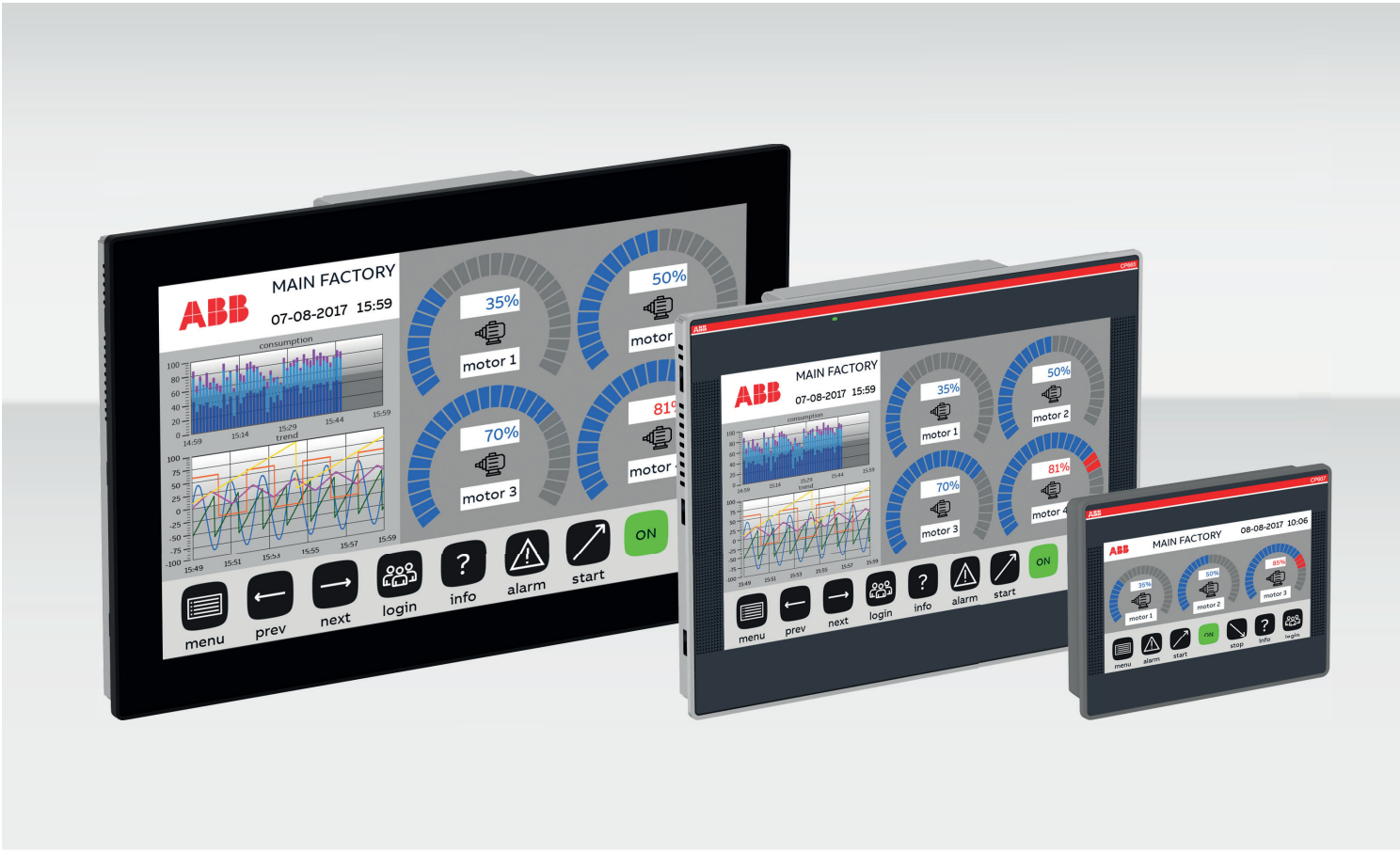
MANUEL D'UTILISATION

PB610 Panel Builder 600

Logiciel de programmation pour afficheurs CP600



Contenu

1 Premiers pas ... 1

Suppositions ... 2

Nouveautés ... 2

Installer l'application ... 4

2 Runtime ... 7

Paramètres de base du périphérique IHM ... 8

Options de menu contextuel ... 8

Service SNTP intégré ... 11

3 Runtime sur PC ... 13

Problèmes d'installation typiques ... 15

4 Mon premier projet ... 19

L'espace de travail ... 20

Protocoles de communication ... 21

Concevoir une page ... 22

La galerie des widgets ... 23

Widget champ de données ... 25

Ajouter des variables ... 28

Associer un widget à des variables ... 31

Pages de dialogues ... 33

Exporter des variables ... 34

Importer des variables ... 35

Rechercher et renommer une variable ... 37

Rechercher et remplacer une variable ... 39

5 Concepts de programmation ... 43

Types de données ... 44

Paramètres « Associer à » ... 44

Formule ... 50

Événements ... 54

Positionnement des widgets ... 57

Gestion widgets chevauchants ... 58

Grouper des widgets ... 59

Changer les propriétés de plusieurs widgets ... 66

Changer la propriété couleur de remplissage en fonction des valeurs de variable ... 67

6 Propriétés de projet ... 69

Runtime ... 70

Module d'exécution (plug-in) ... 75

Projet ... 76

Web ... 80

Événements ... 81

7 Le simulateur IHM ... 83

Méthodes de simulation des données ... 84

Paramètres du simulateur ... 84

Lancer et arrêter le simulateur ... 85

8 Transférer le projet au périphérique IHM ... 87

Télécharger vers le périphérique IHM ... 88

Mettre à jour un paquet ... 91

Le chargeur de runtime ... 93

Téléverser des projets ... 94

9 Variables système (Associer à) ... 95

Variables d'alarmes ... 97

Variables du vibreur sonore ... 97

Variables de communication ... 98

Variables bases de données ... 98

Variables de passage à l'heure d'été (DLS) ... 99

Variables de périphérique ... 100

Variables informations sur les vidages ... 102

Variables client FTP ... 103

Variables réseau ... 103

Variables d'impression ... 104

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Variables client à distance 105
Variables de version 106
Variables d'écran 106
Variables carte SD 106
Variables serveur 107
Variables d'heure 107
Variables d'écran tactile 108
Variables lecteur USB 109
Variables de gestion des utilisateurs 109
10 Variables système (protocole) 111
Paramètres de l'éditeur de protocole 112
Variables par défaut 112
Variables de services 128
Variables de mémoire rémanente 129
Importation de variables 135
11 Actions 139
Actions d'alarmes 140
Actions de base de données 141
Actions d'événements 144
Actions multilangues 144
Actions clavier 145
Actions lecteurs médias 147
Actions FTP 147
Actions sur pages 150
Actions d'impression 154
Actions sur une recette 156
Actions client à distance 160
Actions système 161
Actions de variables 172
Actions de tendance 175
Actions éditeur de texte 179
Actions de gestion des utilisateurs 179
Actions widget 182
12 HMI Client 185
Application client sur ordinateur 186
Application client sur IHM 187
Paramètres et options de fuseau horaire 188
13 Utiliser le serveur FTP intégré 193
Paramètres FTP 193
14 Utiliser VNC pour accès à distance 195
Démarrage du serveur VNC sur les périphériques WinCE 196
Démarrage du serveur VNC sur les périphériques Linux 197
Démarrer la visionneuse VNC 198
15 Alarmes 199
Éditeur d'alarmes 200
Confirmer les alarmes à distance 203
Machine à états pour alarme 204
Configurer des événements 204
Widget alarmes actives 207
Widget historique des alarmes 212
Gestion des alarmes au runtime 212
Activer/désactiver les alarmes au runtime 213
Afficher les données d'alarme actives 213
Exporter la mémoire tampon des alarmes vers fichiers .csv 214
Exporter la configuration des alarmes 215
16 Recettes 219
Gestion des recettes 219
Configuration d'un widget de recette 222
Statut de recette 223
Téléverser/télécharger une recette 224
Sauvegarder et restaurer des données de recettes 225
17 Tendances 227
Enregistrement des données 228

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Exporter les données de mémoire de tendance 231
Widgets tendance 232
Tendances historiques 233
Propriétés du widget tendance 235
Gestes de widget de tendance 236
Valeurs hors de portée ou non valides 237
Afficher les valeurs de tendance 238
Widget diagramme de dispersion 239
Widget tendance de tableau 240
18 Transfert de données 245
Éditeur de transfert de données 246
Exporter données vers fichiers .csv 248
Limites et recommandations du transfert de données 248
19 Gestion de nœud hors ligne 251
Processus de gestion de nœud hors ligne 252
Processus de gestion manuelle de nœud hors ligne 252
Configuration manuelle hors ligne 252
Détection automatique de nœuds hors ligne 253
20 Multilangue 255
L'éditeur multilangue 257
Changer de langue 258
Widgets multilangues 258
Exporter/importer des chaînes multilangues 260
Modifier la langue au runtime 262
Limites de la prise en charge UNICODE 262
21 Planificateur 265
Créer un ordonnancement 266
Ordonnancement Haute Résolution 266
Ordonnancement des tâches répétitives 266
Configurer un emplacement pour les ordonnancements 268
Configurer le widget planificateur 269
Ordonnancement des événements au runtime 270
22 Conformité à 21 CFR partie 11 273
Certificat x.509 279
23 Gestion et mots de passe utilisateur 283
Activer/désactiver la gestion de sécurité 285
Configurer groupes et autorisations 285
Modifier des permissions d'accès 286
Assigner des permissions par widget à partir du mode Page 291
Configurer utilisateurs 292
Utilisateur par défaut 294
Gestion des utilisateurs au runtime 294
Forcer la connexion à distance 294
24 Journaux d'audit 297
Activer/désactiver le journal d'audit 298
Signature électronique 299
Widget audit de tableau 302
Exporter le journal d'audit au format .csv 303
Configurer les événements d'audit 304
25 Rapports 307
Ajouter un rapport 308
Configurer des rapports de texte 308
Configurer les rapports graphiques 309
Événements déclencheurs d'impression 310
Imprimante par défaut 311
26 Écran de veille 315
27 Sauvegarder/restaurer le runtime et le projet 317
28 Claviers 319
Créer et utiliser des claviers personnalisés 321
Supprimer ou renommer des claviers personnalisés 323

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Type de clavier 323
Position de clavier 324
29 Claviers externes 325
Chercher et filtrer 327
Touches affichées 327
Retirer des associations d'actions 327
Disposition du clavier 328
Activer/désactiver clavier 328
Associer des actions à des touches 328
30 Référence croisée de variables 331
Mettre à jour les données dans le volet Référence croisée des variables 334
31 Adressage indexé 335
Générer un ensemble d'adressage indexé 336
Utilisation d'un ensemble de variables indexées dans des pages 339
32 Stocker les données dans des bases de données externes 341
Installer SQL4Automation 343
Configurer le module SQL4Automation 343
Configurer le projet IHM 345
Transférer les données avec JavaScript 346
Tableaux de base de données 347
Tableaux personnalisés 348
Limites de connexion 348
33 Serveur à architecture unifiée OPC 351
Fonctionnalités 351
Réseau 351
Authentification 351
Utilisation de certificats x.509 352
Utilisation de certificats auto-signés 356
Utilisation de certificats externes 359
Carte des alarmes 360
34 Widgets spéciaux 361
Widget BACnet 362
Widget de navigateur 362
Widget Canevas 363
Widget liste déroulante 367
Widget compteur de consommation 371
Widgets de liste de contrôle 373
Widgets Date & Heure 374
Widget de zone de mouvement 375
Widgets Caméra IP 381
Widget blocage de fonction Javascript 384
Widgets de lecture de médias 386
Widget d'image multi-états 389
Widget multi-couche d'image multi-états 389
Widget adaptateurs réseau 391
Widget flux RSS 392
Faire défiler le widget flux RSS 392
Widget tableau 393
Widget ÉditeurTexte 407
Widget de variables 409
35 Widgets personnalisés 411
Créer un widget personnalisé 412
Ajouter des propriétés à un widget personnalisé 415
Utilisation de variables structurées 417
JavaScript dans les widgets personnalisés 419
Galerie de l'utilisateur 422
36 Envoyer un message par courriel 425
Configurer le serveur e-mail 426
Configurer les e-mails 426
37 JavaScript 429
Éditeur JavaScript 431
Exécution des fonctions JavaScript 431
Événements 433

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Widget événements 434
Événements de page 436
Événements système 437
Objets 439
Widget objets de classe 439
Propriétés du widget 440
Méthodes du widget 443
Objet de page 444
Propriétés de l'objet de page 444
Méthodes associées à l'objet de page 445
Objet de groupe 448
Méthodes d'objet de groupe 448
Objet du projet 449
Propriétés d'objet du projet 449
Méthodes d'objet du projet 449
Widgets d'objet du projet 458
Imprimer l'objet des rapports 459
Objet de statut 462
Méthodes pour objet de statut 463
Mots-clés 464
Fonctions générales 464
Prendre en main la lecture/l'écriture de fichiers 465
Se connecter à partir de JavaScript 468
Limites lors du travail avec des widgets dans JavaScript 470
Déboguer JavaScript 470
38 Gestes de prise en main 475
39 Paramètres du système 477
Périphériques Linux 478
Périphériques WinCE 499
40 Accès Web 509
Plateformes et navigateurs pris en charge 510
Créer une page pour accès Web 510
Pages d'accueil de plateforme spécifiques 512
Tester le projet Web 512
Télécharger le projet Web 513
Problèmes de connectivité Web 514
Fonctionnalités web prises en charge 516
Résolution des problèmes et FAQ 520
41 Mettre à jour les composants système des périphériques IHM 523
Mise à jour des composants système à partir de l'application 524
Paramètres 525
42 Protéger l'accès aux périphériques IHM 527
Changer de mot de passe sur le périphérique IHM 528
Ports et pare-feux 528
43 Conseils et astuces pour améliorer la performance 531
Optimisation statique 532
FAQ sur l'optimisation statique 535
Mise en cache de page 536
Image DB 536
Pré-cachage 536
FAQ sur le pré-cachage 537
44 Précisions fonctionnelles et compatibilité 539
Tableau de fonctions et limites 540
Capacités des périphériques IHM 541
Compatibilité 542
Projets de conversion entre les différents périphériques IHM 542
45 Protocoles de communication 545
Ethernet ABB CoDeSys 546
ABB CODESYS Serial 556
ABB IRC5 563
ABB Mint Controller HCP 568

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

ABB Modbus RTU 577
ABB Modbus TCP 584
ABB Pluto 590
BACnet 596
CODESYS V2 ETH 625
CODESYS V3 ETH 638
Ethernet/IP CIP 651
Modbus RTU 677
Modbus RTU Server 693
TCP Modbus 708
Serveur Modbus TCP 725
OPC UA Client 737
Simatic S7 ETH 751
Variables système 791
Variables 793

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

1 Premiers pas

PB610 Panel Builder 600 est un logiciel de création de pages graphiques dans une IHM. PB610 Panel Builder 600 comporte une interface glisser-déposer qui facilite la création de pages complexes. De nombreuses fonctionnalités présentes dans les applications courantes de Windows sont également disponibles dans PB610 Panel Builder 600.

Ce document est divisé en chapitres décrivant les principales fonctions de PB610 Panel Builder 600 et explique comment les utiliser. Chaque chapitre est présenté de manière autonome, vous permettant de sauter d'un chapitre à l'autre, en fonction de la tâche à accomplir.

Suppositions **2**

Nouveautés **2**

Installer l’application **4**

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Suppositions

Nous supposons que les lecteurs possèdent une compréhension de base des ordinateurs, Microsoft Windows, et l'environnement de réseau spécifique où l'application sera exécutée.

Nouveautés

Nouveautés de la v2.8

- Widgets personnalisés et héritage des propriétés des widgets
- Couverture améliorée des fonctions de PB4Web, y compris la gestion des utilisateurs, les widgets personnalisés, le widget canevas. Amélioration de la conception réactive basée sur la disposition en grille.
- Prise en charge de la conformité des applications avec 21 CFR partie 11, y compris les signatures électroniques et la sécurité relative aux certificats
- Calcul d'expressions avec des valeurs de variables
- Nouveaux widgets d'alarme et de tendance basés sur des widget de tableaux
- Widget journal d'audit
- Prise en charge améliorée du tactile multipoint et des gestes
- Trouver une variable avec fonctions renommer et remplacer
- Supprimer les variables inutilisées du projet
- Horodatages des sources dans l'acquisition des données de variables pour les sources OPC UA
- Lecteur vidéo pour les appareils IHM basés sur Linux
- Vidage de tendances une fois pleines
- Protocole Direct Serial pour la création de protocoles de communication personnalisés
- Protocole Direct Socket pour la création de protocoles de communication personnalisés
- Limites de projet étendues pour les périphériques IHM haut de gamme
- Amélioration des performances au runtime avec un temps de démarrage plus rapide
- Logiciel distant xHMI Clientx pour les périphériques IHM sous Linux
- Prise en charge totale de BACnet avec alarmes et tendances BACnet

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Nouveautés de la v2.6

- Disposition des grilles
- Widget tableau
- Widget Canevas
- Widget de blocage de fonction Javascript
- Variables structurées prises en charge pour les widgets personnalisés
- Option « Tout afficher » pour widget Historiques des alarmes (Activation du widget Historique des alarmes pour être en mesure d'afficher toutes les alarmes)
- Support « Joindre à ... » pour les propriétés dans la configuration du serveur de courriels
- Support pour l'écriture des périphériques de réseau pour CP600-eCo (BSP 1.0.25 ou plus grand si nécessaire)
- Sauvegarde / restauration pour CP600-eCo
- Prise en charge de PB4Web des tendances
- Prise en charge de PB4Web des recettes
- Support de protocole BACnet pour le calendrier / Planificateur
- Révision des limites cibles
- Extension du serveur OPC-UA pour prendre en charge les alarmes et les tendances
- Client FTP
- Liste déroulante activée pour travailler en plein écran avec prise en charge des images
- Support ajouté pour éditer et modifier manuellement les liaisons de données
- Amélioration du transfert de données pour prendre ne charge les alias (variables indexées)
- Au besoin, « Télécharger vers cible » invoque la « mise à jour du runtime » lors du téléchargement du projet.
- Une nouvelle variable de système pour vérifier le statut des fichiers vidés

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Nouveautés de la v2.4

- Unifie PB600 Basic et la version Standard avec les limites IHMs basées sur la licence (Basic/Standard)
- Le projet limite le contrôle lors de la conversion d'un projet (non seulement sur le téléchargement comme avant)
- Désactive les protocoles non disponibles basés sur le périphérique sélectionné
- Désactive la sélection de la carte SD dans les périphériques où ce n’est pas disponible
- Widget IP pour définir les paramètres du réseau.
- Introduction de "WebPageRequest" pour contrôler la page affichée par un API.
- Nouvelle propriété de projet pour définir la couleur d'arrière-plan du runtime
- Support serveur à architecture unifiée OPC
- Support SQL4Automation
- Support JS sur PB4Web
- Support multilangue sur PB4Web
- Les chaînes d'état d'alarme peuvent être personnalisées et traduites
- Support de visionneuse PDF pour les produits CP600-eCo
- Support pour JS personnalisé intégré dans la galerie Widget
- Support des gestes
- Introduction d'un moteur d'activation pour les données envoyées/reçues. Maintenant, PB4Web peut actualiser les données et le widget jusqu'à 10 fois par seconde
- Redéfinition de la gestion de cible pour mettre à jour plusieurs périphériques IHM
- Nouvelle option pour activer les alarmes émises par les API
- Nouvelle option pour exécuter une action macro utilisateur en cas d'alarme
- Possibilité de grouper et de filtrer les alarmes

Nouveautés en comparant avec la version v2.0.0

- Prend en charge de Windows 10 32-/64-bit
- Galerie de l’utilisateur pour les widgets Personnalisés
- Nombre maximum de variables pour PB610 PC Runtime : 10 000
- Aide/manuel en ligne en allemand

Installer l’application

L'installation de PB610 Panel Builder 600contient :

- PB610 Panel Builder 600 : une application pour concevoir des projets IHM personnalisés de façon conviviale, assortie d'une palette d'objets dans sa bibliothèque intégrée, la Galerie des widgets.
- HMI Client : une application légère qui peut être utilisée sur les ordinateurs Windows pour visualiser et gérer à distance un projet exécuté sur un périphérique IHM.
- HMI Runtime : une application autonome qui fonctionne sur les périphériques IHM. Le runtime IHM est installé via PB610 Panel Builder 600.
- PB610 PC Runtime : une application autonome qui fonctionne sur les plateformes Win32 (sur les ordinateurs plutôt que sur les périphériques IHM).

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Exigences de système PB610 Panel Builder 600

PB610 Panel Builder 600 a les exigences de système suivantes :

Système d'exploitation	Windows XP Professional Windows XP Embedded Windows Embedded Standard (WES 2009) Windows Server 2003 Windows Vista Business/Ultimate Windows 7 Professional Windows Embedded Standard 7 Windows 8 Windows 10
Stockage	500 MB Minimum
RAM	512 MB
Autres	Une connexion Ethernet

Installer des versions multiples de PB610 Panel Builder 600

Vous pouvez installer différentes instances de PB610 Panel Builder 600 sur un même ordinateur. Chaque installation a ses propres paramètres et peut être désinstallée individuellement.

Trois scénarios d'installation sont possibles :

Scénario d'installation	Résultats
Première installation de PB610 Panel Builder 600 dans le système	Le logiciel est installé dans le dossier de destination spécifié
Système avec seulement une instance de PB610 Panel Builder 600 déjà installée	La version actuelle peut être remplacée ou maintenue.
Système avec de multiples instances de PB610 Panel Builder 600 déjà installées	La dernière version installée peut être remplacée ou maintenue.

Si vous essayez d'installer une deuxième instance d'une version déjà installée de PB610 Panel Builder 600, un message d'avertissement apparaîtra.

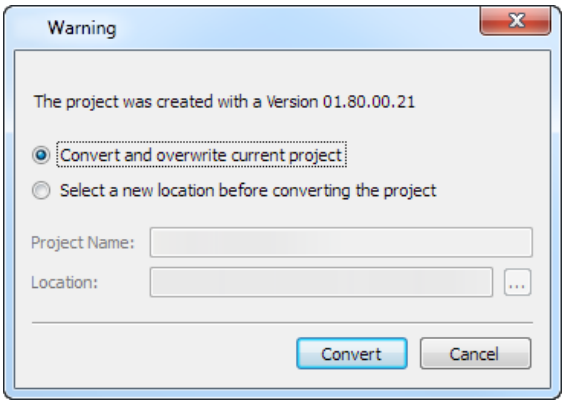
Plusieurs installations PB610 Panel Builder 600 partagent un dossier d'espace de travail commun, chaque sous-dossier comprend le numéro de version, par exemple *C:\Program Files\ABB\Panel Builder 600 Suite 2.5.*. Chaque version installée a son propre ID et peut donc être éliminée individuellement.

Chaque installation est mentionnée séparément dans le menu **démarrage** de Windows.

Ouvrir d'anciens projets

Lorsque vous ouvrez un projet (fichier .jpr) PB610 Panel Builder 600 créé avec une version moins récente du logiciel, PB610 Panel Builder 600 vous demande de convertir le projet dans la version actuelle :

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



Option	Description
Convertir et écraser le projet actuel	Le projet est converti sans version de sauvegarde de la version originale
Sélectionnez un nouvel emplacement avant de convertir le projet	Le projet est copié dans le dossier spécifié, puis converti.

AVERTISSEMENT : n'éditez pas de projets avec une version de PB610 Panel Builder 600 moins récente que celle utilisée pour les créer. Cela pourrait endommager le projet et rendre instable le runtime.

Version multilingue de PB610 Panel Builder 600

PB610 Panel Builder 600 est disponible en plusieurs langues. Toutes les langues sont installées par défaut dans le cadre de PB610 Panel Builder 600.

La langue par défaut est l'anglais. Pour la modifier, allez sur **Aide** > **Changer de langue**.

Rapports de plantage

Une boîte de dialogue de rapport de plantage apparaît toutes les fois que PB610 Panel Builder 600 gèle ou plante.

Important : sauvegardez toujours les fichiers de rapport de crash, car ils peuvent contenir des informations utiles pour l'assistance technique.

Remarque : les rapports de plantage ne sont pas disponibles sur Windows XP.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

2 Runtime

HMI Runtime est conçu pour prendre en charge diverses plateformes et divers systèmes d'exploitation.

Paramètres de base du périphérique IHM 8
Options de menu contextuel 8
Service SNTP intégré 11

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Paramètres de base du périphérique IHM

Les périphériques IHM sont livrés depuis l'usine sans runtime. Si aucun runtime n'est installé sur le périphérique, voir "Le chargeur de runtime" Page 93 pour plus de détails.

Modes de runtime

Le HMI Runtime est composé de deux unités logiques :

- **Serveur** : exécute les protocoles de communication, collecte les données, gère les alarmes, génère des échantillonnages de mémoire tampon des tendances.
- **Client** : affiche les données collectées par le serveur.

L'unité serveur est responsable de la gestion des services IHM tels que les protocoles de communication, de l'acquisition de données, de la génération d'échantillonnages de mémoire des tendances, etc.

L'unité client est la partie responsable du processus de visualisation : utilisez les données collectées par le serveur pour les afficher à l'écran sous forme d'informations graphiques.

L'unité serveur fonctionne selon deux modes opératoires :

- **Mode de configuration** : le serveur est inactif (par exemple quand aucun projet n'est chargé sur le périphérique ou quand il manque des fichiers système).
- **Mode opérationnel** : le serveur fonctionne selon les paramètres définis par les fichiers système et par le projet d'application chargé.

Remarque : les données sur l'unité client peuvent s'afficher même en l'absence d'activité sur le serveur.

Options de menu contextuel

Sur le périphérique IHM, appuyez sur un espace vide de l'écran et maintenez enfoncé pendant quelques secondes pour afficher le menu contextuel.

Zoom avant / arrière

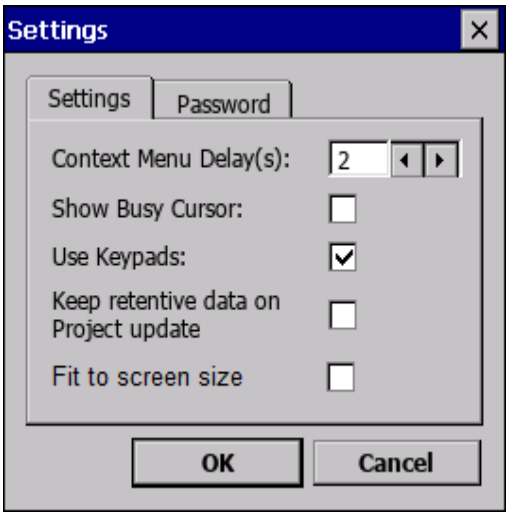
Sélectionner la taille de la vue au runtime

Mode panoramique

Active/désactive le mode panoramique après un zoom avant

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

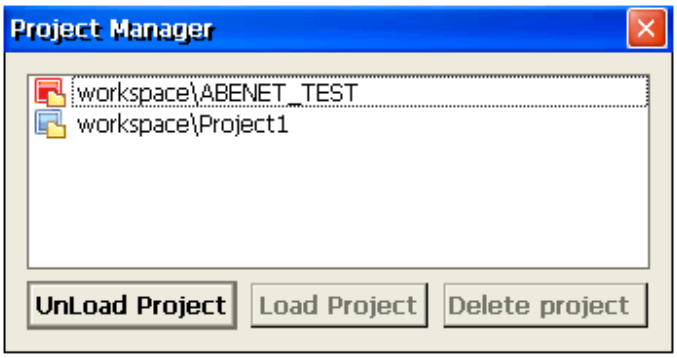
Paramètres



Paramètres principaux	Description
Retard(s) menu contextuel	Retard de l'activation du menu contextuel. Plage : 1 à 60 secondes.
Afficher curseur occupé	Affiche un sablier lorsque le système est occupé
Utiliser des claviers	Affiche les claviers lorsque l'utilisateur touche un champ d'entrée des données. Défini sur **désactiver** lorsqu'un clavier USB externe est connecté au périphérique.
Conserver les données rémanentes à la mise à jour du projet	Conserve le contenu des données non volatiles en chargeant ou téléchargeant le projet.
Ajuster aux dimensions de l'écran	Adapte la vue aux dimensions de l'écran
Mot de passe	Définit les opérations protégées par mot de passe parmi les suivantes : • Télécharger projet/Runtime • Charger projet • Gestion de carte (mise à jour BSP) Voir "Protéger l'accès aux périphériques IHM" Page 527 pour plus de détails.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Gestionnaire de projet



Cet outil vous permet de :

- décharger le projet en cours
- charger un autre projet
- supprimer un projet.

Lorsque vous chargez un nouveau projet, le projet en cours est automatiquement déchargé. Vous devez décharger un projet avant de pouvoir le supprimer.

Mise à jour

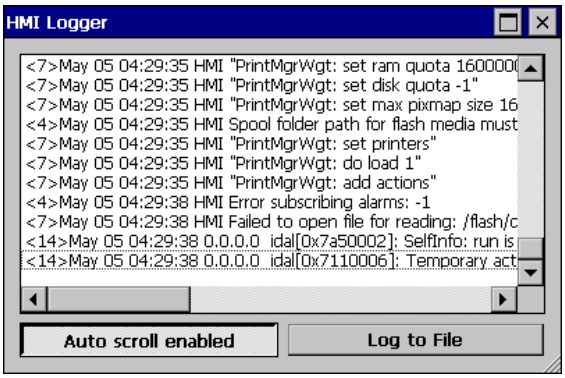
Cette fonction charge des paquets de mise à jour à partir d'un lecteur USB externe. Voir "Mettre à jour un paquet" Page 91pour les détails.

Sauvegarde

Vous pouvez créer une copie de sauvegarde du runtime et du projet.

Enregistrement

Cette fonction affiche un enregistrement des opérations système.



Cliquez sur **Enregistrer dans un fichier** pour sauvegarder les données : un fichier logger.txt file est sauvegardé dans le dossier *...\var\log*.

Ce fichier peut être récupéré avec un client FTP et transféré au support technique.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Remarque : une fois activé, l'enregistrement est maintenu après les cycles d'alimentation et doit être désactivé manuellement.

Afficher le journal au démarrage

Cette fonction active l'enregistreur automatique au démarrage. Si l'option **Enregistrer dans un fichier** a été activée, les fichiers d'enregistrement sont sauvegardés à partir du démarrage.

Déconnexion

Déconnecte l'utilisateur actuel.

Affiche les paramètres système

Autoriser les paramètres IHM et la gestion des composants du système. Voir "Paramètres du système" Page 477 pour les détails.

Outils du développeur

Fonctions utilitaires pour déboguer au runtime. Elles sont visibles uniquement si elles sont activées dans les Propriétés du projet (voir "Outils du développeur" Page 72 pour les détails)

À propos

Cette fonction affiche les informations à propos de la version du runtime.

AVERTISSEMENT : l'action Menu contextuel ne fonctionne pas lorsqu'elle est exécutée depuis une page de dialogue.

Service SNTP intégré

Le périphérique IHM inclut un SNTP intégré qui synchronise le volet de l'horloge en temps réel interne lorsque le serveur prédéfini est disponible. Le système recherche le serveur SNTP à la mise sous tension, ou une fois par semaine si le périphérique IHM n'est pas éteint.

Utilisez un périphérique IHM "Paramètres du système" Page 477pour configurer le service.

Disponibilité : BSP v1.76 ARM/2.79 MIPS ou version ultérieure

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

3 Runtime sur PC

PB610 PC Runtime pour Windows est une plateforme IHM qui combine des fonctionnalités IHM avancées et des graphiques vectoriels avec des technologies web puissantes. Vous pouvez choisir cette plateforme pour superviser et contrôler votre équipement à l'aide de variables, d'alarmes, de planificateurs, de recettes, de tendances, de logique Javascript et d'événements.

PB610 PC Runtime assure une connectivité avec les protocoles d'automatisation industriels et des bâtiments via Ethernet et des interfaces série.

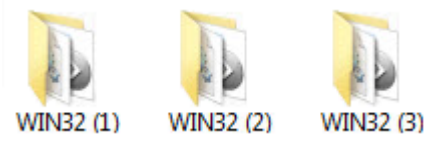
Exigences de système PB610 PC Runtime

PB610 PC Runtime a les exigences minimales de système suivantes :

Système d'exploitation	Windows XP Professional Windows XP Embedded Windows Embedded Standard (WES 2009) Windows Server 2003 Windows Vista Business/Ultimate Windows 7 Professional Windows Embedded Standard 7 Windows 8 Windows 10
Stockage	256 MB Min
RAM	512 MB
CPU	min. 300 MHz Pentium III ou processeurs similaires avec 500 MHz.
Graphique	min. SVGA
Autres	Une connexion Ethernet

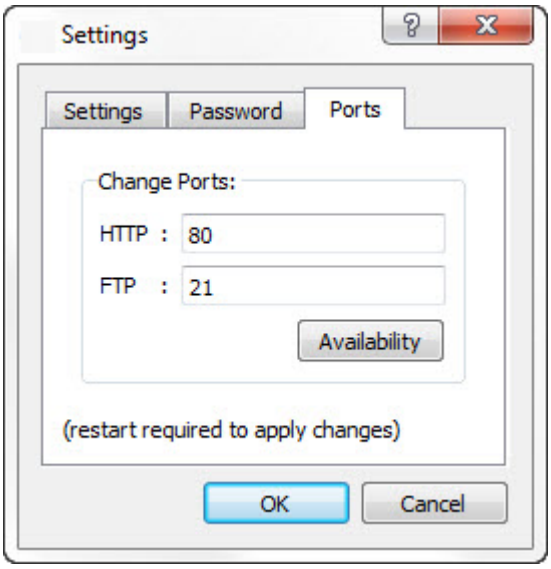
Multiples instances de PB610 PC Runtime

PB610 PC Runtime peut être exécuté en plusieurs instances simultanées. Copiez le dossier d'installation sur un emplacement modifiable et double-cliquez sur l'application IHM dans chaque dossier pour la démarrer.



Le port utilisé par PB610 PC Runtime peut être modifié à partir de la boîte de dialogue des **Paramètres**. Redémarrez l'application pour appliquer la modification du port.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



Limites

Les fonctionnalités suivantes ne sont pas supportées dans PB610 PC Runtime :

Fonction	Fonctionnalité NON supportée
Gérer cible	Section carte
Mode système/ Mode utilisateur	Tap sequence et menu tournant
Lecteurs VNC/PDF	Logiciels informatiques non standards
Sauvegarder/Restaurer	Fonctions sauvegarder et restaurer. Des logiciels informatiques standards peuvent être utilisés à cette fin.
Protocoles	Protocoles série demandant un matériel spécial.

Voir "Précisions fonctionnelles et compatibilité" Page 539 pour plus de détails.

Mode plein écran

PB610 PC Runtime peut démarrer en mode plein écran ou dans une fenêtre.

Pour passer au mode plein écran :

1. Cliquez à l'aide du bouton droit sur la fenêtre principale de PB610 PC Runtime pour afficher le menu contextuel.
2. Sélectionnez **Plein écran**.

Le dossier d'espace de travail

Lorsque vous utilisez PB610 PC Runtime, les fichiers de projet sont stockés dans un dossier d'espace de travail dans :

%appdata%\ABB\[build number]\server\workspace

où [build number] est un dossier nommé en tant que numéro de fabrication (par exemple 01.90.00.608).

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Problèmes d’installation typiques

Important : s’assurer que les ports 80/HTTP et 21/FTP ne sont pas bloqués par le pare-feu.

Si un port est en cours d'utilisation et qu'un conflit est détecté, une boîte de dialogue est affichée pour permettre à l'utilisateur de modifier les ports par défaut.

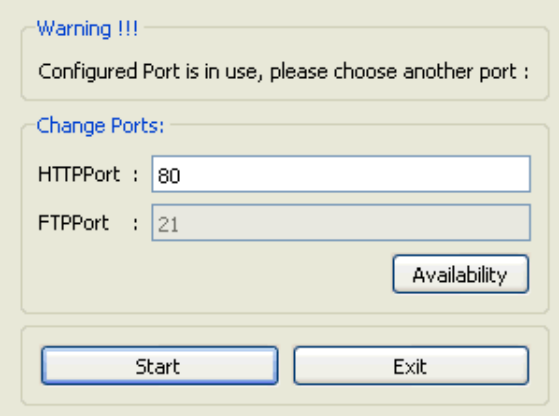
Voir "Protéger l’accès aux périphériques IHM" Page 527 pour les détails.

Dans certaines conditions, PB610 PC Runtime ne peut pas détecter tous les services fonctionnant dans les ports comme 80/HTTP et 21/FTP, ce qui force la fermeture automatique de PB610 PC Runtime. Cela se produit, par exemple, lorsque IIS ou MS SQL Server ou d'autres services Windows s'exécutent sur ces ports. Dans ces cas, désactivez les services Windows.

Si le téléchargement du projet vers PB610 PC Runtime échoue, essayez une des procédures suivantes.

Problèmes avec les numéros de port

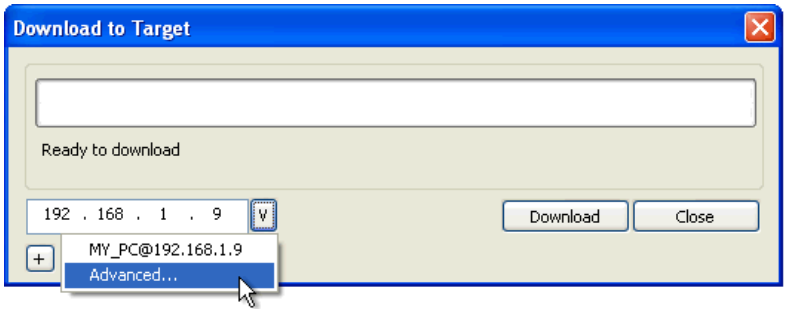
PB610 PC Runtime utilise les ports 80 et 21 par défaut. Si au moins un des ports est occupé, un message d'avertissement s'affiche :



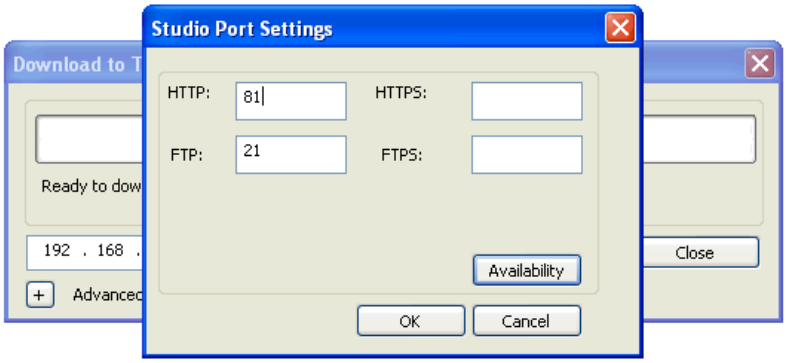
Assurez-vous que lorsque vous modifiez ce port, vous modifiez également le port utilisé pour le téléchargement sur le périphérique IHM dans PB610 Panel Builder 600.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

1. Depuis le **Téléchargement jusqu’à la boîte de dialogue,** sélectionner **Avancé**.



2. Modifiez le numéro de port pour correspondre à celui défini sur PB610 PC Runtime.



3. Cliquez sur **OK** pour confirmer : vous pouvez maintenant télécharger votre projet sur PB610 PC Runtime.

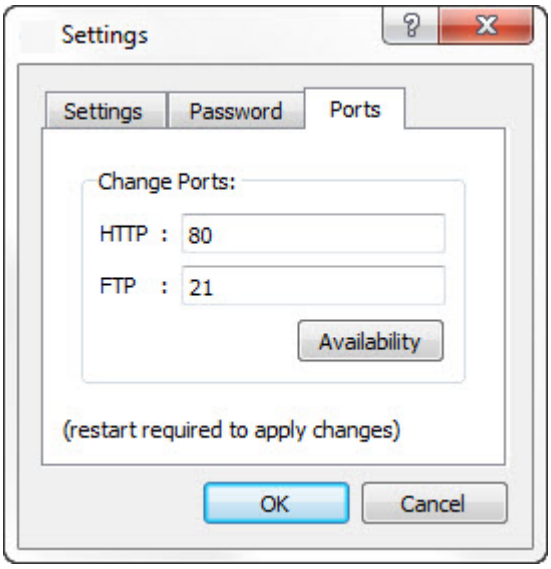
Restauration des informations de port

Si des informations sur les modifications apportées aux ports d'écoute de PB610 PC Runtime ont été perdues, le message d'erreur suivant est renvoyé :

Impossible d'établir une communication avec le Runtime. Veuillez vérifier les paramètres de connexion et si le Runtime est en cours sur le périphérique IHM.

Le port utilisé par PB610 PC Runtime peut être modifié à partir de la boîte de dialogue des **Paramètres**. Redémarrez l'application pour appliquer la modification du port.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



Contourner le pare-Feu ou les blocs antivirus

Si PB610 Panel Builder 600 est en cours d'exécution sur la même machine que le PB610 PC Runtime, votre pare-feu ou antivirus peut bloquer la connexion depuis PB610 Panel Builder 600 jusqu'à PB610 PC Runtime.

1. Depuis la boîte de dialogue **Téléchargement vers la cible**, tapez manuellement l'adresse IP de l'ordinateur hôte local 127.0.0.1.
2. Cliquez sur **Télécharger**.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

4 Mon premier projet

Cette section décrit comment créer un projet PB610 Panel Builder 600simple.

L’espace de travail **20**

Protocoles de communication **21**

Concevoir une page **22**

La galerie des widgets **23**

Widget champ de données **25**

Ajouter des variables **28**

Associer un widget à des variables **31**

Pages de dialogues **33**

Exporter des variables **34**

Importer des variables **35**

Rechercher et renommer une variable **37**

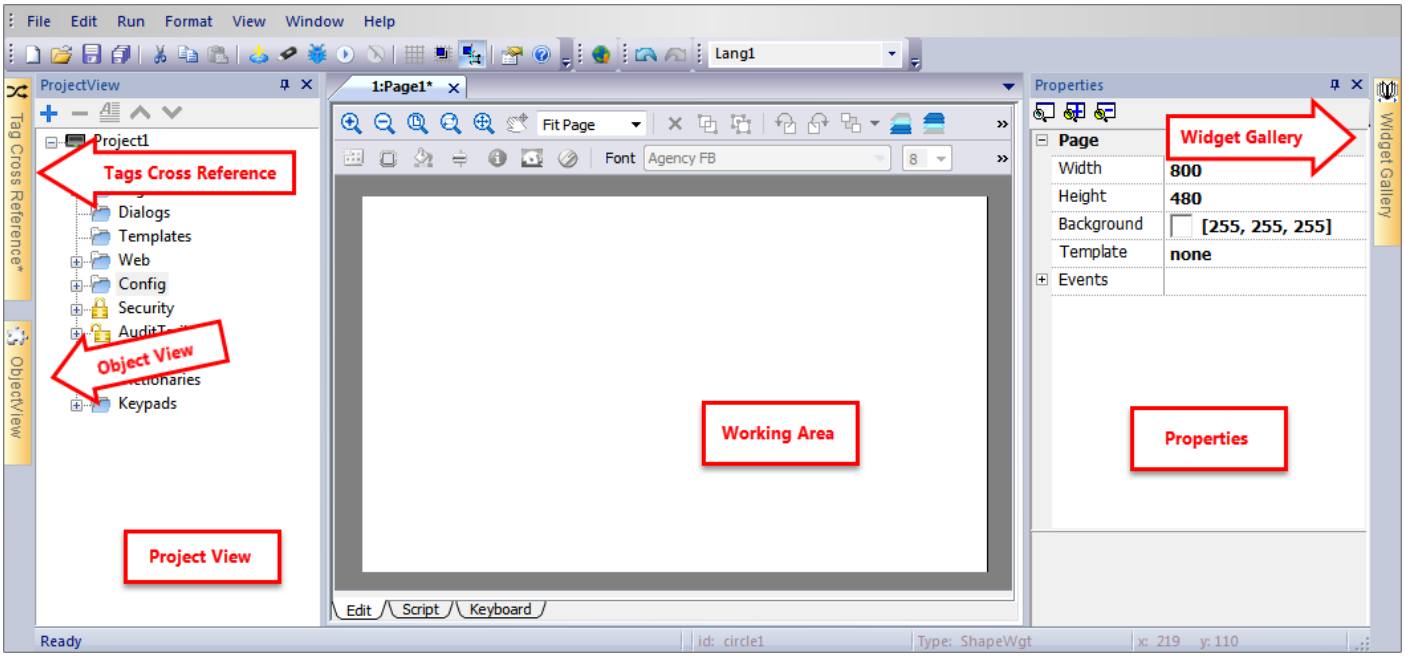
Rechercher et remplacer une variable **39**

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

L’espace de travail

Zones de l’espace de travail

L'espace de travail de PB610 Panel Builder 600 est divisé dans les principaux domaines suivants :



Domaine	Description
Vue du projet	Les éléments du projet dans un arbre hiérarchique du projet.
Vue de l’objet	Arborescence des widgets organisés par page.
Zone de travail	Espace où les pages sont éditées. Les onglets en haut de la zone montrent toutes les pages ouvertes.
Propriétés	Propriétés de l'objet sélectionné.
Galerie des widgets	Bibliothèque des objets graphiques et des symboles.
Référence croisée de variables	Liste des emplacements où une variable donnée est référencée.

Remarque : la mise en page de l'espace de travail peut être modifiée à tout moment, les modifications sont enregistrées et maintenues grâce à des séances de travail.

Réinitialisation de la mise en page de l’espace de travail

Pour restaurer la mise en page par défaut, utilisez la fonction **Fichier > Réinitialisation et redémarrage**.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Protocoles de communication

*Chemin : **VueProjet*> *Configuration* > *Protocoles***

Des pilotes de communication du périphérique sont configurés dans l'**Éditeur de protocole**.Vous pouvez ajouter autant de protocoles que le nombre spécifié dans le Tableau de fonctions. Les Variables et les Variables système ne comptent pas comme des protocoles.

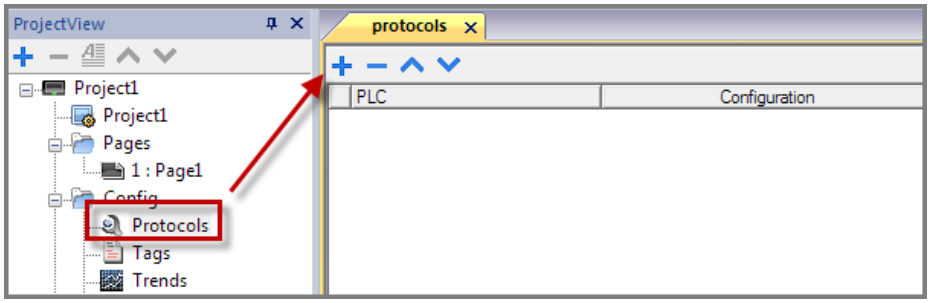
Voir "Protocoles de communication" Page 545 pour plus de détails.

Remarque : vous pouvez exécuter différents protocoles Ethernet sur le même port physique Ethernet, mais vous ne pouvez pas exécuter différents protocoles série en utilisant le même port série. Certains protocoles série prennent en charge l'accès à plusieurs contrôleurs, mais cette option est définie au sein du protocole, qui compte comme un seul protocole.

Ajouter un protocole

Remarque : référez-vous au manuel d'utilisation de CP600 si vous avez besoin d'information sur les câbles.

1. Cliquez sur **+**.

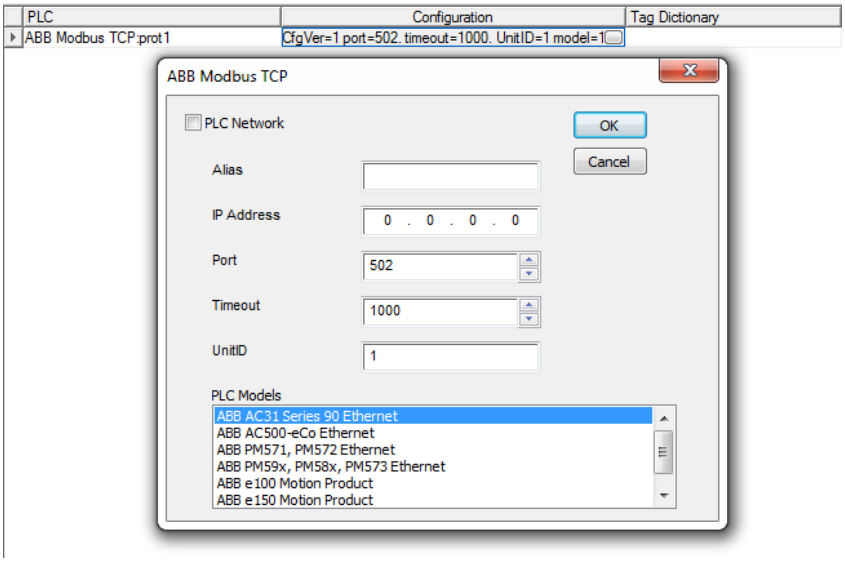


2. Sélectionnez le protocole depuis la liste **API** et entrez les valeurs requises.

Modifier les paramètres du protocole

Pour modifier les paramètres de configuration, cliquez sur le bouton Parcourir dans la colonne **Configuration**.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



Paramètres du protocole

Cliquez sur l'icône **Afficher propriétés avancées** pour afficher tous les paramètres.

Paramètre	Description
Dictionnaires	Variables importées pour le protocole. Voir "Importer des variables" Page 35 pour plus de détails.
Activer Algorithme hors ligne Délai d'attente tentative supplémentaire hors ligne	Voir "Détection automatique de nœuds hors ligne" Page 253 pour plus de détails.
Version	Version du protocole disponible dans PB610 Panel Builder 600 pour le périphérique IHM sélectionné.

Concevoir une page

*Chemin : **VueProjet**> **Pages***

Lorsqu'un projet est créé, la première page est automatiquement ajoutée et affichée dans l'**Éditeur de page**.

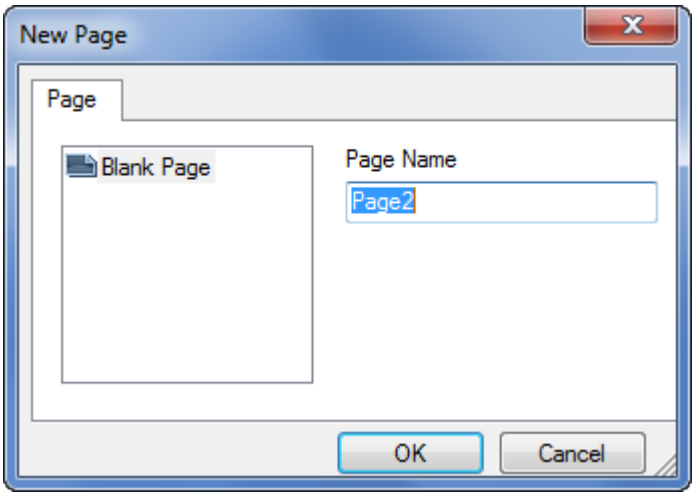
Ajouter des objets sur une page

Glissez et déposez des objets depuis la **Galerie des widgets** vers la page.

Ajouter une page

1. Faites un clic droit sur le nœud **Pages** dans l'arborescence du projet et sélectionnez **Insérer nouvelle page**.
2. Tapez un nom pour la nouvelle page.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



Importer une page

Lorsque vous importez une page, PB610 Panel Builder 600 importe la mise en page et les widgets de la page sans importer les actions et les liaisons de données associées aux widgets. Vous pouvez choisir entre deux comportements différents :

- importer uniquement les pages et les widgets : toutes les actions et les liaisons de données doivent être définies
- importer les pages avec les références aux actions et aux liaisons de données : les variables utilisées doivent être présentes dans le projet pour que ces éléments fonctionnent correctement

Remarque : l'importation de pages peut uniquement être réalisée entre des projets créés avec la même version du logiciel. Sauvegardez l'ancien projet en tant que dernière version, puis réessayez.

1. Faites un clic droit sur le nœud **Pages** dans l'arborescence du projet et sélectionnez **Importer page**.
2. Choisissez la page à importer depuis le projet voulu puis cliquez sur **OK** : un message d'avertissement s'affiche.
3. Cliquez sur **Oui** pour supprimer toutes les liaisons à des données et toutes les actions. Cliquez sur **Non** pour maintenir la référence aux liaisons de données et aux actions.

Groupe de pages

Vous pouvez grouper des pages similaires pour faciliter la maintenance. Grouper des pages n'affecte pas le mode d'apparition des pages au runtime. Pour créer un groupe de pages :

1. Dans **VueProjet**, faites un clic droit sur le nœud **Pages** et sélectionnez **Créer groupe** : un nouveau dossier s'ajoute
2. Pour déplacer une page dans un groupe, faites un clic droit sur une page et sélectionnez **Groupes** > *groupName.*

La galerie des widgets

Chemin : ***Vue> Barres d'outils et Ancrage de fenêtres> Galerie Widget***

Les objets IHM nécessaires à la construction d' une application sont disponibles dans la **Galerie Widget**. La galerie est divisée en plusieurs catégories, chacune contenant une collection de widgets.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Ajouter un widget à une page

1. Sélectionnez le widget depuis la **Galerie Widget**.
2. Glissez et déposez-le sur la page.

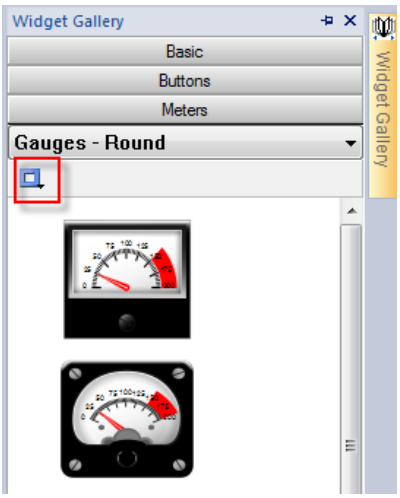
Changer l'apparence d'un widget

Tous les widgets ont des propriétés (volet **Propriétés**) qui peuvent être modifiées, certains widgets sont présentés dans divers styles. Vous pouvez cliquer sur les boutons dans chaque catégorie pour voir les styles disponibles.

Exemple

Pour définir le style du widget pour jauges rondes :

1. Cliquez sur le bouton de style pour afficher les styles disponibles pour le widget.



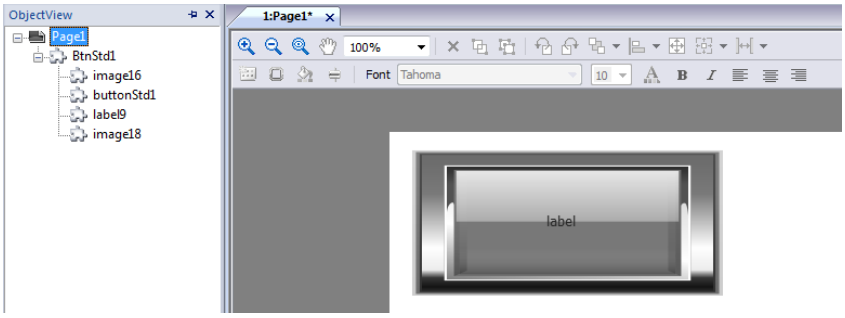
Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

2. Sélectionnez l'un des styles disponibles à partir de la barre d'outils : selon le widget sélectionné, différentes options sont disponibles.

Widgets complexes

Certains widgets sont composés de plusieurs sous-widgets. Par exemple, un bouton est un widget complexe composé d'un widget bouton et d'une étiquette. La structure des widgets peut être vue dans **VueObjet** lorsque le widget est sélectionné.

Vous pouvez sélectionner un sous-widget, tel que l'étiquette dans un bouton, depuis **VueObjet** et le modifier sans dégrouper le widget en entier.

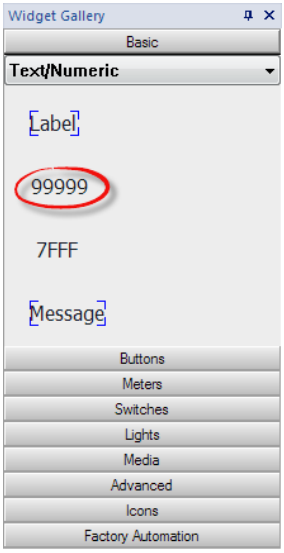


Widget champ de données

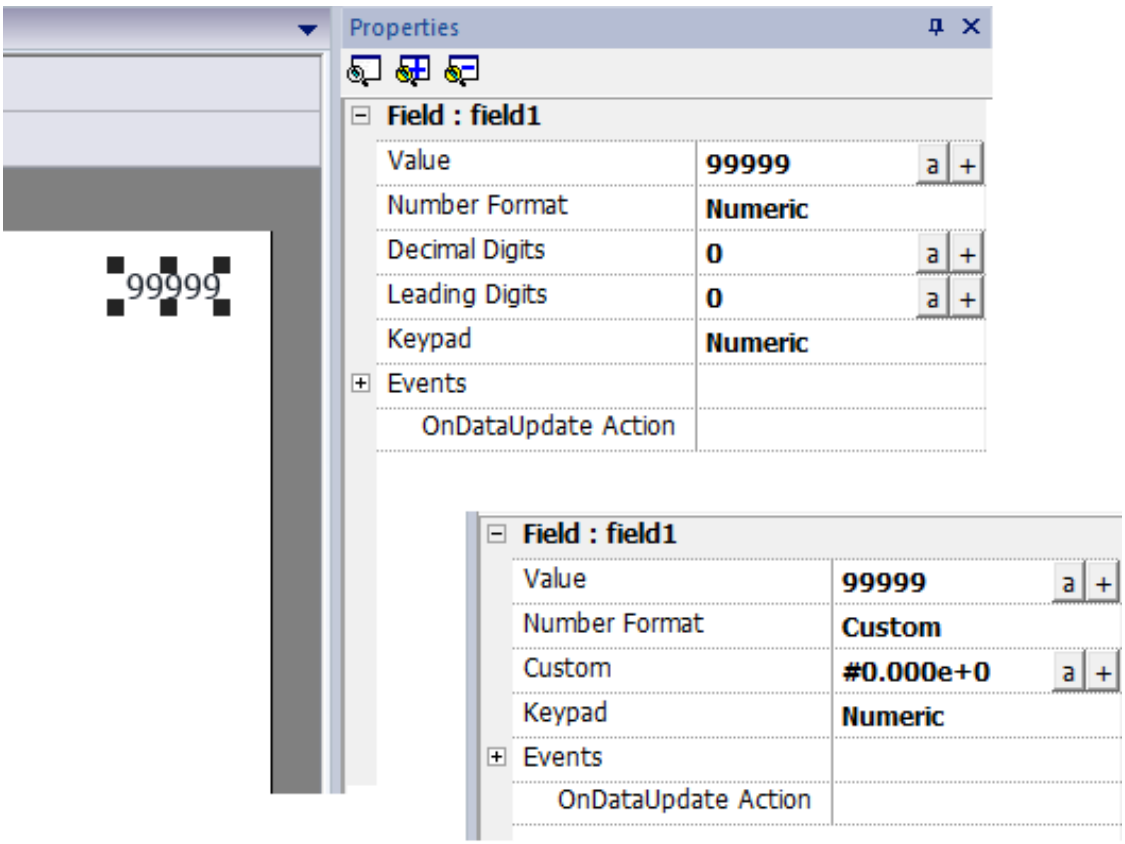
Le widget le plus courant est le widget champ de données qui donne la possibilité d'afficher la valeur des variables.

*Chemin : **Vue> Barres d'outils et Ancrage de fenêtres> Galerie Widget***

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



Propriétés de champ



Propriété	Description
Valeur	Variable contenant les informations à afficher
Format numérique	Format d'affichage

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Propriété	Description
	• Aucun Aucune restriction (le système décide du format à utiliser) • Numérique Format numérique. Les chiffres décimaux et les chiffres en tête peuvent être utilisés pour mieux définir le format des nombres. • hex Format hexadécimal. Les chiffres en tête peuvent être utilisés pour mieux définir le format des nombres. • HEX Le même format « hexadécimal » mais en majuscules. • scientifique Format scientifique. Les chiffres décimaux peuvent être utilisés pour mieux définir le format des nombres. • SCIENTIFIQUE Le même format « scientifique » mais en majuscules. • Personnalisé Utilisez le paramètre "Personnalisé" complémentaire pour mieux définir le format à utiliser (voir tableau ci-dessous).
Chiffres décimaux	Nombre de chiffres décimaux à afficher (non disponible pour tous les types de formats)
Chiffres en tête	Nombre de chiffres en tête à afficher (non disponible pour tous les types de formats)
Personnalisé	Chaîne qui définit le format des nombres à utiliser (disponible uniquement lorsque le format des nombres sélectionné est PERSONNALISÉ)
Clavier	Type de clavier à faire apparaître pour modifier la valeur de l'onglet. • Aucun • Alphabétique, numérique, etc. Fait apparaître un clavier prédéfini ou un clavier utilisateur (voir "Claviers" Page 319) • Molette Le clavier ne sera pas affiché. La molette peut être utilisée pour incrémenter/décrémenter la valeur numérique • Macro Le clavier ne sera pas affiché. La macro clavier permet de saisir des touches (voir "Actions clavier" Page 145 en page 1).
Événements	
Action OnDataUpdate	Liste des commandes pour exécuter chaque fois que la valeur de la variable change (Voir "Actions" Page 139 pour les commandes disponibles)

Certaines propriétés ont quelques boutons :

Entrer dans le mode d'édition :
vous pouvez taper directement le nom de la variable à utiliser.

Attacher à une variable :
la boîte de dialogue où sélectionner la variable à utiliser s'ouvrira

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Formats personnalisés

Dans la propriété personnalisée, les caractères autorisés sont “#” “.” “0” “h” “H” “e” “E”

Utilisez les caractères de remplissage pour gérer l'affichage des chiffres avant et après la virgule décimale. Utilisez le signe numérique (#) si vous voulez afficher uniquement les chiffres significatifs d'un nombre. Ce signe ne permet pas l'affichage de zéros non significatifs. Utilisez le caractère numérique pour zéro (0) si vous voulez afficher des zéros non significatifs lorsqu'un nombre peut avoir moins de chiffres que ce qui a été spécifié dans le code du format.

Si un nombre a plus de chiffres à gauche du séparateur de décimales qu'il n'y a de caractères de remplissage dans le code du format, les chiffres supplémentaires sont affichés. Toutefois, si un nombre a plus de chiffres à droite du séparateur décimal qu'il n'y a de caractères de remplissage dans le code du format, le nombre est arrondi au même nombre de décimales qu'il y a de caractères de remplissage.

Exemples

Pour afficher	En tant que	Caractère de remplissage
123	000123	000#
1500	5DC	#H
1500	5dc	#h
1500	0005DC	000#H
123,456	123,46	#,##
123,456	000123,456000	000#,###000
12 200 000	1,22E+07	#0,00E+00
12 200 000	12,2E+6	#0,0E+0

Ajouter des variables

PB610 Panel Builder 600 utilise les noms de variables pour accéder à toutes les données du périphérique. Tous les champs et les emplacements de référence dans le périphérique doivent se voir attribué un nom de variable pour être utilisé dans le projet IHM.

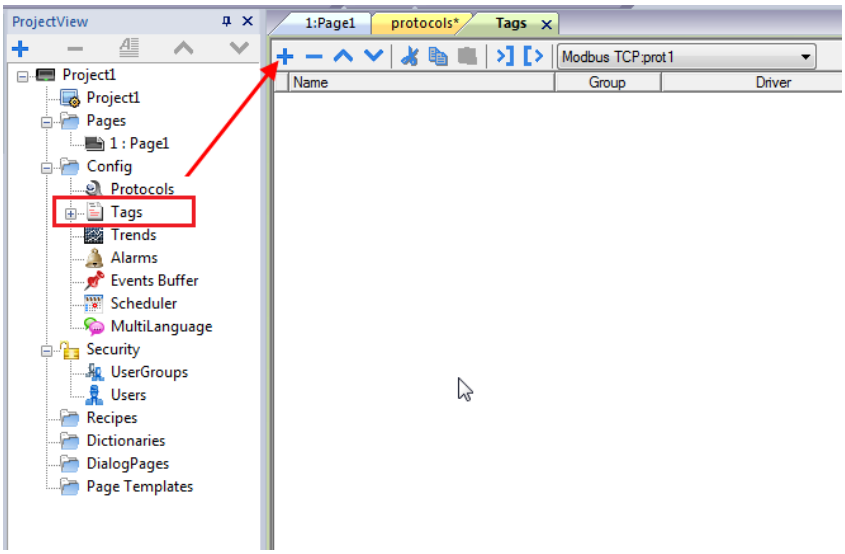
L'éditeur de variables peut être utilisé pour créer et gérer des variables. Après que les variables ont été définies, elles peuvent être utilisées dans le projet en les attachant aux propriétés des widgets.

Voir "Paramètres « Associer à »" Page 44 pour plus de détails.

Éditeur de variable

*Chemin : **VueProjet**> **Variables***

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



Ajouter une variable

1. Cliquez sur **+** et saisissez les données requises.
2. Sélectionnez l'adresse dans la boîte de dialogue d'adresse de protocole de communication : de nouvelles variables sont nommées Variable1, Variable2, etc.
3. Cliquez sur le nom de la variable pour la renommer.

Propriétés de variable

Voir la documentation de protocole spécifique pour les détails.

Propriété	Description
Nom	Nom de variable unique au niveau du projet. Clé primaire pour identifier les informations dans la base de données de variable du runtime. **AVERTISSEMENT : il n'est pas autorisé de dupliquer des noms de variables.**
Groupes	Noms de groupe associés à une variable
Pilote	Protocole de communication
Adresse	Adresse de mémoire du contrôleur. Pour modifier, cliquez sur le côté droit de la colonne pour obtenir la boîte de dialogue dans laquelle vous pouvez entrer les informations d'adresse.
Codage	Type de codage pour le type de données de chaîne (UTF-8, Latin1, UTF-2 et UTF-16)
Commentaire	Description de la variable

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Propriété	Description
Simulateur	Comportement des variables pendant la simulation. Plusieurs profils sont disponibles
Mise à l'échelle	Conversion appliquée à la variable avant le stockage de base de données. **Par formule** = définie comme une transformation linéaire. **Par plage** = définie comme une conversion de plage.

Les propriétés ci-dessous seront visibles uniquement après avoir sélectionné le mode « Afficher colonnes avancées » depuis la barre d'outil de l'éditeur de variable.

Propriété	Description
Nom de variable API	Nom de la variable de l'API d'origine, utilisé pour faire correspondre les variables utilisées par le périphérique (Nom de variable) et les étiquettes exportées depuis l'API L/É uniquement dans une vue avancée pour permettre des réglages en cas d'erreurs d'importation de variable.
Taux (ms)	Temps de rafraîchissement de la variable. Défaut = 500 ms. Quand le taux de rafraîchissement est défini sur « Manuel », le périphérique IHM ne lira pas automatiquement la variable du périphérique distant en arrière-plan. La variable n'est lue et rafraîchie dans la base de données que par l'action « ForceReadTag » ou en utilisant l'option forceRefresh dans le JavaScript getTag (). **AVERTISSEMENT : le taux de rafraîchissement des variables est le taux de rafraîchissement maximum. Le taux de rafraîchissement réel dépend du type de communication (série, bus de terrain, Ethernet), du protocole, de la quantité de données échangées.**
L/É	Attribut de variable L/É (L/É, L ou É). Le contenu de variables d'écriture seule est toujours écrit et jamais lu : lorsque la communication n'est pas active, le contenu de ces variables peut ne pas être disponible dans les widgets.
Actif	Mode mise à jour. **faux** = les variables sont lues à partir du contrôleur uniquement lorsque requis par le périphérique IHM. **vrai** = les variables sont continuellement lues même si non requis par la page affichée. **Important : laissez cette valeur sur faux pour une performance de communication plus élevée.**

Gestion des noms des variables

Les noms de variables doivent être uniques au niveau de projet. Si les mêmes variables, du même fichier de symboles doivent être utilisés pour les deux contrôleurs différents, utilisez la fonction « Alias » pour ajouter un préfixe aux variables importées et les rendre uniques au niveau du projet.

Remarque : seuls certains protocoles prennent en charge la fonctionnalité « Alias ».

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Voir "Protocoles de communication" Page 545 pour plus de détails.

Gestion des groupes de variables

Les variables utilisées dans chaque page sont identifiées comme faisant partie d'un groupe, de sorte que les demandes faites par le protocole de communication au/aux contrôleur(s) connecté(s) peuvent être traitées plus rapidement : seules les variables incluses dans la page affichée sont consultées par le contrôleur.

Associer un widget à des variables

Pour contrôler un widget et l'animer avec des données actives, vous pouvez associer une propriété spécifique à des sources de données différentes. Par exemple, vous pouvez associer la jauge de la propriété **Valeur** à une variable de température testée, ou la propriété **Affichage** à des données de recette.

Sources des données

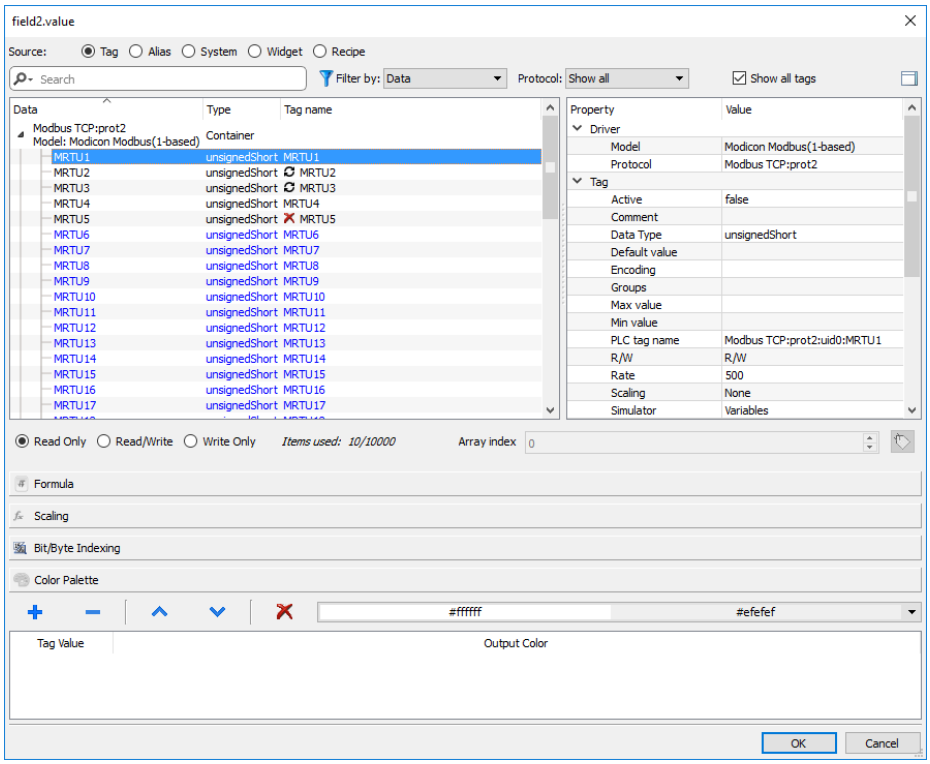
Éléments auxquels vous pouvez associer une propriété d'objet :

Source des données	Description
Variable	Variable définie dans l'éditeur de variables
Alias	Adresse de variables indexées
Système	Variables à système prédéfini (consultez "Variables système (Associer à)" Page 95)
Widget	Connecte à la propriété d'un widget (la valeur d'un widget de barre de défilement, par exemple)
Recette	Données du gestionnaire de recette (consultez "Recettes" Page 219)

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Associer une propriété à une variable

1. Dans le volet **Propriétés**, cliquez sur **+**.
2. Dans **Source**, choisissez la source de données : dans la liste, choisissez un protocole et la variable. Utilisez la barre **Rechercher**pour filtrer les variables.



3. Configurez le type d'accès (par exemple **Lecture seule**). Le champ **Index de tableau** s'affiche lorsque la variable sélectionnée est un tableau, afin d'identifier l'élément du tableau à utiliser. Le mode index indirect est pris en charge grâce à une variable supplémentaire.
4. Cliquez sur **OK** pour confirmer.

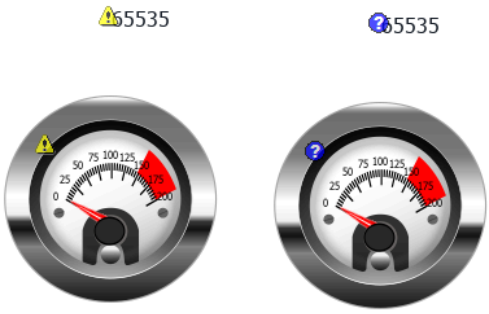
Les icônes adjacentes au nom de variable sont en surbrillance lorsqu'une définition ne correspond pas à la définition de variable dans le dictionnaire, ou lorsqu'une telle définition est manquante. Lorsque vous sélectionnez **Afficher toutes les variables**, toutes les variables du dictionnaire s'affichent, même celles que vous n'avez pas importées dans l'application.

Voir "Paramètres « Associer à »" Page 44 pour plus de détails.

Erreur de communication

Deux icônes peuvent apparaître près des widgets qui ont une variable associée.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



- : erreur de communication
- : données actuellement non disponibles (protocole de communication lent)

Pages de dialogues

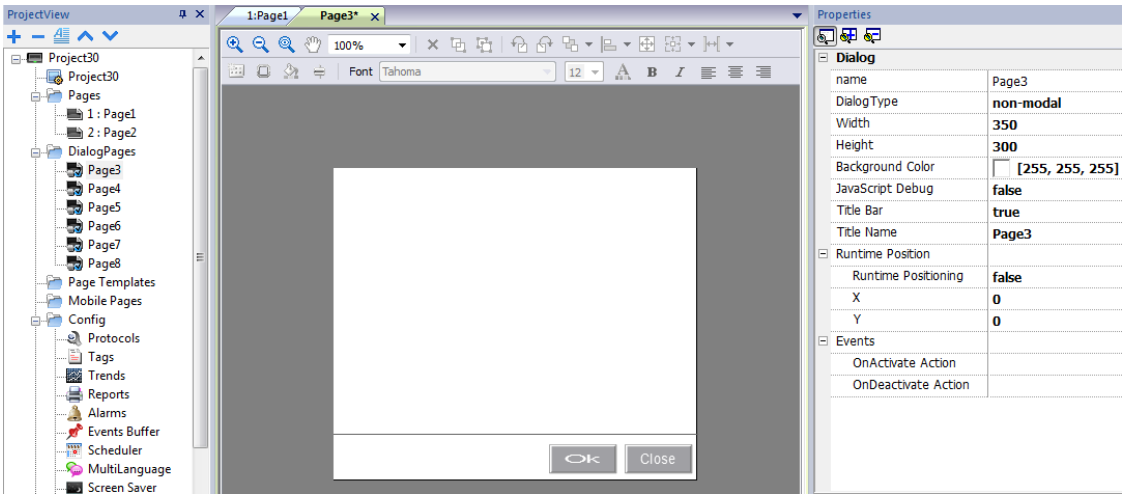
*Chemin : **VueProjet> Dialogues***

Les pages de dialogues s'ouvrent au runtime au-dessus de la page actuelle sur requête du projet. Elles sont utilisées pour notifier les alarmes, les erreurs ou pour requérir une action de l'utilisateur.

Propriétés des dialogues principaux

Propriété	Description
Type de dialogue	**modal** = l'utilisateur ne peut pas retourner sur la fenêtre principale du projet/sur la page tant que le dialogue n'est pas fermé. **non-modal** = l'utilisateur peut continuer d'utiliser la fenêtre principale du projet (ou d'autres dialogues non-modaux) lorsqu'un dialogue est affiché au-dessus.
Barre de titre	**vrai** = titre du dialogue affiché **faux** = titre du dialogue non affiché
Nom de titre	Titre de dialogue. Uniquement si **Barre de titre**= vrai.
Position du runtime	Position fixe du dialogue **faux** = le dialogue s'affiche au centre de l'écran **vrai** = le dialogue s'affiche dans l'angle en haut à gauche à la position X et Y

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



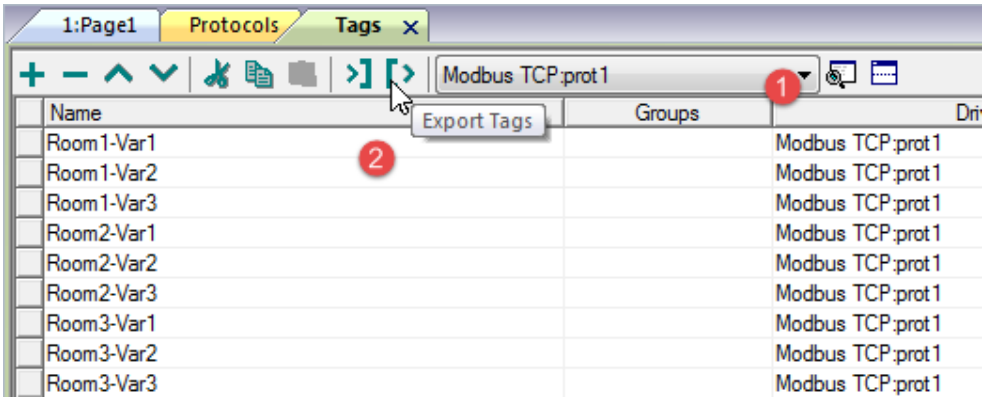
Nombre maximal de dialogues

Le nombre maximum de boîte de dialogue ouvert est défini dans "Précisions fonctionnelles et compatibilité" Page 539.

Lorsque le nombre maximal de dialogues ouverts est atteint, le dialogue le moins récent est fermé pour en ouvrir un nouveau.

Exporter des variables

*Chemin : **VueProjet> Variables***



1. Sélectionnez le protocole pour les variables que vous voulez exporter.
2. Cliquez sur le bouton **Exporter variables** : toutes les configurations des variables pour les protocoles sélectionnés sont exportées dans un fichier .xml.

Vous pouvez éditer le fichier .xml qui en résulte avec des outils tiers (par exemple, Microsoft Excel) puis réimporter le fichier modifié (voir "Importer des variables" Page opposée pour plus de détails).

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Importer des variables

Introduction

Certains protocoles permettent d'importer des variables stockées dans un fichier séparé par une virgule (.csv ou autres formats). Référez-vous à la section Importation des variables de chaque protocole pour obtenir plus de détails (voir "Protocoles de communication" Page 545).

L'importation est un processus en deux étapes :

1. Importation de la définition de variable dans un dictionnaire
2. Importation des variables du dictionnaire vers le projet

AVERTISSEMENT : les caractères spéciaux dans les noms de variables tels que "&" entraînent des erreurs de communication. Voir "Limites de la prise en charge UNICODE" Page 262

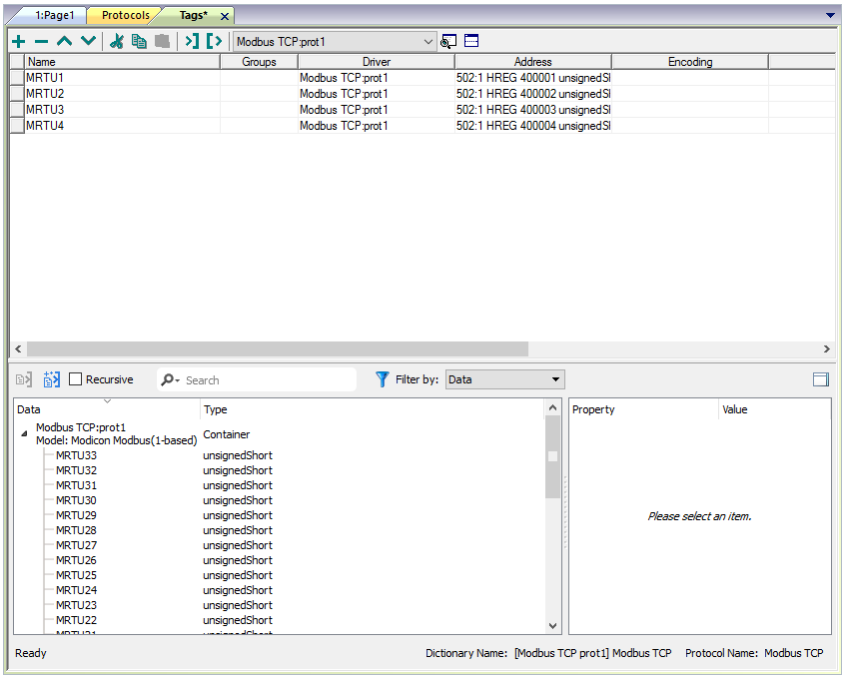
Remarque : au moment de l'importation, le caractère « . » dans les noms de variable est remplacé par « / ». Le protocole utilisera la bonne syntaxe lorsqu'il communiquera avec l'API.

Dictionnaires

*Chemin : **VueProjet> Dictionnaires***

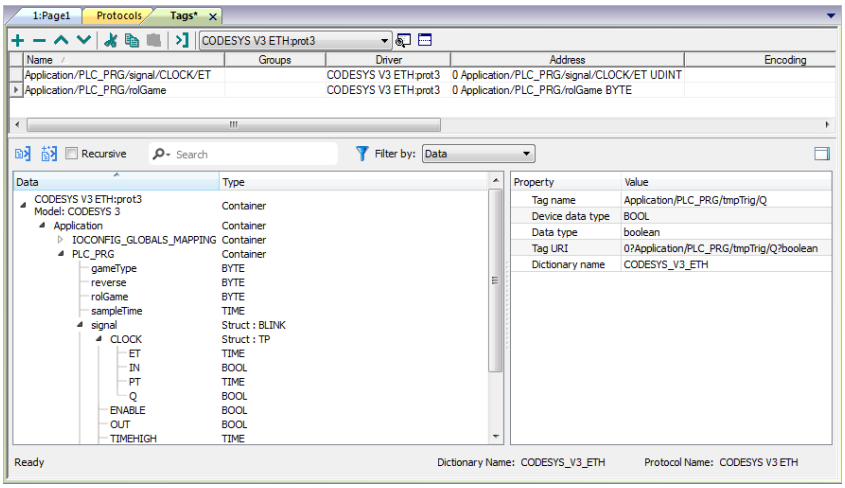
Un dictionnaire est une liste de variables importées dans l'Éditeur de variable pour un protocole spécifique.En fonction du type de protocole, les variables s'affichent en vue linéaire ou en vue hiérarchique.

Vue linéaire



Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

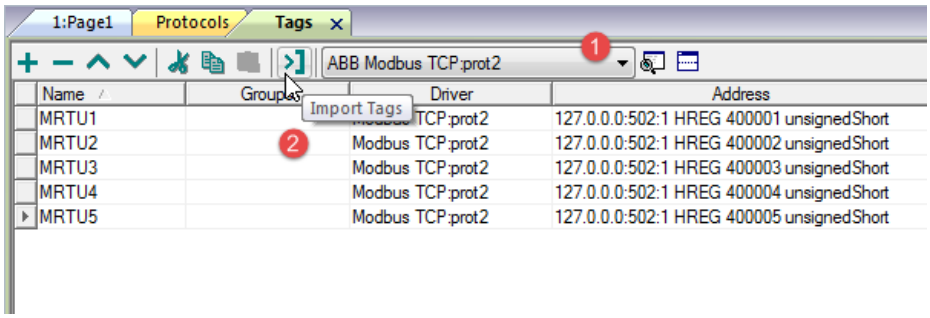
Vue hiérarchique



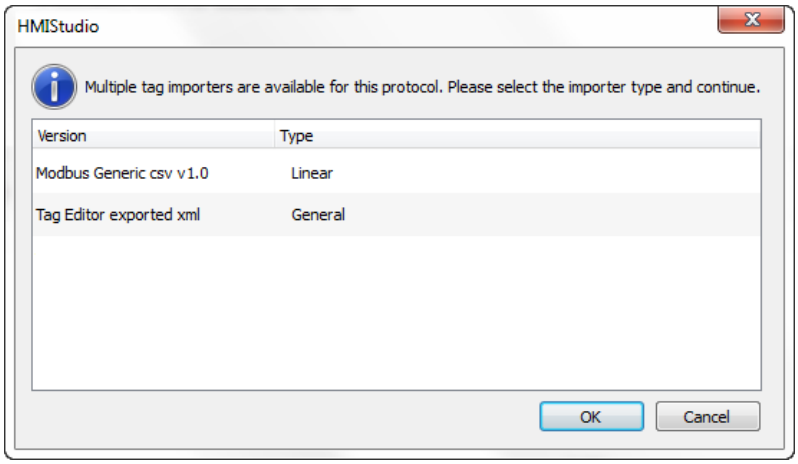
Importer des variables

Pour importer des variables à partir d'un fichier externe :

1. Dans **VueProjet, Variables**, sélectionnez le protocole à partir de la liste du filtre.



2. Cliquez sur le bouton **Importer variables** : la boîte de dialogue sélectionner fichier apparaît. Une boîte de dialogue pour choisir le type de pilote d'importation apparaît.



Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

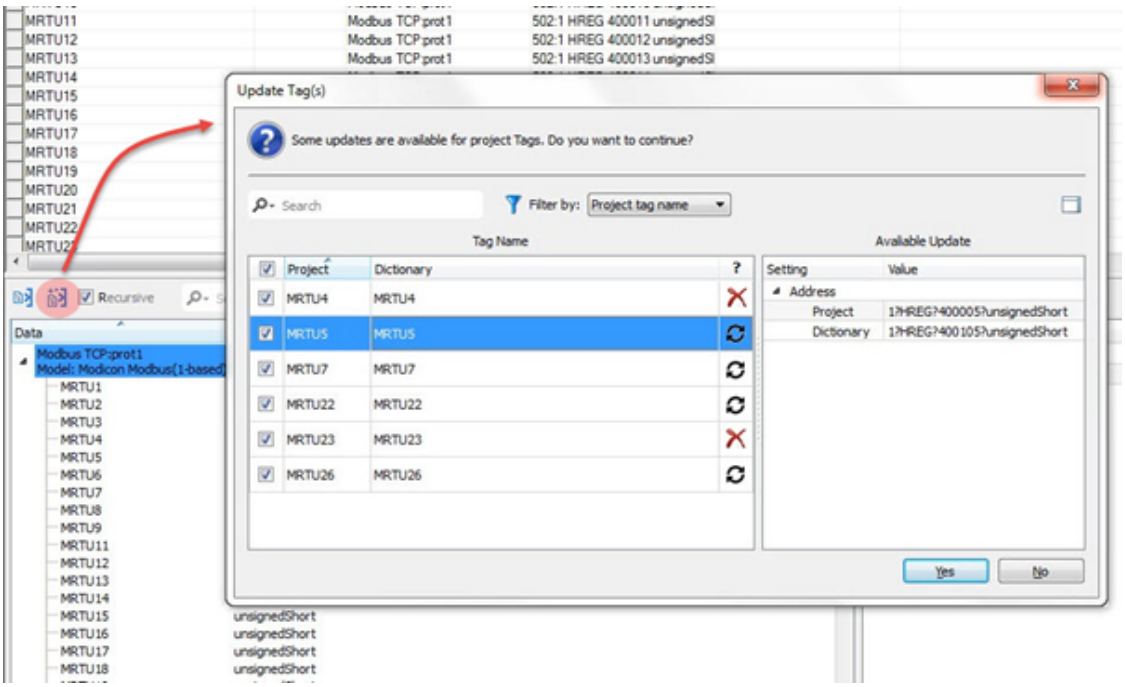
3. Sélectionner le fichier : une liste de variables est représenté dans une vue linéaire ou hiérarchique.
4. Pour importer des variables, sélectionnez une ou plusieurs variables ou bien un Nœud (en vue hiérarchique seulement) et cliquez sur le bouton **Importer variable** : les variables sont copiées vers le projet et apparaissent sous forme de liste dans la partie supérieure de la fenêtre.

Paramètre	Description
Récursif	Tous les éléments de la structure sont importés dans des variables séparées.

Remarque : lorsque le projet a été configuré pour utiliser un réseau de protocole, vous devez également sélectionner le nœud de protocole vers lequel les variables doivent être importées. Vous pouvez importer les mêmes variables sur de multiples protocoles. Lorsque le fichier des variables contient les informations de nœud, vous pouvez choisir d'utiliser ces informations pour filtrer les variables et n'importer que celles qui correspondent aux nœuds sélectionnés.

Mettre à jour les variables importées

En utilisant la commande Mettre à jour variable(s), vous pouvez réimporter des variables. Une boîte de dialogue vous permet de sélectionner les variables à réimporter :



Ces variables doivent être mises à jour. Un liste de différences entre le projet et le dictionnaire s'affiche.

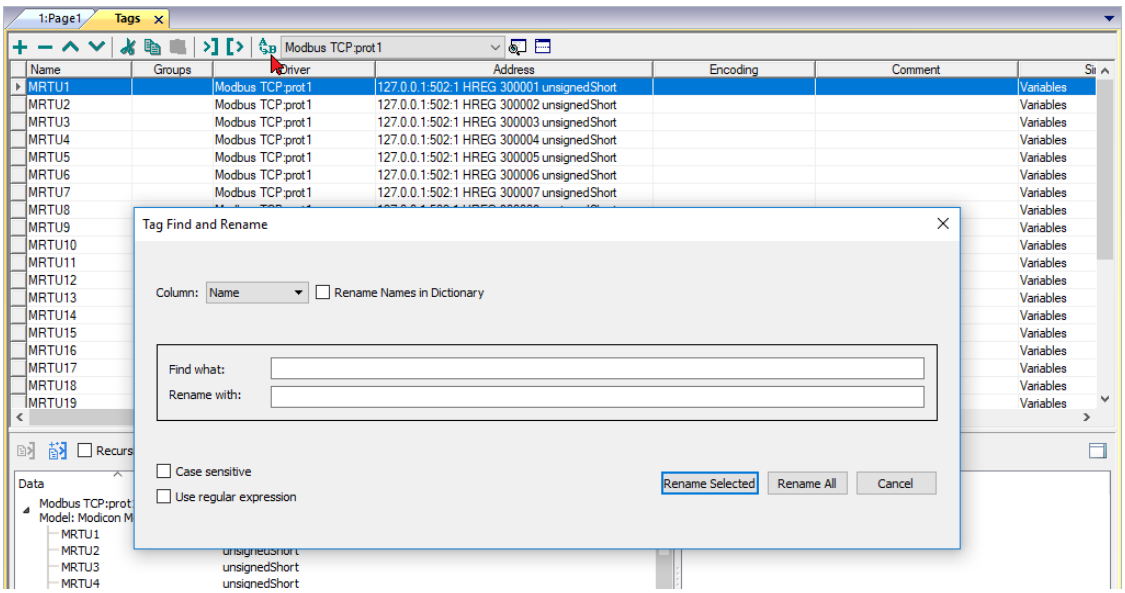
Ces variables ne sont plus disponibles dans le dictionnaire. Si elles sont mises à jour, ces variables seront éliminées du projet.

Rechercher et renommer une variable

La fonction rechercher et renommer des variables renommera une référence de variable à l'intérieur du projet en entier.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Notez que cette fonction peut être utilisée non seulement pour renommer des variables, mais aussi pour changer les valeurs de chaque colonne de l'éditeur de variables



Paramètres	
Colonne	Sélectionnez la colonne à modifier à l'aide de la fonction rechercher et renommer
Renommer des noms dans le dictionnaire	Le renommage des noms des variables sera étendu pour renommer même les noms des variables du dictionnaire interne. Ce paramètre est utile lorsque vous devez remplacer le dictionnaire par un autre dictionnaire contenant des variables renommées. Ce paramètre n'est disponible que lorsque la colonne sélectionnée est « Nom ».
Find what	Chaîne pour chercher
Rename with	Chaîne pour remplacer
Case sensitive	Prend en compte les lettres majuscules et minuscules
Use regular expression	Active l'expression régulière dans le modèle chercher/remplacer Voir https://fr.wikipedia.org/wiki/Expression_régulière pour plus de détails sur les expressions régulières. Lorsque l'expression régulière est activée, le paramètre « Find what » n'offre pas de valeurs prédéfinies mais seulement un traitement de texte libre.

RENAME SELECTED

Exécute le renommage uniquement pour les variables sélectionnées.

RENAME ALL

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Exécute le renommage pour l'ensemble de la base de données des variables.

Les références utilisées dans JavaScript et dans les widgets personnalisés ne seront pas mises à jour. Undo n'est pas autorisé pour cette commande

Exemple d'expression régulière

En utilisant la liste des variables de l'image ci-dessus.

Si vous voulez ajouter un préfixe à toutes les variables, vous n'avez pas besoin d'utiliser une expression régulière :

Find what :	**MRTU**
Rename with	**PLC01_MRTU**

Mais si vous voulez ajouter un postfixe, vous devez utiliser une expression régulière :

Find what :	**MRTU(.*)**
Rename with	**MRTU\1_PLC01**

Où

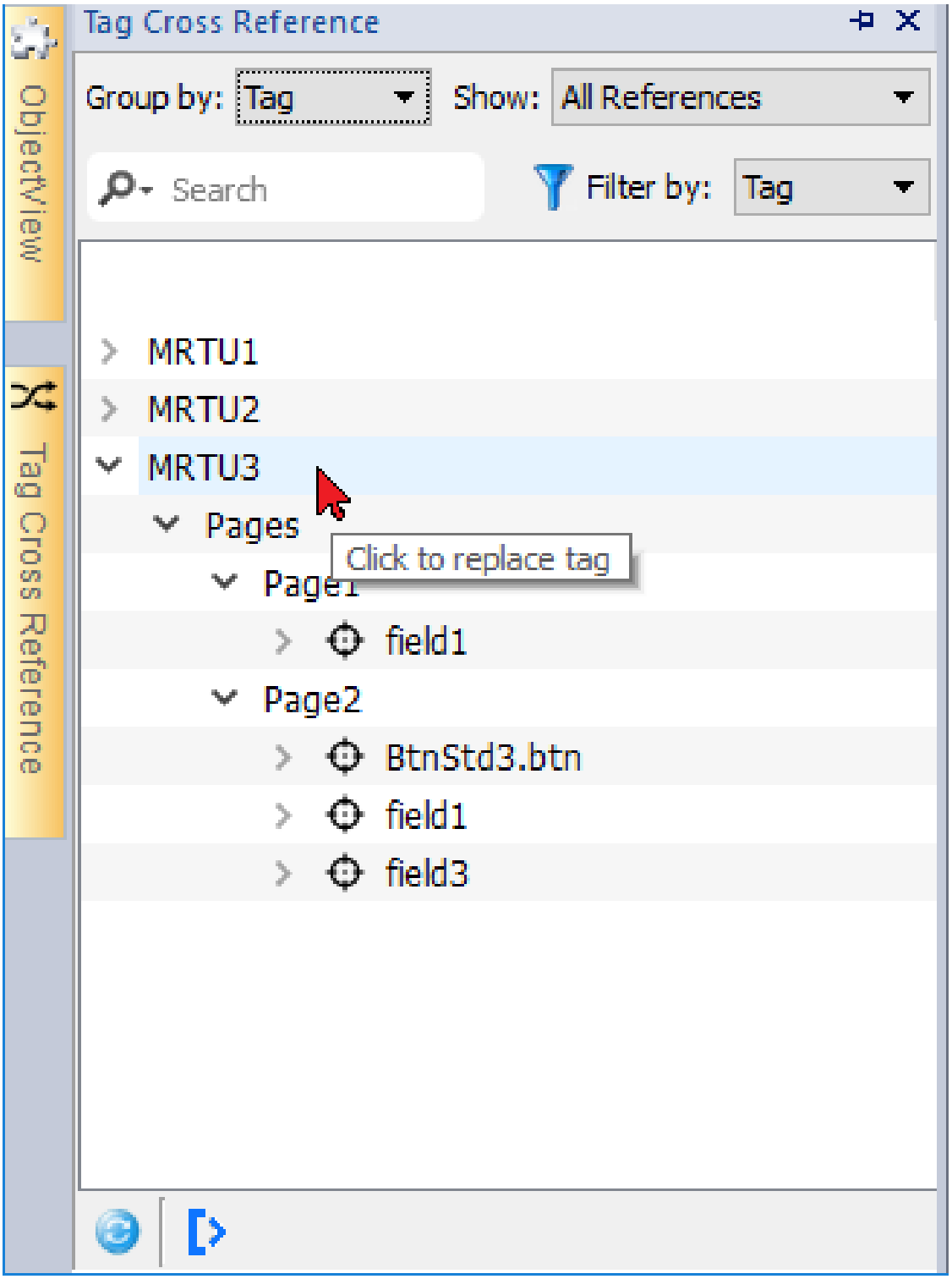
(.*)	signifie toute suite de caractères
\1	est une copie de la première suite de caractères entourée de (...) trouvée dans la chaîne de recherche

Rechercher et remplacer une variable

À l'aide de cette fonction, vous pouvez rechercher toutes les occurrences d'une variable à l'intérieur du projet et la remplacer par une autre variable.

Dans la vue Référence croisée des variables, cliquez sur la variable que vous voulez remplacer

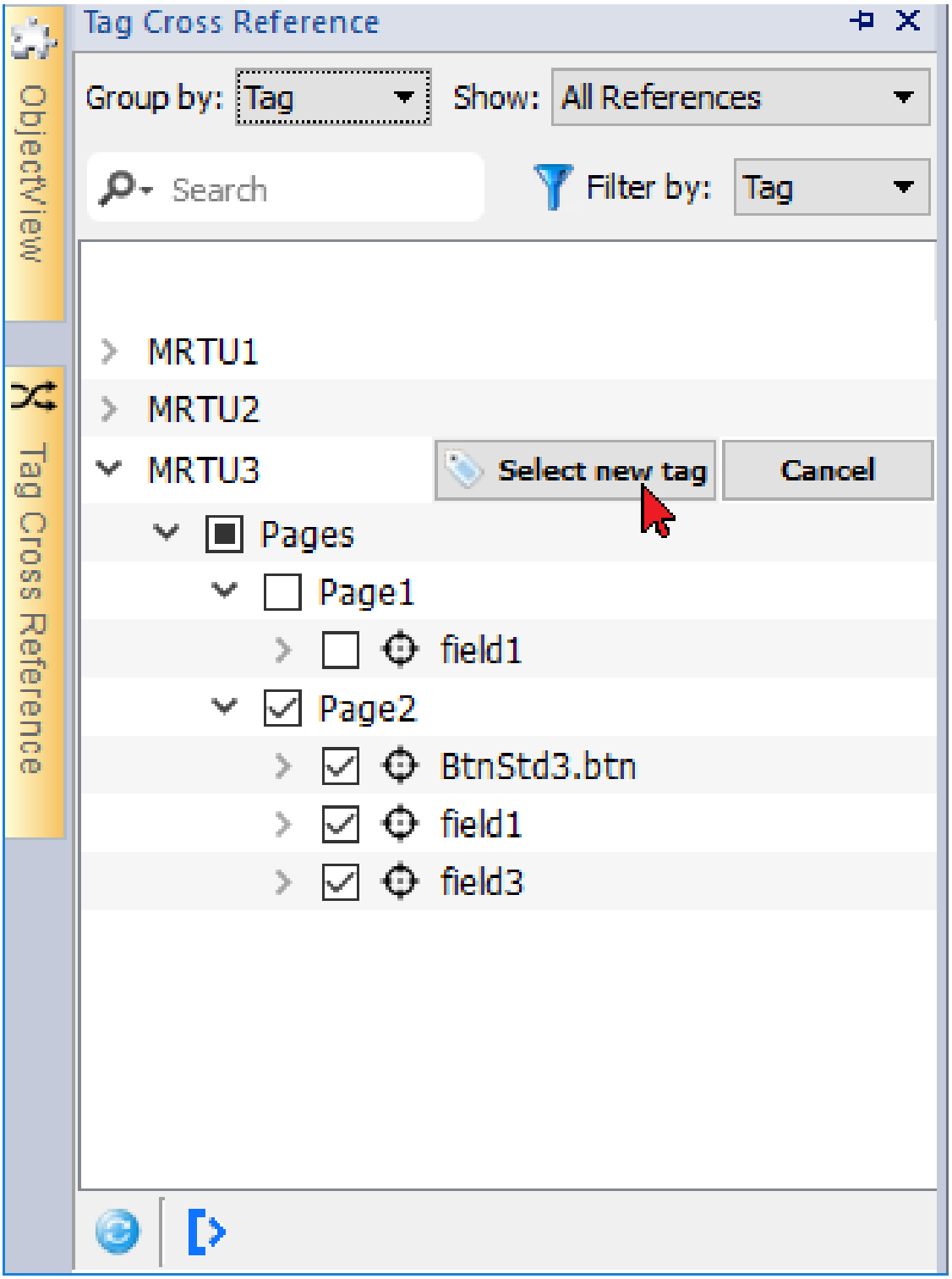
Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



En utilisant les cases à cocher, sélectionnez l'endroit où vous voulez appliquer le remplacement,

puis cliquez sur le bouton « Sélectionner une nouvelle variable pour remplacer les liaisons de données des objets sélectionnés ou appuyez sur « Annuler » pour abandonner l'opération.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



Les références utilisées dans le script Java et dans les widgets personnalisés peuvent ne pas être listées.
Undo n'est pas autorisé pour cette commande

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

5 Concepts de programmation

La programmation de PB610 Panel Builder 600 est basée sur plusieurs concepts et comportements fondamentaux.

Types de données **44**

Paramètres « Associer à » **44**

Formule **50**

Événements **54**

Positionnement des widgets **57**

Gestion widgets chevauchants **58**

Grouper des widgets **59**

Changer les propriétés de plusieurs widgets **66**

Changer la propriété couleur de remplissage en fonction des valeurs de variable **67**

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Types de données

Lorsque vous créez une variable, vous devez spécifier ses propriétés. Les types de données sont spécifiques à PB610 Panel Builder 600, les types de mémoires sont spécifiques au protocole sélectionné. Choisissez la valeur en fonction de la représentation interne dont vous avez besoin pour l'adresse de contrôleur sélectionnée.

Note : les types tableaux utilisent les mêmes types de données suivis par « [ ] » (c-à-d : booléen [ ])

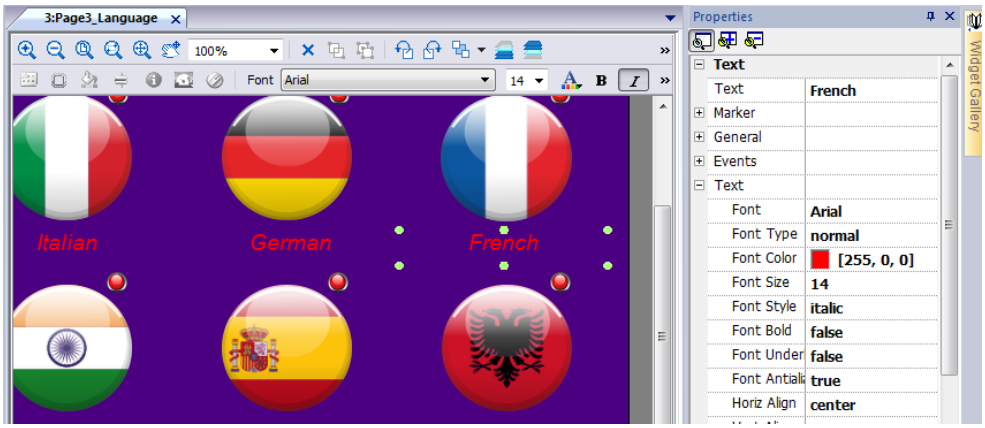
Type de données	Description
booléen	Données d'un bit (0 à 1)
octet	Données signées de 8 bits (-128 à 127)
double	Type virgule flottante de 64 bits double précision conforme IEEE (±2.2e-308 à ±1.79e308)
flottant	Type virgule flottante de 32 bits simple précision conforme IEEE (±1.17e-38 à ±3.40e38)
entier	Données signées de 32 bits (-2.1e9 à 2.1e9)
court	Données signées de 16 bits (-32768 à 32767)
chaîne	Caractères codés selon le format sélectionné
temps	Données de l'heure
octet non signé	Données non signées de 8 bits (0 à 255)
entier non signé	Données non signées de 32 bits (0 à 4.2e9)
court non signé	Données non signées de 16 bits (0 à 65535)
uint64	Données non signées de 64 bits (0 à 264 − 1)

Paramètres « Associer à »

Propriétés d'objet

Dans PB610 Panel Builder 600, les propriétés d'un objet placé sur une page peuvent être définies au moment de la programmation ou configurées pour être dynamiques. Pour changer une propriété au moment de la programmation, utilisez la barre d'outils de la page ou le volet Propriété.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

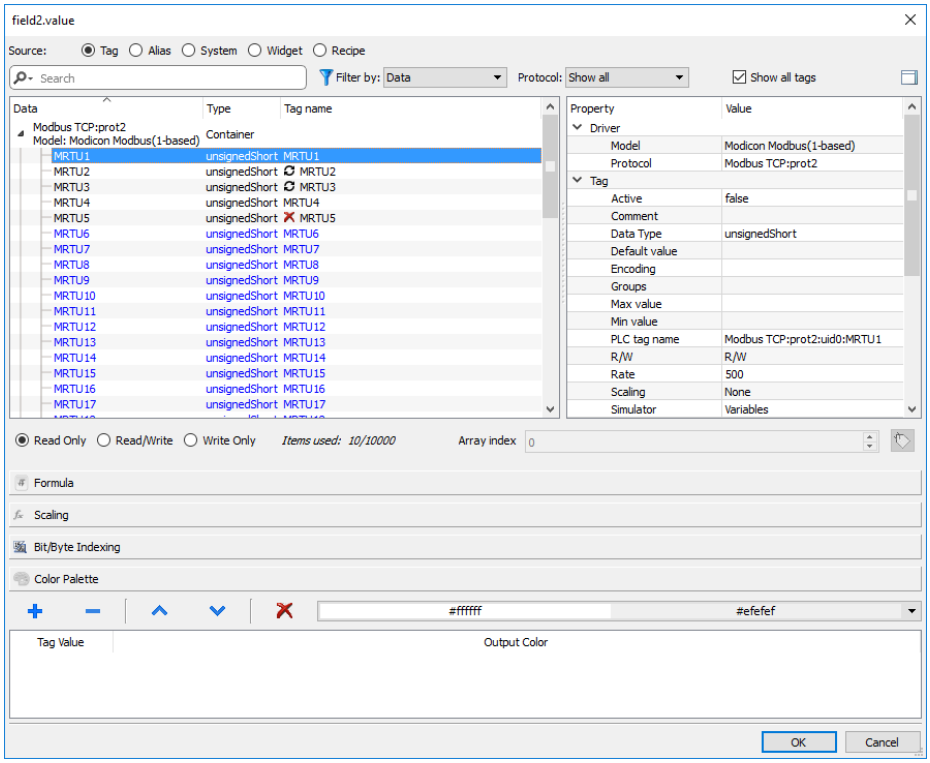


La barre d'outils de la page affiche uniquement les propriétés les plus communes de l'objet, alors que le volet Propriétés affiche toutes les propriétés dans un mode d'affichage simple ou avancé.

Pour changer une valeur de propriété dynamiquement, vous pouvez l'associer à des variables ou à des variables.

Associer une propriété à une variable

1. Dans le volet **Propriétés**, cliquez sur **+**.
2. Dans **Source**, choisissez la source de données : dans la liste, choisissez un protocole et la variable. Utilisez la barre **Rechercher**pour filtrer les variables.



3. Configurez le type d'accès (par exemple **Lecture seule**). Le champ **Index de tableau** s'affiche lorsque la variable sélectionnée est un tableau, afin d'identifier l'élément du tableau à utiliser. Le mode index indirect est pris en charge grâce à une variable supplémentaire.
4. Cliquez sur **OK** pour confirmer.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Les icônes adjacentes au nom de variable sont en surbrillance lorsqu'une définition ne correspond pas à la définition de variable dans le dictionnaire, ou lorsqu'une telle définition est manquante. Lorsque vous sélectionnez **Afficher toutes les variables**, toutes les variables du dictionnaire s'affichent, même celles que vous n'avez pas importées dans l'application.

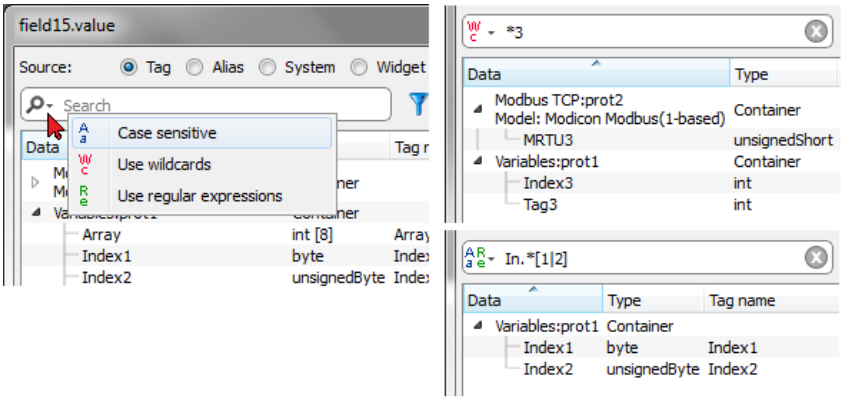
Sources des données

Éléments auxquels vous pouvez associer une propriété d'objet :

Source des données	Description
Variable	Variable définie dans l'éditeur de variables
Alias	Adresse de variables indexées
Système	Variables à système prédéfini (consultez "Variables système (Associer à)" Page 95)
Widget	Connecte à la propriété d'un widget (la valeur d'un widget de barre de défilement, par exemple)
Recette	Données du gestionnaire de recette (consultez "Recettes" Page 219)

Recherche avancée

Diverses options de syntaxe peuvent s'appliquer à une zone de recherche :

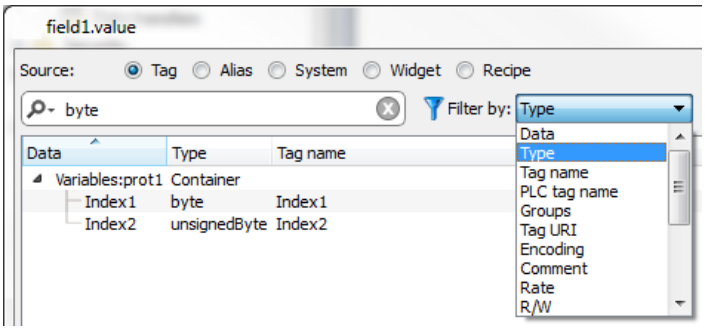


Options principales	Fonction
Métacaractères	Recherche en associant des métacaractères simples. Le caractère '?' correspond à n'importe quel caractère. Le caractère ' *' correspond à aucun ou au-moins un caractère. " […]" : des jeux de caractères peuvent être représentés entre crochets.
Expression régulière	Décrit le modèle de caractère. Voir https://fr.wikipedia.org/wiki/Expression_régulière pour plus de détails sur les expressions régulières.

Filtrage des variables

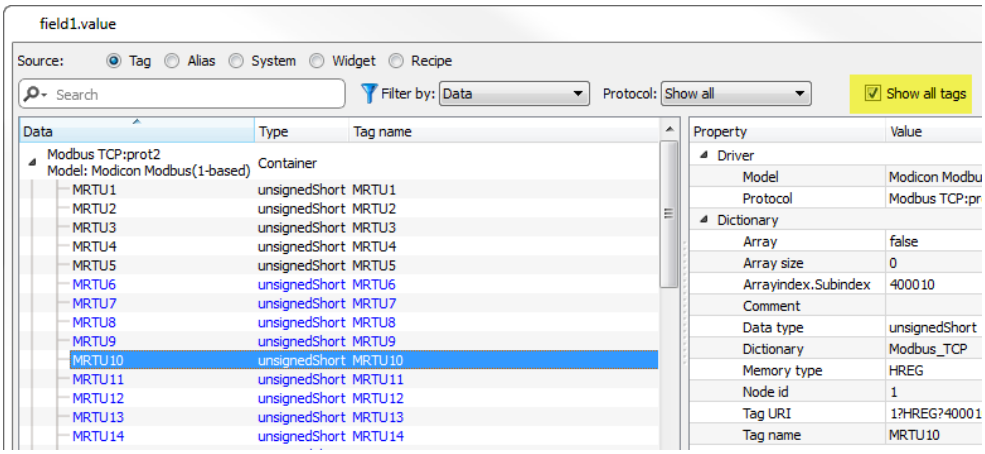
Choisissez différents critères de filtrage des variables :

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

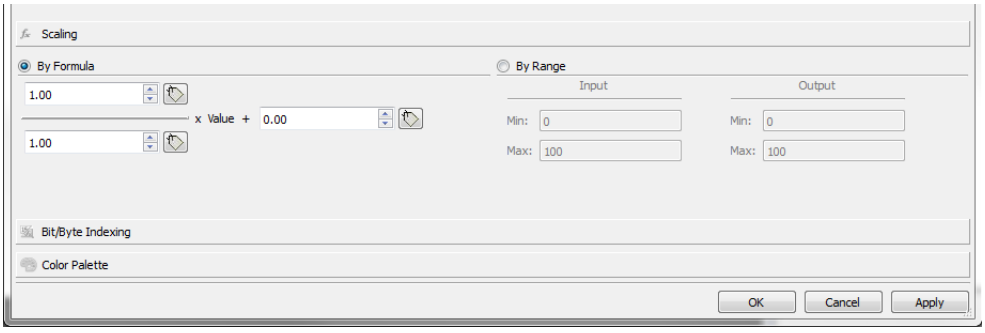


Affichage des variables de dictionnaire

Lorsque **Afficher toutes les variables** est coché, les variables qui appartiennent à un dictionnaire mais qui n'ont pas encore été importées apparaissent en bleu. Vous pouvez sélectionner une variable et cliquer deux fois dessus pour l'importer dans le projet.



Conversion de la valeur de variable

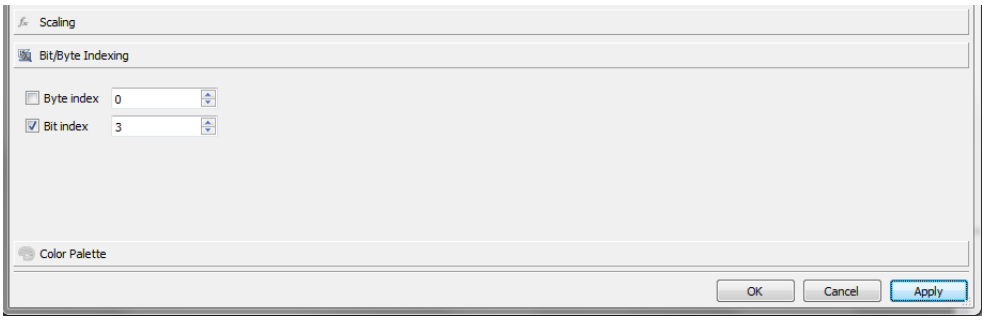


L'onglet **Mise à l'échelle** convertit la valeur de la variable. Dans la section **Par plage**, définissez la plage d'entrée et la plage de sortie : le système calcule automatiquement les facteurs de mise à l'échelle.

Extraction des variables bit/byte basées sur l'index

Permet d'extraire le contenu d'un mot d'un seul bit ou d'un seul byte selon le numéro de bit ou de byte spécifié

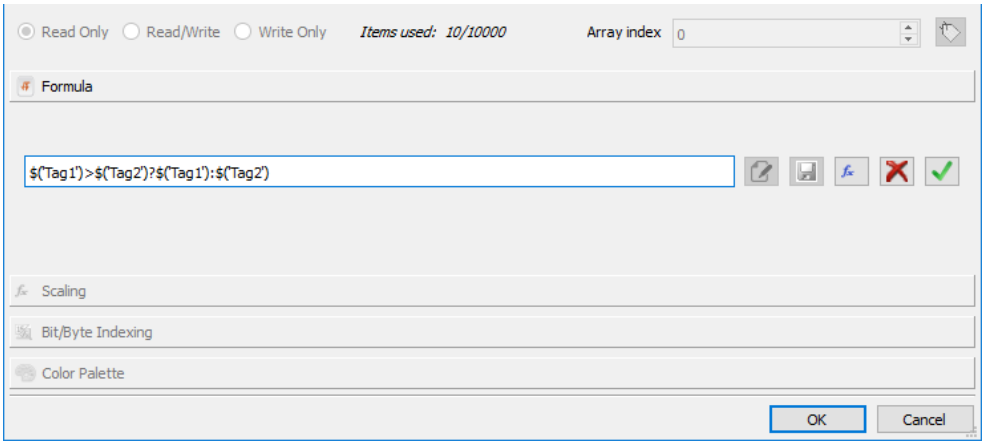
Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



Utilisez une formule pour calculer la valeur à utiliser.

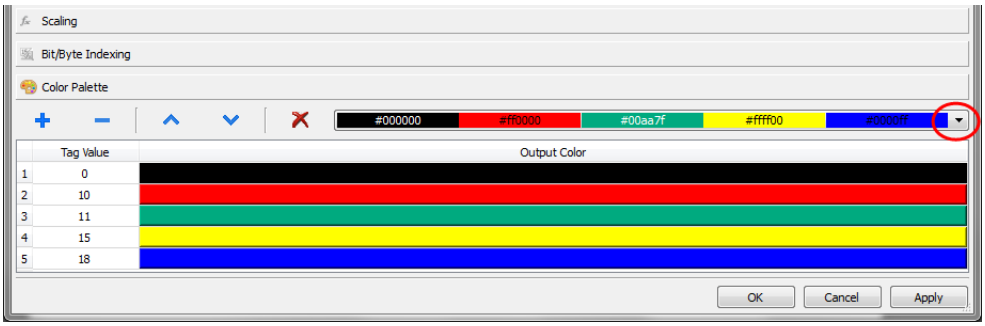
L'utilisation d'une formule permet de calculer la valeur à utiliser. Voir le chapitre "Formule" Page 50 pour plus de détails.

Notez qu'à l'aide d'une formule, la liaison de données sera en lecture seule.



Mappage des valeurs de variable aux couleurs

Permet de mapper des valeurs de variable numérique aux couleurs. Par exemple, vous pouvez utiliser cette option pour changer la couleur d'un bouton.



Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Section	Fonction
	Sur la barre d'outil, ajoutez/supprimez ou déplacez vers le haut/vers le bas les lignes de couleurs. Vous pouvez modifier la valeur de variable et les valeurs de séquence.
	La dernière combinaison de couleurs définie est enregistrée automatiquement et vous pouvez la récupérer dans la barre d'outils des couleurs.

La valeur de la variable peut être une plage de valeurs séparées par une virgule, par exemple :

- 5, 10-15, 20
- A, AB, C

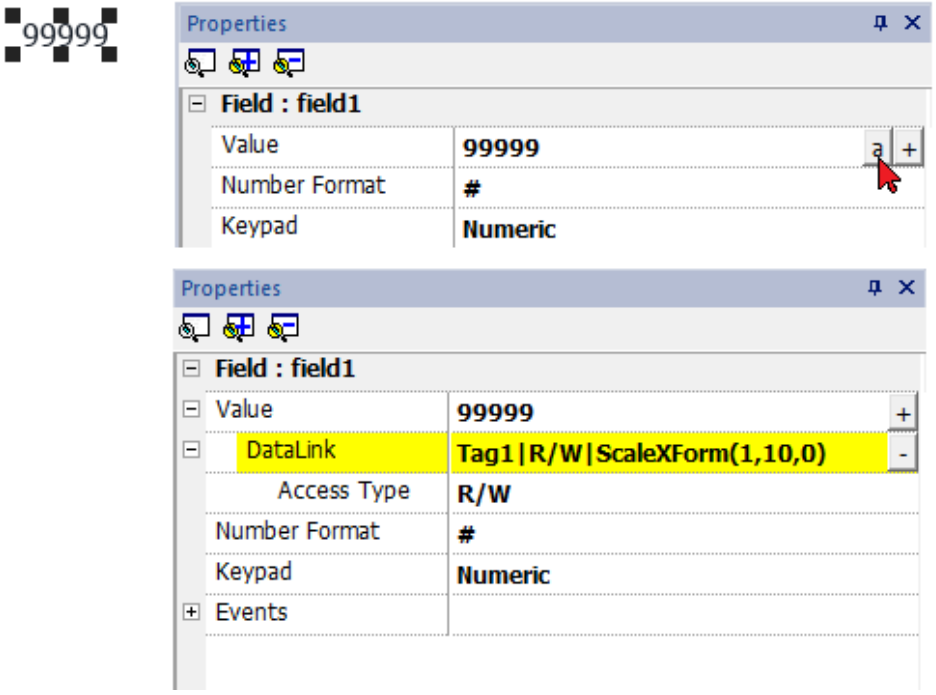
Il est de la responsabilité du développeur de l'application de définir correctement tous les éléments pour couvrir toutes les valeurs possibles de l'application, nous pourrions avoir une couleur inattendue lorsque la valeur n'est pas définie à l'intérieur de la palette des couleurs définies.

Notez que la valeur de variable de mappage de la couleur renverra un type de données de chaîne (par ex. « "#FF0000 »)

Sérialisation de liaison de données

Au lieu de l'utilisation ci-dessus de la boîte de dialogue « attacher à ... », les liaisons de données peuvent être entrées, ou modifiées, manuellement.

Cliquez sur **un** bouton dans le panneau **Propriétés** et entrez le texte qui décrit la liaison de données



Le format de liaison de données est :

```
Tagname [index] | [Atribute] | [XForm] | [XForm] | ...
```

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

en cas de formule :

```
= <formule>
```

Exemple :

- arrayTag[2]
- Tag[0|index]
- Alarm triggered:_SysPropMgr
- Tag|R/W|ScaleXForm(1,10,0)
- Tag|R/W|ScaleXForm(1,10,0)|ByteIndexXForm(1)|ColorPaletteCustomXForm(0#00aa7f,1#ff0000)
- =$('Tag1')>$('Tag2')?$('Tag1'):$('Tag2')
- =$Contains($('Tag4'),$('Tag3'))
- =$Pow(2,$('Tag2'))

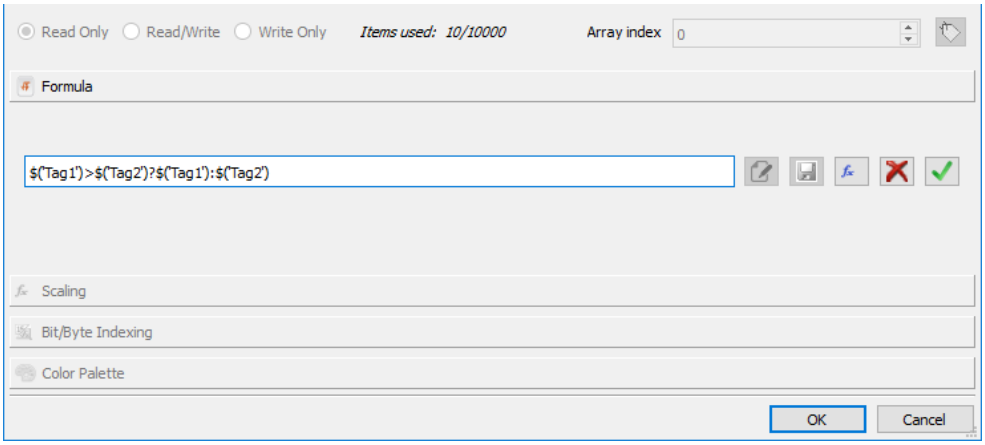
Formule

Une formule est une expression composée de :

- Opérateurs : il peut s'agir d'opérations mathématiques de base, d'opérateurs logiques, d'opérateurs de comparaison ou d'opérateurs de chaîne de base.
- Opérandes : peuvent être des libellés (nombres et chaînes de caractères utilisés comme constantes) et des références à des variables.

Les parenthèses rondes sont prises en charge en tant qu'opérateurs prioritaires. L'opérateur $ sera utilisé pour appeler des fonctions et, en particulier, pour faire référence à une variable (voir les exemples ci-dessous).

La boîte de dialogue Attacher à permet d'utiliser une formule pour calculer la valeur à retourner.



Commandes

Entrer dans le mode d'édition

Sauvegardez la formule saisie dans la bibliothèque de formules pour avoir la possibilité de réutiliser la même formule à d'autres endroits du projet.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Ouvrez la bibliothèque de formules pour sélectionner une formule déjà définie.

retirer la formule saisie

confirmer la formule saisie

Lorsque vous êtes en mode édition, vous pouvez simplement éditer la formule et double-cliquer sur les variables ou fonctions de la bibliothèque pour les ajouter à l'intérieur de la formule.

Exemple de formule :

- =$('Tag1')+$('Tag2')
- =$('Tag1')&$('Tag2')
- =$('Tag1')>$('Tag2')?$('Tag1'):$('Tag2')
- =$Pow(2,$('Tag2'))
- =$Contains($('Tag1'),$('Tag2'))

Syntaxe de formule

Opérations de base	
'Text'	Chaîne littérale
NUMBER	Nombre littéral, p. ex. 169857 ou 13,547
Chaîne(…)	Envoie vers la chaîne (notez qu'il n'y a pas de $)
Nombre(…)	Envoie vers la nombre (notez qu'il n'y a pas de $)
$FuncName(param1, param2, …)	Appel de fonction générale. (Par défaut et utilisateur)
$('TagName')	Variable, ou propriété de widget, ou recette, etc. **Notez que le nom de la variable doit être une chaîne littérale.**
$('TagName')[index]	Elément d'une variable de tableau. **Notez que le nom de la variable doit être une chaîne littérale.**
exp1 ? exp2 : exp3	Expression ternaire. Si exp1 est vrai, alors exp2 est pris, sinon exp3 est pris. C'est comme utiliser l'instruction if/then/else.

Opérateurs mathématiques	
+	Addition
-	Soustraction

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Opérateurs mathématiques	
*	Multiplication
/	Division
%	Module

Opérateurs de type bit	
&	Définit chaque bit sur 1 si les deux bits sont 1.
|	Définit chaque bit sur 1 si un ou deux bits sont 1.
~	Inverse tous les bits
^	Définit chaque bit sur 1 si un ou deux bits sont 1.
<<	Décale vers la gauche en poussant des zéros à partir de la droite et laisse tomber les bits les plus à gauche.
>>	Décale vers la droite en poussant des copies du bit le plus à gauche à partir de la gauche, et laisse tomber les bits les plus à droite.
>>>	Décale vers la droite en poussant des zéros à partir de la gauche et laisse tomber les bits les plus à droite.

Opérateurs logiques	
&&	AND
||	OR
!	NOT

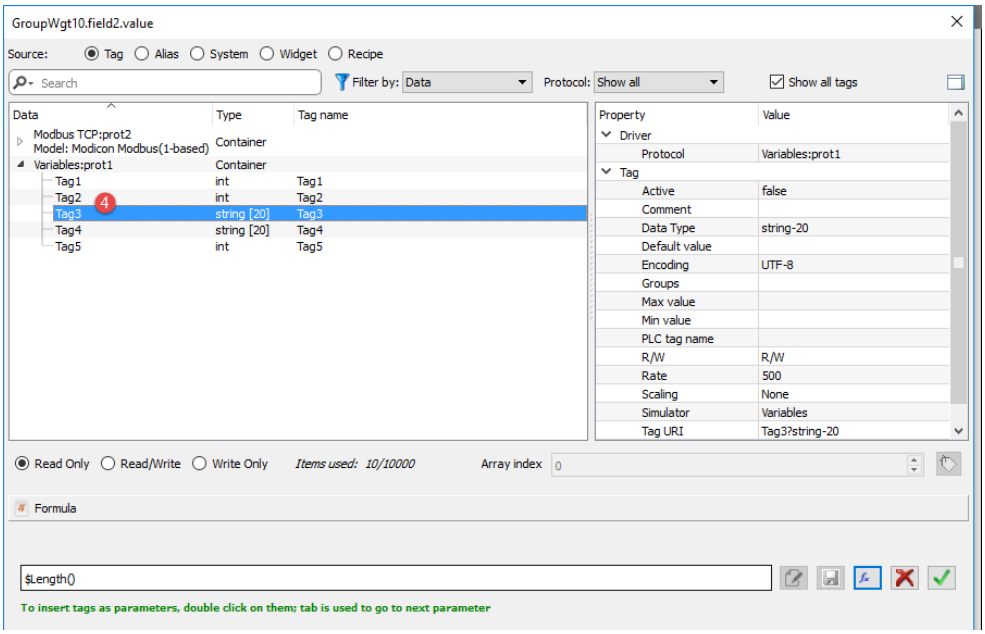
Opérateurs de comparaison	
<	Supérieur à
<=	Inférieur ou égal à
>	Inférieur à
>=	Supérieur ou égal à
==	Égal à
!=	Non égal à

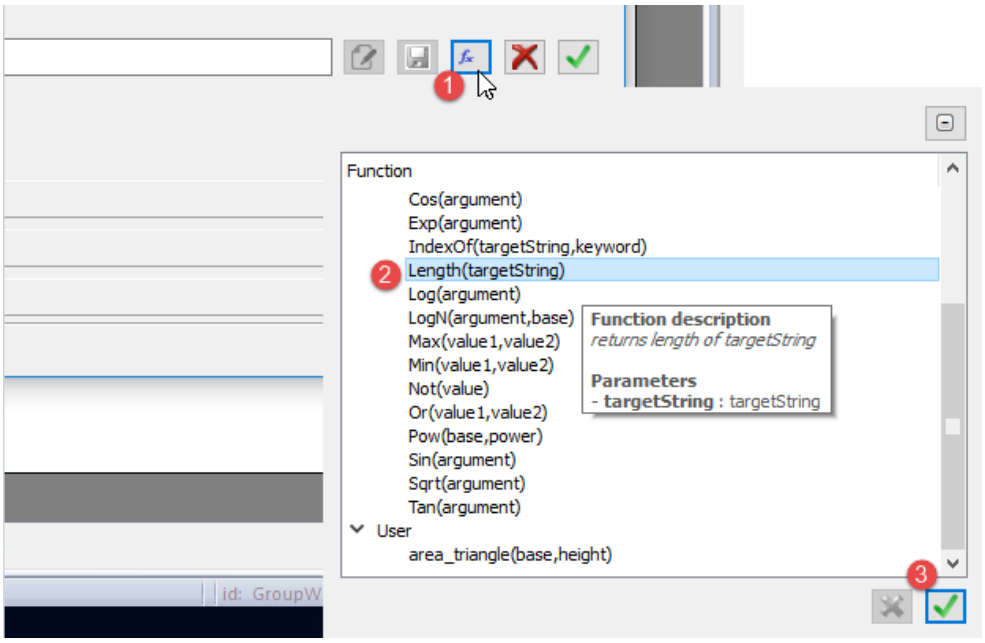
Utiliser une formule prédéfinie de la bibliothèque

Pour insérer un projet ou une formule prédéfinie :

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

1. Ouvrez la bibliothèque de formules
2. Sélectionnez la formule à utiliser
3. Confirmez la formule sélectionnée
4. Saisissez les arguments requis à partir de la formule sélectionnée

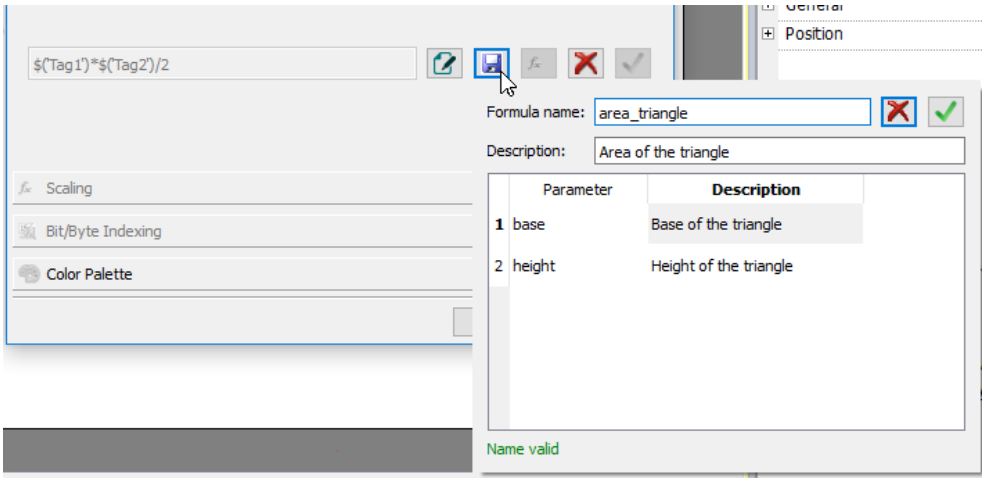




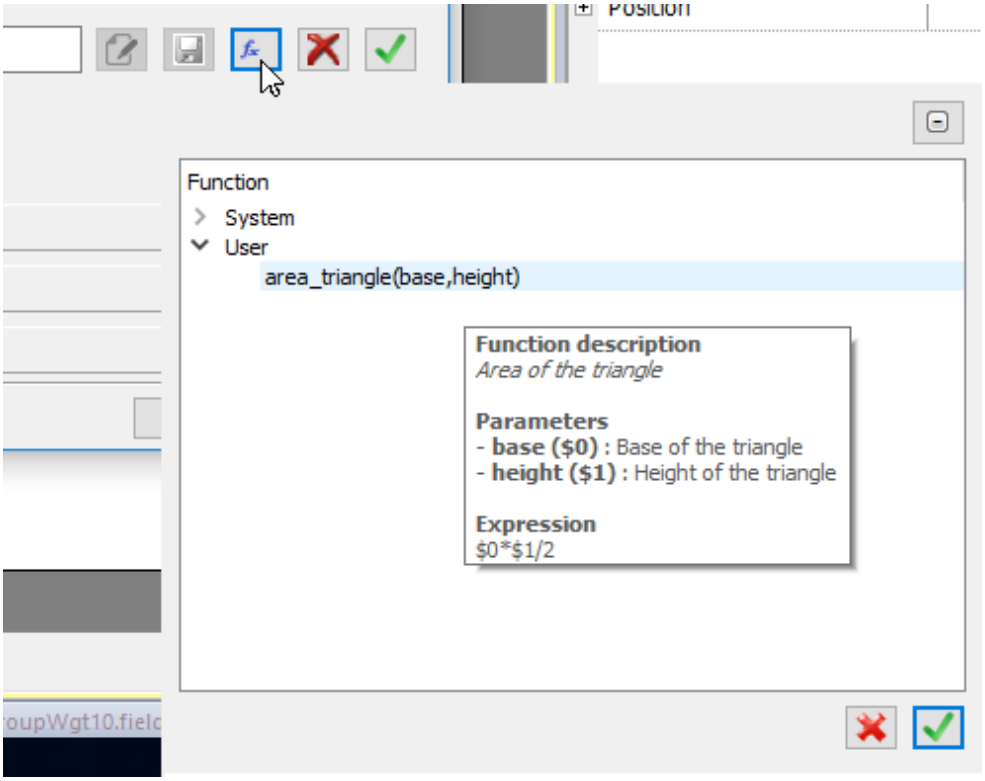
Ajouter des formules utilisateur à la bibliothèque

Après avoir saisi une nouvelle formule, le bouton SAVE permet de stocker la nouvelle formule à l'intérieur du dossier projet pour la rendre disponible à partir de la bibliothèque des formules.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



Une formule utilisateur peut être extraite de la bibliothèque des formules comme pour les autres formules prédéfinies.



Événements

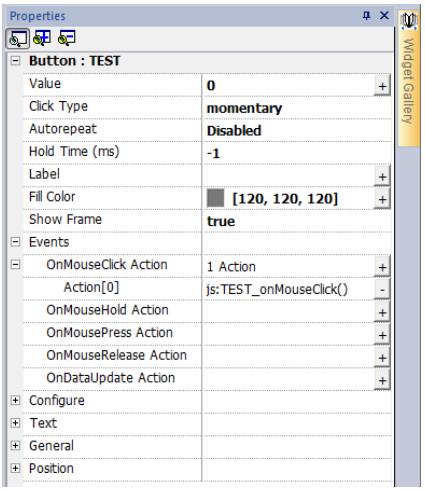
Les événements sont utilisés pour déclencher des actions au niveau du projet et ils peuvent être associés à :

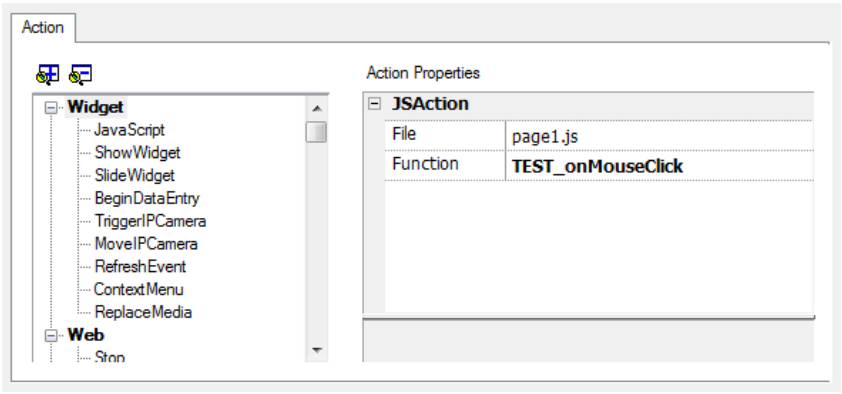
Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

- des boutons/touches (cliquer, appuyer, relâcher)
- Molette de souris
- des périphériques d'entrée externes tels que des claviers et une souris (cliquer, appuyer, maintenir, relâcher, molette)
- des modifications de données (LorsdeMiseàJour)
- des changements de pages (Lorsd'Activation, LorsdeDésactivation)
- Alarmes
- Planificateur

Vous pouvez associer une ou plusieurs actions à un événement, pour qu'elles soient exécutées lorsque l'événement se produit.

Dans cet exemple, une action JavaScript est activée en appuyant sur un bouton.

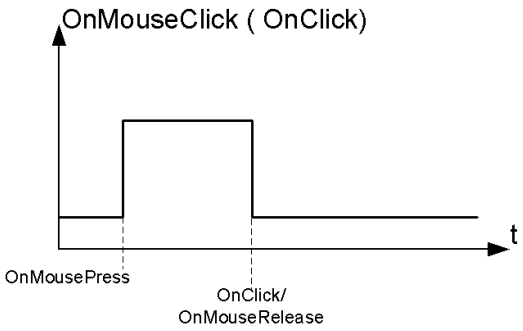




LorsdeClic/LorsdeClicSouris

Déclenche l'événement lorsque vous appuyez sur le bouton/la touche et relâchez rapidement.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

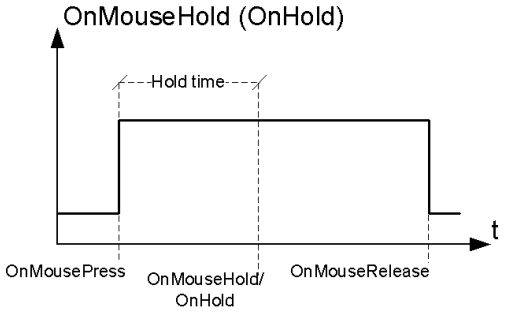


LorsdeMaintienBouton/LorsdeMaintienBoutonSouris

Déclenche l'événement lorsque vous appuyez et maintenez enfoncé le bouton/la touche pendant un certain moment, qui est défini comme **Temps de maintien** dans les propriétés du widget. Les actions programmées pour cet événement sont exécutées uniquement après expiration du temps de maintien.

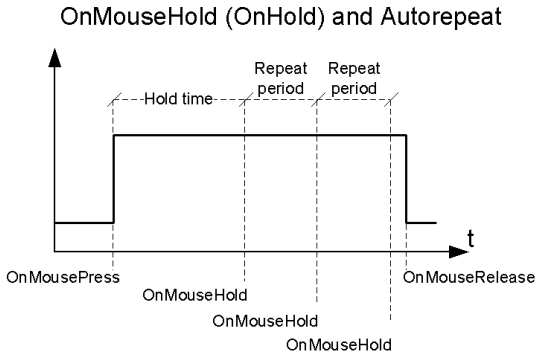
Le **Temps de maintien** par défaut est configuré dans les propriétés du projet mais il peut être redéfini pour chaque bouton/touche. Voir "Propriétés de projet" Page 69.

Remarque : le **Temps de maintien** est défini sur -1 pour le widget, la valeur **Temps de maintien** du projet sera utilisée.

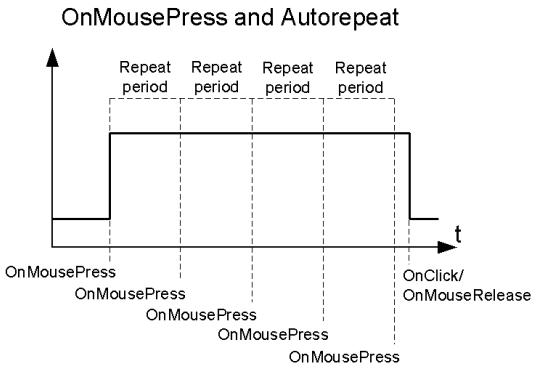


Répétition auto

Active la répétition auto pour un événement d'appui ou de maintien d'un bouton ou d'une touche. Le **Temps de répétition auto** est spécifié dans les propriétés du projet mais il peut également être redéfini pour chaque bouton/touche.



Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



LorsdeMolette

Déclenche l'événement lorsque la valeur d'une molette (par exemple une molette de souris USB) est modifiée. Une molette est en général utilisée pour augmenter/diminuer les valeurs dans une zone de texte ou elle est associée à une variable.

Lorsd'Activation

Déclenche l'événement lorsqu'une page est chargée. L'événement démarre avant que les widgets de la page ne soient initialisés.

LorsdeMiseàJour

Déclenche l'événement de mise à jour lorsque la valeur de la variable change. Par exemple, l'événement **LorsdeMiseàJour** peut être déclenché ou non, selon si le protocole met à disposition les données respectivement après ou avant l'initialisation des widgets pour la première fois. En particulier, il est plus probable que les notifications de modification d'une page s'affichent avec des protocoles lents et des clients à distance.

Remarque : la valeur lue pendant **Lorsd'Activation** peut être similaire à celle obtenue par un événement **LorsdeMiseàJour** postérieur, puisque les notifications **LorsdeMiseàJour** sont envoyées de manière asynchrone.

Positionnement des widgets

Vous pouvez placer des widgets sur la page en utilisant deux méthodes :

- Aligner à la grille
- Aligner à l'objet

Pour afficher la grille, sur le menu **Vue**, cliquez **Afficher grille**.

Aligner à la grille

*Chemin : **Vue**> **Aligner à la grille***

Lorsque vous déplacez ou redimensionnez un objet, son coin supérieur gauche s'alignera à l'intersection la plus proche des lignes dans la grille, même si la grille n'est pas visible.

Paramètres des propriétés de la grille

*Chemin : **Vue**> **Propriétés***

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

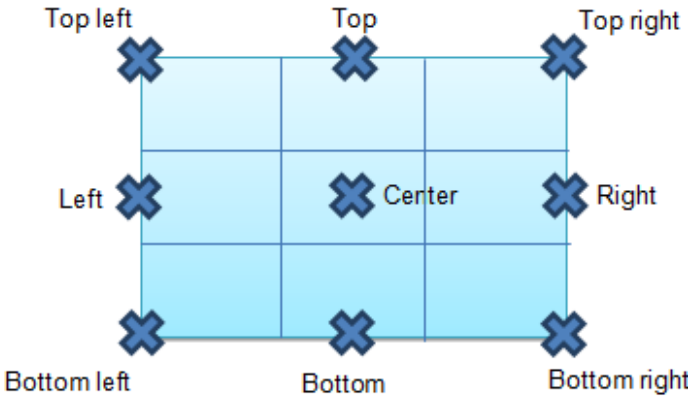
Paramètre	Description
Espacement X	Espace dans des pixels entre deux lignes/points sur l'axe X
Espacement Y	Espace dans des pixels entre deux lignes/points sur l'axe Y
Type	Type de grille (point ou ligne)
Couleur	Couleur de la grille

Aligner à l'objet

Chemin : ***Vue*** *>* ***Aligner à la grille***

Lorsque vous déplacez un objet, il s'alignera avec d'autres objets sur la page.

Lorsque vous sélectionnez un objet, un des points sensibles suivants est sélectionné comme source du point d'alignement, en fonction de la zone où vous avez appuyé: en haut, en haut à gauche, en haut à droite, en bas, en bas à gauche, en bas à droite, à gauche, à droite, au centre :

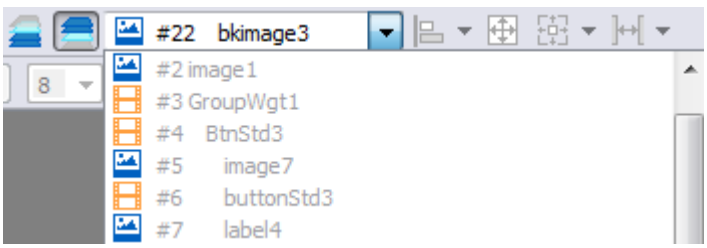


Un algorithme trouve un point sensible correspondant parmi les widgets à proximité de points chauds correspondant aussi bien aux coordonnées x que y du point d'alignement source. Pour les widgets en ligne, les points d'alignement source sont les points terminaux de la ligne.

Gestion widgets chevauchants

Lorsqu'un ou plusieurs widgets se chevauchent sur la page, vous pouvez décider de les organiser de sorte qu'un soit affiché au-dessus des autres.

L'ordre du widget sur la page est affiché dans la liste déroulante. Un widget avec un nombre d'ordre-z supérieur est face à un élément avec un nombre d'ordre-z inférieur. Une icône par image identifie les objets statiques, une icône par film identifie des objets dynamiques.



Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Important : un ordre correct de widgets est essentiel pour la performance au runtime puisque le chevauchement des widgets dynamiques peut rendre invalide l'optimisation statique et réduire les performances des applications IHM.

Cacher/afficher un widget sur ordre z

Pour cacher des widgets au-dessus d'un widget sélectionné :

- Sur la barre d'outils, cliquez sur et sélectionnez un widget : tous les widgets au-dessus de celui-ci seront cachés

Pour cacher des widgets en dessous d'un widget sélectionné :

- Sur la barre d'outils, cliquez sur et sélectionnez un widget : tous les widgets en dessous de celui-ci seront cachés

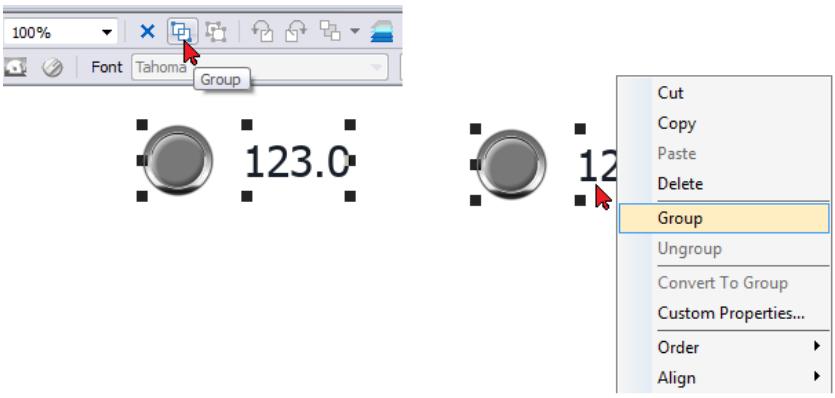
La barre d'outils permet de :

- cacher les widgets empilés au-dessus et/ou en dessous des widgets sélectionnés
- travailler sur différents widgets en utilisant la liste déroulantes qui liste tous les widgets dans leur ordre-z.

Grouper des widgets

Pour grouper des widgets :

1. Sélectionnez tous les widgets à grouper.
2. Faites un clic droit puis cliquez sur **Groupe**.



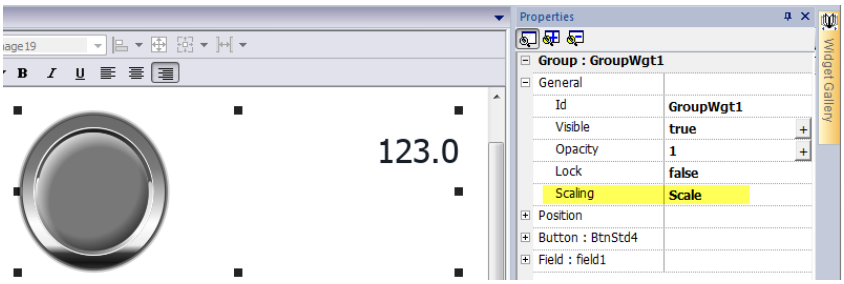
Conseil : double-cliquez pour accéder au mode édition du groupe. Les widgets de groupe ne peuvent être édités et sélectionnés qu'en mode groupe. Tous les autres widgets sont partiellement masqués

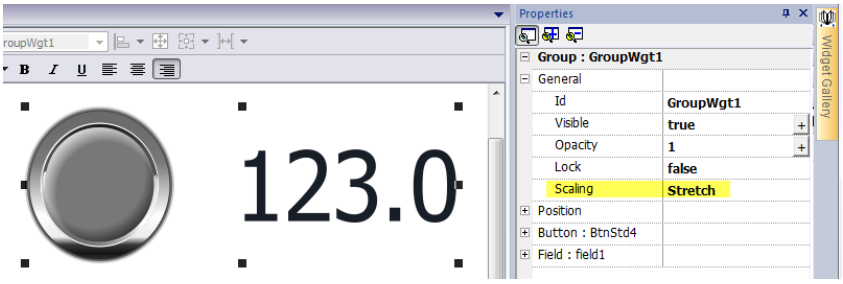
Redimensionner des widgets groupés

Vous pouvez définir les réactions au redimensionnement d'un objet. Utilisez la propriété **Mise à l'échelle** dans la section **Généralités**

- **Mettre à l'échelle** : l'objet et le texte ne sont pas redimensionnés proportionnellement
- **Étirer** : l'objet et le texte sont redimensionnés proportionnellement

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21





Groupe disposition grille

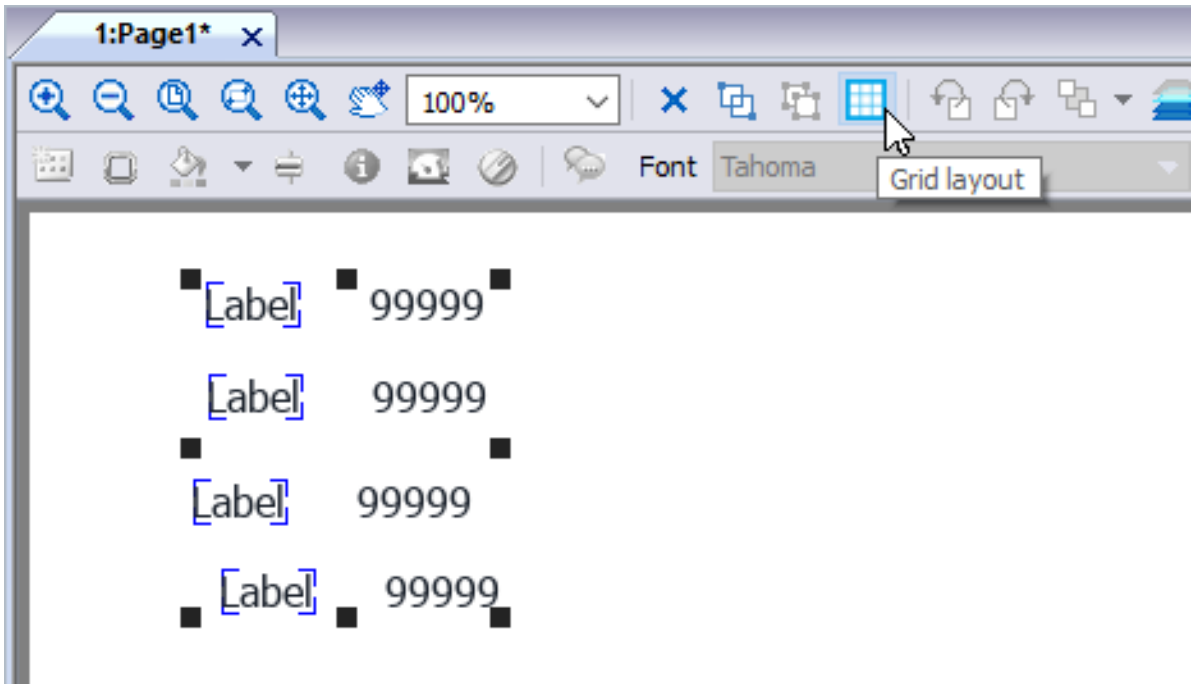
La disposition de la grille ajoute la possibilité de configurer les relations spatiales entre les widgets du groupe.

Pour créer une disposition de la grille :

- Activer le paramètre « *Disposition de grille* » du groupe de widgets.

ou

- Sélectionner les widgets qui seront à l'intérieur du tableau et cliquer sur le bouton « Disposition de la grille » sur la barre d'outils de la page. Les widgets sélectionnés seront alignés et recueillis dans un groupe avec la propriété de disposition de grille activée.

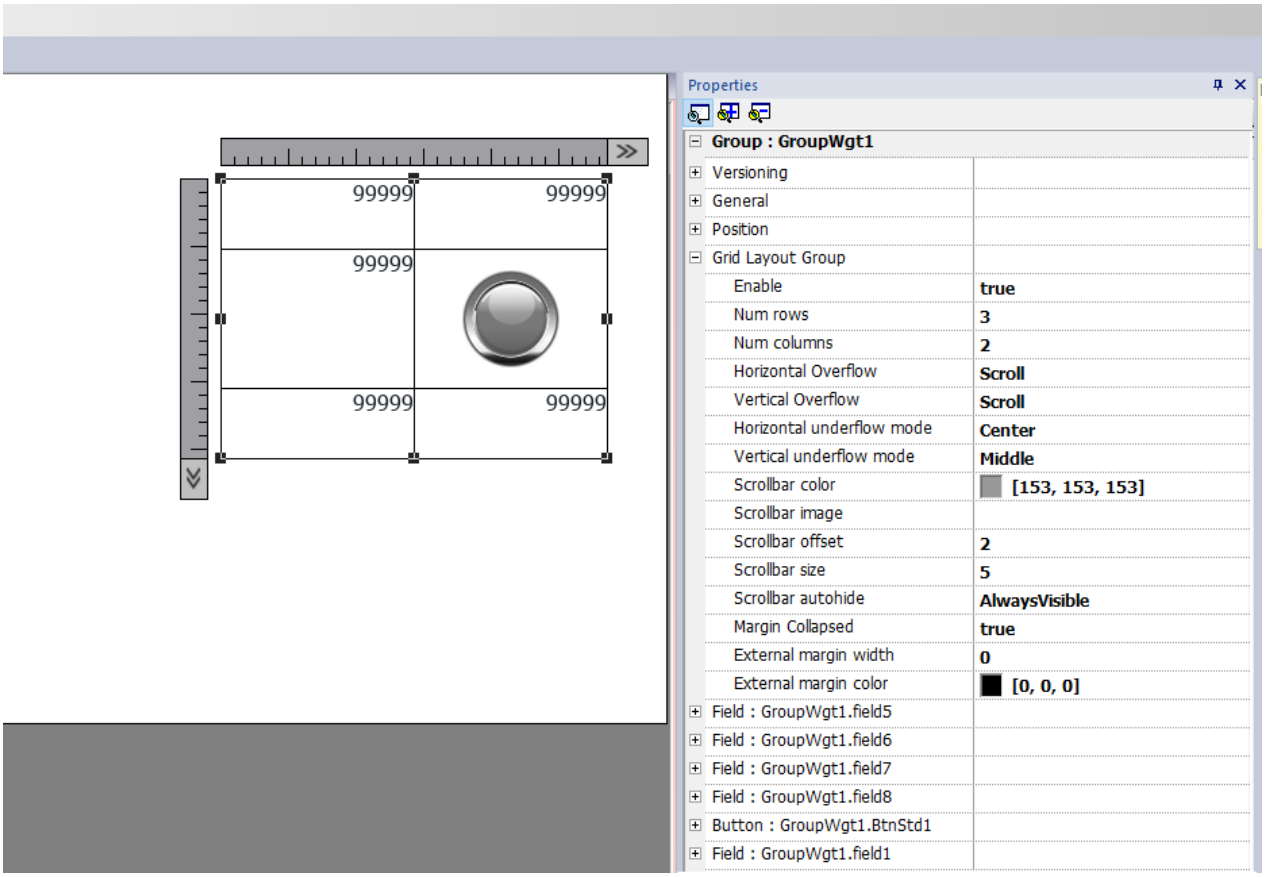


Il existe plusieurs éléments associés à la disposition de la grille qui peuvent être configurés :

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

- Propriétés de grille
- Propriétés de ligne, colonne
- Propriétés de cellules

Propriétés de grille



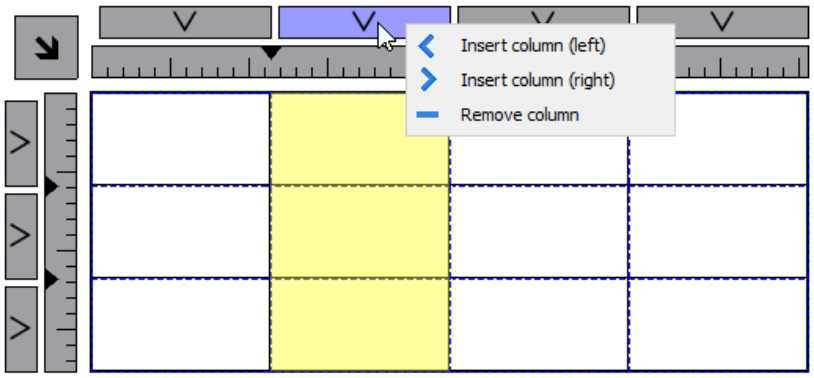
Paramètre	Description
Activer	Activer la disposition de grille. Une grille sera générée autour des widgets du groupe
Lignes num **Colonnes num**	Nombre de lignes et de colonnes des grilles Les lignes et les colonnes peuvent être enlevées uniquement si leurs cellules sont vides.
Débordement horizontal **Débordement vertical**	Ce paramètre permet de définir le comportement de la grille quand il est trop petit pour contenir toutes les lignes et les colonnes. • Caché Les lignes et les colonnes qui ne rentrent pas dans la grille ne sont pas affichées

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Paramètre	Description
	• Visible La grille ne peut pas être rendue plus petite que la taille minimale requise pour contenir toutes les lignes et les colonnes définies • Défilement Lorsque la grille est trop petite pour contenir toutes les lignes et les colonnes définies, les barres de défilement peuvent être utilisées pour déplacer le contenu de la grille.
Sous-débordement horizontal **Sous-débordement vertical**	Ce paramètre définit le comportement de la grille quand il est trop grand que la taille définie pour les lignes et les colonnes. • Bloqué La grille ne peut pas être rendue plus grande que la taille maximale des lignes et des colonnes • Gauche, Centre, Droite - Haut, Milieu, Bas Permet de définir la position des widgets lorsque les cellules sont plus grandes que les tailles maximum définies
Couleur de barre de défilement **Image de barre de défilement** **Décalage de barre de défilement** **Taille de barre de défilement** **Cache auto de barre de défilement**	Paramètre pour définir l'apparence et la position des barres de défilement
Marge réduite	Réduire toute la marge de gauche à droite et de haut en bas en utilisant les paramètres de la course avec une plus grande largeur.
Largeur de la marge externe **Couleur de la marge externe**	Paramètres de la marge externe

Ajouter ou supprimer des lignes ou des colonnes

Pour ajouter ou supprimer des lignes ou des colonnes, double-cliquez sur la grille pour entrer en mode édition et faites un clic droit de souris sur le sélecteur de lignes et de colonnes pour ouvrir le menu contextuel.



Fusionner ou séparer des lignes ou des colonnes

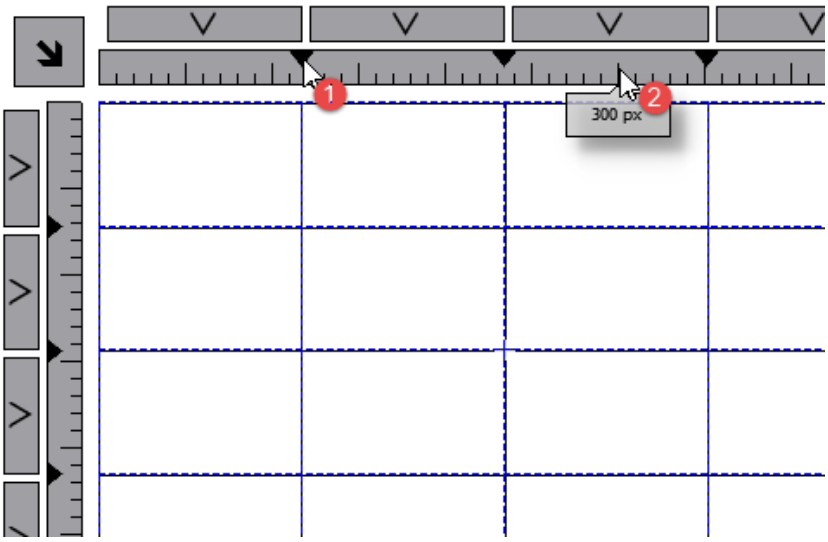
Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Pour fusionner ou séparer des lignes ou des colonnes, double-cliquez sur la grille pour entrer en mode d'édition et déplacez le curseur sur les rubans :

- Double-cliquer sur le triangle noir pour fusionner les deux lignes ou colonnes adjacentes (1)

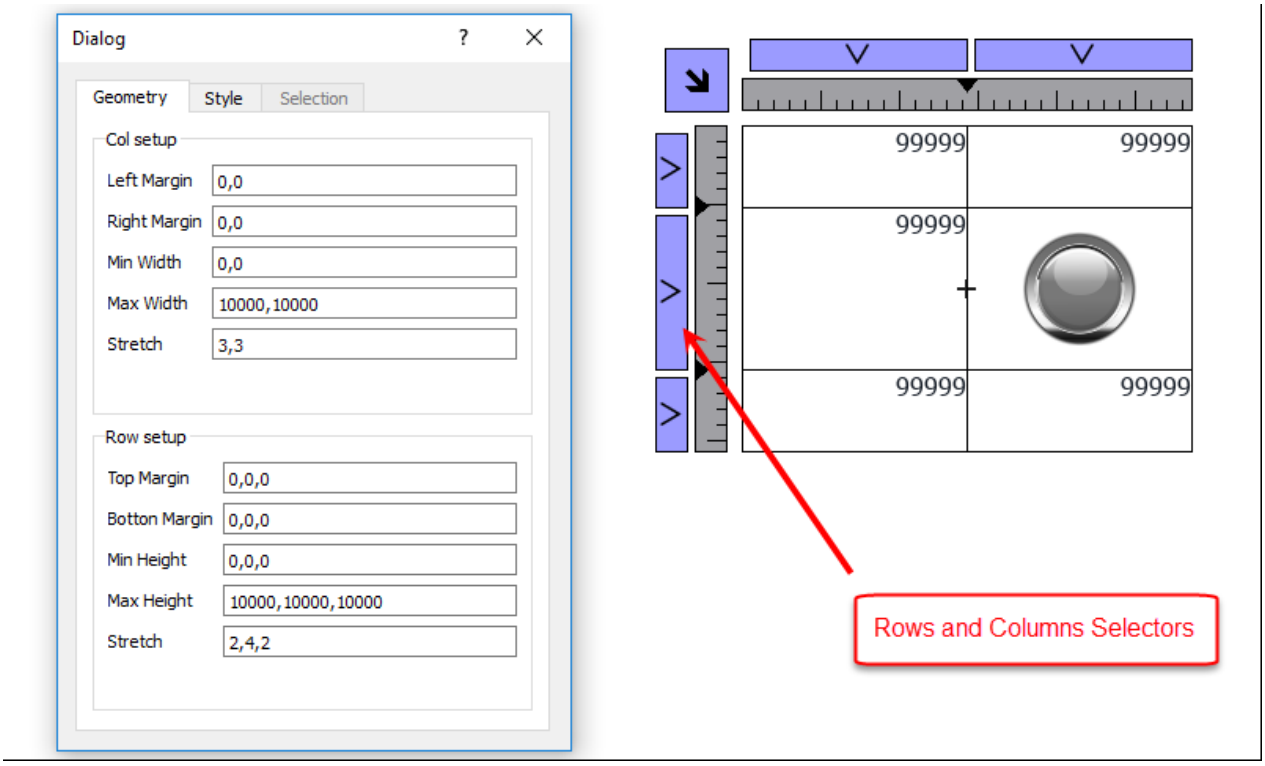
  Veuillez noter que la fusion n'est possible qu'avec des lignes ou colonnes vides.

- Double-cliquer sur le ruban pour diviser la ligne ou colonne sélectionnée (2)



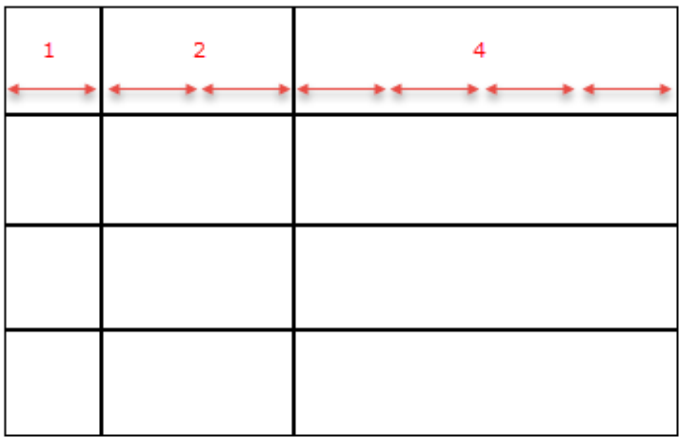
Propriétés de ligne, colonne

Les propriétés de ligne et de colonnes sont disponibles dans une boîte de dialogue pop-up après avoir cliquer sur les sélecteurs de ligne et de colonnes qui sont visibles par une double-clic sur le groupe de widgets.

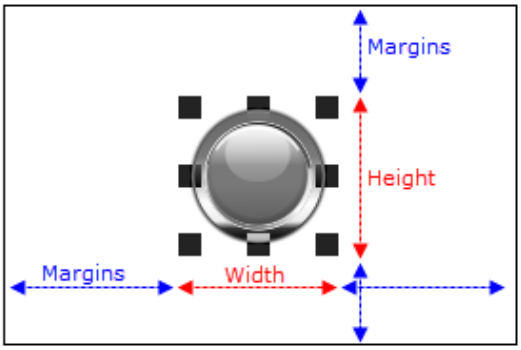


Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Stretch



Margins



Paramètres géométriques

Paramètre	Description
Marge de gauche **Marge de droite**	Distance du widget de le bord de la cellule
Largeur min **Largeur max**	Largeur min/max que le widget peut prendre lorsque la cellule est étirée
Étirement	Définit la relation entre les largeurs des colonnes qui seront maintenues si la grille est étirée
Marge supérieure **Marge inférieure**	Distance du widget de le bord de la cellule
Hauteur min **Hauteur max**	Hauteur min/max que le widget peut prendre lorsque la cellule est étirée
Étirement	Définit la relation entre les hauteurs des lignes qui seront maintenues si la grille est étirée

Paramètres de style

Paramètre	Description
Largeur de course gauche **Largeur de course droite** **Largeur de course haute** **Largeur de course basse**	Largeur des courses
Couleur de course gauche **Couleur de course gauche** **Couleur de course haute** **Couleur de course Basse**	Couleurs des courses
Couleur de fond	Couleur d'arrière-plan de ligne.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

La liste des valeurs séparées par une virgule renvoie aux lignes et colonnes. Exemple, la première valeur est pour la ligne 0, la deuxième valeur pour la ligne 1, et ainsi de suite.

Le format de couleur pourrait être #rrggbb ou #rrggbbaa, où « *aa* » est la valeur alpha qui définit l'opacité de la couleur.

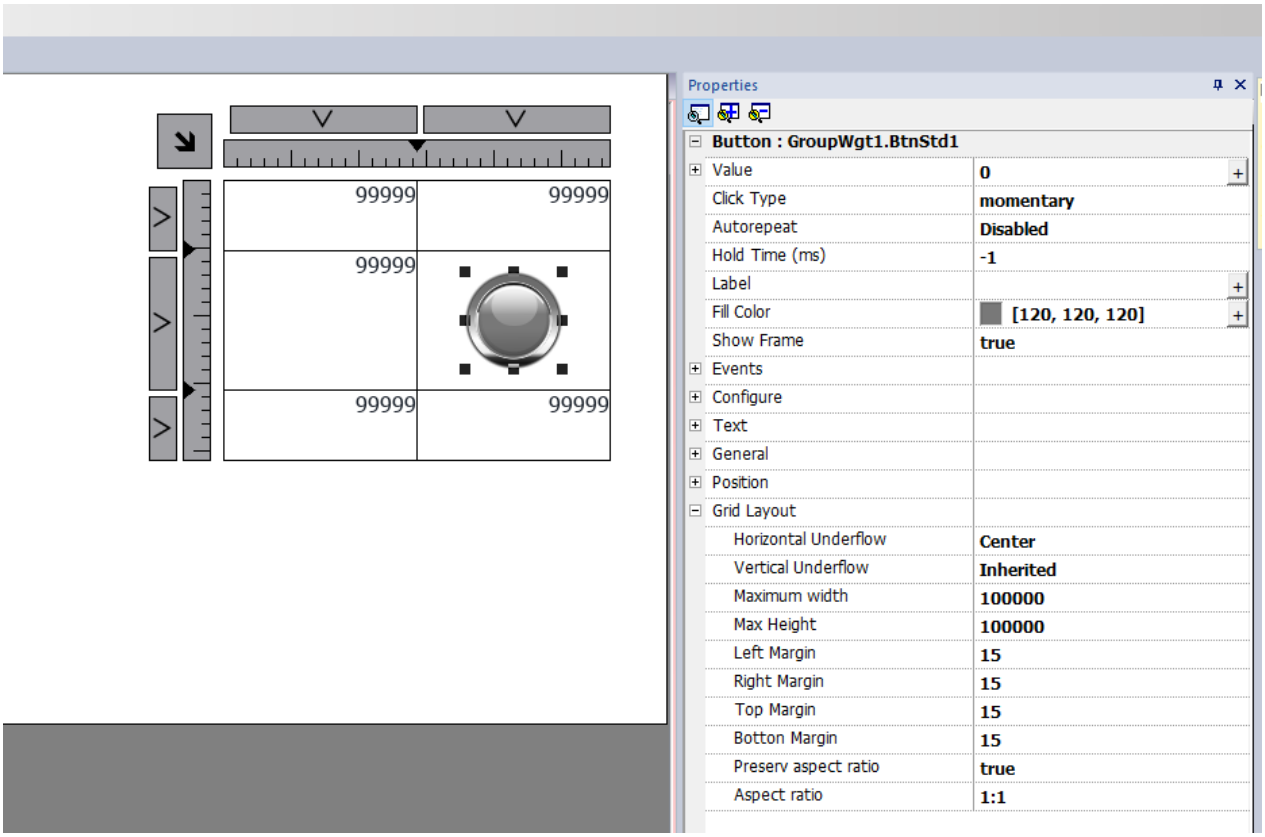
Paramètres de sélection

Les paramètres de sélection sont disponibles uniquement lorsque la grille est utilisée à l'intérieur d'un widget de tableau (voir "Widget tableau" Page 393 pour plus de détails).

Paramètre	Description
Couleur de premier plan **Couleur de fond** **Couleurs de course**	Couleurs que la ligne prend lorsqu'elle est sélectionnée La liste des couleurs est associée à des modèles de ligne. La première couleur est pour le modèle de la ligne 0, la deuxième couleur est pour le modèle de la ligne 1, et ainsi de suite.

Propriétés de cellules

Les propriétés d'une cellule unique sont disponibles à l'intérieur de la vue des propriétés lorsqu'une cellule est sélectionnée. Pour sélectionner une cellule : tout d'abord, double-cliquer sur le groupe de widget, puis cliquer sur la cellule pour la sélectionner.



Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

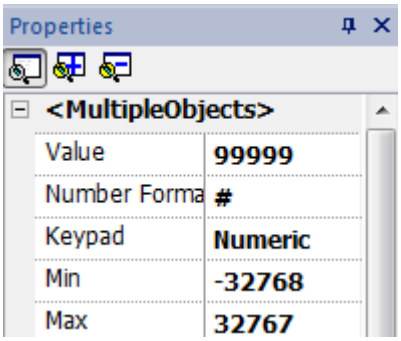
Paramètre	Description
Sous-débordement horizontal **Sous-débordement vertical**	Ce paramètre définit le comportement du widget quand il est trop grand que la taille définie pour le widget. • Héritage Il hérite de la valeur utilisée pour la ligne ou la colonne • Gauche, Centre, Droite - Haut, Milieu, Bas Permet de définir la position des widgets lorsque les cellules sont plus grandes que les tailles maximum définies
Largeur max **Hauteur max**	Écraser les paramètres globaux de la grille
Marge de gauche **Marge de droite** **Marge supérieure** **Marge inférieure**	Écraser les paramètres globaux de la grille Pixels supplémentaires qui sont ajoutés à la marge totale.
Conserver le rapport d'aspect	Conserver le rapport d'aspect du widget
Rapport d'aspect	Disponible uniquement lorsque « *Conserver le rapport d'aspect* » est vrai

Changer les propriétés de plusieurs widgets

Vous pouvez définir les propriétés de plusieurs widgets du même type, tous en une seule fois.

Pour modifier les propriétés :

1. Sélectionnez les widgets.
2. Définissez les propriétés communes dans le volet **Propriétés**.
3. Lorsque plusieurs widgets sont sélectionnés, le titre du volet Propriétés devient **<MultipleObjects>** : tous les changements seront appliqués à tous les widgets sélectionnés.



Remarque : toutes les propriétés ne peuvent pas être modifiées pour plusieurs widgets simultanément, elles doivent donc être modifiées individuellement.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

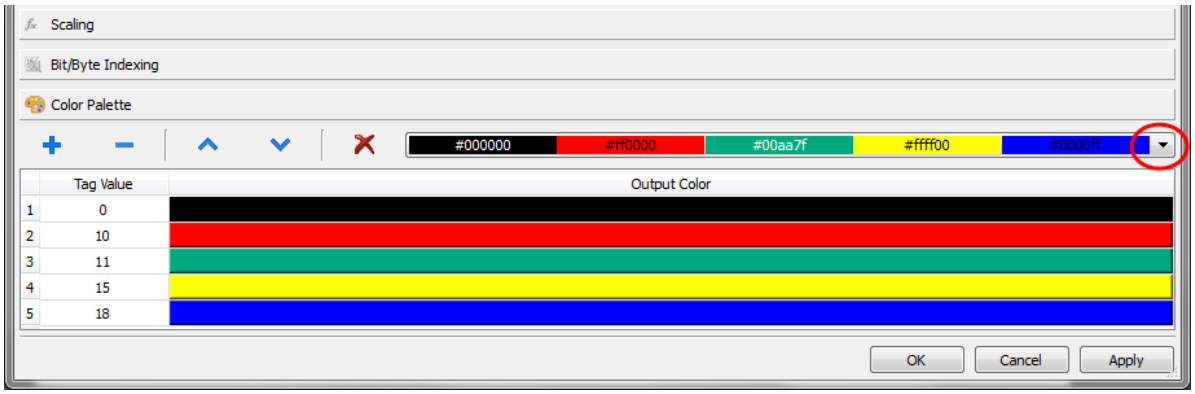
Changer la propriété couleur de remplissage en fonction des valeurs de variable

PB610 Panel Builder 600 permet de modifier la propriété de couleur d'un widget de façon dynamique en fonction des valeurs de variable, ce de deux façons :

- En utilisant ColorPalette
- En associant la propriété de couleur à une variable de type Chaîne

Changer la propriété de couleur en utilisant ColorPalette

1. Créez la variable (interne ou API) de référence pour la gestion des couleurs. La variable peut être de tout type de données. La couleur changera en fonction de la valeur de cette variable.
2. Associez cette variable à la propriété **Couleur de remplissage** d'un objet (par exemple, un bouton).
3. Dans la même boîte de dialogue, sélectionnez l'onglet **Colorpalette** et ajoutez les couleurs à utiliser pour l'objet selon la valeur de la variable.



Remarque : les derniers tableaux de couleurs utilisés sont sauvegardés et peuvent être réutilisés en les sélectionnant à partir de la liste des couleurs dans la barre d'outils.

Changer la propriété de couleur en l'associant à une variable de type Chaîne

1. Créez la variable (interne ou API) de référence pour la gestion des couleurs. La couleur changera en fonction de la valeur de cette variable. La variable doit être de type Chaîne et la propriété **Taille de tableau** de la variable doit être suffisamment grande pour contenir la chaîne formatée comme illustré ici.
2. Associez cette variable à la propriété **Couleur de remplissage** d'un objet (par exemple, un bouton).
3. Écrivez dans la variable de **Chaîne** le code couleur RVB de la couleur souhaitée. Utilisez un des ces formats :

- **#XXYYZZ**, où XX, YY et ZZ sont des composants RVB de la couleur souhaitée exprimée sous forme Hexadécimale (plage 00–FF).
- **rgb(XXX,YYY,ZZZ)**, où XXX, YYY et ZZZ sont les composants RVB des couleurs souhaitées exprimées sous forme Décimale (plage 0–255).

Remarque : cette fonctionnalité peut s'appliquer à tous les objets disponibles dans la Galerie des widgets ayant une propriété de couleur. Le changement de couleur au runtime n'est possible que grâce aux propriétés des SVG

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

dont se compose l'objet. Cette fonctionnalité ne s'applique pas à d'autres formats d'image tels que les fichiers JPEG ou BMP.

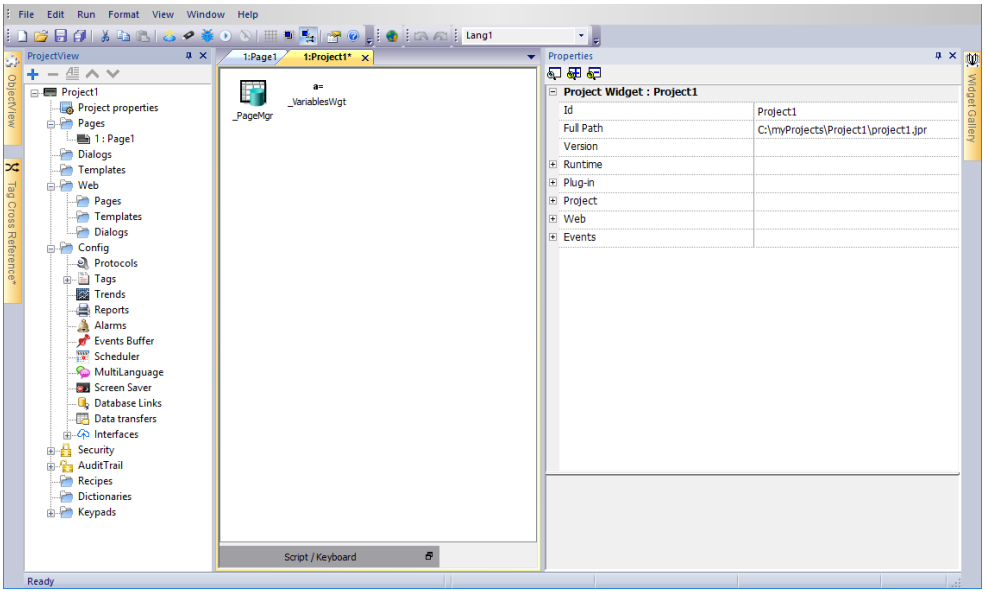
Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

6 Propriétés de projet

Les propriétés de projet contiennent les paramètres pour le projet.

*Chemin : **VueProjet** > double-cliquez sur **Propriétés de projet**> volet **Propriétés***

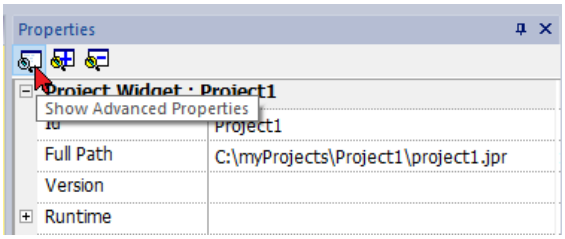
Le volet **Propriétés** de projet contient une liste de données configurables par l'utilisateur au niveau du projet.



Propriétés de base et avancées

Certaines propriétés ne s'affichent qu'en mode avancé. Pour consulter toutes les propriétés de projet :

- Cliquez sur le bouton **Afficher les propriétés avancées** pour agrandir la zone de visualisation du volet **Propriétés**.



Propriétés disponibles

Propriété	Description
ID	Nom de projet (lecture seule)
Chemin complet	Chemin de projet (lecture seule)
Version	Le champ Version permet aux utilisateurs d'indiquer la version du projet.
+ Runtime	Propriétés liées au runtime de l'application. Voir "Runtime" Page suivantepour les détails.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Propriété	Description
+ Plug-in	Modules optionnels. Voir "Module d'exécution (plug-in)" Page 75pour les détails.
+ Projet	Propriétés liées au projet. Voir "Projet" Page 76 pour les détails.
+ Web	Propriétés liées à l'interface web. Voir "Web" Page 80pour les détails.
+ Événements	Événements globaux. Voir "Événements" Page 81 pour les détails.

Runtime

Propriété	Description
Menu contextuel	Définissez les modalités d'affichage du menu contextuel dans le projet IHM. **retard** = le menu contextuel apparaît en touchant/appuyant et maintenant pendant quelques secondes un espace vide de l'écran du runtime, ou via une action du **Menu contextuel** **action** = le menu contextuel n'apparaît que via une action du **Menu contextuel.** Voir "Actions widget" Page 182 pour plus de détails.
Outil de développement	Active/désactive une série d'outils de débogage de runtime.
Signal sonore au contact	Activer le bipeur lorsqu'un widget est touché sur l'écran du périphérique IHM. Widgets pris en charge : • boutons • zones réactives • aiguilles • champs • touches externes • listes déroulantes • éléments de tableaux • éléments de la liste de contrôle *Sur les panneaux de Windows CE, disponibles pour à partir de v1.76 ARM / 2.79 MIPS.*
Durée du signal sonore	200 ms par défaut
Clavier	Permet d'utiliser les macros de clavier au runtime à condition d'utiliser des claviers externes.
Débogage JavaScript	Active le débogueur JavaScript au runtime pour le projet actuel.
Autoriser débogueur à distance JS	Active le débogueur distant JavaScript pour le projet actuel. Débogage distant non supporté sur HMI Client.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Propriété	Description
Activation image base de données	Active un moteur utilisé par le runtime pour optimiser la performance du projet. **AVERTISSEMENT : cette propriété ne doit être désactivée par l'assistance technique que dans un but de débogage, car cette opération pourrait diminuer la performance au runtime.**
Rendu des polices CaractèresLibres	Passe à CaractèresLibres, le rendu de police utilisé par PB610 Panel Builder 600et le runtime.
Retard des icônes de communication (ms)	Retard avant l'affichage de l'icône d'erreur de communication (par défaut 0 ms) La valeur spéciale -1 est toujours désactivée
Démarrage rapide	Lorsque le démarrage rapide est activé et que l'interface utilisateur est démarrée avant le serveur en arrière-plan. • Par défaut : l'interface utilisateur est chargée une fois que le serveur en arrière-plan est prêt à être utilisé. • IU rapide : l'interface utilisateur est chargée avant de charger le serveur en arrière-plan.

Démarrage rapide

Lorsque le démarrage rapide est activé, le périphérique IHM affichera l'écran d'accueil aussi rapidement que possible après la mise sous tension. Dans ce mode, seules les fonctions minimales nécessaires sont chargées avant le démarrage de l'interface utilisateur. Le chargement des protocoles, des événements, des tendances, des alarmes, des actions sont différés après le chargement de l'interface utilisateur.

Il y a deux drapeaux à définir :

- Le drapeau « Démarrage rapide » disponible à l'intérieur des propriétés avancées du projet
- Le drapeau « Démarrage rapide » disponible dans la page Services de l'outil Paramètres du système BSP (voir "Paramètres du système" Page 481)

Lorsque le démarrage rapide est activé et que l'interface utilisateur est démarrée avant le serveur en arrière-plan, l'événement Javascript project.onServerReady peut être utilisé pour synchroniser le serveur.

Exemple :

```
if (!project.serverIsReady) {
	// Set the callback to wait for server ready
	project.onServerReady = onServerReady;
} else {
	// Server is ready, call it now
	onServerReady();
}

function onServerReady()
{
	project.setTag("Tag1", 1);
	project.showMessage("Server is ready, tags can be used: " + project.getTag("Tag1")
```

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

```
)
}
```

Il s'agit d'une fonctionnalité avancée disponible uniquement sur les plateformes Linux

Outils du développeur

Collection de fonctions de débogage au runtime qui peuvent être activées ou désactivées.

1. Dans le volet **Propriétés**, définissez **Outils de développement** sur **Vrai**.
2. Téléchargez le projet.
3. Ouvrez le menu contextuel.
4. Sélectionnez **Outils de développement**.

Liste des outils du développeur

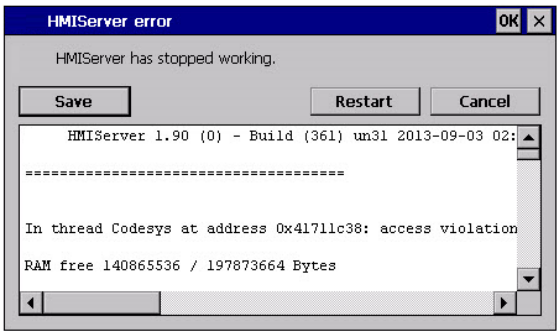
Outil	Description
Afficher/Masquer tout	Affiche une boîte de dialogue contenant des informations sur l'état du périphérique, telles que la charge de l'UC, l'utilisation de la mémoire et les queues d'événements.
Statistiques CPU	Affiche des informations sur la charge de l'UC. Voir "Statistiques CPU" Page opposée.
Statistiques mémoire	Affiche les informations sur la RAM du système. Une valeur négative indique que la mémoire libre diminue.
Queues d'événements	Affiche les informations sur les queues d'événements (taille, taille maximale atteinte, nombre d'événements traités, dernier temps de traitement et temps de traitement maximal). Les statistiques sur le minutage sont disponibles uniquement pour les queues non IU.
Résumé journal en temps réel	Affiche le temps de chargement de la page.
Intégrer la fenêtre	Permet d'intégrer la scène en cours de runtime ou de laisser la fenêtre d'outil de développement comme une fenêtre autonome (dialogue).
Réinitialiser les statistiques de queue	Réinitialise les informations statistiques sur les queues d'événements.
Désactiver le chien de garde	Désactive la fonction de chien de garde et évite que le système ne redémarre en cas de gel ou d'incident des services.
Ignorer les exceptions	Désactive la fonction de rapport d'incident, les exceptions ne sont pas sauvegardées dans la fenêtre de rapport d'incident.
Lancer VNC	Lance le serveur VNC s'il est disponible en runtime. Le serveur VNC est disponible uniquement comme plug-in pour le runtime de Windows CE.
Profilage	Mesure le temps passé pour charger/afficher la page active. Voir "Profilage" Page opposée

Watchdog (Surveillance)

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

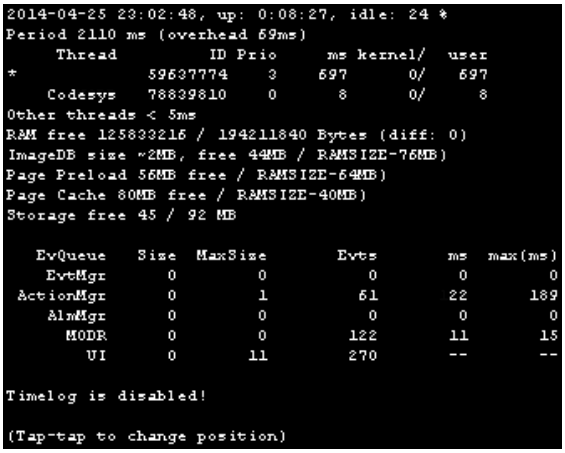
Cette fonction vous permet de désactiver l'horloge de surveillance. Ainsi, vous pouvez éviter que le système ne redémarre en cas d'incident en cours de runtime. Vous aurez donc le temps de sauvegarder le rapport d'incident ou de vérifier les informations sur l'état du système (par exemple, la mémoire disponible, la charge de l'UC, la taille des queues d'événements, etc.).

La boîte de dialogue de rapport d'incident s'affiche automatiquement lorsque le système gèle ou bloque, permettant aux utilisateurs de sauvegarder un fichier journal d'incident.



Important : sauvegardez ce fichier pour le support technique.

Statistiques CPU



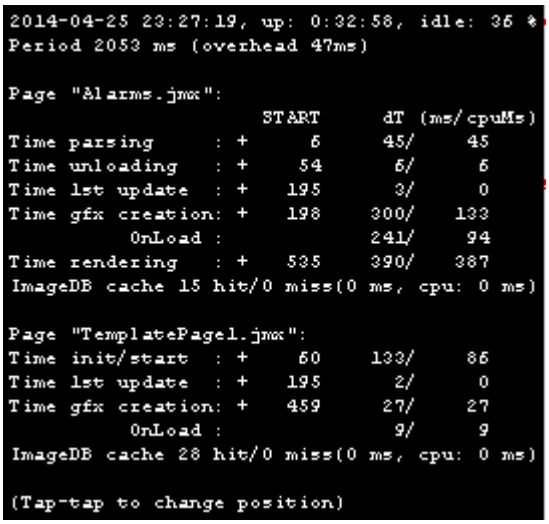
L'heure actuelle de la machine s'affiche sur la ligne supérieure, ainsi que le temps de service total du périphérique.

Les statistiques de l'UC sont collectées toutes les 2000 millisecondes. La période réelle et le traitement requis pour collecter et visualiser les statistiques sont également affichés. Au plus la période réelle s'éloigne de 2000 millisecondes, au plus la charge du système est importante. La consommation en fils de l'UC est listée en rapportant le nom du thread (s'il est disponible, le fil principal est marqué avec un *), l'ID du fil, la priorité du thread et le temps de l'UC passé durant la période de 2000 millisecondes, divisé en temps d'utilisateur et temps du noyau.

Profilage

Le profilage permet de vérifier le temps passé pour charger/afficher la page active. Le profilage démarre à partir du chargement de la page suivante et est actif pour le premier traçage de la page sur l'écran (la configuration est conservée).

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



Option de profilage	Description
Activer journal en temps réel	Active la capture du journal en temps réel. Le minutage est affiché dans la fenêtre « Résumé du journal en temps réel ».
Sauvegarder journal en temps réel sous forme de fichier	Sauvegarde un rapport de détails du profil et le temps passé à charger un projet et ses pages dans un fichier timelog.txt. Ce fichier peut être exporté et partagé pour une analyse approfondie. **Important : l'exécution de cette fonction peut réduire la performance de modification de la page.**
Temps de chargement superposition **Temps de rendu superposition**	Cette vue permet d'afficher le temps passé sur les widgets individuellement et il est disponible uniquement pour les étapes de rendu et de chargement. Cette vue donne un aperçu immédiat d'où le temps est passé. Les zones en rouge représentent les zones critiques qui prennent le plus de temps. Les temps détaillés par widget sont visualisés par une fenêtre en infobulle (sur la plateforme Win32 ils sont associés à l'événement passer la souris, sur Windows CE, appuyez sur la zone d'intérêt, glissez puis relâchez). Dans le cas de widgets hors champ, certaines flèches permettent de naviguer vers ces zones et en les survolant, l'infobulle affichera le résumé de la zone.
Choisir la couleur de superposition	Choisissez la couleur de superposition à utiliser

Données du journal en temps réel

Données	Description
Temps d'analyse	Temps passé à analyser la page actuelle. Dépend de la complexité/nombre de widgets de la page.
Temps de création gfx	Temps passé pour le rendu de l'image. Principalement associé à la méthode *En chargement*.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Données	Description
Temps d'affichage	Temps passé pour afficher la page.
Temps de déchargement	Temps passé à décharger la page, lorsque la page actuelle dépend d'une autre page.

Les temps sont fournis par deux : durée totale d'exécution/temps de l'UC. La durée totale d'exécution est le temps absolu requis par cette partie, qui peut être plus importante que le temps réel de l'UC requis car les fils de priorité plus haute sont aussi en cours d'exécution (pour les protocoles d'instance). La colonne d'heure de début se réfère à l'heure de début de chargement de la page. Elle peut être utilisée pour suivre le temps réel requis pour charger une page, car les temps partiels se réfèrent uniquement aux fonctions au temps le plus critique et n'incluent pas d'autres temps qui contribuent souvent au temps total de manière significative.

Par exemple, la durée totale d'exécution réelle requise pour charger une page est l'heure de début de l'affichage (qui est la dernière étape) + la durée totale d'exécution de l'affichage.

Rendu des polices CaractèresLibres

De nouveaux projets utilisent le moteur de polices CaractèresLibres par défaut. Les projets créés avec les anciennes versions de PB610 Panel Builder 600 pourrait utiliser un moteur de police plus ancien également après la conversion du projet pour éviter tout problème de compatibilité ascendante.

Basculez vers CaractèresLibres chaque fois que vous en avez la possibilité afin d'avoir un meilleur rendu de page.

Une fois que vous avez basculé vers le nouveau rendu de police, sauvegardez le projet et vérifiez que tous les textes s'affichent correctement dans toutes les pages de projet.

Problèmes de rendu de police

Si vous basculez un projet créé avec l'ancien moteur de police vers le moteur de police CaractèresLibres, il est possible que vous rencontriez les problèmes suivants :

- le texte demande plus/moins de pixels pour le rendu, entraînant une modification de la mise en page du texte
- les widgets sont redimensionnés pour contenir du texte
- un meilleur rendu peut être obtenu en utilisant l'Anticrénelage (l'Anticrénelage est une propriété de widget texte)

Module d'exécution (plug-in)

Vous pouvez choisir quels composants logiciels sont téléchargés vers le runtime avec le projet. Le plug-in logiciel a été conçu pour réduire les besoins en mémoire pour le périphérique IHM dans les périphériques IHM où le stockage est limité. Cette fonctionnalité n'est pas supportée par la plateforme Win32.

Propriété	Description
KitWeb	Module requis par le widget NavigateurWeb
EditeurTexte	Module requis par le widget ÉditeurTexte
Lecteur PDF	Lecteur PDF
Serveur VNC	Serveur VNC

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Important : tous les modules de plug-in des logiciels ne sont pas compatibles avec les plate-forme de périphérique IHM.

Une fois activé, le plug-in de logiciel devient une partie du runtime. Utilisez PB610 Panel Builder 600 pour installer en utilisant l'une des procédures suivantes :

- installer le runtime/mettre à jour le runtime
- Paquet de mise à jour

Pour supprimer les plug-ins du runtime, utilisez l'une des fonctions suivantes en Mode système :

- format Flash
- restaure les paramètres d'usine.

Important : le système ne peut pas détecter automatiquement quels plug-ins de logiciel sont requis par le périphérique IHM, assurez-vous de tous les sélectionner dans les propriétés du projet.

Remarque : le support plug-in du logiciel a été conçu pour les périphériques IHM intégrés où le stockage est limité. Cette option n'est pas prise en charge par la plateforme Win32.

Projet

Ces propriétés définissent différents éléments du comportement d'une page.

Propriété	Description
Page d'accueil	La première page chargée au runtime (après la page de connexion si la sécurité est activée dans le projet). Lorsque la sécurité est activée, vous pouvez spécifier une page d'accueil différente pour chaque groupe d'utilisateurs. Voir "Gestion et mots de passe utilisateur" Page 283 pour les détails.
LargeurPage **HauteurPage**	Définit la taille par défaut en pixels d'une page IHM. Lors de la création du projet, la résolution d'affichage du modèle de périphérique IHM sélectionné est par défaut.
Mode d'affichage	Définit l'orientation du périphérique IHM.
Type de projet	Définit le type de périphérique IHM pour le projet. Selon le modèle, certaines fonctionnalités et propriétés du projet sont ajustées automatiquement. **À partir de la version v2, le HMI Runtime vérifie que le projet sélectionné correspond au modèle de périphérique IHM ; lorsque le type sélectionné ne correspond pas, le message suivant apparaît : « Convertissez le projet et téléchargez-le à nouveau »**
Mémoire panneau	Taille de la mémoire interne du panneau disponible.
RequêtePage **PageActuelle** **OptionsSync**	Voir le chapitre ci-dessous

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

<table>
<tr><th>Propriété</th><th>Description</th></tr>
<tr><td>Temps de maintien
Temps de répétition automatique</td><td>Définit les valeurs de temps de maintien et de temps de répétition auto pour les boutons et les claviers externes.

Remarque : ces propriétés peuvent être redéfinies pour chaque bouton ou touche dans le tableau des propriétés de leur widget.</td></tr>
<tr><td>Masquer le chargement du projet au démarrage</td><td>Lorsque masqué, l'image de démarrage reste à l'écran jusqu'à ce que l'application soit prête à être exécutée.</td></tr>
<tr><td>Facteur de zoom cible</td><td>Le facteur de zoom du périphérique IHM est appliqué lorsque le projet est chargé au runtime.
<table><tr><td>Plage</td><td>0.3–2.9</td></tr><tr><td>Valeur par défaut</td><td>1 = pas de zoom</td></tr></table></td></tr>
<tr><td>Options couleur de fond</td><td>Lorsque la page définie est plus petite que l'ensemble de la zone d'affichage, colorez la zone de la page qui n'est pas couverte (par exemple lorsque la page est dézoomée)
<table><tr><td>Aucun</td><td>Ancien mode, la couleur est blanche (par défaut)</td></tr><tr><td>Couleur sélectionnée</td><td>Couleur à utiliser</td></tr><tr><td>Couleur de fond de page</td><td>Ajuste automatiquement la couleur selon la couleur d'arrière-plan du modèle ou de la page</td></tr></table></td></tr>
<tr><td>Signature</td><td>Algorithme à utiliser pour la signature
• sha256
• sha1</td></tr>
<tr><td>Transit de mouvement activé</td><td>Active la possibilité de passer des événements gestuels aux widgets sous-jacents après un délai configurable. L'utilisateur doit maintenir le doigt appuyé puis exécuter le geste.</td></tr>
<tr><td>Délai de transit de geste (ms)</td><td>Lorsqu'activé, les événements de geste sont passés aux widgets sous-jacents après ce délai. (voir "Transit d'événements gestuels" Page 378 pour les détails)</td></tr>
<tr><td>Geste Tactile multipoint</td><td>Active les gestes tactiles multipoints
• faux
• vrai (défaut)

Cette propriété donne la possibilité de désactiver les gestes tactiles multipoints. Ceci pourrait être utile pour éviter les problèmes avec les vieux projets qui n'ont pas été conçus pour gérer les gestes tactiles multipoints.</td></tr>
<tr><td>Accès refusé</td><td>Lorsque l'utilisateur essaie d'utiliser un widget verrouillé par la configuration de sécurité en lecture seule (par exemple, un champ ou un bouton), une icône de cadenas s'affiche pendant quelques secondes pour mettre en évidence que le widget n'est pas accessible.</td></tr>
</table>

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Propriété	Description
	• Aucun • Afficher l'icône
Mode de vue en liste déroulante	Sélectionnez le mode de visualisation de tous les widgets de liste déroulante du projet (voir "Mode « plein écran » avec images du widget à liste déroulante" Page 368 pour les détails) • Contexte Vue classique avec les menus déroulants • Plein écran Vue améliorée avec textes et images configurables qui apparaîtront au milieu de l'écran pour faire défiler facilement et sélectionner.

RequêtePage, PageActuelle et OptionsSync

Avec HMI Runtime, vous pouvez échanger les informations des périphériques sur la page affichée par l'IHM. Vous pouvez synchroniser les pages affichées sur le périphérique IHM et sur HMI Client, ou bien vous pouvez contrôler un projet IHM depuis un contrôleur tel qu'un API.

Les propriétés suivantes peuvent être personnalisées :

Propriété	Description
RequêtePage	Page qui doit s'afficher sur le périphérique IHM et sur HMI Client. Une variable associée doit contenir une valeur entière dans la plage des pages disponibles du projet et elle doit être disponible au moins en tant que Ressource lisible.
PageActuelle	Numéro de page affiché sur le périphérique IHM, sur HMI Client ou sur les deux. Une variable associée doit être disponible au moins en tant que Ressource écrivable et doit avoir un type entier de données.
OptionsSync	Synchronisation des pages du projet avec la valeur contenue dans la propriété **PageActuelle**. Les options sont : • **désactiver** : la valeur du numéro de la page est ignorée, • **local** : le numéro de page est affiché sur l'IHM, • **à distance** : le numéro de page est affiché sur HMI Client. • **local + à distance** : le numéro de page est affiché sur l'IHM et sur HMI Client ; si différentes pages sont affichées, le numéro de page affiché est celui de la dernière page chargée.

Exemple : modification forcée de la page depuis le contrôleur/API au périphérique IHM et au HMI Client

Définissez les propriétés de la façon suivante :

RequêtePage	associée à la variable « A »
PageActuelle	vide
OptionsSync	désactiver

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Définissez la valeur de la variable « A » pour afficher la page demandée sur le périphérique IHM et sur le HMI Client.

Exemple : modification forcée de la page depuis le contrôleur/API à l'IHM et au HMI Client. Lisez la page actuellement chargée sur l'IHM

Définissez les propriétés de la façon suivante :

RequêtePage	associée à la variable « A »
PageActuelle	associée à la variable « B » en tant que lire/écrire
OptionsSync	local

Définissez la valeur de la variable « A » pour afficher la page demandée sur le périphérique IHM et sur le HMI Client. La variable « B » contient le numéro de la page actuellement affichée par le périphérique.

Exemple : modification forcée de la page depuis le contrôleur/API au périphérique IHM et au HMI Client. Lisez la page actuellement chargée sur le HMI Client

Définissez les propriétés de la façon suivante :

RequêtePage	associée à la variable « A »
PageActuelle	associée à la variable « B » en tant que lire/écrire
OptionsSync	à distance

Définissez la valeur de la variable « A » pour afficher la page demandée sur l'IHM et sur le HMI Client. La variable « B » contient le numéro de la page actuellement affichée par HMI Client.

Exemple : modification forcée de la page depuis le contrôleur/API au périphérique IHM et au HMI Client. Force la synchronisation de la page HMI Client avec le périphérique IHM (et non l'inverse).

Définissez les propriétés de la façon suivante :

RequêtePage	associée à la variable « A » en tant que lire/écrire
PageActuelle	associée à la même variable « A » que pour **RequêtePage**
OptionsSync	local

Définissez la valeur de la variable « A » pour afficher la page demandée sur l'IHM et le HMI Client. Changez de page sur l'IHM pour afficher la même page sur HMI Client.

Exemple : modification forcée de la page depuis le contrôleur/API au périphérique IHM et au HMI Client. Force la synchronisation de la page de l'IHM avec HMI Client (et non l'inverse).

Définissez les propriétés de la façon suivante :

RequêtePage	associée à la variable « A » en tant que lire/écrire
PageActuelle	associée à la même variable « A » que pour **RequêtePage**
OptionsSync	à distance

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Changez la valeur de la variable « A » pour afficher la page demandée sur l'IHM et le HMI Client. Changez de page sur HMI Client pour afficher la même page sur l'IHM.

Exemple : synchronisez la page affichée entre le périphérique IHM et le HMI Client

Définissez les propriétés de la façon suivante :

RequêtePage	associée à la variable « A » en tant que lire/écrire
PageActuelle	associée à la même variable « A » que pour **RequêtePage**
OptionsSync	local + à distance

En changeant de page sur le périphérique IHM, la même page sera affichée sur HMI Client et vice-versa.

Web

Propriété	Description	
Délai d'attente inactivité web	Définit un délai d'attente pour un client PB4Web. Lorsque le délai d'attente expire sans activité, l'utilisateur actuel est déconnecté.	
	Plage	1–86 400 s (forme 1 s à 24 h)
	Valeur par défaut	600 s
	Valeurs	0 = désactivé
Icône web	L'icône préférée associée aux pages web	
Temps de rafraîchissement	Définit le temps de rafraîchissement pour la communication entre le runtime et clients PB4Web.	
	Plage	500–10 000 ms
	Valeur par défaut	3000 ms
Optimisation du navigateur	**vrai**	Activation de l'optimisation du moteur Web (par défaut)
	faux	Désactivation de l'optimisation du moteur Web (utile pour les anciens navigateurs qui ne soutiennent l'optimisation du moteur web)
Active le JavaScript Global pour les clients distants	Définit si le code JavaScript défini dans les propriétés du projet, déclenché par les événements Alarmes et Programmateurs, doit être exécuté uniquement sur le périphérique IHM local ou sur les clients distants.	

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Propriété	Description	
	Aucun	Ne sera pas exécuté sur les clients distants (exécution seulement dans le périphérique IHM).
	Client	Sera exécuté sur HMI Client
	Web	Sera exécuté sur un client Web.
	Largeur et hauteur	Sera exécuté sur HMI Client et les clients Web.
Bande passante Max (Kbs)	Limite pour un maximum de données envoyées par le serveur (utile pour les vieux navigateurs lents). Configurez sur 0 pour utiliser toute la bande passante disponible (par défaut)	
Mode de connexion des clients Web	**Auto**	Le mode de connexion est sélectionné par le client (par défaut)
	SSE	Forcer le mode d'événements envoyés par serveur
	Longue scrutation	Forcer le mode de scrutation longue
RequêtePageWeb	Vous pouvez synchroniser les pages affichées sur le Clients PB4Web depuis un contrôleur tel qu'un API. Page à afficher sur le Client PB4Web. Une variable associée doit contenir une valeur entière dans la plage des pages disponibles du projet et elle doit être disponible au moins en tant que Ressource lisible.	
Retard des icônes de communication Web (ms)	Retard avant l'affichage de l'icône d'erreur de communication (par défaut 0 ms) La valeur spéciale -1 est toujours désactivée	

Le project.getClientType() peut être utilisé pour récupérer le type de client en cours d'exécution. Voir "Objet du projet" Page 449 dans le chapitre JavaScript pour plus de détails.

Événements

Propriété	Description
LorsdeMolette	Utilisé uniquement avec des périphériques d'entrée de la molette. Généralement, la molette est utilisée pour augmenter/diminuer la valeur d'une variable sans clavier externe. Combinez cette propriété à un événement de modification de la molette et utilisez une action telle que **BiStep** pour augmenter/diminuer une valeur de variable. **L'action OnWheel du projet n'est exécutée que lorsque l'action OnWheel ne sera pas écrasée depuis la page chargée.**

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

7 Le simulateur IHM

Le simulateur IHM vous permet de tester des projets avant de les télécharger sur le périphérique IHM. Il peut être utilisé pour tester le projet lorsqu'aucun périphérique IHM n'est disponible et pour accélérer les activités de développement et de débogage.

Le simulateur IHM prend en charge :

- simulation en ligne - en communication avec des appareils réels (uniquement pour les protocoles avec Ethernet ou de communication RS-232),
- simulation hors ligne - simulant le comportement de variables

La méthode de simulation de données est définie dans la colonne **Simulateur** de l'éditeur de variable.

Méthodes de simulation des données **84**

Paramètres du simulateur **84**

Lancer et arrêter le simulateur **85**

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Méthodes de simulation des données

Définissez le comportement de simulation de variable dans le champ **Simulateur** de l'Éditeur de variables.

Méthode	Description
Variables	Les données sont stockées dans une variable de simulation. Cette variable maintient la valeur de la variable pour que vous puissiez lire et écrire la valeur.
Signal en dents de scie	Une valeur du nombre est incrémentée de **Offset** à **Amplitude + valeur Offset** avec une **Période** de 60 à 3600 secondes. Lorsque le décompte atteint **Amplitude + Offset**, la valeur est réinitialisée à **Offset** et le décompte recommence.
Signal sinusoïdal	Une valeur de signal sinusoïdal est générée et écrite sur la valeur de la variable. Des valeurs **Min.**, **Max.** et **Période** peuvent être définies pour chaque variable.
Signal triangulaire	Une valeur de signal triangulaire est générée et écrite sur la valeur de la variable. Des valeurs **Min.**, **Max.** et **Période** peuvent être définies pour chaque variable.
Signal carré	Une valeur de signal carré est générée et écrite sur la valeur de la variable. Des valeurs **Min.**, **Max.** et **Période** peuvent être définies pour chaque variable.

Voir "Ajouter des variables" Page 28 pour plus de détails.

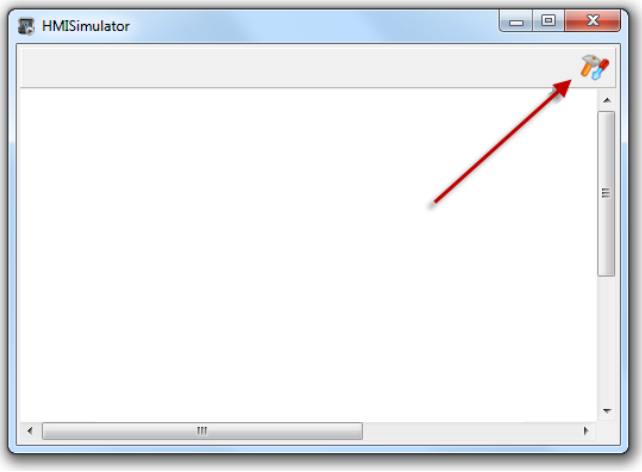
Paramètres du simulateur

Le Simulateur fonctionne par défaut avec des protocoles simulés. Il peut également fonctionner avec de réels protocoles (Ethernet ou des protocoles de série)

Remarque : pour les protocoles ne prenant pas en charge la communication avec des périphériques externes, tels que le protocole Variables, cette option est toujours désactivée.

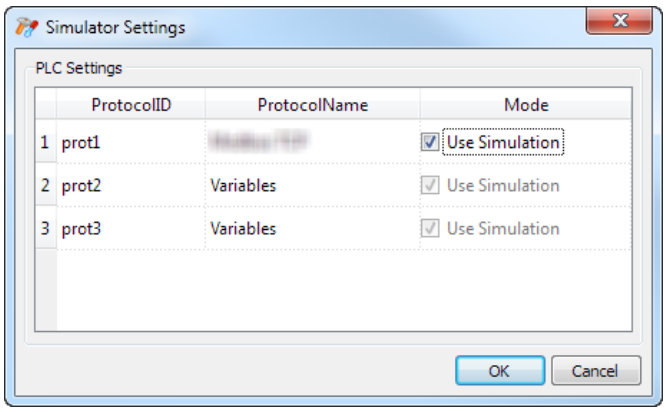
Modification des protocoles simulés

1. Cliquez sur l'icône **Paramètres** du simulateur.



Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

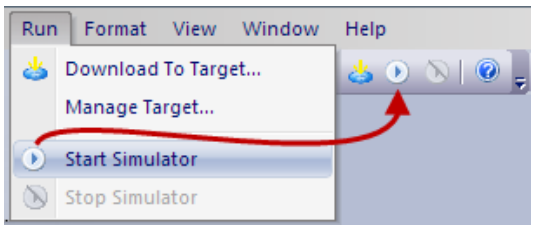
2. Sélectionnez **Utiliser Simulation** pour utiliser des protocoles simulés, sinon des protocoles réels seront utilisés pour la communication avec des périphériques externes.



Lancer et arrêter le simulateur

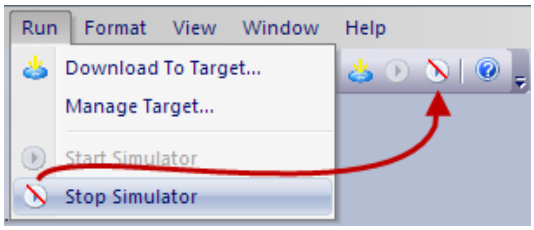
Pour lancer le simulateur :

1. Dans le menu **Exécuter** , cliquez sur **Démarrer Simulateur :** le simulateur fonctionne sur l'ordinateur de la même façon que le serveur sur le périphérique IHM.



Pour arrêter le simulateur :

1. Dans le menu **Exécuter**, cliquez sur **Arrêter le simulateur** ou bien double-cliquez sur le bouton **Quitter** sur la page simulée.



Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

8 Transférer le projet au périphérique IHM

Pour transférer le PB610 Panel Builder 600 projet sur le périphérique IHM, vous pouvez utiliser :

- fonction **Exécuter > Télécharger sur la cible**
- fonction **Exécuter > Mettre à jour un paquet** en utilisant un Périphérique USB

Télécharger vers le périphérique IHM **88**

Mettre à jour un paquet **91**

Le chargeur de runtime **93**

Téléverser des projets **94**

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

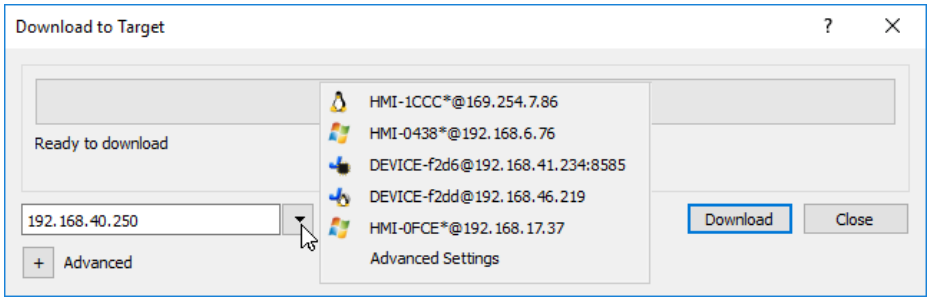
Télécharger vers le périphérique IHM

Chemin : ***Exécuter> Télécharger sur la cible***

Cette fonction transfert un projet et HMI Runtime via Ethernet.

Remarque : le périphérique IHM doit avoir une adresse IP valide. Voir "Paramètres de base du périphérique IHM" Page 8 pour des détails sur la manière d'assigner une adresse IP.

1. Cliquez sur le bouton découverte : une liste des adresses IP détectées s'affiche.
2. Sélectionnez l'adresse IP du périphérique IHM.

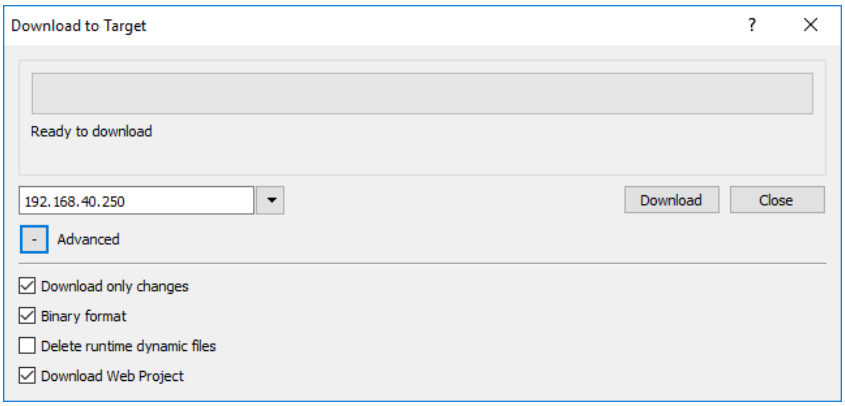


   Vous pouvez même entrer l'adresse IP manuellement ou, le cas échéant, le nom d'hôte fourni par un serveur DNS. Utilisez un outil de service tel que Bonjour, les périphériques IHM basés sur Linux peuvent être découverts en utilisant leur nom d'hôte (par ex. HMI-0d37.local). Bonjour est une marque d'Apple inc.

3. Cliquez sur **Télécharger** : PB610 Panel Builder 600 passe le périphérique IHM en Mode configuration et transfère les fichiers.

Lorsque l'opération de téléchargement est terminée, le périphérique IHM revient automatiquement au mode de fonctionnement et le projet est lancé.

Options avancées



Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

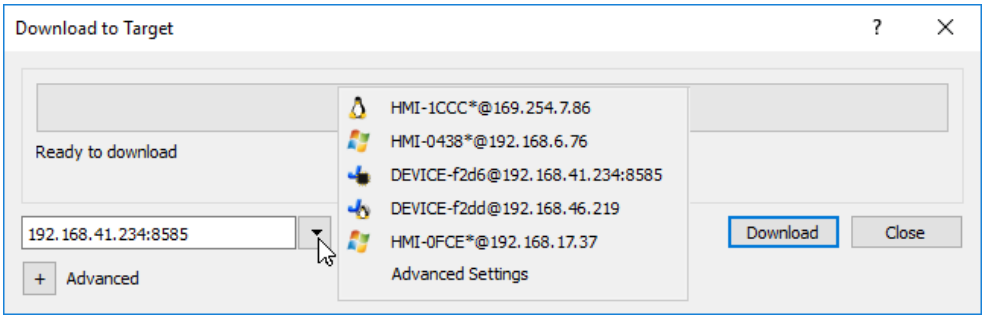
Option	Description
Télécharger uniquement les modifications	Transfère uniquement les fichiers de projet modifiés au périphérique IHM.
Format binaire	Télécharge des fichiers en utilisant le format binaire.
Supprimer les fichiers dynamiques de runtime	Les configurations modifiées de recettes, d'utilisateurs, de planificateurs, etc. qui ont été réalisées au runtime sont supprimées et écrasées par la configuration définie dans le projet. ***MISE EN GARDE : cette opération ne peut pas être annulée, les fichiers dynamiques supprimés ne peuvent pas être restaurés.*** ***MISE EN GARDE : les fichiers dynamiques ne sont pas supprimés lorsqu'ils sont stockés dans des périphériques externes (USB ou cartes SD).***
Télécharger projet web	Télécharge les pages du PB4Web au périphérique IHM.

Lorsque vous transférez un projet, PB610 Panel Builder 600 utilise une combinaison de connexions HTTP et FTP :

- connexion HTTP - envoie les commandes pour passer en mode transfert ou pour charger le projet au runtime,
- session FTP - transfère les fichiers sur la mémoire flash dans le périphérique IHM.

Paramètres avancés

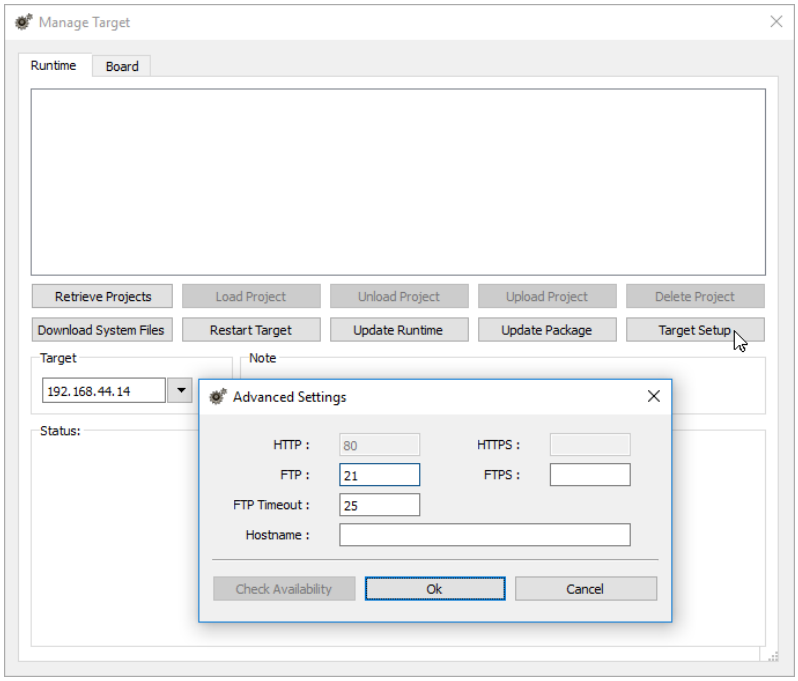
En utilisant l'option « Paramètres avancés », vous pouvez définir les ports à utiliser, mais en général, vous n'avez pas besoin de saisir ces informations parce que les périphériques IHM fourniront les ports à utiliser à l'intérieur de la liste du volet.



Changer de paramètres de connexion du périphérique IHM

Chemin : ***Exécuter> Gérer cible***

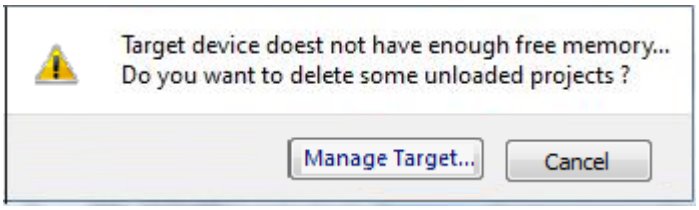
1. Cliquez sur **Configuration cible** : la boîte de dialogue **Paramètres avancés** s'affiche. Le port par défaut pour les connexions HTTPS sur le périphérique IHM est le port 80.



2. Définissez les bons ports HTTP, FTP ou HTTPS, FTPS pour le périphérique IHM. (Ce sont les ports utilisés par le système pour se connecter au périphérique IHM et ils peuvent avoir besoin d'être modifiés lorsque les ports par défaut sont utilisés par d'autres services ou d'autres applications ou si le réseau local nécessite des réglages spécifiques.)
3. Spécifiez un **Nom d'hôte** pour identifier facilement chaque périphérique dans un réseau où plusieurs périphériques sont disponibles. Le nom d'hôte par défaut est « IHM » pour tous les périphériques.
4. Cliquez sur **Télécharger fichiers système**. Au téléchargement suivant, les nouveaux ports sont utilisés dans le périphérique IHM et le nouveau nom d'hôte apparaît dans la liste déroulante

Gérer des projets volumineux

Pour que le téléchargement fonctionne, la taille maximale du projet devrait être au moins inférieure de 2 MB à la mémoire disponible. Dans le cas contraire, vous manquez de mémoire flash dans le périphérique IHM et un message d'avertissement s'affiche.



Pour libérer plus de mémoire :

1. Cliquez sur **Gérer cible**.
2. Supprimez les projets dont vous n'avez plus besoin pour que plus de mémoire soit disponible.

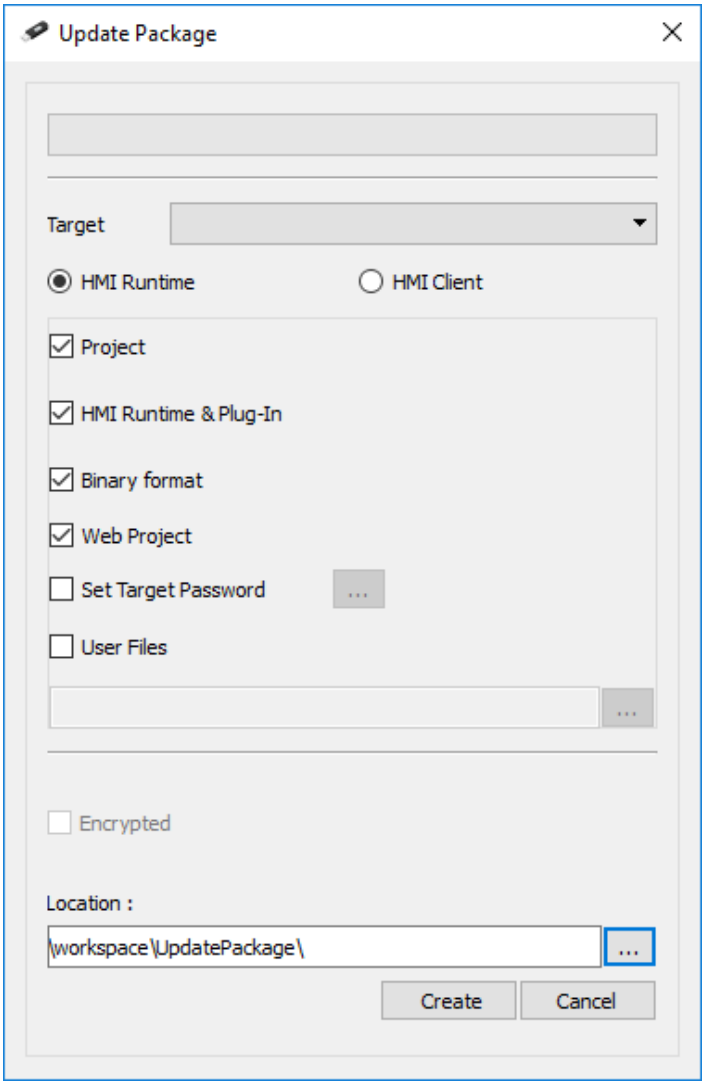
Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Mettre à jour un paquet

Mettre à jour un paquet crée un fichier UpdatePackage.zip pour installer ou mettre à jour l'application dans le périphérique IHM à l'aide d'une clé USB.

Création d'une mise à jour de paquet

Chemin : ***Exécuter> Mettre à jour un paquet***



Option	Description
Cible	Type de périphérique IHM. Sélectionné automatiquement si le projet est ouvert.
Sélecteur d'application	Sélectionne l'application à insérer dans le UpdatePackage.zip • Runtime d'IHM • Client IHM (disponibles seulement sur es les périphériques Linux)

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Option	Description
Projet	Ajouter un projet ouvert à un paquet de mise à jour.
Runtime & Plug-In de l'IHM	Le runtime d'IHM est ajoutée à la mise à jour de paquet. Si le projet est ouvert, les modules d'exécution nécessaires sont également ajoutés pour mettre à jour le paquet.
Format binaire	Télécharge des fichiers en utilisant le format binaire.
Projet Web	Télécharger les pages du PB4Web au périphérique IHM.
Configurer mot de passe cible	Définit le mot de passe pour effectuer des tâches critiques (par exemple, téléchargement/chargement de projet, gestion de la carte) Voir "Protéger l'accès aux périphériques IHM" Page 527.
Fichiers utilisateurs	Sélectionner les fichiers à copier dans le dossier QTHM du périphérique IHM. Taille max = 5 MB
Crypté	Permet le chiffrement du paquet de mise à jour de sorte qu'il ne peut être décompressé par le runtime IHM.
Emplacement	Emplacement du paquet de mise à jour.

Important : Lorsque vous créez un paquet avec l'application Runtime de l'IHM , incluez toujours à la fois le projet et le runtime. Si vous avez besoin d'utiliser un ancien projet avec la dernière version du Runtime, convertissez d'abord le projet. Voir "Installer l'application" Page 4 pour les détails.

Exemple d'emplacement du fichier utilisateur

Ordinateur :

C:\Users\Username\Desktop\myFolder

- *subFolder1/file1*
- *subFolder1/file2*
- *file3*
- *file4*

Périphériques WinCE :

/Flash/QtHmi

- *subFolder1/file1*
- *subFolder1/file2*
- *file3*
- *file4*

Périphériques Linux :

/mnt/data/hmi/qthmi

- *subFolder1/file1*
- *subFolder1/file2*
- *file3*
- *file4*

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Chargement d'une mise à jour de paquet

*Chemin : depuis le menu contextuel > **Mettre à jour***

1. En supposant que vous avez enregistré le paquet dans le dossier racine d'un lecteur USB, retirez le lecteur de l'ordinateur, branchez-le dans le périphérique IHM, affichez le menu contextuel en maintenant votre doigt pendant quelques secondes sur l'écran et sélectionnez **Mettre à jour**.
2. Le système vérifie la présence du paquet de mise à jour à la racine du lecteur USB et demande la confirmation de procéder à la mise à jour.

3. Sélectionnez **Sélectionner automatiquement de la meilleure correspondance** et cliquez sur **Suivant** : la procédure se termine automatiquement. Autrement, utilisez le bouton de navigation pour sélectionner le fichier à utiliser.

Le chargeur de runtime

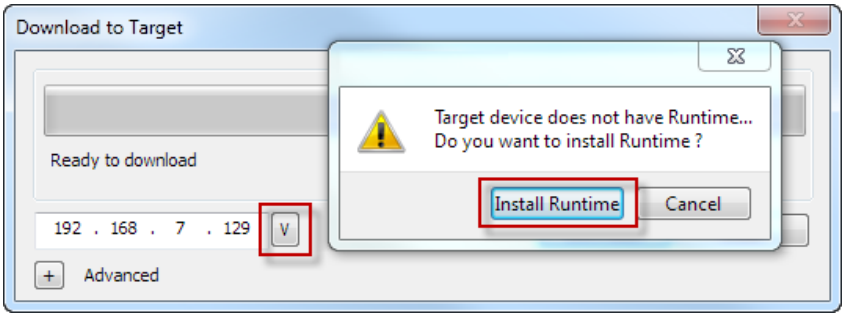
Les périphériques IHM sont livrés depuis l'usine sans runtime.

Lorsque vous mettez le périphérique pour la première fois, la fenêtre de chargement du runtime s'affiche (voir "Paramètres du système" Page 477 pour les détails)

La présence du chargeur de runtime dépend du système d'exploitation du périphérique et peut ne pas être disponible sur toutes les unités. Les anciennes versions des périphériques IHM peuvent ne pas inclure le chargeur de runtime. Contactez le support technique pour obtenir de plus amples informations.

Installation du runtime depuis PB610 Panel Builder 600

Lorsque vous téléchargez un projet, le runtime est installée automatiquement si nécessaire.



Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Voir "Transférer le projet au périphérique IHM" Page 87 pour les détails.

1. Cliquez sur **Installer Runtime** : la procédure est exécutée automatiquement.

Installation du runtime depuis un lecteur USB

1. Préparer le paquet de mise à jour comme décrit dans "Mettre à jour un paquet" Page 91
2. Branchez le lecteur USB dans le périphérique et suivez les instructions sur le type de périphérique (voir "Paramètres du système" Page 477pour les détails)

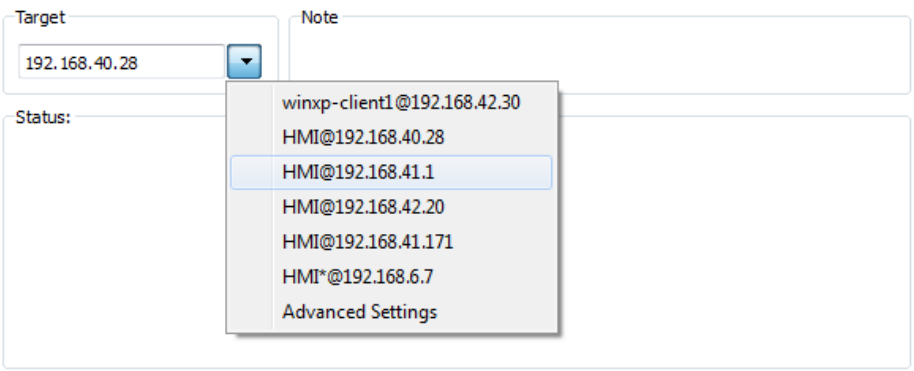
Remarque : les anciennes versions des périphériques IHM ne peuvent pas supporter l'installation automatique du Runtime. Contactez le support technique pour obtenir plus d'informations.

Téléverser des projets

*Chemin : **Exécuter**> **Gérer cible***

Vous pouvez copier un projet depuis le runtime sur un ordinateur où PB610 Panel Builder 600 est en cours d'exécution.

1. Dans l'onglet **Runtime**, sélectionner l'adresse IP du périphérique depuis la **Cible** de la liste déroulante.



2. Cliquez sur **Récupération des projets** : une liste de tous les projets disponible est affichée.
3. Sélectionnez le projet à téléverser
4. Cliquer sur **Téléverser le projet**

Le téléversement peut être protégé par un mot de passe. Voir "Protéger l'accès aux périphériques IHM" Page 527 pour les détails.

5. Si nécessaire, entrez le mot de passe. Le processus de téléversement démarre.

Une copie du projet est enregistré dans :

C:\Users\username\Documents\PB610 Panel Builder 600\workspace\Uploaded\RuntimeIPAddress\workspace\ProjectName

Remarque : si l'opération de téléversement échoue, vérifiez les paramètres du pare-feu de l'Ordinateur où PB610 Panel Builder 600 est en cours d'exécution.

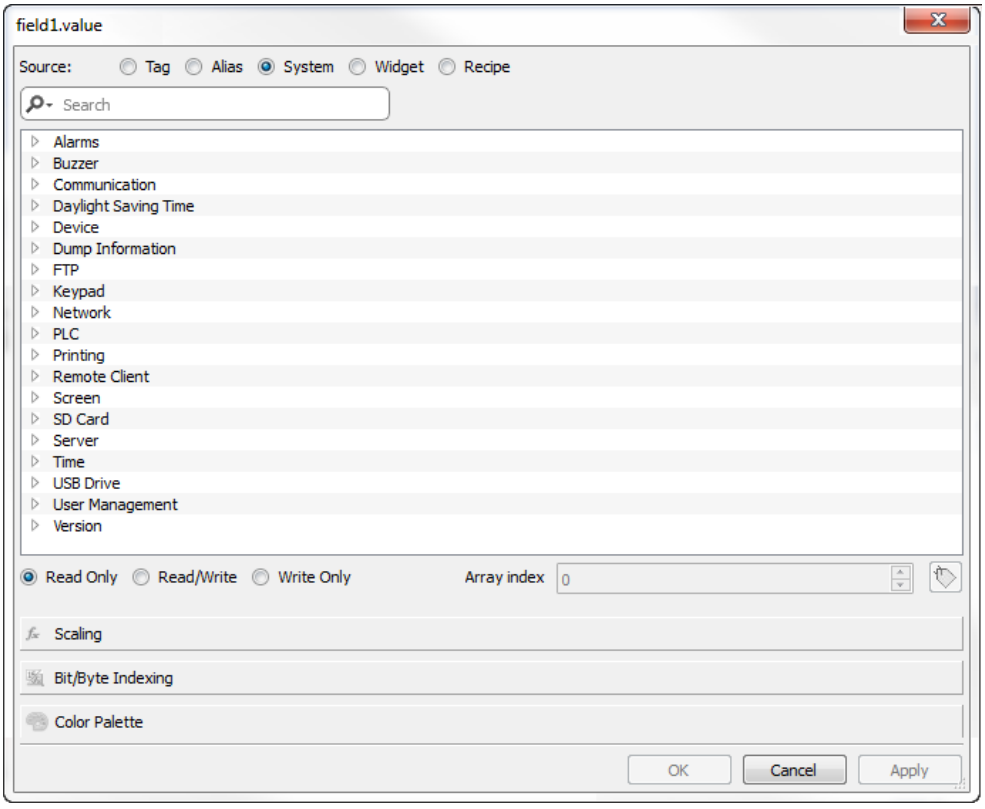
Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

9 Variables système (Associer à)

Chein : ***Source*** *>* ***Associer à***

Les variables système sont des variables spéciales qui contiennent des informations concernant le runtime de l'IHM.

Remarque : Les variables système sont également disponibles en tant que protocole standard dans l'éditeur de protocole. Utilisez les variables système comme protocole lorsque vous devez transférer des données entre des variables système et des variables de périphériques, ou pour sélectionner un taux de rafraîchissement personnalisé pour une variable système.



Variables d'alarmes **97**

Variables du vibreur sonore **97**

Variables de communication **98**

Variables bases de données **98**

Variables de passage à l'heure d'été (DLS) **99**

Variables de périphérique **100**

Variables informations sur les vidages **102**

Variables client FTP **103**

Variables réseau **103**

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Variables d’impression **104**
Variables client à distance **105**
Variables de version **106**
Variables d'écran **106**
Variables carte SD **106**
Variables serveur **107**
Variables d’heure **107**
Variables d'écran tactile **108**
Variables lecteur USB **109**
Variables de gestion des utilisateurs **109**

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Variables d'alarmes

Nombre d'alarmes du type demandé.

Variable	Description	Type de données
Alarme non confirmée	Vrai quand les alarmes non confirmées sont en attente (Non déclenchée Non confirmée<>0) OU (Déclenchée Non confirmée<>0)	booléen Lecture seule
Alarme déclenchée	Vrai quand au moins une alarme a été déclenchée (Déclenchée Confirmée<>0) OU (Déclenchée<>0) OU (Déclenchée Non confirmée<>0)	booléen Lecture seule
Nombre d'événements d'alarme manqués	Alarmes dépassant la file d'attente des événements. La longueur de la file d'attente est définie dans le fichier *engineconfig.xml*.	entier Lecture seule
Nombre d'alarmes non déclenchées, confirmées	Alarmes désactivées ; alarmes déjà confirmées	entier Lecture seule
Nombre d'alarmes non déclenchées, non confirmées	Alarmes désactivées ; en attente de confirmation	entier Lecture seule
Nombre d'alarmes déclenchées, confirmées	Alarmes actives ; alarmes déjà confirmées	entier Lecture seule
Nombre d'alarmes déclenchées	Alarmes actives : confirmation non requise	entier Lecture seule
Nombre d'alarmes déclenchées, non confirmées	Alarmes actives ; en attente de confirmation	entier Lecture seule

Remarque : pour des raisons de compatibilité, les anciens noms restent toujours valides mais leur utilisation est obsolète.

Variables du vibreur sonore

Ajuster le comportement du vibreur sonore.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Variable	Description	Type de données
Configuration du vibreur sonore	**0** = désactivé **1** = activé (le vibreur sonore est audible pour tous les événements de l'écran tactile) **2** = statut du vibreur commandé par la variable système **commande vibreur** ou par la propriété dans les "Propriétés de projet" Page 69 **Le vibreur sur l'écran tactile (Setup = 1) n'est pas disponible sur les plateformes Linux. Voir la propriété « *Vibreur au toucher* » comme alternative.**	entier
Commandes du vibreur	**0** = vibreur éteint **1** = vibreur allumé **2** = vibreur clignotant	entier
Buzzer Off Time	Durée en millisecondes du temps de silence du vibreur lorsque le clignotement a été sélectionné. Défaut = 1000. Plage = 100 à 5000.	entier
Buzzer On Time	Durée en millisecondes du temps d'activation du vibreur lorsque le clignotement a été sélectionné. Défaut = 1000. Plage = 100 à 5000.	entier

Variables de communication

État de communication entre le périphérique IHM et les contrôleurs.

Variable	Description	Type de données
Protocol Communication Status	Résume l'état des protocoles de communication. **0** = Aucun protocole en cours d'exécution, il est possible que les pilotes des protocoles n'aient pas été téléchargés correctement sur le périphérique IHM. **1** = Protocoles chargés et démarrés, aucune erreur de communication. **2** = Au moins un protocole de communication signal une erreur.	entier Lecture seule
Protocol Error Message	Erreur de communication avec source de l'erreur. Par exemple : « [xxxx] » où « xxxx » est l'abréviation du protocole, source de l'erreur. Des acronymes multiples apparaissent dans le cas de sources d'erreurs multiples. Vide lorsqu'aucune erreur n'est signalée.	Chaîne ASCII Lecture seule
Protocol Error Count	Nombre d'erreurs de communication survenues depuis la dernière réinitialisation. Réinitialisez la valeur avec l'action Réinitialiser le nombre d'erreurs du protocole, voir "Actions système" Page 161.	entier Lecture seule

Variables bases de données

État de connexion de la base de données.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Variable	Description	Type de données
Database link error message	Description de la dernière erreur détectée	chaîne Lecture seule
État du lien de la base de données	0 = Non défini (pas encore initialisé) 1 = EnLigne (prêt) 2 = HorsLigne (non disponible) 3 = Transfert en cours 4 = Erreur	entier Lecture seule
Database link error count	Compteur d'erreurs. Augmente après chaque erreur.	entier Lecture seule

Chaque variable de base de données est un tableau où l'index sélectionne la connexion de lien de base de données (plage 1 à 10)

Remarque : les variables sont mises à jour uniquement lorsqu'une action de connecteur de base de données est exécutée.

Remarque : ces variables sont disponibles sous forme de variables dans le protocole « Variables système ».

Variables de passage à l'heure d'été (DLS)

Informations sur l'horloge système. Les variables contiennent des informations sur l'heure « locale ». Les options disponibles sont l'heure standard (heure solaire) et l'heure d'été (DST).

Remarque : toutes les variables sont en lecture seule ; il n'est pas possible de les utiliser pour mettre à jour l'horloge système.

Variable	Description
Standard Offset	Décalage en minutes par rapport au GMT lorsque l'heure est réglée sur l'heure standard (par exemple : -8x60 = -480 minutes).
Standard Week	Semaine à laquelle l'heure standard commence (par exemple : Premier = 1).
Standard Month	Mois auquel l'heure standard commence . Plage : 0 - 11. (par exemple : Novembre = 10).
Standard Day	Jour de la semaine auquel l'heure standard commence (par exemple : Dimanche = 0).
Standard Hour	Heure à laquelle l'heure standard commence (par exemple : 02 = 2).

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Variable	Description
Standard Minute	Minute à laquelle l'heure standard commence (par exemple : 00 = 0).
DST Offset	Décalage en minutes par rapport au GMT lorsque le passage à l'heure est défini
DST Week	Semaine à laquelle le passage à l'heure d'été commence
DST Month	Mois auquel le passage à l'heure d'été commence. Plage : 0 - 11.
DST Day	Jour de la semaine auquel le passage à l'heure d'été commence
DST Hour	Heure à laquelle le passage à l'heure d'été commence
DST Minute	Minute à laquelle le passage à l'heure d'été commence

Variables de périphérique

Paramètres des périphériques et informations sur l'état de fonctionnement.

Variable	Description	Type de données
Available System Memory	Mémoire RAM libre en octets.	uint64 Lecture seule
Backlight Time	Temps d'activation en heures du rétroéclairage de l'affichage depuis la fabrication du périphérique.	entier non signé Lecture seule
Battery LED	Active/désactive le voyant de batterie faible de la DEL (si disponible). **0** = désactivé **1** = activé Ce n'est pas disponible sur les plateformes Linux (consultez "Capacités des périphériques IHM" Page 541 pour plus de détails sur les panneaux)	entier
Battery Timeout	Réservé Ce n'est pas disponible sur les plateformes Linux (consultez "Capacités des périphériques IHM" Page 541 pour plus de détails sur les panneaux)	entier
Display Brightness	Réinitialise et adapte le niveau de luminosité. Même lorsqu'il est réglé sur 0, le rétroéclairage reste allumé et le compteur **Durée de rétroéclairage** augmente.	entier

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Variable	Description	Type de données
	Plage : 0-255 Sur périphérique WinCE uniquement : Lorsque le rétroéclairage est réglé sur un niveau de luminosité faible (0…3), il reste allumé à un niveau plus élevé pendant 8 secondes afin de permettre à l'utilisateur d'effectuer les réglages, puis il s'éteint.	
External Timeout	Délai d'inactivité après lequel le rétroéclairage de l'écran s'éteint automatiquement. Le rétroéclairage s'allume automatiquement dès que l'utilisateur touche l'écran. **-1** = éteint le rétroéclairage et désactive les commandes tactiles (éteindre l'affichage). Le compteur **Backlight Time** s'arrête. **0** = allumer le rétroéclairage (allume l'affichage) **1..n** = délai d'attente, en secondes, pour éteindre le rétroéclairage (minuteur de l'écran de veille) Sur le périphérique Linux (voir "Capacités des périphériques IHM" Page 541 pour les détails du panneau), délai d'attente géré en quelques minutes. La valeur saisie est convertie en minutes arrondies à la valeur suivante, par exemple, 60, 120, 180.	entier
Flash Free Space	Espace flash libre restant dans la mémoire flash interne.	uint64 Lecture seule
Manufacturer Code	Code interne qui identifie le type d'IHM	entier non signé Lecture seule
System Font List	Liste des polices du système	chaîne Lecture seule

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Variable	Description	Type de données
Mode système	État de service du runtime. **1** = démarrage en cours **2** = mode de configuration **3** = mode opérationnel **4** = redémarrage **5** = arrêt	entier
Temps de montée du système	Durée pendant laquelle le système a été alimenté depuis la fabrication de l'unité (en heures).	entier non signé Lecture seule

Variables informations sur les vidages

Statut du processus de copie vers des lecteurs externes (USB ou carte SD) pour les mémoires tampons de tendances d'événements.

Variable	Description	Type de données
Message d'erreur Vidage	Retour du message d'erreur si une erreur se produit pendant l'opération de vidage	chaîne Lecture seule
Statut du vidage d'archive	**0** = état initial par défaut **1** = opération déclenchée **2** = opération terminée avec succès **3** = opération terminée avec erreurs	entier Lecture seule
Statut du vidage de recette	**0** = état initial par défaut **1** = opération déclenchée **2** = opération terminée avec succès **3** = opération terminée avec erreurs	entier Lecture seule
Statut du vidage de tendance	**0** = état initial par défaut **1** = opération déclenchée **2** = opération terminée avec succès **3** = opération terminée avec erreurs	entier Lecture seule

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Variable	Description	Type de données
Statut de la réinitialisation de recette	**0** = état initial par défaut **1** = opération déclenchée **2** = opération terminée avec succès **3** = opération terminée avec erreurs	entier Lecture seule
Statut de la restauration de recette	**0** = état initial par défaut **1** = opération déclenchée **2** = opération terminée avec succès **3** = opération terminée avec erreurs	entier Lecture seule

Variables client FTP

Les variables client FTP sont mises à jour lorsque les actions FTP sont utilisées.

Variable	Description	Type de données
Commande en cours FTP	Dernière commande FTP	chaîne Lecture seule
Message d'erreur FTP	Dernier message d'erreur FTP	chaîne Lecture seule
Progression FTP	Progression téléchargement / chargement (0/100%)	court Lecture seule
État FTP	État de la dernière commande FTP : • 0 = inactif • 1 = actif • 2 = effectué • 3 = erreur	court Lecture seule

Variables réseau

Paramètres réseau du périphérique.

Variable	Description	Type de données
Paramètres de l'adaptateur	Chaîne JSON qui permet de lire ou de mettre à jour les paramètres d'adaptateurs réseau	chaîne
Gateway	Adresse de passerelle de l'interface Ethernet principale du périphérique	chaîne Lecture

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Variable	Description	Type de données
		seule
IP Address	Adresse IP de l'interface Ethernet principale du périphérique	chaîne Lecture seule
Mac ID	Mac ID de l'interface Ethernet principale du périphérique	chaîne Lecture seule
Status	Contient les résultats de la dernière opération requise par l'écriture dans les Paramètres de l'adaptateur. Cette valeur est actualisée après chaque opération d'écriture. • Si la chaîne est vide, cela signifie qu'il n'y a aucune erreur • Descriptions des erreurs les plus récentes	chaîne Lecture seule
Subnet Mask	Masque de sous réseau de l'interface Ethernet principale du périphérique	chaîne Lecture seule

Variables d'impression

Informations concernant les fonctions d'impression.

Variable	Description	Type de données
Completion percentage	Progression de la tâche d'impression actuelle en pour cent. Plage : 0-100	Lecture seule
Current disk usage	Taille en octets du dossier de stockage des rapports PDF. Si *Flash* a été sélectionné en tant que *Type de média spool*, cette valeur correspond à *rapportspool*.	Lecture seule
Current job	Nom du rapport que la tâche traite actuellement. La tâche actuelle est la suivante : • [nom de rapport] pour un **Rapport graphique** • [première ligne de texte] pour un **Rapport texte**	Lecture seule
Current RAM usage	Taille en octets de la mémoire RAM utilisée pour le traitement de la tâche actuelle	Lecture seule
Disk quota	Taille maximale en octets du dossier de stockage des rapports PDF	Lecture seule
Graphic job queue size	Nombre de tâches de graphique dans la file d'impression	Lecture seule

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Variable	Description	Type de données
Last error message	Description de la dernière erreur affichée	chaîne Lecture seule
RAM quota	Taille maximale en octets de la mémoire RAM utilisée pour la génération des rapports	Lecture seule
Status	Statut du système d'impression. Valeurs : • **idle** • **Erreur** • **paused** • **Impression**	chaîne Lecture seule
Text job queue size	Nombre de tâches de texte dans la file d'attente d'impression	Lecture seule

Variables client à distance

Sur les clients distants, la variable système ci-dessous peut être utilisée pour savoir si le serveur (périphérique IHM) est accessible.

Variable	Description	Type de données
Statut de connexion	**0** = le client ne peut pas atteindre le client du serveur. La connexion avec le serveur est perdue. **1** = le client peut atteindre le client du serveur. La connexion avec le serveur est active. Il ne s'agit que d'une variable côté client. Sur HMI Runtime ou PB610 PC Runtime elle sera toujours 0.	entier (32 bits) Lecture seule

Les variables système suivantes sont associées aux fichiers lors du transfert sur un périphérique IHM distant.

Variable	Description	Type de données
Download from HMI error message	Description de l'erreur	Chaîne ASCII Lecture seule
Download from HMI percentage	Progression du téléchargement (0→100)	Lecture seule
Download from HMI status	**0** = inactif, action non utilisée ou terminée **1**= téléchargement du fichier en cours	entier (32 bits) Lecture seule

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Variable	Description	Type de données
	2 = erreur	
Upload to HMI error message	Description de l'erreur	Chaîne ASCII Lecture seule
Upload to HMI percentage	Progression du téléversement (0→100)	Lecture seule
Upload to HMI status	**0** = inactif, action non utilisée ou terminée **1**= chargement du fichier en cours **2** = erreur	entier (32 bits) Lecture seule

Variables de version

Versions du système d'exploitation et du runtime.

Variable	Description	Type de données
Main OS Version	Version de l'OS principal.	chaîne
Runtime Version	Version du runtime.	chaîne

Variables d'écran

Statut de l'écran.

Variable	Description
Time remaining to unlock	Temps restant avant le déverrouillage de l'écran (voir action **Verrouiller écran**, "Actions sur pages" Page 150)
X Screen resolution	Affiche la dimension horizontale de l'écran en pixels
Y Screen resolution	Affiche la dimension verticale de l'écran en pixels

Variables carte SD

Informations concernant la carte SD externe.

Variable	Description	Type de données
SD Card FreeSpace	Espace de mémoire libre sur la carte en octets	uint64 Lecture seule
SD Card Name	Nom de la carte SD	chaîne

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Variable	Description	Type de données
		Lecture seule
SD Card Size	Tailles en octets de la carte enfichée dans le port	uint64 Lecture seule
SD Card Status	0 = carte SD débranchée 1 = carte SD branchée	entier Lecture seule

Variables serveur

Statut du serveur.

Important : toutes les variables se réfèrent au serveur, non à HMI Client.

Variable	Description	Type de données
PageActuelle	Nom de la page actuellement affichée	chaîne
Current project	Nom du projet actuel	chaîne
Operating mode time	Secondes écoulées depuis que le périphérique a démarré en mode opérationnel	uint64
Project load time	Date de chargement du projet sur le HMI Runtime au format **Date système** (millisecondes).	uint64

Variables d’heure

Heure système au format UTC.

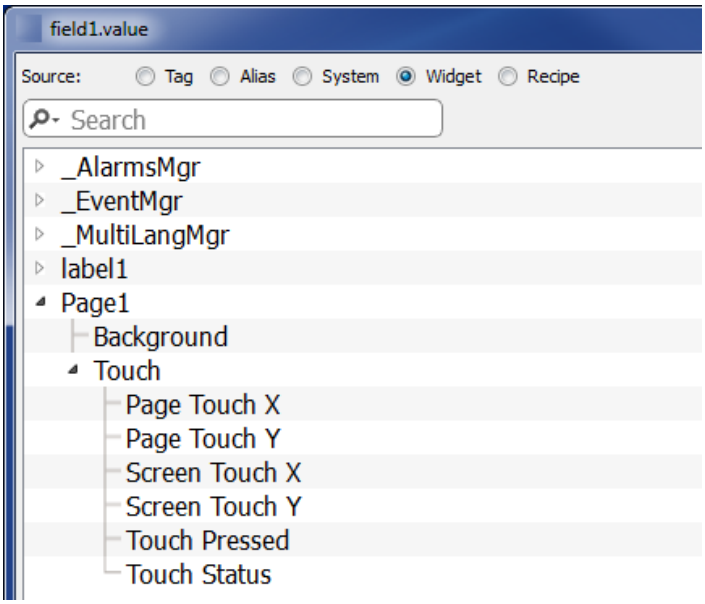
Variable	Description	Type de données
Jour du mois	Plage : 1-31	entier
Jour de la semaine	Plage : 0 = Dimanche, .. , 6 = Samedi	entier
Heure	Plage : 0-23	entier
Minute	Plage : 0-59	entier
Mois	Plage : 1-12	entier
Seconde	Plage : 0-59	entier
Heure du	Identique à l'heure UTC. Il est également possible de définir cette variable au	entier non signé

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Variable	Description	Type de données
système	format date/heure.	
Année	Année en cours	entier

Variables d'écran tactile

Statut et position du curseur sur l'écran tactile. Ce sont des propriétés de la page active et elles peuvent être sélectionnées dans la section **Widget**.



remarque : les dimensions de page peuvent différer de celles de l'affichage du périphérique IHM.

Variable	Description	Java Script
Page Touch X **Page Touch Y**	Position du curseur par rapport à la page	page.primaryTouch.x page.primaryTouch.y
Screen Touch X **Screen Touch Y**	Position du curseur par rapport à l'écran tactile	page.primaryTouch.screenX page.primaryTouch.screenY

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Variable	Description	Java Script
Touch Press	**0** = écran non touché **1** = écran touché	page.primaryTouch.pressed
Touch Status	Modifications génériques sur l'écran tactile. Cette variable contient la concaténation des variables **Screen Touch X**, **Screen Touch Y** et **Touch Press** (par exemple, « 924,129,0 »). La fonction principale de cette variable est de déclencher un événement à l'aide de la fonctionnalité OnDataUpdate dès qu'un changement intervient (x, y ou clic).	page.primaryTouchStatus

Variables lecteur USB

Informations concernant le lecteur USB externe connecté au périphérique.

Variable	Description	Type de données
USB Drive free space	Espace de mémoire libre en octets	uint64 Lecture seule
USB Drive Name	Nom du périphérique USB	chaîne Lecture seule
USB Drive Size	Taille en octets du lecteur branché au port USB	uint64 Lecture seule
USB Drive Status	0 = lecteur USB débranché 1 = lecteur USB branché	entier Lecture seule

Variables de gestion des utilisateurs

Informations concernant les utilisateurs et les groupes.

Variable	Description	Type de données
This Client User-Name	Nom de l'utilisateur connecté au client où la variable système est affichée.	chaîne Lecture seule
This Client Group-Name	Groupe de l'utilisateur actuellement connecté	chaîne Lecture seule
This Client ID	Uniquement pour HMI Client. Les clients locaux et distants connectés au même	court

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Variable	Description	Type de données
	serveur (par exemple, le runtime) reçoivent un ID unique.	Lecture seule
No of Remote-Clients Alive	Nombre de HMI Client connectés au serveur	court Lecture seule

JavaScript

Depuis JavaScript, les variables sont accessibles en tant que propriétés de l'objet _SysPropMgr.

Exemple :

```
var sysVar = project.getWidget( "_SysPropMgr" ); var UserName = sysVar.getProperty
("This Client User-Name"); var UserGroup = sysVar.getProperty("This Client Group-
Name"); var clientId = sysVar.getProperty("This Client ID"); var numClients =
sysVar.getProperty("No of Remote-Clients Alive");
```


Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

10 Variables système (protocole)

Le pilote de communication des Variables système permet de créer des variables pointant sur les informations du système.

Le pilote de communication de variables n'est pas compté comme protocole physique.
Se référer au **Tableau des fonctions et limites** de manuel principal dans la ligne « Nombre de protocoles physiques ».

Paramètres de l'éditeur de protocole **112**

Variables par défaut **112**

Variables de services **128**

Variables de mémoire rémanente **129**

Importation de variables **135**

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Paramètres de l’éditeur de protocole

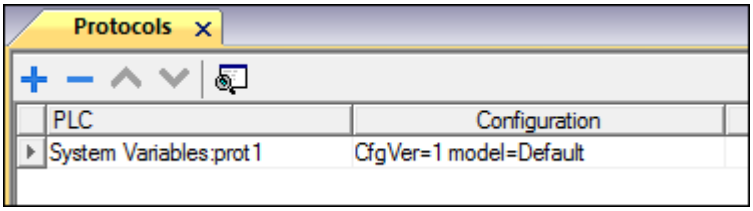
Ajouter un protocole

Pour configurer le protocole :

1. Dans le nœud **Config**, double-cliquez sur **Protocoles**.
2. Pour ajouter un pilote, cliquez sur **+** : une nouvelle ligne est ajoutée.
3. Sélectionnez le protocole dans la liste **API**.

La boîte de dialogue de configuration du protocole s’affiche.

Dans la liste de modèles d'API, sélectionnez le type particulier de Variables système.



Variables par défaut

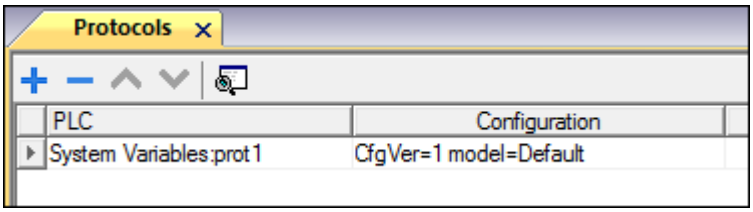
Variables système - Le protocole par défaut permet de créer des variables qui pointent vers les variables système de l'IMH concernant :

- Alarmes
- Vibreur sonore
- Communication
- Base de données
- Heure d'été
- Périphérique
- Informations sur les vidages
- Réseau
- Écran
- Carte SD
- Serveur
- Heure
- Lecteur USB
- Version
- Commutateur de communication virtuelle

Paramètres de l’éditeur du protocole

Dans la liste des modèles d'API de la boîte de dialogue Éditeur de protocole, sélectionnez Défaut.

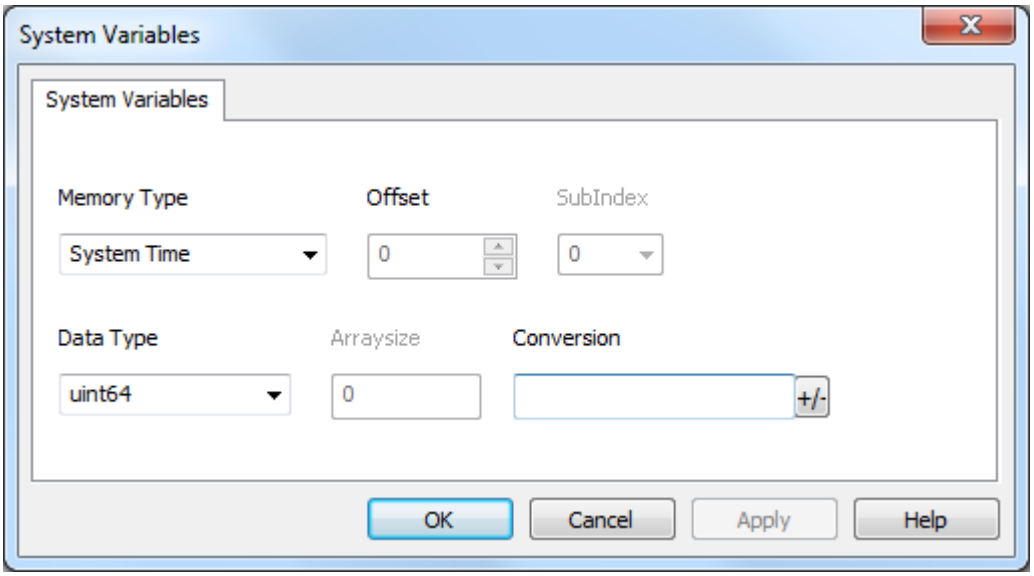
Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



Paramètres de l'éditeur de variables

Chemin : ***VueProjet*** *>* ***Configuration*** *> double-cliquez sur* ***Variables***

1. Pour ajouter une variable, cliquez sur **+** : une nouvelle ligne est ajoutée.
2. Sélectionnez **Variables système** dans la liste **Driver** : le dialogue de définition de variables s'affiche.



Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

<table>
<tr><th>Élément</th><th>Description</th></tr>
<tr><td>Type de mémoire</td><td>
Représente la variable système à laquelle la variable renvoie.

La rubrique ci-dessous montre la liste complète des variables système possibles, regroupées par catégorie.
<table>
<tr><th colspan="3">Variables d'alarmes</th></tr>
<tr><th>Nom de variable</th><th>Description</th><th>Type de données</th></tr>
<tr><td>Alarme non confirmée</td><td>Vrai quand les alarmes non confirmées sont en attente

(Non déclenchée Non confirmée<>0) OU (Déclenchée Non confirmée<>0)</td><td>booléen

Lecture seule</td></tr>
<tr><td>Alarme déclenchée</td><td>Vrai quand au moins une alarme a été déclenchée

(Déclenchée Confirmée<>0) OU (Déclenchée<>0) OU (Déclenchée Non confirmée<>0)</td><td>booléen

Lecture seule</td></tr>
<tr><td>Nombre d'événements d'alarme manqués</td><td>Alarmes dépassant la file d'attente des événements. La longueur de la file d'attente est définie dans le fichier engineconfig.xml.</td><td>entier

Lecture seule</td></tr>
<tr><td>Nombre d'alarmes non déclenchées, confirmées</td><td>Alarmes désactivées ; alarmes déjà confirmées</td><td>entier

Lecture seule</td></tr>
<tr><td>Nombre d'alarmes non déclenchées, non confirmées</td><td>Alarmes désactivées ; en attente de confirmation</td><td>entier

Lecture seule</td></tr>
<tr><td>Nombre d'alarmes déclenchées, confirmées</td><td>Alarmes actives ; alarmes déjà confirmées</td><td>entier

Lecture seule</td></tr>
<tr><td>Nombre d'alarmes déclenchées</td><td>Alarme active : confirmation non requise</td><td>entier

Lecture seule</td></tr>
<tr><td>Nombre d'alarmes déclenchées, non confirmées</td><td>Alarmes actives ; en attente de confirmation</td><td>entier

Lecture seule</td></tr>
</table>
</td></tr>
</table>

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Élément	Description

Variables du vibreur sonore

Nom de variable	Description	Type de données
Configuration du vibreur sonore	**0** = désactivé **1** = activé (le vibreur sonore est audible pour tous les événements de l'écran tactile) **2** = statut du vibreur commandé par la variable système Commande vibreur ou par la propriété Vibreur au toucher dans les « Propriétés du projet » du manuel principal **Le vibreur sur l'écran tactile (Setup = 1) n'est pas disponible sur les plateformes Linux. Voir la propriété « Vibreur au toucher » comme alternative.**	entier
Commandes du vibreur	**0** = vibreur éteint **1** = vibreur allumé **2** = vibreur clignotant	entier
Buzzer Off Time	Durée en millisecondes du temps de silence du vibreur lorsque le clignotement a été sélectionné. Défaut = 1000. Plage = 100 à 5000.	entier
Buzzer On Time	Durée en millisecondes du temps d'activation du vibreur lorsque le clignotement a été sélectionné. Défaut = 1000. Plage = 100 à 5000.	entier

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

<table>
<tr><th>Élément</th><th>Description</th></tr>
<tr><td></td><td>
<table>
<tr><th colspan="3">Variables de communication</th></tr>
<tr><th>Nom de variable</th><th>Description</th><th>Type de données</th></tr>
<tr><td>Protocol Communication Status</td><td>Résume l'état des protocoles de communication.
0 = Aucun protocole en cours d'exécution, il est possible que les pilotes des protocoles n'aient pas été téléchargés correctement sur le périphérique IHM.
1 = Protocoles chargés et démarrés, aucune erreur de communication.
2 = Au moins un protocole de communication signal une erreur.</td><td>entier
Lecture seule</td></tr>
<tr><td>Protocol Error Message</td><td>Erreur de communication avec source de l'erreur.
Par exemple : « [xxxx] » où « xxxx » est l'abréviation du protocole, source de l'erreur.
Des acronymes multiples apparaissent dans le cas de sources d'erreurs multiples. Vide lorsqu'aucune erreur n'est signalée.</td><td>chaîne
Lecture seule</td></tr>
<tr><td>Protocol Error Count</td><td>Nombre d'erreurs de communication survenues depuis la dernière réinitialisation. Réinitialisez la valeur avec l'action Réinitialiser le nombre d'erreurs du protocole, voir « Actions système » du manuel principal.</td><td>entier
Lecture seule</td></tr>
</table>
<table>
<tr><th colspan="3">Variables bases de données</th></tr>
<tr><th>Nom de variable</th><th>Description</th><th>Type de données</th></tr>
<tr><td>Database link error message</td><td>Description de la dernière erreur détectée</td><td>chaîne
Lecture seule</td></tr>
<tr><td>État du lien de la base de données</td><td>0 = Non défini (pas encore initialisé)
1 = EnLigne (prêt)
2 = HorsLigne (indisponible)</td><td>entier
Lecture seule</td></tr>
</table>
</td></tr>
</table>

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Élément	Description

Variables bases de données

Nom de variable	Description	Type de données
	3 = Transfert en cours **4** = Erreur	
Database link error count	Compteur d'erreurs. Augmente après chaque erreur	entier Lecture seule

Chaque variable de base de données est un tableau où l'index sélectionne la connexion de lien de base de données (plage 1 à 10)
Les variables sont mises à jour uniquement lorsqu'une action de connecteur de base de données est exécutée

Variables de passage à l'heure d'été

Nom de variable	Description	Type de données
Standard Offset	Décalage en minutes par rapport au GMT lorsque l'heure est réglée sur l'heure standard (par exemple : -8x60 = -480 minutes).	entier Lecture seule
Standard Week	Semaine à laquelle l'heure standard commence (par exemple : Premier = 1).	entier Lecture seule
Standard Month	Mois auquel l'heure standard commence . Plage : 0 - 11. (par exemple : Novembre = 10).	entier Lecture seule
Standard Day	Jour de la semaine auquel l'heure standard commence (par exemple : Dimanche = 0).	entier Lecture seule
Standard Hour	Heure à laquelle l'heure standard commence (par exemple : 02 = 2).	entier Lecture seule
Standard Minute	Minute à laquelle l'heure standard commence (par exemple : 00 = 0).	entier

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Élément	Description

Variables de passage à l'heure d'été

Nom de variable	Description	Type de données
		Lecture seule
DST Offset	Décalage en minutes par rapport au GMT lorsque le passage à l'heure est défini	entier Lecture seule
DST Week	Semaine à laquelle le passage à l'heure d'été commence	entier Lecture seule
DST Month	Mois auquel le passage à l'heure d'été commence. Plage : 0 - 11.	entier Lecture seule
DST Day	Jour de la semaine auquel le passage à l'heure d'été commence	entier Lecture seule
DST Hour	Heure à laquelle le passage à l'heure d'été commence	entier Lecture seule
DST Minute	Minute à laquelle le passage à l'heure d'été commence	entier Lecture seule

Toutes les variables sont en lecture seule ; il n'est pas possible de les utiliser pour mettre à jour l'horloge système.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Élément	Description

Variables de périphériques

Nom de variable	Description	Type de données
Available System Memory	Mémoire RAM libre en octets	uint64 Lecture seule
Backlight Time	Temps d'activation en heures du rétroéclairage de l'affichage depuis la fabrication du périphérique	entier non signé Lecture seule
Battery LED	Active/désactive le voyant de batterie faible de la DEL (si disponible) **0** = désactivé **1** = activé Ce n'est pas disponible sur les plateformes Linux (consultez "Capacités des périphériques IHM" Page 541 pour plus de détails sur les panneaux)	entier
Battery Timeout	Réservé à Ce n'est pas disponible sur les plateformes Linux (consultez "Capacités des périphériques IHM" Page 541 pour plus de détails sur les panneaux)	entier
Display Brightness	Réinitialise et adapte le niveau de luminosité. Lorsque le rétroéclairage est réglé sur un niveau de luminosité faible (0…3), il reste allumé à un niveau plus élevé pendant 8 secondes afin de permettre à l'utilisateur d'effectuer les réglages, puis il s'éteint. Même lorsqu'il est réglé sur 0, le rétroéclairage reste allumé et le compteur Durée de rétroéclairage augmente. Plage 0-255	entier
External Timeout	Délai d'inactivité après lequel le rétroéclairage de l'écran s'éteint automatiquement. Le rétroéclairage s'allume automatiquement dès	entier

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Élément	Description

Variables de périphériques

Nom de variable	Description	Type de données
	que l'utilisateur touche l'écran. éteint le rétroéclairage et désactive les commandes tactiles (éteindre l'affichage). Le compteur **Backlight Time** s'arrête. **0** = Allumer le rétroéclairage (allume l'affichage) **1..n** = délai d'attente, en secondes, pour éteindre le rétroéclairage (minuteur de l'écran de veille) Sur le périphérique Linux (voir "Capacités des périphériques IHM" Page 541 pour les détails du panneau), délai d'attente géré en quelques minutes. La valeur saisie est convertie en minutes arrondies à la valeur suivante, par exemple, 60, 120, 180.	
Flash Free Space	Espace flash libre restant dans la mémoire flash interne	uint64 Lecture seule
Manufacturer Code	Numéro de code identifiant l'IHM	court Lecture seule
System RAM Usage	Mémoire RAM actuelle utilisée par l'IHM, exprimée en octets	uint64 Lecture seule
System Font List	Liste des polices du système	chaîne Lecture seule
Mode système	État de service du runtime. **1** = démarrage en cours **2** = mode de configuration **3** = mode opérationnel	entier

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Élément	Description

Variables de périphériques

Nom de variable	Description	Type de données
	4 = redémarrage **5** = arrêt	
Temps de montée du système	Durée pendant laquelle le système a été alimenté depuis la fabrication de l'unité (en heures)	entier non signé Lecture seule

Variables informations de vidage

Nom de variable	Description	Type de données
Message d'erreur Vidage	Retour du message d'erreur si une erreur se produit pendant l'opération de vidage	chaîne Lecture seule
Statut du vidage d'archive	**0** = état initial par défaut **1** = opération déclenchée **2** = opération terminée avec succès **3** = opération terminée avec erreurs	entier Lecture seule
Statut du vidage de recette	**0** = état initial par défaut **1** = opération déclenchée **2** = opération terminée avec succès **3** = opération terminée avec erreurs	entier Lecture seule
Statut du vidage de tendance	**0** = état initial par défaut **1** = opération déclenchée **2** = opération terminée avec succès **3** = opération terminée avec erreurs	entier Lecture seule
Statut de la réinitialisation de recette	**0** = état initial par défaut **1** = opération déclenchée **2** = opération terminée avec succès **3** = opération terminée avec erreurs	entier Lecture seule
Statut de la restauration de recette	**0** = état initial par défaut **1** = opération déclenchée **2** = opération terminée avec succès **3** = opération terminée avec erreurs	entier Lecture seule

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

<table>
<tr><th>Élément</th><th>Description</th></tr>
<tr><td></td><td>
<table>
<tr><th colspan="3">Variables réseau0</th></tr>
<tr><th>Nom de variable</th><th>Description</th><th>Type de données</th></tr>
<tr><td>Gateway</td><td>Adresse de passerelle de l'interface Ethernet principale de l'IHM</td><td>chaîne
Lecture seule</td></tr>
<tr><td>IP Address</td><td>Adresse IP de l'interface Ethernet principale de l'IHM</td><td>chaîne
Lecture seule</td></tr>
<tr><td>Mac ID</td><td>ID Mac de l'interface Ethernet principale de l'IHM</td><td>chaîne
Lecture seule</td></tr>
<tr><td>Paramètres de l'adaptateur réseau</td><td>Chaîne JSON qui permet de lire ou de mettre à jour les paramètres de l'adaptateur réseau</td><td>chaîne</td></tr>
<tr><td>Statut du réseau</td><td>Contient les résultats de la dernière opération requise par l'écriture dans les Paramètres de l'adaptateur. Cette valeur est actualisée après chaque opération d'écriture.
Si la chaîne est vide, cela signifie qu'il n'y a aucune erreur
Descriptions des erreurs les plus récentes</td><td>chaîne
Lecture seule</td></tr>
<tr><td>Subnet Mask</td><td>Masque de sous réseau de l'interface Ethernet principale de l'IHM</td><td>chaîne
Lecture seule</td></tr>
</table>
<table>
<tr><th colspan="3">Variables écran</th></tr>
<tr><th>Nom de variable</th><th>Description</th><th>Type de données</th></tr>
<tr><td>X Screen resolution</td><td>Affiche la dimension horizontale de l'écran en pixels</td><td>entier
Lecture seule</td></tr>
<tr><td>Y Screen resolution</td><td>Affiche la dimension verticale de l'écran en pixels</td><td>entier
Lecture seule</td></tr>
</table>
</td></tr>
</table>

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

<table>
<tr><th>Élément</th><th>Description</th></tr>
<tr><td></td><td>
<table>
<tr><th colspan="3">Variables carte SD</th></tr>
<tr><th>Nom de variable</th><th>Description</th><th>Type de données</th></tr>
<tr><td>SD Card FreeSpace</td><td>Espace de mémoire libre sur la carte en octets</td><td>uint64
Lecture seule</td></tr>
<tr><td>SD Card Name</td><td>Nom de la carte SD</td><td>chaîne
Lecture seule</td></tr>
<tr><td>SD Card Size</td><td>Tailles en octets de la carte enfichée dans le port</td><td>uint64
Lecture seule</td></tr>
<tr><td>SD Card Status</td><td>0 = carte SD débranchée
1 = carte SD branchée</td><td>entier
Lecture seule</td></tr>
</table>
<table>
<tr><th colspan="3">Variables serveur</th></tr>
<tr><th>Nom de variable</th><th>Description</th><th>Type de données</th></tr>
<tr><td>Nom de page</td><td>Nom de la page actuellement affichée</td><td>chaîne
Lecture seule</td></tr>
<tr><td>Current project</td><td>Nom du projet actuel</td><td>chaîne
Lecture seule</td></tr>
<tr><td>Project load time</td><td>Date de chargement du projet sur le HMI Runtime au format Date système (millisecondes).</td><td>uint64
Lecture seule</td></tr>
<tr><td>Last operating mode start time</td><td>Secondes écoulées depuis que le périphérique a démarré en mode opérationnel</td><td>uint64
Lecture seule</td></tr>
</table>

Toutes les variables se réfèrent au serveur, non à HMI Client.
</td></tr>
</table>

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Élément	Description

Variables d'heure

Nom de variable	Description	Type de données
Jour du mois	Plage : 1-31	entier
Jour de la semaine	Plage : 0 = Dimanche, .. , 6 = Samedi	entier
Heure	Plage : 0-23	entier
Minute	Plage : 0-59	entier
Mois	Plage : 1-12	entier
Seconde	Plage : 0-59	entier
Heure du système	Identique à l'heure UTC. Il est également possible de définir cette variable au format date/heure.	entier non signé
Année	Année en cours	entier

Heure système au format UTC

Variables lecteur USB

Nom de variable	Description	Type de données
USB Drive FreeSpace	Espace de mémoire libre en octets	uint64 Lecture seule
USB Drive Name	Nom du périphérique USB	chaîne Lecture seule
USB Drive Size	Taille en octets du lecteur branché au port USB	uint64 Lecture seule
USB Drive Status	**0** = lecteur USB débranché **1** = lecteur USB branché	entier Lecture seule

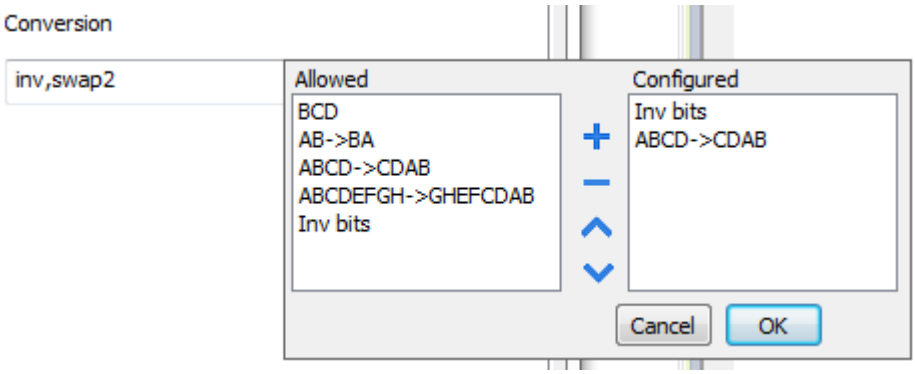
Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

<table>
<tr><th>Élément</th><th>Description</th></tr>
<tr><td></td><td>
<table>
<tr><th colspan="3">Variables de version</th></tr>
<tr><th>Nom de variable</th><th>Description</th><th>Type de données</th></tr>
<tr><td>Main OS Version</td><td>Version principale du SE</td><td>chaîne
Lecture seule</td></tr>
<tr><td>Runtime Version</td><td>Version du runtime</td><td>chaîne
Lecture seule</td></tr>
</table>
<table>
<tr><th colspan="3">Variables du commutateur de communication virtuelle</th></tr>
<tr><th>Nom de variable</th><th>Description</th><th>Type de données</th></tr>
<tr><td>VCS Status</td><td>Indique le statut du périphérique VCS.
0 = Service activé
1 = Client connecté en mode intercalé
2 = Client connecté en mode exclusif
3 = Service désactivé (par défaut)</td><td>octet non signé
Lecture seule</td></tr>
<tr><td>VCS disable</td><td>Permet la désactivation manuelle du périphérique VCS.
0 = Service VCS activé
1 = Service VCS désactivé (par défaut)</td><td>booléen</td></tr>
<tr><td>Port VCS</td><td>Indique le port d'écoute en cours TCP sur l'IHM par le service VCS</td><td>court non signé</td></tr>
</table>
</td></tr>
<tr><td>Type de données</td><td>Chaque variable système a un type de données spécifique, décrit dans les tableaux ci-dessus.
Le tableau suivant présente les détails de tous les types de données utilisés pour les variables système.</td></tr>
</table>

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

<table>
<tr><th>Élément</th><th>Description</th></tr>
<tr><td></td><td>
<table>
<tr><th>Type de données</th><th>Espace mémoire</th><th>Limites</th></tr>
<tr><td>court</td><td>Donnée à 16 bits</td><td>-32768 ... 32767</td></tr>
<tr><td>entier</td><td>Donnée à 32 bits</td><td>-2.1e9 ... 2.1e9</td></tr>
<tr><td>octet non signé</td><td>Donnée à 8 bits</td><td>0 ... 255</td></tr>
<tr><td>court non signé</td><td>Donnée à 16 bits</td><td>0 ... 65535</td></tr>
<tr><td>entier non signé</td><td>Donnée à 32 bits</td><td>0 ... 4.2e9</td></tr>
<tr><td>uint64</td><td>Donnée à 64 bits</td><td>0 ... 1.8e19</td></tr>
<tr><td>chaîne</td><td colspan="2">Tableau des éléments contenant du code de caractères défini par l'encodage sélectionné</td></tr>
</table>
</td></tr>
<tr><td>Taille de tableau</td><td>En cas de variable de chaîne, cette propriété représente le nombre maximum d’octets disponible pour la variable de chaîne.

Remarque : le nombre d'octets correspond au nombre de caractères de chaîne si la propriété Encoding est définie sur UTF-8 ou Latin1 dans l'Éditeur de variables.
Si la propriété Encoding est définie sur UCS-2BE, UCS-2LE, UTF-16BE ou UTF-16LE, un caractère nécessite 2 octets.</td></tr>
<tr><td>Conversion</td><td>Conversion à appliquer à la variable.

En fonction du type de données choisi, la liste Autorisée affiche un ou plusieurs types de conversion.
<table>
<tr><th>Valeur</th><th>Description</th></tr>
<tr><td>Inv bits</td><td>inv : inverse tous les bits de la variable.

Exemple :
1001 → 0110 (format binaire)
9 → 6 (format décimal)</td></tr>
<tr><td>Inverser</td><td>neg : inverse la valeur de la variable.</td></tr>
</table>
</td></tr>
</table>

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Élément	Description

Valeur	Description
	Exemple : 25.36 → -25.36
AB -> BA	**swapnibbles** : échange les quartets en un octet. *Exemple :* 15D4 → 514D (format hexadécimal) 5588 → 20813 (format décimal)
ABCD -> CDAB	**swap2** : échange les octets en un mot. *Exemple :* 9ACC → CC9A (format hexadécimal) 39628 → 52378 (format décimal)
ABCDEFGH -> GHEFCDAB	**swap4** : échange les octets en un mot double. *Exemple :* 32FCFF54 → 54FFFC32 (format hexadécimal) 855441236 → 1426062386 (format décimal)
ABC...NOP -> OPM...DAB	**swap8** : échange les octets en un mot long. Exemple : 142,366 → -893553517,588905 (format décimal) 0 10000000110 0001110010111011011001000101101000011100101011000001 → 1 10000011100 1010101000010100010110110110110010110110000010011110 1 (format binaire)
BCD	**bcd** : sépare l'octet en deux quartets, les lit au format décimal (de 0 à 9) *Exemple :* 23 → 17 (format décimal) 0001 0111 = 23 0001 = 1 (premier quartet) 0111 = 7 (second quartet)

Sélectionnez la conversion et cliquez sur le bouton +. La conversion sélectionnée s'affiche dans la liste **Configurée**.

Si vous configurez d'autres conversions, celles-ci s'affichent dans l'ordre (du haut vers le bas de la liste **Configurée**).

Utilisez les boutons de direction pour changer l'ordre des conversions configurées.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Variables de services

Les variables de services donnent la possibilité de lire l'état et de transmettre des commandes au serveur VNC.

Paramètres de l'éditeur de protocole

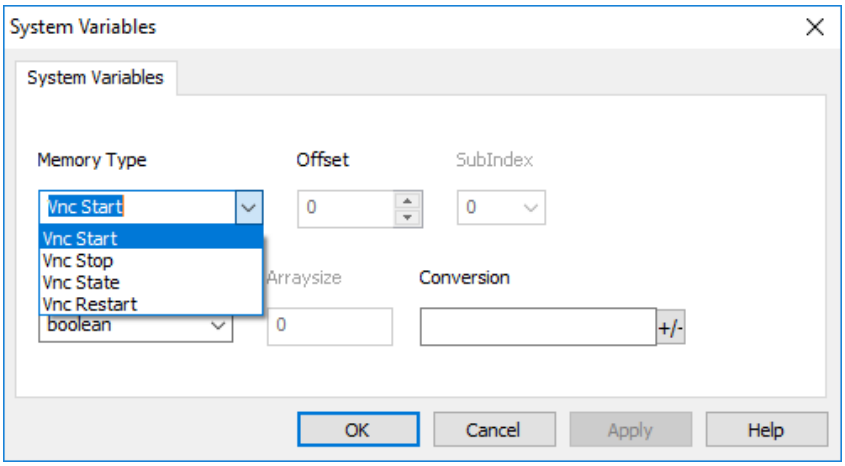
*Chemin : **VueProjet**> **Protocoles***

1. Cliquez sur **+** et sélectionnez **Variables système** : la boîte de dialogue **Variables système** s'affiche.
2. Sélectionnez **Services** dans la liste **PLCModels**.

Paramètres de l'éditeur de variables

*Chemin : **VueProjet**> **Configuration** > double-cliquez sur **Variables***

1. Pour ajouter une variable, cliquez sur **+** : une nouvelle ligne est ajoutée.
2. Sélectionnez **Variables système** dans la liste **Driver** : le dialogue de définition de variables s'affiche.



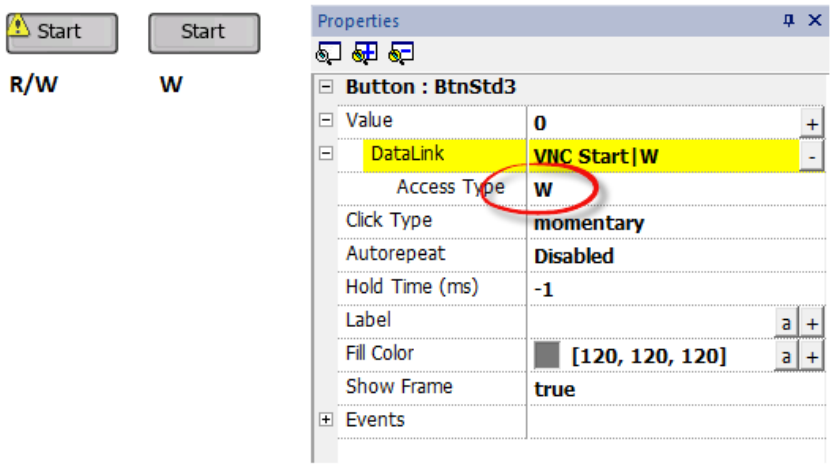
Les variables d'état VNC ne sont prises en charge que par les périphériques Linux ayant un BSP version 1.0.344 ou plus. (Voir x"Capacités des périphériques IHM" Page 541x)

Élément	Description	Type de données
VNC Start	Écrivez 1 à l'intérieur de cette variable pour forcer le démarrage du serveur VNC. C'est une variable en écriture seule, la commande sera exécutée à chaque fois que vous la réécrivez.	booléen Écriture seule
VNC Stop	Écrivez 1 à l'intérieur de cette variable pour forcer l'arrêt du serveur VNC. C'est une variable en écriture seule, la commande sera exécutée à chaque fois que vous la réécrivez.	booléen Écriture seule
VNC Restart	Écrivez 1 à l'intérieur de cette variable pour forcer le redémarrage du serveur VNC. C'est une variable en écriture seule, la commande sera	booléen Écriture seule

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Élément	Description	Type de données
	exécutée à chaque fois que vous la réécrivez.	
VNC State	Statut du serveur VNC 0 INACTIF 1 DÉMARRAGE EN COURS 10 FONCTIONNEMENT EN COURS 100 CONNEXION EN COURS 200 CONNECTÉ -1 ERREUR	entier Lecture seule

Les variables en écriture seulement ne peuvent pas être lues. Assurez-vous de ne pas utiliser le mode d'accès L/É pour éviter l'icône d'erreur de lecture.



Variables de mémoire rémanente

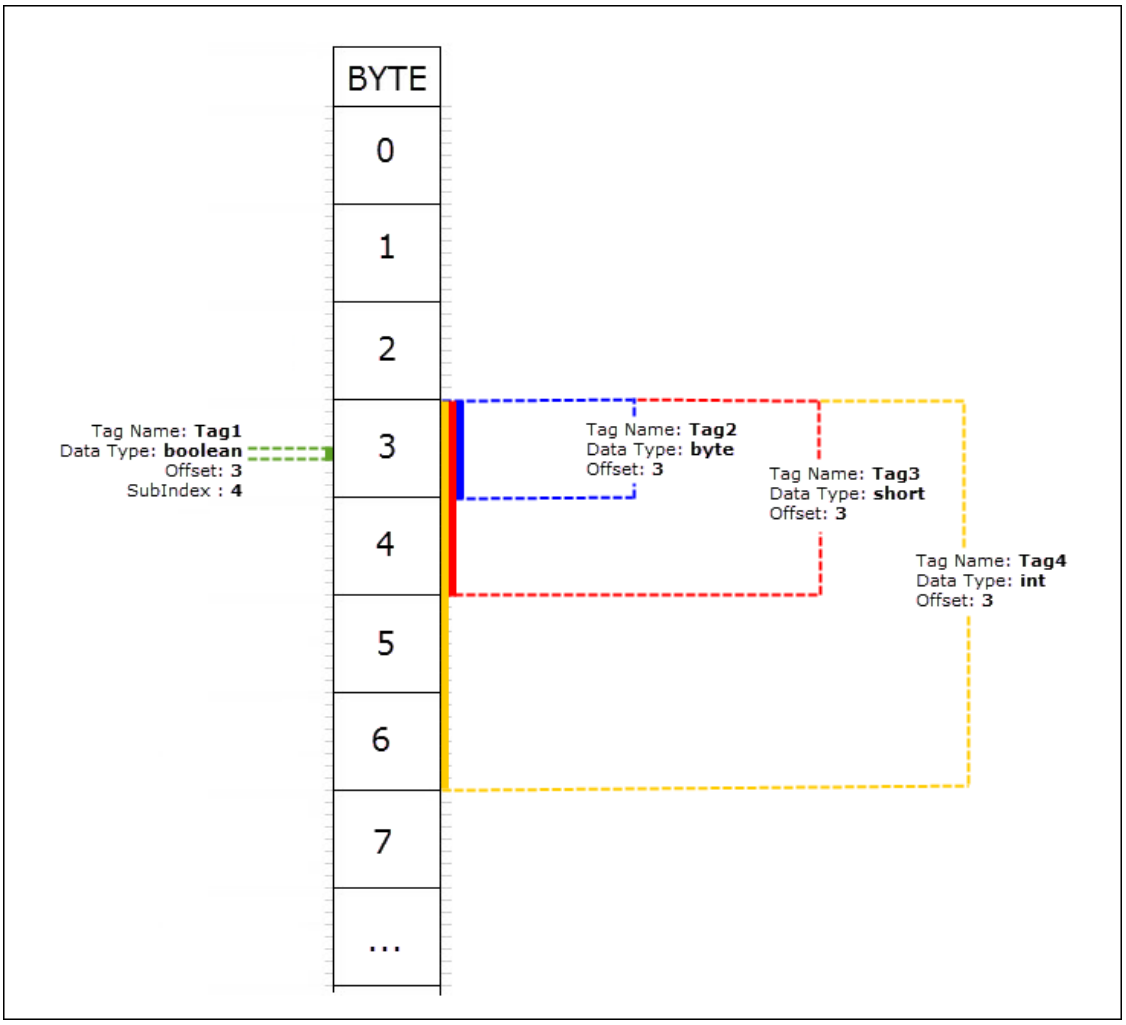
Variables système - Le protocole Mémoire rémanente permet de créer des variables pointant sur une zone de mémoire dont le contenu est maintenu lorsque l'IHM est hors tension.

Le support physique de la mémoire rétentive est basé sur la technologie FRAM.

Important : si la mémoire FRAM n'est pas disponible, la rémanence est prise en charge en utilisant la mémoire de l'utilisateur (Flash ou disque dur). La technologie Flash est limitée dans le nombre maximum d'opérations d'écriture. L'utilisation de Flash comme support de stockage pour la mémoire rémanente avec des opérations d'écriture fréquentes peut endommager les composants de la mémoire. Vérifier la disponibilité de la mémoire FRAM dans les données du périphérique IHM.

Important : la mémoire rémanente est une zone de mémoire linéaire de 16 Ko organisée en octets et accessible par un décalage.

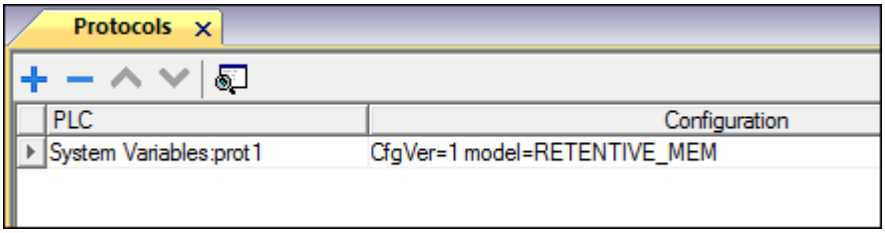
Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



Mémoire rémanente vs stockage des recettes
Les données de recette sont sauvegardées dans la mémoire flash (sauf pourPB610 PC Runtime) tandis que les données rémanentes sont sauvegardées dans un FRAM. La mémoire flash n'est pas adaptée à un nombre élevé d'opérations d'écriture, tandis que FRAM supporte un nombre pratiquement illimité d'opérations d'écriture et est préférable lorsque des opérations d'écriture fréquentes sont nécessaires.

Paramètres de l'éditeur de protocole

Dans la liste des modèles d'API de la boîte de dialogue Éditeur de protocole, sélectionnez Mémoire rémanente.

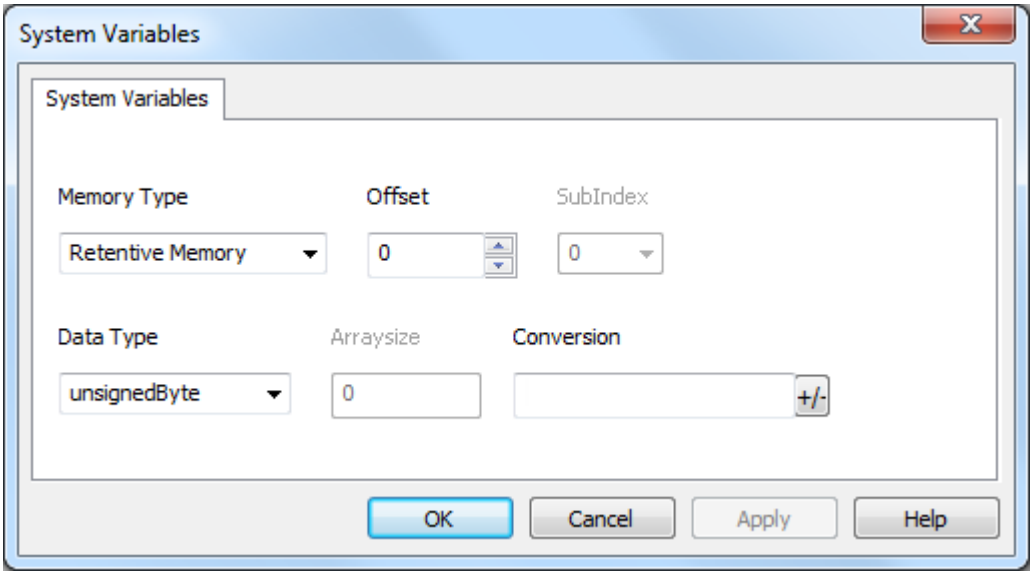


Paramètres de l'éditeur de variables

Chemin : ***VueProjet*** *>* ***Configuration*** *> double-cliquez sur* ***Variables***

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

1. Pour ajouter une variable, cliquez sur **+** : une nouvelle ligne est ajoutée.
2. Sélectionnez **Variables système** dans la liste **Driver** : le dialogue de définition de variables s'affiche.



Élément	Description
Type de mémoire	Fixée à la mémoire rémanente
Offset	Adresse de décalage où se situe la variable. Plage : 0-16383
Sousindex	Ce paramètre permet la sélection du décalage de ressources en fonction du Type de données sélectionné.
Type de données	

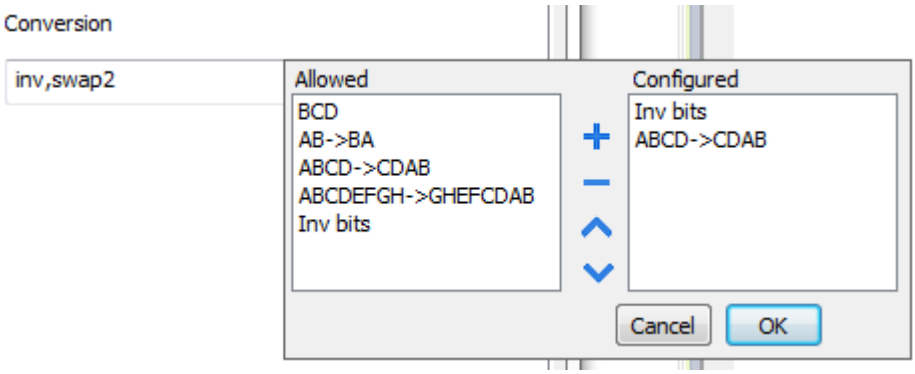
Type de données	Espace mémoire	Limites
booléen	Donnée à 1 bit	0 ... 1
octet	Donnée à 8 bits	-128 ... 127
court	Donnée à 16 bits	-32768 ... 32767
entier	Donnée à 32 bits	-2.1e9 ... 2.1e9
int64	Donnée à 64 bits	-9.2e18 ... 9.2e18
octet non signé	Donnée à 8 bits	0 ... 255
court non signé	Donnée à 16 bits	0 ... 65535
entier non signé	Donnée à 32 bits	0 ... 4.2e9
uint64	Donnée à 64 bits	0 ... 1.8e19
flottant	Type virgule flottante de 32 bits simple précision conforme IEEE	1.17e-38 ... 3.4e38

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

<table>
<tr><th>Élément</th><th>Description</th></tr>
<tr><td></td><td>
<table>
<tr><th>Type de données</th><th>Espace mémoire</th><th>Limites</th></tr>
<tr><td>double</td><td>Type virgule flottante de 64 bits double précision conforme IEEE</td><td>2.2e-308 ... 1.79e308</td></tr>
<tr><td>chaîne</td><td colspan="2">Tableau des éléments contenant du code de caractères défini par l'encodage sélectionné</td></tr>
<tr><td>binaire</td><td colspan="2">Données binaires arbitraires</td></tr>
</table>

Remarque : pour définir des tableaux, sélectionnez l’un des formats de type de données suivi de crochets tels que « byte[] », « short[] », etc.
</td></tr>
<tr><td>Taille de tableau</td><td>

En cas de variable vectorielle, cette propriété représente le nombre d’éléments vectoriels.
En cas de variable de chaîne, cette propriété représente le nombre maximum d’octets disponible pour la variable de chaîne.

Remarque : le nombre d'octets correspond au nombre de caractères de chaîne si la propriété Encoding est définie sur UTF-8 ou Latin1 dans l'Éditeur de variables.
Si la propriété Encoding est définie sur UCS-2BE, UCS-2LE, UTF-16BE ou UTF-16LE, un caractère nécessite 2 octets.
</td></tr>
<tr><td>Conversion</td><td>
Conversion à appliquer à la variable.


En fonction du type de données choisi, la liste Autorisée affiche un ou plusieurs types de conversion.
<table>
<tr><th>Valeur</th><th>Description</th></tr>
<tr><td>Inv bits</td><td>inv : inverse tous les bits de la variable.

Exemple :
1001 → 0110 (format binaire)
9 → 6 (format décimal)</td></tr>
<tr><td>Inverser</td><td>neg : inverse la valeur de la variable.

Exemple :</td></tr>
</table>
</td></tr>
</table>

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Élément	Description

Valeur	Description
	25.36 → -25.36
AB -> BA	**swapnibbles** : échange les quartets en un octet. *Exemple :* 15D4 → 514D (format hexadécimal) 5588 → 20813 (format décimal)
ABCD -> CDAB	**swap2** : échange les octets en un mot. *Exemple :* 9ACC → CC9A (format hexadécimal) 39628 → 52378 (format décimal)
ABCDEFGH -> GHEFCDAB	**swap4** : échange les octets en un mot double. *Exemple :* 32FCFF54 → 54FFFC32 (format hexadécimal) 855441236 → 1426062386 (format décimal)
ABC...NOP -> OPM...DAB	**swap8** : échange les octets en un mot long. Exemple : 142,366 → -893553517,588905 (format décimal) 0 10000000110 00011100101110110110010001011010000111001010110000001 → 1 10000011100 10101010000101000101101101101100101101100001001111101 (format binaire)
BCD	**bcd** : sépare l'octet en deux quartets, les lit au format décimal (de 0 à 9) *Exemple :* 23 → 17 (format décimal) 0001 0111 = 23 0001 = 1 (premier quartet) 0111 = 7 (second quartet)

Sélectionnez la conversion et cliquez sur le bouton +. La conversion sélectionnée s'affiche dans la liste **Configurée**.

Si vous configurez d'autres conversions, celles-ci s'affichent dans l'ordre (du haut vers le bas de la liste **Configurée**).

Utilisez les boutons de direction pour changer l'ordre des conversions configurées.

.

Nettoyer la mémoire rémanente

Utilise l'action **ClearRetentiveMemory** pour nettoyer le contenu de la mémoire rémanente.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Conseil : utilisez cette action pour définir le contenu de la mémoire à un statut connu à tout moment.

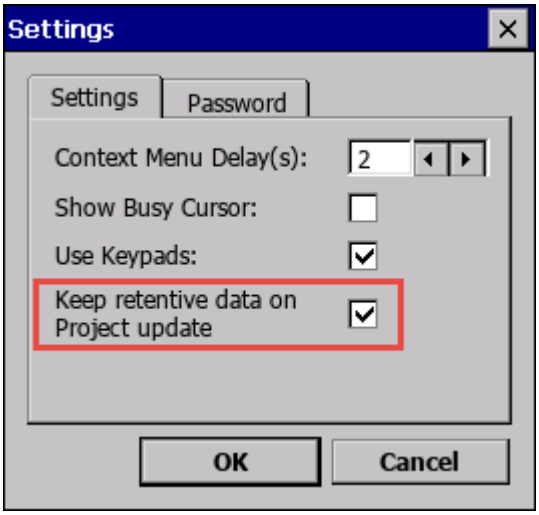
Voir la section *Actions* > *Tag Action* du manuel principal pour les détails.

L'interface JavaScript pour cette action est :
project.clearRetentiveMemory();

Préserver la mémoire rémanente lors du téléchargement du projet

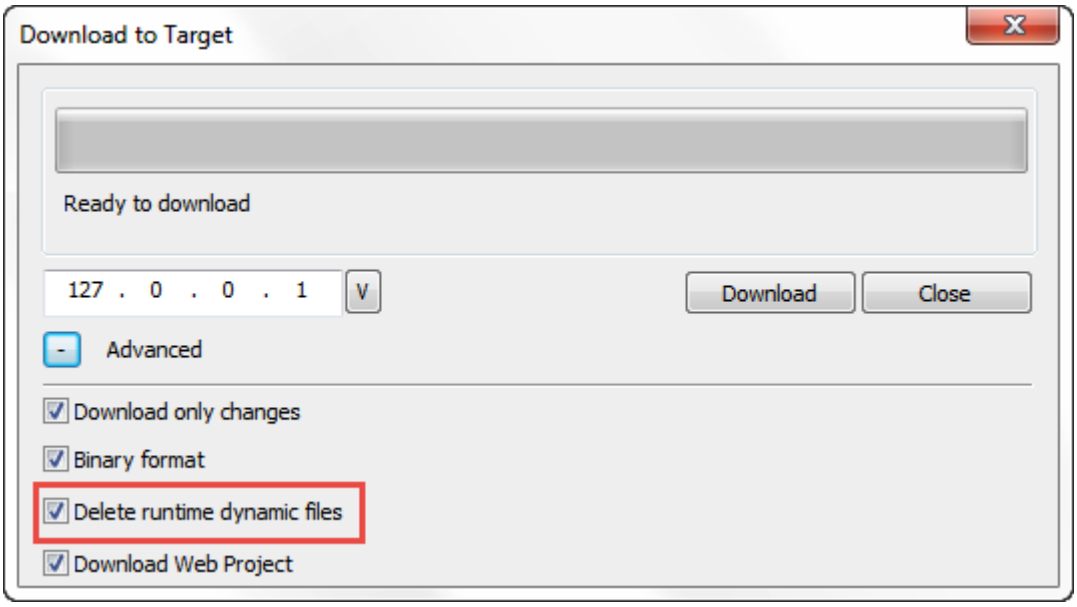
Lorsqu'un fichier de projet est téléchargé vers une IHM, ou lorsque le projet actif est modifié, le contenu de la mémoire rémanente est généralement supprimé.

Si cela est nécessaire pour préserver le contenu des données rémanente lors du téléchargement ou de la mise à jour du projet, sélectionnez l'option **Conserver les données rémanente lors de la mise à jour du projet** dans les onglets de configuration de l'IHM.



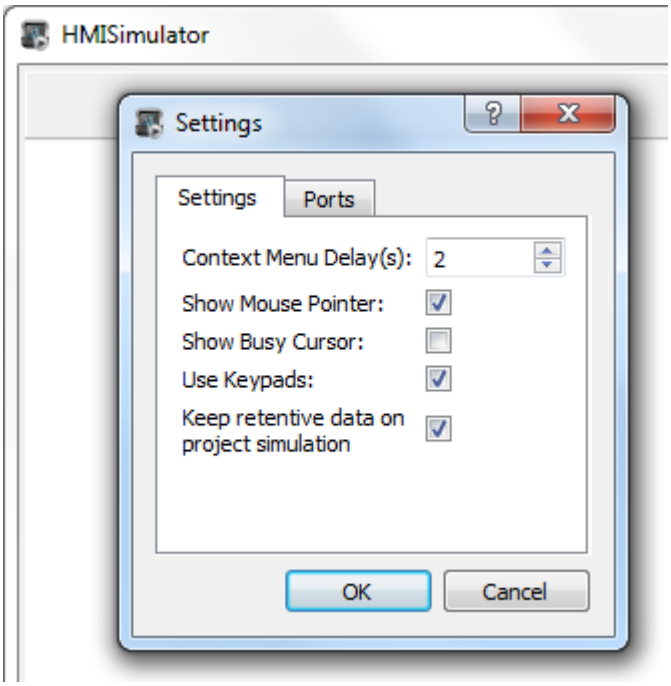
Ce paramètre sera ignoré si l'option **Supprimer les fichiers dynamiques de runtime** est sélectionnée dans la fenêtre *Télécharger vers la cible*.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



Préserver la mémoire rémanente dans le simulateur

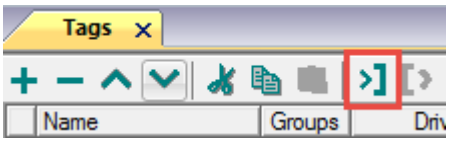
Le simulateur du PB610 Panel Builder 600 prend en charge la mémoire rémanente. Pour activer la mémoire rémanente pendant la simulation de projet, utilisez l'option « Conserver les données rémanentes sur la simulation de projet » dans le menu contextuel.



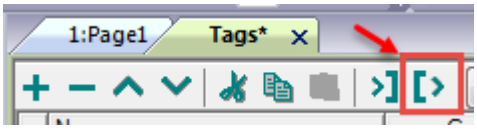
Importation de variables

Sélectionnez le pilote dans l'éditeur de variables et cliquez sur le bouton **Importer variables** pour lancer le pilote d'importation.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

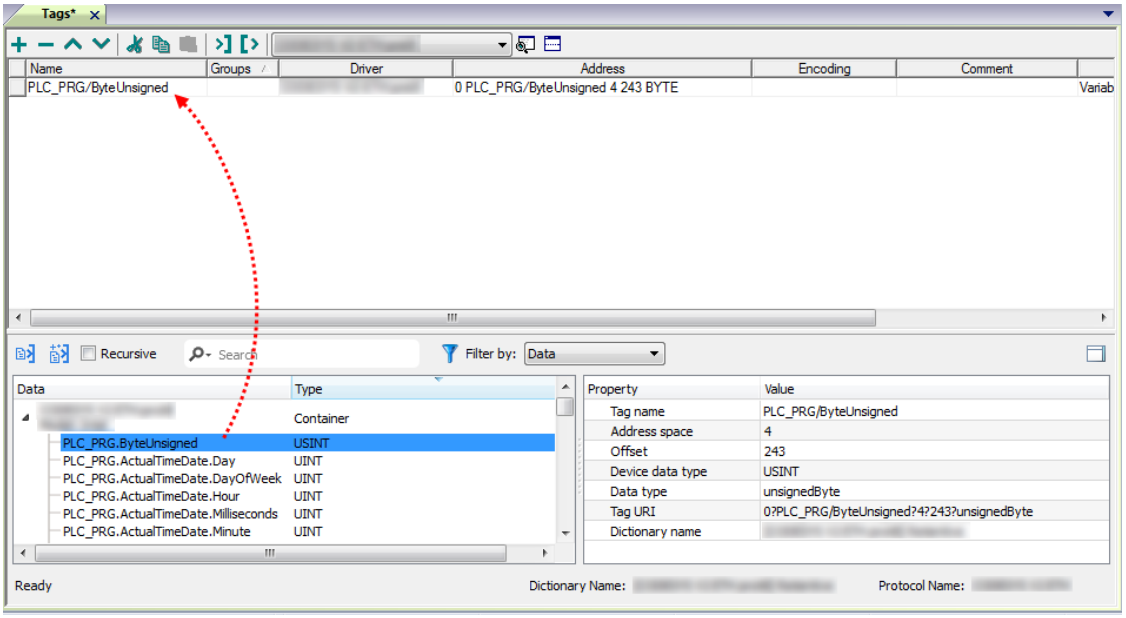


Le système aura besoin d'un fichier XML générique exporté de l'éditeur de variables avec le bouton approprié.



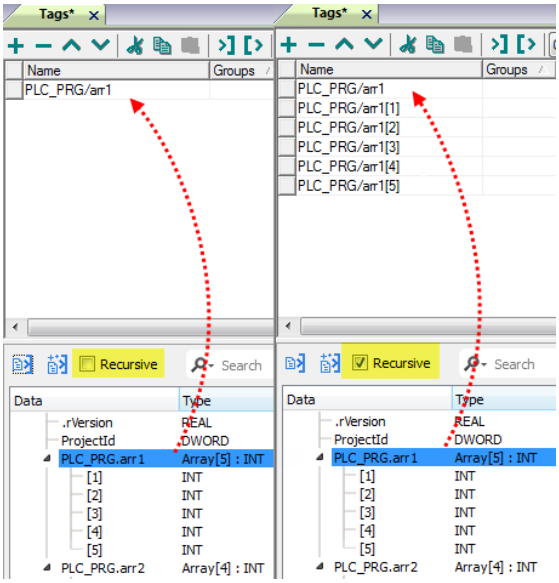
Lorsque vous avez sélectionné le pilote d'importation, localisez le fichier de symboles et cliquez sur **Ouvrir**.

Les variables comprises dans le fichier de symboles sont listées dans le dictionnaire des variables. Le dictionnaire des variables est affiché en bas de l'écran.



Élément de la barre d'outils	Description
	Importer variable(s). Sélectionnez les variables à importer et cliquez sur cette icône pour ajouter les variables du dictionnaire au projet.
	Mettre à jour variable(s). Cliquez sur cette icône pour mettre à jour les variables du projet grâce à un nouvel import du dictionnaire.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Élément de la barre d'outils	Description
Recursive	Cochez cette case pour importer tous les sous-éléments d'une variable. Exemples de résultats avec la case cochée et non cochée : 
Search Filter by: Tag name	Recherche des variables dans le dictionnaire en filtrant en fonction de l'élément sélectionné dans la liste déroulante.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

11 Actions

Les actions sont des fonctions permettant d'interagir avec le système. Elles sont exécutées en général lors du déclenchement d'événements.

Divers widgets peuvent déclencher des événements, par exemple, en appuyant et en relâchant un bouton. Toutes les actions ne sont pas disponibles pour tous les événements d'un objet.

Les actions sont associées à des widgets dans la section **Événements** du volet Propriétés (Éditeur de pages).

Actions d'alarmes **140**

Actions de base de données **141**

Actions d'événements **144**

Actions multilangues **144**

Actions clavier **145**

Actions lecteurs médias **147**

Actions FTP **147**

Actions sur pages **150**

Actions d'impression **154**

Actions sur une recette **156**

Actions client à distance **160**

Actions système **161**

Actions de variables **172**

Actions de tendance **175**

Actions éditeur de texte **179**

Actions de gestion des utilisateurs **179**

Actions widget **182**

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Actions d'alarmes

Permet principalement d'accuser réception ou de réinitialiser des alarmes.

SelectAllAlarms

Sélectionne toutes les alarmes.

Paramètre	Description
Mode	**TOGGLE** Inverse le statut sélectionné. Les alarmes qui ne sont pas déclenchées ou qui n'ont pas de demande d'accusé de réception ou de réinitialisation en attente ne seront jamais sélectionnées. **SELECT** Sélectionne toutes les alarmes qui sont déclenchées ou qui ont une demande d'accusé de réception ou de réinitialisation en attente **UNSELECT** Désélectionne toutes les alarmes

SelectAlarm

Sélectionne une alarme spécifique.

Paramètre	Description
AlarmID	ID d'alarme
Sélection de drapeau	**TRUE** Sélectionne l'alarme. Les alarmes qui ne sont pas déclenchées ou qui n'ont pas de demande d'accusé de réception ou de réinitialisation en attente ne seront pas sélectionnées. **FALSE** Désélectionne une alarme.

AckAlarm

Accuse réception d'une alarme spécifique ou de toutes les alarmes sélectionnées.

Paramètre	Description
AlarmID	ID d'alarme spécifique **SELECTED** Toutes les alarmes sélectionnées

Accuse réception des alarmes sélectionnées.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

RéinitialiserAlarme

Réinitialise une alarme spécifique ou toutes les alarmes sélectionnées qui ne sont ni déclenchées, ni confirmées.

Paramètre	Description
AlarmID	ID d'alarme spécifique **SELECTED** Toutes les alarmes sélectionnées

EnableAlarms

Active ou désactive une alarme spécifique ou toutes les alarmes sélectionnées.

Les alarmes désactivées ne produiront pas d'événements d'alarme.

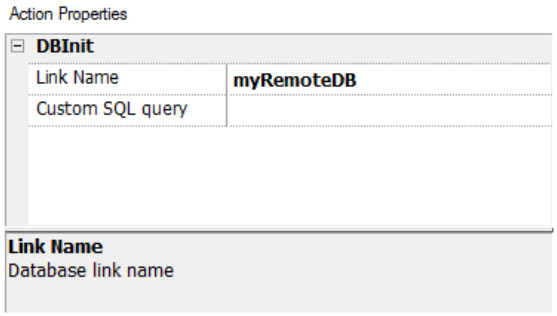
Paramètre	Description
AlarmID	ID d'alarme spécifique **SELECTED** Toutes les alarmes sélectionnées
Sélection de drapeau	**TRUE** Active le(s) alarme(s). **FALSE** Désactive le(s) alarme(s).

Actions de base de données

DBInit

Important : cette action est exécutée une seule fois sur une base de données vide. Il ne s'agit pas d'une commande d'initialisation à appeler à chaque démarrage du périphérique IHM.

Crée le jeu de tableaux requis par le projet. Vous n'avez pas besoin d'utiliser cette action si la base de données contient déjà les tableaux nécessaires.



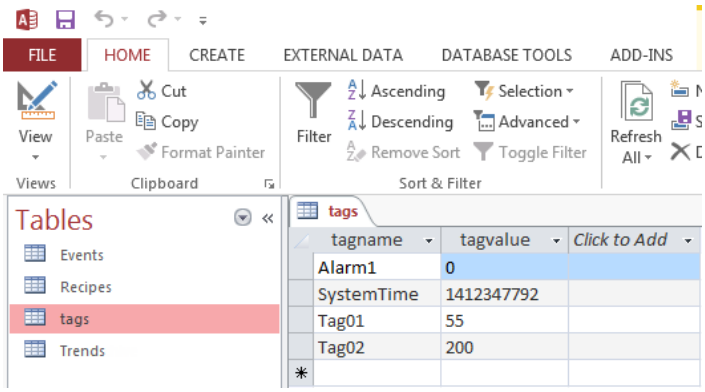
Utilisez le paramètre **Requête SQL personnalisée** pour définir les pages à créer. Laissez vide pour générer des noms de tableau par défaut

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Conseil : ajoutez cette commande à l'intérieur d'une page de montage de votre projet, utilisée par le personnel autorisé uniquement lors de la première installation de l'application.

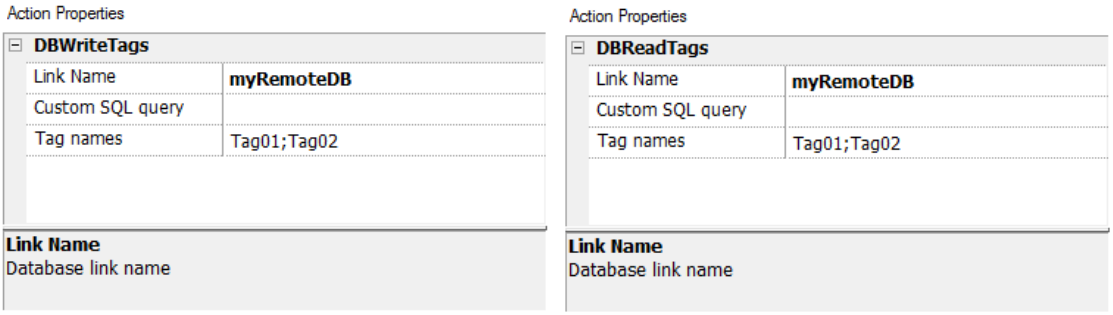
Interface JavaScript

```
project.dbInit(dbLinkName, sqlCustomQuery);
```



DBWriteTags, DBReadTags

Transfèrent les valeurs des variables sélectionnées vers/à partir de la base de données distante.



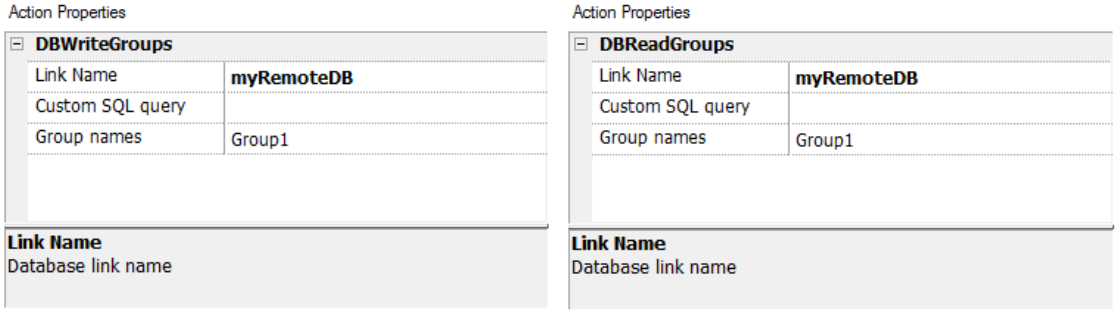
Interface JavaScript

```
project.dbWriteTags(dbLinkName, sqlCustomQuery, Tags);
```

```
project.dbReadTags(dbLinkName, sqlCustomQuery, Tags);
```

DBWriteGroups, DBReadGroups

Transfèrent des groupes de variables entre le périphérique IHM et la base de données.



Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

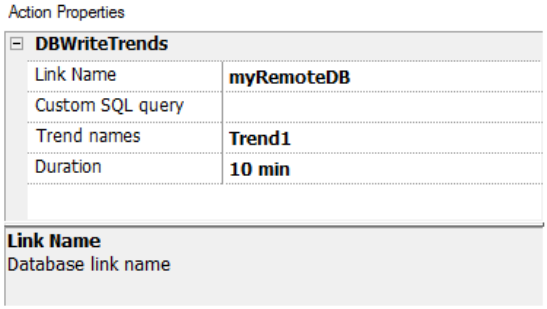
Interface JavaScript

```
project.dbWriteGroups(dbLinkName, sqlCustomQuery, Groups);
```

```
project.dbReadGroups(dbLinkName, sqlCustomQuery, Groups);
```

DBWriteTrend

Ajoute les valeurs des données échantillonnées les plus récentes dans la plage de temps sélectionnée dans le tableau des tendances de la base de données distante.

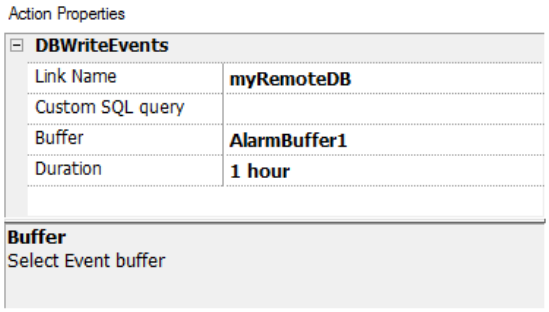


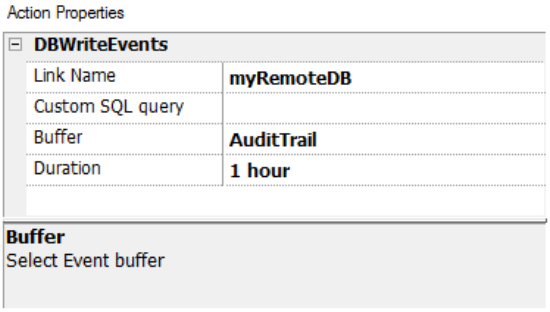
Interface JavaScript

```
project.dbWriteTrends(dbLinkName, sqlCustomQuery, trendName, durationIndex)
```

DBWriteEvents

Insère les valeurs des événements les plus récents dans la plage de temps sélectionnée dans le tableau des événements de la base de données distante.





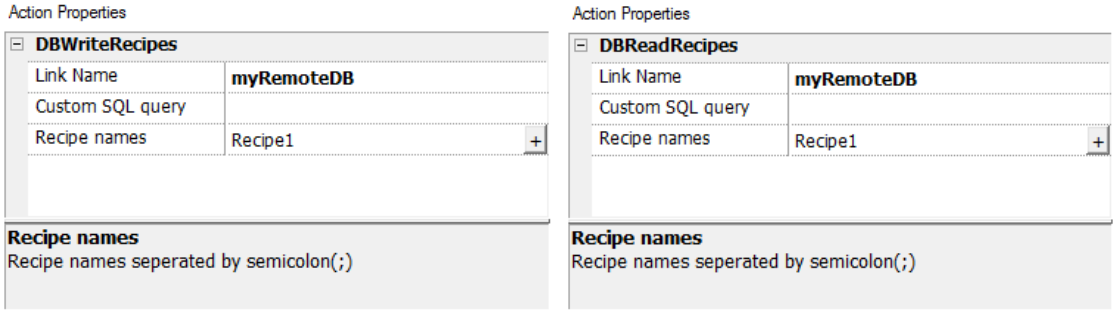
Interface JavaScript

```
project.dbWriteEvents (dbLinkName, sqlCustomQuery, archiveName, durationIndex)
```

DBWriteRecipes, DBReadRecipes

Transfèrent les données des recettes vers/à partir de la base de données distante.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



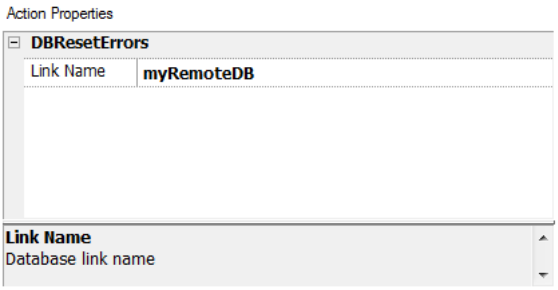
Interface JavaScript

```
project.dbWriteRecipes(dbLinkName, sqlCustomQuery, recipeNames)
```

```
project.dbReadRecipes(dbLinkName, sqlCustomQuery, recipeNames)
```

DBResetErrors

Réinitialise les trois variables de statut pour le lien de base de données sélectionné. Voir "Variables bases de données" Page 98.



Interface JavaScript

```
project.dbResetErrors(dbLinkName)
```

Actions d'événements

Utilisées par le widget de l'Historique des alarmes pour le défilement des événements/alarmes dans la vue de tableau (widget de mémoire d'événements).

FaireDéfilerÉvénementsArrière

Faire défiler les événements/actions en arrière dans la vue de tableau (widget de mémoire d'événements).

FaireDéfilerÉvénementsAvant

Faire défiler les événements/actions en avant dans la vue de tableau (widget de mémoire d'événements).

Actions multilangues

Sélectionne la langue de l'application.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

ConfigurerLangue

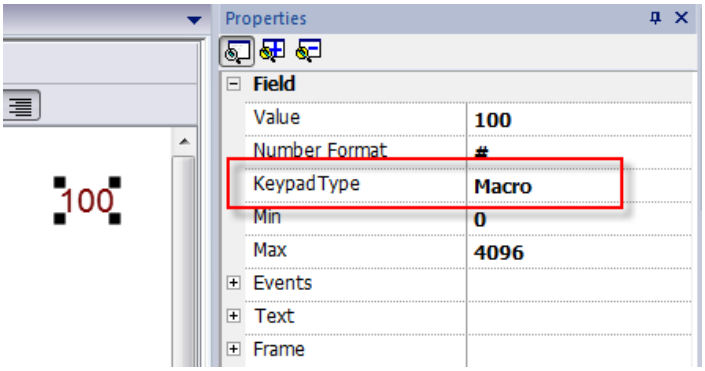
Définit la langue utilisée. La langue sélectionnée sera appliquée au runtime à tous les widgets concernés.

Actions clavier

Modifie les claviers utilisés.

SendKey

Envoie un caractère à un widget numérique. La propriété **TypeClavier** du widget numérique doit être définie sur **Macro**.



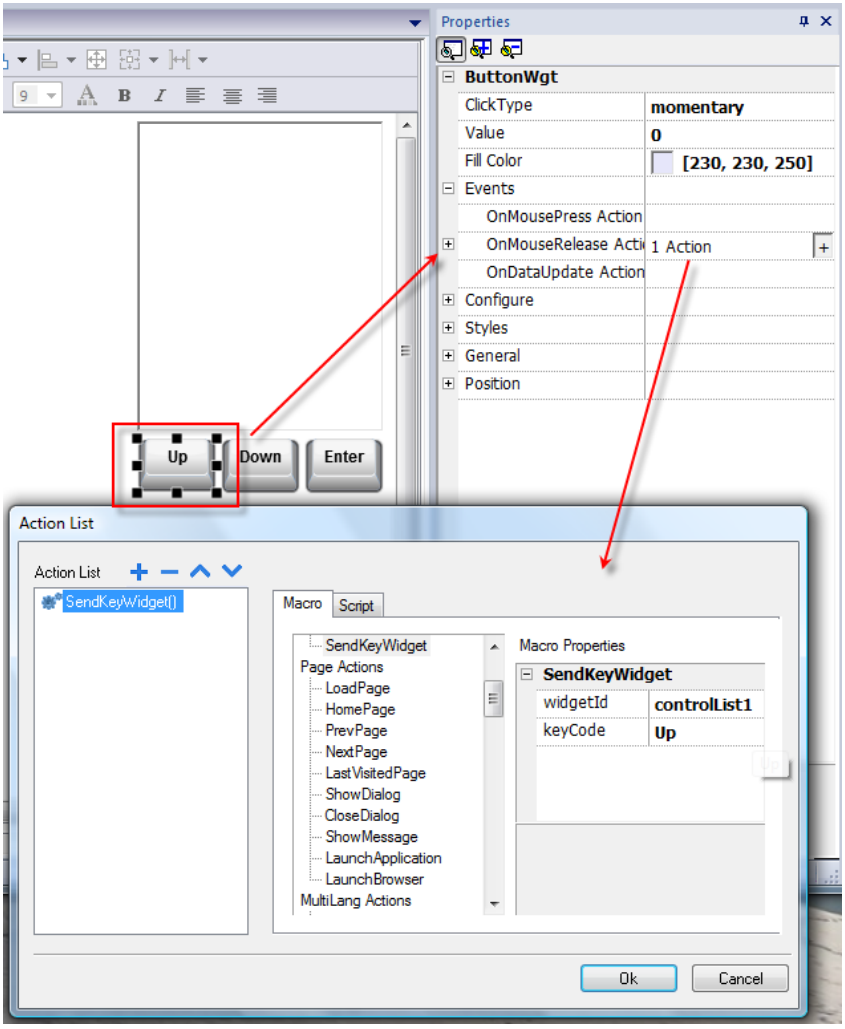
SendKeyWidget

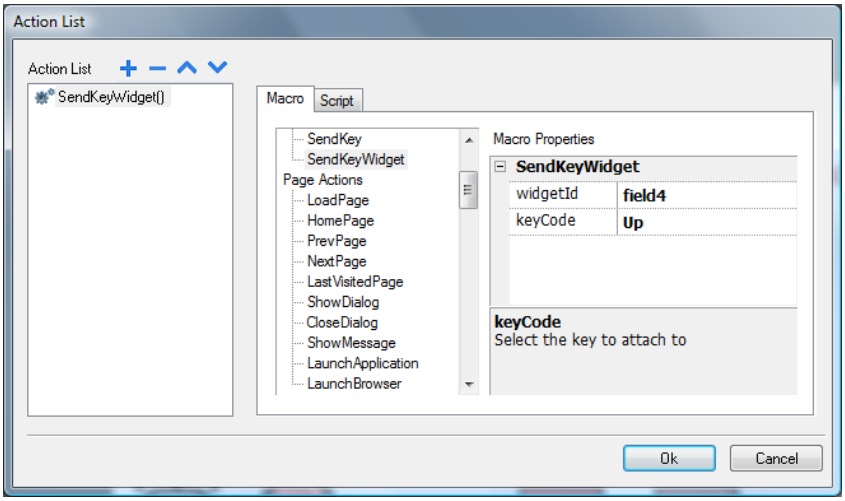
Envoie un caractère à un widget particulier.

Exemple

Les boutons **Haut** et **Bas** utilisent l'action **SendKeyWidget** en combinaison avec **Control List Widget**.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21





ShowKeyPad

Affiche le clavier tactile du système d'exploitation par défaut.

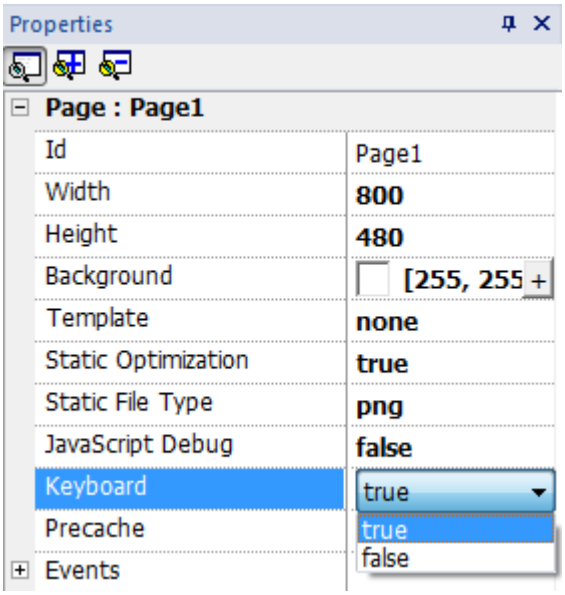
Remarque : peut ne pas être pris en charge par tous les systèmes d'exploitation.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Clavier

Active/désactive l'utilisation d'actions lors de l'utilisation de claviers externes. L'exécution des actions peut être activée/désactivée tant au niveau du projet qu'au niveau de la page.

L'effet est équivalent à l'utilisation de la propriété Clavier pour le projet et la page.



Actions lecteurs médias

Interagit avec le widget du lecteur de médias au runtime.

Action	Description
LireMédia	Lance la lecture de la vidéo.
FermerMédia	Arrête la lecture de la vidéo.
RechargerMédia	Reprend la lecture de la vidéo depuis le début.
PauseMédia	Pause la vidéo.
ParcourirMédia	Sélectionne la vidéo à lire.

Actions FTP

Utilisé pour le téléversement et le téléchargement de fichiers vers et depuis un serveur FTP distant.

ftpGET

Télécharge des fichiers depuis un serveur FTP distant

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Paramètre	Description
FtpConfig	Configuration des paramètres FTP
FtpRemoteFileName	Nom du fichier sur un serveur FTP à télécharger (source)
FtpLocalFileName	Nom du fichier sur un périphérique IHM local (destination)

ftpPUT

Charge les fichiers sur un serveur FTP distant

Paramètre	Description
FtpConfig	Configuration des paramètres FTP
FtpLocalFileName	Nom du fichier sur un périphérique IHM local (source)
FtpRemoteFileName	Nom du fichier sur un serveur FTP à télécharger (destination)

Les noms de fichier peuvent contenir des métacaractères.

Une fois transférées, les variables système sont mises à jour avec le statut des opérations en cours (voir "Variables client FTP" Page 103pour les détails).

Configuration serveur FTP

Pour configurer le paramètre FTP, entrez les informations suivantes pour le paramètre **FtpConfig** :

Paramètre	Description
Adresse FTP	Adresse IP du serveur FTP.
Port de serveur	Port pour connexion FTP (par défaut = 21).
Authentification	Choisissez l'authentification FTP à utiliser : • Normale (nom d'utilisateur et mot de passe requis) • Anonyme
Identifiant	Nom d'utilisateur du compte FTP distant
Mot de passe	Mot de passe du compte FTP distant

Cliquez sur **+** pour ajouter ajouter des configurations de serveurs FTP.

Conseil : utilisez les variables si vous voulez changer les paramètres du serveur de manière dynamique depuis le HMI Runtime.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Interface JavaScript FTP

ftpConfig

```
ftpCONFIG (IPAddress, Port, Authentication, UserName, Password)
```

Régle les paramètres FTP pour utiliser les prochains appels FTP

Paramètre	Description
Adresse IP	Adresse IP du serveur FTP.
Port	Port pour connexion FTP (par défaut = 21).
Authentification	Choisissez l'authentification FTP à utiliser : • Normale (nom d'utilisateur et mot de passe requis) • Anonyme
Identifiant	Nom d'utilisateur du compte FTP distant
Mot de passe	Mot de passe du compte FTP distant

ftpGET

```
ftpGET (remoteFileName, localFileName, [callback])
```

Télécharge des fichiers depuis un serveur FTP distant

Paramètre	Description
remoteFileName	Nom du fichier sur un serveur FTP à télécharger (source)
localFileName	Nom du fichier sur un périphérique IHM local (destination)
callback	Fonction qui sera appelée à la fin du transfert FTP

ftpPUT

```
ftpPUT (remoteFileName, localFileName, [callback])
```

Charge les fichiers sur un serveur FTP distant

Paramètre	Description
remoteFileName	Nom du fichier sur un serveur FTP à télécharger (source)
localFileName	Nom du fichier sur un périphérique IHM local (destination)
callback	Fonction qui sera appelée à la fin du transfert FTP

Exemple :

```
project.ftpCONFIG("192.168.0.200", "21", "true", "admin", "admin");
```

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

```
project.ftpGET( "data.txt",
                "\\USBMemory\\data.txt",
                function(ftpStatus) {fnFtpGetFinished(ftpStatus);} );

function fnFtpGetFinished(ftpStatus) {
       alert(ftpStatus);
}
```

Actions sur pages

Navigation sur pages. Les actions sur pages peuvent être utilisées avec les événements suivants :

- OnMouseClick,
- OnMouseRelease,
- OnMouseHold
- OnActivate
- OnDeactivate
- Alarmes
- Planificateurs.

LoadPage

Se rendre à la page sélectionnée du projet.

HomePage

Se rendre à la page d'accueil.

Vous pouvez configurer la page d'accueil dans la section **Comportement** du **Widget de projet**, voir "Projet" Page 76

PrevPage

Retourner à la page précédente.

NextPage

Passer à la page suivante.

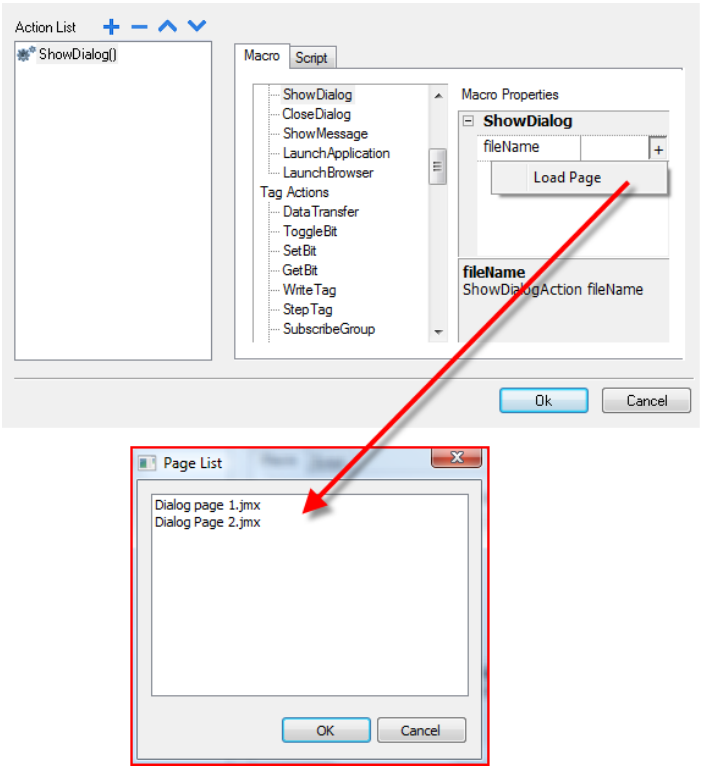
LastVisitedPage

Retourner à la page affichée précédemment

ShowDialog

Ouvrir une page de dialogue définie dans le projet.

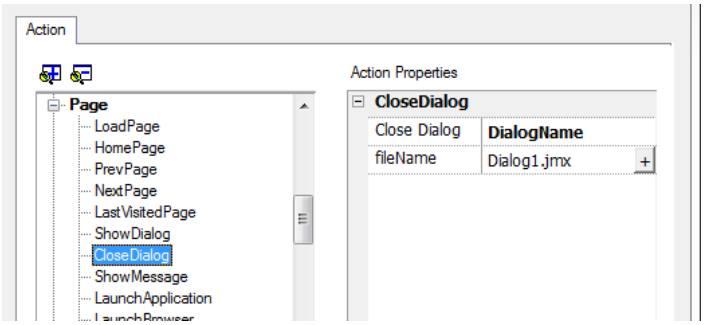
Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



CloseDialog

Fermer pages de dialogue.

Remarque : cette action s'applique uniquement aux pages de dialogue.



CloseDialog options

Option	Description
All	Ferme tous les dialogues ouverts
Selected	Ferme uniquement le dialogue actif
DialogName	Ferme le dialogue indiqué par la propriété **nomFichier**

Interface JavaScript

project.closeDialog(DialogID);

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Où *DialogID* :

All	Ferme tous les dialogues ouverts
Selected	Ferme uniquement le dialogue actif
DialogName.jmx	Ferme le dialogue indiqué par le paramètre **nomFichier**

Exemples

Exemple	Comportement
project.closeDialog("All");	Tous les dialogues ouverts sont fermés
project.closeDialog("Selected");	Le dialogue sélectionné est fermé
project.closeDialog("Dialog1.jmx");	Toutes les instances du Dialogue1 sont fermées

Sans paramètres, la fonction *project.closeDialog();* fonctionne comme *project.closeDialog("Selected");*.

ShowMessage

Affiche un message contextuel. Saisir le texte du message à afficher.

LaunchApplication

Lance une application externe.

Paramètre	Description
App Name	Nom exécutable avec extension (par exemple, « notepad.exe » pour exécuter Notepad)
Path	Chemin de l'application.
Arguments	Arguments spécifiques à l'application (par exemple, *\flash\qthmi\Manuel.pdf* pour ouvrir le document « Manuel.pdf »)
Instance unique	Argument pour démarrer l'application en une seule instance ou plusieurs instances. Lorsqu'une seule instance est sélectionnée, le système vérifie si l'application est déjà en cours d'exécution ; si oui, l'application est mise au premier plan, sinon, l'application est lancée.
FlushRuntimeCache	Efface tous les caches du runtime pour libérer le plus de ram possible avant de lancer l'application.

Remarque : les arguments contenant des espaces doivent être mis entre guillemets (par exemple, "\Carte de stockage\Manuel.pdf")

Exemple :

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

LaunchApplication	
Application Name	\Windows\cmd.exe
Executable path	
arguments	/c "\Flash\New Folder\test.bat" Par1 Par2
Single Instance	true

LaunchBrowser

Ouvre le navigateur Web par défaut. Vous pouvez définir une adresse URL comme argument.

Remarque : fonctionne uniquement sur les plateformes dotées d'un navigateur Web natif (par exemple, Windows CE PRO avec Internet Explorer activé)

LaunchVNC

Démarre le serveur VNC et ouvre la configuration.

Macro disponible uniquement pour les périphériques IHM basés sur la plateforme Windows CE.
Sur les périphériques IHM basés sur la plateforme Linux, le service VNC peut être activé depuis l'onglet « *Service* » de l'"Périphériques Linux" Page 478 BSP v. 1.0.44 ou plus récent nécessaire.

Voir "Module d'exécution (plug-in)" Page 75 pour l'inclure sur les périphériques Windows CE.

LaunchPDFViewer

Démarre la visionneuse de PDF.

Remarque : fonctionne uniquement sur les périphériques dotés d'une visionneuse de PDF.
Voir "Module d'exécution (plug-in)" Page 75 pour l'inclure sur les périphériques Windows CE.
Inclus sur les périphériques Linux depuis BSP v1.00.44.

LaunchUpdater

Met à jour le projet et le runtime depuis un périphérique externe.

Utilise le paramètre **Chemin**pour spécifier le dossier qui contiendra le fichier de mise à jour du paquet .Laissez le paramètre chemin vide si vous préférez sélectionner manuellement le fichier sur le périphérique IHM lorsque la macro est appelée.

Lorsque la macro LaunchUpdater est exécutée, le dialogue ci-dessous s'affiche sur le périphérique IHM

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Remarque : non pris en charge sur les périphériques basés sur Win32.

Interface JavaScript

project.launchUpdater(strPath)

Exemples

```
project.launchUpdater("\\USBMemory")
```

LockScreen

Verrouille temporairement l'écran tactile. Permet de nettoyer l'écran tactile.

La variable système **Time remaining to unlock** affiche le temps restant avant le déverrouillage. Voir "Variables d'écran" Page 106.

LoadProject

Décharge le projet en cours et charge le projet sélectionné vers le périphérique IHM.

Le nom du projet doit être spécifié par son chemin relatif, comme dans l'exemple ci-dessous :

LastVisitedProject

Décharge le projet en cours et revient au projet précédent.

Actions d'impression

Gère les tâches d'impression.

PrintGraphicReport

Imprime un rapport graphique.

Paramètre	Description
reportName	Attribue un nom au rapport
silent	**faux** = permet de configurer les propriétés de l'imprimante au runtime

PrintText

Imprime une chaîne.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Paramètre	Description
text	Chaîne à imprimer
silent	**faux** = permet de configurer les propriétés de l'imprimante au runtime

Cette action fonctionne en mode d'impression ligne par ligne et utilise un protocole normalisé commun à toutes les imprimantes qui le prennent en charge. Le texte est imprimé immédiatement ligne par ligne ou imprimé après un délai d'attente défini pour chaque modèle d'imprimante.

Remarque : l'impression pourrait prendre quelques minutes sur les modèles non conçus pour l'impression par ligne.

Ce n'est pas disponible sur les plateformes Linux (consultez "Capacités des périphériques IHM" Page 541 pour plus de détails sur les panneaux)

PrintBytes

Imprime une chaîne hexadécimale qui représente les données à imprimer (par ex. « 1b30 » pour imprimer < ESC 0 >.

Paramètre	Description
octets	Chaîne hexadécimale à imprimer
silent	**faux** = permet de configurer les propriétés de l'imprimante au runtime

Cette action fonctionne en mode d'impression ligne par ligne et utilise un protocole normalisé commun à toutes les imprimantes qui le prennent en charge. Le texte est imprimé immédiatement ligne par ligne ou imprimé après un délai d'attente défini pour chaque modèle d'imprimante.

Remarque : l'impression pourrait prendre quelques minutes sur les modèles non conçus pour l'impression par ligne.

Ce n'est pas disponible sur les plateformes Linux (consultez "Capacités des périphériques IHM" Page 541 pour plus de détails sur les panneaux)

EmptyPrintQueue

Efface la queue d'impression en cours. Lorsque l'action est exécutée au cours d'une tâche d'impression, la queue est vidée en fin de tâche.

PausePrinting

Pause la queue d'impression en cours. Lorsque l'action est exécutée au cours d'une tâche d'impression, la queue est vidée en fin de tâche.

ResumePrinting

Relance la queue précédemment mise en attente.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

AbortPrinting

Arrête l'exécution de la tâche en cours et la supprime de la queue. Si la queue contient une autre tâche, alors, après son interruption, la tâche suivante démarre.

Actions sur une recette

Utilisées pour programmer la gestion des recettes.

DownLoadRecipe

Copie les données d'une recette de mémoire flash du périphérique IHM vers le contrôleur (par ex. API, variable locale, en fonction du protocole) .

Paramètre	Description
RecipeName	Nom de la recette à télécharger
RecipeSet	Numéro de l'ensemble de recettes à copier. **curSet** = télécharge l'ensemble de recettes sélectionné

UpLoadRecipe

Enregistre les données de recettes du contrôleur (par ex. API, variable locale, selon le protocole) dans la mémoire flash du périphérique.

Paramètre	Description
RecipeName	Nom de la recette à charger
RecipeSet	Numéro de l'ensemble de recettes à copier. **curSet** = téléverse l'ensemble de recettes sélectionné

WriteCurrentRecipeSet

Définit la recette sélectionnée en tant qu'ensemble de recettes actuel.

Paramètre	Description
RecipeName	Nom de la recette à définir comme recette actuelle
RecipeSet	Ensemble de recettes à définir comme ensemble de recettes actuel

DownLoadCurRecipe

Télécharge les données de l'ensemble de recettes actuel vers le contrôleur.

Aucun paramètre requis.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

UploadCurRecipe

Téléverse un ensemble de données du contrôleur vers l'ensemble actuel de recettes.

Aucun paramètre requis

ResetRecipe

Restaure les paramètres usine des données de recettes. Les données de recettes d'origine écraseront les recettes téléversées

Sélectionnez la recette que vous souhaitez réinitialiser aux données d'usine.

DumpRecipeData

Déplace les données de recette vers un périphérique de stockage interne ou externe. Les données sont enregistrées au format .csv.

Paramètre	Description
RecipeName	Nom de la recette à déplacer
FilePath	Dossier de destination • Interne = *\Flash\QTHMI\workspace\Dump* • Lecteur USB = *\USBMemory* • Carte SD = *\Storage Card* • Réseau public = *\\<nom d'hôte ou IP>\sharePath* • Réseau privé = *\\<nom d'utilisateur>:<mot de passe>@<nom d'hôte ou IP>\sharePath* Note : les formats supportés pour la mémoire externe sont FAT ou FAT32 (le format NTFS n'est pas pris en charge). Remarque : les réseaux privés sont supportés uniquement par les périphériques Linux avec BSP à partir de la version 1.0.25.
FileName	Variable définissant un nom de fichier. Les caractères de remplacement ci-dessous sont pris en charge • %r = nom de recette • %d = nom de jeu de données Exemple : %r_%d
DateTimePrefixFileName	**vrai** = un préfixe composé de la date et de l'heure sera ajouté au fichier déplacé (par exemple D2012_01_01_H10_10_recette1.csv)
TimeSpec	Format de l'heure : • **Locale** = les valeurs de temps exportées sont l'heure du périphérique IHM. • **Globale** = les valeurs de temps sont exportées au format UTC.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

RestoreRecipeData

Restaure les données de recette sauvegardées auparavant.

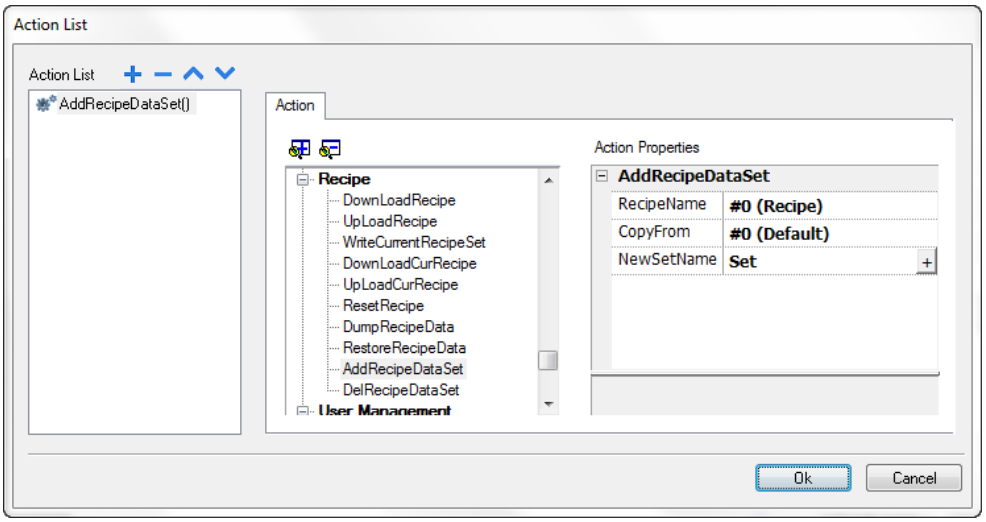
Paramètre	Description
RecipeName	Recettes à restaurer : • AllRecipes Les données de toutes les recettes seront remplacées par les données lues à partir du fichier externe • CurrentRecipe Seule les données de la recette en cours de sélection seront remplacées par les données lues à partir du fichier externe
RecipeDataSet	Disponible uniquement lorsque RecipeName=CurrentRecipeNomRecette=RecetteActuelle. Sélectionner les ensembles de données à restaurer : • AllRecipeDataSet Tous les jeux de données seront restaurés • curSet Seul le jeu de données des données en cours de sélection sera restauré
Type de restauration	Disponible uniquement lorsque RecipeDataSet=AllRecipeDataSet. Ce paramètre définit le comportement lorsque le nombre d'ensembles de données dans le fichier à restaurer ne correspond pas au nombre d'ensembles de données à l'intérieur du périphérique IHM. • Replace Tous les ensembles de données du périphérique sont retirés et remplacés par les ensembles de données du fichier CSV • Match Remplace uniquement l'ensemble de données du périphérique qui ont le même identifiant d'ensemble de données • MatchAndAdd Remplace l'ensemble des données du périphérique qui ont le même identifiant d'ensemble de données et ajoute l'ensemble complémentaire de données trouvé dans le fichier csv (remarque : les ensembles de données présents dans le périphérique mais pas dans le fichier csv ne sont pas supprimés du périphérique)
FilePath	Dossier source • Interne = *\Flash\QTHMI\workspace\Dump* • Lecteur USB = *\USBMemory* • Carte SD = *\Storage Card* • Réseau public = *<nom d'hôte ou IP>\sharePath* • Réseau privé = *<nom d'utilisateur>:<mot de passe>@<nom d'hôte ou IP>\sharePath* Note : les formats supportés pour la mémoire externe sont FAT ou FAT32 (le format NTFS n'est pas pris en charge).

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Paramètre	Description
	Remarque : les réseaux privés sont supportés uniquement par les périphériques Linux avec BSP à partir de la version 1.0.25.
FileName	Variable associée à partir de laquelle lire le nom du fichier au runtime.
BrowseForFile	**vrai** = affiche le dialogue Ouvrir permettant de rechercher le fichier à lire. **faux** = aucun dialogue ne s'affiche,

AddRecipeDataSet

Ajoute un nouvel ensemble de données à la recette sélectionnée. Le nouvel ensemble de données est ajouté à la fin des ensembles de données déjà définis.

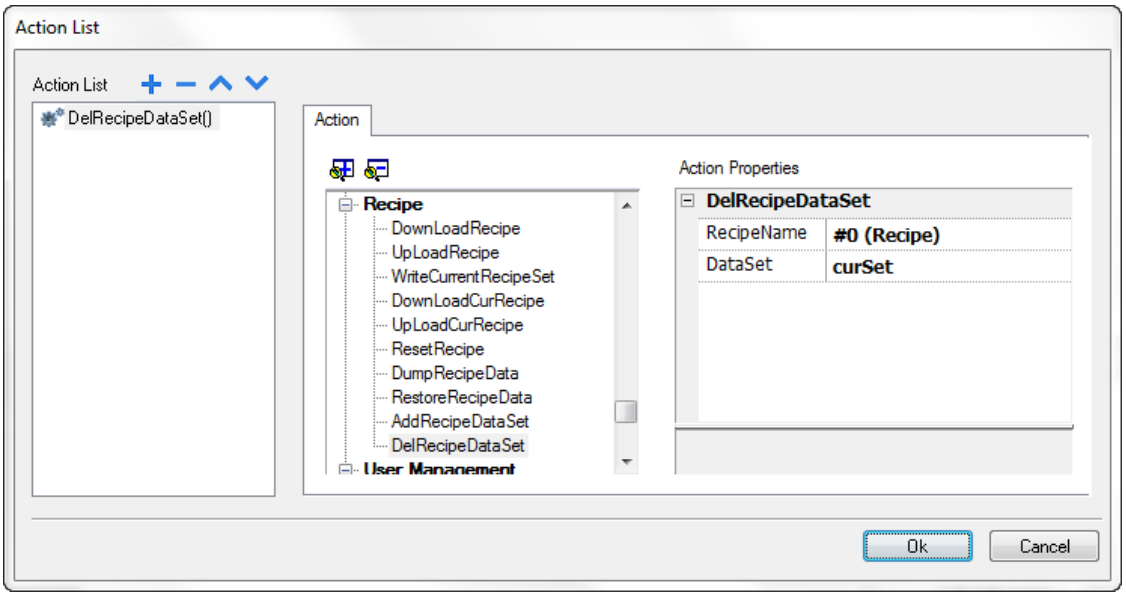


Paramètre	Description
RecipeName	Recette à laquelle l'ensemble de données est ajouté.
CopyFrom	Ensemble de données à partir duquel les valeurs des paramètres sont copiées pour initialiser le nouvel ensemble de données.
NewSetName	Nom d'un nouvel ensemble de données. Il est possible ici d'indiquer une référence de variable.

DelRecipeDataSet

Supprime un ensemble de données de la recette sélectionnée. La suppression d'un ensemble de données réorganisera le numéro de position des ensembles de données suivants.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



Paramètre	Description
RecipeName	Recette dans laquelle l'ensemble de données sera supprimé.
DataSet	Ensemble de données à supprimer.

Actions client à distance

Utilisées pour téléverser et télécharger des fichiers vers et depuis un périphérique IHM distant. Ces actions peuvent être utilisées uniquement à partir d'un HMI Client distant pour accéder à des fichiers distants via FTP.

Important : activez la prise en charge de FTP et attribuez tous les droits d'utilisateurs nécessaires aux dossiers utilisés pour transférer des fichiers.

ChargerSurIHM

Ouvre un dialogue Ouvrir un fichier pour sélectionner un fichier à téléverser vers le périphérique IHM distant.

Paramètre	Description
Destination	Chemin de destination du fichier à téléverser vers le périphérique IHM
Filtrer	Extensions des fichiers à afficher séparées par des virgules (par ex., *.txt)

DownloadFromHMI

Ouvre un dialogue Ouvrir un fichier pour sélectionner un fichier à télécharger depuis le périphérique IHM distant.

Remarque : seuls les fichiers conformes au filtre défini sont affichés et peuvent être téléchargés.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Paramètre	Description
Source	Chemin source sur le périphérique IHM où télécharger des fichiers
Filtrer	Extensions des fichiers à afficher séparées par des virgules (par ex., *.txt)

Interface JavaScript

```
boolean project.uploadToHMI(dirPath, strFilter);
```

```
boolean project.downloadFromHMI(dirPath, strFilter);
```

Paramètre	Description
dirPath	Chemin source du fichier à charger/télécharger du périphérique IHM
strFilter	Extensions des fichiers à afficher séparées par des virgules (par ex., *.txt)

Valeurs retournées :

vrai	Transfert réussi
faux	Transfert échoué

Remarque : une fois transférées, les variables système sont mises à jour avec le statut des opérations en cours.

Actions système

Permet de gérer les propriétés système.

Restart

Redémarre le runtime.

DumpTrend

Stocke les données des tendances passées sur des lecteurs externes (lecteur USB ou carte SD).

Paramètre	Description
TrendName	Nom de la tendance passée à stocker
FolderPath	Dossier de destination : • Interne = *\Flash\QTHMI\workspace\Dump* • Lecteur USB = *\USBMemory* • Carte SD = *\Storage Card* • Réseau public = *\\<nom d'hôte ou IP>\sharePath*

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Paramètre	Description
	• Réseau privé = \\<*nom d'utilisateur*>:<*mot de passe*>@<*nom d'hôte ou IP*>*sharePath* Note : les formats supportés pour la mémoire externe sont FAT ou FAT32 (le format NTFS n'est pas pris en charge). Remarque : les réseaux privés sont supportés uniquement par les périphériques Linux avec BSP à partir de la version 1.0.25.
FileFormat	**Binaire** = la mémoire tampon est vidée au format binaire (fichiers .dat et .inf). Les deux fichiers sont ensuite requis pour convertir les données au format .csv à l'aide d'un utilitaire externe. **Compatibilité CSV** = la mémoire tampon est vidée à l'emplacement spécifié dans un format de fichier .csv compatible avec les versions 1.xx **CSV compact** = la mémoire tampon est vidée à l'emplacement spécifié dans un fichier .csv utilisant un format plus récent Voir "Exporter les données de mémoire de tendance" Page 231
DateTimePrefix	**vrai** = le fichier vidé aura la date et l'heure comme préfixe de son nom (par exemple D2012_01_01_H10_10_Tendance1.csv)
TimeSpec	Format de l'heure : • **Locale** = les valeurs de temps exportées sont l'heure du périphérique IHM. • **Globale** = les valeurs de temps sont exportées au format UTC.
FileName	Activé lorsque le DateTimePrefixFileName=vrai Les caractères de remplacement ci-dessous sont pris en charge • %n = nom de tendance • %y = année • %M = mois • %d = jour • %h = heure • %m = minutes • %s = secondes Exemple : \%n\%y%M%d\%h%m%s

Paramètres supplémentaires disponibles uniquement lorsque le Format de fichier sélectionné est **CSV Compact**

Lorsque les paramètres « Select Fields » et « Select Curves » sont vides, le fichier.csv est déposé dans l'ancien « CSV compact » sans le format sélection de colonnes. Voir aussi "Exporter les données de mémoire de tendance" Page 231

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

<table>
<tr><th>Paramètre</th><th>Description</th></tr>
<tr><td>Select Fields</td><td>Sélectionne les colonnes à exporter dans le fichier déposé.

Colonnes disponibles :

• Date et heure
• Date
• Heure
• Valeur
• Qualité

Notez que « Attach to tag » peut être utilisé pour définir les colonnes à exporter au runtime à partir de l'application IHM. La variable doit contenir une chaîne de caractères comprenant la liste des champs à exporter séparés par des virgules.

Exemple :

• "" (chaîne vide = tous champs disponibles)
• "Date,Heure,Valeur,Qualité"
• "Date,Heure,Qualité"</td></tr>
<tr><td>Select Curves</td><td>Sélectionne les courbes à exporter dans le fichier déposé.

Notez que « Attach to tag » peut être utilisé pour définir les courbes à exporter au runtime à partir de l'application IHM. La variable doit contenir une chaîne de caractères comprenant la liste des noms de courbes à exporter séparés par des virgules.

Exemple :

• "" (chaîne vide = toutes tendances disponibles)
• "Nom1,Nom2,Nom3"
• "Nom1,Nom3"</td></tr>
<tr><td>Date Format</td><td>Sélectionne le format pour la Date et l'Heure

L'utilisation de « Attach to tag » est possible, définit le format de la date au runtime par une chaîne de caractères

<table>
<tr><th colspan="2">Marque substitutive de la date</th></tr>
<tr><td>d</td><td>Le jour au format numérique sans zéro en tête (1 à 31)</td></tr>
<tr><td>dd</td><td>Le jour au format numérique avec un zéro en tête (01 à 31)</td></tr>
<tr><td>ddd</td><td>Le nom abrégé du jour localisé (p. ex. 'Lun' à 'Dim')</td></tr>
<tr><td>dddd</td><td>Le nom en entier du jour localisé (p. ex. 'Lundi' à 'Dimanche')</td></tr>
<tr><td>M</td><td>Le mois au format numérique sans zéro en tête (1 à 12)</td></tr>
<tr><td>MM</td><td>Le mois au format numérique avec un zéro en tête (01 à 12)</td></tr>
</table>
</td></tr>
</table>

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Paramètre	Description

Marque substitutive de la date	
MMM	Le nom abrégé du mois localisé (p. ex. 'Jan' à 'Déc')
MMMM	Le nom en entier du mois localisé (p. ex. 'Janvier' à 'Décembre')
yy	L'année au format numérique à 2 chiffres (00 à 99)
yyyy	L'année au format numérique à 4 chiffres

Marque substitutive de l'heure	
h	L'heure sans zéro en tête (0 à 23 ou 1 à 12 si l'affichage est AM/PM)
hh	L'heure avec zéro en tête (00 à 23 ou 01 à 12 si l'affichage est AM/PM)
m	La minute sans zéro en tête (0 à 59)
mm	La minute avec un zéro en tête (00 à 59)
s	La seconde sans zéro en tête (0 à 59)
ss	La seconde sans zéro en tête (00 à 59)
zzz	La milliseconde sans zéro en tête
z	La milliseconde
AP	Utilisé pour l'affichage AM/PM. AP sera remplace par « AM » ou « PM »
ap	Utilisé pour l'affichage am/pm. ap sera remplace par « am » ou « pm»

Remarque : l'exécution de l'action DumpTrend forcera automatiquement le vidage vers le disque des données temporaires conservées dans la mémoire RAM. Voir "Tendances historiques" Page 233 en page 1 pour les détails sur la sauvegarde de données échantillonnées sur le disque.

Remarque : les lecteurs externes connectés par port USB doivent être au format FAT ou FAT32. Le format NFTS n'est pas pris en charge.

AVERTISSEMENT : sachez qu'il existe des limites sur le nombre maximal de fichiers pouvant être créés dans un dossier. Ces limites dépendent de divers facteurs et leur calcul est complexe mais vous pouvez considérer un maximum de 999 fichiers peuvant être utilisés dans un dossier.

Conversion des fichiers clichés binaires au format .csv

L'outil TrendBufferReader.exe est stocké dans le dossier *Outils* du dossier d'installation PB610 Panel Builder 600.

Utilisez la syntaxe suivante :

```
TrendBufferReader -r Trend1 Trend1.csv 1
```


Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

où :

`Trend1` = nom de la mémoire des tendances sans extension résultant du vidage (le nom du fichier d'origine étant trend1.dat)

`Trend1.csv` = nom du fichier de sortie.

AVERTISSEMENT : TrendBufferReader.exe est un ancien utilitaire qui ne fonctionne pas avec les nouveaux tampons multivariables. L'utilisation de cet utilitaire n'est pas recommandée. L'utilitaire n'est plus maintenu parce qu'il y a maintenant la possibilité de vider la mémoire tampon de tendance directement au format.csv.

Structure du fichier .csv

Le fichier .csv généré comprend cinq colonnes

Colonne	Description
Type de données	Type de données de la variable échantillonnée : 0 = vide 1 = booléan 2 = octet 3 = court 4 = entier 5 = octet non signé 6 = court non signé 7 = entier non signé 8 = flottant 9 = double
Valeur	Valeur de l'échantillon
Timestamp (UTC)	Horodatage au format UTC
Sampling Time(ms)	Intervalle entre les échantillons en millisecondes
Qualité	Qualité de la valeur de la variable. Les informations codées selon les normes d'accès aux données OPC DA et stockées sous forme de données d'octets (8 bits) sont définies sous forme de trois champs de bits : qualité, sous-état et état des limites. Les huit bits de qualité sont disposés dans l'ordre suivant : QQSSSSLL. Pour des descriptions complètes et détaillées de tous les champs, veuillez consulter la documentation officielle OPC DA.

Valeurs de qualité régulièrement utilisées

Les valeurs de qualité les plus fréquemment utilisées retournées par le moteur d'acquisition de l'IHM sont les suivantes :

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Code de qualité	Qualité	Description
0	BAD	La valeur est mauvaise mais aucune raison particulière n'est indiquée
4	BAD	Problème de serveur particulier avec la configuration. Par exemple, la variable a été supprimée du fichier de configuration (tags.xml).
8	BAD	Aucune valeur ne pourrait être disponible pour le moment, par exemple, la valeur n'a pas été fournie par la source de données.
12	BAD	Défaillance de périphérique détectée
16	BAD	Délai d'attente dépassé avant la réponse du périphérique.
24	BAD	Échec de communication
28	BAD	Aucune donnée trouvée pour le drapeau spécifique des valeurs de limite inférieures et supérieures de l'interface des tendances.
32	BAD	Aucune donnée collectée (par exemple, l'archivage n'est pas actif. Drapeau spécifique de l'interface tendances. Cette valeur est également utilisée pour indiquer un état hors ligne temporaire (dans toute situation où l'échantillonnage a été arrêté).
64	UNCERTAIN	Aucune raison spécifique.
65	UNCERTAIN	Aucune raison spécifique. La valeur a été fixée à une limite inférieure.
66	UNCERTAIN	Aucune raison spécifique. La valeur a été fixée à une limite supérieure.
67	UNCERTAIN	Aucune raison spécifique. La valeur est une constante et ne peut pas varier.
84	UNCERTAIN	La valeur retournée se situe en-dehors des limites définies. Dans ce cas, le champ **Limites** indique quelle limite a été dépassée mais la valeur peut sortir de cette plage.
85	UNCERTAIN	La valeur retournée se situe en-dehors des limites définies. Dans ce cas, le champ **Limites** indique quelle limite a été dépassée mais la valeur peut sortir de cette plage. La valeur a été fixée à une limite inférieure.
86	UNCERTAIN	La valeur retournée se situe en-dehors des limites définies. Dans ce cas, le champ **Limites** indique quelle limite a été dépassée mais la valeur peut sortir de cette plage. La valeur s'est accrochée à une limite supérieure

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Code de qualité	Qualité	Description
87	UNCERTAIN	La valeur retournée se situe en-dehors des limites définies. Dans ce cas, le champ **Limites** indique quelle limite a été dépassée mais la valeur peut sortir de cette plage. La valeur est une constante et ne peut pas varier.
192	GOOD	-

DeleteTrend

Supprime les données de tendance enregistrées.

Définissez le nom de la tendance de laquelle vous souhaitez supprimer les journaux.

DumpEventArchive

Stocke le journal des alarmes passées et les données du journal d'audit passées sur des lecteurs externes (lecteur USB ou carte SD).

Paramètre	Description
EventArchive	Nom de la mémoire tampon dans laquelle déposer les données
FolderPath	Dossier de destination • Interne = *\Flash\QTHMI\workspace\Dump* • Lecteur USB = *\USBMemory* • Carte SD = *\Storage Card* • Réseau public = \\<*nom d'hôte ou IP*>*sharePath* • Réseau privé = \\<*nom d'utilisateur*>:<*mot de passe*>@<*nom d'hôte ou IP*>*sharePath* Note : les formats supportés pour la mémoire externe sont FAT ou FAT32 (le format NTFS n'est pas pris en charge). Remarque : les réseaux privés sont supportés uniquement par les périphériques Linux avec BSP à partir de la version 1.0.25.
DumpConfigFile	Dépose les fichiers description des archives
DumpAsCSV	**true** = la mémoire tampon est vidée à l'emplacement spécifié dans un fichier .csv **faux** = la mémoire tampon est vidée au format binaire (fichiers .dat et .inf). Les deux fichiers sont ensuite requis pour convertir les données au format .csv à l'aide d'un utilitaire externe.
DateTimePrefix	**vrai** = un préfixe composé de la date et de l'heure sera ajouté au fichier dépose (par exemple D2012_01_01_T10_10_alarmBuffer1.csv)

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Paramètre	Description
timeSpec	Format de l'heure : • **Locale** = les valeurs de temps exportées sont l'heure du périphérique IHM. • **Globale** = les valeurs de temps sont exportées au format UTC.
Colonnes csv	Sélectionnez les colonnes à déposer dans le fichier .csv. Disponible uniquement lorsque le Supprimer Archive des Événements est une mémoire tampon d'alarmes
FileName	Les métacaractères ci-dessous sont pris en charge • %n = nom d'archive d'événements • %y = année • %M = mois • %d = jour • %h = heure • %m = minutes • %s = secondes Exemple : \%n\%y%M%d\%h%m%s Disponible uniquement lorsque DateTimePrefixFileName=true
Langue	Sélectionnez la langue à utiliser. Lorsque vide, le déchargement exécutera sur toutes les langues. Disponible uniquement lorsque le Supprimer Archive des Événements est une mémoire tampon d'alarmes

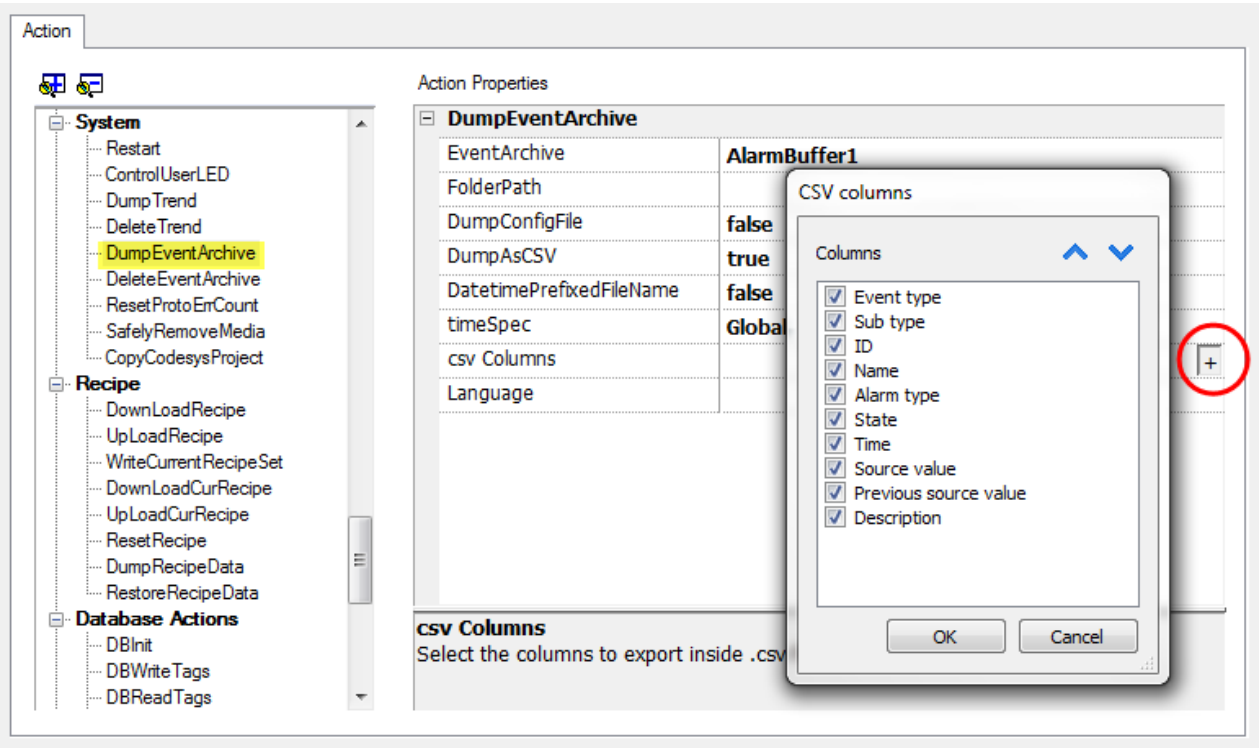
Déposer au format CSV

DumpAsCSV = true

Pour les tampons d'alarmes, le paramètre supplémentaire « csv Colums » donne la possibilité de sélectionner les colonnes à exporter à l'intérieur du fichier.csv.

Note : disponible uniquement pour les tampons d'alarmes.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



Déposer au format BINAIRE

DumpAsCSV = false

Lors de l'export des mémoires tampons d'événements au format binaire et lorsque le paramètre **DumpConfigFile** est défini sur vrai (paramètres recommandés), il y a deux dossiers :

- **data**, contenant les fichiers de données,
- **config**, contenant les fichiers de configuration pour la conversion au format .csv.

Une fois les deux dossiers copiés du lecteur USB sur le disque de l'ordinateur, la structure des dossiers sera la suivante :

config\

alarms.xml

eventconfig.xml

data\

AlarmBuffer1.dat

AlarmBuffer1.inf

\

AlarmBufferReader.exe

Conversion des fichiers déposés au format .csv

L'outil AlarmBufferReader.exe est stocké dans le dossier *Utilitaires* du dossier d'installation PB610 Panel Builder 600.

Utilisez la syntaxe suivante :

```
AlarmBufferReader AlarmBuffer1 FICHIER ./AlarmBuffer1.csv
```

où :

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

- `AlarmBuffer1` = nom du fichier déposé .dat sans extension
- `AlarmBuffer1.csv` = nom du fichier de sortie.

L'utilitaire AuditTrailBufferReader.exe est disponible pour les mémoires tampon des journaux d'audit.

Utilisez la syntaxe suivante :

```
AuditTrailBufferReader AuditTrail FILE ./AuditTrail.csv
```

où :

- `AuditTrail` = nom de la mémoire vidée sans extension et
- `AuditTrail1.csv` = nom du fichier de sortie.

DeleteEventArchive

Supprime les données du journal des mémoires d'événements enregistrées.

Indiquez le nom de la mémoire d'événements à supprimer des journaux d'événements.

ResetProtoErrCount

Réinitialise la variable système Compteur d'erreurs de protocole

Voir "Variables système (Associer à)" Page 95 pour les détails.

SafelyRemoveMedia

Permet de retirer la carte SD ou le lecteur USB en toute sécurité de l'IHM.

Ce n'est pas disponible sur les plateformes Linux (consultez "Capacités des périphériques IHM" Page 541 pour plus de détails sur les panneaux)

SaveEventArchive

Sauvegardez les enregistrements situés dans le journal d'audit dans un fichier signé. La signature du fichier garantira que les enregistrements dans le rapport ne sont pas modifiés.

Paramètre	Description
EventArchive	Nom de la mémoire tampon dans laquelle déposer les données
FolderPath	Dossier de destination • Interne = *\Flash\QTHMI\workspace\Dump* • Lecteur USB = *\USBMemory* • Carte SD = *\Storage Card* • Réseau public = \\<*nom d'hôte ou IP*>*sharePath* • Réseau privé = \\<*nom d'utilisateur*>:<*mot de passe*>@<*nom d'hôte ou IP*>*sharePath* Note : les formats supportés pour la mémoire externe sont FAT ou FAT32 (le format NTFS n'est pas pris en charge).

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

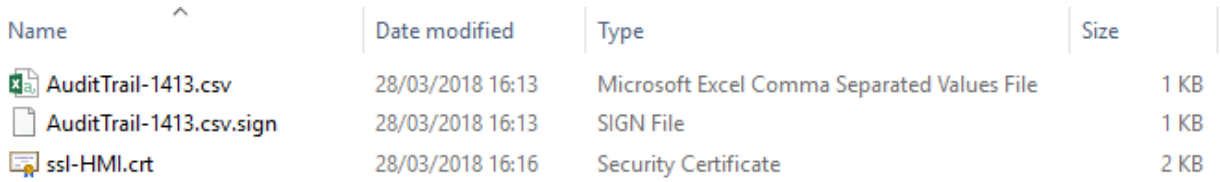
Paramètre	Description
	Remarque : les réseaux privés sont supportés uniquement par les périphériques Linux avec BSP à partir de la version 1.0.25.
FileName	Les métacaractères ci-dessous sont pris en charge • %n = nom d'archive d'événements • %y = année • %M = mois • %d = jour • %h = heure • %m = minutes • %s = secondes Exemple : \%n\%y%M%d\%h%m%s
Format	Format du fichier de sortie • CSV
Signed	Génère la signature de fichier. L'algorithme à utiliser pour la signature est défini dans les paramètres des propriétés du projet. Voir "Projet" Page 76 pour les algorithmes disponibles
TimeSpec	Format de l'heure : • **Locale** = les valeurs de temps exportées sont l'heure du périphérique IHM. • **Globale** = les valeurs de temps sont exportées au format UTC.
PeriodMode	Définit la fenêtre d'heure à exporter • Tous événements • Aujourd'hui • Hier • Semaine dernière • Mois dernier • Semaine en cours • Mois en cours • Personnalisé Les paramètres supplémentaires «periodFrom » et « periodTo » seront affichés
Séparer date et heure	Utilise 2 colonnes séparées pour la date et l'heure
Date Format	Sélectionne le format pour la Date et l'Heure

Signed file

Lorsque le paramètre Fichier signé est vrai, deux fichiers seront ajoutés à fileame.csv :

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

- filename.csv.sign
  La signature du fichier garantira que les enregistrements dans le fichier filename.csv n'ont pas été modifiés
- ssl-HMI.crt
  Une copie du certificat du périphérique IHM requis pour vérifier l'authenticité du rapport.



Pour plus d'informations sur le certificat et la vérification des fichiers signés, voir "Certificat x.509" Page 279.

Pour plus d'informations sur les informations exportées, voir "Exporter le journal d'audit au format .csv" Page 303.

LogMessage

Ajouter un message dans la mémoire tampon du journal d'audit.

Cette macro donne la possibilité au développeur de décider de garder la trace de certains événements (par exemple lorsqu'un bouton est pressé, lorsqu'une page est activée, etc. La variable attach to pour avoir la possibilité de définir le message à enregistrer au runtime est prise en charge.

Paramètre	Description
EventArchive	Nom de la mémoire tampon d'audit où ajouter le message
Message	Message à ajouter à la mémoire tampon d'audit

DeleteOldFiles

Cette macro supprime les fichiers plus anciens qui donnent le nombre de jours.

Sur PC, il n'y a pas de restriction de l'utilisation du chemin. Dans les panneaux il est autorisé dans les médias dynamiques et la partition de données (/Flash dans WinCE et /mnt/data dans Linux)

Il sera de la responsabilité du développeur de configurer l'application afin d'éviter la possibilité de supprimer des fichiers système.

Paramètre	Description
FolderPath	Le chemin du dossier dans lequel chercher les fichiers à supprimer
FileTypes	Liste des fichiers à supprimer séparés par des virgules. Les caractères de substitution sont pris en charge Exemple : *.png,*.jpg
OlderDays	Nombre minimum de jours sans modifications

Actions de variables

Permet d'interagir avec les variables.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

DataTransfer

Échanger des données entre :

- deux contrôleurs,
- des registres au sein d'un contrôleur,
- des variables systèmes et des contrôleurs,
- des contrôleurs et des variables système

Les divers types de variables incluent une variable de contrôleur, une variable système, une variable de recette et une propriété de widget.

ToggleBit

Commuter une valeur de bit d'une variable.

BitIndex vous permet de sélectionner le bit à commuter : la commutation requiert l'opération de lecture-modification-écriture ; la valeur en lecture est inversée puis réécrite dans la variable.

SetBit

Définit le bit sélectionné sur « 1 ».

BitIndex vous permet de sélectionner la position du bit à l'intérieur de la variable.

ResetBit

Réinitialise le bit sélectionné sur « 0 »

BitIndex vous permet de sélectionner la position du bit à l'intérieur de la variable.

WriteTag

Écrit des valeurs constantes dans la mémoire du contrôleur. Indiquez le nom et la valeur de la variable.

StepTag

Augmente ou diminue la valeur de variable.

Paramètre	Description
TagName	Nom de la variable à augmenter/diminuer
Étape	Valeur de pas
Ne pas dépasser les limites	Active la limite de pas
Step Limit	Valeur de la limite de pas, si activée.

BiStep (par pas)

Cette action est similaire à l'action StepTag mais le sens d'Incrémentation/Décrémentation est automatiquement défini par la rotation de la molette. La valeur de la variable sera augmentée lorsque la molette est tournée en sens horaire. La valeur de la variable sera diminuée lorsque la molette est tournée en sens inverse d'horloge.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Propriété	Description
TagName	Nom de la variable pour laquelle exécuter l'action BiStep
Step	Valeur à ajouter/soustraire à la variable à chaque rotation de roulette (dépend de la propriété du pas de l'événement)
Pas de l'événement	Cette propriété permet de choisir d'ajouter/soustraire des valeurs de pas à chaque pas de molette ou à chaque événement de rotation. **false** = La valeur de pas est ajoutée ou soustraite à la variable à chaque événement de rotation. *Exemple : si on tourne la molette en effectuant 5 pas de molette en un seul événement, la variable sera augmentée/diminuée de 1.* **true** = La valeur de pas est ajoutée ou soustraite à la variable à chaque événement unique de rotation. *Exemple : si on tourne la molette en effectuant 5 pas de molette en un seule rotation, la variable sera augmentée/diminuée de 5.*
Ne pas dépasser les limites	Si vrai, active les limites inférieure et supérieure, lesquelles représentent la valeur inférieure et supérieure que la variable peut assumer suite à une action BiStep de variable.
LowerLimit	Si « Ne pas faire de pas au-delà de la limite » est vrai, cette propriété représente la valeur la plus basse que la variable peut assumer suite à une action BiStep de variable.
UpperLimit	Si « Ne pas faire de pas au-delà de la limite » est vrai, cette propriété représente la valeur la plus grande que la variable peut assumer suite à une action BiStep de variable.

Seulement disponibles pour les actions de molette (OnWheel)

ActivateGroup

Force la mise à jour d'un groupe de variables.

Les variables sont mises à jour, soit lorsqu'elles sont utilisées dans la page en cours, soit en continu, si elles ont été définies comme actives dans l'éditeur de variables. Cette action force la mise à jour en continu de toutes les variables d'un groupe.

DeactivateGroup

Désactive un groupe de variables, c'est-à-dire arrête de forcer la mise à jour d'un groupe de variables.

EnableNode

Active/désactive une action pour une gestion de nœud hors ligne. Aucune communication n'est effectuée avec un nœud désactivé.

Paramètre	Description
Protocol ID	Identificateur unique du protocole sélectionné
NodeID	Identificateur de nœud dans le protocole sélectionné. Peut être associé à une variable.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Paramètre	Description
Activer	Statut de communication de nœud **Faux** = désactivé **Vrai** = activé Lorsque le nœud est associé à une variable, variable = 0 signifie **Faux**

BACnetClearPriority (effacer propriété BACnet)

Reportez-vous au manuel BACnet dans le dossier « Pilotes de communication » pour une description détaillée des actions BACnet.

BACnetClearAllPriorities (effacer toutes propriétés BACnet)

Reportez-vous au manuel BACnet dans le dossier « Pilotes de communication » pour une description détaillée des actions BACnet.

BACnetSetPriority (Définir propriété BACnet)

Reportez-vous au manuel BACnet dans le dossier « Pilotes de communication » pour une description détaillée des actions BACnet.

ClearRetentiveMemory

Si défini sur 0, efface le contenu de la mémoire rémanente.

ForceReadTag

Force le rafraîchissement de la variable spécifiées depuis un contrôleur distant.

Actions de tendance

Utilisées pour les Tendances de données actives et le Widget de tendances passées.

RefreshTrend

Rafraîchit la fenêtre **Tendance**.

Cette action peut être utilisée dans tous les widgets de Tendances/Graphiques. Indiquez le widget en tant que paramètre de l'action.

ScrollLeftTrend

Fait défiler la fenêtre **Tendance** vers la gauche au dixième (1/10) de la durée de page.

Remarque : pour les tendances en temps réel, mettez en pause la tendance à l'aide de l'action **PauseTrend**, sinon la fenêtre sera ramenée en continu à la valeur actuelle.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

ScrollRightTrend

Fait défiler la fenêtre **Tendance** vers la droite au dixième (1/10) de la durée de page.

Remarque : pour les tendances en temps réel, mettez en pause la tendance à l'aide de l'action **PauseTrend**, sinon la fenêtre sera ramenée en continu à la valeur actuelle.

PageLeftTrend

Fait défiler la fenêtre **Tendance** d'une page. Par exemple, si la taille de la page correspond à 10 minutes, vous pouvez utiliser l'action **PageLeftTrend** pour faire avancer la tendance vers la gauche de 10 minutes.

PageRightTrend

Fait défiler la fenêtre **Tendance** d'une page. Par exemple, si la taille de la page correspond à 10 minutes, vous pouvez utiliser l'action **PageRightTrend** pour faire avancer la tendance vers la droite de 10 minutes.

PageDurationTrend

Configure la durée de la page de la fenêtre **Tendance**.

Définit le nom de la tendance et la durée de la page.

Remarque : vous pouvez configurer la durée de la page au runtime en utilisant un widget de liste déroulante.

ZoomInTrend

Réduit la durée de page.

ZoomOutTrend

Étend la durée de page.

ZoomResetTrend

Réinitialise le niveau de zoom au niveau de zoom d'origine.

ZoomInYAxisTrend

Réduit l'axe Y.

ZoomOutYAxisTrend

Étend l'axe Y.

ZoomResetYAxisTrend

Réinitialise le niveau de zoom de l’axe Y au niveau de zoom d'origine.

MettreTendanceEnPause

Arrête le traçage des courbes de tendance dans la fenêtre **Tendance**.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Lorsque cette action est utilisée pour une tendance en temps réel, le traçage s'arrête une fois que la courbe a atteint le bord droit du graphique. Cette action ne met pas fin à l'enregistrement de tendance.

ResumeTrend

Reprend le traçage de la tendance après une pause.

ShowTrendCursor

Affiche la valeur de la courbe pour un point donné sur l'axe X.

Ceci active le curseur de tendance Un curseur (ligne verticale) s'affichera dans le widget de tendance.

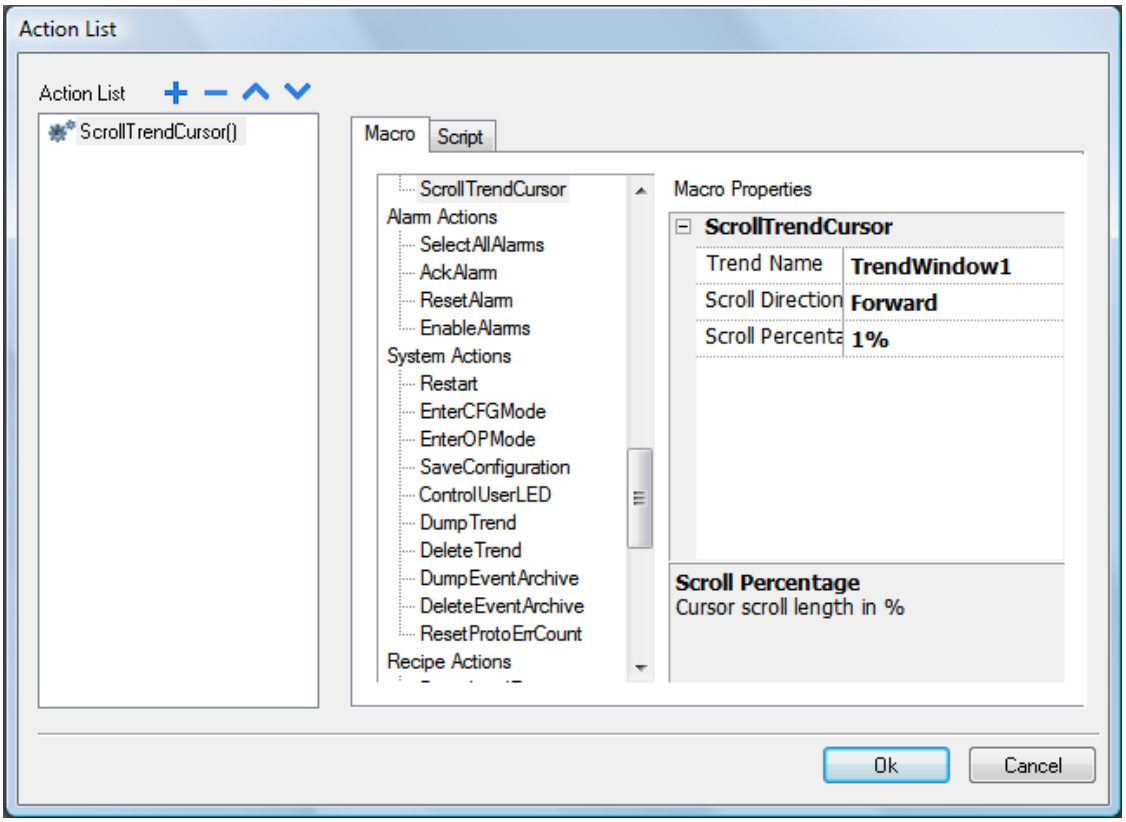
Lorsque le curseur graphique est activé, le défilement de la tendance s'arrête.

L'action **ScrollCursor** déplace le curseur graphique sur les courbes ou sur toute la fenêtre de **Tendance**.

ScrollTrendCursor

Fait défiler le curseur de tendance vers l'avant ou vers l'arrière.

La valeur du curseur Y affichera la valeur de la tendance située sous la pointe du curseur. Le pourcentage de défilement peut être défini à 1 % ou 10 %. Le pourcentage est calculé d'après la durée de la fenêtre de tendance.

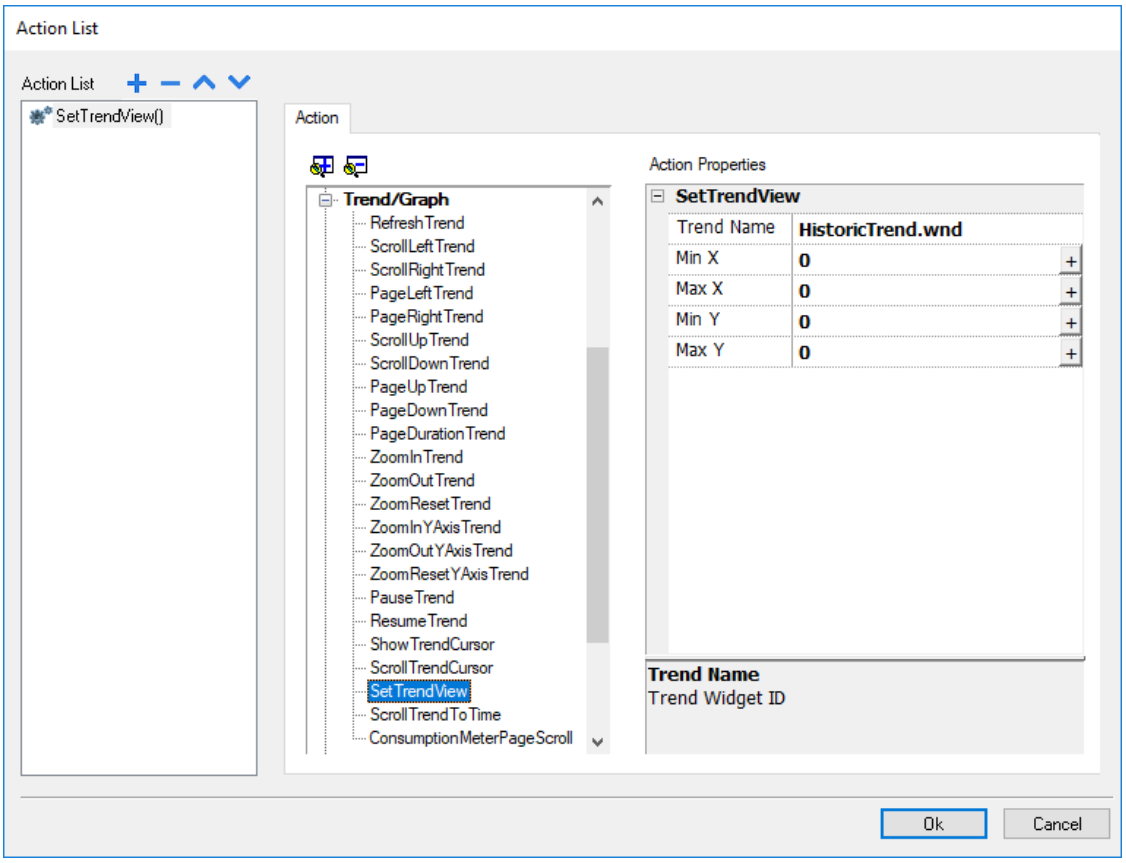


SetTrendView

Utilisez cette macro pour modifier les plages d'axes de la vue de tendance.

Lorsque Min X=0 et Max X=0, les valeurs statiques définies dans les propriétés du widget sont utilisées. Il en va de même pour l'axe Y.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



ScrollTrendToTime

Fait défiler la fenêtre **Tendance** à un moment donné dans le temps.

Utilisez cette action lorsque vous souhaitez faire défiler la fenêtre de tendance jusqu'à une position spécifique dès qu'un événement spécifique survient.

Exemple

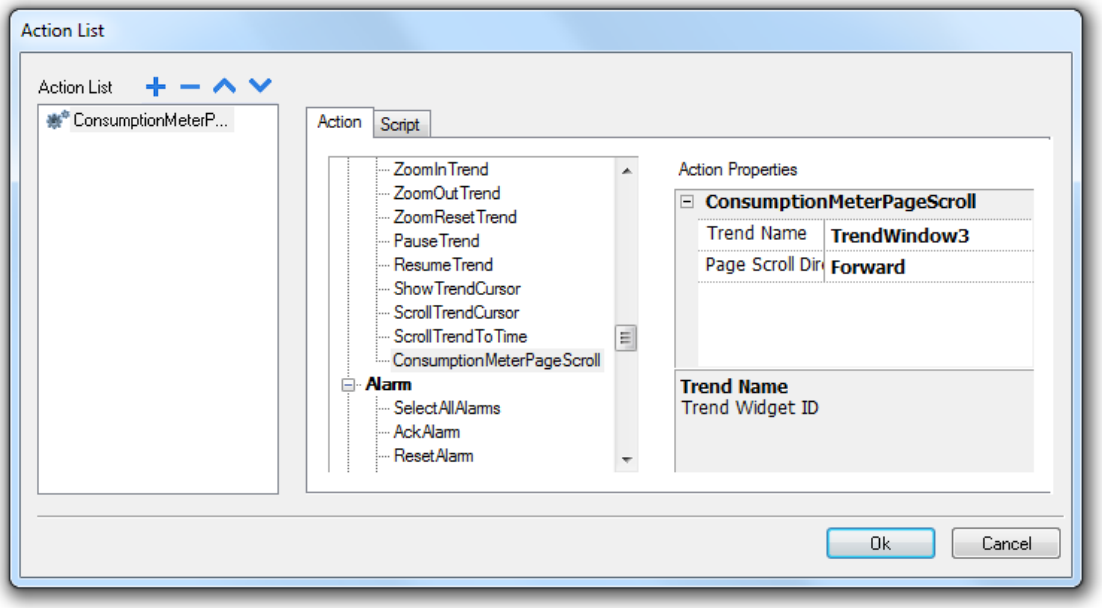
1. Configurer une action pour un événement (par exemple, une alarme) qui exécute un transfert de données de l'heure système vers une variable.
2. Sélectionnez cette variable en tant que paramètre **ScrollTrendtoTime** : les fenêtres de tendance seront centrées dès le déclenchement de l'événement.

ConsumptionMeterPageScroll

Fait défiler la page vers la gauche ou vers la droite dans un widget du compteur de consommation.

Paramètre	Description
Trend Name	ID de widget de tendance (par exemple, FenêtreTendances3)
Page Scroll Direction	Direction du défilement d'une page (Avant/Arrière)

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



Actions éditeur de texte

Macros utilisées pour interagir avec le widget TextEditor.

Voir "Widget ÉditeurTexte" Page 407 pour les détails

Actions de gestion des utilisateurs

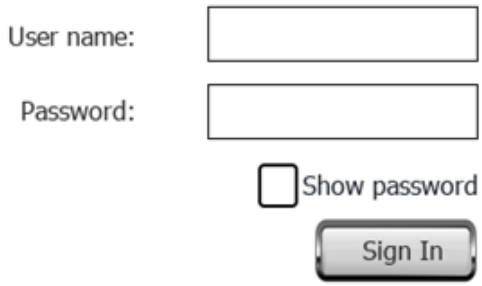
Paramètres de gestion des utilisateurs et de sécurité.

LogOut

Ferme la session de l'utilisateur en cours. L'utilisateur par défaut est ensuite connecté automatiquement. Si aucun utilisateur par défaut n'a été configuré, la fenêtre de connexion s'affiche.

SwitchUser

Commute entre deux utilisateurs sans déconnecter l'utilisateur connecté : le dialogue de connexion utilisateur s'affiche. L'utilisateur peut cliquer sur **Retour** pour retourner à l'utilisateur précédemment connecté.



Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Le serveur continue de fonctionner avec l'utilisateur connecté précédemment jusqu'à ce que l'utilisateur suivant se connecte. Un utilisateur reste toujours connecté au système.

ChangePassword

Modifie le mot de passe de l'utilisateur en cours : un dialogue s'affiche

Aucun paramètre requis.

ResetPassword

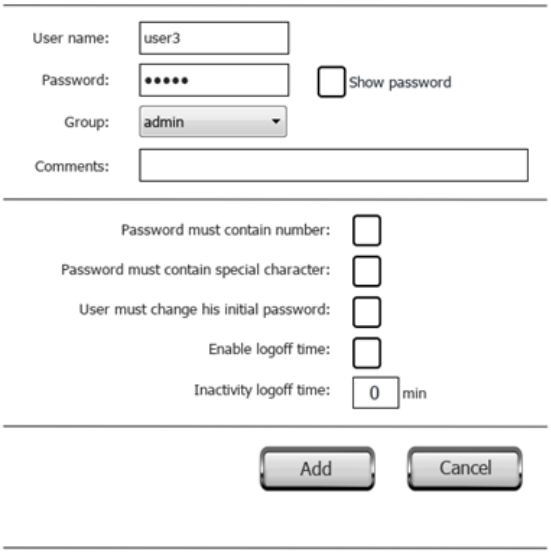
Restaure le mot de passe d'origine avec les paramètres spécifiés dans le projet pour l'utilisateur en cours.

Aucun paramètre requis.

AddUser

*Réservé aux utilisateurs pour lesquels la propriété **Can manage other users** est définie.*

Ajoute un utilisateur au runtime : un dialogue s'affiche.



DeleteUser

*Réservé aux utilisateurs pour lesquels la propriété **Can manage other users** est définie.*

Supprime un utilisateur au runtime : un dialogue s'affiche.

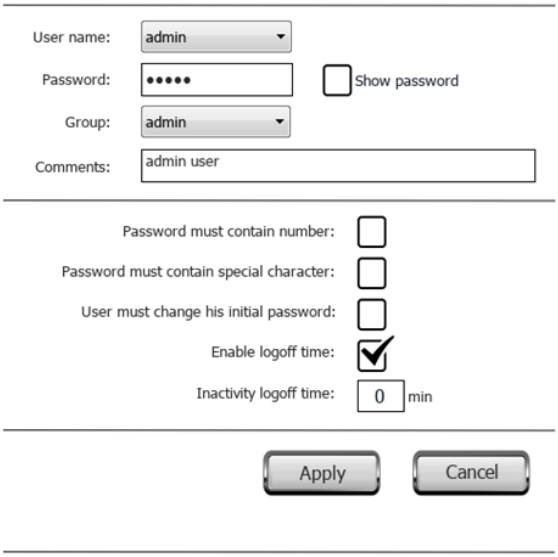
Aucun paramètre requis.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

EditUsers

*Réservé aux utilisateurs pour lesquels la propriété **Can manage other users** est définie.*

Permet de modifier les paramètres des utilisateurs.



DeleteUMDynamicFile

Supprime le fichier de gestion dynamique des utilisateurs. Les modifications apportées aux paramètres utilisateur au runtime sont effacées. Les paramètres d'origine sont restaurés à partir des informations du projet.

Aucun paramètre requis.

ExportUsers

Exporte les paramètres des utilisateurs dans un fichier .xml (*usermgnt_user.xml*) crypté qui permet la restauration le cas échéant.

Définissez le fichier de destination du fichier d'export.

Important : le fichier d'utilisateur est crypté et ne peut pas être modifié.

Remarque : les formats pris en charge sont FAT ou FAT32. NTFS n'est pas pris en charge.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

ImportUsers

Importe les paramètres des utilisateurs à partir d'un fichier d'export .xml précédemment auvegardé (*usermgnt_user.xml*).

Définissez le fichier source du fichier d'import.

Remarque : les formats pris en charge sont FAT ou FAT32. NTFS n'est pas pris en charge.

Actions widget

ShowWidget

Affiche ou masque les widgets de la page.

Propriété	Description
Widget	Widget à afficher/masquer

SlideWidget

Affiche l'effet de glissement d'un widget ou d'un groupe de widgets.

Remarque : le widget ou les widgets groupés peuvent se situer en-dehors de la partie visible de la page du projet et glisser pour s'afficher et sortir de la vue.

Propriété	Description
Widget	Widget à glisser
Direction	Direction du glissement
Vitesse	Vitesse de déplacement du widget glissant
Distance X	Distance parcourue par la coordonnée X en pixels
Distance Y	Distance parcourue par la coordonnée Y en pixels
Limite de défilement	Activer/désactiver les limites de mouvement du widget par rapport aux coordonnées x et y
Limite X	Position de limite de l'action de glissement par rapport à la coordonnée x
Limite Y	Position de limite de l'action de glissement par rapport à la coordonnée y
Inverser visibilité	Affiche/masque le widget après chaque action de glissement
Image Widget	Image affichée pendant l'action de glissement

BeginDataEntry

Affiche un clavier et démarre l'entrée de données dans un champ de données sans toucher le widget même. Cette action permet d'activer l'entrée de données à l'aide d'un scanner de code-barres.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Interface Java Script

```
project.beginDataEntry(wgtName [, pageName])
```

Paramètre	Description
wgtNameWidget	Nom de widget
Nom de page	Active la page pour l'entrée de données. Utile pour sélectionner un champ de données dans une boîte de dialogue non-modale active.

TriggerIPCamera

Capture une image à l'aide d'une caméra IP. Fonctionne uniquement avec les pages qui contiennent un widget caméra IP.

MoveIPCamera

Envoie des commandes à distance à une caméra qui les prend en charge. Voir "Widgets Caméra IP" Page 381 pour les détails. Assurez-vous que la caméra IP prend en charge les commandes de déplacement.

RefreshEvent

Rafraîchit la mémoire d'événements pour le widget de l'**Historique d'alarmes**. Voir "Widget historique des alarmes" Page 212 pour les détails.

ContextMenu

Affiche le menu contextuel.

Si la propriété **Context Menu** du Widget Projet a été définie sur **On delay**, le menu contextuel peut aussi apparaître en touchant l'arrière-plan de l'écran pendant quelques secondes. voir "Propriétés de projet" Page 69

ReplaceMedia

Remplace les fichiers média existants par de nouveaux fichiers stockés sur un lecteur USB /une carte SD. peut être utilisé pour remplacer les fichiers vidéo des widgets MediaPlayer ou des images du projet.

Remarque : les nouveaux fichiers médias doivent avoir le même nom et être au même format que les fichiers à remplacer.

Paramètre	Description
Media Type	Type du fichier à mettre à jour
Périphérique	Périphérique où les nouveaux fichiers médias sont fournis
sourcePath	Fichier où sont stockés les nouveaux fichiers médias (par exemple "\\USBMemory")
Image Resize	Redimensionne les nouvelles images à la taille des images à remplacer. Ne s'applique pas aux fichiers vidéo.
Silent	Remplace automatiquement les médias. Par défaut, un dialogue demande à l'utilisateur d'indiquer l'emplacement du fichier.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Interface Java Script

```
void replaceMedia(var sourcePath, var bSilent, var Device, var nMediaType, var
bResize)
```

```
project.replaceMedia("Images", true, "\USBMemory", 1, true);
```


Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

12 HMI Client

HMI Client est une application autonome qui fournit un accès à distance à HMI Runtime et est inclus dans le PB610 Panel Builder 600. HMI Client utilise le même système de rendu graphique que le runtime dans les périphériques IHM, il repose sur un HMI Runtime spécifique en tant que serveur pour les données en direct.

HMI Client agit comme un client à distance et communique avec le serveur, en partageant la visualisation locale avec les valeurs de variable qui sont maintenues ou mises à jour par le protocole de communication.

Les projets IHM contiennent des propriétés indiquant quelle page est actuellement affichée sur l'IHM et peut forcer l'IHM pour passer à une page spécifique. Vous pouvez utiliser ces propriétés pour synchroniser les pages affichées sur le périphérique IHM et HMI Client ou pour commander un périphérique IHM avec un API. Voir "Projet" Page 76 pour les détails.

Pour éviter tout comportement inattendu, assurez-vous d'utiliser la même version de HMI Runtime.

Application client sur ordinateur **186**

Application client sur IHM **187**

Paramètres et options de fuseau horaire **188**

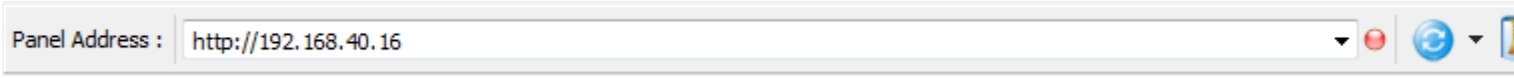
Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Application client sur ordinateur

Pour exécuter l'application HMI Client sur ordinateur :

1. Depuis le menu **Start** > **PB610 Panel Builder 600** >**HMI Client** : le client ouvre dans une fenêtre style navigateur.
2. Saisissez l'adresse IP du serveur/périphérique dans la barre d'adresse (par exemple : http://192.168.1.12) : HMI Client se connectera au serveur et la même application graphique s'exécutant sur le périphérique sera chargée dans la fenêtre du client.

La barre d'outils de l'application client



Élément	Description
Adresse de serveur IHM.	Adresse IP du périphérique IHM distant (p. ex. 192.168.0.1:80)
État de la connexion	État de la demande de réseau. Rouge lors de l'échange de données.
Recharger à partir du cache	Recharger projet
Marque-page	Marquer les pages préférées et les recharger.
Paramètres	Ouvrir la boîte de dialogue des **réglages**

Options de rechargement

Option	Description
F5	Recharge le projet depuis le cache
Maj + F5	Télécharge le projet sur le client

Transférer des fichiers à un périphérique IHM distant

Vous pouvez charger et télécharger des fichiers du et vers un périphérique IHM distant en utilisant deux actions dédiées. Ces actions peuvent être exécutées uniquement à partir d'un HMI Client distant et accéder à des fichiers distants via FTP.

Voir "Actions client à distance" Page 160 et "Variables client à distance" Page 105.

Important : activez la prise en charge du FTP et attribuez tous les droits d'utilisateurs aux dossiers utilisés pour le transfert des fichiers.

Espace de travail

Les fichiers du projet sont chargés depuis le périphérique et stockés dans HMI Client dans le dossier cache suivant.

%appdata%\ABB\[build number]\client\cache

où :

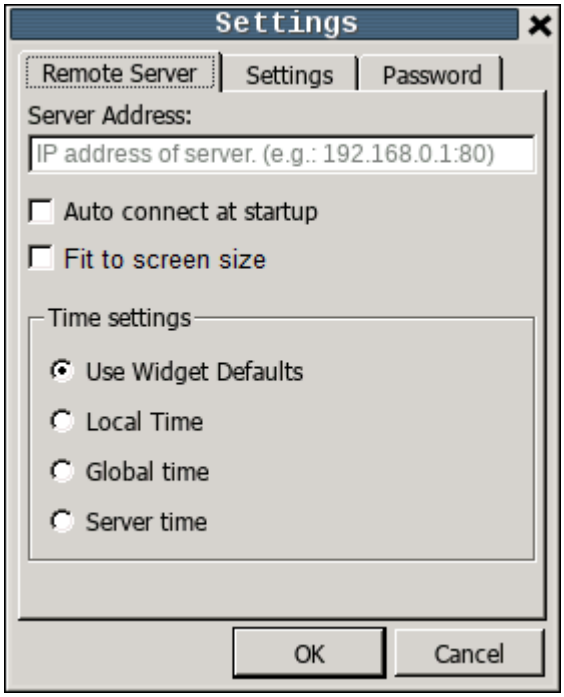
[build number] = dossier nommé en tant que numéro de fabrication, par exemple 01.90.00.608.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Application client sur IHM

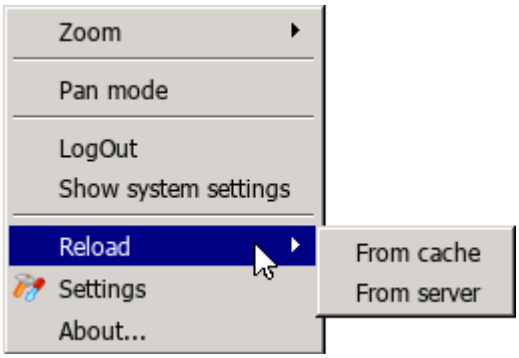
Pour exécuter l'application HMI Client sur un périphérique IHM sous Linux :

1. Dans le menu **Exécuter** > **Mettre à jour un paquet**, créez un paquet mis à jour et installez l'application Client IHM sur le périphérique IHM. (voir "Mettre à jour un paquet" Page 91 pour des informations complémentaires)
2. Saisissez l'adresse IP du serveur/périphérique dans la boîte de dialogue Configuration qui sera disponible au démarrage du périphérique IHM (par exemple : http://192.168.1.12) : le client IHM se connectera au serveur et la même application graphique s'exécutant sur le périphérique sera chargée dans la fenêtre du client.



Menu contextuel

Le menu contextuel, disponible par un clic droit de souris, affichera les commandes ci-dessous :



Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

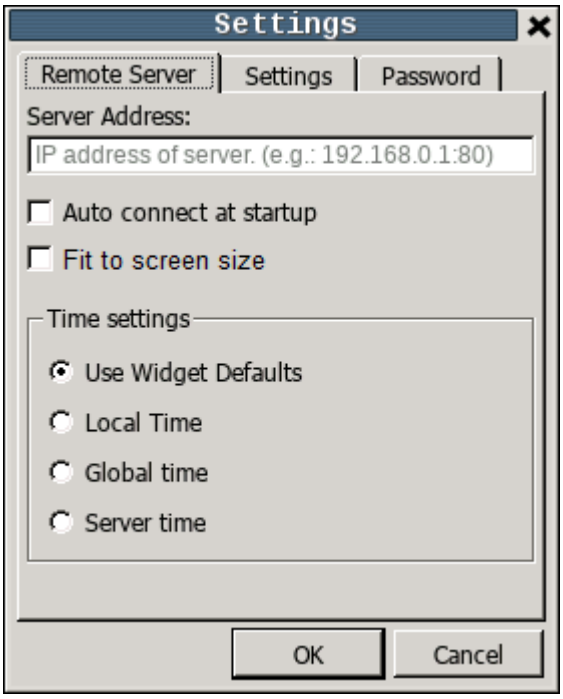
Option	Description
Zoom	Sélectionner la taille de la vue au runtime • Zoom avant • Zoom arrière • Zoom 100%
Mode panoramique	Active/désactive le mode panoramique après un zoom avant
Déconnexion	Déconnecte l'utilisateur actuel.
Affiche les paramètres système	Autoriser les paramètres IHM et la gestion des composants du système. Voir "Paramètres du système" Page 477 pour les détails.
Recharger	Recharge le projet distant • Depuis le cache • Depuis le serveur
Paramétrage	Ouvrez les Paramètres du Client IHM. Voir "Paramètres et options de fuseau horaire" bas pour les détails Peut être protégé par un mot de passe
À propos	Affiche les informations sur la version du Client IHM.

Paramètres et options de fuseau horaire

Dans la boîte de dialogue **Paramètres** vous pouvez configurer les paramètres du client et décider comment afficher les informations d'horodatage du projet.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Serveur distant



Paramètres de connexion

Paramètre	Description
Adresse du serveur	Adresse IP du périphérique IHM distant (p. ex. 192.168.0.1:80)
Connecter automatiquement au démarrage	Lorsque le panneau démarre, utilisez l'adresse du serveur pour essayer de vous connecter automatiquement au serveur distant.
Ajuster aux dimensions de l'écran	Adapte la vue aux dimensions de l'écran

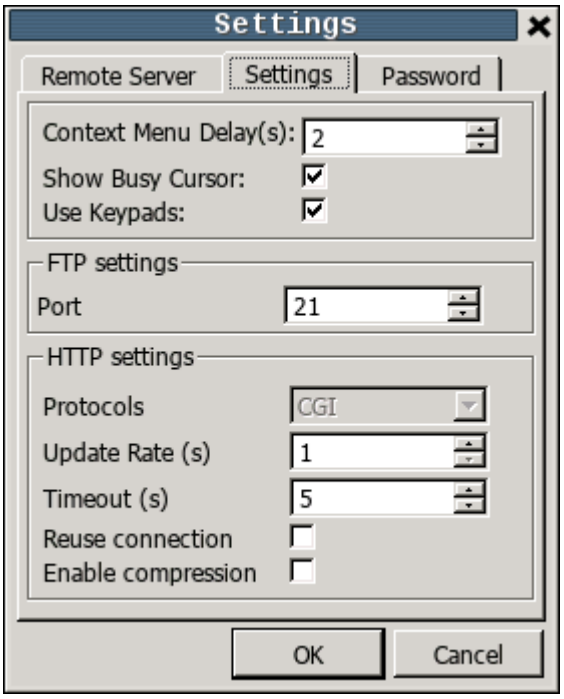
Paramètres de temps

Paramètre	Description
Utiliser widgets par défaut	Affiche des informations de temps en fonction des paramètres du widget.
Heure locale	Traduit tous les horodatages du projet dans l'heure locale de l'ordinateur, où le client est installé.
Heure globale	Traduit tous les horodatages du projet au format UTC
Heure du serveur	Traduit tous les horodatages du projet dans le même projet utilisé par le périphérique/serveur IHM afin d'afficher dans le même temps.

Important : assurez-vous de définir la zone horaire correcte RTC de l'IHM et les options DST.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Paramètres



Paramètres d'interface

Paramètre	Description
Retard(s) menu contextuel	Retard de l'activation du menu contextuel. Plage : 1 à 60 secondes.
Afficher curseur occupé	Affiche un sablier lorsque le système est occupé
Utiliser des claviers	Affiche les claviers lorsque l'utilisateur touche un champ d'entrée des données. Défini sur **désactiver** lorsqu'un clavier USB externe est connecté au périphérique.

Paramètres FTP

Paramètre	Description
Port	Port communication FTP

Paramètres HTTP

Paramètre	Description
Protocoles	Le protocole de communication utilisé par HMI Client pour communiquer avec un périphérique IHM.
Fréquence de mise à jour	La fréquence de scrutation pour synchroniser les données depuis le serveur. Défaut = 1 s.
Délai d'attente	Temps d'attente maximum avant qu'une demande est répétée par le HMI Client. Défaut = 5 s.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Paramètre	Description
Réutiliser connexion	Permet la réutilisation de la même connexion TCP pour plusieurs requêtes HTTP pour réduire le trafic réseau. Remarque : lorsqu'elle est activée, cette option peut entraîner une latence élevée si le serveur proxy ne met pas fin immédiatement aux anciennes demandes saturant ainsi les prises de connexion. C'est souvent le cas avec les connexions 3G.
Activer la compression	Compresse les données afin de réduire le temps de téléchargement. Défaut = désactivé. ***ATTENTION : activer cette option pourrait provoquer une surcharge excessive du processeur.***
Paramètres de temps	Utilisé par le client pour adapter les informations d'horodatage des widgets.

Mot de passe

Cette boîte de dialogue offre la possibilité de changer le mot de passe interne de l'IHM pour l'utilisateur administrateur (le mot de passe par défaut est « admin »).

La protection par mot de passe n'est pas disponible sur la version ordinateur du client IHM

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

13 Utiliser le serveur FTP intégré

HMI Runtime le système utilise un serveur FTP intégré.

Connectez-vous au serveur FTP du périphérique IHM en utilisant n'importe quelle application client FTP ordinaire. Le serveur FTP répond sur le port 21 par défaut.

Important : le serveur ne prend en charge qu'une seule connexion à la fois ; si vous utilisez un client FTP avec des connexions multiples, désactivez cette fonction sur le programme client ou définissez le nombre maximal de connexions par session sur 1.

Paramètres FTP

Identifiants par défaut FTP

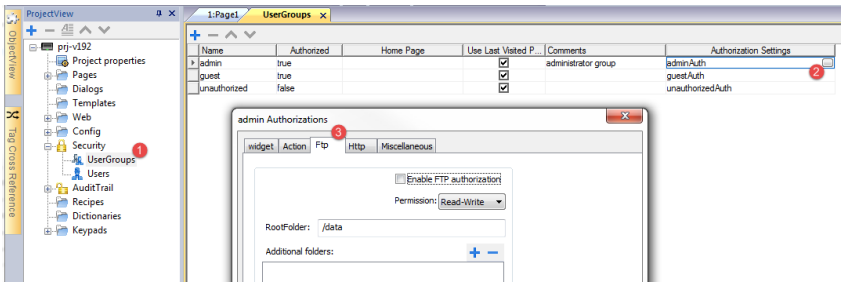
Lorsque la gestion/sécurité de l'utilisateur est désactivée, utiliser les identifiants suivants pour les connexions entrantes :

Identifiant	admin
Mot de passe	admin

Modifier les paramètres FTP

*Chemin : **VueProjet*> *Sécurité*> *GroupesUtilisateurs*** > ***Paramètres autorisation***

Vous pouvez modifier les autorisations FTP et les informations de compte dans l'onglet **Ftp** de la boîte de dialogue des **autorisations Administrateur**.



Voir "Configurer groupes et autorisations" Page 285 pour les détails.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

14 Utiliser VNC pour accès à distance

VNC est un logiciel de contrôle à distance qui vous permet de voir et de commander l'application IHM à distance en utilisant votre souris et votre clavier.

L'accès à distance est particulièrement utile pour l'administration et le support technique. Pour l'utiliser, vous devez :

- démarrer un serveur dans le périphérique IHM
- installer un afficheur sur le périphérique à distance

Démarrage du serveur VNC sur les périphériques WinCE **196**

Démarrage du serveur VNC sur les périphériques Linux **197**

Démarrer la visionneuse VNC **198**

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Démarrage du serveur VNC sur les périphériques WinCE

Le serveur VNC est un module d'exécution. Il peut être activé et téléchargé comme faisant partie du Runtime. Voir "Module d'exécution (plug-in)" Page 75.

Installer le serveur VNC

*Chemin : **VueProjet** > **Propriétés du projet***

1. Dans le volet **Propriété**, définissez le **Serveur VNC** sur **vrai** pour activer le module d'exécution.
2. Installez ou mettez à jour le runtime pour ajouter le serveur VNC.

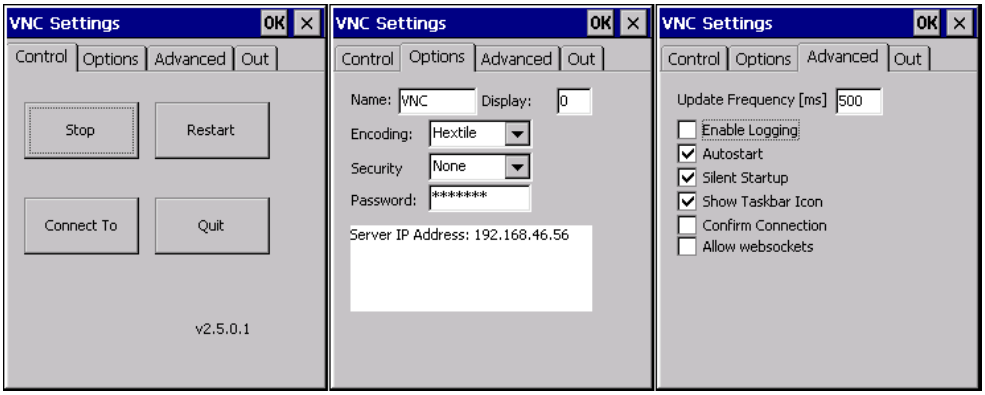
Démarrage/arrêt du serveur VNC

Le serveur VNC se trouve dans le dossier *\Flash\qthmi\VNC* et est activé par l'action *launchVNC*. S'il est activé à partir des propriétés du projet, il peut également être activé à partir du menu contextuel du runtime **Outils du développeur**> **Lancer VNC**.

Pour activer le menu contextuel du runtime, voir "Propriétés de projet" Page 69

Boîte de dialogue Options VNC

Dans la boîte de dialogue **Options VNC**, il est possible d'exécuter plusieurs tâches.



Onglet	Fonctions
Commande	Démarrer/arrêter le serveur VNC et connecter l'afficheur
Options	Définir les informations de sécurité pour l'accès au serveur en utilisant un afficheur VNC

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Onglet	Fonctions
Paramètres avancés	Activer l'activation automatique du serveur VNC au démarrage du périphérique IHM. Sélectionner **Démarrage silencieux** pour garder la boîte de dialogue des **Options VNC** en arrière-plan lorsque l'**Autodémarrage** est activé. *Activer **Afficher l'icône de la barre des tâches** en déboguant le mode KIOSK.*
Out	Contient les paramètres de configuration pour une connexion sortante à un logiciel d'affichage VNC d'écoute.

Important : les paramètres dans l'onglet Avancé sont réservés aux utilisateurs experts et doivent être modifiés lorsque le serveur VNC est utilisé en conjonction avec un répéteur VNC pour surmonter les problèmes de pare-feu ou optimiser les performances du VNC en fonction de la configuration du réseau.

Connexion à la visionneuse

De nombreux afficheurs VNC modernes offrent la possibilité de lancer le logiciel en mode écoute. C'est parce que la plupart du temps les périphériques mobiles n'ont pas d'adresse IP publique de référence. Il est donc utile d'avoir une adresse IP publique sur un ordinateur de bureau qui exécute une visionneuse VNC d'écoute. Un utilisateur peut alors facilement appeler le support en appuyant sur le bouton **Se connecter à la visionneuse** sur l'onglet Commande.

Paramètres VNC par défaut

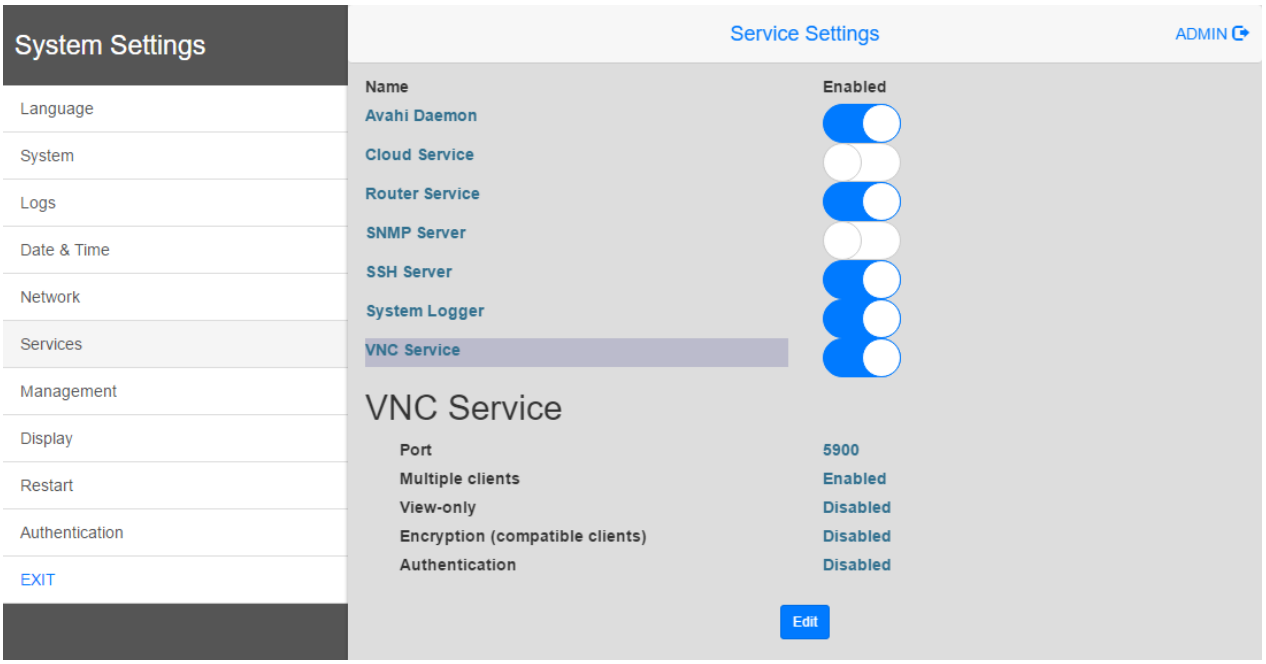
Port TCP	5900
Mot de passe	null

Important : le serveur VNC n'autorise qu'un seul client.

Démarrage du serveur VNC sur les périphériques Linux

Le serveur VNC est un service intégré au BSP qui peut être activer à partir de l'onglet Services des paramètres système du périphérique. Voir "Paramètres du système" Page 481 pour les détails.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



Démarrer la visionneuse VNC

Aucune visionneuse VNC n’est fournie avec PB610 Panel Builder 600.

De nombreuses visionneuses VCN compatibles sont disponibles en téléchargement gratuit (par exemple, TightVNC).

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

15 Alarmes

Le système de gestion des alarmes a été conçu pour fournir des alertes par messages affichés dans une fenêtre contextuelle, qui sont généralement des messages d'avertissement indiquant toute condition ou tout dysfonctionnement anormal dans le système sous contrôle.

Un message s'affiche lorsqu'un bit change ou lorsque la valeur d'une variable dépasse un seuil fixé dans la configuration de l'alarme. Vous pouvez également programmer des actions spécifiques qui doivent s'exécuter lorsqu'une alarme est déclenchée.

Important : aucune action par défaut n'est associée aux alarmes.

Vous pouvez définir le mode d'affichage d'une alarme sur le périphérique IHM, lorsque l'utilisateur doit la confirmer, ainsi que le moment de l'enregistrement et le mode d'enregistrement dans la liste d'événements.

Les alarmes sont configurées dans l'Éditeur de configuration des alarmes et sont ainsi disponibles pour toutes les pages du projet. Un widget d'alarme peut afficher plus d'une alarme à la fois, si elles sont dimensionnées de manière appropriée. Vous pouvez déclencher l'ouverture ou la fermeture de la fenêtre de l'alarme avec un événement.

Vous travaillez avec les alarmes de la même manière qu'avec tout autre événement. Vous ne pouvez pas afficher de boîte de dialogue lorsqu'une alarme est déclenchée et vous pouvez l'associer à toute autre action disponible.

Éditeur d'alarmes **200**

Confirmer les alarmes à distance **203**

Machine à états pour alarme **204**

Configurer des événements **204**

Widget alarmes actives **207**

Widget historique des alarmes **212**

Gestion des alarmes au runtime **212**

Activer/désactiver les alarmes au runtime **213**

Afficher les données d'alarme actives **213**

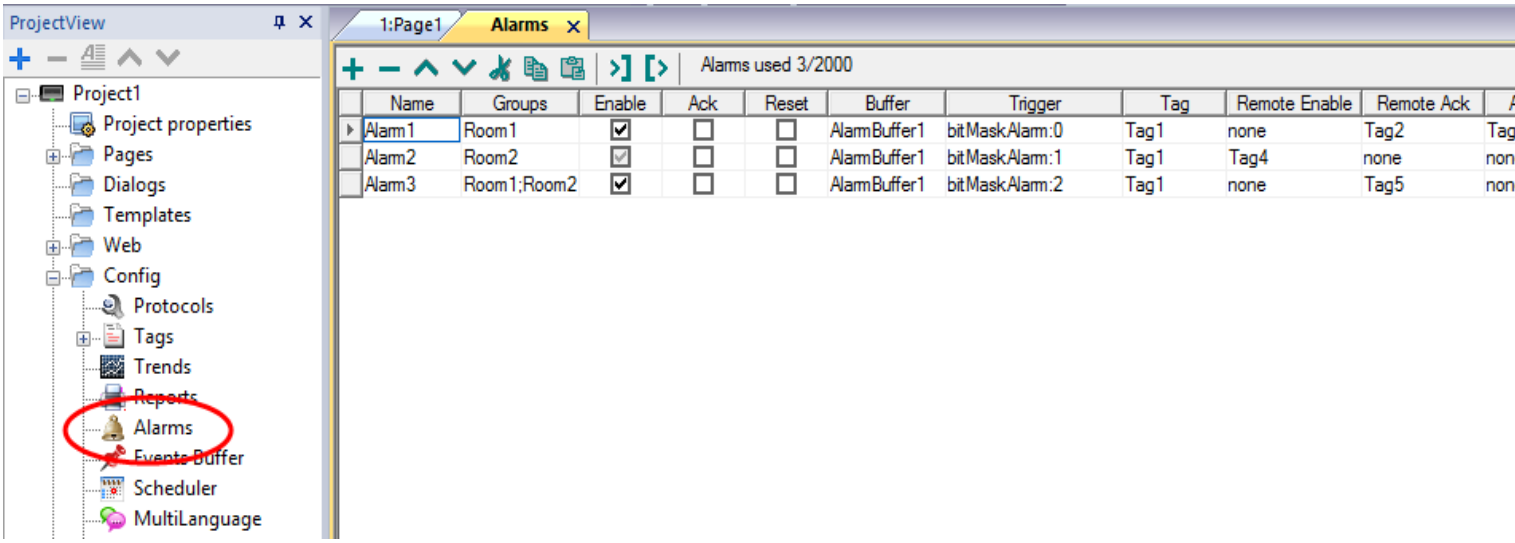
Exporter la mémoire tampon des alarmes vers fichiers .csv **214**

Exporter la configuration des alarmes **215**

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Éditeur d'alarmes

Chemin : ***VueProjet*** *>* ***Configuration*** *> double-cliquez sur* ***Alarmes***



Ajouter une alarme

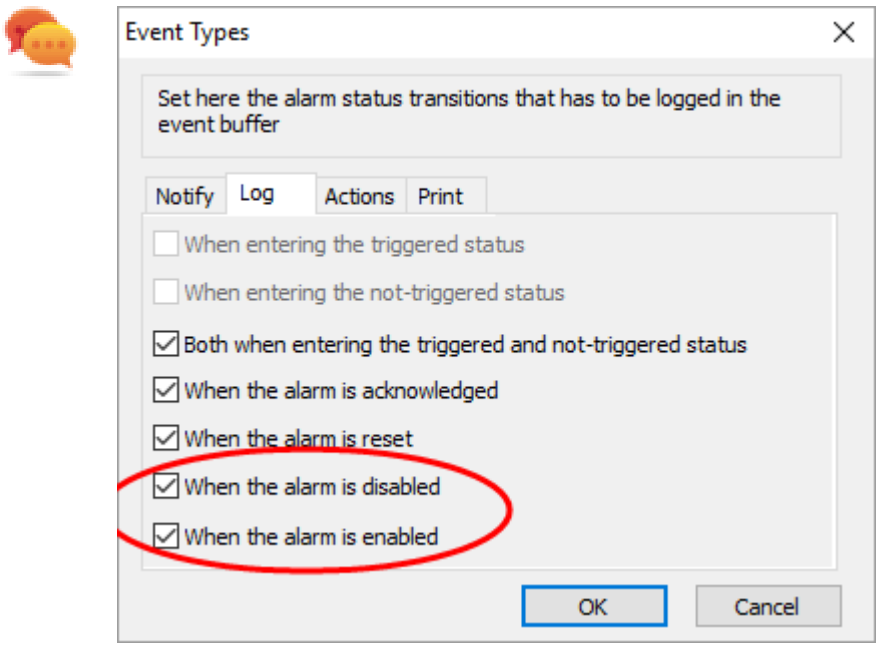
Cliquez sur **+** pour ajouter une alarme.

Paramètre	Description
Nom	Nom de l'alarme
Groupes	Groupes associés à une alarme. Ils peuvent être utilisés dans des filtres d'affichage des widgets.
Activer	Activer/désactiver le déclenchement de l'alarme. Les alarmes peuvent également être activées ou désactivées au runtime (voir "Activer/désactiver les alarmes au runtime" Page 213 pour plus de détails).
Conf	Active/désactive la confirmation de l'alarme. Si elle est sélectionnée, l'opérateur doit confirmer l'alarme une fois déclenchée pour la supprimer du widget **Alarme active**.
Réinitialiser	Utilisé avec l'option **Conf** lorsqu'elle est sélectionnée, les alarmes confirmées restent dans la liste des alarmes, étiquetées comme **Conf non déclenchées**, jusqu'à ce que l'opérateur appuie sur le bouton **Réinitialiser** dans le widget de l'alarme.
Mémoire tampon	Fichier tampon où l'historique des alarmes est enregistré.
Déclencheur	La condition de déclenchement dépend du type d'alarme : • **limitAlarm** : alarme déclenchée lorsque la valeur de variable dépasse ses limites. L'alarme n'est pas déclenchée si la valeur atteint les limites. • **valueAlarm** alarme déclenchée lorsque la valeur de variable est égale à la valeur configurée

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Paramètre	Description
	• **bitMaskAlarm** : l'opérateur ET au niveau des bits compare chaque bit du masque binaire avec la valeur de variable correspondant à cette alarme. Si les deux bits sont activés, l'alarme est définie sur Vraie. Vous pouvez spécifier une ou plusieurs positions binaires (à partir de 0) à l'intérieur de la variable. La position du bit doit être donnée en format décimal ; si plusieurs bits sont spécifiés, chaque position doit être séparée par une « , ». • **deviationAlarm** : alarme déclenchée si le pourcentage de déviation de la valeur de variable depuis le point défini dépasse une déviation définie. $$\left|Value_{now} - SetPoint\right| > \left(\frac{deviation}{100} \times SetPoint\right)$$ • **protAlarm:BACN** : l'alarme est déclenchée à partir du périphérique BACnet distant. Ce mode de déclenchement n'est disponible que lorsque la configuration des protocoles contient un périphérique BACnet. Reportez-vous au manuel BACnet dans le dossier « Pilotes de communication » pour une description détaillée des événements d'alarmes BACnet.
Variable	Variable dont la valeur déclenche l'alarme lorsqu'elle dépasse les limites définies. L'alarme peut faire référence à la valeur de cette variable ou à l'état d'un bit lorsque **bitMaskAlarm** a été sélectionné comme déclencheur.
Activer à distance	Variable utilisée par l'API pour activer/désactiver l'alarme. • Changer l'état Activer du widget Alarmes change la valeur de variable • Lorsque la variable ne peut pas être lue (p. ex. erreur de communication), l'alarme est désactivée • Aucune variable reliée à l'alarme n'est rafraîchie lorsque l'alarme est désactivée. Conseil : il peut être utile d'activer l'enregistrement du drapeau d'activation de l'alarme.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Paramètre	Description
	
Acquittement à distance	Variable utilisée par l'API pour confirmer l'alarme. Une transition de cette variable de 0 à une valeur différente de zéro est considérée comme une requête de confirmation. Laisser vide lorsque l'acquittement à distance n'est pas requise. Voir "Confirmer les alarmes à distance" Page opposée pour plus de détails.
Notification d'acquittement	Variable utilisée par le périphérique IHM pour notifier lorsque l'alarme est acquittée depuis le périphérique ou depuis l'API. 0 = définir cette valeur lorsque l'alarme est déclenchée 1 = définir cette valeur lorsque l'alarme est acquittée.
Action	Actions exécutées lorsque l'alarme est déclenchée. Des conditions supplémentaires peuvent être spécifiées dans la colonne **Événements**. Voir "Configurer des événements" Page 204 pour plus de détails.
Action utilisateur	Actions exécutées lorsque l'utilisateur appuie sur le bouton d'action dans le widget de l'alarme active. Voir "Widget alarmes actives" Page 207 pour plus de détails.
Description	Description de l'alarme. Ce texte prend en charge les fonctionnalités multilangue et peut être une combinaison de parties statiques et dynamiques, ces dernières incluant une ou plusieurs valeurs de variable. Voir "Afficher les données d'alarme actives" Page 213 pour plus de détails.
Champ personnalisé #	Il s'agit d'une description complémentaire de l'alarme pouvant être utilisée pour afficher des informations supplémentaires à l'intérieur des widgets d'alarme. Par exemple, ce pourrait être un index à utiliser pour afficher une image relative à l'alarme.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

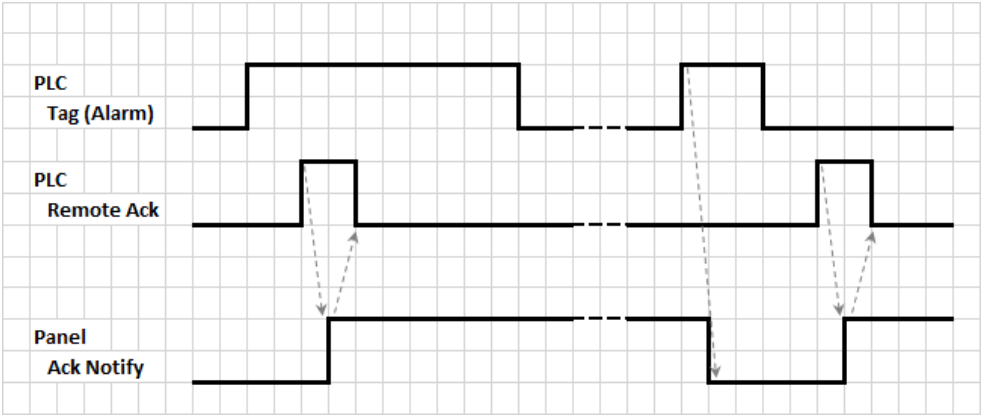
Paramètre	Description
Couleur	Couleurs de premier plan et d'arrière-plan des lignes des alarmes, basées sur l'état de l'alarme.
ConfirmationClignotement	Clignotement si alarmes déclenchées. Lorsque cette action est sélectionnée, les lignes de l'alarme clignotent jusqu'à leur confirmation. Effectif seulement si **Confirmation** est sélectionné.
Sévérité	Sévérité de l'alarme. Si plusieurs alarmes sont déclenchées simultanément, les actions seront exécutées selon les paramètres de sévérité. 0 = pas importante 1 = basse 2 = en dessous de la normale 3 = normale 4 = au-dessus de la normale 5 = élevée 6 = critique
Événements	Conditions de notification, d'enregistrement ou d'impression des alarmes. Voir "Configurer des événements" Page suivante pour plus de détails.

Confirmer les alarmes à distance

Lorsque le paramètre **Confirmation à distance** est activé, une alarme peut être confirmée à partir d'un périphérique API en introduisant une valeur de variable différente de zéro. Le statut confirmé est notifié sur le périphérique API par le drapeau **Notification de confirmation**.

Processus de confirmation des alarmes

La variable **Acquittement à distance** est réglée/réinitialisée par l'API pour demander la confirmation, tandis que la **Notification d'acquittement** est réglée/réinitialisée par le périphérique IHM pour notifier l'exécution de la confirmation.



Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

1. Lorsqu'une condition d'alarme est détectée, le périphérique IHM règle **Notification d'acquittement** sur 0 et l'ensemble des actions correspondantes sont exécutées.
2. Lorsque l'alarme est acquittée (par le périphérique IHM ou à distance), la **Notification d'acquittement** est réglée sur 1
3. Il appartient au contrôleur de régler **Acquittement à distance** sur 1 pour confirmer l'alarme ou de la remettre sur 0 lorsque le périphérique IHM envoie une notification pour signaler que l'alarme a été confirmée (**Confirmation de notification** = 1)

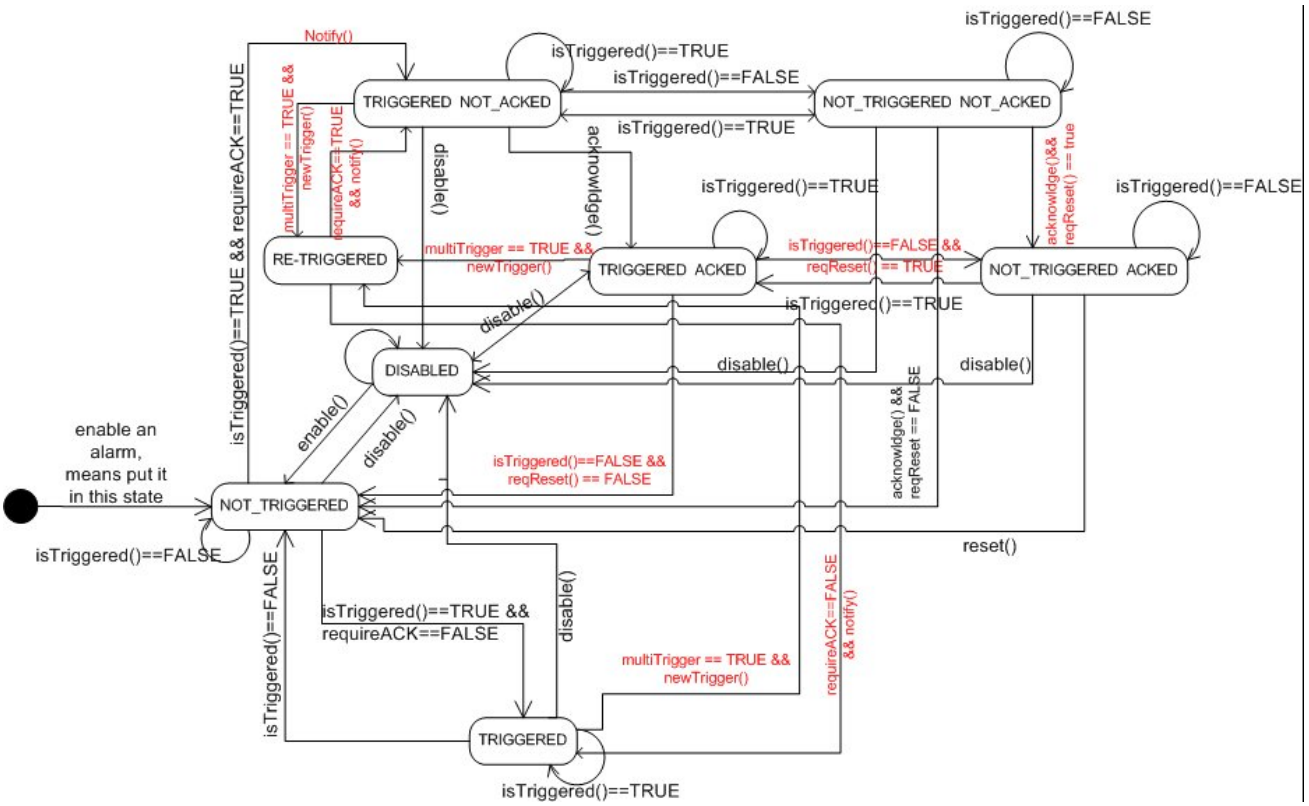
AVERTISSEMENT : lorsqu'une alarme est déclenchée, certains signaux doivent être mis à jour/communiqués vers les périphériques connectés. La Confirmation est censée être un signal activé par un opérateur, non un signal déclenché automatiquement par un périphérique de contrôleur. Ceci permet de tenir compte du temps nécessaire pour communiquer les signaux originaux.

Conseil : l'utilisation d'une même variable pour **Confirmation à distance** et **Notification de confirmation** permet de connecter plusieurs périphériques au même contrôleur et de confirmer les alarmes à partir de n'importe quel périphérique IHM.

Machine à états pour alarme

Le runtime implémente la machine à états pour alarme décrite dans ce schéma.

Les états et les transitions entre états sont décrits selon les options sélectionnées et le comportement voulu.



Configurer des événements

Chemin : **VueProjet**> **Config** > **Alarmes** > colonne **Événements**

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

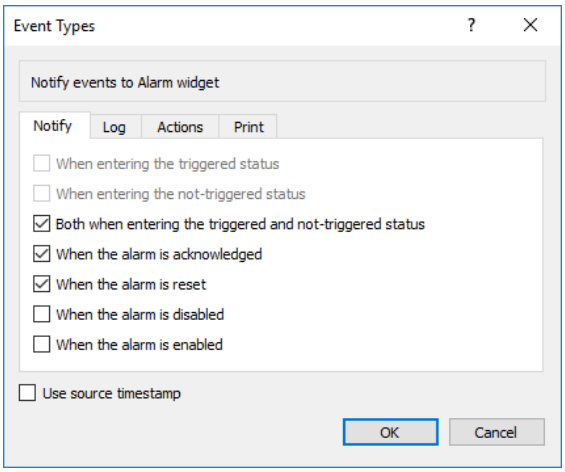
Les événements sont définis à l'aide de l'éditeur Alarmes.

Voir "Éditeur d'alarmes" Page 200 pour plus de détails.

Notification des événements

Chemin : **VueProjet**> **Config** > **Alarmes** > colonne **Événements** > onglet **Notification**

Conditions de réglage dans lesquelles les alarmes seront affichées dans le widget d'alarme.



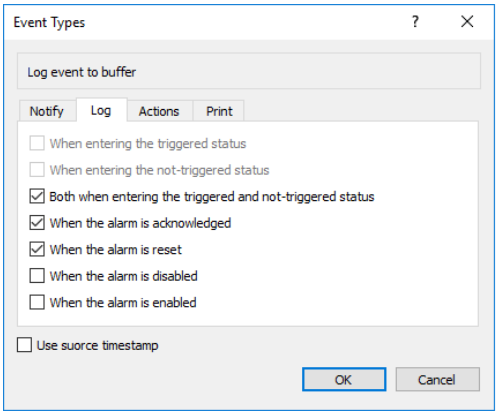
Vous définissez ici le comportement du widget d'alarme par défaut disponible dans la galerie de widgets et décidez dans quels cas le widget est mis à jour par un changement de statut d'alarme.

MISE EN GARDE : faire uniquement les ajustements requis par l'application spécifique tout en laissant tous les autres paramètres sur défaut.

Enregistrement d'événements

Chemin : **VueProjet**> **Config** > **Alarmes** > colonne **Événements** > onglet **Journaliser**

Définit les conditions sous lesquelles vous voulez stocker l'événement spécifique dans une mémoire tampon d'historique d'alarmes.



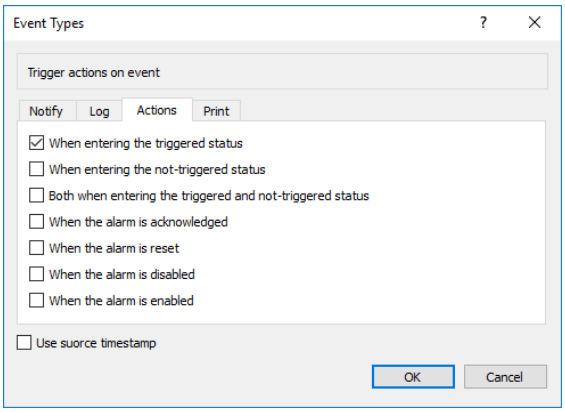
L'historique des alarmes est enregistré dans la mémoire tampon d'événements.

Actions d'exécution

Chemin : **VueProjet**> **Config** > **Alarmes** > colonne **Événements** > onglet **Actions**

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Conditions de réglage sous laquelle les/l'action(s), configuré pour l'alarme spécifique, doivent être exécutées.

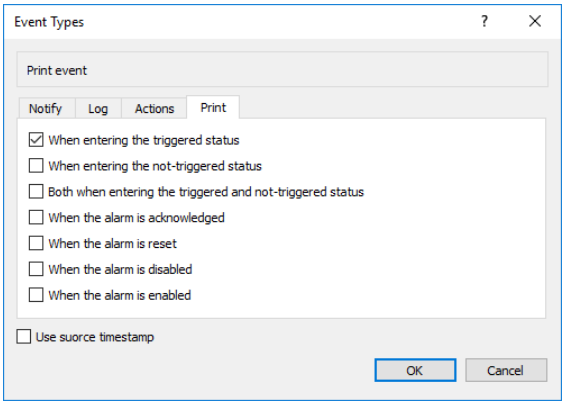


Par défaut, les actions sont exécutées uniquement lorsque l'alarme est déclenchée ; d'autres statuts d'alarme peuvent également être configurés pour exécuter des actions.

Événements d'impression

Chemin : **VueProjet**> **Config** > **Alarmes** > colonne **Événements** > onglet **Imprimer**

Définit les conditions sous lesquelles vous voulez imprimer l'événement spécifique

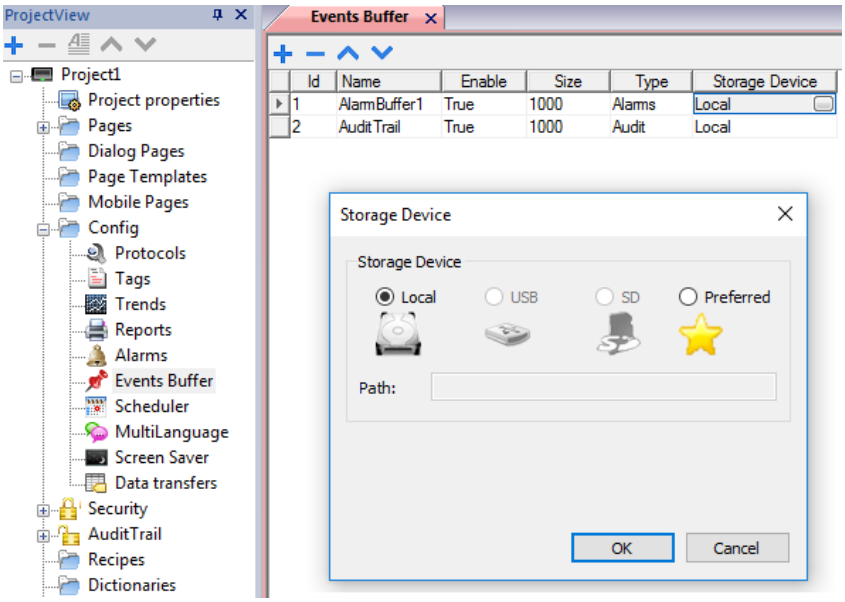


Paramétrage du périphérique de stockage

*Chemin : **VueProjet**> **Config** > **Mémoire Tampon événements**> onglet **Périphérique de stockage***

1. Ouvrez la boîte de dialogue du **périphérique de stockage**.
2. Sélectionnez un périphérique de stockage de données d'événements.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



Les données sont automatiquement enregistrées toutes les cinq minutes, sauf pour les données d'alarme qui sont enregistrée immédiatement.

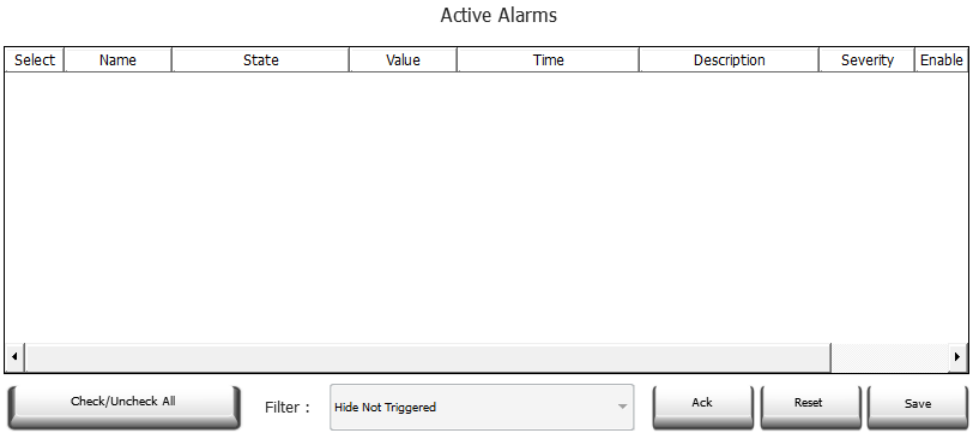
Utiliser l'horodatage de la source

Les événements sont stockés avec l'horodatage de l'instant où l'appareil IHM détecte l'événement. Lorsque « Utiliser l'horodatage de la source » est sélectionné, les événements sont stockés avec l'horodatage reçu du périphérique distant.

Disponible uniquement pour les protocoles du périphérique qui prennent en charge cette fonctionnalité (OPC UA Client et BACnet)

Widget alarmes actives

Vous pouvez insérer le widget **Alarmes actives** dans une page pour afficher les alarmes et pour en prendre connaissance, les réinitialiser ou les activer/désactiver.



Filtres d'alarme

*Chemin : **Alarme active** widget> **Propriétés** volet> **Filtre***

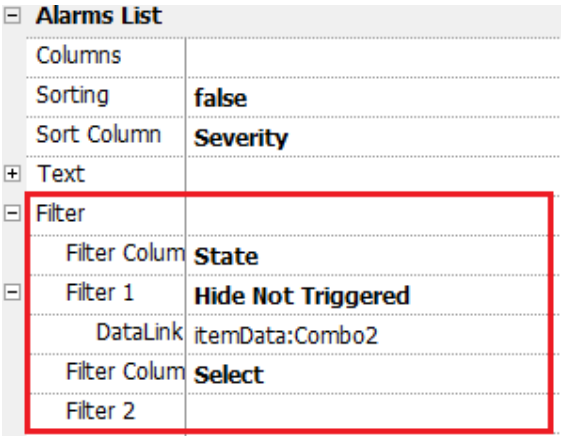
Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Définissez les filtres utilisés pour afficher uniquement certaines alarmes configurées. Les filtres sont basés sur des champs d'alarme, vous pouvez donc filtrer les alarmes en fonction du nom, de la sévérité, de la description, etc.

Filtre 1 est le filtre par défaut. Il est géré avec la liste déroulante **Filtre 1** et a 2 options : **Afficher toutes les alarmes** et **Masquer les alarmes non déclenchées** ; cette option une fois sélectionnée permet d'afficher uniquement les alarmes actives.

Le filtre 2 n'est par défaut pas configuré et peut être personnalisé.

Les expressions du filtre utilisent le langage AWK, les expressions sont appliquées aux données contenues dans la colonne **Filtre** sélectionnée du widget Alarme.



Paramétrer les filtres

Chemin : ***Alarme active*** *widget>* ***Propriétés*** *volet>* ***Filtre***

Pour définir un des deux filtres disponibles :

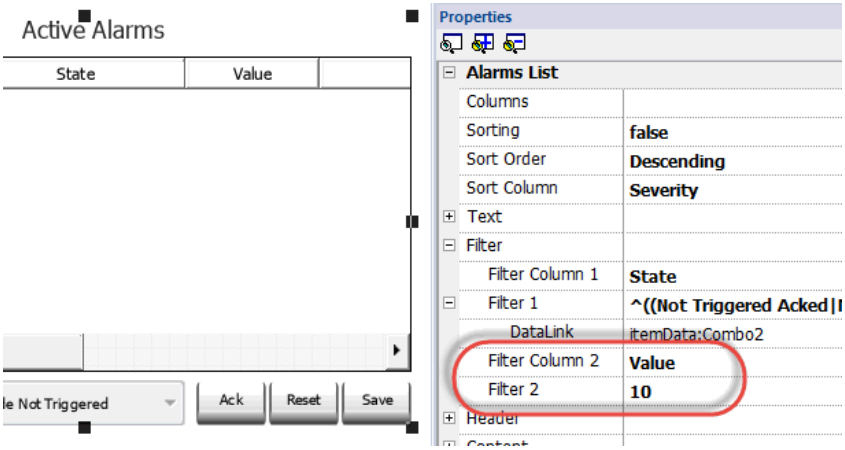
1. Sélectionnez **Filtre Colonne 1** et choisissez la valeur à filtrer (p. ex. : Nom, État, Temps, Groupes)
2. Dans **LiaisonDonnées**, associez un widget à liste déroulante. Appuyez sur Maj + faites un clic gauche pour sélectionner la liste déroulante.
3. Dans le volet **Propriétés**, sélectionnez la propriété dans la liste et ouvrez la boîte de dialogue pour personnaliser les valeurs de la liste déroulante
4. Dans la boîte de dialogue de configuration de la liste déroulante, spécifiez **Liste de chaînes** et l'expression régulière pour filtrer les valeurs.

Voir https://fr.wikipedia.org/wiki/Expression_régulière pour plus de détails sur les expressions régulières.

Filtres premier exemple

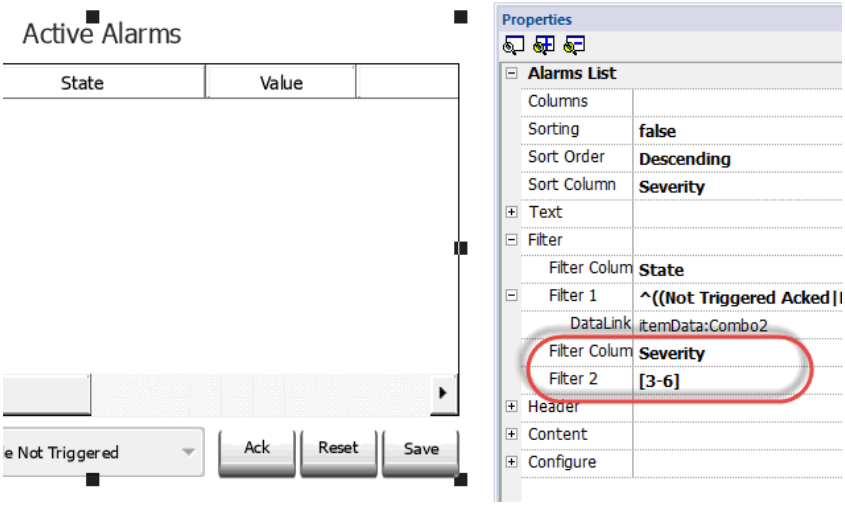
Vous voulez afficher toutes les alarmes correspondant au Filtre 1 d'une valeur égale à 10. Puis paramètres de propriétés :**Filtre Colonne 2** = Valeur, **Filtre 2** = 10

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



Deuxième exemple de filtres

Vous voulez afficher toutes les alarmes correspondant à une valeur de Sévérité de 1 à 6. Puis paramètres de propriétés :**Filtre Colonne 2** = Sévérité, **Filtre 2** = [3-6]



Troisième exemple de filtres

Vous voulez afficher toutes les alarmes correspondant à une valeur de 1 à 19. Puis paramètres de propriétés :**Filtre Colonne 2** = Sévérité, **Filtre 2** = ^(1[1-9]$)

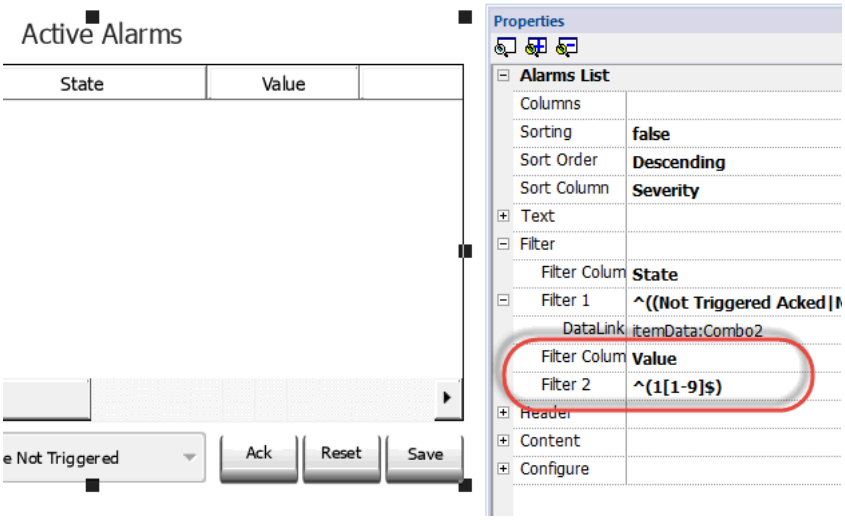
Signification :

^ = la correspondance doit commencer au début de la chaîne

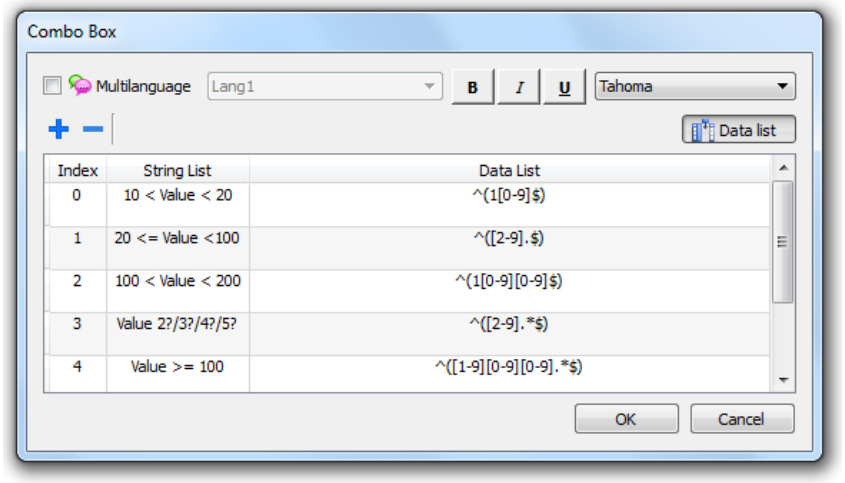
1[1-9] = le premier char doit être 1 et le deuxième char doit être compris entre 1 et 9

$ = fin de la comparaison.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



Exemples d'expression de filtres



Filtrer par	Liste des chaînes	Liste de données
État	Masquer les alarmes non déclenchées	^((Not Triggered Acked|Not Triggered Not Acked|Triggered).*$)
Valeur	10 < Valeur < 20	^(1[0-9]$)
Valeur	20 <= Valeur <100	^([2-9].$)
Valeur	100 < Valeur < 200	^(1[0-9][0-9]$)
Valeur	Valeur 2?/3?/4?/5?	^([2-9].*$)
Valeur	Valeur >= 100	^([1-9][0-9][0-9].*$)
Valeur	Valeur >= 20	^([2-9].*$\|[1-9][0-9][0-9].*$)

Trier les alarmes

Chemin : ***Alarme active*** *widget>* ***Propriétés*** *volet>* ***Tri***

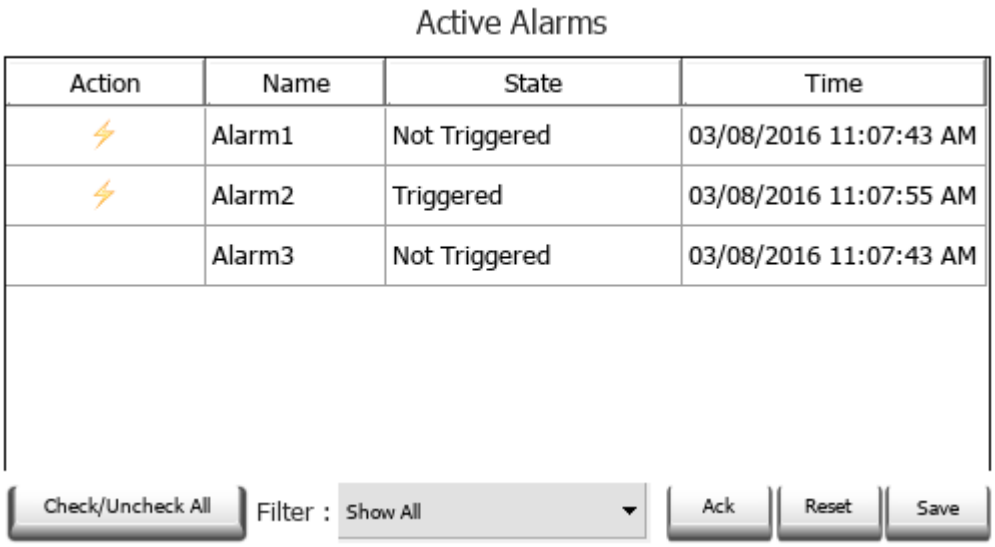
Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

La fonction de tri vous permet de trier les alarmes au runtime dans le widget des alarmes en cliquant sur l'en-tête de la colonne.

Remarque : la valeur de sévérité affichée ici est définie dans l'Éditeur d'alarme.

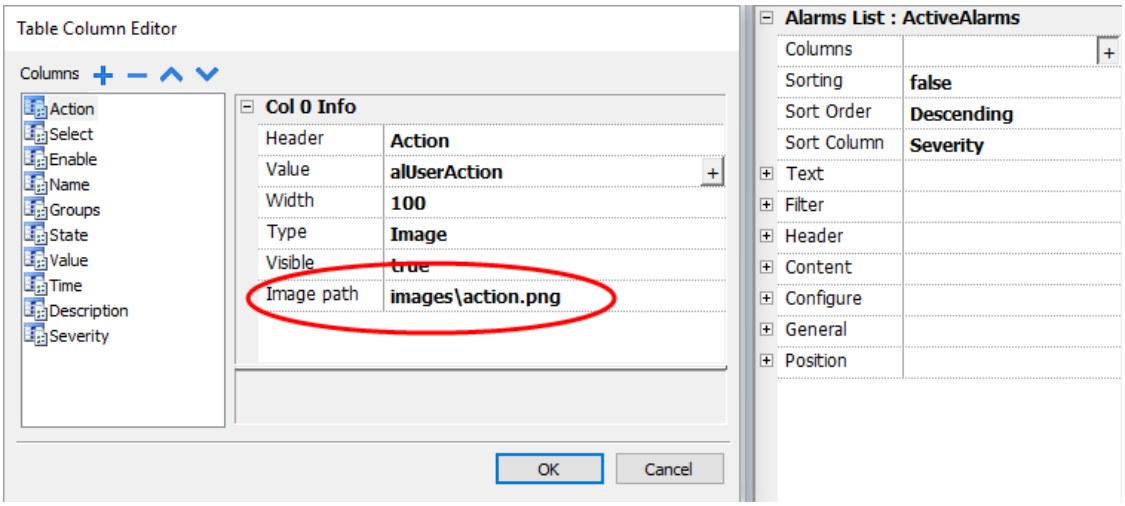
Action

Lorsque l'« Action utilisateur » associée avec l'alarme (voir "Éditeur d'alarmes" Page 200 pour plus de détails) contient des actions valides, l'icône Action s'affiche. En appuyant sur l'icône, les actions configurées seront exécutées.



AVERTISSEMENT : si vous utilisez un ancien projet converti, vous devez remplacer l'ancien widget Alarmes actives par le nouveau widget, situé dans la galerie des widgets

Remarque : l'image peut être modifiée dans les propriétés des colonnes du widget Alarmes actives

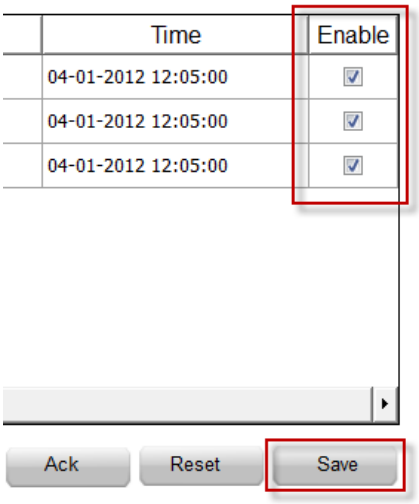


Activer/désactiver des alarmes

Au runtime, le widget Alarmes peut être utilisé pour activer ou désactiver les alarmes.

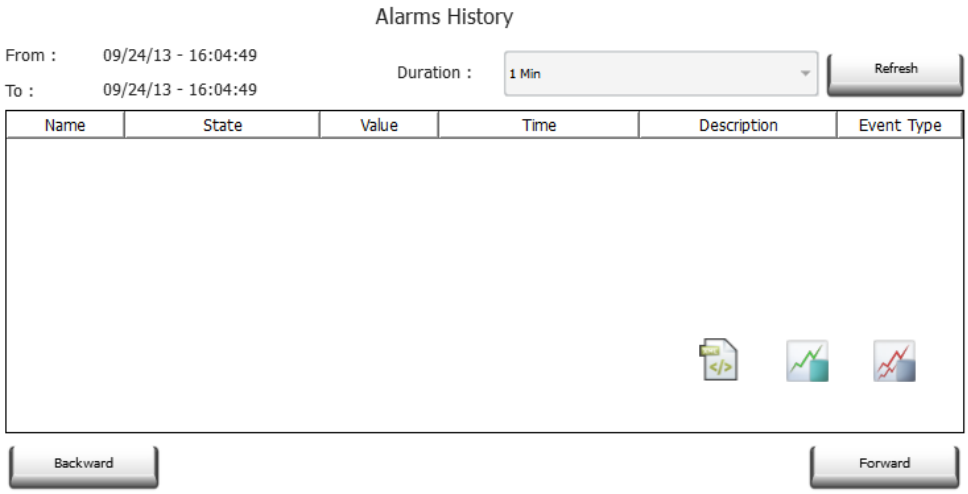
Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Sauvegarde les modifications dans la colonne **Activer** du widget Alarmes. Cette action s'effectue par le bouton **Sauvegarder** dans le widget Alarmes.



Widget historique des alarmes

Créé et affiche une liste d'alarme lorsque la propriété **Mémoire tampon** est définie dans l'éditeur de configuration des alarmes.



Associer un widget à la mémoire tampon

Chemin : widget **HistoriqueAlarme** > volet **Propriétés** > **Mémoire tampon** > **MémoireÉvénement**

Dans le volet **Propriétés** > **Événement**, sélectionnez la **Mémoire d'événements** qui contient la liste d'alarmes

Gestion des alarmes au runtime

Lorsqu'une alarme est déclenchée, elle s'affiche dans le widget Alarmes Actives, où vous pouvez la confirmer et la réinitialiser. Vous pouvez filtrer les alarmes affichées en utilisant une série de filtres, par exemple vous pouvez masquer les alarmes non déclenchées ou bien afficher toutes les alarmes.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Voir "Widget alarmes actives" Page 207 pour plus de détails.

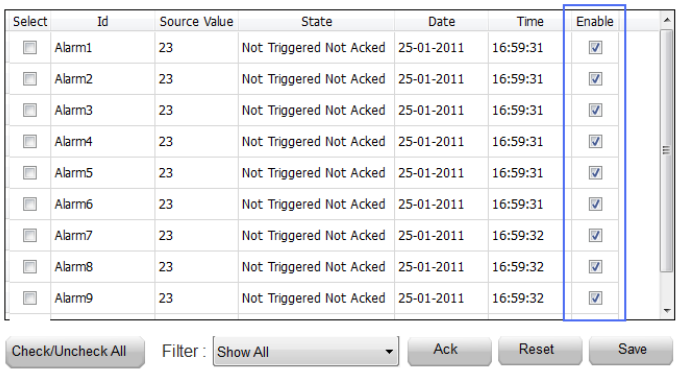
IMPORTANT : le widget alarmes actives ne s'affiche pas automatiquement. Vous devez ajouter une action dédiée qui ouvrira la page contenant le widget d'alarme dès que l'alarme se déclenche.

Activer/désactiver les alarmes au runtime

Vous pouvez activer ou désactiver les alarmes au runtime.

Pour activer une alarme, sélectionnez l'option **Activer** dans le widget de l'alarme.

Les alarmes désactivées ne sont pas déclenchées, elles ne sont donc pas affichées au runtime.



Remarque : vous pouvez configurer les alarmes pour qu'elles soient activées/désactivées même depuis l'API. Voir « Éditeur de configuration d'alarme » pour plus de détails.

Afficher les données d'alarme actives

*Chemin : **VueProjet**> **Configuration** > double-cliquez sur **Alarmes***

Vous pouvez paramétrer la description d'alarme de manière à afficher les données de variable active, aussi bien dans les Alarmes Actives que dans le widget Alarmes Historiques.

Id	Name	Enable	Ack	Reset	Tag	Buffer	Trigger	Action	Description
1	Alarm1	☑	☑	☑	Tag1	AlarmBuffer1	bitMaskAlarm:	ShowDialog	Alarm 1 Tag Value is [Tag1]
2	Alarm2	☑	☑	☑	Tag1	AlarmBuffer1	bitMaskAlarm:1	ShowDialog	Alarm 2 Tag Value is [Tag2]
3	Alarm3	☑	☑	☑	Tag1	AlarmBuffer1	bitMaskAlarm:1	ShowDialog	Alarm 3 Tag Value is [Tag3]

Pour afficher la valeur de variable, définir un caractère fictif dans **Description** et entrez le nom de variable entre crochet, par exemple « [variable1] ». Au runtime, la valeur actuelle de la variable apparaîtra dans la colonne **Description** du widget Alarmes actives.

Il y a deux types de caractères de remplissage

- [TagName]
  Lorsque l'alarme est déclenchée, la valeur de la variable est lue et continuellement mise à jour
- [!TagName]
  Lorsque l'alarme est déclenchée, la valeur de la variable est lue et gelée

Dans le widget Historique des alarmes ou dans le fichier.csv, les valeurs de la variable en temps réel sont les valeurs prises lorsque le statut de l'alarme change (pour les deux types de caractères de remplissage)

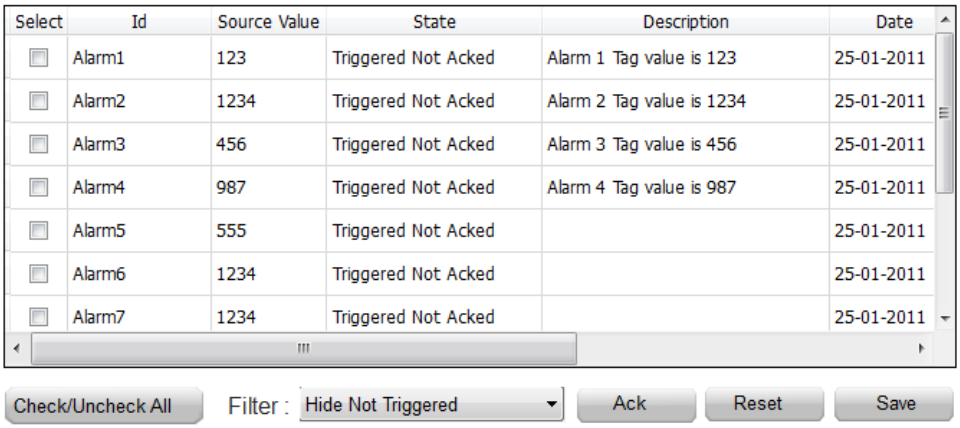
Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Utilisez '\' avant '[ ]' si vous souhaitez afficher le '[ ]' dans la chaîne de description, par exemple : *\[Variable\[1\]\]* affichera la chaîne *"[Variable[1]]"*.

Utiliser '\', même lorsque l'étiquette de variable contient des crochets. Par exemple, pour afficher la valeur de la variable en direct de la variable « TAG]3 » ou « TAG[3] », utiliser :

- TAG\]3 = **[**TAG\]3**]**
- TAG\[3\] = **[**TAG\[3\]**]**

Exemple d'un widget d'alarme



Le nombre de variables en temps réel pouvant être utilisées dans la description de chaque alarme dépend de la taille des variables utilisées. PB610 Panel Builder 600 vérifiera et affichera un message d'avertissement lorsque trop de variables sont utilisées.

La somme des octets calculée à l'aide de l'algorithme sous-jacent doit être inférieure ou égale à 50

```
    4 octets + (longueur de variable + 2) + (longueur de variable + 2) +
(longueur de variable + 2) + . . .   <= 50
```

Exemple :

Alarm Description:
Tag1=**[TagInt]**, Tag2=**[TagBool]**, Tag3=**[TagStr8]**

Fixed	4	
Tag1	6	4 (sizeof-INT) +2
Tag2	3	1 (sizeof-BOOL) +2
Tag3	10	8 (sizeof-STR8) +2
Total:	23	

Exporter la mémoire tampon des alarmes vers fichiers .csv

Pour exporter une mémoire d'événements contenant une liste de l'historique des alarmes, utilisez l'action **ExporterArchiveÉvénements**.

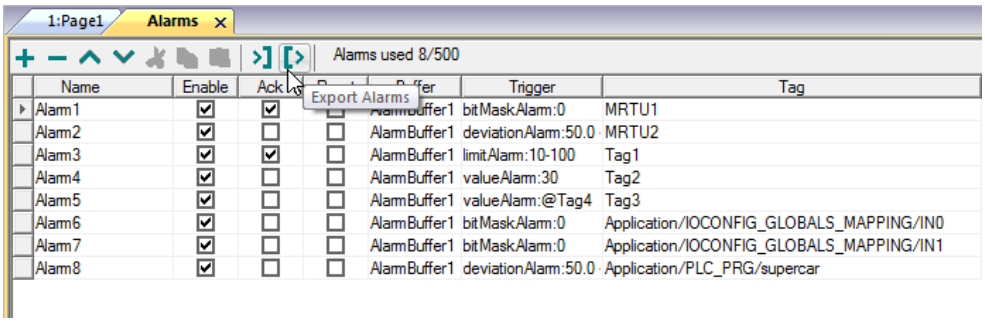
Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Voir "Actions système" Page 161 pour plus de détails.

Remarque : les valeurs de variable affichées dans la description des alarmes sont aussi incluses dans la mémoire tampon. Les variables sont échantillonnées lorsque l'alarme est déclenchée et lorsque la valeur est enregistrée et incluse dans la description.

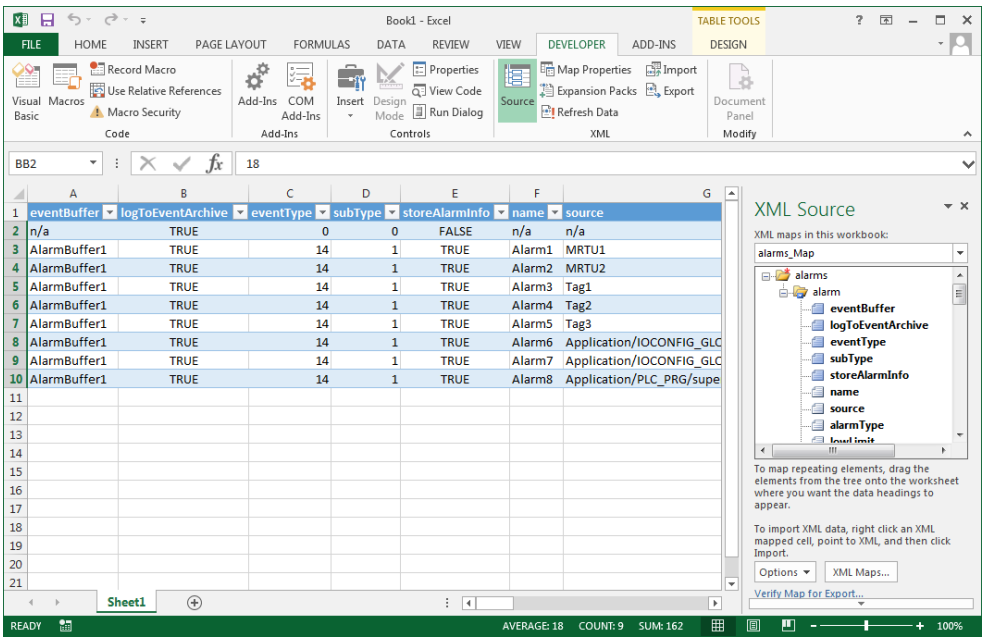
Exporter la configuration des alarmes

*Chemin : **VueProjet**> **Configuration** > double-cliquez sur **Alarmes***



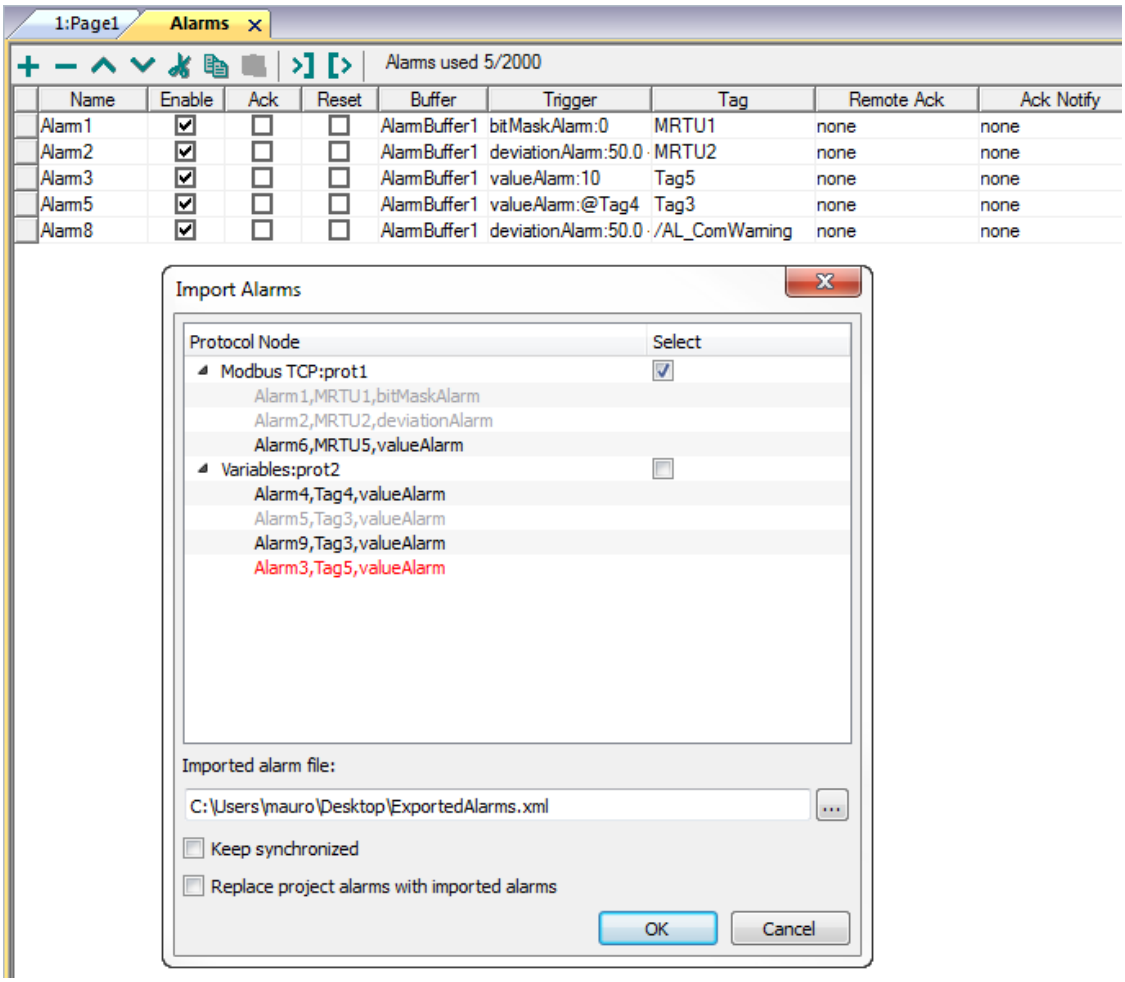
Cliquez sur le bouton **Exporter alarmes** : le tableau de configuration des alarmes est exporté dans un fichier .xml.

Vous pouvez éditer le fichier .xml qui en résulte avec des outils tiers (par exemple, Microsoft Excel).



Importer la configuration des alarmes

*Chemin : **VueProjet**> **Configuration** > double-cliquez sur **Alarmes***



1. Cliquez sur le bouton **Importer Alarmes** et sélectionnez le fichier .xml à partir duquel vous souhaitez importer la configuration des alarmes : la boîte de dialogue **Importer Alarmes** s'affiche.
2. Sélectionnez le groupe d'alarmes à importer et cliquez sur **OK** pour confirmer.

Les différences sont mises en évidence dans la boîte de dialogue **Importer Alarmes** à l'aide de couleurs différentes

Couleur	Description
Noir	C'est une nouvelle alarme et elle sera importée
Rouge	Cette alarme n'a pas été trouvée et sera supprimée (seulement si la case « Remplacer les alarmes de projet par les alarmes importées » est sélectionnée)
Bleu	Cette alarme a été modifiée et sera mise à jour.
Gris	Cette alarme fait déjà partie du projet et sera sautée.

Synchronisation automatique

Sélectionnez l'option **Synchroniser en temps réel** dans la boîte de dialogue **Importer Alarmes** pour activer la synchronisation automatique du fichier de configuration d'alarme.

À chaque modification de la configuration des alarmes, le fichier sera automatiquement mis à jour en mode silencieux.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Conseil : activez cette fonction lorsque le fichier d'alarme est géré par un autre outil (par exemple, logiciel de programmation API), en plus du PB610 Panel Builder 600.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

16 Recettes

Les recettes sont des collections de valeurs de variable organisées selon des ensembles répondant à des exigences spécifiques de l'application.

Par exemple, si vous devez contrôler des variables d'une pièce (température et humidité) le matin, l'après-midi et le soir. Vous créerez trois ensembles (matin, après-midi, soir) dans lesquels vous définirez les valeurs des variables appropriées.

Pour une utilisation plus efficace, chaque élément de la recette est associé à une variable et peut être indexé dans des ensembles. Cette fonctionnalité vous permet d'étendre les capacités des contrôleurs ayant une mémoire limitée.

Vous pouvez ajouter des données du contrôleur à une page en utilisant un widget recette. Les données de recette contiennent tous les éléments de données du contrôleur ; cependant, ces données ne peuvent plus être consultées directement à partir du contrôleur, mais à partir de l'élément de recette associée sur le périphérique IHM.

Les données de recette sont configurées dans l'espace de travail PB610 Panel Builder 600 ; l'utilisateur peut préciser les valeurs par défaut pour chaque élément des enregistrements de données. Dans HMI Runtime, les données peuvent être modifiées et sauvegardées dans un nouveau fichier de données, toutes les modifications des données de recette sont donc stockées sur disque. En utilisant un fichier de données séparé, HMI Runtime assure le maintien des valeurs de recette modifiées à travers les différentes mises à jour de projet. Autrement dit, une mise à jour de projet postérieure n'affecte pas les données de recette modifiées par l'utilisateur dans le HMI Runtime.

Voir "Actions sur une recette" Page 156pour plus de détails sur la réinitialisation des données de recette.

Remarque : les données de recette peuvent être stockées sur une mémoire flash, un lecteur USB ou une carte SD.

Gestion des recettes **219**

Configuration d'un widget de recette **222**

Statut de recette **223**

Téléverser/télécharger une recette **224**

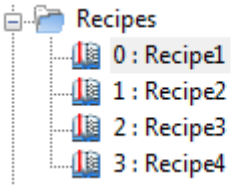
Sauvegarder et restaurer des données de recettes **225**

Gestion des recettes

Créer une recette

Pour créer une recette pour votre projet :

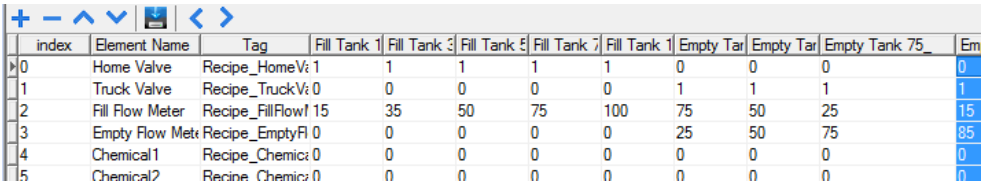
1. Dans **VueProjet**, cliquez à l'aide du bouton droit sur **Recettes** et sélectionnez **Insérer une recette** : une recette vide s'ajoute. Vous créez et configurez vos recettes à l'aide de l'Éditeur de recettes.



Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Éditeur de recettes

*Chemin : **VueProjet** > **Recettes** > double-cliquez sur **NomRecette***

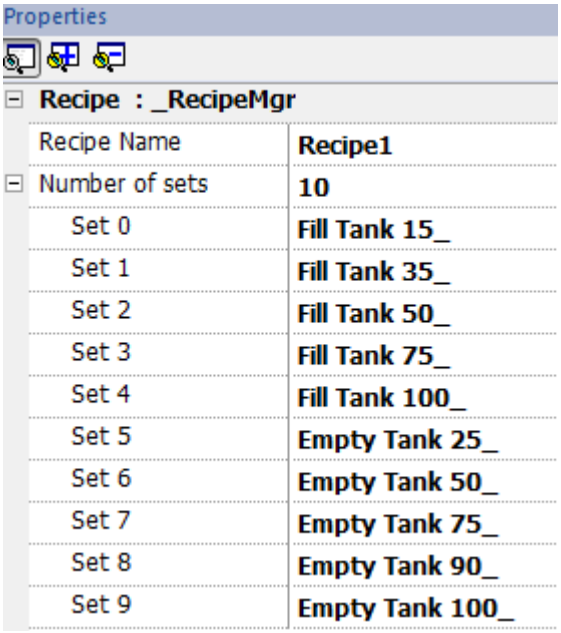


index	Element Name	Tag	Fill Tank 1	Fill Tank 3	Fill Tank 5	Fill Tank 7	Fill Tank 1	Empty Tar	Empty Tar	Empty Tank 75_	Em
0	Home Valve	Recipe_HomeV	1	1	1	1	1	0	0	0	0
1	Truck Valve	Recipe_TruckV	0	0	0	0	0	1	1	1	1
2	Fill Flow Meter	Recipe_FillFlowI	15	35	50	75	100	75	50	25	15
3	Empty Flow Met	Recipe_EmptyFl	0	0	0	0	0	25	50	75	85
4	Chemical1	Recipe_Chemica	0	0	0	0	0	0	0	0	0
5	Chemical2	Recipe_Chemica	0	0	0	0	0	0	0	0	0

Configurer les propriétés de recette

Dans le volet **Propriétés** de chaque recette, vous pouvez définir les paramètres suivants :

Paramètre	Description
Nom de recette	Nom de la recette
Nombre d'ensembles	Nombre d'ensembles de valeurs pour chaque élément de recette. Chaque ensemble a un nom différent qui peut être configuré.



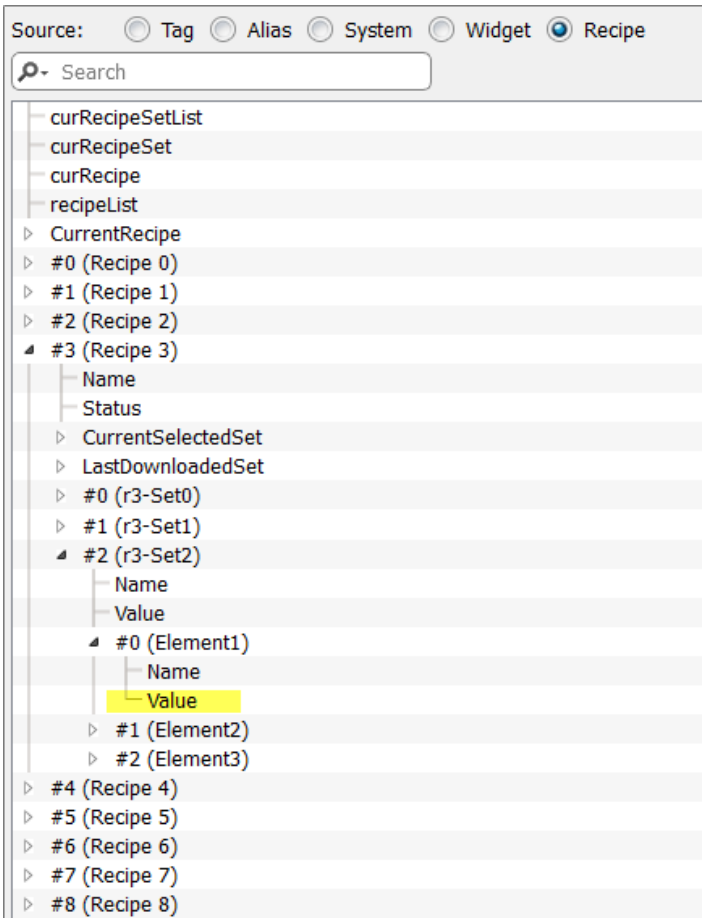
Paramétrer une recette

1. Cliquez sur **+** pour ajouter un élément à la recette.
2. Associez les variables à chaque élément de recette.

Définir des champs de recette

Créez un champ de recette dans la page en utilisant un widget numérique et en l'associant à un élément de recette après voir sélectionné la recette en tant que source.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



Dans la boîte de dialogue **Associer à**, vous avez le choix parmi toutes les différentes variables de recette, telles que :

- Recette actuelle >Ensemble de recette actuellement sélectionné> Élément > Valeur
- Recette sélectionnée > Ensemble0 sélectionné > Élément > Valeur
- liste de recettes

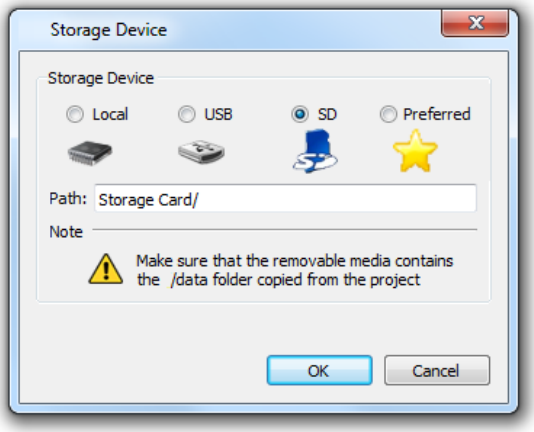
Quand les widgets numériques sont définies en lecture/écriture, les données de recette par défaut peuvent être modifiées au runtime. Ces nouvelles valeurs sont stockées sur un fichier séparé en tant que données de recette modifiées.

Remarque : étant donné qu'on utilise les fonctions de l'API JavaScript, les éléments de recette et les ensembles peuvent être référencés par nom ou par position. Afin d'éviter toute ambiguïté entre les noms et les index, les noms des éléments de recette et des ensembles doivent inclure au moins un caractère alphanumérique.

Stocker les données de recette

Dans l'Éditeur de recette, cliquez sur l'icône du type de stockage pour sélectionner l'endroit où vous souhaitez stocker les données de recette : la boîte de dialogue du **Périphérique de stockage** s'affiche.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



Pour les lecteurs USB et les cartes de stockage SD, vous pouvez préciser l'emplacement du dossier.

AVERTISSEMENT : les fichiers de configuration des recettes sont automatiquement créés au moment de la sauvegarde du projet et sont stockés dans le sous-dossier de données du projet. Pour utiliser les périphériques de stockage externes, vous devez copier ce dossier dans le périphérique externe. Notez que vous avez la responsabilité de gérer le dossier de données à l'intérieur des périphériques externes. Même les fichiers dynamiques ne sont pas supprimés lorsque le projet est mis à jour en utilisant l'option « Supprimer le fichier dynamique ».

Important : vous pouvez ajouter un sous-dossier mais vous ne devez pas renommer le sous-dossier « données ».

Importer/exporter des recettes

Pour importer/exporter la configuration des recettes de votre projet :

Dans **VueProjet**, cliquez droit sur **Recettes**et sélectionnez **Exporter une recette** ou **Importer une recette**

Les formats suivants sont pris en charge pour l'importation :

- Valeurs séparées par des virgules (.csv)
- Texte Unicode (.txt)

Remarque : utilisez le format de fichier en Texte Unicode lorsque vous importez un fichier modifié avec Microsoft® Excel®.

Configuration d'un widget de recette

Vous pouvez choisir un des deux widgets de recettes disponibles dans la **Galerie des widgets** :

- **Ensemble de recettes** : vous permet de sélectionner un ensemble de recettes pour le charger ou le télécharger. Voir "Téléverser/télécharger une recette" Page 224.
- **Menu de recettes** : lorsque plusieurs recettes ont été créées pour un projet, utilisez ce widget pour gérer toutes les recettes et sélectionnez les ensembles voulus pour chacune d'entre elles.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Configurer le widget d'ensemble de recettes

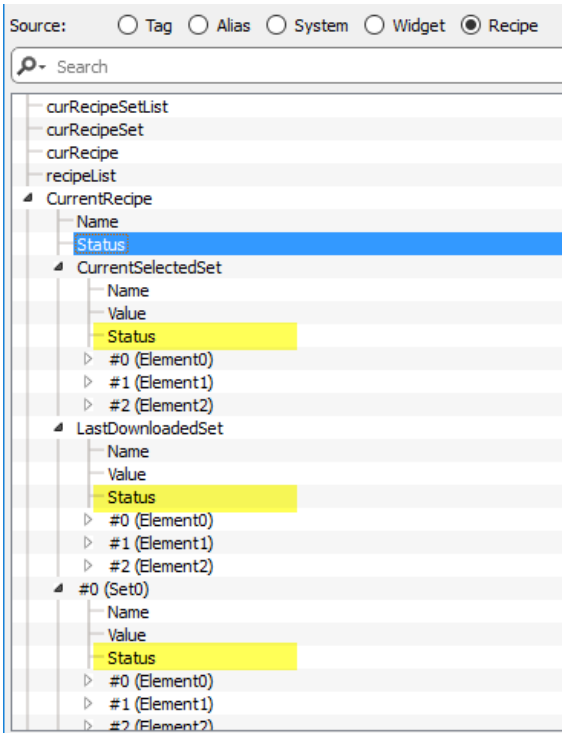
Dans le volet **Propriétés** de chaque widget **Ensemble de recettes**, définissez les paramètres suivants :

Paramètre	Description
Nom de recette	Nom de la recette

Statut de recette

Chaque recette contient deux types de paramètres de statut

- Statut de la recette (bleu dans l'image ci-dessous)
  Donne des informations sur la dernière opération de téléchargement ou de téléversement
- Statut de l'ensemble de données (jaune dans l'image ci-dessous)
  Donne des informations sur les ensembles de données modifiées



Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

État de la recette

Après chaque chargement ou téléchargement de recette, ou modification d'un ensemble de recette, les paramètres **Statut de recette** contiennent une valeur avec le résultat de l'opération.

Code	Fonction	Description
0	Ensemble modifié	Ensemble sélectionné modifié
1	Téléchargement déclenché	Demande de téléchargement déclenchée
2	Téléchargement effectué	Opération de téléchargement terminée
3	Erreur de téléchargement	Erreur pendant le téléchargement (par exemple, ensemble inconnu, recette inconnue, contrôleur pas prêt, échec d'écriture des variables etc.)
4	Chargement déclenché	Demande de chargement déclenchée
5	Chargement effectué	Opération de chargement terminée
6	Erreur de chargement	Erreur pendant le chargement - pareil que pour le téléchargement
7	Erreur générale	Erreur générale (par exemple, données indisponibles)

État d'ensemble de données

Le statut de chaque ensemble de données indique qu'il a été modifié. Ces informations peuvent être utiles pour ne pas oublier de télécharger la recette pour synchroniser l'API. Les deux opérations de téléchargement ou de chargement réinitialiseront le statut de l'ensemble de données à 0.

Code	Fonction	Description
0	Synchronisé	API synchronisé par l'utilisateur avec les valeurs de l'ensemble de données
1	Modifié	L'utilisateur a modifié certaines valeurs de l'ensemble de données

Remarque : Après un démarrage du périphérique ou une réinitialisation/restauration d'une recette, toutes les valeurs de statut sont remises à 0.

Téléverser/télécharger une recette

Téléverser une recette

Vous téléversez une recette vers un périphérique IHM en utilisant un widget de recette et l'action **Téléverser Recette**, **TéléverserRecetteEncours** de l'une des façons suivantes :

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

- joindre l'action à un événement d'un bouton ou d'un interrupteur (voir "Paramètres « Associer à »" Page 44 pour les détails)
- configurer l'action dans une liste d'action d'alarme (voir "Actions d'alarmes" Page 140 pour les détails)
- configurer l'action dans une liste d'action de planificateur (voir "Ordonnancement des événements au runtime" Page 270 pour les détails)

Télécharger une Recette

Vous téléchargez une recette depuis un périphérique IHM en utilisant un widget de recette et l'action **Télécharger Recette**, **TéléchargerRecetteEncours**. Voir "Actions sur une recette" Page 156

Sauvegarder et restaurer des données de recettes

Les données de recette stockées dans un périphérique IHM peuvent être exportées pour être sauvegardées et restaurées par la suite. Pour ce faire, il faut utiliser les actions **ViderDonnéesRecette** ou **RestaurerDonnéesRecette**.

Voir "Actions sur une recette" Page 156 pour plus de détails.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

17 Tendances

Les tendances vous permettent d'échantillonner et d'enregistrer les valeurs des variables spécifiées en fonction des conditions spécifiques d'échantillonnage. La fonction de tendance comprend l'acquisition de tendance et l'affichage de tendance.

Les paramètres d'acquisition de tendance sont définies dans l'éditeur de tendance afin que les données puissent être stockées. Les données stockées peuvent ensuite être affichées dans un format graphique en utilisant un widget tendance.

Enregistrement des données **228**

Exporter les données de mémoire de tendance **231**

Widgets tendance **232**

Tendances historiques **233**

Propriétés du widget tendance **235**

Gestes de widget de tendance **236**

Valeurs hors de portée ou non valides **237**

Afficher les valeurs de tendance **238**

Widget diagramme de dispersion **239**

Widget tendance de tableau **240**

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Enregistrement des données

Les données peuvent être enregistrées et stockées dans la mémoire IHM. L'enregistrement des données vous permet de stocker les valeurs d'un groupe de variables au même moment dans une mémoire tampon. L'enregistrement des données peut être déclenché par un minuteur ou par une variable dédiée. Les données enregistrées peuvent être exportées dans un fichier .csv ou affichées en utilisant le widget de tendance historique. Les données enregistrées peuvent être sauvegardées localement sur un périphérique USB ou sur une carte SD, ou bien sur tout dossier de réseau personnalisé et disponible.

AVERTISSEMENT : l'opération avec des périphériques mémoire amovibles (lecteurs flash USB, cartes mémoires SD) contenant un très grand nombre de fichiers peut entraîner une perte de performance du système.

AVERTISSEMENT : le nombre maximal de fichiers dans une carte mémoire SD dépend du type de formatage (par ex. FAT32 max. 65536 fichiers, FAT max. 513 fichiers).

AVERTISSEMENT : les cartes flash prennent en charge un nombre limité d'opérations d'écriture. Nous vous suggérons de n'utiliser que des cartes mémoires de bonne qualité. Lorsque votre application utilise la carte mémoire de manière intensive, pensez à changer régulièrement de carte mémoire.

AVERTISSEMENT : si la date/heure est reculée, les échantillons avec des dates/heures non valides sont supprimés de la mémoire des tendances. Lorsque le système détecte que la date/l'heure n'est pas valide (ex. batterie faible), un avertissement s'affiche pour informer l'utilisateur que la date/l'heure du dernier échantillon est utilisée pour éviter de perdre des données.

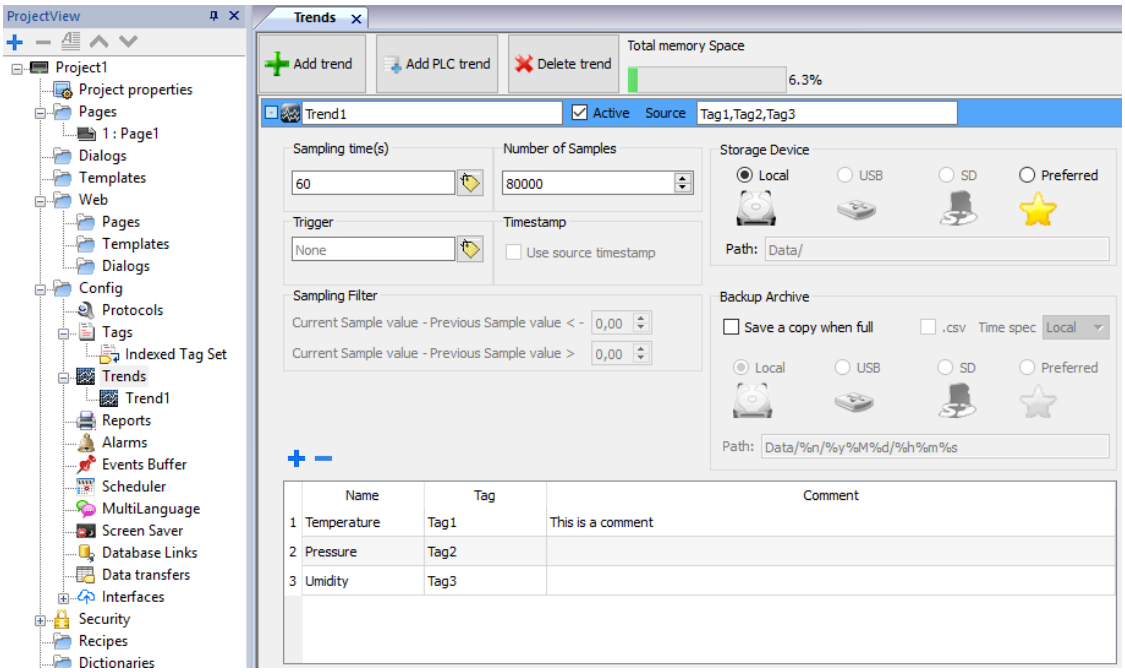
Le stockage est basé sur les mémoires des tendances. Les mémoires des tendances sont organisées comme une queue FIFO : lorsque la mémoire tampon est pleine, les valeurs les plus anciennes sont rejetées sauf si vous configurez votre tendance pour créer une copie de sauvegarde de la mémoire tampon.

Ajouter une mémoire des tendances

*Chemin : **VueProjet**> **Configuration** > double-cliquez sur **Tendances***

1. Cliquez sur **Ajouter** pour ajouter une nouvelle mémoire tampon.
2. Cliquez sur **+** à côté de chaque mémoire des tendances pour afficher tous les paramètres de configuration.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



Élément	Description
Espace mémoire total	Mémoire en cours d'utilisation par la mémoire des tendances. Voir "Tableau de fonctions et limites" Page 540 pour le nombre maximal d'échantillons autorisé pour un projet. Ce pourcentage est calculé de la manière suivante : $Total\ Memory\ Space\ = \frac{Total\ Number\ of\ Samples\ used\ in\ the\ Project}{Max\ Number\ of\ Samples\ allowed\ for\ a\ Project} * 100$
Nom de tendance	Nom de tendance qui sera affiché dans le volet des propriétés de la fenêtre.
Actif	Lorsque cet élément est activé, la tendance s'exécute par défaut au démarrage du système. Remarque : les tendances ne peuvent pas être activées au runtime.
Source	Variables échantillonnées par la tendance.
Temps d'échantillonnage	Intervalle d'échantillonnage en secondes.
Déclencheur	Variable déclenchant l'échantillon. Lorsque la valeur de cette variable change, un échantillon est collecté. Remarque : Déclencheur et Source peuvent faire référence à la même variable.
Horodatage	Une fois coché, utilisez l'horodatage fourni par le périphérique distant. Disponible :

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Élément	Description
	• uniquement pour les protocoles du périphérique qui prennent en charge cette fonctionnalité (OPC UA Client et BACnet) • lorsque la mémoire tampon de tendance est configurée pour contenir une seule variable.
Nombre d'échantillons	Taille de la mémoire tampon.
Périphérique de stockage	Périphérique où les données de mémoire des tendances sont stockées.
Sauvegarder archive	Lorsque l'option **Sauvegarder une copie lorsque la mémoire est saturée** est activée, une copie de sauvegarde des données de mémoire tampon est créée avant qu'elle ne soit écrasée par de nouvelles données. **.csv** : données sauvegardées utilisant le format texte CSV. **Spéc heure** : vider l'heure locale ou UTC au format CSV **Chemin** : périphérique où les données de mémoire des tendances sont copiées Les caractères de remplacement ci-dessous sont pris en charge • %n = nom de tendance • %y = année • %M = mois • %d = jour • %h = heure • %m = minutes • %s = secondes
Filtre d'échantillonnage/Filtre déclencheur	Ce paramètre permet de spécifier un filtre dynamique si nécessaire. Lorsque l'échantillonnage est effectué sur une base temporelle, le décalage est appliqué à la valeur de la variable échantillonnée. Si la nouvelle valeur dépasse les limites spécifiées, la nouvelle valeur est considérée comme valide et stockée, sinon le nouvel enregistrement conservera la valeur sauvegardée précédente. Lorsque l'échantillonnage est effectué sur le déclencheur, le décalage est appliqué à la valeur de variable du déclencheur. Si le changement de la valeur de variable du déclencheur dépasse les limites spécifiées, un nouvel échantillon est prélevé et stocké, sinon aucun échantillonnage ne sera effectué.
Tableau de variables échantillonné	**Nom** : nom de la tendance **Variable** : variable à échantillonner. **Commentaire** : description de la tendance

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Exporter les données de mémoire de tendance

Utilisez l'action **ExporterTendance** pour exporter des données de mémoire de tendance vers un fichier .csv. Voir "DumpTrend" Page 161 pour les détails des paramètres de la macro.

Le fichier.csv exporté peut avoir différents formats définis à partir des paramètres de la macro Vidage de tendance. Les différents formats sont maintenus principalement pour des raisons de compatibilité.

Format de fichier : **Compatibilité CSV**

	A	B	C	D	E	F	G	H	I	J	K
1	Type	Value	Time Stamp	Refresh Time	Quality	Type	Value	Quality	Type	Value	Quality
2	4	0	2015-09-18T14:42:22.000Z	1000	192	8	0.00E+00	192	3	0	192
3	4	0	2015-09-18T14:42:23.000Z	1000	192	8	0.00E+00	192	3	0	192
4	4	0	2015-09-18T14:42:24.000Z	1000	192	8	0.00E+00	192	3	0	192
5	4	40	2015-09-18T14:42:25.000Z	1000	192	8	0.00E+00	192	3	0	192
6	4	40	2015-09-18T14:42:26.000Z	1000	192	8	0.00E+00	192	3	0	192
7	4	40	2015-09-18T14:42:27.000Z	1000	192	8	0.00E+00	192	3	0	192
8	4	40	2015-09-18T14:42:28.000Z	1000	192	8	5.00E+01	192	3	0	192
9	4	40	2015-09-18T14:42:29.000Z	1000	192	8	5.00E+01	192	3	0	192
10	4	40	2015-09-18T14:42:30.000Z	1000	192	8	5.00E+01	192	3	0	192

Format de fichier : **CSV compact**

	A	B	C	D	E	F	G
1	Timestamp	Tag1	4	Tag2	8	Tag3	3
2		Value	Quality	Value	Quality	Value	Quality
3	2015-09-18T14:42:22.000Z	0	192	0.00E+00	192	0	192
4	2015-09-18T14:42:23.000Z	0	192	0.00E+00	192	0	192
5	2015-09-18T14:42:24.000Z	0	192	0.00E+00	192	0	192
6	2015-09-18T14:42:25.000Z	40	192	0.00E+00	192	0	192
7	2015-09-18T14:42:26.000Z	40	192	0.00E+00	192	0	192
8	2015-09-18T14:42:27.000Z	40	192	0.00E+00	192	0	192
9	2015-09-18T14:42:28.000Z	40	192	5.00E+01	192	0	192
10	2015-09-18T14:42:29.000Z	40	192	5.00E+01	192	0	192

REmarque : la première ligne de l'en-tête contient les noms de variables et les types de données de variables

Format de fichier : **CSV compact** avec sélection de colonnes

	A	B	C	D	E	F	G	H
1	Date	Time	Name1(int)	Quality	Name2(int)	Quality	Name3(boolean)	Quality
2	17/04/2018	07:24:29	0	192	10	192	0	192
3	17/04/2018	07:24:30	1	192	11	192	1	192
4	17/04/2018	07:24:31	2	192	12	192	0	192
5	17/04/2018	07:24:32	3	192	13	192	1	192
6	17/04/2018	07:24:33	4	192	14	192	0	192
7	17/04/2018	07:24:34	5	192	15	192	1	192
8	17/04/2018	07:24:35	6	192	16	192	0	192
9	17/04/2018	07:24:36	7	192	17	192	1	192
10	17/04/2018	07:24:37	8	192	18	192	0	192

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Widgets tendance

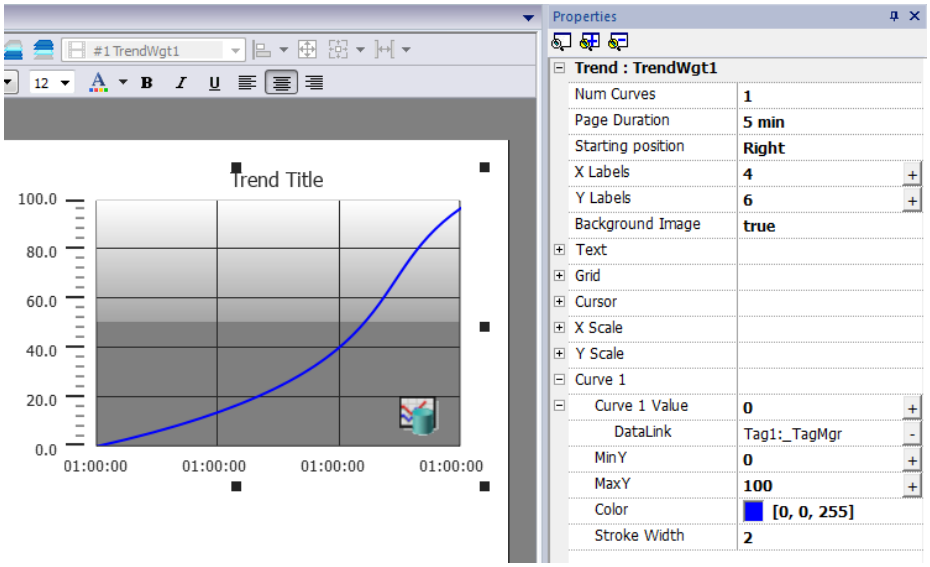
Il est possible d'afficher les données enregistrées par le périphérique IHM sous forme de graphiques en utilisant les widgets de tendance.

Widget tendance en temps réel

Le widget tendance en temps réel peut être utilisé pour afficher les modifications de valeur d'une variable. Les données ne sont pas stockées dans une mémoire des tendances et ne peuvent pas être récupérées pour analyse ultérieure.

Pour afficher une tendance en temps réel :

1. Glissez-déposez le **Widget de tendance en temps réel** de la galerie des widgets jusqu'à la page.



2. Associez la variable que vous voulez échantillonnez à la **valeur de la Courbe *n***. Les données doivent toujours être représentées en fonction du temps.

Propriétés du widget tendance en temps réel

Propriété	Description
Nombre de courbes	Nombre de courbes de tendance à afficher (Max. 5)
Durée de page	Intervalle de temps de l'axe x. Conseil : vous pouvez associer un widget **Date Heure** combiné à la propriété de **Durée de Page** et modifier de façon dynamique la durée de la page au runtime.
Position initiale	Spécifie le point de départ de la courbe lorsque la page est ouverte.
Labels X	Nombre de graduations sur l'échelle de l'axe x

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Propriété	Description
Labels Y	Nombre de graduations sur l'échelle de l'axe y.
Texte	Titre de tendance et propriétés de police (taille de police, label, etc.)
Grille	Propriétés de présentation des grilles (couleurs)
Curseur	Propriétés de présentation du curseur (activation et couleur)
Échelle X	Propriétés de présentation de l'échelle X
Échelle Y	Propriétés de présentation de l'échelle Y
Courbe « n »	Variable qui sera représentée dans le widget tendance. Voir "Propriétés du widget tendance" Page 235 pour les détails. Vous pouvez définir le minimum et le maximum des courbes (**MinY**, **MaxY**). Vous pouvez associer une variable aux propriétés minimum et maximum. Cela permet de modifier plus aisément les valeurs minimum et maximum de façon dynamique au runtime.

Mise à l'échelle des données

Les valeurs de variable peuvent être mises à l'échelle en utilisant les Formes X dans la boîte de dialogue **Associer à**. Voir "Paramètres « Associer à »" Page 44 pour les détails.

Tendances historiques

Les données de tendance stockées dans les mémoire des tendances peuvent être analysées à l'aide du widget de **Tendance historique**.

Il s'agit d'un processus en deux étapes :

- tout d'abord, vous créez une mémoire des tendances qui collectera les données de variables spécifiques à des moments spécifiques,
- puis vous configurez un widget de Tendance historique de manière à ce qu'il affiche les données collectées sous forme de graphique.

Voir "Enregistrement des données" Page 228 pour plus détails sur la façon de créer une mémoire des tendances.

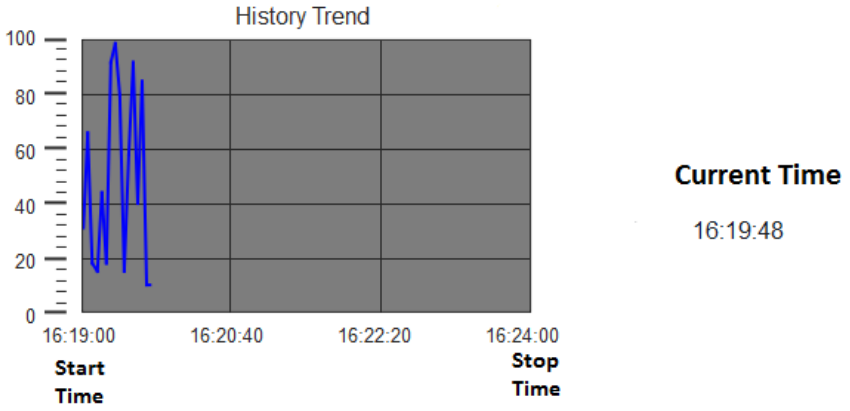
Widget de Tendance historique

Le widget de Tendance historique affiche le contenu de la mémoire des tendances sous forme de graphique.

L'heure de début correspond à l'heure actuelle et l'heure de fin correspond à l'heure actuelle plus la durée de la fenêtre. La courbe part de la gauche et progresse vers la droite, les données sont automatiquement rafraîchies pendant un certain intervalle de temps, jusqu'à l'heure de fin.

Lorsqu'elle atteindra l'heure de fin, la courbe défilera vers la gauche et sa mise à jour continuera jusqu'à ce qu'elle atteigne à nouveau l'heure de fin. À ce moment, un nouveau défilement est automatiquement exécuté et le processus se répète.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

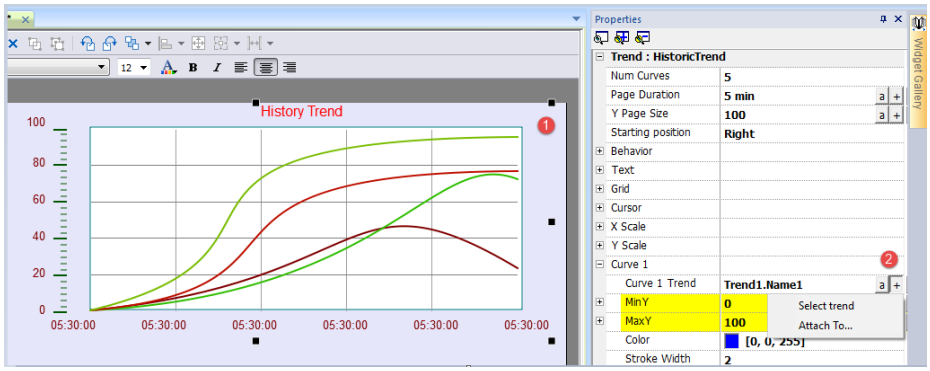


Les tendances historiques demandent une configuration appropriée de la mémoire des tendances.

Voir "Enregistrement des données" Page 228 pour plus détails sur l'utilisation de l'Éditeur de tendances.

Configurer le widget de Tendance historique

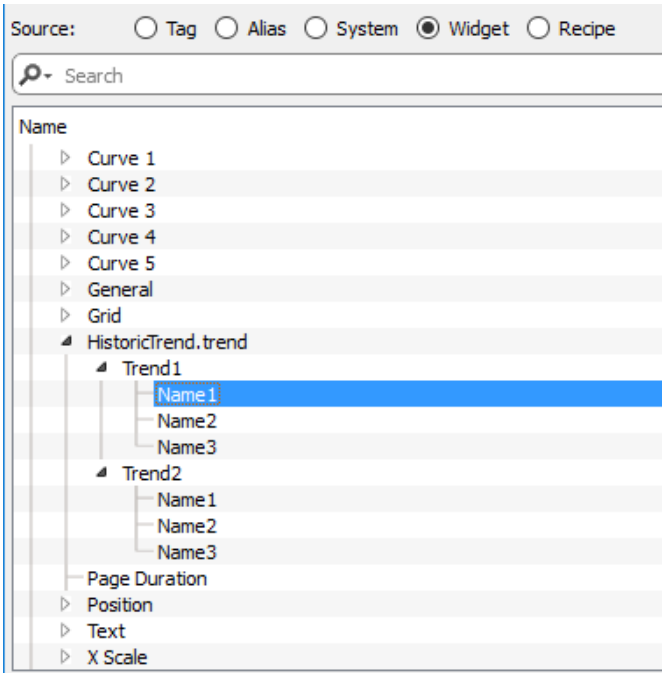
1. À partir de la section **Tendances/Graphiques** de la **Galerie des widgets,** glissez-déposez le widget de **Tendance historique** dans la page.



2. Dans le volet **Propriétés,** associez la mémoire des tendances à tracer dans le widget.

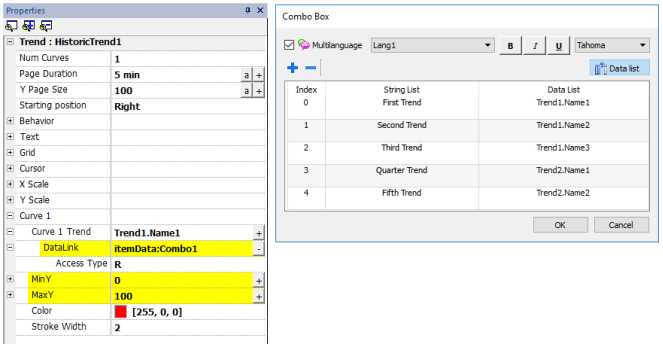
   Vous pouvez **Sélectionner une tendance** directement par le widget Tendance historique

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



ou utiliser la variable **Associer à...** pour identifier la variable qui contiendra le nom de la tendance à afficher. En utilisant la variable Associer à, la courbe à afficher peut être sélectionnée à partir de HMI Runtime

Par exemple, connectez-vous à la propriété Données actuelles d'un Widget liste déroulante configuré avec la liste des tendances à afficher.



Propriétés du widget tendance

Certaines propriétés de tendance du widget ne sont disponibles que lorsque le volet Propriétés est en mode Avancé.

Demande d'échantillons

La propriété de **demande d'échantillons** peut être réglée pour chaque courbe et indique le nombre maximum d'échantillons lus par le widget à un moment donné de la mémoire des tendances.

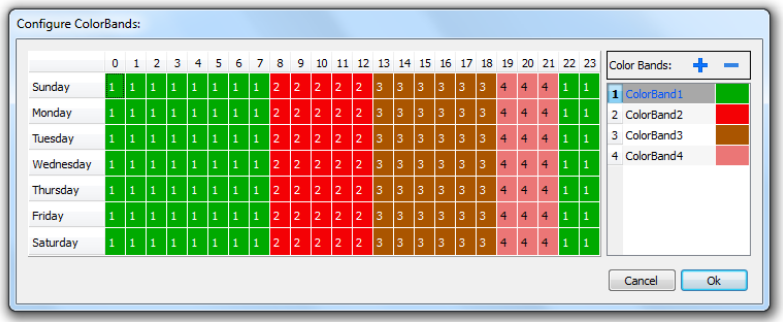
Conseil : vous n'avez normalement pas besoin de modifier la valeur par défaut. Ajustez pour affiner les performances accordées dans le rafraîchissement du widget tendance, en particulier quand on travaille avec des clients distants.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Bandes de couleur

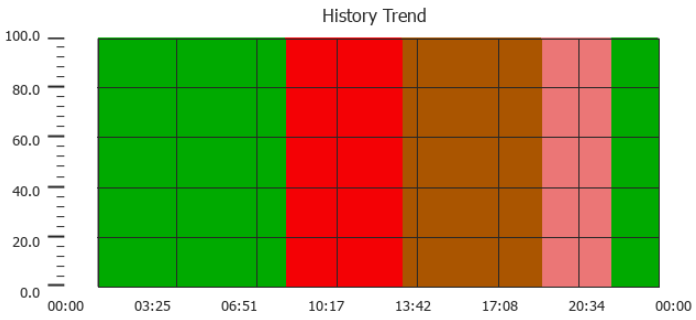
Utiliser la configuration des bandes de couleur pour personnaliser votre arrière-plan de graphique, par exemple pour mettre en évidence certains jours ou certaines heures (week-ends, heures de nuit, etc.).

1. Dans le volet **Propriétés**, dans la propriété **Bandes de couleur,** cliquez sur + : la fenêtre **Configurer bandes** apparaît.
2. Cliquez sur **+** pour ajouter le nombre de couches souhaitées.
3. Sélectionnez plusieurs cellules et cliquez sur une bande de couleur pour affecter la couleur à la plage sélectionnée des cellules.



Cette fonctionnalité utilise uniquement l'heure locale dans le widget tendance, non l'option de temps international.

Exemple de bandes de couleurs dans le calendrier



Gestes de widget de tendance

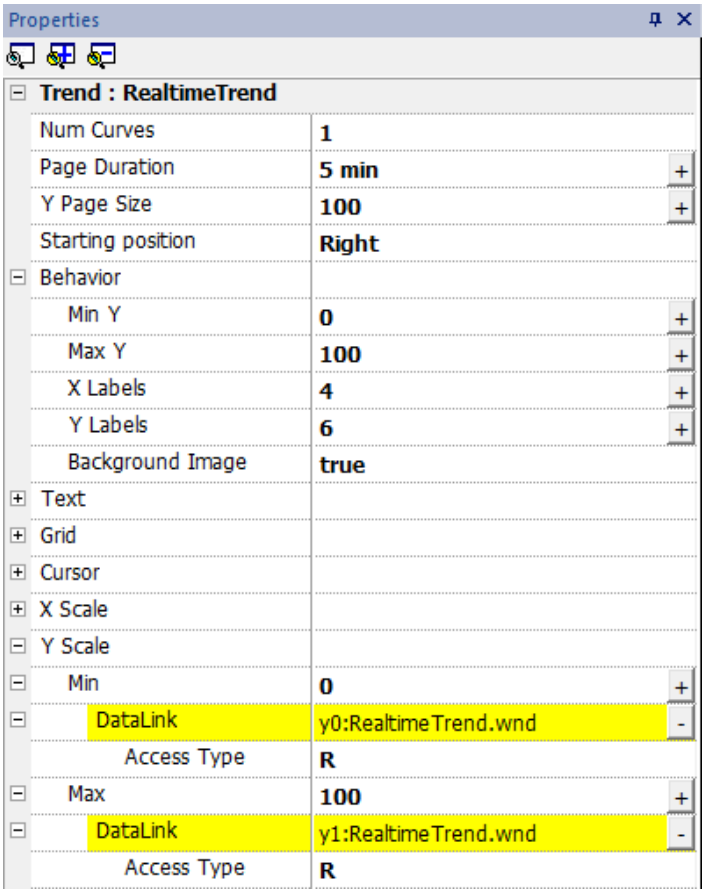
Les widgets de tendance supportent les commandes gestuelles :

Geste	Description
faire défiler	Touchez le widget pour faire défiler la courbe dans la zone de widget
pincer	Utilisez deux doigts pour pincer la courbe et effectuer des opérations de zoom

AVERTISSEMENT : seuls les dispositifs IHM tactiles multipoints peuvent générer des événements de pincement

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

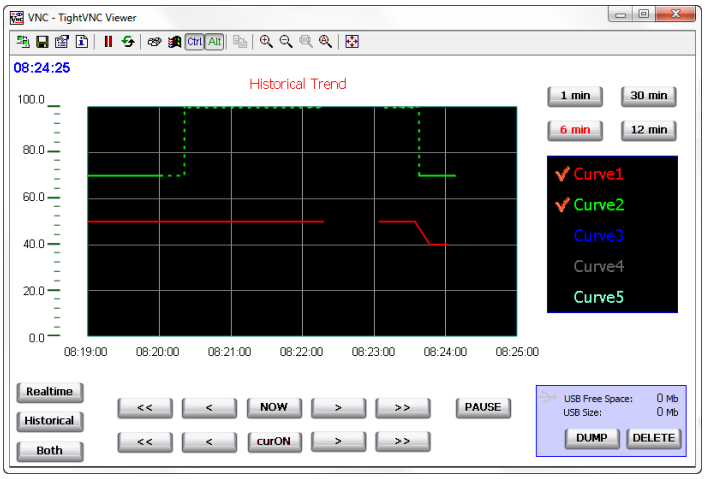
Remarque : afin de prendre ne charge les gestes sur l'axe Y, les propriétés min/max du widget de tendance doivent être liées aux valeurs Min/Max des paramètres de comportement (par défaut pour les nouvelles tendances).



Valeurs hors de portée ou non valides

Lorsque la valeur de tendance dépasse les limites définies pour le widget de tendance, une ligne pointillée apparaît. Lorsque la valeur de la variable n'est pas disponible, par exemple lorsque le périphérique du contrôleur est hors ligne, aucune courbe n'est tracée.

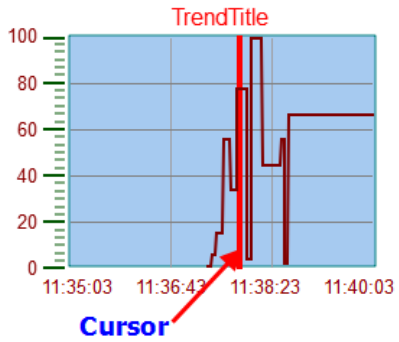
Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



Afficher les valeurs de tendance

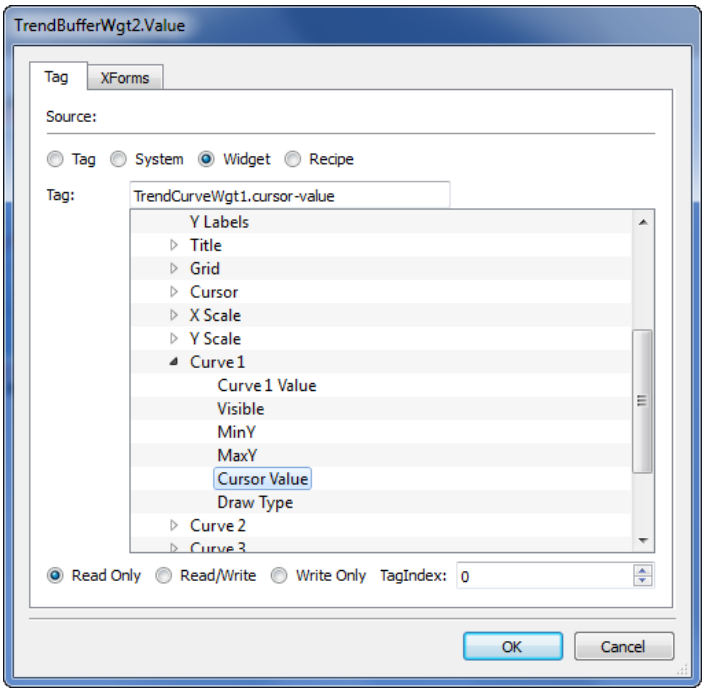
Le curseur de tendance affiche la valeur de la tendance à un moment précis.

Utilisez les actions **AfficherCurseurTendance** et **DéfilerCurseurTendance** pour activer le curseur de la tendance et le déplacer vers le point nécessaire pour obtenir la valeur de la courbe à ce moment précis dans le temps.



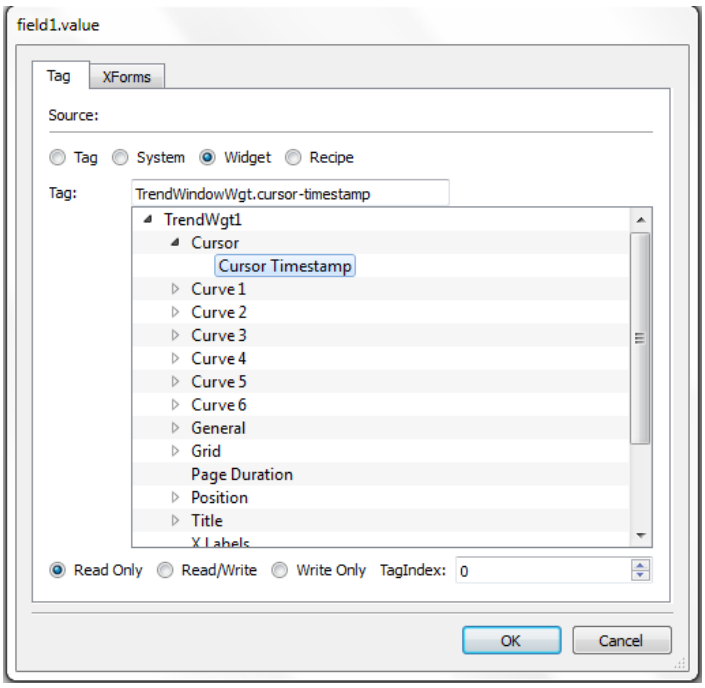
Pour afficher la valeur du curseur de tendance sur la page, définir un champ numérique et l'attacher à la variable du widget **Valeur du Curseur**.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



Dans cet exemple, la valeur de l'axe Y du curseur est affichée.

Pour afficher l'horodatage de tendance sur la position du curseur, définissez un champ numérique et l'attacher à la variable du widget **Horodatage de curseur**.

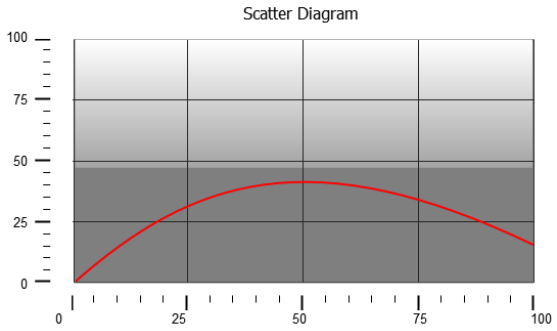


Widget diagramme de dispersion

Un diagramme de dispersion est un type de diagramme pour afficher les valeurs de deux variables à partir d'un ensemble de données en coordonnées cartésiennes. Les données sont affichées comme un ensemble de points, chacune ayant la

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

valeur d'une variable déterminant la position sur l'axe horizontal et la valeur de l'autre variable déterminant la position sur l'axe vertical. C'est la raison pour laquelle il est souvent appelé *graphique XY*.



Les courbes de diagramme de dispersion sont obtenues par une interpolation linéaire de points. Pour créer un nouveau diagramme de dispersion :

1. Ajoutez un widget **Diagramme de dispersion** à la page.
2. Sélectionnez le nombre de courbes à afficher : chaque courbe est nommée comme Graph1, Graph2, etc.
3. Personnalisez les propriétés des graphes généraux tels que les détails **X Min, X Max, de la grille**.
4. Définir le nombre maximum d'échantillons/valeurs pour chaque courbe en définissant le paramètre **Échantillon max**.

   Vous définissez ici le nombre maximum de valeurs à afficher dans le graphique à partir du premier élément du tableau.

   Par exemple : tag1[20] et échantillons Max = 10 montreront seulement les 10 premiers éléments du tableau Tag1.
5. Définir pour chaque courbe les deux variables de tableau type à afficher (**variable X** et **variable Y**).

Lorsque les variables de tableau changent, vous pouvez forcer un rafraîchissement à l'action de **RafraîchirTendance**.

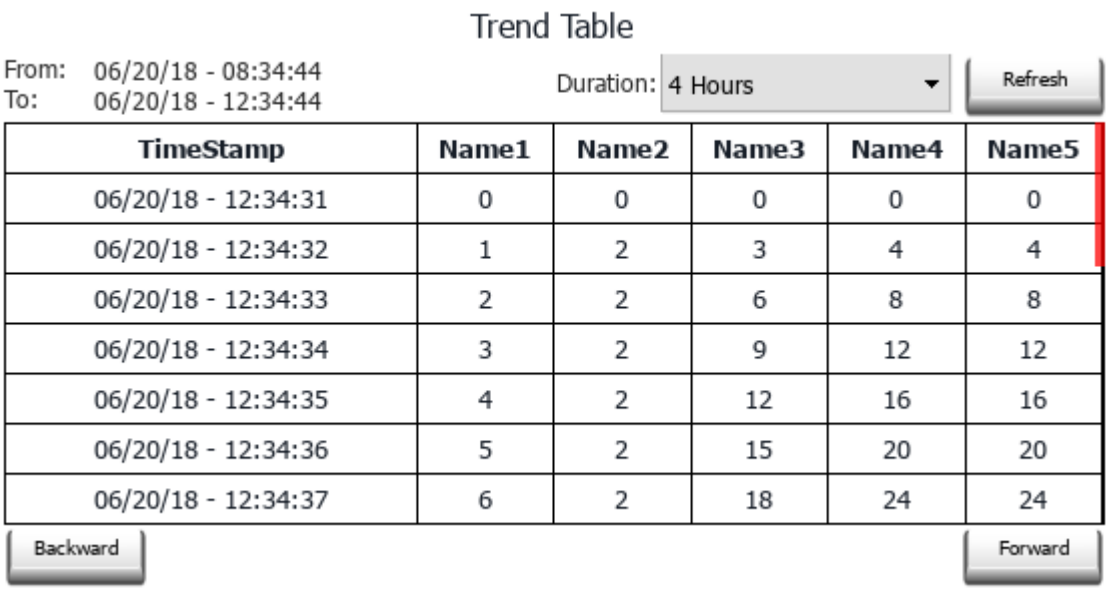
Remarque : les diagrammes de dispersion ne supportent que l'action **RafraîchirTendance**.

Widget tendance de tableau

Chemin : **Galerie des widgets**> **Basique**> **Tendances/Graphiques**

Affiche le contenu d'une mémoire tampon de tendance dans un widget

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



TimeStamp	Name1	Name2	Name3	Name4	Name5
06/20/18 - 12:34:31	0	0	0	0	0
06/20/18 - 12:34:32	1	2	3	4	4
06/20/18 - 12:34:33	2	2	6	8	8
06/20/18 - 12:34:34	3	2	9	12	12
06/20/18 - 12:34:35	4	2	12	16	16
06/20/18 - 12:34:36	5	2	15	20	20
06/20/18 - 12:34:37	6	2	18	24	24

Boutons :

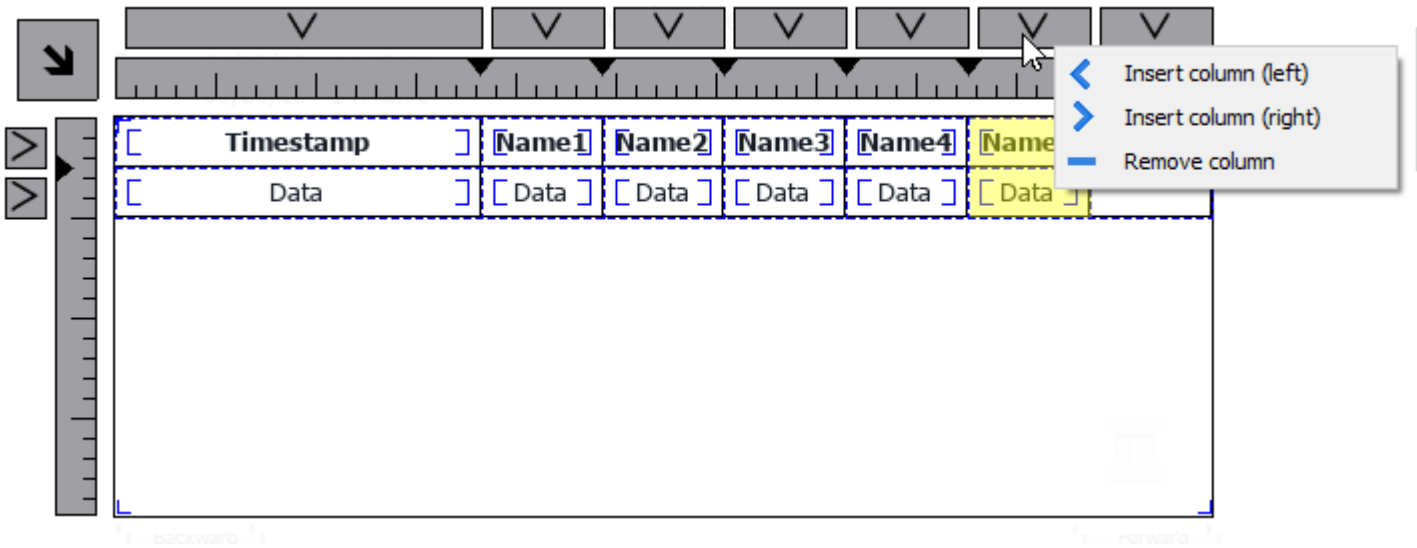
- REFRESH
  Récupère les données de tendance de la mémoire tampon interne et rafraichit la vue du tableau
- BACKWARD/FORWARD
  Déplace la fenêtre d'affichage vers l'avant ou vers l'arrière comme spécifié dans le paramètre de durée.

Paramètre	Description
TrendName	Mémoire tampon de tendance à partir de laquelle les échantillons sont récupérés (voir "Enregistrement des données" Page 228)
En-tête	Étiquette En-tête
Durée de page	Valeur initiale de la fenêtre de temps à afficher
Spéc heure	Format de l'heure : • **Locale** = affiche les valeurs de temps du périphérique IHM. • **Globale** = affiche les valeurs de temps au format UTC.
Format de date	Sélectionnez le format de Date et d'Heure
Disposition du tableau	Définit les caractéristiques de la barre de défilement et permet de supprimer l'en-tête du tableau.

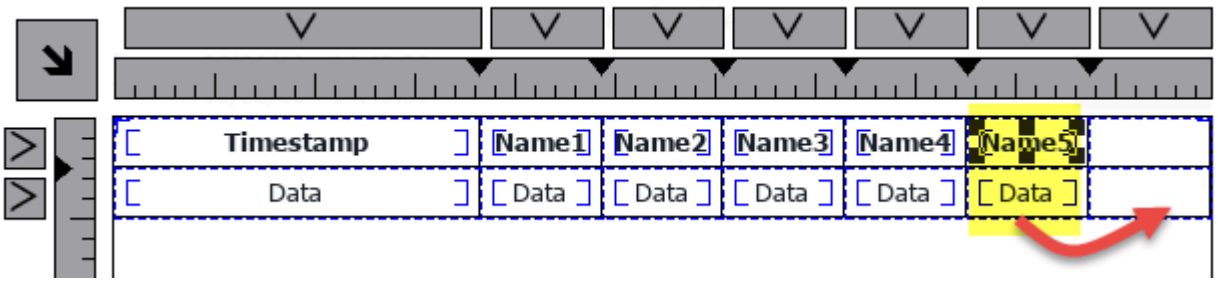
Ajouter ou retirer des colonnes de tendance

Pour ajouter ou retirer une colonne, double-cliquez sur la grille pour entrer en mode édition et cliquez avec le bouton droit de la souris sur le sélecteur de colonne pour ouvrir le menu contextuel à partir duquel insérer ou supprimer une colonne.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

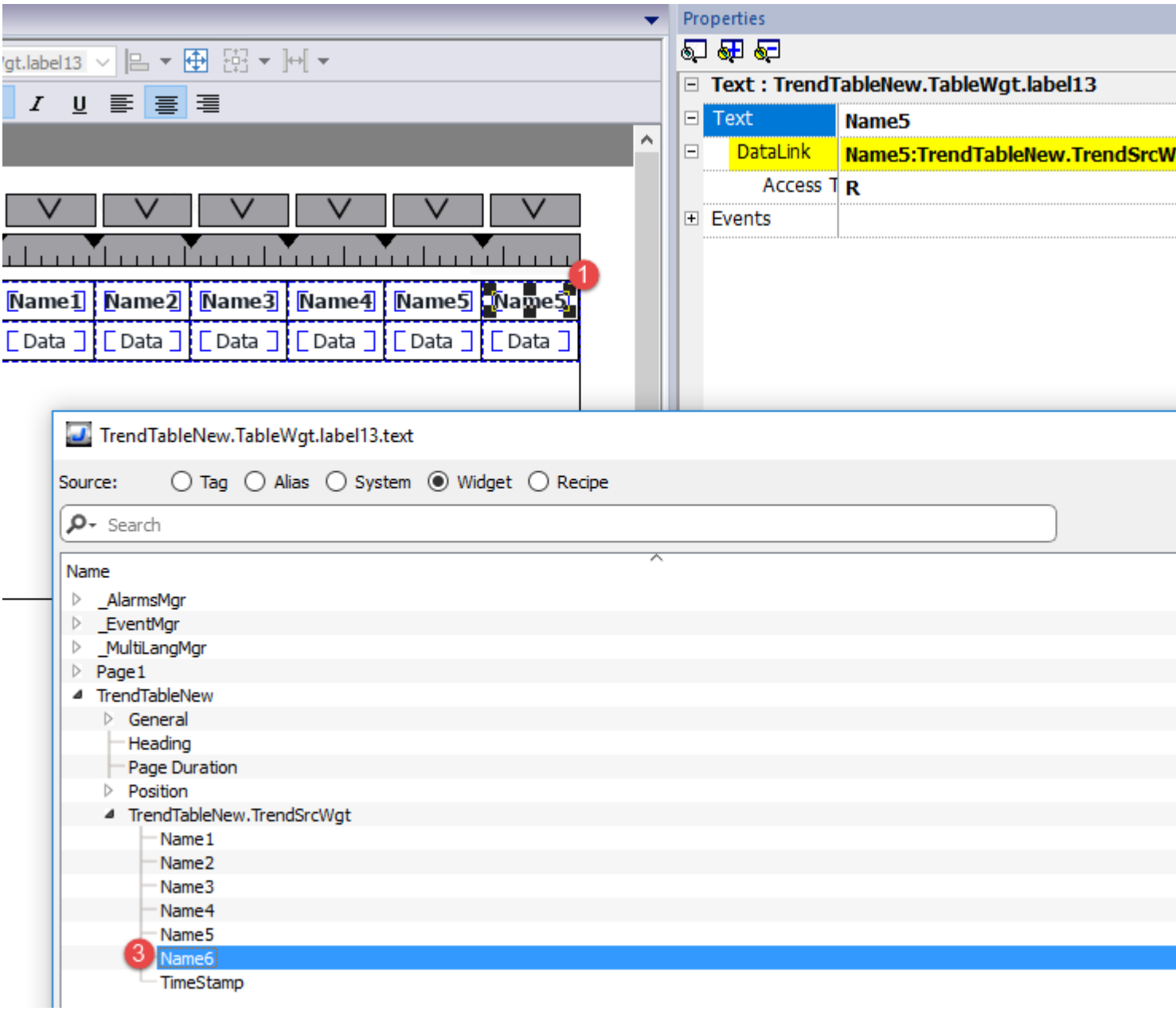


Copiez et collez les champs d'une autre colonne.

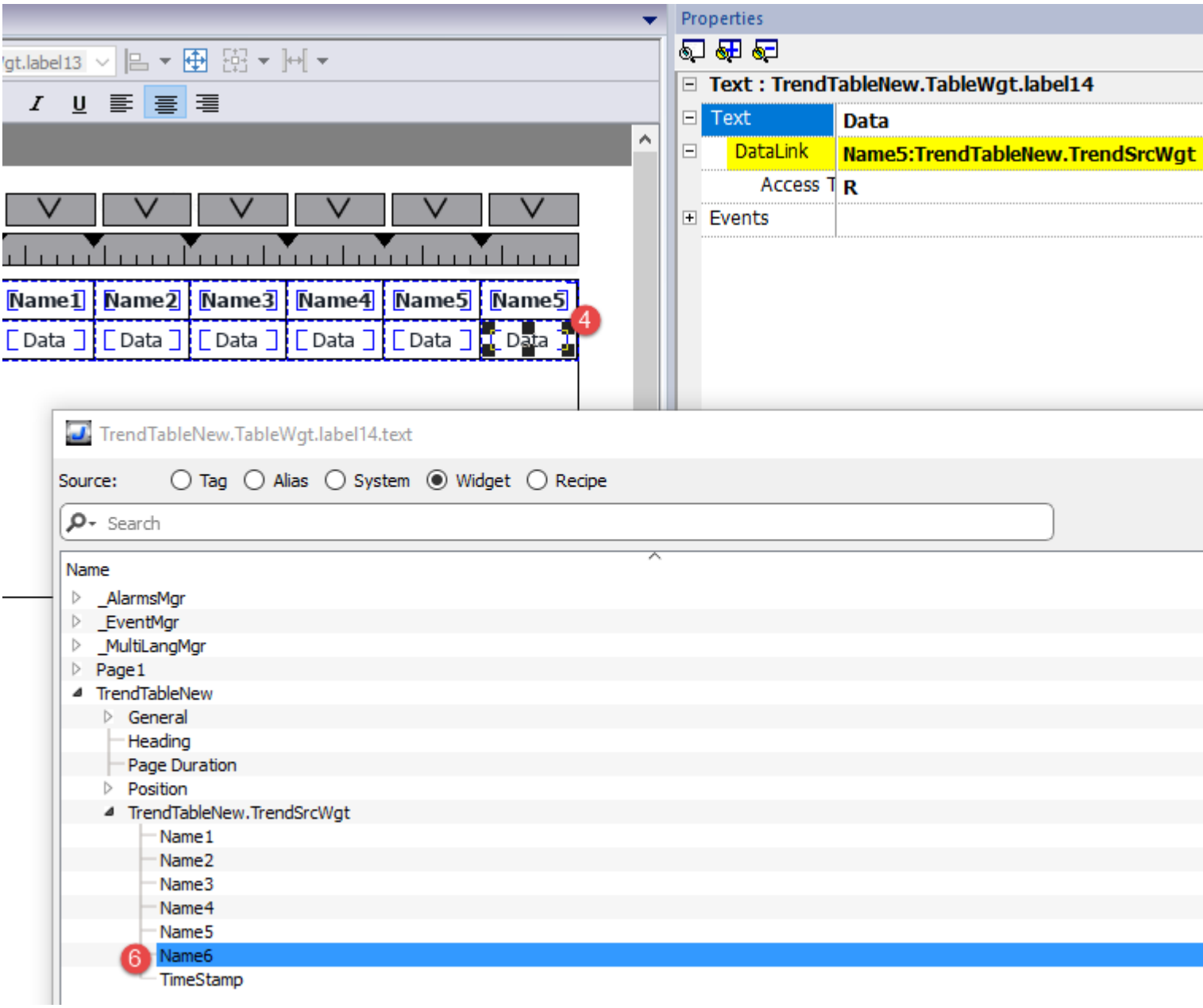


Allez ensuite dans le panneau des propriétés pour sélectionner l'élément de tendance à ajouter aux nouvelles colonnes.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

18 Transfert de données

Le transfert de données vous permet de transférer des données de variables d'un périphérique à l'autre. En utilisant cette fonctionnalité, un périphérique IHM peut opérer comme une passerelle entre deux périphériques, même lorsqu'ils n'utilisent pas le même protocole de communication.

Éditeur de transfert de données **246**

Exporter données vers fichiers .csv **248**

Limites et recommandations du transfert de données **248**

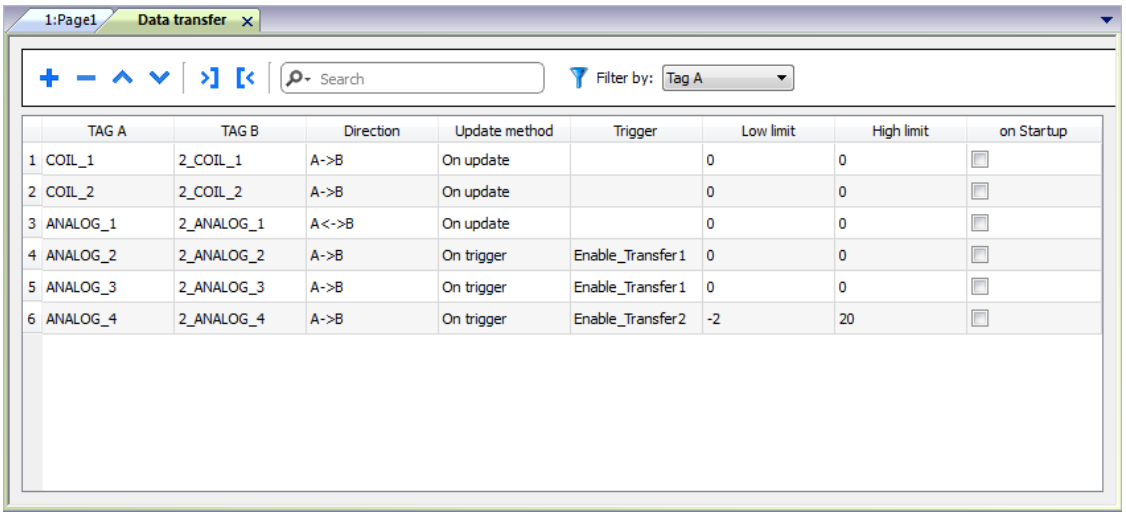
Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Éditeur de transfert de données

Chemin : ***VueProjet*** *>* ***Configuration*** *> double-cliquez sur* ***Transfer de données***

Utilisez l'éditeur de transfert de données pour cartographier les règles de transfert.

Chaque ligne de l'éditeur de transfert de données définit une règle de correspondance entre deux variables. Définissez plusieurs règles de correspondance si vous avez besoin d'un sens, d'une méthode de mise à jour ou d'un déclencheur différents.



	TAG A	TAG B	Direction	Update method	Trigger	Low limit	High limit	on Startup
1	COIL_1	2_COIL_1	A->B	On update		0	0	
2	COIL_2	2_COIL_2	A->B	On update		0	0	
3	ANALOG_1	2_ANALOG_1	A<->B	On update		0	0	
4	ANALOG_2	2_ANALOG_2	A->B	On trigger	Enable_Transfer1	0	0	
5	ANALOG_3	2_ANALOG_3	A->B	On trigger	Enable_Transfer1	0	0	
6	ANALOG_4	2_ANALOG_4	A->B	On trigger	Enable_Transfer2	-2	20	

Pour ajouter une nouvelle règle, cliquez sur **+** : une nouvelle ligne de variable apparaît.

Barre d'outil de transfert de données

Paramètre	Description
Import / Export	Importer ou exporter les paramètres de transfert de données depuis ou vers un fichier .csv.
Rechercher	Affiche uniquement les lignes contenant le mot-clé de recherche.
Filtrer par	Afficher uniquement les lignes filtre et champ de recherche correspondant.

Paramètres de transfert de données

Paramètre	Description
TAG A / TAG B	Paire de variables à cartographier pour l'échange via le dispositif IHM.
Direction	Direction de transfert. **A->B** et **B->A** : transferts unidirectionnels, les valeurs sont toujours copiées d'une variable et envoyées à l'autre variable dans le sens indiqué. **A<->B** : transfert bidirectionnel, les valeurs sont transférées vers et depuis les deux variables.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Paramètre	Description
Méthode de mise à jour	**On trigger** : le transfert de données se produit lorsque la valeur de la variable définie comme déclencheur change au-dessus ou en dessous des valeurs définies comme limites. Les limites sont recalculées d'après la valeur de la variable précédente, celle qui a déclenché la mise à jour. Remarque : cette méthode s'applique uniquement aux transferts unidirectionnels (A->B ou B->A). **On Update** : le transfert de données se produit à chaque fois que la valeur des variables source change. Remarque : cette méthode vaut aussi bien pour les transferts unidirectionnels que bidirectionnels (A->B, B->A et A<->B). Remarque : le runtime surveille cycliquement les changements des variables sources (variable de déclenchement lors de l'utilisation de On Trigger ou variables à transférer lors de l'utilisation de On Update) en fonction du paramètre **Rate** (Fréquence). Si le paramètre **Rate** pour la variable source est 500 ms (par défaut), le système vérifie les mises à jour toutes les 500 ms. Remarque : les modifications sur les variables sources plus rapides que la **Rate** ne peuvent être pas détectées.
Déclencheur, Limite supérieure, Limite inférieure	Variable qui déclenche le processus de transfert de données. Lorsque cette variable modifie sa valeur en dehors des limites fixées comme **Limite supérieure** et **Limite inférieure**, le transfert de données démarre. La plage de tolérance est recalculée en fonction des limites spécifiées sur la valeur de la variable ayant déclenché la mise à jour précédente. Aucune action n'est prise si le changement se situe dans les limites. Ce mécanisme permet de déclencher des transferts de données uniquement lorsque les variations significatives des valeurs de référence se produisent. La **limite inférieure** est inférieure ou égale à zéro. Si les deux **Limite inférieure** et **Limite supérieure** sont définies sur « 0 », le transfert de données se produit à chaque fois que la valeur de la variable de déclenchement change.
au démarrage	Lorsqu'il est sélectionné, le transfert de données est forcé : • sur un démarrage IHM si la qualité de la variable source est bonne • après des erreurs de communication, lorsque les nœuds de périphérique associé redeviennent actifs Voir "Objets" Page 439 pour les détails sur la qualité. **Important : les transferts de données exécutés au démarrage peuvent avoir un impact majeur sur le temps de démarrage du périphérique IHM. Activez cette option seulement si nécessaire.**

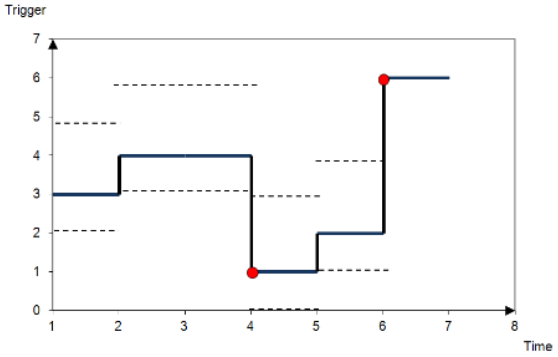
Exemple de réglage de la limite

Limite supérieure = 1,9

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Limite inférieure = - 0,9

• = points où le transfert de données est déclenché



Exporter données vers fichiers .csv

Les informations de configuration pour les transferts de données peuvent être exportées vers un fichier .csv.

Exemple de paramètres de transfert de données vers un fichier .csv

A	B	C	D	E	F	G	H	I	J
COIL_1	2_COIL_1	A->B	On update		0	0	data1	true	1
COIL_2	2_COIL_2	A->B	On update		0	0	data2	true	1
ANALOG_1	2_ANALOG_1	A<->B	On update		0	0	data3	true	1
ANALOG_2	2_ANALOG_2	A->B	On trigger	Enable_Transfer1	0	0	data4	true	1
ANALOG_3	2_ANALOG_3	B->A	On trigger	Enable_Transfer1	0	0	data5	true	1
ANALOG_4	2_ANALOG_4	A->B	On trigger	Enable_Transfer2	-10	20	data6	true	1

Colonne	Description
A à G	Même données que dans l'éditeur de transfert de données
H	Identifiant uniquement associé automatiquement à chaque ligne. **Important : lorsque vous éditez le fichier .csv et que vous ajoutez toute ligne supplémentaire, vérifiez que vous entrez bien un identifiant unique dans cette colonne.**
I et J	Réservés à une utilisation ultérieure.

L'import / export utilise le caractère de séparation définie à l'intérieur des paramètres régionaux de Windows.

Limites et recommandations du transfert de données

Une définition correcte des règles de transfert des données est critique pour la performance des périphériques IHM. Pour garantir une opération et une performance fiables, gardez à l'esprit les règles suivantes.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Méthode dès le déclenchement

La méthode **Dès le déclenchement** ne permet que des transferts unidirectionnels (A->B ou B->A).

Le transfert de données basé sur le mode **Dès le déclenchement** doit être privilégié car il vous permet de forcer le transfert et il surveille uniquement les variables de déclenchement et non toutes les variables impliquées dans le transfert.

Méthode dès la mise à jour

La méthode **Dès la mise à jour** permet de changer les valeurs selon les paramètres de direction uniquement lorsque la valeur de la source est modifiée.

En utilisant la méthode **Dès la mise à jour**, vous forcez le système à lire continuellement toutes les variables de source définies pour vérifier si des modifications doivent être transférées. La valeur par défaut de la fréquence des mises à jour de chaque variable est de 500 ms et peut être modifiée avec l'Éditeur de variables.

Observations sur la performance

La performance du transfert de données dépend :

- du nombre de transferts de données définis,
- du nombre de transferts de donnés qui pourrait se produire en même temps,
- de la fréquence des modifications des variables API qui sont surveillées,

Important : testez toujours la performance de l'opération durant le développement du projet.

Important : si les tâches de transfert de données sont définies de manière inappropriée, elles peuvent entraîner des conditions où les variables impliquées créent des boucles. Identifiez et évitez de telles conditions.

Conseil : utilisez le planificateur pour calibrer la fréquence de mise à jour sur la performance de l'ensemble de votre projet.

Conseil : utilisez des variables de type tableau pour optimiser le transfert de données et réduire la charge de travail.

Conseil : réduisez le nombre de transferts de données pour réduire le temps de modification de la page et le temps de chargement.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

19 Gestion de nœud hors ligne

Lorsqu'un contrôleur en communication avec les périphériques IHM est mis hors connexion, la performance de communication du système pourrait diminuer.

La fonctionnalité de gestion de nœud hors ligne reconnaît les contrôleurs hors connexion et les écarte de la communication jusqu'à leur remise en ligne.

De plus, si vous savez qu'un contrôleur inclus dans l'installation sera mis hors connexion pendant un certain temps, vous pouvez le désactiver manuellement afin d'optimiser la performance du système.

Remarque : cette fonctionnalité n'est pas supportée par tous les protocoles de communication. Vérifiez la documentation de protocole pour savoir si elle est supportée ou pas.

Processus de gestion de nœud hors ligne **252**

Processus de gestion manuelle de nœud hors ligne **252**

Configuration manuelle hors ligne **252**

Détection automatique de nœuds hors ligne **253**

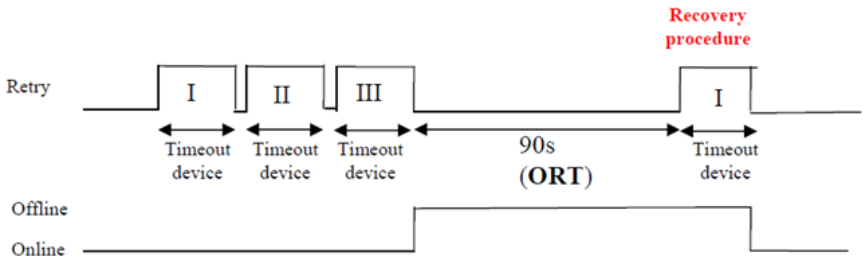
Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Processus de gestion de nœud hors ligne

Les étapes de ce processus sont :

- Le système communique normalement avec un périphérique donné. En cas d'absence de réponse du périphérique à une demande de communication, le système répète la demande deux fois, puis déclare le périphérique hors ligne.
- Quand un périphérique est hors ligne, le système envoie des demandes de communication au périphérique selon un intervalle plus long, que l'on appelle Délai d'attente avant nouvelle tentative. Si le périphérique répond à l'une de ces demandes, le système le déclare en ligne et reprend une communication normale.

Le diagramme montre les trois tentatives de communication et la procédure de restauration qui démarre dès l'expiration du Délai d'attente avant nouvelle tentative.



Processus de gestion manuelle de nœud hors ligne

La gestion de nœud hors ligne peut être faite manuellement. Lorsqu'un périphérique spécifique est connecté et communique normalement, vous pouvez :

- utiliser une action pour déclarer le périphérique hors ligne : le système interrompra la communication avec le périphérique.
- utiliser une action pour déclarer le périphérique en ligne : le système reprendra la communication normale avec le périphérique.

Configuration manuelle hors ligne

Lorsque vous savez que certains périphériques en communication avec le périphérique IHM resteront déconnectés pendant un certain temps, vous pouvez les exclure de la communication en recourant à l'action **ActiverNœud**.

AVERTISSEMENT : tous les nœuds de périphérique désactivés resteront désactivés si le même projet est téléchargé sur le périphérique, a contrario, si un projet différent est téléchargé, tous les périphériques désactivés seront réactivés. Il en va de même en cas de mise à jour de paquet.

Conseil : pour rendre cette fonctionnalité plus dynamique, vous pouvez décider de ne pas indiquer d'**ID de nœud** spécifique, mais de l'associer à la valeur d'une variable ou à une variable interne créée pour identifier les différents périphériques éventuellement installés sur votre réseau.

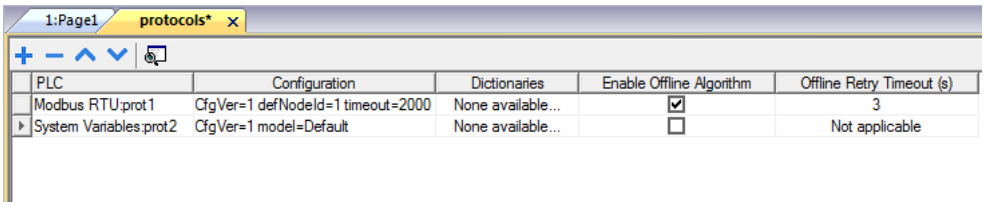
Remarque : lorsque vous utilisez l'action **ActiverNœud** pour forcer la remise en ligne d'un nœud de périphérique, la communication reprend immédiatement.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Détection automatique de nœuds hors ligne

Lorsqu'un périphérique ne répond pas aux demandes de communication, il est désactivé. Le périphérique IHM arrête d'envoyer des demandes à ce périphérique. Après trois secondes, le périphérique IHM envoie une commande pour vérifier si le périphérique est disponible, auquel cas la communication est redémarrée. Dans le cas contraire, le périphérique est désactivé pour un autre intervalle de délai d'attente.

Les paramètres par défaut peuvent être modifiés dans l'éditeur Protocole.



Remarque : seuls certains protocoles prennent en charge cette fonction.

Paramètre	Description
Activer algorithme hors ligne	Active la gestion hors ligne du protocole
Délai d'attente nouvelle tentative hors ligne	Intervalle en secondes du cycle de nouvelle tentative après la désactivation d'un périphérique. Plage : 1–86.400 seconds (24 h).

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

20 Multilangue

La fonctionnalité multilangue a été conçue pour créer des applications IHM contenant des textes en plusieurs langues simultanées

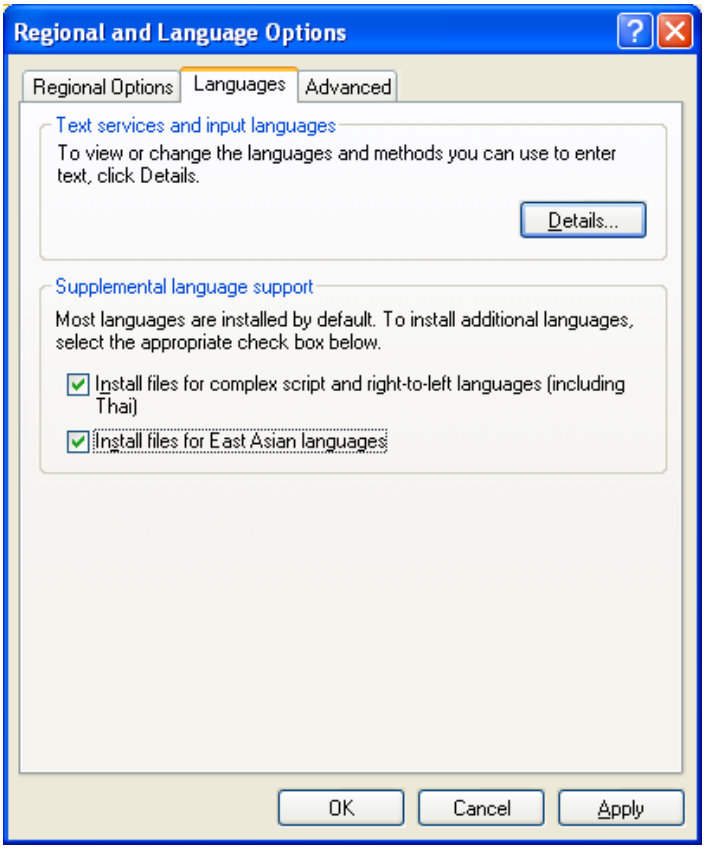
La fonctionnalité multilangue utilise un support de pages de code pour gérer les différentes langues. Une page de code (ou un fichier de script) est une série de formes de lettres utilisées au sein de chaque langue.

La fonctionnalité multilangue peut être utilisée pour définir les langues et les jeux de caractères d'un projet. PB610 Panel Builder 600 développe les polices TrueType proposées par les systèmes Windows afin de fournir différentes familles de police associées à différents jeux de caractères.

PB610 Panel Builder 600 vous permet également de fournir des chaînes pour chaque langue supportée.

PB610 Panel Builder 600 vous permet enfin de modifier la langue d'affichage afin de vous donner une idée de l'aspect et de la convivialité de la page pendant la phase de conception.

Important : dans les systèmes d'exploitation Windows XP, vous devez installer le support pour les scripts complexes et les langues de l'Asie de l'est.



Polices supportées pour le Chinois Simplifié

Pour le Chinois Simplifié, les polices suivantes sont supportées :

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Nom de police	Nom de fichier
Fangsong	simfang.ttf
Arial Unicode MS	ARIALUNI.TTF
Kaiti	simkai.ttf
Microsoft Yahei	msyh.ttf
NSImsun	simsun.ttc
SimHei	simhei.ttf
Simsun	simsun.ttc

Polices supportées pour le Chinois Traditionnel

Pour le Chinois Traditionnel, les polices suivantes sont supportées :

Nom de police	Nom de fichier
DFKai-SB	kaiu.ttf
Microsoft Sheng Hai	msjh.ttf
Arial Unicode MS	ARIALUNI.TTF
MingLiU	mingliu.ttc
PMingLiU	mingliu.ttc
MingLiU_HKSCS	mingliu.ttc

L'éditeur multilangue **257**

Changer de langue **258**

Widgets multilangues **258**

Exporter/importer des chaînes multilangues **260**

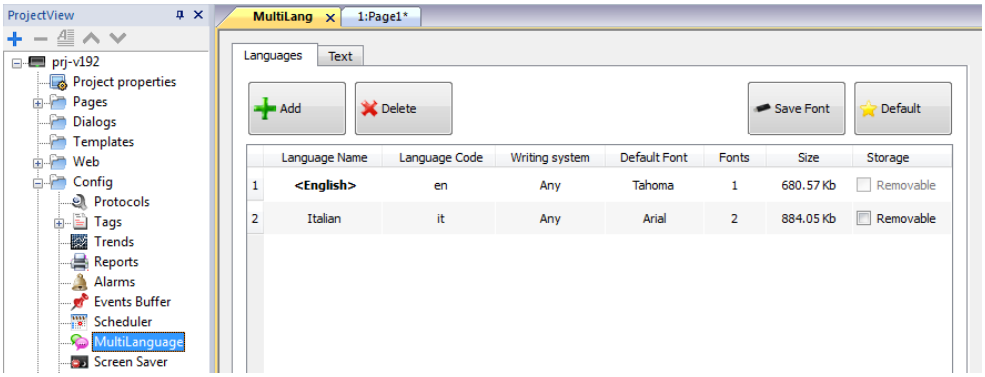
Modifier la langue au runtime **262**

Limites de la prise en charge UNICODE **262**

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

L'éditeur multilangue

*Chemin : **VueProjet**> **Configuration** > double-cliquez sur **Multilangue***



Paramètres de langue

Paramètre	Description
Nom de la langue	Nom identifiant la langue dans le projet.
Code de la langue	Identificateur du code de la langue ISO 639, utilisé pour associer des éléments de langue lors de l'importation de ressources depuis des fichiers xml externes.
Système d'écriture	Sélectionnez le jeu de polices à utiliser avec la langue
Police par défaut	Police par défaut pour les widgets du projet. Remarque : lorsque vous choisissez une nouvelle police, vous êtes invité à remplacer la police utilisée dans les widgets que vous avez déjà créés.
Polices	Nombre de polices associées avec la langue sélectionnée.
Taille	Mémoire utilisée pour stocker les fichiers de polices.
Stockage	Le fichier de polices est stocké dans une mémoire externe amovible. Conseil : stockez les fichiers de polices volumineux sur une mémoire amovible pour libérer de la mémoire requise dans le périphérique IHM.

Ajouter une langue

1. Dans l'onglet **Langues**, cliquez sur **+** : une ligne s'ajoute au tableau.
2. Entrez tous les paramètres de langue.
3. Cliquez sur **Par défaut** pour définir la langue sélectionnée comme langue par défaut lorsque le runtime démarre.
4. Cliquez sur **Sauvegarder police** pour copier les polices que vous avez marquées comme **Amovibles** sur une mémoire externe.

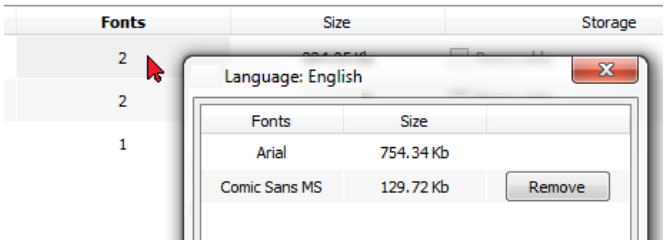
Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Important : les fichiers de polices configurés pour être stockés sur une mémoire amovible doivent être fournis à l'utilisateur final pour terminer l'installation de la police sur le périphérique IHM.

Supprimer des polices

Pour supprimer des polices qui ne sont plus nécessaires :

1. Cliquez sur le numéro de police dans l'éditeur multilangue : une boîte de dialogue contenant la liste des polices utilisée s'affiche.

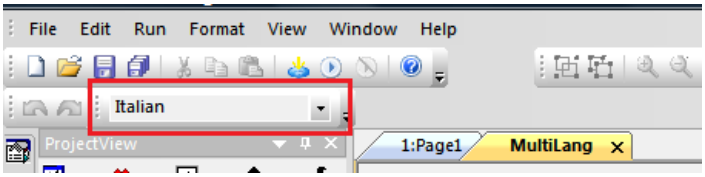


2. Sélectionnez les polices à supprimer et cliquez sur **Supprimer** : les polices supprimées sont remplacées par la police par défaut.

Changer de langue

Changer la langue pendant la conception de page

Une liste déroulante est disponible pour modifier la langue pendant la conception de page. Si aucun texte n'apparaît, veuillez sélectionner l'onglet **Texte** dans l'éditeur multilangue et entrez la chaîne manquante.



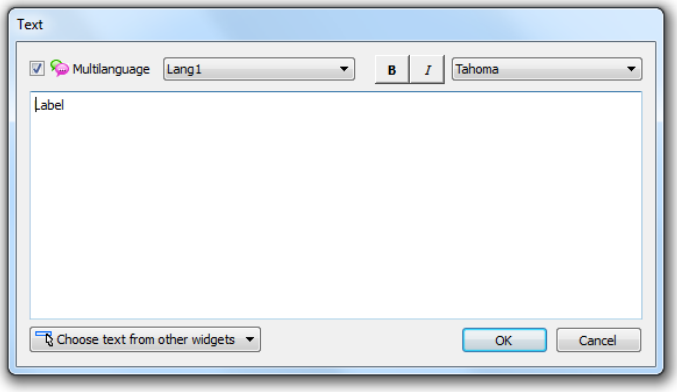
Widgets multilangues

Le support multilangue est disponible pour les objets tels que les boutons, les textes statiques, les messages, les descriptions d'alarme et les messages contextuels.

Multilangue pour les widgets d'étiquette

Double-cliquez sur un widget de texte dans une page pour ouvrir la boîte de dialogue **Texte.**

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



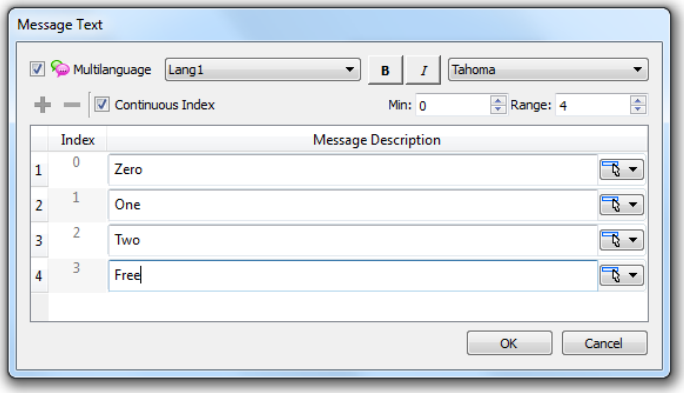
Activez/Désactivez la fonction multilangue, éditez le texte pour la langue sélectionnée et choisissez la police.

Remarque : les propriétés caractère gras, italique et de couleurs définies ici pour le widget sont applicables à toutes les langues.

Paramètre	Description
Multilangue	Activez/désactivez la fonction multilangue pour le widget.
Choisir texte depuis un autre widget	Cliquez sur le bouton pour parcourir les chaînes de message présentes dans le projet afin de sélectionner un texte pour le widget.

Multilangue pour les widgets de message

Double-cliquez sur un widget de message dans une page pour ouvrir la boîte de dialogue **Texte de message.**



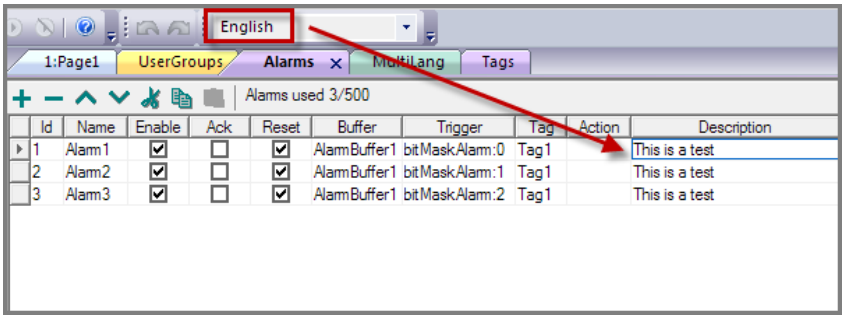
Paramètre	Description
Multilangue	Activez/désactivez la fonction multilangue pour le widget.
Index continu	L'index pour le widget est une série de chiffres consécutifs (exemple 3, 4,5,6)
Min	Chiffre de départ pour l'index
Plage	Nombre de messages
Choisir texte depuis un autre widget	Cliquez sur le bouton pour parcourir les chaînes de message présentes dans le projet afin de sélectionner un texte pour le widget.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Multi-langue pour les messages d'alarme

Pour ajouter des chaînes multilangues pour les messages d'alarme :

1. Ouvrez l'Éditeur d'alarme.
2. Sélectionnez une langue en utilisant la liste déroulante des langues.
3. Entrez le texte pour l'alarme dans la colonne **Description**.

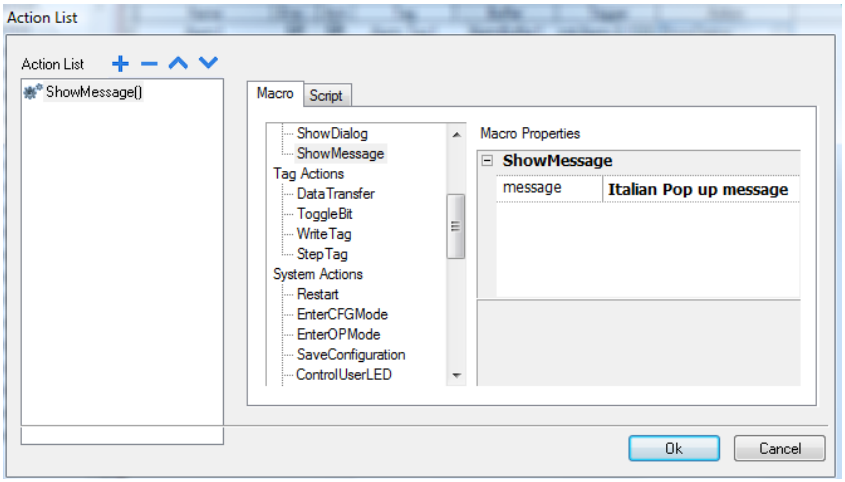


Conseil : les étiquettes de texte avec les états d'alarme affichés par les widgets d'alarme peuvent être traduites ou personnalisées à l'aide de l'éditeur de texte multilangue.

Multilangue pour les messages contextuels

Pour ajouter un message contextuel multilangue :

1. Sélectionnez une langue dans la liste déroulante des langues.
2. Ajoutez l'action de page **Afficher Message** et entrez le texte dans la langue sélectionnée.



Exporter/importer des chaînes multilangues

La manière la plus simple de traduire un projet dans plusieurs langues est d'exporter tout le texte dans un fichier .csv, de traduire le document qui en résulte puis de réimporter le texte traduit dans le projet.

Important : le fichier .csv exporté par PB610 Panel Builder 600 est codé en Unicode, vous avez donc besoin d'un outil spécifique qui prend en charge les fichiers .csv encodés en Unicode pour l'éditer.

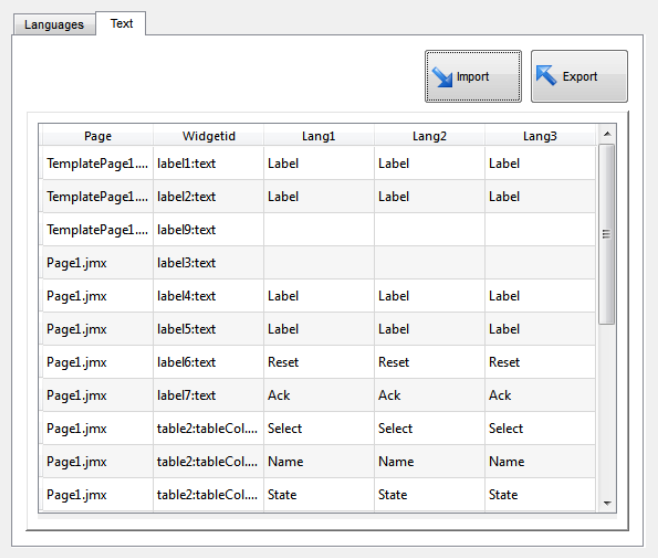
Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Exporter et réimporter des chaînes

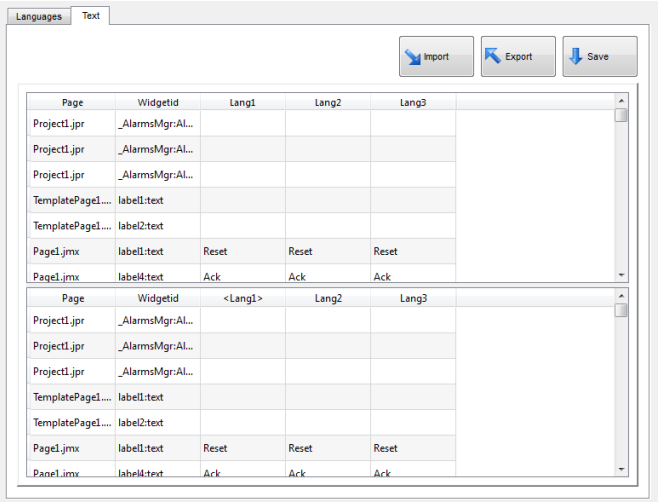
*Chemin : **VueProjet**> **Configuration** > double-cliquez sur **Multilangue***

Pour exporter et réimporter des chaînes multilangues :

1. Dans l'onglet **Texte**, cliquez sur **Exporter** : toutes les chaînes multilangues sont exportées dans un fichier .csv.



Important : définissez toutes les langues qui sont utilisées dans le projet avant d'exporter le fichier. Cela garantit que le fichier exporté contienne toutes les définitions de colonnes et de langue.



2. Lorsque les chaînes ont été traduites, cliquez sur **Importer** pour les réimporter dans le projet : les chaînes sont importées selon l'ID du widget et le numéro de page de chaque widget.
3. Cliquez sur **Sauvegarder** pour sauvegarder les données du nouveau widget.

Remarque : pour changer le séparateur utilisé dans le fichier exporté, changez les paramètres régionaux de votre ordinateur. Lors de l'importation, l'information du séparateur est récupérée dans le fichier. Si elle est introuvable, le caractère par défaut « , » est utilisé.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Contraintes d'importation

Les formats suivants sont pris en charge pour l'importation :

- Valeurs séparées par des virgules (.csv)
- Texte Unicode (.txt)

Remarque : utilisez le format de fichier en Texte Unicode lorsque vous importez un fichier modifié avec Microsoft® Excel®.

Modifier la langue au runtime

Modifier la langue avec une action

Après le téléchargement du projet, le runtime IHM utilise la langue définie par défaut. Vous pouvez modifier la langue en utilisant l'action **ConfigurerLangue**. Voir "Actions multilangues" Page 144.

Remarque : lorsque la langue est modifiée, elle sera également utilisée lors des sessions ultérieures.

Le code de langue active est disponible à partir de l'API JavaScript. Voir "curLangCode" Page 459 pour plus de détails.

Polices manquantes

Lorsque vous changez de langue, si les polices nécessaires ne sont pas disponibles dans la mémoire du périphérique, un message contextuel vous invite à insérer la carte mémoire contenant les polices manquantes. À la fin de l'opération, vous pouvez retirer la carte mémoire.

Limites de la prise en charge UNICODE

PB610 Panel Builder 600 a été conçu pour accepter le texte Unicode. Cependant, en cas de problèmes d'incompatibilité avec certaines plateformes, Unicode n'est pris en charge que dans un sous-ensemble de propriétés.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Domaine	Propriété	Jeu de caractères accepté	Caractères/Chaînes réservés
Éditeur de protocole	Alias	ASCII [32..126]	(espace) , ; : . < * >'
Éditeur de variable	Nom	ASCII [32..126]	. \ / * ? : > < | " & # %;=
	Groupe	ASCII [32..126]	<Nouveau> \ / * ? : > < | " & # % ;
	Commentaire	Unicode	
Tendances	Nom	ASCII [32..126]	\ / * ? : > < | " & # % ;
Rapports d'impression	Nom	ASCII [32..126]	\ / * ? : > < | " & # % ;
Alarmes	Nom	ASCII [36..126]	\ / * ? : > < | " & # % ;
	Description	Unicode	[] - pour les variables actives, \ échap pour [et \
Événements	Nom de mémoire tampon	ASCII [32..126]	\ / * ? : > < | " & # % ;
Planificateur	Nom	ASCII [32..126]	\ / * ? : > < | " & # % ;
Langues	Nom de la langue	ASCII [32..126]	\ / * ? : > < | " & # % ;
	Textes dans les widgets	Unicode	-
	Textes issus de fichiers importés	Unicode	-
Groupe utilisateur	Nom de groupe	a-z A-Z _	admin,guest,unauthorized
	Commentaires	Unicode	-
Utilisateur	Nom	ASCII [32..126]	\ / * ? : > < | " & # % ;
	Mot de passe	Unicode	-
	Commentaire	Unicode	-
Recettes	Nom	ASCII [32..126]	\ / * ? : . > < | " & % ; ,
	Nom d'ensemble	ASCII [32..126]	\ / * ? : . > < | " & % ; ,
	Nom d'élément	ASCII [32..126]	\ / * ? : . > < | " & % ; ,
Généralités	Nom du projet	A-Z,a-z,0-9,-,_	“PUBLIC”, “readme”, “index.html”
	Nom de page	A-Z,a-z,0-9,-,_	-

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Domaine	Propriété	Jeu de caractères accepté	Caractères/Chaînes réservés
	Nom de page de dialogue	A-Z,a-z,0-9,-,_	-
	Nom de page de modèle	A-Z,a-z,0-9,-,_	-
	Nom du clavier	A-Z,a-z,0-9,-,_	-
	Fichiers (Images/Vidéos/etc.)	A-Z,a-z,0-9,-,_	-
	ID de widget	A-Z,a-z,0-9,-,_	-
Runtime	Communication API	UTF-8, Latin1, UCS-2BE, UCS-2LE, UTF-16BE, UTF-16LE	-

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

21 Planificateur

PB610 Panel Builder 600 fournit un moteur de planificateur qui peut exécuter des actions spécifiques à des intervalles définis, ou sur une base temporelle.

La création d'un emploi du temps est généralement un processus en deux étapes :

1. Vous créez un calendrier avec une liste d'actions à exécuter lorsque l'événement programmé se produit. Vous faites cela dans l'éditeur du planificateur.
2. Vous créez une interface utilisateur d'exécution qui permet à l'utilisateur final de modifier les paramètres pour chaque calendrier. Vous faites cela en ajoutant un widget **Planificateur** à une page de votre projet et la configuration pour l'adapter aux besoins de planification de l'utilisateur.

Créer un ordonnancement **266**

Ordonnancement Haute Résolution **266**

Ordonnancement des tâches répétitives **266**

Configurer un emplacement pour les ordonnancements **268**

Configurer le widget planificateur **269**

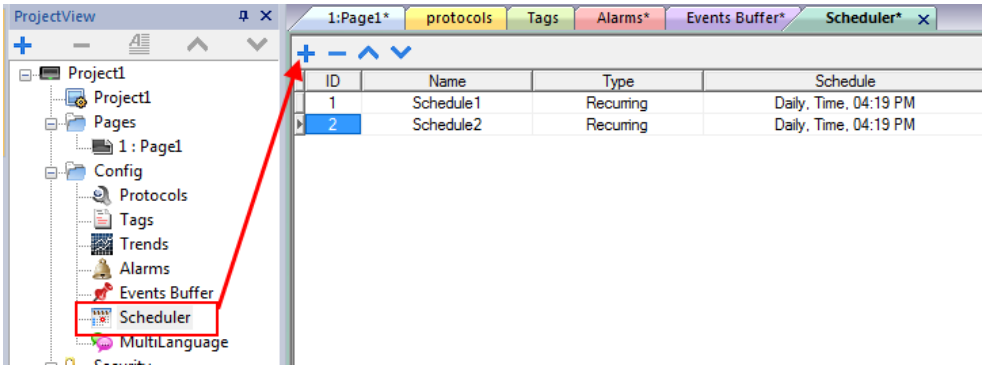
Ordonnancement des événements au runtime **270**

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Créer un ordonnancement

Chemin : ***VueProjet*** *>* ***Configuration*** *> double-cliquez sur* ***Planificateurs***

- Cliquez sur **+** pour ajouter un ordonnancement.



Paramètres d'ordonnancement

Paramètre	Description
ID	Code unique assigné automatiquement à l'ordonnancement
Nom	Nom de l'ordonnancement
Type	Type d'ordonnancement : • **Répétitif**, voir "Ordonnancement des tâches répétitives" bas pour plus de détails. • **Haute Fréquence**, voir "Ordonnancement Haute Résolution" bas pour plus de détails
Ordonnancer	Paramètres et options du planificateur. Voir "Ordonnancement des tâches répétitives" bas pour plus de détails.
Action	Actions devant être exécutées à un moment programmé
Priorité	Niveau de priorité pour l'événement. Si deux ordonnancements se produisent au même moment, l'événement avec la priorité la plus haute sera exécuté en premier.

Ordonnancement Haute Résolution

L'ordonnancement **HauteRésolution** sert à exécuter des actions qui demandent à être répétées à des intervalles donnés. L'intervalle entre les exécutions est défini en millièmes de seconde dans la colonne **Ordonnancement**.

Remarque : vous ne pouvez pas modifier les paramètres de ce type d'ordonnancement au runtime. Vous devez modifier les paramètres de temps d'action au runtime, sélectionnez **Ordonnancement répétitif** et configurez le **Type** sur **Chaque**. Voir "Ordonnancement des tâches répétitives" bas pour plus de détails.

Ordonnancement des tâches répétitives

L'ordonnancement des tâches répétitives est utilisé pour exécuter des actions à des moments précis. Les paramètres peuvent être modifiés au runtime.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Paramètres du planificateur des tâches répétitives

Paramètre	Description
Type	Fréquence des actions planifiées
Mode	Paramètres spécifiques demandés par chaque type de planificateur
Condition	Une variable booléenne (vrai/faux) pour activer les actions spécifiées au moment où la minuterie se déclenche. Les actions seront exécutées si la variable = vraie. Par défaut, les actions sont exécutées lorsque la minuterie est déclenchée. Remarque : seules les variables associées au type de données booléennes sont affichées.
Actions	Actions devant être exécutées par l'ordonnancement. **Important : les actions et les paramètres d'ordonnancement ne peuvent pas être modifiés au runtime**
Date	Date à laquelle les actions planifiées seront exécutées
Heure/Décalage	Ce champ montre l'une des données suivantes : **Heure** = moment où les actions planifiées seront exécutées **Décalage** = retard ou avance par rapport au mode sélectionné.
Location (Localité)	Lieu de référence pour calculer l'heure de lever/coucher du soleil.
jours de la semaine	Jours de la semaine où les actions planifiées seront exécutées.
au démarrage	Exécute l'ordonnancement au démarrage
Activer ordonnancement	Active/désactive l'ordonnancement
Exécuter seulement au démarrage	Exécute l'ordonnancement une fois seulement au démarrage

Options du type d'ordonnancement

Option	Description
Par date	Les actions sont exécutées à la date et à l'heure spécifiés.
Quotidiennement	Les actions sont exécutées quotidiennement à l'heure spécifiée.
Chaque	Les actions sont exécutées selon l'intervalle spécifié (Plage : 1 s - 1 jour)
Chaque heure	Les actions sont exécutées chaque heure, à la minute spécifiée.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Option	Description
Chaque mois	Les actions sont exécutées chaque mois, à la date et à l'heure spécifiés.
Hebdomadaire	Les actions sont exécutées chaque semaine, au(x) jour(s) et heures spécifiés.
Annuel	Les actions sont exécutées chaque année à la date et à l'heure spécifiés.

Options du mode d'ordonnancement

Option	Description
Time	Dépend du type d'ordonnancement. Permet de préciser les informations sur la date/heure/semaine.
Random10	Les actions sont exécutées selon un intervalle de 10 minutes avant ou après l'heure définie. Si, par exemple, l'heure définie est 10:30, les actions pourront être exécutées à tout moment entre 10:20 et 10:40.
Random20	Les actions sont exécutées selon un intervalle de 20 minutes avant ou après l'heure définie. Si, par exemple, l'heure définie est 10:30, les actions pourront être exécutées à tout moment entre 10:10 et 10:50.
Sunrise+	Les actions sont exécutées selon un délai spécifié après le lever du soleil. Ce délai s'exprime en minutes/heures et l'heure de lever du soleil dépend du lieu.
Sunrise-	Les actions sont exécutées selon une anticipation spécifiée avant le lever du soleil. Cette anticipation s'exprime en minutes/heures et l'heure de lever du soleil dépend du lieu.
Sunset+	Les actions sont exécutées selon un délai spécifié après le coucher du soleil. Ce délai s'exprime en minutes/heures et l'heure de coucher du soleil dépend du lieu.
Sunset-	Les actions sont exécutées selon une anticipation spécifiée avant le coucher du soleil. Cette anticipation s'exprime en minutes/heures et l'heure de coucher du soleil dépend du lieu.

Voir "Configurer un emplacement pour les ordonnancements" bas pour plus de détails sur les paramètres lever du soleil et coucher du soleil.

Remarque : les options de **mode** ne sont pas disponible pour tous les types d'ordonnancement.

Configurer un emplacement pour les ordonnancements

Les actions programmées peuvent être configurées pour être exécutées à un moment donné par rapport au lever et/ou au coucher du soleil. Pour ce faire, vous devez définir la localisation correcte, selon les informations UTC. Le système calcule automatiquement l'heure du lever du soleil et du coucher du soleil.

Seules quelques localisations sont disponibles par défaut. Si votre localisation n'est pas listée, vous pouvez l'ajouter en entrant la latitude, la longitude et les informations UTC dans le fichier Target_Location.xml.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Important : chaque plateforme a son propre fichier Target_Location.xml.

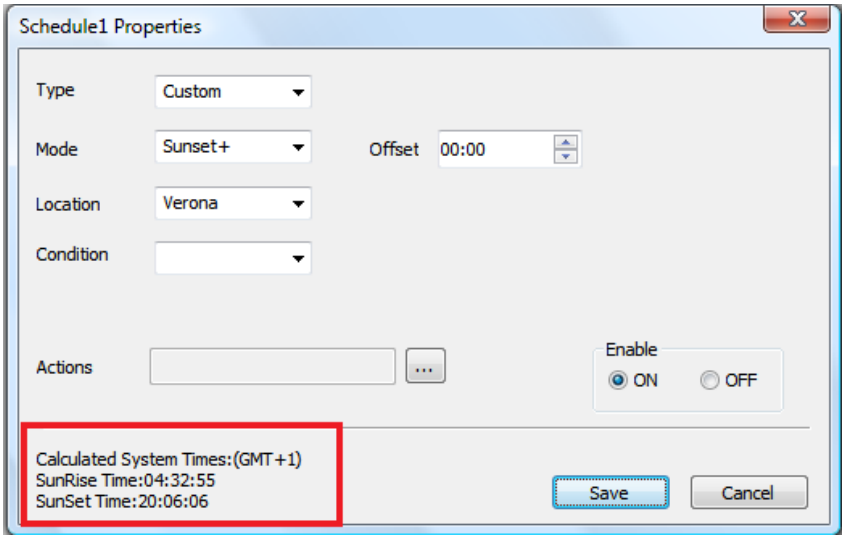
Position des fichiers de localisation

Application	Chemin des fichiers de localisation
PB610 Panel Builder 600	*ABB\Panel Builder 600 Suite\languages\shared\studio\config\Target_Location.xml*
Périphériques IHM	*ABB\Panel Builder 600 Suite\runtime\<HW Platform>\config\Target_Location.xml*
Simulateur	*ABB\Panel Builder 600 Suite\simulator\config\Target_Location.xml*
PB610 PC Runtime	*ABB\Panel Builder 600 Suite\server\config\Target_Location.xml*

Par exemple, les informations pour la ville de Vérone (Italie) sont affichées ci-dessous :

```
<file city="Verona" latitude="45.44" longitude="10.99" utc="1"/>
```

Les informations de localisation sont aussi affichées dans la boîte de dialogue, avec les heures de coucher et de lever du soleil.

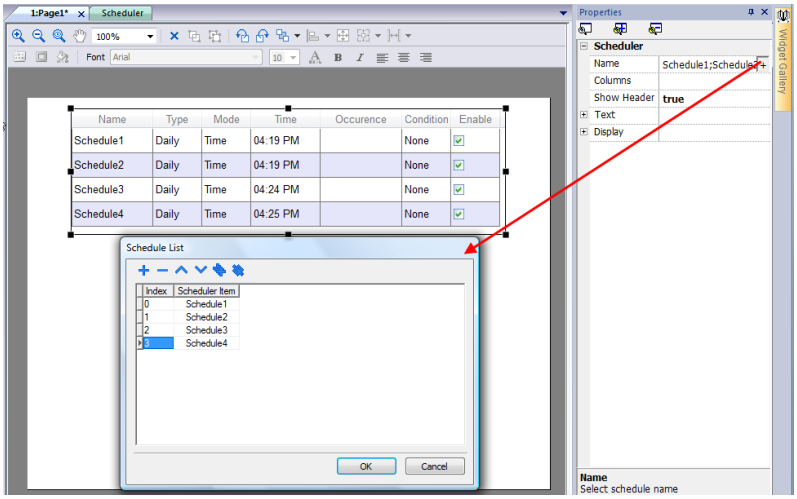


Configurer le widget planificateur

Pour afficher les données du planificateur sur une page :

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

1. Glissez-déposez un widget **Planificateur** depuis la galerie des widgets dans la page.
2. Dans le volet **Propriétés**, cliquez sur **+** pour le paramètre **Nom** : la boîte de dialogue **Liste d'ordonnancement** s'affiche.
3. Ajoutez tous les ordonnancements que vous voulez afficher sur la page.



4. Dans le volet **Propriétés**, personnalisez tous les paramètres.

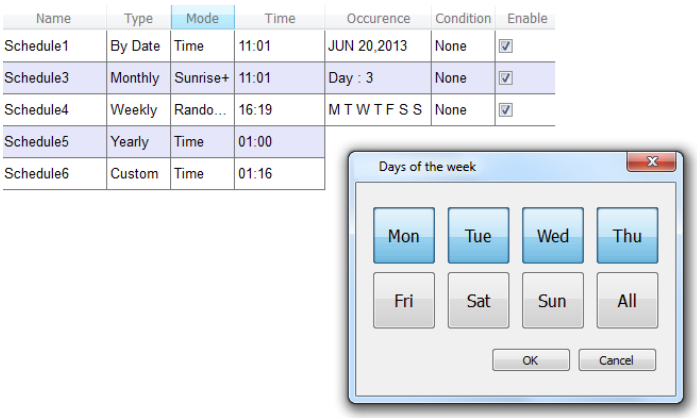
Paramètres des planificateurs

Paramètre	Description
Nom	Ordonnancement à afficher
Colonnes	Colonnes à afficher et leurs caractéristiques
Afficher en-tête	Affiche/masque les en-têtes des colonnes
Spéc heure	Heure à afficher au runtime
Texte	Police utilisée pour le texte
Affichage	Styles de tableaux

Ordonnancement des événements au runtime

Au runtime, vous pouvez modifier les paramètres d'ordonnancement suivants.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



Name	Type	Mode	Time	Occurence	Condition	Enable
Schedule1	By Date	Time	11:01	JUN 20,2013	None	☑
Schedule3	Monthly	Sunrise+	11:01	Day : 3	None	☑
Schedule4	Weekly	Rando...	16:19	M T W T F S S	None	☑
Schedule5	Yearly	Time	01:00			
Schedule6	Custom	Time	01:16			

Paramètre	Description
Occurrence	Informations sur le type de programme et l'heure d'exécution
Condition	Condition appliquée à l'exécution de l'action
Activer	Active/désactive l'exécution des actions programmées sans supprimer le calendrier.

Voir "Ordonnancement des tâches répétitives" Page 266 pour les détails sur les paramètres de planification.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

22 Conformité à 21 CFR partie 11

PB610 Panel Builder 600 inclut un ensemble de fonctions visant à répondre aux exigences spécifiées dans la FDA 21 CFR partie 11. Cette norme est destinée à fournir une solution pour le traitement sécurisé des documents électroniques et des signatures électroniques dans les applications industrielles.

Le tableau énumère toutes les exigences spécifiées par le règlement et indique les fonctions disponibles dans PB610 Panel Builder 600 sur la conformité.

La conformité à la FDA 21 CFR partie11 est facultative pendant le développement de l'application et le développeur de l'application est responsable de la configurer correctement.

Chapitre	Description	PB610 Panel Builder 600 niveau de conformité (v2.8)
11.10(a)	(a) Validation des systèmes pour assurer l'exactitude, la fiabilité, l'uniformité du rendement visé et la capacité de discerner les dossiers non valides ou modifiés.	Les rapports générés par PB610 Panel Builder 600 peuvent être signés à l'aide de certificats x.509. Un certificat incluant la clé publique, nécessaire pour vérifier la signature des rapports, sera exporté avec le rapport. Références : • "SaveEventArchive" Page 170 • "Planificateur" Page 265
11.10(b)	La capacité de produire des copies exactes et complètes des documents sous forme écrite et sous forme électronique qui se prêtent à l'inspection, à l'examen et à la reproduction par l'organisme. Les personnes doivent contacter l'agence s'il y a des questions concernant la capacité de l'agence à effectuer un tel examen et une telle copie des documents électroniques.	Le développeur d'application peut sélectionner les ressources (valeurs de processus, alarmes, etc.) dont les changements seront suivis jusqu'au journal d'audit. Chaque modification des ressources sélectionnées sera enregistrée avec le nom de l'opérateur qui effectue la modification. Les rapports de journal d'audit peuvent être exportés dans des fichiers.csv. Références : • "Activer/désactiver le journal d'audit" Page 298 • "Exporter le journal d'audit au format .csv" Page 303 • "SaveEventArchive" Page 170
11.10(c)	Protection des enregistrements pour permettre leur récupération précise et rapide tout au long de leur période de conservation.	Les applications peuvent être développées afin de générer des rapports signés vers la mémoire externe ou des dossiers en réseau à des intervalles prédéfinis (par exemple en fin de journée) ou lorsque la mémoire tampon circulaire est pleine. L'utilisateur est responsable de conserver ces rapports pendant

Chapitre	Description	PB610 Panel Builder 600 niveau de conformité (v2.8)
		la période de conservation. Références : • "SaveEventArchive" Page 170 • "Planificateur" Page 265
11.10(d)	Limiter l'accès au système aux personnes autorisées.	Le développeur d'application est responsable de la configuration correcte de l'application au sujet de la sécurité. Références : • "Gestion et mots de passe utilisateur" Page 283
11.10(e)	Utilisation de journaux d'audit sécurisés, générés par ordinateur et horodatés pour l'enregistrement indépendamment la date et l'heure des saisies et des actions de l'opérateur qui créent, modifient ou suppriment des enregistrements électroniques. Les modifications apportées aux enregistrements ne doivent pas masquer les informations précédemment enregistrées. Cette documentation sur les journaux d'audit est conserve pendant une période au moins aussi longue que celle requise pour les documents électroniques en question et sont disponibles aux fins d'examen et de copie par l'agence.	Les enregistrements des journaux d'audit sont stockés à l'aide d'une mémoire tampon circulaire (pour garantir que le périphérique ne sera pas à court de mémoire). Les journaux d'audit ne peuvent pas être modifiés par l'opérateur. Chaque enregistrement contient un numéro séquentiel visant à vérifier facilement la présence de tous les enregistrements. L'application peut être développée pour sauvegarder/exporter une copie des données à intervalles réguliers (par exemple en fin de chaque journée) ; l'opérateur est responsable de garder une copie des rapports dans un endroit sûr. Références : • "SaveEventArchive" Page 170 • "Planificateur" Page 265 • "Exporter le journal d'audit au format .csv" Page 303
11.10(f)	Utilisation de vérifications du système opérationnel visant à faire respecter l'enchaînement autorisé des étapes et des événements, s'il y a lieu.	Des macros ou JavaScript peuvent être utilisés pour configurer les séquences de commandes dans l'application.
11.10(g)	Utilisation de vérifications d'autorité pour garantir que seules les personnes autorisées peuvent utiliser le système, signer électroniquement un document, accéder au dispositif d'entrée ou de sortie du système informatique, modifier un document ou effectuer manuellement l'opération.	L'application IHM peut être configurée • pour n'être accessible qu'après que l'utilisateur s'est connecté avec son propre mot de passe. • Les objets peuvent être configurés pour être disponibles ou non disponibles en fonction de l'utilisateur qui s'est connecté au système.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Chapitre	Description	PB610 Panel Builder 600 niveau de conformité (v2.8)
		• Les ressources peuvent être configurées pour exiger une confirmation de mot de passe avant d'être modifiées. Références : • "Gestion et mots de passe utilisateur" Page 283 • "Signature électronique" Page 299
11.10(h)	Vérifications de périphériques (p. ex., terminaux) pour déterminer, le cas échéant, la validité de la source entrées de données ou d'instruction opérationnelle.	Les ressources peuvent être configurées pour n'être accessibles qu'à partir de groupes d'utilisateurs sélectionnés. La liste des adresses IP autorisées peut être configurée à partir des paramètres de gestion des utilisateurs. Références : • "Modifier des permissions d'accès" Page 286
11.10(i)	Détermination que les personnes qui développent, maintiennent ou utilisent des systèmes d'enregistrement électronique et de signature électronique ont reçu l'enseignement, la formation et ont l'expérience afin d'accomplir les tâches qui leur sont assignées.	Le développeur d'application est responsable de la définition et de l'attribution des droits d'utilisateur appropriés à chaque utilisateur ayant accès à l'interface homme-machine.
11.10(j)	L'établissement et le respect de politiques écrites qui tiennent les personnes responsables des mesures prises en vertu de leur signature électronique, afin de décourager la falsification d'enregistrements et de signatures.	Le développeur d'application est responsable de la mise en place des procédures appropriées.
11.10(k)	Utilisation de contrôles appropriés sur la documentation des systèmes, y compris : (1) Contrôles appropriés de la distribution, de l'accès et de l'utilisation de la documentation pour l'exploitation et la maintenance du système. (2) Révision et procédures de contrôle des changements afin de maintenir un journal d'audit qui documente le développement et la modification échelonnés dans le temps de la documentation des systèmes.	Le développeur d'application est responsable de la mise en place des procédures appropriées.
11.30	Les personnes qui utilisent des systèmes ouverts pour créer, modifier, maintenir ou transmettre des documents électroniques emploient des procédures et des contrôles conçus pour assurer l'authenticité, l'intégrité et, le cas échéant, la confidentialité des	PB610 Panel Builder 600 a été conçu pour fonctionner en systèmes fermés.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Chapitre	Description	PB610 Panel Builder 600 niveau de conformité (v2.8)
	documents électroniques depuis leur création jusqu'à leur réception. Ces procédures et contrôles doivent comprendre les procédures et contrôles mentionnés en section 11.10, selon le cas, ainsi que des mesures supplémentaires comme le chiffrement des documents et l'utilisation de normes de signature numérique appropriées pour assurer, au besoin dans les circonstances, l'authenticité, l'intégrité et la confidentialité des documents.	
11.50(a)	Les dossiers électroniques signés doivent contenir des renseignements accompagnant la signature qui indiquent clairement tous les éléments suivants : (1) Le nom du signataire, (2) la date et l'heure auxquelles la signature a été apposée et (3) La signification (comme l'examen, l'approbation, la responsabilité ou la qualité d'auteur) associée à la signature.	Tous les enregistrements seront ajoutés au journal d'audit avec l'horodatage et l'identifiant de l'utilisateur connecté. Références : • "Exporter le journal d'audit au format .csv" Page 303 • "Widget audit de tableau" Page 302
11.50(b)	Les éléments visés aux paragraphes (a)(1), (a)(2) et (a)(3) de la présente section sont soumis aux mêmes contrôles que les enregistrements électroniques et doivent être inclus dans toute forme lisible d'enregistrement électronique (en tant qu'affichage électronique ou document imprimé).	
11.70	Les signatures électroniques et les signatures manuscrites exécutées sur des enregistrements électroniques doivent être liées à leurs enregistrements électroniques respectifs afin de garantir que les signatures ne peuvent pas être supprimées, copiées ou transférées de toute autre façon dans le but de falsifier un enregistrement électronique par des moyens ordinaires.	Le développeur d'application est responsable d'éviter d'utiliser des macros qui permettent l'importation/exportation des mots de passe des utilisateurs.
11.100(a)	Chaque signature électronique doit être la propriété d'une seule personne et ne doit pas être réutilisée ou réaffectée à quelqu'un d'autre.	Le système garantira que deux utilisateurs avec le même identifiant ne peuvent pas être définis. Il est de la responsabilité de l'utilisateur d'éviter la suppression et la réaffectation du même nom d'utilisateur à un autre utilisateur.
11.100(b)	Avant qu'une organisation établisse, assigne, certifie ou sanctionne la signature électronique d'un individu, ou tout élément d'une telle signature électronique, l'organisation doit vérifier l'identité de l'individu.	Responsabilité de l'utilisateur.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Chapitre	Description	PB610 Panel Builder 600 niveau de conformité (v2.8)
11.100(c)	Les personnes qui utilisent des signatures électroniques doivent, avant ou au moment de es utiliser, certifier à l'agence que les signatures électroniques dans leur système, utilisées le 20 août 1997 ou après cette date, sont destinées à être l'équivalent tenu de par la loi des signatures manuscrites traditionnelles. (1) La certification doit être adressée sur papier et signée d'une signature manuscrite traditionnelle au Bureau des opérations régionales (HFC-100), 5600 Fishers Lane, Rockville, MD 20857, États-Unis. (2) Les personnes utilisant des signatures électroniques doivent, à la demande de l'agence, fournir une certification ou un témoignage supplémentaire attestant qu'une signature électronique spécifique est l'équivalent l'équivalent tenu de par la loi des signatures manuscrites traditionnelles.	Responsabilité de l'utilisateur.
11.200(a)	(a) Les signatures électroniques qui ne sont pas fondées sur des données biométriques doivent : (1) Utilisez au moins deux éléments d'identification distincts, comme un code d'identification et un mot de passe.	PB610 Panel Builder 600 Les fonctions de sécurité sont basées sur la combinaison Identifiant / Mot de passe.
	(i) Lorsqu'une personne exécute une série de signatures au cours d'une seule période continue d'accès contrôlé au système, la première signature doit être exécutée à l'aide de tous les composants de signature électronique ; les signatures ultérieures sont exécutées à l'aide d'au moins un composant de signature électronique qui n'est exécutable que par cette personne et conçu pour n'être utilisé que par elle. (ii) Lorsqu'une personne exécute une ou plusieurs signatures qui n'ont pas été effectuées au cours d'une seule période continue d'accès contrôlé au système, chaque signature doit être exécutée en utilisant tous les composants de signature électronique.	Les utilisateurs doivent entrer un nom et un mot de passe pour accéder au système. Les actions critiques peuvent être configurées de manière à exiger la saisie du mot de passe avant le démarrage de l'exécution. Références : • "Gestion et mots de passe utilisateur" Page 283 • "Signature électronique" Page 299
	(2) Utilisé uniquement par leurs propriétaires authentiques ; et (3) Administré et exécuté de manière à ce que toute tentative d'utilisation de la signature électronique d'une personne autre que son propriétaire	Chaque utilisateur est responsable de ne pas divulguer son propre mot de passe. Les mots de passe définis par l'administrateur pour le premier accès peuvent être redéfinis lors de la première utilisation.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

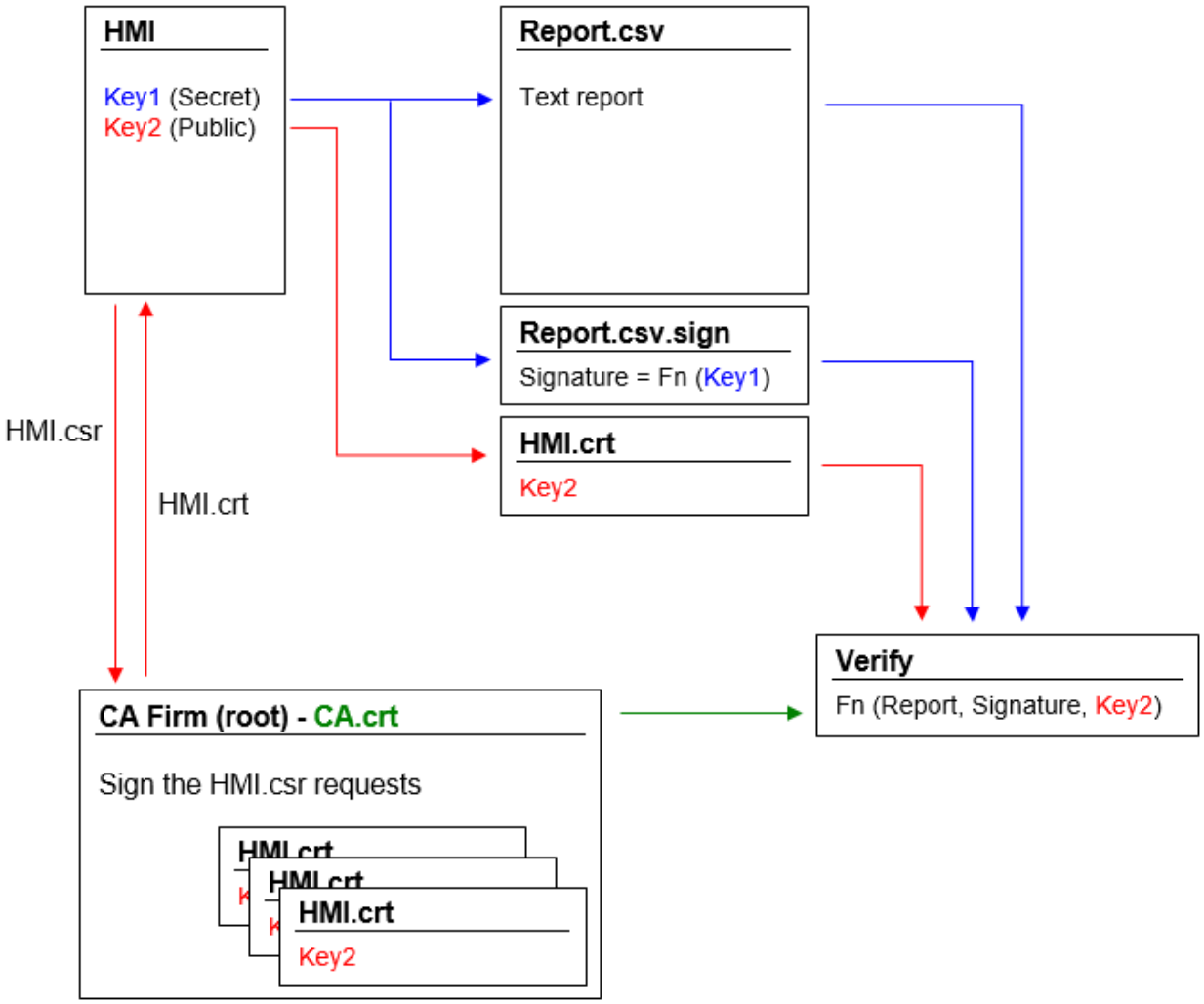
Chapitre	Description	PB610 Panel Builder 600 niveau de conformité (v2.8)
	authentique nécessite la collaboration de deux personnes ou plus.	Références : • "Configurer utilisateurs" Page 292
11.200(b)	Les signatures électroniques fondées sur des données biométriques sont conçues de manière à ce qu'elles ne puissent être utilisées que par leurs propriétaires authentiques	PB610 Panel Builder 600 ne prend pas en charge les données biométriques.
11.300(a)	Maintenir le caractère unique de chaque combinaison code d'identification et mot de passe, de sorte qu'il n'y a pas deux personnes ayant la même combinaison code d'identification et mot de passe.	Il n'est pas possible de définir deux utilisateurs avec le même ID d'utilisateur
11.300(b)	S'assurer que les codes d'identification et les mots de passe sont vérifiés, rappelés ou modifiés périodiquement (p. ex. pour couvrir des événements comme le vieillissement des mots de passe).	Le système peut être configuré pour forcer chaque utilisateur à définir un nouveau mot de passe différent après un certain nombre configurable de jours Références : • "Configurer utilisateurs" Page 292
11.300(c)	Suivre les procédures de gestion des pertes pour désautoriser électroniquement les jetons, cartes et autres dispositifs perdus, volés, manquants ou potentiellement compromis qui portent ou génèrent un code d'identification ou des informations sur les mots de passe, et pour émettre des remplacements temporaires ou permanents en utilisant des commandes rigoureuses et appropriées.	Les utilisateurs peuvent changer leur mot de passe à tout moment. L'administration peut redéfinir le mot de passe de chaque utilisateur et les forcer à le redéfinir lors de la première connexion. Références : • "Actions de gestion des utilisateurs" Page 179 • "Configurer utilisateurs" Page 292
11.300(d)	Utilisation de mesures de protection des transactions pour empêcher l'utilisation non autorisée de mots de passe et/ou de codes d'identification, ainsi que pour détecter et signaler immédiatement et de manière urgente toute tentative d'utilisation non autorisée à l'unité de sécurité du système et, le cas échéant, à la direction de l'organisation.	Les tentatives de connexion échouées sont enregistrées dans le journal d'audit.
11.300(e)	Effectuez des contrôles initiaux et périodiques des dispositifs, tels que les jetons ou les cartes, qui portent ou génèrent un code d'identification ou un mot de passe pour s'assurer qu'ils fonctionnent correctement et n'ont pas été modifiés de manière non autorisée.	Il incombe à l'utilisateur de s'assurer que les mesures appropriées sont prises.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Certificat x.509

Pour garantir l'authenticité des rapports générés par les périphériques IHM, HMI Runtime peut générer des rapports avec des fichiers signés pour vérifier l'authenticité et l'intégrité des rapports générés.

HMI Runtime utilise des clés de cryptographie asymétrique pour signer les fichiers et la norme x.509 pour gérer les certificats de clé publique. La photo montre l'architecture.



Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Flux de travail

1. Chaque périphérique IHM contient deux clés:
   - Key1 est la clé secrète utilisée pour signer les rapports générés par le périphérique IHM. Cette clé est stockée en toute sécurité à l'intérieur du périphérique IHM.
   - Key2 est la clé publique que n'importe qui peut utiliser pour vérifier l'authenticité des rapports signés par le périphérique IHM.
2. La macro SaveEventArchive peut être utilisé pour générer des rapports signés (voir "SaveEventArchive" Page 170 pour plus de détails)
3. Les utilisateurs peuvent utiliser la clé publique et le fichier signé pour vérifier qu'un rapport est authentique et n'a pas été modifié après sa génération.

Authenticité de la clé publique

Avant de vérifier un rapport (Report.csv) avec son fichier signé (Report.csv.sign), les utilisateurs doivent s'assurer de l'authenticité de la clé publique (HMI.crt).

La clé publique peut être signée par une autorité de certification (AC) qui garantit son authenticité. La clé publique peut être utilisé comme solution de rechange, directement à partir du périphérique IHM.

Les fichiers de certificat interne x.509

Vous pouvez générer un certificat auto-signé avec une fonction dédiée dans Panneau de configuration. Vous pouvez également exporter ou importer le fichier de certificat interne.

Les utilisateurs peuvent personnaliser les informations contenues dans le certificat. Ils peuvent conserver le certificat auto-signé ou récupérer la "Demande de certificat signé" pour la soumettre à une autorité de certification pour validation. Ensuite, le certificat signé par l'autorité de certification peut être importé à l'intérieur du périphérique IHM. (Voir la section Certificat x.509 dans la section "Paramètres du système" Page 477 à la page 1)

Vous pouvez importer des certificats avec ou sans clé privée. Lorsque le certificat contient la clé privée, la clé privée interne du périphérique IHM sera remplacée par la clé trouvée dans le certificat. Les certificats sont encodés en base64.

Vous pouvez importer à l'intérieur de chaque périphérique IHM le même fichier de certificat pour avoir un fichier de certificat public unique pour tous vos périphériques IHM.

Exemple de certificat avec des clés publiques et privées.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

ssl-certificate.crt

```
-----BEGIN CERTIFICATE-----
MIIDBDCCAewCCQDcBYW7PYwJsDANBgkqhkiG9w0BAQsFADBEMQswCQYDVQQGEwJJ
VDEPMA0GA1UEBwwGVmVyb25hMRMwEQYDVQQKDApUZXN0T2ZmaWNlMQ8wDQYDVQQD
DAZITUktMDQwHhcNMTcwNjI2MDgwOTQ1WhcNMTgwNjI2MDgwOTQ1WjBEMQswCQYD
VQQGEwJJVDEPMA0GA1UEBwwGVmVyb25hMRMwEQYDVQQKDApUZXN0T2ZmaWNlMQ8w
DQYDVQQDDAZITUktMDQwggEiMA0GCSqGSIb3DQEBAQUAA4IBDwAwggEKAoIBAQDc
Nlp2kswcbLh4IxS6eeCgQ4EAUHCRpaZ5YPfQ/un9/s0tejaa3Si3Pcqv3JqddJM8
mJEZaPF/+HhAEhtC+rv57TbgullUQJdoQpfoGChofpULforXZt2BfdWNx67plNoa
YM3ElaNtAKIW2o6S9HGEvlkf09XFLGkFgeMgC59+SejgguCNT0m99m6fNa5910I7
UDJFINkC3bxtONj+WiL/iEZYkHXacaN9q06fx+2NfmiSsXGPnmSys5mocqo89tMa
TjyeF7jYpDccCpJ9pY4xRjRpcIkDCM7PabVoG/ascSMUU6XPE2R0W4UJ6bPAygD6
QLKCCq0BUi6/eUj0pnanAgMBAAEwDQYJKoZIhvcNAQELBQADggEBAMLfIEXQOEjS
OpwVkzNxXmL/A6PLU5BKlhVYHb7ofb2Z37zN69vCn8ESglAFYK7QhkhJu3zAD+jH
fYBVKVdxfd3HS8EmcDWxpC6F2lfgqsSqepMRTbKbsSaO53a7JsXtwnHVNfG6EBZV
8tqSlGc4RwtJeVZJelUdmWSBD4Fc7asFeBCKqLrHJinz7buj3I4fLcyscTaMTBI9
fsE7poEpWvKc7NWtKYZglGG3AG6xONu3sEahcJ5k+UVdh/QQdAiCt3vG+JJ/owYU
sd30WIZ4pNzG/GUH9MbJyvI4ftA8IvEhGxHvi3xt7slJnvYQDaghOEDhdtGvilOr
nJZ2FZOBCEI=
-----END CERTIFICATE-----
-----BEGIN RSA PRIVATE KEY-----
MIIEpAIBAAKCAQEA3DZadpLMHGy4eCMUunngoEOBAFBwkaWmeWD30P7p/f7NLXo2
mt0otz3Kr9yanXSTPJiRGWjxf/h4QBIbQvq7+e024LtZVECXaEKX6BgoaH6VC36K
l2bdgX3Vjceu6dTaGmDNxJWjbQCiFtqOkvRxhL5ZH9PVxSxpBYHjIAuffkno4ILg
jU9JvfZunzWufZdCO1AyRSDZAt28bTjY/loi/4hGWJB12nGjfatOn8ftjX5okrFx
j55ksrOZqHKqPPbTGk48nhe42KQ3HAqSfaWOMUY0aXCJAwjOz2mlaBv2rHEjFFOl
zxNkdFuFCemzwMoA+kCyggqtAVIuv3lI9KZ2pwIDAQABAoIBAGnamsuqrwDu5hGh
02H8GhUPvd/3ytTISujHyvgkwTf+FoTI3Zy9uMe0pUy5/3y2v9v9/qm3P3djafJq
gb5Fprxx4dJPXJZaYi2U7U585lesmVqoHneCk/GeGlyH4zWlwo2xgNgBkkhgaIoR
zz0m0bachVz+SCD6wxUJpbMOw0FBw54oPL0XS/gD+76S9ET7xmqZAS5xV/w8Khht
PtjPfT58GKhqVIC9cMrrBrkuGQPrNrDaJMPsQDxrFp7POQm4+GivrUJ0FA9Vtx46
C5QhXqVps/BODo3mjeOcj2b/FqsvG7WCc5PWOAcCqStmDxl+DQZOIVFSTrE4kdpg
mNn/80kCgYEA88Xfmqg0ta83lpe9b6U0BaLvvslgxgXmCmkyvK7Ru+iKyPUMzxB+
BjGWeeiZuigmIhXfFu3eBs5xOgDrUxf9j55sJAFamljG4LTyun378RnOdA87ff1q
rpF4oPKVfTrfXXzZkeIg0eX2tD6Lsn3+MJwYqpefovxmyJA3kPgcGv0CgYEA50H0
HQififZZ2nApgPf/jJpU7hBLC45cSXvE2MX2I3rd3ptGwzKRo/lZkslbvQutqRln
slyEF+c9LCz6g7FYhJoewChLqCVfeZ9GxBzHeJloxZwmxDXi8L4vmEDphwlcV8b3
ExHqUlMGuINHGelPIRlLKeEsbTQU+OVHuNv443MCgYEA7rMKYhllC6bYCsjowSMG
TqKembX84cqyl+zstp+EVbi99Usm0Lc4f/4cd6EQrplTwbqi6YPgDdAmRQLTalkp
e3FIOPVub4aQr0XgDEcC5bI8W57yxUrZJLjjYs5HHQoB4Dw5m0TOmFnS+enoxs3i
kly3Nowjz+fRCYFWN8kLVE0CgYEA43CLLK7ZcW9XKa2cNBo0PE1g8A4YMJJfk2nl
zKjNjlF9ujyO2NV4RYOsI+RSsFe3ARdJcS6xP20OTc8ixrh57VhCnAxFdGblQpFy
oNgJGkf9zjPoMJsqykjSOHTG+CctqaqmPxwkkLScbIW4PPSn/U6KDPNHpVNOuQeO
hXHak58CgYBLW1719vgYhUiSWc9Gd3mCSxpAb6y8RcyTgqF76K8v4MalLPqFkEtD
0BaFtlA+PtMLk2ODTRH4XU18oc9eV+7VDFkPJ8T0A2VwjzjMgNAd+vKlm4nOEBTt
UhegY0k8yLxS1ZvuYiVnHvKBIoF/G2ckwrxjO9KVE+SA45Ex0Px5qQ==
-----END RSA PRIVATE KEY-----
```

PB610 PC Runtime

Lorsque vous utilisez PB610 PC Runtime les fichiers de certificats se trouvent dans le dossier : %AppData%\ABB\<Version>\server\config\ssl-certificate

Fichiers batch pour gérer les fichiers de certificat à l'aide de l'outil public OpenSSL-Win32 public

Voici un exemple de gestion des fichiers de certificat à l'aide d'une bibliothèque publique OpenSSL-Win32

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Référence: https://www.openssl.org/

Création de certificats auto-signés

```
@echo off set OpenSSL="C:\Program Files (x86)\OpenSSL-Win32\bin\openssl.exe" set
CertificateName=HMIDeviceID rem Generate an RSA key %OpenSSL% genrsa -out ssl-
certificate.key 2048 rem Creating Certificate Signing Requests %OpenSSL% req -new -key
ssl-certificate.key -out ssl-certificate.csr -subj "/C=IT/L=Verona/O=CompanyName/OU=RD
Team/CN=%CertificateName%" rem Creating Certificate %OpenSSL% x509 -req -days 3650 -in
ssl-certificate.csr -signkey ssl-certificate.key -out ssl-certificate.crt pause
```

Vérifier un rapport signé

```
@echo off set OpenSSL="C:\Program Files (x86)\OpenSSL-Win32\bin\openssl.exe" set
FileToCheck=Report.csv rem Extract public key from the certificate %OpenSSL% x509 -in
HMI.crt -pubkey -noout > publicKey.pem rem Verify Signature %OpenSSL% dgst -sha256 -
verify publicKey.pem -signature %FileToCheck%.sign %FileToCheck% rem Remove public key
del publicKey.pem pause
```

SaveEventArchive (see "SaveEventArchive" Page 170) génère des rapports signés en utilisant l'algorithme « sha256 ».
Voir Propriétés du projet en "Projet" Page 76 pour les autres algorithmes disponibles.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

23 Gestion et mots de passe utilisateur

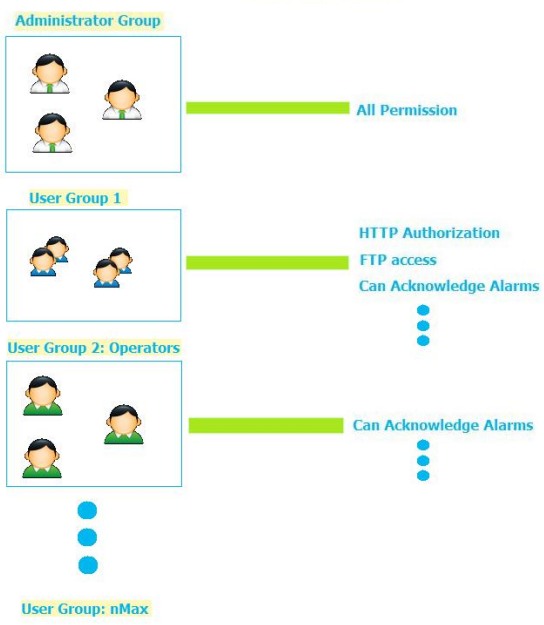
Vous pouvez restreindre l'accès à divers widgets et opérations en configurant les utilisateurs, les groupes d'utilisateurs et en attribuant des autorisations spécifiques à chaque groupe.

Chaque utilisateur doit être membre d'un seul groupe. Chaque groupe dispose d'autorisations et de permissions spécifiques.

Les autorisations et les permissions sont divisées en deux catégories :

- Permissions de widget : masqué, lecture seule, accès total
- Permissions d’action : autorisé ou non autorisé.

En organisant les permissions et les groupes, vous pouvez définir les options de sécurité d’un projet.



Activer/désactiver la gestion de sécurité **285**

Configurer groupes et autorisations **285**

Modifier des permissions d’accès **286**

Assigner des permissions par widget à partir du mode Page **291**

Configurer utilisateurs **292**

Utilisateur par défaut **294**

Gestion des utilisateurs au runtime **294**

Forcer la connexion à distance **294**

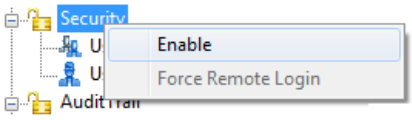
Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Activer/désactiver la gestion de sécurité

*Chemin : **VueProjet**> faites un clic droit sur **Sécurité**> **Activer***

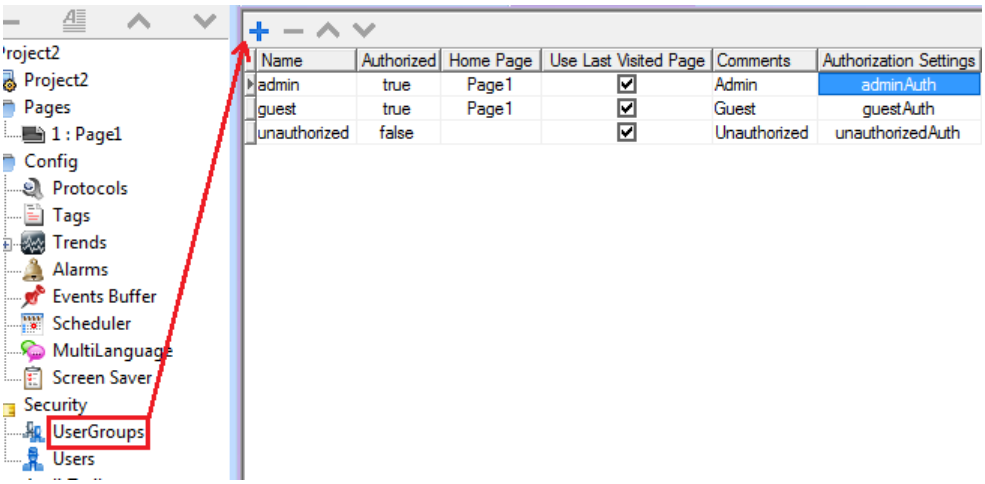
Le symbole du cadenas indique si la fonction est activée ou désactivée.



Important : les paramètres de sécurité fonctionnent uniquement si la fonction Sécurité est activée.

Configurer groupes et autorisations

*Chemin : **VueProjet**> **Sécurité** > double-cliquez sur **GroupesUtilisateurs***



Trois groupes prédéfinis sont disponibles par défaut (**administrateur**, **invité** et **non autorisé**) : ils ne peuvent pas être effacés ou renommés. Cependant, vous pouvez modifier les autorisations et les autres paramètres.

Ajouter un groupe d'utilisateurs

Cliquez sur **+** pour ajouter un groupe d'utilisateurs.

Paramètre	Description
Nom	Nom du groupe d'utilisateurs
Autorisé	Autorisation accordée
Page d'accueil	Page affichée lorsque les utilisateurs appartenant à ce groupe se connectent
Utiliser la dernière page visitée	Lorsque ce paramètre est sélectionné, la dernière page affichée par l'utilisateur précédent s'affiche lorsque les utilisateurs appartenant à ce groupe se connectent

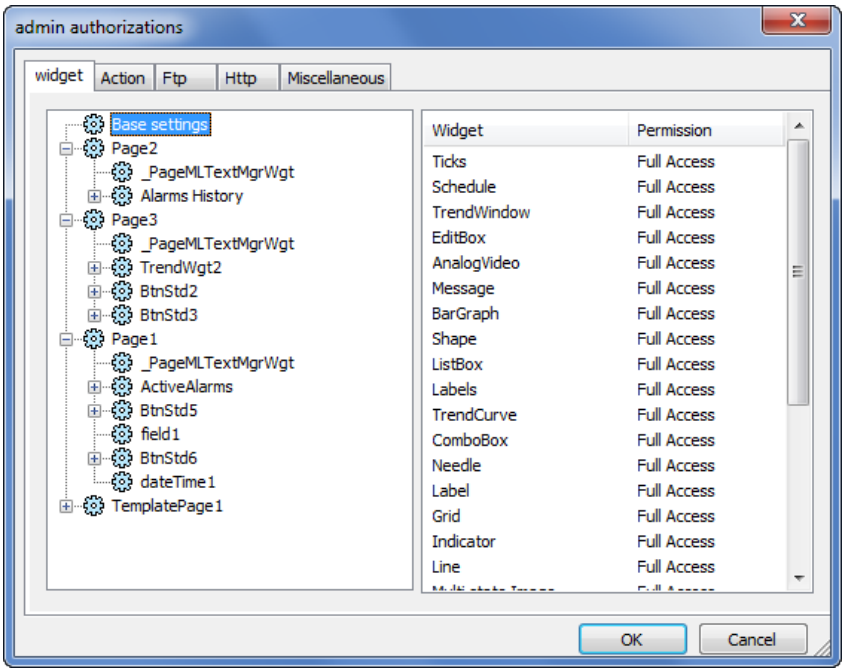
Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Paramètre	Description
Commentaires	Tout commentaire ou toute description pour le groupe
Paramètres des autorisations	Ouvre la boîte de dialogue Autorisation Administrateur pour définir les permissions d'accès. Voir "Modifier des permissions d’accès" bas pour plus de détails.

Modifier des permissions d’accès

Chemin : ***VueProjet*** *>* ***Sécurité****> double-cliquez sur* ***GroupesUtilisateurs*** *>* *colonne* ***Paramètres des autorisations***

Cliquez sur le bouton : une boîte de dialogue apparaît avec une liste de widgets et d'actions. Vous pouvez modifier les permissions d'accès pour chaque élément de la liste.



Permissions d'accès aux widgets

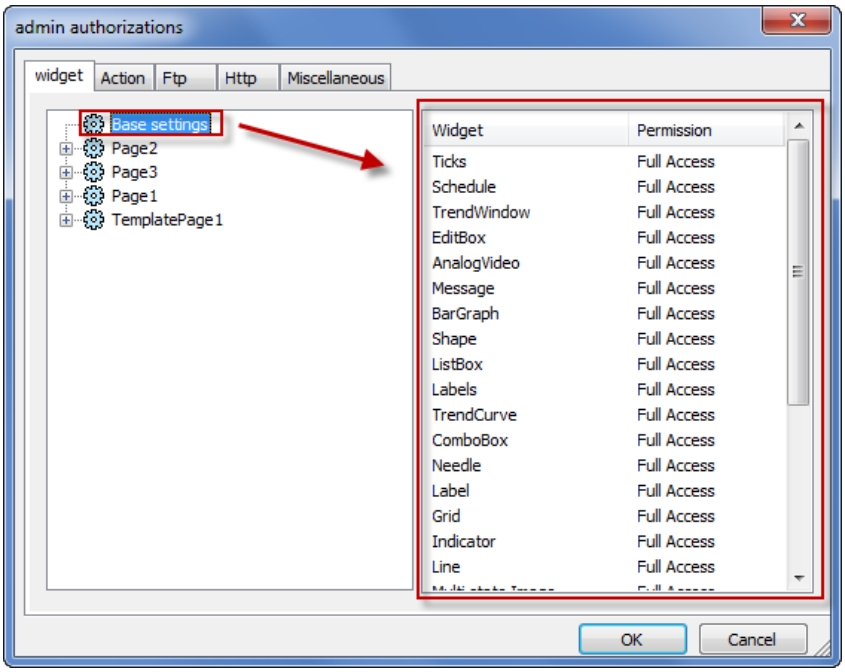
Dans l'onglet **Widget**, vous pouvez définir les options d'accès au widget au niveau du projet, de la page ou du widget, ce pour l'ensemble des widgets utilisés dans le projet. Les permissions des niveaux inférieurs (par exemple, au niveau du widget) écrasent celles des niveaux supérieurs (c'est à dire, les niveaux de page et de projet).

Utilisez les **Paramètres de base** pour définir les permissions par défaut au niveau du projet.

Les paramètres possible sont :

- **Accès complet** pour autoriser l'accès au widget en mode lecture/écriture
- **Lecture seule** pour n'autoriser l'accès au widget qu'en mode lecture
- **Masquer** pour cacher le widget pour le groupe sélectionné

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



Modifier une permission d'accès à un widget

Pour modifier une permission d'accès à un widget spécifique dans une page du projet, naviguez jusqu'à ce widget à l'intérieur de sa page sur le volet droit et personnalisez ses options d'accès. À défaut, tous les widgets comportent les permissions définies au niveau du projet ou de la page.

Par exemple, si la permission de page pour un widget a été accordée en mode **Lecture seule** au niveau du projet, l'ensemble des widgets de même type comporteront une permission **Lecture seule**. Lorsque vous sélectionnez un widget à l'intérieur d'une page à partir de l'arborescence, la permission est configurée sur **Utiliser les paramètres de base**. Vous ne pouvez modifier ce paramètre et les permissions d'accès que pour ce widget et dans cette page.

Priorité d'accès

Les permissions d'accès aux widgets suivent les priorités suivantes :

Niveau de permission	Priorité
Niveau du projet - Paramètres de base	Faible
Niveau page	Moyen
Niveau widget	Élevé

Cela vous permet de définir des exceptions pour une action ou un widget directement à partir de la page affichée.

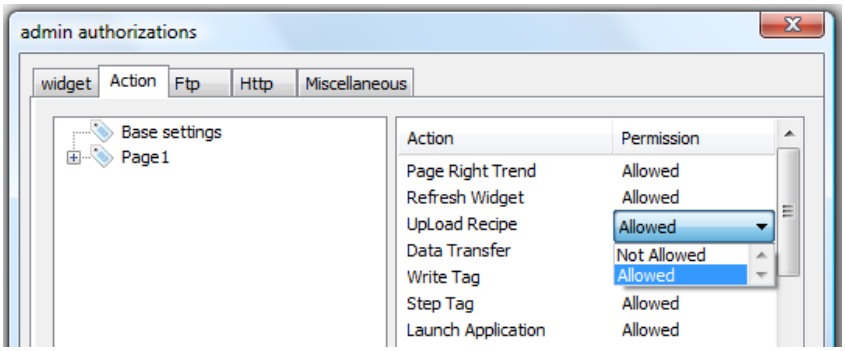
Par exemple, si vous accordez des permissions Lecture Seule pour un widget au niveau du projet et un Accès Complet au niveau de la page, les paramètres du niveau page prévaudront.

Les permissions d'accès peuvent être modifiées directement à partir de la page de projet. Voir "Assigner des permissions par widget à partir du mode Page" Page 291 pour les détails.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Permissions pour les actions

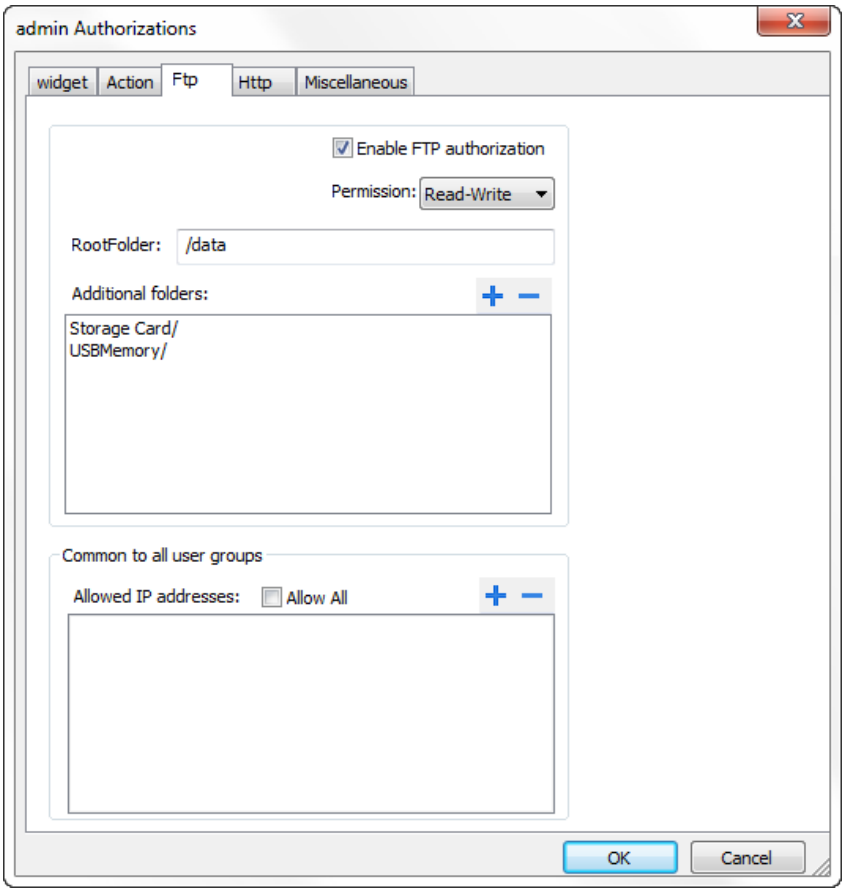
Dans Onglet **Action**, vous pouvez définir les autorisations concernant les actions au niveau du projet, de la page ou du widget. Les actions peuvent être **Autorisées** ou **Non Autorisées**.



Les permissions pour les actions peuvent être modifiées directement à partir de la page de projet. Voir "Assigner des permissions par widget à partir du mode Page" Page 291 pour les détails.

Autorisations FTP

Dans l'onglet **Ftp** vous pouvez définir des autorisations spécifiques pour le serveur FTP.



Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

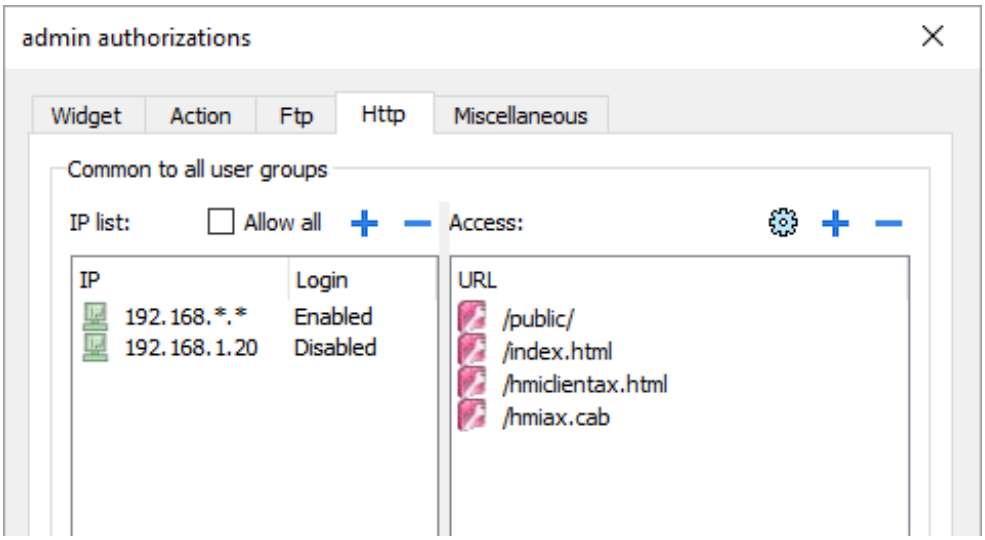
Élément	Description
Autoriser Accès FTP	Active la fonction FTP pour le groupe spécifique
Permission	Type de permission : • **Lecture seule** • **Lecture-écriture**
Dossier racine	Dossier à utiliser comme racine pour l'accès FTP. C'est un chemin relatif.
Dossier supplémentaire	Dossiers supplémentaires à utiliser comme racine pour l'accès FTP (par exemple, sur une clé USB ou une carte SD)
Adresses IP autorisées	Liste des adresses IP à partir desquelles la connexion FTP peut être acceptée. **Important : le paramètre est commun à tous les groupes d'utilisateurs.**

Autorisations HTTP

Dans l'onglet **HTTP**, vous définissez des restrictions à l'accès HTTP au serveur web intégré dans HMI Runtime.

Les métacaractères peuvent être utilisés pour identifier une plage d'adresses IP.

Par exemple, les deux règles suivantes définissent que l'unité de périphérique IHM ne peut être accessible par toutes les adresses IP 192.168.*.* sur votre réseau local dans lequel seule l'adresse IP 192.168.1.20 peut accéder au périphérique sans entrer un nom de connexion.



Élément	Description
Liste IP	Adresses IP autorisées à accéder au serveur HTTP. Toutes par défaut.
Ouverture de session	Lorsque désactivé, le nom d'utilisateur et mot de passe ne sont pas nécessaires.
Limites d'accès	Liste des ressources pour lesquelles l'accès est limité

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

L'effet de ces paramètres dépend de si l'option **Imposer l'identification à distance** a été sélectionnée. Voir "Forcer la connexion à distance" Page 294 pour plus de détails.

Forcer connexion à distance	Accès par défaut à l'espace de travail	Limites d'accès
-	Complet	-
Désactiver	Complet	Peut être utilisé pour bloquer l'accès à certains fichiers/dossiers ou pour demander l'autorisation
Activer	Pas d'accès	Peut être utilisé pour ouvrir l'accès à des fichiers/dossiers

Important : le paramètre est commun à tous les groupes d'utilisateurs.

Ajout d'une configuration HTTP

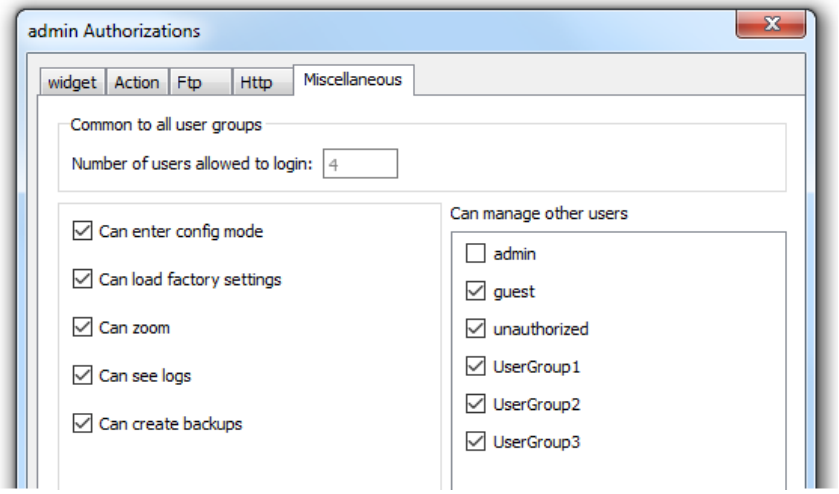
Pour ajouter et configurer un nouvel accès, cliquez sur **+** : la boîte de dialogue **Limites d'accès** s'affiche.

Pour restaurer la configuration par défaut, cliquez sur l'icône **Configurer les limites d'accès par défaut**. La configuration par défaut permet d'accéder aux éléments suivants :

- Dossier PUBLIC et Index.html

Paramètres divers

Dans l'onglet **Divers**, vous pouvez définir divers paramètres d'autorisation.



Remarque : certains de ces paramètres sont spécifiques à un groupe, d'autres sont communs pour tous les groupes.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

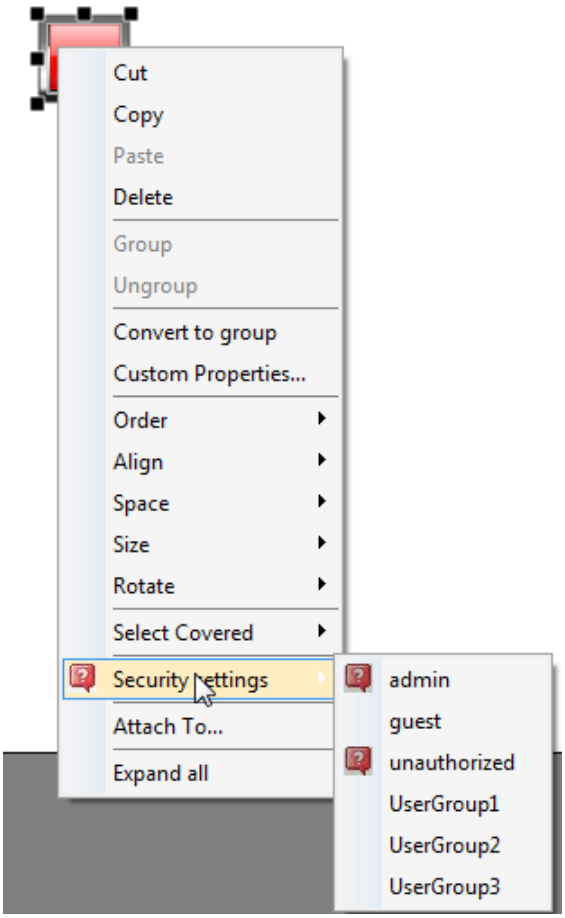
Option	Description
Passage au mode de configuration possible	Permet de passer du runtime au mode configuration. Normalement utilisé pour la maintenance.
Gestion des autres utilisateurs possible	Attribue des privilèges de super-utilisateur au runtime pour gérer les groupes sélectionnés. Permet d'ajouter, supprimer et modifier les permissions des utilisateurs.
Chargement des paramètres usine possible	Restaure les paramètres d'usine.
Zoom autorisé	Permet de zoomer ou de dézoomer dans le menu contextuel au runtime
Affichage des journaux possible	Permet à l'utilisateur de consulter les journaux au runtime
Création de sauvegardes autorisée	Permet à l'utilisateur de sauvegarder un projet.
Nombre autorisé d'utilisateurs connectés	Nombre maximum d'utilisateurs pouvant être connectés au runtime IHM au même moment. Défaut = 3

Assigner des permissions par widget à partir du mode Page

Vous pouvez assigner différents niveaux de sécurité pour un seul widget à différents groupes d'utilisateurs, directement à partir des pages du projet.

1. Faites un clic droit sur le widget et sélectionnez **Paramètres de sécurité.**
2. Choisissez le groupe : la boîte de dialogue d'autorisation pour le groupe s'affiche.
3. Définissez les propriétés de sécurité pour accéder au widget.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

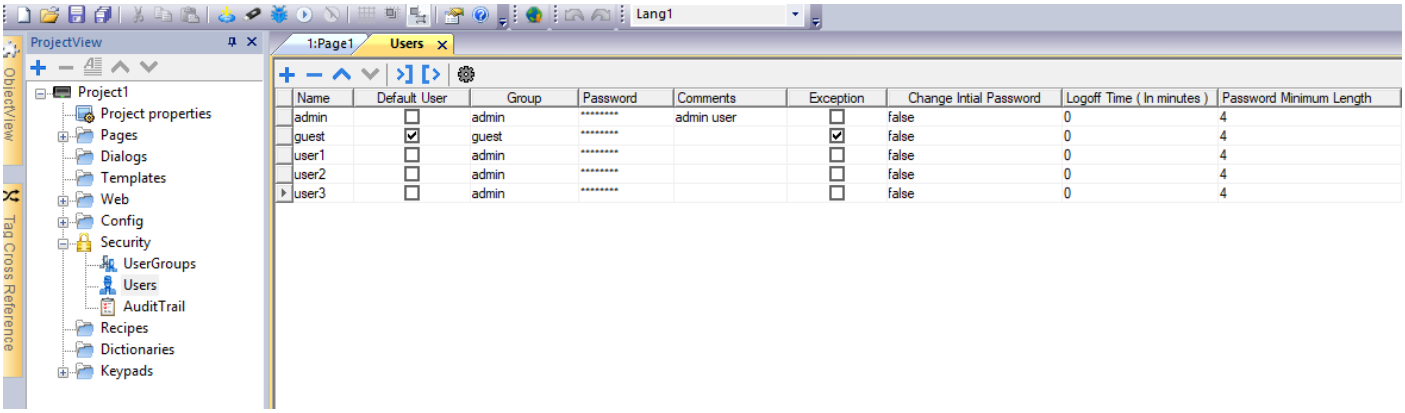


Voir "Modifier des permissions d'accès" Page 286 pour plus de détails.

Configurer utilisateurs

Chemin : ***VueProjet*** *>* ***Sécurité*** *> double-cliquez sur* ***Utilisateurs***

Dans l'éditeur Utilisateurs, cliquez sur **+** pour ajouter un utilisateur : une ligne est ajoutée au tableau.



Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

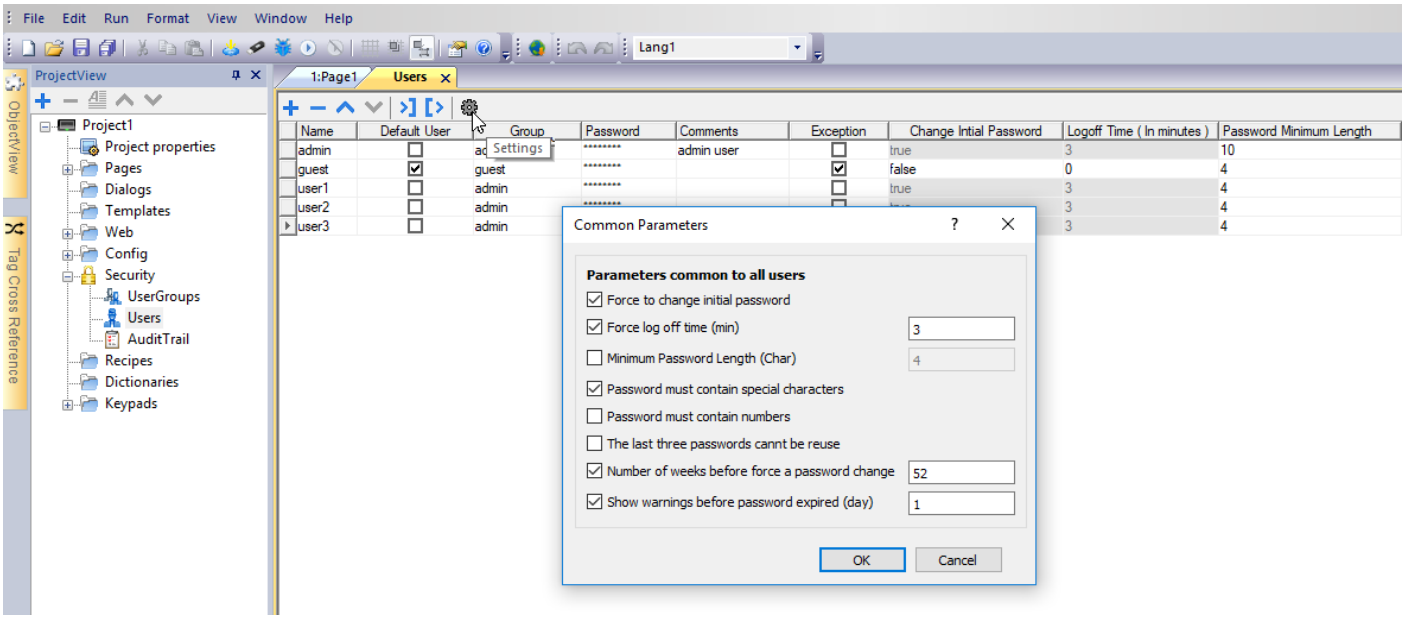
Paramètre	Description
Nom	Identifiant
Utilisateur par défaut	Cet utilisateur est automatiquement connecté au démarrage du système ou après qu'un autre utilisateur s'est déconnecté. Un seul Utilisateur par défaut peut être défini
Groupe	Groupe utilisateur
Mot de passe	Mot de passe de l'utilisateur. Notez que pour des raisons de sécurité, le mot de passe ne sera jamais affiché **Les mots de passe sont cryptés et ne peuvent pas être récupérés, même par des techniciens spécialisés**
Commentaires	Description détaillée de l'utilisateur
Exception	Permet de modifier les valeurs forcées à partir des paramètres de réglages utilisateur
Changer mot de passe initial	Cet utilisateur doit modifier son mot de passe lors de sa première connexion
Temps de déconnexion (en minutes)	Minutes d'inactivité après lesquelles l'utilisateur est déconnecté. Définir sur 0 pour désactiver
Nombre de caractères minimum du mot de passe	Nombre de caractères minimum du mot de passe
Doit contenir des caractères spéciaux	Le mot de passe doit contenir au moins un caractère spécial
Doit contenir des chiffres	Le mot de passe doit contenir au moins un caractère numérique
Le mot de passe ne peut pas être réutilisé	Le nouveau mot de passe doit être différent des 3 derniers mots de passe utilisés
Ancienneté du mot de passe (semaines)	Nombre de semaines avant de forcer un changement de mot de passe (1/52 semaines)
Avertissement (jours)	Afficher un message d'avertissement avant l'expiration du mot de passe (1/30 jours)

Paramètres des utilisateurs

À partir de la commande Paramètres, il est possible de définir des valeurs de paramètres qui seront communes à tous les utilisateurs.

Les utilisateurs avec le drapeau Exception coché ne sont pas obligés d'utiliser les paramètres communs.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



Utilisateur par défaut

Vous ne pouvez définir qu'un utilisateur par défaut dans un projet. Il s'agit de l'utilisateur connecté automatiquement au démarrage du système et lorsque l'utilisateur actuellement connecté se déconnecte ou est déconnecté après un délai d'attente.

Pour se connecter à HMI Runtime avec un utilisateur différent, utilisez une de ces actions :

- **ChangerUtilisateur**
- **Déconnexion**

Voir "Actions de gestion des utilisateurs" Page 179pour plus de détails.

Gestion des utilisateurs au runtime

L'utilisateur par défaut est, le cas échéant, automatiquement connecté au démarrage du runtime IHM. Si aucun utilisateur par défaut n'a été configuré, le système demande un identifiant et un mot de passe. Voir "Actions de gestion des utilisateurs" Page 179 pour plus de détails sur les actions pouvant êtres exécutées sur les utilisateurs.

Supprimer les données d'utilisateur

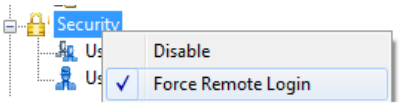
Toutes les informations d'utilisateur modifiées au runtime sont stockées dans des fichiers dédiés. Pour supprimer ces fichiers dynamiques et l'ensemble des modifications apportées à la configuration de l'utilisateur au runtime, vous pouvez :

- sur le runtime IHM : exécuter l'action SuppressionFichiersDynamiquesUM
- avec PB610 Panel Builder 600: sélectionner **Suppression des fichiers dynamiques** dans la boîte de dialogue de téléchargement.

Forcer la connexion à distance

*Chemin : **VueProjet**> cliquez avec le bouton droit sur **Sécurité**> **Forcer connexion à distance***

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



Sélectionnez cette option pour forcer l'utilisateur à se connecter lorsqu'il utilise l'accès à distance viaHMI Client. Si vous ne la sélectionnez pas, l'accès à distance utilisera le même niveau de protection que l'accès local.

Important : cette fonction ne fonctionne que quand la gestion utilisateur est désactivée.

Conseil : utilisez cette option lorsque vous avez un utilisateur par défaut mais que vous souhaitez sécuriser l'accès à distance.

Voir "Activer/désactiver la gestion de sécurité" Page 285pour plus de détails.

Les seuls fichiers/dossiers encore accessibles lorsque ce drapeau est activé sont :

- Dossier PUBLIC et Index.html.

Voir "Modifier des permissions d’accès" Page 286 pour plus de détails sur les limites d'accès HTTP.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

24 Journaux d’audit

La liste d'audit est une séquence chronologique des enregistrements d'audit. Chaque enregistrement contient des informations sur les actions exécutées et sur l'utilisateur qui les a réalisées.

Cette fonction fournit le suivi du processus, l'identification de l'utilisateur et l'horodatage des événements.

Activer/désactiver le journal d’audit **298**

Signature électronique **299**

Widget audit de tableau **302**

Exporter le journal d’audit au format .csv **303**

Configurer les événements d'audit **304**

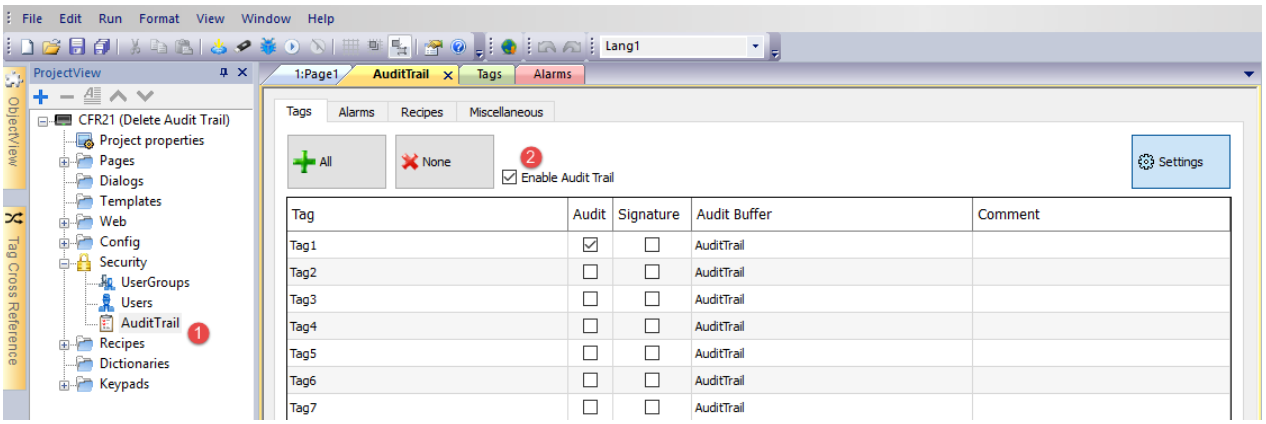
Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Activer/désactiver le journal d'audit

Chemin : ***VueProjet*** *>* ***Sécurité*** *> double-cliquez sur* ***Journal d'audit***

L'enregistrement du journal d'audit peut être activé à partir de la case à cocher « Activer le journal d'audit ».

Lorsque activée, toutes les modifications apportées aux ressources sélectionnées seront enregistrées dans la mémoire tampon d'audit avec l'horodatage, le nom de l'utilisateur qui a effectué l'opération et des informations supplémentaires concernant la ressource modifiée (par exemple, nouvelle valeur et valeur précédente des variables).



À partir des onglets principaux (Variables, Alarmes, Recettes et Divers) de l'éditeur du journal d'audit, vous pouvez basculer entre les vues de liste des ressources disponibles.

Paramètre	Description
Audit	Active le suivi de la ressource sélectionnée
Signature	Le mot de passe utilisateur est requis avant d'autoriser la modification de la ressource par l'utilisateur (voir "Signature électronique" Page opposée pour plus d'informations).
Mémoire tampon d'audit	Mémoire tampon d'événements à partir de laquelle la liste des événements est récupérée (voir "Configurer les événements d'audit" Page 304).
Commentaire	Espace pour les commentaires des développeurs

Variables

- Gardez un suivi du moment où la valeur d'une variable est modifiée.

Alarmes

- Garde un suivi du moment où l'utilisateur accuse réception ou réinitialise un événement d'alarme.

Recettes

- Garde un suivi du moment où les utilisateurs téléchargent ou téléversent des recettes.

Ressources diverses

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

- Détails de connexion de l'utilisateur
  Garde un suivi du moment de connexion, déconnexion ou changement de mot de passe.
- Actions de gestion des utilisateurs
  Garde un suivi du moment où un utilisateur est ajouté, supprimé ou lorsque les propriétés de l'utilisateur sont modifiées.
- Actions système
  Garde un suivi des actions du système (Redémarrage du périphérique IHM, Mise sous tension, Sauvegarde, Mise à jour, Téléchargement, entrée dans Paramètres du système, ouverture du Gestionnaire de projets).
- Actions FTP
  Garde un suivi de ftpGET, ftpPUT, OpenTextEditor, SaveTextEditor
- Actions mémoire tampon
  Garde le suivi des actions de vidage et de suppression des actions pour les tampons d'alarmes, d'audit ou de tendances.

Macro LogMessage

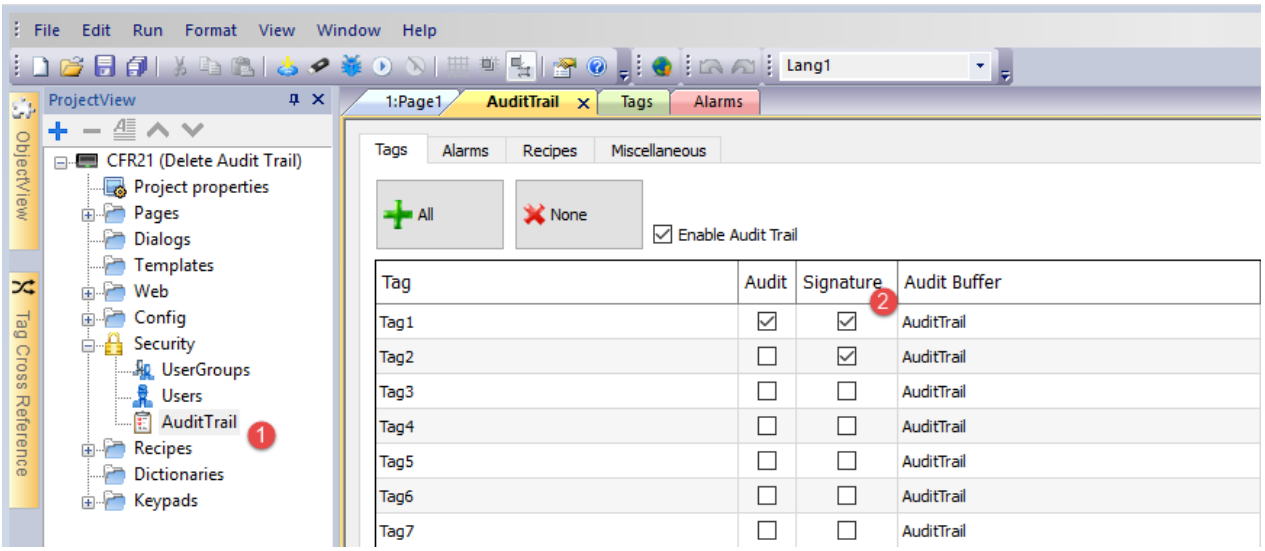
En outre, la macro LogMessage donne la possibilité de définir des événements supplémentaires pour la journalisation dans la mémoire tampon du journal d'audit.

Voir "LogMessage" Page 172 pour plus de détails.

Signature électronique

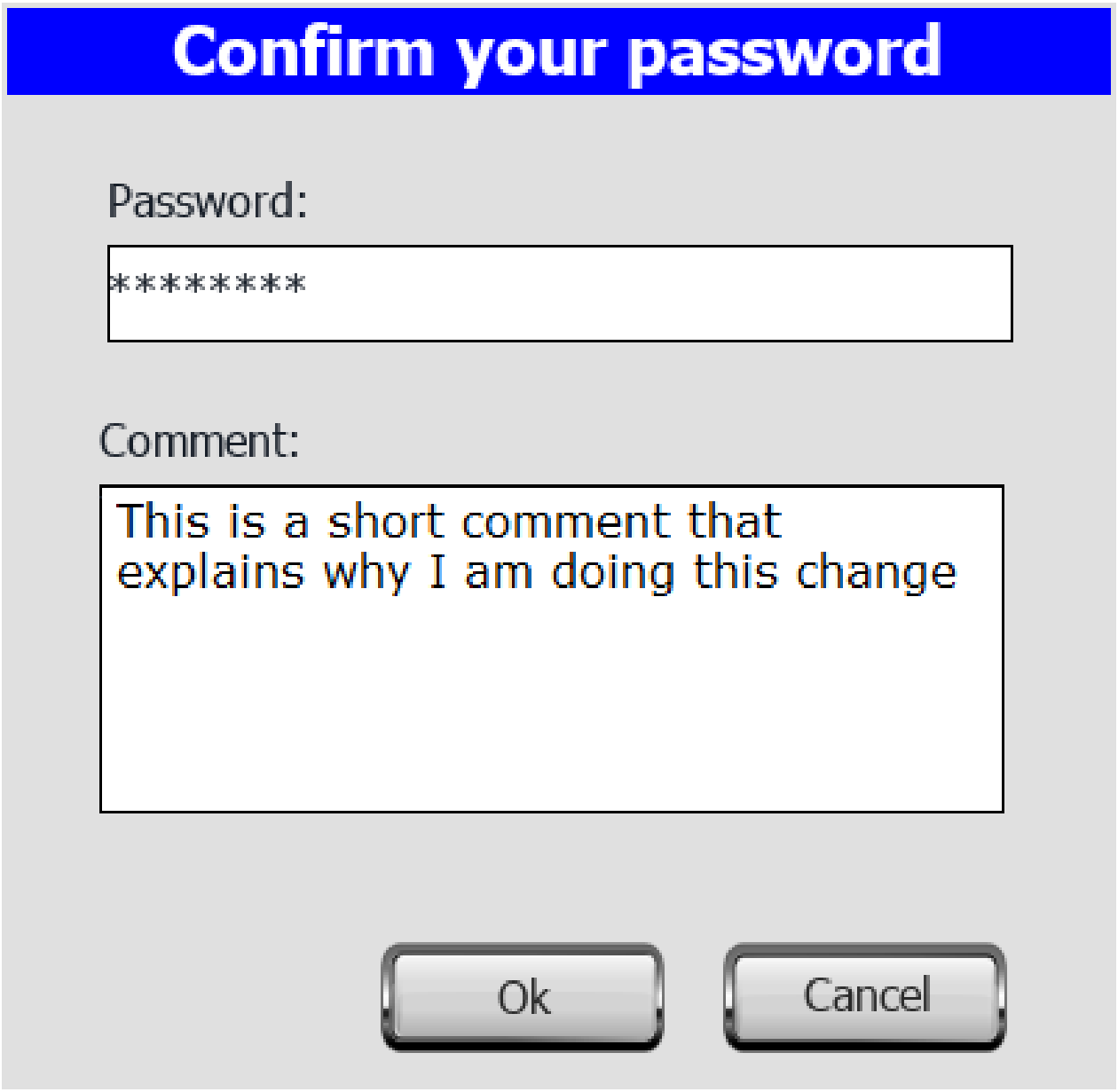
Pour chaque ressource listée dans l'éditeur du Journal d'Audit, il est possible de configurer le HMI Runtime de façon à exiger la confirmation du mot de passe avant de le modifier. Si l'enregistreur du journal d'audit est activé, l'utilisateur a la possibilité d'ajouter un commentaire qui sera enregistré dans l'enregistreur de suivi.

*Chemin : **VueProjet**> **Sécurité** > double-cliquez sur **Journal d'audit***

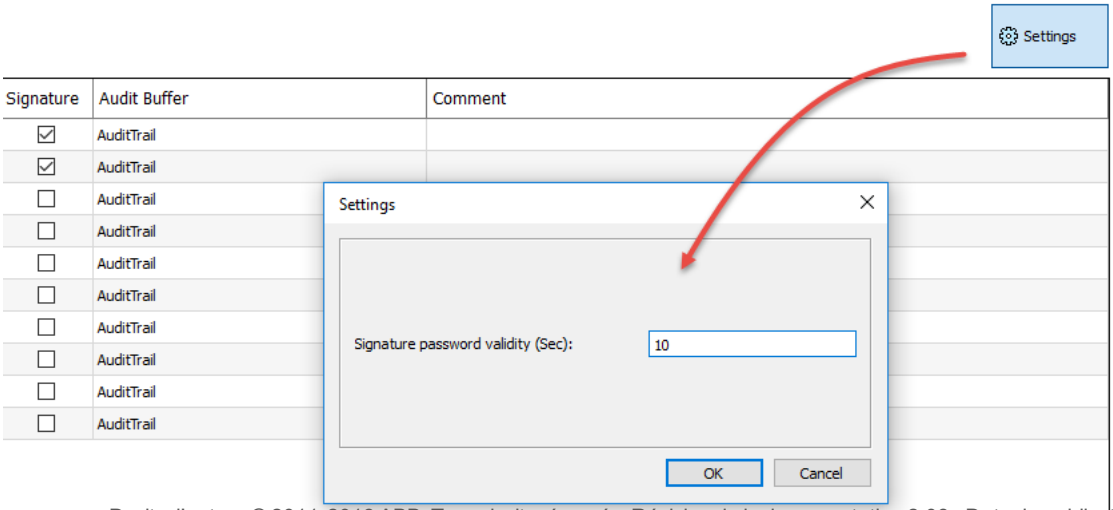


Le mot de passe utilisateur est requis avant de permettre à l'utilisateur de modifier la ressource.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



Le mot de passe introduit ne sera plus nécessaire pour les commandes lancées dans les 10 secondes suivantes.
La durée de validité peut être modifiée dans la boîte de dialogue Paramètres.



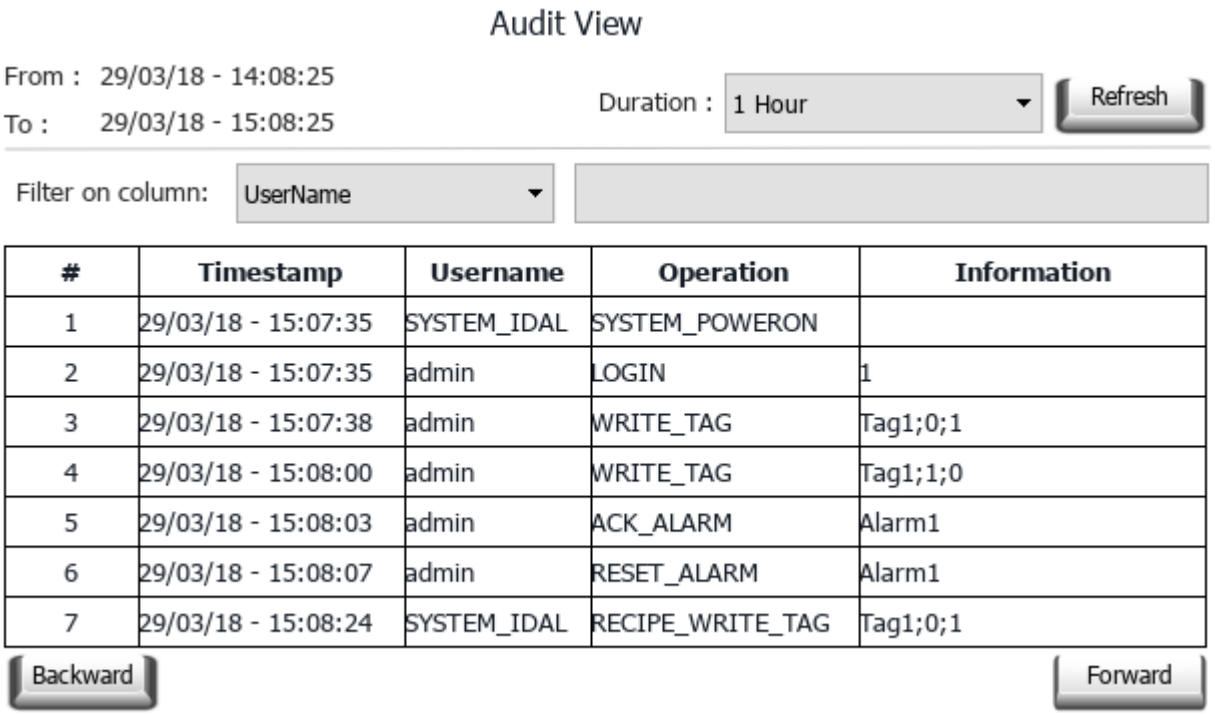
Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Widget audit de tableau

Chemin : **Galerie des widgets**> **Basique**> **Audit de tableaux**

Affiche le contenu du journal d'audit dans un widget.



Boutons :

- REFRESH
  Récupère les données de tendance de la mémoire tampon interne et rafraichit la vue du tableau
- BACKWARD/FORWARD
  Déplace la fenêtre d'affichage vers l'avant ou vers l'arrière comme spécifié dans le paramètre de durée.

Filtrer :

Utilisez la liste déroulante pour sélectionner la colonne où rechercher et le texte listé à droite pour saisir la chaîne de caractères à rechercher.

Paramètre	Description
AuditBuffer	Mémoire tampon d'événements à partir de laquelle la liste des événements est récupérée (voir "Configurer les événements d'audit" Page 304).
En-tête	Étiquette En-tête
Durée par défaut	Valeur initiale de la fenêtre de temps à afficher
Spéc heure	Format de l'heure :

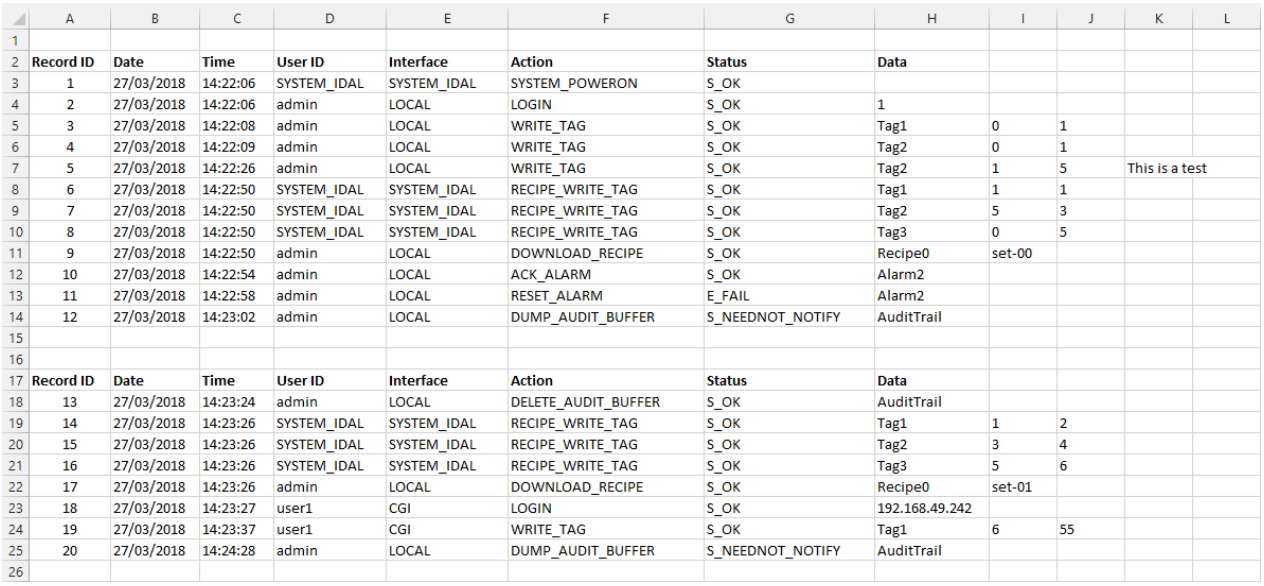
Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Paramètre	Description
	• **Locale** = affiche les valeurs de temps du périphérique IHM. • **Globale** = affiche les valeurs de temps au format UTC.
Format de date	Sélectionnez le format de Date et d'Heure
Liste des filtres	Étiquettes à afficher dans la sélection de la colonne de filtre
Disposition du tableau	Définit les caractéristiques de la barre de défilement et permet de supprimer l'en-tête du tableau.

Exporter le journal d’audit au format .csv

Pour afficher les données du journal d’audit, vous devez les exporter au format .csv avec l'action **SauvegarderArchiveÉvénements**. Voir "SaveEventArchive" Page 170 pour les détails.

Structure du fichier



	A	B	C	D	E	F	G	H	I	J	K	L
1												
2	**Record ID**	**Date**	**Time**	**User ID**	**Interface**	**Action**	**Status**	**Data**				
3	1	27/03/2018	14:22:06	SYSTEM_IDAL	SYSTEM_IDAL	SYSTEM_POWERON	S_OK					
4	2	27/03/2018	14:22:06	admin	LOCAL	LOGIN	S_OK	1				
5	3	27/03/2018	14:22:08	admin	LOCAL	WRITE_TAG	S_OK	Tag1	0	1		
6	4	27/03/2018	14:22:09	admin	LOCAL	WRITE_TAG	S_OK	Tag2	0	1		
7	5	27/03/2018	14:22:26	admin	LOCAL	WRITE_TAG	S_OK	Tag2	1	5	This is a test	
8	6	27/03/2018	14:22:50	SYSTEM_IDAL	SYSTEM_IDAL	RECIPE_WRITE_TAG	S_OK	Tag1	1	1		
9	7	27/03/2018	14:22:50	SYSTEM_IDAL	SYSTEM_IDAL	RECIPE_WRITE_TAG	S_OK	Tag2	5	3		
10	8	27/03/2018	14:22:50	SYSTEM_IDAL	SYSTEM_IDAL	RECIPE_WRITE_TAG	S_OK	Tag3	0	5		
11	9	27/03/2018	14:22:50	admin	LOCAL	DOWNLOAD_RECIPE	S_OK	Recipe0	set-00			
12	10	27/03/2018	14:22:54	admin	LOCAL	ACK_ALARM	S_OK	Alarm2				
13	11	27/03/2018	14:22:58	admin	LOCAL	RESET_ALARM	E_FAIL	Alarm2				
14	12	27/03/2018	14:23:02	admin	LOCAL	DUMP_AUDIT_BUFFER	S_NEEDNOT_NOTIFY	AuditTrail				
15												
16												
17	**Record ID**	**Date**	**Time**	**User ID**	**Interface**	**Action**	**Status**	**Data**				
18	13	27/03/2018	14:23:24	admin	LOCAL	DELETE_AUDIT_BUFFER	S_OK	AuditTrail				
19	14	27/03/2018	14:23:26	SYSTEM_IDAL	SYSTEM_IDAL	RECIPE_WRITE_TAG	S_OK	Tag1	1	2		
20	15	27/03/2018	14:23:26	SYSTEM_IDAL	SYSTEM_IDAL	RECIPE_WRITE_TAG	S_OK	Tag2	3	4		
21	16	27/03/2018	14:23:26	SYSTEM_IDAL	SYSTEM_IDAL	RECIPE_WRITE_TAG	S_OK	Tag3	5	6		
22	17	27/03/2018	14:23:26	admin	LOCAL	DOWNLOAD_RECIPE	S_OK	Recipe0	set-01			
23	18	27/03/2018	14:23:27	user1	CGI	LOGIN	S_OK	192.168.49.242				
24	19	27/03/2018	14:23:37	user1	CGI	WRITE_TAG	S_OK	Tag1	6	55		
25	20	27/03/2018	14:24:28	admin	LOCAL	DUMP_AUDIT_BUFFER	S_NEEDNOT_NOTIFY	AuditTrail				
26												

Le fichier des données exportées contient les éléments suivants	
ID d'enregistrement	Chaque enregistrement est stocké avec un numéro progressif qui donne la possibilité d'identifier facilement les enregistrements manquants ou de confirmer qu'ils ne sont pas perdus. Notez que le numéro progressif n'est pas remis à zéro lorsque la mémoire tampon est supprimée.
Date, Heure	Horodatage d’événement. L'heure peut être configurée comme heure locale ou globale pour l'action de vidage.
ID d'utilisateur	L'utilisateur qui accomplit l'opération
Interface	LOCALE : lorsque l'action est réalisée dans le périphérique IHM

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

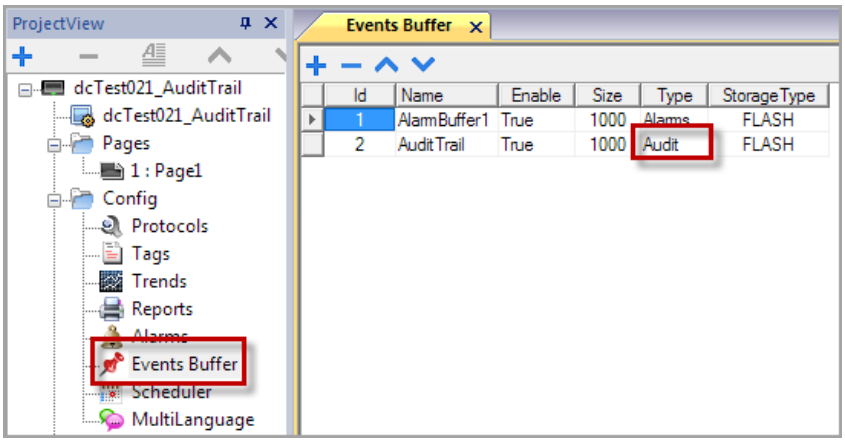
Le fichier des données exportées contient les éléments suivants	
	CGI : lorsque l'action est réalisée par un client à distance. SYSTEM_IDAL : lorsque l'action est réalisée dans l'application HMI Runtime
Action	Action exécutée.
Status	Résultat de l'action exécutée • S_OK Action exécutée correctement • E_FAIL Action non exécutée • S_NEEDNOT_NOTIFY Action déclenchée (sera exécutée de manière asynchrone)
Informations	Informations supplémentaires relatives à l'action exécutée.

Configurer les événements d'audit

Vous avez plus d'un ensemble d'entrées d'audit. Vous devez configurer une mémoire tampon d'événements dédiée.

Créer une mémoire d'événements

*Chemin : **VueProjet**> **Configuration** > double-cliquez sur **Mémoire d'événements***



1. Dans l'éditeur **Mémoire d'événements**, cliquez sur **+** : une ligne s'ajoute au tableau.
2. Sélectionnez **Audit** pour **Type**.
3. Configurez les paramètres de mémoire tampon.

Paramètre	Description
ID	Numéro d'identification mémoire tampon
Nom	Nom de mémoire tampon
Activer	Activer/désactiver l'enregistrement

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Paramètre	Description
Taille	Taille du fichier journal. Les données sont automatiquement sauvegardées sur un disque toutes les 5 minutes.
Type	Type d'événements enregistrés : • **Alarmes** • **Audit** • **Générique**
Périphérique de stockage	Périphérique où les données d'audit sont stockées

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

25 Rapports

Un rapport est une collection d'informations dont l'impression est déclenchée par un événement. Lorsqu'un événement programmé est déclenché, l'impression démarre en arrière-plan.

Vous pouvez configurez les rapports, leurs contenus, les conditions de déclenchement et l'imprimante de sortie dans l'éditeur de rapport.

Tous les widgets ne peuvent pas être utilisés dans les rapports. Lorsque vous configurez vos rapports, PB610 Panel Builder 600 fournit l'accès à une galerie de widgets spécifique où n'apparaissent que les widgets disponibles pour les rapports.

Le format des rapports peut être personnalisé à l'aide de modèles de mise en page prédéfinis.

Remarque : l'impression des rapports n'est pas prise en charge par HMI Client.

Ajouter un rapport **308**

Configurer des rapports de texte **308**

Configurer les rapports graphiques **309**

Événements déclencheurs d'impression **310**

Imprimante par défaut **311**

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Ajouter un rapport

Chemin : ***VueProjet*** *>* ***Configuration*** *> double-cliquez sur* ***Rapport***

Dans l'éditeur des **Rapports**, cliquez sur **Rapport graphique** ou **Rapport texte** : une nouvelle ligne est ajoutée au tableau.

Types de rapport

Type de rapport	Description
Rapports texte	Utilisés pour l'impression ligne par ligne d'alarmes. Utilisés uniquement pour des imprimantes par ligne. Le texte est envoyé à l'imprimante sans pilote spécifique. **Important : ce mode d'impression requiert l'utilisation d'un port physique et fonctionne uniquement sur les plateformes Windows CE.**
Rapports graphiques	Ils contiennent des éléments graphiques et peuvent inclure des widgets complexes tels que des captures d'écran ou des alarmes. **Important : chaque imprimante requiert un pilote d'imprimante spécifique. Voir "Configurer les rapports graphiques" Page opposée pour consulter la liste des pilotes d'imprimante pris en charge.**

Configurer des rapports de texte

Utilisez l'éditeur de **Rapports**. **Taille du papier** en nombre de caractères.

Réglage des options d'impression

Utilisez les options de l'imprimante pour commander le vidage des pages sur l'imprimante.

L'impression commence immédiatement ou après un délai d'attente. Dans les options d'impression, vous pouvez forcer le vidage dès qu'une condition spécifique se produit, après un certain nombre d'événements, de lignes ou de secondes.

Remarque : les rapports de texte ne prennent pas en charge le format PDF.

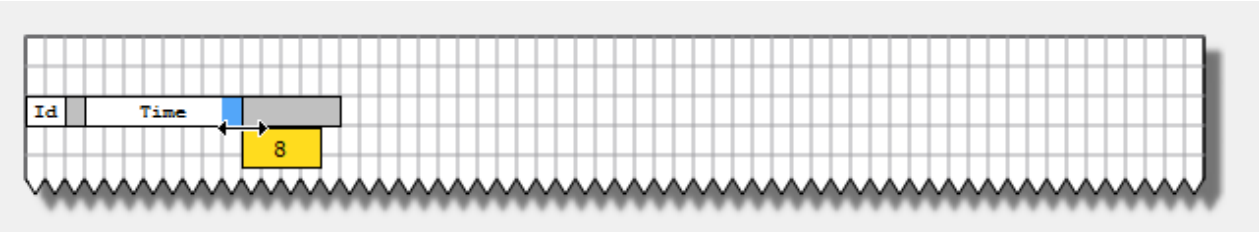
Réglage de la mise en page des alarmes

La **Taille du papier** est la largeur du papier en nombre de caractères.

Ajout de champs au rapport

Pour ajouter un élément au rapport, faites-le glisser et déposez-le sur la page de modèle dans la liste des **Champs disponibles**.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

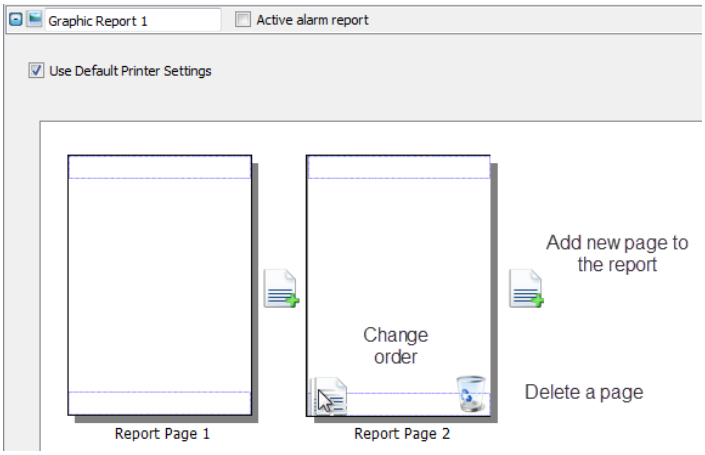


Recadrage du champ à l'aide de la souris, un outil de pointe montre la dimension en nombre de caractères.

Remarque : si le texte ne tient pas dans l'espace dédié, l'enveloppement automatique est appliqué.

Configurer les rapports graphiques

Utilisez l'éditeur de **Rapport** pour configurer les rapports graphiques.



Ajouter une page de rapport

Cliquez sur **+** pour ajouter une nouvelle page au modèle de rapport.

Quand la souris va sur une page, deux icônes s'affichent et permettent de réordonner ou de supprimer les pages.

Modifier le contenu de la page de rapport

1. Double-cliquez sur une page pour modifier son contenu : l'éditeur de **Rapport graphique** apparaît.

   Chaque page se subdivise en : en-tête, pied de page et corps de la page.

2. Double-cliquez sur la zone que vous souhaitez modifier : la zone à modifier apparaît en blanc, tandis que les autres sont grisées.

La Galerie des widgets est sensible au contexte et n'affiche que les widgets disponibles pour la zone que vous êtes en train de modifier.

Widgets disponibles pour les rapports

Widgets pouvant être utilisés pour un rapport graphique :

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Widget	Fonction
Numéro de page	Numérotation de page automatique
Capture d'écran	Capture d'écran de la page actuellement affichée sur le périphérique IHM. La page de rapport est automatiquement redimensionnée pour s'adapter à la page du périphérique IHM. Remarque : l'intégralité de l'écran est imprimée, y compris les boîtes de dialogue ouvertes.
Alarme	La totalité du contenu de la mémoire d'événements (la mémoire tampon par défaut est Alarm Buffer1).
Texte	Les widgets tels que les labels et les champs numériques

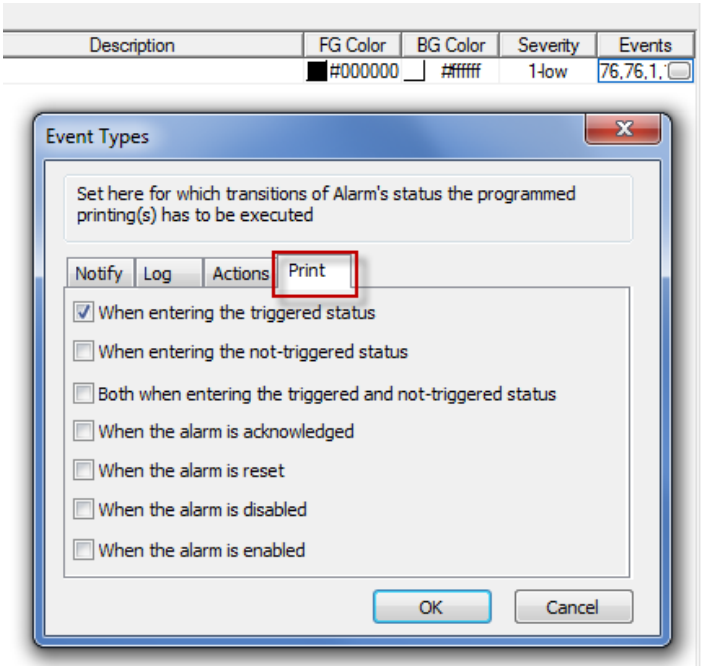
Événements déclencheurs d'impression

L'impression des rapports peut être déclenchée par des événements.

Configurer l'impression d'alarme

*Chemin : **VueProjet**> **Configuration** > double-cliquez sur **Alarmes***

1. Dans l'Éditeur d'alarme, ouvrez la boîte de dialogue **Types d'événements** à partir de la colonne **Événements**.
2. Dans l'onglet **Impression**, sélectionnez toutes les conditions dont vous souhaiteriez qu'elles déclenchent une impression.



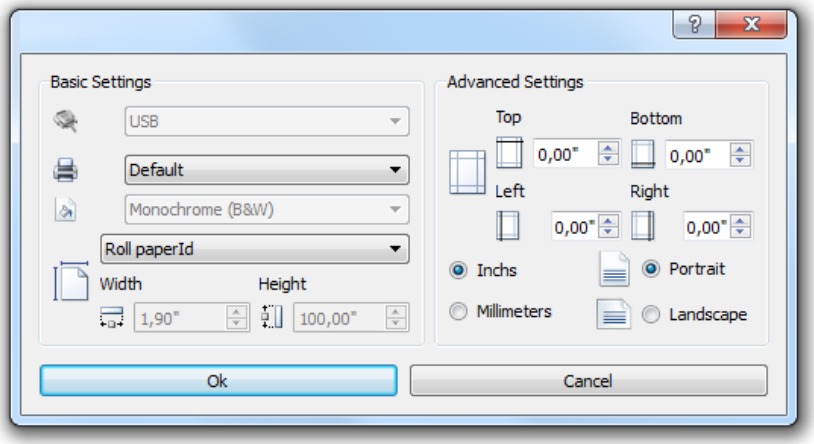
Important : il n'est possible de définir qu'un seul rapport en tant que Rapport d'alarme active au sein d'un projet et ce rapport peut se présenter sous forme de texte ou de graphique.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Ajustement des paramètres de l'imprimante au runtime

L'impression d'un rapport graphique peut également être lancée en utilisant l'action **ImpressionRapportGraphique**.

Configurez la propriété **silencieux** de l'action sur **faux** pour faire apparaître la boîte de dialogue.



Imprimante par défaut

Paramètres de l'imprimante

Vous pouvez définir une imprimante par défaut pour tous les rapports graphiques. Chaque rapport peut ensuite être configuré pour utiliser l'imprimante par défaut ou toute autre imprimante disponible. Cliquez sur le bouton **Paramètres de l'imprimante** pour définir les paramètres de l'imprimante.

Pour les imprimantes PDF, vous définissez aussi le dossier de sauvegarde des fichiers avec **Emplacement des fichiers imprimés**.

Imprimantes prises en charge

Liste d'imprimantes et de langages d'imprimantes pris en charge par le pilote de Windows CE printCE.dll. Les imprimantes qui ne figurent pas sur la liste mais qui sont compatibles avec ces langages sont prises en charge.

Imprimante	Langues
HP PCL 3, HP PCL 5e, HP PCL3GUI	HP PCL3/PCL5e/PCL3GUI, notamment DeskJet, LaserJet et DesignJet
Epson ESC/P2	ESC/P2 et LQ
Epson Stylus Color	Epson Stylus Color
Epson LX (9 broches)	Imprimantes 9 broches, Epson LX, FX et PocketJet
Cannon iP100, iP90, BubbleJet	BubbleJet, iP90, iP100
PocketJet II, 200, 3	Pocket Jet

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Imprimante	Langues
MTE Mobile Pro Spectrum	MTE Mobile Pro Spectrum
Fichier Adobe PDF	Fichier Adobe PDF
SPT-8	SPT-8
M1POS	M1POS
MP300	MP300
Zebra	Langage Zebra CPCL
Intermec PB42, PB50, PB51, PB2, PB3	Intermect PB42/50/51/2/3 avec langage ESC/P
Datamax Apex	Datamax Apex

Ports pris en charge

Les ports suivants sont pris en charge :

- LPT1 (imprimantes USB)
- Fichier (PDF)

Remarque : sur la plateforme Win32, seules les imprimantes PDF et celles par défaut sont prises en charge. L'imprimante par défaut est l'imprimante par défaut du système d'exploitation. Elle peut être connectée avec tout type de port (pas uniquement USB).

Imprimantes testées

Les imprimantes suivantes ont été testées avec les pilotes printCE sur les périphériques IHM sous Windows CE.

Pilote	Modèle de l'imprimante	Graphique	Ligne
Personnalisé	Plus 4 Kube II	Oui	Oui
Epson ESC/P 2	Epson AcuLaser M2310	Oui	Simuler
Epson LX (9 broches)	Epson LX-300+II	Non	Oui
HP PCL 3	HP LaserJet P2015dm	Oui	Simuler
	HP LaserJet 4700dtn	Oui	Oui
HP PCL 3 GUI	HP Deskjet 1010	Oui	Non
	HP Deskjet D5560	Oui	Non
	HP LaserJet 4700dtn	Non	Oui
HP PCL 5e	HP LaserJet P2015dm	Oui	Simuler
	HP LaserJet 4700dtn		

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Pilote	Modèle de l'imprimante	Graphique	Ligne
INTERMEC	Intermec PB50 avec langage ESC/P avec du papier en rouleau de 4 pouces (environ 10 cm). Remarque : le périphérique IHM se bloque lorsque vous essayez d'imprimer avec les imprimantes Intermec PB50 en mode veille après une première impression réussie.	Oui	Oui
PDF	-	Oui	Non

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

26 Écran de veille

L'écran de veille peut être utilisé pour afficher un diaporama lorsque le périphérique IHM n'est pas utilisé. Le diaporama commence après un délai d'attente si aucun des événements suivants se produit :

- touche d'écran
- déplacement de souris
- touche de clavier externe appuyée

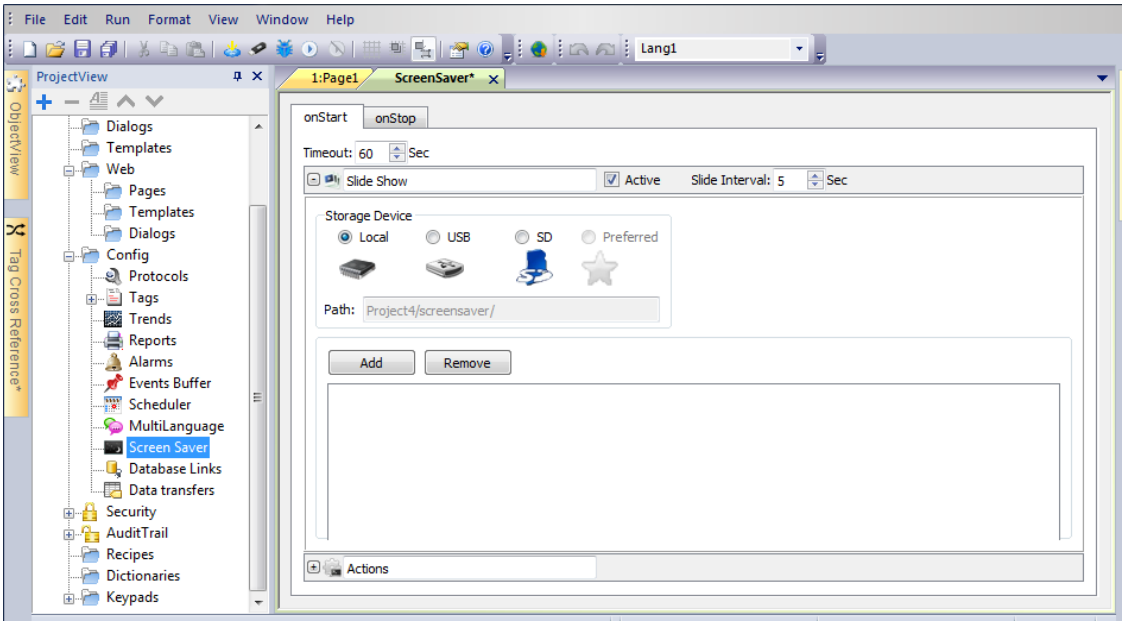
Activation de la fonction d'écran de veille

*Chemin : **VueProjet**> **Config** > clic droit sur **Écran de veille**> **Activé***

Important : vous devez activer l'économiseur d'écran avant de pouvoir le configurer.

Configuration d'un écran de veille

*Chemin : **VueProjet**> **Configuration** > double-clic sur **Écran de veille***



Paramètres du diaporama.

Paramètre	Description
Délai d'attente	Temps après lequel le diaporama démarre.
Intervalle	Intervalle entre les diapos

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Paramètre	Description
entre les diapos	
Périphérique de stockage	Emplacement des images utilisées dans le diaporama. Les images stockées localement sont enregistrées dans *workspace\projectname\screensaver* et peuvent être téléchargées sur le périphérique IHM lorsque les projet est téléchargé. Les images stockées sur les périphériques USB ou SD sont enregistrées dans un dossier d'écran de veille sur le périphérique même. **Important : seules les images JPEG et PNG sont prises en charge.**

Association d’actions à l'écran de veille

Les actions peuvent être déclenchées par le début et/ou l’arrêt de l'écran de veille.

- Cliquez sur **+** à côté d'Actions dans l'onglet **onStart** pour configurer les actions à exécuter lorsque l'écran de veille démarre.
- Cliquez sur **+** à côté d'**Actions** dans l'onglet **onStop** pour configurer les actions à exécuter lorsque l'économiseur d'écran s’arrête.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

27 Sauvegarder/restaurer le runtime et le projet

Vous pouvez sauvegarder tout le contenu du périphérique IHM, comprenant

- HMI Runtime
- Projet d'application IHM

pour une mémoire externe. Cette copie de sauvegarde peut être utilisée pour restaurer le contenu du périphérique IHM par la suite ou pour le copier dans un nouveau périphérique IHM.

Cette fonction de sauvegarde est uniquement disponible lorsqu'elle est activée pour l'utilisateur connecté. Voir "Modifier des permissions d'accès" Page 286 pour les détails.

Remarque : la sauvegarde n'est pas supportée dans Win32 / HMI Client.

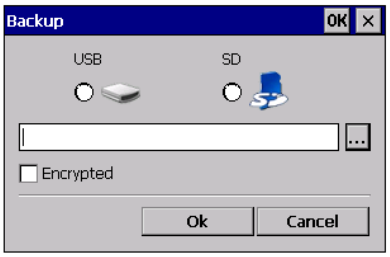
Fonction de sauvegarde

La fonction de sauvegarde réalise automatiquement la procédure suivante :

1. Décharge le projet en cours pour déverrouiller les fichiers utilisés.
2. Archive le contenu du dossier \QTHMI (qui contient HMI Runtime, les projets, les fichiers dynamiques tels que les recettes, les alarmes, les tendances, etc.) dans un fichier .zip (standard ou crypté).
3. Réinitialise le périphérique IHM (recharge le projet).

Pour démarrer la procédure de sauvegarde :

1. Dans HMI Runtime, faites un clic droit pour ouvrir le menu contextuel.
2. Sélectionnez **Exporter variables** : le dialogue **Exportation** s'affiche.



3. Sélectionnez le chemin de stockage du fichier de sauvegarde.

Remarque : le processus de sauvegarde n'inclut pas les fichiers stockés dans des clés USB et dans des cartes SD. Les données dynamiques telles que les recettes, les tendances ou les événements stockés dans ces périphériques ne seront pas incluses dans la sauvegarde.

Fonction restauration

La restauration du paquet de sauvegarde peut être effectué sur périphérique IHM

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

- depuis le Menu Contextuel (voir "Mettre à jour un paquet" Page 91 pour obtenir plus de détails)
- ou depuis les Paramètres du Système (voir "Paramètres du système" Page 477 pour obtenir plus de détails)

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

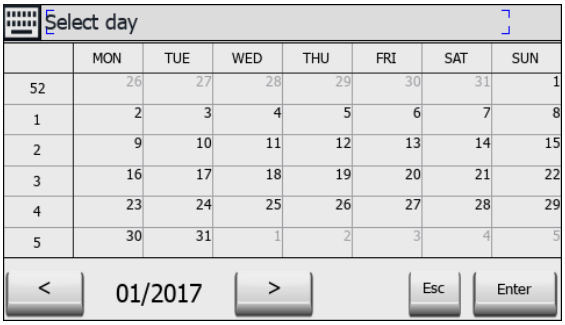
28 Claviers

De nombreux claviers sont fournis par défaut dans le PB610 Panel Builder 600 et peuvent être utilisés pour la saisie de données.

Le clavier alphabétique peut être associé à une utilisation avec un type de données chaîne

Le clavier numérique peut être associé à une utilisation avec un type de données numérique

Le clavier calendrier peut être associé à une utilisation avec un type de données de date



Créer et utiliser des claviers personnalisés **321**

Supprimer ou renommer des claviers personnalisés **323**

Type de clavier **323**

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Position de clavier **324**

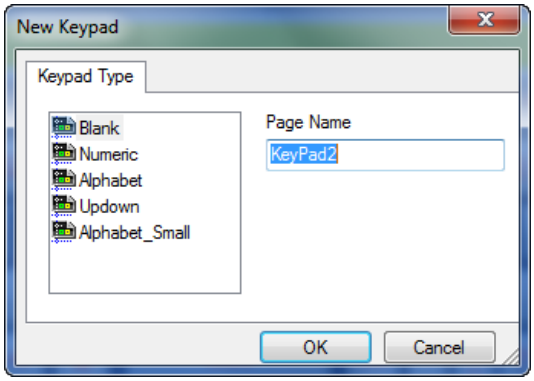
Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Créer et utiliser des claviers personnalisés

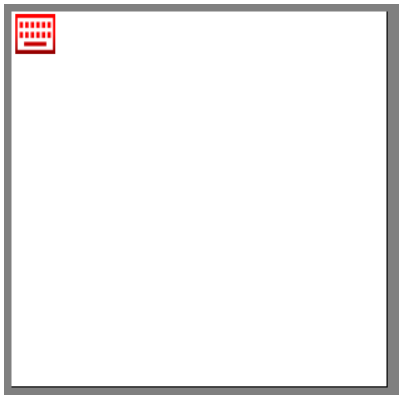
Vous pouvez soit créer un nouveau clavier soit personnaliser un clavier existant.

Créer un clavier

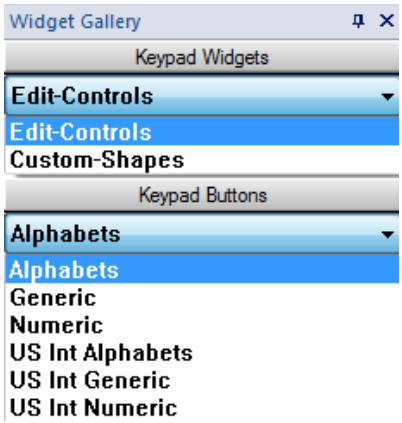
1. Dans **VueProjet**, faites un clic droit sur **Claviers** et sélectionnez **Insérer clavier** : la boîte de dialogue **Nouveau clavier** s'affiche.



2. Sélectionnez un des claviers disponible ou **Vierge** pour créer un clavier de A à Z. Dans ce cas, un clavier vierge est affiché.



3. Utilisez **Widgets clavier** et **Boutons clavier** de la galerie des widgets pour créer votre clavier personnalisé.

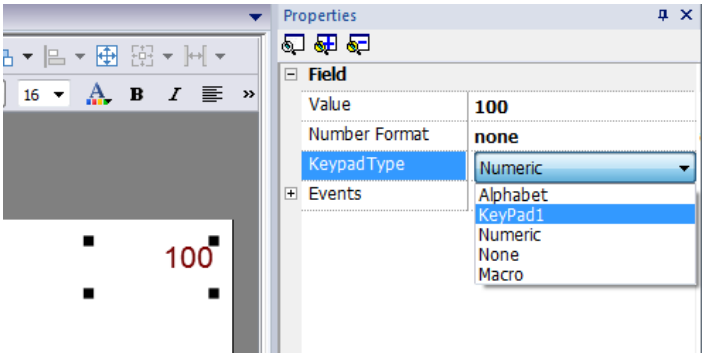


Comme dans cet exemple, le clavier que vous créez est sauvegardé dans le dossier du projet.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Associer des claviers personnalisés à des champs

Les claviers personnalisés peuvent être réutilisés pour tout champ où la propriété **Clavier** pointe dessus, comme dans l'exemple.



Astuces avec les claviers personnalisés

Par défaut, le clavier numérique est assigné à tous les widgets numériques (écrire/lire un champ numérique).

Si vous voulez appliquer une version personnalisée du clavier numérique à tous les widgets numériques que vous ajoutez à votre projet, procédez de la façon suivante :

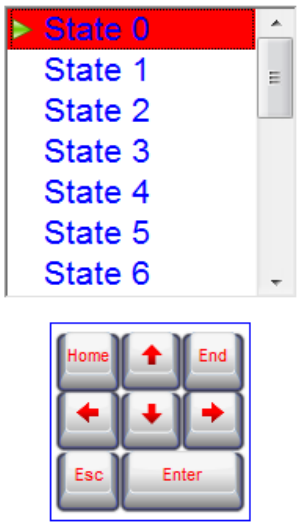
1. Créez un nouveau clavier et sélectionnez **Numérique**comme type de **Clavier**. C'est une sauvegarde des paramètres d'origine pour le clavier numérique.
2. Personnalisez le clavier numérique par défaut et enregistrez-le. Cette version personnalisée du clavier numérique est assignée par défaut dans le projet.

Voir "Supprimer ou renommer des claviers personnalisés" Page opposée pour des détails sur la façon de renommer un clavier personnalisé.

Flèches haut-bas du clavier

Ce type de clavier est particulièrement utile pour déplacer le curseur de haut en bas dans un widget qui requiert cette fonctionnalité. Voici un exemple avec le widget **Liste de contrôle**. Voir "Widgets de liste de contrôle" Page 373 pour plus de détails.

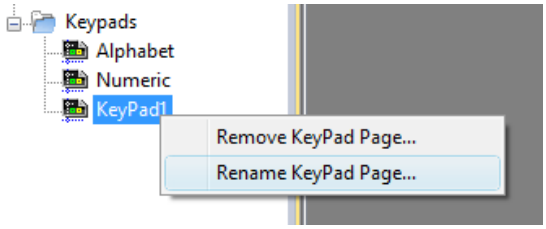
Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



Supprimer ou renommer des claviers personnalisés

Dans **VueProjet**, faites un clic droit sur un clavier personnalisé et sélectionnez une des options suivantes :

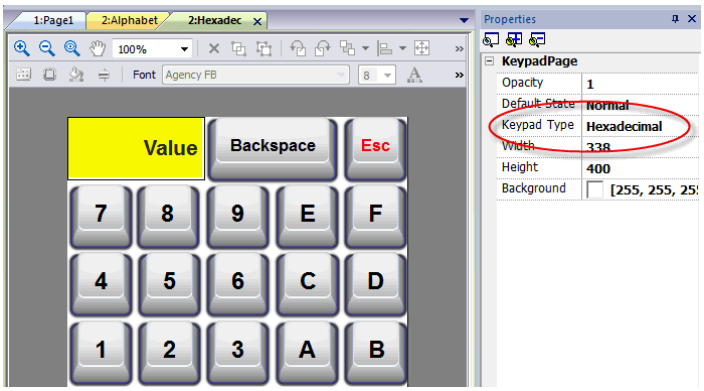
- **Supprimer page de clavier** pour supprimer le clavier du projet
- **Renommer page de clavier** pour renommer le clavier.



Type de clavier

*Chemin : **VueProjet** > **Claviers** > double-cliquez sur un clavier > **Propriétés***

Configurez le paramètre **Type de clavier** pour un clavier afin de définir le type de saisie de données.



Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Type de clavier	Description
Auto	Paramètres par défaut
Décimal	Seuls les touches numériques sont acceptées. En entrant 10, le clavier retourne 10, qui s'affichera comme « 10 » si le champ associé est numérique ou ASCII, comme 'A' si le fichier associé est hexadécimal.
Hexadécimal	Seuls les touches hexadécimales sont acceptées. En entrant 10, le clavier retourne 16, qui s'affichera comme « 16 » si le champ associé est numérique ou ASCII, comme « 10 » si le champ associé est hexadécimal.
Ascii	Toutes les touches sont acceptées. En entrant 1A, le clavier retourne 1A, qui s'affichera comme « 1 » si le champ associé est numérique, comme « 1A » si le champ associé est ASCII ou hexadécimal.

Position de clavier

La propriété **Emplacement de runtime** des claviers peut être utilisée pour définir l'emplacement des claviers sur l'écran.

Option	Description
Automatique	Sélection du meilleur emplacement en fonction de l'endroit où la saisie de données est requise.
Absolu	Des coordonnées X,Y sont renseignées pour définir l'emplacement exact
Gauche-haut **Gauche-centre** **Gauche-bas** **Centre-haut** **Centre-centre** **Centre-bas** **Droite-haut** **Droite-centre** **Droite-bas**	Emplacements écran prédéfinis

Sélectionnez l'option **Verrouiller l'emplacement du clavier** pour empêcher le déplacement du clavier par glissement.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

29 Claviers externes

HMI Runtime a été conçu pour fonctionner avec des claviers externes connectés via USB.

Les claviers peuvent être utilisés pour :

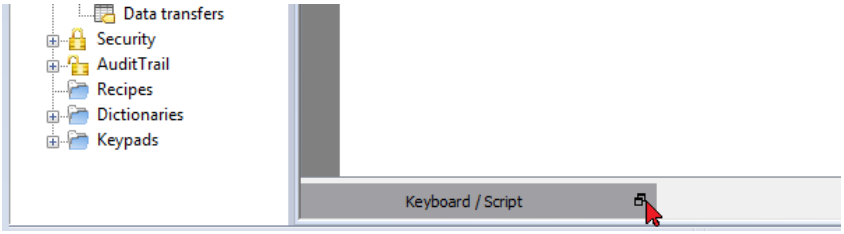
- l'entrée de données (par défaut)
- l'exécution d'actions associées à des touches spécifiques

Par exemple, l'événement **LorsdeClic** sur la flèche droite peut être associé à l'action **ChargerPage**.

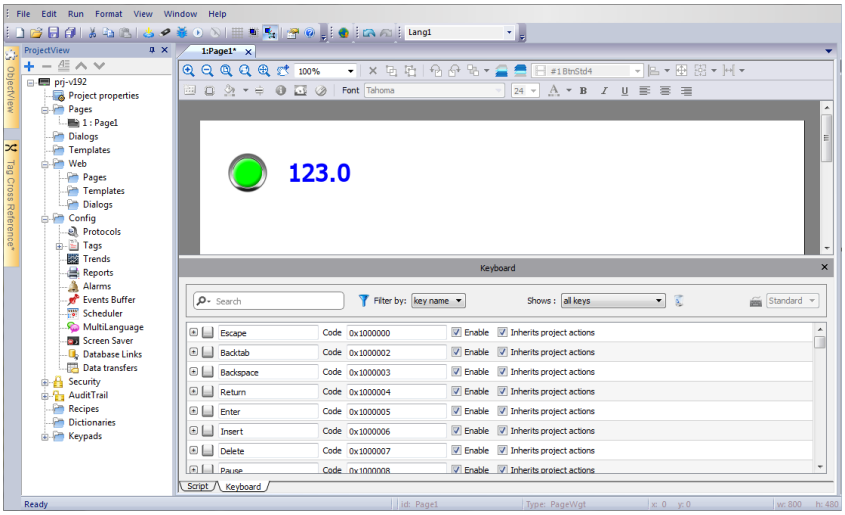
Le clavier peut être programmé au niveau du projet pour que les paramètres soient hérités par toutes les pages. Dans chaque page, vous pouvez ensuite choisir quel paramètre de touche est hérité depuis le projet et quel paramètre est personnalisé pour la page spécifique.

Ouvrir des claviers externes

1. Sur l'éditeur de page, cliquez sur l'icône à droite de **Clavier/Script**, en bas de l'espace de travail : l'éditeur Clavier/Script s'affiche.
2. Sélectionnez l'onglet **Clavier**.



Chaque ligne dans l'éditeur de clavier correspond à une touche.



Pour chaque touche, les informations suivantes sont affichées :

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Élément	Description
Étiquette	Nom de la touche
Code	Code de la touche
Activer	État d'activation de la touche
Hérite des actions du projet	Définit si la touche hérite de l'action programmée au niveau du projet

Voici les configurations possibles :

Activer	Hérite des actions du projet	Apparence d'éditeur	Comportement HMI Runtime
Vérifié	Non vérifié	Les listes d'action affichent les actions de page (ou rien si la liste est vide)	Seules les actions de page (le cas échéant) sont exécutées.
Vérifié	Vérifié	Les listes d'action affichent uniquement les actions du projet et ne peuvent pas être éditées	Seules les actions de page (le cas échéant) sont exécutées.
Non vérifié	Vérifié	Hérite de la case à cocher des actions de projet et toutes les listes d'action sont désactivées. Les listes d'action affichent uniquement les actions du projet.	Aucune action de page ou de projet n'est exécutée.
Non vérifié	Non vérifié	Hérite de la case à cocher des actions de projet et toutes les listes d'action sont désactivées. Les listes d'action affichent uniquement les actions du projet.	Aucune action de page ou de projet n'est exécutée.

Chercher et filtrer **327**

Touches affichées **327**

Retirer des associations d'actions **327**

Disposition du clavier **328**

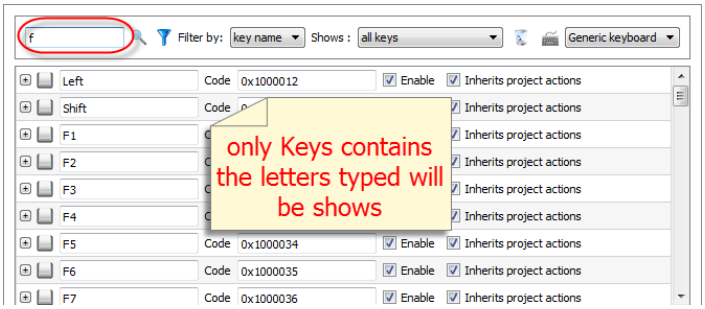
Activer/désactiver clavier **328**

Associer des actions à des touches **328**

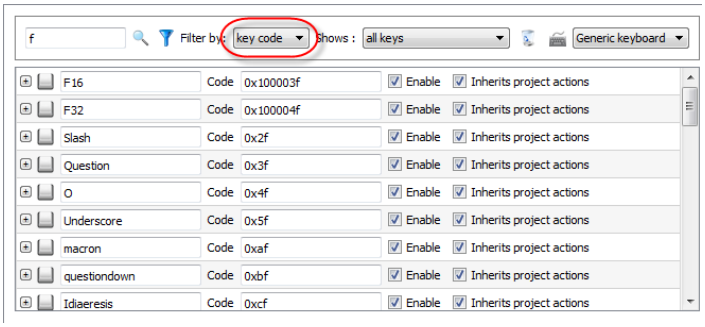
Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Chercher et filtrer

Pour afficher un ensemble filtré de touches, dans **Filtrer par** sélectionner le **nom clé** et tapez une lettre dans le champ de recherche : seules les touches contenant cette lettre dans leur nom seront affichés dans l'éditeur du clavier.



Comme alternative, dans **Filtrer par** sélectionner le **code clé** et tapez une lettre dans le champ de recherche : seules les clés contenant cette lettre dans leur code seront affichés dans l'éditeur du clavier.



Touches affichées

Vous pouvez facilement sélectionner quelles touches seront répertoriées dans la fenêtre de l'éditeur de Clavier. Pour afficher un ensemble limité de touches, sélectionnez une option dans **Affichages**.

Option	Description
toutes les touches	Toutes les touches disponibles dans la configuration du clavier sont répertoriées
touches modifiées	Seules les touches associées aux actions au niveau page sont répertoriés
touches modifiées dans le projet	Seules les touches associées aux actions au niveau du projet sont répertoriées

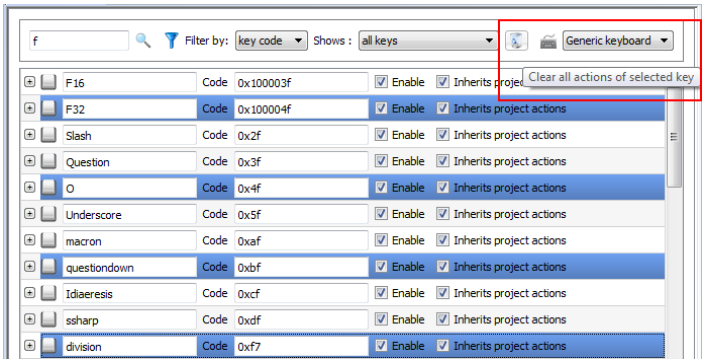
Retirer des associations d'actions

Pour retirer toutes les associations que vous avez créées entre des touches et des actions :

1. Sélectionnez les touches desquelles vous voulez retirer l'association.
2. Cliquez sur le bouton **Effacer toutes les actions des touches sélectionnées**.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Si vous travaillez au niveau d'une page, les actions de la page seront retirées. Si vous travaillez au niveau d'un projet, les actions du projet seront retirées.

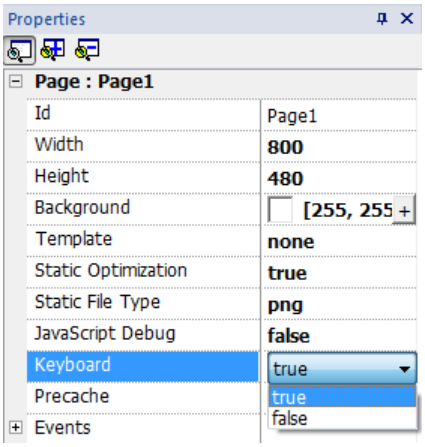


Disposition du clavier

Sélectionnez la disposition de clavier à partir de la liste déroulante **Disposition du clavier**. **Clavier générique** se réfère à une disposition de clavier générique internationale.

Activer/désactiver clavier

Vous pouvez activer/désactiver les actions du clavier dans le projet et au niveau de la page. Pour activer les actions du clavier, définissez dans le volet **Propriétés** le **Macro clavier** sur **Vrai**.



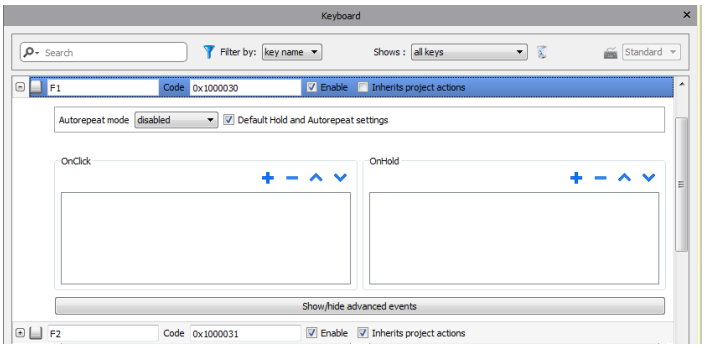
Vous pouvez également activer/désactiver les actions du clavier au runtime avec l'action KeyboardMacros. Voir "Actions clavier" Page 145 pour plus de détails.

Associer des actions à des touches

Vous associez des actions à des touches depuis l'éditeur Clavier.

1. Cliquez sur **+** à côté de la touche que vous voulez programmer : les champs pour la configuration de la touche s'affichent.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

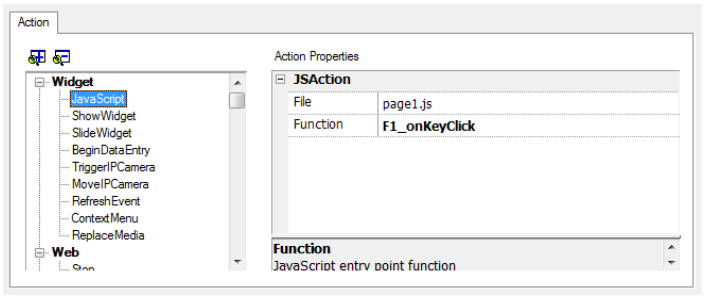


1. Cliquez sur **+** pour ajouter des actions.

Vous pouvez associer des actions à l'événement **LorsdeClic** et à l'événement **LorsdeMaintienBouton**.

Voir "Événements" Page 54 pour plus de détails.

Remarque : un code JavaScript peut également être associé à un événement-clé.



Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

30 Référence croisée de variables

Le volet **Référence croisée des variables** affiche une liste de noms de variables utilisées dans le projet en cours organisé en fonction de leur emplacement et de leur utilisation.

Depuis ce volet, vous pouvez :

- vérifier où chaque variable est utilisée (alarmes, pages, recettes, planificateurs, tendances, etc.)
- identifier les références invalides de variable (références aux variables non définies dans l'éditeur de variable)
- identifier les variables non utilisées dans le projet

Remarque : le volet Référence croisée de variables ne liste pas nécessairement les variables utilisées dans le code JavaScript.

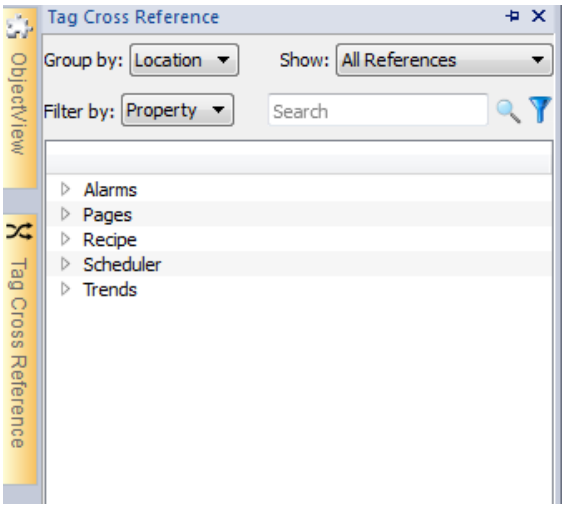
Mettre à jour les données dans le volet Référence croisée des variables **334**

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Ouvrir le volet Référence croisée des variables

Chemin : ***Affichage*** *>* ***Barre d'outils et ancrage de fenêtres*** *>* ***Référence croisée de variable***

Cliquez sur l'onglet **Référence croisée de variable** pour ouvrir le volet Référence croisée de variable.



Travailler dans le volet Référence croisée des variables

Le volet Référence croisée des variables fournit un ensemble de fonctions standards.

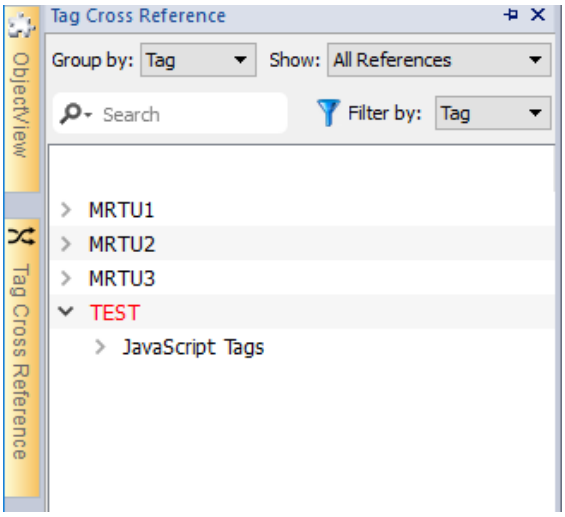
Élément	Fonction
Grouper par	Variables de groupes par **Emplacement** (alarmes, pages, tendances, etc.) ou Nom de **variables**
Afficher	Variables de filtres et affichages : • **Toutes les références** : toutes les variables • **Référence variables invalide** : variables non listées dans l'Éditeur de variables. • **Variables non utilisées** : variables listées dans l'éditeur de variables mais non utilisées dans le projet.
Recherche de champ	Appliquer un filtre pour afficher un nombre limité de variables
Filtrer par	Variables de filtres par **Emplacement**, **variable** ou **Propriété**.

Naviguer dans les variables listées pour trouver où elles sont utilisés dans le projet.

Double-cliquez sur une variable pour ouvrir l'éditeur ou la page où elle est utilisée.

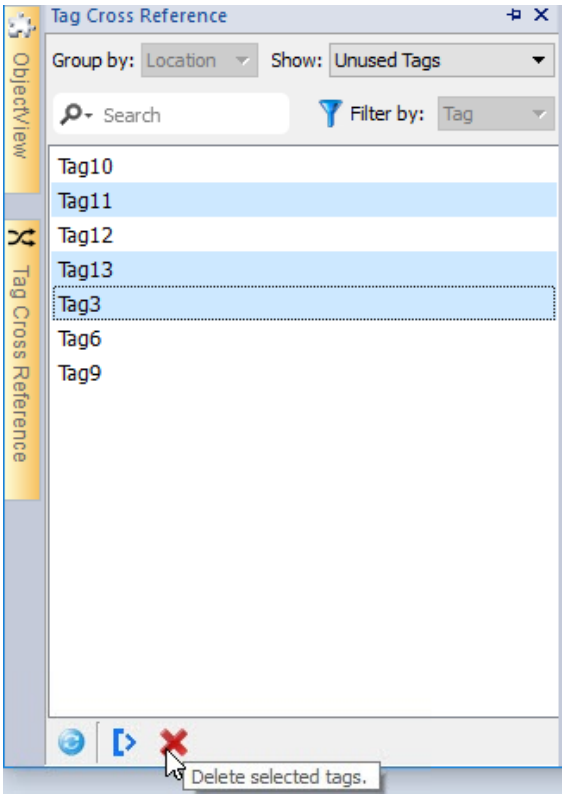
Les références de variables non valides seront listées en rouge :

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



Supprimez les variables non utilisées

Dans la vue des variables inutilisées, il est possible de sélectionner une ou plusieurs variables et de les supprimer de l'éditeur de variables. Pour sélectionner une variable, cliquez sur une variable, pour sélectionner plusieurs variables, utilisez les touches MAJ ou CTRL.



Sachez que les variables éventuellement référencées à l'intérieur de JavaScript peuvent ne pas être trouvées (selon la façon dont le code a été écrit). Même s'il n'est pas pratique d'utiliser les références des variables à l'intérieur de widgets personnalisés, même les variables référencées à l'intérieur de Widgets personnalisés peuvent ne pas être détectées à partir du moteur Référence croisée de variables.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Mettre à jour les données dans le volet Référence croisée des variables

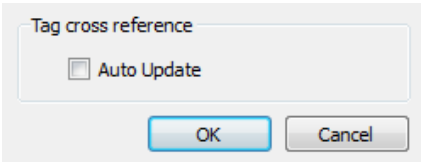
Mise à jour manuelle

Par défaut, les informations affichées dans le volet Référence croisée des variables doivent être mises à jour manuellement. Pour ce faire, cliquez sur le bouton de rafraîchissement . Un panneau d'avertissement s'affiche lorsqu'un rafraîchissement est nécessaire.

Mise à jour automatique

Chemin : ***Vue*** *>* ***Propriétés***

Vous activez la mise à jour automatique du volet Référence croisée des variables depuis la PB610 Panel Builder 600 page **Propriétés**.



Sélectionnez l'option **mise à jour automatique**.

Exporter les données

Les données affichées dans le volet Référence croisée des variables peuvent être exportées dans un fichier .csv.

Les données sont organisées dans le fichier exporté selon la manière dont elles sont groupées dans le volet.

Groupées par	Format de fichier
Emplacement	RESSOURCES, BUREAU RESSOURCES, ID WIDGET, ATTRIBUT, variable
Variable	VARIABLE, RESSOURCES, BUREAU RESSOURCES, ID WIDGET, ATTRIBUT

Remarque : les séparateurs utilisés lors de l'exportation dépendent des paramètres régionaux de votre ordinateur.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

31 Adressage indexé

L'adressage indexé vous permet de sélectionner un ensemble de variables dépendant de la valeur d'une autre variable. Il est par exemple très utile de pouvoir utiliser les mêmes graphiques pour visualiser un ensemble de données provenant de différentes sources. L'utilisateur n'a plus qu'à sélectionner la source à surveiller parmi une liste.

Générer un ensemble d’adressage indexé **336**

Utilisation d’un ensemble de variables indexées dans des pages **339**

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Générer un ensemble d’adressage indexé

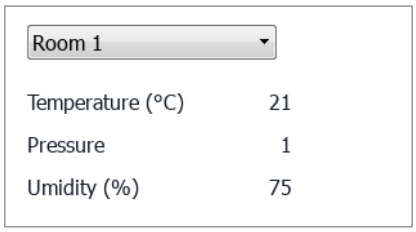
Scénario

Dans ce scénario, les données environnementales sont collectées depuis quatre pièces, chacune équipée de capteurs de température, de pression et d'humidité. Les données sont disponibles de la façon suivante :

Numéro de pièce	Température	Pression	Humidité
1	Pièce1-Température	Pièce1-Pression	Pièce1-Humidité
2	Pièce2-Température	Pièce2-Pression	Pièce2-Humidité
3	Pièce3-Température	Pièce3-Pression	Pièce3-Humidité
4	Pièce4-Température	Pièce4-Pression	Pièce4-Humidité

En utilisant la fonctionnalité d'adressage indexé, vous pouvez utiliser un seul format de tableau pour organiser toutes les données dans le périphérique IHM.

Les données provenant des trois capteurs peuvent être affichées sur une seule page, où le numéro de pièce est utilisé comme sélecteur (liste déroulante) pour choisir le bon ensemble de variables.



Créer un ensemble de variables indexées

*Chemin : **VueProjet**> **Variables***

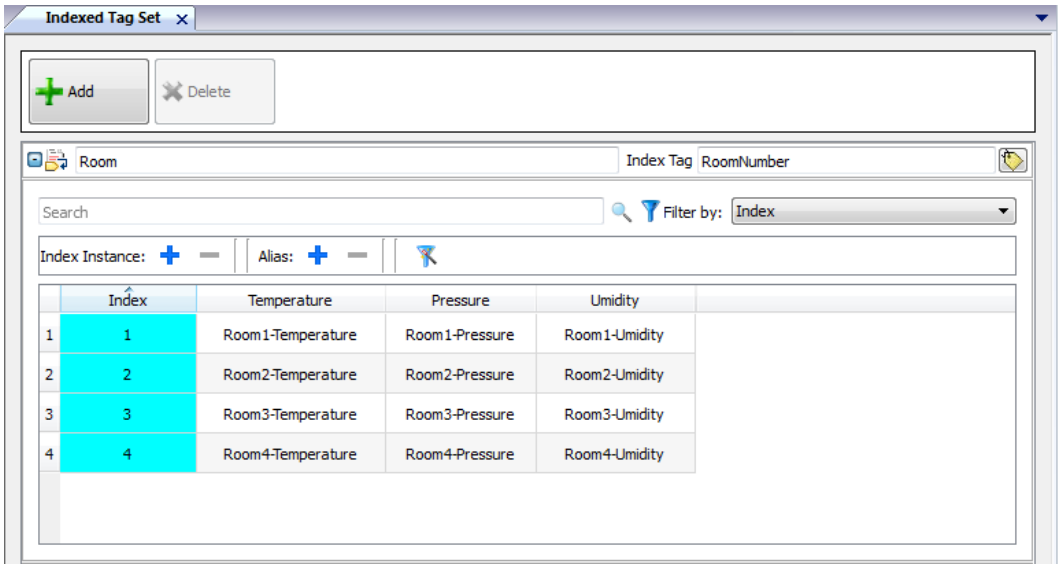
Pour ce faire, vous devez créer un ensemble de variables indexées.

1. Dans l'Éditeur de variables, définissez les protocoles et les variables. Définissez une variable pour chaque donnée à indexer, dans cet exemple vous devez créer une variable pour chaque capteur dans chaque pièce.

Name	Group	Driver	Address
Room1-Temperature		Modbus TCP:prot1	192.168.0.34:502:1 HREG 400001 unsignedShort
Room1-Pressure		Modbus TCP:prot1	192.168.0.34:502:1 HREG 400002 unsignedShort
Room1-Umidity		Modbus TCP:prot1	192.168.0.34:502:1 HREG 400003 unsignedShort
Room2-Temperature		Modbus TCP:prot1	192.168.0.34:502:1 HREG 400004 unsignedShort
Room2-Pressure		Modbus TCP:prot1	192.168.0.34:502:1 HREG 400005 unsignedShort
Room2-Umidity		Modbus TCP:prot1	192.168.0.34:502:1 HREG 400006 unsignedShort
Room3-Temperature		Modbus TCP:prot1	192.168.0.34:502:1 HREG 400007 unsignedShort
Room3-Pressure		Modbus TCP:prot1	192.168.0.34:502:1 HREG 400008 unsignedShort
Room3-Umidity		Modbus TCP:prot1	192.168.0.34:502:1 HREG 400009 unsignedShort
Room4-Temperature		Modbus TCP:prot1	192.168.0.34:502:1 HREG 400010 unsignedShort
Room4-Pressure		Modbus TCP:prot1	192.168.0.34:502:1 HREG 400011 unsignedShort
Room4-Umidity		Modbus TCP:prot1	192.168.0.34:502:1 HREG 400012 unsignedShort

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

2. Créez une variable à utiliser comme variable indexée. Dans cet exemple, vous créez une variable « NuméroPièce » qui pourrait être de type UnsignedInt utilisant le protocole Variable.
3. Dans **VueProjet**, sélectionnez **Configuration> variables**, puis double-cliquez sur **Ensemble de variables indexées** : l'éditeur d'Ensemble de variables indexées s'affiche.
4. Cliquez sur + pour afficher un Ensemble de variables indexées. Dans cet exemple, vous le nommez « Pièce ».
5. Sélectionnez la variable « NuméroPièce » à utiliser comme sélecteur pour le numéro de pièce.
6. Créez une **Instance d'index** pour chaque ensemble de données. Dans cet exemple, créez-en une pour chaque pièce.
7. Créez un **Alias** pour chaque type de données et renommez les colonnes du tableau de manière appropriée. Dans cet exemple, renommez-les « Température », « Pression » et « Humidité ».
8. Double-cliquez sur chaque cellule pour associer la variable correcte.



Remarque : le type de données de variable indexée peut être un nombre, une chaîne, ou tout type de données simples.

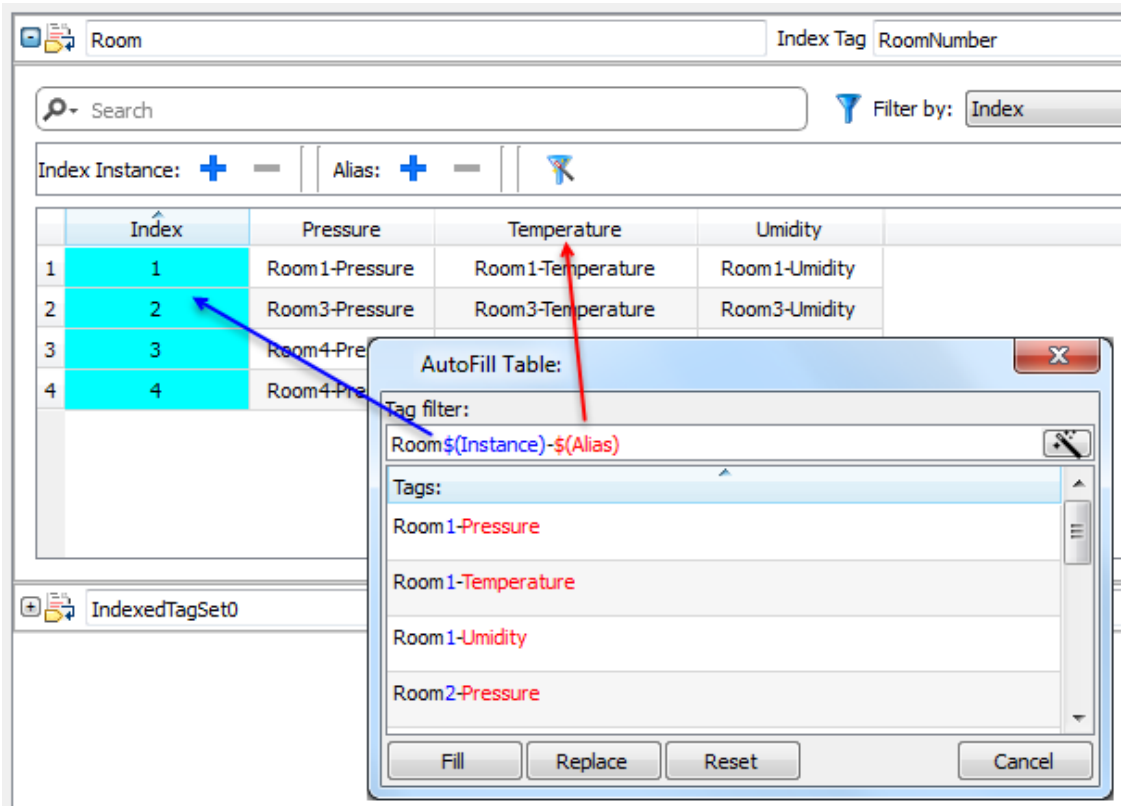
Remarque : pour référencer un type de données de tableau, utilisez le tableau index = -1

Fonction de remplissage automatique

Un tableau d'ensemble de variables indexées peut devenir très complexe et le remplir peut conduire à des erreurs. Activez la fonctionnalité Remplissage automatique pour vous assurer que les alias sont correctement entrés.

Cliquez sur pour activer la fonctionnalité Remplissage automatique : le **Tableau de remplissage automatique** s'affiche.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



Cette fonction utilise une expression régulière pour remplir le tableau avec des variables tentant de correspondre au filtre, où le mot-clé $(Instance) est remplacé par les valeurs Index définies et le mot-clé $(Alias) par les étiquettes d'alias définies.

Voir https://fr.wikipedia.org/wiki/Expression_régulière pour plus de détails sur les expressions régulières.

Exemple de remplissage automatique

« Room$(Instance)-$(Alias) » correspond à tous les noms de variables :

Pièce1-Température,

Pièce1-Pression,

Pièce1-Humidité,

Pièce2-Température,

…

« Room0*$(Instance)-$(Alias) » correspond à tous les noms de variables :

Pièce1-Température,

Pièce01-Pression,

Pièce001-Humidité,

Pièce2-Température,

Pièce02-Pression,

Pièce002-Humidité,

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

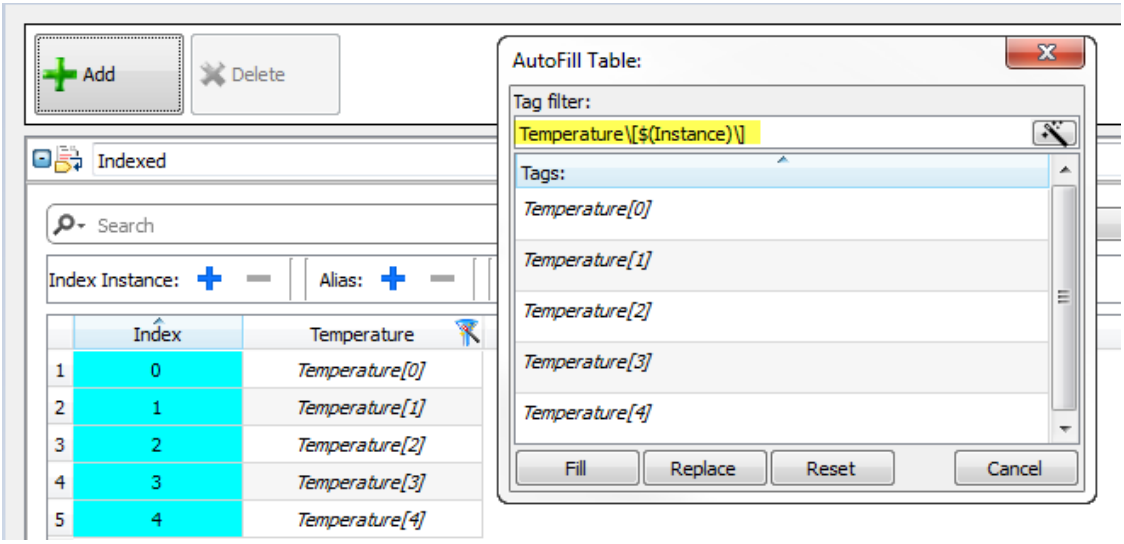
…

Éléments de tableau à remplissage automatique

Élément	Description
Remplir	Remplit les entrées manquantes dans le tableau de variables en utilisant le filtre défini (le cas échéant). Par exemple, lorsque de nouvelles instances ou de nouveaux alias sont ajoutés, vous pouvez utiliser cette option pour remplir les nouvelles entrées.
Remplacer	Remplace toutes les entrées du tableau par celles fournies par le tableau à remplissage automatique.
Réinitialiser	Réinitialise le filtre de variables pour vider ; aucun remplissage automatique n'est effectué.
	Suggère une expression de filtre valide pour votre projet.

Remarque : les filtres sont sauvegardés comme préférences de projet et peuvent être définis pour l'ensemble du tableau ou pour une colonne. Lorsque le filtre est défini pour une colonne, le filtre de tableau est ignoré. Ainsi, vous pouvez modifier de manière sélective le filtre pour gérer un seul alias en particulier.

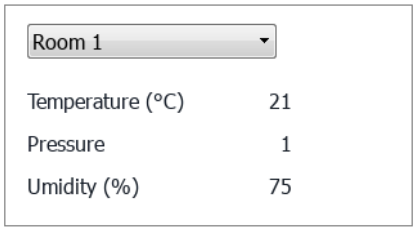
Remarque : pour référencer les éléments d'un tableau, utilisez le caractère \ pour désactiver l'interprétation de l'expression régulière des crochets (les variables de tableau sont différenciées en étant en italique).



Utilisation d’un ensemble de variables indexées dans des pages

Une fois qu'un jeu de variables indexées a été créé, vous pouvez l'utiliser pour créer une page pour le périphérique IHM comme dans cet exemple.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

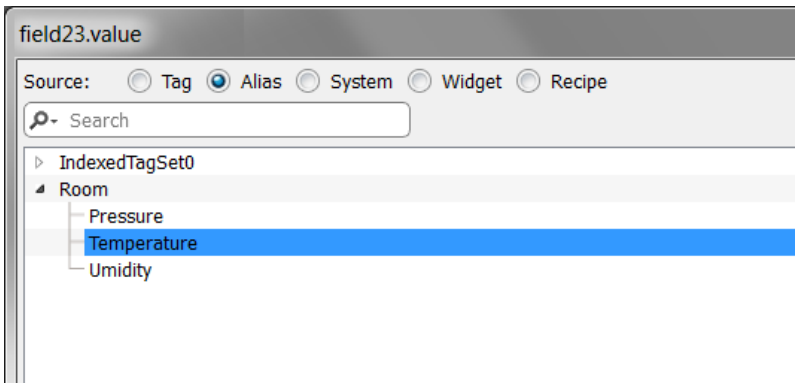


Pour créer cette page :

1. Créer une page et ajouter une liste déroulante, trois étiquettes et trois champs numériques.
2. Utiliser la variable d'index créé pour le numéro de la pièce pour la liste déroulante, « RoomNumber » dans cet exemple. Ce sera le sélecteur pour le numéro de la pièce.
3. Créer une liste pour la liste déroulante. Dans cet exemple, utiliser la liste suivante.

Index	Liste des chaînes
0	Numéro de pièce
1	Pièce 1
2	Pièce 2
3	Pièce 3
4	Pièce 4

4. Joindre à chaque valeur de champ numérique la variable Alias correspondant (**Pièce > Température, Pièce > Humidité, Pièce > Pression**).

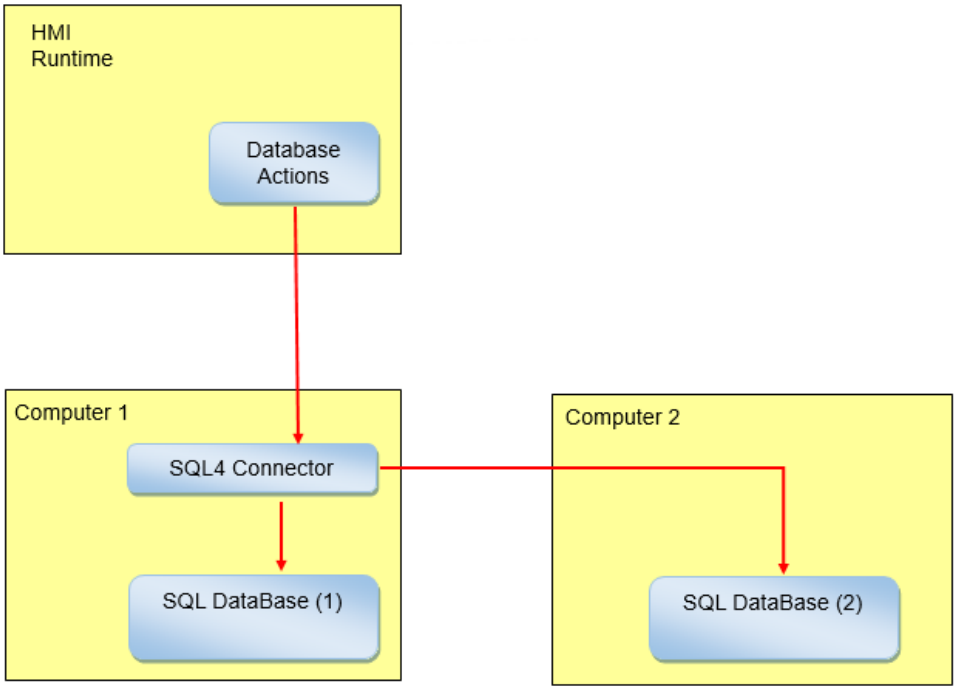


Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

32 Stocker les données dans des bases de données externes

PB610 Panel Builder 600 permet de se connecter à SQL4Automation Connector, solution logicielle pour un usage industriel. Il connecte l'IHM, l'API et les contrôles robotiques directement avec des base de données SQL. L'IHM accède directement aux bases de données SQL via le connecteur et peut interroger des données de tableaux, insérer, modifier et effacer des données dans des tableaux en utilisant les commandes SQL [langage de requêtes structuré].

Le site de base de données communique par ODBC. Par conséquent, vous pouvez intégrer toutes les bases de données SQL qui prennent en charge une interface ODBC. La syntaxe SQL doit être adaptée à la base de données concernée, par exemple MS SQL Server, mySQL, MS Office Access, SQLite, Oracle, PostgreSQL...



Pour stocker des données dans une base de données externe :

1. Installez l'outil SQL4Automation sur l'ordinateur qui héberge la base de données ou sur un ordinateur entre le périphérique IHM et la base de données.
2. Configurez l'outil SQL4Automation.
3. Créez un projet qui utilise les actions de base de données dédiées pour accéder à la base de données externe.

Installer SQL4Automation **343**

Configurer le module SQL4Automation **343**

Configurer le projet IHM **345**

Transférer les données avec JavaScript **346**

Tableaux de base de données **347**

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Tableaux personnalisés **348**
Limites de connexion **348**

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Installer SQL4Automation

Téléchargez la toute dernière version de SQL4automation et installez-la sur l'ordinateur. Reportez-vous à www.sql4automation.com pour les détails et le téléchargement.

Les procédures décrites dans ce document se réfèrent à SQL4Automation Connector Version 3.3.2.0

Configurer le module SQL4Automation

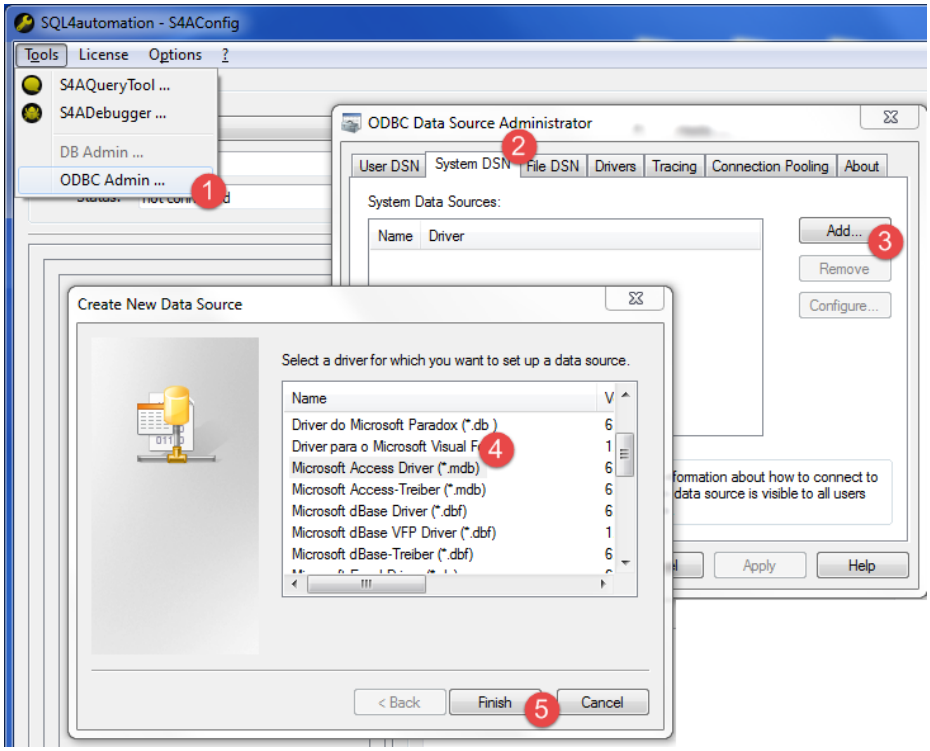
Important : référez-vous au manuel d'utilisation de SQL4Automation pour obtenir des instructions de configuration détaillées.

Il s'agit ici d'une rapide description de la manière d'accéder à une base de données MS Office (MS Access).

Vous devez avoir installé la suite MS Office sur un ordinateur et avoir créé une base de données vide en utilisant Microsoft Access.

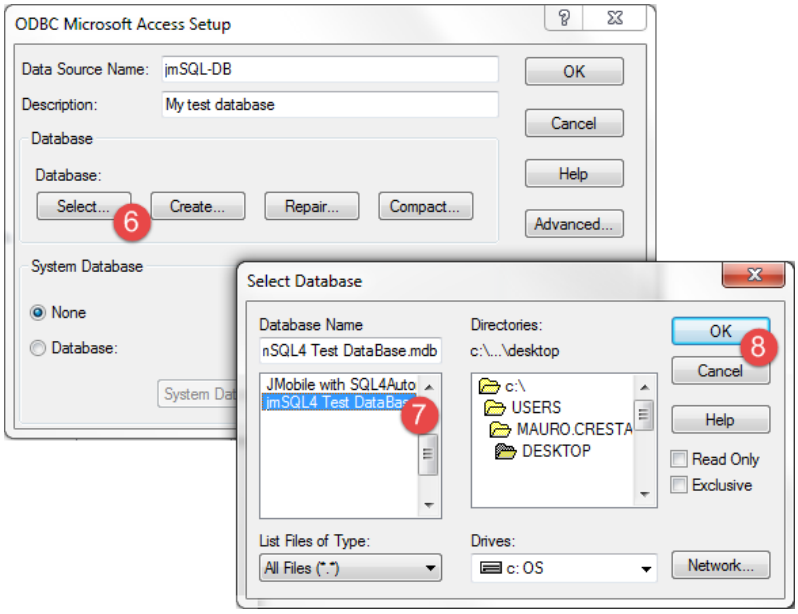
Démarrez SQL4Automation et suivez la procédure pour configurer votre connecteur SQL4Automation Connector :

1. Sélectionnez **Administrateur ODBC** : la boîte de dialogue **Administrateur de la source de données d'ODBC** s'affiche.
2. Sélectionnez l'onglet **Système DSN**.
3. Cliquez sur **Ajouter** : la boîte de dialogue **Créer une nouvelle source de données** s'affiche.
4. Sélectionnez le pilote Microsoft Access
5. Cliquez sur **Terminer** pour confirmer.

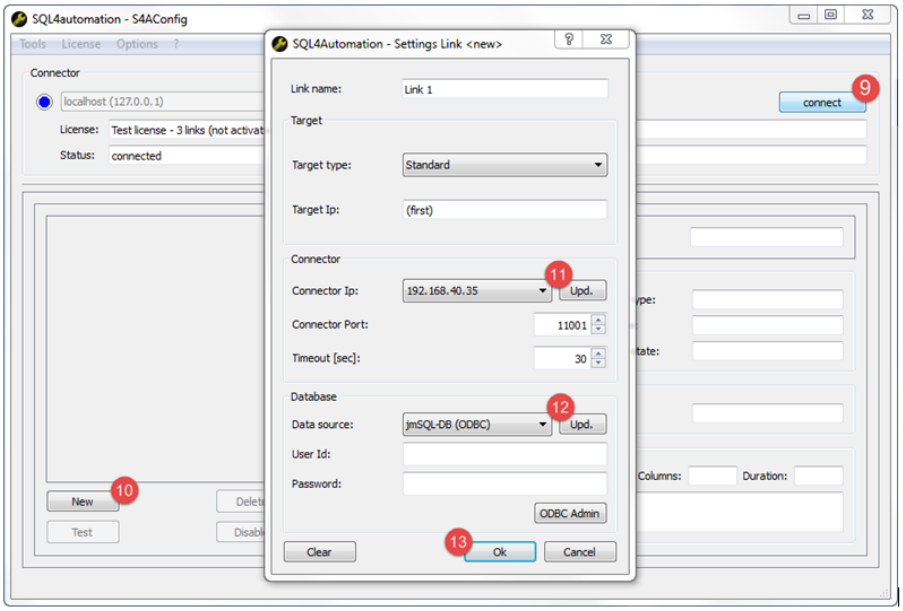


Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

6. Entez le **Nom de la source de données** et la **Description**, puis cliquez sur **Sélectionner** : la boîte de dialogue **Sélectionner la base de données** s'affiche.
7. Sélectionnez votre base de données Access.
8. Cliquez sur **OK** pour confirmer.

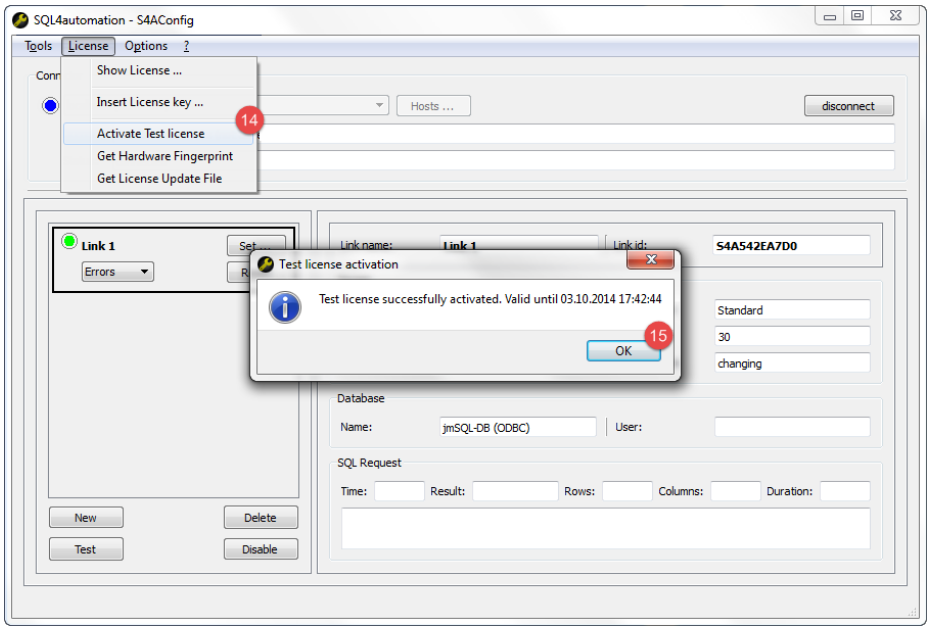


9. Lors de la première connexion, cliquez sur **Nouveau** pour sélectionner votre source de données
10. Sélectionnez l'adresse IP de votre ordinateur. Cette adresse IP de connexion sera utilisée par votre périphérique IHM.
11. Sélectionnez la source de données.
12. Cliquez sur **OK** pour confirmer.
13. Cliquez sur **Connecter**.



Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

14. Sélectionnez **Licence> Activer la licence test** : lorsque la LED **Link 1** devient verte, la procédure a été correctement exécutée.
15. Cliquez sur **OK** pour confirmer.

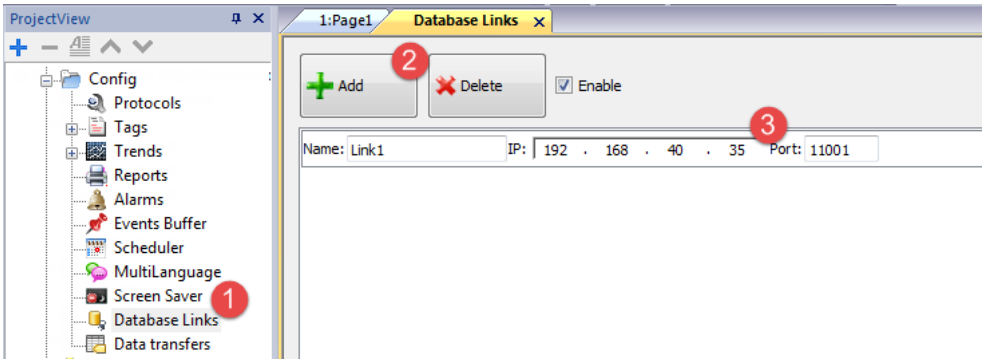


Configurer le projet IHM

Chemin : ***VueProjet*** *>* ***Configuration*** *> double-cliquez sur* ***Liens vers base de données***

Pour sauvegarder les données d'un projet dans une base de données externe, vous devez créer un lien avec la base de données concernée

1. Dans l'éditeur de **Liens vers base de données**, sélectionnez **Activer** pour activer cette fonction.
2. cliquez sur **Ajouter** pour créer un nouveau lien.
3. Entrez l'adresse IP sur l'ordinateur qui héberge le connecteur SQL4Automation Connector.



Important : le nom du lien n'est pas nécessairement le même que celui défini dans SQL4Automation Connector. Il s'agit cependant du nom à utiliser dans toutes les actions utilisant la base de données à distance.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Transférer les données avec JavaScript

Certaines actions utilisées pour transférer des données à partir d'un périphérique IHM à une base de données à distance peuvent être utilisées comme des macros à l'intérieur d'un code JavaScript comme dans l'exemple ci-dessous.

L'état de la connexion de base de données est disponible grâce à des variables du système. Voir "Variables bases de données" Page 98.

Le statut d'erreur peut être réinitialisé avec des actions. Voir "Actions de base de données" Page 141

```
fonction myButton1_onMouseClick(me, eventInfo) {
     var CustomSQL = '' ;
     var DatabaseLink ='Link1';
     project.dbInit(DatabaseLink, CustomSQL);
};

fonction myButton2_onMouseClick(me, eventInfo) {
     var CustomSQL = '' ;
     var DatabaseLink ='Link1';
     var Tags ='Alarm1;SystemTime;Tag01;Tag02;';
     project.dbReadTags(DatabaseLink, CustomSQL, Tags);
};

fonction myButton3_onMouseClick(me, eventInfo) {
     var CustomSQL = '' ;
     var DatabaseLink ='Link1';
     var Tags ='Alarm1;SystemTime;Tag01;Tag02;';
     project.dbWriteTags(DatabaseLink, CustomSQL, Tags);
};
```

dbQuery

```
project.dbQuery(databaseLink, customSQL, dbCallback);
```

En utilisant cette requête, vous pouvez exécuter les requêtes SQL.

Paramètre	Description
databaseLink	Lien à la base de données à utiliser
customSQL	Chaîne avec la requête SQL
dbCallback()	Fonction qui sera appelée lorsque les données de requête sont prêtes

dbCallBack

```
project.dbCallBack(dbStatus, dbResponse);
```


Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Paramètre	Description
dbStatus	0: no error found
dbResponse	Réponse à la requête. Noms de colonne de tableau suivi par ses lignes : Dans l'exemple : TagnName - Tagvalue Tag09 - 103 Tag10 - 302

Script

```
function JS1_onMouseClick(me, eventInfo) {

    var customSQL = "SELECT Tagname, Tagvalue FROM Tags WHERE Tagname='Tag09' OR Tagname='Tag10' ORDER BY Tagname"
    var databaseLink ="Link1";
    project.dbQuery(databaseLink, customSQL, dbCallback)
};


function dbCallback(dbStatus, dbResponse){

    alert("SQL Answer = " + dbResponse + "\ndbStatus =" + dbStatus);
};
```

Tableaux de base de données

Vous trouverez ci-dessous la structure des tableaux de base de données utilisés par les actions de base de données.

Remarque : ces tableaux peuvent être générés sur une base de données vide depuis l'action **DBInit**.

Tableau : variables

FieldName	Text(255)	CLÉ PRIMAIRE
TagValue	Text(255)	

Tendance de tableau

ID	Long entier	CLÉ PRIMAIRE
Nom de tendance	Text(255)	
SampleTime	Text(255)	
TrendValue	Text(255)	
Qualité	Text(255)	
RefreshTime	Text(255)	

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Tableau : recettes

Recette	Text(255)	CLÉ PRIMAIRE
SetName	Text(255)	CLÉ PRIMAIRE
ElementName	Text(255)	CLÉ PRIMAIRE
SetValue	Text(255)	

Tableau : événement

ID	Long entier	CLÉ PRIMAIRE
EventName	Text(255)	
SampledTime	Text(255)	
Type d'événement	Text(255)	
EventSubTime	Text(255)	
EventValue	Text(255)	

Tableaux personnalisés

Les requêtes SQL libérées des actions de bases de données sont listées dans le fichier de projet config\dbconnector.xml.

Modifiez les commandes définies dans le fichier pour personnaliser les chaînes SQL libérées des actions de bases de données puis accédez à une base de données structurée différente.

Exemple

```
CREATE TABLE myTagsTable (tagname VARCHAR(255) PRIMARY KEY, tagvalue VARCHAR(255))
UPDATE myTagsTable SET Tagvalue= '%_JMV' WHERE Tagname= '%_JMT'
INSERT INTO myTagsTable (Tagname, Tagvalue) Values ('%_JMT', '%_JMV')
```

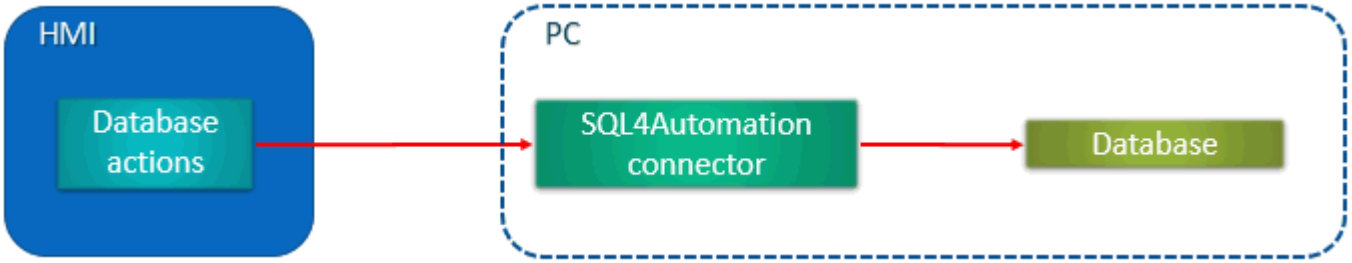
Où « %_JMV » est remplacé par la valeur de la variable et « %_JMT » par le nom de la variable.

Limites de connexion

SQL4Automation est livré comme une clé USB avec une licence pour un nombre prédéfini de connexions. Les connexions sont appelés « Liens » dans le SQL4Automation Connector. Le nombre de connexions dépend de la licence que vous avez achetée.

Vous pouvez installer SQL4Automation Connector sur le même ordinateur/serveur exécutant la base de données :

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



Ou sur un ordinateur/serveur séparé :

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

33 Serveur à architecture unifiée OPC

Chemin : ***VueProjet*** > ***Config*** > ***Interfaces*** > *double-cliquer sur* ***OPC UA***

Utilisez un serveur à architecture unifiée OPC pour publier les données selon le standard de l'architecture unifiée OPC.

Paramètre	Description
Enable OPC UA Server	Drapeau principal pour activer le serveur à architecture unifiée OPC. Les valeurs de données définies dans l'IHM sont publiées par le serveur à architecture unifiée OPC.

Fonctionnalités

Paramètre	Description
Activer alarmes	Active la publication des données d'alarme en temps réel (Alarmes actives).
Activer alarmes historiques	Active la publication des données de l'historique des alarmes.
Activer tendances	Active la publication des données de tendance.
Groupes de variables	Seules les variables appartenant aux groupes sélectionnés seront disponibles pour le serveur à architecture unifiée OPC.
Groupes d'alarme	Seules les alarmes appartenant aux groupes sélectionnés seront disponibles pour le serveur à architecture unifiée OPC.

Réseau

Paramètre	Description
Nom de nœud	Saisir le nom du nœud ou le laisser en blanc pour utiliser le nom de l'hôte.
Port	Le numéro de port du Serveur à architecture unifiée OPC. Le numéro de port proposé par défaut peut être différent du port utilisé par le Serveur à architecture unifiée OPC.

Authentification

Sélectionnez les options d'authentification au Serveur à architecture unifiée OPC.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Les clients du Serveur à architecture unifiée OPC auront la responsabilité de choisir, parmi les options disponibles, l'option la plus appropriée à utiliser en fonction de leurs capacités.

Authentification par identifiant

Paramètre	Description
Anonyme	Clients anonymes acceptés.
Identifiant/Mot de passe	L'authentification par identifiant est acceptée. Tout utilisateur valide a un accès illimité au Serveur à architecture unifiée OPC (voir"Configurer utilisateurs" Page 292).

Utilisation de certificats x.509

Le Serveur à architecture unifiée OPC fournit un canal de communication sécurisé à l'aide de certificats numériques. Des niveaux configurables de sécurité de bout en bout garantissent que le cryptage, la confidentialité et l'intégrité de chaque message sont disponibles. Lorsqu'il est activé, le serveur valide le certificat client et vice versa.

Les clients du Serveur à architecture unifiée OPC auront la responsabilité de choisir, parmi les options disponibles, l'option la plus appropriée à utiliser en fonction de leurs capacités.

Mode sécurité	Description
Aucun	Une connexion sans certificat est permise. Non recommandé pour des réseaux publics.
Signature	Le client du Serveur à architecture unifiée OPC doit fournir son propre certificat : les communications par message signé sont autorisées.
SignAndEncrypt	Le client du Serveur à architecture unifiée OPC doit fournir son propre certificat : les communications par message signé et crypté sont autorisées.

Politique de scurité	Description
Basic128Rsa15	Niveau de cryptage accepté (utilisé seulement quand le mode Sécurité est actif).
Basic256	Niveau de cryptage accepté (utilisé seulement quand le mode Sécurité est actif).
Basic256Sha256	Niveau de cryptage accepté (utilisé seulement quand le mode Sécurité est actif).

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Paramètre	Description
Certificats de confiance	Seuls les clients du Serveur à architecture unifiée OPC qui fournissent l'un des certificats listés sont acceptés. Pour ajouter un nouveau certificat à la liste des certificats de confiance, vous devez avoir le fichier de certificat fourni par le propriétaire du périphérique client du Serveur à architecture unifiée OPC. Les formats de fichier de certificat binaire et ASCII sont acceptés.

Serveur découverte globale

Le Serveur à architecture unifiée OPC est compatible avec le modèle GDS Push. Ceci signifie que vous pouvez utiliser un outil distant GDS pour la gestion centralisée des certificats.

Pour se connecter correctement au Serveur à architecture unifiée OPC, vous devez récupérer le certificat de l'outil GDS et l'ajouter à la liste des certificats de confiance du serveur en question.

Exemple

Lorsqu'un client du serveur OPC tente une connexion, le serveur vérifie si le certificat client est disponible dans sa propre liste de certificats de confiance. S'il n'est pas trouvé, la communication sera rejetée et le certificat sera stocké dans une liste de certificats non fiables.

A l'aide d'un outil GDS, vous pouvez vous connecter au serveur OPC, inspecter les certificats disponibles et définir le statut de confiance ou non de chaque certificat.

Fichiers de certificats

L'IHM stockera les certificats dans des sous-dossiers du dossier *"/workspace/<ApplicationName>/config/pkiserver"*

- own (personnel)
  Certificat personnel et clef privée
- trusted (fiable)
  Certificats de confiance auto-signés et certificats CA
- rejected
  Certificats rejetés
- issuers (émetteurs)
  CA intermédiaires de confiance (non directement fiables) nécessaires pour valider la chaîne de confiance

Identité de serveur

Paramètre	Description
Nom du fabricant	Nom lisible par l'homme du fabricant du produit. Le client OPC peut récupérer cette information à partir de la variable : *ServerName|Objects|Server|ServerStatus|BuildInfo|ManufacturerName*
Nom du produit	Un nom lisible par l'homme désignant le produit qui s'exécute dans le serveur. Le client OPC peut récupérer cette information à partir de la variable :

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Paramètre	Description
	ServerName|Objects|Server|ServerStatus|BuildInfo|ProductName

Paramètres de certificat

Le certificat du serveur peut soit être généré automatiquement, soit par l'ajout de fichiers de certificat existants.

Certificat auto-signé automatiquement généré

Si certificat généré automatiquement est activé, le certificat est régénéré après chaque modification apportée par l'utilisateur aux paramètres du certificat. Le certificat est également remplacé par tout certificat explicitement défini.

Paramètres de certificat

Chaque certificat doit contenir les informations permettant d'identifier le certificat et ses restrictions. Si vous avez choisi d'utiliser un certificat auto-généré, entrez les informations que vous souhaitez trouver dans le certificat auto-généré. Sinon, les paramètres sont lus à partir du certificat que vous avez fourni.

Paramètre	Description
Nom de serveur (Nom courant)	Nom du certificat (p. ex. nom du périphérique).
Organization	Nom de l'organisation
Unit	Unité dans l'organisation Ce champ peut être utile pour différencier les diverses divisions au sein d'une organisation.
Location (Localité)	Le champ Localité indique la ville où se trouve l'organisation
State (État)	Le champ État ou province indique l'emplacement physique de l'organisation. Le contenu du champ État ou Province ne doit pas être abrégé. Par exemple, CA n'est pas un nom d'état valide. Californie est le nom correct de l'État.
Country (Pays)	La norme de nommage X.509 exige un indicatif de pays à 2 caractères. L'indicatif de pays pour les États-Unis est US ; l'indicatif de pays pour l'Italie est IT.

Paramètre	Description
Produc URI	Un identifiant globalement unique pour le serveur. Exemple : "*urn:NodeName:CompanyName:ServerName*"
Noms DNS	Nom DNS ou adresse IP de l'appareil sur lequel ce serveur OPC est installé.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Paramètre	Description
Adresses IP	Plusieurs noms DNS et/ou adresses IP peuvent figurer dans un seul certificat. **Le certificat ne sera valide que si l'adresse IP à laquelle tourne le serveur OPC est incluse dans cette liste.**
Validity	Période de validité du certificat à partir de la date de création
Key Length (longueur de clef)	Longueur de la clef utilisée par l'algorithme de cryptage RSA

Script pour générer un certificat

Si vous voulez fournir votre propre certificat, notez que le certificat doit inclure le paramètre nom alternatif « Subject Alternative Name » (SAN) comme l'exige la norme sur les serveurs à architecture unifiée OPC.

Voici un exemple de la façon de générer un certificat à l'aide d'une bibliothèque publique OpenSSL-Win32 (Référence : https://www.openssl.org/)

```
@echo off
set OpenSSL="C:\Program Files (x86)\OpenSSL-Win32\bin\openssl.exe"
set NodeName=HMI-Server
set IPAddress=192.168.44.165

rem Generate an RSA key
    %OpenSSL% genrsa -out server-key.pem 2048

rem Creating Certificate Signing Requests
    %OpenSSL% req -new -key server-key.pem -out server.csr -subj
"/ST=Italy/C=IT/L=Verona/O=CompanyName/OU=R&D Team/CN=OPCUAServer@%NodeName%

rem Creating Certificate (.pem)
    echo subjectAltName=URI:urn:%NodeName%:CompanyName:OPCUAServer,IP:%IPAddress% >
san.txt
    echo
keyUsage=digitalSignature,nonRepudiation,keyEncipherment,dataEncipherment,keyCertSign
>> san.txt
    echo extendedKeyUsage=critical,serverAuth,clientAuth >> san.txt
    echo authorityKeyIdentifier=keyid,issuer >> san.txt
    echo basicConstraints=CA:TRUE >> san.txt
    %OpenSSL% x509 -req -days 3650 -in server.csr -signkey server-key.pem -out
server.crt -extfile san.txt

rem Convert Certificate (.der)
    %OpenSSL% x509 -in server.crt -outform der -out server.der

rem Not necessary files
    del san.txt

pause
```

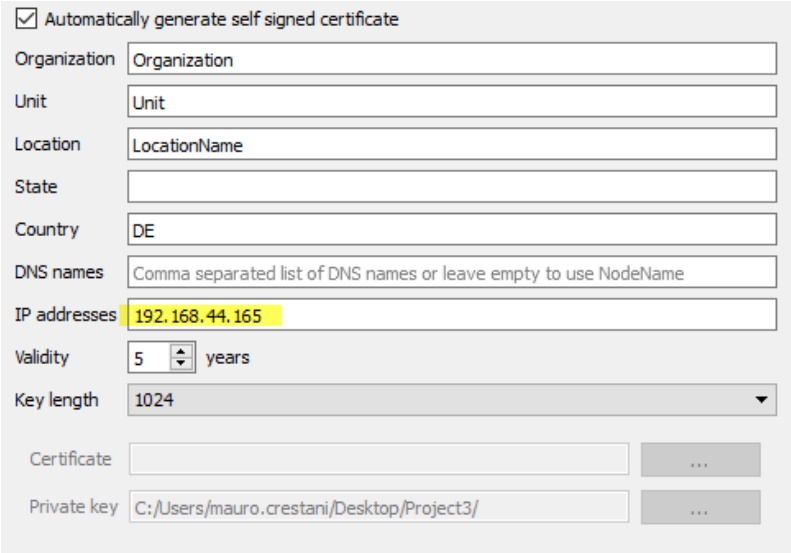
Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Utilisation de certificats auto-signés

Ce chapitre est un exemple étape par étape qui explique comment configurer deux périphériques IHM pour communiquer à l'aide de certificats auto-signés

Serveur à architecture unifiée OPC

1. Créer un projet simple comprenant quelques variables
2. Ouvrez la boîte de dialogue de l'OPC UA et activez le serveur OPC UA. Assurez-vous d'activer les groupes de variables (par exemple, sélectionnez Toutes).
3. Entrez dans le champ Adresses IP, l'adresse IP de l'IHM où le serveur OPC tournera

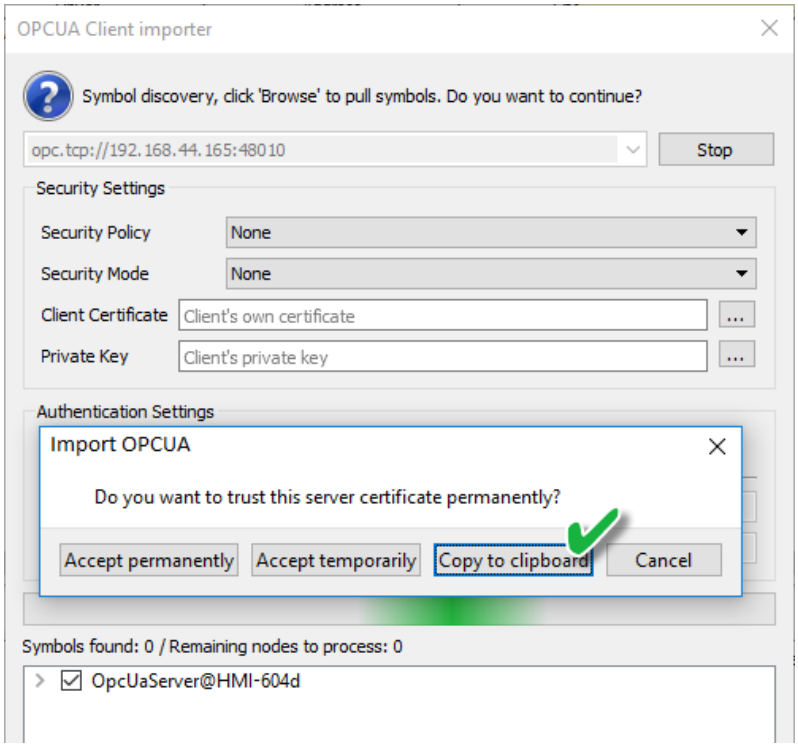


4. Téléchargez le projet vers le périphérique IHM

Client du serveur à architecture unifiée OPC

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

5. Créer un projet simple
6. Ajoutez le protocole Client de l'OPC. Entrez l'adresse IP du serveur OPC distant et son numéro de port (48010). Laissez les paramètres de certificat vides.
7. Ouvrez l'éditeur de variables et importez les variables. Sélectionnez le mode « OPC UA Discovery »
8. Choisissez de copier le certificat dans le presse-papiers comme indiqué sur la figure. Collez-le dans le champ « Certificat du serveur » de la boîte de dialogue configuration du protocole.



9. Répétez l'étape 7, acceptez le certificat du serveur OPC, importez des variables. Notez que vous pouvez accepter le certificat de façon permanente ou temporaire. Si vous acceptez le certificat de façon permanente, une copie du certificat est sauvegardée dans votre ordinateur pour une utilisation ultérieure sans vous demander de confirmation.
10. Ouvrez de nouveau la boîte de dialogue. Sélectionnez la politique de sécurité = Basic256 and Security Mode = SignAndEncrypt
11. Téléchargez le projet vers le périphérique IHM

Comme nous avons laissé vide le champ Certificat du serveur dans le client OPC, l'application a généré son propre certificat et l'a envoyé au serveur OPC UA mais comme le serveur ne connaît pas ce certificat, il rejette la demande de connexion. Maintenant, nous devons dire au serveur de faire confiance à ces certificats. Il y a différentes façons de le faire.

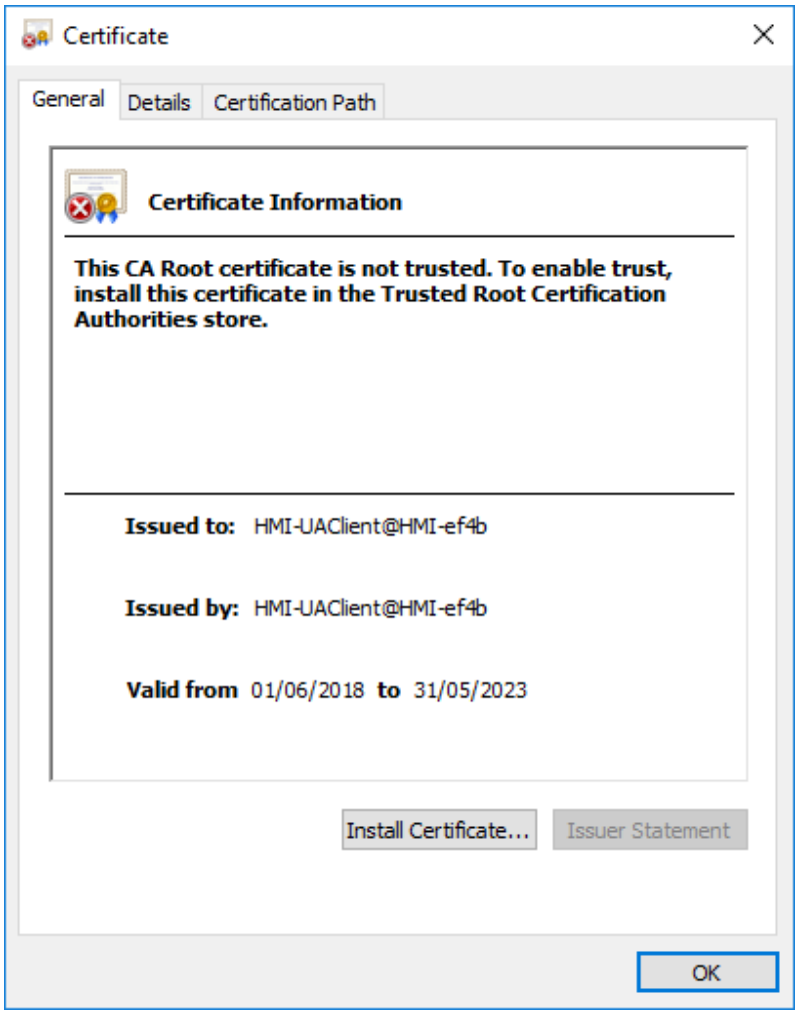
Faites en sorte que les certificats rejetés soient approuvés à l'aide d'un client FTP :

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

1. Connectez-vous au serveur OPC avec un client FTP.
2. Regardez dans les dossiers de certificats et déplacez le certificat rejeté du dossier « rejetés » vers le dossier de confiance.

   /workspace/<YourProjectName>/config/pkiserver/rejected
   /workspace/<YourProjectName>/config/pkiserver/trusted/certs

   Vous pouvez double-cliquer sur le fichier du certificat pour l'ouvrir et consulter les paramètres du certificat pour vous assurer que c'est bien le certificat que vous validez.



Maintenant la communication va démarrer

Faites en sorte que les certificats rejetés soient approuvés à l'aide d'un outil GDS :

1. Ouvrez l'outil GDS et exportez son certificat
2. Ouvrez le projet et ajoutez le certificat de l'outil GDS à la liste des certificats de confiance
3. Téléchargez les mises à jour du projet vers le périphérique IHM

Vous pouvez maintenant gérer les certificats à l'aide des outils de l'IHM.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Utilisation de certificats externes

Ce chapitre est un exemple étape par étape qui explique comment configurer deux périphériques IHM pour communiquer à l'aide de certificats externes.

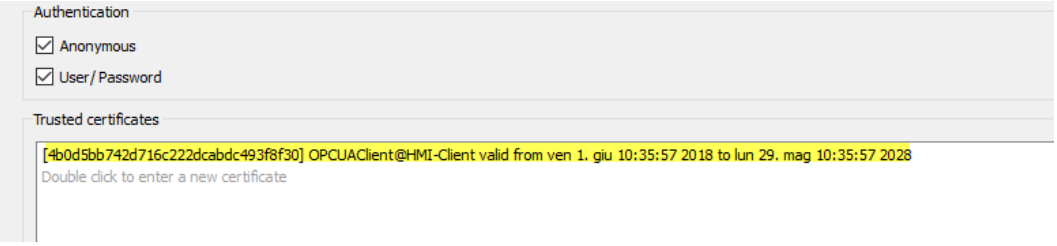
Générer des certificats

Vous pouvez utiliser le script donné dans ce manuel pour générer une copie de vos propres certificats, un pour le Serveur OPC UA et un autre pour le Client OPC UA.

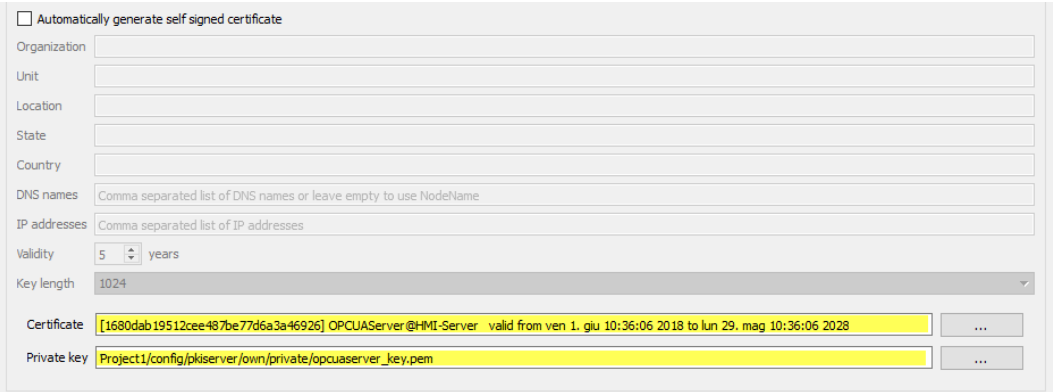
1. Installez une bibliothèque OpenSSL-Win32 (Référence : https://www.openssl.org/)
2. Utilisez le script ("Script pour générer un certificat" Page 355) pour générer le certificat du Serveur à architecture unifiée OPC. Assurez-vous de définir la variable IPAddress avec l'adresse IP du périphérique IHM à laquelle le serveur OPC tournera avant d'exécuter le script.
3. Repérez dans le manuel du protocole client de l'OPC un exemple de script pour générer un certificat pour le protocole client de l'OPC

Serveur à architecture unifiée OPC

4. Créer un projet simple avec quelques variables
5. Ouvrez la boîte de dialogue de l'OPC UA et activez le serveur OPC UA. Assurez-vous d'activer les groupes de variables (par exemple, sélectionnez Toutes).
6. Ajoutez le certificat client.der à la zone Certificat de confiance pour permettre au client OPC UA de communiquer avec le serveur OPC UA.



7. Décochez *Automatically generate self-signed certificate* et ajoutez le certificat (server.der) et la clef privée du certificat (server-key.pem)



8. Téléchargez le projet vers le périphérique IHM

Client du serveur à architecture unifiée OPC

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

9. Créer un projet simple
10. Ajoutez un protocole Client OPC UA.
11. Entrez l'adresse IP du serveur OPC UA distant et son numéro de port (48010).
12. Ouvrez la version ASCII du certificat du serveur (server.crt), supprimez tous les caractères de la Nouvelle ligne, puis copiez et collez les caractères ASCII de votre certificat dans le champ Certificat de serveur.
13. Faites de même avec le certificat Client (client.crt) et la clef privée Client (client-key.pem)
14. Sélectionnez la politique de sécurité Basic256 et le mode de sécurité = SignAndEncrypt
15. Ouvrez l'éditeur de variables et importez les variables. Sélectionnez le mode « OPC UA Discovery »
16. Acceptez le certificat du serveur OPC, importez des variables. Notez que vous pouvez accepter le certificat de façon permanente ou temporaire. Si vous acceptez le certificat de façon permanente, une copie du certificat est sauvegardée dans votre ordinateur pour une utilisation ultérieure sans vous demander de confirmation.
17. Téléchargez le projet vers le périphérique IHM

Vous remarquerez que le Client de l'OPC UA récupère les données du serveur OPC UA en utilisant les certificats donnés.

Carte des alarmes

Les états d'alarme sont associés aux états de l'architecture unifiée OPC selon les règles suivantes :

État alarme architecture unifiée OPC	PB610 Panel Builder 600 État alarme
Opcua.Alarm.Active	Vrai si le statut des alarmes est sur déclenché • TRIGGERED • TRIGGERED_NOT_ACKED • TRIGGERED_ACKED
Opcua.Alarm.Acked	Vrai si la confirmation des alarmes n'est pas requise • TRIGGERED_ACKED • NOT_TRIGGERED_ACKED • NOT_TRIGGERED
Opcua.Alarm.Retain	Vrai quand l'alarme est en attente • TRIGGERED | • TRIGGERED_NOT_ACKED | • TRIGGERED_ACKED | • NOT_TRIGGERED_ACKED mais une RÉINITIALISATION est requise
Opcua.Alarm.Confirmed	Vrai lorsque l'alarme est retournée (non déclenchée, acquittée et réinitialisée) • NOT_TRIGGERED Cette information n'est disponible que lorsque l'alarme est configurée avec une RÉINITIALISATION obligatoire

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

34 Widgets spéciaux

Les widgets conçus à des fins spéciales sont appelées widgets spéciaux et comprennent des listes de commandes, des widgets date et heure, des widgets de variables, etc.

Widget BACnet **362**

Widget de navigateur **362**

Widget Canevas **363**

Widget liste déroulante **367**

Widget compteur de consommation **371**

Widgets de liste de contrôle **373**

Widgets Date & Heure **374**

Widget de zone de mouvement **375**

Widgets Caméra IP **381**

Widget blocage de fonction Javascript **384**

Widgets de lecture de médias **386**

Widget d'image multi-états **389**

Widget multi-couche d'image multi-états **389**

Widget adaptateurs réseau **391**

Widget flux RSS **392**

Faire défiler le widget flux RSS **392**

Widget tableau **393**

Widget ÉditeurTexte **407**

Widget de variables **409**

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

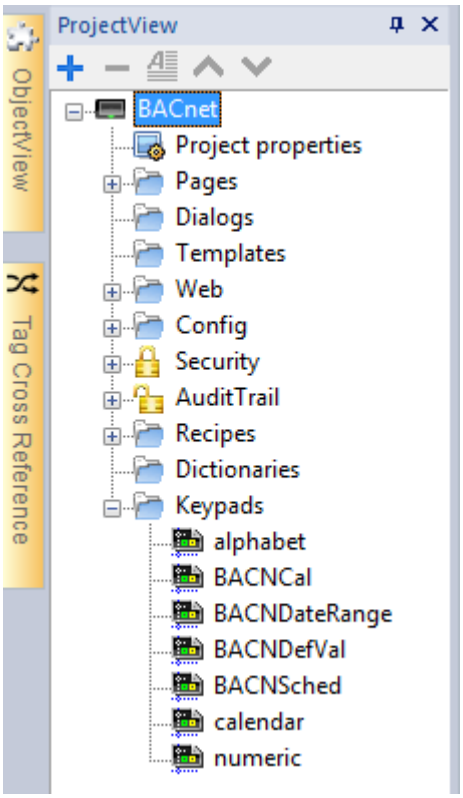
Widget BACnet

Chemin : **Galerie Widget> BACnet**

Les widgets BACnet sont des widgets spéciaux qui vous permettent d'interagir avec des objets BACnet natifs.

- Calendrier BACnet
- Planificateur BACnet
- Période effective BACnet

Ces widgets utilisent des claviers spéciaux qui sont ajoutés dans le dossier des claviers lorsque les widgets sont utilisés. En règle générale, vous ne devez considérer que ces claviers, sauf si vous voulez les personnaliser.



Reportez-vous au manuel BACnet dans le dossier « Pilotes de communication » pour une description détaillée des widgets spéciaux BACnet.

Widget de navigateur

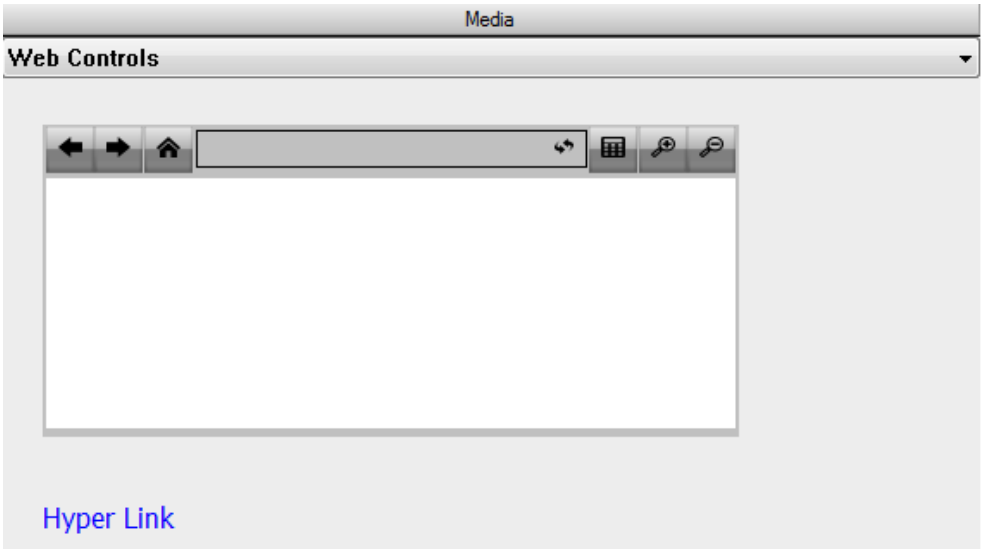
*Chemin : **Galerie des widgets**> **Média**> **Contrôles Web***

Utilisez ce widget pour intégrer des pages web aux pages de votre périphérique IHM. Il s'agit d'un widget de navigateur compatible HTML5 basé sur le moteur WebKit.

Remarque : la bibliothèque WebKit est disponible en tant que module d'exécution (plug-in) (voir"Module d'exécution (plug-in)" Page 75 pour les détails) à télécharger vers le HMI Runtime uniquement si nécessaire.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Important : les périphériques avec MIPS ne prennent pas en charge ce widget.



Paramètre	Description
Page d'accueil	URL à ouvrir par défaut lorsque le widget est affiché sur la page.
Zoom pour ajuster	Met automatiquement le contenu à l'échelle de la taille de la zone d'affichage.
Délai d'attente	Délai d'attente pour le chargement de la page en secondes.
Effacer l'historique	Efface automatiquement l'historique lors du chargement
Défilement	Affiche/masque les barres de défilement
Afficher curseur de progrès	Affiche/masque le curseur de chargement

Cela vous permet de gagner environ 3 MB de mémoire lorsque le widget n'est pas requis dans votre projet.

Un widget **Hyperlien** est disponible pour créer des hyperliens vers des pages. Lorsque vous cliquez sur ces liens, ils notifient au widget de navigateur qu'une page Web donnée va être chargée.

Important : le protocole HTTPs n'est pas pris en charge.

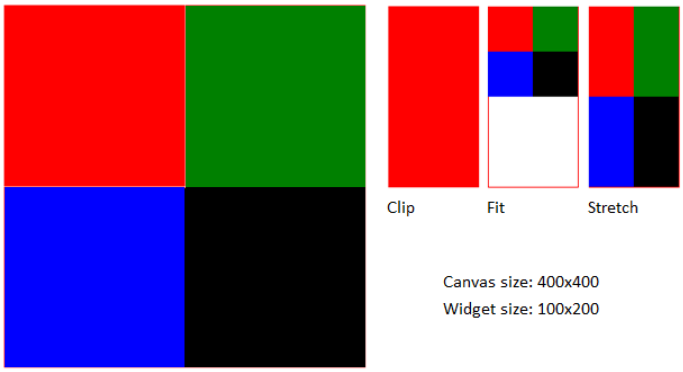
Widget Canevas

Chemin : **Galerie des widgets**> **Basique**> **Canevas générique**

Le Widget Canevas peut être utilisé pour dessiner le graphique via le script JavaScript.

Remarque : les méthodes JavaScript sont les mêmes qui sont disponibles pour la variable HTML5 <canevas>

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Paramètre	Description
Largeur de canevas **Hauteur de canevas**	Taille du canevas. Notez que ce n'est pas la taille du widget. Par exemple, la taille du canevas peut être 500 x 500 pixels lorsque la taille du widget serait de 100 x 100 pixels. Le paramètre Dessiner indice définit comment étirer la taille du canevas pour l'adapter à la taille du widget.
Dessiner indice	Définir comment adapter le canevas à l'intérieur de la taille du widget • Clip Aucune transformation n'est appliquée, le système de coordonnées n'est pas mis à l'échelle et le dessin est clipsé dans le rectangle englobant du widget. • Ajuster à la taille Ajuster à la taille du widget en préservant le rapport d'aspect du modèle de canevas. • Étirement Ajuster à la taille du widget en ignorant le rapport d'aspect du modèle de canevas. Exemple en utilisant une taille de canevas plus grande que la taille du widget : 
Aperçu du temps de design	Aperçu du canevas dans PB610 Panel Builder 600 Notez que le code JavaScript peut utiliser les données non disponibles dans PB610 Panel Builder 600 mais seulement dans le périphérique IHM
Effacer automatiquement l'arrière-plan	Efface automatiquement l'arrière-plan avant de dessiner le canevas. Lorsqu'il est désactivé, les éléments peints restent et il n'est pas nécessaire de tout redessiner.
Action LorsdeDessin	L'événement LorsdeDessin est exécuté lorsque la page est peinte. Cet événement doit être lié au code JavaScript qui dessine le graphique de canevas.
Action OnMousePress **Actions OnMouseRelease** **Action OnMouseDrag**	Événements de la souris

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Méthodes de canevas disponibles

// Enregistrer / restaurer peintre

- annuler enregistrement(); // appelle enregistrement peintre
- annuler restauration(); // appelle restauration peintre

// Échelle / Transformation

- annuler échelle(qreal x, qreal y);
- annuler rotation(qreal angle);
- annuler translation(qreal x, qreal y);
- annuler transformation(qreal m11, qreal m12, qreal m21, qreal m22, qreal dx, qreal dy);
- annuler setTransform(qreal m11, qreal m12, qreal m21, qreal m22, qreal dx, qreal dy);

// Gradient

- CanvasGradient createLinearGradient(qreal x0, qreal y0, qreal x1, qreal y1);
- CanvasGradient createRadialGradient(qreal x0, qreal y0, qreal r0, qreal x1, qreal y1, qreal r1);

// Fonctions Rectangle

- annuler clearRect(qreal x, qreal y, qreal w, qreal h);
- annuler fillRect(qreal x, qreal y, qreal w, qreal h);
- annuler strokeRect(qreal x, qreal y, qreal w, qreal h);
- annuler rect(qreal x, qreal y, qreal w, qreal h);

// Chemin

- annuler beginPath();
- annuler closePath();
- annuler moveTo(qreal x, qreal y);
- annuler lineTo(qreal x, qreal y);
- annuler quadraticCurveTo(qreal cpx, qreal cpy, qreal x, qreal y);
- annuler bezierCurveTo(qreal cp1x, qreal cp1y, qreal cp2x, qreal cp2y, qreal x, qreal y);

// Texte dessin

- annuler fillText(const QString &text,qreal x, qreal y);

// Arc

- annuler arcTo(qreal x1, qreal y1, qreal x2, qreal y2, qreal radius);
- annuler arc(qreal x, qreal y, qreal radius, qreal startAngle, qreal endAngle, bool anticlockwise);

// Remplissage/course

- annuler remplissage();
- annuler stroke();
- annuler clip();
- bool isPointInPath(qreal x, qreal y) const;

// Manipulation de l’image (Dessiner CImageWgt en utilisant la rectification cible et source)

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

- Annuler drawImage(QObject *pObjImage, qreal sx, qreal sy,qreal sw, qreal sh, qreal dx, qreal dy,qreal dw, qreal dh);
- Annuler drawImage(QObject *pObjImage, qreal dx, qreal dy);
- Annuler drawImage(QObject *pObjImage, qreal dx, qreal dy, qreal dw, qreal dh);
- annuler drawImage(const QVariant& image, int width, int height, const QString& format, qreal sx, qreal sy,qreal sw, qreal sh, qreal dx, qreal dy,qreal dw, qreal dh);

// manipulation des pixels

- ImageData createImageData(double sw, double sh);//Empty Image
- ImageData createImageData(ImageData fromImage);//from another Image
- ImageData createImageData(ArrayBuffer value); //From arraybuffer
- annuler putImageData(ImageData imgData,double dx, double dy);
- annuler putImageData(ImageData imagedata, double dx, double dy, double dirtyX, double dirtyY, double dirtyWidth, double dirtyHeight);
- ImageData getImageData(qreal sx, qreal sy, qreal sw, qreal sh);

Exemple JavaScript Canevas

Le canevas est d'abord vierge. Pour afficher quelque chose, un script doit d'abord accéder au contexte de rendu et y dessiner :

```
var ctx = me.context2d;
```

alors vous pouvez utiliser les méthodes de toile, comme pour l'exemple ci-dessous

```
function GenericCanvasWgt1_onDraw(me, eventInfo)
{
    var ctx = me.context2d;
    ctx.fillStyle = 'red';
    ctx.fillRect(0,0,250,250);
    ctx.fillStyle = 'green';
    ctx.fillRect(250,0,250,250);
    ctx.fillStyle = 'blue';
    ctx.fillRect(0,250,250,250);
    ctx.fillStyle = 'black';
    ctx.fillRect(250,250,250,250);
}

function GenericCanvasWgt1_onMouseDown(me, eventInfo)
{
    alert("X = " + eventInfo.posX + "\nY = " + eventInfo.posY );
}
```

La méthode de mise à jour peut être utilisée pour redessiner dynamiquement un widget canevas

```
function BtnStd1_btn_onMouseClick(me, eventInfo)
{
    var myCanvasWidget = page.getWidget("GenericCanvasWgt1");
```

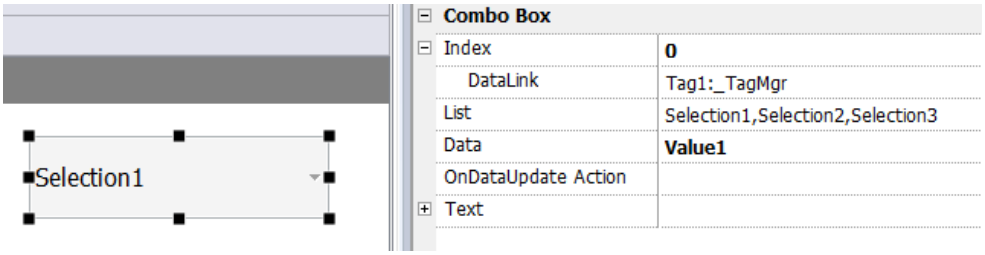

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

```
    myCanvasWidget.update()
}
```

Widget liste déroulante

Chemin : ***Galerie des widgets*** *>* ***Basique*** *>* ***Contrôles***

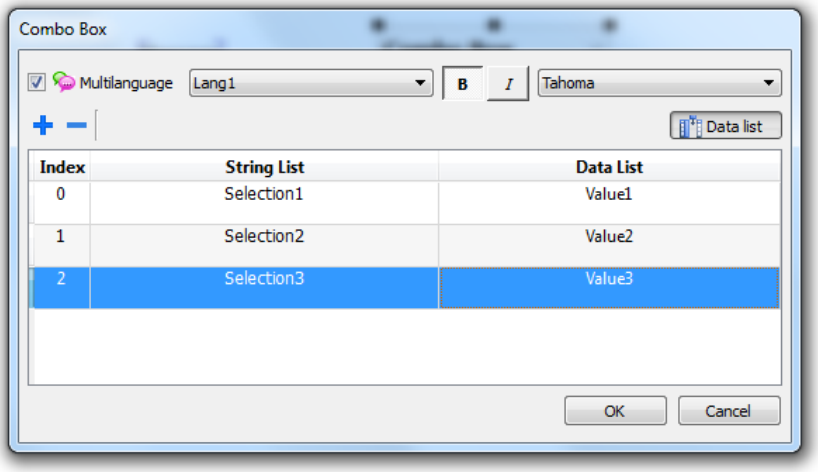
Utilisez ce widget comme widget de sélection ou pour filtrer les lignes dans un tableau, afin d'afficher uniquement les valeurs sélectionnées dans la liste déroulante.



Paramètre	Description
Index	Index de l'élément sélectionné.
Liste/Liste de chaîne	Chaînes d'élément dans la liste déroulante. Remarque : ce champ est multilangue.
Données/Liste de données	Retourne la valeur de la colonne Liste de données (comme chaîne) dans le champ Données du widget. Conseil : utilisez ce paramètre pour retourner une valeur personnalisée basée sur un élément sélectionné dans la liste déroulante.
Texte	Format du texte affiché.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Associer des données VS associer des index



Dans de nombreux projets, vous pouvez être amené à associer des champs tels que **Index** ou **Données** à des variables pour connaître les valeurs de l'élément sélectionné dans la liste déroulante. Utilisez :

- **Index** : pour afficher l'index (entier) de l'élément sélectionné (0...n).
- **Données** : pour afficher la valeur des données (chaîne) spécifiée dans la colonne Liste des données.

Mode « plein écran » avec images du widget à liste déroulante

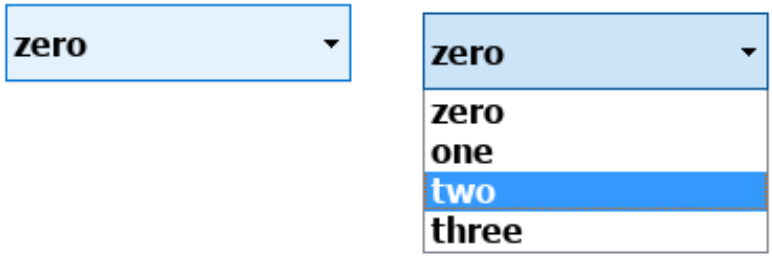
Depuis les "Propriétés de projet" Page 69, l'apparence et le comportement des listes déroulantes du mode Contextuel au mode Plein Écran

Chemin : **VueProjet**> double-cliquez sur **Propriétés du projet**> **Panneau de propriétés**> **Projet**> **Mode de vue de la liste déroulante**

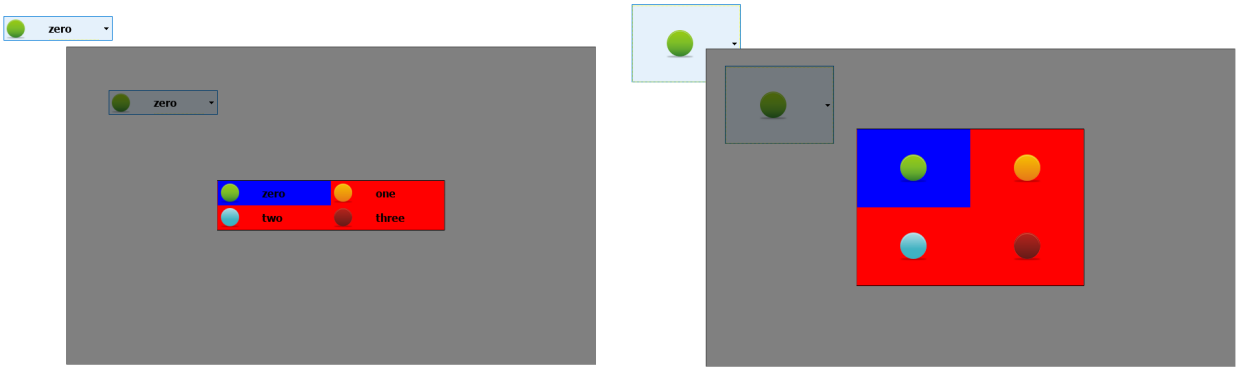
Paramètre	Description
Mode de vue en liste déroulante	Sélectionnez le mode de visualisation de tous les widgets de liste déroulante du projet **Contexte** Vue classique avec les menus déroulants **Plein écran** Vue améliorée avec textes et images configurables qui apparaîtront au milieu de l'écran pour faire défiler facilement et sélectionner.

Exemple de vue **Contextuel**

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



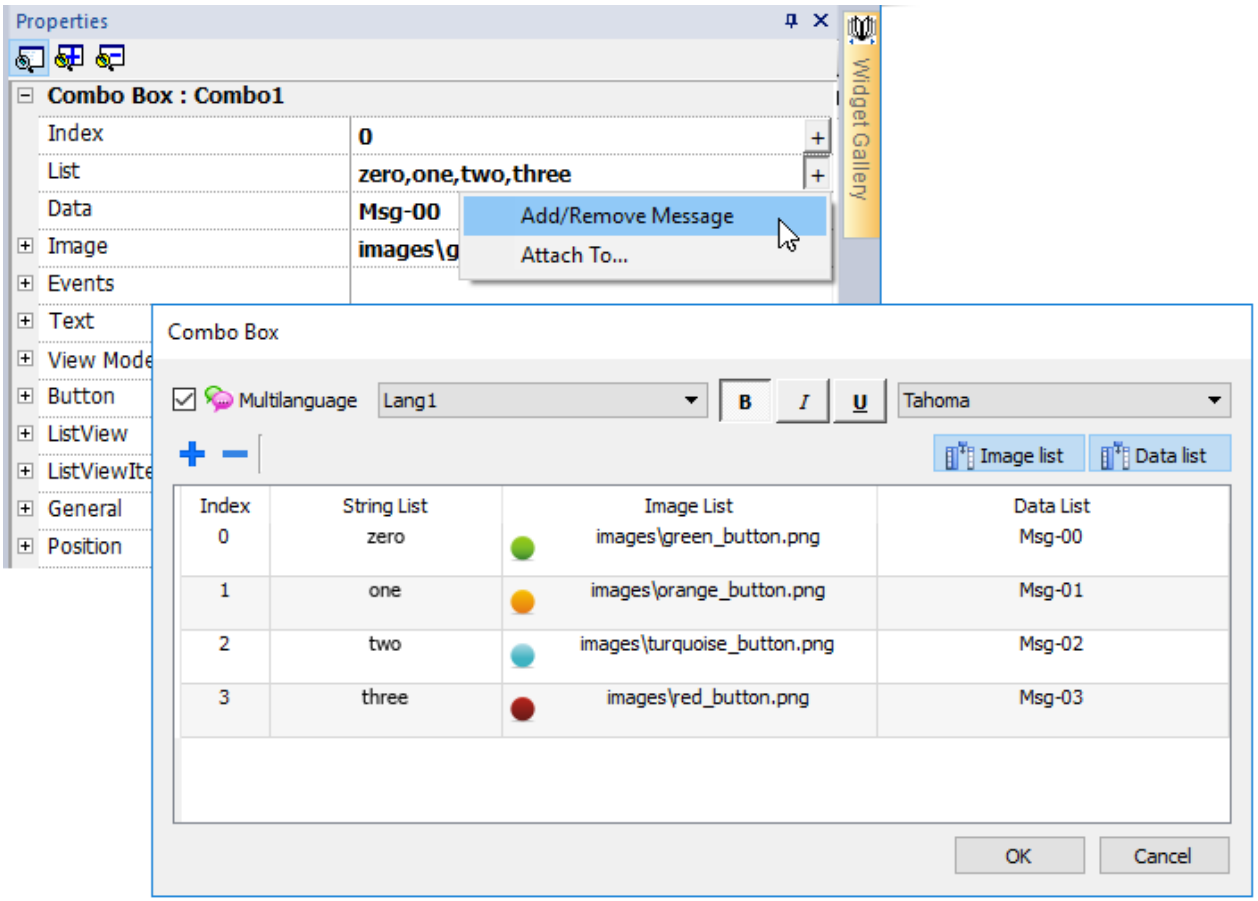
Exemple de vue **Plein écran**



Paramètres supplémentaires disponibles en mode plein écran

La colonne supplémentaire « *Liste des images* » sera disponible dans le paramètre **Liste déroulante**> **List**e :

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



Remarque : certaines propriétés ne s'affichent qu'en mode avancé.

Paramètre	Description
Image	Retour, à l'intérieur de la variable attachée, le nom du fichier de l'image sélectionnée
Bouton	Définir l'apparence de la liste déroulante • Afficher l'arrière-plan = vrai Bouton liste déroulante affiché • Afficher l'arrière-plan = faux Seul l'image ou le texte est affiché
VueListes	Paramètres de mise en page de la liste déroulante en mode édition
ÉlémentsVueListe	Définir le type d'éléments à l'intérieur de la liste déroulante Mode Image : • Seul texte • Seules images • Texte et images

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

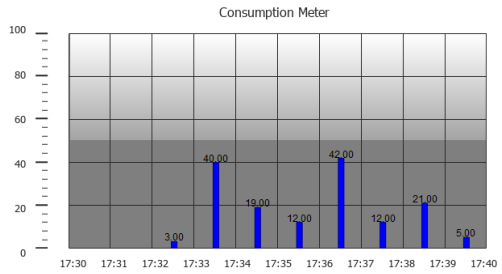
Widget compteur de consommation

*Chemin : **Galerie des widgets**> **Basique**> **Tendances/Graphiques***

Utilisez ce widget pour gérer une ressource qui est en constante hausse. Le système lit la valeur de la ressource et calcule l'incrément dans un ensemble de plages de temps. L'incrément est ensuite affiché dans un histogramme dans une fenêtre de type de tendance.

Différentes couleurs peuvent être utilisées dans le graphique, selon le délai d'exécution.

Conseil : utilisez ce widget pour calculer la consommation d'énergie d'un système.



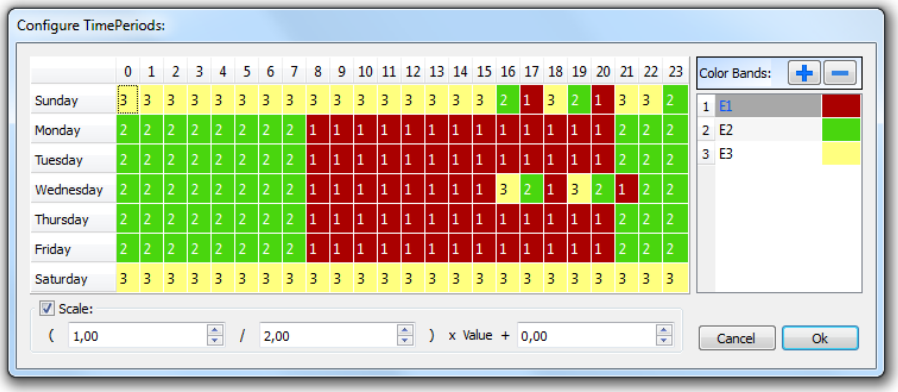
Paramètre	Description
Valeur	Ressource surveillée
Durée du graphique **Unités de durée du graphique**	Période de temps affichée dans la fenêtre
Durée de la barre **Unités de durée des barres**	Période de temps représentée par chaque barre du graphique
Périodes de temps	Assigne une couleur spécifique pour surligner l'incrément de la ressource surveillée dans une période de temps donnée (résolution minimale = 1 heure).
Couleur **Largeur de barre**	Couleur et largeur de la barre
Valeur de la barre	Afficher / Masquer la valeur de chaque barre
Compteur de consommation	Nombre d'étiquettes à afficher dans le graphique.

Exemple : surveiller la consommation d'énergie

Dans l'exemple suivant, un widget est conçu pour surveiller la consommation d'énergie avec une échelle hebdomadaire et une unité quotidienne.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

1. Associez une variable à la variable physique à surveiller. Dans cet exemple, l'énergie totale consommée (variable KWh). Cette variable contient un nombre incrémentiel qui indique la consommation en KW/h depuis le début de la consommation d'énergie.
2. Ajoutez une tendance et liez-la à la variable à surveiller, la variable KWh.
3. Ajoutez un widget **Compteur de consommation** à une page.
4. Associez la propriété **Valeur** du compteur de consommation à la tendance créée à l'étape 2.
5. Définissez **Durée du graphique/Unités** à 1 semaine : vous obtiendrez un graphique hebdomadaire de l'énergie consommée.
6. Définissez **Durée de la barre/Unités** à 1 jour. C'est l'intervalle de temps pendant lequel l'énergie consommée est calculée.
7. Dans **Compteur de consommation**, définissez le nombre d'étiquettes à afficher dans l'histogramme, dans le cas présent 7, pour afficher un graphique hebdomadaire.
8. Depuis la propriété **Périodes de temps**, ouvrez la boîte de dialogue **Configurer les périodes de temps** : définissez les différentes couleurs pour différentes valeurs de la variable KWh dans chaque barre.



Conseil : pour assigner la couleur aux cellules du tableau, sélectionnez les cellules et cliquez sur la couleur voulue, ou entrez la valeur d'index de la bande (1, 2, 3) dans la cellule.

9. Ajoutez autant de couleurs de bandes que nécessaire, dans cet exemple 3 couleurs de bandes.
10. Assignez une bande à chaque heure dans le tableau hebdomadaire, dans cet exemple une bande rouge (E1), qui indique la plage de temps dans la journée/semaine pendant laquelle le coût de l'énergie est le plus élevé.

Remarque : vous pouvez appliquer un facteur d'échelle à chaque bande de couleur, si nécessaire.

Il en résulte un histogramme représentant le compteur de consommation et affichant la consommation d'énergie en KW/h, avec des couleurs qui indiquent les différents coûts de l'énergie. La hauteur de chaque barre représente la quantité d'énergie dans l'intervalle de temps concerné, 1 jour dans cet exemple.

Utilisez l'action ConsumptionMeterPageScroll pour faire défiler l'histogramme en avant et en arrière, ainsi que l'action RefreshTrend pour rafraîchir l'histogramme car les données ne sont pas rafraîchies automatiquement.

Important : aucune autre action de tendance n'est actuellement prise en charge par le widget du compteur de consommation.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

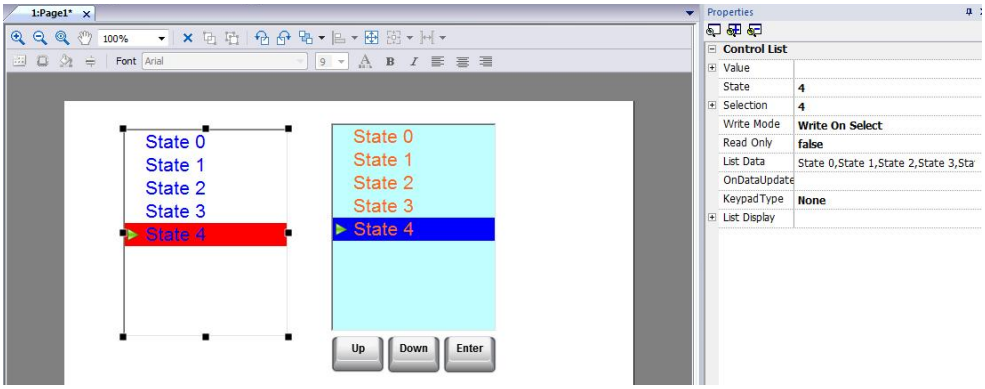
Widgets de liste de contrôle

*Chemin : **Galerie des widgets**> **Avancé**> **Liste de contrôle***

Utilisez ces widgets pour représenter l'état associé à un processus particulier et pour contrôler ce processus depuis le même widget.

Deux types de listes de contrôle sont disponibles :

- une liste de contrôle de groupe, avec un ensemble limité de boutons de navigation qui sont déjà inclus, et
- une liste de contrôle basique avec aucun bouton configuré au préalable, sur laquelle vous pouvez naviguer en utilisant la fonction écran tactile.



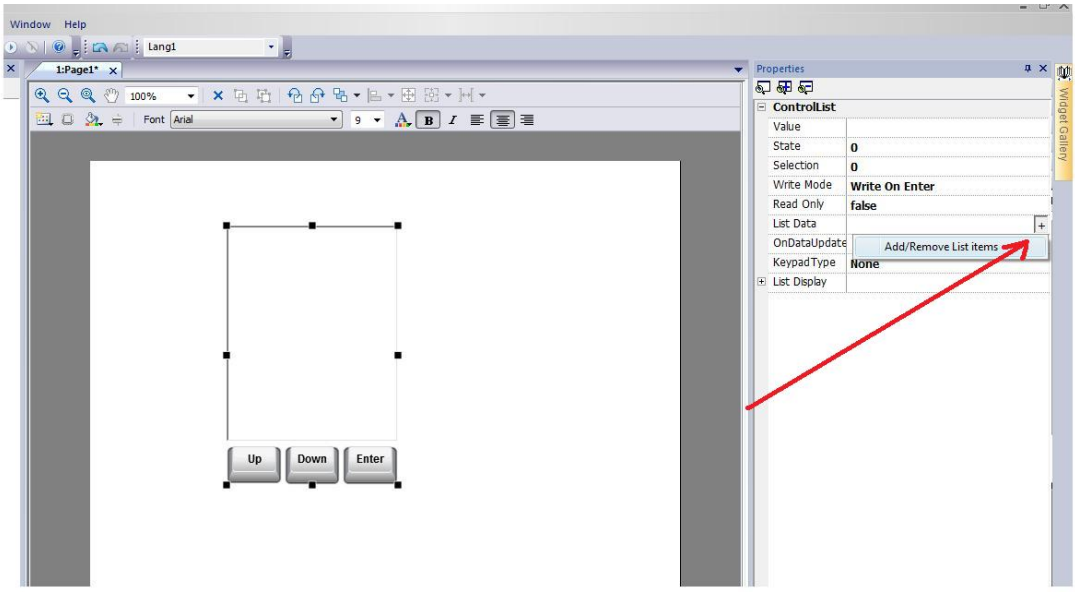
Paramètre	Description
Valeur	Si le **Mode d'écriture** est **Écrire après sélection** : valeur de l'élément sélectionné. Si le **Mode d'écriture** est **Écrire après entrée** : valeur de l'élément sélectionné et confirmé en appuyant sur le bouton entrée. Ce champ peut être associé à une variable pour contrôler l'élément sélectionné et confirmé.
État	État par défaut lorsque le widget est chargé.
Sélection	Élément actuellement sélectionné, affiché comme un curseur surligné se déplaçant de haut en bas. Cette propriété peut être associée à une variable.
Mode écriture	**Écrire après sélection** : la valeur est automatiquement écrite sur la variable lorsque vous sélectionnez un des éléments. **Écrire après entrée** : la valeur est écrite sur la variable uniquement lorsque vous sélectionnez un des éléments et que vous appuyez sur la touche entrée.
Lecture seule	Définit si la liste est uniquement un indicateur.
Liste de données	Ajoute/supprime des éléments de la liste.

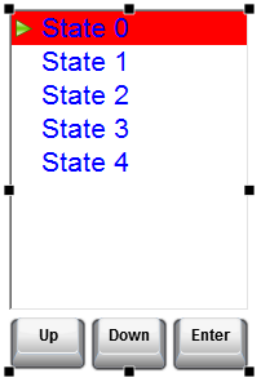
Définir les états

Ajoute/supprime des états, c'est-à-dire des éléments de la liste, depuis la propriété **Liste de données**.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Toute valeur peut être assignée à un état. Lorsque vous activez l'état, en sélectionnant l'élément associé si vous êtes en mode **Écrire après sélection** ou bien en le sélectionnant et en confirmant en appuyant sur entrée si vous êtes en mode **Écrire après entrée**, la valeur associée à un état sera écrite sur la variable liée au widget de contrôle de la liste **Valeur** .





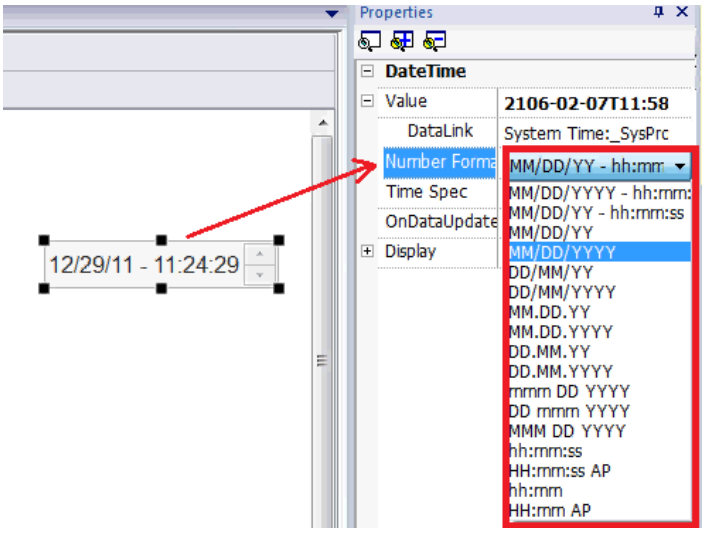
Widgets Date & Heure

Chemin : ***Galerie des widgets*** *>* ***Basique*** *>* ***Contrôles***

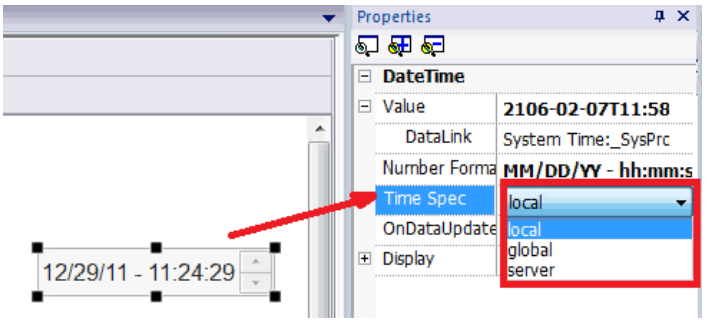
Utilisez ce widget pour afficher et éditer la date et l'heure actuelles.

Dans le volet **Propriétés**, différents formats sont disponibles pour représenter la date et l'heure.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



Pour la propriété **Spéc heure**, sélectionnez quelle heure le widget doit afficher au runtime.



Options pour l'heure

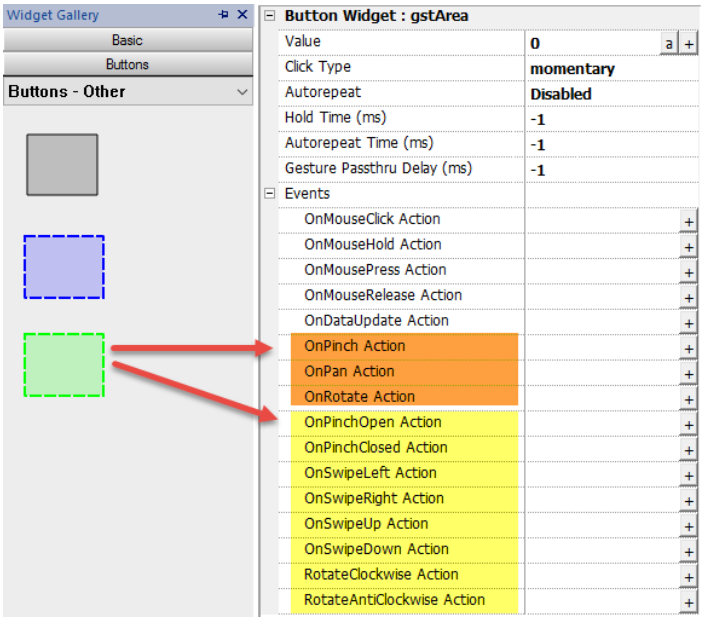
Option	Description
local	affiche l'heure locale, l'heure du périphérique IHM où le projet est exécuté
GMT	affiche l'heure de Greenwich (GMT)
serveur	affiche les informations relatives à l'heure telles que gérées au niveau du serveur du périphérique IHM

Widget de zone de mouvement

*Chemin : **Galerie des widgets**> **Boutons**> **Autres***

Le widget de zone de mouvement est un bouton à zone réactive qui génère des événements gestuels.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



Événements gestuels	Description
BalayageGauche **BalayageDroite**	Un événement est déclenché lorsqu'un geste de balayage (swipe) est détecté
PincementOuvert **PincementFermé**	Un événement est déclenché lorsqu'un geste de pincement (pinch) est détecté
RotationSensHoraire **RotationSensAntihoraire**	Un événement est déclenché lorsqu'un geste de rotation est détecté
OnPan (Panoramique) **OnPinch (Pincement)** **OnRotate (Rotation)**	Une série d'événements déclenchés pendant le geste. Seul JavaScript peut être utilisé pour gérer ces événements, grâce au code JavaScript, le développeur peut gérer les événements des gestes comme il le souhaite. **AVERTISSEMENT : seuls les dispositifs IHM multi-touch peuvent générer des événements de pincement**

OnPan (Panoramique)

```
boolean onGesturePan (me, eventInfo)
```

Cet événement se produit lorsqu'un point à l'intérieur de la zone a été appuyé et qu'un mouvement a été détecté.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Paramètre	Description
me	Objet déclencheur de l'événement.
eventInfo	**id** = ID Geste ; utilisé pour identifier les différents gestes. **running** = Vrai, sauf pour le dernier événement livré pour informer l'achèvement du geste. **dx** = Total de mouvements de l'axe X en unités de pixels de l'écran depuis la position initiale de contact. **dy** = Total de mouvements de l'axe Y en unités de pixels de l'écran depuis la position initiale de contact.

OnPinch (Pincement)

```
boolean onGesturePinch(me, eventInfo)
```

Cet événement se produit lorsque 2 points à l'intérieur de la zone ont été appuyés et qu'un mouvement a été détecté.

Paramètre	Description
me	Objet déclencheur de l'événement
eventInfo	**id** = ID Geste ; utilisé pour identifier les différents gestes. **running** = Vrai, sauf pour le dernier événement livré pour informer l'achèvement du geste. **dx** = Total de mouvements de l'axe X en unités de pixels de l'écran depuis la position initiale de contact. La valeur positive signifie que la distance augmente ; La valeur négative signifie que la distance diminue. Cette quantité peut être utilisée pour contrôler une valeur de zoom. **dy** = Total de mouvements de l'axe Y en unités de pixels (voir dx).

OnRotate (Rotation)

```
boolean onGesturePinch(me, eventInfo)
```

Cet événement se produit lorsque 2 points à l'intérieur de la zone ont été appuyés et qu'un mouvement de rotation a été détecté.

Paramètre	Description
me	Objet déclencheur de l'événement
eventInfo	**id** = ID Geste ; utilisé pour identifier les différents gestes. **running** = Vrai, sauf pour le dernier événement livré pour informer l'achèvement du geste. **drot** = combien de degrés (0/360) ont été ajoutés depuis le dernier événement. **trot** = nombre total de degrés (0/360) du mouvement en entier.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Paramètre	Description
	Les nombres positifs signifient une rotation dans le sens horaire, les nombres négatifs signifient une rotation dans le sens antihoraire.

Transit d'événements gestuels

Pour utiliser un widget (par exemple un bouton ou un curseur) recouvert d'un objet gestuel, vous devez maintenir enfoncé le widget 200 ms pour déplacer la commande vers l'objet sous-jacent. Le temps d'attente pour envoyer la commande vers l'objet sous-jacent peut être modifié à partir du paramètre « Délai de transit de geste » disponible dans la vue des propriétés avancées.

Paramètre	Description
Transit de mouvement activé	Active la possibilité de passer des événements gestuels aux widgets sous-jacents après un délai configurable. L'utilisateur doit maintenir le doigt appuyé puis exécuter le geste. **défaut** = utilise la valeur définie dans les propriétés du projet. Voir "Projet" Page 76 **vrai** = transit de mouvement activé **faux** = transit de mouvement désactivé
Délai de transit de geste (ms)	Le temps qu'il faut attendre pour envoyer la commande à l'objet sous-jacent 0/500 ms -1 Utilise la valeur définie dans les propriétés du projet. Voir "Projet" Page 76

Exemples d'utilisation d'événements gestuels en association avec JavaScript

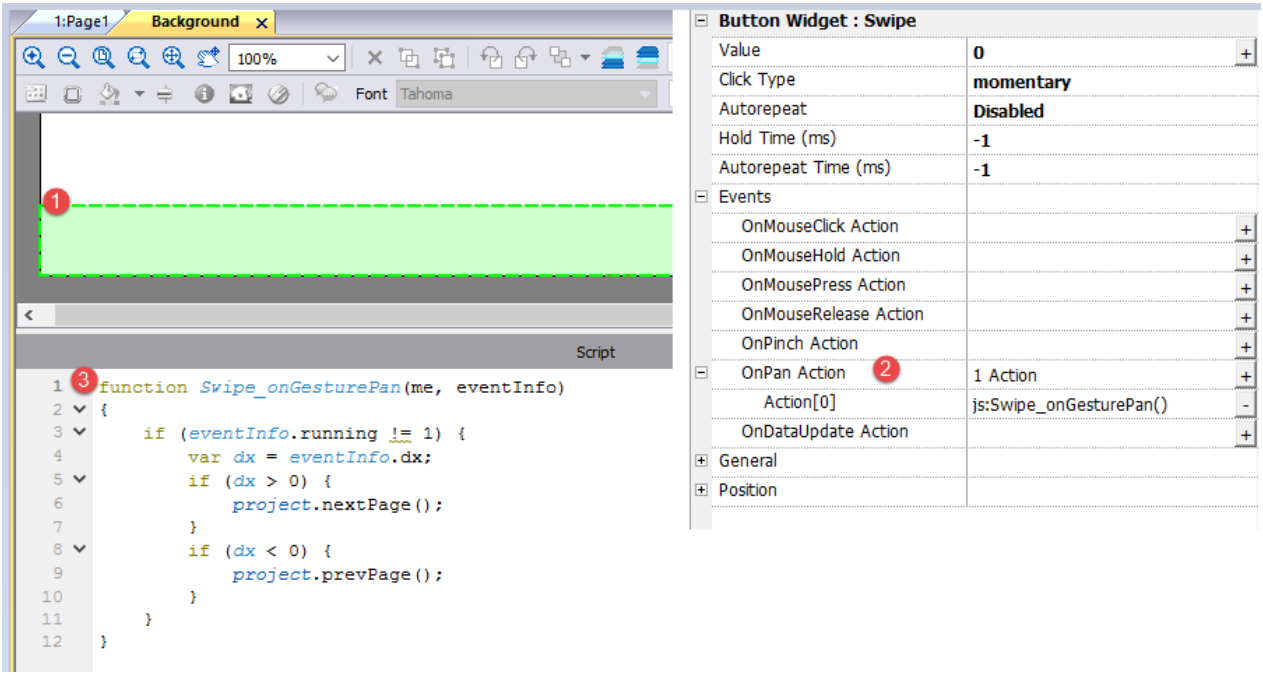
Ici figurent quelques exemples de l'utilisation d'événements en association avec le code JavaScript pour identifier les gestes et programmer les actions demandées.

Geste de balayage

Comment reconnaître un geste de « balayage » de changement de page dans l'application.

1. Mettez un widget de zone de Gestion dans la page
2. Configurer l'action OnPan pour déclencher une fonction JavaScript
3. Écrire le code JavaScript qui reconnait et gère le geste de balayage

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

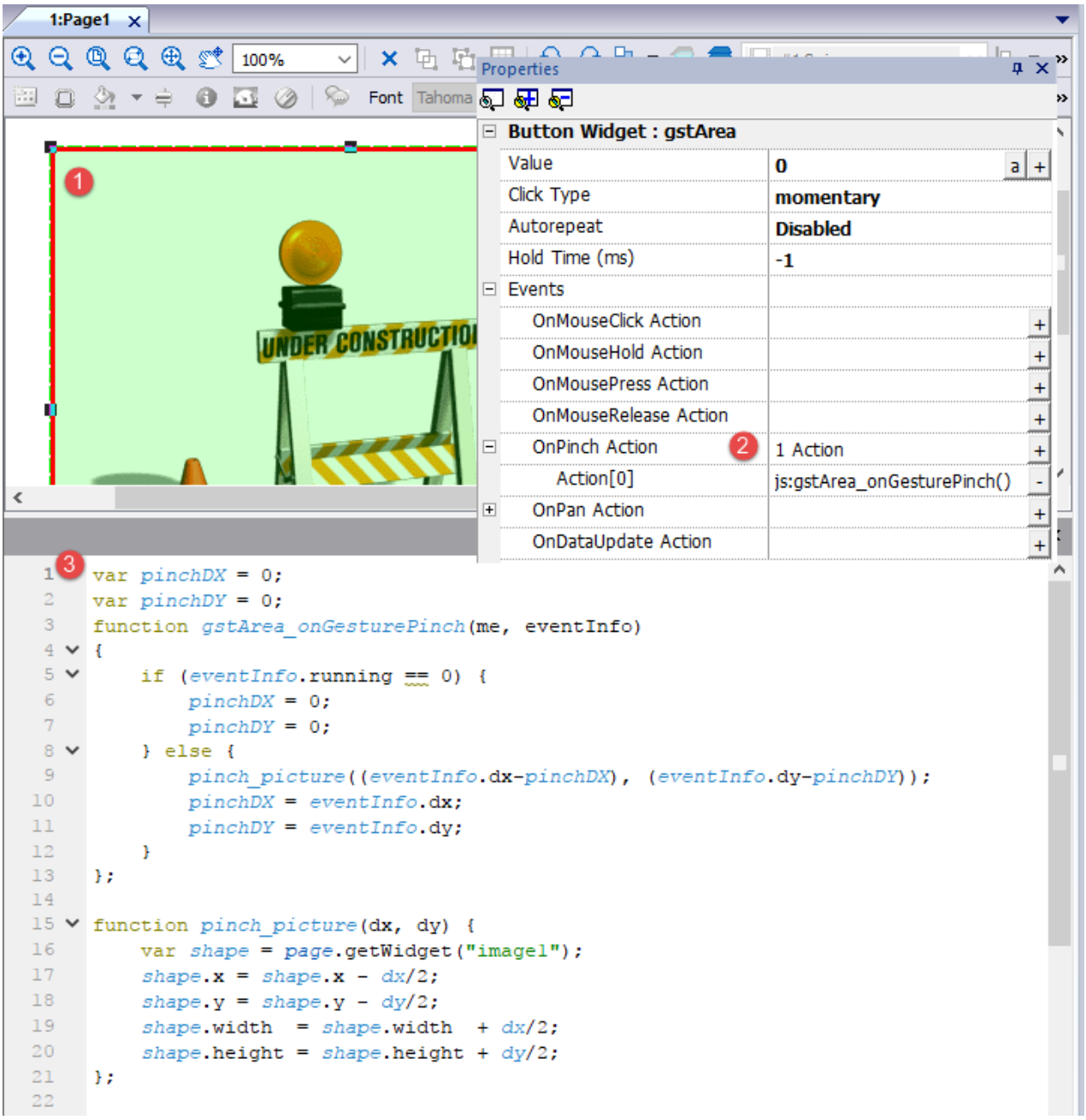


Geste de pincement

Comment reconnaître un geste de « pincement » pour redimensionner une image.

1. Placer un widget de zone de Geste dans la page sur l'image
2. Configurer l'action OnPinch pour déclencher une fonction JavaScript
3. Écrire le code JavaScript qui reconnait et gère le geste de pincement

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

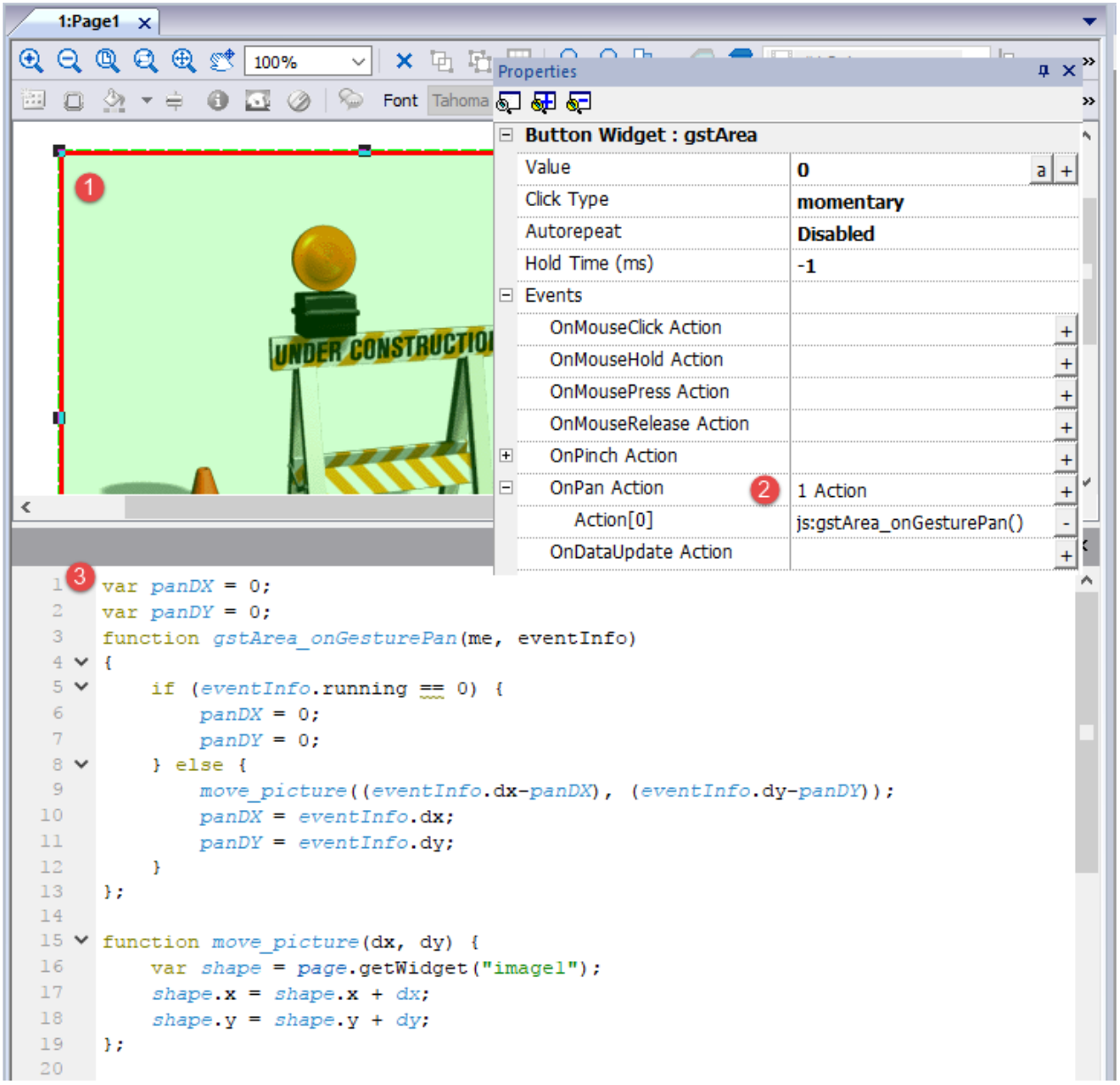


Geste panoramique

Comment reconnaître un geste « panoramique » pour déplacer une image.

1. Placer un widget de zone de Geste dans la page sur l'image
2. Configurer l'action OnPan pour déclencher une fonction JavaScript
3. Écrire le code JavaScript qui reconnait et gère le geste panoramique

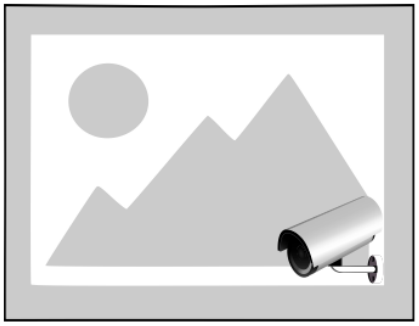
Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



Widgets Caméra IP

Chemin : ***Galerie des widgets*** *>* ***Média*** *>* ***Caméra IP***

Utilisez ces widgets pour afficher des images capturées avec une Caméra IP ou un flux vidéo.



Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Paramètre	Description
URL de Caméra	URL de la Caméra IP lorsqu'elle est utilisée dans le format JPEG.
Fréquence de rafraîchissement	Nombre d'images JPEG par seconde autorisé. Fréquence max = 1 fps.
Identifiant	Nom d'utilisateur autorisé à accéder à la caméra. Configurez ce paramètre lorsque l'accès à la caméra est protégé par un mot de passe.
Mot de passe	Mot de passe d'accès à la caméra.
URL de caméra MJPEG	URL de flux MJPEG (par exemple, http://192.168.0.1/video.cgi)

Lorsque ce widget est utilisé pour diffuser HTTP MJPEG, **l'URL de caméra** et **la Fréquence de rafraîchissement** sont ignorés.

La performance du flux n'est pas fixe et dépend de nombreux facteurs, tels que : la taille de l'image, le niveau de compression de l'image, la CPU du périphérique IHM, la qualité de la Caméra IP. En fonction de ces facteurs, le widget peut atteindre jusqu'à 25 fps.

Vous pouvez ajouter de multiples widgets de caméra IP, mais cela réduira la fréquence d'image pour chaque widget.

Caméras IP supportées

Les Caméras IP suivantes ont été testées :

Caméra IP	Protocole	URL
Apexis APM-J901-Z-WS PTZ IP Camera	MJPEG HTTP	http://{ip_address}/videostream.cgi http://{ip_address}/snapshot.cgi
AXIS M3027-PVE Network Camera	MJPEG HTTP	http://{ip_address}/axis-cgi/mjpg/video.cgi http://{ip_address}/axis-cgi/jpg/image.cgi
DAHUA DH-IPC-HD2100P-080B 1.3mp Outdoor Vandalproof	HTTP	http://{ip_address}:9988/onvif/media_service/snapshot
D-Link DCS-5605 PTZ	MJPEG	http://{ip_address}/video/mjpg.cgi
D-Link DCS-900W IP Camera	MJPEG	http://{ip_address}/video.cgi
D-Link DCS-932L	MJPEG	http://{ip_address}/video.cgi
Edimax IC-7100P PTZ	MJPEG HTTP	http://{ip_address}/mjpg/video.mjpg http://{ip_address}/picture.jpg
Foscam FI8916W	MJPEG HTTP	http://{ip_address}/videostream.cgi http://{ip_address}/snapshot.cgi

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

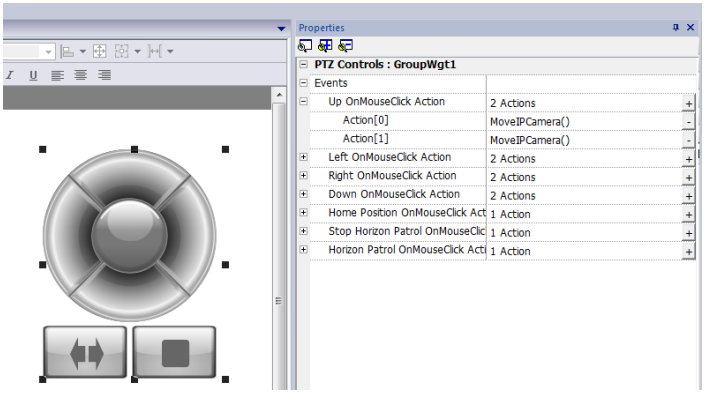
Caméra IP	Protocole	URL
Foscam FI9803 EP	MJPEG	http://{ip_address}:88/cgi-bin/CGIStream.cgi?cmd=GetMJStream&usr={user}&pwd={pass} Remarque : • le port 88 peut être différent en fonction des paramètres de la Caméra IP • {user} = identifiant défini dans les paramètres de la Caméra IP • {pass} = mot de passe défini dans les paramètres de la Caméra IP
Hamlet HNIPCAM IP Camera	MJPEG HTTP	http://{ip_address}/video.cgi http://{ip_address}/image.jpg
MOXA VPort 254 (Encodeur vidéo industriel robuste à 4 canaux MJPEG/MPEG4)	MJPEG HTTP	http://{ip_address}/moxa-cgi/mjpeg.cgi http://{ip_address}/moxa-cgi/getSnapShot.cgi?chindex=1
NVS30 network video server	MJPEG HTTP	http://{ip_address}:8070/video.mjpeg http://{ip_address}/jpg/image.jpg
Panasonic WV-Series Network Camera	MJPEG	http://{ip_address}/cgi-bin/mjpeg
Ubiquiti UniFi Video Camera	HTTP	http://{ip_address}:7080/images/snapshot/camera/{camera_guid}?force=true Remarque : • {camera_guiID} peut être trouvé sur la page web de la Caméra IP • le port 7080 peut être différent en fonction des paramètres de la Caméra IP
Zavio F3210 2MP Day & Night Compact IP Came	MJPEG HTTP	http://{ip_address}/stream?uri=video.pro3 http://{ip_address}/cgi-bin/view/image?pro_0 Remarque : • Le flux vidéo MJPEG peut être configuré en sélectionnant « profil vidéo 3 » avec une résolution 640x480 dans les paramètres de la Caméra IP.

Widget de contrôles PTZ

Les caméras PTZ (pan–tilt–zoom) sont des caméras capables de contrôler le zoom et la direction à distance.

Le widget de contrôles PTZ utilise l'action MoveIPCamera pour envoyer des commandes HTTP/cgi à la Caméra IP PTZ.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



Paramètre	Description
URL de Caméra	URL de la Caméra IP
Identifiant	Nom d'utilisateur autorisé à accéder à la caméra. Configurez ce paramètre lorsque l'accès à la caméra est protégé par un mot de passe.
Mot de passe	Mot de passe d'accès à la caméra.
Commande	Commande à envoyer au contrôleur PTZ (par exemple, decoder_ control.cgi?command=0)

Méthodes d'authentification

La méthode d'authentification est automatiquement définie par le serveur web de la caméra auquel le widget est connecté. Les méthodes d'authentification supportées sont :

- Basique
- NTLM version 1
- Digest-MD5

Widget blocage de fonction Javascript

Chemin : ***Galerie des widgets*** *>* ***Basique*** *>* ***Blocage de fonction Javascript***

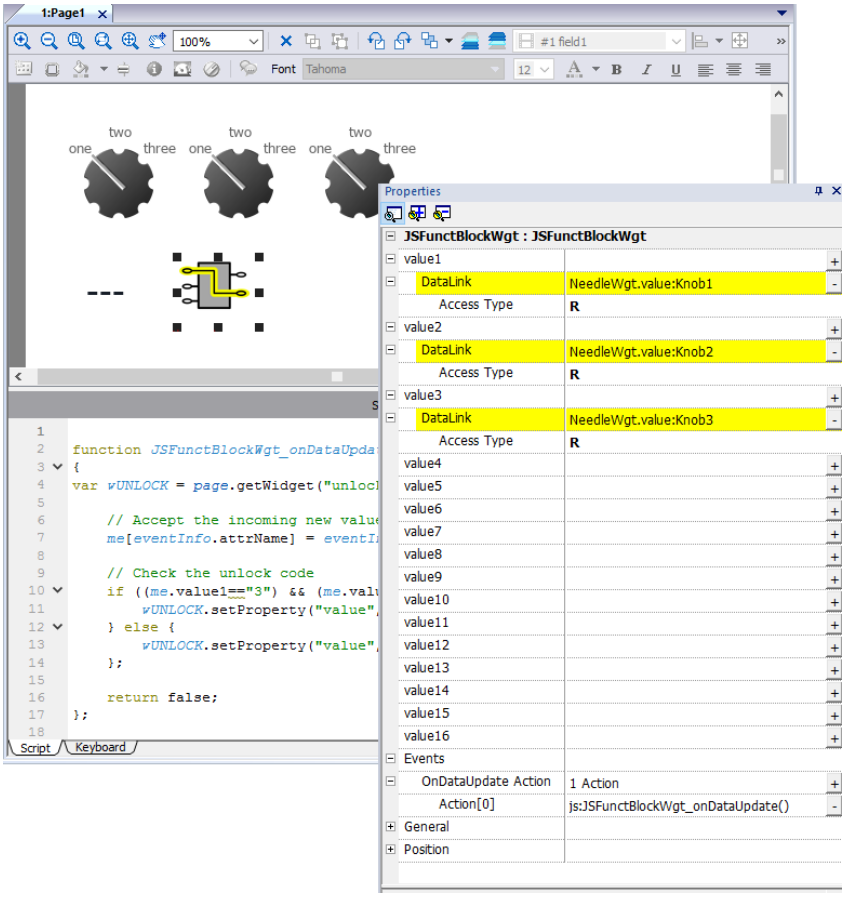
Le blocage de fonction Javascript est un widget contenant une logique Javascript qui est exécutée lorsque les valeurs de variable changent.

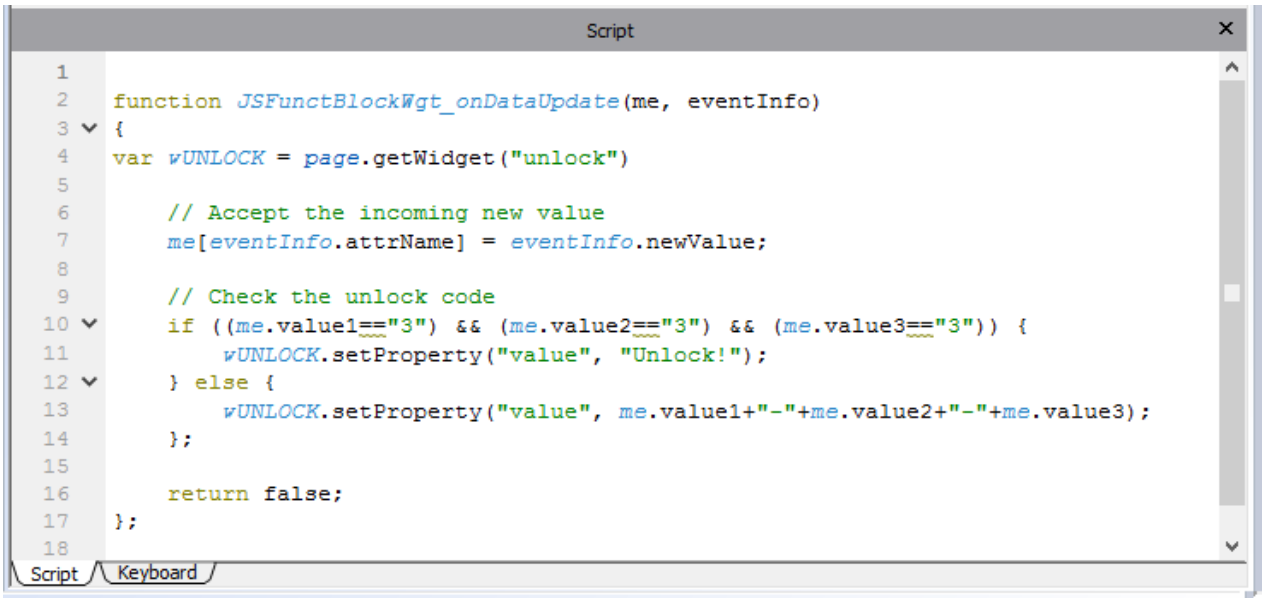
Paramètre	Description
valeur1 **...** **valeur16**	Objets qui déclencheront l'action OnDataUpdate.
LorsdeMiseàJour	Action qui sera exécutée si une modification de valeur associée est détectée

Remarque : le widget n'est rendu que dans PB610 Panel Builder 600, et n'est pas rendu dans le périphérique IHM.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Exemple :
Un code Javascript qui contrôle la serrure à combinaison de trois sélecteurs





Voir "Widget événements" Page 434pour la description des paramètres onDataUpdate

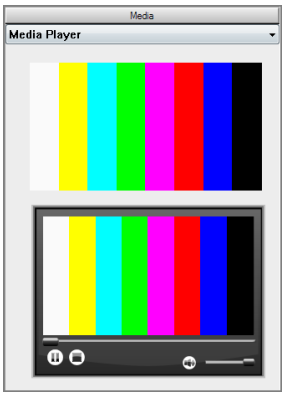
Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Widgets de lecture de médias

Chemin : ***Galerie des widgets*** *>* ***Média*** *>* ***Lecture de médias***

Utilisez ces widgets pour lire des vidéos à partir d'une liste de lecture. Les fichiers vidéo peuvent être stockés sur une clé USB, une carte mémoire flash ou une carte SD.

Deux widgets sont disponibles : l'un inclut un cadre multimédia avec des boutons pour lire et interrompre la vidéo, l'autre est un simple cadre où la vidéo est lu sans que l'utilisateur ne dispose de commandes.



Paramètre	Description
Liste de lecture médias	Ouvrez le navigateur de fichiers de Windows pour sélectionner les fichiers vidéo à intégrer à la liste de lecture. Les fichiers sélectionnés seront téléchargés à l'intérieur du périphérique IHM avec le projet.
Style en boucle	Définissez la modalité de lecture de la vidéo. • **NoLoop** : lit toutes les vidéos de la liste de lecture, puis s'arrête. • **LoopOne** : répète la première vidéo dans la liste de lecture. • **LoopAll** : répète toute la liste de lecture. • **Random** : lit les vidéos selon un ordre aléatoire.

Remarque : Le widget lecteur de médias ne fonctionne qu'avec certains périphériques IHM (voir "Capacités des périphériques IHM" Page 541). Cela ne fonctionne pas avec le HMI Client.

Remarque : vous ne pouvez avoir qu'un seul widget de lecture de médias dans une page.

Codage vidéo supporté

Deux groupes de codecs sont supportés :

- Codecs vidéo basés sur DSP
- Codecs logiciels vidéo

La liste des appareils IHM qui supportent le DSP (accélération matérielle vidéo) est disponible dans "Capacités des périphériques IHM" Page 541.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Codecs vidéo DSP

Ils incluent :

- H264 en utilisant un conteneur AVI/MP4, CABAC désactivé et Niveau 3 (suggéré)
- MPEG4 en utilisant un conteneur MP4

Avec le système d'exploitation WinCE, BSP v1.55 ou plus est nécessaire
Sous Linux , BSP v1.0.269 ou plus est nécessaire

Codecs logiciels vidéo

Concerne uniquement :

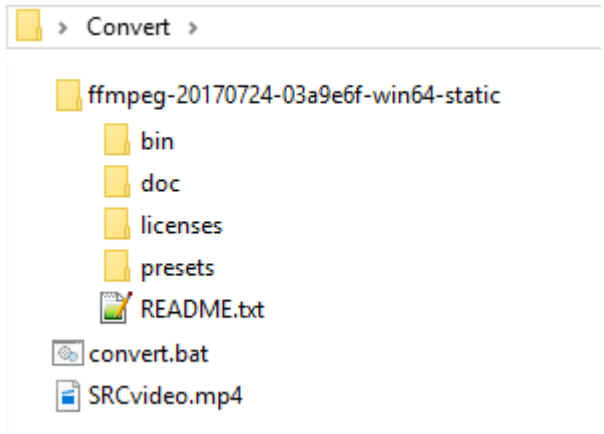
- Microsoft MPEG4 v3 en utilisant un conteneur AVI.

Sachez que les performances vidéo dépendent du choix de la résolution, du débit binaire et des capacités du périphérique. Si le rendu vidéo n'est pas lisse, essayez de réduire la résolution ou le débit binaire de votre vidéo.

Les vidéos codées avec Microsoft MPEG4 v3 n'utilisent pas l'accélération matérielle et sont davantage limitées. Pour éviter une lecture saccadée des vidéos, nous suggérons une résolution maximale de 640x512 pixels et un débit binaire de 1300 kb/s. De plus, la taille du widget Media Player utilisé sur la page doit être la même que celle des vidéos de la liste de lecture pour éviter de devoir agrandir et réduire. L'audio n'est pas pris en charge.

Convertir une vidéo

Le FFMPEG (www.ffmpeg.org) peut être utilisé pour convertir une vidéo dans le codec correct pris en charge par le périphérique IHM. En utilisant la structure de dossiers de l'image ci-dessous, le fichier de lot suivant pourrait être utilisé pour convertir n'importe quel fichier vidéo.



Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

convert.bat

```
@echo off
set FFMPEG=ffmpeg-20170724-03a9e6f-win64-static\bin\ffmpeg.exe

%FFMPEG% -i SRCvideo.mp4 ^
         -y ^
         -an ^
         -s 240x160 ^
         -b:v 4200k ^
         -maxrate 4200k ^
         -c:v libx264 ^
         -profile:v baseline ^
         -level:v 3 ^
         -bufsize 3000k ^
         -minrate 0 ^
         -f avi ^
         -preset slow ^
         HMIvideo.avi

pause
```

Vous pouvez maintenant ouvrir les vidéos avec un lecteur vidéo classique, tel que Windows Media Player et contrôler la qualité. Vous pouvez maintenant ouvrir la vidéo convertie avec un lecteur vidéo ordinaire, tel que Windows Media Player et en vérifier la qualité. Vous pouvez ajouter la vidéo obtenue à la liste de lecture du widget Lecteur de médias.

Remarque : l'outil FFMPEG n'est pas fourni avec le PB610 Panel Builder 600.

Utilisation du lecteur de médias dans Javascript

Le widget de lecture de médias peut aussi être référencé dans les programmes Javascript avec la syntaxe suivante :

```
//get the mediaplayer widget.
var mediaWgt = page.getWidget('MediaPlayerWgt2');
//load the play list
mediaWgt.setProperty('medialist', '/Storage Card/demo_3.avi,/Storage Card/video1_
3.avi');
// set the loopstyle 0 - noloop, 1 - loop one, 2- loop all, 3 - random
mediaWgt.setProperty('loopstyle', 2);
//start playing the first file.
mediaWgt.mediapath = '/Storage Card/demo_3.avi';
```

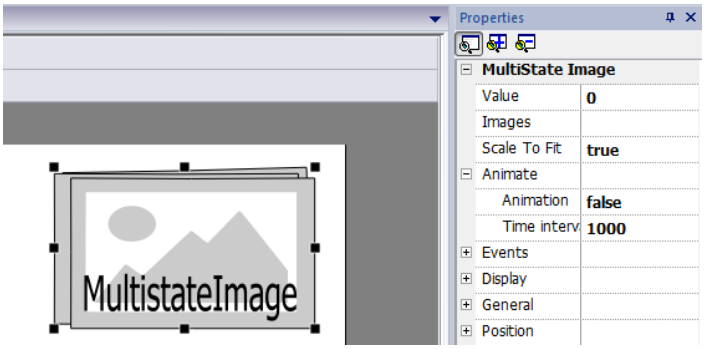
Voir "JavaScript" Page 429 pour plus de détails sur les modalités de fonctionnement avec JavaScript.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Widget d'image multi-états

*Chemin : **Galerie des widgets**> **Basique**> **Images***

Utilisez ce widget pour afficher une image parmi la série proposée, en fonction de la valeur d'une variable utilisée comme Index. Vous pouvez aussi utiliser ce widget pour de simples animations.



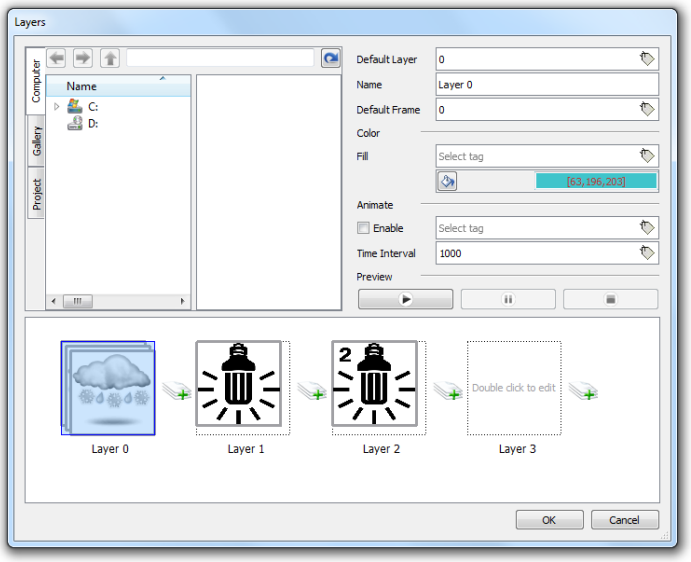
Paramètre	Description
Valeur	Index de l'image à afficher. Par exemple, définissez Valeur=0 pour afficher l'image avec l'index 0 dans la série d'images.
Images	Série d'images associées à un index.
Animer	Configurez sur vrai pour activer un diaporama.
Intervalle de temps	Intervalle entre les images dans le diaporama.

Widget multi-couche d’image multi-états

*Chemin : **Galerie des widgets**> **Basique**> **Images***

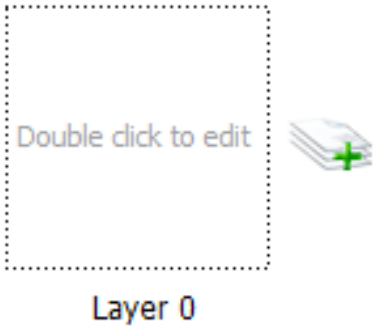
Utilisez ce widget pour créer différentes animations et sélectionnez la plus appropriée au runtime.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

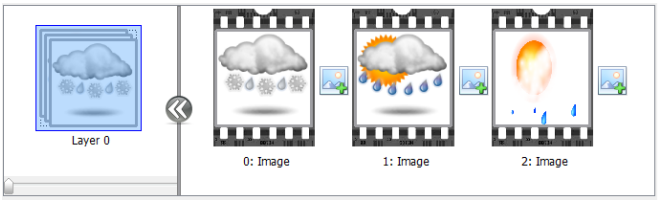


Paramétrer les couches de widget

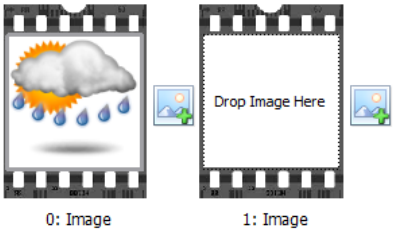
1. Ouvrez la boîte de dialogue **Couches** à partir du volet **Propriétés**.
2. Cliquez sur **+** pour ajouter le nombre de couches souhaitées.



3. Double-cliquez sur chaque couche pour ajouter le nombre d'images souhaitées à inclure dans la couche.



4. Glissez-déposez les images dans le cadre pour les ajouter à la couche actuelle.



5. Définissez les propriétés de widget.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

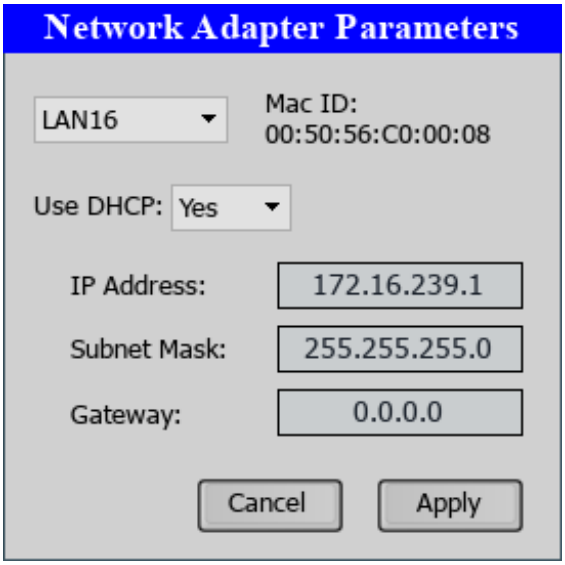
Paramètre	Description
Couche par défaut	Couche affichée au runtime.
Nom	Nom de la couche sélectionnée.
Cadre par défaut	Cadre apparaissant lorsque la couche actuelle est affichée.
Couleur / Remplissage	Couleur de remplissage pour les images de la couche actuelle.
Animer	Active le diaporama pour la couche active. Les animations peuvent être lancées/interrompues au runtime en les associant à une variable.
Intervalle de temps	Intervalle de temps du diaporama, si activé.
Aperçu	Simulation du diaporama.

Remarque : les **Couche par défaut**, **Cadre par défaut**, **Couleur** et **Remplissage** peuvent être modifiés au runtime, en les associant à une variable.

Widget adaptateurs réseau

*Chemin : **Galerie des widgets**> **Basique**> **Contrôle***

Utilisez le widget IP pour configurer les paramètres d'adaptateurs réseau.



La variable système Réseau->État contient le résultat de la dernière opération exécutée par le widget IP (voir "Variables réseau" Page 103pour plus d'informations)

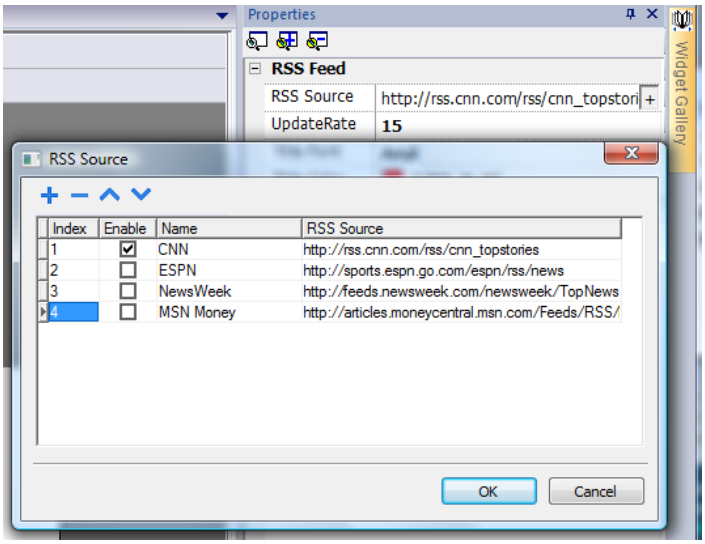
Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Widget flux RSS

*Chemin : **Galerie des widgets**> **Média**> **Source de flux RSS***

Utilisez ce widget pour afficher sur le périphérique IHM votre flux RSS préféré directement à partir d'Internet.

Paramètre	Description
Source RSS	URL du flux Remarque : les sources de flux ne peuvent pas être modifiées au runtime.
UpdateRate	Rafraîchissement l'heure



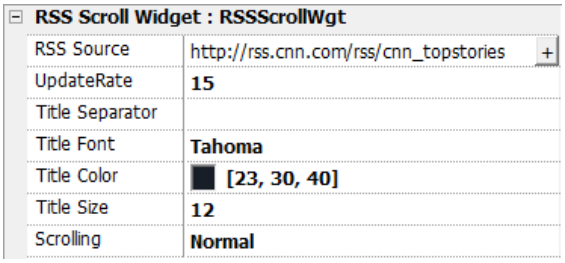
Le widget de flux RSS a été spécifiquement conçu pour fonctionner avec Pocket Internet Explorer.

Faire défiler le widget flux RSS

*Chemin : **Galerie des widgets**> **Média**> **Défilement flux RSS***

Utilisez cette version du widget principal de flux RSS pour afficher les faits saillants à l'intérieur d'une ligne de texte à l'aide d'un texte défilant.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



Ce widget a des propriétés supplémentaires.

Paramètre	Description
Défilement	Vitesse de défilement
Séparateur de titres	Caractère de séparation entre les éléments surlignés

Widget tableau

Chemin : **Galerie des widgets**> **Basique**> **Tableau**

Utiliser ce widget pour créer une tableau avec des données fournies à partir d'une source de données.

Pour configurer un tableau :

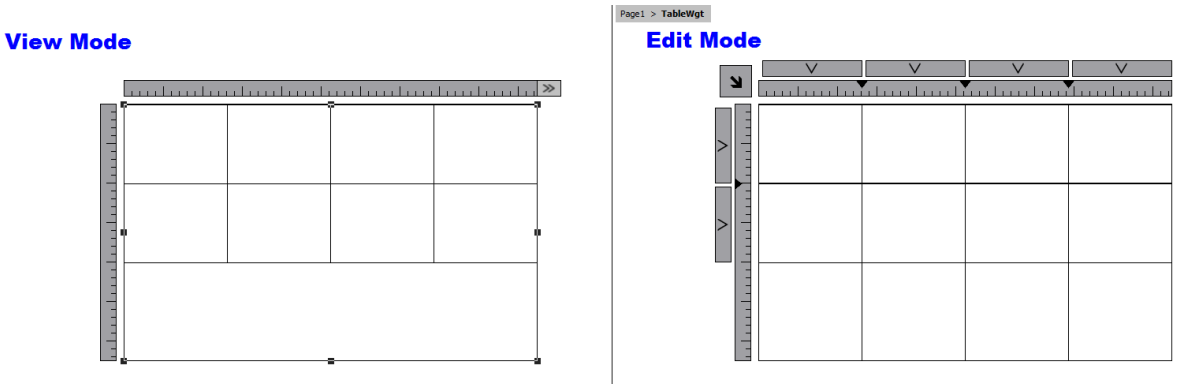
1. Placer un widget de tableau sur l'écran et configurer le modèle de tableau.
2. Ajouter des widgets dans les cellules pour configurer une ou plusieurs lignes qui seront utilisées en tant que modèles de ligne lorsque le tableau sera rempli de données fournies par la source de données.
3. Sélectionner une source de données qui seront utilisées pour remplir les lignes du tableau
4. Définir les liens de widgets et source de données.

Configurer le widget tableau

Le Widget tableau a deux états :

- Mode vue
- Mode édition.

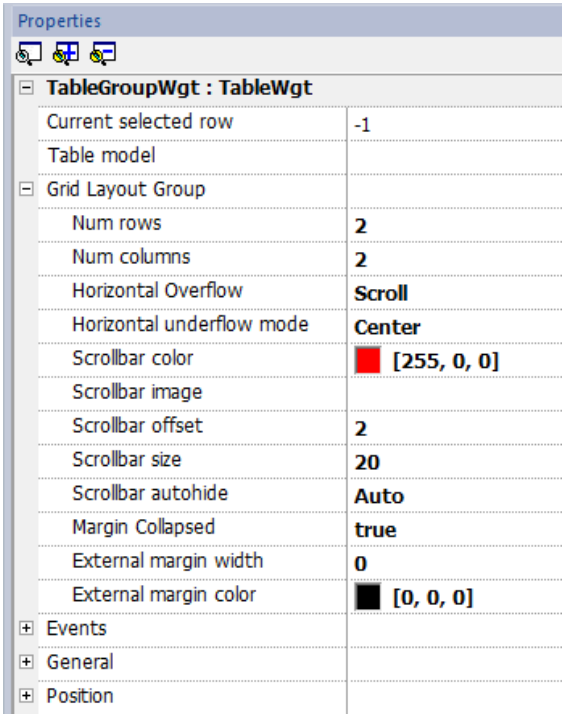
Cliquer sur le tableau pour gérer le widget en mode vue, cliquer deux fois pour entrer dans le mode édition. Pour quitter et revenir au mode vue, cliquer en dehors du tableau.



Mode vue

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

En mode vue, vous pouvez configurer la disposition de tableau. Glisser et déposer le tableau sur la page, redimensionner le tableau, définir le nombre de lignes modèle, nombre de colonnes et les principales propriétés du tableau.

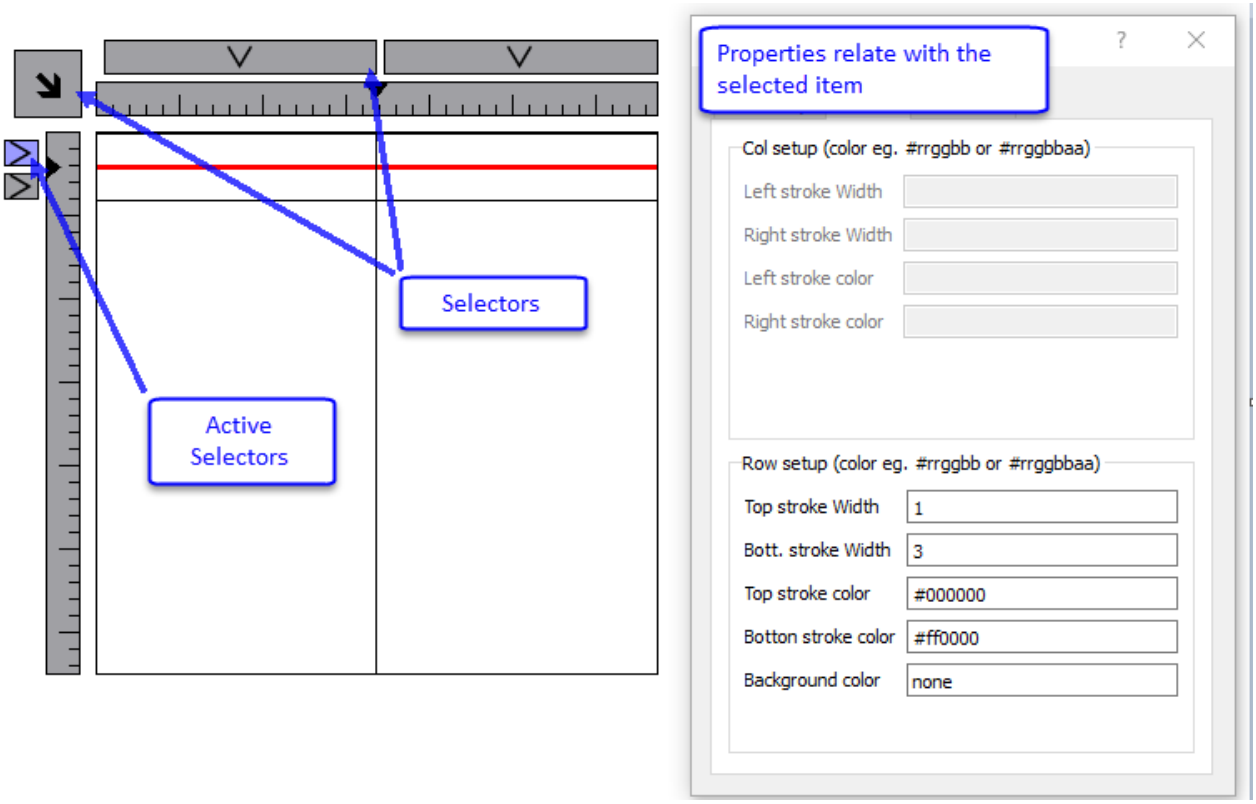


Mode édition

En mode édition, il est possible de configurer le format et le contenu de chaque cellule du tableau. Caque ligne du tableau se comportera comme un modèle de ligne.

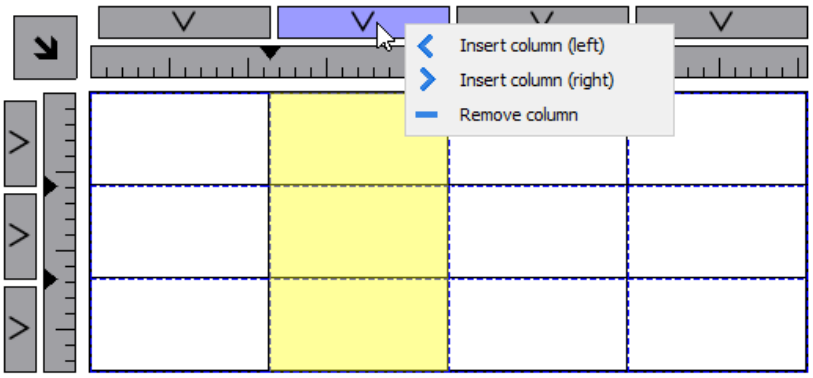
Pour configurer l'apparence du tableau, cliquer sur les sélecteurs de tableau pour sélectionner l'élément à configurer.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



Ajouter ou supprimer des lignes ou des colonnes

Pour ajouter ou supprimer des lignes ou des colonnes, double-cliquez sur la grille pour entrer en mode édition et faites un clic droit de souris sur le sélecteur de lignes et de colonnes pour ouvrir le menu contextuel.



Fusionner ou séparer des lignes ou des colonnes

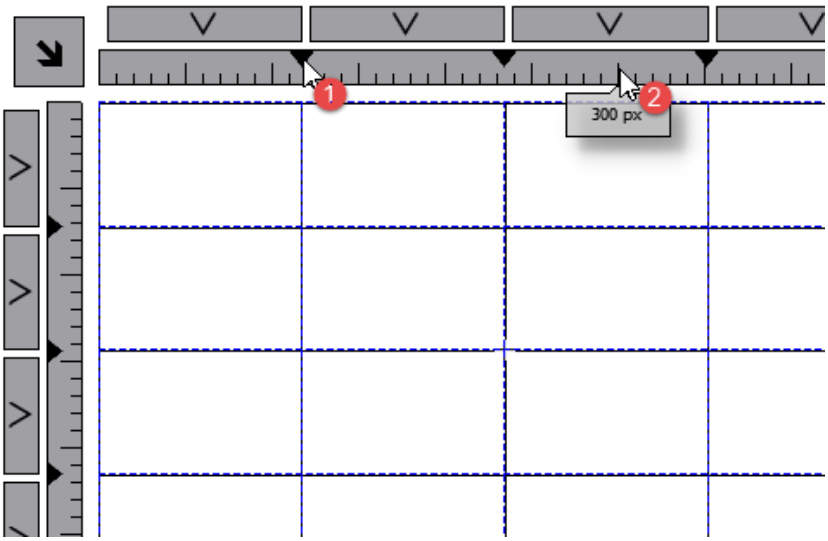
Pour fusionner ou séparer des lignes ou des colonnes, double-cliquez sur la grille pour entrer en mode d'édition et déplacez le curseur sur les rubans :

- Double-cliquer sur le triangle noir pour fusionner les deux lignes ou colonnes adjacentes (1)

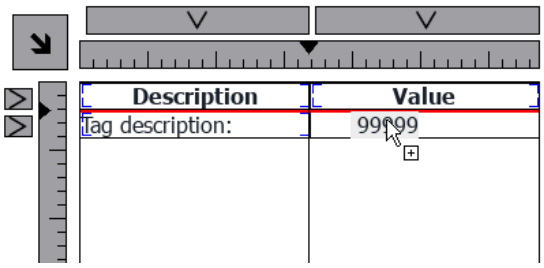
  Veuillez noter que la fusion n'est possible qu'avec des lignes ou colonnes vides.

- Double-cliquer sur le ruban pour diviser la ligne ou colonne sélectionnée (2)

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



Pour configurer le contenu des cellules, faire glisser les widgets à l'intérieur des cellules.



Si vous avez besoin plus de widgets dans une seule cellule, créer un groupe de widgets et copier le groupe depuis la page sur la cellule.

Configuration de la source de données

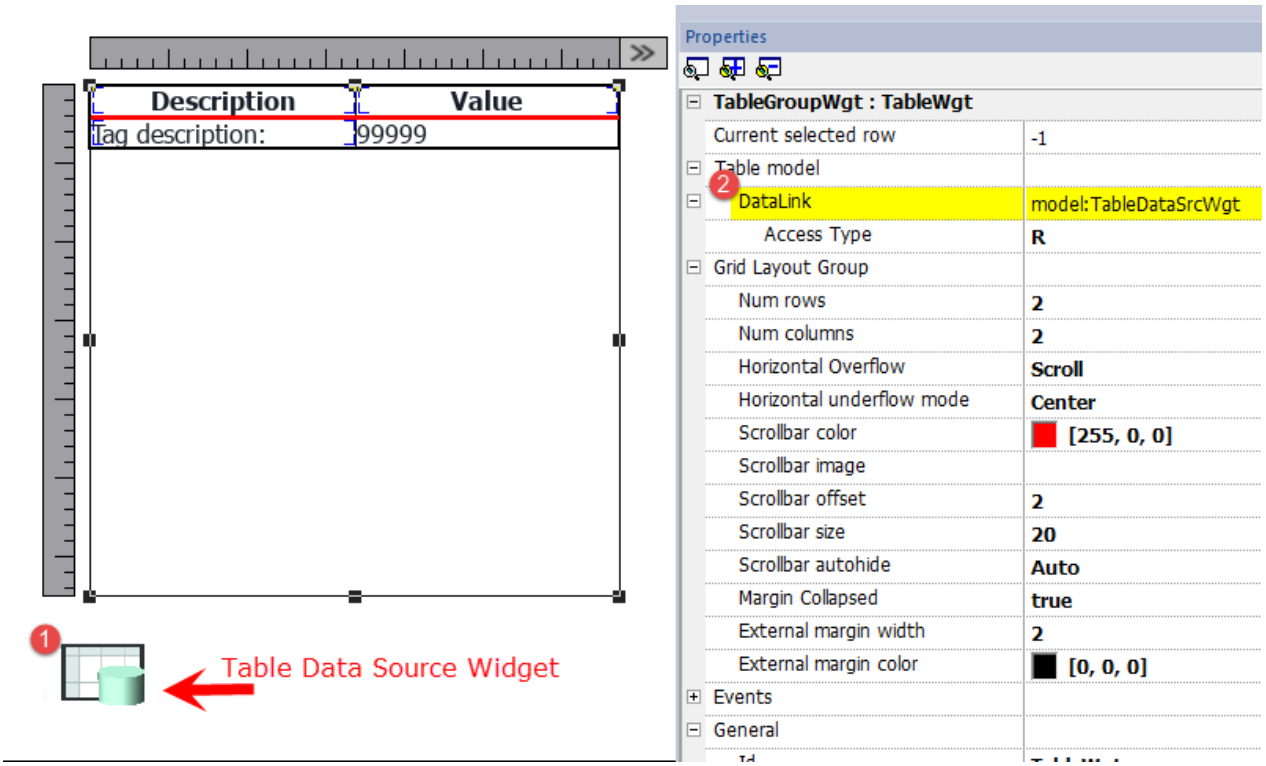
La source de données, qui fournit les données pour remplir le tableau, pourrait être un Widget source de données de tableau ou un objet JavaScript.

Widget source de données tableau

Chemin : **Galerie des widgets**> **Basique**> **Tableau**

1. Glisser et déposer un *Widget de source de donnée de tableau* sur la page
2. Voir le paramètre *Modèle de tableau* pour relier à la source de données.

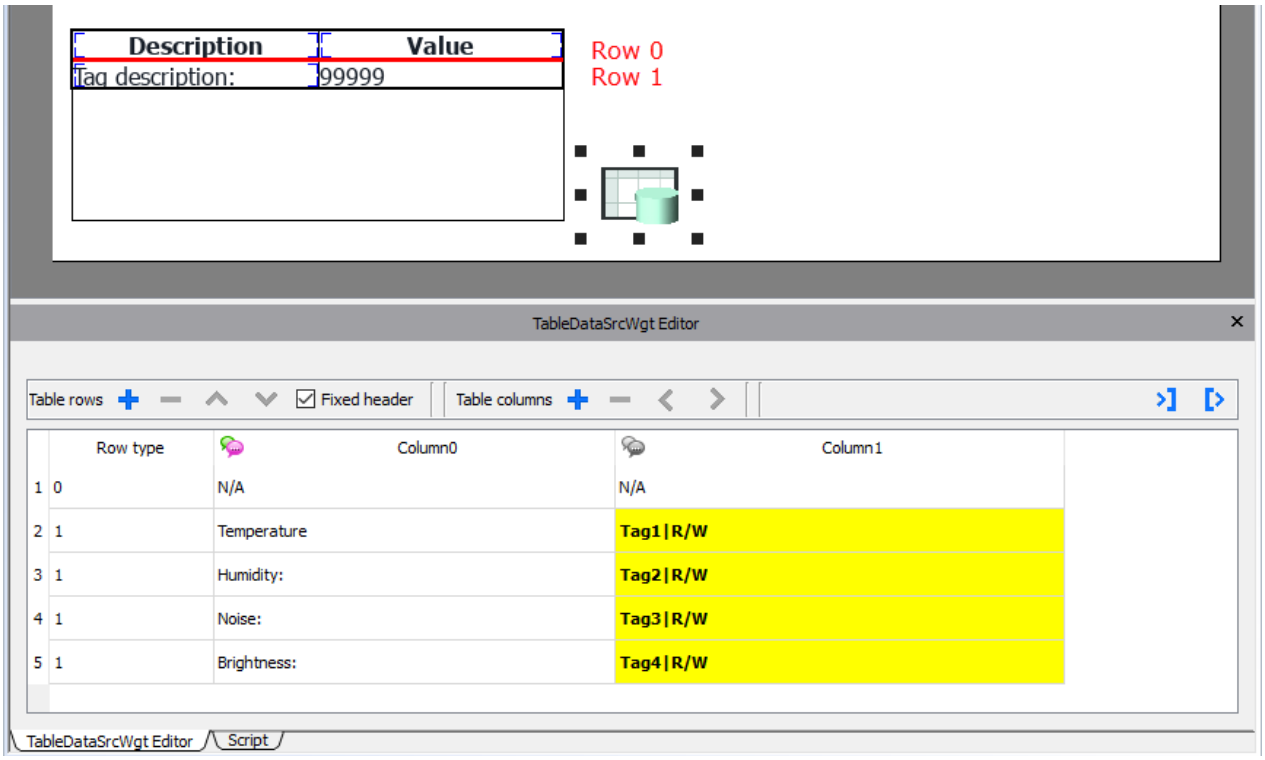
Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



Sélectionner la source de données et à l'intérieur de l'éditeur TableDataSrcWgt ajouter les lignes et les colonnes nécessaires. Dans l'exemple suivant, nous avons défini deux modèles de lignes :

- Ligne 0
  En-tête du tableau. Contient uniquement du texte statique.
- Ligne 1
  Modèle de lignes avec données. Sur la première colonne, nous avons ajouté une étiquette qui contiendra la description et sur la seconde colonne, un champ qui contiendra la valeur.

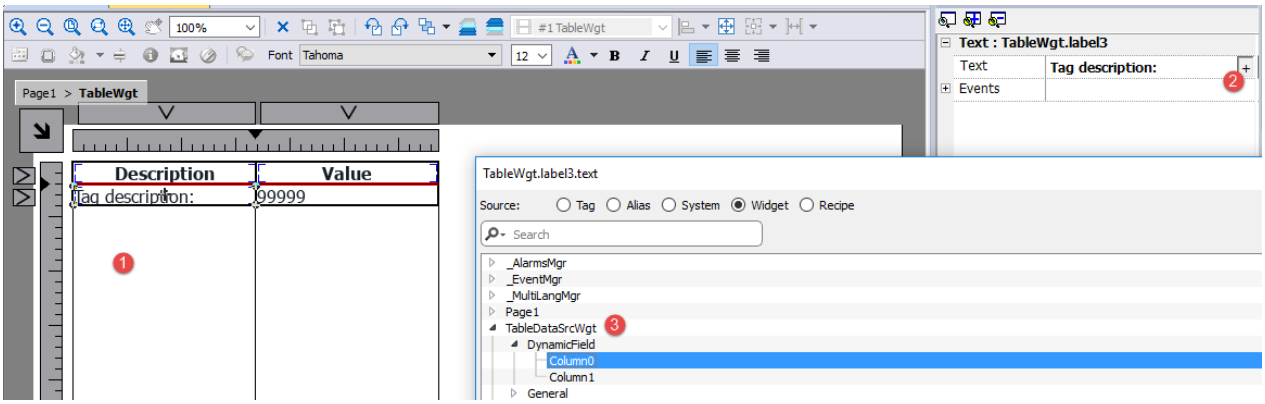
Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



Chaque ligne doit être affectée d'un type de ligne. La ligne prend le format du modèle de ligne correspondant. Les widgets placés dans chaque cellule du modèle de ligne apparaîtront dans les rangées de ce type.

Définir des liens avec la source de données

1. Double-cliquer sur le widget de tableau pour entrer en mode d'édition et sélectionnez un widget
2. Sélectionner la propriété qui doit être lu à partir de la source de données
3. Sélectionner la colonne de la source de données qui fournira les données



L'image ci-dessous montre comment notre exemple sera rendu au runtime.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Description	Value
Temperature	111
Humidity:	222
Noise:	333
Brightness:	444

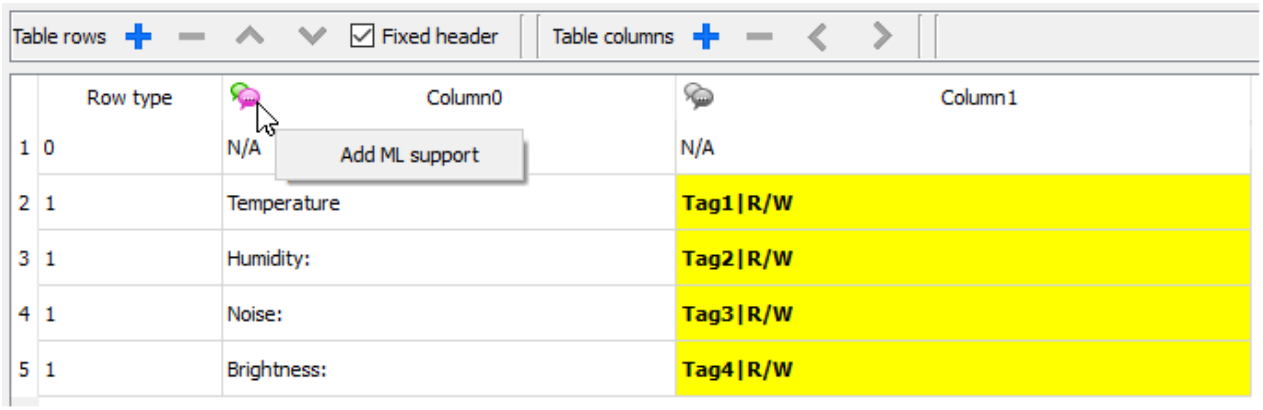
En-tête fixe

Si vous voulez que la première ligne ne soit pas déroulante, cochez la case *"Corriger l'en-tête"*.

Multilangue

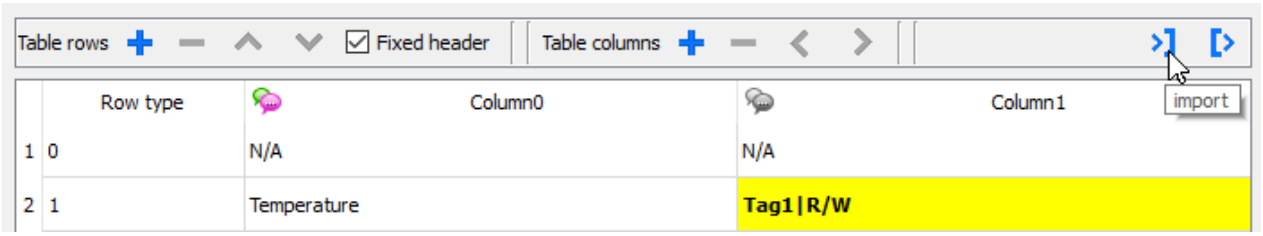
Pour activer le support Multilingue, faire un clic droit sur l'icône Multilingue de la colonne. L'icône changera de couleur pour mettre en évidence que le support est activé.

Éviter en permettant le support multilingue lorsqu'ils ne sont pas nécessaires pour de meilleures performances.



Importer/Exporter la source des données

La configuration de la source de données peut être importée/exportée à l'aide de fichiers xml.



Objet JavaScript

Comme alternative au Widget de source de données, les données devant remplir le tableau peuvent être fournies à partir d'un objet JavaScript. Dans ce cas, nous devons remplir un tableau d'éléments avec les données à utiliser et attribuer le tableau au widget tableau.

```
var myTable = page.getWidget("TableWgt1");
```

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

```
myTable.model = model;
```

modèle est un tableau d'éléments avec la définition et les données de tableau. Le premier élément du tableau contiendra le modèle des lignes tandis que les autres éléments contiendront les données à remplir dans les lignes du tableau.

```
model[0] = row_templates;    // row templates
model[1] = row_data1;        // data of the row1
model[2] = row_data2;        // data of the row2
model[3] = row_data3;        // data of the row3
model[4] = row_data4;        // data of the row4
model[5] = row_data4;        // data of the row5
```

Le **modèle de ligne** est un tableau multidimensionnel où chaque tableau définit la liaison de données d'un modèle de ligne.

Sur l'exemple ci-dessous, nous avons un modèle pour deux rangées.

```
var row_templates = {
    _h : [
            [ []       , []        ],   //rowType = 0
            [ ["text"] , ["value"] ]    //rowType = 1
        ]
}
```

La première ligne comporte deux colonnes qui ne contiennent pas de liens vers des données. Nous utilisons ce modèle pour l'en-tête sur la première ligne de notre tableau.

La deuxième ligne définit le modèle d'une ligne avec la propriété « texte » du widget dans la première colonne et la propriété « valeur » du widget dans la deuxième colonne. Elles seront remplies dynamiquement à l'aide des données fournies dans la variable de modèle.

Dans l'exemple ci-dessous, nous définissons une **ligne de données**

```
var row_data = {
    _t : 1,
    _v : ["Temperature:", { _c : "dl" , s : "_TagMgr", a : "Tag1", i: 0, m : 2 }]
}
```

Le premier élément est le modèle de ligne à utiliser tandis que le deuxième élément est le tableau avec les données à utiliser. Dans notre exemple « Température : » est le texte à utiliser à l'intérieur du widget sur la première colonne, tandis que l'autre élément est une liaison de données qui fournira la valeur pour remplir la propriété de valeur du widget dans la deuxième colonne.

L'élément de liaison de données :

Paramètre	Description
_c : "dl"	Identifier l'élément comme une liaison de données
s : "_TagMgr"	Indiquer la source des données est le Gestionnaire de variables
a : "Tag1", i: 0, m:2	Spécifier le nom de la variable, l'indice (nécessaire quand une variable est un tableau) et le mode lire/écrire.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Paramètre	Description
	• m=0 est Read Only • m=1 est Write Only • m=2 est Read/Write

Le code JavaScript ci-dessous générera le même tableau que l'exemple précédent lié avec le widget de source des données de tableau.

```
var model = [];

var row_templates = {
    _h : [
            [ []       , []        ],   //rowType = 0
            [ ["text"] , ["value"] ]    //rowType = 1
        ]
}

var row_data1 = {
    _t : 0,
    _v : [],
    _h : true
}

var row_data2 = {
    _t : 1,
    _v : ["Temperature:", { _c : "dl" , s : "_TagMgr", a : "Tag1", i: 0, m : 2 }]
}

var row_data3 = {
    _t : 1,
    _v : ["Humidity:",    { _c : "dl" , s : "_TagMgr", a : "Tag2", i: 0, m : 2 }]
}

var row_data4 = {
    _t : 1,
    _v : ["Noise:",       { _c : "dl" , s : "_TagMgr", a : "Tag3", i: 0, m : 2 }]
}

var row_data5 = {
    _t : 1,
    _v : ["Brightness:",  { _c : "dl" , s : "_TagMgr", a : "Tag4", i: 0, m : 2 }]
}

model[0] = row_templates;
model[1] = row_data1;
model[2] = row_data2;
model[3] = row_data3;
model[4] = row_data4;
model[5] = row_data5;
```

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

```
var myTable = page.getWidget("TableWgt1");
myTable.model = model;
```

Notez que la première ligne (row_data1) contient la directive _h: vrai pour éviter que la première ligne soit déroulante.

```
var row_data1 = {
    _t : 0,
    _v : [],
    _h : true
}
```

Multilangue

Un texte de langues multiples peut être entré en utilisant l'élément ci-dessous :

```
{ _c : "ml" , mltext : {"en-US" : "Temperature:" , "it-IT" : "Temperatura:"} }
```

Paramètre	Description
_c : "ml"	Identifier l'élément comme un texte multilangue
mltext : { ... }	Liste de couples : "ID Language":"Text" Exemple : • "en-US" : "Temperature:" • "it-IT" : "Temperatura:"

Exemple :

```
var row_data2 = {
    _t : 1,
    _v : [ { _c : "ml" , mltext : { "en-US" : "Temperature:",
                                    "it-IT" : "Temperatura:"}   },
           { _c : "dl" , s : "_TagMgr", a : "Tag1", i: 0, m : 2 }
         ]
}
```

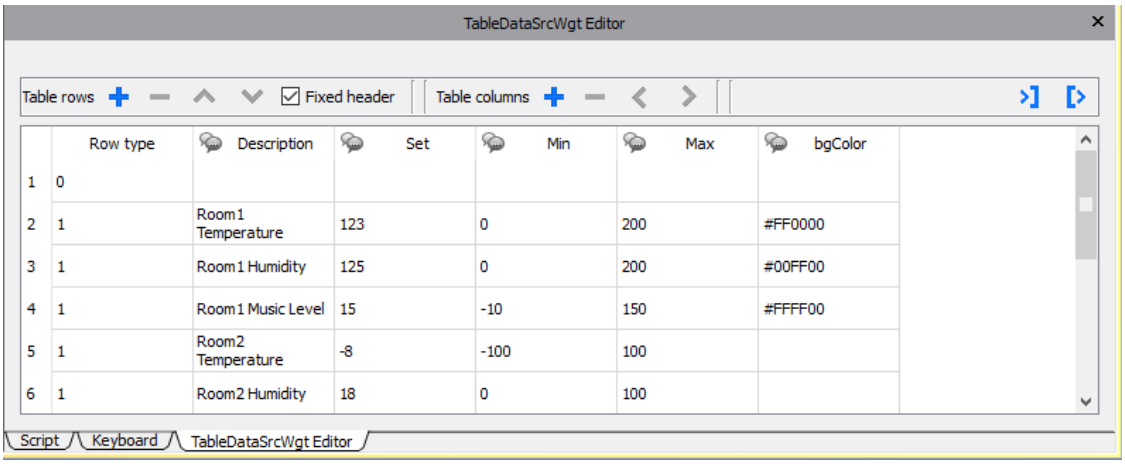
Couleur d'arrière-plan de ligne.

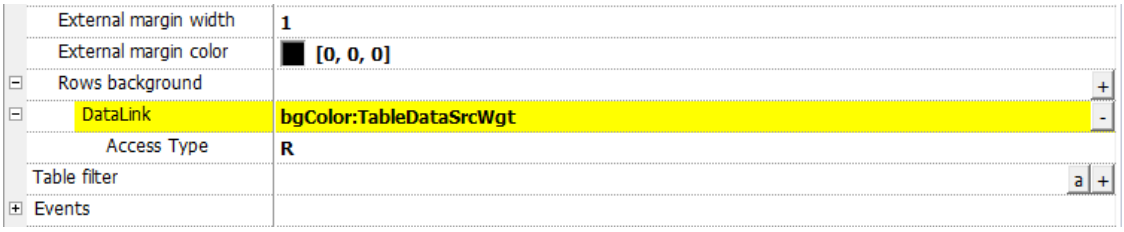
À l'aide du paramètre Couleur d'arrière-plan de ligne, il est possible de définir la colonne du Widget Source de données qui contiendra la couleur d'arrière-plan de la ligne associée.

Pour configurer la couleur d'arrière-plan des lignes du tableau :

1. Ajouter une nouvelle colonne à l'intérieur du Widget Source de données devant contenir la couleur de fond de chaque ligne
2. Configurer le paramètre de couleur « d'arrière-plan de ligne » du tableau pour pointer vers la colonne couleur du Widget Source de données

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21





Filter:

Description	Set	Min	Max
Room1 Temperature	123	0	200
Room1 Humidity	125	0	200
Room1 Music Level	15	-10	150
Room2 Temperature	-8	-100	100
Room2 Humidity	18	0	100
Room2 Music Level	12	0	150
Room3 Temperature	15	0	150
Room3 Humidity	134	0	500

Filtre de tableau

Le contenu visible à l'intérieur du tableau peut être filtré à l'aide de la propriété " Filtre de tableau ". Dans la liaison de données, vous pouvez utiliser une formule (voir "Formule" Page 50 pour plus de détails) afin de définir les critères à utiliser pour filtrer les données.

Chaque ligne du tableau ne sera visible que lorsque le lien de données du Filtre de tableau renvoie la valeur vraie.

Exemple 1

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Si vous voulez choisir de ne voir que les lignes qui contiennent « quelque chose » à l'intérieur de la colonne Description, vous pouvez utiliser la formule ci-dessous :

```
=$Contains($('Description:TableDataSrcWgt'),$('value:SearchOnTable'))
```

Où

- 'Description:TableDataSrcWgt' est un champ dynamique du Widget Source de données utilisé à partir du tableau pour identifier la colonne à vérifier
- 'value:SearchOnTable' est la valeur d'un champ de texte qui contient la chaîne de caractères à rechercher

Filter:

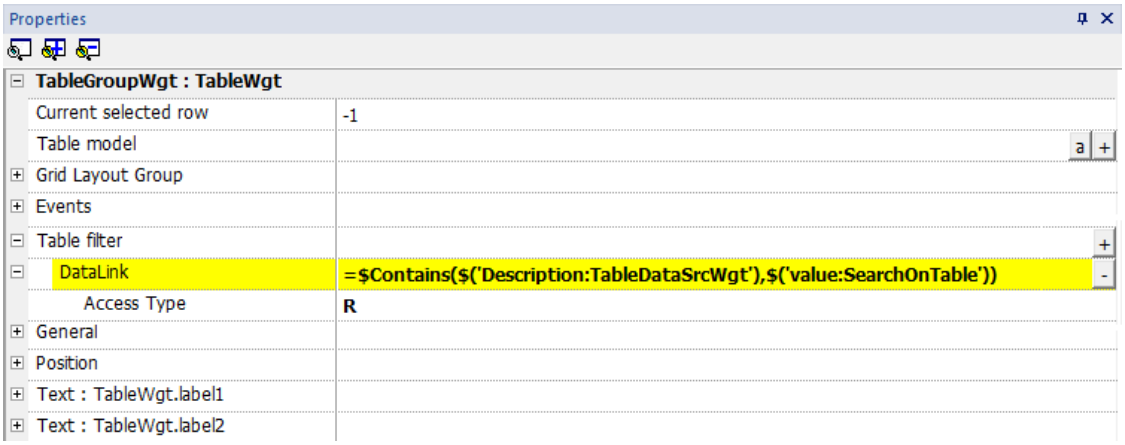
Description	Set	Min	Max
Room1 Temperature	123	0	200
Room1 Humidity	125	0	200
Room1 Music Level	15	-10	150
Room2 Temperature	-8	-100	100
Room2 Humidity	18	0	100
Room2 Music Level	12	0	150
Room3 Temperature	15	0	150
Room3 Humidity	134	0	500

Filter: Room2

Room2 Temperature	-8	-100	100
Room2 Humidity	18	0	100
Room2 Music Level	12	0	150

Filter: Temperature

Room1 Temperature	123	0	200
Room2 Temperature	-8	-100	100
Room3 Temperature	15	0	150
Room4 Temperature	2	0	10



Exemple 2

Pour utiliser des drapeaux afin de définir les paramètres à faire figurer à l'intérieur du tableau :

Tout d'abord, ajoutez une nouvelle colonne à l'intérieur de la source de données qui contiendra les drapeaux et activera la ligne associée. Ensuite, associez le lien de données du filtre de tableau à la nouvelle colonne contenant les drapeaux

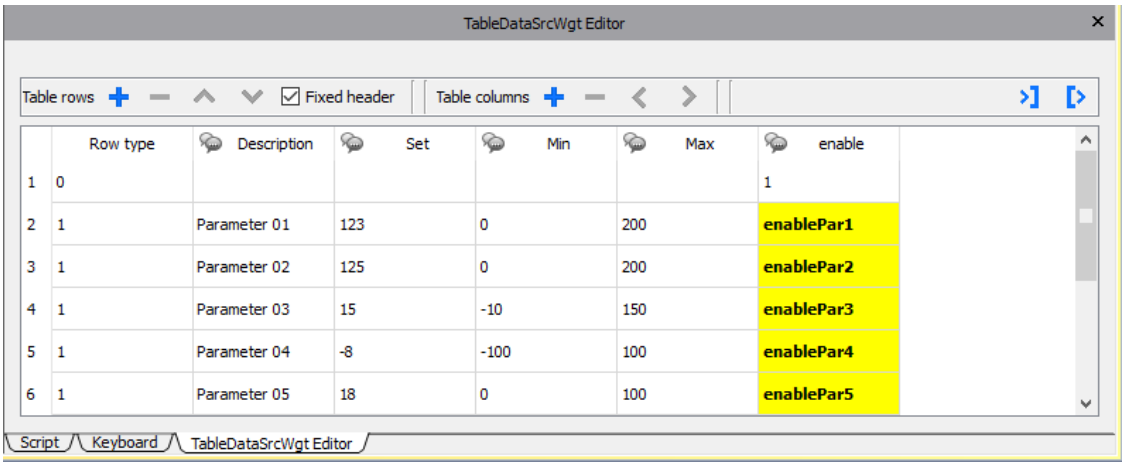
Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

enablePar1 = 1
enablePar2 = 1
enablePar3 = 1
enablePar4 = 1
enablePar5 = 1

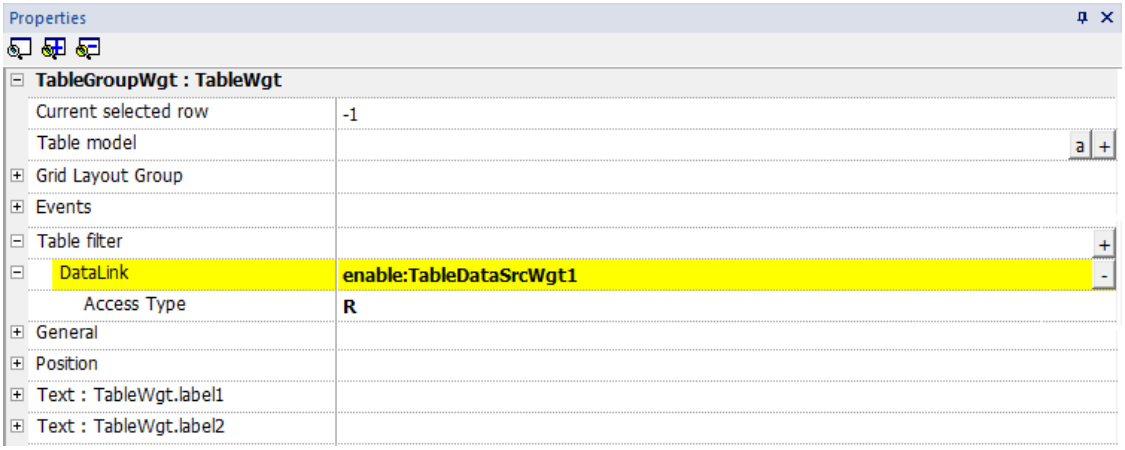
Description	Set	Min	Max
Parameter 01	123	0	200
Parameter 02	125	0	200
Parameter 03	15	-10	150
Parameter 04	-8	-100	100
Parameter 05	18	0	100
Parameter 06	12	0	150
Parameter 07	15	0	150
Parameter 08	134	0	500

enablePar1 = 1
enablePar2 = 1
enablePar3 = 0
enablePar4 = 0
enablePar5 = 1

Description	Set	Min	Max
Parameter 01	123	0	200
Parameter 02	125	0	200
Parameter 05	18	0	100
Parameter 06	12	0	150
Parameter 07	15	0	150
Parameter 08	134	0	500
Parameter 09	44	0	50
Parameter 10	2	0	10

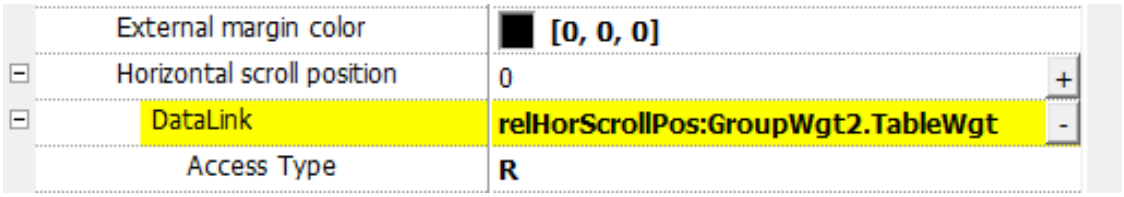


Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



Position défilement horizontal

La « Position du défilement horizontal » donne la possibilité de synchroniser les mouvements de défilement horizontal de deux tableaux.



Le paramètre de position du défilement horizontal n'est disponible qu'en mode Vue des propriétés avancées

Pages préchargées

Normalement, le HMI Runtime ne récupère que les données qui seront visibles à l'écran. Pour rendre le défilement des tableaux plus agréable, il peut être utile de précharger les données des lignes suivantes et précédentes des lignes affichées. En utilisant le paramètre Pages préchargées, il est possible de définir combien de pages seront préchargées

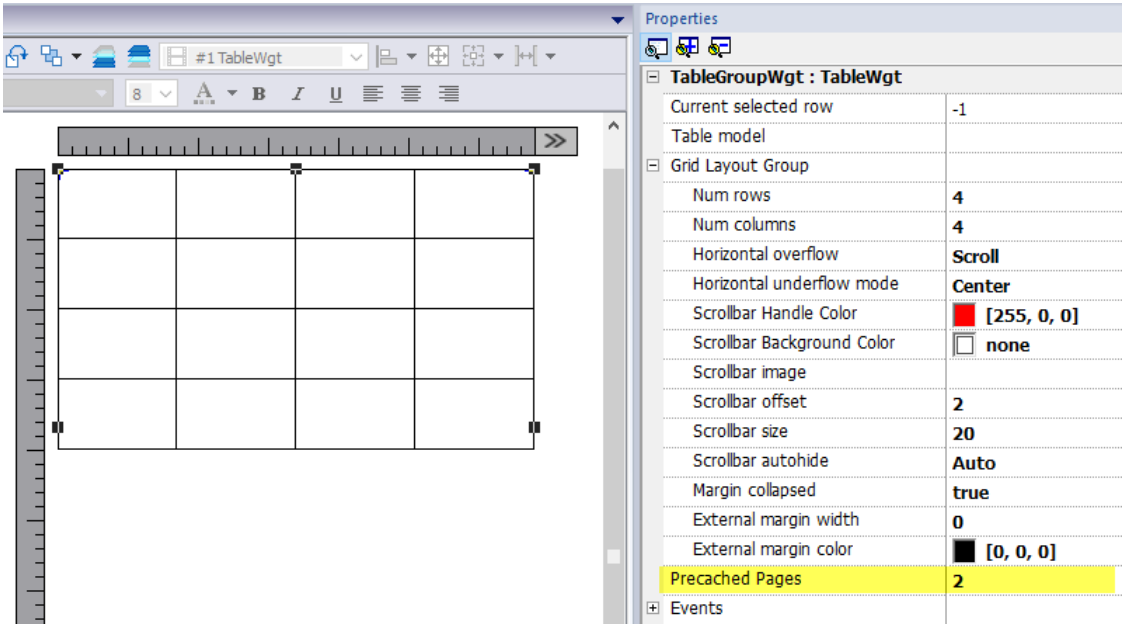
- 0 = aucune page préchargée
- N = nombre de pages à précharger

Exemple :

Pour un tableau de 4 lignes et de pages préchargées = 2

- Le nombre de lignes à précharger est de 8 (2 pages x 4 lignes).
- 4 lignes avant (pour être prêt à gérer le défilement du tableau vers le haut)
- 4 lignes au-dessus (pour être prêt à gérer le défilement du tableau vers le bas)

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



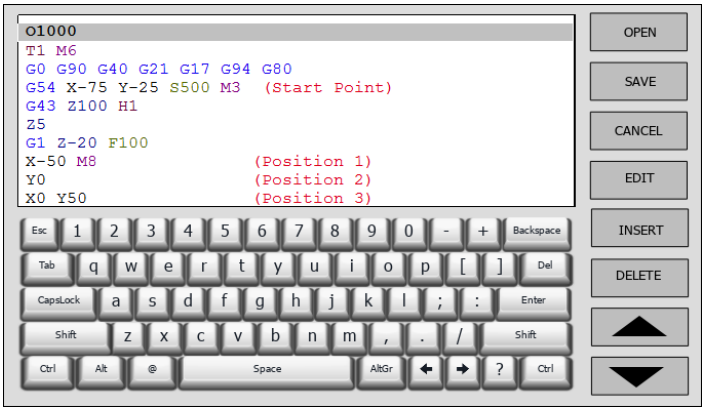
Le paramètre Pages préchargées n'est disponible qu'en mode Vue des propriétés avancées

Widget ÉditeurTexte

*Chemin : **Galerie des widgets**> **Avancé**> **Éditeur***

Utilisez ce widget pour éditer des fichiers textes .Le widget peut charger le fichier texte depuis le périphérique IHM local ou télécharger le fichier depuis un périphérique distant à l'aide d'une connexion ftp.

Remarque : le widget TextEditor est disponible en tant que module d'excécution (voir"Module d'exécution (plug-in)" Page 75 pour les détails) à télécharger vers le HMI Runtime uniquement si nécessaire.



Boutons widget

Bouton	Description
Ouvrir	Charger le fichier texte dans TextEditor
Sauvegarder	Sauvegarder le fichier texte
Annuler	Supprimer toutes les modifications résultant de la dernière commande OUVRIR ou SAUVEGARDER
Éditer	Entrer dans le mode Éditer
Insérer	Insérer une nouvelle ligne
Supprimer	Supprimer la ligne en cours
Haut/bas	Déplacer le curseur vers le Haut/bas

Propriétés du widget

Option	Description
Clavier	Le widget TextEditor est pourvu d'un clavier intégré. Lorsque le widget est utilisé sans le clavier intégré, le clavier alphabétique s'affichera lorsque vous entrerez en mode d'édition. • Caché • Visible
Config FTP	Paramètres FTP pour télécharger le fichier texte depuis un serveur FTP distant. Laissez ce champ vide pour charger le fichier texte depuis le périphérique IHM.

Paramètre	Description
Adresse FTP	Adresse IP du serveur FTP.
Port de serveur	Port pour connexion FTP (par défaut = 21).
Authentification	Choisissez l'authentification FTP à utiliser : • Normal (nom d'utilisateur et mot de passe requis) • Anonyme
Identifiant	Nom d'utilisateur du compte FTP distant
Mot de passe	Mot de passe du compte FTP distant

Nom de fichier	Nom de fichier à modifier. Si vide, un navigateur de fichiers servant à charger un fichier local s'ouvrira
Type de surbrillance syntaxique	Affiche le texte en différentes couleurs selon la langue choisie • Aucun • GCode

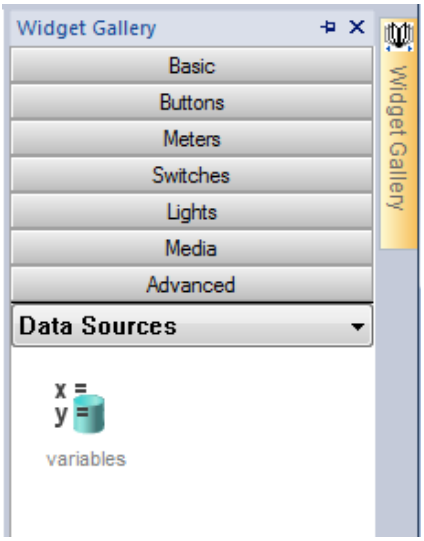
Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Widget de variables

Chemin : ***Galerie de widgets*** *>* ***Avancé*** *>* ***Source de données***

Utilisez ce widget pour ajouter des variables internes aux opérations telles que le transfert de données ou à utiliser dans des programmes JavaScript.

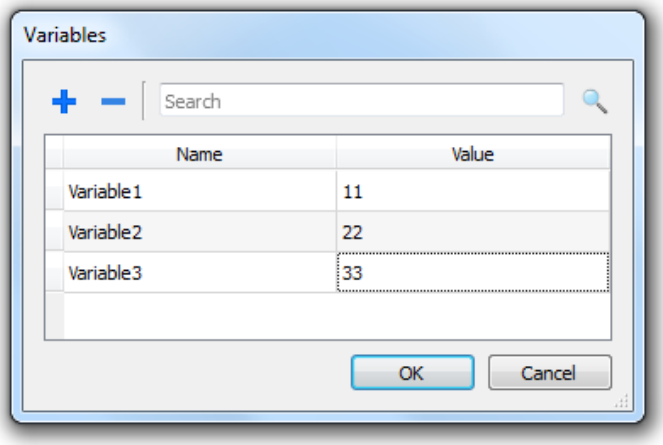
Remarque : les variables sont locales dans la page où le widget a été inséré.



Lorsque vous glissez-déposez ce widget dans votre page, un caractère de position sera affiché pour indiquer l'emplacement du widget, mais il ne sera pas visible au runtime.

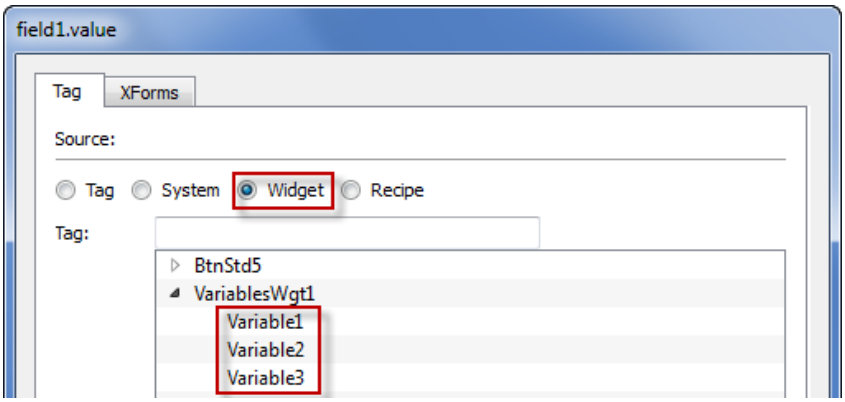
Paramétrage du widget

Pour créer des variables et attribuer des valeurs à celles-ci, ouvrir les boîtes de dialogue **Variables** depuis la propriété **Variables** dans le volet **Propriétés**.

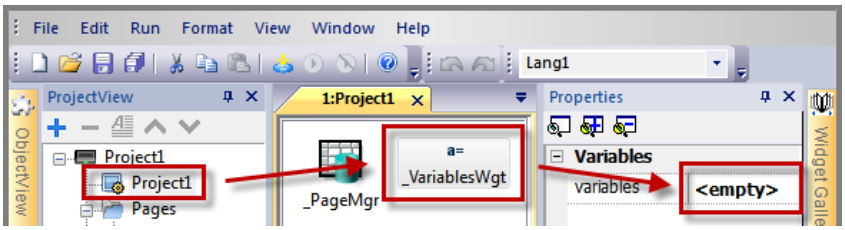


Ces variables peuvent ensuite être référencés à partir de la boîte de dialogue de **variable Attacher**, à partir de l'éditeur de page.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



Si vous avez besoin des variables globales, les configurer au niveau du projet, en ajoutant les variables souhaitées au widget variable global.



Utiliser les variables dans JavaScript

Les variables peuvent également être référencées dans les programmes JavaScript avec la syntaxe suivante :

Pour les variables locales :

```
var varWgt = page.getWidget("_VariablesWgt");
var compVar = varWgt.getProperty("VariableName");
```

Pour les variables globales :

```
var varWgt = project.getWidget("_VariablesWgt");
var compVar = varWgt.getProperty("VariableName");
```


Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

35 Widgets personnalisés

PB610 Panel Builder 600 a une grande bibliothèque de widgets qui inclut des widgets dynamiques prédéfinis (boutons, voyants, jauges, sélecteurs, tendances, recettes et éléments de boîte de dialogue), ainsi que des images statiques (formes, tuyaux, réservoirs ou moteurs).

Vous pouvez glisser et déposer un objet de la galerie vers la page, puis le dimensionner, le déplacer, le pivoter ou le transformer. Tous les widgets de la galerie sont vectoriels, ils ne perdent donc pas de définition lorsqu'ils sont redimensionnés.

Cependant, vous pouvez modifier tous les widgets prédéfinis pour créer votre widget personnalisé. Les widgets personnalisés peuvent être créés à partir de plusieurs éléments incluant seulement les propriétés nécessaires pour leur fonctionnement.

Créer un widget personnalisé **412**

Ajouter des propriétés à un widget personnalisé **415**

Utilisation de variables structurées **417**

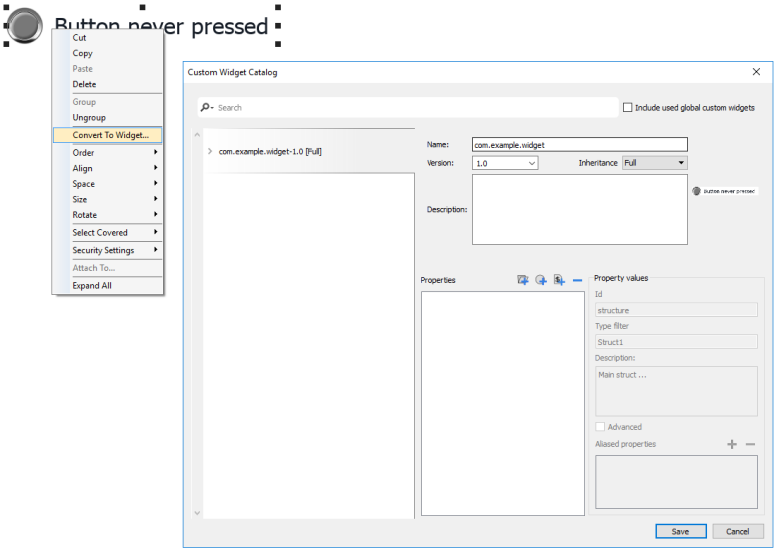
JavaScript dans les widgets personnalisés **419**

Galerie de l’utilisateur **422**

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Créer un widget personnalisé

1. Glissez et déposez sur une page tous les widgets que vous voulez utiliser pour composer votre widget personnalisé.
2. Sélectionnez-les et groupez-les.
3. Faites un clic droit sur l'objet groupé et sélectionnez le **Convertir en Widget** : la boîte de dialogue **Catalogue de Widgets personnalisés** s'affiche.



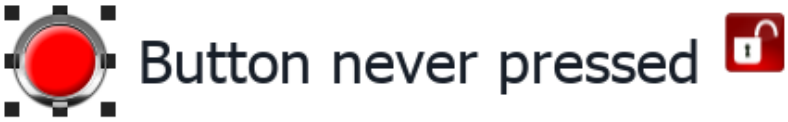
Paramètre	Description
Inclut des widgets personnalisés utilisés	Une fois coché, lister tous les widgets utilisés à l'intérieur du projet. Même les widgets du système.
Nom	Vous pouvez définir tout ce que vous préférez, mais il est commun de garder une structure de nom. Le dossier com.hmi est réservé aux widgets du système
Description	Description du widget.
Version	Version du widget. Tous les widgets qui partagent la même version partagent les propriétés définies à partir du paramètre d'héritage.
Héritage	Propriétés partagées entre widgets avec la même version • Toutes (aussi bien graphique que logique) • Uniquement graphique • Uniquement logique • Désactiver

Modifier un widget personnalisé

Pour modifier un widget personnalisé, il suffit de double-cliquer pour sélectionner le widget personnalisé en mode édition.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Si le drapeau Héritage est activé, une icône de verrouillage apparaîtra pour vous avertir que vous êtes en train d'ajouter des changements qui seront propagés à tous les autres widgets personnalisés partageant la même version. Cliquez sur l'icône du cadenas pour confirmer l'entrée en mode d'édition, le cadenas sera ouvert. Cliquez à nouveau lorsque la modification sera effectuée.

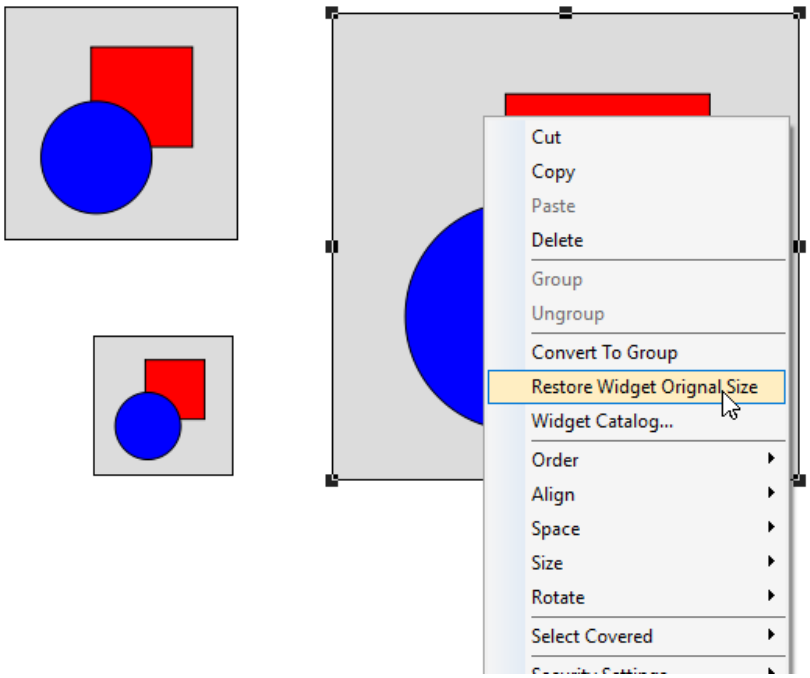


Le cadenas n'est affiché que lorsque l'héritage est activé.

Redimensionner un widget personnalisé

Lorsque les dimensions d'un widget personnalisé sont modifiées, les nouvelles dimensions ne se propageront pas à d'autres widgets.

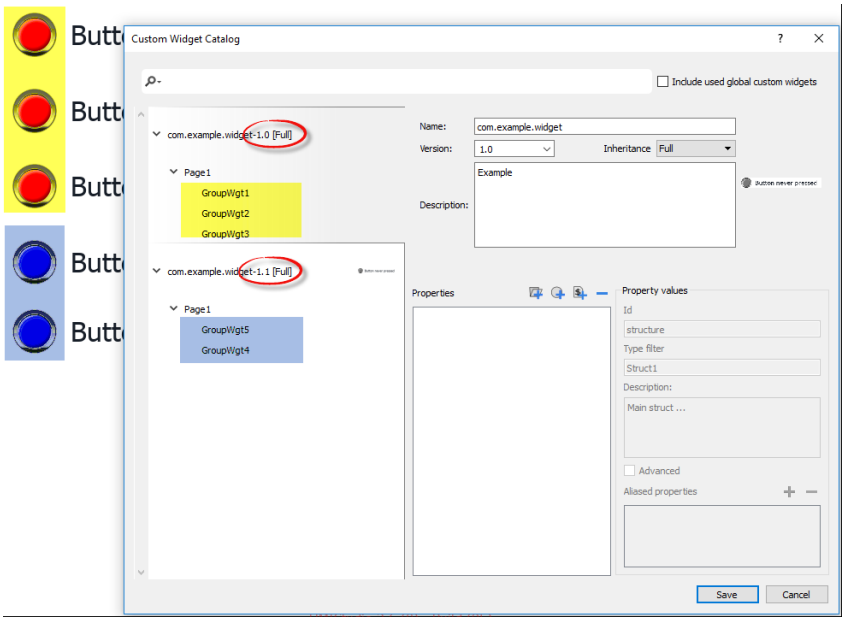
La commande « Restaurer le format original du widget » peut être utilisée.



Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Partager les propriétés

Lorsqu'un widget personnalisé est modifié, toutes les modifications seront propagées à tous les autres widgets personnalisés qui partage la même version et qui sont configurés pour hériter des propriétés du widget.

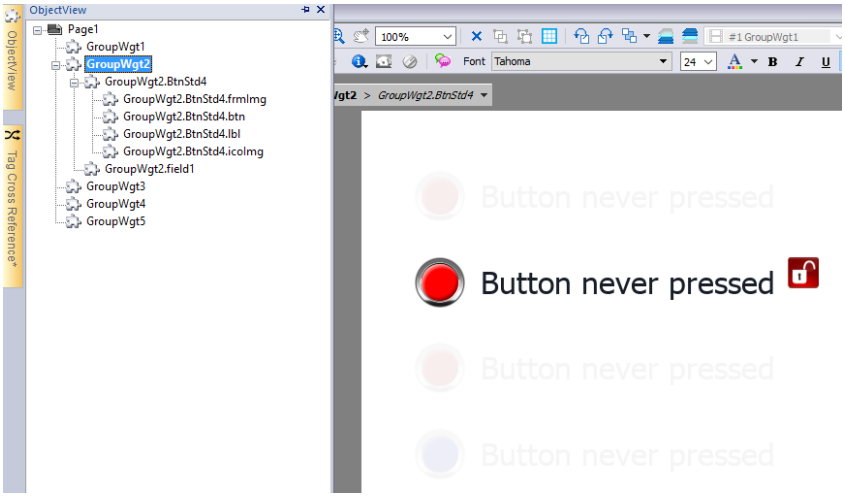


Utiliser des composants de widgets

En général, les widgets sont composés de nombreuses parties, par exemple un bouton est un widget complexe qui inclut deux widgets d'image, un widget de bouton et une étiquette.

Pour afficher une liste de tous les éléments qui font partie d'un widget, sélectionnez le widget, ouvrir le cadenas et ouvrir le volet **VueObjet** : tous les éléments composant un widget complexe sont listés en ordre hiérarchique.

Pour sélectionner un seul widget, le sélectionner directement depuis le volet **VueObjet**.

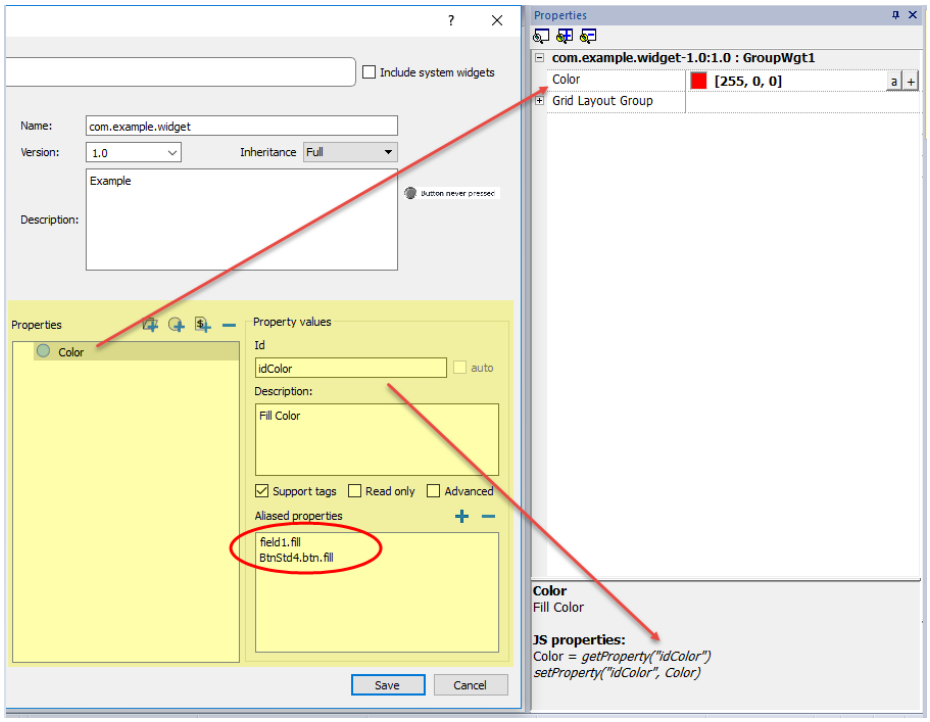


Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

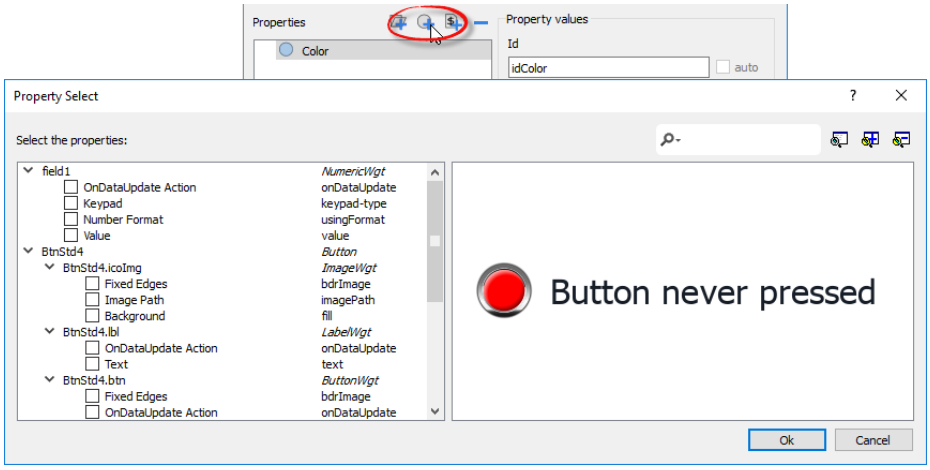
Ajouter des propriétés à un widget personnalisé

Lorsque vous créez un widget personnalisé, vous devez définir les propriétés qui seront affichées pour ce widget dans le volet **Propriétés**.

1. Faites un clic droit sur l'objet groupé et sélectionnez le **Catalogue Widget** : la boîte de dialogue des propriétés s'affichent.



2. Cliquez sur **+** pour ouvrir la boîte de dialogue **Sélection des propriétés** : elle liste toutes les propriétés de tous les widgets groupés.



3. Sélectionnez les propriétés que vous voulez définir pour votre widget personnalisé.
4. Définissez les détails de chaque propriété.

Notez que vous pouvez créer des dossiers et utiliser la fonction glisser-déposer pour déplacer ou de réorganiser la liste des **Propriétés**

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Paramètre	Description
Propriétés	Nom affiché dans le volet **Propriétés**.
Description	Tout commentaire sur la propriété à afficher dans le volet **Propriétés**.
ID	Le nom affiché par PB610 Panel Builder 600 à des fonctions JavaScript et à la boîte de dialogue variable associée.
Prise en charge des variables	Spécifie si la propriété prend en charge l'attribut « Associer à ».
Lecture seule	Propriété exposée uniquement en mode de lecture
Paramètres avancés	Spécifie si chaque propriété doit apparaître dans le mode d'affichage avancé ou simple du volet **Propriétés**.
Propriétés dénommées	Propriétés internes liées à la propriété exposée

Associer des propriétés

Pour associer deux propriétés ou plus :

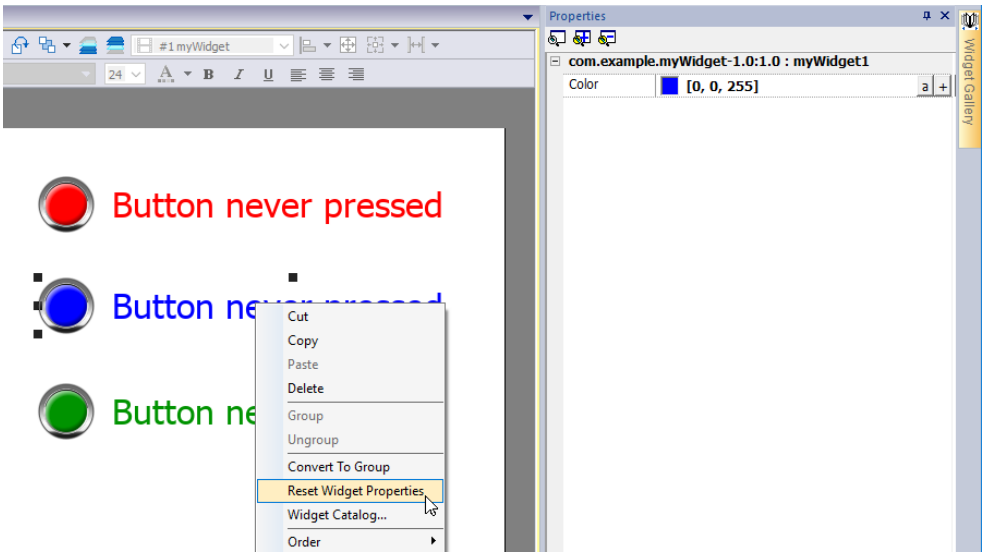
1. Sélectionnez la propriété primaire dans la boîte de dialogue de la liste des **Propriétés.**
2. Cliquez sur **+** dans la barre d'outils **propriétés alias** : la boîte de dialogue **Sélectionner propriété** s'affiche.
3. Sélectionnez les propriétés que vous voulez associer.
4. Cliquez sur **OK :** les attributs combinés s'afficheront dans la zone de liste **Propriétés alias**.

Exemple

Si vous insérez dans une propriété « Couleur », la couleur de remplissage de tous les widgets (par ex., filed1.fill et BtnStd4.btn.fill) lorsque vous définissez la propriété Couleur exposée du widgets personnalisé, toutes les couleurs des widgets inclus changeront.

Réinitialiser les propriétés du widget

La fonction « Réinitialiser les propriétés du widget » réinitialise les valeurs des propriétés modifiées aux valeurs d'origine.

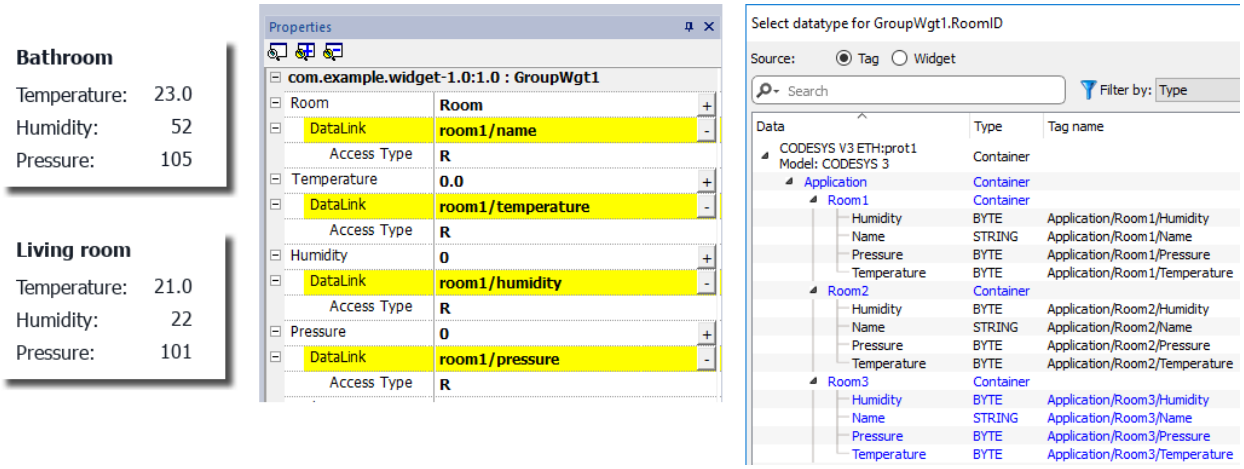


Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

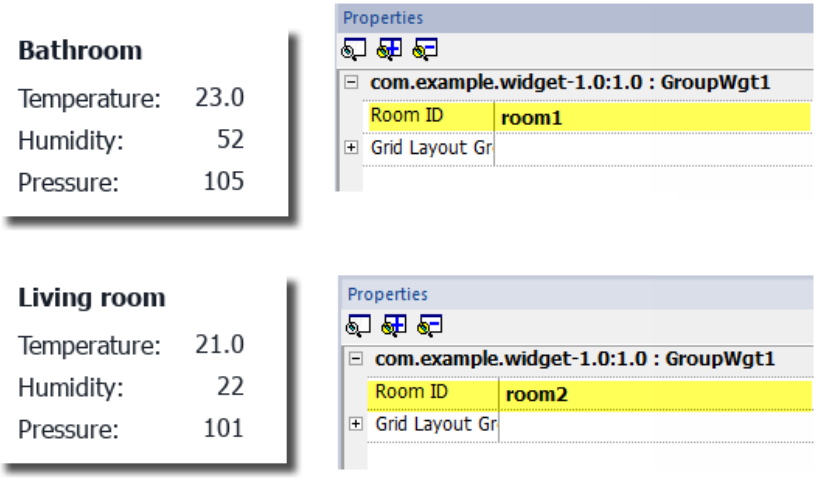
Utilisation de variables structurées

Un problème commun en utilisant un widget qui utilisent de nombreuses variables est la nécessité de créer des instances du widget en donnant uniquement le nom de la variable de la structure qui contient toutes les variables au lieu de configurer chaque variable individuelle.

Par exemple, considérez le widget ci-dessous. Il utilise quatre variables, le nom de la pièce, la température, l'humidité et la pression. Si nous voulons utiliser deux instances de ce widget pour deux pièces différentes, nous devons configurer huit variables, quatre variables pour chaque pièce.

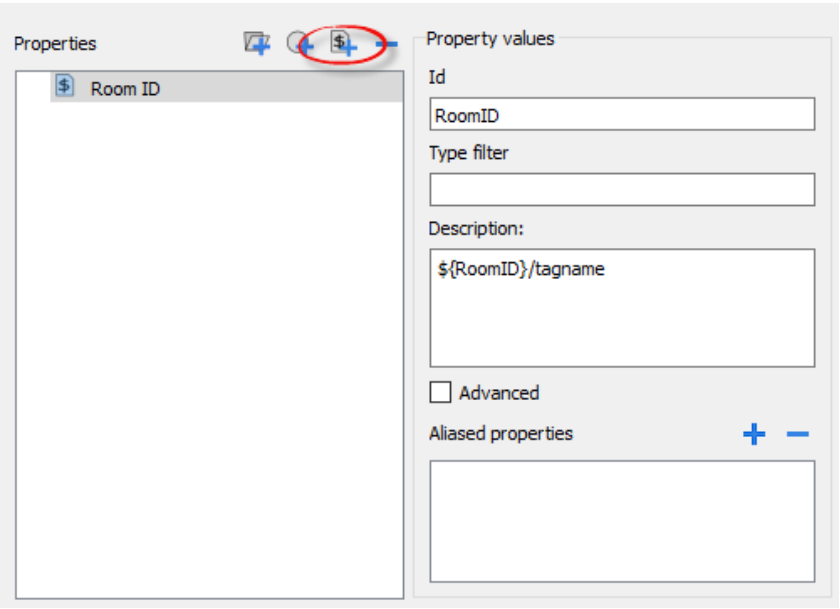


En utilisant une propriété de **paramètre**, il est possible de définir toutes les liaisons de données du widget en donnant uniquement le nom de la structure.

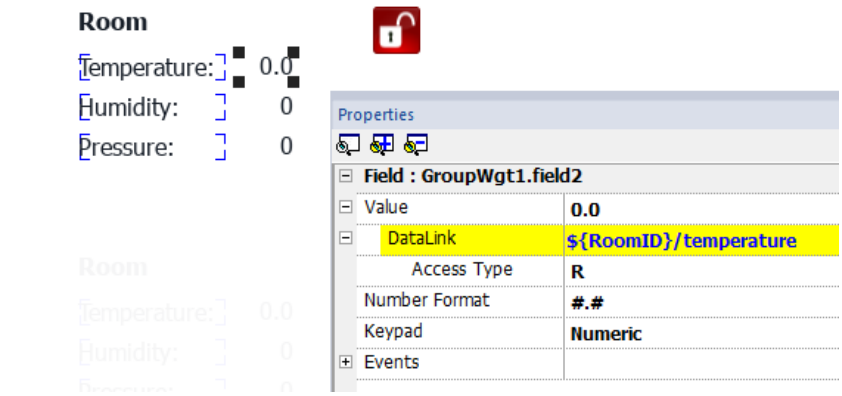


Un champ « Paramètre » peut être ajouté à l'intérieur du widget personnalisé en utilisant l'icône « *Ajouter Paramètre* » :

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

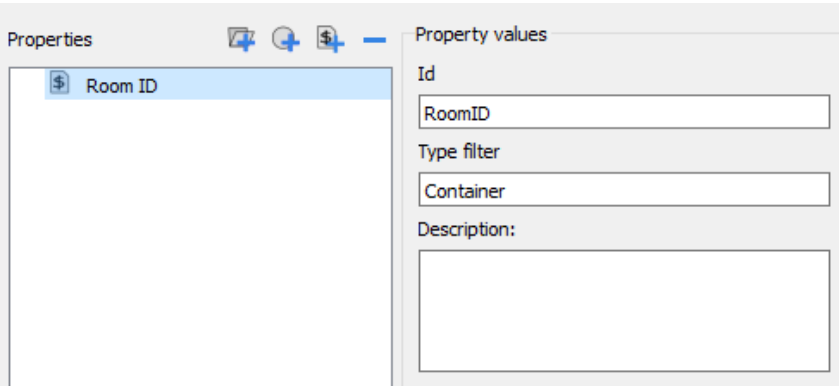


Pour configurer les liaisons de données du widget personnalisé, le mot-clé ${RoomID} peut être utilisé pour faire référence à l'instance de structure

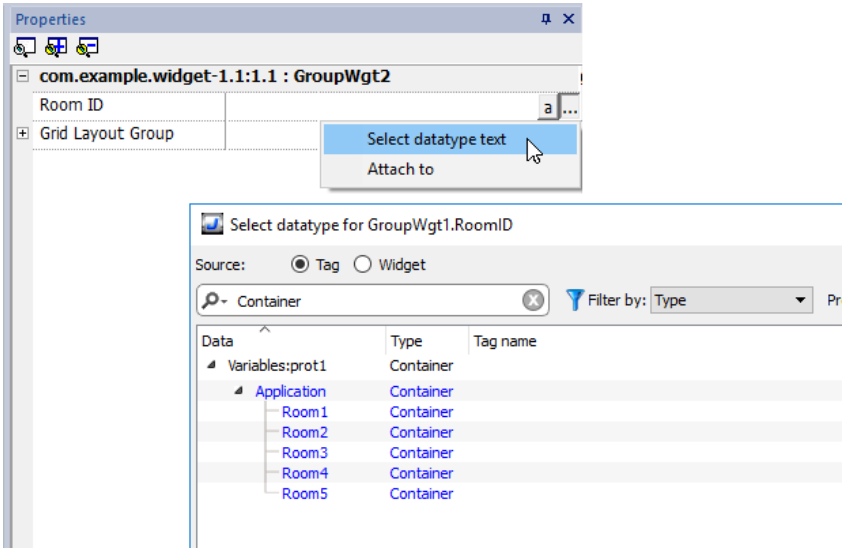


Filtre type

Généralement, la valeur du paramètre sera un élément d'une variable structurée. À l'aide du paramètre *« type filter »*, *« Select datatype text »* listera les variables filtrées.



Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



« Select datatype text » retournera une chaîne alors que *« Attach to »* retournera un lien de données vers une variable qui contiendra la chaîne de caractères à utiliser.

getParameter

À l'aide de JavaScript, vous pouvez lire la valeur des paramètres à l'aide de la commande getParameter()

```
object getParameter(paramID)
```

Exemple :

```
var myWidget = page.getWidget("myWidget");
function BtnStd3_btn_onMouseClick(me, eventInfo)
{
     alert("Room is: " + myWidget.getParameter("RoomID"));
}
```

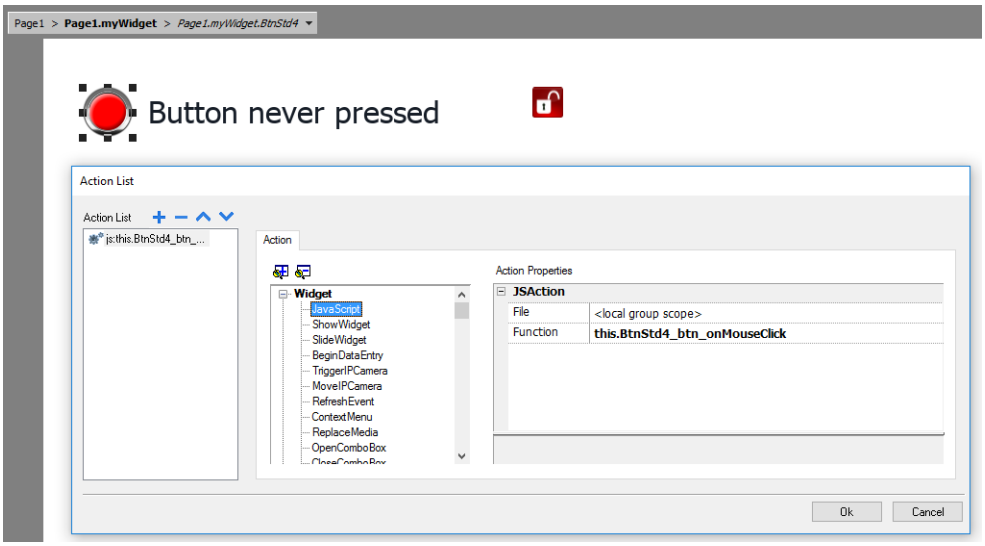
Vous pouvez aussi utiliser getProperty(), mais getParameter() est plus efficace pour lire les paramètres des widgets personnalisés

JavaScript dans les widgets personnalisés

Les fonctions JavaScript peuvent être intégrées dans les widgets personnalisés.

Après avoir double-cliquer sur le widget personnalisé et cliqué sur le cadenas, le mode d'édition est actif et il est possible d'associer le code JavaScript pour les événements disponibles.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

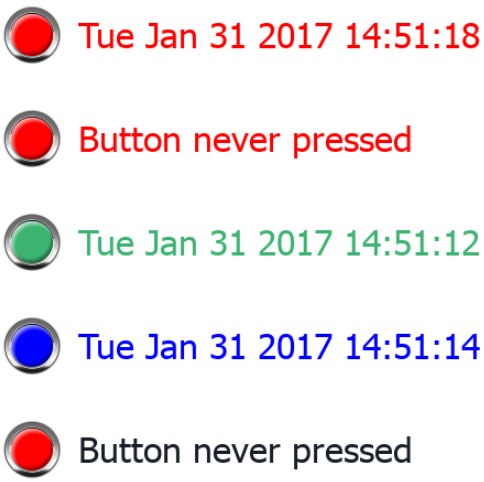


Noter l'utilisation de l'opérateur **this.** qui est nécessaire pour permettre l'instance multiple du widget personnalisé.

Si vous avez besoin de faire référence à un élément du widget, vous pouvez utiliser le mot-clé **wgt.** Par exemple, utilisez wgt.id pour faire référence à l'id de l'instance active du widget.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Si vous coupez et collez des instances du widget personnalisé de l'exemple ci-dessus et les exécutez, p. ex. dans le simulateur, vous obtiendrez le résultat ci-dessous.



Propriété onActivate

Pour initialiser le widget personnalisé, il est possible de définir la propriété onActive avec une fonction d'initialisation comme dans l'exemple ci-dessous.

La fonction onActivate() sera exécutée au chargement de la page.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Script

```
/*!
javascript module: widget-1.0.js
javascript source file path: lib\com\example\widget-1.0\widget-1.0.js
*/

this.wMSG = wgt.getWidget(wgt.id+".field1")

this.BtnStd1_btn_onMouseClick = function (me, eventInfo)
{
    var now = new Date();
    this.wMSG.setProperty("value", now.toString().slice(0, 24));
}


this.onActivate = function()
{
    this.wMSG.setProperty("value", "Button never pressed");
};
this.onActivate();
```

Noter que le widget personnalisé peut aussi passer à l'intérieur de la galerie de l'utilisateur pour une réutilisation ultérieure.

Code JavaScript utilisé dans les exemples de ce chapitre

```
/*! javascript module: widget-1.0.js javascript source file path:
lib\com\example\widget-1.0\widget-1.0.js */ this.wMSG = wgt.getWidget
(wgt.id+".field1") this.BtnStd1_btn_onMouseClick = function (me, eventInfo) { var now
= new Date(); this.wMSG.setProperty("value", now.toString().slice(0, 24)); }
this.onActivate = function() { this.wMSG.setProperty("value", "Button never pressed");
}; this.onActivate();
```

Galerie de l'utilisateur

Vous pouvez sauvegarder les widgets créés par les développeurs dans la galerie des widgets, afin de pouvoir les utiliser durant le développement de nouveaux projets.

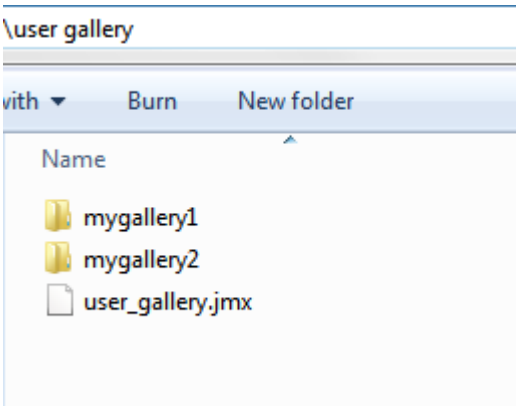
Barre d'outils widgets utilisateur

Commande	Description
	Ouvrez le dossier des widgets sélectionnés dans l'éditeur PB610 Panel Builder 600
	Ajoutez un nouveau dossier de widgets
	Effacez le dossier actuellement sélectionné
	Sélectionnez le dossier des widgets utilisateur

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Pour ajouter un nouveau widget dans la galerie utilisateur, ouvrez le dossier de widgets, puis éditez la page de galeries en créant ou en ajoutant le nouveau widget.

Conseil : pour importer un sous-dossier galerie utilisateur, copiez simplement le dossier à importer et collez-le dans le principal fichier galerie utilisateur.



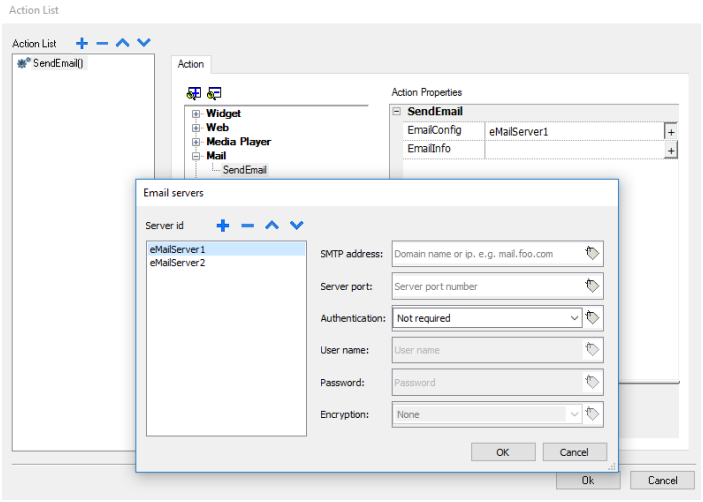
Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

36 Envoyer un message par courriel

Envoyer des courriels à l'aide de l'action EnvoyerCourriel, y compris des variables dans le corps et des pièces jointes.

L'action EnvoyerCourriel a été créée pour fonctionner avec des alarmes et des planificateurs, mais peut être déclenchée et exécutée par de nombreux autres événements.



Configurer le serveur e-mail **426**

Configurer les e-mails **426**

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Configurer le serveur e-mail

Pour configurer le serveur e-mail, entrez les informations suivantes pour le paramètre **ConfigEmail** :

Paramètre	Description
Adresse SMTP	Adresse de serveur SMTP.
Port de serveur	Port pour la connexion au serveur SMTP (par défaut = 25).
Requiert authentification	Sélectionnez ce paramètre si le serveur SMTP requiert une authentification.
Identifiant	Identifiant pour envoyer des e-mails avec le serveur SMTP.
Mot de passe	Mot de passe pour envoyer des e-mails avec le serveur SMTP.
Cryptage	Type de cryptage (aucun ou SSL).

Cliquez sur **+** pour ajouter plus de serveurs e-mails.

Conseil : utilisez les variables si vous voulez changer les paramètres du serveur de manière dynamique depuis le HMI Runtime.

Configurer les e-mails

Entrez les informations suivantes pour le paramètre **InfoEmail** :

Paramètre	Description
Nom	Optionnel, cette information sert uniquement pour l'enregistrement.
Description	Optionnel, cette information sert uniquement pour l'enregistrement.
De	Optionnel, adresse e-mail de l'expéditeur (par exemple, John@domain.com).
À	Adresses électroniques des destinataires. Pour entrer plusieurs adresses, séparez-les avec un point-virgule.
Objet	Objet de l'e-mail.
Pièce jointe	Le chemin du fichier est envoyé en pièce jointe. Seule une pièce jointe à la fois peut être envoyée. Remarque : la taille maximale des pièces jointes est généralement définie par le serveur SMTP.
Texte	Contenu principal de l'e-mail. Vous pouvez y insérer des variables live lorsque vous les incluez entre crochets. Par exemple, un corps de message « Valeur Tag1 est [Tag1] » est envoyé comme « Valeur Tag1 est 45 », si la valeur actuelle de Tag1 est 45.

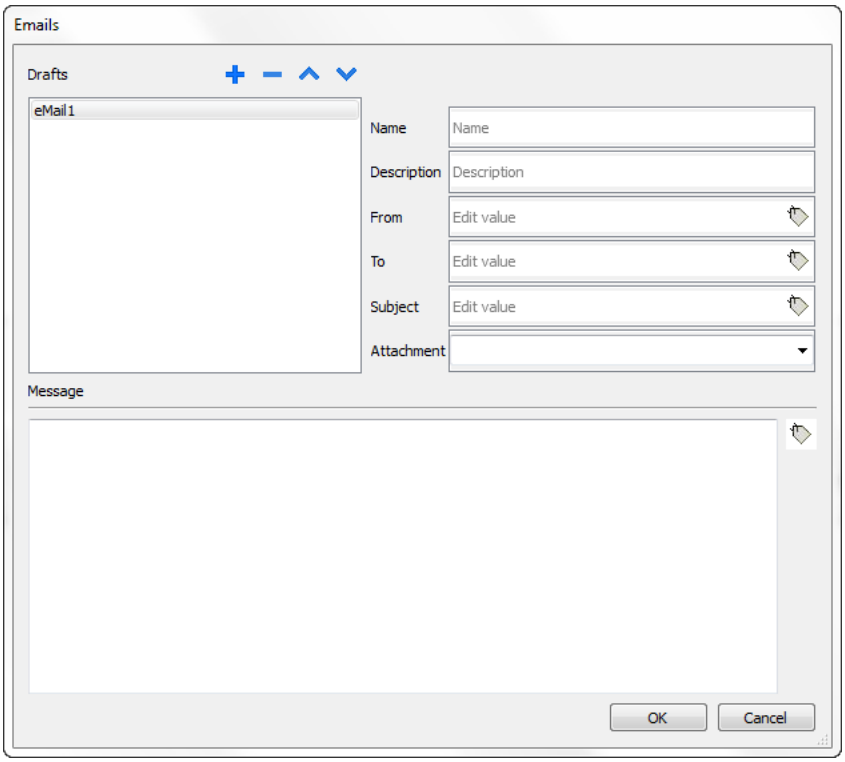
Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Conseil : associez une variable de chaîne aux champs **De**, **À** et **Objet** pour que leurs valeurs soient changées dans le HMI Runtime.

AVERTISSEMENT : la taille maximale du corps du message est de 4096 octets, le texte qui dépasse sera tronqué.

Ajouter des modèles d'e-mails

Cliquez sur **+** pour ajouter plus de modèles.



Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

37 JavaScript

L'objectif de cette section est de décrire comment est utilisé JavaScript dans les applications PB610 Panel Builder 600. Il n'a en revanche pas pour but d'expliquer le langage JavaScript lui-même.

JavaScript PB610 Panel Builder 600se base sur le langage de programmation ECMAScript http://www.ecmascript.org , tel que défini par la norme ECMA-262.

Si vous maîtrisez JavaScript, vous pouvez utiliser le même type de commandes dans PB610 Panel Builder 600 que celles que vous utiliseriez dans un navigateur Web. Si vous ne maîtrisez pas le langage ECMAScript, veuillez consulter la page :

https://developer.mozilla.org/en/JavaScript

Éditeur JavaScript **431**

Exécution des fonctions JavaScript **431**

Événements **433**

Widget événements **434**

Événements de page **436**

Événements système **437**

Objets **439**

Widget objets de classe **439**

Propriétés du widget **440**

Méthodes du widget **443**

Objet de page **444**

Propriétés de l'objet de page **444**

Méthodes associées à l'objet de page **445**

Objet de groupe **448**

Méthodes d'objet de groupe **448**

Objet du projet **449**

Propriétés d'objet du projet **449**

Méthodes d'objet du projet **449**

Widgets d'objet du projet **458**

Imprimer l'objet des rapports **459**

Objet de statut **462**

Méthodes pour objet de statut **463**

Mots-clés **464**

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

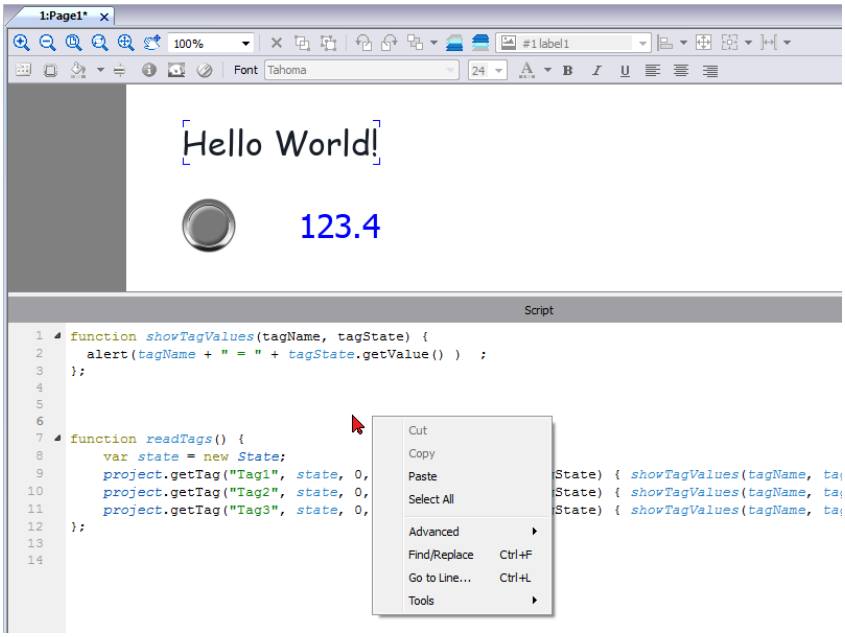
Fonctions générales **464**
Prendre en main la lecture/l'écriture de fichiers **465**
Se connecter à partir de JavaScript **468**
Limites lors du travail avec des widgets dans JavaScript **470**
Déboguer JavaScript **470**

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Éditeur JavaScript

PB610 Panel Builder 600 inclut un éditeur JavaScript puissant.

Double-cliquez sur l'éditeur pour afficher l'ensemble des commandes disponibles.



Exécution des fonctions JavaScript

Les fonctions JavaScript sont exécutées lorsque des événements se produisent. Par exemple, un utilisateur peut définir un script pour l'événement OnMouseClick et le script JavaScript est exécuté lorsque vous appuyez sur le bouton sur le périphérique IHM.

Les fonctions JavaScript sont exécutées uniquement lorsque l'événement programmé se produit et n'est pas cyclique. Cette approche minimise le traitement requis pour exécuter une logique dans le périphérique IHM.

PB610 Panel Builder 600 fournit un moteur JavaScript fonctionnant au niveau du client. Chaque page de projet peut contenir des scripts ayant une étendue locale sur la page où ils sont ajoutés. Vous pouvez créer des scripts globaux pour qu'ils soient exécutés par des événements de planificateur ou par des événements d'alarme.

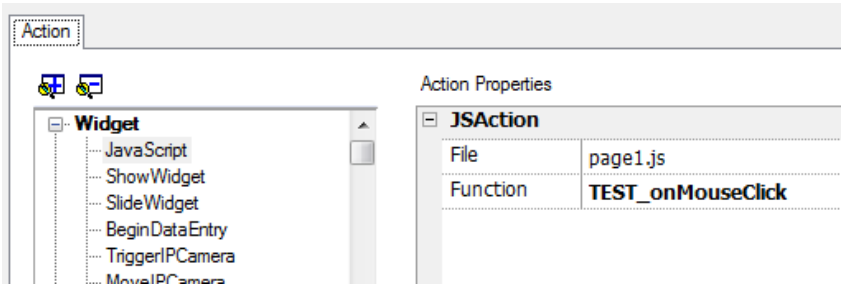
Dans les deux cas, les scripts sont exécutés sur le client. Cela signifie que lorsque plus d'un client est connecté au périphérique IHM (pour un ordinateur externe exécutant le HMI Client), chaque client exécute le même script, fournissant des résultats de sortie différents selon l'entrée, puisque les entrées fournies à différents clients peuvent être différentes.

Par exemple, si un script agit selon la position d'une barre de défilement et que cette position est différente sur les différents clients, le résultat du script sera différent sur chaque client.

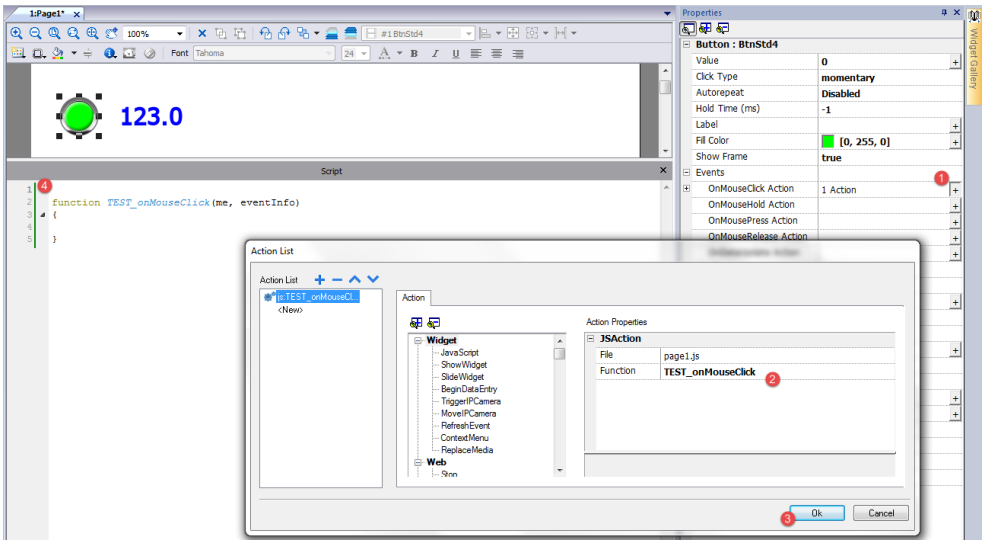
Fonctions JavaScript pour événements de page

L'éditeur JavaScript s'ouvre lorsque vous ajoutez une action JavaScript dans une liste d'actions.

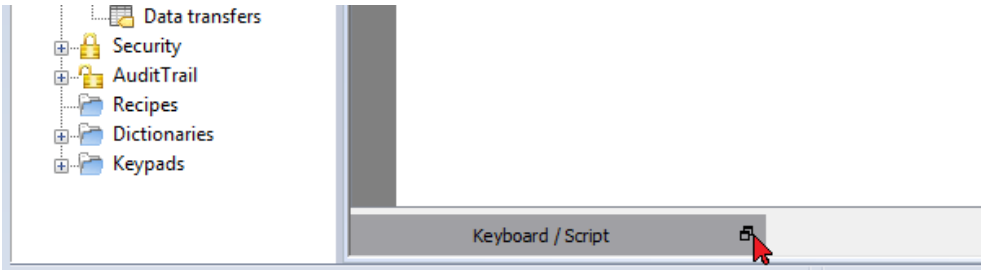
Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



1. Sélectionnez l'événement qui exécute l'action.
2. Ajoutez une action **JavaScript** de la catégorie **Widget**.
3. Laissez le nom de fonction par défaut ou tapez-en un nouveau.
4. Cliquez sur **OK** pour confirmer : l'éditeur JavaScript affiche la structure de votre fonction.



Vous pouvez aussi ouvrir l'éditeur JavaScript depuis l'onglet **Script** situé en bas de l'espace de travail.

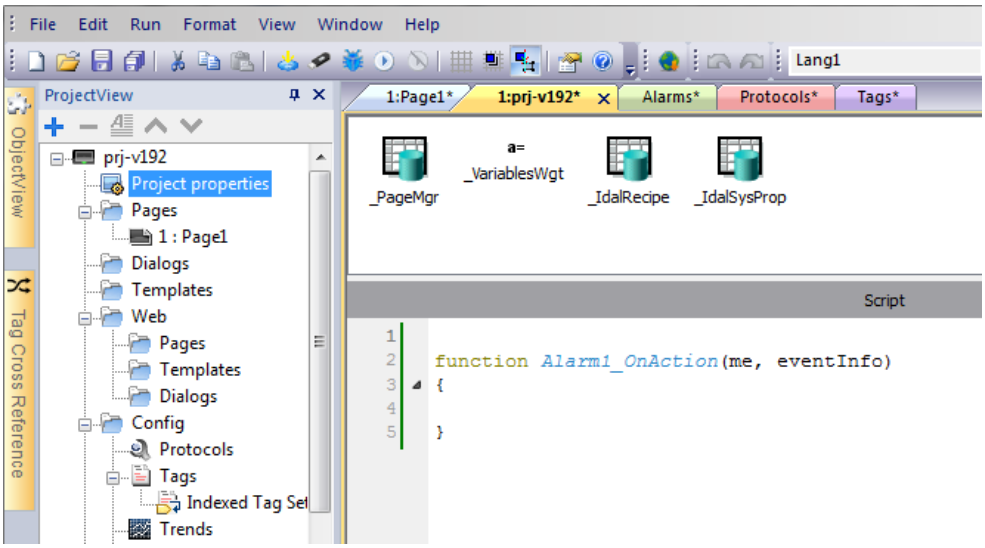


Fonctions JavaScript pour les alarmes et les événements programmés

Le code JavaScript, associé aux alarmes et aux événements, qui n'est pas associé à une page spécifique peut être édité depuis la page principale **Propriétés projet**.

*Chemin : **VueProjet**> double-cliquez sur **Propriétés projet***

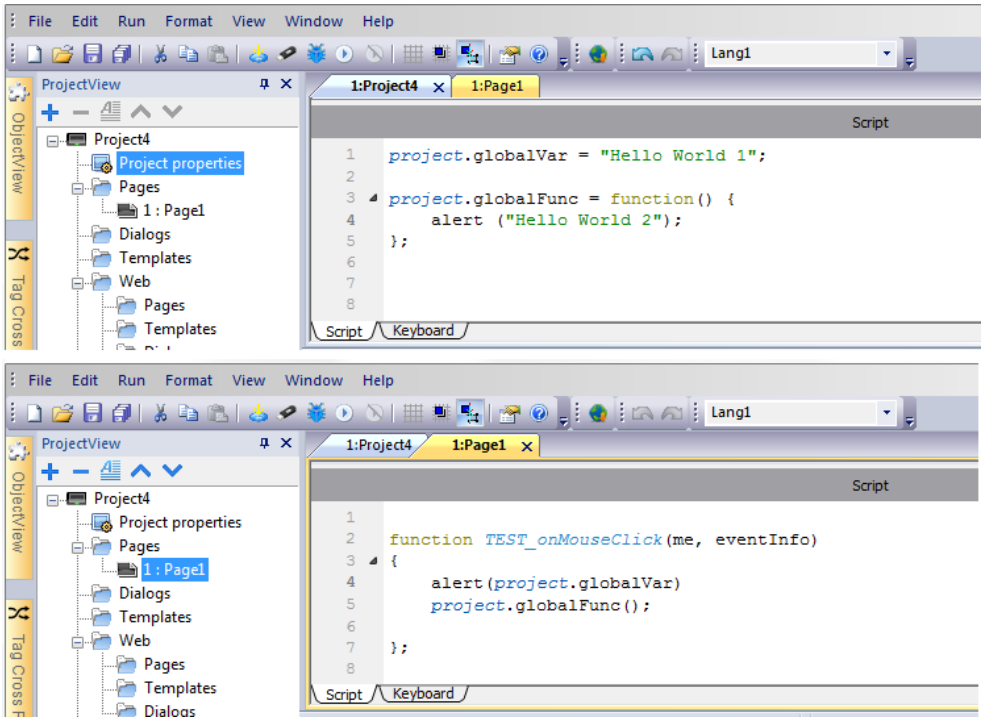
Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



REmarque : les actions JavaScript sont des actions client, elles sont donc exécutées uniquement lorsqu'un client est connecté.

Code JavaScript partagé

La variable globale **projet** peut être utilisée pour partager du code JavaScript entre les pages. Les variables sont créées/initialisées depuis le code JavaScript principal depuis la page **Propriétés projet** principale et peuvent ensuite être utilisées depuis les pages du projet.



Événements

Vous pouvez ajouter JavaScript aux catégories d'événements suivantes :

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

- Événements de widget
- Événements de page
- Événements système

Pour les événements de type :

- Lorsd'AppuiSouris
- LorsdeRelâchementSouris
- LorsdeClicSouris
- LorsdeMolette

Le paramètre **Infos événement** de JavaScript contient les propriétés supplémentaires suivantes :

Paramètre	Description
eventInfo.posX	Coordonnée de la souris locale/touche X par rapport aux coordonnées du widget
eventInfo.posY	Coordonnée de la souris locale/touche Y par rapport aux coordonnées du widget
eventInfo.pagePosX	Coordonnée de la souris/touche de la page X
eventInfo.pagePosY	Coordonnée de la souris/touche de la page Y
eventInfo.wheelDelta	Delta de la molette de la souris. Valeur entière avec un signe représentant la direction de rotation. La valeur réelle est le nombre de rotations en huitièmes de degré. La valeur minimale dépend de la résolution de la souris. Elle est en général de 120, ce qui correspond à 15 degrés.

Widget événements

onMouseClick

```
void onMouseClick (me, eventInfo)
```

Cet événement est disponible uniquement pour les boutons et il se produit lorsque le bouton est enfoncé et relâché rapidement.

Paramètre	Description
me	Objet déclencheur de l'événement
eventInfo	Détails sur l'événement déclenché

```
function buttonStd1_onMouseClick(me, eventInfo) {
     //faire quelque chose…
}
```


Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

onMouseHold

```
void onMouseHold (me, eventInfo)
```

Cet événement est disponible uniquement pour les boutons et il se produit lorsque le bouton est enfoncé et relâché après le nombre de secondes défini comme **temps de maintien** dans les propriétés du widget.

Paramètre	Description
me	Objet déclencheur de l'événement
eventInfo	Détails sur l'événement déclenché

```
function buttonStd1_onMouseHold(me, eventInfo) {
     //faire quelque chose…
}
```

onMousePress

```
void onMousePress (me, eventInfo)
```

Cet événement n'est disponible que pour les boutons et il se produit lorsque vous appuyez sur le bouton.

Paramètre	Description
me	Objet déclencheur de l'événement
eventInfo	Détails sur l'événement déclenché

```
function buttonStd1_onMousePress(me, eventInfo) {
     //faire quelque chose…
}
```

onMouseRelease

```
void onMouseRelease (me, eventInfo)
```

Cet événement est disponible uniquement pour les boutons et il se produit lorsque l'on relâche le bouton.

Paramètre	Description
me	Objet déclencheur de l'événement
eventInfo	Détails sur l'événement déclenché

```
function buttonStd1_onMouseRelease(me, eventInfo) {
     //faire quelque chose…
}
```

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

LorsMiseàJourDonnées

```
boolean onDataUpdate (me, eventInfo)
```

Cet événement se produit lorsque les données associées au widget changent.

Paramètre	Description
me	Objet déclencheur de l'événement
eventInfo	Un objet avec les champs indiqués ci-dessous (vous pouvez vous référer aux champs en utilisant "." notation point) **oldValue** = Valeur du widget avant le changement **newValue** = Valeur qui sera actualisée au widget **attrName** = Attribut sur lequel l'événement est généré **index** = Index d'attribut entier, le cas échéant, par défaut = 0 **mode =** W lorsque l'utilisateur est en train d'écrire au widget. R dans tous les autres statuts.

L'événement est déclenché avant que la valeur ne soit transmise au widget. Un code JavaScript peut intercepter l'événement et décider d'éviter de mettre à jour le widget en retournant la valeur réelle.

Remarque : s'il y a d'autres macros associées à l'événement, toutes les macros seront exécutées quelle que soit la valeur de retour utilisée dans le code JavaScript.

```
function buttonStd1_onDataUpdate(me, eventInfo) {
if ( eventInfo.newValue > 100) {
   //do something...
   return true;  // To avoid to continue and update
                 // the widget (e.g. not update the linked tag)
}
   return false; // To continue and update the widget
                 // (e.g. update the linked tag)
}
```

Événements de page

onActivate (Lorsd'Activation)

```
void onActivate( me, eventInfo )
```

Cet événement survient toutes les fois que la page s'affiche.

Paramètre	Description
me	Objet déclencheur de l'événement
eventInfo	Réservés à une utilisation ultérieure

JavaScript sera exécuté lorsque la page sera active, autrement dit lorsque la page sera chargée.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

```
function Page1_onActivate(me, eventInfo) {
     //faire quelque chose…
}
```

onDeactivate

```
void onDeactivate( me, eventInfo )
```

Cet événement survient lorsque vous quittez la page.

Paramètre	Description
me	Objet déclencheur de l'événement
eventInfo	Réservés à une utilisation ultérieure

```
function Page1_onActivate(me, eventInfo) {
     //faire quelque chose…
}
```

onWheel (LorsdeMolette)

```
void onDeactivate( me, eventInfo )
```

Cet événement survient lorsqu'un dispositif à molette bouge (par exemple, une molette de souris).

Paramètre	Description
me	Objet déclencheur de l'événement
eventInfo	Détails sur l'événement déclenché

```
fonction Page1_onMouseWheelClock(me, eventInfo) {
     //faire quelque chose…
}
```

Événements système

Les événements système peuvent être liés à :

- planificateur
- alarmes
- un périphérique à molette

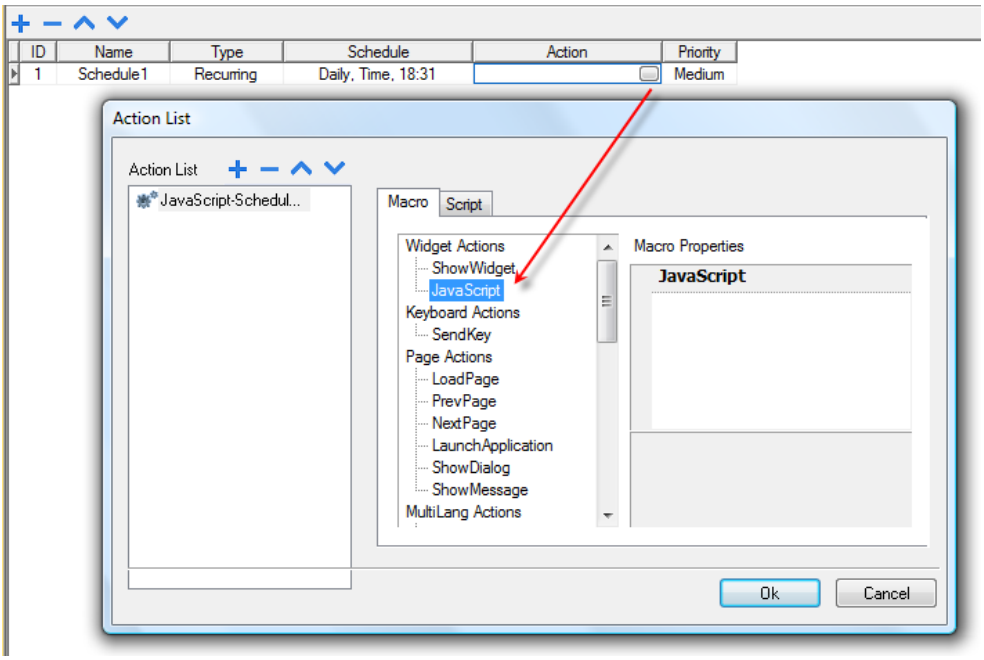
Important : assurez-vous de ne pas dupliquer les noms de fonctions JavaScript au niveau de la page et du projet. Quand un conflit se produit, c'est-à-dire deux fonctions avec le même nom dans la page en cours et au niveau du projet, le système exécute la fonction de rappel JavaScript au niveau de la page.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Quand un rappel JavaScript ne se trouve pas dans la page courante, le système le recherche automatiquement au niveau du projet.

Événements du planificateur

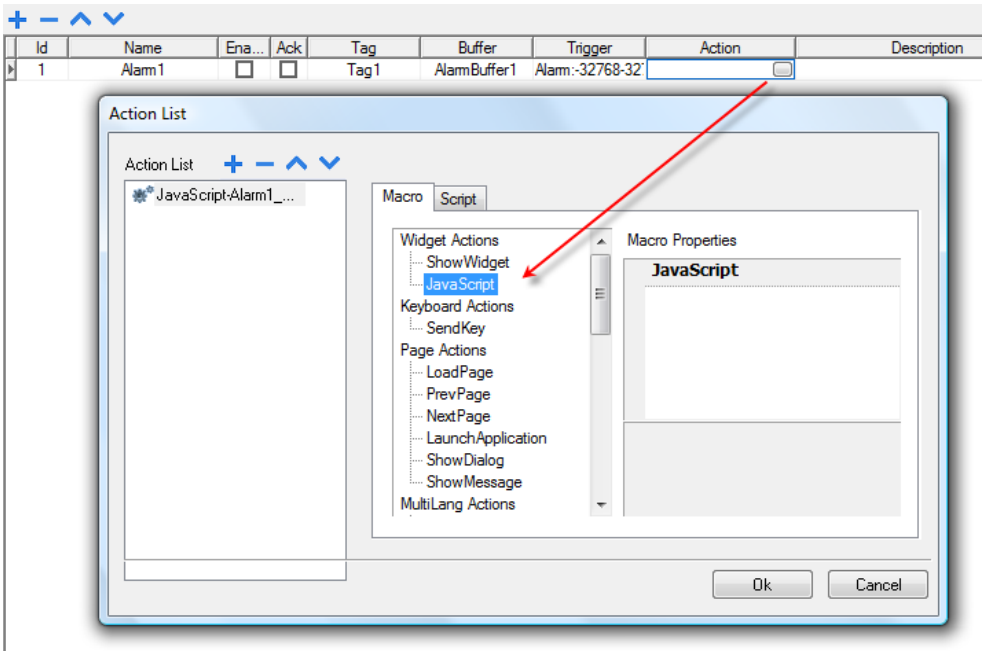
Ces événements se produisent lorsqu'ils sont déclenchés par l'action associée dans le planificateur.



Vous pouvez modifier le JavaScript depuis l'onglet **Propriétés du projet**.

Événements d'alarmes

Ces événements se produisent lorsqu'ils sont déclenchés par la condition d'alarme associée.



Vous pouvez modifier le JavaScript depuis l'onglet **Propriétés du projet**.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

onWheel

```
void onDeactivate( me, eventInfo )
```

Cet événement survient lorsqu'un dispositif à molette bouge (par exemple, une molette de souris).

Paramètre	Description
me	Objet déclencheur de l'événement
eventInfo	Détails sur l'événement déclenché

```
fonction Project1_onMouseWheelClock(me, eventInfo) {
     //faire quelque chose…
}
```

Objets

PB610 Panel Builder 600 utilise des objets JavaScript pour accéder aux éléments de la page. Chaque objet se compose de propriétés et de méthodes utilisées pour définir l'opération et l'aspect de l'élément de page. Les objets suivants sont utilisés pour interagir avec les éléments de la page du périphérique IHM :

Objet	Description
Widget	C'est la classe de base pour l'ensemble des éléments sur la page, élément de page inclus
Page	Cet objet référence la page actuelle du périphérique IHM. La page est l'objet supérieur de l'écran.
Groupe	Cet objet associe un ensemble de variables pour permettre un fonctionnement uniforme sur un ensemble de variables logiquement connectées
Projet	Cet objet définit le widget de projet. Le widget de projet est utilisé pour récupérer les données du projet, telles que les variables, les alarmes, les recettes, les ordonnancements, etc. Il n'y a qu'un seul widget pour le projet, qui peut être référencé par les variables du projet.
État	Cet objet est la classe maintenant l'état d'une variable obtenue à partir de l'environnement contrôlé. En plus de la valeur elle-même, il contient l'horodatage indiquant à quel moment la valeur a été collectée ainsi que les drapeaux signalant la qualité de la valeur.

Widget objets de classe

Le Widget classe est la base pour l'ensemble des éléments sur une page, élément de page inclus.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Le widget, dans ce cas, ne sert pas à indiquer un objet d'écran spécifique, mais une classe JavaScript.

Changer les propriétés du widget avec JavaScript

Si vous souhaitez modifier les propriétés des widgets avec JavaScript, définir la propriété du widget de l'**optimisation statique** à **dynamique**.

Important : la propriété widget de l'optimisation statique n'est pas réglée sur dynamique, les modifications apportées aux propriétés seront ignorées.

Si un appel à `getWidget` échoue, le débogueur distant signale l'erreur suivante :

« Tentative d'accès au widget statique optimisé 'étiquette1' ». Désactivez l'optimisation statique du widget pour accéder auwidget depuis le script ».

Cette erreur est également visible en utilisant le fragment de code suivant :

```
var wgt;
try {
wgt = page.getWidget('label1');
} catch(err) {
alert("" + err);
}
```

Propriétés du widget

Certaines propriétés sont communes à tous les widgets.

Nom d'objet

`string objectName`

Obtenez le nom du widget, un identifiant unique.

```
function btnStd04_onMouseRelease(me) {
     var wgt = page.getWidget("rect1");
     var name = wgt.objectName;
}
```

(Disponible sur pages Web)

x

`nombre x`

Obtenez ou définissez la position « x » du widget dans les pixels.

```
function btnStd1_onMouseRelease(me) {
     var wgt = page.getWidget("rect1");
```


Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

```
    wgt.x = 10;
}
```

(Disponible sur pages Web)

y

```
nombre y
```

Obtenez ou définissez la position « y » du widget dans les pixels.

```
function btnStd1_onMouseRelease(me) {
    var wgt = page.getWidget("rect1");
    wgt.y = 10;
}
```

(Disponible sur pages Web)

width

```
Nombre largeur
```

Donne ou définit la position « y » du widget en pixels.

```
function btnStd1_onMouseRelease(me) {
    var wgt = page.getWidget("rect1");
    wgt.width = 10;
}
```

(Disponible sur pages Web)

height

```
Nombre hauteur
```

Donne ou définit la hauteur du widget en pixels.

```
function btnStd1_onMouseRelease(me) {
    var wgt = page.getWidget("rect1");
    wgt.height = 10;
}
```

(Disponible sur pages Web)

visible

```
boolean visible
```

Obtenez ou définissez l'état visible du widget.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

```
function btnStd4_onMouseRelease(me) {
     var wgt = page.getWidget("rect1");
     wgt.visible = false;
}

function btnStd5_onMouseRelease(me) {
     var wgt = page.getWidget("rect1");
     wgt.visible = true;
}
```

Valeur

`nombre valeur`

Obtenez ou définissez la valeur du widget.

```
function btnStd6_onMouseRelease(me) {
     var wgt = page.getWidget("field1");
     wgt.value = 100;
}
```

opacité

`nombre opacité (compris entre 0 et 1)`

Récupère ou définit l'opacité du widget. Les valeurs sont des décimales de 0 à 1, où 1 signifie 100 % opaque.

```
function btnStd8_onMouseRelease(me) {
     var wgt = page.getWidget("rect1");
     wgt.opacity = 0.5;
}
```

(Disponible sur pages Web)

rotation

`nombre rotation (en degrés)`

Récupère ou définit l'angle de rotation du widget. La rotation se fait dans le sens horaire et par degrés, en commençant à la position Est.

```
function btnStd9_onMouseRelease(me) {
     var wgt = page.getWidget("rect1");
     wgt.rotation = 45;
}
```

(Disponible sur pages Web)

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

userValue

```
string userValue
```

Récupère ou définit une valeur définie par l'utilisateur pour le widget. Ce champ peut être utilisé par les fonctions JavaScript pour stocker des données supplémentaires avec le widget.

```
function btnStd9_onMouseRelease(me) {
     var wgt = page.getWidget("rect1");
     wgt.userValue = "Here I can store custom data";
}
```

Chaque widget a des propriétés spécifiques auxquelles vous pouvez accéder en utilisant la notation avec point. Pour obtenir une liste actualisée et détaillée des propriétés, vous pouvez utiliser le Débogueur JavaScript qui inspecte les méthodes et les propriétés de widget.

Méthodes du widget

Certaines méthodes sont communes à tous les widgets.

getProperty

```
object getProperty( propertyName, [index] )
```

Revient à une propriété.

Paramètre	Description
propertyName	Chaîne contenant le nom de la propriété à obtenir
index	Index de l'élément à obtenir à partir du tableau (par défaut = 0)

Presque toutes les propriétés qui sont affichées dans le PB610 Panel Builder 600 volet**Propriétés** peuvent être récupérées en utilisant la méthode `getProperty`. La valeur de l'indice est facultative et est uniquement utilisée pour les widgets qui prennent en charge des tableaux.

```
function buttonStd1_onMouseRelease(me, eventInfo) {
     var shape = page.getWidget("rect2");
     var y_position = shape.getProperty("y");
}
```

```
function buttonStd2_onMouseRelease(me, eventInfo) {
     var image = page.getWidget("multistate1");
     var image3 = image.getProperty("imageList", 2);
     //…
}
```

(Disponible sur pages Web)

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

setProperty

`boolean setProperty( propertyName, value, [index] )`

Définissez une propriété pour le widget.

Paramètres

Paramètre	Description
propertyName	Chaîne contenant le nom de la propriété à définir
Valeur	Chaîne contenant la valeur à définir la propriété.
index	Index de l'élément à définir à partir du tableau (par défaut = 0)

Presque toutes les propriétés affichées dans le PB610 Panel Builder 600 volet**Propriétés** peuvent être récupérées en utilisant cette méthode. La valeur de l'index est facultative et n'est utilisée que pour les Widgets qui supportent les tableaux (par exemple, un widget MultiState Image). La méthode `setProperty` retourne une valeur booléenne (vrai ou faux) pour indiquer si la propriété a été définie ou non.

```
function buttonStd1_onMouseRelease(me, eventInfo) {
     var setting_result = shape.setProperty("y", 128);
     if (setting_result)
     alert("Shape returned to start position");
}
```

```
function buttonStd2_onMouseRelease(me, eventInfo) {
     var image = page.getWidget("multistate1");
     var result = image.setProperty("imageList", "Fract004.png", 2);
     //…
}
```

(Disponible sur pages Web)

Objet de page

Cet objet référence la page actuelle du périphérique IHM. La page est l'objet supérieur de l'écran.

Propriétés de l'objet de page

Propriétés disponibles au niveau de la page.

backgroundColor

`chaîne backgroundColor (Couleur d'arrière-plan) (au format rgb(xxx, xxx, xxx) où xxx varie de 0 à 255)`

Couleur de fond de page.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

```
function btnStd11_onMouseRelease(me) {
     page.backgroundColor = "rgb(128,0,0)";
}
```

(Disponible sur pages Web)

width

```
Nombre largeur
```

Largeur de page en pixels.

```
function btnStd05_onMouseRelease(me) {
     var middle_x = page.width / 2;
}
```

(Disponible sur pages Web, avec récupération uniquement)

height

```
Nombre hauteur
```

Hauteur de la page en pixels.

```
function btnStd05_onMouseRelease(me) {
     var middle_y = page.height / 2;
}
```

(Disponible sur pages Web, avec récupération uniquement)

userValue

```
string userValue
```

Obtient ou établit une valeur définie par l'utilisateur pour le widget. Ce champ peut être utilisé par les fonctions JavaScript pour stocker des données supplémentaires avec la page.

```
function btnStd9_onMouseRelease(me) {
     page.userValue = "Here I can store custom data";
}
```

(Disponible sur pages Web)

Méthodes associées à l'objet de page

Méthodes pouvant être utilisées au niveau de la page.

getWidget

```
object getWidget( wgtName )
```

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Retourne le widget portant le nom indiqué.

Paramètre	Description
wgtName	Chaîne contenant le nom du widget

Return value

Un objet représentant le widget. Si le widget n'existe pas, le retour sera nul.

```
function btnStd1_onMouseRelease(me) {
     var my_button = page.getWidget("btnStd1");
}
```

(Disponible sur pages Web)

setTimeout

```
number setTimeout( functionName, delay )
```

Démarre un minuteur qui appelle une fonction donnée après un délai donné.

Paramètre	Description
functionName	Chaîne contenant le nom de la fonction à appeler
Retard	Délai en millièmes de secondes

Return value

Nombre correspondant au timerID.

```
var duration = 3000;
var myTimer = page.setTimeout("innerChangeWidth()", duration);
```

(Disponible sur pages Web)

clearTimeout

```
void clearTimeout( timerID )
```

Interrompt et efface le minuteur de délai d'attente avec le minuteur indiqué.

Paramètre	Description
timerID	Minuteur à effacer et interrompre

```
var duration = 3000;
var myTimer = page.setTimeout("innerChangeWidth()", duration);
// do something
page.clearTimeout(myTimer);
```


Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

(Disponible sur pages Web)

setInterval

```
number setInterval( functionName, interval )
```

Démarre un minuteur qui exécute la fonction indiquée selon l'intervalle indiqué.

Paramètre	Description
functionName	Chaîne contenant le nom de la fonction à appeler
interval	Intervalle en millièmes de secondes

Return value

Nombre correspondant au timerID.

```
var interval = 3000;
var myTimer = page.setInterval("innerChangeWidth()", interval);
```

(Disponible sur pages Web)

clearInterval

```
void clearInterval( timerID )
```

Interrompt et efface le minuteur à intervalle avec le minuteur indiqué.

Paramètre	Description
timerID	Minuteur à effacer et interrompre

```
var interval = 3000;
var myTimer = page.setInterval("innerChangeWidth()", interval);
// do something
page.clearInterval(myTimer);
```

(Disponible sur pages Web)

clearAllTimeouts

```
void clearAllTimeouts()
```

Efface tous les minuteurs démarrés.

```
page.clearAllTimeouts();
```

(Disponible sur pages Web)

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Objet de groupe

Un groupe est un élément logique de base qui associe un ensemble de variables logiques.

Méthodes d'objet de groupe

Méthodes pouvant être utilisées avec les objets de groupe.

getTag

```
object getTag( TagName )
```

Obtient la variable spécifiée par TagName à partir de l'objet de groupe.

Paramètre	Description
TagName	Chaîne représentant le nom de variable

Return value

Un objet représentant la valeur de la variable ou, si la valeur de variable est un tableau, l'intégralité du tableau. Si vous avez besoin de récupérer un élément du tableau, sélectionnez la méthode `getTag` disponible dans l'objet du projet. Indéfini est retourné si la variable n'est pas valide.

```
var group = new Group();
project.getGroup("GroupName", group);
var value = group.getTag("Tag1");
```

(Disponible sur pages Web)

getCount

```
number getCount()
```

Retourne le nombre total de variables dans ce groupe.

```
var group = new Group();
project.getGroup("GroupName", group);
var value = group.getCount();
```

(Disponible sur pages Web)

getTags

```
object getTags()
```

Retourne la liste de toutes les variables du groupe.

```
Fonction
var group = new Group();
```


Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

```
project.getGroup("enginesettings", group);
var tagList = group.getTags();
for(var i = 0; i < tagList.length; i++){
     var tagName = tagList[i];
     //faire quelque chose…
};
```

(Disponible sur pages Web)

Objet du projet

Cet objet définit le widget de projet. Le widget de projet est utilisé pour récupérer les données du projet, telles que les variables, les alarmes, les recettes, les ordonnancements, etc. Il n'y a qu'un seul widget pour le projet, qui peut être référencé par les variables du projet.

Propriétés d'objet du projet

Propriétés à définir au niveau du projet.

startPage

`chaîne startPage`

Page affichée au démarrage du projet.

```
var startPage = project.startPage;
project.startPage = "Page2.jmx";
```

Méthodes d'objet du projet

Méthodes à utiliser au niveau du projet.

nextPage

`void nextPage()`

Le script exécute l'action de la Page suivante.

```
project.nextPage();
```

(Disponible sur pages Web)

prevPage

`void prevPage()`

Le script exécute l'action de la page précédente.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

```
project.prevPage();
```

(Disponible sur pages Web)

lastVisitedPage

```
void lastVisitedPage()
```

Le script exécute l'action de la dernière page visitée.

```
project.lastVisitedPage();
```

(Disponible sur pages Web)

Page d'accueil

```
void homePage()
```

Le script exécute l'action de la Page d'accueil.

```
project.homePage();
```

(Disponible sur pages Web)

loadPage

```
void loadPage(pageName)
```

Le script exécute le chargement de la page demandée définie dans le script.

```
project.loadPage("Page5.jmx");
```

(Disponible sur pages Web)

AVERTISSEMENT : à chaque changement de page, la suppression de l'ensemble des événements de temps actifs est forcée et la procédure JavaScript continuera à être exécutée jusqu'à la fin avant de passer à la nouvelle page.

showDialog

```
void showDialog(pageName)
```

Le script exécute l'affichage de la page de dialogue.

```
project.showDialog("Dialog.jmx");
```

(Disponible sur pages Web)

closeDialog

```
void closeDialog()
```

Le script exécute la fermeture de la page de dialogue ouverte.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

```
project.closeDialog();
```

(Disponible sur pages Web)

showMessage

```
void showMessage( message )
```

Le script exécute l'affichage de la fenêtre contextuelle du message.

```
project.showMessage("Hi This is test message");
```

(Disponible sur pages Web)

getGroup

```
number getGroup( groupName, groupInstance, [callback] )
```

Méthode de lecture rapide ; permet d'obtenir les valeurs de l'ensemble des variables d'un groupe.

Paramètre	Description
Nom de groupe	Chaîne contenant le nom du groupe
InstanceGroupe	Élément de groupe à remplir
callback	Chaîne contenant le nom de la fonction à rappeler quand le groupe est prêt

Return value

Un chiffre représentant l'état : 1 pour réussite, 0 pour échec.

```
var group = new Group();
var status = project.getGroup ("enginesettings", group);
if (status == 1) {
     var value = group.getTag("Tag1");
     if (value!=undefined) {
     // do something with the value
     }
}
```

```
var g = new Group();
var status = project.getGroup ("enginesettings", g,
     function (groupName, group) { fnGroupReady(groupName, group);} );

function fnGroupReady(groupName, group) {
     var val = group.getTag("Tag1");
     if (val!=undefined) {
     // do something with the value
```

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

```
    }
}
```

(Disponible sur pages Web)

getTag

```
object getTag( tagName, state, index, forceRefresh)
```

```
void getTag( tagName, state, index, callback, forceRefresh)
```

Retourne la valeur de variable ou le tableau complet si la valeur d'index est -1 du NomVariable indiqué.

Paramètre	Description
tagName	Chaîne du nom de variable
État	Élément d'état à remplir
index	Index en cas de variable sous forme de tableau. -1 retourne le tableau complet. Défaut = 0.
callback	Nom de fonction si une lecture asynchrone est requise. Défaut = “”.
forceRefresh	(Paramètre optionnel) Vrai = le runtime lira une valeur mise à jour de la variable directement à partir du périphérique. Défaut = faux.

Return value

La valeur de la variable est retournée. Si la variable est un tableau et l'index = -1, le tableau complet est retourné. Pour les variables autres que les tableaux, définissez un index à 0.

```
var state = new State();
var value = project.getTag("Tag1", state, 0);
//
//for non array type
//tags index is not considered, so can be left as 0
//
if (value!=undefined) {
//...do something with s
}
```

```
var state = new State();
project.getTag("Tag1", state, -1,
     function(tagName, tagState) { fnTagReady(tagName, tagState); });
function fnTagReady(tagName, tagState) {
     if (tagName=="Tag1") {
     var myValue = tagState.getValue();
     }
```


Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

```
}
```

(Disponible sur pages Web)

setTag

```
number setTag( tagName, tagValue, [index], [forceWrite] )
```

Définit la variable donnée dans le projet. Le nom et la valeur sont en chaînes.

Paramètre	Description
tagName	Chaîne du nom de variable
ValeurVariable	Objet contenant la valeur à écrire
index	Index en cas de variable de type tableau. -1 retourne le tableau complet. Défaut = 0.
forcerÉcrire	Valeur booléenne pour activer l'écriture forcée des variables, la fonction attendra que la valeur soit écrite avant de revenir. Défaut = faux.

Return value

Valeur entière indiquant le succès ou l'échec d'une action lorsque forceWrite est vrai. 0 signifie réussite et -1 signifie échec. Si forceWrite est faux, la valeur de retour sera indéfinie.

```
var val = [1,2,3,4,5];
var status = project.setTag("Tag1", val, -1, true);
if (status == 0) {
     // Success
} else {
     // Failure
}
```

```
var val = "value";
project.setTag("Tag1", val);
```

(Disponible sur pages Web)

updateSystemVariables

```
void project.updateSystemVariables()
```

Force le rafraîchissement des variables du système.

```
project.updateSystemVariables()
```

selectAllAlarms

void project.selectAllAlarms(bool selected)

Sélectionne/désélectionne toutes les alarmes

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

```
project.selectAllAlarms(true)
```

(Disponible sur pages Web)

ackAlarms

void project.ackAlarms()

Confirme toutes les alarmes sélectionnées

```
project.selectAllAlarms(true);
project.ackAlarms();
project.selectAllAlarms(true);
```

(Disponible sur pages Web)

resetAlarms

void project.resetAlarms()

Réinitialise toutes les alarmes sélectionnées

```
project.selectAllAlarms(true);
project.resetAlarms();
project.selectAllAlarms(true);
```

(Disponible sur pages Web)

Activer alarmes

void project.enableAlarms()

Active toutes les alarmes sélectionnées

```
project.selectAllAlarms(true);
project.enableAlarms();
project.selectAllAlarms(true);
```

(Disponible sur pages Web)

getRecipeItem

```
object getRecipeItem (recipeName, recipeSet, recipeElement)
```

Obtient la valeur de l'élément d'ensemble de recette indiqué.

Paramètre	Description
Nom de recette	Chaîne représentant le nom de recette
recipeSet	Chaîne représentant l'ensemble de recette, peut être soit le nom de l'ensemble de recette, soit l'index

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Paramètre	Description
	d'ensemble basé sur 0.
recipeElement	Chaîne représentant l'Élément de recette, peut être soit le nom d'élément, soit l'index d'élément basé sur 0.

Return value

Un objet ayant la valeur de la recette. Non défini est retourné si non valide. Si de type tableau, un type d'objet sous forme de tableau est retourné.

```
var value = project.getRecipeItem("recipeName", "Set", "Element");
```

setRecipeItem

```
number setRecipeItem (recipeName, recipeSet, recipeElement, value )
```

Obtient la valeur de l'élément d'ensemble de recette indiqué.

Paramètre	Description
recipeName	Chaîne représentant le nom de recette
recipeSet	Chaîne représentant l'ensemble de recette, peut être soit le nom de l'ensemble de recette, soit l'index d'ensemble basé sur 0.
recipeElement	Chaîne représentant l'Élément de recette, peut être soit le nom d'élément, soit l'index d'élément basé sur 0.
Valeur	Un objet contenant la valeur à stocker dans la recette. Ce peut être un type de tableau.

Return value

Valeur entière indiquant le succès ou l'échec d'une action lorsque forceWrite est vrai. ‘0’ signifie réussite et ‘-1’ signifie échec.

```
var val = [2,3,4];
project.setRecipeItem("recipeName", "Set", "Element", val);
if (status == 0) {
     // Success
} else {
     // Failure
}
```

downloadRecipe

```
void downloadRecipe (recipeName, recipeSet )
```

Télécharge l'ensemble de recette vers la variable correspondante.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Paramètre	Description
recipeName	Chaîne représentant le nom de recette
recipeSet	Chaîne représentant l'ensemble de recette, peut être soit le nom de l'ensemble de recette, soit l'index d'ensemble basé sur 0.

```
project.downloadRecipe("recipeName", "Set");
```

uploadRecipe

```
void uploadRecipe (recipeName, recipeSet )
```

Charge la valeur des variables dans l'ensemble de recette fourni.

Paramètre	Description
recipeName	Chaîne représentant le nom de recette
recipeSet	Chaîne représentant l'ensemble de recette, peut être soit le nom de l'ensemble de recette, soit l'index d'ensemble basé sur 0.

```
project.uploadRecipe("recipeName", "Set");
```

launchApp

```
void launchApp( appName, appPath, arguments, singleInstance)
```

Exécute une application externe.

Paramètre	Description
appName	Chaîne contenant le nom de l'application
appPath	Chaîne contenant le chemin absolu de l'application
Arguments	Chaîne contenant les arguments à envoyer à l'application
singleInstance	vrai = une seule instance autorisée, faux = multiples instances autorisées

```
project.launchApp("PDF.exe","\\Flash\\QTHMI\\PDF","\\USBMemory\\file.pdf","true");
```

getClientType

```
chaîne getCurrentPageName()
```

Retourne le type de client

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Type de client	Description
local	En cours d'exécution sur périphérique IHM
à distance	En cours d'exécution sur client HMI Client
Web	En cours d'exécution sur client Web

```
var clientType = project.getClientType();
if (clientType=="web") {
       // Currently running on web client
} else  if (clientType=="remote") {
       // Currently running on HMI Client
} else  if (clientType=="local")  {
       // Currently running on HMI Device
}
```

(Disponible sur pages Web)

Ouverture de session

```
int project.login("username", "password")
```

Accès au système avec les identifiants donnés

```
var ReplyCode;
ReplyCode = project.login("admin", "admin");
if (ReplyCode != 0) {
    alert("Access denied);
}
```

Return value	
0	Pas d'erreur
1	Erreur : vous n'êtes pas autorisé.
2	Erreur : connexion perdue avec le runtime.
3	Erreur : identifiant ou mot de passe saisi incorrect
4	Erreur : le mot de passe saisi est incorrect
5	Erreur : impossible d'exécuter l'action
6	Erreur : les mots de passent sont différents
7	Erreur : le mot de passe est trop court
8	Erreur : le mot de passe doit contenir des chiffres
9	Erreur : le mot de passe doit contenir des caractères spéciaux

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Return value	
10	Erreur : le mot de passe doit être différent des mots de passe antérieurs.
11	Erreur : utilisateur déjà existant
12	Erreur : le mot de passe ne peut pas être vide
13	Erreur : votre mot de passe a expiré
14	Erreur : votre mot de passe expire bientôt

Déconnexion

```
project.logout(AllowDefaultUser)
```

Quitter le système

```
project.logout();       // Logout even from default user
project.logout(true);   // Logout even from default user
project.logout(false);  // Logout only if not logged as default user
```

Widgets d'objet du projet

getCurrentPageName

```
string getCurrentPageName()
```

Revenir au nom de la page active en cours

```
// Get PageMgr widget
var pageMgr = project.getWidget( "_PageMgr" );

// Show Current Page
var currentPageName = pageMgr.getCurrentPageName();
project.showMessage( "Current active page is: " + currentPageName );
```

(Disponible sur pages Web)

hasPage

```
boolean hasPage(string pageName)
```

Revenir sur vrai si la page existe, sinon sur faux

```
// Get PageMgr widget
var pageMgr = project.getWidget( "_PageMgr" );

//Page exists
var pageExists = pageMgr.hasPage( "Page10" );
```


Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

```
if (pageExists) {
    project.showMessage( "Page10 exists" );
} else {
    project.showMessage( "Hei Page10 not exists!" );
}
```

(Disponible sur pages Web)

curLangCode

`chaîne curLangCode`

Propriété du widget MultiLangMgr. Contient le code de la langue active.

```
// Get MultiLangMgr widget
var MultiLangMgr = project.getWidget( "_MultiLangMgr" );

// Show curLangCode
var curLangCode = MultiLangMgr.curLangCode;
project.showMessage( "Current active language is: " + curLangCode );
```

Imprimer l'objet des rapports

printGfxReport

`void printGfxReport( reportName, silentMode)`

Imprime le rapport graphique spécifié par reportName.

Paramètre	Description
reportName	Chaîne contenant le nom de rapport
silentMode	Vrai = mode silencieux activé. Aucune boîte de dialogue des paramètres d'impression n'est affichée.

```
project.printGfxReport("Report Graphics 1", true);
```

ImprimerTexte

`void printText( text, silentMode)`

Imprime un texte fixe.

Paramètre	Description
text	Chaîne à imprimer
silentMode	Vrai = mode silencieux activé. Aucune boîte de dialogue des paramètres d'impression n'est affichée.

```
project.printText("Hello I Am Text Printing",true);
```

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

ImpressionOctets

```
void printBytes( text, silentMode)
```

Imprime une chaîne hexadécimale représentant les données à imprimer. Par exemple, « 1b30 » pour imprimer < ESC 0 >

Paramètre	Description
text	Chaîne hexadécimale à imprimer
silentMode	Vrai = mode silencieux activé. Aucune boîte de dialogue des paramètres d'impression n'est affichée.

```
project.printText("Hello I Am Text Printing",true);
```

emptyPrintQueue

```
void emptyPrintQueue()
```

Vide la queue d'impression. La tâche actuelle ne sera pas annulée.

```
project.emptyPrintQueue();
```

pausePrinting

```
void pausePrinting();
```

Suspend les opérations d'impression. Ne suspendra pas l'impression d'une page déjà envoyée à l'imprimante.

```
project.pausePrinting();
```

resumePrinting

```
void resumePrinting();
```

Reprend l'impression suspendue.

```
project.resumePrinting();
```

abortPrinting

```
void abortPrinting();
```

Annule l'opération d'impression en cours et continue avec l'élément suivant dans la queue. Cette commande ne suspendra pas l'impression d'une page déjà envoyée à l'imprimante.

```
project.abortPrinting();
```

printStatus

```
project.printStatus;
```

Retourne une chaîne représentant l'état d'impression actuel.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Chaîne de statut	Description
Erreur	Une erreur est survenue pendant l'impression
Impression	Impression en cours
idle	Le système est prêt à accepter de nouvelles tâches
paused	L'impression a été suspendue

```
var status = project.printStatus;
project.setTag("PrintStatus",status);
```

printGfxJobQueueSize

```
project.printGfxJobQueueSize;
```

Retourne le nombre de rapports graphiques dans la queue d'impression.

```
var gfxqueuesize = project.printGfxJobQueueSize;
project.setTag("printGfxJobQueueSize",gfxqueuesize);
```

printTextJobQueueSize

```
project.printTextJobQueueSize;
```

Retourne le nombre de rapports textes dans la queue d'impression.

```
var textjobqueuesize = project.printTextJobQueueSize;
project.setTag("printTextJobQueueSize",textjobqueuesize);
```

printCurrentJob

```
project.printCurrentJob;
```

Retourne une chaîne représentant la tâche actuelle en cours d'impression

```
var currentjob = project.printCurrentJob;
project.setTag("printCurrentJob",currentjob);
```

printActualRAMUsage

```
project.printActualRAMUsage;
```

Retourne une estimation sur l'utilisation de RAM pour les queues d'impression

```
var myVar = project.printActualRAMUsage;
alert(" actual ram usage is "+ myVar);
```

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

printRAMQuota

`project.printRAMQuota;`

Retourne l'utilisation maximum de RAM autorisée pour les queues d'impression

```
var ramquota = project.printRAMQuota;
project.setTag("printRAMQuota",ramquota);
```

printActualDiskUsage

`project.printActualDiskUsage;`

Retourne l'utilisation du disque de dossier spool (pour les impressions PDF)

```
var myVar1 = project.printActualDiskUsage;
alert(" actual disk usage is "+ myVar1);
```

printDiskQuota

`project.printDiskQuota;`

Retourne la taille maximale autorisée du dossier spool (pour les impressions PDF).

```
var ramquota = project.printRAMQuota;
var diskquota = project.printDiskQuota;
```

printSpoolFolder

`project.printSpoolFolder;`

Retourne le chemin du dossier spool actuel (pour les impressions PDF).

```
var spoolfolder = project.printSpoolFolder;
project.setTag("printSpoolFolder",spoolfolder);
```

printPercentage

`project.printPercentage;`

Retourne le pourcentage d'avancement de la tâche actuelle (significatif seulement pour les rapports graphiques multipages)

```
var percentage = project.printPercentage;
project.setTag("printPercentage",percentage);
```

Objet de statut

Il s'agit de la classe maintenant le statut d'une variable obtenue à partir de l'environnement contrôlé.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Méthodes pour objet de statut

Méthodes à utiliser avec les objets de statut.

getQualityBits

```
number getQualityBits()
```

Renvoie un entier - une combinaison de bits indiquant la qualité de valeur de la variable.

```
var state = new State();
var value = project.getTag("Tag1", state, 0);
var qbits = state.getQualityBits();
```

(Disponible sur pages Web)

getTimestamp

```
number getTimestamp()
```

Renvoie la valeur du temps échantillonnée.

Return value

Un nombre contenant l'horodatage (par exemple 1315570524492).

Remarque : la date est un type natif de données de JavaScript.

```
var state = new State();
var value = project.getTag("Tag1", state, 0);
var ts = state.getTimestamp();
```

isQualityGood

```
boolean isQualityGood()
```

Renvoie si la valeur contenue dans cet objet de statut est fiable.

Return value

Une valeur booléenne de vrai si la qualité est bonne, sinon faux.

```
var state = new State();
var value = project.getTag("Tag1", state, 0);
if (state.isQualityGood()) {
     // faire quelque chose…
}
```

(Disponible sur pages Web)

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Mots-clés

Des objets globaux sont prédéfinis et peuvent être référencés par les noms suivants.

page

```
object page
```

Référence l'objet de page pour la page courante.

```
function btnStd04_onMouseRelease(me) {
     var wgt = page.getWidget("rect1");
     var name = wgt.objectName;
}
```

projet

```
object project
```

Référence le widget de projet.

```
var group = new Group();
project.getGroup("GroupName", group);
var value = group.getCount("Tag1");
```

Fonctions générales

print

```
void print( message )
```

Imprime un message à la fenêtre d'enregistrement IHM.

Paramètre	Description
message	Chaîne de message

```
print("Test message");
```

alert

```
void alert( message )
```

Affiche une boîte de dialogue contextuelle avec un message prédéfini. L'utilisateur doit appuyer sur le bouton **OK** dans la boîte de dialogue pour poursuivre l'exécution du script.

Paramètre	Description
message	Chaîne de message

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Remarque : la fonction alerte peut être utilisée pour le débogage des fonctions JavaScript.

```
alert("Test message");
```

(Disponible sur pages Web)

Prendre en main la lecture/l'écriture de fichiers

Créer un dossier

```
boolean fs.mkdir(strPath);
```

Crée un dossier, s'il n'existe pas déjà, dans le chemin spécifié. Retourne vrai en cas de succès et faux en cas d'échec.

Paramètre	Description
strPath	Chaîne de chemin

Remove folder

```
boolean fs.rmdir(dirPath);
```

Supprime le répertoire de strPath s'il existe et est vide. Retourne vrai en cas de succès et faux en cas d'échec.

Paramètre	Description
dirPath	Chaîne de dossier

Read folder content

```
object fs.readdir(dirPath);
```

Lit les contenus d'un dossier. Retourne un tableau des noms des fichiers dans le dossier en excluant '.' and '..'. Retourne une liste vide en cas d'échec.

Paramètre	Description
dirPath	Chaîne de dossier

Read file

```
object fs.readFile(strfile [,strFlag]);
```

Ouvre le fichier strFile en mode lecture, lit son contenu et le retourne.

Paramètre	Description
strFile	Chaîne de nom de fichier
strFlag	Mode lecture de fichier : "b" lit et retourne en tant que fichier binaire (autrement retourne un fichier de texte)

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Write file

```
fs.writeFile(strFile, fileData, [strFlag]);
```

Crée le fichier strFile s'il n'est pas présent. Ouvre le fichier strFile en mode écriture et écrit les données fileData au fichier.

Paramètre	Description
strFile	Chaîne de nom de fichier
fileData	Données destinées à être écrites sur le fichier dans un tableau d'octets
strFlag	Mode écriture de fichier : • "a" : ajoute le fileData à la fin du fichier de texte • "r" : remplace les contenus du fichier par le fileData • "ab" : ajoute le fileData à la fin du fichier binaire • "rb" : remplace les contenus du fichier binaire par le fileData

Le drapeau par défaut sert à écrire des fichiers de texte en mode ajout et écriture. Le chemin de fichier sera créé s'il n'est pas présent.

Retourne -1 en cas d'erreurs d'écriture.

Ajouter un fichier

```
int fs.appendFile(strFile, fileData);
```

Si les fichiers n'existe pas, les créer, sinon ajouter au fichier existant. Renvoie le nombre de caractères écrits ou -1 en cas d'erreur.

Paramètre	Description
strFile	Chaîne de nom de fichier
fileData	Données destinées à être écrites sur le fichier dans un tableau d'octets

File exists

```
boolean fs.exists(strPath)
```

Retourne vrai si le fichier ou le dossier existent au strPath.

Paramètre	Description
strPath	Chaîne de chemin

Remove file

```
boolean fs.unlink(strPath)
```

Supprime le fichier donné de strPath du système de fichiers s'il existe. Retourne vrai en cas de succès et faux en cas d'échec.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Paramètre	Description
strPath	Chaîne de chemin

File status

```
object fs.stat(strPath)
```

Récupère les informations sur le fichier/dossier présent au chemin spécifié.

Paramètre	Description
strPath	Chaîne de chemin de fichier/dossier

```
var fileStats = var fs.stat(strPath)
```

fileStats.isFile	Vrai si le chemin est un fichier
fileStats.isDir	Vrai si le chemin est un dossier
fileStats.size	Taille en octets de ce fichier
fileStats.atime	Objet de date représentant le moment du dernier accès en mode lecture
fileStats.mtime	Objet de date représentant le moment du dernier accès en mode écriture
fileStats.ctime	L'objet de date représentant le moment de création
fileStats.perm	Permissions de fichier

Si le chemin n'est pas valide, les champs isFile et isDir retournent tous les deux faux.

File permission table

0x4000	Le fichier est lisible par le propriétaire du fichier
0x2000	Le fichier est modifiable par le propriétaire du fichier
0x1000	Le fichier est exécutable par le propriétaire du fichier
0x0400	Le fichier est lisible par l'utilisateur
0x0200	Le fichier est modifiable par l'utilisateur
0x0100	Le fichier est exécutable par l'utilisateur
0x0040	Le fichier est lisible par le groupe
0x0020	Le fichier est modifiable par le groupe
0x0010	Le fichier est exécutable par le groupe
0x0004	Le fichier est lisible par tous
0x0002	Le fichier est modifiable par tous

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Notes importantes sur la gestion des fichiers

Le chemin des fichiers et des dossiers doit être de style UNIX. Ce qui implique que le caractère barre oblique inversée (\) n'est pas reconnu. Veuillez utiliser la barre oblique classique (/) à la place.

L'objet système de fichier est un objet côté client. Les opérations sont donc effectuées sur le système de fichier local plutôt que sur le système de fichier du serveur.

L'actuelle API JavaScript d'accès au système de fichier du périphérique a été conçue pour manipuler des fichiers de petite taille. Lorsqu'un fichier est lu, l'ensemble des contenus du fichier est temporairement stocké dans la RAM disponible pour l'environnement JavaScript (16MB), avec des exceptions possibles au cas où la mémoire disponible ne serait pas suffisante. Une bonne pratique en matière de programmation serait d'inclure l'appel `fs.readFile()` dans un bloc d'essai/détection.

Se connecter à partir de JavaScript

En utilisant les fonctions project.login() et project.logout(), il est possible d'automatiser la connexion de l'utilisateur à partir d'un périphérique distant. Ceci pourrait être utile, par exemple, pour effectuer la connexion par la lecture d'un badge d'utilisateur avec un lecteur de badge.

Ce chapitre montre un exemple de configuration de l'application pour gérer la connexion par un périphérique distant.

L'application doit avoir un utilisateur par défaut

Comme les fonctions du projet ne fonctionnent que lorsque l'application est active, l'application doit démarrer avec un utilisateur par défaut, peut-être avec un privilège de lecture seule. En lisant le badge, l'application peut être commutée vers un utilisateur disposant de privilèges supplémentaires. Plus tard, la commande de déconnexion réactivera l'utilisateur par défaut sans privilèges particuliers.

Dans l'exemple ci-dessous, nous utilisons trois variables pour communiquer avec le périphérique distant :

- TAG_USERNAME
- TAG_PASSWORD
- TAG_LOGIN

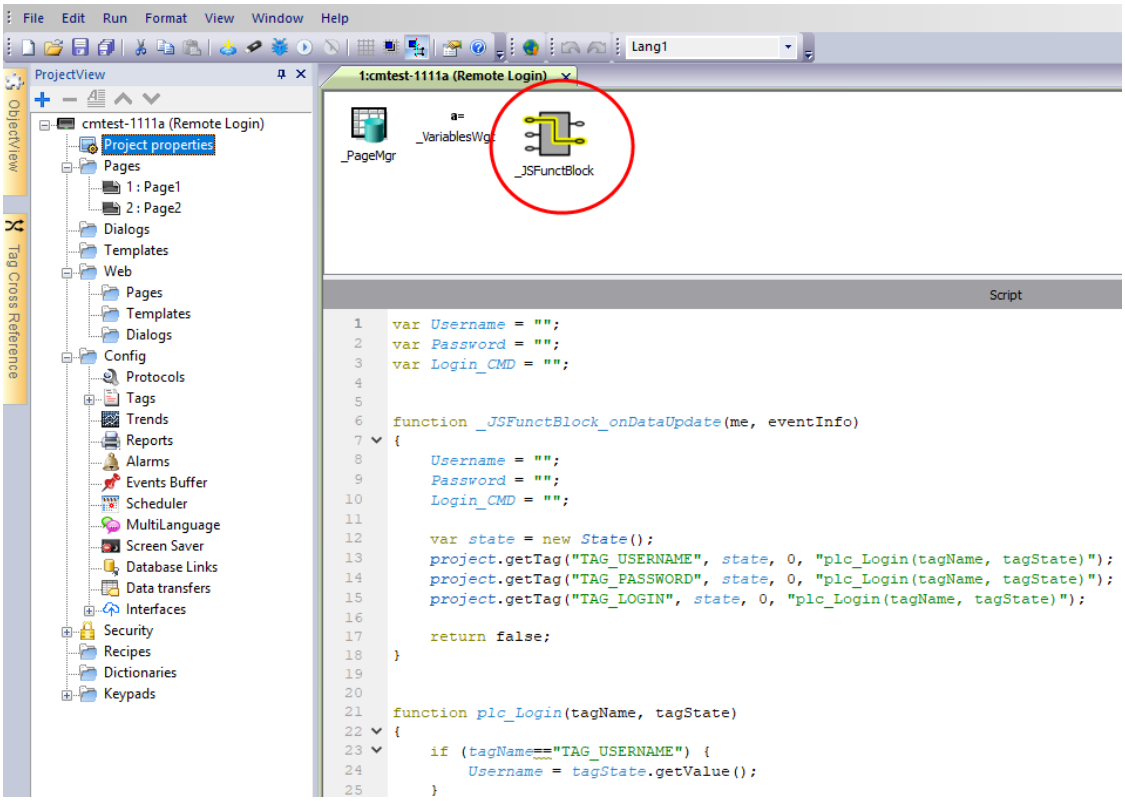
Le TAG_LOGIN sera le code de commande à exécuter.

Le périphérique distant doit remplir les paramètres TAG_USERNAME et TAG_PASSWORD, puis remplir le paramètre TAG_LOGIN avec la commande de connexion ou de déconnexion requise. Le moteur sur HMI-RUNTIME détectera les changements de TAG_LOGIN et exécutera la commande requise, puis réinitialisera le TAG_LOGIN à son statut inactif.

Commandes TAG_LOGIN	
0	Inactif
1	Demande de connexion
2	Demande de déconnexion

Au niveau du projet, nous devons ajouter un bloc fonction JavaScript pour détecter quand TAG_LOGIN changera. Le code JavaScript attaché à l'action OnDataUpdate du bloc fonction JavaScript exécutera la commande connexion.déconnexion requise.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



Le code JavaScript associé à l'action OnDataUpdate.

```
var Username;
var Password;
var Login_CMD;

function _JSFunctBlock_onDataUpdate(me, eventInfo)
{
    Username = "";
    Password = "";
    Login_CMD = "";

    var state = new State();
    project.getTag("TAG_USERNAME", state, 0, "plc_Login(tagName, tagState)");
    project.getTag("TAG_PASSWORD", state, 0, "plc_Login(tagName, tagState)");
    project.getTag("TAG_LOGIN", state, 0, "plc_Login(tagName, tagState)");
    return false;
}

function plc_Login(tagName, tagState)
{
    if (tagName=="TAG_USERNAME") {
        Username = tagState.getValue();
    }
    if (tagName=="TAG_PASSWORD") {
        Password = tagState.getValue();
    }
    if (tagName=="TAG_LOGIN") {
```

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

```
        Login_CMD = tagState.getValue();
    }

    if (Username!="" && Password!="" && Login_CMD!=""){
        if (Login_CMD==1) {
            Reply = project.login(Username, Password);
        };
        if (Login_CMD==2) {
            Reply = project.logout(false); // Logout only if not logged as default
user
        };
        project.setTag("TAG_LOGIN", 0);
        project.setTag("TAG_REPLY", parseInt(Reply));
    }
}
```

Voir aussi :

- "Ouverture de session" Page 457

Limites lors du travail avec des widgets dans JavaScript

Les widgets ne peuvent pas être instanciés par JavaScript, ils ne peuvent qu'être accédés et modifiés. Si vous avez besoin de widgets supplémentaires sur la page, vous pouvez ajouter des widgets masqués sur la page, puis les afficher ou les positionner en utilisant JavaScript.

Déboguer JavaScript

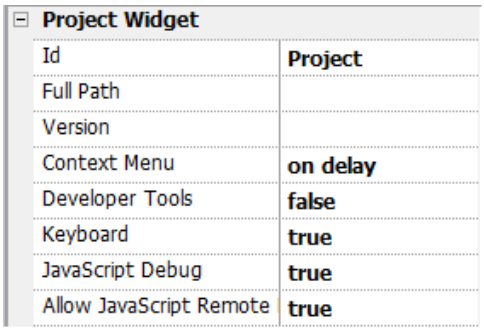
PB610 Panel Builder 600 et HMI Runtime incluent un débogueur JavaScript.

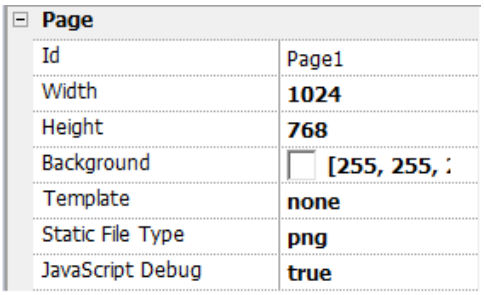
Deux types de débogueurs sont disponibles :

- Débogueur de runtime : un débogueur qui s'exécute directement sur le périphérique IHM
- Débogueur à distance : un débogueur qui s'exécute sur un ordinateur à distance connecté au périphérique IHM via Ethernet (en général, un ordinateur qui exécute PB610 Panel Builder 600)

Activer le débogage

Dans le volet **Propriétés** d'une page, régler le **Débogage JavaScript** sur **vrai**.

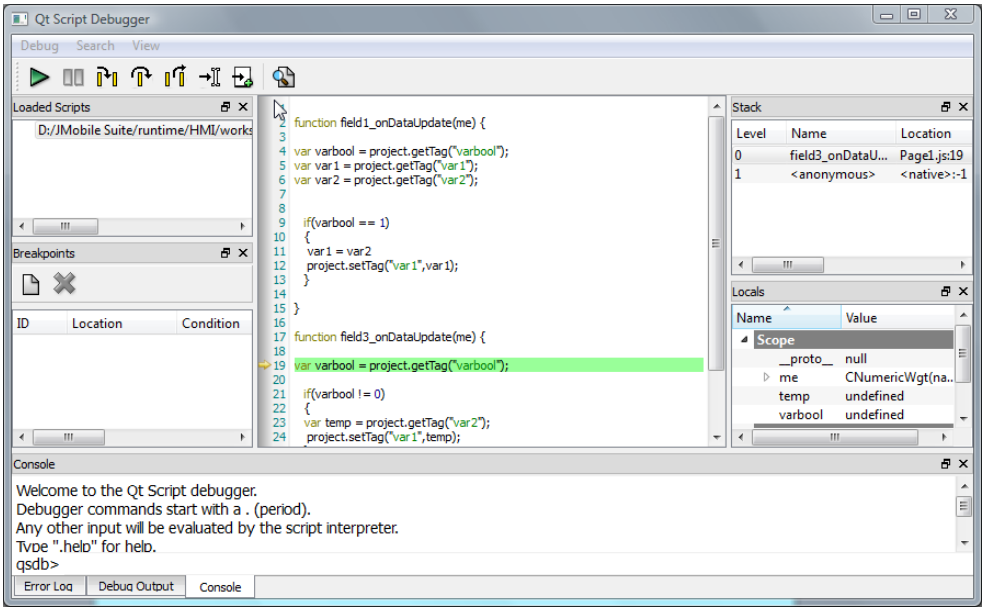




Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Pour le débogage des planificateurs et d'alarmes, activez le Débogage JavaScript dans Propriétés du projet.

Dans HMI Runtime, le débogueur affiche les informations de débogage lorsque les événements sont appelés. Dans le volet **Locaux**, vous pouvez inspecter toutes les variables et tous les éléments.



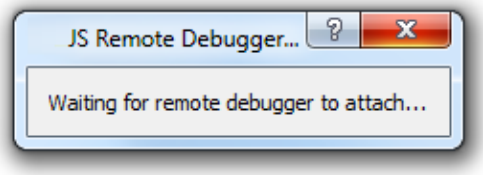
Pour obtenir un guide de référence complet sur le débogueur de JavaScript, consultez :

http://qt-project.org/doc/qt-4.8/qtscriptdebugger-manual.html

Débogueur à distance JavaScript

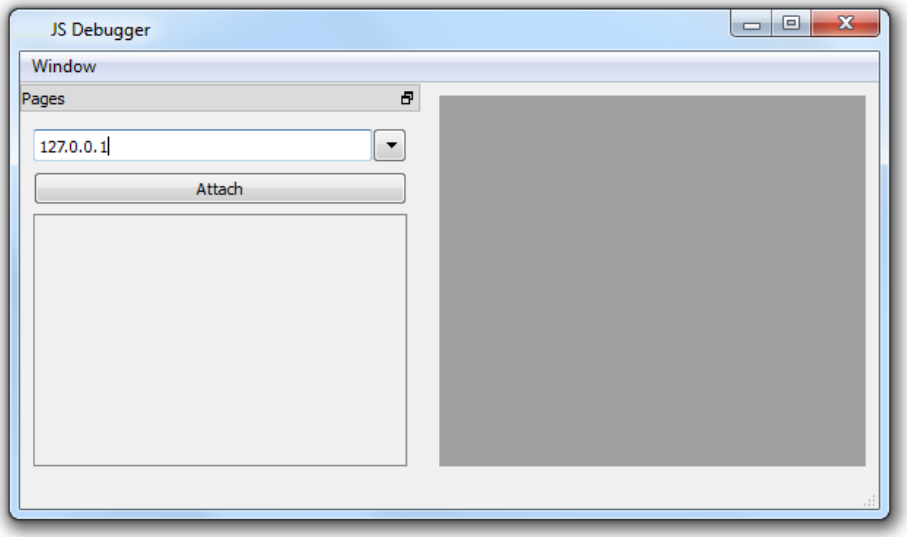
*Chemin : **Exécuter> Démarrer débogueur à distance JS***

1. Dans le projet Propriétés, définissez les paramètres **Permettre JavaScript à distance** et **Débogage JavaScript** sur Vrai dans toutes les pages qui requièrent un débogage.
2. Téléchargez le projet : le message suivant s'affiche en cours de runtime.



Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

3. Dans la fenêtre **Débogueur JS**, sélectionnez l'IP du périphérique IHM et cliquez sur **Associer** pour connecter le débogueur au périphérique IHM.



Le débogueur à distance JavaScript se connecte à HMI Runtime via un port 5100/TCP.

L'outil de débogage à distance de JavaScript n'est pas pris en charge dans HMI Client.

Utilisation de la mémoire JavaScript

Lorsque la mémoire dépasse le maximum, une exception mémoire pleine est lancée avec un message personnalisé. Notez que nous n'avons pas un contrôle précis de l'utilisation réelle de la mémoire, il s'agit donc d'une limite floue. De plus, nous ne pouvons pas interdire l'allocation (cela arrêterait l'implémentation du moteur), une exception est donc lancée uniquement lorsque la mémoire a déjà dépassé la limite. Avant de lever l'exception, un nettoyage de la mémoire est forcé pour voir si de la mémoire peut être libérée.

Vous pouvez accéder à la limite de la mémoire JavaScript depuis l'objet global $EngineMemory. Elle est par défaut de 16 MB, ce qui devrait suffire pour une utilisation habituelle de JavaScript (principalement des commandes, avec peu d'allocations).

- $EngineMemory.setLimit()
  définir la mémoire maximale autorisée pour JavaScript (la limite par défaut est 0x00FFFFFF)
- $EngineMemory.getLimit()
  obtenir la mémoire maximale autorisée pour JavaScript
- $EngineMemory.getSize()
  obtenir la mémoire en cours d'utilisation par JS (fastMallocStat)

Tester l'exception de mémoire

Pour générer et tester les exceptions de mémoire, vous pouvez utiliser le bout de code suivant. Notez que nous devons réinitialiser la limite de la mémoire à 0xffffffff pour exécuter l'alerte, sinon les allocations de mémoire requises pour afficher l'alerte échoueraient.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

```
try
{
    // Generate out-of-memory error
    var a = [];
    while(1)
    {
        a.push("a");
    };
} catch(e)
{
    // Ensure there is enough memory to pop up error message
    $EngineMemory.setLimit(0xffffffff);
    alert("Exception: " + e);
};
```

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

38 Gestes de prise en main

Certains widgets ont la capacité de détecter et de gérer les gestes de déplacement et de pincement.

- Tendances (voir "Gestes de widget de tendance" Page 236 pour plus de détails)
- Widget d'alarmes
- Widget liste déroulante
- Widget tableau
- Afficheur PDF
- widget de zone de Geste. Widget spécial conçu pour personnaliser la gestion des événements de geste (voir "Widget de zone de mouvement" Page 375 pour plus de détails)

Pour les widgets avec une présentation en tableau, quand le paramètre **Type de défilements** a été réglé sur « Gestes », le mouvement de déplacement est utilisé pour faire défiler le tableau en douceur.

- Alarmes
- Liste de contrôle

AVERTISSEMENT : Le geste de pincement et de rotation nécessite deux doigts. Il est disponible uniquement avec les périphériques IHM supportant un fonctionnement tactile multipoint (voir "Capacités des périphériques IHM" Page 541

Astuce : Si vous utilisez un périphérique IHM multi-touch, vous pouvez mettre en place des commandes sécurisées en programmant des commandes qui ne seront exécutables que par pression simultanée de deux boutons.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

39 Paramètres du système

Les paramètres du système sont un outil interne du périphérique IHM qui peut être utilisé pour les réglages de base du périphérique ou pour la mise à jour des composants du système.

Remarque : les composants du système peuvent être mis à jour dans PB610 Panel Builder 600 (voir "Mettre à jour les composants système des périphériques IHM" Page 523 pour les détails)

AVERTISSEMENT : travailler avec l'outil Paramètres système est une opération critique et, lorsqu'elle n'est pas effectuée correctement, peut causer des dommages au produit nécessitant une réparation du produit. Contactez le support technique pour de l'aide.

Périphériques Linux **478**

Périphériques WinCE **499**

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Périphériques Linux

les produits CP600-eCo offrent un outil intégré puissant appelé Paramètres du Système qui permet la gestion et la mise à niveau des composants du système. Les opérations peuvent être effectuées directement sur l'IHM ou à distance à l'aide d'un navigateur Web.

Installation du runtime

Les périphériques IHM sont livrés d'usine sans Runtime, à la première mise sous tension, l'IHM affiche l'écran « Chargeur de runtime ».

Le runtime peut être installé :

- Automatiquement, via Ethernet lors du premier téléchargement de projet avec PB610 Panel Builder 600
- Manuellement via la mémoire USB, en créant un « Mise à jour de paquet ». (Voir "Mettre à jour un paquet" Page 91 pour créer un paquet de runtime)

Installation du Runtime via Ethernet

Pour l'installation du Runtime via Ethernet, suivre la procédure "Télécharger vers le périphérique IHM" Page 88.

AVERTISSEMENT : le téléchargement via Ethernet de l'installation du runtime requière le périphérique IHM avec une adresse IP valide.

L'adresse IP peut être attribuée de trois manières :

- *Automatiquement via serveur DHCP.* Cette option est activée par défaut. Si un serveur DHCP est disponible sur le réseau, l'adresse IP sera attribuée automatiquement par le serveur.
- *Automatiquement via la fonctionnalité Auto-IP.* Si l'assignation DHCP est activée mais qu'aucun serveur DHCP n'est disponible sur le réseau, le périphérique IHM s'attribue une adresse IP dans la plage 169.254.xx avec un masque de sous-réseau 255.255.0.0
- *Manuellement via les Paramètres système.* Dans le menu Paramètres du Système, dans la section réseau, l'adresse IP peut être attribuée manuellement, désactivant la fonction d'attribution du serveur DHCP.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

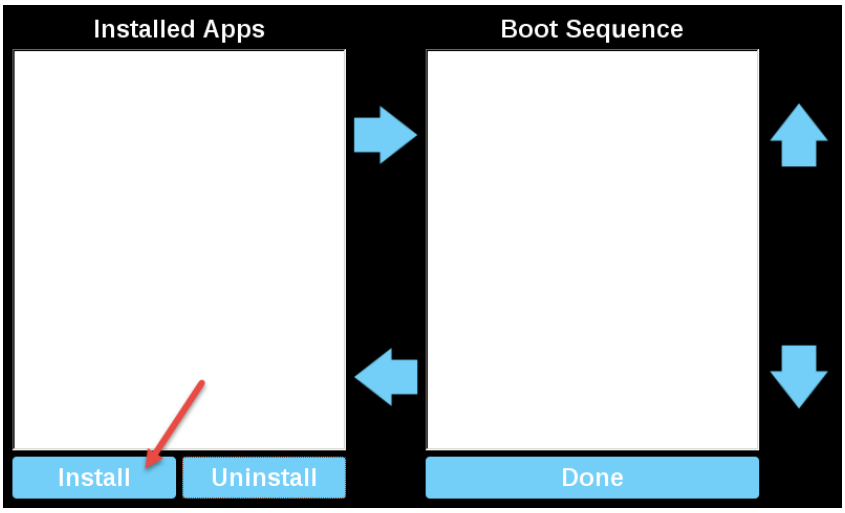
Installation du runtime via une mémoire USB

Pour l'installation du runtime, UpdatePackage ou Backup Package via un périphérique USB, suivre cette procédure :

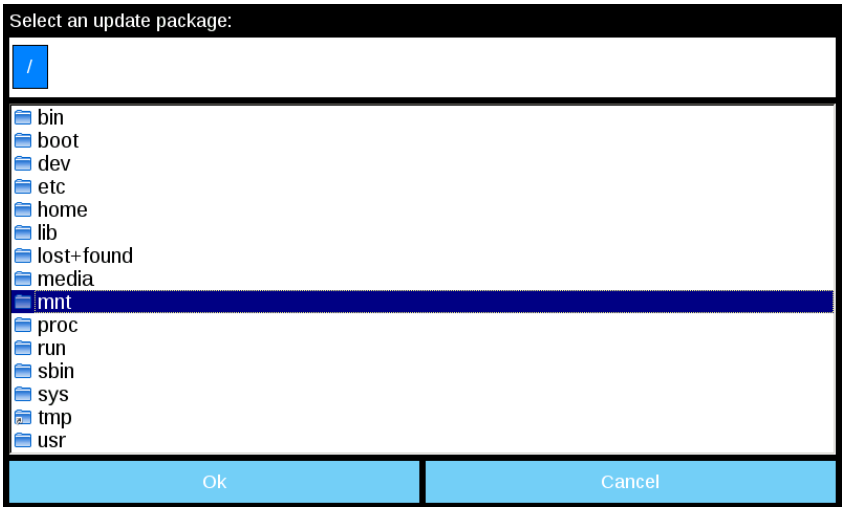
1. Créer un paquet de mise à jour depuis PB610 Panel Builder 600 et le copier sur une clé USB vierge

Remarque : les systèmes de fichiers pris en charge sont FAT16/32 et Linux Ext2, Ext3 et Ext4.

2. Sur le périphérique IHM, sélectionner [Startup sequence], puis [Install]

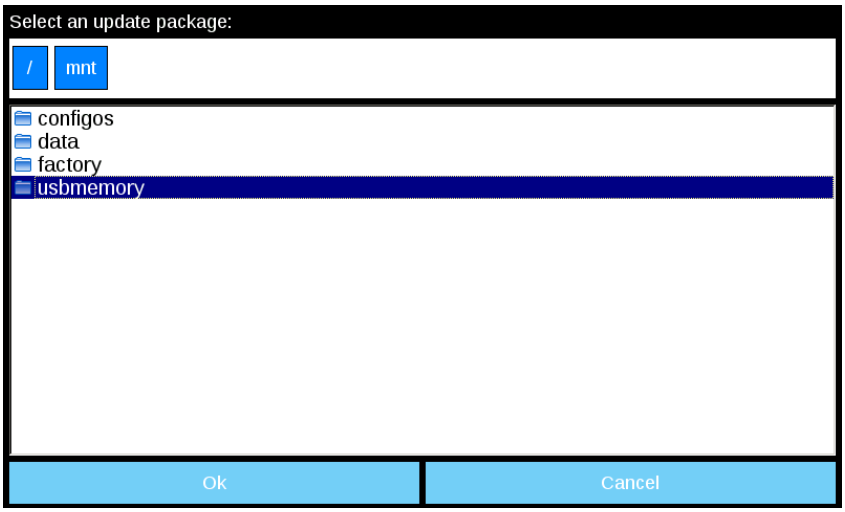


3. Double-cliquer sur « mnt » pour accéder au dossier

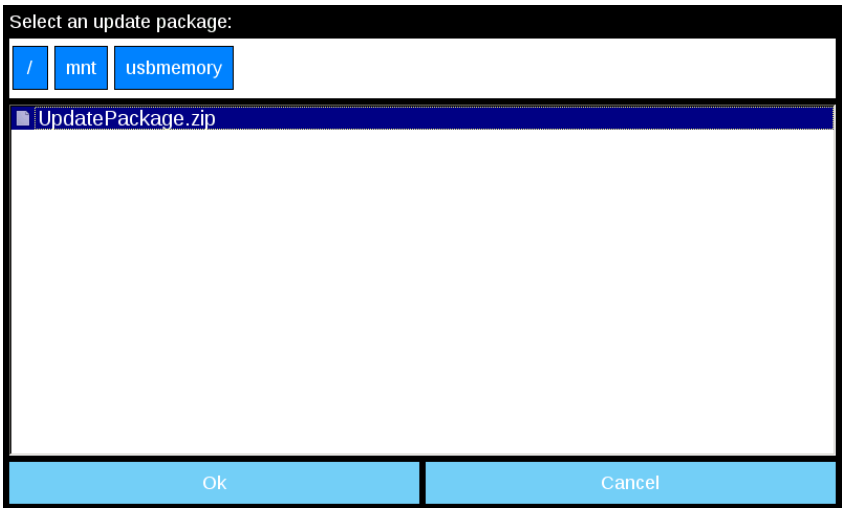


4. Puis sur « usbmemory »

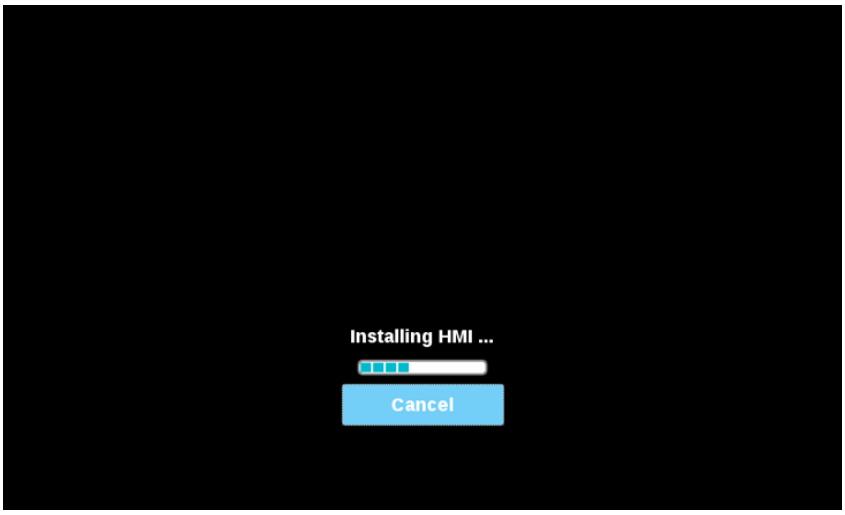
Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



5. Sélectionner « UpdatePackage.zip » et confirmer en appuyant sur [Ok]

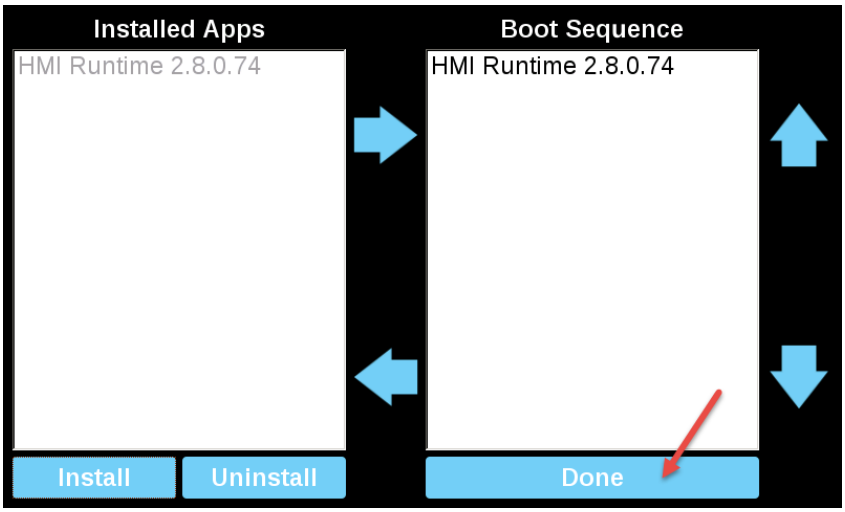


6. L'installation du runtime commence

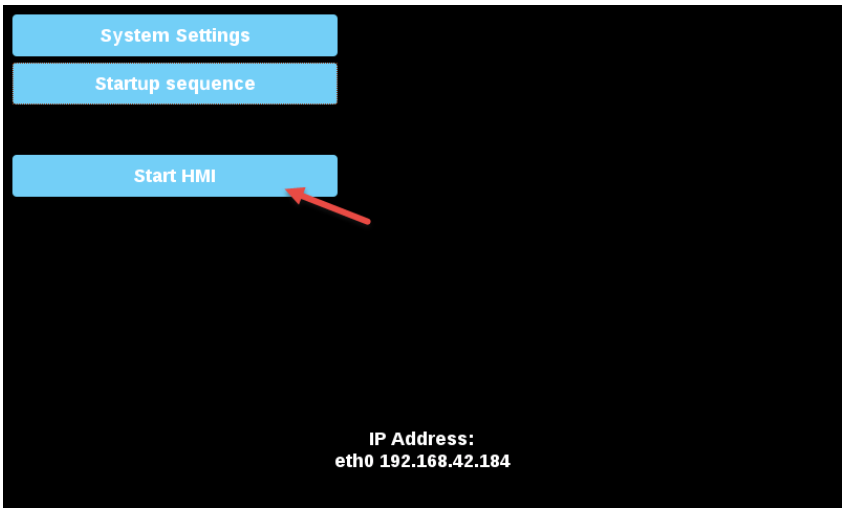


7. À la fin, appuyez sur le bouton « Done » (Terminé)

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



8. Puis sur le bouton « Start IHM »



Paramètres du système

L'interface utilisateur des Paramètres systèmes se base sur des pages HTML et il est possible d'y accéder soit localement sur l'écran du périphérique IHM, soit à distance en utilisant un navigateur web.

L'identifiant de l'administrateur en possession de tous les droits d'accès est « admin » et le mot de passe par défaut est « admin ». L'identifiant générique est « user » et le mot de passe par défaut est « user »

AVERTISSEMENT : Pour des raisons de sécurité, veuillez changer les mots de passe par défaut pour les deux identifiants (les mots de passe peuvent être modifiées à partir de la commande « Paramètres système -> Authentification »)

L'accès aux paramètres du système du périphérique IHM ne nécessite pas d'entrer un mot de passe tant que le mot de passe « admin » par défaut ne'est pas modifié.

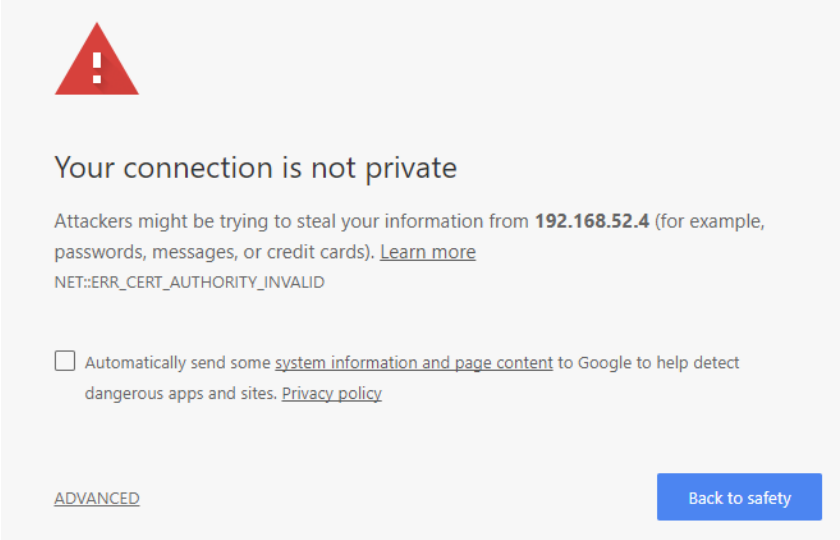
Accès aux paramètres du système depuis un navigateur Web

Pour accéder aux Paramètres système via un navigateur Web, saisir l'adresse IP du périphérique, selon le format suivant :

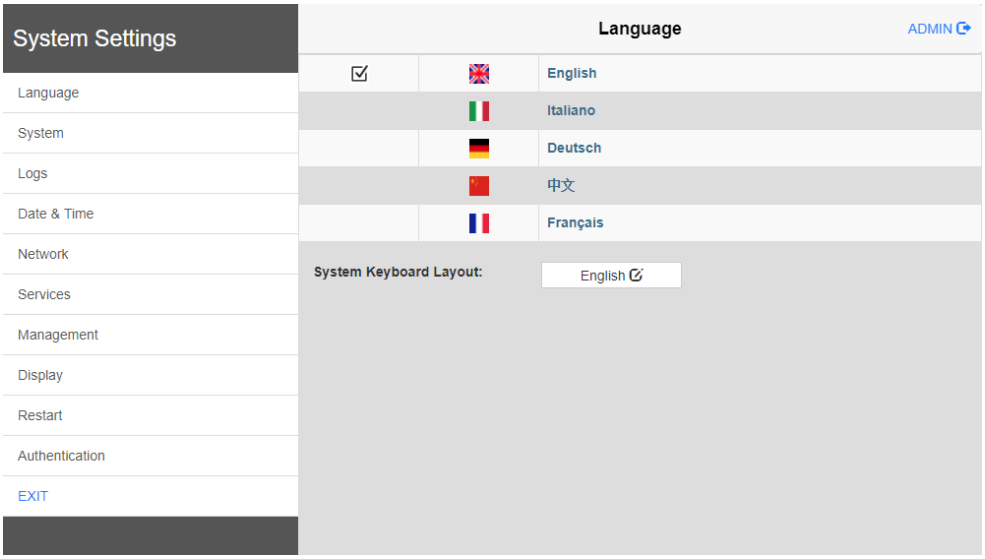
https://IP/machine_config

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Notez que l'accès à distance utilise le protocole https crypté sur le port 443. Lorsque la connexion est établie, l'IHM envoie un certificat à utiliser pour le cryptage. Comme le certificat n'est pas signé par une autorité de certification, vous recevrez un message d'avertissement. Veuillez cliquer sur les options avancées et le choix pour continuer.



Parcourir les options disponibles dans le menu à gauche : l'élément actif est surligné et les informations correspondantes s'affichent sur la droite.



Les protocoles de sécurité par défaut proposés par le serveur HTTPS dans le CP600-eCo périphérique IHM sont :

- SSLv3 256 bits ECDHE-RSA-AES256-SHA
- TLSv1 256 bits ECDHE-RSA-AES256-SHA

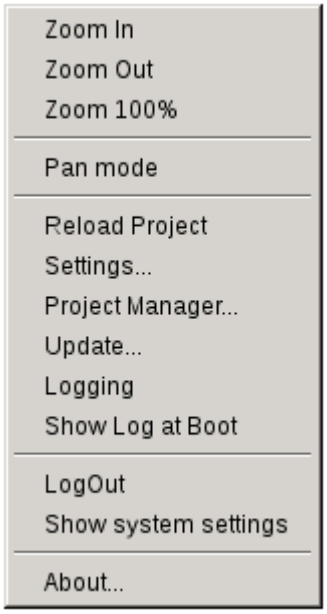
AVERTISSEMENT : nous déconseillons l'utilisation de suites de chiffrement CBC en cas de connexions SSL3 ou TLSv1.0, car elles pourraient être affectées par des vulnérabilités.

Accès aux paramètres du système depuis un périphérique IHM

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Lorsque le Runtime n’est pas installé, les Paramètres du Système sont accessibles à partir de l'écran du chargeur de runtime,

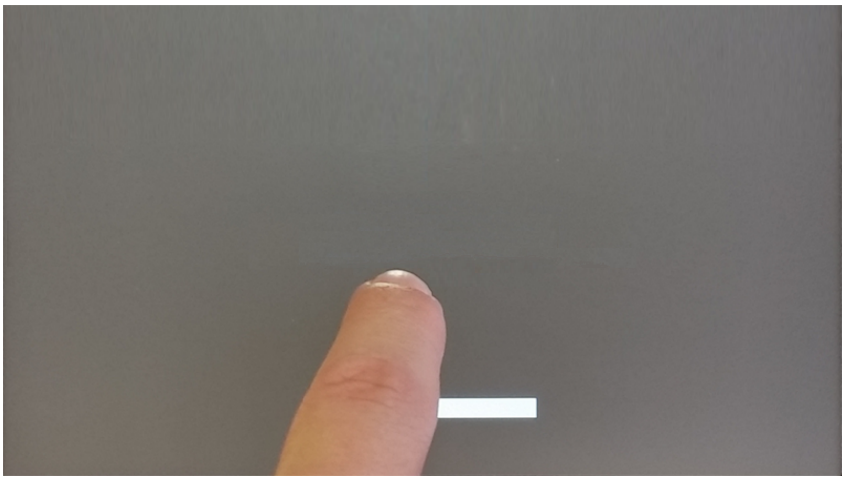
Lorsque le Runtime est installé, les Paramètres du système sont accessibles en sélectionnant l'option « Show System Settings » du menu contextuel,



Entrer dans les Paramètres du système via la procédure tape-tape

Tape-tape consiste en une suite de plusieurs activations tactiles par des moyens simples du doigt en tapant sur l'écran tactile pendant la phase de mise sous tension et commencé immédiatement après la mise sous tension de l’IHM.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



Lorsque le message « tape-tape détecté » apparaît sur le haut de l'écran, attendre 5 secondes (sans toucher l'écran) pour entrer dans le sous-menu Paramètres de système

Attendre encore 5 secondes (sans toucher l'écran) pour entrer en mode par Défaut

Sélectionner « Paramètres du Système » depuis l'écran du mode par défaut IHM

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Sections de Paramètres du système

Pour modifier les valeurs des paramètres du système, entrer en mode édition en cliquant sur le bouton Éditer en haut à droite.

Le bouton Éditer est disponible uniquement à l'intérieur des boîtes de dialogue qui contient les paramètres modifiables.

Langues

Sélectionner la langue de l'interface des paramètres du système

- Disposition du clavier système : sélectionnez la disposition du clavier virtuel.

Système

Paramètre	Description
Info	Informations relatives au périphérique
Status	État du périphérique (Free RAM, Up time, CPU Load)
Timers	Minuteurs périphérique (System on, Back light on)
Module d'exécution (plug-in)	Informations sur les modules d'exécution matériels

Journaux

Définissez l'option journal permanent si vous souhaitez conserver les fichiers journaux enregistrés après une réinitialisation de l'alimentation.

Utilisez le bouton Enregistrer pour exporter une copie des fichiers journaux.

Le gestionnaire de fichiers journaux remplit cycliquement 3 fichiers de 4Mo.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Date et Heure

Date et heure du périphérique.

Paramètre	Description
Fuseau horaire actuel	Fuseau horaire
Date actuelle **Heure locale**	La date et l'heure peuvent être défini manuellement uniquement lorsque la mise à jour automatique est désactivée.
Mise à jour automatique (NTP)	Activer pour garder date et l'heure synchronisées à partir d'un serveur distant • Serveur NTP Spécifier l'adresse Internet du serveur NTP

Réseaux

Paramètre de réseau. Paramètre disponible en mode édition :

Paramètre	Description
Paramètres généraux	Adresse Interner du périphérique Nom d'hôte Avahi (voir "Démon Avahi" bas).
Interface réseau	Paramètres réseau des interfaces disponibles • DHCP • Adresse IP • Masque réseau • Gateway
DNS	Serveurs DNS Fourni généralement à partir des serveurs DHCP, mais peut être modifié en mode édition Domaines e recherche Domaines en option qui seront utilisés en concaténation avec les url fournies

Services

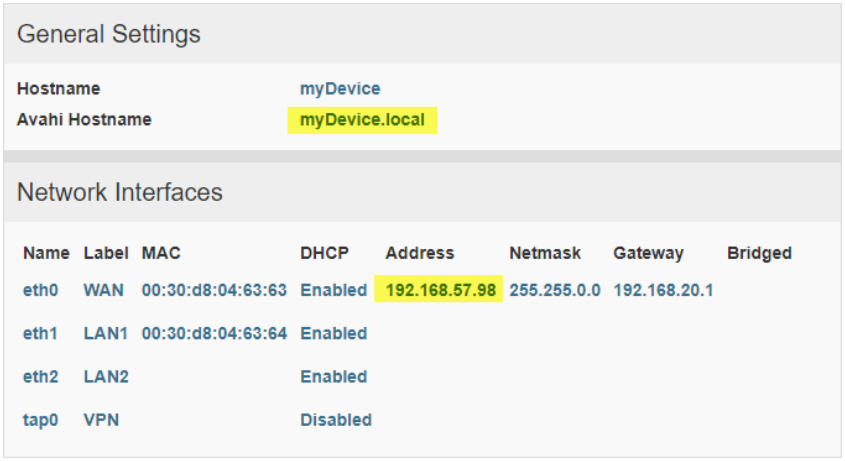
Les services sont disponibles uniquement lorsque vous êtes connecté en tant qu'administrateur.

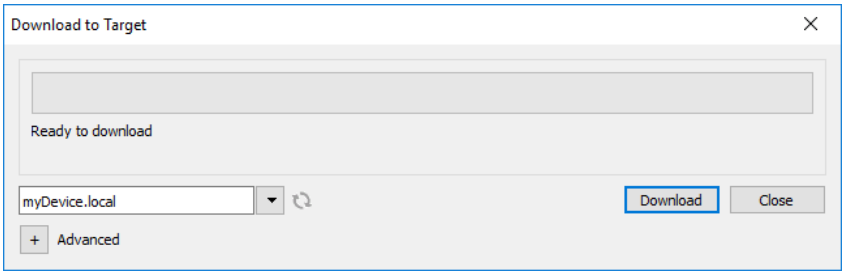
Cliquez avec la souris sur le bouton d'activation pour activer /désactiver le service. Cliquez sur le nom du service pour lister les paramètres associés.

Démon Avahi

Avahi est un système qui permet aux programmes de publier et de détecter les services et les hôtes fonctionnant sur un réseau local. Lorsqu'il est activé, le périphérique IHM peut être atteint même en utilisant le nom d'hôte du périphérique (en alternative à l'adresse IP).

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21





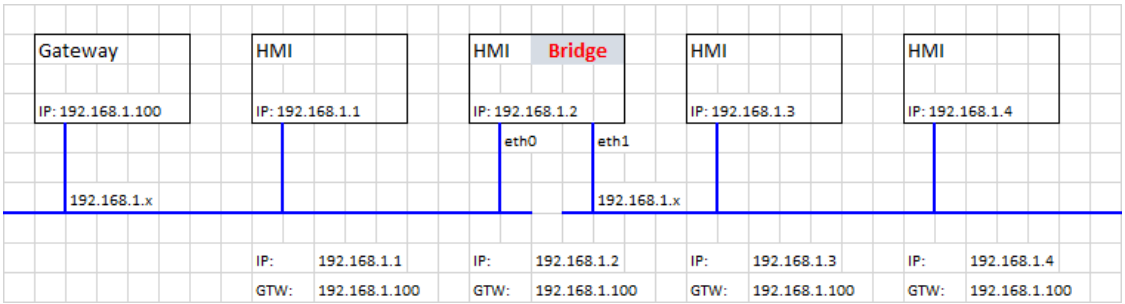
Le démon Avahi tourne sur UDP port 5353

Sur les PC Linux et Apple, le service Avahi est fourni gratuitement avec le système d'exploitation. Sur les PC Windows, vous devez installer un service Avahi pour pouvoir accéder au panneau par son nom d'hôte Avahi (par exemple, vous devez installer l'application Apple Bonjour - Bonjour est une marque de commerce d'Apple inc.

Service Pont

À l'aide du service pont, il est possible de connecter ensemble l'adaptateur réseau WAN (eth0) avec les adaptateurs réseau LAN (eth1 ou eth2). Lorsqu'ils sont utilisés, les deux adaptateurs Ethernet sont pontées et les deux adaptateurs Ethernet partagent la même adresse IP.

Le service Pont crée un pont réseau de couche 2 basé sur Linux entre deux ou plusieurs interfaces réseau. Si des dispositifs WAN et des terminaux sont connectés à ce pont, les deux réseaux seront physiquement reliés et les terminaux seront disponibles comme s'ils étaient directement connectés au WAN (Note : le scénario en nuage exige toujours que le service de routeur soit actif).



Démarrage rapide

Lorsque le démarrage rapide est activé, à la mise sous tension, le périphérique IHM démarrera l'application IHM aussi rapidement que possible. Dans ce mode, il n'y a pas d'informations de diagnostic affichées (par exemple, la barre de

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

chargement) mais seulement les fonctions minimales nécessaires sont chargées avant de charger l'interface utilisateur (par exemple, les paramètres système, VNC, SSH, etc. seront chargés après le chargement de l'application IHM).

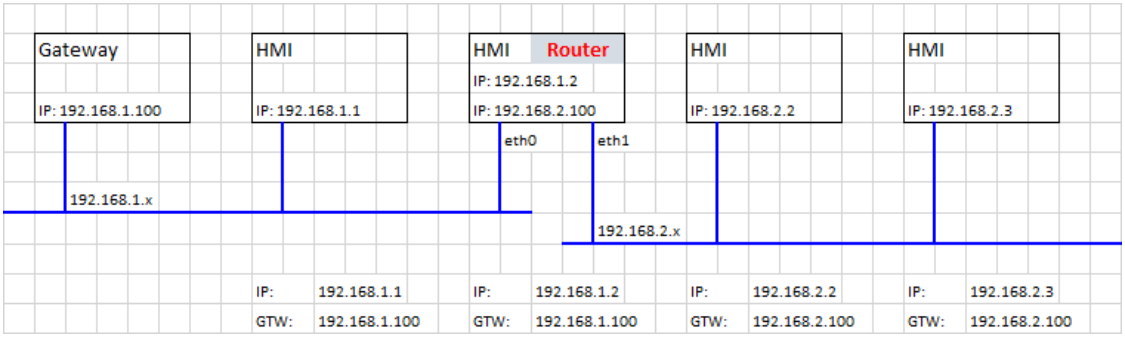
Pour obtenir les meilleures performances, en plus d'activer le mode de démarrage rapide, il est recommandé de :

- désactiver tout service qui n'est pas nécessaire
- éviter de garder activé le journal permanent
- utiliser l'adresse IP statique au lieu du service DHCP

BSP v1.0.242 requis ou plus

Service Router

Ce service utilise la redirection de l'IP et la traduction d'adresse réseau pour partager les connexions du WAN (eth0) vers le LAN (eth1 ou eth2) : les terminaux connectés peuvent atteindre les mêmes réseaux accessibles par la passerelle (y compris Internet si disponible)

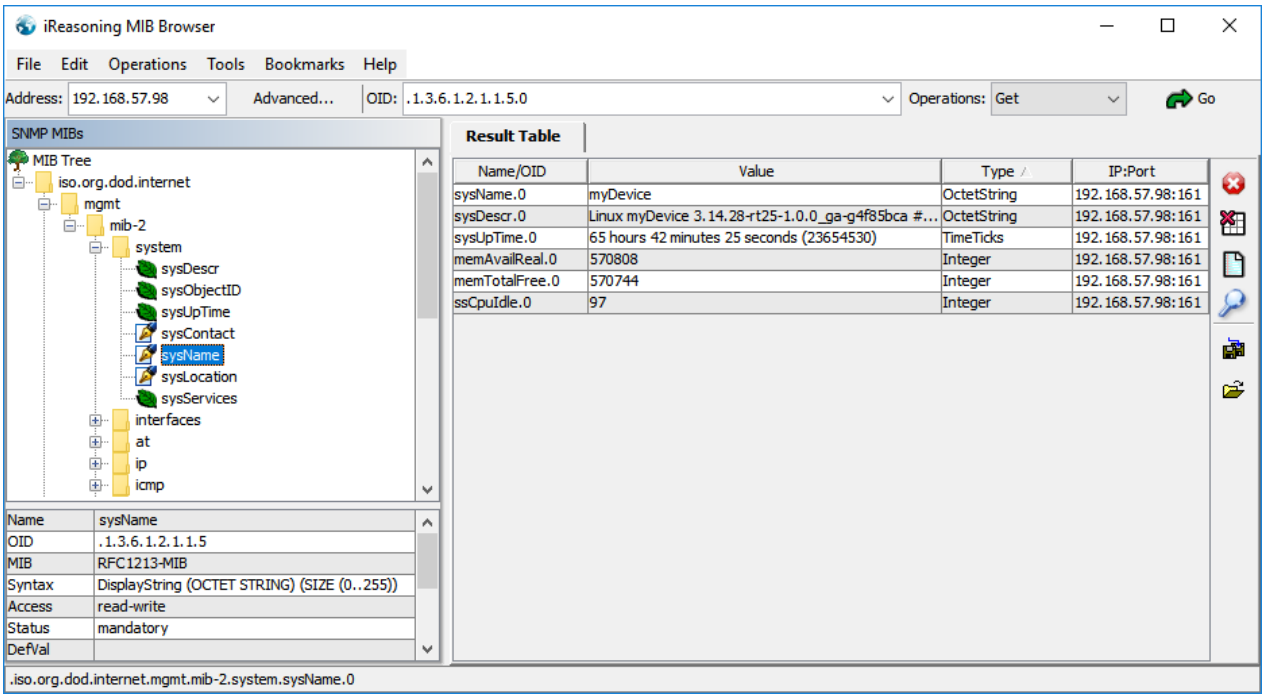


Serveur SNMP

SNMP est un protocole réseau qui permet de gérer les infrastructures réseau. Il est couramment utilisé pour surveiller les périphériques réseau tels que les commutateurs, les routeurs, etc. connectés à un réseau LAN.

Lorsque le service SNMP est activé, un gestionnaire SNMP peut récupérer des informations à partir du périphérique IHM en utilisant le protocole SNMP. Actuellement, il n'y a pas de MIB propriétaires disponibles. Seuls les MIB usuels de la communauté publique sont disponibles en mode lecture seule.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



Exemple :

Nom du système :	.1.3.6.1.2.1.1.5.0
Description du	.1.3.6.1.2.1.1.1.0
système :	.1.3.6.1.2.1.1.3.0
Temps de montée du	.1.3.6.1.4.1.2021.4.6.0
système	.1.3.6.1.4.1.2021.4.11.0
RAM totale utilisée :	.1.3.6.1.4.1.2021.11.11.0
RAM totale libre :	
Temps UC inactif (%) :	

Le serveur SNMP tourne sur UDP port 161

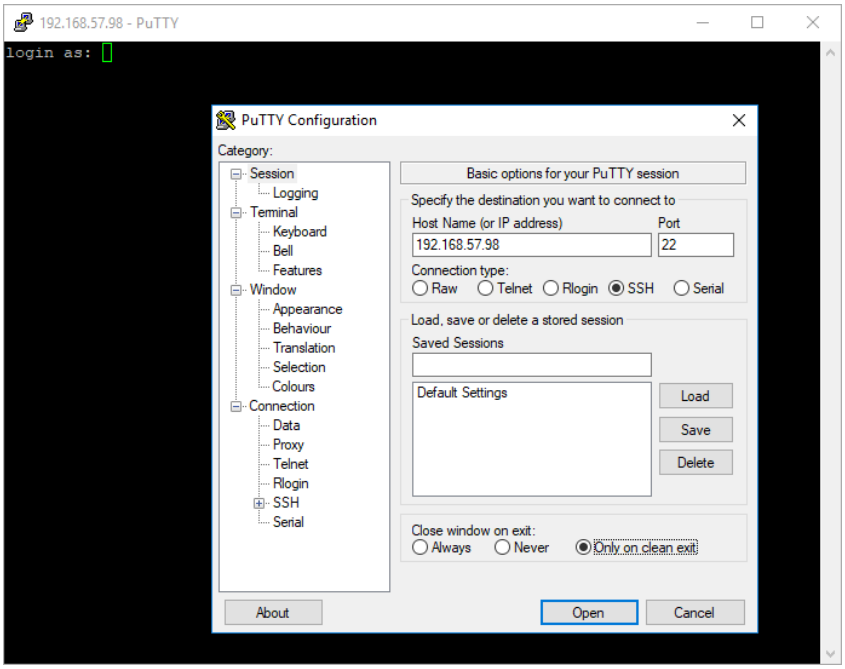
Ce service est conçu pour être utilisé pendant la phase de développement. Pour des raisons de sécurité, pensez à désactiver ce service avant de passer à la production.

Serveur SSH

Le service SSH a été conçu uniquement pour les utilisateurs avancés. Il permet de se connecter à distance au périphérique IHM en utilisant le protocole interpréteur de commandes (shell) sécurisé. Sur PC, vous pouvez exécuter un Client SSH comme, par exemple, PuTTY qui est un logiciel libre distribué sous licence MIT.

Le mot de passe par défaut pour le nom d'utilisateur de l'administrateur est « admin ». Voir le chapitre "Authentification" Page 491 pour plus d'informations.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



Le serveur SSH tourne sur TCP port 22

Ce service est conçu pour être utilisé pendant la phase de développement. Pour des raisons de sécurité, pensez à désactiver ce service avant de passer à la production.

Service VNC

VNC est un service qui permet d'accéder à distance à l'affichage de l'IHM. Les clients VNC peuvent être utilisés pour bénéficier de la commande à distance du périphérique IHM.

- Port : Le serveur VNC écoute les connexions sur le port TCP 5900 (par défaut)
- Clients multiples : autoriser plusieurs sessions sur le même port (si désactivé, les clients précédemment enregistrés sont déconnectés lors d'une nouvelle connexion entrante).
- Visualisation seulement : n'autorise pas les interactions entre utilisateurs actifs (les clients peuvent seulement regarder)
- Cryptage : active le cryptage SSL des connexions (pas beaucoup pris en charge - vérifiez la compatibilité client)
- Authentification : dans le cas où les utilisateurs sont authentifiés lors de la création de session. Un mot de passe spécifique personnalisé VNC peut être défini ou des mots de passe système peuvent être utilisés (cette option n'est disponible que si le Cryptage est aussi activé).

Modules d'exécution

Cette page affichera les paramètres disponibles pour configurer les modules d'exécution optionnels associés à l'IHM. Voir la description de chaque module pour plus d'informations.

Gestion

La gestion est disponible uniquement lorsque vous êtes connecté en tant qu'administrateur.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Depuis la zone Gestion, il est possible d'effectuer la "Mise à jour des composants du système" Page suivante du périphérique IHM.

AVERTISSEMENT : travailler dans la zone Gestion est une opération critique et, lorsqu'elle n'est pas effectuée correctement, peut causer des dommages au produit nécessitant une réparation du produit. Contactez le support technique pour de l'aide.

Utilisez la commande « Effacer » dans la section « Données » pour supprimer le runtime IHM du périphérique (restauration paramètres d'usine)

Affichage

Paramètre	Description
Luminosité	Niveau de luminosité de l'écran
Délai de retour de lumière	Délai d'attente inactivité arrière plan
Orientation	Orientation de l'écran

Restart

Commande de redémarrage du périphérique IHM

Authentification

Entrer en mode édition pour modifier les mots de passe d'authentification.

L'identifiant de l'administrateur en possession de tous les droits d'accès est « admin » et le mot de passe par défaut est « admin ». L'identifiant générique est « user » et le mot de passe par défaut est « user »

AVERTISSEMENT : Pour des raisons de sécurité, veuillez changer les mots de passe par défaut pour les deux identifiants

Certificat x.509

Les périphériques IHM utilisent un autocertificat pour crypter la communication Internet via le protocole HTTPS. Vous pouvez personnaliser le certificat avec les données de votre entreprise et demander à une autorité de certification de le valider.

La procédure pour personnaliser et valider votre certificat est :

1. Entrez en mode édition et remplissez les paramètres nécessaires, puis appuyez sur la touche GENERATE pour générer un certificat auto-signé avec vos données.
2. Exporter la « Demande de certificat signé »
3. Envoyez la « Demande de certificat signée » à une autorité de certification pour validation (en général, il s'agit d'un service payant)
4. Importez le certificat signé dans le périphérique IHM

Paramètre du certificat

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Paramètre	Description
Device Name	Le nom de votre périphérique
Organization	Le nom légal de votre organisation
Unit	Le service de votre organisation qui traite le certificat
État	L'état/ la région où votre organisation est située
Location (Localité)	La ville où votre organisation est située
Country	Le code ISO à deux lettres du pays où se trouve votre organisation
Valid (days)	Validité du certificat
Key Length	Nombre de bits de la clé utilisée à partir de l'algorithme cryptographique

Les certificats gérés sont codés en base 64

BSP v1.0.239 requis ou plus

EXIT

Sortir de l'outil des paramètres du système.

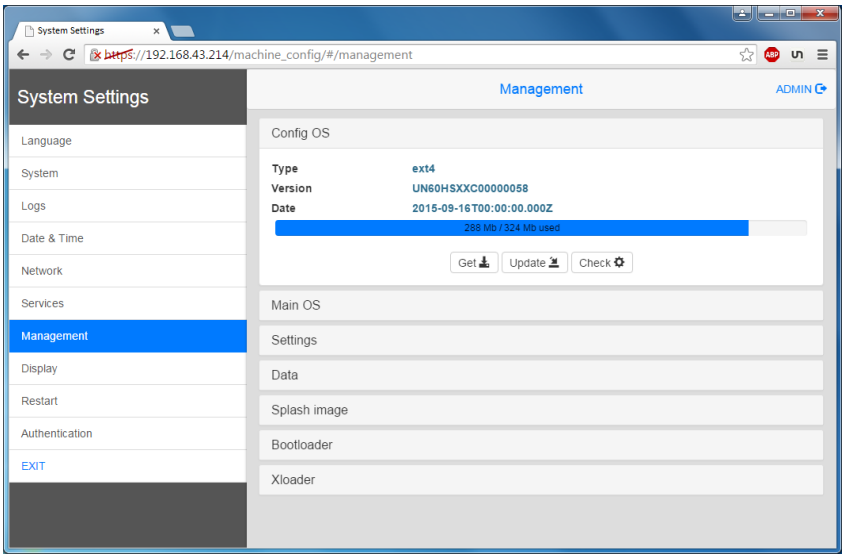
Mise à jour des composants du système

AVERTISSEMENT : travailler dans la zone Gestion est une opération critique et, lorsqu'elle n'est pas effectuée correctement, peut causer des dommages au produit nécessitant une réparation du produit. Contactez le support technique pour de l'aide. (les tout derniers fichiers BSP seront fournis par le support technique).

Les composants du système du périphérique CP600-eCo peuvent se mettre à jour localement à l'aide d'une clé USB ou à distance via un navigateur Web.

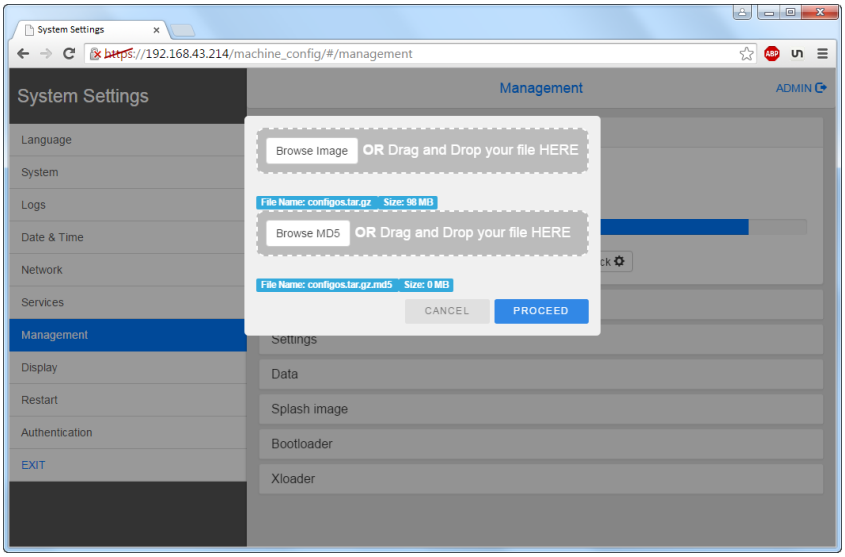
Pour mettre à jour les composants du système, entrer dans Paramètres du système en mode Config OS via la procédure tape-tape sur IHM ou ouvrir le navigateur Web sur https: // <HMI-IP-address> et sélectionner la section « Gestion ».

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



Développer le composant à mettre à jour et sélectionner [Mise à jour]

Dans la boîte de dialogue ouverte, cliquez sur [Parcourir image], puis sélectionnez le fichier « xxx-Mainos-xxx.tar.gz ». Cliquez ensuite sur [Parcourir MD5] et sélectionnez le fichier « xxx-Mainos-xxx.tar.gz.md5 ».



Important : ne pas éteindre le périphérique alors qu'un composant du système est en cours de mise à niveau.

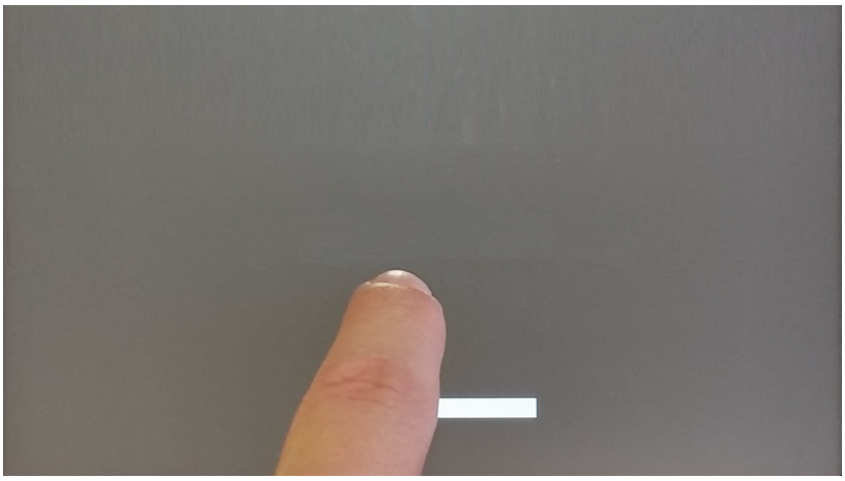
À la fin de la mise à jour des composants, redémarrez le périphérique IHM et laissez-le démarrer normalement.

Entrer dans les Paramètres en mode Config OS du système via la procédure tape-tape

Le Paramètre en mode Config OS est disponible via la séquence tape-tape, ce mode est accessible également lorsque l'IHM subit une panne de logiciel.

Tape-tape consiste en une suite de plusieurs activations tactiles par des moyens simples du doigt en tapant sur l'écran tactile pendant la phase de mise sous tension et commencé immédiatement après la mise sous tension de l'IHM.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



Lorsque le message « tape-tape détecté » apparaît en haut de l'écran, appuyez et mainteniz le doigt sur l'écran tactile, pour sélectionner « Restart: Config OS »

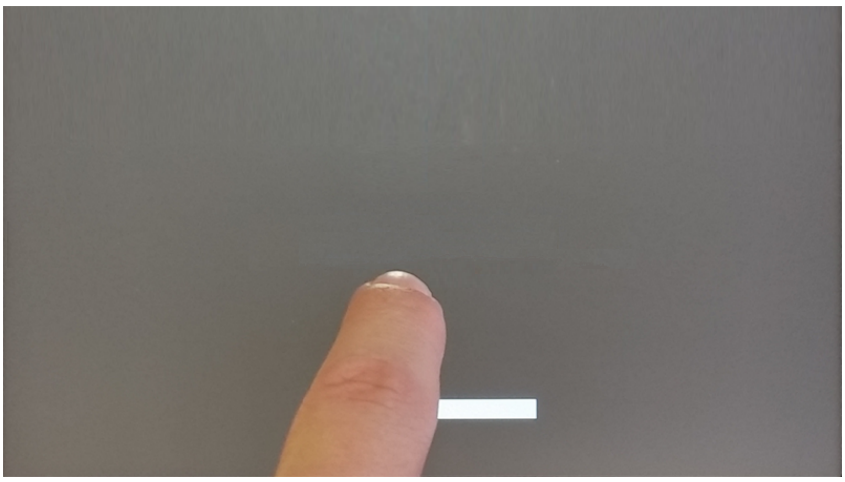
L'IHM redémarrera dans Paramètres système en mode Configuration du système d'exploitation :

Calibrage de l'écran tactile

Le calibrage des paramètres du système permet de calibrer l'écran tactile, accessible par la procédure tape-tape.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Tape-tape consiste en une suite de plusieurs activations tactiles par des moyens simples du doigt en tapant sur l'écran tactile pendant la phase de mise sous tension et commencé immédiatement après la mise sous tension de l'IHM.



Lorsque le message « tape-tape détecté » apparaît en haut de l'écran, attendre 5 secondes (sans toucher l'écran) pour entrer dans le sous-menu Paramètres système

Appuyez sur l'écran tactile, la voix « calibrage de l'écran tactile » sera surlignée en jaune, maintenez enfoncé pendant quelques secondes jusqu'à ce que la procédure de calibrage de l'écran tactile commence

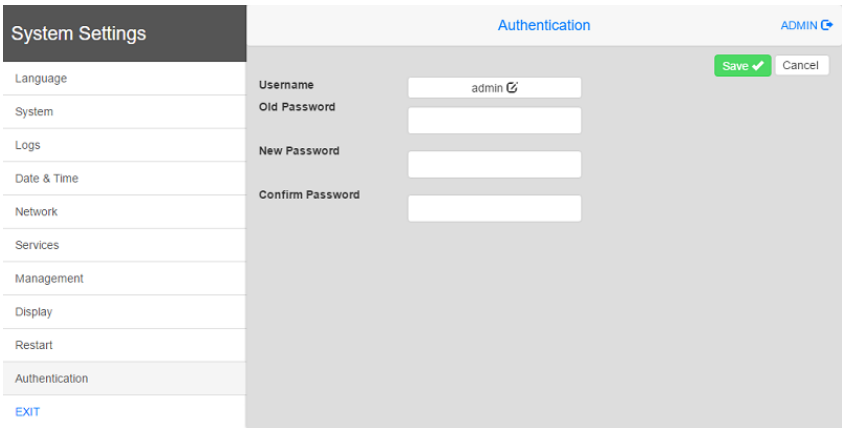
Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Suivez les instructions affichées sur l'écran pour terminer la procédure de calibrage, le système vous demandera de toucher des points spécifiques pour calibrer le périphérique écran tactile.

Protection par mot de passe

Mot de passe interne au périphérique IHM.

Dans l'onglet Authentification, dans les "Paramètres du système" Page 481, activez le mode édition et sélectionnez le nom d'utilisateur pour modifier le mot de passe associé.

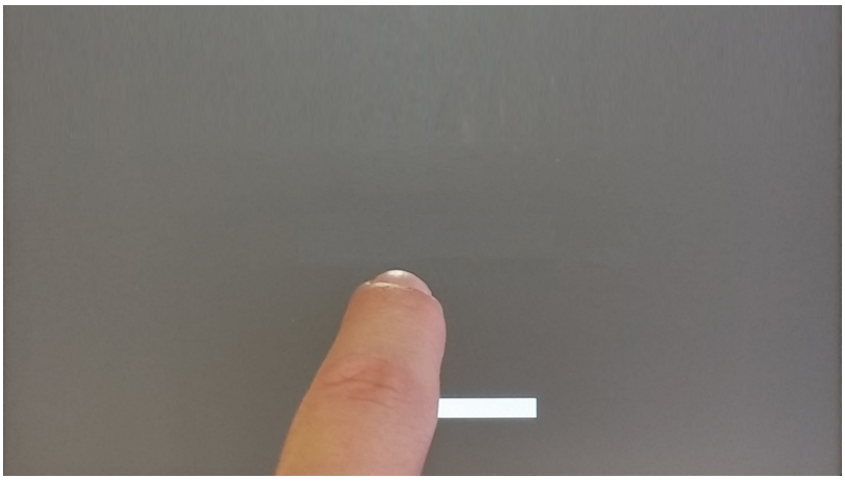


Le mot de passe pour l'utilisateur admin peut être modifié, même dans le menu contextuel de HMI Runtime (voir "Options de menu contextuel" Page 8 pour les détails).

Rétablir la configuration d'usine

Le mode par défaut des Paramètres du système permet de désinstaller le runtime IHM ou de modifier la séquence de démarrage, ce mode est disponible via la séquence tape-tape et peut être consultée également lorsque l'IHM est confronté à une panne de logiciel.

Tape-tape consiste en une suite de plusieurs activations tactiles par des moyens simples du doigt en tapant sur l'écran tactile pendant la phase de mise sous tension et commencé immédiatement après la mise sous tension de l'IHM.

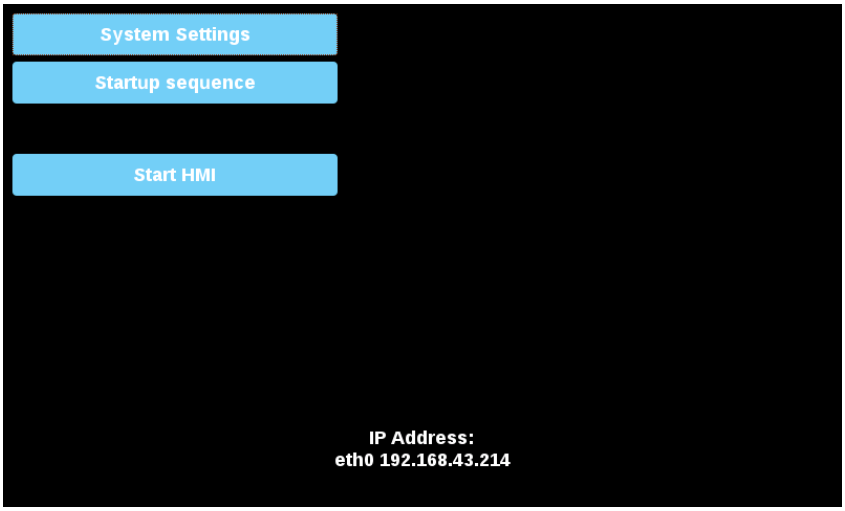


Lorsque le message « tape-tape détecté » apparaît sur le haut de l'écran, attendre 5 secondes (sans toucher l'écran) pour entrer dans le sous-menu Paramètres de système

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Attendre encore 5 secondes (sans toucher l'écran) pour entrer en mode par Défaut

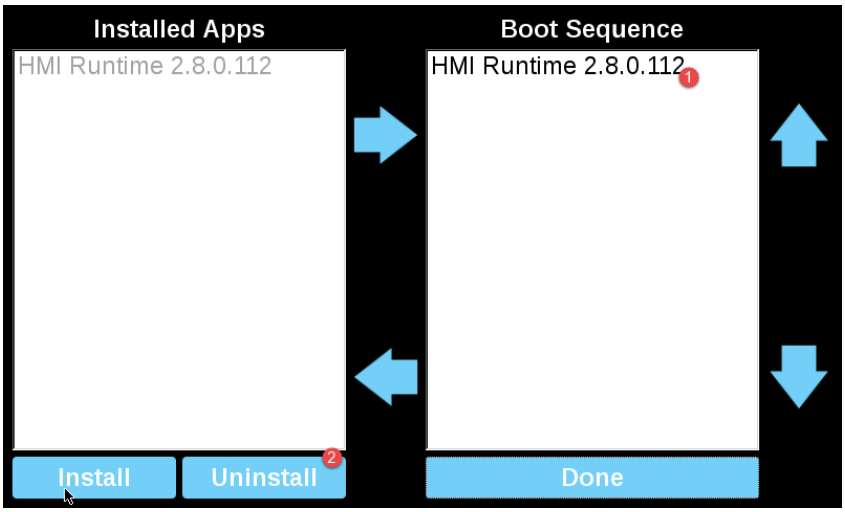
Pour désinstaller le runtime depuis l'IHM dans l'écran Mode par Défaut sélectionnez [Startup Sequence] :



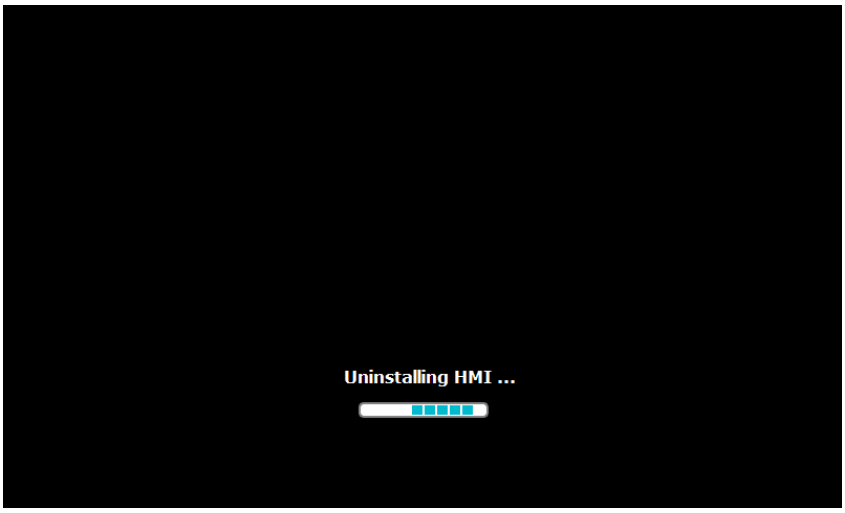
Depuis la vue des applications installées :

1. Sélectionnez le Runtime que vous voulez supprimer.
2. Désinstallez le Runtime sélectionné.

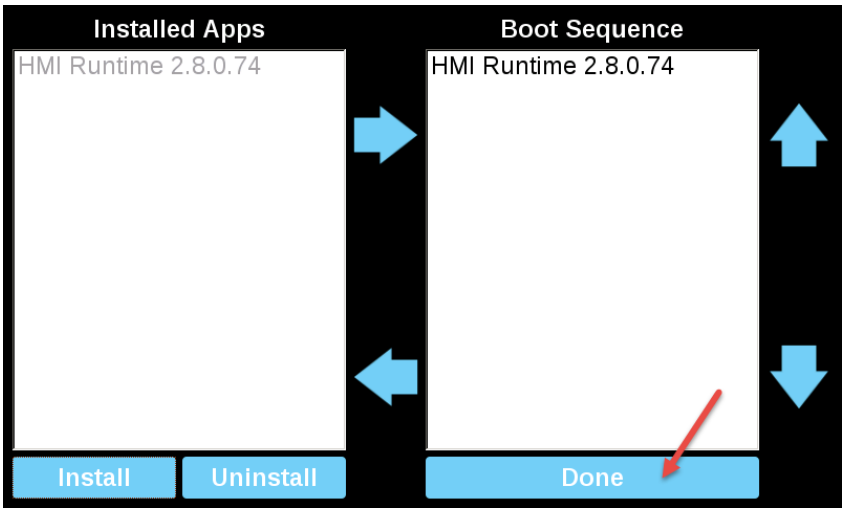
Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



Le processus de désinstallation au runtime sera effectué :



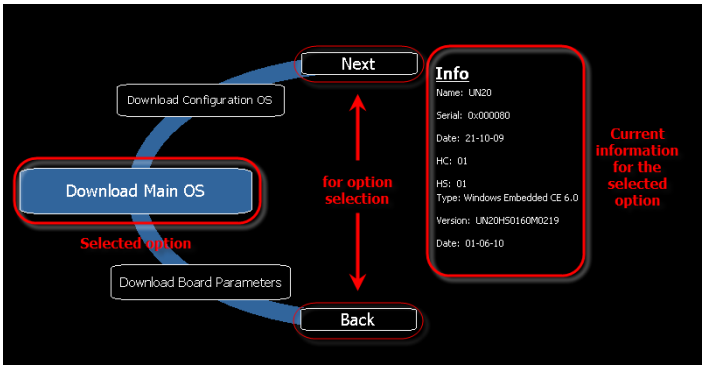
À la fin, appuyez sur le bouton « Done » (Terminé)



Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Périphériques WinCE

L'outil Paramètres système comprend un menu rotatif et des boutons de navigation pour faire défiler les options disponibles.



Pour chaque fonction et composant sur la gauche, le volet **Info** sur la droite affiche toutes les informations disponibles. Dans l'exemple, la version du composant Système d'exploitation principal est représentée.

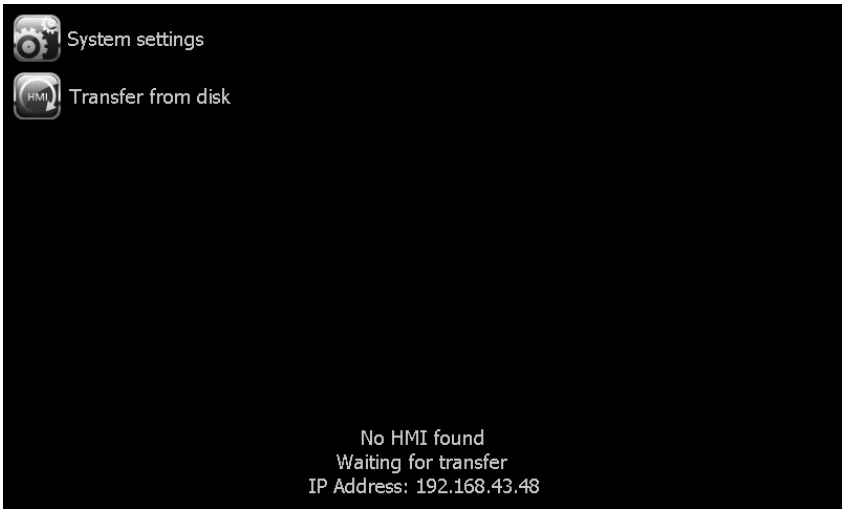
L'outil des paramètres du système peut être utilisé selon deux modes de fonctionnement :

- Mode utilisateur
- Mode système.

Pour chaque mode, différentes options sont disponibles.

Installation du runtime

Les périphériques HMI sont livrés d'usine sans Runtime, à la première mise sous tension, l'IHM affiche l'écran « Chargeur de runtime ».



Le runtime peut être installé :

- Automatiquement, via Ethernet lors du premier téléchargement de projet avec PB610 Panel Builder 600
- Manuellement via clé USB, création d'un « paquet de mise à jour »

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Installation du Runtime via Ethernet

Pour l'installation du Runtime via Ethernet, suivre la procédure "Télécharger vers le périphérique IHM" Page 88.

AVERTISSEMENT : le téléchargement via Ethernet de l'installation du runtime requière le périphérique IHM avec une adresse IP valide.

L'adresse IP peut être attribuée de trois manières :

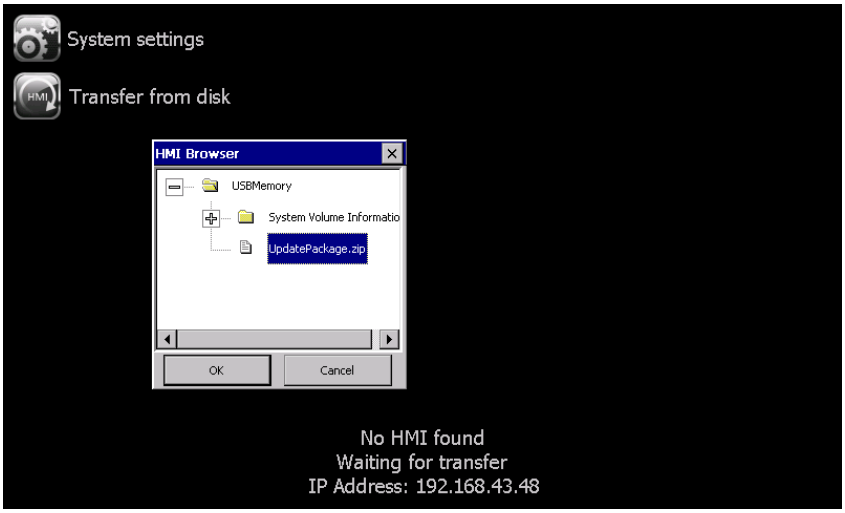
- *Automatiquement via serveur DHCP.* Cette option est activée par défaut. Si un serveur DHCP est disponible sur le réseau, l'adresse IP sera attribuée automatiquement par le serveur.
- *Automatiquement via la fonctionnalité Auto-IP.* Si l'assignation DHCP est activée mais qu'aucun serveur DHCP n'est disponible sur le réseau, le périphérique IHM s'attribue une adresse IP dans la plage 169.254.xx avec un masque de sous-réseau 255.255.0.0
- *Manuellement via les Paramètres système.* Dans le menu Paramètres du Système, dans la section réseau, l'adresse IP peut être attribuée manuellement, désactivant la fonction d'attribution du serveur DHCP.

Installation du runtime via une mémoire USB

Pour l'installation du runtime, UpdatePackage ou Backup Package via un périphérique USB, suivre cette procédure :

1. Créer un paquet de mise à jour depuis PB610 Panel Builder 600 et le copier sur une clé USB vierge

2. Sur le périphérique HMI, sélectionner [Transfer from disk] et sélectionner UpdatePackage.zip à charger.



Paramètres du système

Les paramètres du système ont deux modes de fonctionnement :

- Mode utilisateur
  une interface simplifiée qui permet aux utilisateurs d'accéder au paramétrage de base du périphérique IHM.
- Mode système
  une interface complète qui donne aux utilisateurs l'accès à toutes les options de l'outil.

Lorsque vous accédez à l'outil au runtime, sélectionnez « *Show System Settings* » dans le menu contextuel, l'outil est démarré par défaut en mode utilisateur.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

REMARQUE : appuyez longuement sur une zone de l'écran sans boutons ou autres éléments tactiles pour afficher le menu contextuel.

Pour accéder au Mode système :

- Exécutez une suite de tapotements sur l'écran tactile pendant la phase de mise sous tension. Une suite de tapotements et une suite à haute fréquence d'activations tactiles exécutées immédiatement après la mise sous tension du périphérique.
- Depuis la page Paramètres du système en mode utilisateur, redémarrez le panneau en mode configuration OS

Éléments disponibles en Mode utilisateur

Élément	Description
Calibrate Touch	Calibre l'écran tactile
Display settings	Contrôle la clarté et le délai d'attente d'inactivité du rétroéclairage
Time	Définit manuellement la date et l'heure sur le périphérique IHM ou configure les serveurs NTP.
Regional Settings	Sélectionne ou personnalise les paramètres régionaux
BSP Settings	Affiche la version du système d'exploitation et les minuteries de fonctionnement de l'unité pour contrôler le vibreur sonore et la DEL de la batterie.
Network	Définit l'adresse IP et d'autres paramètres réseau
Plug-in List	Liste des modules d'exécution installés et reconnus par le système. Remarque : cette option n'est pas prise en charge par toutes les plateformes ni par toutes les versions.
Close	Ferme la page de paramétrage du système
Certificat x.509	Vous pouvez personnaliser le certificat avec les données de votre entreprise et demander à une autorité de certification de le valider. La procédure pour personnaliser et valider votre certificat est : 1. Remplissez les paramètres nécessaires, puis appuyez sur la touche GENERATE pour générer un certificat auto-signé avec vos données. 2. Exporter la « Demande de certificat signé » 3. Envoyez la « Demande de certificat signée » à une autorité de certification pour validation (en général, il s'agit d'un service payant) 4. Importez le certificat signé dans le périphérique IHM **Paramètre du certificat**

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

<table>
<tr><th>Élément</th><th>Description</th></tr>
<tr><td></td><td>
<table>
<tr><th>Paramètre</th><th>Description</th></tr>
<tr><td>Device Name</td><td>Le nom de votre périphérique</td></tr>
<tr><td>Organization</td><td>Le nom légal de votre organisation</td></tr>
<tr><td>Unit</td><td>Le service de votre organisation qui traite le certificat</td></tr>
<tr><td>État</td><td>L'état/ la région où votre organisation est située</td></tr>
<tr><td>Location (Localité)</td><td>La ville où votre organisation est située</td></tr>
<tr><td>Country</td><td>Le code ISO à deux lettres du pays où se trouve votre organisation</td></tr>
<tr><td>Valid (days)</td><td>Validité du certificat</td></tr>
<tr><td>Key Length</td><td>Nombre de bits de la clé utilisée à partir de l'algorithme cryptographique</td></tr>
</table>
Les certificats gérés sont codés en base 64
 BSP v1.2.29 requis ou plus
</td></tr>
<tr><td>Restart</td><td>Redémarre le périphérique IHM
Système d'exploitation principal
Redémarrer le périphérique IHM dans le mode de fonctionnement
Système d'exploitation de configuration
Redémarrer le périphérique HIM avec l'outil Paramètre du système actif en Mode système</td></tr>
</table>

Éléments disponibles en Mode système

En plus de ceux disponibles en mode utilisateur, les fonctionnalités suivantes sont disponibles :

Élément	Description
Format Flash	Formate le disque flash du périphérique interne. Tous les projets et le HMI Runtime seront effacés, restituant le périphérique aux réglages d'usine.
Restore Factory Settings	Restaure les paramètres d'usine comme une alternative à la Formatation flash, d'une manière plus flexible. **Désinstaller IHM** : efface le HMI Runtime (tout le dossier qthmi) au prochain démarrage, le périphérique se comportera comme une nouvelle unité. Cette commande ne réinitialise pas les paramètres tels que l'adresse IP, la luminosité ou le RTC. **Effacer les paramètres de système** : réinitialise les paramètres du système (paramètres de registre) et supprime les fichiers suivants :

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Élément	Description
	*Flash**Documents and Settings**system.hv* *Flash**Documents and Settings**default**user.hv* *Flash**Documents and Settings**default.mky* *Flash**Documents and Settings**default.vol* Le mot de passe du Mode système est également réinitialisé. **Efface les paramètres sysdata** : efface le dossier \Flash\$SysData$ *Appel de service : à être utilisé uniquement par le support technique pour résoudre les problèmes d'affichage.* Remarque : toutes ces options ne sont pas disponibles pour tous les périphériques IHM et BSP.
Resize Image Area	Redimensionne la mémoire flash réservée pour stocker l'image de démarrage affiché à la mise sous tension. Les paramètres par défaut conviennent normalement à toutes les unités.
Download Configuration OS	Vérifie et met à niveau la version actuelle du système d'exploitation utilisé en mode système
Download Main OS	Vérifie et met à niveau la version actuelle du système d'exploitation principal
Download Splash Image	Charge un nouveau fichier pour l'image de démarrage affichée à l'unité au moment du démarrage. Conseil : mettre à jour l'image de démarrage directement à partir du logiciel de programmation PB610 Panel Builder 600. Voir "Mise à jour des composants système à partir de l'application" Page 524 pour plus de détails.
Download Bootloader	Vérifie et met à niveau la version actuelle du chargeur de démarrage du système.
Download Main FPGA	Vérifie et met à niveau la version actuelle du fichier FPGA principal. Cette fonction peut ne pas être disponible pour toutes les plateformes et versions.
Download Safe FPGA	Vérifie et met à niveau la version actuelle de la copie de sauvegarde du fichier FPGA. Cette fonction peut ne pas être disponible pour toutes les plateformes et versions.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

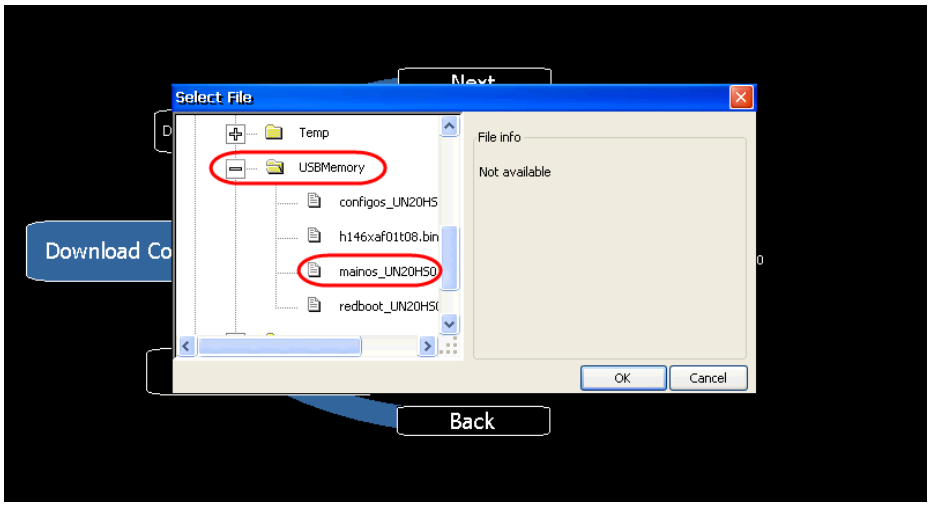
Élément	Description
Download System Supervisor	Vérifie et met à jour la version actuelle du micrologiciel du superviseur du système (utilisé pour la gestion du RTC et de l'alimentation électrique).
Upload Configuration OS **Upload Main OS** **Upload Splash Image** **Upload Bootloader** **Upload Main FPGA** **Upload Safe FPGA** **Upload System Supervisor**	Copier les fichiers système à partir du panneau de commande sur le périphérique externe (généralement une clé USB).

Mise à jour des composants du système

Les composants du système peuvent être mis à jour à l'aide de lecteurs flash USB. Pour chaque composant, un couple de fichiers de mise à jour spécifiques est fourni.

Remarque : les procédures de mise à niveau dépendent des versions du matériel et du système d'exploitation. Contactez le support technique pour de l'aide.

1. Copiez tous les fichiers de mise à niveau dont vous avez besoin sur un lecteur USB et branchez-le sur le port USB du périphérique IHM.
2. Démarrez l'outil des Paramètres du système en mode système (voir "Paramètres du système" Page 500pour les détails).
3. Cliquez sur la fonction de téléchargement souhaité.
4. Parcourez le contenu de la clé USB pour trouver les fichiers à télécharger. L'exemple montre les composants principaux du système d'exploitation.



Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

5. Cliquer sur **Télécharger** pour transférer les fichiers vers le périphérique IHM.

   Remarque : depuis cette boîte de dialogue, cliquez sur **Charger** pour transférer les fichiers vers le périphérique USB.

6. Suivez les instructions affichées pour terminer la mise à jour : l'état d'avancement de l'opération est affiché dans une barre de progression.

   Cette opération peut nécessiter quelques minutes.

Important : ne pas éteindre le périphérique alors qu'un composant du système est en cours de mise à niveau.

Liste des composants pouvant être mis à jour

Les périphériques IHM supportent la mise à jour des composants suivants :

Composant	Description
Application	L'application IHM et le runtime IHM générés à partir de la commande **Exécuterr>Paquet de mise à jour**
Système d'exploitation principal	Système d'exploitation principal
Système d'exploitation de configuration	Système d'exploitation de sauvegarde qui assure la restauration des unités en cas de corruption du système d'exploitation principal
Splash	L'écran initial affiché au démarrage du périphérique IHM
Bootloader	Chargeur pour gérer le démarrage du périphérique
Main FPGA	Microprogramme FPGA
Safe FPGA	Copie de sauvegarde du FPGA principal qui assure le démarrage de l'unité en cas de corruption du FPGA principal **Important : utilisez le même fichier pour mettre à jour les composants du FPGA Principal et de Sécurité.**
System Supervisor	Microprogramme du contrôleur du superviseur de système (par exemple : packaged_GekkoZigBee_v4.13.bin). *Le composant Superviseur de système peut être mis à niveau à partir de la v4.13 ou plus.* **Important : ne tentez pas de mettre à jour les versions V4.08, V4.09, V4.10 et V4.11 car elles ne prennent pas en charge la mise à jour automatique à partir des Paramètres Système.**

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Calibrage de l'écran tactile

Le calibrage du paramètre du système permet de calibrer le périphérique à écran tactile, peut être consulté depuis les paramètres du système

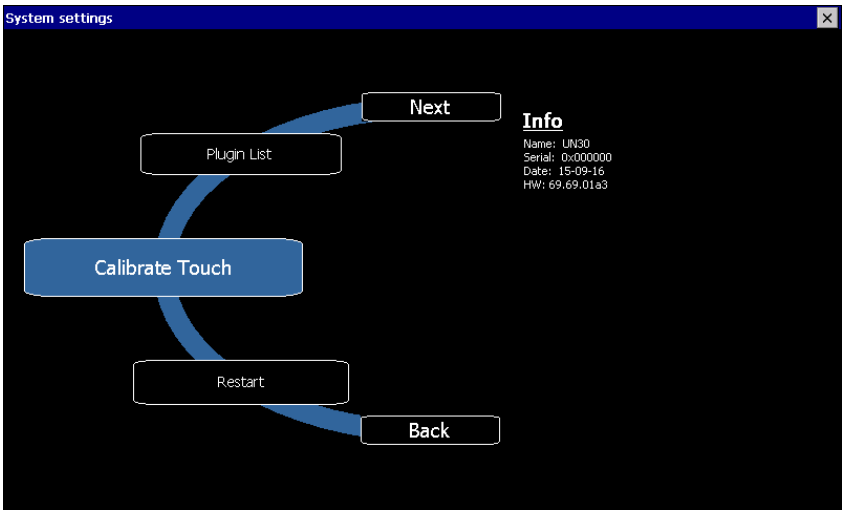
Pour accéder aux paramètres du système :

- Exécutez une suite de tapotements sur l'écran tactile pendant la phase de mise sous tension. Une suite de tapotements et une suite à haute fréquence d'activations tactiles exécutées immédiatement après la mise sous tension du périphérique.

ou

- Appuyer sur un espace vide de l'écran et maintenez enfoncé pendant quelques secondes pour afficher le menu contextuel.

Depuis le menu rotatif, sélectionner « *Calibrate Touch* » et suivre les instructions affichées sur l'écran pour terminer la procédure de calibrage, le système vous demandera de toucher des points spécifiques pour calibrer le périphérique à écran tactile.

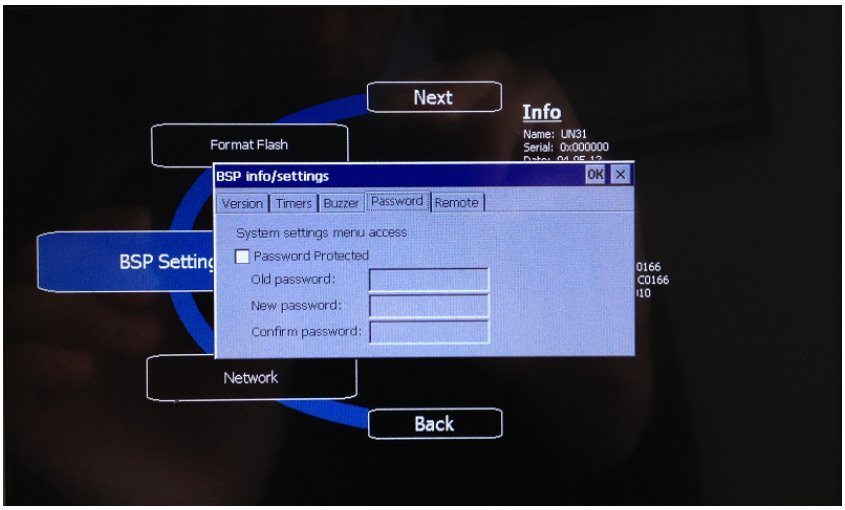


Protection par mot de passe

Le mot de passe interne du périphérique IHM peut être défini à partir des paramètres système en mode système (voir "Paramètres du système" Page 500 pour plus de détails)

Dans le menu de rotation, sélectionnez « *Paramètres BSP* », puis l'onglet Mot de passe pour ouvrir la boîte de dialogue Définir mot de passe.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



Le mot de passe doit avoir au moins 5 caractères.

Laissez « Ancien mot de passe » vide par défaut si le mot de passe cible n'est pas défini.

Cette fonctionnalité est disponible pour BSP à partir des versions V1.64 ARM UN30/31 et V2.73 MIPS UN20 basées sur WCE.

Rétablir la configuration d'usine

Si vous rencontrez des problèmes avec votre périphérique IHM, vous pouvez tenter de rétablir la configuration d'usine par défaut à partir du Mode système.

1. Passez en **Mode système.**
2. Utilisez l'une des opérations suivantes disponibles dans le menu tournant :
   - **Formater Flash**, pour nettoyer le périphérique flash et la configuration de registre.
   - **Rétablir la configuration d'usine**, pour nettoyer uniquement les composants sélectionnés.

Remarque : aucune de ces opérations ne concerne le rétablissement de la configuration d'usine pour les logiciels (MainOS, ConfigOS, Bootloader, FPGA, images, etc.).

Voir "Paramètres du système" Page 500pour plus de détails.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

40 Accès Web

PB4Web permet aux utilisateurs d'accéder aux projets IHM à partir d'un navigateur web distant exécuté sur un ordinateur ou sur un périphérique portable, tel qu'une tablette ou un téléphone. Avec PB4Web, les utilisateurs peuvent créer un projet web qui leur permet d'afficher sur un emplacement distant le même affichage graphique que celui apparaissant sur le périphérique IHM. Les projets PB4Web se basent sur une technologie HTML5, qui ne nécessite pas de plug-ins ni de logiciels externes pour afficher les informations.

Ce document suppose que vous avez une connaissance de base du mode de fonctionnement du navigateur web sur vos périphériques portables ainsi que de la manière d'établir une connexion au périphérique IHM qui exécute le serveur. Par exemple, si vous travaillez avec une tablette ou un téléphone, vous devez savoir comment configurer un accès Wi-Fi afin de pouvoir accéder aux pages PB4Web sur le périphérique IHM.

Plateformes et navigateurs pris en charge **510**

Créer une page pour accès Web **510**

Pages d'accueil de plateforme spécifiques **512**

Tester le projet Web **512**

Télécharger le projet Web **513**

Problèmes de connectivité Web **514**

Fonctionnalités web prises en charge **516**

Résolution des problèmes et FAQ **520**

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Plateformes et navigateurs pris en charge

PB4Web prend en charge 3 plateformes :

- Web, pour les navigateurs de bureau,
- téléphone, pour les périphériques multifonctions
- tablette, pour les périphériques tablettes

Vous pouvez donc créer des pages de contenu et de taille différentes pour les diverses plateformes. Par exemple, vous pouvez créer un ensemble de petites pages dans votre projet pour les téléphones alors que vous utiliserez des pages en pleine grandeur pour les navigateurs Web de bureau et de tablettes.

Travailler avec un ordinateur

PB4Web fonctionne avec tous les navigateurs Web modernes. Les navigateurs suivants ont été testés pour leur compatibilité avec PB4Web :

- Mozilla Firefox 50+
- Microsoft Internet Explorer 11+
- Apple Safari 10+
- Google Chrome 52+

Travailler avec des tablettes ou des téléphones

PB4Web fonctionne avec la plupart des tablettes et téléphones. Les tablettes suivantes ont été testées pour leur compatibilité avec PB4Web :

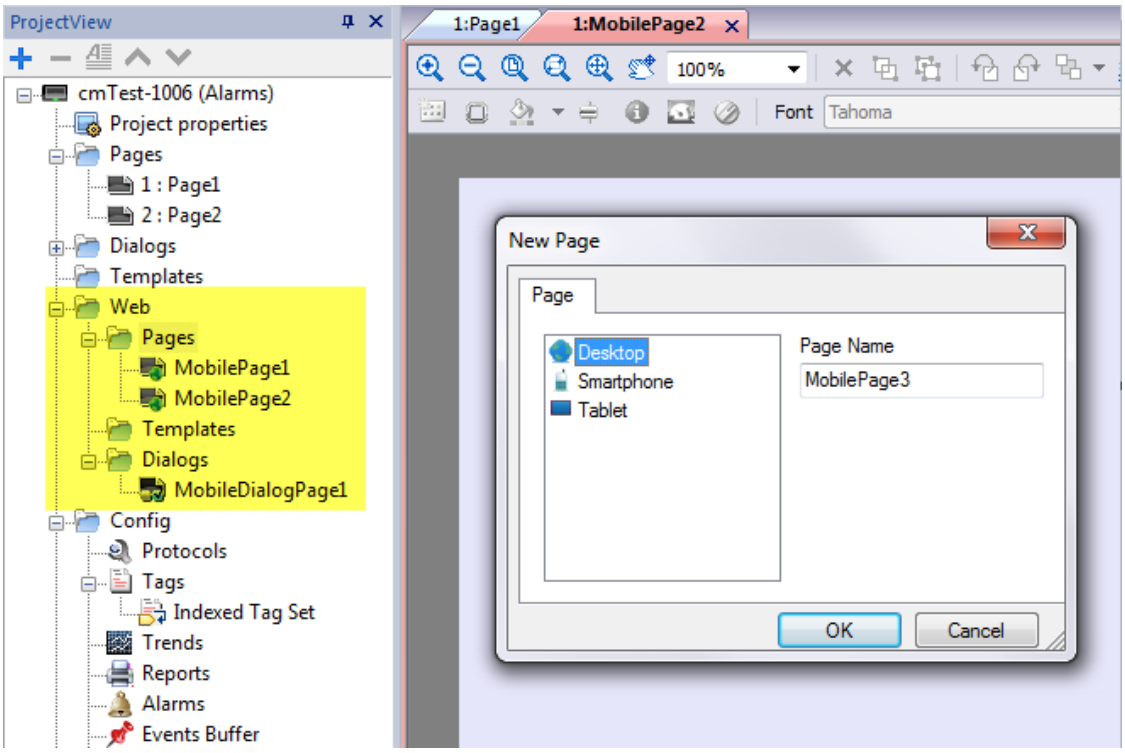
- iOS 10+ - Mobile Safari
- Android 7+ - Chrome pour Android 55+

Créer une page pour accès Web

VueProjet> Web > Pages

Cliquez avec le bouton droit sur le nœud des **Pages** et sélectionnez **Insérer une page** pour ajouter une page web.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



Tous les widgets et les fonctionnalités peuvent être utilisés dans PB610 Panel Builder 600 ; cependant, toutes les fonctionnalités ne sont pas actuellement disponibles dans PB4Web. Si le projet inclut une fonctionnalité qui n'est pas disponible, PB4Web fonctionnera quand même correctement, mais cette fonctionnalité ne sera pas disponible sur le périphérique client distant.

Voir "Fonctionnalités web prises en charge" Page 516 pour avoir la liste des fonctionnalités supportées dans PB4Web et des limites existantes.

Propriétés des pages Web

En plus des propriétés classiques de la page, il y a une propriété supplémentaire

Propriété	Description
Ajuster aux dimensions de l'écran	Comment la page sera adaptée à la fenêtre de visualisation du navigateur. • Aucun • Ajuster à l'écran Il suffit de modifier le niveau de zoom pour adapter la page à la fenêtre de visualisation du navigateur • Conception réactive Modifie intelligemment le niveau de zoom pour adapter la page à la fenêtre de visualisation du navigateur en respectant les restrictions définies à l'intérieur de la disposition en grille

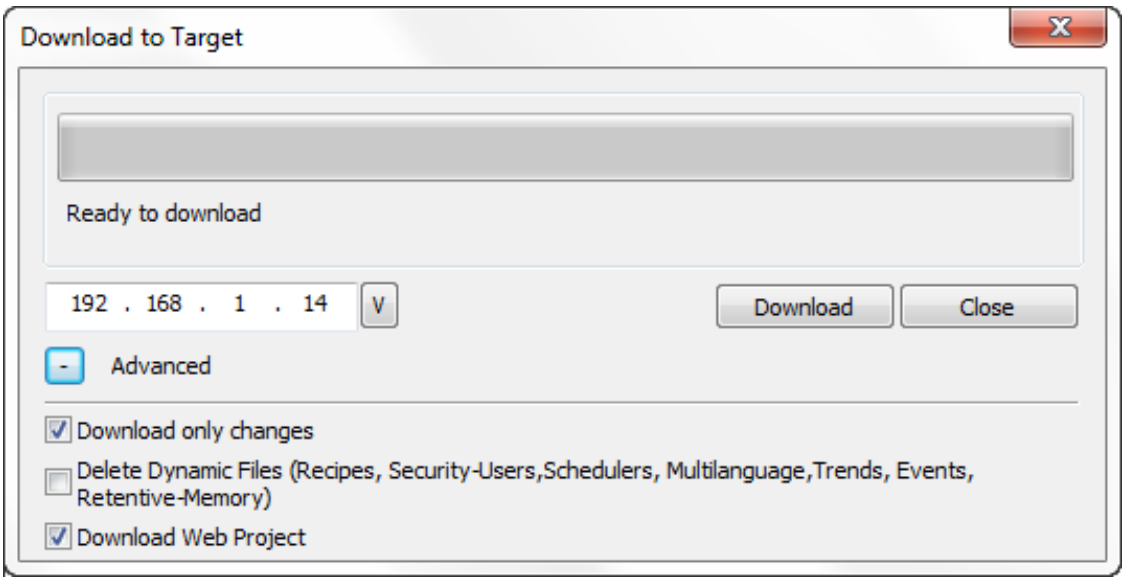
Exporter des pages

Pour sélectionner des pages à exporter à partir du projet actuel.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

1. Dans le menu **Exécuter**, cliquez sur **Paramètres de projet web** : la boîte de dialogue des **Paramètres de projet web** apparaîtra.
2. Sélectionnez les pages web que vous souhaitez exporter. Par défaut, toutes les pages de projet et de dialogue sont sélectionnées.
3. Pour chaque plateforme, sélectionnez l'icône d'accueil près de la page que vous souhaitez définir comme page d'accueil. Il n'est possible de sélectionner qu'une page d'accueil pour chaque plateforme.

AVERTISSEMENT : lorsque vous téléchargez un projet vers le périphérique IHM, assurez-vous que l'option Télécharger le projet Web est sélectionnée.



Pages d'accueil de plateforme spécifiques

La Page d'accueil du projet PB4Web correspond à la première page qui s'affiche dans le navigateur pour chaque type de plateforme et représente le point de départ pour votre projet Web. Les pages accessibles à partir de la page d'accueil dépendent dont la façon dont les autres pages sont liées dans le projet.

Par exemple, si vous avez conçu un ensemble de pages destiné à une plateforme téléphonique, définissez comme page d'accueil une page de taille appropriée à un écran de téléphone portable. Puis, n'incluez dans cette page que des liens vers d'autres pages de téléphone : l'utilisateur n'accèdera qu'aux pages de téléphonie lorsqu'il naviguera dans le projet PB4Web à partir d'un téléphone.

Tester le projet Web

Vous pouvez tester votre PB4Web projet en utilisant le simulateur en ligne ouvrant une page Web autonome directement à partir d'un navigateur.

Test avec le simulateur en ligne

PB610 Panel Builder 600 comprend un serveur Web dans le simulateur en ligne. Vous pouvez démarrer le simulateur et accéder à votre projet PB4Web à partir d'un navigateur Web. Les pages seront diffusées à partir du simulateur.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

1. Créer votre projet (voir "Créer une page pour accès Web" Page 510).
2. Sur le fichier **Exécution**, choisissez **Démarrage Simulateur** : le projet démarrera dans une fenêtre séparée.
3. Ouvrir un navigateur Internet (voir "Plateformes et navigateurs pris en charge" Page 510 pour une liste de navigateur compatible avec PB4Web).
4. Entrez l'adresse suivante : http://localhost:81 : cela indique au navigateur Internet de lire les pages Web à partir de l'ordinateur local et d'utiliser le port 81, utilisé par défaut par le simulateur en ligne PB4Web.
5. Testez votre projet dans le navigateur.

Important : si vous apportez des modifications aux pages de projets dans PB610 Panel Builder 600, vous devez arrêter et redémarrer le simulateur.

Remarque : si vous utilisez un périphérique (par exemple, un mobile multifonction) qui n'est pas l'hôte local où le simulateur est au runtime, il vous sera demandé de saisir le nom d'utilisateur et le mot de passe.

Télécharger le projet Web

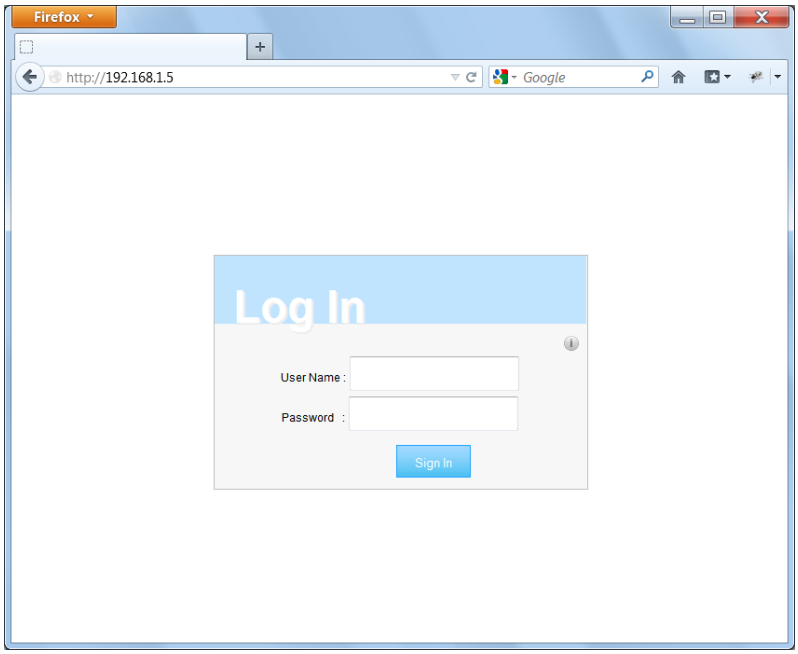
Après avoir testé les pages du PB4Web, vous pouvez télécharger le projet sur le périphérique IHM choisi.

Le projet PB4Web est téléchargé avec le projet PB610 Panel Builder 600, voir "Télécharger vers le périphérique IHM" Page 88 pour plus de détails.

Lorsque le processus de téléchargement est terminé, le projet IHM démarre automatiquement sur le périphérique IHM et le projet PB4Web est prêt à être utilisé.

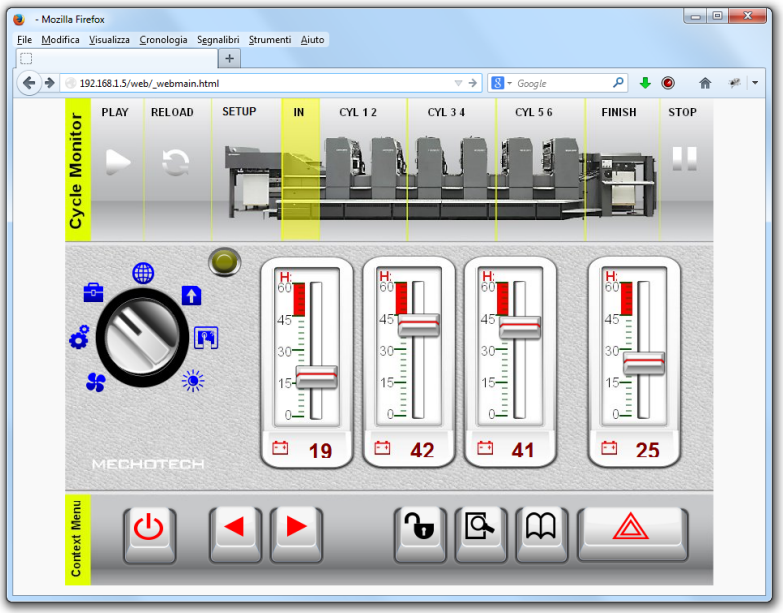
Exécuter PB4Web depuis un navigateur

1. Ouvrez un navigateur Web et entrez l'adresse IP de votre périphérique IHM : la page de connexion s'affiche.



Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

2. Entez le **Nom d'utilisateur** et le **Mot de passe**, puis cliquez sur **Se connecter** : la page d'accueil s'affiche.



Voir "Gestion et mots de passe utilisateur" Page 283 pour des détails sur la création d'informations d'identification.

Vous pouvez interagir avec le projet en utilisant le navigateur de la même manière que vous interagissez avec un périphérique lorsque vous touchez l'écran : cliquez sur les boutons pour changer de page, affichez les indicateurs et les jauges, déplacez les poignées de barre de défilement pour changer de valeurs, etc. Le projet PB4Web gère toutes les communications avec le serveur Web lorsque vous interagissez avec le périphérique IHM à distance.

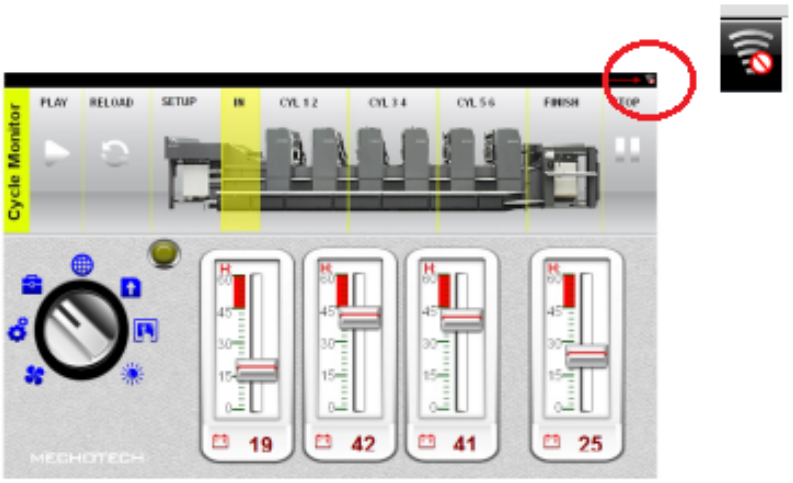
Problèmes de connectivité Web

Vous trouverez ci-dessous une description des problèmes les plus fréquemment rencontrés lors des connexions à distance avec le périphérique IHM.

Déconnexion du serveur

Comme PB4Web est exécuté à distance depuis le périphérique IHM, le serveur pourrait se déconnecter du navigateur (par exemple si le serveur est interrompu ou si le câble de réseau est débranché). Si cela se produit, une icône 'déconnexion' apparaîtra dans la barre d'outils en haut du PB4Web comme dans cet exemple.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



Dès que le serveur retourne en ligne, l'icône représentant un cercle rouge barré disparaît, signe que la communication avec le périphérique est redevenue normale.

La variable système « Statut de connexion » permet de connaître l'état de la connexion. Voir "Variables client à distance" Page 105 pour les détails).

Remarque : si vous faites des modifications dans les pages PB4Web pendant que le serveur est déconnecté, ces modifications seront visibles sur le client mais ne seront pas transmises au serveur tant que la connexion n'aura pas été rétablie.

Délai d'attente d'inactivité

PB4Web vous demandera de vous réidentifier si le navigateur est resté inactif pendant plusieurs minutes. Si aucune activité n'est détectée dans un délai de 10 minutes, l'écran d'ouverture de session apparaîtra et vous devrez indiquer vos informations d'identification avant de pouvoir poursuivre vos activités. Cette fonctionnalité de délai d'attente vous garantit contre tout accès non autorisé. Le délai d'attente d'inactivité web peut être modifié à partir du tableau **Propriété du projet**.

Fermeture de session utilisateur

Une session utilisateur peut être fermée soit par le serveur, soit pas l'utilisateur.

Dans des situations particulières, le serveur peut envoyer une demande au client (navigateur) d'effectuer la procédure de connexion. Dans ce cas, l'utilisateur est redirigé vers la page de connexion, puis directement vers la page sur laquelle il travaillait. C'est le cas par exemple lorsque l'utilisateur efface le cache ou les cookies du navigateur.

Remarque : si l'utilisateur travaillait dans une boîte de dialogue au moment d'être redirigé vers la page de connexion, il sera ensuite redirigé vers la page à partir de laquelle la boîte de dialogue avait été ouverte.

Projet PB4Web inactif

La page PB4Web qui s'affiche dans votre navigateur pourrait provenir d'un projet qui n'est plus actif dans le périphérique. Dans ce cas, une boîte de dialogue de confirmation s'affiche et vous pouvez retourner au projet actif.

Remarque : cette redirection suppose que le projet actif actuel comporte des pages PB4Web.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Si vous choisissez de rester dans le projet inactif, les actions que vous réalisez dans le navigateur pourraient ne pas être exécutées correctement, étant donné que le PB4Webne peut pas effectuer de communications liées au serveur.

Fonctionnalités web prises en charge

Actuellement, toutes les fonctionnalités PB610 Panel Builder 600 ne sont pas prises en charge dans PB4Web. Voici une liste des fonctionnalités supportées et les limites correspondantes, classées par catégorie.

Lorsque vous copiez et collez des objets depuis les pages standard vers une des pages Web, assurez-vous que tous les objets sont pris en charge dans les pages Web. Supprimez enfin les objets non pris en charge dans la page Web après le collage.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Catégorie	Fonctionnalités supportées	Limites
Widgets	• Basique (Texte/Numérique, Images, Formes, Tendances/Graphiques, Recettes, Contrôles, Alarmes, Texture, Blocage de fonction Javascript, Canevas générique disposition e-grille) • Boutons • Compteurs • Sélecteurs • Voyants • Médias (Caméra IP) • Icônes • Automatisation industrielle	• La fonction AttachToTag des variables du système n'est pas supportée • Les fichiers de police pour lesquels le drapeau de permission de téléchargement via Internet n'a pas été activé, ne sont pas chargés à partir de PB4Web • Les propriétés de widget avec le comportement dynamique Associer à... pourraient ne pas fonctionner pour toutes les propriétés supportées par PB610 Panel Builder 600. • Le multi-couche d'image multi-état n'est pas supporté. • La couleur d'alarme en fonction de la condition de déclenchement n'est pas supportée sur le web • Ne peut pas modifier les widgets d'alarme en runtime • Le mode de visualisation ComboBox en plein écran n'est pas pris en charge

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Catégorie	Fonctionnalités supportées	Limites
Alarmes	• Les limites d'alarme dans PB4Web sont les mêmes que pour le périphérique IHM (500..2000 selon la cible)	• Sur les smartphones/tablettes (périphériques généralement intégrés) en fonction du matériel, un utilisateur peut s'attendre à des problèmes de performance avec > 500 alarmes.
Actions	• Widgets (Javascript) • Page (PageAccueil, ChargerPage, Page Suiv, Page Préc, Dernière page visitée, Afficher Dialogue, Fermer dialogue, Afficher Message, Lancer Navigateur) • Multilangue (ConfigurerLangue) • Variable (ÉcrireVariable, VariablePas, Activer Bit, Désactiver Bit, BasculerBit) • Tendance/Graphique (RafraîchirTendance, DéfilementGaucheTendance, DéfilementDroitTendance, PageGaucheTendance, PageDroiteTendance, DéfilementTendanceVersLeHaut, DéfilementTendanceVersLeBas, TendanceHautPage, TendanceBasPage, DuréePageTendance, ZommerTendance, DézoomerTendance, RéinitialiserZoomTendance, ZommerTendanceAxe, DézoomerTendanceAxe, RéinitialiserZoomTendanceAxeX, MettreTendanceEnPause, Reprise tendance, AfficherCurseurTendance, DéfilerCurseurTendance, DéfilerTendance jusqu'à l'heure, DéfinirVueTendances) • Alarme (RéinitialiserAlarme, AcquitterAlarme, SélectionnerToutesIesAlarmes, ActiverAlarmes) • Système (ExporterTendance, SupprimerTendance, SupprimerArchiveÉvénements) • Recettes (Télécharger Recette, Chargement Recette, ÉcrireEnsembleRecettesActuel, TéléchargerRecetteAct, ChargerRecetteAct, RéinitialiserRecette, DumpRecipeData, RestoreRecipeData, AjouterEnsembleDonnéesRecette, SupprEnsembleDonnéesRecette) • Gestion des utilisateurs (Déconnexion et Changer le mot de passe)	• JavaScript est supporté (voir "JavaScript" Page 429chapitre JavaScript pour consulter la liste des fonctionnalités supportées) • Les actions de page ne sont pas supportées dans des conditions de déclenchement d'alarme • L'événement OnWheel (Molette) de page n'est pas pris en charge
XForms	• Formule • Mise à l'échelle • Indexation bit/octet	

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Catégorie	Fonctionnalités supportées	Limites
	• ColorPalette	
Claviers	• Seuls les widgets de claviers numériques sont supportés.	• Les claviers personnalisés ne sont pas pris en charge. Le clavier numérique s'affichera en tant que widget numérique en mode lecture/écriture ou écriture seule.
Page de dialogue	• Supportée, vous pouvez l'afficher et la fermer à l'aide des actions Afficher Dialogue et Fermer Dialogue.	• Les pages de dialogue ne supportent que les boîtes de dialogue modales.
Gestion des utilisateurs	• Le mécanisme de connexion vérifie les informations d'identification de l'utilisateur sur le serveur. L'identifiant et le mot de passe dépendent des informations d'identification définies dans "Gestion et mots de passe utilisateur" Page 283.	
Connexions d'utilisateurs simultanés	• Le serveur web du périphérique IHM supporte jusqu'à trois connexions simultanées.	• Si plus de 3 utilisateurs tentent de se connecter à partir de navigateurs distants, seuls les 3 premiers seront autorisés.

Secure Socket Layer (HTTPS)

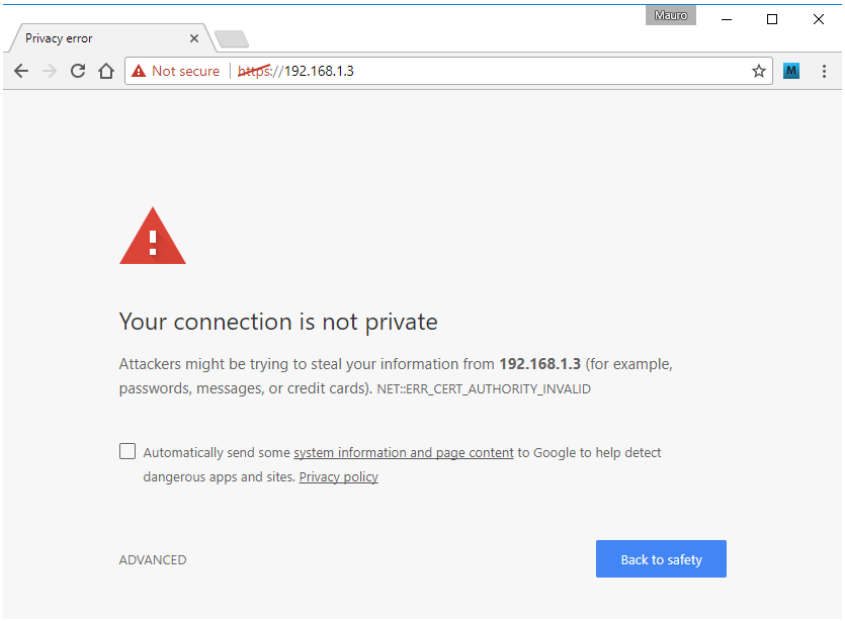
Les périphériques Linux prennent en charge le protocole de transfert sur Secure Socket Layer (HTTPS). Pour utiliser ce protocole, accédez à la page Web en utilisant la syntaxe ci-dessous :

https://<device_ip_address>

Notez que puisque l'auto-certificat fourni par l'IHM n'est pas confirmé par une autorité connue, vous obtiendrez un message d'avertissement.

Cliquez simplement sur le bouton AVANCER pour continuer.

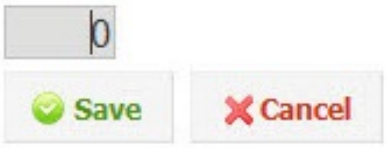
Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



Travailler avec des claviers dans PB4Web

L'utilisateur peut cliquer sur le widget numérique et une zone de texte apparaîtra dans laquelle la nouvelle valeur pourra être renseignée.

Une fois la valeur renseignée, l'utilisateur peut soit appuyer sur la touche **Entrée**, ou équivalent sur les périphériques tactiles, soit cliquer sur **Sauvegarder** pour rendre cette valeur permanente. Seuls les nombres pertinents seront acceptés lors du processus de sauvegarde. Les autres seront ignorés et n'entraîneront aucune modification de valeur.



Résolution des problèmes et FAQ

Activer JavaScript

PB4Web requiert JavaScript pour fournir l'interactivité avec le serveur et l'utilisateur. PB4Web ne fonctionnera pas si JavaScript est désactivé dans votre navigateur.

Par défaut, la plupart des navigateurs ont JavaScript activé. Si vous avez désactivé JavaScript auparavant, veuillez réactiver JavaScript avant d'accéder aux PB4Web pages.

Cache du navigateur

PB4Web comprend des ressources qui changent rarement telles que les fichiers CSS, les fichiers image et les fichiers JavaScript. Ces ressources prennent du temps à télécharger sur le réseau, ce qui augmente le temps nécessaire pour charger la page PB4Web dans votre navigateur. La mise en cache du navigateur permet à ces ressources d'être sauvegardées par un navigateur et utilisées sans effectuer la demande à chaque fois depuis le serveur. I en résulte un chargement plus rapide des pages PB4Web.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

La mise en cache est normalement activée par défaut, pour des performances optimales de PB4Web, assurez-vous qu'elle n'a pas été désactivée.

Remarque : les pages PB4Web fonctionneront toujours correctement avec la mise en cache désactivée du navigateur, mais le temps de chargement des ressources sera plus lent par rapport aux opérations normales mises en cache.

Utilisation d'un proxy

Certains utilisateurs pourraient accéder au projet PB4Web par proxy. Les proxys peuvent contrôler le nombre de connexion en parallèle du navigateur.

Assurez-vous que les connexions parallèles maximales autorisées (connexions max) sont entre 5 et 10.

Pourquoi je ne suis pas en mesure de voir les modifications dans les pages Web ?

Chaque fois qu'une nouvelle page Web est ajoutée au projet, vous devez télécharger le projet sur le périphérique. Toutefois, lorsque vous vous connectez à l'adresse IP du périphérique, le navigateur Web peut afficher des pages mises en cache au lieu des dernières pages téléchargées. Pour éviter ce comportement, vous pouvez :

- désactiver le cache de votre navigateur Web
- forcer le rafraîchissement de la page Web
- contourner le cache du navigateur

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

41 Mettre à jour les composants système des périphériques IHM

La plupart des composants logiciels du système peut être facilement mis à jour en assurant un degré élevé de flexibilité dans la fourniture de mises à jour et des correctifs aux systèmes existants et en cours d'exécution.

Des nouveaux modules logiciels peuvent être mis à jour

- Directement sur le périphérique IHM en utilisant un des lecteurs flash USB (voir "Paramètres du système" Page 477 pour plus de détails)
- Depuis l'application PB610 Panel Builder 600 (voir "Mise à jour des composants système à partir de l'application" Page suivante pour les détails)

Chaque périphérique IHM est étiqueté avec un code de produit, comprenant tous les paramètres d'usine (matériel, logiciels et composants de microprogramme). L'outil de mise à jour du périphérique IHM fournit également des détails sur les composants effectivement en cours d'exécution sur le périphérique.

MISE EN GARDE : S'assurer d'utiliser les fichiers de mise à niveau corrects, puisque le chargement de fichiers de mise à niveau inappropriés pour votre périphérique provoquera un dysfonctionnement grave du système. Toujours vérifier le code produit de votre périphérique.

Remarque : la mise à niveau de fichiers est fournie sur demande en tant une partie de l'activité du support technique.

Appel de service : les opérations de déclassement sont des tâches complexes qui pourraient causer de graves dommages à votre équipement si elles ne sont pas effectuées correctement. Ces opérations sont réservés au support technique.

Mise à jour des composants système à partir de l'application **524**

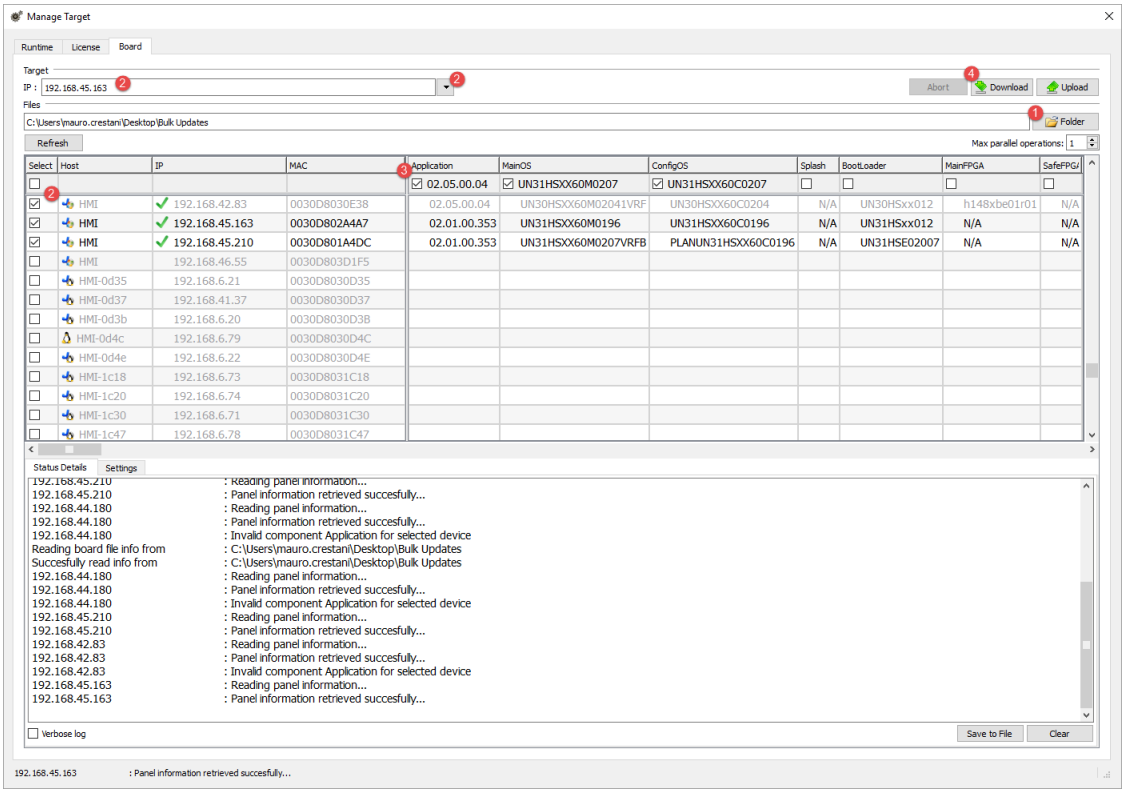
Paramètres **525**

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Mise à jour des composants système à partir de l'application

Vous pouvez télécharger les composants du système sur un périphérique IHM unique ou sur un grand nombre de périphériques IHM du même type en utilisant l'interface de communication Ethernet.

Chemin : ***Exécuter*** *>* ***Gérer cible*** *>* ***Carte***



1. Sélectionnez le dossier contenant les fichiers à télécharger sur le périphérique IHM ou l'endroit où téléverser les fichiers depuis le périphérique IHM.
2. Sélectionnez un ou plusieurs périphériques IHM.
3. Sélectionnez les composants que vous téléchargerez (ou chargerez) vers/depuis les dispositifs
4. Démarrez l'opération de téléchargement vers le périphérique IHM ou de téléversement depuis l'IHM

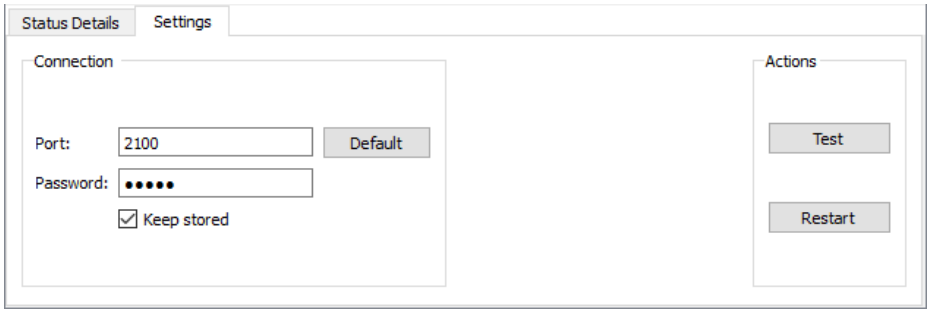
Remarque :

- L'outil est conçu pour mettre à jour un grand nombre de périphériques HMI du même type. Évitez de mettre des fichiers pour le type de périphérique différent dans le même dossier
- Si l'IP cible souhaitée n'est pas listée, tapez-la directement dans la boîte de dialogue. Le service Découverte est un service de diffusion. Quand une connexion à distance se fait via VPN ou depuis des réseaux externes, elle ne fonctionnera pas et vous devrez saisir l'adresse manuellement.
- Le téléchargement des composants sélectionnés s'effectuera uniquement vers des périphériques compatibles
- Sur la base de vos capacités de réseau et de matériel, vous pouvez augmenter le nombre de périphériques à mettre à jour simultanément.
- Vous devez redémarrer le périphérique IHM pour finaliser la mise à jour.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Paramètres

Depuis l'onglet **Paramètres**, vous pouvez spécifier le Port et les paramètres de Mot de passe à utiliser pour communiquer avec les périphériques IHM. Laissez un mot de passe vide si aucun mot de passe n'est défini côté périphérique IHM.



AVERTISSEMENT : le mode en grand nombre fonctionne uniquement avec les périphériques IHM ayant les mêmes paramètres de connexion

Chargement de l'image de démarrage

Vous pouvez remplacer l'image de démarrage par défaut affichée par les périphériques pendant la phase de mise sous tension.

L'image utilisée comme image de démarrage doit être conforme aux exigences suivantes :

Nom de fichier	splash.bmp
Format	Bitmap, RGB 565 format
Taille	< 500 KB
Largeur bitmap	Même des nombres (par exemple 430x239)

Pour charger l'image de démarrage :

1. Renommez la nouvelle image splash.bmp et copiez-la dans le dossier source.
2. Sélectionnez les périphériques IHM
3. Cliquez sur **Télécharger**.

Pour assurer les meilleurs résultats visuels, les images de démarrage doivent avoir un arrière-plan noir.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

42 Protéger l’accès aux périphériques IHM

Les opérations suivantes sont protégées par mot de passe sur le périphérique IHM :

- Gestion de HMI Runtime : installer HMI Runtime et mettre à jour HMI Runtime
- Gestion de carte : remplacer les principaux composants BSP tels que le système d'exploitation principal, la configuration du système d'exploitation, le bootloader, etc.
- Télécharger et charger des fichiers de projet
- Services en option sur les périphériques Linux (par ex. protocole SSH, Serveur VNC)

AVERTISSEMENT : L’accès non autorisé au périphérique peut causer des dommages ou défaillances. Lorsque vous connectez le périphérique à un réseau, protégez le réseau contre les accès non autorisés.

Voici une liste non exhaustive des mesures vous permettant de protéger le réseau :

- Pare-feu
- Système de prévention d'intrusion (IPS)
- Segmentation du réseau
- Réseau local virtuel (VLAN)
- Réseau privé virtuel (VPN)
- Sécurité au niveau de l’accès physique (sécurité de port).

Pour plus d’informations, des directives et des normes concernant la sécurité dans le domaine de l’informatique : IEC 62443, ISO/IEC 27001.

Changer de mot de passe sur le périphérique IHM **528**

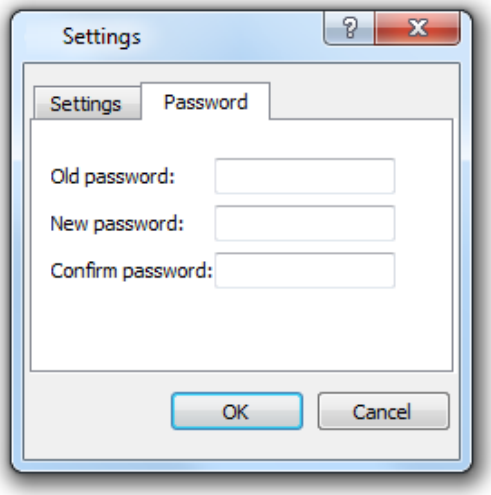
Ports et pare-feux **528**

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Changer de mot de passe sur le périphérique IHM

Pour changer de mot de passe sur le périphérique IHM, utilisez l'une des méthodes suivantes :

- Dans le HMI Runtime menu contextuel : **Paramètres**> onglet**Mot de passe**.



- Utilisez la fonction **Configurer mot de passe cible** dans le paquet de mise à jour : le mot de passe est mis à jour par HMI Runtime après la fin du processus de mise à jour.
- Utilisation de l'outil "Paramètres du système" Page 477 du périphérique IHM

Laisser « Ancien mot de passe » vide par défaut si le mot de passe cible n'est pas réglé.

Pour Win32 HMI Runtime, le mot de passe est sauvegardé dans *Users\[username]\AppData\Roaming\ABB\buildNumber\server\config\RemoteUpdateConfig.xml*.

Ports et pare-feux

Voici une liste de tous les ports utilisés par les composants PB610 Panel Builder 600.

Port	Utilisation	Accès à distance	Gestion de carte	Gestion de Runtime/Projet
80/tcp	Port HTTP	Oui	-	Oui
21/tcp	Port cmd FTP	-	-	Oui
2100/tcp	Port carte	-	Oui	-
16384-17407/tcp	Port de données FTP (mode passif)	-	Oui	Oui
990/udp	Diffusion UDP (Découverte de périphériques)	-	Optionnel	Optionnel

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Port	Utilisation	Accès à distance	Gestion de carte	Gestion de Runtime/Projet
991/udp	Diffusion UDP (Découverte de périphériques)	-	Optionnel	Optionnel
998/udp	Diffusion UDP (Découverte de périphériques)	-	Optionnel	Optionnel
999/udp	Diffusion UDP (Découverte périphérique)	-	Optionnel	Optionnel
5900/tcp	Serveur VNC	Seul VNC	-	-
5100/tcp	Débogueur à distance JS	-	-	Optionnel

Accès à distance

L’accès à distance est nécessaire pour se connecter aux HMI Runtime à l'aide de:

- HMI Client
- Navigateur Internet

Ports de gestion de runtime et projet

Vous utilisez ces ports pour vous connecter à HMI Runtime pour des opérations telles que la mise à jour, l'installation et le téléchargement de projet.

Ports de gestion de carte

Vous utilisez ces ports pour vous connecter au périphérique IHM pour les opérations de la carte telles que la mise à jour BSP, le téléchargement des images de démarrage, etc.

Remarque : Lorsque le service de diffusion n'est pas disponible, par exemple dans les réseaux VPN, tapez l'adresse IP exacte pour vous connecter au périphérique IHM du PB610 Panel Builder 600.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

43 Conseils et astuces pour améliorer la performance

PB610 Panel Builder 600 permet une grande flexibilité pour un concepteurs de projets.

Suivez ces instructions pour créer des projets qui fonctionnent mieux en termes de temps de démarrage, de changement de page et d'animations.

Optimisation statique **532**

FAQ sur l'optimisation statique **535**

Mise en cache de page **536**

Image DB **536**

Pré-cachage **536**

FAQ sur le pré-cachage **537**

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Optimisation statique

L'optimisation statique est une technique utilisée dans PB610 Panel Builder 600 pour améliorer les performances au runtime.

L'utilisation d'un grand nombre d'images et de photos dans un projet pourrait dégrader les performances, l'optimisation statique fusionne plusieurs images dans une image d'arrière plan unique, réduisant ainsi les temps de rendu et de chargement. En utilisant cette méthode, une seule image tramée doit être chargée et rendue au lieu de plusieurs images tramées et/ou d'images vectorielles.

Lorsque vous créez un projet dans PB610 Panel Builder 600, les pages peuvent contenir des widgets tels que textes, images, images d'arrière-plan, couleurs d'arrière-plan, etc. qui peuvent être classés comme suit :

- Statique : valeurs ou propriétés ne changent pas au runtime.
- Dynamique : valeurs ou propriétés changent au runtime.

Remarque : basées sur les paramètres de sécurité, les parties statiques des widgets peuvent ne pas être fusionnées avec l'arrière plan. Ceci se produit lorsqu'un widget est configuré sur « caché » dans les paramètres de sécurité.

Important : lorsque vous modifiez les propriétés des widgets avec JavaScript, définissez le widget d'optimisation statique sur dynamique, sinon les modifications des propriétés seront ignorées.

Lors du téléchargement ou de la validation un projet, PB610 Panel Builder 600 identifie les composants statiques et les rend sous forme d'images d'arrière-plan en fichiers .png. Ces images d'arrière-plan sont enregistrées en tant que partie du projet dans le dossier */opt*.

Les images d'arrière-plan peuvent être créés comme suit :

- images d'arrière-plan de pleine page, contenant tous les widgets fusionnés à la page d'arrière-plan
- groupez les images d'arrière-plan, contenant un groupe de widgets statiques fusionnés pour former un groupe d'arrière-plan. Par exemple, le groupe Jauge est normalement composé d'un fond, d'une échelle, d'une étiquette et d'une aiguille, où l'échelle et l'étiquette d'arrière-plan peuvent tous être fusionnés en une image d'arrière-plan unique.

L'attribut de la page **Optimisation statique** active et désactive l'optimisation statique de la page entière.

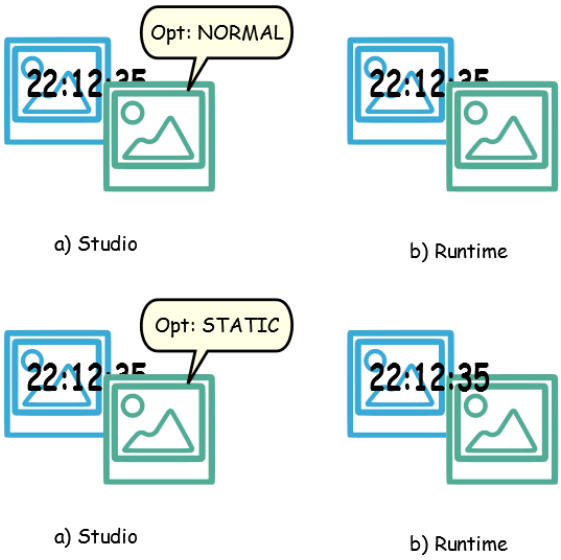
Un contrôle plus fin peut être obtenu en définissant l'attribut **Optimisation statique** de chaque widget individuel comme suit :

- **Normal** : PB610 Panel Builder 600 détecte automatiquement si le widget peut être fusionné avec l'arrière-plan. Ceci peut être utilisé si le widget n'est pas un widget dynamique et qu'il ne chevauche pas, c'est-à-dire qu'il n'est pas empilé au-dessus d'un widget dynamique.
- **Statique** : l'image est forcée d'être fusionnée avec l'arrière-plan. Ceci peut être utilisé lorsque le widget statique chevauche un widget transparent dynamique.

Remarque : dans ce cas, l'optimisation automatique échouera car elle ne fait aucune hypothèse sur les zones invisibles qui pourraient être rendus au runtime.

- **Dynamique** : le widget n'est pas optimisé du tout. Utilisez cette option quand un widget statique doit être changé par Javascript.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



Conseils pour une meilleure performance

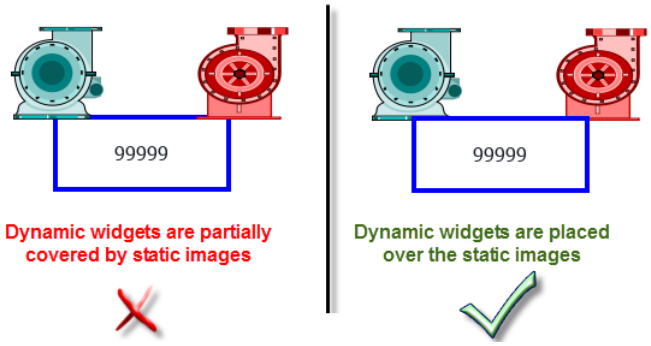
1. Tout d'abord : évitez de placer des widgets statiques sur un widget dynamique. La zone de chevauchement est calculée à partir des rectangles englobant les widgets, c'est-à-dire les rectangles délimités par la modification des poignées.
2. Ne pas utiliser l'optimisation statique si vos pages contiennent presque uniquement des objets dynamiques. L'optimisation statique permettrait d'économiser beaucoup d'images presque identiques en taille réelle pour chaque page, consommant beaucoup d'espace mémoire qui pourrait être utilisé plus efficacement pour améliorer les performances du projet avec d'autres techniques (comme, par exemple, la mise en cache des pages).
3. Les rectangles délimitants peuvent inclure des zones transparentes, minimisez les zones transparentes (par exemple divisez l'image en plusieurs images) car ceci peut représenter un gaspillage de ressources, même lorsque optimisées.
4. Optimisez la taille d'image. L'image sera rendue à la taille du widget contenant l'image. Pour de meilleures performances, le widget doit avoir la même taille que l'image.
5. Évitez d'utiliser **mettre à l'échelle pour convenir** pour les widgets d'images, car ceci force un redimensionnement au runtime pour les images dynamiques et « cache » la taille réelle de l'image lors de l'édition.
6. Utilisez **dimensionner pour convenir** pour rendre le widget à la taille réelle de son contenu.
7. Si le chevauchement ne peut être évité, assurez-vous de placer les widgets statiques derrière le widget dynamique.
8. Choisissez le format de fichier d'image basé sur le périphérique IHM auquel vous êtes connecté.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

9. Évitez d'utiliser trop de widgets sur une même page. Souvent, les widgets sont placés en dehors de la zone visible ou bien leur transparence est commandée par une variable. Alors que les widgets sont chargés, même s'ils ne sont pas visibles, avoir trop de widgets sur une page peut ralentir considérablement le temps de changement de page.
10. Divisez une page contenant de nombreux widgets en plusieurs pages avec moins de widgets.
11. Pour afficher de nouveaux éléments graphiques dans une page, préférez les pages de dialogue avec positionnement contrôlé aux widgets transparents.
12. Vérifiez le dossier *opt* pour voir si l'optimisation statique fonctionne comme prévu, les widgets ordre de plan peuvent avoir besoin d'être ajustés.
13. Les champs numériques sont souvent utilisés pour exécuter un code JavaScript sur l'événement OnDataUpdate même si le widget n'a pas besoin d'être visible sur la page. Dans ce cas, placez le widget en dehors de la zone de page visible au lieu de le rendre invisible, en modifiant la couleur de police ou la propriété visibilité. Dans ce dernier cas, vous pourriez vous retrouver avec de nombreux coins en trop.
14. Utilisez un bouton Zone sensible si vous avez besoin d'une zone tactile pour réagir aux entrées de l'utilisateur.
15. Si vous réutilisez un widget depuis la galerie ou que vous créez votre propre widget, rappelez-vous de définir les propriétés correctes d'optimisation. Par exemple, les widgets de boutons sont des widgets dynamiques, si vous utilisez un widget de bouton juste pour son cadre, il ne sera pas optimisé car le widget de bouton est dynamique. Si vous avez juste besoin du cadre, vous devez utiliser l'image Vers le haut.
16. Avec de nombreuses pages ayant de nombreux widgets dynamiques et en utilisant un modèle commun :
    1. définir l'optimisation statique de modèle sur **vrai,**
    2. définir l'optimisation statique de la page sur **faux**, puisque l'arrière-plan est déjà prévu par le modèle.

    Dans ce scénario, l'image d'arrière-plan peut être réutilisée sur de nombreuses pages différentes économisant ainsi de l'espace mémoire.
17. Ne pas utiliser des widgets dynamiques, tels que les boutons, uniquement à des fins graphiques, lorsque la fonction du bouton n'est pas nécessaire, utiliser plutôt les widgets image pour obtenir le même effet graphique.

Voici un exemple d'une bonne et d'une mauvaise utilisation d'optimisation statique.



Formats d'image pris en charge

PB610 Panel Builder 600 prend en charge de nombreux formats tramés tels que BMP, PNG, JPEG, TIFF et le format vectoriel SVG.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Format d'image	Avantages	Inconvénients
À TRAME	• Rendu rapide • Bien normalisé	• Grande taille de fichier • Résolution fixe
VECTEUR (SVG)	• Petite taille de fichier • Redimensionner sans perte de qualité • Peut gérer les propriétés dynamiques	• Les images SVG complexes avec des nombreux éléments graphiques et des couches peuvent être plus lents à rendre. • Créer un SVG optimisé n'est pas simple. • Seul Tiny 1.2 (http://www.w3.org/TR/SVGTiny12/) est supporté.

Remarque : le logiciel Scour est un outil gratuit qui peut être utilisé pour supprimer le code étranger d'un fichier (http://www.codedread.com/scour/).

Optimisation statique des modèles

Les pages de modèles peuvent avoir de grandes quantités de contenu statique. Cependant, l'optimisation statique ne peut pas être appliquée à un modèle de page, puisque l'endroit où le modèle est utilisé est basé sur la conception de la page.

Si une image d'arrière-plan énorme doit être répétée à chaque page utilisant le même modèle, ceci augmenterait l'empreinte du périphérique de sorte que la même image statique serait créée pour chacune des pages utilisant la page modèle.

FAQ sur l'optimisation statique

Q : dans une page où il y a des widgets identiques, dans le dossier *opt* , je vois un PNG pour chacun d'entre eux. S'ils sont vraiment identiques, pourquoi le logiciel les double au lieu d'avoir un seul PNG ?

R : le logiciel ne sait pas si les images statiques sont en fait les mêmes puisque chaque widget peut avoir différents paramètres/propriétés modifiant le rendu réel au runtime.

Q : pourquoi les images statiques sont-elles stockées dans un dossier séparé appelé *opt* au lieu de les stocker directement dans le dossier du projet ?

R : cela permet d'éviter des collisions de noms et permet de sauter le chargement des images d'optimisation

Q : pourquoi les images statiques sont stockées en tant que fichiers PNG au lieu de fichiers JPEG classiques ?

R : Le format PNG utilise une compression sans perte des images et prend en charge les transparences. Les fichiers JPEG seraient plus flous par rapport aux fichiers PNG avec un résultat différent dans PB610 Panel Builder 600(n'utilisant pas d'optimisation) et HMI Runtime.

Q : que se passera-t-il lorsqu'aucune optimisation n'est faite dans le logiciel ?

R : chaque widget est rendu au runtime. En particulier, les images SVG peuvent nécessiter beaucoup de temps à être rendues dans une plateforme embarquée.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Mise en cache de page

Une fois accédées, les pages sont conservées dans un cache RAM jusqu'à atteindre la taille maximale du cache, qui dépend de la RAM effectivement disponible de la plateforme. Cela permet un accès beaucoup plus rapide, car lorsque l'on charge à nouveau les pages mises en cache, seul leur contenu doit être retracé, sans qu'il soit nécessaire de charger à nouveau l'ensemble des ressources de la page.

Image DB

L'image DB est une technique utilisée pour tracer l'utilisation de fichiers d'image et pour réduire le coût de chargement des images moyennant la mise en cache des images les plus fréquemment utilisées (par exemple, les images Bouton-Poussoir, Aiguilles de calibre, Barres de défilement etc.). La même image, utilisée à plusieurs emplacements différents, n'est donc chargée qu'une seule fois.

L'image DB préchargera les images les plus utilisées au démarrage jusqu'à ce que la limite de mémoire soit atteinte. Cela permet d'écourter les délais de chargement de la page individuelle.

Le fichier imagecachelist.xml est créé dans le dossier *projet/opt*, et contient des informations utiles :

- Couleur de remplissage (en cas d'images SVG)
- Taille de l'image SVG
- Nombre de fois qu'une image est utilisée dans le projet
- Nombre de tailles différentes pour la même image

Astuces d'utilisation de la fonction Image DB

1. Dans la mesure du possible, utilisez une taille uniforme pour les boutons, les calibres et les autres widgets.
2. Utilisez des thèmes de même couleur parmi les widgets d'un même type.

Pré-cachage

L'attribut pré-cache des pages peut être utilisé pour notifier à HMI Runtime de précharger certaines pages dans la RAM au moment de l'initialisation pour un accès plus rapide. Le pré-cachage est utile pour les pages complexes ayant de nombreux widgets dynamiques.

Lorsque cette fonction est activée dans une page, l'accès à cette page est plus rapide, mais elle ralentit également l'initialisation, le système n'étant prêt qu'une fois que toutes les pages à pré-cacher ont été sauvegardées dans la RAM.

Astuces pour le pré-cachage

1. Activez la fonction pré-cache uniquement pour quelques pages ayant plusieurs widgets dynamiques ou pour des pages utilisées fréquemment par les utilisateurs.
2. N'activez pas la fonction précache pour toutes les pages du projet, vous risquez d'épuiser la mémoire et de n'obtenir aucun avantage.
3. Désactivez l'optimisation statique pour les pages où la fonction précache est activée afin de limiter l'utilisation de la mémoire.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

FAQ sur le pré-cachage

Limite de page au pré-cachage

En fonction de la taille et de la complexité d'une page, l'espace requis pour le pré-cachage peut aller de 1,5Mb à 3Mb.

Lorsqu'un projet est chargé, HMI Runtime procède de la façon suivante :

1. Les images de page sont préchargées jusqu'à ce que l'espace mémoire encore disponible atteigne 76 MB (imageDBLowMem)
2. Les pages où la mémoire pré-cache est configurée sur **vrai** sont préchargées jusqu'à ce que l'espace mémoire encore disponible atteigne 64 MB (pageCacheLowMemMax). Les images de ces pages sont chargées dans la RAM (dans l'Image DB).

Lorsque le projet est prêt :

1. Toute page nouvelle visitée est sauvegardée dans le cache (RAM) avec toutes les images associées jusqu'à ce que l'espace mémoire encore disponible atteigne 40 MB (pageCacheLowMemMin)
2. Lorsqu'un changement de page survient et que l'espace dans la RAM est critique (<40MB), le HMI Runtime commence à vider le cache (RAM) en supprimant des pages et les images associées jusqu'à ce que 64 MB d'espace mémoire soient rendus disponibles. HMI Runtime supprime les données stockées dans le cache dans l'ordre suivant :
    1. les dernières pages visitées, les images les plus grandes et celles inutilisées (>320x240),
    2. s'il faut libérer plus de mémoire, les pages en pré-cache ainsi que toutes les images chargées dans Image DB peuvent également être supprimées.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

44 Précisions fonctionnelles et compatibilité

Voici un aperçu des fonctions prises en charge et des limites connexes. Les limites indiquées ici sont des limites prudentes, au-delà desquelles le fonctionnement correct et la performance du système ne sont plus garantis.

Tableau de fonctions et limites **540**

Capacités des périphériques IHM **541**

Compatibilité **542**

Projets de conversion entre les différents périphériques IHM **542**

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Tableau de fonctions et limites

Fonction	Limite maximum
Nombre de pages	1,000
Nombre de widgets de base	2,000 x page
Nombre de variables	10,000
Nombre de pages de dialogue	50
Nombre de pages de dialogue qui peuvent être ouvertes en même temps	5
Nombre de recettes	32
Nombre d'ensembles de paramètres pour une recette	1,000
Nombre d'éléments par recette	1,000
Nombre de groupes d'utilisateurs	50
Nombre d'utilisateurs	50
Nombre de clients distants simultanés	4
Nombre de planificateurs	30
Nombre d'alarmes	2.000 (voir "Capacités des périphériques IHM" Page opposée)
Nombre de transferts de données	1000
Nombre de pages de modèles	50
Nombre d'actions programmables par état de bouton	32
Nombre de mémoires des tendances	30
Nombre de courbes par widget de tendance	5
Nombre de courbes par widget de diagramme de dispersion	10
Nombre d'échantillons par mémoire des tendances	200,000
Nombre de variables par mémoire des tendances	200
Nombre d'échantillons de mémoire des tendances pour un projet	1.200.000 (voir "Capacités des périphériques IHM" Page opposée)
Nombre de messages dans un champ de message	1024
Nombre de langues	12
Nombre d'événements par mémoire tampon	2,048

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Fonction	Limite maximum
Nombre d'événements par mémoire des tendances	4
Taille de fichier JavaScript par page	16 KB
Taille de projet sur disque	60 Mo (voir "Capacités des périphériques IHM" bas)
Nombre d'instances indexées	100
Nombre d'alias indexés	100
Nombre d'ensembles de variables indexés	30
Nombre de protocoles physiques	4
Nombre de rapports	32
Nombre de pages de rapports	32
Nombre max. de variables dans le widget de variables	255
Taille du dossier utilisateur (UpdatePackage.zip)	5 MB
Nombre de sessions FTP simultanées	4
Dossiers FTP supplémentaires	5

Capacités des périphériques IHM

Voir "Tableau de fonctions et limites" Page précédente pour les capacités classiques.

Panel	Device OS	Touch	Media Player	PDF	Max Project Size	Dialogs	Alarms	Protocols	JavaScript	Reports	Trend Buffers	Trend Curves	Events Buffers	User Folder Size
Runtime PC	Win32		MPEG4	Yes	60 MB	200	10.000	8	64 KB	64	2.400.000	10	10 K	na
CP620	WEC		na	na	30 MB	50	500	4	16 KB	32	1.200.000	5	2 K	10 MB
CP630	WEC		na	na	30 MB	50	500	4	16 KB	32	1.200.000	5	2 K	10 MB
CP635	WEC		na	na	30 MB	50	500	4	16 KB	32	1.200.000	5	2 K	10 MB
CP651	WEC		MPEG4/H264	Yes	60 MB	50	2000	4	16 KB	32	1.200.000	5	2 K	100 MB
CP661	WEC		MPEG4/H264	Yes	60 MB	50	2000	4	16 KB	32	1.200.000	5	2 K	100 MB
CP665	WEC		MPEG4/H264	Yes	60 MB	50	2000	4	16 KB	32	1.200.000	5	2 K	100 MB
CP676	WEC		MPEG4/H264	Yes	60 MB	50	2000	4	16 KB	32	1.200.000	5	2 K	100 MB
CP635-Fx	WEC		MPEG4/H264	Yes	60 MB	50	2000	4	16 KB	32	1.200.000	5	2 K	100 MB
CP604	Linux		na	Yes	60 MB	50	500	4	64 KB	32	1.200.000	5	2 K	100 MB
CP607	Linux		na	Yes	60 MB	50	500	4	64 KB	32	1.200.000	5	2 K	100 MB
CP610	Linux		MPEG4	Yes	60 MB	50	2000	4	64 KB	32	1.200.000	5	10 K	100 MB
CP6607	Linux	Multi	MPEG4/H264	Yes	240 MB	200	4.000	8	64 KB	64	2.400.000	10	10 K	100 MB
CP6610	Linux	Multi	MPEG4/H264	Yes	240 MB	200	4.000	8	64 KB	64	2.400.000	10	10 K	100 MB
CP6615	Linux	Multi	MPEG4/H264	Yes	240 MB	200	4.000	8	64 KB	64	2.400.000	10	10 K	100 MB
CP6621	Linux	Multi	MPEG4/H264	Yes	240 MB	200	4.000	8	64 KB	64	2.400.000	10	10 K	100 MB
CP6605	Linux	Multi	MPEG4	Yes	60 MB	50	2000	4	64 KB	32	1.200.000	5	10 K	100 MB

Fonctions non disponibles dans les périphériques Linux :

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

- LaunchBrowser macro
- Les périphériques d'impression ne sont pas pris en charge. Les rapports ne peuvent être imprimés que sur des fichiers PDF. L'impression de rapports textuels et d'événements d'alarme n'est pas prise en charge.

Fonction non disponible sur PB610 PC Runtime :

- Plug-in VNC et PDF Readers
- Gérer cible
- Outil de réglage du système
- Sauvegarder/Restaurer
- Protocoles de série qui requièrent un matériel spécial.
- Les fonctions tactiles multipoints seront disponibles si elles sont prises en charge par le matériel et le système d'exploitation de l'ordinateur.

Compatibilité

La politique de compatibilité suivante a été adoptée :

- La version PB610 Panel Builder 600 doit toujours être alignée avec HMI Runtime sur le périphérique,
- l'utilisateur est responsable de la mise à jour des composants de HMI Runtime sur le périphérique IHM à chaque mise à jour de PB610 Panel Builder 600,
- la mise à jour de HMI Runtime peut être réalisée directement à partir de PB610 Panel Builder 600 avec la commande Mise à jour de la cible, qui est disponible dans la boîte de dialogue Exécuter\Gérer cible,
- les projets créés dans une version de PB610 Panel Builder 600 postérieure à la version V1.00 (00) peuvent être ouverts et gérés par les nouvelles versions,
- les projets qui sont créés avec des versions antérieures de PB610 Panel Builder 600, ouverts avec des versions postérieures et déployés sur HMI Runtime compatible, maintiennent leurs performances et leurs fonctionnalités,
- la compatibilité n'est pas assurée entre les versions les plus récentes de HMI Runtime et les projets créés et déployés avec des versions moins récentes de PB610 Panel Builder 600.

Important : n'éditez pas de projets avec une version de PB610 Panel Builder 600 moins récente que celle utilisée pour les créer.Cela pourrait endommager le projet et rendre HMI Runtime instable.

Projets de conversion entre les différents périphériques IHM

La conversion de projet à partir de différents modèles de périphériques IHM est prise en charge, cependant, certaines opérations manuelles peuvent être nécessaires si le projet utilise des fonctions non supportées dans le périphérique de destination.

Lignes directrices

Avant de convertir un projet, vérifiez s'il existe des fonctions non supportées (voir "Capacités des périphériques IHM" Page précédente), et ajustez votre projet en supprimant les fonctionnalités non prises en charge avant de convertir le projet.

En particulier :

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

- Vérifiez les limitations et les fonctionnalités non prises en charge par le nouveau périphérique IHM (voir "Tableau de fonctions et limites" Page 540 pour plus de détails).
- Supprimez les widgets, les actions, les variables système, les protocoles et les propriétés de projet qui ne sont pas pris en charge.
- Si le projet utilise un stockage externe, vérifier si le même chemin de stockage est toujours disponible.
- Ajustez les applications ou les chemins externes spécifiques à un système d'exploration.
- Si besoin, réduire la taille du projet selon les nouvelles limitations de type de périphérique IHM (voir « Limitations » pour plus de détails).
- Puisque les périphériques IHM sont basés sur différentes plateformes matérielles avec différente vitesse de processeur, taille de la mémoire RAM, taille du cache, assurez-vous de vérifier le temps de démarrage du projet et le temps de chargement de la page pour chaque page dans le projet.
- Vérifiez le code JavaScript pour les opérations spécifiques au système d'exploitation.

Fonctionnalités spécifiques au système d'exploitation

Linux est sensible à la casse alors que Windows CE ne l'est pas. Par conséquent, les projets sur les périphériques IHM Linux peuvent avoir différents fichiers nommés en fonction des majuscules et minuscules, par exemple « dump1.csv » et « Dump1.csv », ce qui n’est pas possible sur les périphériques IHM Windows CE.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

45 Protocoles de communication

Cette section décrit les protocoles disponibles.

Remarque : Il est possible que le matériel ou les protocoles du contrôleur aient été modifiés depuis la création de cette documentation. Pour intégrer les évolutions du matériel et des protocoles du contrôleur, les pilotes font l'objet d'une mise à jour continue. Pour cette raison, veuillez vous assurer que l'application utilise le pilote le plus récent.

Il est possible d'utiliser divers médias physiques, passerelles, routeurs et concentrateurs dans le réseau de communication. De plus, les autres périphériques peuvent utiliser le réseau de manière autonome et simultanée. Toutefois, il est important de garantir que le trafic généré par ces périphériques n'amène pas de dégradation inacceptable de la vitesse de connexion (temps aller-retour).

Ethernet ABB CoDeSys **546**

ABB CODESYS Serial **556**

ABB IRC5 ..... **563**

ABB Mint Controller HCP **568**

ABB Modbus RTU **577**

ABB Modbus TCP **584**

ABB Pluto **590**

BACnet **596**

CODESYS V2 ETH **625**

CODESYS V3 ETH **638**

Ethernet/IP CIP **651**

Modbus RTU **677**

Modbus RTU Server **693**

TCP Modbus **708**

Serveur Modbus TCP **725**

OPC UA Client **737**

Simatic S7 ETH **751**

Variables système **791**

Variables **793**

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Ethernet ABB CoDeSys

Le pilote de communication ABB CODESYS pour Ethernet est spécialement conçu pour prendre en charge la communication avec des contrôleurs ABB de la série AC500 en programmation standardisée CEI 61131-3, qui se base sur le système CODESYS V2.3.

Remarque : le pilote Ethernet CODESYS est pris en charge à partir de la version AC500 FW 2.1 et non pas des versions AC500 FW 1.xx

Remarque : pour intégrer les évolutions du matériel et des protocoles du contrôleur, les pilotes font l'objet d'une mise à jour continue.

Remarque : Il est possible que le matériel ou les protocoles du contrôleur aient été modifiés depuis la création de cette documentation. Ceci peut interférer avec la fonctionnalité de ce pilote. Par conséquent, testez et vérifiez toujours la fonctionnalité de l'application.

Limites

Le niveau 4 de CODESYS n'est pas pris en charge. La taille maximale d'un bloc est 1024.

Paramètres de l’éditeur de protocole

Ajouter un protocole

Pour configurer le protocole :

1. Dans le nœud **Config**, double-cliquez sur **Protocoles**.
2. Pour ajouter un pilote, cliquez sur **+** : une nouvelle ligne est ajoutée.
3. Sélectionnez le protocole dans la liste **API**.

La boîte de dialogue de configuration du protocole s’affiche.

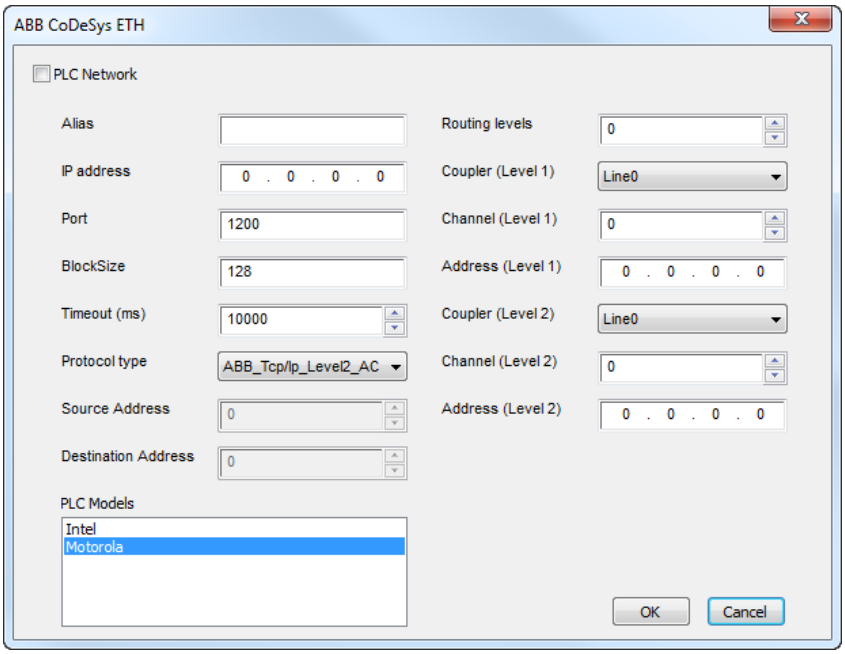
Ajoutez un pilote dans l'éditeur de protocoles et sélectionnez **ABB CODESYS ETH** dans la liste des protocoles disponibles.

Les types de protocoles suivants sont pris en charge :

- Route TCP/IP de niveau 2
- Route TCP/IP ABB de niveau 2 AC
- TCP/IP

Sélectionnez le type de protocole dans la liste déroulante **Type de protocole** du dialogue **ABB CODESYS ETH**.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

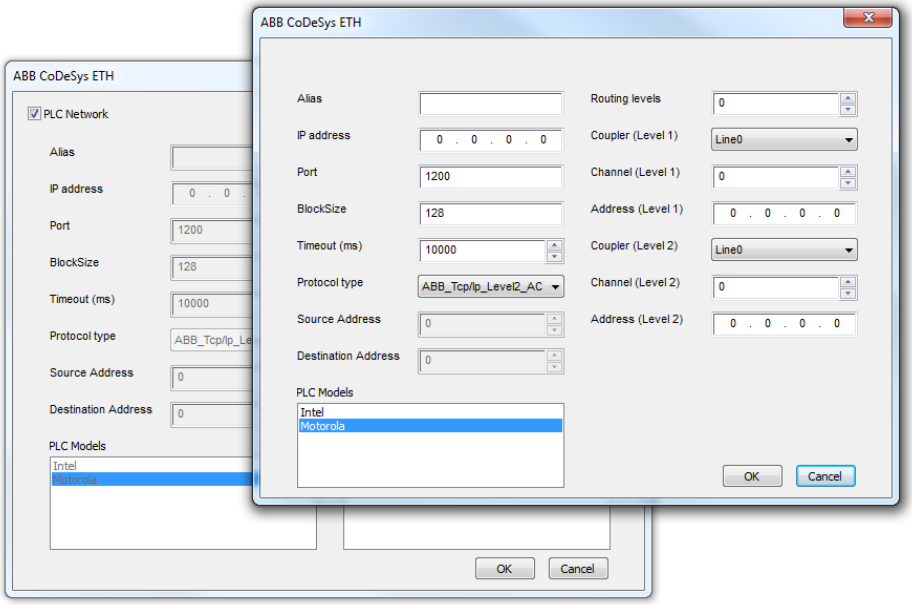


Certains paramètres de ce dialogue sont disponibles pour différents types de protocoles, d'autres sont spécifiques.

Les paramètres disponibles pour différents types de protocoles sont les suivants :

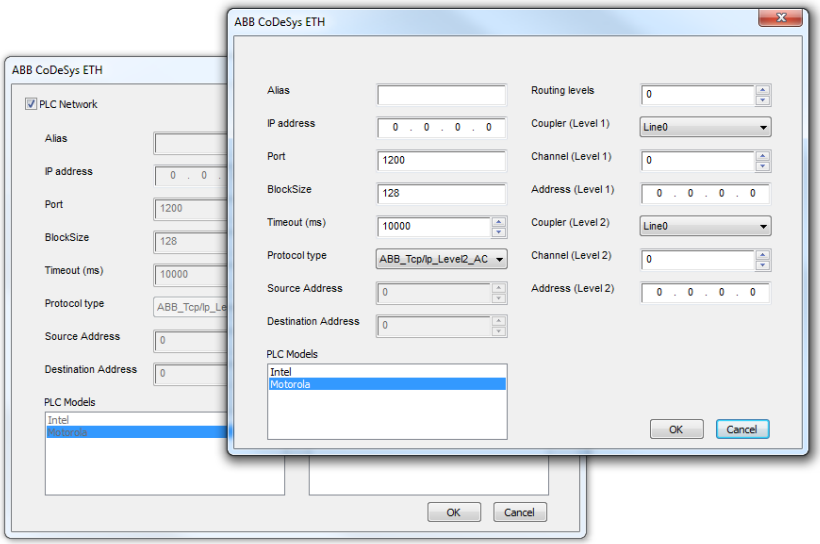
Élément	Description
Alias	Nom d'identification des nœuds dans les configurations réseau. Il est ajouté en préfixe à chaque nom de variable importé pour chaque nœud de réseau.
Adresse IP	Adresse IP Ethernet du contrôleur
Port	La valeur par défaut pour les pilotes ABB est 1200. Pour les pilotes AC500 et 3S, sélectionnez le numéro de port 1201.
Taille de bloc	Taille de bloc maximale prise en charge par votre contrôleur
Délai d'attente	Délai d'attente (en millisecondes) entre deux tentatives en cas d'absence de réponse du périphérique esclave.
Modèle d'API	Format d'octets qui sera utilisé par le pilote de communication en envoyant des trames de communication à l'API : pour Intel, il s'appelle généralement « little-endian », pour Motorola, on parle de « big-endian » Sélectionnez « Motorola » pour AC500.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Élément	Description
Type de protocole	Trois types de protocoles différents sont disponibles : • **TCP/IP** • **Route TCP/IP de niveau 2** • **TCP/IP ABB de niveau 2 AC**
Réseau d'API	Le protocole permet la connexion de contrôleurs multiples. Pour configurer les connexions multiples, cochez la case « Réseau d'API » et entrez une adresse IP pour chaque esclave auquel vous souhaitez accéder. 

Types de protocoles

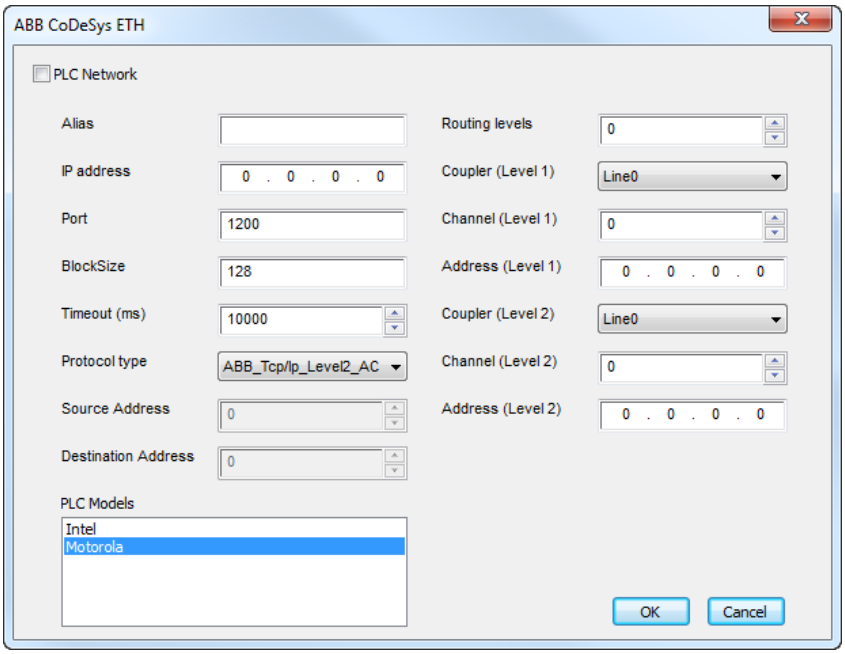
Le type de protocole **TCP/IP** correspond au pilote 3S de niveau 4 et ne requiert pas de configuration supplémentaire à part les paramètres partagés.



Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Le type de protocole **Route TCP/IP de niveau 2** correspond au pilote de Route 3S de niveau 2 standard et requiert deux paramètres supplémentaires :

Paramètre	Description
Adresse source (SrcAdr), adresse de destination	La destination est le nœud de l'API qui permet au protocole de lire les variables d'un sous-réseau. L'adresse est utilisée pour la lecture de variables lorsque plusieurs API sont connectés à un sous-réseau (réseau en série) mais qu'un seul est doté d'interface Ethernet. *Ce paramètre ne s'applique actuellement pas pour les API AC500.*



Le type de protocole **TCP/IP ABB de niveau 2 AC** met en œuvre une variation spécifique du protocole standard de niveau 2 en utilisant un pilote routeur supplémentaire. Ce type de protocole est généralement utilisé pour se connecter aux API via d'autres API faisant office de passerelles.

Ce type de protocole requiert les paramètres supplémentaires suivants :

- Niveaux de routage
- Coupleur (niveau 1)
- Canal (niveau 1)
- Adresse (niveau 1)
- Coupleur (niveau 2)
- Canal (niveau 2)
- Adresse (niveau 2)

Pour plus d'informations, voir la documentation *AC500 et Control Builder*, chapitre *Interfaces de programmation de l'AC500 utilisé par Control Builder*.

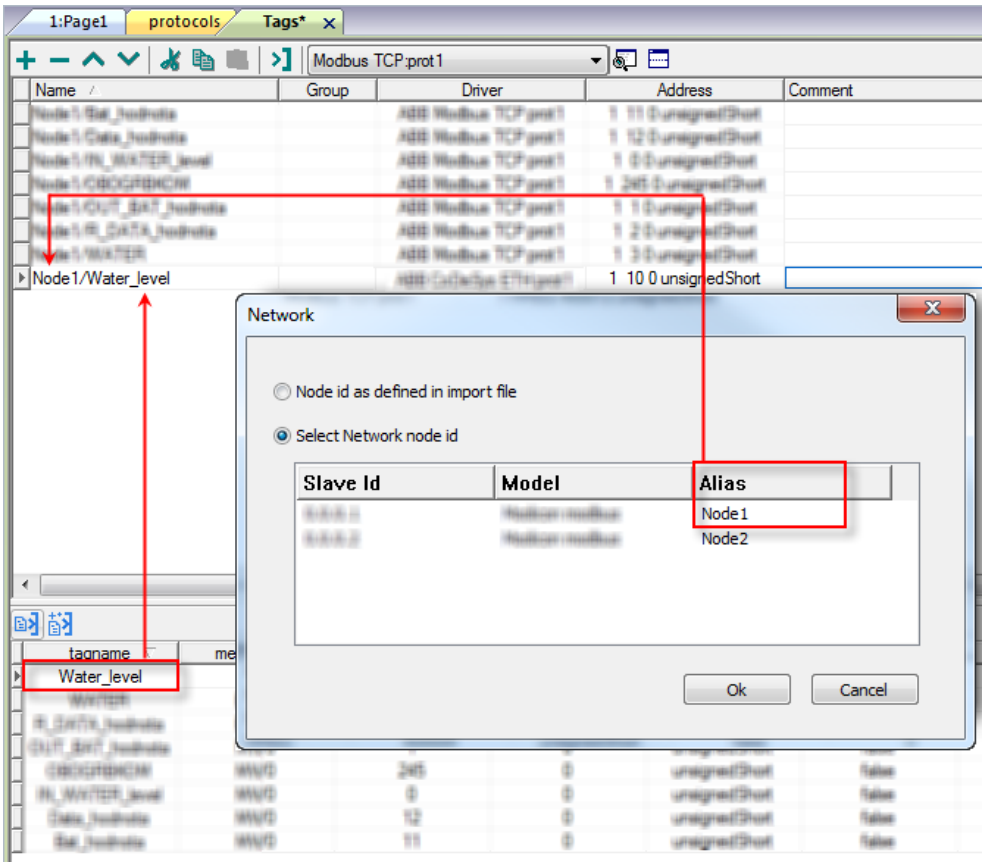
Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Ajouter un nom d'alias à un protocole

Les noms de variables doivent être uniques au niveau d'un projet. Vous pouvez cependant être amené à utiliser les mêmes noms de variables pour différents nœuds de contrôleur (par exemple lorsque le périphérique IHM est connecté à deux périphériques exécutant la même application).

Lorsque vous générez un protocole, vous pouvez ajouter un nom d'alias qui s'ajoutera aux noms de variables importés pour ce protocole.

Dans l'exemple, la connexion à un certain contrôleur porte le nom **Nœud1**. Lorsque vous importez les variables pour ce nœud, tous les noms de variables porteront le préfixe **Nœud1**, les rendant ainsi uniques au niveau projet/réseau.

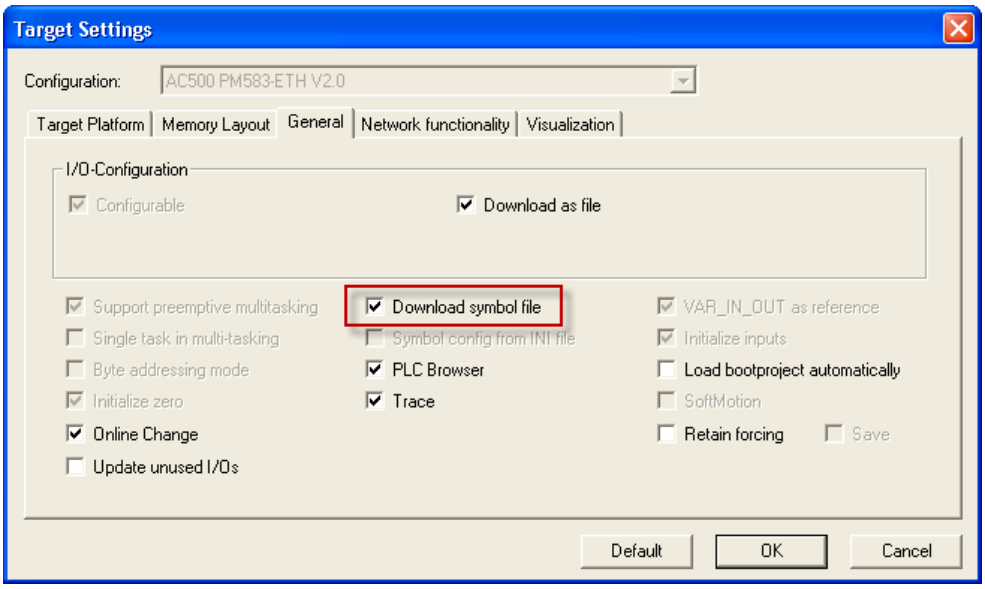


Remarque : vous pouvez donner un nom d'alias uniquement aux variables importées. Il n'est pas possible pour les variables ajoutées manuellement dans l'éditeur de variables de porter un préfixe d'alias dans leur nom.
La chaîne d'alias est associée à l'importation des variables. Une fois l'importation des variables effectuée, une modification de la chaîne d'alias n'aura aucun effet sur les noms figurant déjà dans le dictionnaire. Lorsque vous modifiez la chaîne d'alias et réimportez les variables, toutes les variables réimportées porteront la nouvelle chaîne de préfixe.

Paramètres logiciels CODESYS

Lors de la création du projet sous CODESYS V2, sélectionnez **Télécharger le fichier de symboles** (*Paramètres cibles* > *Général*).

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



Remarque : le périphérique CODESYS ABB Ethernet prend en charge le chargement automatique du fichier de symboles (SDB) depuis le contrôleur. Aucune modification dans le décalage de variable due à une nouvelle compilation du côté du logiciel de l'API ne nécessite la réimportation du fichier de symboles. Le fichier de variables doit être réimporté uniquement si vous renommez ou définissez de nouvelles variables.

Types de données standard

Le module d'importation prend en charge des variables des types de données standard et définies par l'utilisateur.

Les types suivants sont considérés comme types de données standard :

Types de données pris en charge

- BOOL
- WORD
- DWORD
- INT
- UINT
- UDINT
- DINT
- STRING*
- REAL
- TIME
- DATE & TIME

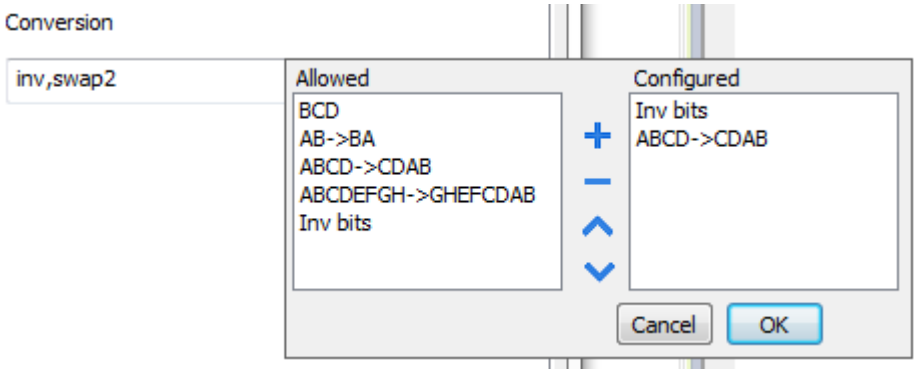
et un TABLEAU (ARRAY) unidimensionnel des types ci-dessus. Voir la section « Concepts de programmation » dans le manuel principal.

Note * : La longueur de chaîne d'une variable STRING dans l'API doit être de 80 caractères maximum. Déclarez une variable STRING soit avec une taille spécifiée (str : STRING(35) soit la taille par défaut (str : STRING) qui est de 80 caractères.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Conversion de variables

Conversion à appliquer à la variable.



En fonction du type de données choisi, la liste **Autorisée** affiche un ou plusieurs types de conversion.

Valeur	Description
Inv bits	**inv** : inverse tous les bits de la variable. *Exemple :* 1001 → 0110 (format binaire) 9 → 6 (format décimal)
Inverser	**neg** : inverse la valeur de la variable. *Exemple :* 25.36 → -25.36
AB -> BA	**swapnibbles** : échange les quartets en un octet. *Exemple :* 15D4 → 514D (format hexadécimal) 5588 → 20813 (format décimal)
ABCD -> CDAB	**swap2** : échange les octets en un mot. *Exemple :* 9ACC → CC9A (format hexadécimal) 39628 → 52378 (format décimal)
ABCDEFGH -> GHEFCDAB	**swap4** : échange les octets en un mot double. *Exemple :* 32FCFF54 → 54FFFC32 (format hexadécimal) 855441236 → 1426062386 (format décimal)
ABC...NOP -> OPM...DAB	**swap8** : échange les octets en un mot long. Exemple : 142,366 → -893553517,588905 (format décimal) 0 10000000110 00011100101110110110010001011010000111001010110000001 → 1 10000011100

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Valeur	Description
	10101010000101000101101101101100101101100000100111101 (format binaire)
BCD	**bcd** : sépare l'octet en deux quartets, les lit au format décimal (de 0 à 9) *Exemple :* 23 → 17 (format décimal) 0001 0111 = 23 0001 = 1 (premier quartet) 0111 = 7 (second quartet)

Sélectionnez la conversion et cliquez sur le bouton +. La conversion sélectionnée s'affiche dans la liste **Configurée**.

Si vous configurez d'autres conversions, celles-ci s'affichent dans l'ordre (du haut vers le bas de la liste **Configurée**).

Utilisez les boutons de direction pour changer l'ordre des conversions configurées.

IP d'écrasement de nœud

Le protocole propose le type de données spécial IP d'écrasement de nœud, qui vous permet de modifier l'adresse IP du contrôleur cible au moment du runtime.

Ce type de mémoire est un tableau de 4 octets non signés, un pour chaque octet de l'adresse IP.

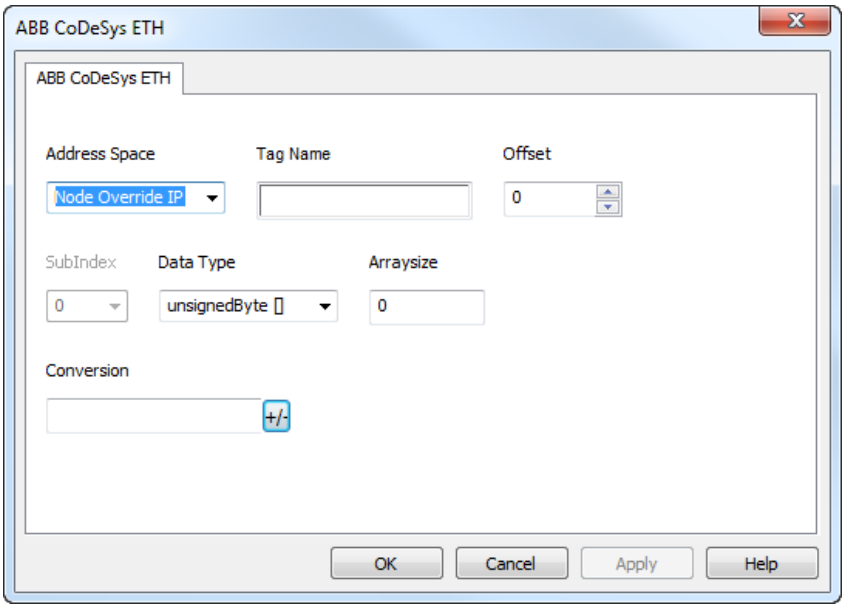
L'IP d'écrasement de nœud est initialisée avec la valeur de l'IP du contrôleur spécifiée dans le projet au moment de la programmation.

IP d'écrasement de nœud	Exécution de l'API
0.0.0.0	La communication avec le contrôleur est coupée, plus aucune trame de requête ne sera générée.
Autre que 0.0.0.0	Interprétée comme étant l'écrasement de l'IP de nœud, et remplace l'adresse IP cible au moment du runtime.

Lorsque le périphérique IHM est connecté à un réseau avec plus d'un nœud de contrôleur, chaque nœud dispose de sa propre variable d'IP d'écrasement de nœud.

Remarque : le maintien des valeurs de l'IP d'écrasement de nœud assignées au runtime est assuré par des cycles d'alimentation.

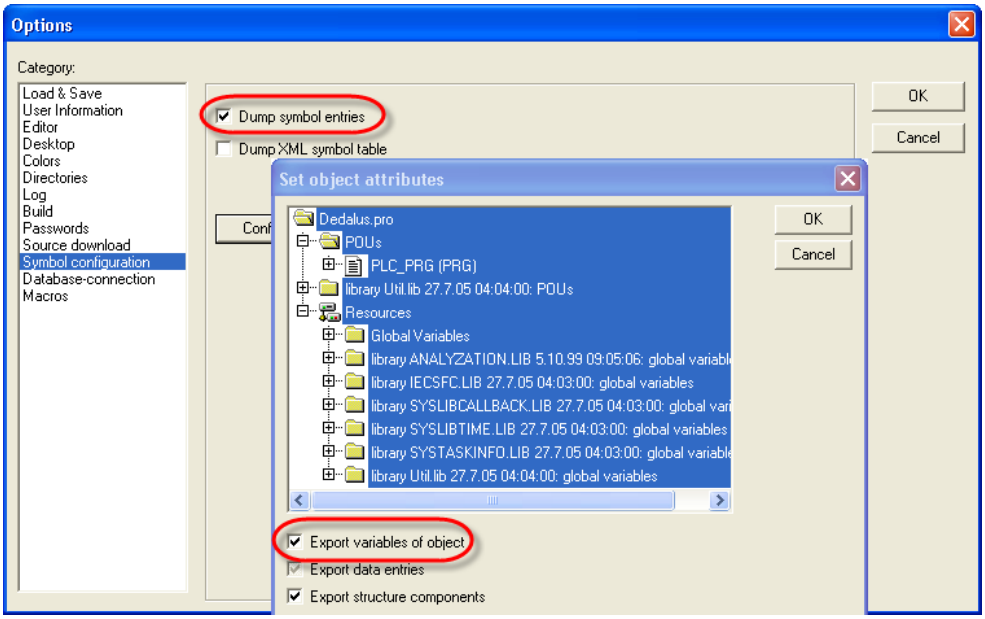
Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



Exporter les variables du contrôleur

Lors de la configuration de l'API grâce au logiciel de configuration du fabricant, autorisez la création d'un Fichier de symboles (dont l'extension est .sym) sous le logiciel de programmation CODESYS :

1. Dans le menu **Projet**, cliquez sur **Options**.
2. Sélectionnez **Configuration de symbole**.
3. Sélectionnez **Vider les entrées de symboles**.
4. Cliquez sur **Configurer un fichier de symbole** : la boîte de dialogue **Définir les attributs d'objet** s'affiche.
5. Sélectionnez **Exporter les variables de l'objet**.
6. Cliquez sur **OK**.



Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Importer des variables

Vous pouvez importer les variables d'un fichier .sym exporté d'un contrôleur. Voir la section « Mon premier projet » du manuel principal.

Statut de communication

Vous pouvez afficher l'état de communication actuel grâce aux variables système.

Voir la section « Variables système » du manuel principal.

Codes pris en charge pour ce pilote de communication :

Erreur	Cause et action
Fichier de symboles absent	Vérifiez Fichier de symboles et téléchargez à nouveau le programme de l'API
« variable » absente du fichier de symboles	Vérifiez la présence de cette variable dans le projet de l'API
Délai d'attente dépassé pour Confirmer	Le contrôleur n'a pas envoyé de confirmation
Délai d'attente dépassé pour la dernière action à Confirmer	Le contrôleur n'a pas envoyé de confirmation pour la dernière action
Délai d'attente dépassé pour la réception des données	Le contrôleur n'a pas renvoyé les données
Délai d'attente de connexion dépassé	Le périphérique n'est pas connecté

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

ABB CODESYS Serial

Le pilote de communication ABB CODESYS Serial est spécialement conçu pour prendre en charge la communication avec des contrôleurs de la série ABB 500 en programmation CEI 61131-3, qui se base sur le système CODESYS V2.3.

Limites

Ce protocole ne prend pas en charge les versions de logiciel AC500 antérieures à la version V2.0.

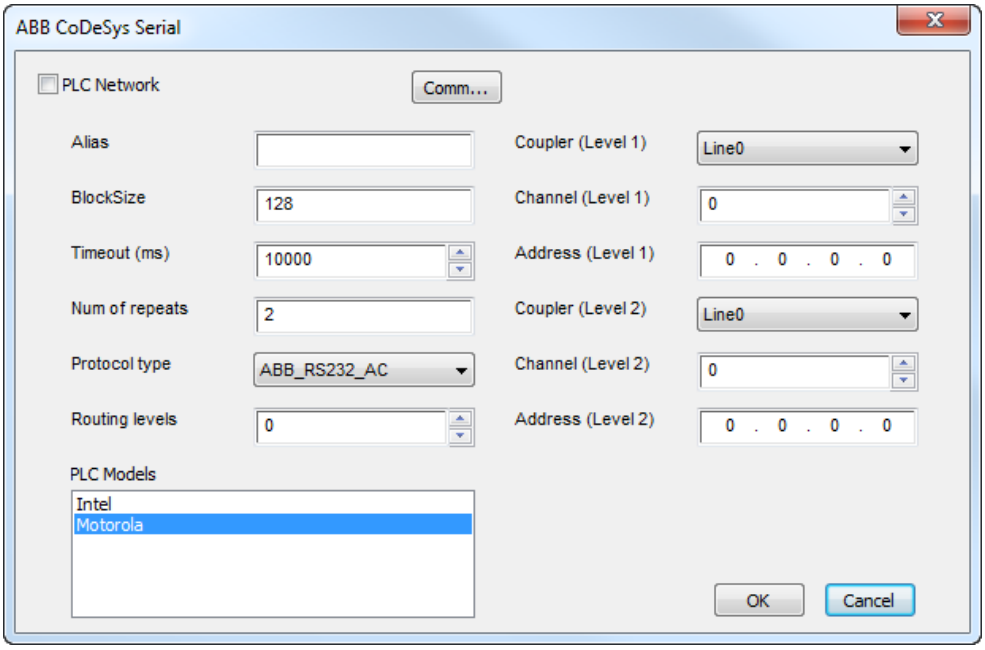
Paramètres de l'éditeur de protocole

Ajouter un protocole

Pour configurer le protocole :

1. Dans le nœud **Config**, double-cliquez sur **Protocoles**.
2. Pour ajouter un pilote, cliquez sur **+** : une nouvelle ligne est ajoutée.
3. Sélectionnez le protocole dans la liste **API**.

La boîte de dialogue de configuration du protocole s'affiche.

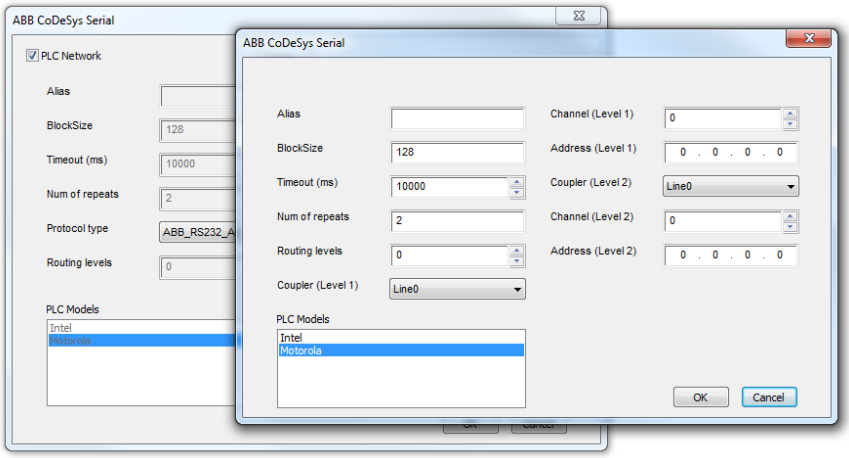
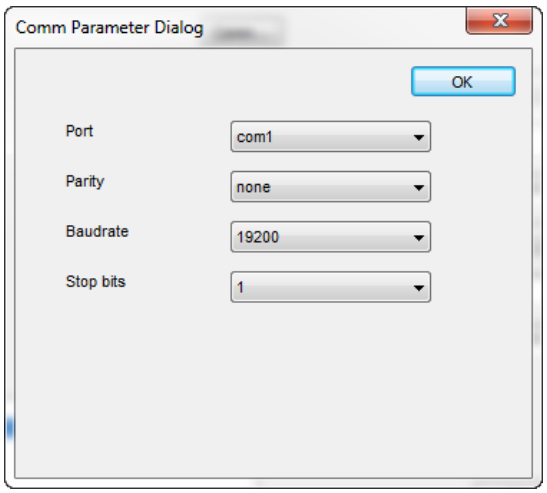


Élément	Description
Alias	Nom d'identification des nœuds dans les configurations réseau. Il est ajouté en préfixe à chaque nom de variable importé pour chaque nœud de réseau.
Taille de bloc	Taille de bloc maximale prise en charge par votre contrôleur (limitée à 1 024 Ko).
Délai d'attente (ms)	Délai d'attente (en millisecondes) entre deux tentatives en cas d'absence de réponse du périphérique esclave.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Élément	Description
Nombre de répétitions	Nombre d'envois d'un certain message au contrôleur avant l'envoi d'un rapport d'erreur de communication.
Type de protocole	Deux types de protocoles disponibles : • **Serial_RS232** : correspond au pilote 3S standard. • **ABB_RS232_AC** : implémente une variation spécifique du protocole standard de niveau 2 en utilisant un pilote routeur supplémentaire. Ce type de protocole est généralement utilisé pour se connecter aux API via d'autres API faisant office de passerelles. Pour le type de protocole ABB_RS232_AC, il convient de configurer convenablement les paramètres supplémentaires suivants : • Niveaux de routage • Coupleur (niveau 1) • Canal (niveau 1) • Adresse (niveau 1) • Coupleur (niveau 2) • Canal (niveau 2) • Adresse (niveau 2) Pour plus d'informations, voir la documentation *AC500 et Control Builder*, chapitre *Interfaces de programmation de l'AC500 utilisé par Control Builder*.
Modèles d'API	La liste vous permet de sélectionner le modèle d'API auquel vous allez vous connecter. Cette sélection influence le décalage des plages de données pour chaque type de données en fonction des ressources spécifiques de mémoire d'API.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Élément	Description
Réseau d'API	Le protocole permet la connexion de contrôleurs multiples. Pour configurer les connexions multiples, cochez la case « Réseau d'API » et entrez un ID de nœud pour chaque esclave auquel vous souhaitez accéder. 
Comm...	En cliquant dessus, vous affichez la boîte de dialogue de configuration des paramètres de communication. 

Élément	Description
Port	Sélection du port série. • **COM1** : port d'API du périphérique. • **COM2** : port de l'ordinateur/imprimante.
Parité, débit en bauds, bits d'arrêt	Paramètres de ligne série. **Vous devez configurer la parité sur « Aucune » pour l'AC500.**

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

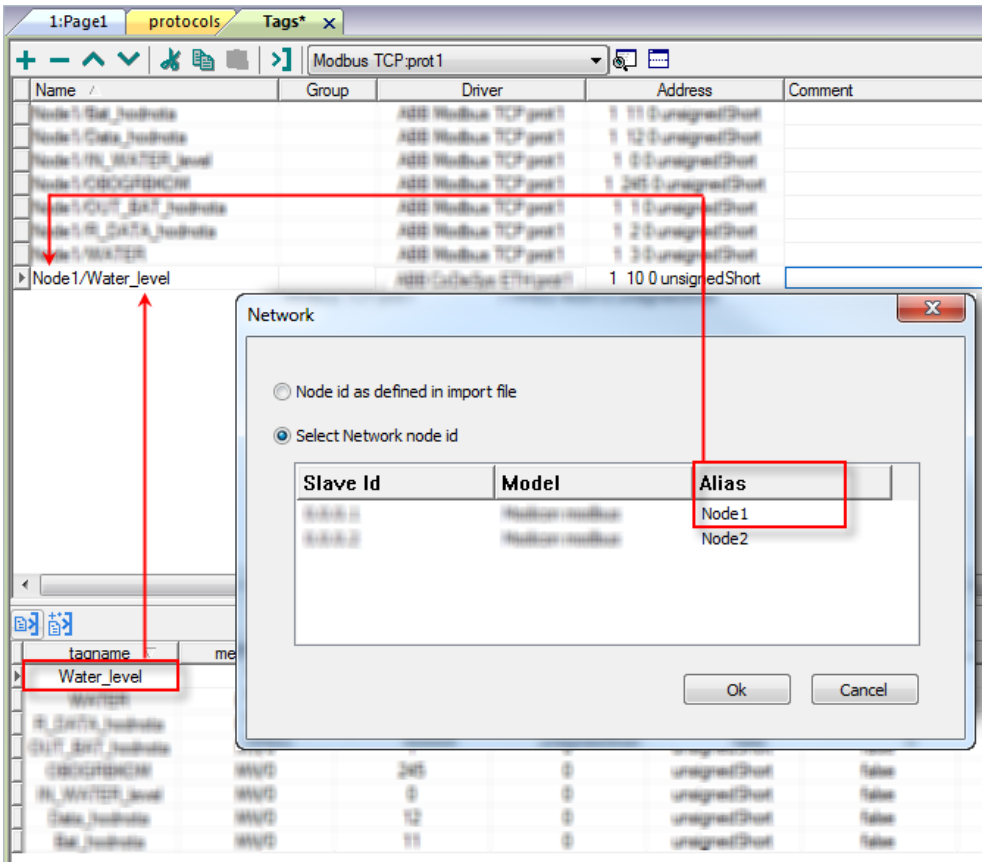
Élément	Description
	<table><tr><th>Élément</th><th>Description</th></tr><tr><td>Mode</td><td>Mode de port série. modes disponibles : - RS-232 - RS-485 (2 fils) - RS-422 (4 fils) </td></tr></table>

Ajouter un nom d'alias à un protocole

Les noms de variables doivent être uniques au niveau d'un projet. Vous pouvez cependant être amené à utiliser les mêmes noms de variables pour différents nœuds de contrôleur (par exemple lorsque le périphérique IHM est connecté à deux périphériques exécutant la même application).

Lorsque vous générez un protocole, vous pouvez ajouter un nom d'alias qui s'ajoutera aux noms de variables importés pour ce protocole.

Dans l'exemple, la connexion à un certain contrôleur porte le nom **Nœud1**. Lorsque vous importez les variables pour ce nœud, tous les noms de variables porteront le préfixe **Nœud1**, les rendant ainsi uniques au niveau projet/réseau.



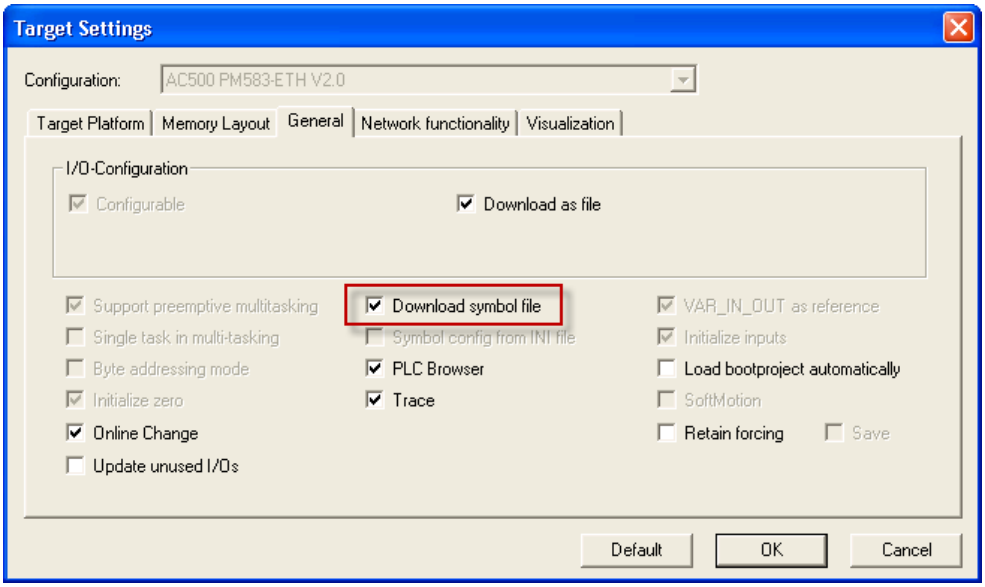
Remarque : vous pouvez donner un nom d'alias uniquement aux variables importées. Il n'est pas possible pour les variables ajoutées manuellement dans l'éditeur de variables de porter un préfixe d'alias dans leur nom.
La chaîne d'alias est associée à l'importation des variables. Une fois l'importation des variables effectuée, une modification de la chaîne d'alias n'aura aucun effet sur les noms figurant déjà dans le dictionnaire. Lorsque vous

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

modifiez la chaîne d'alias et réimportez les variables, toutes les variables réimportées porteront la nouvelle chaîne de préfixe.

Paramètres logiciels CODESYS

Lors de la création du projet sous CODESYS V2, sélectionnez **Télécharger le fichier de symboles** (*Paramètres cibles* > *Général*).



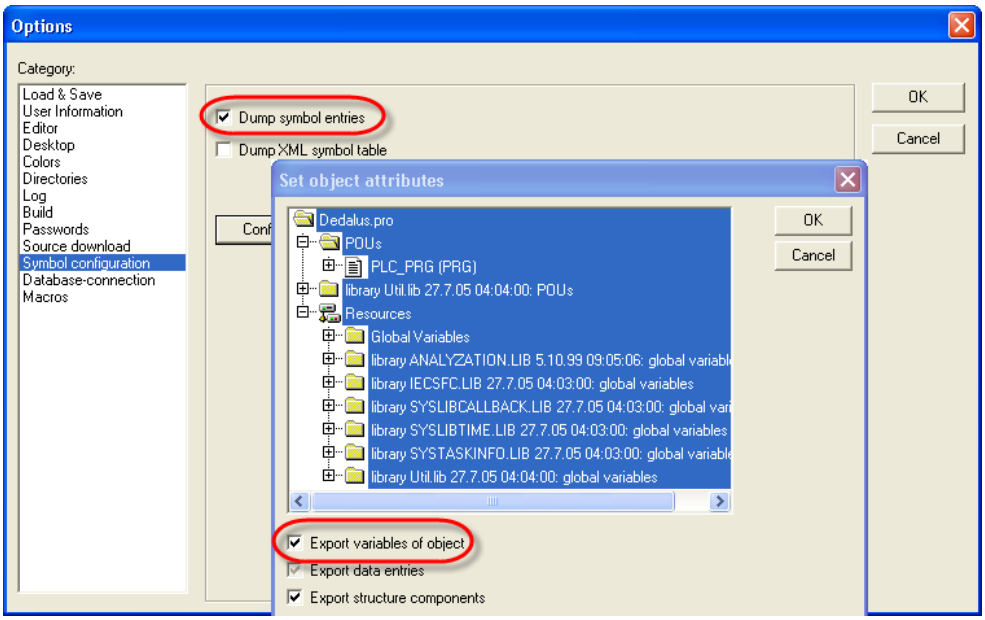
Remarque : le périphérique CODESYS ABB Serial prend en charge le chargement automatique du fichier de symboles (SDB) depuis le contrôleur. Aucune modification dans le décalage de variable due à une nouvelle compilation du côté du logiciel de l'API ne nécessite la réimportation du fichier de symboles. Le fichier de variables doit être réimporté uniquement si vous renommez ou définissez de nouvelles variables.

Exporter les variables du contrôleur

Lors de la configuration de l'API grâce au logiciel de configuration du fabricant, autorisez la création d'un Fichier de symboles (dont l'extension est .sym) sous le logiciel de programmation CODESYS :

1. Dans le menu **Projet**, cliquez sur **Options**.
2. Sélectionnez **Configuration de symbole**.
3. Sélectionnez **Vider les entrées de symboles**.
4. Cliquez sur **Configurer un fichier de symbole** : la boîte de dialogue **Définir les attributs d'objet** s'affiche.
5. Sélectionnez **Exporter les variables de l'objet**.
6. Cliquez sur **OK**.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



Importer des variables

Vous pouvez importer les variables d'un fichier .sym exporté d'un contrôleur. Voir la section « Mon premier projet » du manuel principal.

Types de données

Le module d'importation prend en charge des variables des types de données standard et définies par l'utilisateur.

Types de données pris en charge	• BOOL • WORD • DWORD • INT • UINT • UDINT • DINT • STRING* • REAL • TIME • DATE & TIME et un TABLEAU (ARRAY) unidimensionnel des types ci-dessus. Voir la section « Concepts de programmation » dans le manuel principal. Note * : La longueur de chaîne d'une variable STRING dans l'API doit être de 80 caractères maximum. Déclarez une variable STRING soit avec une taille spécifiée (str : STRING(35) soit la taille par défaut (str : STRING) qui est de 80 caractères.
Types de données non pris en charge	• LWORD • LINT • LREAL

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Statut de communication

Vous pouvez afficher l'état de communication actuel grâce aux variables système. Voir la section « Variables système » du manuel principal.

Codes pris en charge pour ce pilote de communication :

Erreur	Cause et action
Fichier de symboles absent	Vérifiez Fichier de symboles et téléchargez à nouveau le programme de l'API
« variable » absente du fichier de symboles	Vérifiez la présence de cette variable dans le projet de l'API
Délai d'attente dépassé pour Confirmer	Le contrôleur n'a pas envoyé de confirmation
Délai d'attente dépassé pour la dernière action à Confirmer	Le contrôleur n'a pas envoyé de confirmation pour la dernière action
Délai d'attente dépassé pour la réception des données	Le contrôleur n'a pas renvoyé les données
Délai d'attente de connexion dépassé	Le périphérique n'est pas connecté

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

ABB IRC5

Le pilote de communication pour contrôleur de robot ABB IRC5 a été conçu pour établir la communication vers la gamme de contrôleurs de robot ABB Robotics IRC5. Cette version prend en charge la communication vers le système E/S (signaux d'entrée et de sortie) du contrôleur de robot IRC5.

Le pilote de communication n'est pas capable de répondre à des demandes difficiles en temps réel pour de nombreuses raisons :

- Il fonctionne sous un système d'exploitation qui n'est pas en temps réel.
- La communication s'effectue en TCP/IP sur un réseau.
- Le contrôleur peut avoir des tâches de plus haute priorité à effectuer.

Vous pouvez vous attendre à un temps de réponse minimal de l'ordre de 10 à 100 ms.

Détails sur l'implantation

Un système de contrôleur de robot IRC5 utilise des signaux d'entrée et de sortie pour contrôler le processus. Il peut s'agir de signaux numériques, analogiques ou de groupe. Vous pouvez accéder à ces signaux grâce à l'outil PanelBuilder.

Les modifications de signal dans le système robot sont souvent significatives. Dans de nombreuses situations, l'utilisateur final doit recevoir des notifications de ces modifications.

Dans le mode manuel du contrôleur IRC5, la valeur d'un signal ne peut être modifiée que lorsque le niveau d'accès du signal est configuré sur TOUS et que le FlexPendant n'est pas connecté. Sinon, vous devez passer le contrôleur en mode auto. Utilisez les outils du contrôleur de robot IRC5 RobotStudio ou le FlexPendant pour modifier le niveau d'accès à un signal.

Une application PanelBuilder fait office de client distant comparé au FlexPendant, qui est un client local.

Les clients distants ne disposent pas de tous les privilèges d'un client local, en particulier en mode manuel. Par exemple, en mode manuel le FlexPendant peut démarrer et arrêter les programmes du contrôleur de robot IRC5 grâce aux entrées du système. Pour un client distant, cette fonctionnalité est disponible uniquement en mode automatique.

Un client distant peut surveiller plusieurs contrôleurs de robot et y accéder depuis un lieu différent.

Sécurité du personnel

AVERTISSEMENT : un robot est lourd et extrêmement puissant quelle que soit sa vitesse. Une pause ou un long arrêt dans le mouvement peut être suivi d'un mouvement dangereux et rapide. Même si un modèle de mouvement est prédit, un changement dans le fonctionnement peut être déclenché par un signal externe résultant en un mouvement inattendu.

Vous devez suivre toutes les règles de sécurité lorsque vous pénétrez dans un espace protégé. Assurez-vous de connaître les règles de sécurité décrites dans les guides d'utilisation IRC5.

Pilote ABB IRC5

Le pilote IRC5 est pris en charge à partir de la version PB610 V1.90.0.778.

Vous pouvez connecter des périphériques IHM aux réseaux de robot ABB en tant que clients utilisant ce pilote de communication.

Le pilote ABB IRC5 facilite le traitement des connexions des contrôleurs de robot ABB, assurant une prise en charge spécifique des fonctions d'importation de variables.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

L’exigence minimale est la version du système d'exploitation (version BSP) sur le périphérique IHM. Consultez le manuel d’utilisation du logiciel pour connaître la version BSP de votre périphérique IHM.

Produits	Index des changements	Date de production	Version BSP
CP650, CP660, CP675	B2 ou plus récent	WK10 2013 (1103) ou ultérieur	V2.80 ou plus récent
CP620, CP630, CP635	B2 ou plus récent	WK10 2013 (1103) ou ultérieur	V1.76 ou plus récent
CP651, CP661, CP665, CP676	A0		V1.76 ou plus récent

Remarque : vous devrez configurer le masque de sous-réseau IP du périphérique IRC5 sur 255.255.255.0.

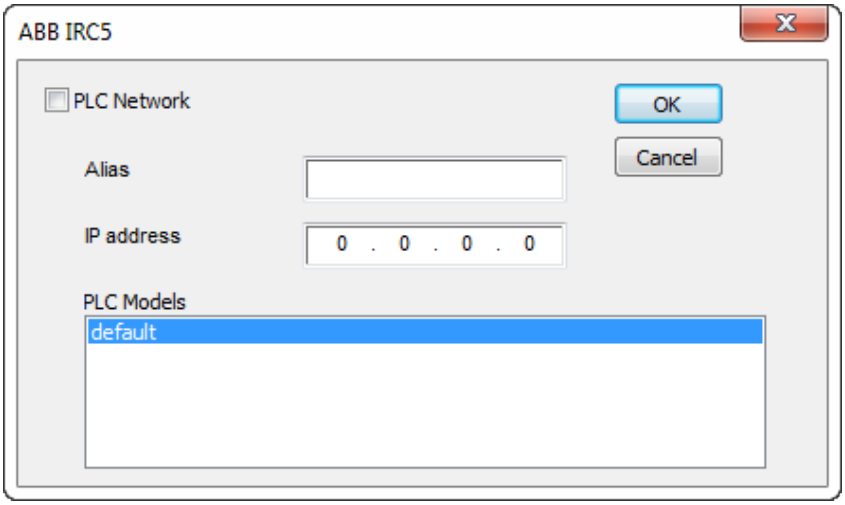
Paramètres de l’éditeur de protocole

Ajouter un protocole

Pour configurer le protocole :

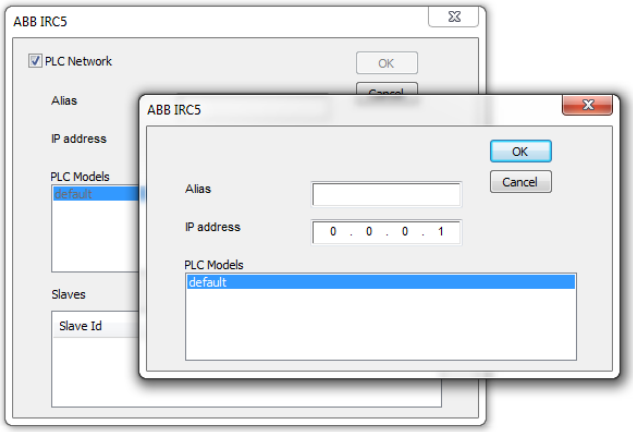
1. Dans le nœud **Config**, double-cliquez sur **Protocoles**.
2. Pour ajouter un pilote, cliquez sur **+** : une nouvelle ligne est ajoutée.
3. Sélectionnez le protocole dans la liste **API**.

La boîte de dialogue de configuration du protocole s’affiche.



Élément	Description
Alias	Nom à utiliser pour l’identification des nœuds dans les configurations réseau. Il est ajouté en préfixe à chaque nom de variable importé pour chaque nœud de réseau
Adresse IP	Adresse IP Ethernet du contrôleur.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Élément	Description
Modèles d'API	Seul un type de modèle est actuellement disponible.
Réseau d'API	Adresse IP de tous les contrôleurs en connexions multiples. Vous devez sélectionner la case à cocher **Réseau d'API** pour permettre les connexions multiples. 

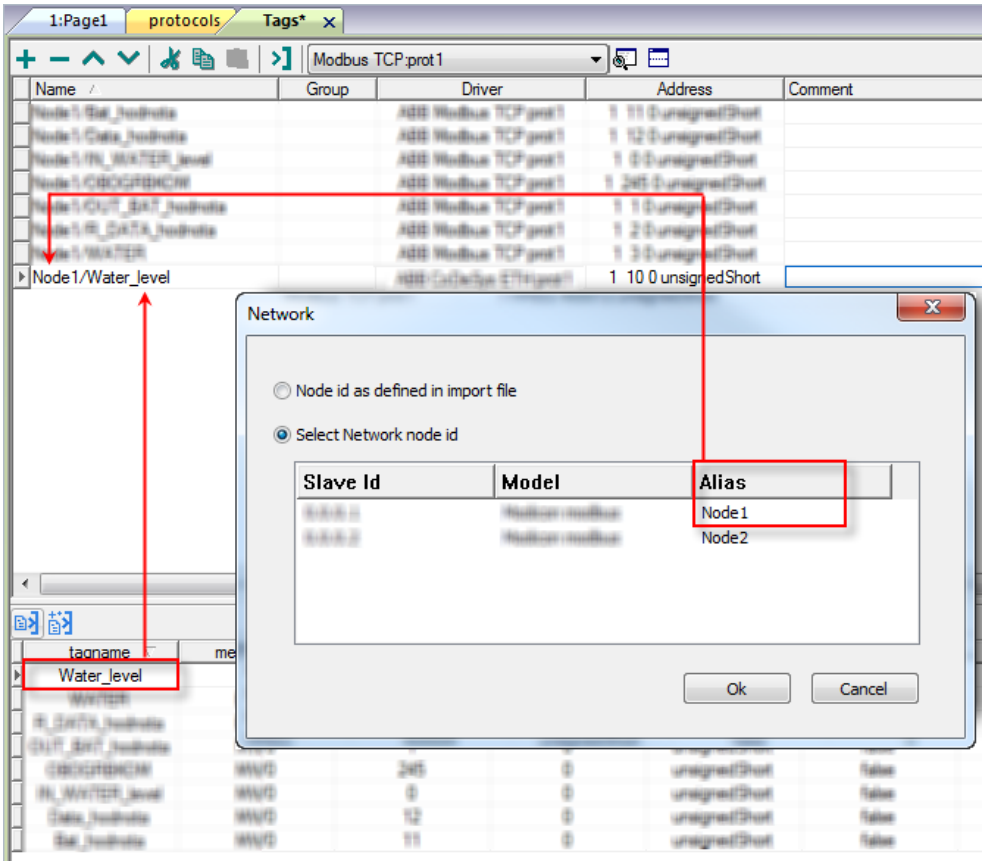
Ajouter un nom d'alias à un protocole

Les noms de variables doivent être uniques au niveau d'un projet. Vous pouvez cependant être amené à utiliser les mêmes noms de variables pour différents nœuds de contrôleur (par exemple lorsque le périphérique IHM est connecté à deux périphériques exécutant la même application).

Lorsque vous générez un protocole, vous pouvez ajouter un nom d'alias qui s'ajoutera aux noms de variables importés pour ce protocole.

Dans l'exemple, la connexion à un certain contrôleur porte le nom **Nœud1**. Lorsque vous importez les variables pour ce nœud, tous les noms de variables porteront le préfixe **Nœud1**, les rendant ainsi uniques au niveau projet/réseau.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



Remarque : vous pouvez donner un nom d'alias uniquement aux variables importées. Il n'est pas possible pour les variables ajoutées manuellement dans l'éditeur de variables de porter un préfixe d'alias dans leur nom.
La chaîne d'alias est associée à l'importation des variables. Une fois l'importation des variables effectuée, une modification de la chaîne d'alias n'aura aucun effet sur les noms figurant déjà dans le dictionnaire. Lorsque vous modifiez la chaîne d'alias et réimportez les variables, toutes les variables réimportées porteront la nouvelle chaîne de préfixe.

Exporter les variables du contrôleur

L'outil de programmation des contrôleurs ABB Robotics peut générer un fichier de symboles .cfg.

Le module d'importation prend en charge des variables des types de données standard suivants.

- booléen
- entier non signé
- flottant

Remarque : Le périphérique CP600 peut uniquement utiliser les signaux E/S et modifier les signaux de sortie du contrôleur de robot. Il peut les modifier uniquement lorsque le contrôleur IRC5 est en mode automatique ou lorsque le FlexPendant n'est pas connecté et que le niveau d'accès du signal est configuré sur TOUS.

Importer des variables

Vous pouvez importer les variables d'un fichier .cfg exporté d'un contrôleur. Voir la section « Mon premier projet » du manuel principal.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Statut de communication

Vous pouvez afficher l'état de communication actuel grâce aux variables système. Voir la section « Variables système » du manuel principal.

Codes pris en charge par ce pilote de communication :

Erreur	Cause	Action
Impossible de trouver le nœud x.x.x.x "	La requête n'a obtenu aucune réponse dans le délai d'attente spécifié.	Vérifiez que le contrôleur est connecté et correctement configuré pour accéder au réseau.
Erreur de lecture du signal E/S	Le contrôleur a envoyé une réponse ou un contenu non valide au périphérique.	Vérifiez la cohérence entre les données programmées dans le projet et les ressources du contrôleur.
Signal E/S de faible qualité	Le périphérique a reçu une réponse du contrôleur avec un signal de faible qualité.	-
Erreur requête maître	Le périphérique n'a pas l'autorisation d'écrire des données dans le contrôleur.	-

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

ABB Mint Controller HCP

Ce protocole de communication permet de connecter les périphériques IHM aux périphériques de mouvement et de servo-variateur ABB grâce aux protocoles de communication HCP et HCP2.

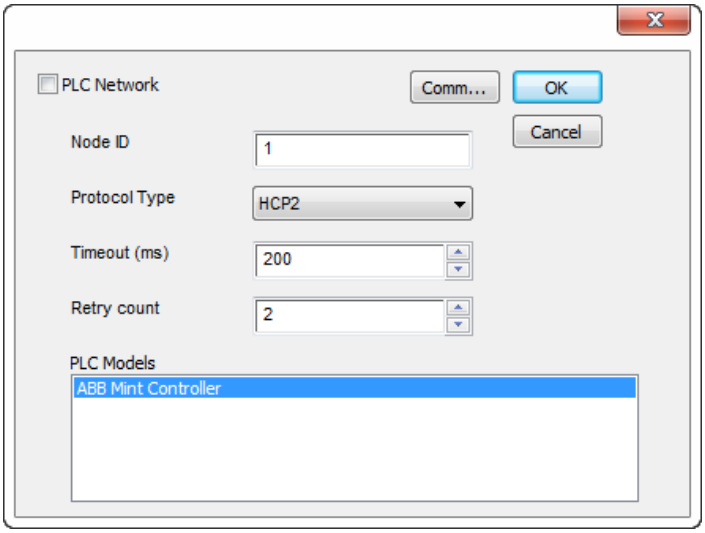
Paramètres de l'éditeur du protocole

Ajouter un protocole

Pour configurer le protocole :

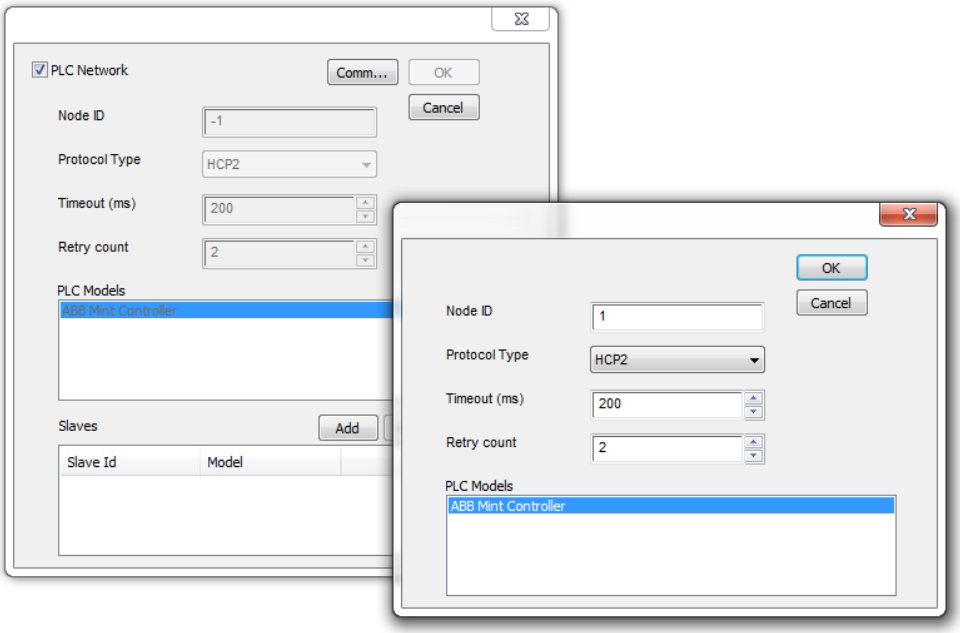
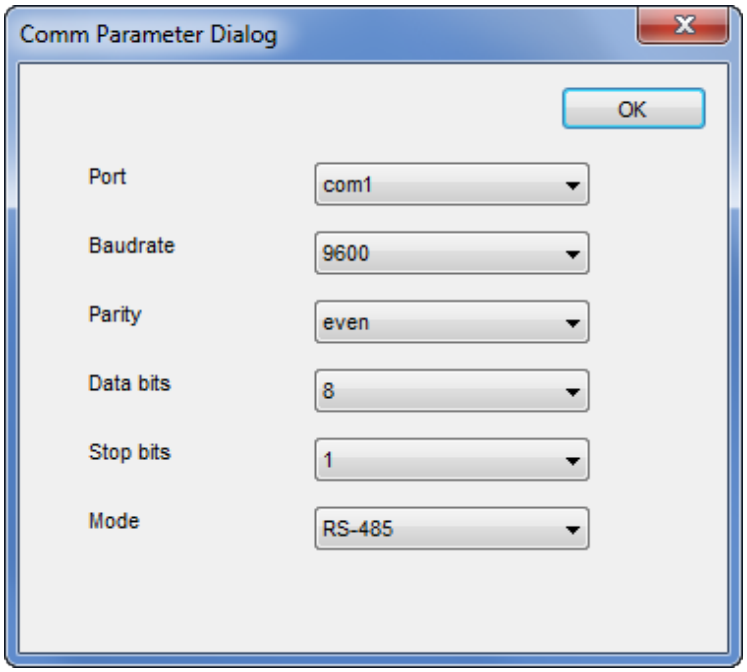
1. Dans le nœud **Config**, double-cliquez sur **Protocoles**.
2. Pour ajouter un pilote, cliquez sur **+** : une nouvelle ligne est ajoutée.
3. Sélectionnez le protocole dans la liste **API**.

La boîte de dialogue de configuration du protocole s'affiche.



Élément	Description
ID de nœud	ID de nœud assigné au périphérique du contrôleur.
Type de protocole	Deux protocoles sont disponibles : • **HCP** • **HCP2**
Délai d'attente (ms)	Délai d'attente (en millisecondes) entre deux tentatives en cas d'absence de réponse du périphérique esclave.
Nombre de tentatives	Nombre d'envois d'un certain message au contrôleur avant l'envoi d'un rapport d'erreur de communication.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Élément	Description
Modèles d'API	Modèle d'API auxquels vous allez vous connecter.
Réseau d'API	Le protocole permet la connexion de contrôleurs multiples à un périphérique IHM. Pour configurer les connexions multiples, cochez la case « Réseau d'API » et entrez un ID de nœud pour chaque esclave auquel vous souhaitez accéder. 
Comm...	En cliquant dessus, vous affichez la boîte de dialogue de configuration des paramètres de communication. 

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Élément	Description
	<table><tr><th>Élément</th><th>Description</th></tr><tr><td>Port</td><td>Sélection du port série. - COM1 : port d’API du périphérique. - COM2 : port de l’ordinateur/imprimante. </td></tr><tr><td>Débit en bauds, parité, bits de données, bits d’arrêt</td><td>Paramètres de ligne série.</td></tr><tr><td>Mode</td><td>Mode de port série. modes disponibles : - RS-232. - RS-485 (2 fils) - RS-422 (4 fils) </td></tr></table>

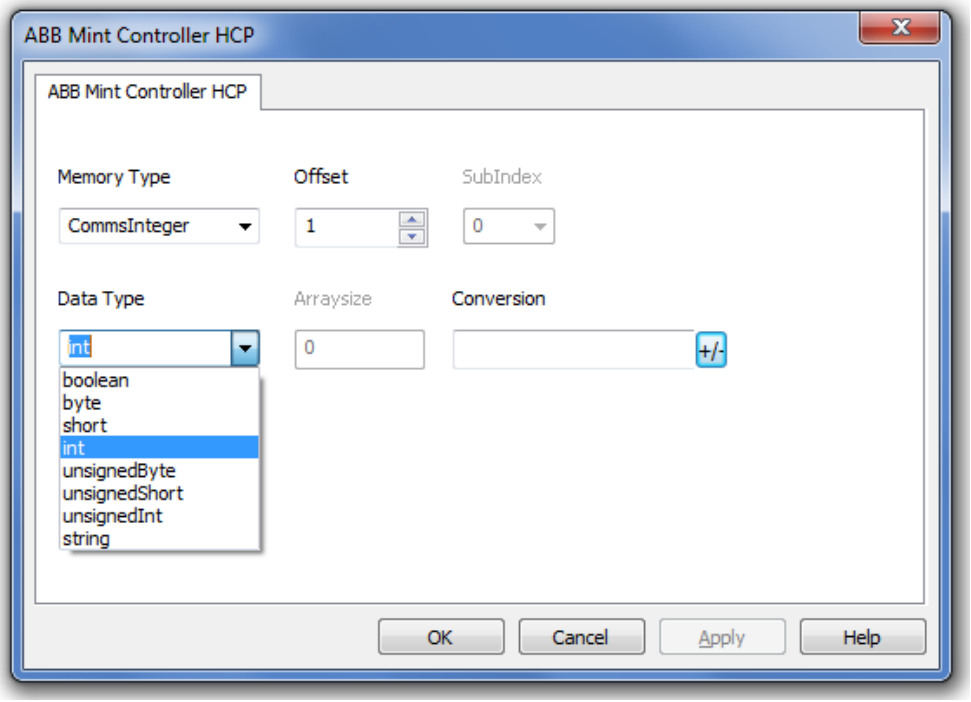
Types de données

Le pilote ABB Mint Controller HCP prend en charge deux types de mémoire, qui correspondent à la même zone de mémoire physique dans le contrôleur Mint :

- **Comms** : s’utilise uniquement avec des valeurs en virgule flottante. Le programme Mint du contrôleur ABB doit utiliser COMMS pour accéder à ces données.
- **CommsInteger** : permet de sélectionner de nombreux types de données basés sur une valeur entière.

Si le programme du contrôleur Mint utilise...	alors...
Mot-clé COMMS pour la configuration d'une variable permettant d’utiliser le type de mémoire Commsinteger	seuls les 23 derniers octets seront exacts (en raison de la précision en virgule flottante du mot-clé COMMS).
Mot-clé COMMSINTEGER pour la configuration d'une variable permettant d’utiliser le type de mémoire Commsinteger	la valeur est exacte pour l’intégralité des 32 octets.

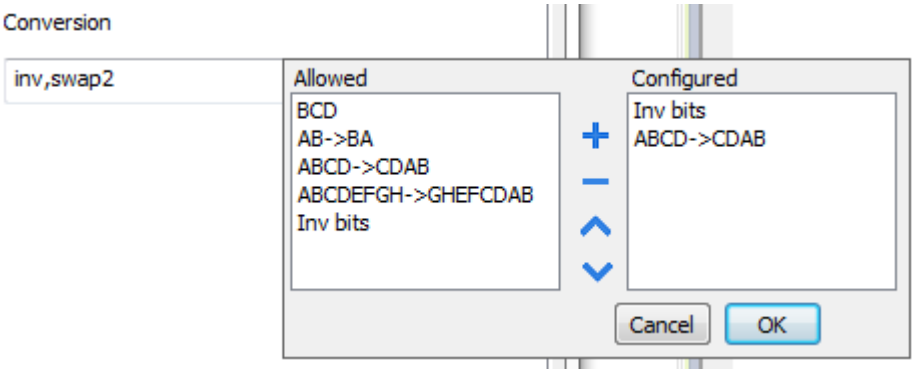
Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



Consultez la section « Concepts de programmation » du manuel principal.

Conversion de variables

Conversion à appliquer à la variable.



En fonction du type de données choisi, la liste **Autorisée** affiche un ou plusieurs types de conversion.

Valeur	Description
Inv bits	**inv** : inverse tous les bits de la variable. *Exemple :* 1001 → 0110 (format binaire) 9 → 6 (format décimal)
Inverser	**neg** : inverse la valeur de la variable. *Exemple :*

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Valeur	Description
	25.36 → -25.36
AB -> BA	**swapnibbles** : échange les quartets en un octet. *Exemple :* 15D4 → 514D (format hexadécimal) 5588 → 20813 (format décimal)
ABCD -> CDAB	**swap2** : échange les octets en un mot. *Exemple :* 9ACC → CC9A (format hexadécimal) 39628 → 52378 (format décimal)
ABCDEFGH -> GHEFCDAB	**swap4** : échange les octets en un mot double. *Exemple :* 32FCFF54 → 54FFFC32 (format hexadécimal) 855441236 → 1426062386 (format décimal)
ABC...NOP -> OPM...DAB	**swap8** : échange les octets en un mot long. Exemple : 142,366 → -893553517,588905 (format décimal) 0 10000000110 0001110010111011011001000101101000011100101011000001 → 1 10000011100 1010101000010100010110110110110010110110000100111101 (format binaire)
BCD	**bcd** : sépare l'octet en deux quartets, les lit au format décimal (de 0 à 9) *Exemple :* 23 → 17 (format décimal) 0001 0111 = 23 0001 = 1 (premier quartet) 0111 = 7 (second quartet)

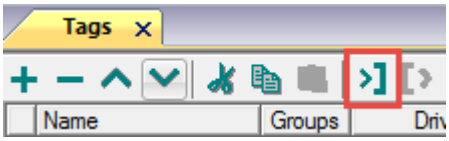
Sélectionnez la conversion et cliquez sur le bouton +. La conversion sélectionnée s'affiche dans la liste **Configurée**.

Si vous configurez d'autres conversions, celles-ci s'affichent dans l'ordre (du haut vers le bas de la liste **Configurée**).

Utilisez les boutons de direction pour changer l'ordre des conversions configurées.

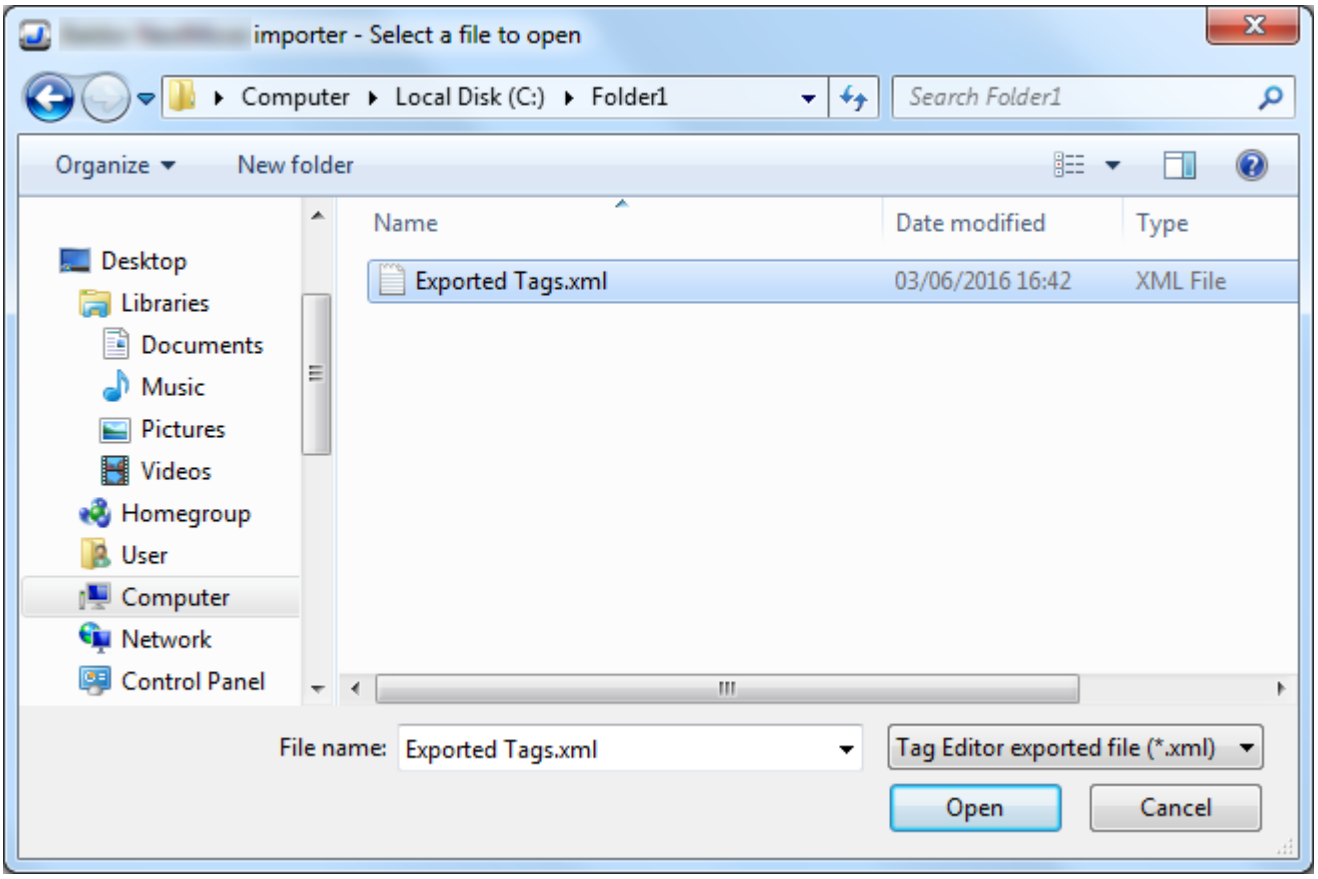
Importation des variables

Sélectionnez le pilote dans l'éditeur de variables et cliquez sur le bouton **Importer variables** pour lancer le pilote d'importation.



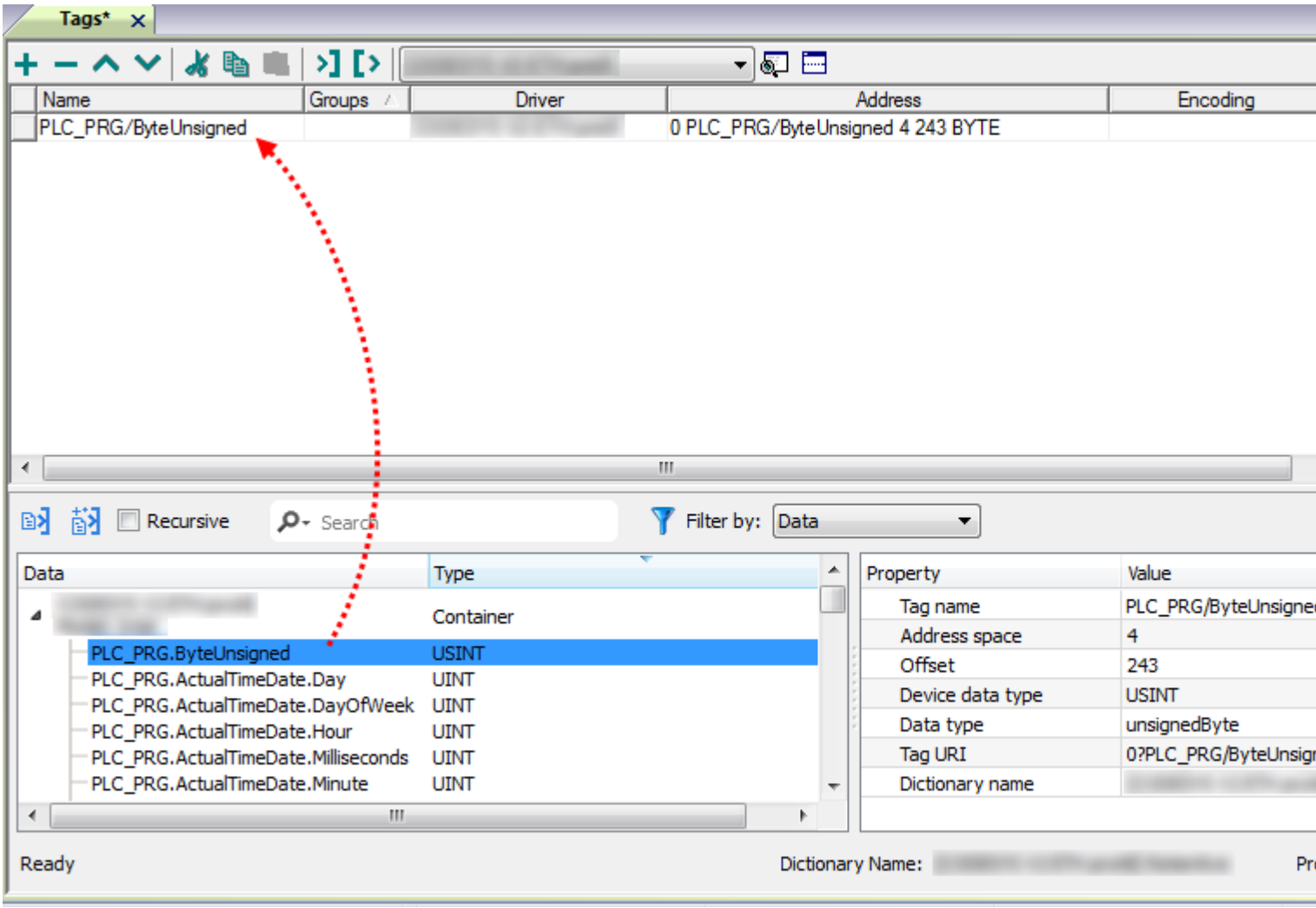
Localisez le fichier **.xml** exporté depuis l'éditeur de variables et cliquez sur **Ouvrir**.

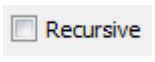
Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



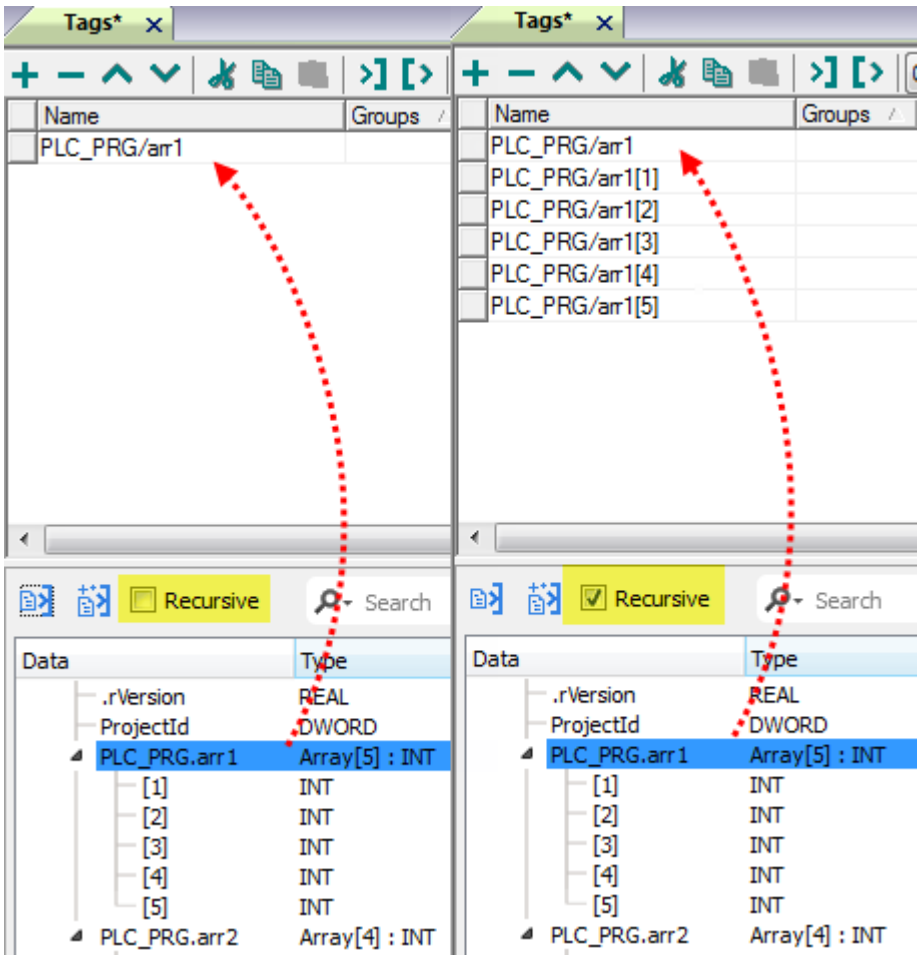
Les variables comprises dans le fichier de symboles sont listées dans le dictionnaire des variables. Le dictionnaire des variables est affiché en bas de l'écran.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



Élément de la barre d'outils	Description
	Importer variable(s). Sélectionnez les variables à importer et cliquez sur cette icône pour ajouter les variables du dictionnaire au projet.
	Mettre à jour variable(s). Cliquez sur cette icône pour mettre à jour les variables du projet grâce à un nouvel import du dictionnaire.
	Cochez cette case pour importer tous les sous-éléments d'une variable. Exemples de résultats avec la case cochée et non cochée :

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Élément de la barre d'outils	Description
	
	Recherche des variables dans le dictionnaire en filtrant en fonction de l'élément sélectionné dans la liste déroulante.

Statut de communication

Vous pouvez afficher l'état de communication actuel grâce aux variables système. Voir la section « Variables système » du manuel principal.

Codes pris en charge par ce pilote de communication :

Erreur	Cause	Action
NAK	Le contrôleur répond « Ne pas confirmer ».	-
Délai d'attente	La requête n'a obtenu aucune réponse dans le délai d'attente spécifié.	Vérifiez que le contrôleur est connecté et correctement configuré pour accéder au réseau.
Erreur de ligne	Une erreur a été détectée dans la configuration des paramètres de communication (parité, débit en bauds, bits de données, bits d'arrêt).	Vérifiez que les paramètres de communication du contrôleur sont compatibles avec ceux du périphérique.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Erreur	Cause	Action
Réponse non valide	Le contrôleur a envoyé une réponse ou un contenu non valide au périphérique.	Vérifiez la cohérence entre les données programmées dans le projet et les ressources du contrôleur.
Erreur générale	Erreur inconnue. Ne doit jamais être signalée.	Contactez l'assistance technique.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

ABB Modbus RTU

Il est possible de connecter les périphériques IHM à un réseau Modbus, qui fera office de réseau maître du pilote Modbus RTU.

L'implantation spécifique du pilote ABB Modbus RTU facilite le traitement des connexions des contrôleurs ABB, assurant une prise en charge spécifique des modèles d'API et des fonctions d'importation de variables.

Détails sur l'implantation

La mise en œuvre du ABB Modbus RTU prend seulement en charge un sous-ensemble des codes fonction du RTU Modbus standard.

Code	Fonction	Description
01	Lecture du statut de la bobine	Lecture de bits multiples dans la zone du périphérique bobine.
02	Lecture du statut d'entrée	Lecture du statut « MARCHE/ARRÊT » des entrées discrète (référence 1x) de l'esclave
03	Lecture des registres de maintien	Lecture de registres multiples
04	Lecture des registres d'entrée	Lecture du contenu binaire des registres d'entrée (référence 3x) de l'esclave
05	Forcer bobine seule	Force une bobine unique à passer en état « MARCHE » ou « ARRÊT »
06	Prédéfinir registre seul	Prédéfinit une valeur dans un registre
16	Prédéfinir registres multiples	Prédéfinit une valeur dans des registres multiples

Remarque : vitesse de communication avec les contrôleurs s'élève à 115 200 bauds.

Remarque : le format de données en virgule flottante est conforme à la norme IEEE.

Paramètres de l'éditeur de protocole

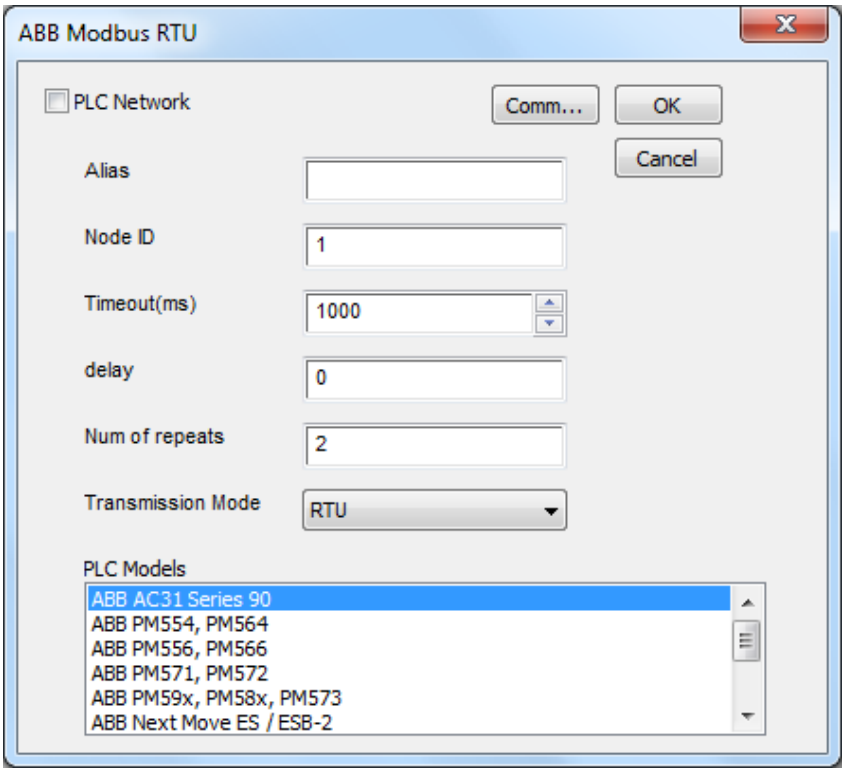
Ajouter un protocole

Pour configurer le protocole :

1. Dans le nœud **Config**, double-cliquez sur **Protocoles**.
2. Pour ajouter un pilote, cliquez sur **+** : une nouvelle ligne est ajoutée.
3. Sélectionnez le protocole dans la liste **API**.

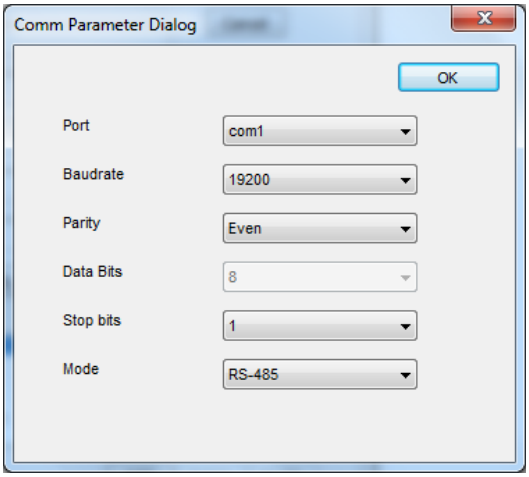
La boîte de dialogue de configuration du protocole s'affiche.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



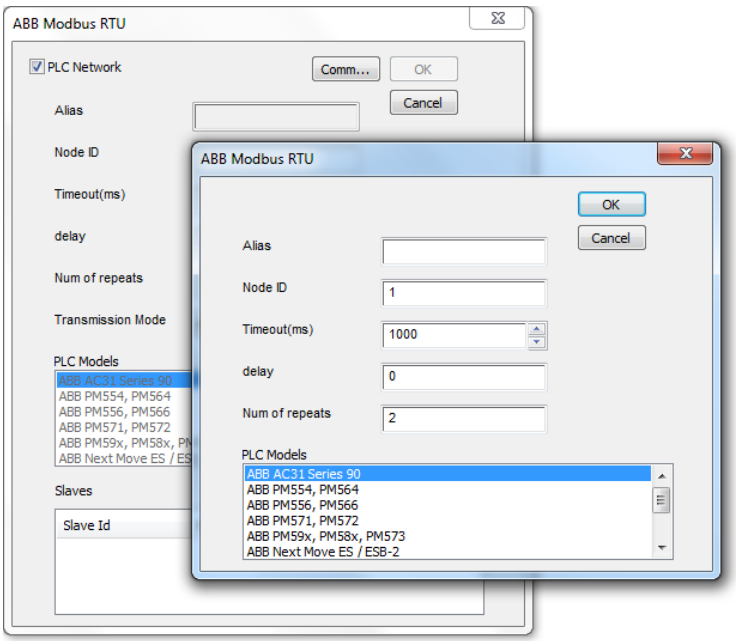
Élément	Description
Alias	Nom d'identification des nœuds dans les configurations réseau. Il est ajouté en préfixe à chaque nom de variable importé pour chaque nœud de réseau.
ID de nœud	Nœud Modbus du périphérique esclave.
Délai d'attente (ms)	Délai d'attente (en millisecondes) entre deux tentatives en cas d'absence de réponse du périphérique esclave.
Retard	Retard en millisecondes entre la fin de la dernière trame reçue et le début d'une nouvelle requête. S'il est à 0, le système interne envoie une nouvelle requête dès qu'il est capable de la programmer.
Nombre de répétitions	Nombre d'envois d'un certain message au contrôleur avant l'envoi d'un rapport d'erreur de communication.
Mode de transmission	• **RTU** : utilise le mode RTU • **ASCII** : utilise le mode ASCII Remarque : lorsque le réseau API est actif, tous les nœuds sont configurés sur le même mode de transmission.
Modèles d'API	Modèles d'API auxquels vous allez vous connecter. Cette sélection influence le décalage des plages de données pour chaque type de données en fonction des ressources spécifiques de mémoire d'API.
Comm...	En cliquant dessus, vous affichez la boîte de dialogue de configuration des paramètres de

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

<table>
<tr><th>Élément</th><th>Description</th></tr>
<tr><td></td><td>communication.
 

<table>
<tr><th>Élément</th><th>Paramètre</th></tr>
<tr><td>Port</td><td>Sélection du port série.
COM1 : port d’API du périphérique.
COM2 : port de l’ordinateur/imprimante.</td></tr>
<tr><td>Débit en bauds, parité, bits de données, bits d’arrêt</td><td>Paramètres de ligne série.</td></tr>
<tr><td>Mode</td><td>Mode de port série. modes disponibles :
RS-232.
RS-485 (2 fils)
RS-422 (4 fils)</td></tr>
</table>
Lorsque vous utilisez les contrôleurs suivants :
ABB NextMove ES / ESB-2
ABB e100 Motion Product
ABB e150 Motion Product
assurez-vous de configurer la parité sur « Aucune »</td></tr>
<tr><td>Réseau d’API</td><td>Le protocole permet la connexion de multiples contrôleurs à un seul panneau opérateur. Vous devez sélectionner Réseau d’API pour activer des connexions multiples.</td></tr>
</table>

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Élément	Description
	 Remarque : le modèle d'**API Pluto Safety** est disponible pour des raisons de compatibilité. Si vous devez vous connecter au modèle d'API, veuillez sélectionner le protocole **ABB Pluto**.

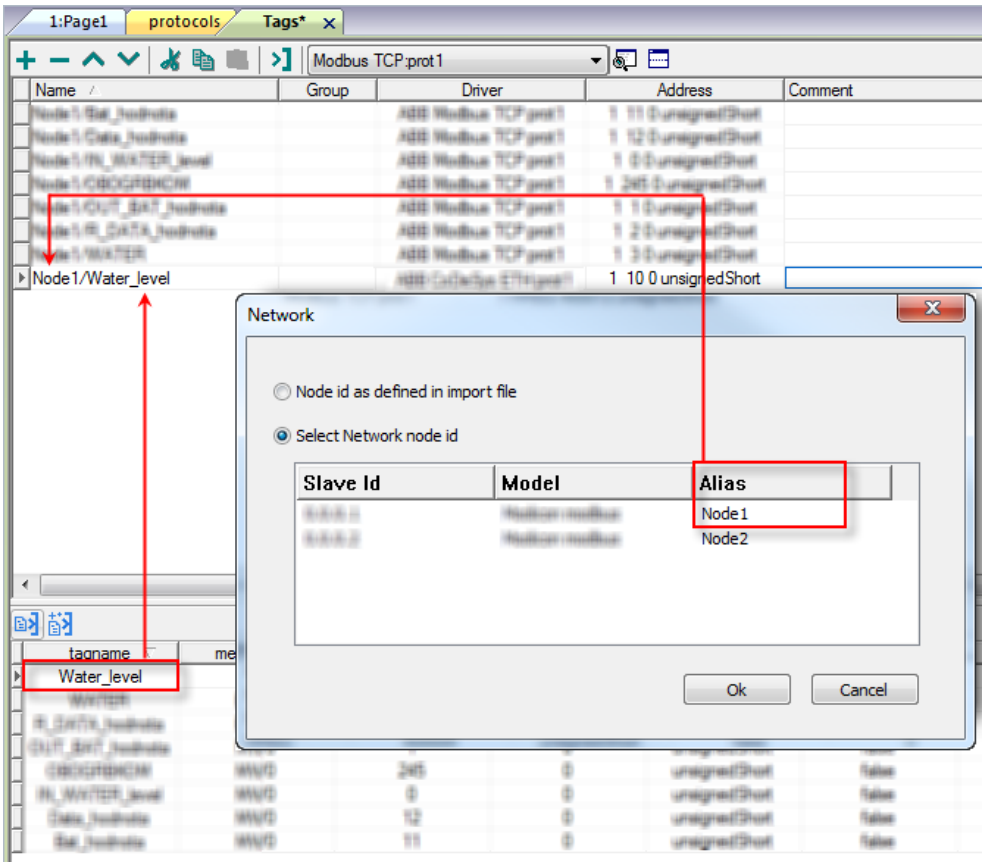
Ajouter un nom d'alias à un protocole

Les noms de variables doivent être uniques au niveau d'un projet. Vous pouvez cependant être amené à utiliser les mêmes noms de variables pour différents nœuds de contrôleur (par exemple lorsque le périphérique IHM est connecté à deux périphériques exécutant la même application).

Lorsque vous générez un protocole, vous pouvez ajouter un nom d'alias qui s'ajoutera aux noms de variables importés pour ce protocole.

Dans l'exemple, la connexion à un certain contrôleur porte le nom **Nœud1**. Lorsque vous importez les variables pour ce nœud, tous les noms de variables porteront le préfixe **Nœud1**, les rendant ainsi uniques au niveau projet/réseau.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



Remarque : vous pouvez donner un nom d'alias uniquement aux variables importées. Il n'est pas possible pour les variables ajoutées manuellement dans l'éditeur de variables de porter un préfixe d'alias dans leur nom.
La chaîne d'alias est associée à l'importation des variables. Une fois l'importation des variables effectuée, une modification de la chaîne d'alias n'aura aucun effet sur les noms figurant déjà dans le dictionnaire. Lorsque vous modifiez la chaîne d'alias et réimportez les variables, toutes les variables réimportées porteront la nouvelle chaîne de préfixe.

ID d'écrasement de nœud (périphériques maîtres)

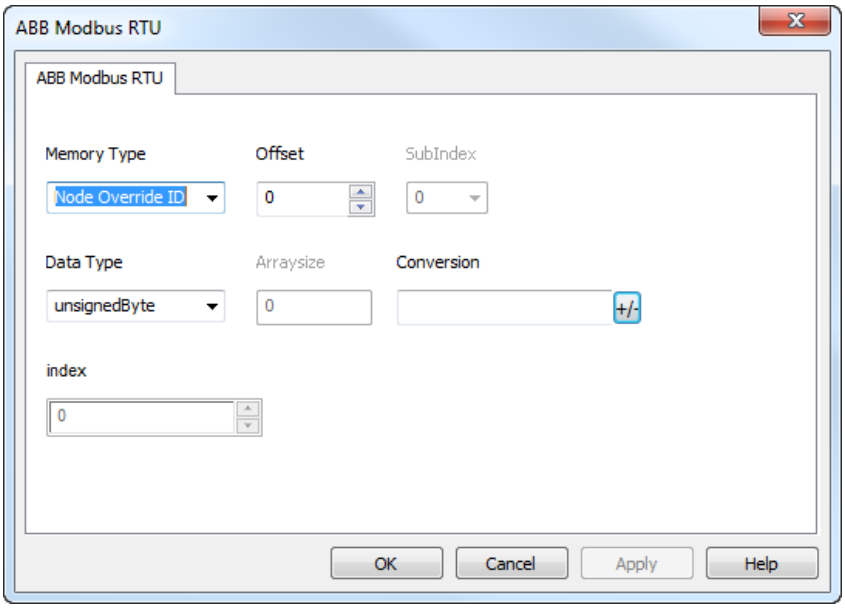
Le protocole propose le type de données spécial ID d'écrasement de nœud, qui vous permet de modifier l'ID de nœud du contrôleur cible au runtime. Ce type de mémoire est un octet non signé.

L'ID d'écrasement de nœud est initialisé avec la valeur de l'ID de nœud spécifiée dans le projet au moment de la programmation.

ID d'écrasement de nœud	Opération Modbus
0	La communication avec le contrôleur est coupée. En cas d'opération d'écriture, la requête sera transmise sans attente de réponse.
1 à 254	Interprété comme étant la valeur du nouvel ID de nœud et remplacé pour l'opération de runtime.
255	La communication avec le contrôleur est coupée, aucun message de requête ne sera généré.

Remarque : le maintien de la valeur de l'ID d'écrasement de nœud assignée au runtime est assuré par des cycles d'alimentation.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

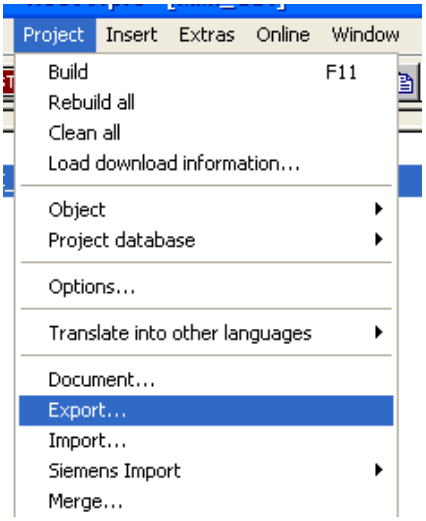


Exporter les variables d'un contrôleur

La programmation des contrôleurs ABB prend en charge l'exportation des variables au format .exp.

Pour exporter des variables :

Sélectionnez **Projet**> **Exporter...** : cela entraîne la création d'un fichier .exp.



Importer des variables

Vous pouvez importer les variables d'un fichier .exp exporté d'un contrôleur. Voir la section « Mon premier projet » du manuel principal.

Statut de communication

Vous pouvez afficher le statut de communication actuel grâce aux variables système. Voir la section « Variables système » du manuel principal.

Codes pris en charge pour ce pilote de communication :

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Erreur	Cause	Action
NAK	Le contrôleur répond « Ne pas confirmer ».	-
Délai d'attente	La requête n'a obtenu aucune réponse dans le délai d'attente spécifié.	Vérifiez que le contrôleur est connecté et correctement configuré pour accéder au réseau.
Erreur de ligne	Une erreur a été détectée dans la configuration des paramètres de communication (parité, débit en bauds, bits de données, bits d'arrêt).	Vérifiez que les paramètres de communication du contrôleur sont compatibles avec ceux du périphérique.
Réponse non valide	Le contrôleur a envoyé une réponse ou un contenu non valide au périphérique.	Vérifiez la cohérence entre les données programmées dans le projet et les ressources du contrôleur.
Erreur générale	Erreur inconnue. Ne doit jamais être signalée.	Contactez l'assistance technique.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

ABB Modbus TCP

La mise en œuvre spécifique du pilote ABB Modbus TCP facilite le traitement des connexions des contrôleurs ABB, assurant une prise en charge spécifique des modèles d'API et des fonctions d'importation de variables.

Il est possible de connecter divers périphériques compatibles TCP Modbus aux périphériques IHM. Pour configurer votre périphérique TCP Modbus, consultez la documentation reçue avec ce périphérique.

L'implantation du protocole s'effectue en tant que client TCP Modbus.

Détails sur l'implantation

Le pilote ABB Modbus TCP prend seulement en charge un sous-ensemble des codes fonction du Modbus TCP standard.

Code	Fonction	Description
01	Lecture du statut de la bobine	Lecture de bits multiples dans la zone du périphérique bobine.
02	Lecture du statut d'entrée	Lecture du statut « MARCHE/ARRÊT » des entrées discrète (référence 1x) de l'esclave
03	Lecture des registres de maintien	Lecture de registres multiples
04	Lecture des registres d'entrée	Lecture du contenu binaire des registres d'entrée (référence 3x) de l'esclave
05	Forcer bobine seule	Force une bobine unique à passer en état « MARCHE » ou « ARRÊT »
06	Prédéfinir registre seul*	Prédéfinit une valeur dans un registre
16	Prédéfinir registres multiples*	Prédéfinit une valeur dans des registres multiples

Paramètres de l'éditeur de protocole

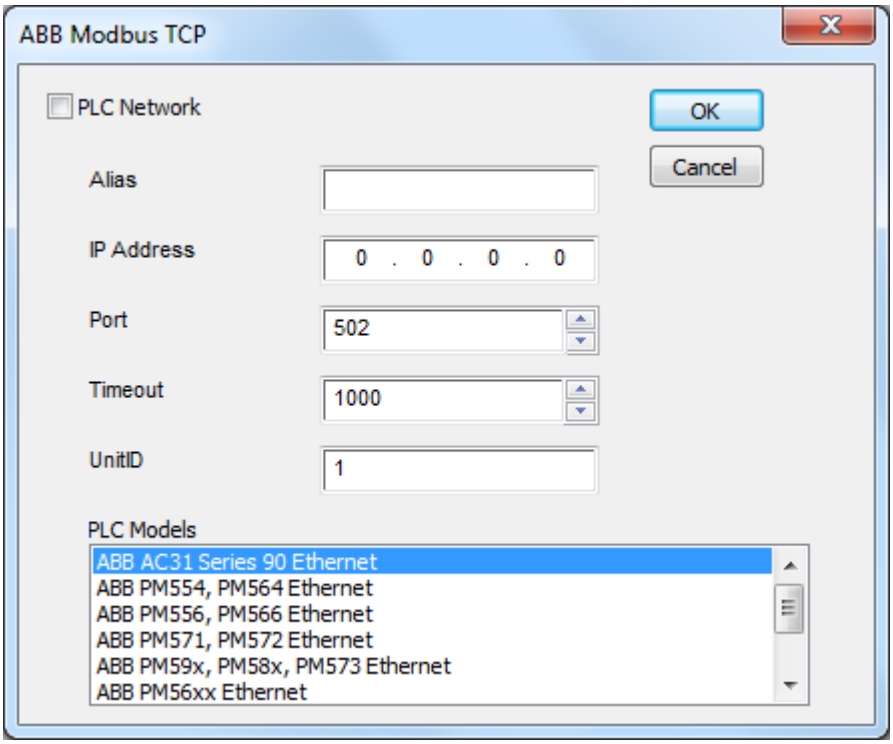
Ajouter un protocole

Pour configurer le protocole :

1. Dans le nœud **Config**, double-cliquez sur **Protocoles**.
2. Pour ajouter un pilote, cliquez sur **+** : une nouvelle ligne est ajoutée.
3. Sélectionnez le protocole dans la liste **API**.

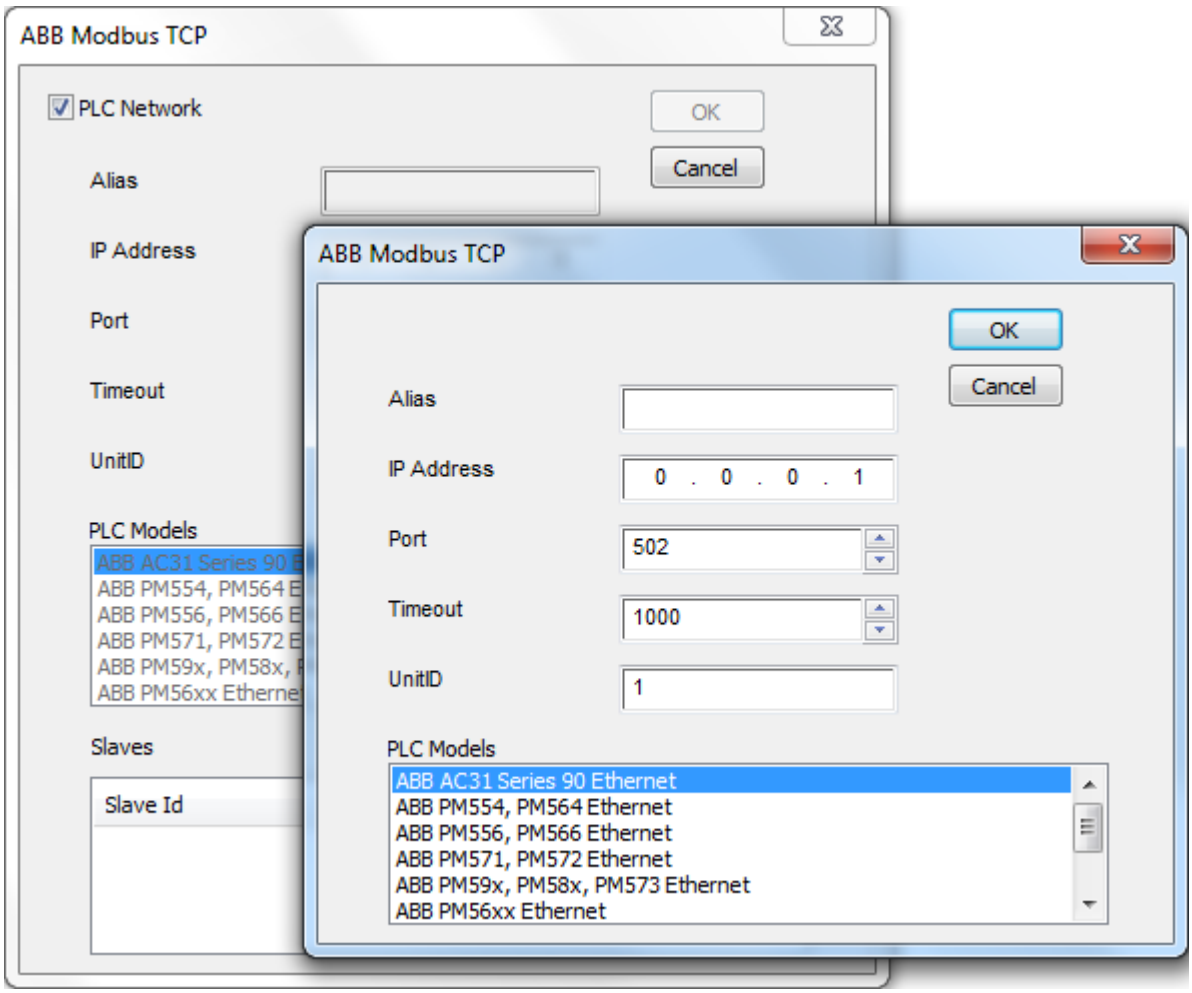
La boîte de dialogue de configuration du protocole s'affiche.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



Élément	Description
Alias	Nom d'identification des nœuds dans les configurations réseau. Il est ajouté en préfixe à chaque nom de variable importé pour chaque nœud de réseau.
IP Address	Adresse IP Ethernet du contrôleur.
Port	Numéro du port utilisé par le pilote Modbus TCP. Vous pouvez changer la valeur par défaut lorsque la communication se fait par des routeurs ou des passerelles Internet où que le numéro de port par défaut est déjà utilisé.
Délai d'attente	Délai d'attente (en millisecondes) entre deux tentatives en cas d'absence de réponse du périphérique esclave.
Identificateur d'unité	Généralement utilisé en cas de communication par passerelles Ethernet vers série, interprété comme l'ID esclave. Sa valeur est simplement copiée dans le champ Identificateur d'unité de la trame de communication du Modbus TCP. Il est rarement utilisé, vous pouvez le laisser à zéro dans la plupart des cas.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Élément	Description
Modèles d'API	Modèles d'API auxquels vous allez vous connecter. Cette sélection influence le décalage des plages de données pour chaque type de données en fonction des ressources spécifiques de mémoire d'API.
Réseau d'API	Adresse IP de tous les contrôleurs en connexions multiples. Vous devez sélectionner la case à cocher **Réseau d'API** pour permettre les connexions multiples. 

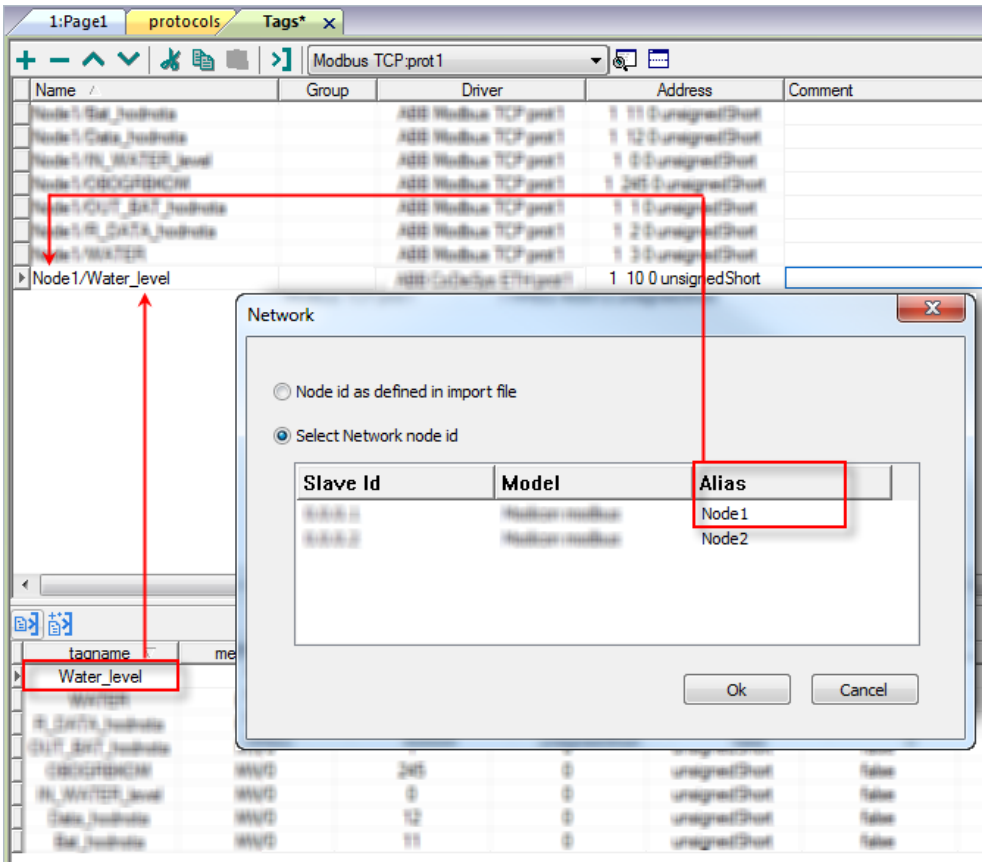
Ajouter un nom d'alias à un protocole

Les noms de variables doivent être uniques au niveau d'un projet. Vous pouvez cependant être amené à utiliser les mêmes noms de variables pour différents nœuds de contrôleur (par exemple lorsque le périphérique IHM est connecté à deux périphériques exécutant la même application).

Lorsque vous générez un protocole, vous pouvez ajouter un nom d'alias qui s'ajoutera aux noms de variables importés pour ce protocole.

Dans l'exemple, la connexion à un certain contrôleur porte le nom **Nœud1**. Lorsque vous importez les variables pour ce nœud, tous les noms de variables porteront le préfixe **Nœud1**, les rendant ainsi uniques au niveau projet/réseau.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



Remarque : vous pouvez donner un nom d'alias uniquement aux variables importées. Il n'est pas possible pour les variables ajoutées manuellement dans l'éditeur de variables de porter un préfixe d'alias dans leur nom.
La chaîne d'alias est associée à l'importation des variables. Une fois l'importation des variables effectuée, une modification de la chaîne d'alias n'aura aucun effet sur les noms figurant déjà dans le dictionnaire. Lorsque vous modifiez la chaîne d'alias et réimportez les variables, toutes les variables réimportées porteront la nouvelle chaîne de préfixe.

ID d'écrasement de nœud (périphériques maîtres)

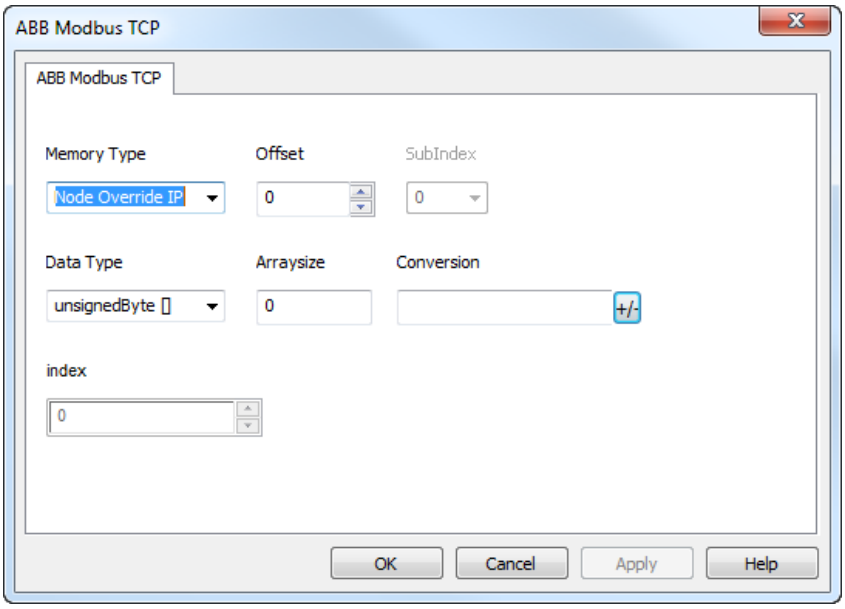
Le protocole propose le type de données spécial ID d'écrasement de nœud, qui vous permet de modifier l'ID de nœud du contrôleur cible au runtime. Ce type de mémoire est un octet non signé.

L'ID d'écrasement de nœud est initialisé avec la valeur de l'ID de nœud spécifiée dans le projet au moment de la programmation.

ID d'écrasement de nœud	Opération Modbus
0	La communication avec le contrôleur est coupée. En cas d'opération d'écriture, la requête sera transmise sans attente de réponse.
1 à 254	Interprété comme étant la valeur du nouvel ID de nœud et remplacé pour l'opération de runtime.
255	La communication avec le contrôleur est coupée, aucun message de requête ne sera généré.

Remarque : le maintien de la valeur de l'ID d'écrasement de nœud assignée au runtime est assuré par des cycles d'alimentation.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

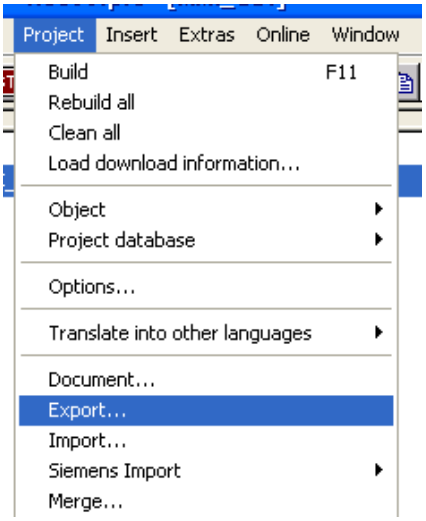


Exporter les variables du contrôleur

La programmation des contrôleurs ABB prend en charge l'exportation des variables au format .exp.

Pour exporter des variables :

Sélectionnez **Projet**> **Exporter...** : cela entraîne la création d'un fichier .exp.



Importer des variables

Vous pouvez importer les variables d'un fichier .exp exporté d'un contrôleur. Voir la section « Mon premier projet » du manuel principal.

Statut de communication

Vous pouvez afficher l'état de communication actuel grâce aux variables système. Voir la section « Variables système » du manuel principal.

Codes pris en charge pour ce pilote de communication :

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Erreur	Cause	Action
NAK	Le contrôleur répond « Ne pas confirmer ».	-
Délai d'attente	La requête n'a obtenu aucune réponse dans le délai d'attente spécifié.	Vérifiez que le contrôleur est connecté et correctement configuré pour accéder au réseau.
Réponse non valide	Le contrôleur a envoyé une réponse ou un contenu non valide au périphérique.	Vérifiez la cohérence entre les données programmées dans le projet et les ressources du contrôleur.
Erreur générale	Erreur non identifiable.	Contactez l'assistance technique.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

ABB Pluto

The HMI devices can be connected to a Modbus network as the network master using this generic driver.

This specific implementation of the Modbus RTU driver provides easy handling of the connections to the ABB controllers providing specific support for ABB Pluto Safety PLC and tag import facilities.

Implementation details

This Modbus RTU implementation supports only a subset of the standard Modbus function codes.

Code	Function	Description
01	Read Coil Status	Reads multiple bits in the device Coil area
02	Read Input Status	Read the ON/OFF status of the discrete inputs (1x reference) in the slave
03	Read Holding Registers	Read multiple Registers
04	Read Input Registers	Reads the binary contents of input registers (3x reference) in the slave
05	Force Single Coil	Forces a single Coil to either ON or OFF
06	Preset Single Register	Presets a value in a Register
16	Preset Multiple Registers	Presets value in multiple Registers

Note: Communication speed with controllers is supported up to 115200 baud.

Note: Floating point data format is IEEE standard compliant.

Protocol Editor Settings

Ajouter un protocole

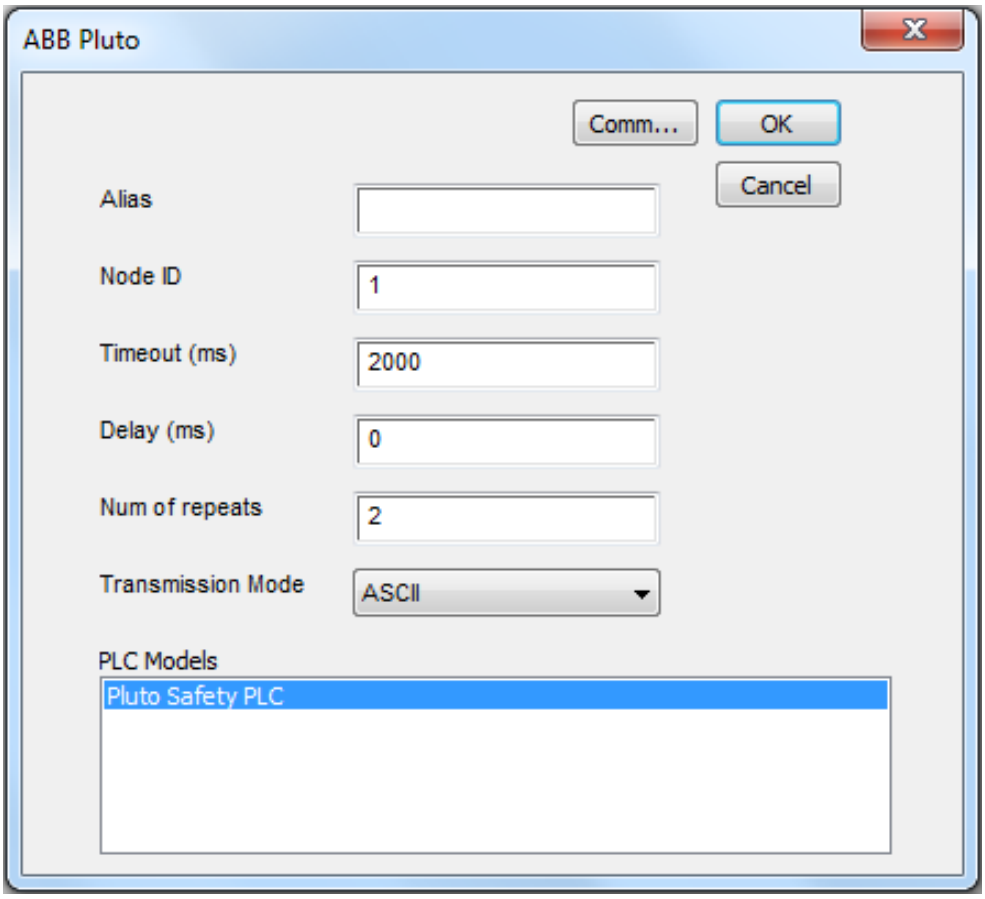
Pour configurer le protocole :

1. Dans le nœud **Config**, double-cliquez sur **Protocoles**.
2. Pour ajouter un pilote, cliquez sur **+** : une nouvelle ligne est ajoutée.
3. Sélectionnez le protocole dans la liste **API**.

La boîte de dialogue de configuration du protocole s’affiche.

The driver configuration dialog is shown in the following figure.

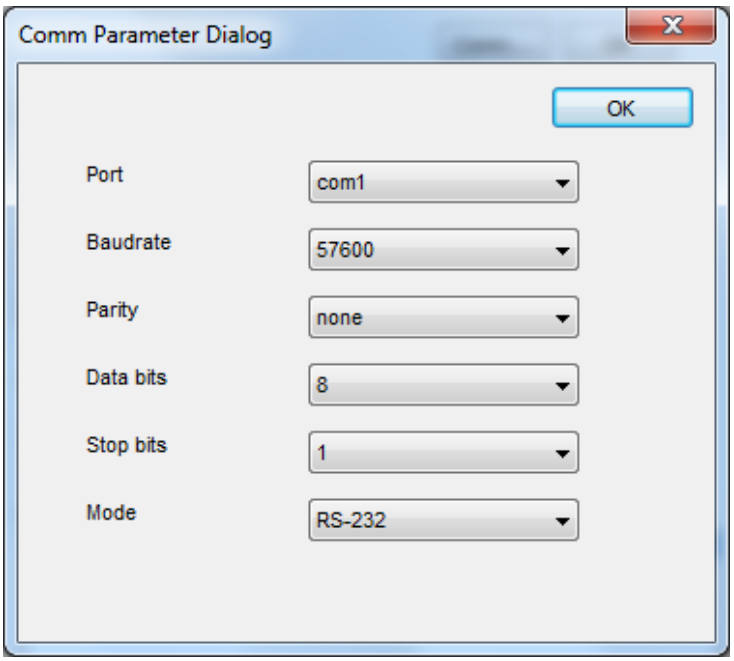
Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



Element	Description
Alias	Name identifying nodes in network configurations. The name will be added as a prefix to each tag name imported for each network node.
Node ID	Modbus node of the slave device.
Timeout (ms)	Time delay in milliseconds between two retries in case of missing response from the slave device.
Delay (ms)	Time delay in milliseconds between the end of the last received frame and the starting of a new request. If set to 0, the new request will be issued as soon as the internal system is able to reschedule it.
Num of repeats	Number of times a certain message will be sent to the controller before reporting the communication error status. When set to 1 the panel will report the communication error if the response to the first request packet is not correct.
Transmission Mode	• **RTU**: use RTU mode • **ASCII**: use ASCII mode

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Element	Description
PLC Models	PLC model you are going to connect to. The selection influences the data range offset per each data type according to the specific PLC memory resources.
Comm…	If clicked displays the communication parameters setup dialog.



Element	Description
Port	Serial port selection. • **COM1**: device PLC port. • **COM2**: computer/printer port (if available)
Baudrate, Parity, Data bits, Stop bits	Serial line parameters.
Mode	Serial port mode. Available modes: • **RS-232**. • **RS-485** (2 wires). • **RS-422** (4 wires).

ID d'écrasement de nœud (périphériques maîtres)

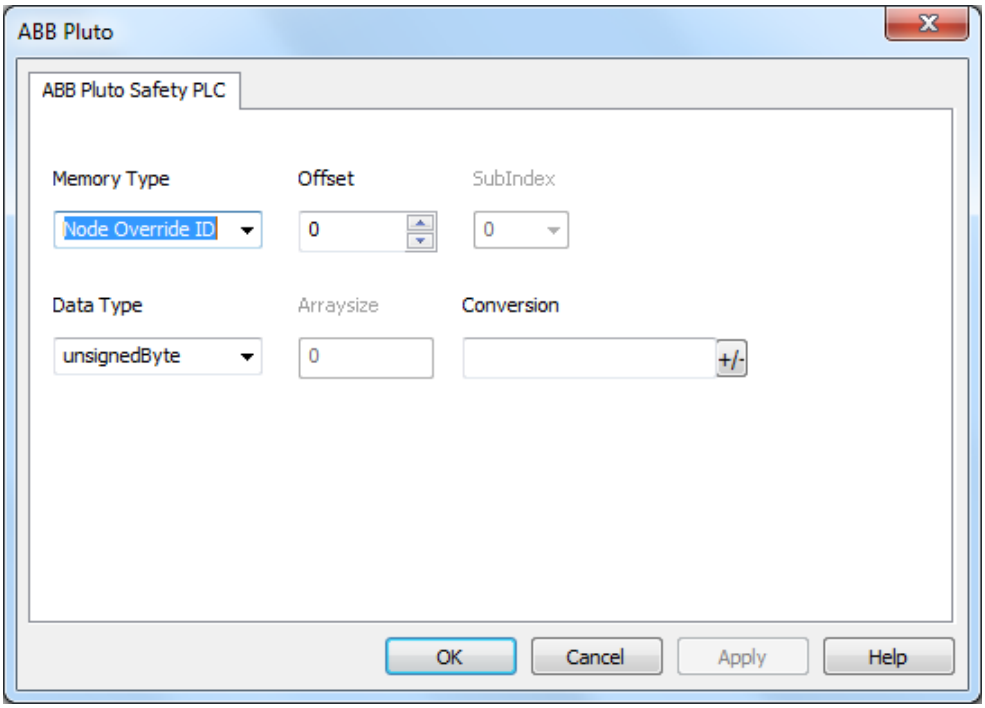
Le protocole propose le type de données spécial ID d'écrasement de nœud, qui vous permet de modifier l'ID de nœud du contrôleur cible au runtime. Ce type de mémoire est un octet non signé.

L'ID d'écrasement de nœud est initialisé avec la valeur de l'ID de nœud spécifiée dans le projet au moment de la programmation.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

ID d'écrasement de nœud	Opération Modbus
0	La communication avec le contrôleur est coupée. En cas d'opération d'écriture, la requête sera transmise sans attente de réponse.
1 à 254	Interprété comme étant la valeur du nouvel ID de nœud et remplacé pour l'opération de runtime.
255	La communication avec le contrôleur est coupée, aucun message de requête ne sera généré.

Remarque : le maintien de la valeur de l'ID d'écrasement de nœud assignée au runtime est assuré par des cycles d'alimentation.

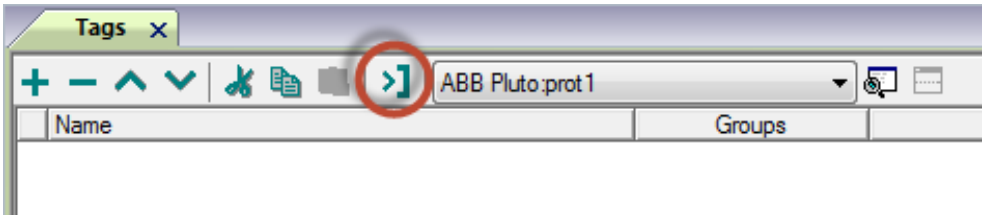


Tag import

ABB Pluto driver supports tag import.

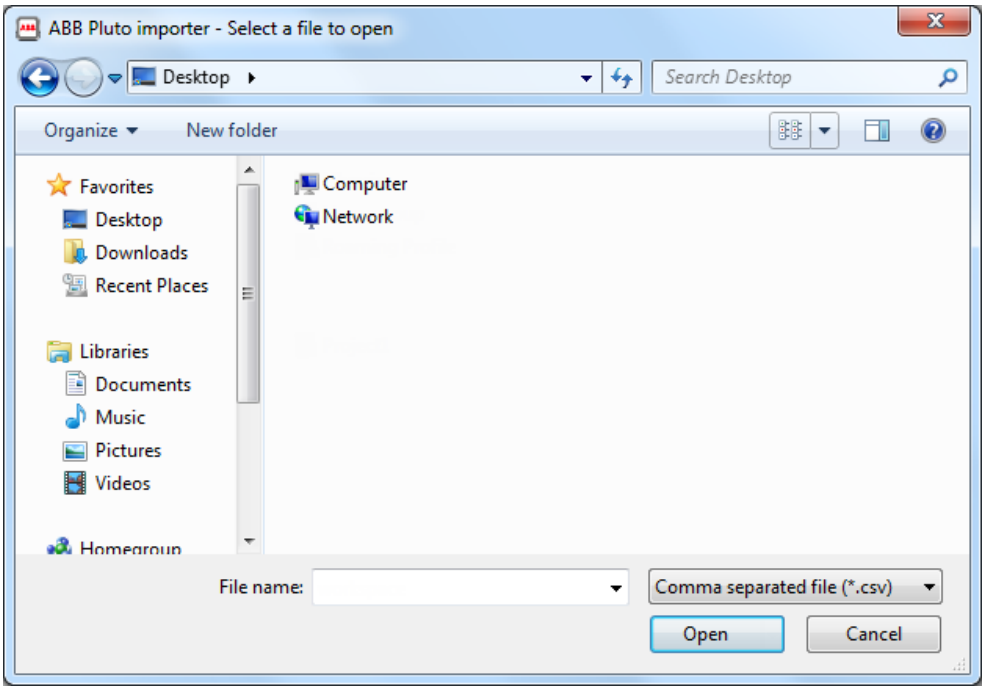
The ABB Pluto Safety PLC programming suite allows to export tags in .csv format.

1. In the Tag Editor select the driver.
2. Click the **Import Tags** button to start the import.

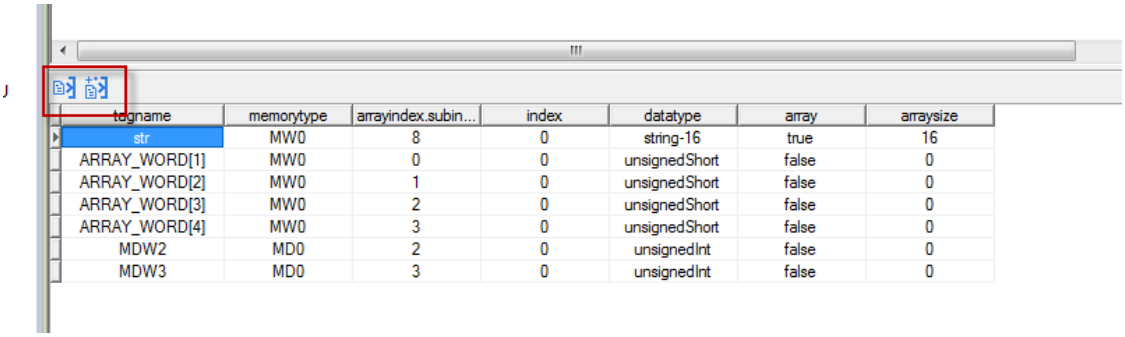


3. Locate the .csv file and confirm.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



4. To import tags, select one or more tags in the .csv file and click the **Import tag** button: tags are copied to the project.



tagname	memorytype	arrayindex.subin...	index	datatype	array	arraysize
str	MW0	8	0	string-16	true	16
ARRAY_WORD[1]	MW0	0	0	unsignedShort	false	0
ARRAY_WORD[2]	MW0	1	0	unsignedShort	false	0
ARRAY_WORD[3]	MW0	2	0	unsignedShort	false	0
ARRAY_WORD[4]	MW0	3	0	unsignedShort	false	0
MDW2	MD0	2	0	unsignedInt	false	0
MDW3	MD0	3	0	unsignedInt	false	0

See "My first project" section in the main manual.

Communication status

Current communication status can be displayed using System Variables. See "System Variables" section in the main manual.

Codes supported for this communication driver:

Error	Cause	Action
NAK	The controller replies with a not acknowledge.	-
Timeout	A request is not replied within the specified timeout period.	Check if the controller is connected and properly configured to get network access.
Line Error	An error on the communication parameter setup is detected (parity, baud rate, data bits, stop	Check if the communication parameter settings of the controller is compatible with the device communication

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Error	Cause	Action
	bits).	setup.
Invalid response	The device did received a response with invalid format or contents from the controller .	Ensure the data programmed in the project are consistent with the controller resources.
General Error	Unidentifiable error. Should never be reported.	Contact technical support.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

BACnet

Le pilote de communication BACnet a été conçu pour connecter les périphériques IHM aux réseaux BACnet. Il prend en charge la communication IP et MS/TP.

Le périphérique IHM agit alors comme un périphérique BACnet.

Paramètres de l'éditeur du protocole

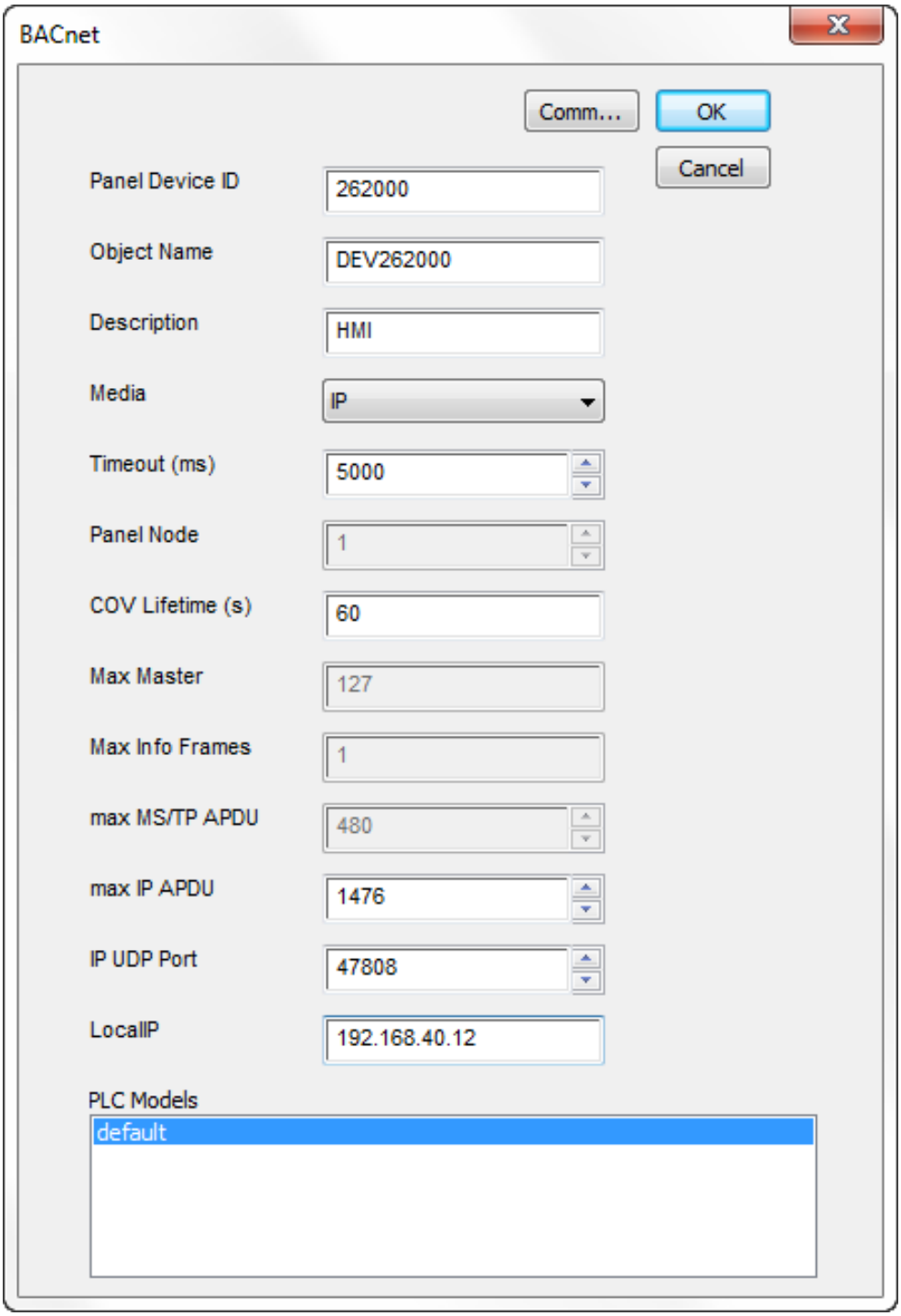
Ajouter un protocole

Pour configurer le protocole :

1. Dans le nœud **Config**, double-cliquez sur **Protocoles**.
2. Pour ajouter un pilote, cliquez sur **+** : une nouvelle ligne est ajoutée.
3. Sélectionnez le protocole dans la liste **API**.

La boîte de dialogue de configuration du protocole s'affiche.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



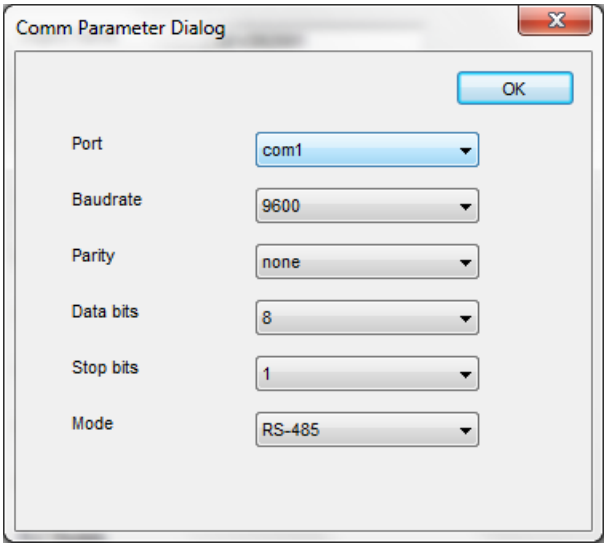
Élément	Description
ID de périphérique panneau	Identifie le périphérique IHM dans le réseau.
Nom d'objet	Nom d'objet BACnet pour le périphérique IHM.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Élément	Description
Description	Description du périphérique IHM, à des fins de documentation.
Médias	Type de communication du protocole. • **MS/TP** : Communication Maître-Esclave/Token-Passing (RS-485). • **IP** : basé sur une communication UDP/IP standard.
Délai d'attente (ms)	Délai d'attente (en millisecondes) entre deux tentatives en cas d'absence de réponse du périphérique BACnet.
Nœud de panneau *	Adresse MS/TP. Adresse du périphérique physique sur le lien, sans passage par routeurs.
Durée de vie COV (s)	Durée de vie souhaitée de l'abonnement (en secondes) avant son annulation automatique. Une valeur à zéro indique une durée de vie indéfinie, sans annulation automatique.
Maître max *	Adresse la plus haute autorisée pour les nœuds maître. Doit être inférieure ou égale à 127.
Max Info Frames *	Nombre maximal de trames d'informations que le nœud peut envoyer avant de devoir passer le jeton. Les trames d'info max peuvent avoir différentes valeurs sur différents nœuds, et peuvent être utilisées pour allouer plus ou moins de bande passante de lien disponible à certains nœuds.
ME/PJ APDU max *	Longueur maximale de l'APDU (Application Protocol Data Unit), c'est-à-dire la longueur réelle du paquet sur le réseau BACnet. Cette valeur ne peut dépasser 480 (valeur par défaut).
IP APDU max **	Longueur maximale de l'APDU (Application Protocol Data Unit), c'est-à-dire la longueur réelle du paquet sur le réseau BACnet. Cette valeur ne peut dépasser 1476 (valeur par défaut).
Port UDP/IP **	Numéro de port pour la communication IP.
IP locale **	Adresse IP de l'adaptateur réseau à utiliser pour le protocole. Non obligatoire lorsque le périphérique ne possède qu'un seul adaptateur Ethernet.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Élément	Description
Modèles d'API	Réservés à une utilisation ultérieure.
Comm... *	En cliquant dessus, vous affichez la boîte de dialogue de configuration des paramètres de communication.



Élément	Description
Port	Port de communication.
Débit en bauds, parité, bits de données, bits d'arrêt	Paramètres de communication.
Mode	Mode de communication. modes disponibles : • **RS-232** • **RS-485** • **RS-422**

Remarque * : disponible uniquement lorsque le média est défini sur **MS/TP**.

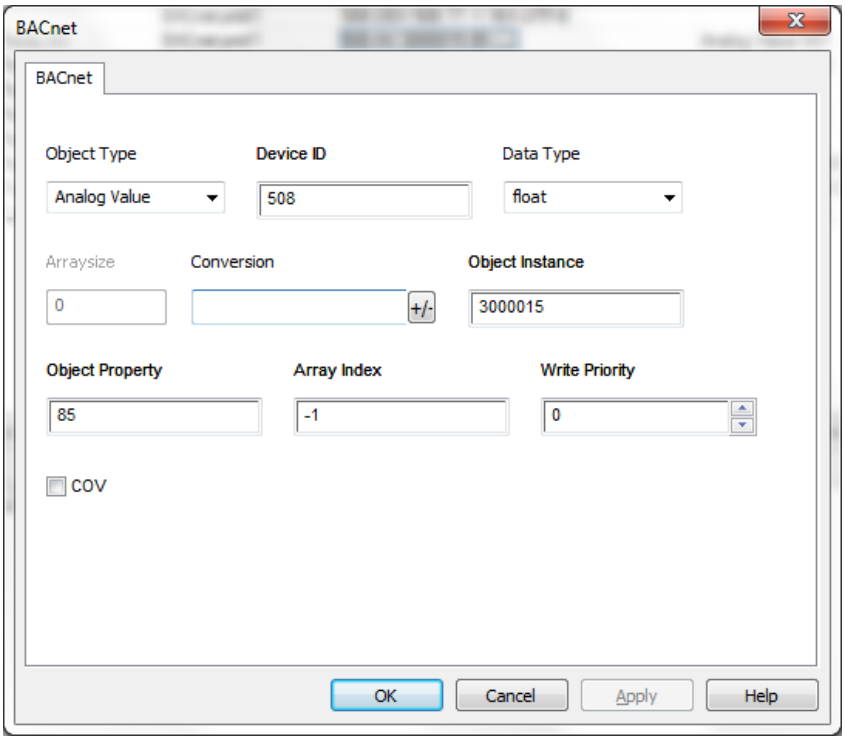
Remarque **: disponible uniquement lorsque le média est défini sur **MS/TP**.

Paramètres de l'éditeur de variables

*Chemin : **VueProjet**> **Configuration** > double-cliquez sur **Variables***

1. Pour ajouter une variable, cliquez sur **+** : une nouvelle ligne est ajoutée.
2. Dans la liste **Pilote**, sélectionnez **BACnet** : la boîte de dialogue de définition de variable s'affiche.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



Élément	Description
Object_ Type	Type d'objet BACnet à référencer. Types d'objets disponibles : • **Périphérique** • **Entrée analogique** • **Sortie analogique** • **Valeur analogique** • **Entrée binaire** • **Sortie binaire** • **Valeur binaire** • **Entrée multi-statut** • **Sortie multi-statut** • **Valeur multi-statut** • **Valeur entière** • **Valeur entière positive** • **Grande valeur analogique**
ID de périphérique	ID du périphérique contenant l'objet.
Type de données	Type de données pour l'affichage. Types de données disponibles : • **booléen**

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

<table>
<tr><th>Élément</th><th>Description</th></tr>
<tr><td></td><td>

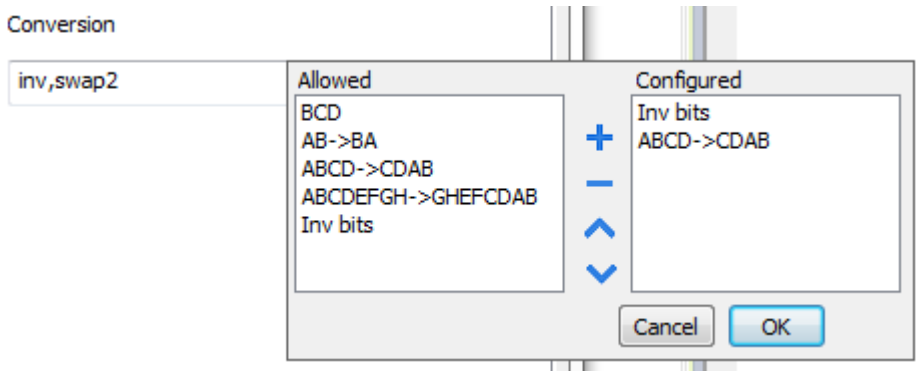
entier
entier non signé
flottant
double
chaîne
binaire
boolean[]

Ces types de données sont tels que définis dans le logiciel.
Le tableau suivant présente leur équivalence avec les types de données BACnet :
<table>
<tr><th>Type de données BACnet</th><th>Type de données logiciel</th><th>Notes</th></tr>
<tr><td>BOOLEAN</td><td>Boolean</td><td>-</td></tr>
<tr><td>INTEGER</td><td>Int</td><td>-</td></tr>
<tr><td>UNSIGNED_INTEGER</td><td>entier non signé</td><td>-</td></tr>
<tr><td>REAL</td><td>flottant</td><td>-</td></tr>
<tr><td>BIT_STRING</td><td>boolean-x</td><td>x = taille</td></tr>
<tr><td>CHARACTER_ STRING</td><td>string-x</td><td>x = taille</td></tr>
<tr><td>OCTET_STRING</td><td>binary-x</td><td>x = taille</td></tr>
<tr><td>DATE</td><td>int ou unsignedInt</td><td>-</td></tr>
<tr><td>TIME</td><td>int ou unsignedInt</td><td>-</td></tr>
<tr><td>BACnetObjectIdentifier</td><td>int ou unsignedInt</td><td>Utilise l’instance des conversions et objType pour un affichage approprié</td></tr>
</table>
</td></tr>
<tr><td>Taille de tableau</td><td>

En cas de variable vectorielle, cette propriété représente le nombre d’éléments vectoriels.
En cas de variable de chaîne, cette propriété représente le nombre maximum d’octets disponible pour la variable de chaîne.

Remarque : le nombre d'octets correspond au nombre de caractères de chaîne si la propriété Encoding est définie sur UTF-8 ou Latin1 dans l'Éditeur de variables.
Si la propriété Encoding est définie sur UCS-2BE, UCS-2LE, UTF-16BE ou UTF-16LE, un caractère nécessite 2 octets.
</td></tr>
<tr><td>Convers</td><td>Conversion à appliquer à la variable.</td></tr>
</table>

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Élément	Description
ion	 En fonction du type de données choisi, la liste **Autorisée** affiche un ou plusieurs types de conversion.

Valeur	Description
Inv bits	**inv** : inverse tous les bits de la variable. *Exemple :* 1001 → 0110 (format binaire) 9 → 6 (format décimal)
Inverser	**neg** : inverse la valeur de la variable. *Exemple :* 25.36 → -25.36
AB -> BA	**swapnibbles** : échange les quartets en un octet. *Exemple :* 15D4 → 514D (format hexadécimal) 5588 → 20813 (format décimal)
ABCD -> CDAB	**swap2** : échange les octets en un mot. *Exemple :* 9ACC → CC9A (format hexadécimal) 39628 → 52378 (format décimal)
ABCDEFGH -> GHEFCDAB	**swap4** : échange les octets en un mot double. *Exemple :* 32FCFF54 → 54FFFC32 (format hexadécimal) 855441236 → 1426062386 (format décimal)
ABC...NOP -> OPM...DAB	**swap8** : échange les octets en un mot long. Exemple : 142,366 → -893553517,588905 (format décimal) 0 10000000110

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

<table>
<tr><th>Élément</th><th>Description</th></tr>
<tr><td></td><td>
<table>
<tr><th>Valeur</th><th>Description</th></tr>
<tr><td></td><td>0001110010111011011001000101101000011100101011000001
→
1 10000011100
10101010000101000101101101101100101101100000100111101
(format binaire)</td></tr>
<tr><td>BCD</td><td>bcd : sépare l’octet en deux quartets, les lit au format décimal (de 0 à 9)

Exemple :
23 → 17 (format décimal)
0001 0111 = 23
0001 = 1 (premier quartet)
0111 = 7 (second quartet)</td></tr>
</table>

Sélectionnez la conversion et cliquez sur le bouton +. La conversion sélectionnée s’affiche dans la liste Configurée.

Si vous configurez d’autres conversions, celles-ci s’affichent dans l’ordre (du haut vers le bas de la liste Configurée).

Utilisez les boutons de direction pour changer l’ordre des conversions configurées.</td></tr>
<tr><td>Instance d'objet</td><td>ID BACnet de l’objet à référencer.</td></tr>
<tr><td>Propriété d'objet</td><td>Valeur numérique de la propriété à référencer (exemple : une valeur de 85 signifie present-value pour la plupart des objets standard).
Le tableau ci-dessus présente toutes les propriétés d’objet BACnet.
<table>
<tr><th>Propriété</th><th>Valeur</th><th>Propriété</th><th>Valeur</th><th>Propriété</th><th>Valeur</th><th>Propriété</th><th>Valeur</th></tr>
<tr><td>accepted-modes</td><td>175</td><td>effective-period</td><td>32</td><td>max-info-frames</td><td>63</td><td>reason-for-halt</td><td>100</td></tr>
<tr><td>acked-transitions</td><td>0</td><td>elapsed-active-time</td><td>33</td><td>max-master</td><td>64</td><td>recipient-list</td><td>102</td></tr>
<tr><td>ack-required</td><td>1</td><td>error-limit</td><td>34</td><td>max-pres-value</td><td>65</td><td>records-since-notification</td><td>140</td></tr>
<tr><td>action</td><td>2</td><td>event-enable</td><td>35</td><td>max-segments-</td><td>167</td><td>record-count</td><td>141</td></tr>
</table>
</td></tr>
</table>

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Élément	Description

Propriété	Valeur	Propriété	Valeur	Propriété	Valeur	Propriété	Valeur
				accepted			
action-text	3	event-state	36	member-of	159	reliability	103
active-text	4	event-time-stamps	130	minimum-off-time	66	relinquish-default	104
active-vt-sessions	5	event-type	37	minimum-on-time	67	required	105
active-cov-subscriptions	152	event-parameters	83	minimum-output	68	resolution	106
adjust-value	176	exception-schedule	38	minimum-value	136	scale	187
alarm-value	6	fault-values	39	minimum-value-timestamp	150	scale-factor	188
alarm-values	7	feedback-value	40	min-pres-value	69	schedule-default	174
all	8	file-access-method	41	mode	160	segmentation-supported	107
all-writes-successful	9	file-size	42	model-name	70	setpoint	108
apdu-segment-timeout	10	file-type	43	modification-date	71	setpoint-reference	109
apdu-timeout	11	firmware-revision	44	notification-class	17	slave-address-binding	171
application-software-version	12	high-limit	45	notification-threshold	137	paramètre	162

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Élément	Description

Propriété	Valeur	Propriété	Valeur	Propriété	Valeur	Propriété	Valeur
archive	13	inactive-text	46	notify-type	72	silenced	163
attempted-samples	124	in-process	47	number-of-APDU-retries	73	start-time	142
auto-slave-discovery	169	input-reference	181	number-of-states	74	state-text	110
average-value	125	instance-of	48	object-identifier	75	status-flags	111
backup-failure-timeout	153	integral-constant	49	object-list	76	stop-time	143
bias	14	integral-constant-units	50	object-name	77	stop-when-full	144
buffer-size	126	last-notify-record	173	object-property-reference	78	system-status	112
change-of-state-count	15	last-restore-time	157	object-type	79	time-delay	113
change-of-state-time	16	life-safety-alarm-values	166	operation-expected	161	time-of-active-time-reset	114
client-cov-increment	127	limit-enable	52	optional	80	time-of-state-count-reset	115
configuration-files	154	limit-monitoring-interval	182	out-of-service	81	time-synchronization-recipients	116
controlled-variable-reference	19	list-of-group-members	53	output-units	82	total-record-count	145

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

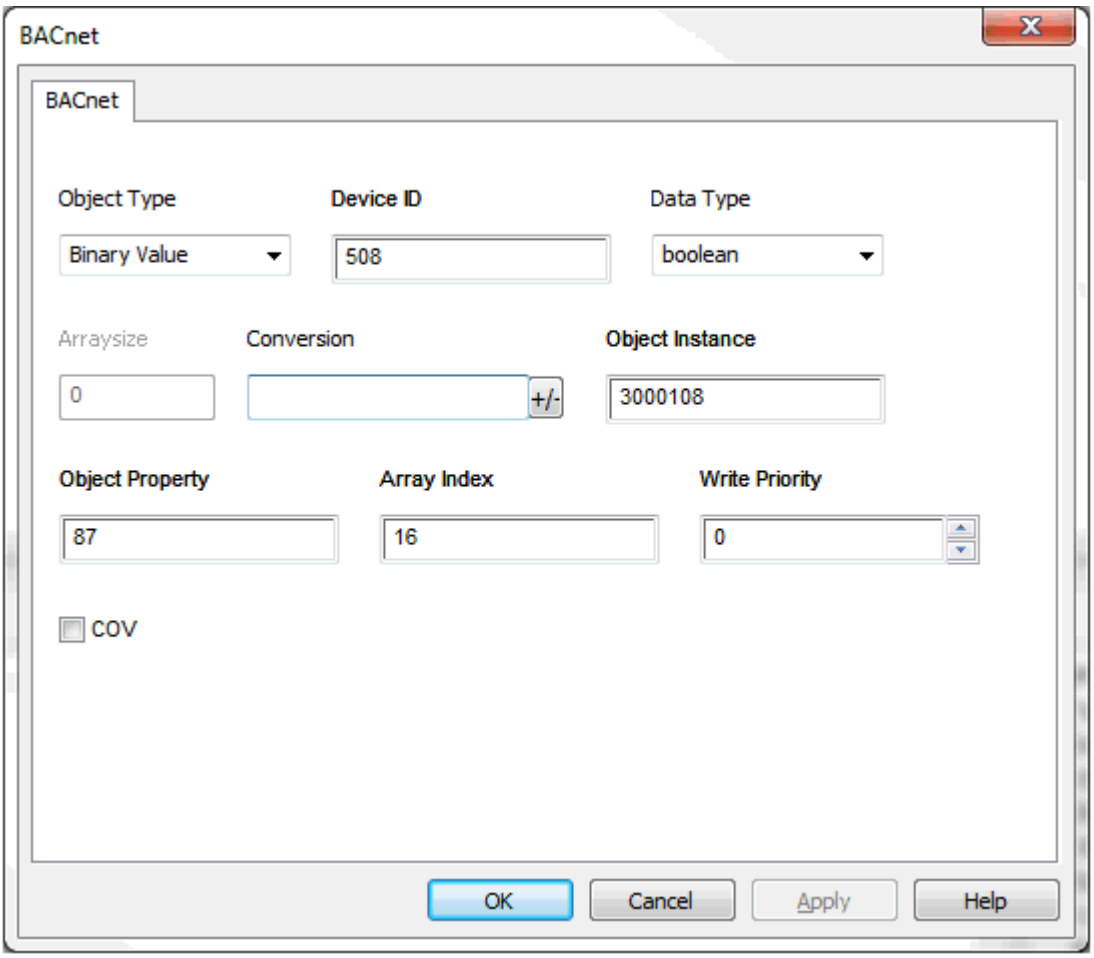
Élément	Description

Propriété	Valeur	Propriété	Valeur	Propriété	Valeur	Propriété	Valeur
controlled-variable-units	20	list-of-object-property-references	54	polarity	84	tracking-value	164
controlled-variable-value	21	list-of-session-keys	55	prescale	185	units	117
count	177	local-date	56	present-value	85	update-interval	118
count-before-change	178	local-time	57	Priorité	86	update-time	189
count-change-time	179	location	58	pulse-rate	186	utc-offset	119
cov-increment	22	log-buffer	131	priority-array	87	valid-samples	146
cov-period	180	log-device-object-property	132	priority-for-writing	88	value-before-change	190
cov-resubscription-interval	128	log-enable	133	process-identifier	89	value-set	191
database-revision	155	log-interval	134	profile-name	168	value-change-time	192
date-list	23	logging-object	183	program-change	90	variance-value	151
daylight-savings-status	24	logging-record	184	program-location	91	vendor-identifier	120
deadband	25	low-limit	59	program-state	92	vendor-name	121
derivative-	26	maintena	158	proportio	93	vt-classes-	122

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

<table>
<tr><th>Élément</th><th>Description</th></tr>
<tr><td></td><td>
<table>
<tr><th>Propriété</th><th>Valeur</th><th>Propriété</th><th>Valeur</th><th>Propriété</th><th>Valeur</th><th>Propriété</th><th>Valeur</th></tr>
<tr><td>constant</td><td></td><td>nce-required</td><td></td><td>nal-constant</td><td></td><td>supported</td><td></td></tr>
<tr><td>derivative-constant-units</td><td>27</td><td>manipulated-variable-reference</td><td>60</td><td>proportional-constant-units</td><td>94</td><td>weekly-schedule</td><td>123</td></tr>
<tr><td>description</td><td>28</td><td>manual-slave-address-binding</td><td>170</td><td>protocol-object-types-supported</td><td>96</td><td>window-interval</td><td>147</td></tr>
<tr><td>description-of-halt</td><td>29</td><td>maximum-output</td><td>61</td><td>protocol-revision</td><td>139</td><td>window-samples</td><td>148</td></tr>
<tr><td>device-address-binding</td><td>30</td><td>maximum-value</td><td>135</td><td>protocol-services-supported</td><td>97</td><td>zone-members</td><td>165</td></tr>
<tr><td>device-type</td><td>31</td><td>maximum-value-timestamp</td><td>149</td><td>protocol-version</td><td>98</td><td></td><td></td></tr>
<tr><td>direct-reading</td><td>156</td><td>max-apdu-length-accepted</td><td>62</td><td>read-only</td><td>99</td><td></td><td></td></tr>
</table>
</td></tr>
<tr><td>Array Index</td><td>Index permettant de soumettre des éléments dans des tableaux BACnet.
-1 signifie lire tous les éléments
0 à n signifie lire un élément spécifique
Exemple de tableau de priorité
Pour lire un objet de tableau de priorité, il est nécessaire de définir la propriété de l’objet = 87 et l’index du tableau doit se référer à l'article de la priorité à lire.
La figure suivante montre comment lire le 16ème objet d'un tableau de priorité.</td></tr>
</table>

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Élément	Description
	
Définition de priorité	Définit un niveau de priorité des requêtes. L valeur est comprise entre 1 et 16. 16correspond à 0.
COV	Active la notification COV (Modifier la valeur).

Effacer/Définir une priorité

Le système permet différentes actions permettant une prise en main plus facile de la définition de priorité.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Action	Description
BACnetClearPriority (effacer propriété BACnet)	Efface le tableau de priorité à la position associée à la variable BACnet configurée en tant que paramètre. Cette action a un effet immédiat sur le périphérique BACnet.
BACnetClearAllPriorities (effacer toutes propriétés BACnet)	Efface toutes les priorités du tableau. Cette action a un effet immédiat sur le périphérique BACnet.
BACnetSetPriority (Définir propriété BACnet)	Écrase la valeur de définition de priorité configurée dans la définition de variable BACnet. Cette action a deux paramètres : • **nomVariable** : nom de la variable BACnet. • **prioritéVariable** : nouvelle valeur de la définition de priorité pour la variable BACnet configurée en tant que paramètre. Cette action écrase uniquement la valeur de la définition de priorité dans la définition de variable BACnet. Elle n'effectue aucune communication avec le périphérique BACnet. Toute commande d'écriture qui sera effectuée dans la propriété valeur présente du périphérique BACnet identifié par la variable, sera exécutée en utilisant la nouvelle valeur de priorité d'écriture. La valeur de priorité reste valide : • Jusqu'à un nouvel appel de l'action BACnetSetPriority qui la modifie. • Le périphérique IHM a redémarré. La valeur de la priorité en écriture dans le projet est valide dans ce cas.

Importation de variables

Les informations d'objet BACnet peuvent être importées à partir de fichiers BACnet EDE (échange de données d'ingénierie). Le fichier EDE doit avoir l'extension .csv.

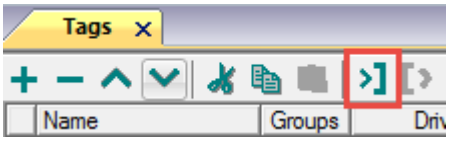
L'importateur utilise les caractères « , » et « ; » comme délimiteurs. Ils sont considérés comme des caractères réservés et vous ne pouvez pas les utiliser dans le nom de fichier.

Utilisez le pilote d'importation hiérarchique pour avoir une liste ordonnée d'objets et de propriétés BACnet.

Les variables seront créés en utilisant la chaîne spécifiée dans l'objet nom de la colonne du fichier EDE. Le pilote d'importation ajoutera l'ID du périphérique comme préfixe pour éviter la duplication des noms de variables.

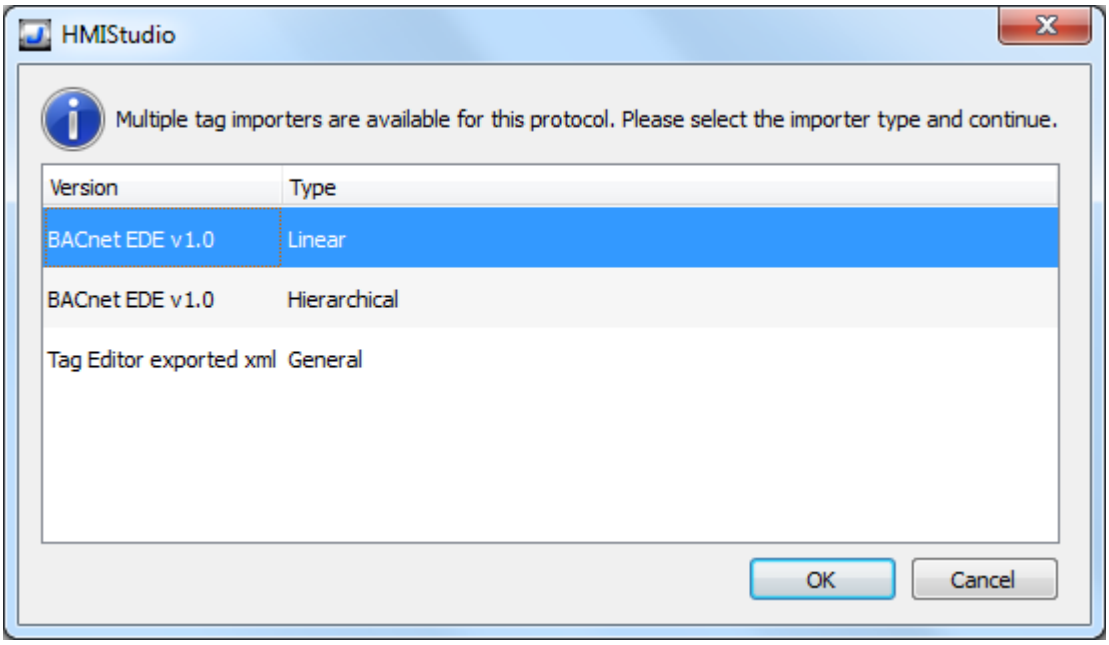
Remarque : le pilote d'importation demandera de localiser les fichiers State-Texts, Unit-Texts et Object-Types. Cliquez sur Annuler pour ignorer.

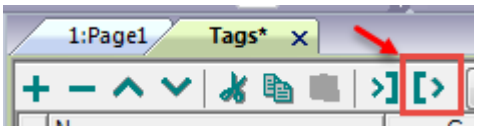
Sélectionnez le pilote dans l'éditeur de variables et cliquez sur le bouton **Importer variables** pour lancer le pilote d'importation.



Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

La boîte de dialogue suivante affiche le type d'importation que vous pouvez sélectionner.

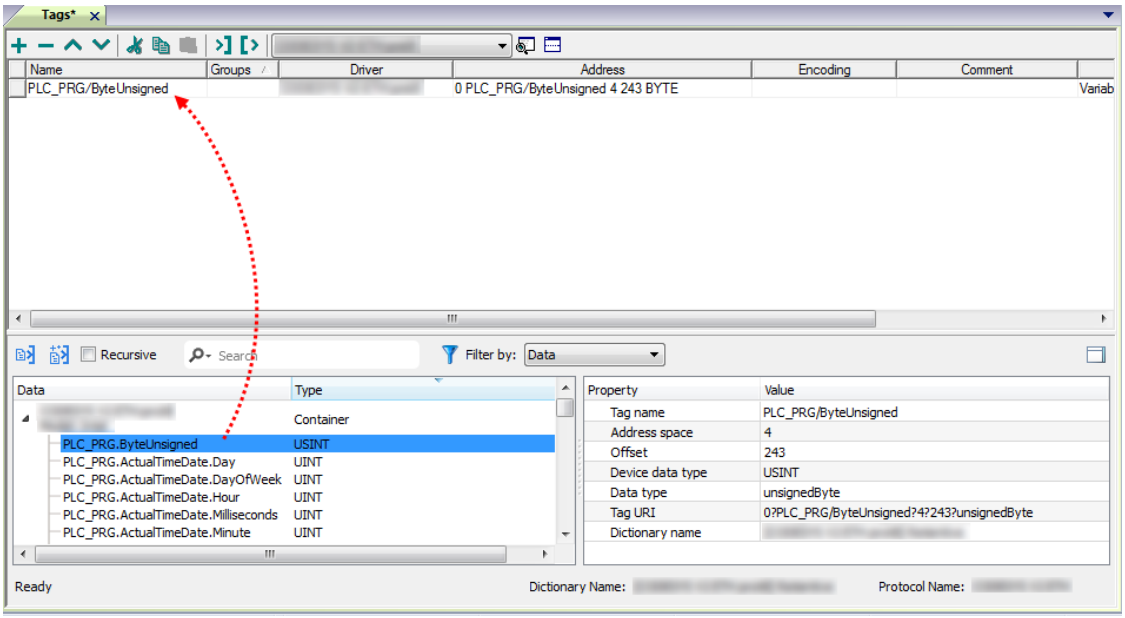


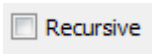
Importateur	Description
BACnet EDE v1.0 Linéaire	Nécessite un fichier **.csv**. Toutes les variables s'affichent au même niveau.
BACnet EDE v1.0 Hiérarchique	Nécessite un fichier **.csv**. Toutes les variables s'affichent en fonction de l'affichage hiérarchique BACnet EDE.
Fichier .xml exporté de l'éditeur de variables	Sélectionnez ce pilote d'importation pour lire un fichier XML générique exporté de l'éditeur de variables avec le bouton approprié. 

Lorsque vous avez sélectionné le pilote d'importation, localisez le fichier de symboles et cliquez sur **Ouvrir**.

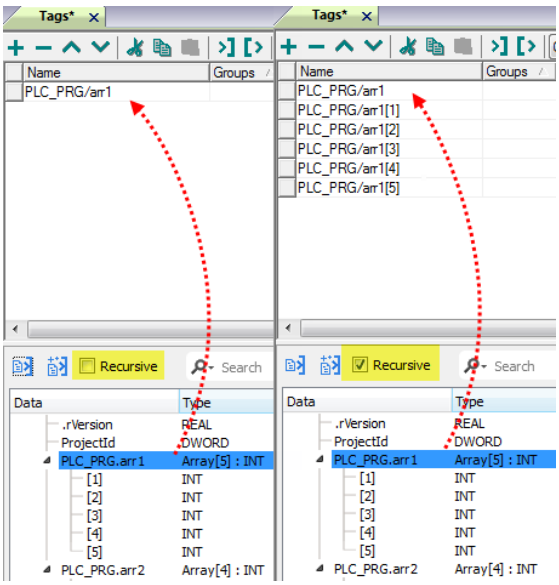
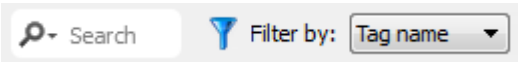
Les variables comprises dans le fichier de symboles sont listées dans le dictionnaire des variables. Le dictionnaire des variables est affiché en bas de l'écran.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



Élément de la barre d'outils	Description
	Importer variable(s). Sélectionnez les variables à importer et cliquez sur cette icône pour ajouter les variables du dictionnaire au projet.
	Mettre à jour variable(s). Cliquez sur cette icône pour mettre à jour les variables du projet grâce à un nouvel import du dictionnaire.
	Cochez cette case pour importer tous les sous-éléments d'une variable. Exemples de résultats avec la case cochée et non cochée :

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Élément de la barre d'outils	Description
	
	Recherche des variables dans le dictionnaire en filtrant en fonction de l'élément sélectionné dans la liste déroulante.

Pour les variables faisant référence à des objets BACnet de type Calendrier ou Ordonnancement, le taux de rafraîchissement de la variable est réglé sur « Manuel ».

Les propriétés de l'objet BACnet suivantes sont nécessaires pour le fonctionnement des widgets.

Objet	Variables à importer
Calendrier	Date_List
Ordonnancer	Weekly_Schedule Exception_Ordonnancement Valeur par défaut Effective_Period

Propriétés d'objet du PÉRIPHÉRIQUE

Un scanneur réseau BACnet peut détecter les propriétés en explorant le réseau et en obtenant des données du périphérique IHM.

Voici les propriétés d'objet du PÉRIPHÉRIQUE prises en charge :

Propriété	Description
Object_Identifier	BACnetObjectIdentifier
Object_Name	CharacterString

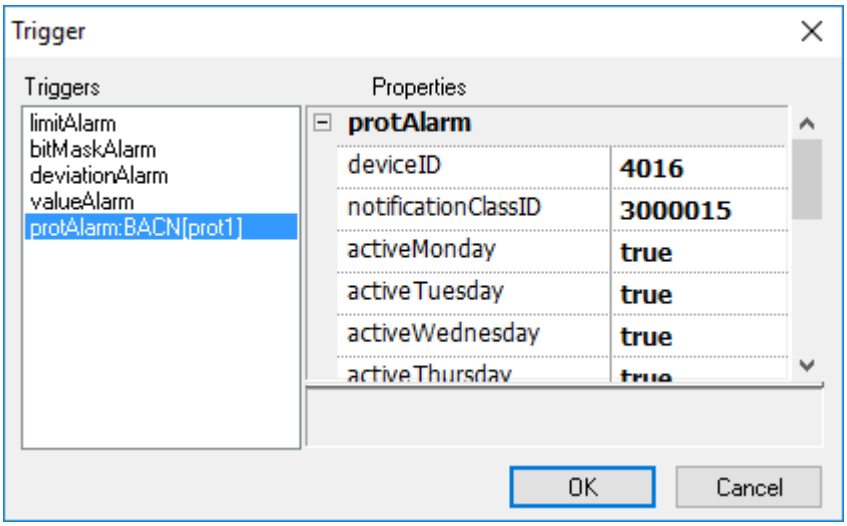
Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Propriété	Description
Object_Type	BACnetObjectType
System_Status	BACnetDeviceStatus
Vendor_Name	CharacterString
Vendor_Identifier	Unsigned16
Model_Name	CharacterString
Firmware_Revision	CharacterString
Application_Software_Version	CharacterString
Protocol_Version	Unsigned
Protocol_Revision	Unsigned
Protocol_Services_Supported	BACnetServicesSupported
Protocol_Object_Types_Supported	BACnetObjectTypesSupported
Object_List	BACnetARRAY[N]of BACnetObjectIdentifier
Max_APDU_Length_Accepted	Unsigned
Segmentation_Supported	BACnetSegmentation
APDU_Timeout	Unsigned
Number_Of_APDU_Retries	Unsigned
Device_Address_Binding	List of BACnetAddressBinding
Database_Revision	Unsigned

Événement d'alarme BACnet

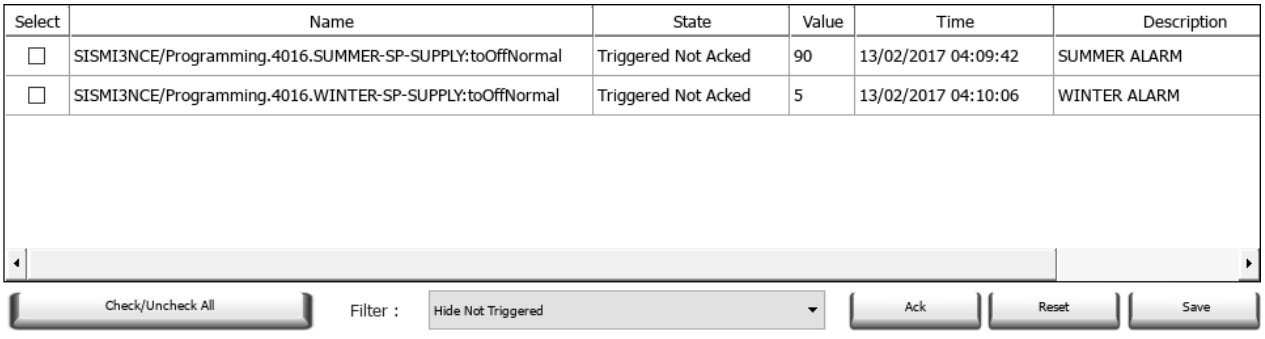
Le mode de déclenchement spécial « protAlarm: BACN », disponible dans l'Éditeur d'alarmes, permet de recevoir des événements d'alarme à partir du module natif d'alarmes BACnet.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



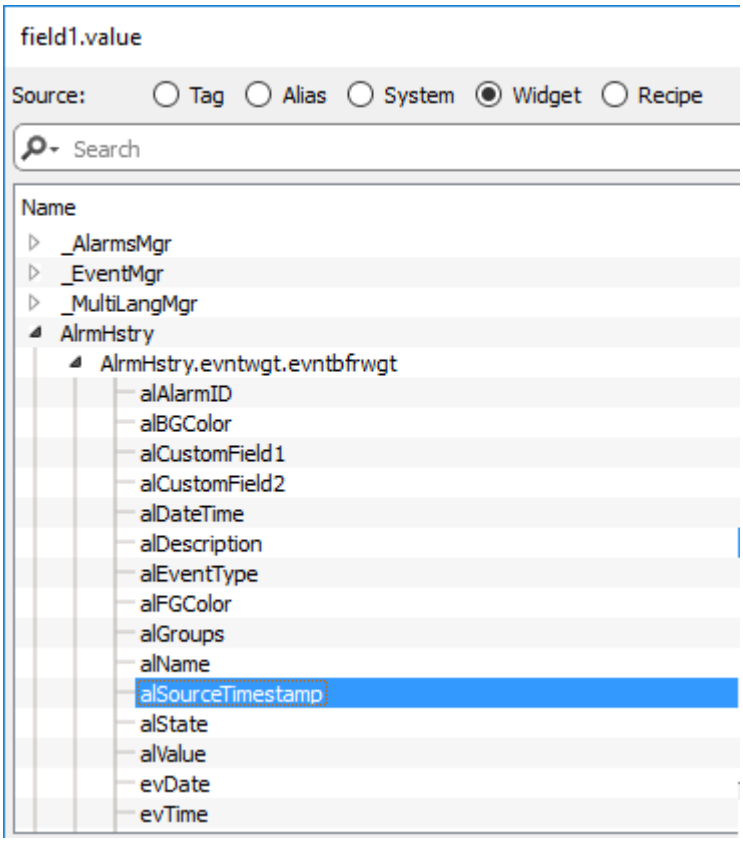
Propriété	Description
deviceID	Identifie le périphérique BACnet sur le réseau.
notificationClassID	ID de classe de notification pour vous abonner au journal des événements d'alarme
processID	Non utilisé
activeMonday **activeTuesday** **activeWednesday** **activeThrusday** **activeFriday** **activeSaturday** **activeSunday**	Définissez les jours où l'abonnement aux événements d'alarme doit rester actif. • faux Abonnement inactif • vrai Abonnement actif
startHour **startMinute** **startSecond** **endHour** **endMinute** **endSecond**	Définissez l'intervalle de temps pendant lequel l'abonnement aux événements d'alarme sera actif

Les vignettes d'alarme signaleront les informations d'alarme fournies à partir du périphérique BACnet.



Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Lorsque le mode de déclenchement spécial "protAlarm:BACN" est utilisé, le widget des alarmes actives affiche l'horodatage fourni par le dispositif BACnet tandis que le widget de l'historique des alarmes affiche l'horodatage du moment où les événements d'alarme sont reçus du dispositif IHM. Généralement, les deux horodatages sont identiques mais si vous avez besoin d'afficher l'horodatage du dispositif BACnet même à l'intérieur du widget de l'historique des alarmes, vous pouvez ajouter une nouvelle colonne configurée pour utiliser la valeur "alSourceTimestamp" du widget de l'historique des alarmes.



L'alarme BACnet est une alarme spéciale qui nécessite un double espace à l'intérieur du tampon d'événements. Cela signifie, par exemple, si le tampon d'événements est configuré pour contenir 1.000 événements, seuls les 500 derniers événements BACnet seront stockés.

Widget Calendrier BACnet

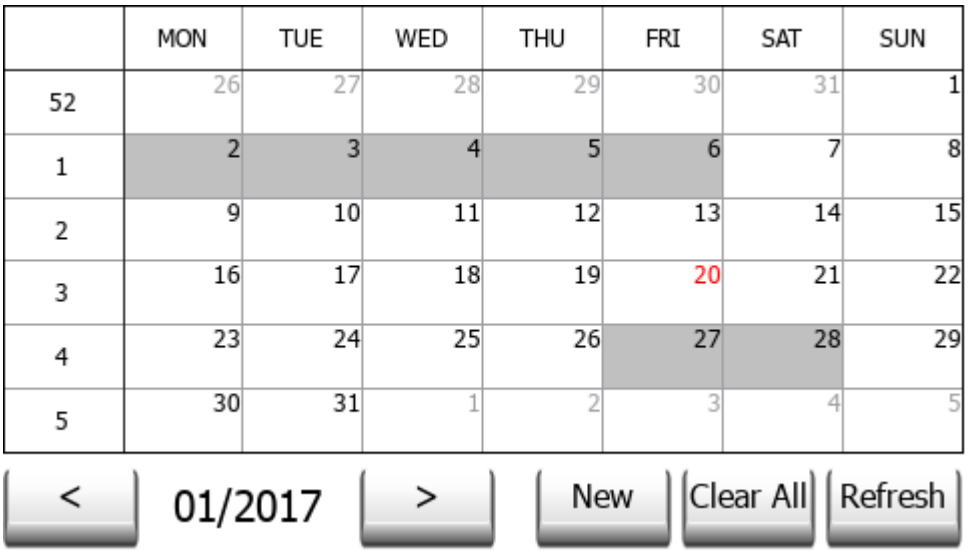
Utiliser le widget Calendar pour afficher le contenu d'un objet calendrier BACnet.

Propriété	Description
Date_List	Connectez-vous à la variable « Date_List » d'un objet de calendrier BACnet en ReadOnly ou en Lecture / écriture. Remarque : il peut être connecté à un alias qui indexe une liste de Date_List(s) de calendrier BACnet, dans le but d'utiliser un widget calendrier pour plus d'un objet de calendrier.

Fonctionnement du widget calendrier

Le widget affiche les données pendant un mois.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

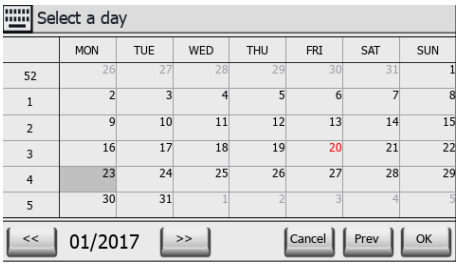
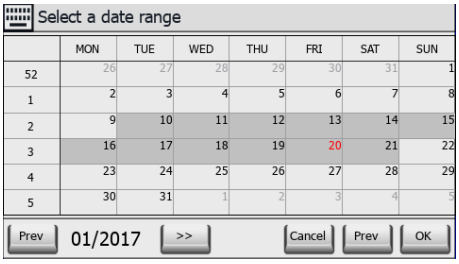


Utiliser les touches < et > pour sélectionner le mois à afficher. La date du premier jour du mois est affichée.

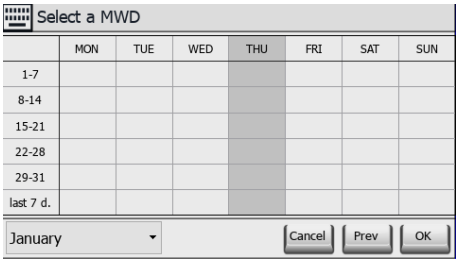
Le geste d'oscillation peut être utilisé sur le widget pour sélectionner la date.

Nouveau

Appuyer sur le bouton « Nouveau » pour entrer un nouvel élément de calendrier. Le bouton est actif uniquement si la variable associée au calendrier a été configurée en tant que lecture / écriture.

Élément calendrier	Description
Individuel	Cliquer sur un jour pour sélectionner un seul jour dans le calendrier 
Plage	Cliquer sur le premier jour et le dernier jour pour sélectionner une plage de jours dans le calendrier. • Un seul clic sur une journée pour modifier le précédent dernier jour sélectionné de la gamme. • Double-clic sur une journée pour modifier le précédent premier jour sélectionné de la gamme. 

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Élément calendrier	Description
MWD	Sélectionner un jour ou une semaine pour chaque année ou chaque mois. 

Tout effacer

Appuyer sur le bouton « Tout effacer » pour effacer le contenu du calendrier. Le bouton est actif uniquement si la variable associée au calendrier a été configurée en lecture /écriture. Le bouton est configuré pour réagir à un événement LorsdeMaintienBoutonSouris, pour réduire le risque de perte de données.

Rafraîchir

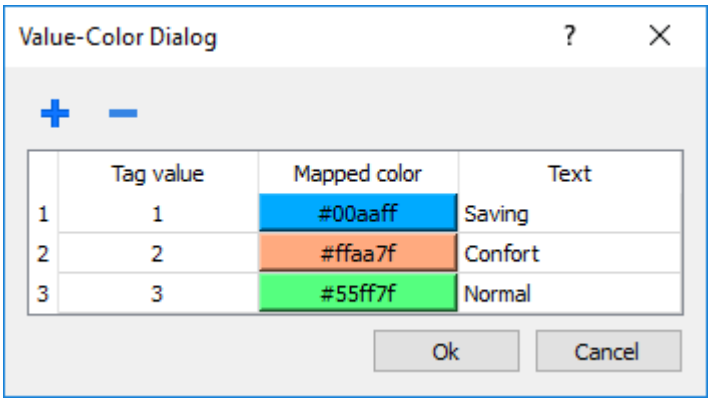
Appuyer sur le bouton « Actualiser » pour lancer une actualisation manuelle des données du widget. Toujours appuyer sur le bouton Actualiser après la saisie des données dans le calendrier.

Widget ordonnancement BACnet

Utiliser le widget ordonnancement pour afficher le contenu d'un objet ordonnancement BACnet.

Propriété	Description
Type	Sélectionnez le type d'objet BACnet contrôlé par l'ordonnancement. Les options sont : • binaire • Real • Multicouche
Weekly_Schedule	Attacher à la variable Weekly_Schedule de l'objet d'ordonnancement. La variable peut être Lecture seule ou Lecture / écriture.
Exception_ Ordonnancement	En option, attachez à la variable Exception_Schedule de l'objet d'ordonnancement. La variable peut être en lecture seule ou en lecture/écriture. Ne joignez cette propriété que si des exceptions sont utilisées.
Default_Value	En option, attachez à la variable Default_Value de l'objet d'ordonnancement. La variable peut être en lecture seule ou en lecture/écriture. Ne joignez cette propriété que si des valeurs par défaut sont utilisées.
Cal. 0 (Date_List)	En option, attachez à la variable Date_List du widget d'ordonnancement en mode Lecture seule. Utilisez cette option pour afficher les exceptions « de référence du calendrier ».

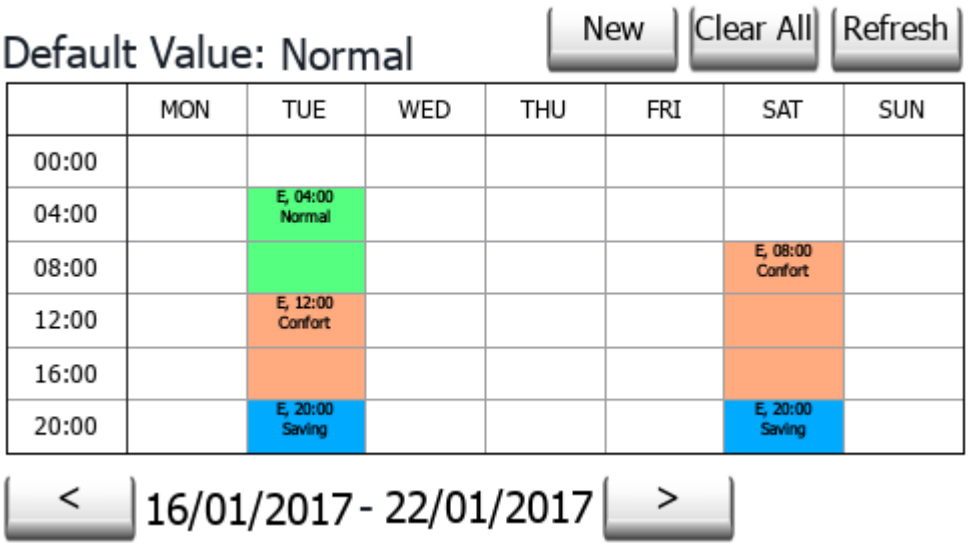
Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Propriété	Description
	Remarque : une exception peut être une date unique, une plage de dates, un mwd ou une référence de calendrier. Dans ce dernier cas, l'exception_list ne contient pas les informations de date, mais seulement de priorité-valeur-temporelle et une référence au calendrier. La date_list nécessaire pour afficher la programmation dans le widget est stockée dans le BACNCalendar relative, et c'est la raison pour laquelle nous avons besoin de cette liaison de données. S'il n'est pas nécessaire d'afficher des exceptions de calendrier dans le calendrier, cette propriété peut être laissée vide. Remarque : si cela n'est pas attaché à un calendrier, il est impossible d'insérer l'exception dans le calendrier. Voir BACNSchedKeypad pour plus de détails.
Cal. 0 (Object_Name)	Éventuellement joindre à la propriété du calendrier. Ce nom est utilisé pour identifier le calendrier dans le BACNSchedKeypad utilisé pour insérer des exceptions de calendrier. Si Object_Name n'est pas fixé, le calendrier est identifié avec son numéro d'instance. Cette propriété est utilisée uniquement si un Cal. 0 (Date_List) est fixé à un calendrier.
Cal. 1 (Date_List)	Option pour un second calendrier.
Cal. 1 (Object_Name)	Option pour un second calendrier.
Carte Valeur-couleur-texte	Définit la valeur de l'association – Couleur / texte affiché dans le tableau. Utilisez cette option pour définir toutes les valeurs possibles disponibles dans le clavier BACNSched. 

Fonctionnement du widget Ordonnancement

Le widget affiche les données pendant une semaine.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

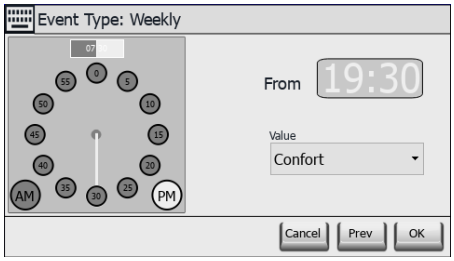
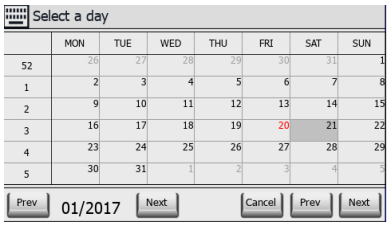
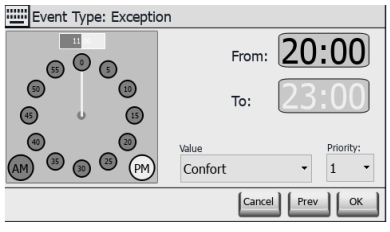


Utilisez les touches < et > pour sélectionner la semaine à afficher. La date du premier jour et du dernier jour de la semaine est affichée.

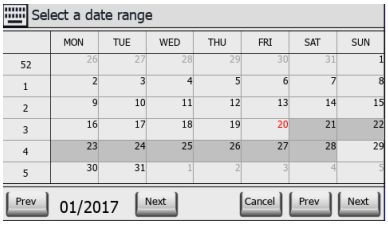
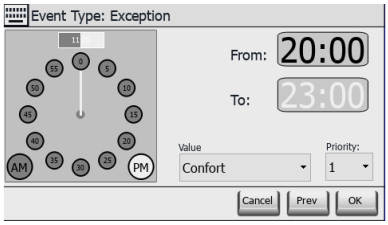
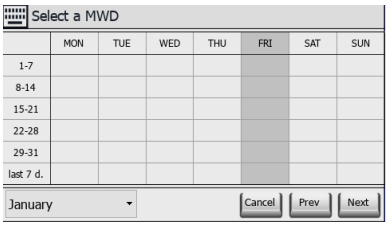
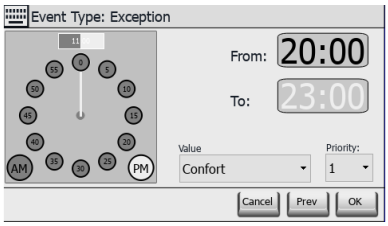
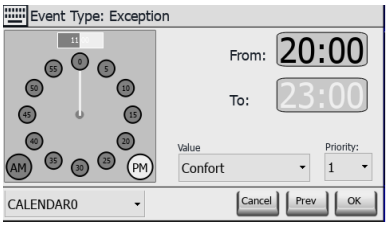
Le geste d'oscillation peut être utilisé sur le widget pour sélectionner la date.

Nouveau

Appuyez sur la touche « Nouveau » pour saisir un nouvel élément de planification. Le bouton n'est actif que si la variable associée au calendrier hebdomadaire ou au calendrier des exceptions a été configurée en lecture/écriture.

Élément tableau	Description
Hebdomadaire	Sélectionnez le jour et cliquez sur le bouton Hebdomadaire, la boîte de dialogue suivante s'affiche. Puis, sélectionnez la valeur souhaitée et le moment où il doit être réglé. 
Exception Individuel	Cliquer sur un jour pour sélectionner un seul jour dans le calendrier. Sur la boîte de dialogue suivante, sélectionnez la fenêtre de temps, la valeur désirée et sa priorité.  

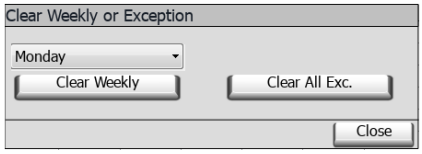
Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Élément tableau	Description
Exception Plage	Cliquer sur le premier jour et le dernier jour pour sélectionner une plage de jours dans le calendrier. • Un seul clic sur une journée pour modifier le précédent dernier jour sélectionné de la gamme. • Double-clic sur une journée pour modifier le précédent premier jour sélectionné de la gamme. Sur la boîte de dialogue suivante, sélectionnez la fenêtre de temps, la valeur désirée et sa priorité.  
Exception MWD	Sélectionner un jour ou une semaine pour chaque année ou chaque mois. Sur la boîte de dialogue suivante, sélectionnez la fenêtre de temps, la valeur désirée et sa priorité.  
Exception Cal Réf	Cette option est disponible uniquement si l'ordonnancement est lié à un calendrier (configuré en tant que lecture / écriture) Sélectionnez la fenêtre horaire, la valeur souhaitée et sa priorité. La valeur sera définie pour tous les jours définis à partir du calendrier. Si d'autres calendriers sont associés au widget Ordonnancement, sélectionnez le calendrier à utiliser. 

Tout effacer

Appuyer sur le bouton « Tout effacer » pour effacer le contenu de l'objet du calendrier. La touche n'est active que si la variable associée au calendrier a été configurée en lecture/écriture. La touche est configurée pour réagir aux événements onMouseClick et onMouseHold. L'événement onMouseHold effacera toutes les données du calendrier. L'événement onMouseClick rappellera une boîte de dialogue pour la sélection des données à effacer. Il est nécessaire de choisir d'effacer les données hebdomadaires ou les données d'exception.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



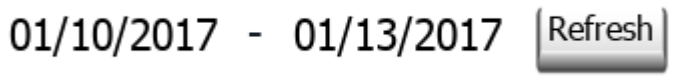
Refresh

Appuyer sur le bouton « Actualiser » pour lancer une actualisation manuelle des données du widget. Toujours appuyer sur le bouton Actualiser après la saisie des données dans le planificateur.

Widget Période effective BACnet

Utilisez le Widget Période effective pour fournir les informations à la variable Effective_Period d'un objet de l'ordonnancement, si cela est demandé.

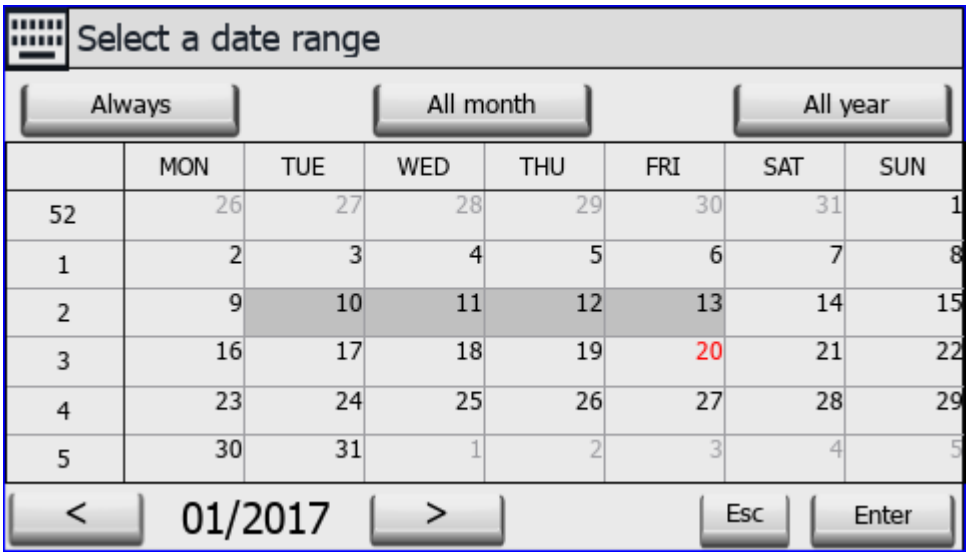
Propriété	Description
Période effective BACnet	Joinder à la variable Effective_Period de l'objet de l'ordonnancement



Fonctionnement du Widget de la période effective

Le widget indique la date de début et la date de fin de la période.

Cliquer sur la zone indiquant les dates pour activer la procédure de saisie des données montrant le clavier BACNDateRange.



Le clavier affiche les données pendant un mois.

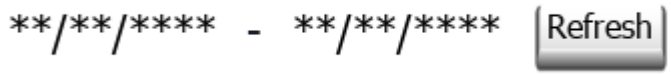
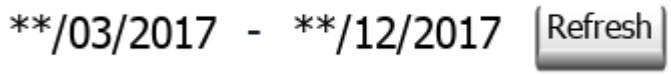
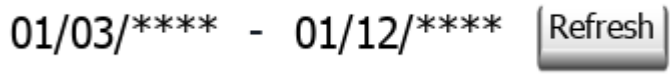
Utiliser les touches < et > pour sélectionner le mois à afficher. La date du premier jour du mois est affichée.

Vous pouvez utiliser le geste d'oscillation sur le widget pour sélectionner la date.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Sélectionnez la période en cliquant le premier jour et le dernier jour de la période. Effective_Period s'affiche avec une couleur différente.

Le clavier offre trois options prédéfinies :

Option	Description
Toujours	L'ordonnancement sera toujours actif. 
Tous les mois	La période sélectionnée sera étendue à tous les mois. 
Toute l'année	La période sélectionnée sera étendue tous les ans. 

Rafraîchir

Appuyer sur le bouton « Actualiser » pour lancer une actualisation manuelle des données du widget. Toujours appuyer sur le bouton Actualiser après la saisie des données dans le widget.

Claviers BACnet

Les widgets BACnet requièrent des claviers dédiés pour la saisie des données.

Clavier	Description
BACNCal	Clavier pour le calendrier BACnet.
BACNDateRange	Clavier pour BACnet Effective_Period.
BACNDefVal	Clavier pour la valeur par défaut (intégrée dans l'ordonnancement BACnet).
BACNSched	Clavier pour l'ordonnancement BACnet. Ce clavier est sensible au contexte. Il montrera différentes options en fonction du type d'ordonnancement.

Le système est configuré pour rappeler le clavier approprié pour chaque widget BACnet.

Statut de communication

Vous pouvez afficher l'état de communication actuel grâce aux variables système. Voir la section « Variables système » du manuel principal.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Codes pris en charge par ce pilote de communication :

Erreur	Cause
ID du périphérique introuvable	Impossible d'établir la communication avec l'ID du périphérique fournie pour cette variable.
Type de propriété de données illisible	Impossible de déterminer le type de propriété à définir.
erreur dans la conversion d'écriture	Échec d'une conversion associée à cette variable.
Impossible d'écrire le type d'ICOM …. BACnet type ….	Un type de données sélectionné pour cette variable n'est pas compatible avec la propriété BACnet à configurer.
Délai d'attente pour la souscription COV dépassé	Le délai d'attente pour la demande de souscription COV pour cette variable a été dépassé.
Délai d'attente pour la mise à jour COV dépassé	Aucune notification COV n'a été reçue pour cette variable dans le délai d'attente.
Impossible d'obtenir COV pour cette propriété	La propriété sélectionnée pour la notification COV n'est pas prise en charge.
erreur dans la conversion d'un élément du datagramme	Échec de conversion associée à une variable faisant partie d'un datagramme.
Délai d'attente de réponse dépassé	Aucune réponse obtenue dans le délai d'attente pour une demande de lecture ou d'écriture de propriété.
élément de datagramme …., aucune donnée disponible	Aucune donnée disponible pour une variable faisant partie d'un datagramme.
élément de datagramme …., type de données BACnet non pris en charge	L'élément de datagramme de lecture fait partie d'un type BACnet non pris en charge.
élément de datagramme …., impossible de convertir le type BACnet en ….	Un type de données sélectionné pour une variable faisant partie d'un datagramme n'est pas compatible avec la propriété BACnet à lire.
Aucune donnée en réponse	Aucune donnée disponible pour une variable.
Élément de datagramme 'element_URI' error: 'error_class': error_code	La lecture d'un élément de datagramme indiqué 'element_URL' a été reportée comme une erreur. Les descriptions de l'erreur **error_class** et **error_code** sont incluses dans le message.
l'objet de datagramme ne correspond pas	L'objet de l'élément de datagramme reçu ne correspond pas à l'objet demandé.
la propriété de datagramme ne correspond pas	La propriété de l'élément de datagramme reçue ne correspond pas à la propriété demandée.
Abandon BACnet : reason_of_abort	Un message d'abandon de BACnet a été reçu. La raison de l'abandon est donnée.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Erreur	Cause
Rejet BACnet : raison_of_ rejection	Un message de rejet de BACnet a été reçu. La raison du rejet est donnée.
Erreur BACnet : error_class : error_code	Un message d'erreur de BACnet a été reçu. La description de l'erreur est une combinaison d'**error_class** et d'**error_code**.
paramètre « parameter_ name » hors de portée	La valeur du paramètre de protocole **parameter_name** est hors de portée.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

CODESYS V2 ETH

Le pilote de communication CODESYS V2 ETH pour communication de supports via une connexion Ethernet avec des contrôleurs basés sur la version CODESYS V2.3.

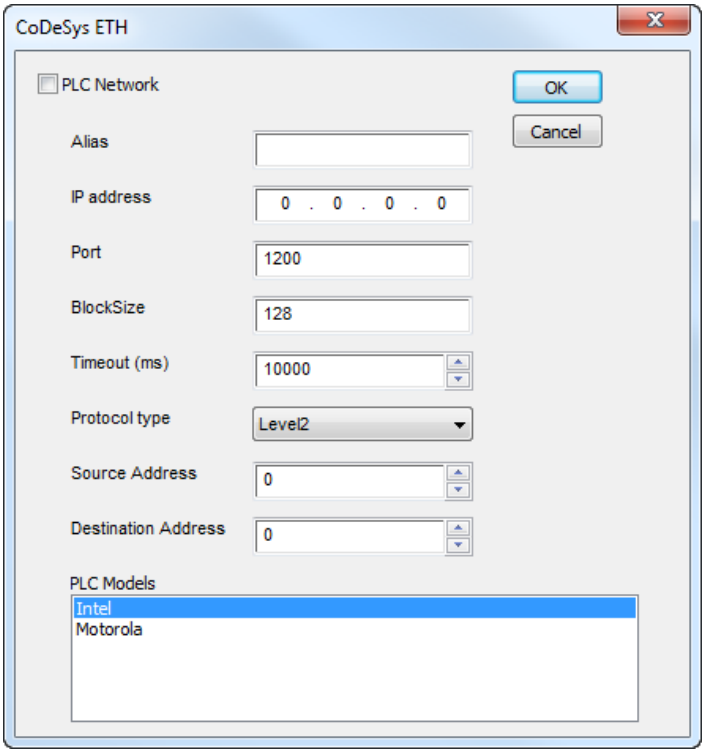
Paramètres de l'éditeur du protocole

Ajouter un protocole

Pour configurer le protocole :

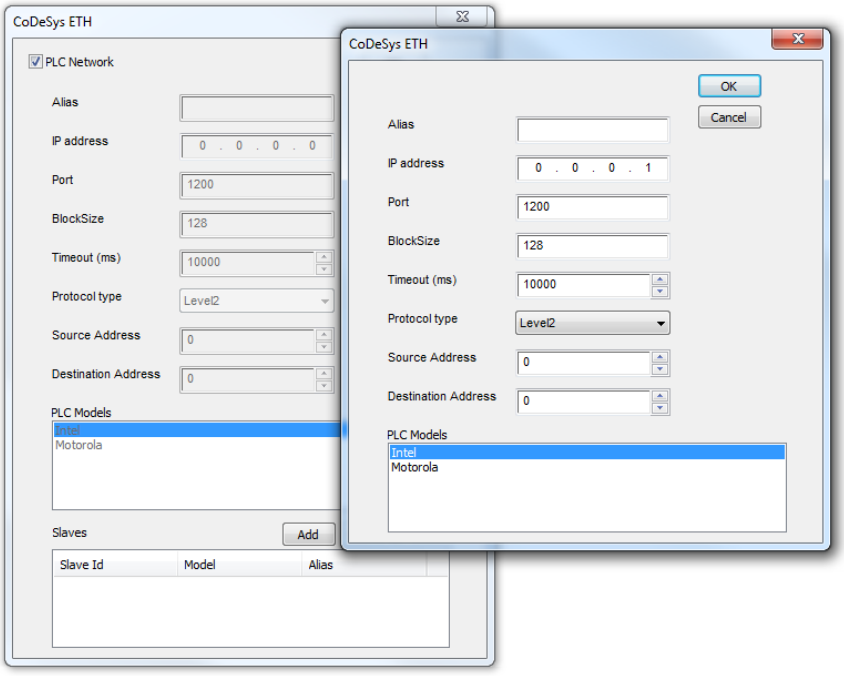
1. Dans le nœud **Config**, double-cliquez sur **Protocoles**.
2. Pour ajouter un pilote, cliquez sur **+** : une nouvelle ligne est ajoutée.
3. Sélectionnez le protocole dans la liste **API**.

La boîte de dialogue de configuration du protocole s'affiche.



Élément	Description
Alias	Nom d'identification des nœuds dans les configurations réseau. Il est ajouté en préfixe à chaque nom de variable importé pour chaque nœud de réseau.
Adresse IP	Adresse IP Ethernet du contrôleur.
Port	Numéro du port utilisé par le pilote CODESYS V2 Ethernet. Par défaut, la valeur est configurée à **1200**, ce qui correspond au paramètre par défaut des contrôleurs basés sur CODESYS.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Élément	Description
Taille de bloc	Taille de bloc maximale prise en charge par votre contrôleur (limitée à 1 024 KB).
Délai d'attente (ms)	Délai d'attente entre deux tentatives d'envoi du même message en cas d'échec de communication.
Type de protocole	Variante du protocole à utiliser. Veuillez vérifier quelle variante du protocole est prise en charge par le runtime CODESYS à laquelle vous souhaitez vous connecter.
Adresse source, adresse de destination	Disponible uniquement si l'option **Route TCP/IP de niveau 2** a été sélectionné pour le paramètre **Type de protocole**. La destination est le nœud de l'API qui permet au protocole de lire les variables d'un sous-réseau. L'adresse est utilisée pour la lecture de variables lorsque plusieurs API sont connectés à un sous-réseau (réseau en série) mais qu'un seul est doté d'interface Ethernet.
Modèles d'API	Deux modèles d'API sont disponibles. • **Intel** • **Motorola**
Réseau d'API	Adresse IP de tous les contrôleurs en connexions multiples. Vous devez sélectionner la case à cocher **Réseau d'API** pour permettre les connexions multiples. 

À partir de la version V1.60, le pilote CODESYS V2 Ethernet prend en charge la connexion à plusieurs contrôleurs.

Remarque : le pilote CODESYS V2 Ethernet est recommandé lors de la création de projets pour le contrôleur interne iPLC CODESYS. Pour utiliser le pilote CODESYS V2 Ethernet avec iPLC, configurez l'adresse IP de l'API comme hôte local (127.0.0.1).

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

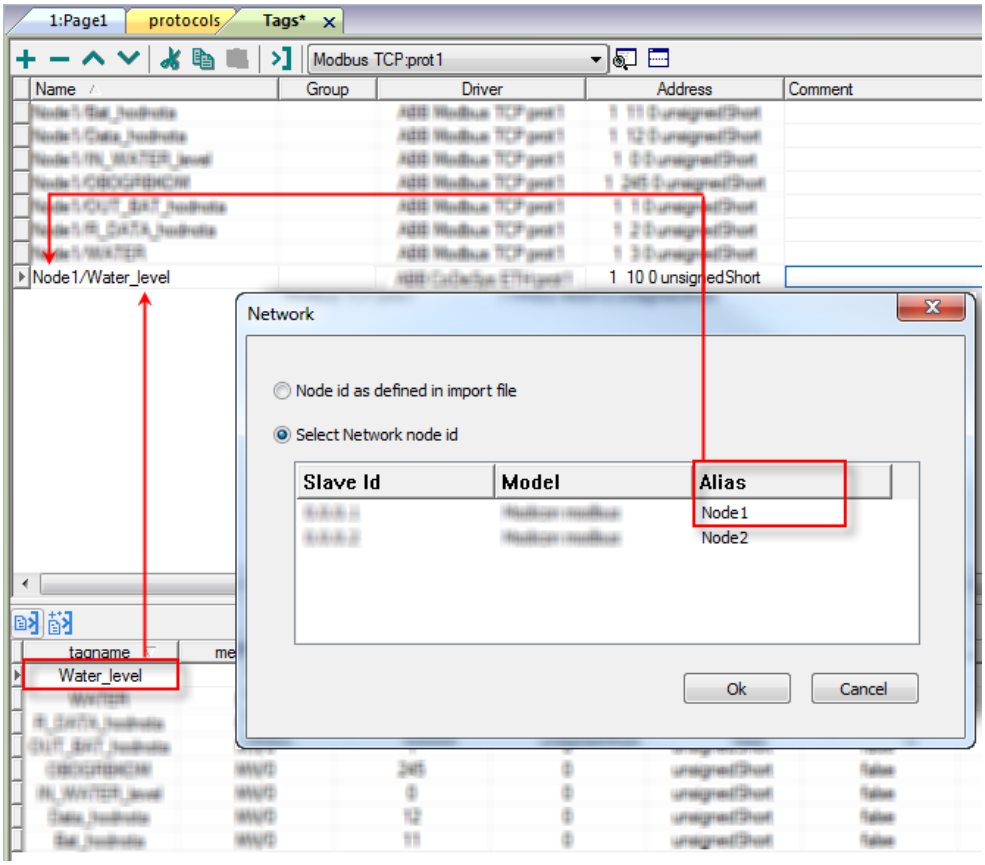
iPLC CODESYS prend en charge la communication avec le pilote CODESYS V2 Ethernet avec support à base de symboles à partir de la version V1.55.

Ajouter un nom d'alias à un protocole

Les noms de variables doivent être uniques au niveau d'un projet. Vous pouvez cependant être amené à utiliser les mêmes noms de variables pour différents nœuds de contrôleur (par exemple lorsque le périphérique IHM est connecté à deux périphériques exécutant la même application).

Lorsque vous générez un protocole, vous pouvez ajouter un nom d'alias qui s'ajoutera aux noms de variables importés pour ce protocole.

Dans l'exemple, la connexion à un certain contrôleur porte le nom **Nœud1**. Lorsque vous importez les variables pour ce nœud, tous les noms de variables porteront le préfixe **Nœud1**, les rendant ainsi uniques au niveau projet/réseau.

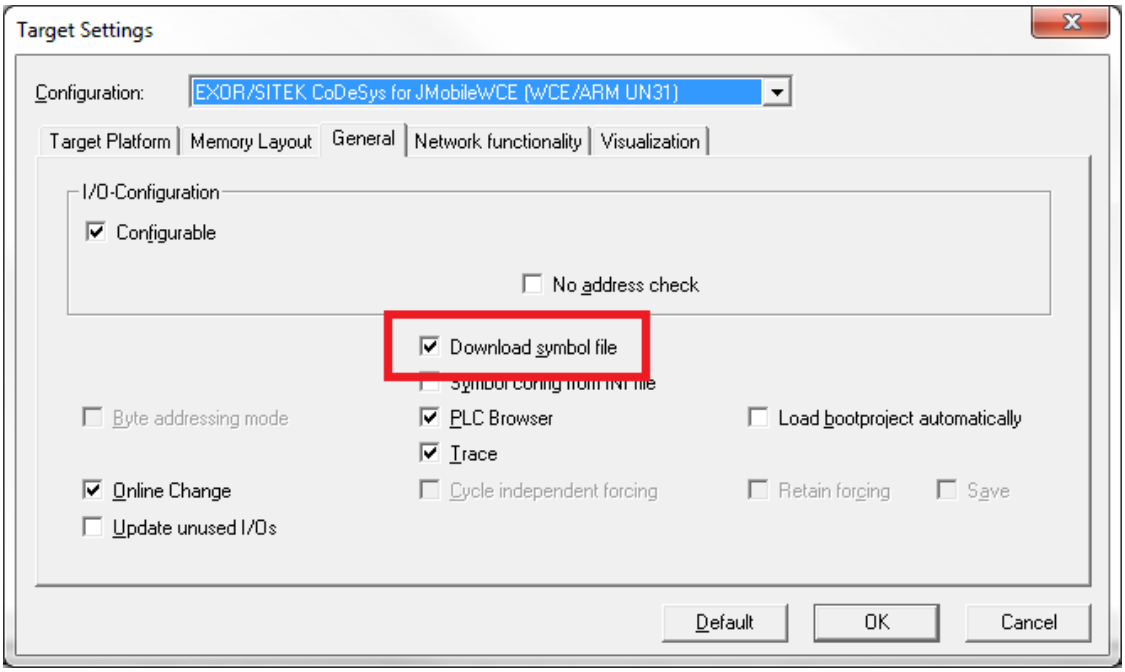


Remarque : vous pouvez donner un nom d'alias uniquement aux variables importées. Il n'est pas possible pour les variables ajoutées manuellement dans l'éditeur de variables de porter un préfixe d'alias dans leur nom.
La chaîne d'alias est associée à l'importation des variables. Une fois l'importation des variables effectuée, une modification de la chaîne d'alias n'aura aucun effet sur les noms figurant déjà dans le dictionnaire. Lorsque vous modifiez la chaîne d'alias et réimportez les variables, toutes les variables réimportées porteront la nouvelle chaîne de préfixe.

Paramètres logiciels CODESYS

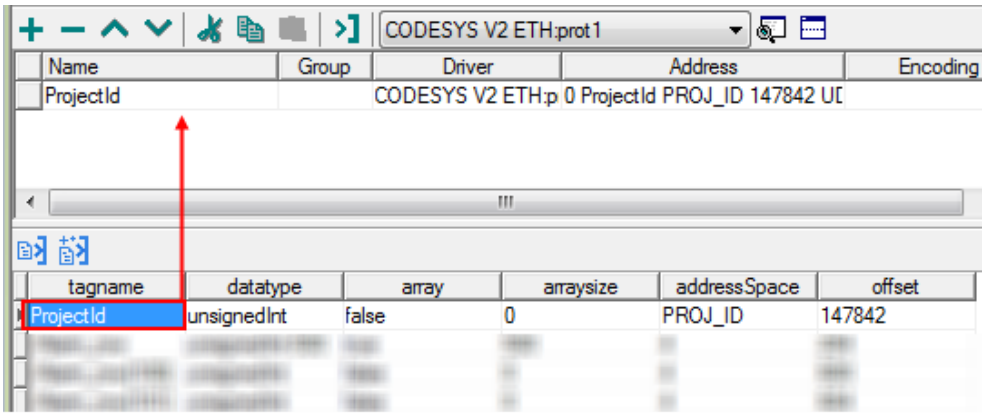
Lors de la création du projet sous CODESYS, sélectionnez **Télécharger le fichier de symboles**.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



Remarque : le périphérique CODESYS V2 Ethernet prend en charge le chargement automatique du fichier de symboles (SDB) depuis l'API. Aucune modification dans le décalage de variable due à une nouvelle compilation du côté du logiciel de l'API ne nécessite la réimportation du fichier de symboles. Le fichier de variables doit être réimporté uniquement si vous renommez ou définissez de nouvelles variables.

Lorsque l'option **Télécharger fichier symbole** n'est pas disponible ou effacée, le protocole ne peut fonctionner que si la variable **ProjectID** est importée. Si le décalage de la variable change en raison d'une nouvelle compilation du programme API, le fichier des symboles doit être réimporté.



Types de données

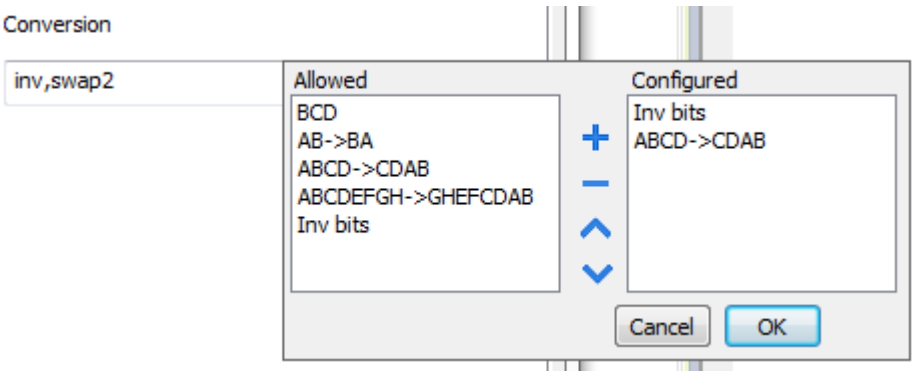
Le module d'importation prend en charge des variables des types de données standard et définies par l'utilisateur.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Types de données pris en charge	• BOOL • WORD • DWORD • INT • UINT • UDINT • DINT • STRING * • REAL • TIME • DATE & TIME et un TABLEAU (ARRAY) unidimensionnel des types ci-dessus. Voir la section « Concepts de programmation » dans le manuel principal. Note * : La longueur de chaîne d'une variable STRING dans l'API doit être de 80 caractères maximum. Déclarez une variable STRING soit avec une taille spécifiée (str : STRING(35) soit la taille par défaut (str : STRING) qui est de 80 caractères.
Types de données non pris en charge	• LWORD • LINT • LREAL

Conversion de variables

Conversion à appliquer à la variable.



En fonction du type de données choisi, la liste **Autorisée** affiche un ou plusieurs types de conversion.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Valeur	Description
Inv bits	**inv** : inverse tous les bits de la variable. *Exemple :* 1001 → 0110 (format binaire) 9 → 6 (format décimal)
Inverser	**neg** : inverse la valeur de la variable. *Exemple :* 25.36 → -25.36
AB -> BA	**swapnibbles** : échange les quartets en un octet. *Exemple :* 15D4 → 514D (format hexadécimal) 5588 → 20813 (format décimal)
ABCD -> CDAB	**swap2** : échange les octets en un mot. *Exemple :* 9ACC → CC9A (format hexadécimal) 39628 → 52378 (format décimal)
ABCDEFGH -> GHEFCDAB	**swap4** : échange les octets en un mot double. *Exemple :* 32FCFF54 → 54FFFC32 (format hexadécimal) 855441236 → 1426062386 (format décimal)
ABC...NOP -> OPM...DAB	**swap8** : échange les octets en un mot long. Exemple : 142,366 → -893553517,588905 (format décimal) 0 10000000110 0001110010111011011001000101101000011100101011000001 → 1 10000011100 1010101000010100010111011011011001011011000010011101 (format binaire)
BCD	**bcd** : sépare l'octet en deux quartets, les lit au format décimal (de 0 à 9) *Exemple :* 23 → 17 (format décimal) 0001 0111 = 23 0001 = 1 (premier quartet) 0111 = 7 (second quartet)

Sélectionnez la conversion et cliquez sur le bouton +. La conversion sélectionnée s'affiche dans la liste **Configurée**.

Si vous configurez d'autres conversions, celles-ci s'affichent dans l'ordre (du haut vers le bas de la liste **Configurée**).

Utilisez les boutons de direction pour changer l'ordre des conversions configurées.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

IP d'écrasement de nœud

Le protocole propose le type de données spécial IP d'écrasement de nœud, qui vous permet de modifier l'adresse IP du contrôleur cible au moment du runtime.

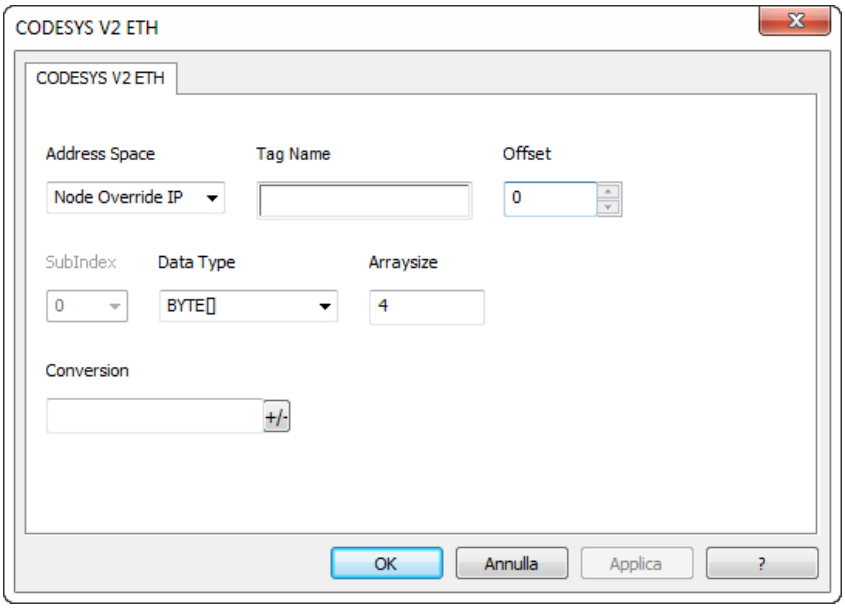
Ce type de mémoire est un tableau de 4 octets non signés, un pour chaque octet de l'adresse IP.

L'IP d'écrasement de nœud est initialisée avec la valeur de l'IP du contrôleur spécifiée dans le projet au moment de la programmation.

IP d'écrasement de nœud	Exécution de l'API
0.0.0.0	La communication avec le contrôleur est coupée, plus aucune trame de requête ne sera générée.
Autre que 0.0.0.0	Interprétée comme étant l'écrasement de l'IP de nœud, et remplace l'adresse IP cible au moment du runtime.

Lorsque le périphérique IHM est connecté à un réseau avec plus d'un nœud de contrôleur, chaque nœud dispose de sa propre variable d'IP d'écrasement de nœud.

Remarque : le maintien des valeurs de l'IP d'écrasement de nœud assignées au runtime est assuré par des cycles d'alimentation.



Importation de variables

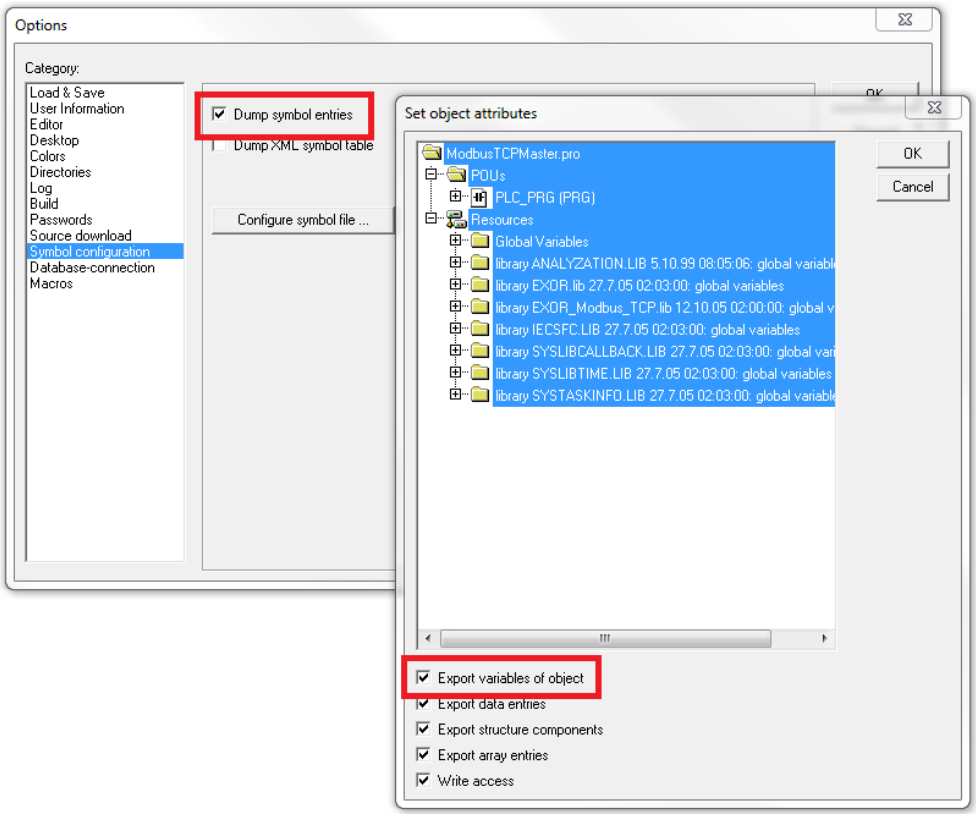
Exporter des variables depuis l'API

Lors de la configuration de l'API grâce au logiciel de configuration du fabricant, autorisez la création d'un Fichier de symboles (extension .sym) sous le logiciel de programmation CODESYS :

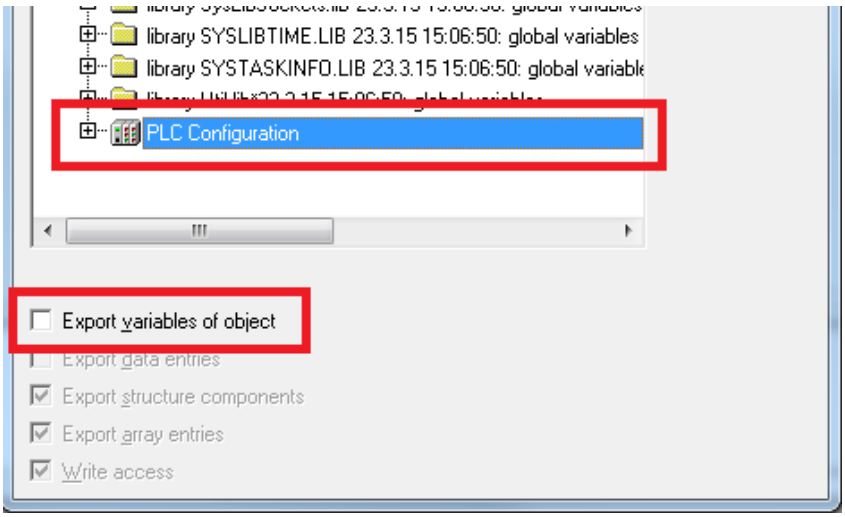
Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

1. Dans le menu **Projet**, cliquez sur **Options**.
2. Cliquez sur **Configuration de symbole**.
3. Sélectionnez **Vider les entrées de symboles**.
4. Cliquez sur **OK**.

Remarque : cliquez ensuite sur **Configurer fichier symboles...** et sélectionnez **Exporter variables de l'objet**. Nous recommandons de décocher la case et de la resélectionner pour s'assurer que les paramètres sont corrects.



Dans certains cas, il est possible que les symboles de variables associées ou intégrées dans les modules I/O du fichier .sym soient dupliqués. Pour supprimer les doublons, sélectionnez l'entrée « Configuration de l'API » dans la liste des objets et décochez l'option « Exporter variables de l'objet ».

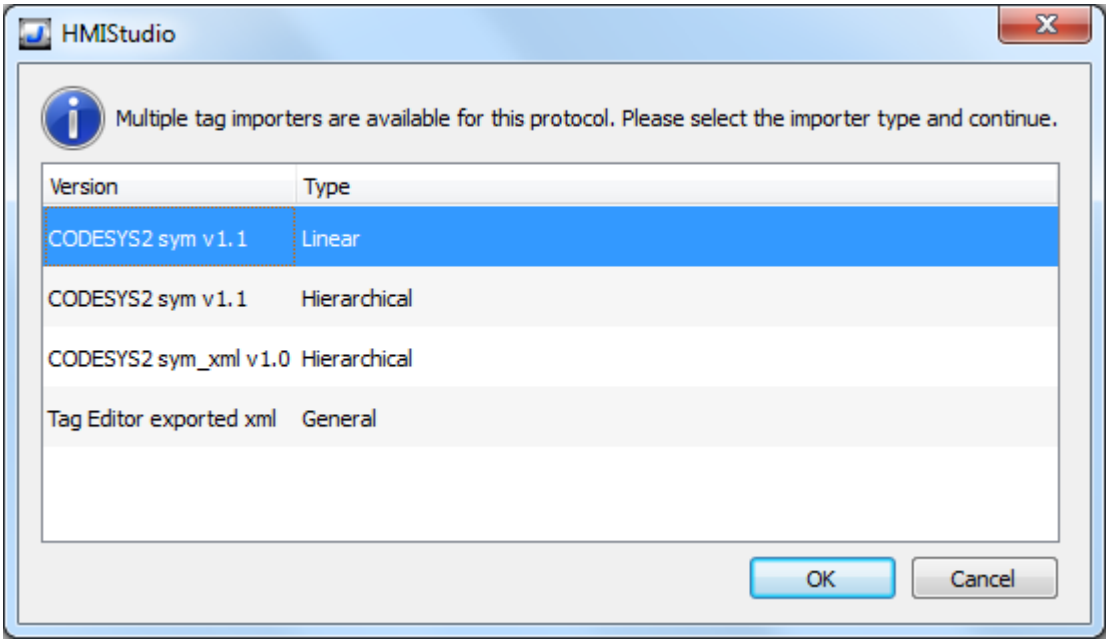


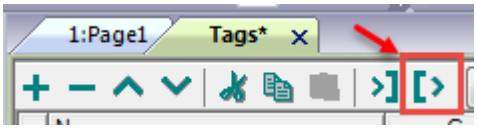
Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Importer des variables dans l'éditeur

Sélectionnez le pilote dans l'éditeur de variables et cliquez sur le bouton **Importer variables** pour lancer le pilote d'importation.

La boîte de dialogue suivante affiche le type d'importation que vous pouvez sélectionner.

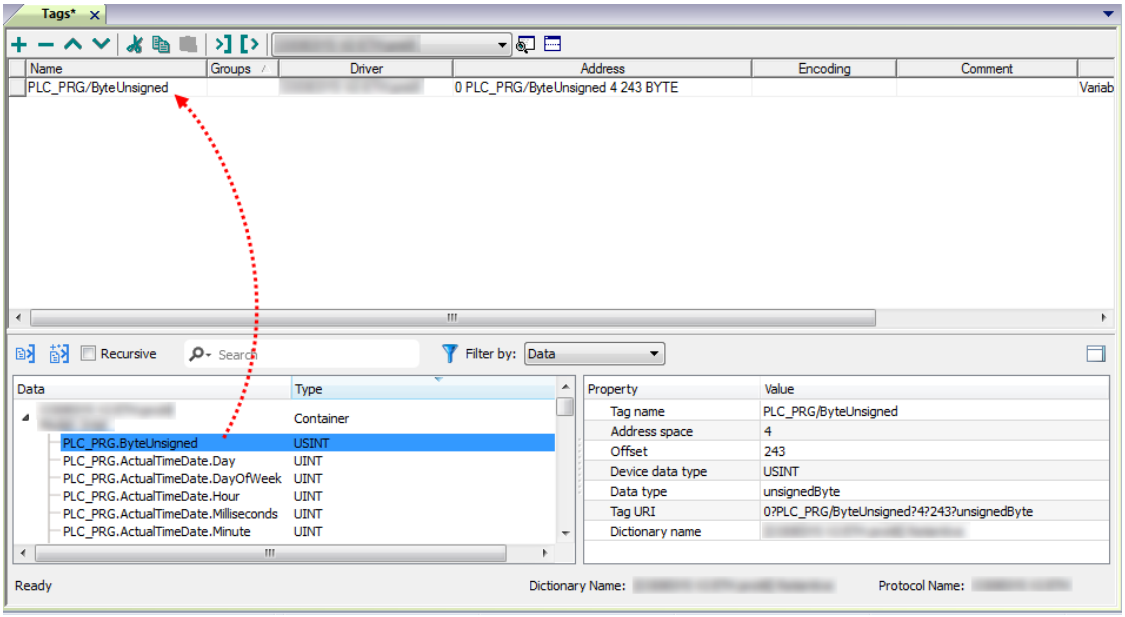


Importateur	Description
CODESYS2 sym v1.1 Linéaire	Nécessite un fichier **.sym**. Toutes les variables s'affichent au même niveau.
CODESYS2 sym v1.1 Hiérarchique	Nécessite un fichier **.sym**. Toutes les variables s'affichent en fonction de la vue hiérarchique CODESYS V2.
CODESYS2 sym_xml v1.0 Hiérarchique	Nécessite un fichier **.sym_xml**. Toutes les variables s'affichent en fonction de la vue hiérarchique CODESYS V2.
Fichier .xml exporté de l'éditeur de variables	Sélectionnez ce pilote d'importation pour lire un fichier XML générique exporté de l'éditeur de variables avec le bouton approprié. 

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

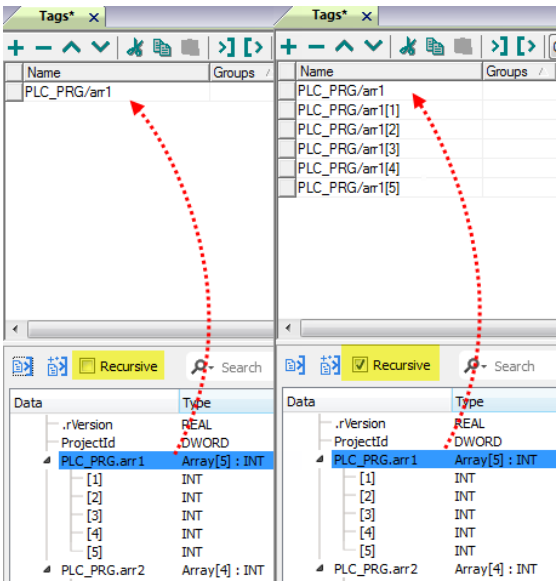
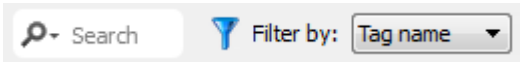
Lorsque vous avez sélectionné le pilote d'importation, localisez le fichier de symboles et cliquez sur **Ouvrir**.

Les variables comprises dans le fichier de symboles sont listées dans le dictionnaire des variables. Le dictionnaire des variables est affiché en bas de l'écran.



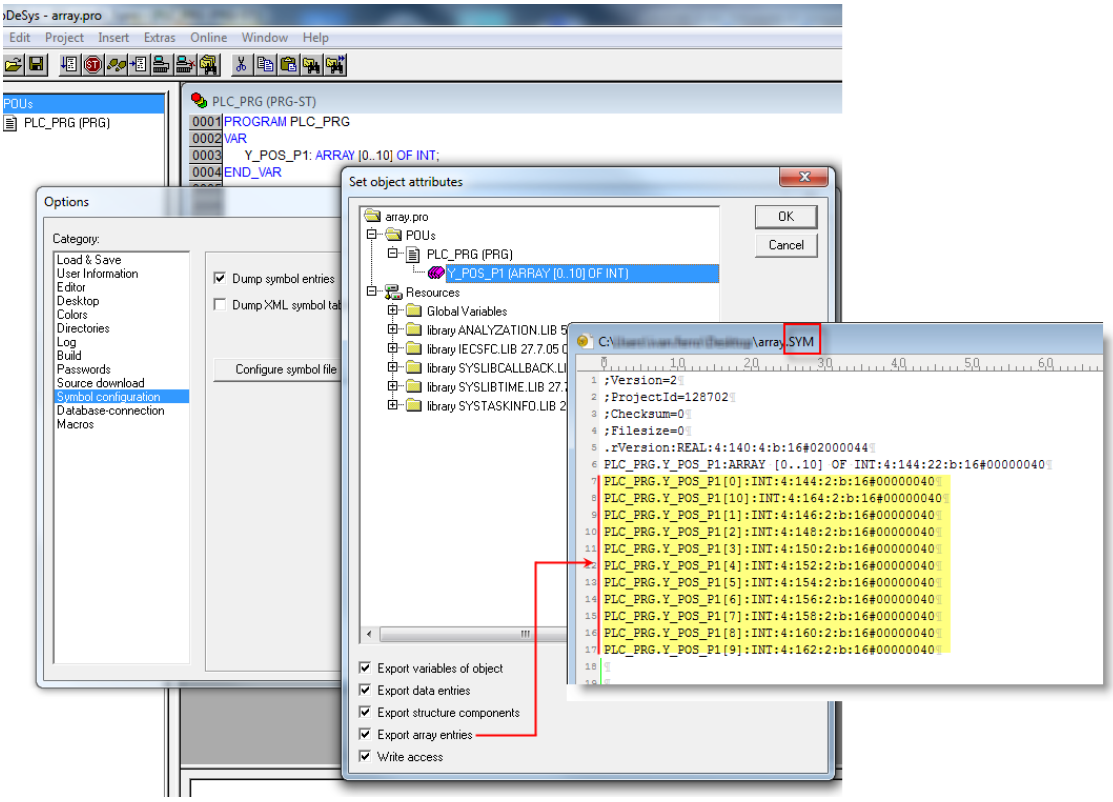
Élément de la barre d'outils	Description
	Importer variable(s). Sélectionnez les variables à importer et cliquez sur cette icône pour ajouter les variables du dictionnaire au projet.
	Mettre à jour variable(s). Cliquez sur cette icône pour mettre à jour les variables du projet grâce à un nouvel import du dictionnaire.
	Cochez cette case pour importer tous les sous-éléments d'une variable. Exemples de résultats avec la case cochée et non cochée :

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Élément de la barre d'outils	Description
	
	Recherche des variables dans le dictionnaire en filtrant en fonction de l'élément sélectionné dans la liste déroulante.

Exportation des tableaux de variables

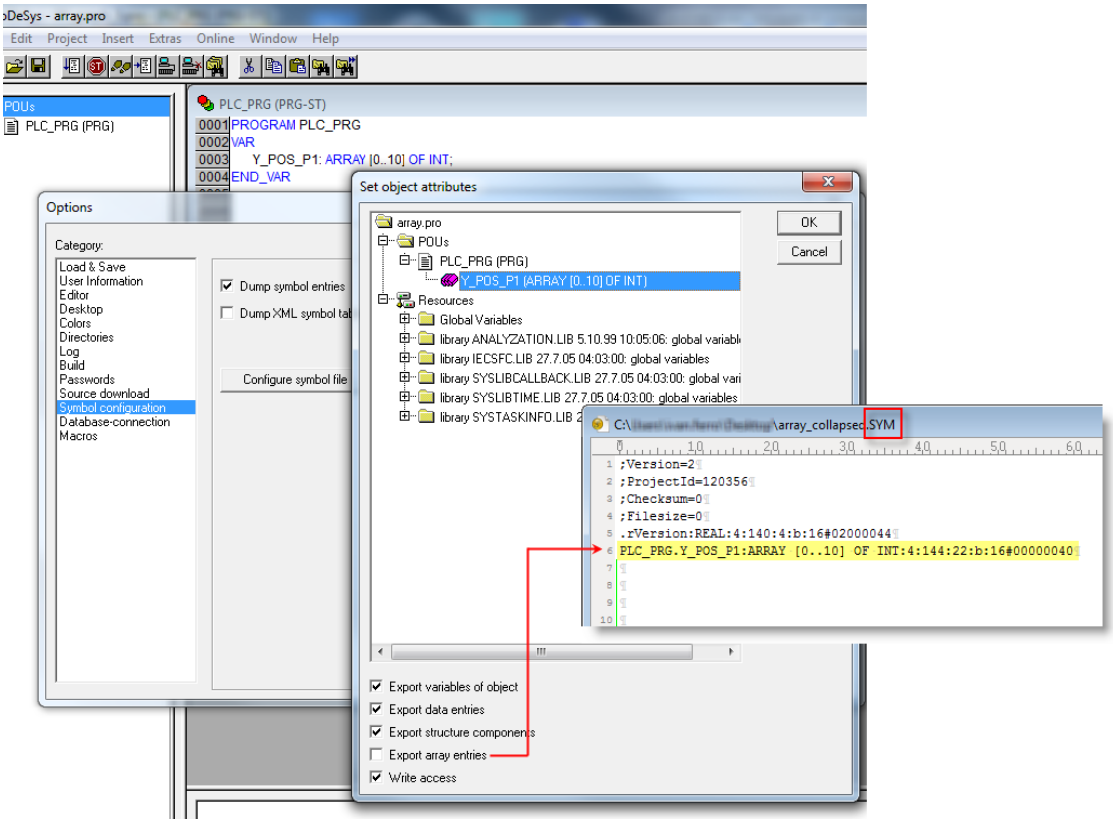
Dans le logiciel CODESYS V2, les tableaux de variables sont divisés en éléments individuels et une variable est créée pour chaque élément. L'exemple suivant présente un tableau avec 10 éléments.



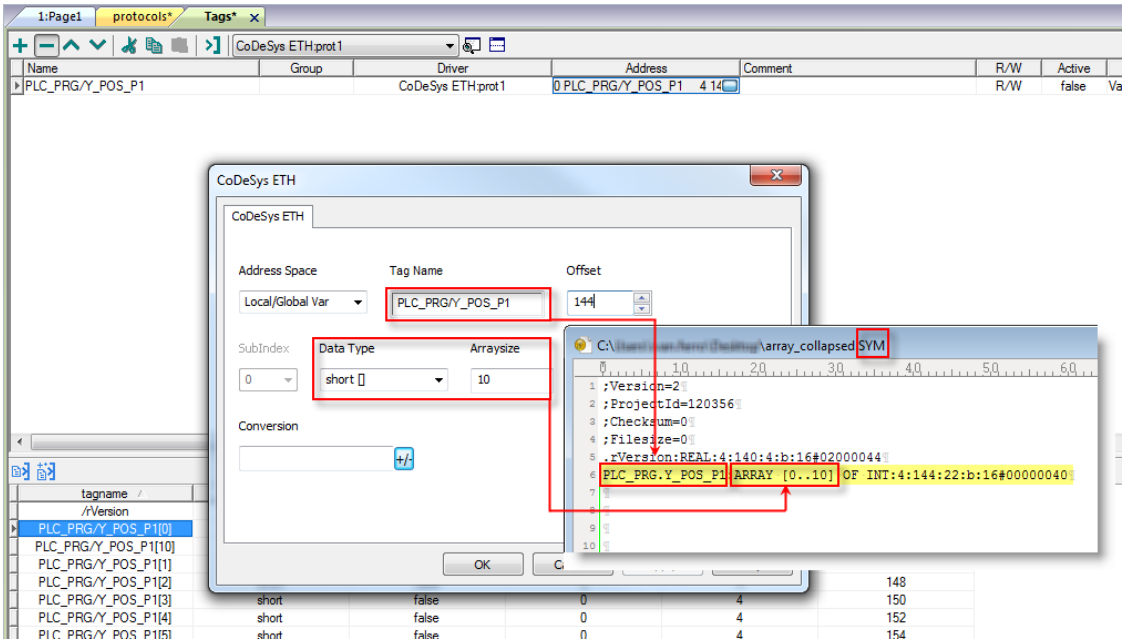
Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Si **Exporter entrées tableau** est sélectionné, une variable sera créée pour chaque élément et exportée vers le fichier .sym. La liste de variables sera importée dans son intégralité et automatiquement dans l'éditeur de variables.

En désactivant les **entrées du tableau d'exportation**, seule une variable pour chaque tableau peut être créée.

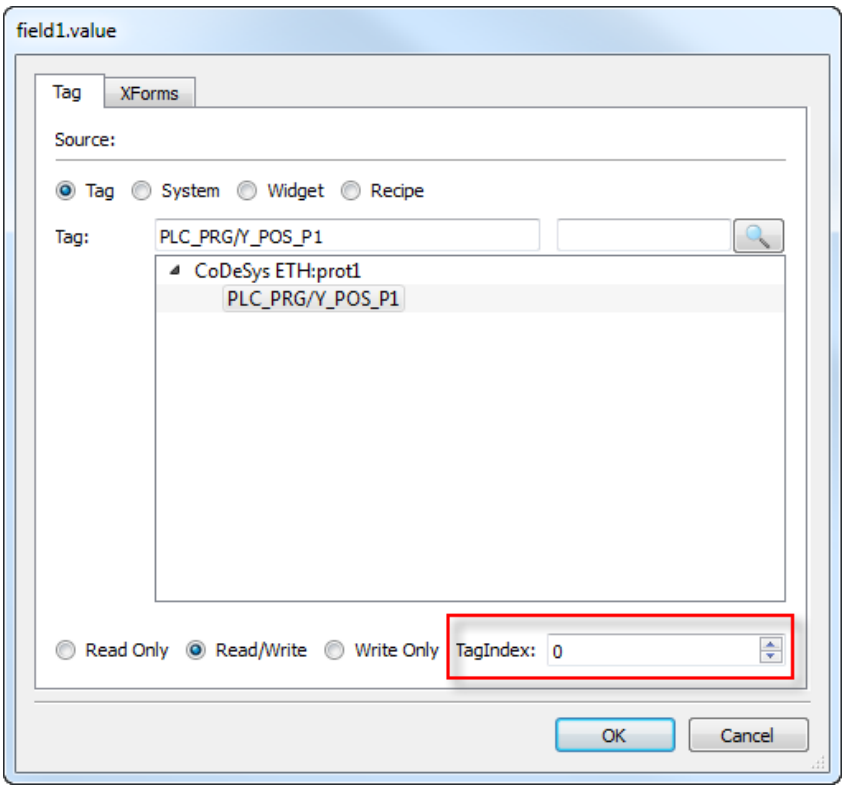


Remarque : lorsque les **Entrées de tableau d'exportation** ont été effacées, une seul variable est créée et exportée dans le fichier .sym. Le tableau n'est alors pas importé automatiquement vers l'éditeur de variables et il est nécessaire de configurer les variables manuellement dans l'éditeur de variables.



Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Tous les éléments de variables peuvent être référencés dans l'éditeur en utilisant **IndexVariables** dans le dialogue **Associer à variable**.



Statut de communication

Vous pouvez afficher l'état de communication actuel grâce aux variables système. Voir la section « Variables système » du manuel principal.

Codes pris en charge par ce pilote de communication :

Erreur	Cause et action
Fichier de symboles absent	Vérifiez Fichier de symboles et téléchargez à nouveau le programme de l'API.
« variable » absente du fichier de symboles	Vérifiez la présence de cette variable dans le projet de l'API.
Délai d'attente dépassé pour Confirmer	Le contrôleur n'a pas envoyé de confirmation.
Délai d'attente dépassé pour la dernière action à Confirmer	Le contrôleur n'a pas envoyé de confirmation pour la dernière action.
Délai d'attente dépassé pour la réception des données	Le contrôleur ne retourne pas de données.
Délai d'attente de connexion dépassé	Le périphérique n'est pas connecté.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

CODESYS V3 ETH

Le pilote de communication CODESYS V3 ETH pour communication de supports via une connexion Ethernet avec les contrôleurs basés sur le logiciel CODESYS V3 PLC par l'entreprise 3S.

Remarque : pour intégrer les évolutions du matériel et des protocoles du contrôleur, les pilotes font l'objet d'une mise à jour continue.

Remarque : Il est possible que le matériel ou les protocoles du contrôleur aient été modifiés depuis la création de cette documentation. Ceci peut interférer avec la fonctionnalité de ce pilote. Par conséquent, testez et vérifiez toujours la fonctionnalité de l'application.

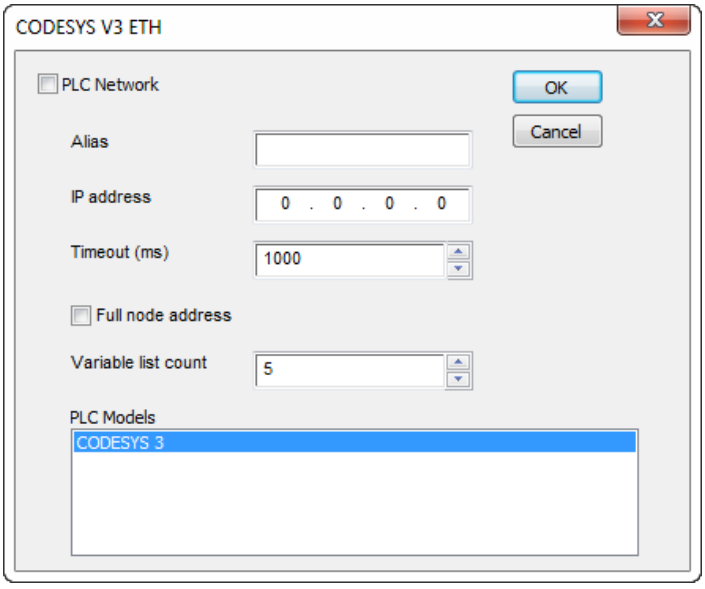
Paramètres de l'éditeur de protocole

Ajouter un protocole

Pour configurer le protocole :

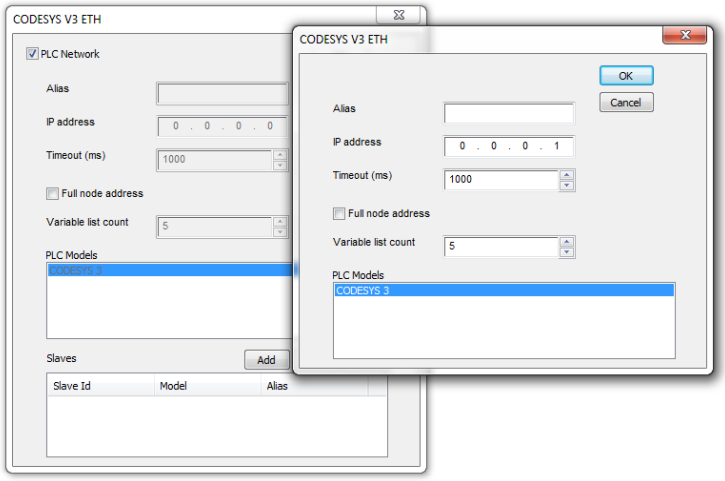
1. Dans le nœud **Config**, double-cliquez sur **Protocoles**.
2. Pour ajouter un pilote, cliquez sur **+** : une nouvelle ligne est ajoutée.
3. Sélectionnez le protocole dans la liste **API**.

La boîte de dialogue de configuration du protocole s'affiche.



Élément	Description
Alias	Nom à utiliser pour l'identification des nœuds dans les configurations réseau. Il est ajouté en préfixe à chaque nom de variable importé pour chaque nœud de réseau.
Adresse IP	Adresse IP Ethernet du contrôleur
Adresse	Dans la mesure où certaines mises en œuvre de CODESYS V3 dans le runtime nécessitent

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Élément	Description
de nœud complète	l'indication des quatre valeurs de l'adresse IP dans les trames de protocoles, ce drapeau force le protocole à créer des adresses IP qui utilisent les quatre champs d'adresses de l'IP.
Décompte liste de variables	La liste de variables représente la meilleure méthode pour augmenter la performance du protocole de communication CODESYS V3 comme elle permet de demander plusieurs éléments de données lors d'une seule session de protocole. Dans la mesure où certaines mises en œuvre de CODESYS V3 dans le runtime ne contiennent qu'un nombre limité de listes de variables à attribuer, ce paramètre vous permet de configurer le nombre maximal de listes de variables que le pilote de communication essayera de créer dans l'API.
Modèle d'API	Format d'octets qui sera utilisé par le pilote de communication en envoyant des trames de communication à l'API.
Délai d'attente	Temps en millisecondes entre les nouvelles tentatives en cas d'échec de communication.
Réseau d'API	Active l'accès aux contrôleurs multiples en réseau. Configurez la bonne option pour chaque contrôleur (esclave). 

Remarque : veuillez consulter la documentation du contrôleur pour vérifier les valeurs requises pour les paramètres **Adresse de nœud complète** ou **Décompte liste de variables**.

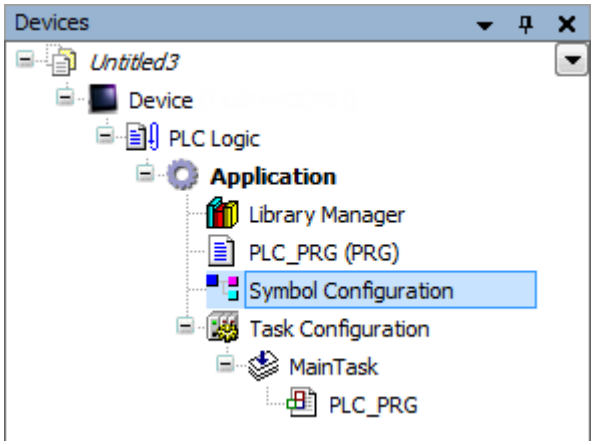
Importation de variables

Exporter des variables depuis l'API

Lors de la création du projet en utilisant CODESYS V3, veillez à configurer correctement le fichier des symboles qui contiendra les variables requises.

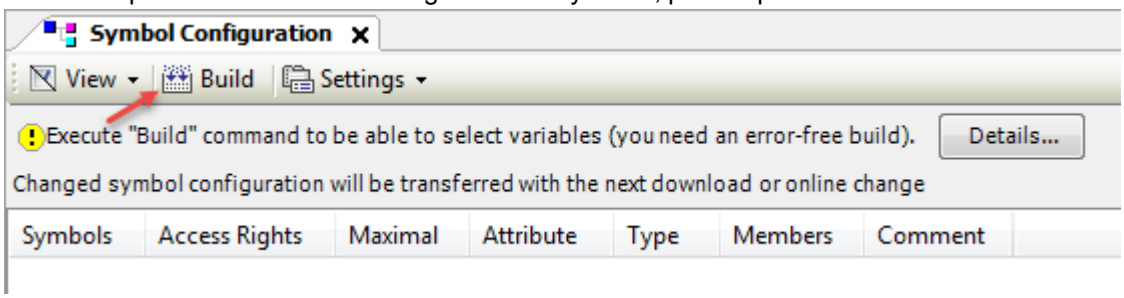
Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

1. Pour ajouter une configuration de Symbole dans un projet CODESYS V3, effectuez un clic droit sur l'élément Application dans l'arborescence du projet, puis sélectionnez Ajouter objet > Configuration de symbole dans le menu contextuel. L'élément de configuration de symbole sera ajouté à l'arborescence du projet.

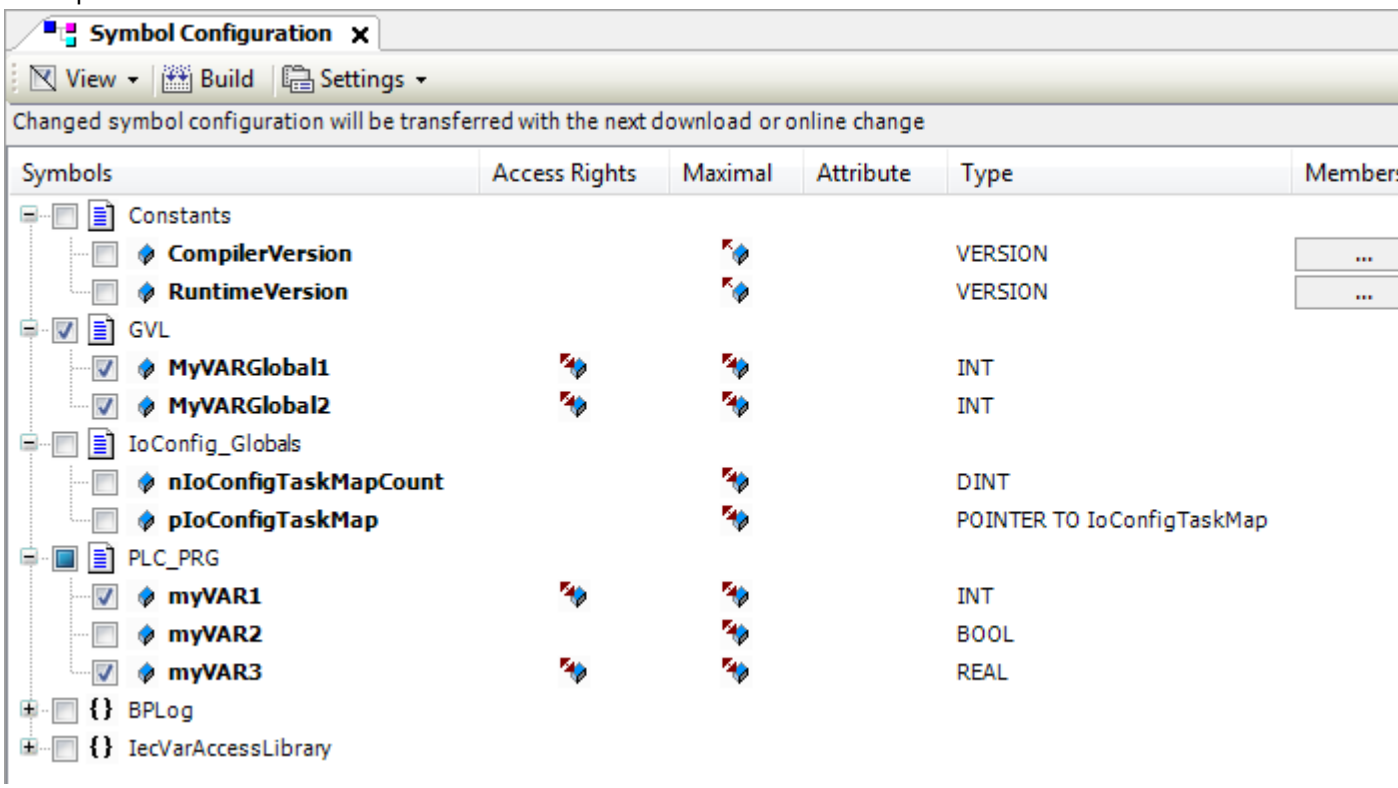


Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

2. Double-cliquez sur l'élément de configuration de symbole, puis cliquez sur le bouton « Construire »

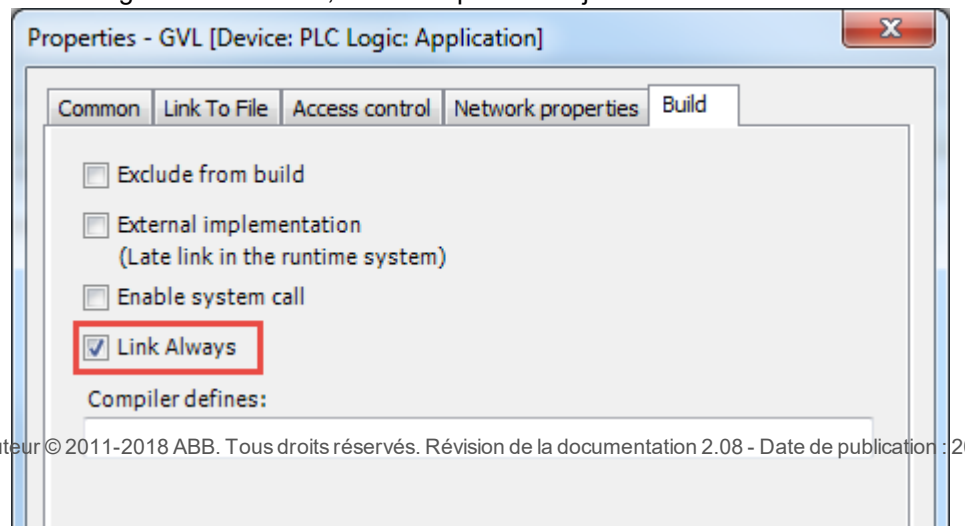


3. L'élément de configuration de symbole contient une liste de toutes les variables disponibles dans le projet CODESYS V3, des variables individuelles ou des groupes de variables peuvent être sélectionnés dans l'élément correspondant de la liste.



4. Une fois la configuration des symboles terminée, téléchargez le projet ou utilisez la fonction **Générer code** (Construire > Générer code) pour créer un fichier .xml qui contient toutes les variables de lecture à importer dans l'éditeur de variables.

Remarque : les variables globales GVL ne sont listées dans la configuration des symboles que si elles sont utilisées dans un programme API.
Pour lister en permanence les variables globales, faites un clic droit sur GVL et sélectionnez « Propriétés ».
Dans l'onglet « Construire », vérifiez l'option « Toujours lier ».

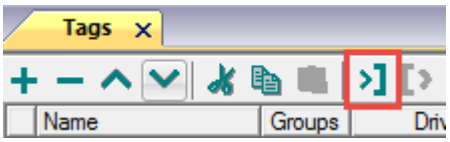


Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

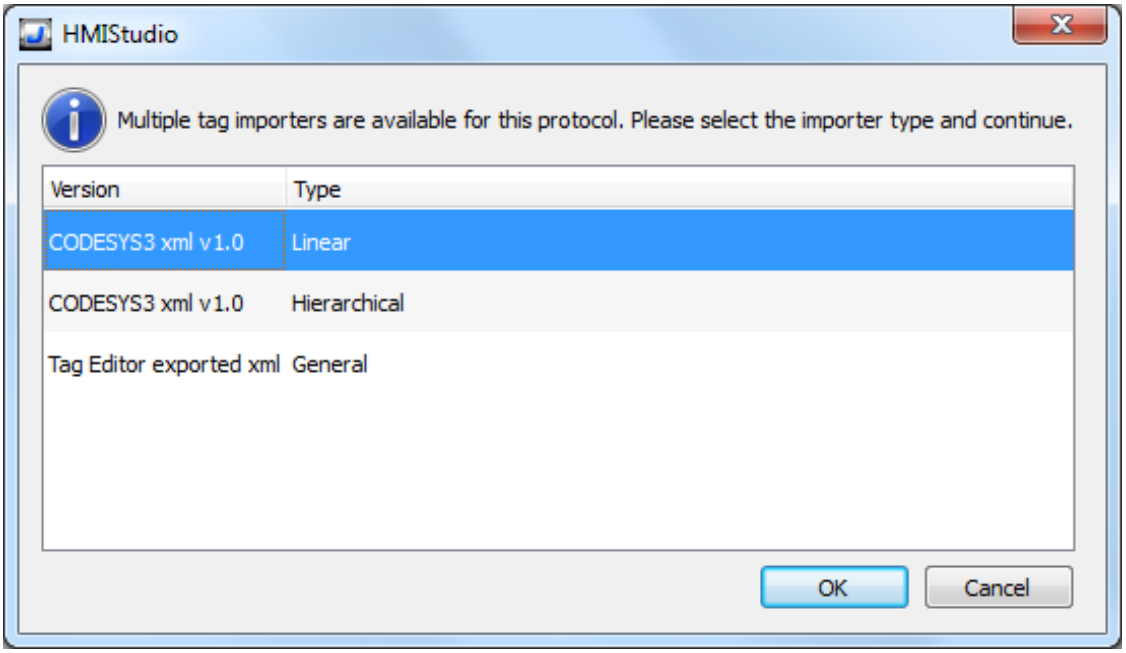
Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

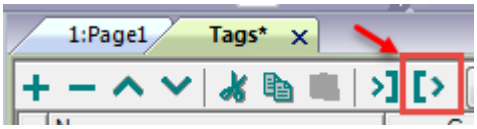
Importer des variables dans l'éditeur

Sélectionnez le pilote dans l'éditeur de variables et cliquez sur le bouton **Importer variables** pour lancer le pilote d'importation.



La boîte de dialogue suivante affiche le type d'importation que vous pouvez sélectionner.

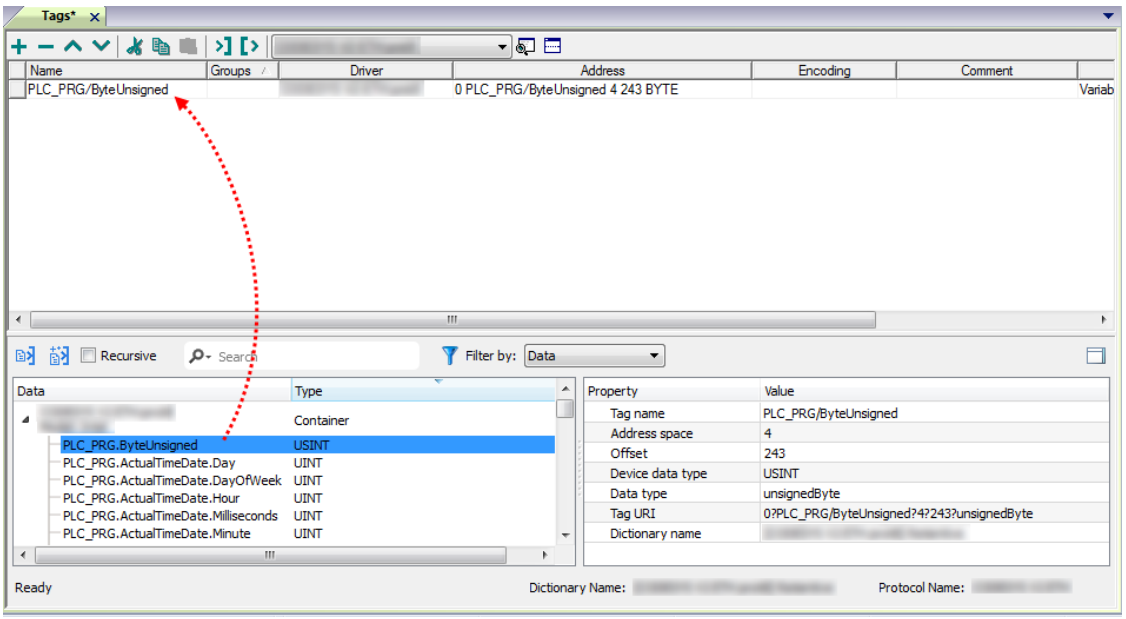


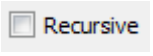
Importateur	Description
CODESYS3 xml v1.0 Linéaire	Nécessite un fichier **.xml**. Toutes les variables s'affichent au même niveau.
CODESYS3 xml v1.0 Hiérarchique	Nécessite un fichier **.xml**. Toutes les variables s'affichent en fonction de la vue hiérarchique CODESYS V3.
Fichier .xml exporté de l'éditeur de variables	Sélectionnez ce pilote d'importation pour lire un fichier XML générique exporté de l'éditeur de variables avec le bouton approprié. 

Lorsque vous avez sélectionné le pilote d'importation, localisez le fichier de symboles et cliquez sur **Ouvrir**.

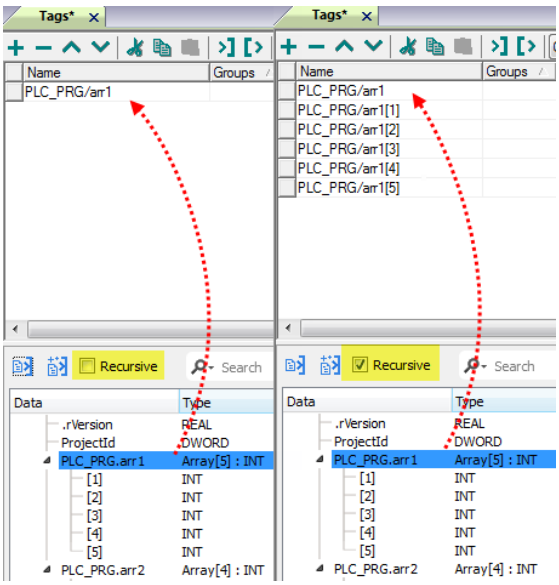
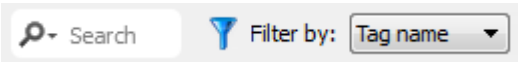
Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Les variables comprises dans le fichier de symboles sont listées dans le dictionnaire des variables. Le dictionnaire des variables est affiché en bas de l'écran.



Élément de la barre d'outils	Description
	Importer variable(s). Sélectionnez les variables à importer et cliquez sur cette icône pour ajouter les variables du dictionnaire au projet.
	Mettre à jour variable(s). Cliquez sur cette icône pour mettre à jour les variables du projet grâce à un nouvel import du dictionnaire.
	Cochez cette case pour importer tous les sous-éléments d'une variable. Exemples de résultats avec la case cochée et non cochée :

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Élément de la barre d'outils	Description
	
	Recherche des variables dans le dictionnaire en filtrant en fonction de l'élément sélectionné dans la liste déroulante.

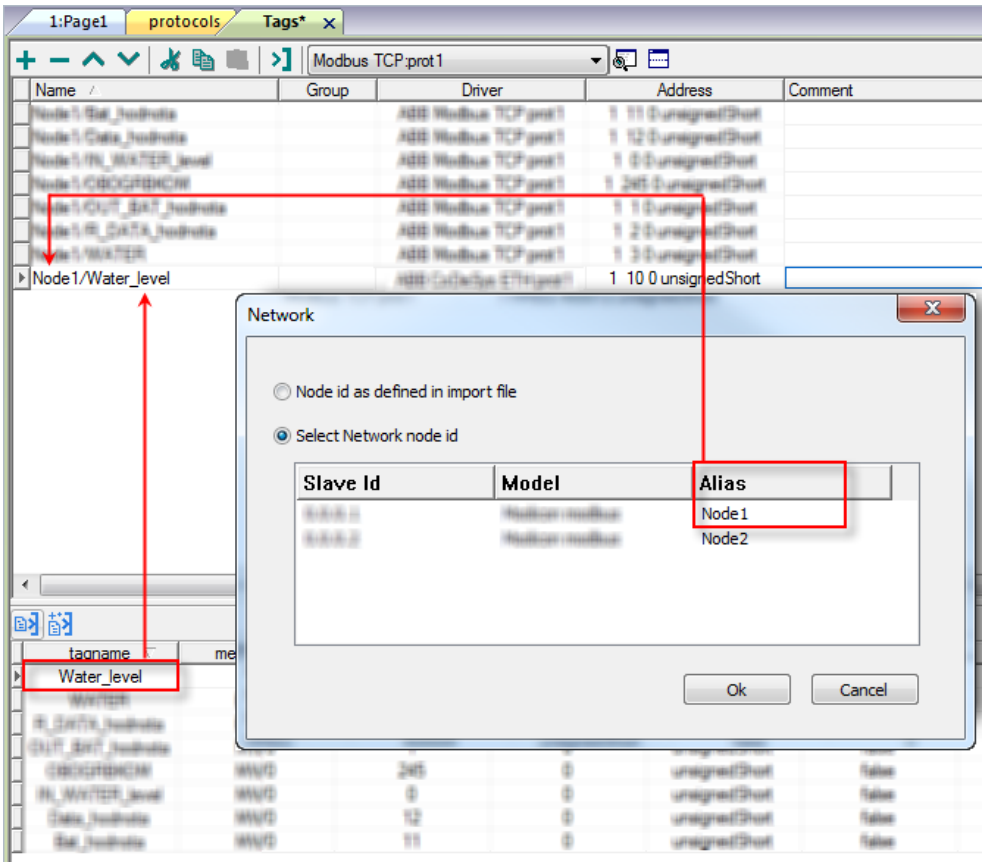
Ajouter un nom d'alias à un protocole

Les noms de variables doivent être uniques au niveau d'un projet. Vous pouvez cependant être amené à utiliser les mêmes noms de variables pour différents nœuds de contrôleur (par exemple lorsque le périphérique IHM est connecté à deux périphériques exécutant la même application).

Lorsque vous générez un protocole, vous pouvez ajouter un nom d'alias qui s'ajoutera aux noms de variables importés pour ce protocole.

Dans l'exemple, la connexion à un certain contrôleur porte le nom **Nœud1**. Lorsque vous importez les variables pour ce nœud, tous les noms de variables porteront le préfixe **Nœud1**, les rendant ainsi uniques au niveau projet/réseau.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



Remarque : vous pouvez donner un nom d'alias uniquement aux variables importées. Il n'est pas possible pour les variables ajoutées manuellement dans l'éditeur de variables de porter un préfixe d'alias dans leur nom.
La chaîne d'alias est associée à l'importation des variables. Une fois l'importation des variables effectuée, une modification de la chaîne d'alias n'aura aucun effet sur les noms figurant déjà dans le dictionnaire. Lorsque vous modifiez la chaîne d'alias et réimportez les variables, toutes les variables réimportées porteront la nouvelle chaîne de préfixe.

Types de données

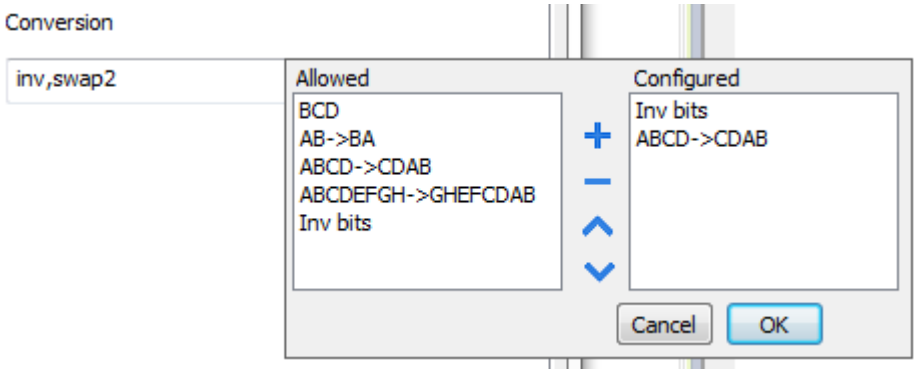
Le module d'importation prend en charge des variables des types de données standard et définies par l'utilisateur.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Types de données pris en charge	• BOOL • INT • SINT • UINT • UDINT • DINT • STRING* • REAL • LREAL • octet • ULINT • LINT et un TABLEAU (ARRAY) unidimensionnel des types ci-dessus. Voir la section « Concepts de programmation » dans le manuel principal. Note * : La longueur de chaîne d'une variable STRING dans l'API doit être de 80 caractères maximum. Déclarez une variable STRING soit avec une taille spécifiée (str : STRING(35) soit la taille par défaut (str : STRING) qui est de 80 caractères.
Types de données non pris en charge	• LWORD • LINT

Conversion de variables

Conversion à appliquer à la variable.



En fonction du type de données choisi, la liste **Autorisée** affiche un ou plusieurs types de conversion.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Valeur	Description
Inv bits	**inv** : inverse tous les bits de la variable. *Exemple :* 1001 → 0110 (format binaire) 9 → 6 (format décimal)
Inverser	**neg** : inverse la valeur de la variable. *Exemple :* 25.36 → -25.36
AB -> BA	**swapnibbles** : échange les quartets en un octet. *Exemple :* 15D4 → 514D (format hexadécimal) 5588 → 20813 (format décimal)
ABCD -> CDAB	**swap2** : échange les octets en un mot. *Exemple :* 9ACC → CC9A (format hexadécimal) 39628 → 52378 (format décimal)
ABCDEFGH -> GHEFCDAB	**swap4** : échange les octets en un mot double. *Exemple :* 32FCFF54 → 54FFFC32 (format hexadécimal) 855441236 → 1426062386 (format décimal)
ABC...NOP -> OPM...DAB	**swap8** : échange les octets en un mot long. Exemple : 142,366 → -893553517,588905 (format décimal) 0 10000000110 0001110010111011011001000101101000011100101011000001 → 1 10000011100 1010101000010100010111011011011001011011000010011101 (format binaire)
BCD	**bcd** : sépare l'octet en deux quartets, les lit au format décimal (de 0 à 9) *Exemple :* 23 → 17 (format décimal) 0001 0111 = 23 0001 = 1 (premier quartet) 0111 = 7 (second quartet)

Sélectionnez la conversion et cliquez sur le bouton +. La conversion sélectionnée s'affiche dans la liste **Configurée**.

Si vous configurez d'autres conversions, celles-ci s'affichent dans l'ordre (du haut vers le bas de la liste **Configurée**).

Utilisez les boutons de direction pour changer l'ordre des conversions configurées.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

IP d'écrasement de nœud

Le protocole propose le type de données spécial IP d'écrasement de nœud, qui vous permet de modifier l'adresse IP du contrôleur cible au moment du runtime.

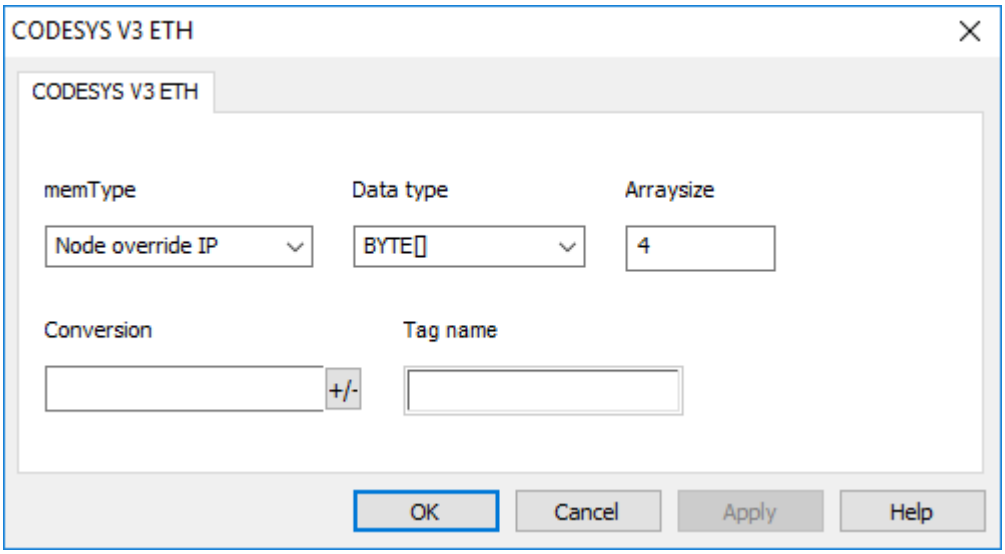
Ce type de mémoire est un tableau de 4 octets non signés, un pour chaque octet de l'adresse IP.

L'IP d'écrasement de nœud est initialisée avec la valeur de l'IP du contrôleur spécifiée dans le projet au moment de la programmation.

IP d'écrasement de nœud	Exécution de l'API
0.0.0.0	La communication avec le contrôleur est coupée, plus aucune trame de requête ne sera générée.
Autre que 0.0.0.0	Interprétée comme étant l'écrasement de l'IP de nœud, et remplace l'adresse IP cible au moment du runtime.

Lorsque le périphérique IHM est connecté à un réseau avec plus d'un nœud de contrôleur, chaque nœud dispose de sa propre variable d'IP d'écrasement de nœud.

Remarque : le maintien des valeurs de l'IP d'écrasement de nœud assignées au runtime est assuré par des cycles d'alimentation.

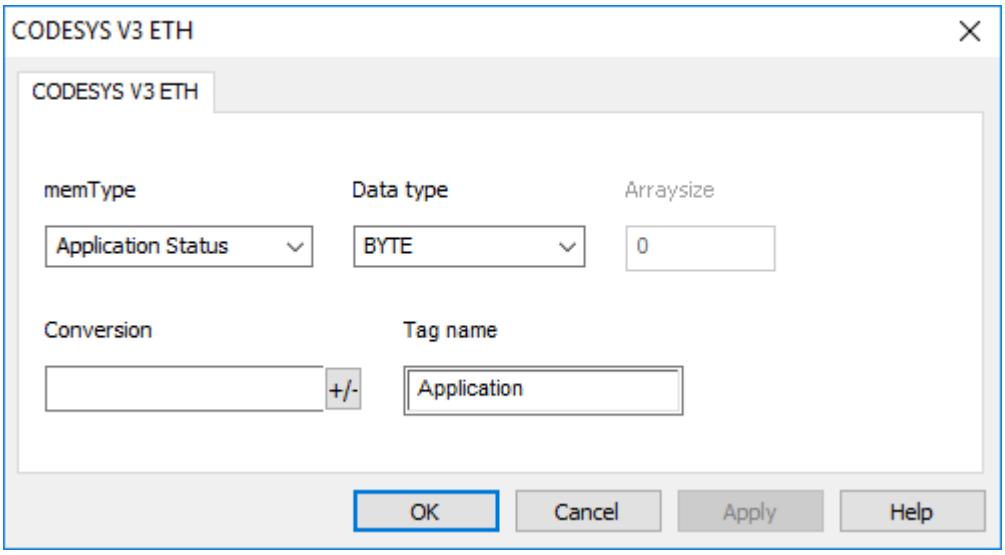


Statut d'application

Le protocole fournit le Statut de l'application type de données qui vous permet de vérifier ou de modifier l'état des applications.

Les variables pointant sur le statut de l'application doivent contenir dans le champ « Nom de variable » le nom de l'application API (le nom par défaut est souvent « Application »)

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



Lorsque le périphérique IHM est connecté à un réseau avec plus d'un nœud de contrôleur, chaque nœud dispose de sa propre variable de statut d'application.

Statut d'application	Description
0	FONCTIONNEMENT EN COURS
1	À L'ARRÊT
2	ARRÊTÉ à un point d'interruption

Remarque : Il n'est pas possible d'écrire 2 comme nouveau statut

Statut de communication

Vous pouvez afficher le statut de communication actuel grâce aux variables système. Voir la section « Variables système » du manuel principal.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Ethernet/IP CIP

Ce protocole a été mis en œuvre selon les spécifications Ethernet/IP publiées (disponibles sur www.odva.org).

Le pilote Ethernet/IP CIP a été conçu pour fournir la meilleure performance tout en ralentissant le moins possible la performance globale du système. Malgré le fait que le pilote Ethernet/IP CIP est rapide, nous recommandons d'utiliser des noms de variables brefs. Les variables sont lues et écrites par le périphérique en indiquant leur nom symbolique dans les requêtes de communication. Pour cette raison, un nom de variable plus long signifie un plus grand volume pour la requête.

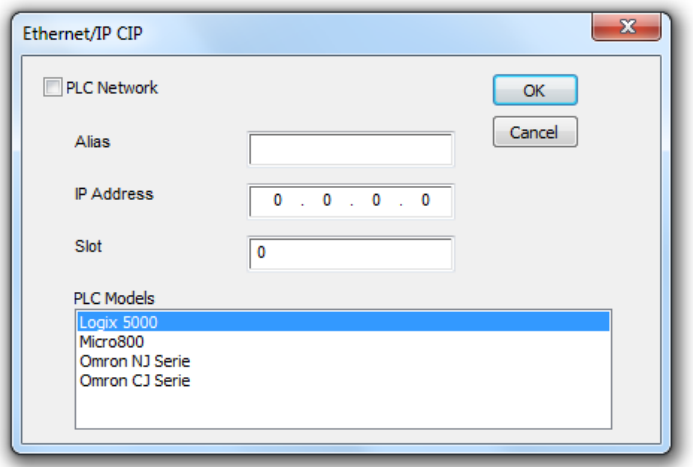
Paramètres de l'éditeur de protocole

Ajouter un protocole

Pour configurer le protocole :

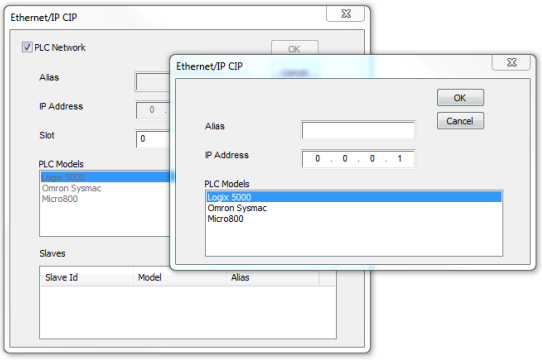
1. Dans le nœud **Config**, double-cliquez sur **Protocoles**.
2. Pour ajouter un pilote, cliquez sur **+** : une nouvelle ligne est ajoutée.
3. Sélectionnez le protocole dans la liste **API**.

La boîte de dialogue de configuration du protocole s'affiche.



Champ	Description
Alias	Nom d'identification des nœuds dans les configurations réseau. Il est ajouté en préfixe à chaque nom de variable importé pour chaque nœud de réseau.
IP Address	Adresse IP Ethernet du contrôleur.
Fente	Numéro de fente CPU pour modèles Logix 5000 (typiquement 0). Pour plus de détails, veuillez consulter la documentation du contrôleur.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Champ	Description
Modèles d'API	Modèle d'API utilisé pour importer le fichier de variables.
Réseau d'API	Active l'accès aux contrôleurs multiples en réseau. Configurez la bonne option pour chaque contrôleur (esclave). 

Modèle contrôleur Logix 5000

Le pilote Ethernet/IP CIP permet de connecter des contrôleurs Allen-Bradley ControlLogix et CompactLogix Ethernet.

Pour assurer la communication avec les contrôleurs ControlLogix® 5500, veuillez utiliser un module de communication Ethernet/IP pour Ethernet tel que les modèles 1756-EN2T ou 1756-ENET.

La communication Ethernet avec les contrôleurs CompactLogix™ 5300 requiert un processeur à port Ethernet/IP intégré tel que le modèle 1769-L32E.

Toutes les marques déposées sont de la propriété de leurs propriétaires respectifs

Les CPU Logix ne sont pas dotés d'une organisation de mémoire interne fixe mais la mémoire est configurée au moment du développement par l'utilisateur. Le logiciel RSLogix 5000 peut alors exporter sur l'application de la liste des variables créés pour chaque contrôleur.

Le projet chargé sur le périphérique IHM doit se référer aux noms de variables attribués dans le logiciel RSLogix 5000 au moment du développement. L'éditeur de variable supporte l'importation directe du fichier de variable généré par le logiciel RSLogix 5000 au format .csv.

La mise en œuvre du pilote Ethernet/IP prend en charge également les types de données structurées qui peuvent être importés à partir de fichiers .L5X.

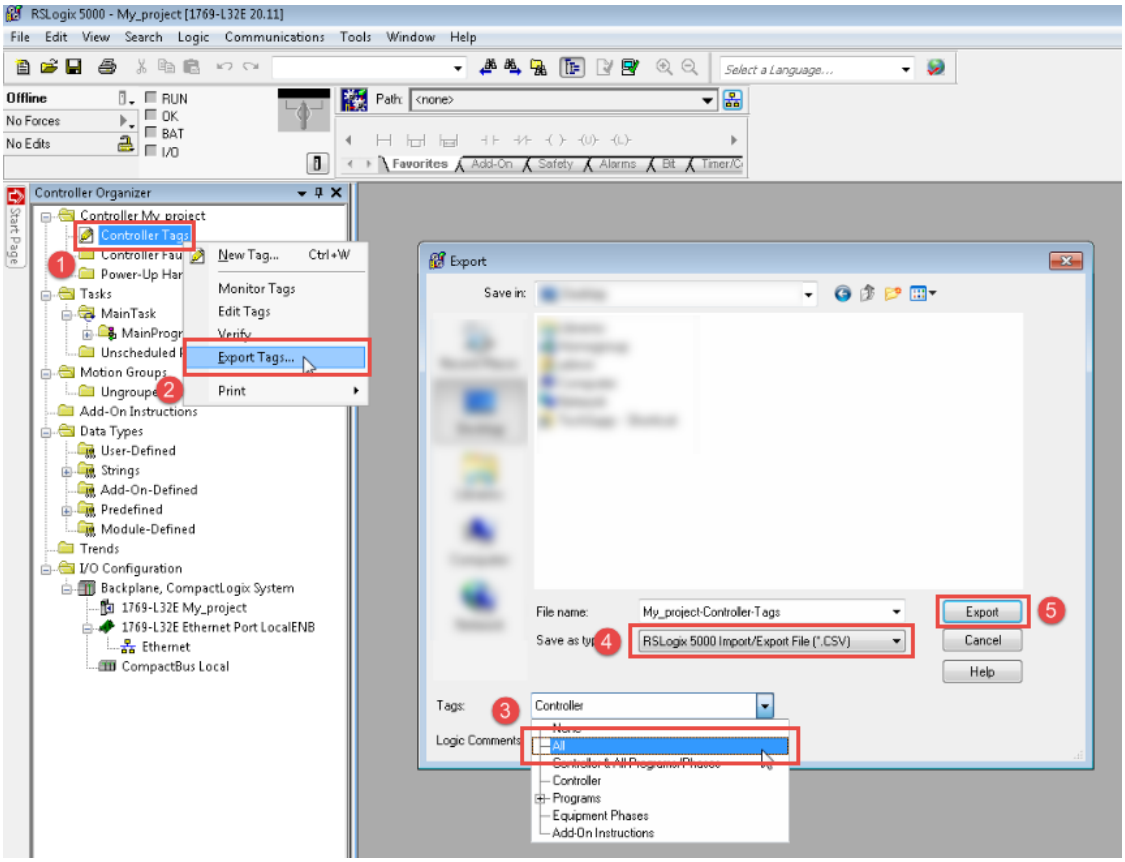
Le pilote prend en charge l'accès à la fois aux variables du contrôleur et de programme.

Exportation CSV et fichiers L5X en utilisant RSLogix5000

Pour exporter le fichier de variable **.CSV** :

1. Dans le volet **Organisateur de contrôleur**, effectuez un clic droit sur **variables du contrôleur**.
2. Sélectionnez **Exporter variables** : le dialogue **Exportation** s'affiche.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

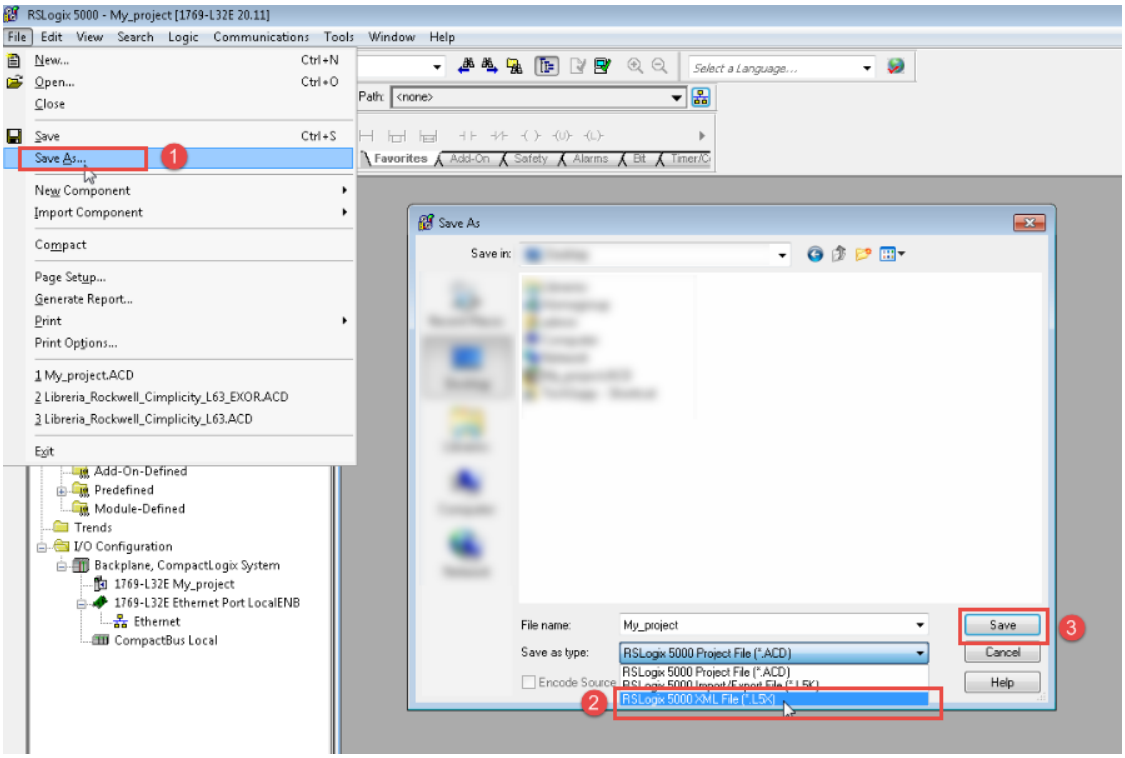


3. Choisir **Tous** depuis la liste des **variables** pour exporter tous les variables.
4. Sélectionner l'option **Enregistrer en tant que type** en **.CSV**.
5. Cliquer sur **Export** : toutes les variables sont exportées vers un fichier **.CSV**.

Pour exporter le fichier type de données **.L5X** :

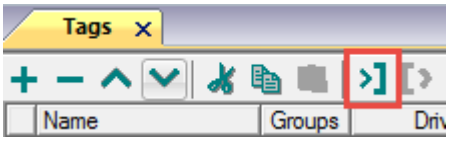
1. Sélectionnez **Fichier > Enregistrer sous**.
2. Sélectionner l'option **Enregistrer en tant que type** en **.L5X**.
3. Cliquer sur **Enregistrer** : toutes les variables sont exportées vers un fichier **.L5X**.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



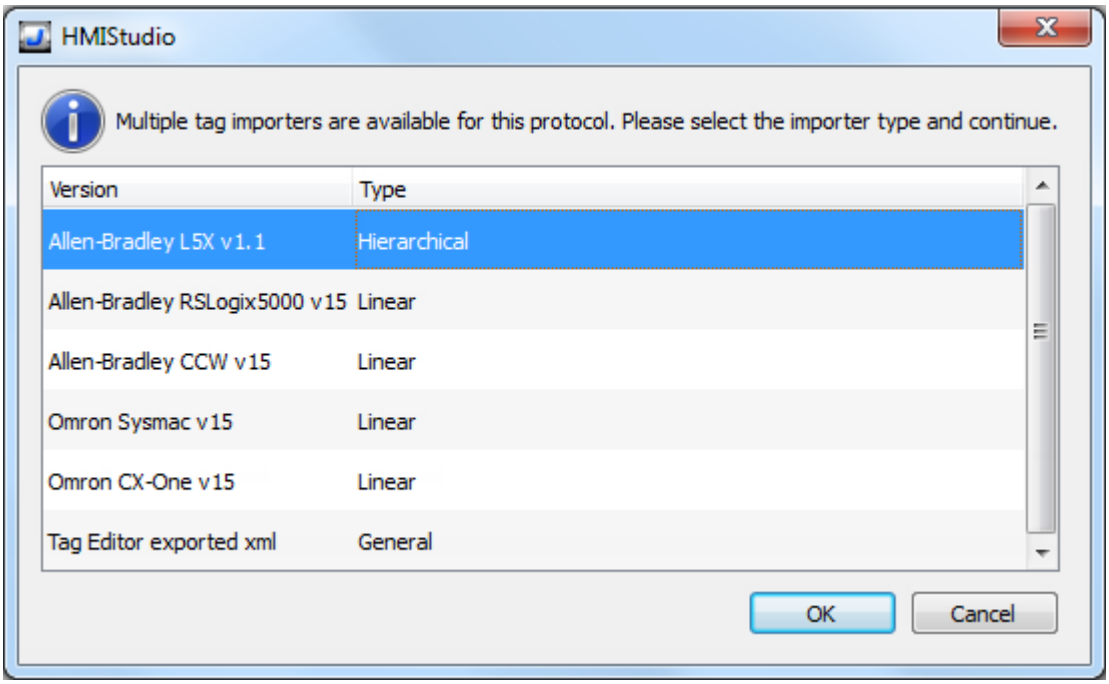
Importer des fichiers dans l'éditeur

Sélectionnez le pilote dans l'éditeur de variables et cliquez sur le bouton **Importer variables** pour lancer le pilote d'importation.



La boîte de dialogue suivante affiche le type d'importation que vous pouvez sélectionner.

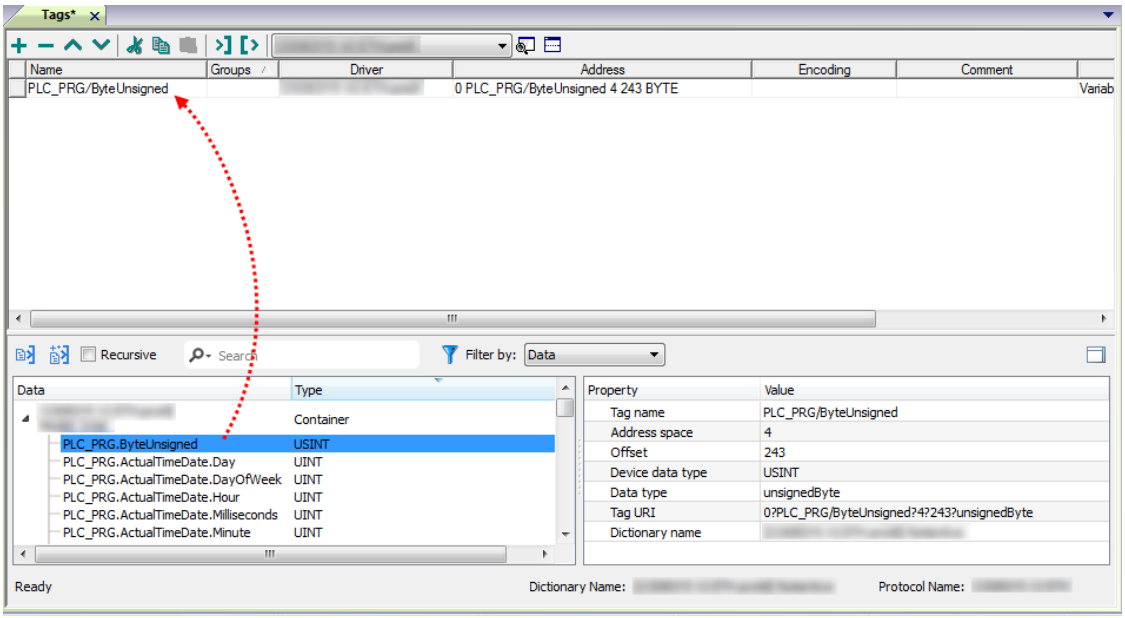
Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



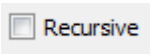
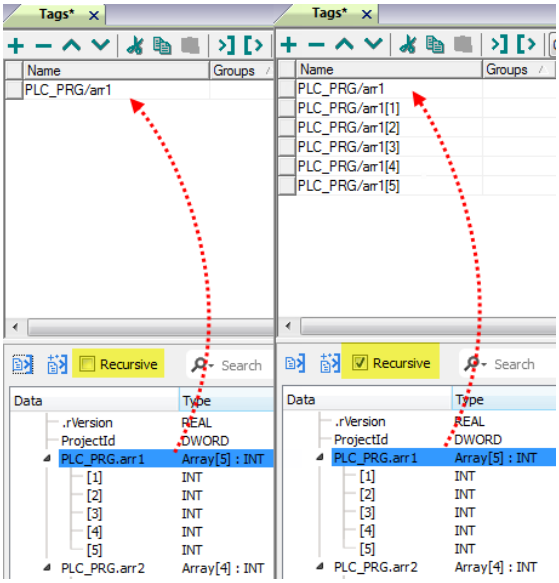
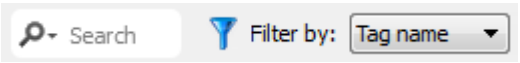
Sélectionner l'option **Allen-Bradley RSLogix5000 v15**.

Lorsque vous avez sélectionné le pilote d'importation, localisez le fichier de symboles et cliquez sur **Ouvrir**.

Les variables comprises dans le fichier de symboles sont listées dans le dictionnaire des variables. Le dictionnaire des variables est affiché en bas de l'écran.



Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Élément de la barre d'outils	Description
	Importer variable(s). Sélectionnez les variables à importer et cliquez sur cette icône pour ajouter les variables du dictionnaire au projet.
	Mettre à jour variable(s). Cliquez sur cette icône pour mettre à jour les variables du projet grâce à un nouvel import du dictionnaire.
	Cochez cette case pour importer tous les sous-éléments d'une variable. Exemples de résultats avec la case cochée et non cochée : 
	Recherche des variables dans le dictionnaire en filtrant en fonction de l'élément sélectionné dans la liste déroulante.

Remarque : lors de l'importation des types de données de tableau, l'importateur les étend en créant des variables individuelles par chaque élément du tableau ; ceci est valable pour tous les types de données, sauf pour les tableaux booléen. Dans ce cas, ils sont importés en tant que « booléen-32 » et l'élément de tableau unique peuvent être traité en utilisant le paramètre « Tag Index » depuis la boîte de dialogue « Joindre à ... ».

Types de données définies par module et définies par l'utilisateur

RSLogix 5000 vous permet de définir des variables avec plusieurs types de données.

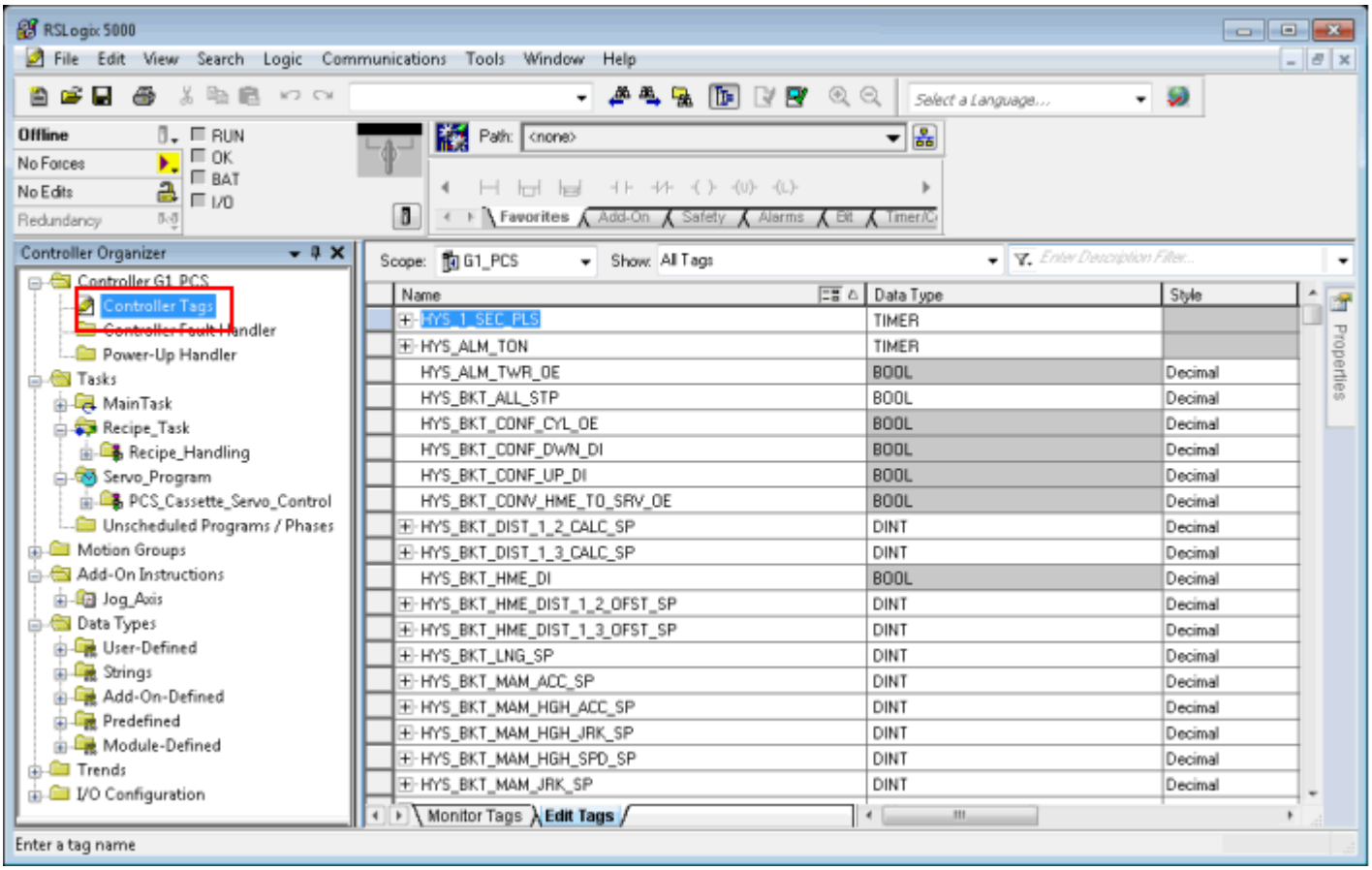
Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Groupe de types de données	Description
Prédéfinies	Types de données standard tels que BOOL, DINT, SINT, INT et types de données moins courants tels que PID, COUNTER, TIMER.
Définies par module	Type de données associées à des modules I/O optionnelles généralement référencés par des alias.
Définies par utilisateur	Type de données personnalisé défini par l'utilisateur

Pour importer un type de données Prédéfini (sauf types de données standard qui sont toujours importés) et Défini par le module, vous devez éditer le fichier ETIPSpecialDataTypes.xml à l'emplacement *langues\partagées\studio\importationvariables* ou (en fonction de la version du logiciel) *studio\importationvariables*.

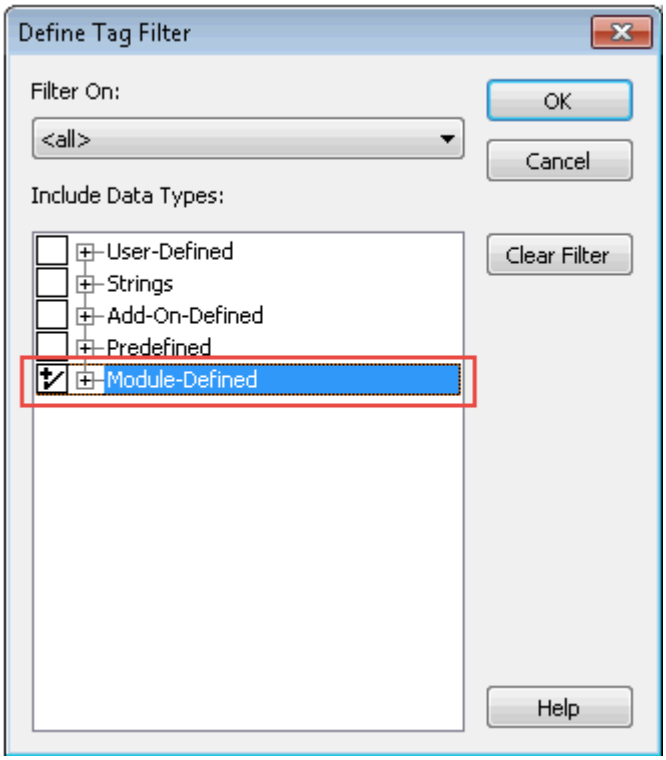
Dans le logiciel RSLogix5000 :

1. Dans le volet **Organisateur de contrôleur**, sélectionnez **variables du contrôleur**.

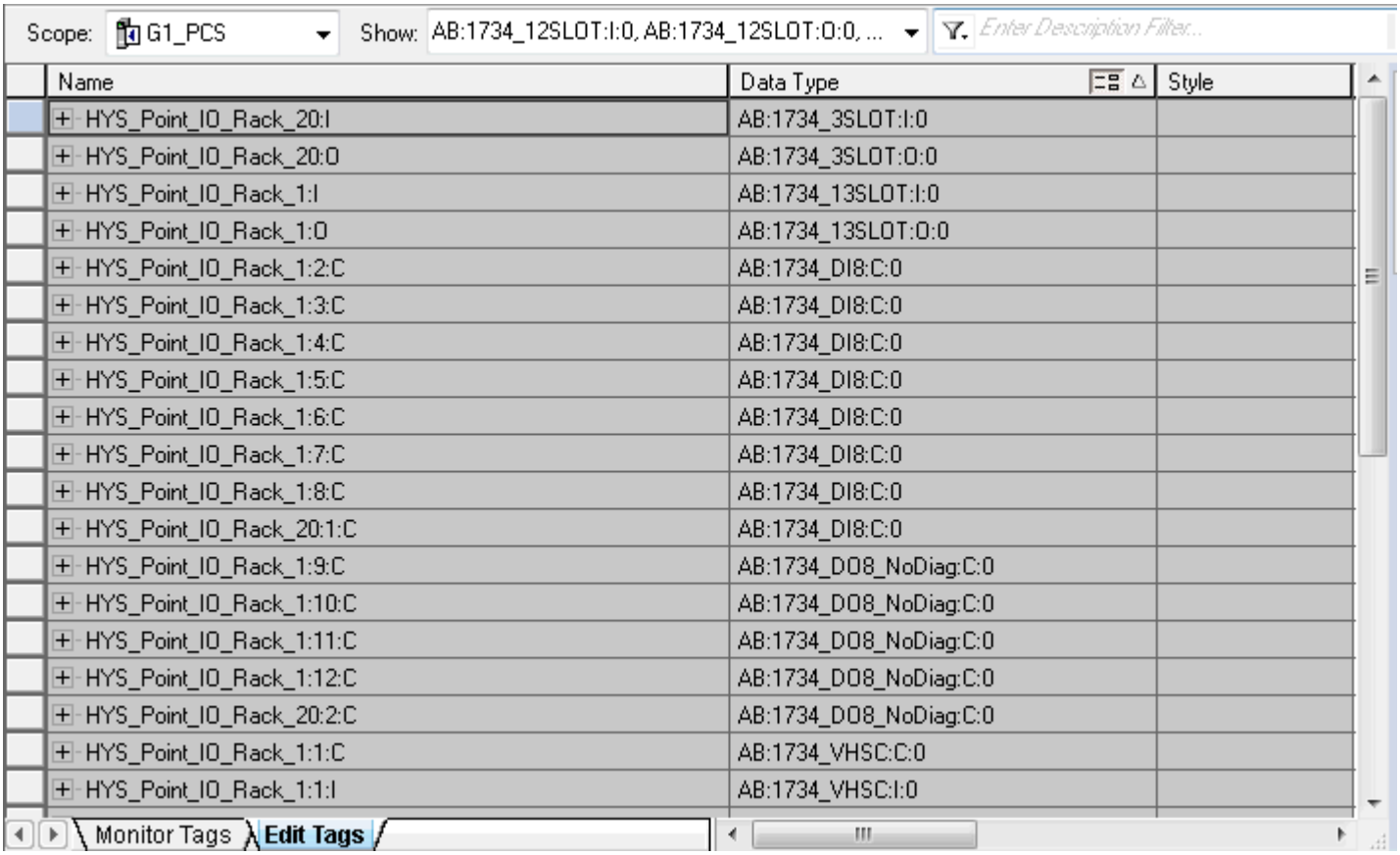


2. Variables de filtre pour afficher uniquement les variables **Module-Defined**.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

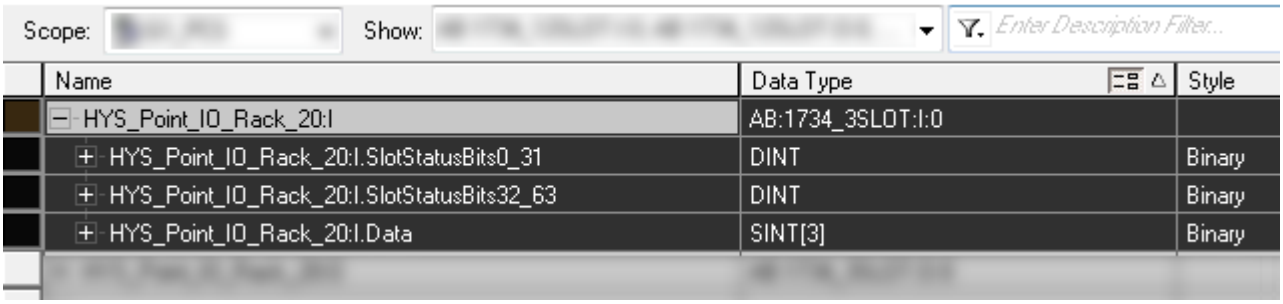


Seules les variables (alias) avec des types de données qui appartiennent à des modules I/O en option s'afficheront.



Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Dans cet exemple, l'alias HYS_Point_IO_Rack_20:I désigne le type de données AB:1734_3SLOT:I:0. Développez cette variable pour afficher la structure de ce type de données :



Pour garantir que HYS_Point_IO_Rack_20:I, et toutes ses sous-variables soient importées dans le projet, ouvrez le fichier ETIPSpecialDataTypes.xml dans un éditeur de texte et vérifiez s'il contient le type de données AB:1734_3SLOT:I:0. Si c'est le cas, vous pouvez continuer par le type de données suivant.

La structure peut se présenter, par exemple, comme suit :

```
<DataType Name="aaa">
<Members>
<Member Name="bbb" DataType="ccc" Dimension="ddd" Radix="eee"/>
</Members>
</DataType>
```

où :

- `aaa` = Alias/Type de données de variable
- `bbb` = Nom de sous-variable (le nom de la sous-variable suit le point)
- `ccc` = Type de données de la sous-variable
- `ddd` = Dimension de tableau (0 s'il ne s'agit pas d'un tableau)
- `eee` = Style

Dans l'exemple ci-dessus :

ETIPSpecialDataTypes.xml

```
238
239  <DataType Name="AB:1734_3SLOT:I:0">
240  <Members>
241   <Member Name="SlotStatusBit0_31" DataType="DINT" Dimension="0" Radix="Binary"/>
242   <Member Name="SlotStatusBit32_63" DataType="DINT" Dimension="0" Radix="Binary"/>
243   <Member Name="Data" DataType="SINT" Dimension="3" Radix="Binary"/>
244  </Members>
245  </DataType>
```

3. Répétez l'étape 2 pour tous les types de données Définis par module.
4. Répétez la procédure depuis l'étape 2, en filtrant les variables pour afficher uniquement les variables **Prédéfinies**.

Modèle contrôleur Omron Sysmac

Il est possible d'accéder aux données dans les contrôleurs NJ et CJ via le protocole CIP.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Chaque élément de données peut être identifié par une chaîne de caractères appelée « variable ». Utilisez des outils de programmation appropriés pour le contrôleur d'exportation de la liste des variables.

Les contrôleurs de la série NJ sont programmés dans Sysmac Studio :

- NJ301-xxxx
- NJ501-xxxx

Les contrôleurs de la série CJ sont programmés dans CX-One :

- CJ2M CPU-3x
- CJ2H CPU 6x-EIP
- Tout processeur avec une unité CJ1W-EIP21 associée.

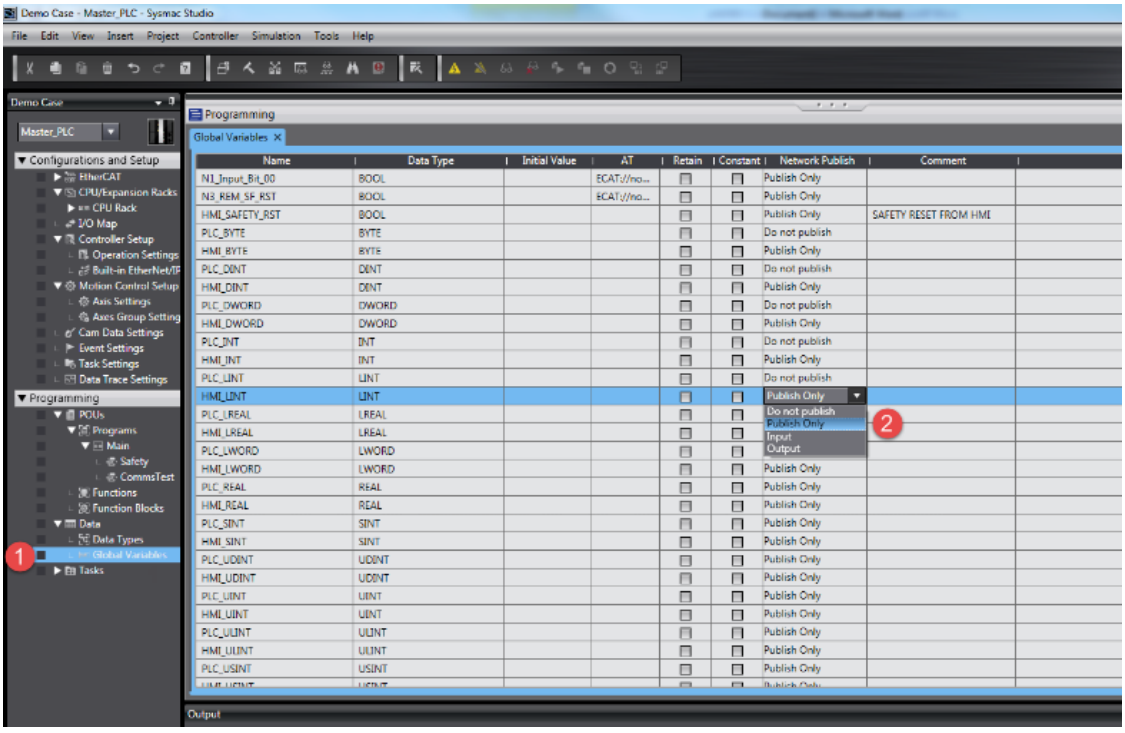
Le projet chargé sur le périphérique IHM doit se référer aux noms de variables attribués dans le logiciel de programmation au moment du développement. L'éditeur de variable supporte l'importation directe du fichier de variable généré par le logiciel Sysmac Studio au format .NJF ou généré par CX-One au format.CJF.

Toutes les variables qui peuvent être consultées par le périphérique IHM doivent être déclarées comme des Variables globales.

Exportation des fichiers NJF en utilisant Sysmac Studio

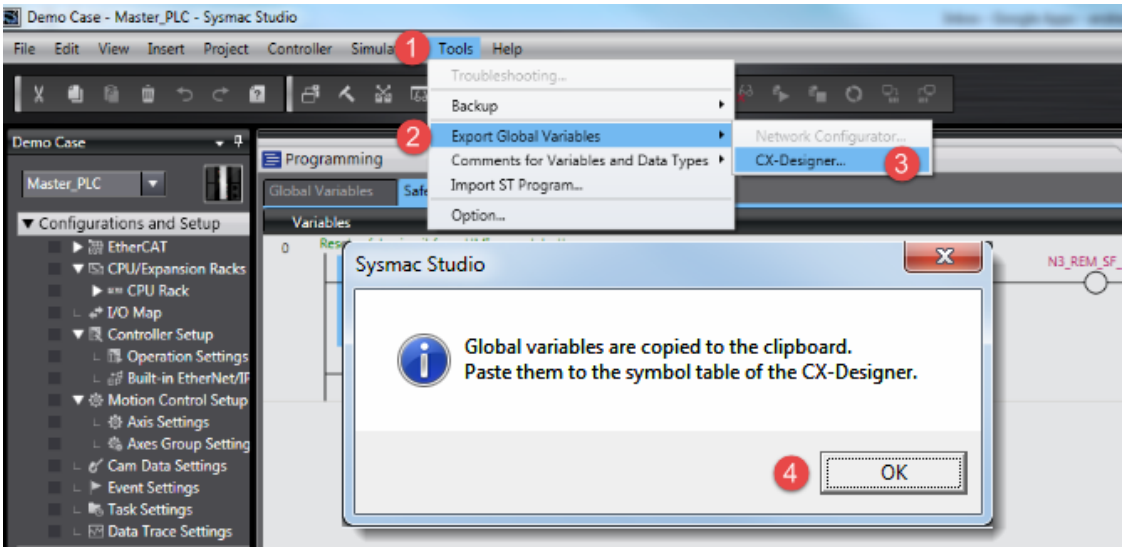
Pour exporter le fichier variable **.NJF** :

1. Dans Sysmac Studio, déclarez les variables comme **Variables globales**.
2. Configurez l'attribut de **Publication réseau** sur **Publier uniquement**.

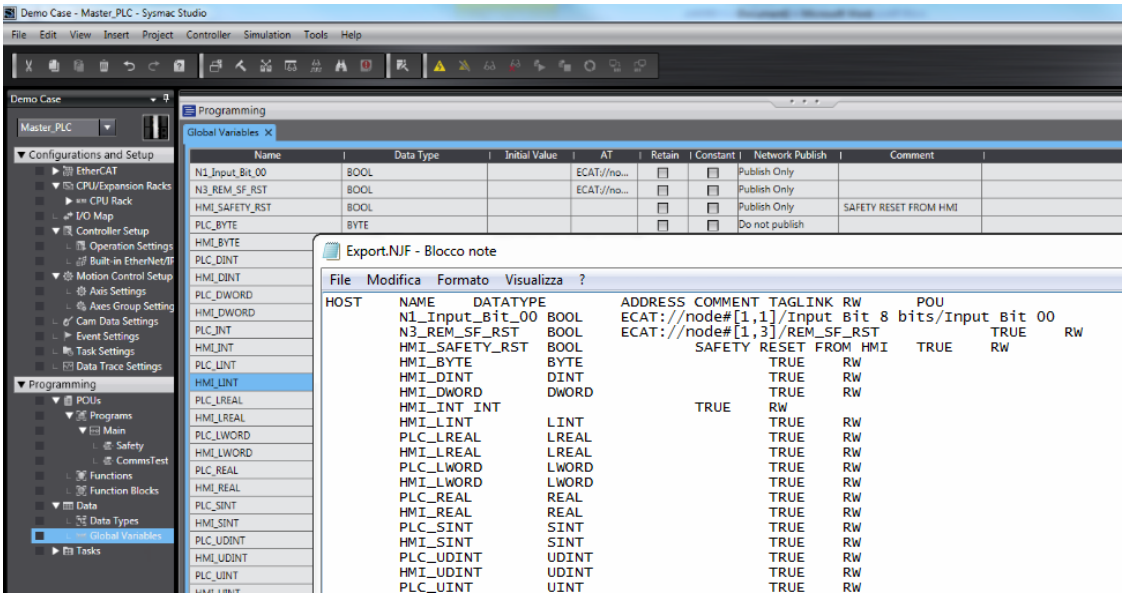


2. Dans le menu **Outils**, sélectionnez **Exporter variables globales > CX-Designer**.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



3. Cliquez sur **OK** pour confirmer.
4. Coupez et collez les contenus du presse-papier dans un éditeur de texte.



4. Enregistrez le fichier sous **.NJF**.

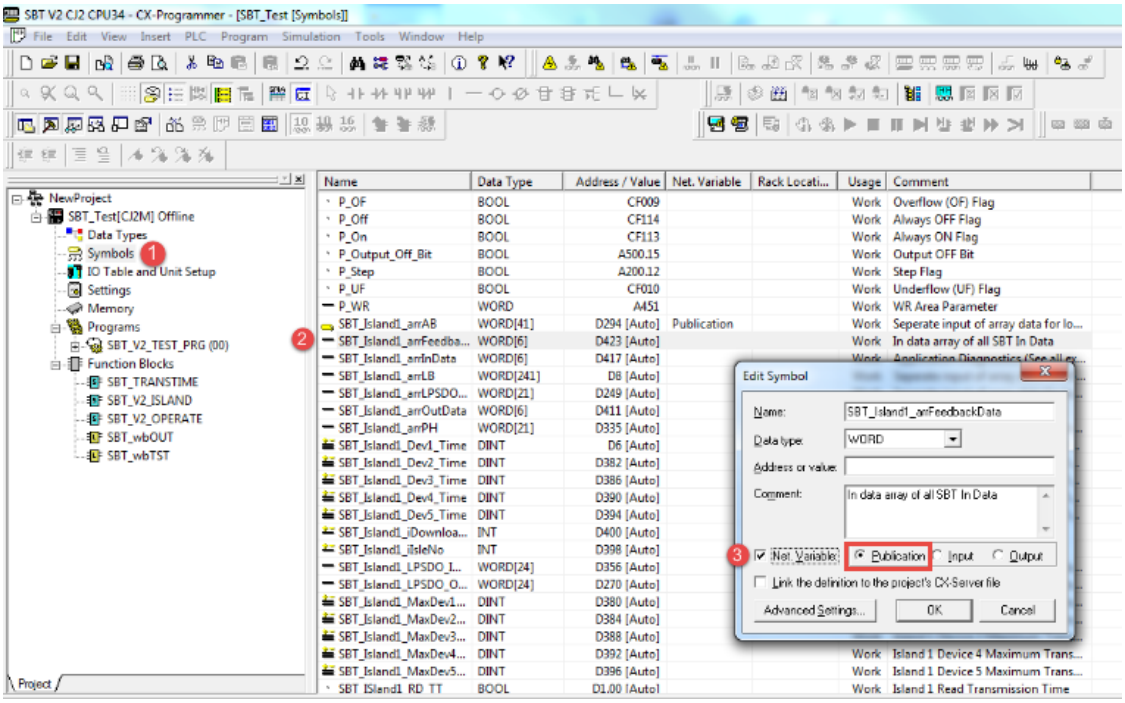
Remarque : en utilisant le Bloc-notes comme éditeur de texte, assurez-vous d'enregistrer le fichier texte avec l'extension **.NJF** en sélectionnant « Enregistrer sous » comme « tous les fichiers » bien que le fichier sera nommé *.njf.txt et il ne sera pas visible depuis l'importateur.

Exportation du fichier CJF en utilisant CX-One

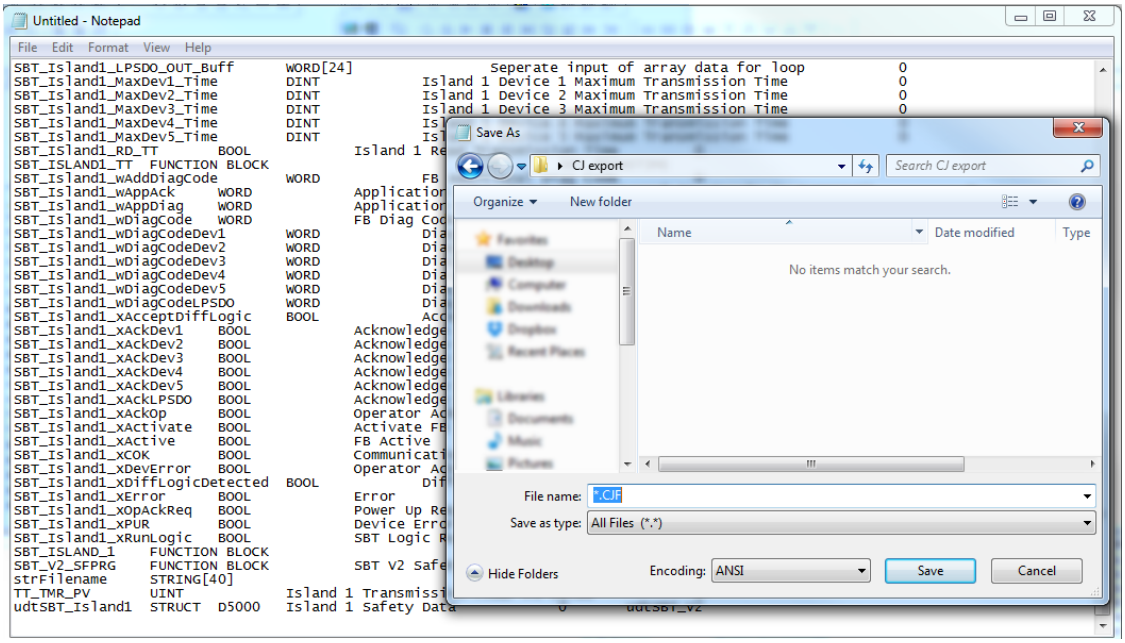
Pour exporter le fichier variable **.CJF** :

1. Dans CX-One, ouvrez le fichier de symboles du projet.
2. Dans le dialogue **Éditer symbole**, configurez l'attribut des **Variables réseau** sur **Publication**.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



3. Copiez et collez tous les variables dans un éditeur de texte.



4. Enregistrez le fichier sous **.CJF**.

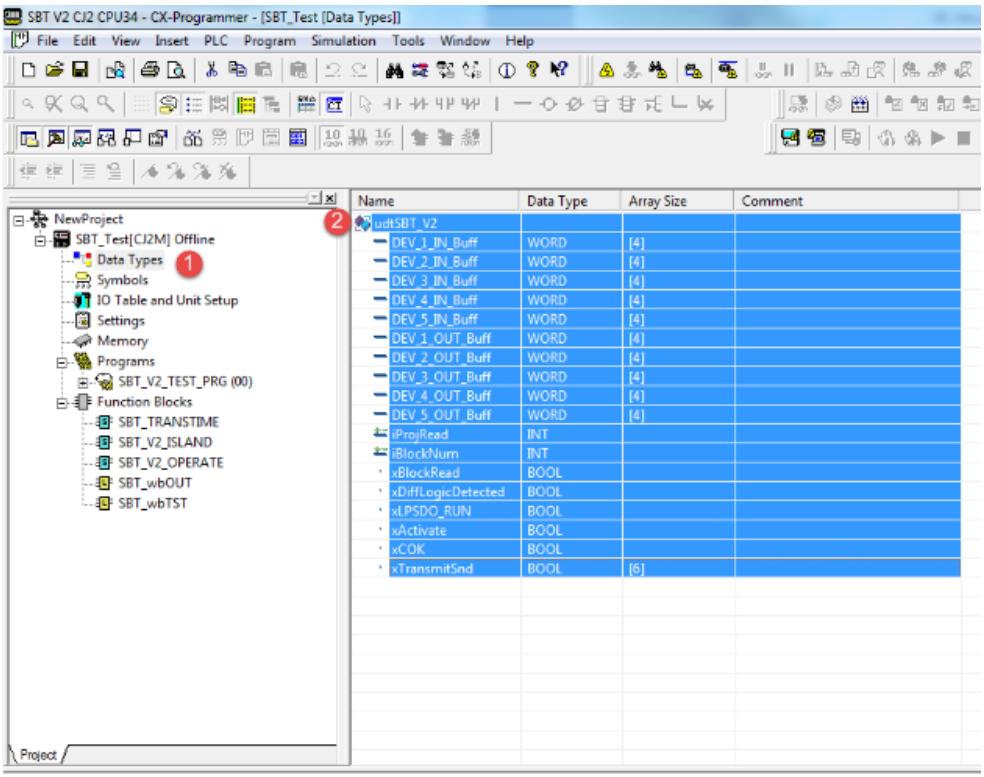
Remarque : en utilisant le Bloc-notes comme éditeur de texte, assurez-vous d'enregistrer le fichier texte avec l' extension **.CJF** en sélectionnant « Enregistrer sous » comme « tous les fichiers » bien que le fichier sera nommé *.cjf.txt et il ne sera pas visible depuis l'importateur.

Exportation des structures définies de l'utilisateur

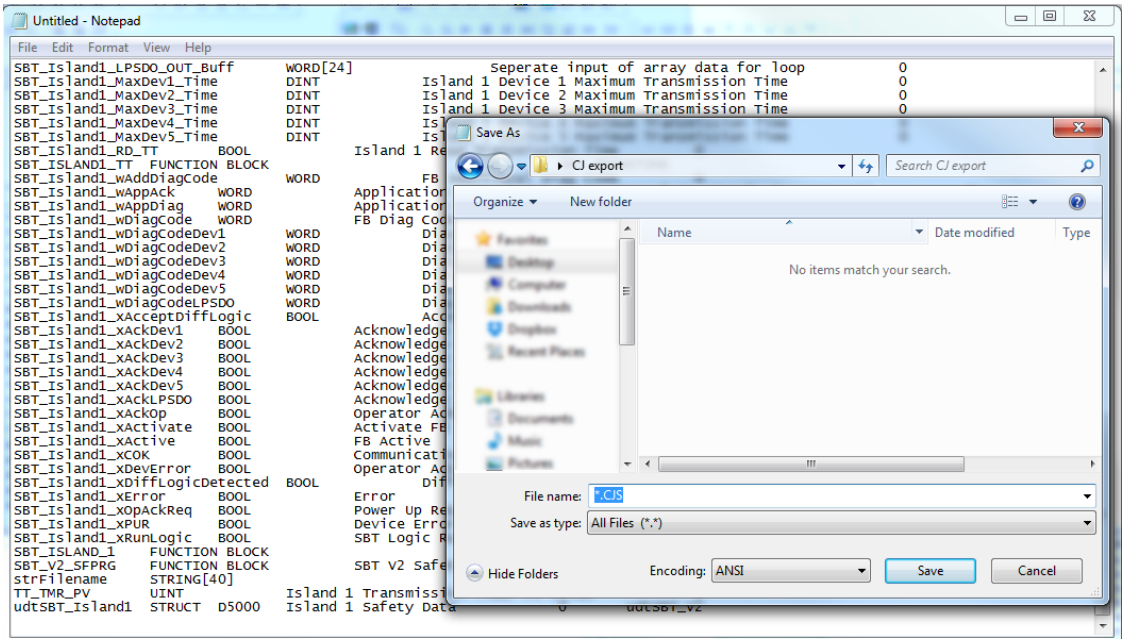
Pour exporter le fichier variable **.CJS** :

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

1. Dans CX-One, ouvrez le fichier de types de données du projet.



2. Copiez et collez tous les variables dans un éditeur de texte.



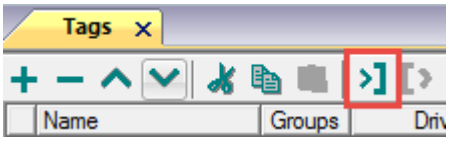
3. Enregistrez le fichier sous **.CJS**.

Remarque : en utilisant le Bloc-notes comme éditeur de texte, assurez-vous d'enregistrer le fichier texte avec l' extension **.CJS** en sélectionnant « Enregistrer sous » comme « tous les fichiers » bien que le fichier sera nommé *.cjs.txt et il ne sera pas visible depuis l'importateur.

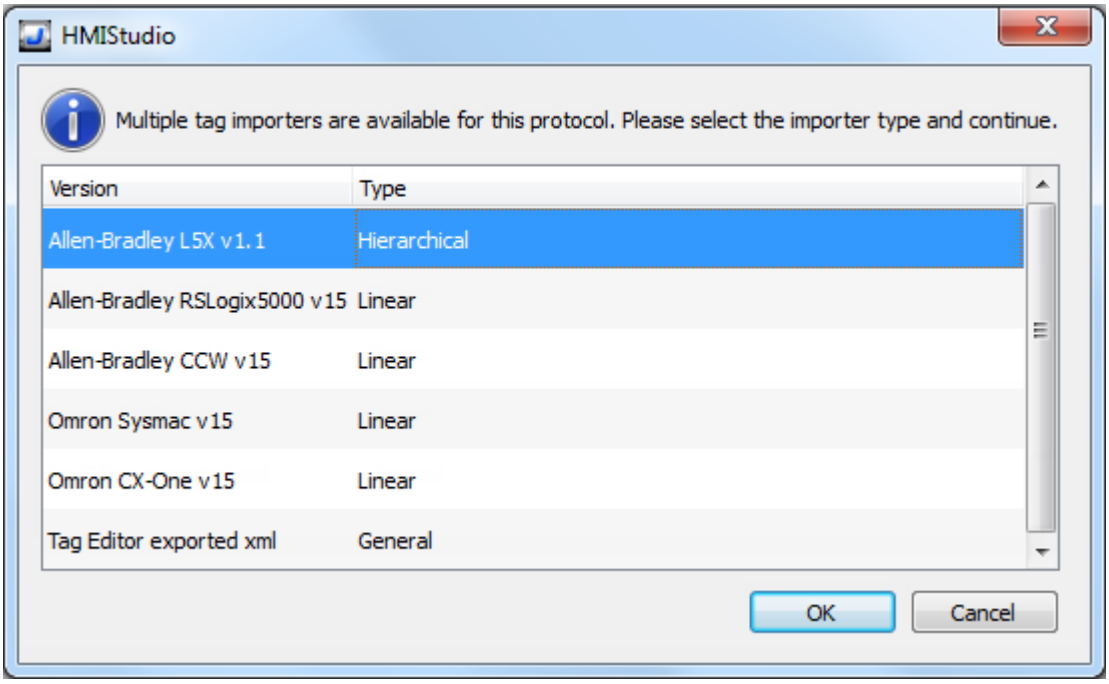
Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Importer des fichiers dans l'éditeur

Sélectionnez le pilote dans l'éditeur de variables et cliquez sur le bouton **Importer variables** pour lancer le pilote d'importation.



La boîte de dialogue suivante affiche le type d'importation que vous pouvez sélectionner.

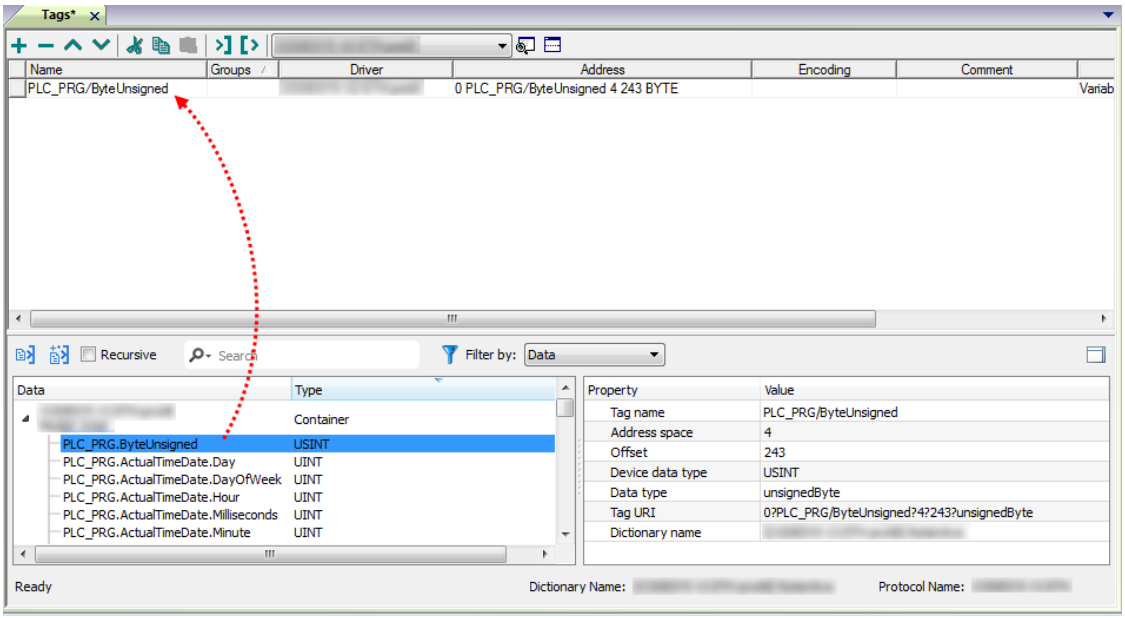


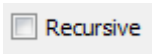
Sélectionnez **Omron Sysmac** pour importer un fichier de variables **.NJF** ou **Omron CX-One** pour importer un fichier de variables **.CJF**.

Lorsque vous avez sélectionné le pilote d'importation, localisez le fichier de variables et cliquez sur **Ouvrir**. Le système vous demandera le fichier **.CJS** des structures définies par l'utilisateur. Si non requis, passez le dialogue en cliquant sur le bouton Annuler.

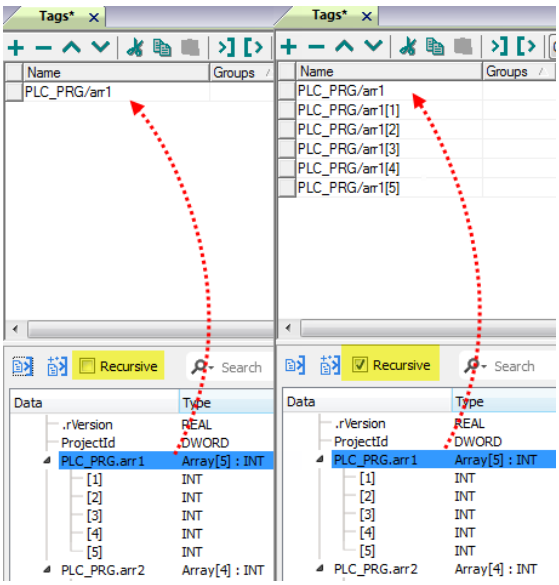
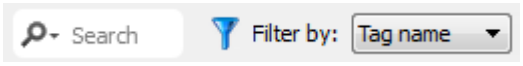
Les variables comprises dans le fichier de symboles sont listées dans le dictionnaire des variables. Le dictionnaire des variables est affiché en bas de l'écran.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



Élément de la barre d'outils	Description
	Importer variable(s). Sélectionnez les variables à importer et cliquez sur cette icône pour ajouter les variables du dictionnaire au projet.
	Mettre à jour variable(s). Cliquez sur cette icône pour mettre à jour les variables du projet grâce à un nouvel import du dictionnaire.
	Cochez cette case pour importer tous les sous-éléments d'une variable. Exemples de résultats avec la case cochée et non cochée :

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Élément de la barre d'outils	Description
	
	Recherche des variables dans le dictionnaire en filtrant en fonction de l'élément sélectionné dans la liste déroulante.

Remarque : lors de l'importation des types de données de tableau, l'importateur les étend en créant des variables individuelles par chaque élément du tableau ; ceci est valable pour tous les types de données, sauf pour les tableaux booléen. Dans ce cas, ils sont importés en tant que « booléen-32 » et l'élément de tableau unique peuvent être traité en utilisant le paramètre « Tag Index » depuis la boîte de dialogue « Joindre à ... ».

Modèle contrôleur Micro800

Le pilote Ethernet/IP CIP constitue un moyen facile et fiable de connexion aux contrôleurs Allen-Bradley Micro800.

La portée des variables dans un contrôleur Micro800 peut être limitée à un programme local ou elle peut être globale :

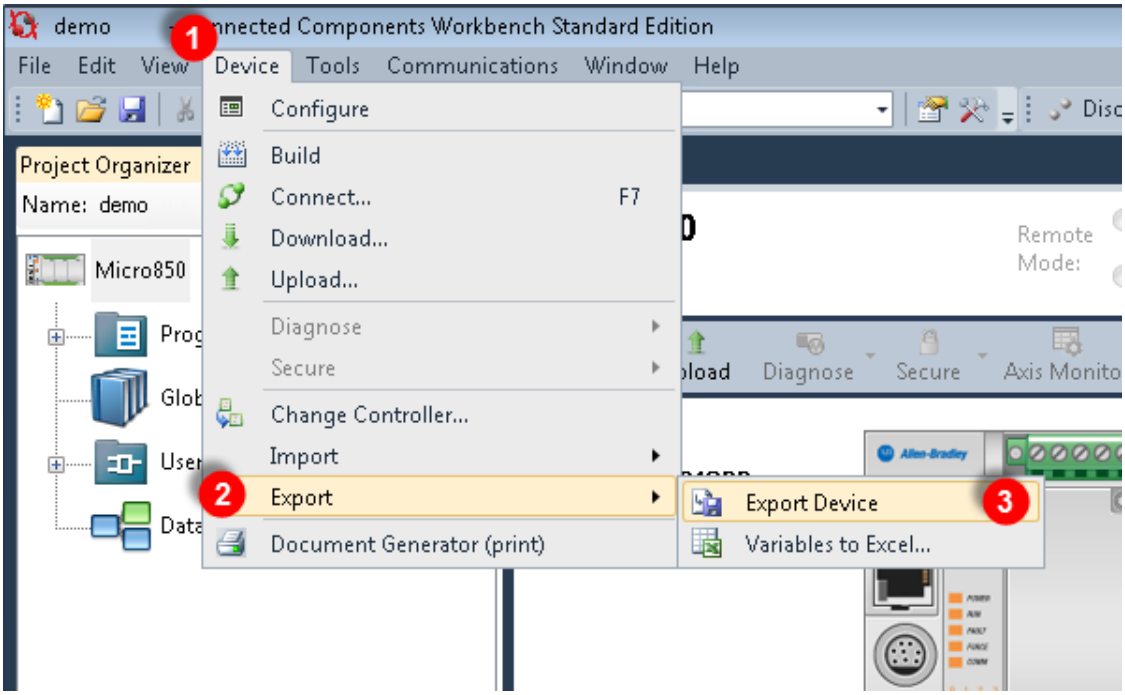
But	Description
Variables locales	Variables à portée de programme. Les variables sont affectées au programme spécifique dans le projet et disponibles uniquement pour ce programme. Ces variables ne sont **pas supportées** par ce pilote.
Variables globales	Variables à portée de contrôleur. Les variables appartiennent au contrôleur dans le projet et sont disponibles à tout programme dans le projet. Ces variables sont **supportées** dans ce pilote.

Exportation du fichier ISAXML en utilisant le Workbench de composant connecté

Pour exporter les variables globales **.ISAXML** y compris les variables IE/S :

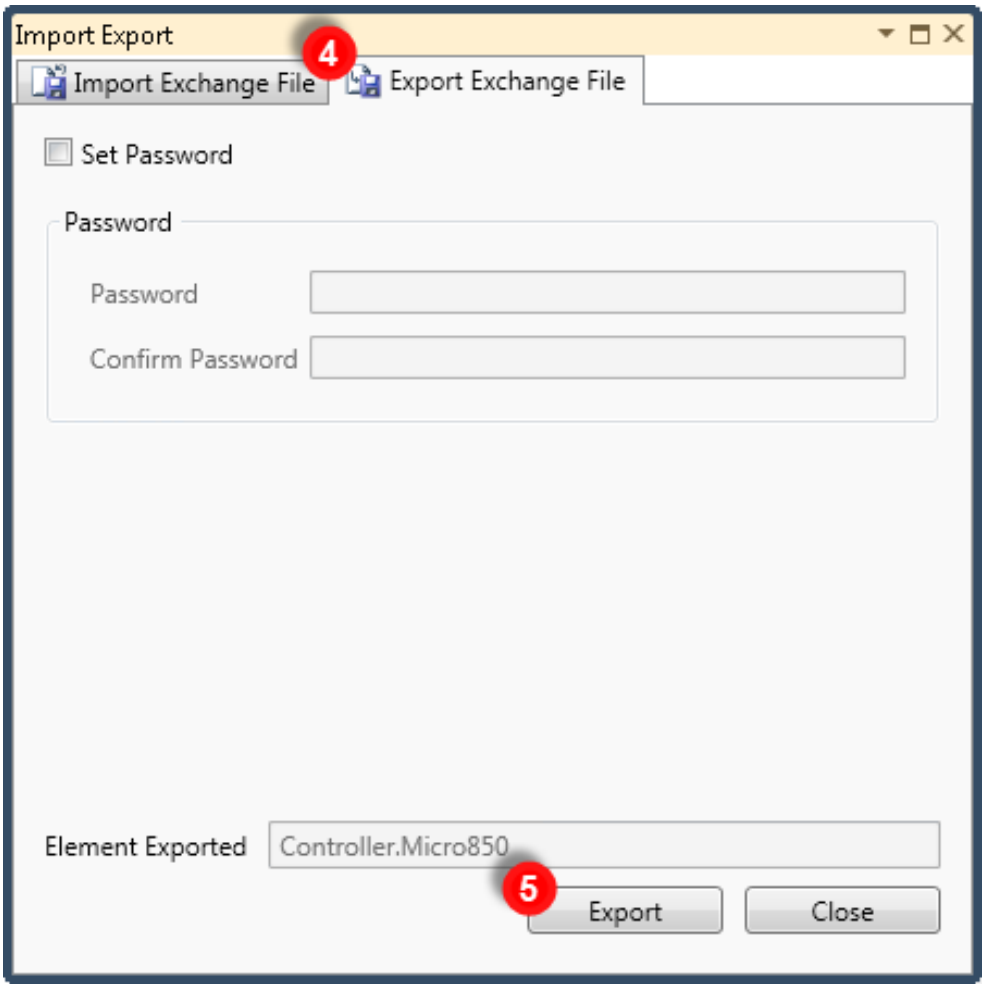
Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

1. Sélectionnez l'onglet **Périphérique**.
2. Étendez l'élément **Export**.
3. Sélectionnez l'**export du périphérique**.

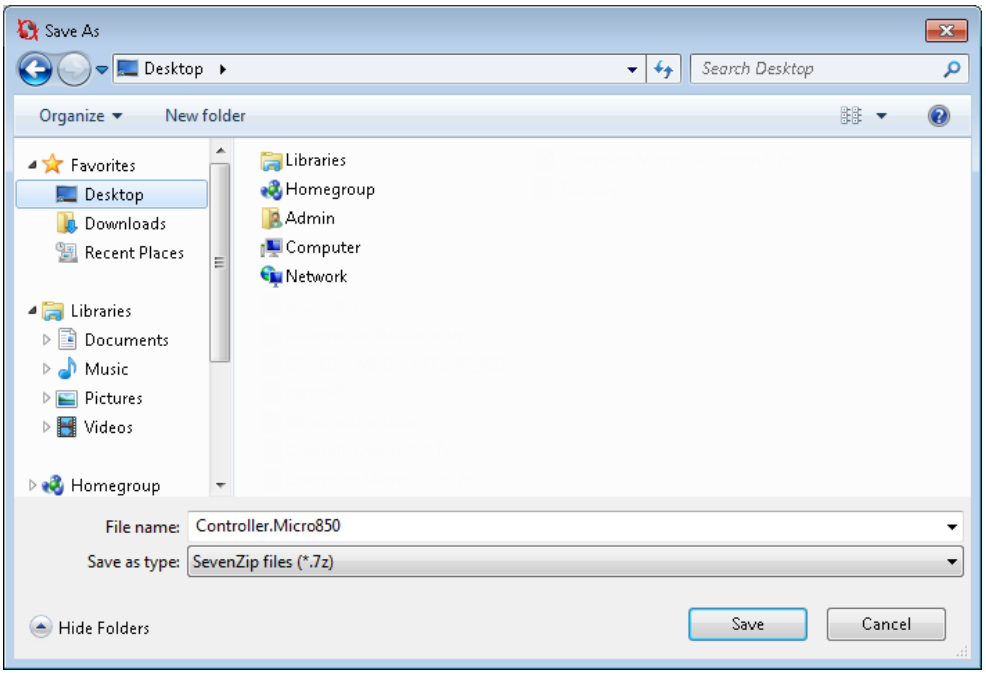


4. Cliquez sur l'onglet **Exportation du fichier d'échange**.
5. Cliquez sur le bouton **Export**.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

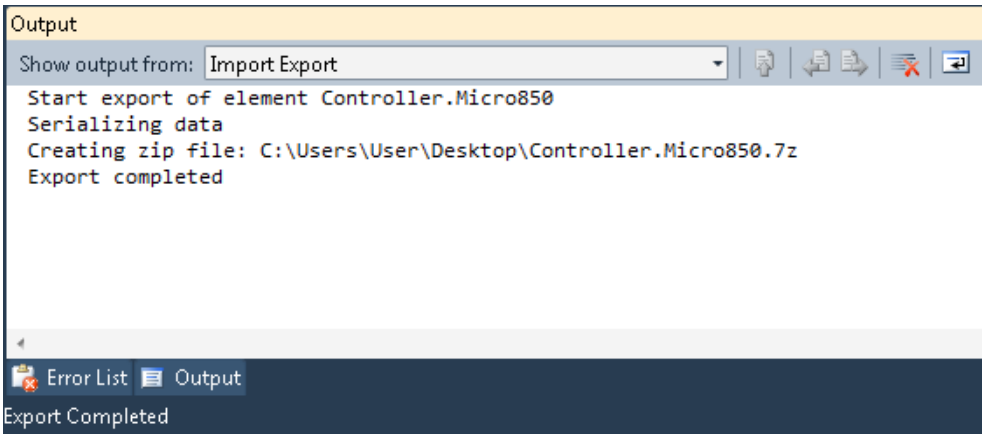


6. Sélectionnez un emplacement pour l'enregistrement du fichier d'exportation et cliquez sur **Enregistrer**.



7. Lorsque l'exportation s'est terminée avec succès, les informations de sortie s'affichent :

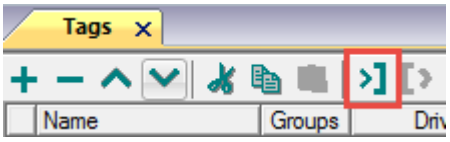
Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



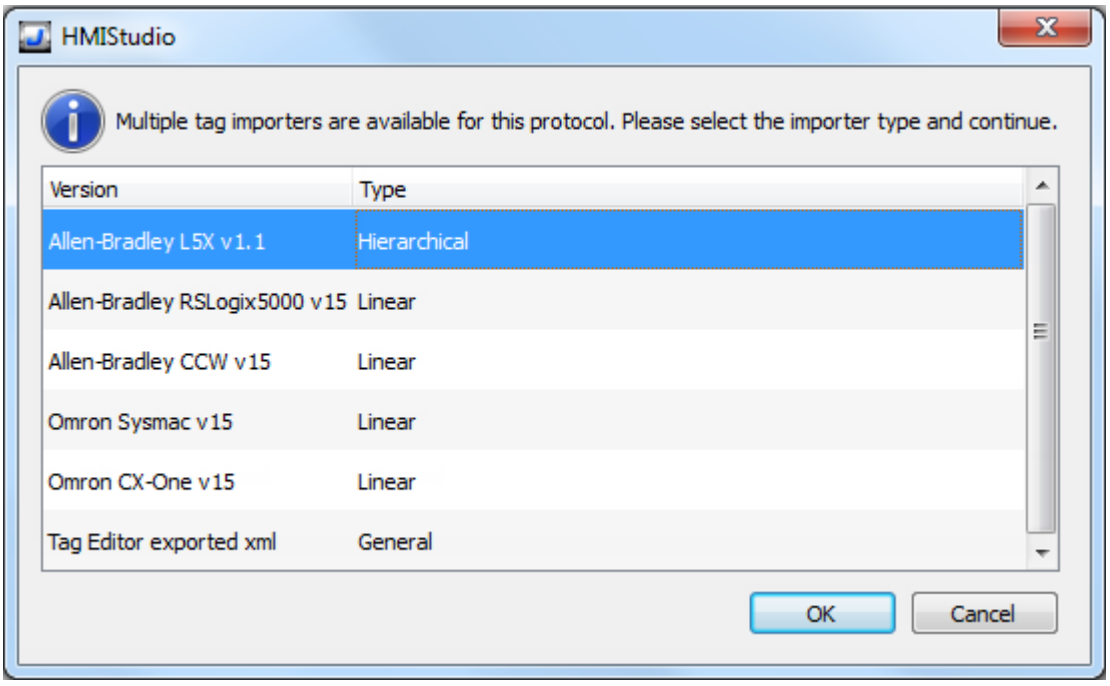
Remarque : Le fichier d'exportation est un fichier d'archives comprimé avec 7-zip. Utilisez un utilitaire zip approprié pour extraire le contenu archive dans un dossier local.

Importer des fichiers dans l'éditeur

Sélectionnez le pilote dans l'éditeur de variables et cliquez sur le bouton **Importer variables** pour lancer le pilote d'importation.



La boîte de dialogue suivante affiche le type d'importation que vous pouvez sélectionner.

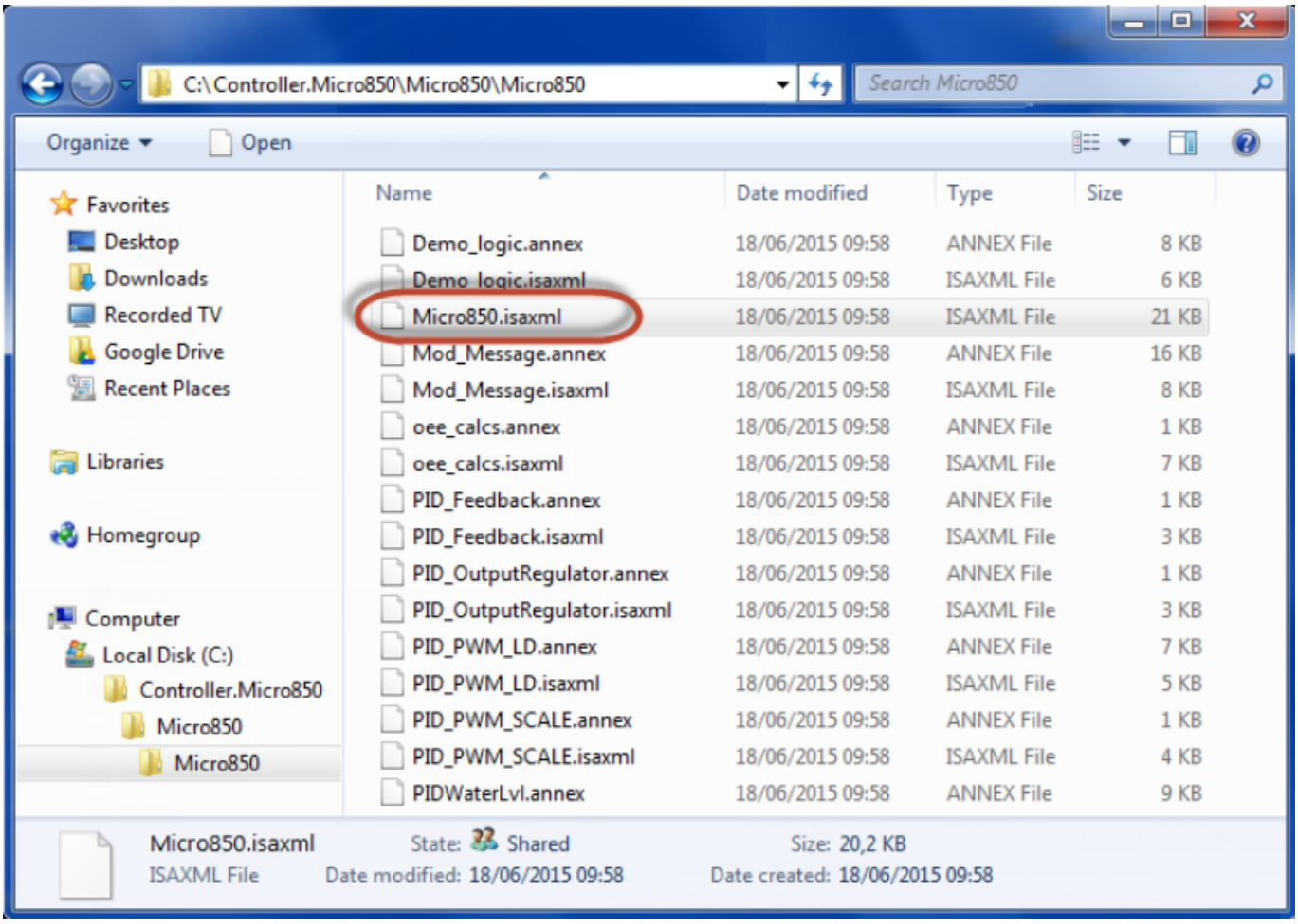


Sélectionner l'option **Allen-Bradely CCW v15**.

Structure du répertoire extraite de fichier 7z est quelque chose comme : "..\<folder_name>\Micro8xx\Micro8xx\"

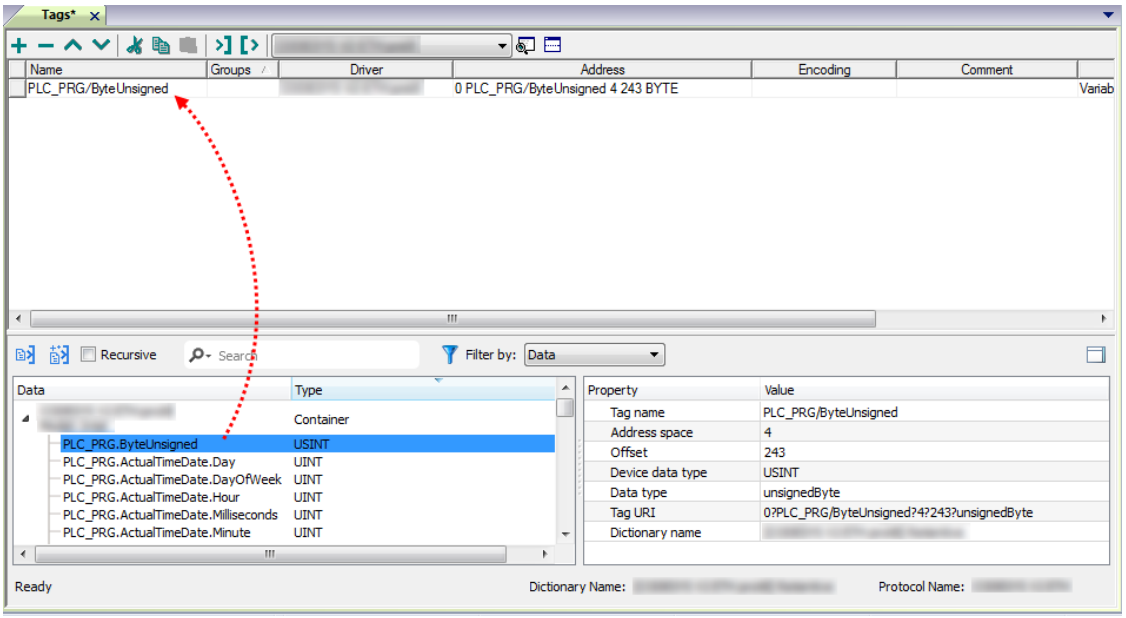
Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Dans ce dernier dossier, sélectionnez le fichier Micro8xx.isaxml comme montré ci-dessous :

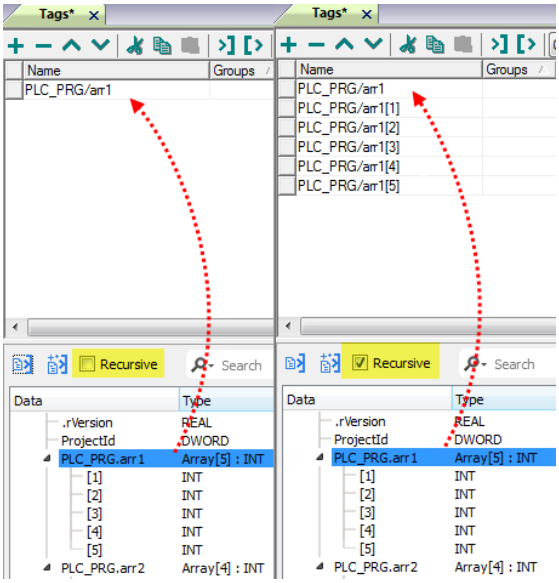
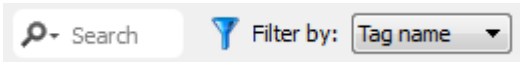


Lorsque vous avez sélectionné le pilote d'importation, localisez le fichier de symboles et cliquez sur **Ouvrir**.

Les variables comprises dans le fichier de symboles sont listées dans le dictionnaire des variables. Le dictionnaire des variables est affiché en bas de l'écran.



Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Élément de la barre d'outils	Description
	Importer variable(s). Sélectionnez les variables à importer et cliquez sur cette icône pour ajouter les variables du dictionnaire au projet.
	Mettre à jour variable(s). Cliquez sur cette icône pour mettre à jour les variables du projet grâce à un nouvel import du dictionnaire.
	Cochez cette case pour importer tous les sous-éléments d'une variable. Exemples de résultats avec la case cochée et non cochée : 
	Recherche des variables dans le dictionnaire en filtrant en fonction de l'élément sélectionné dans la liste déroulante.

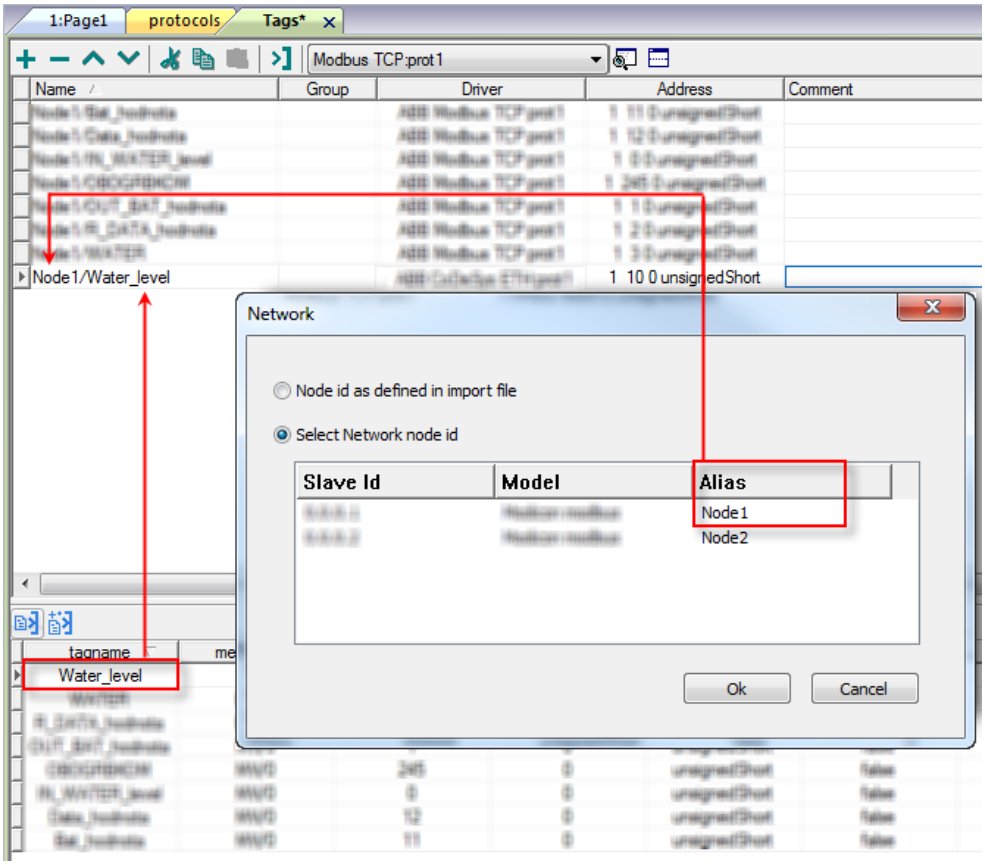
Ajouter un nom d'alias à un protocole

Les noms de variables doivent être uniques au niveau d'un projet. Vous pouvez cependant être amené à utiliser les mêmes noms de variables pour différents nœuds de contrôleur (par exemple lorsque le périphérique IHM est connecté à deux périphériques exécutant la même application).

Lorsque vous générez un protocole, vous pouvez ajouter un nom d'alias qui s'ajoutera aux noms de variables importés pour ce protocole.

Dans l'exemple, la connexion à un certain contrôleur porte le nom **Nœud1**. Lorsque vous importez les variables pour ce nœud, tous les noms de variables porteront le préfixe **Nœud1**, les rendant ainsi uniques au niveau projet/réseau.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



Remarque : vous pouvez donner un nom d'alias uniquement aux variables importées. Il n'est pas nécessaire pour les variables ajoutées manuellement dans l'éditeur de variables de porter un préfixe d'alias dans leur nom. La chaîne d'alias est associée à l'importation. Une fois l'importation des variables effectuée, une modification de la chaîne d'alias n'aura aucun effet sur les noms figurant déjà dans le dictionnaire. Lorsque vous modifiez la chaîne d'alias et réimportez les variables, toutes les variables réimportées porteront la nouvelle chaîne de préfixe.

IP d'écrasement de nœud

Le protocole propose le type de données spécial IP d'écrasement de nœud, qui vous permet de modifier l'adresse IP du contrôleur cible au moment du runtime.

Ce type de mémoire est un tableau de 4 octets non signés, un pour chaque octet de l'adresse IP.

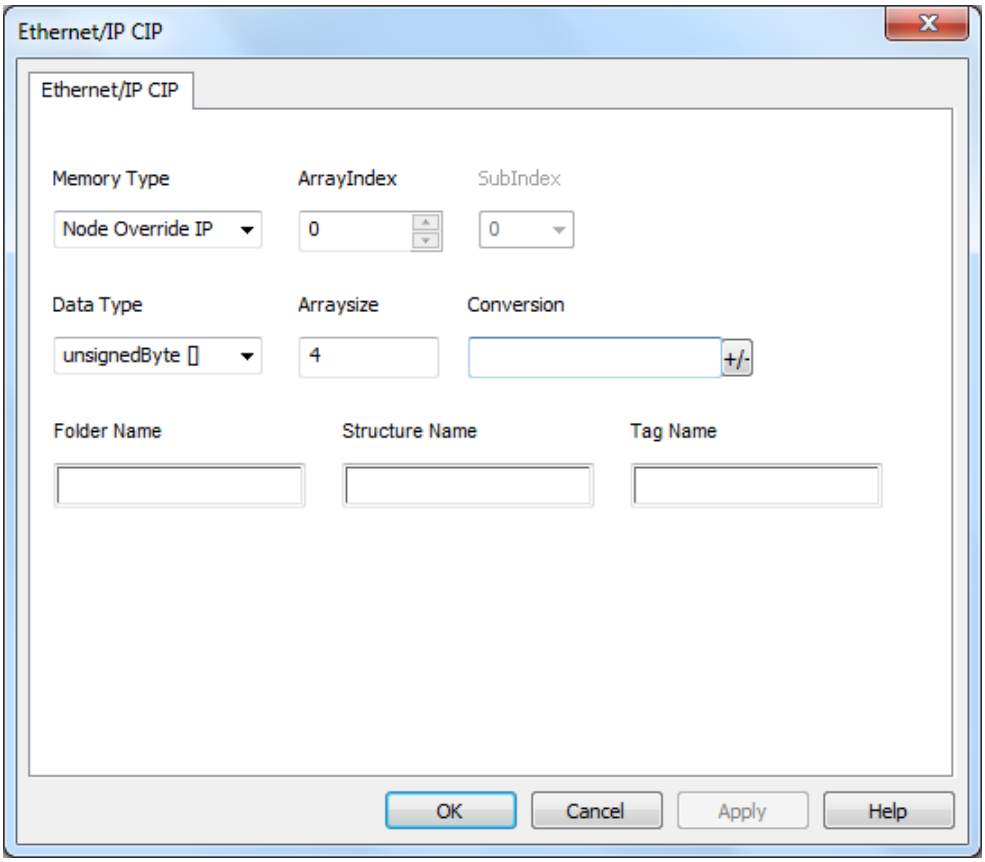
L'IP d'écrasement de nœud est initialisée avec la valeur de l'IP du contrôleur spécifiée dans le projet au moment de la programmation.

IP d'écrasement de nœud	Exécution de l'API
0.0.0.0	La communication avec le contrôleur est coupée, plus aucune trame de requête ne sera générée.
Autre que 0.0.0.0	Interprétée comme étant l'écrasement de l'IP de nœud, et remplace l'adresse IP cible au moment du runtime.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

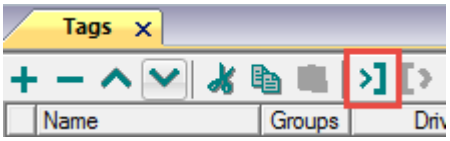
Lorsque le périphérique IHM est connecté à un réseau avec plus d'un nœud de contrôleur, chaque nœud dispose de sa propre variable d'IP d'écrasement de nœud.

Remarque : le maintien des valeurs de l'IP d'écrasement de nœud assignées au runtime est assuré par des cycles d'alimentation.



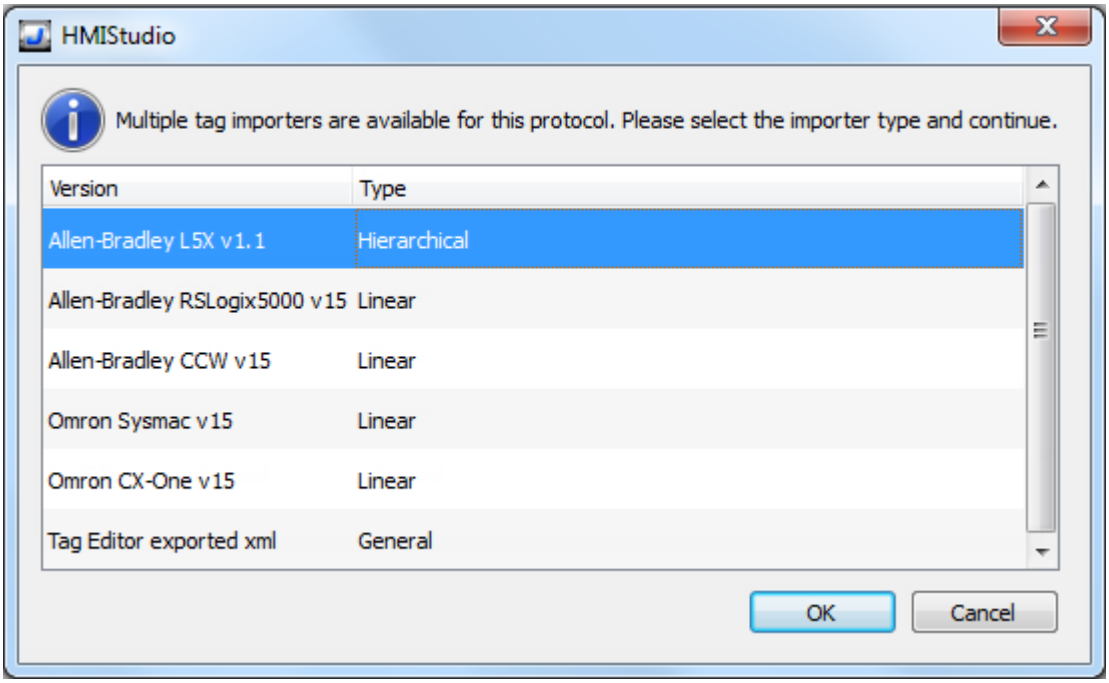
Importation de variables

Sélectionnez le pilote dans l'éditeur de variables et cliquez sur le bouton **Importer variables** pour lancer le pilote d'importation.



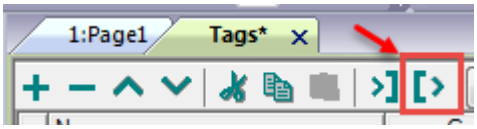
La boîte de dialogue suivante affiche le type d'importation que vous pouvez sélectionner.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



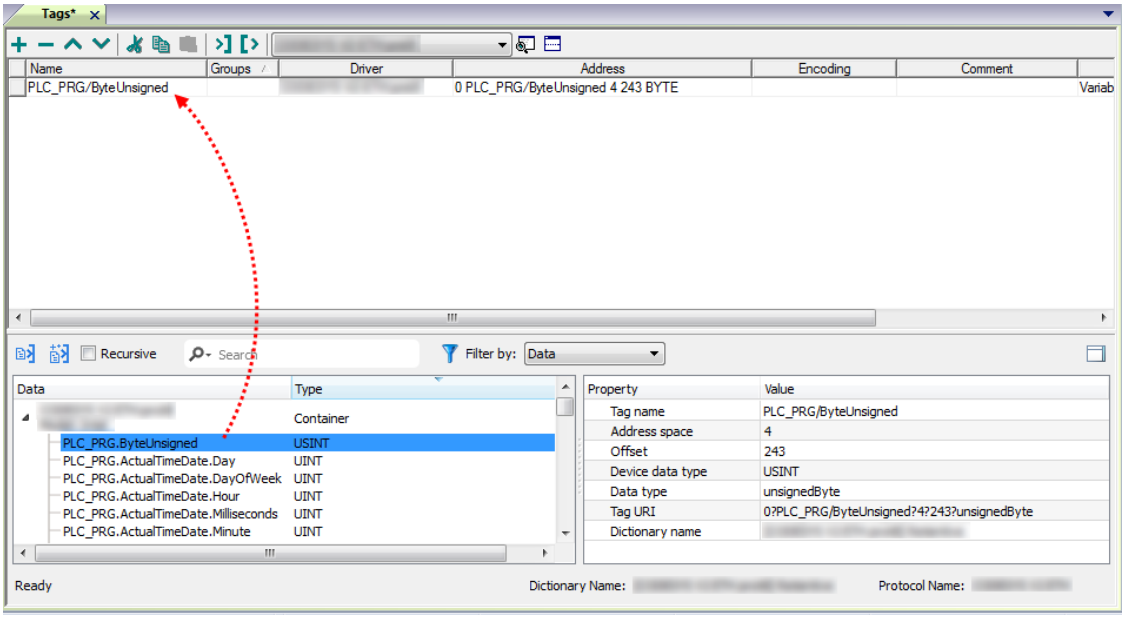
Importateur	Description
Allen-Bradley L5X v1.1 Hiérarchique	Requiert un fichier .L5X. Vérifier le **Modèle du contrôleur Logix 5000** pour plus de détails. Toutes les variables s'affichent en fonction de l'affichage hiérarchique RSLogix5000.
Allen-Bradley RSLogix5000 v15 Linéaire	Requiert un fichier **.CSV** et des fichiers **.L5X** (en option). Vérifier le **Modèle du contrôleur Logix 5000** pour plus de détails. Toutes les variables s'affichent au même niveau.
Allen-Bradley CCW v15 Linéaire	Requiert un fichier **.ISAXML**. Vérifier le **Modèle du contrôleur Micro800** pour plus de détails. Toutes les variables s'affichent au même niveau.
Omron Sysmac v15 Linéaire	Requiert un fichier **.NJF**. Vérifier le **Modèle du contrôleur Omron Sysmac** pour plus de détails. Toutes les variables s'affichent au même niveau.

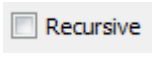
Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Importateur	Description
Omron CX-One v15 Linéaire	Requiert un fichier **.CJF** et des fichiers **.CJS** (en option). Vérifier le **Modèle du contrôleur Omron Sysmac** pour plus de détails. Toutes les variables s'affichent au même niveau.
Fichier .xml exporté de l'éditeur de variables	Sélectionnez ce pilote d'importation pour lire un fichier XML générique exporté de l'éditeur de variables avec le bouton approprié. 

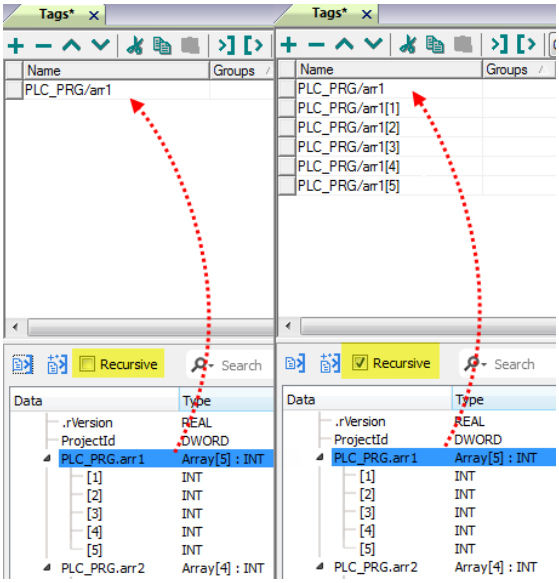
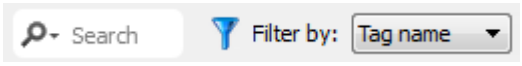
Lorsque vous avez sélectionné le pilote d'importation, localisez le fichier de symboles et cliquez sur **Ouvrir**.

Les variables comprises dans le fichier de symboles sont listées dans le dictionnaire des variables. Le dictionnaire des variables est affiché en bas de l'écran.



Élément de la barre d'outils	Description
	Importer variable(s). Sélectionnez les variables à importer et cliquez sur cette icône pour ajouter les variables du dictionnaire au projet.
	Mettre à jour variable(s). Cliquez sur cette icône pour mettre à jour les variables du projet grâce à un nouvel import du dictionnaire.
	Cochez cette case pour importer tous les sous-éléments

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Élément de la barre d'outils	Description
	d'une variable. Exemples de résultats avec la case cochée et non cochée : 
	Recherche des variables dans le dictionnaire en filtrant en fonction de l'élément sélectionné dans la liste déroulante.

Statut de communication

Vous pouvez afficher le statut de communication actuel grâce aux variables système. Voir la section « Variables système » du manuel principal.

Codes pris en charge pour ce pilote de communication :

Erreur	Cause	Action
NAK	Le contrôleur répond « Ne pas confirmer ».	-
Délai d'attente	La requête n'a obtenu aucune réponse dans le délai d'attente spécifié.	Vérifiez que le contrôleur est connecté et correctement configuré pour accéder au réseau.
Réponse non valide	Le contrôleur a envoyé une réponse ou un contenu non valide au périphérique.	Vérifiez la cohérence entre les données programmées dans le projet et les ressources du contrôleur.
Erreur générale	Erreur inconnue. Ne doit jamais être signalée.	Contactez l'assistance technique.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Modbus RTU

Vous pouvez connecter les panneaux opérateurs à un réseau Modbus qui fera office de réseau maître utilisant ce pilote de communication.

Détails sur l'implantation

La mise en œuvre du Modbus RTU prend seulement en charge un sous-ensemble des codes fonction du Modbus RTU standard.

Code	Fonction	Description
01	Lecture du statut de la bobine	Lecture de bits multiples dans la zone du périphérique bobine.
02	Lecture du statut d'entrée	Lecture du statut « MARCHE/ARRÊT » des entrées discrète (référence 1x) de l'esclave
03	Lecture des registres de maintien	Lecture de registres multiples
04	Lecture des registres d'entrée	Lecture du contenu binaire des registres d'entrée (référence 3x) de l'esclave
05	Forcer bobine seule	Force une bobine unique à passer en état « MARCHE » ou « ARRÊT »
06	Prédéfinir registre seul	Prédéfinit une valeur dans un registre
16	Prédéfinir registres multiples	Prédéfinit une valeur dans des registres multiples

Remarque : vitesse de communication avec les contrôleurs s'élève à 115 200 bauds.

Remarque : le format de données en virgule flottante est conforme à la norme IEEE.

Paramètres de l'éditeur de protocole

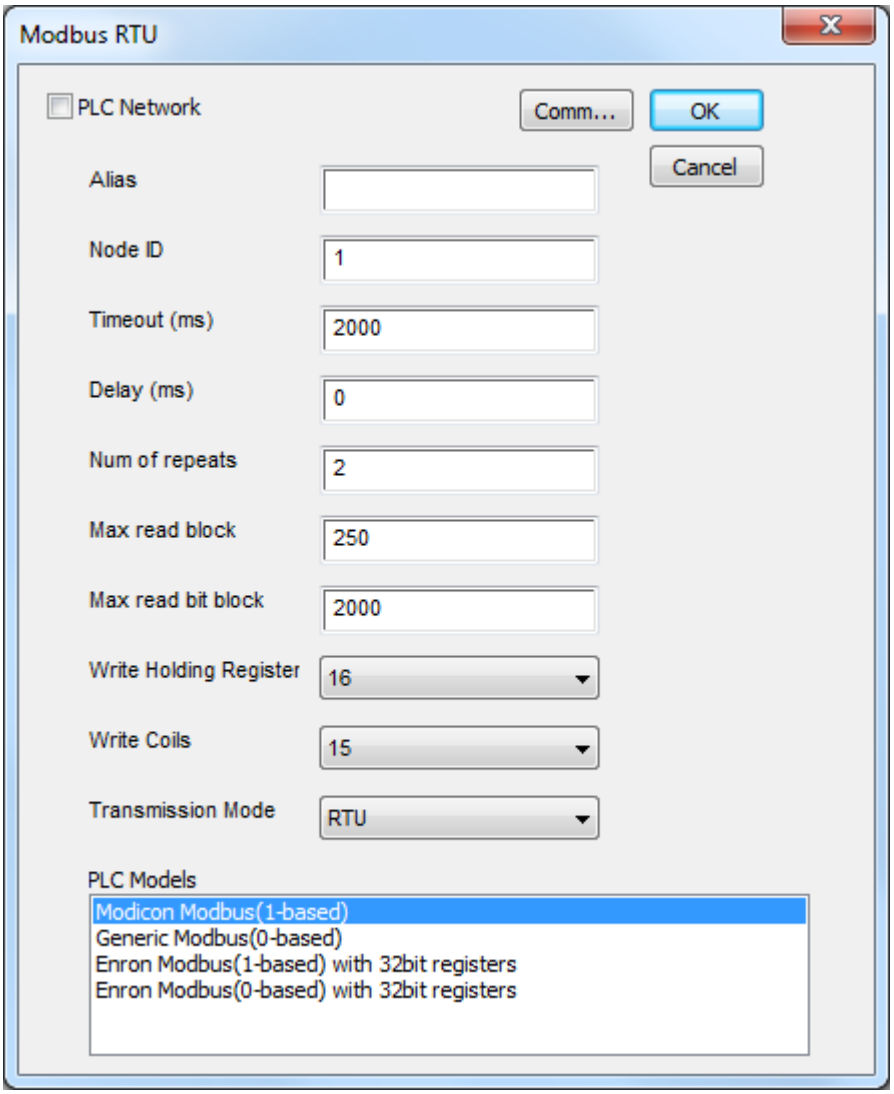
Ajouter un protocole

Pour configurer le protocole :

1. Dans le nœud **Config**, double-cliquez sur **Protocoles**.
2. Pour ajouter un pilote, cliquez sur **+** : une nouvelle ligne est ajoutée.
3. Sélectionnez le protocole dans la liste **API**.

La boîte de dialogue de configuration du protocole s'affiche.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

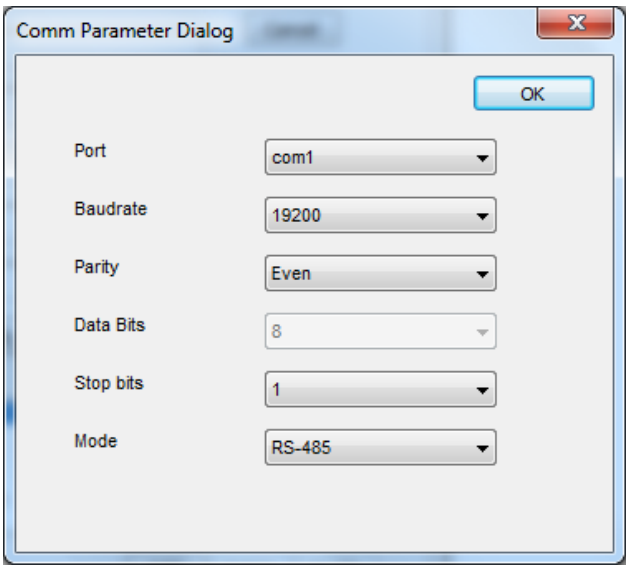


Élément	Description
Alias	Nom d'identification des nœuds dans les configurations réseau. Il est ajouté en préfixe à chaque nom de variable importé pour chaque nœud de réseau.
ID de nœud	Nœud Modbus du périphérique esclave.
Délai d'attente (ms)	Délai d'attente (en millisecondes) entre deux tentatives en cas d'absence de réponse du périphérique esclave.
Retard (ms)	Retard en millisecondes entre la fin de la dernière trame reçue et le début d'une nouvelle requête. S'il est à 0, le système interne envoie une nouvelle requête dès qu'il est capable de la programmer.
Nombre de répétitions	Nombre d'envois d'un certain message au contrôleur avant l'envoi d'un rapport d'erreur de communication. Lorsque ce nombre est configuré sur 1, le panneau envoie un rapport d'erreur de communication si la réponse au paquet de la première requête est incorrecte.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Élément	Description
Bloc lecture max	Longueur maximale en octets d'une requête de bloc de données. Cette valeur s'applique uniquement à l'accès en lecture aux Registres d'exploitation.
Bloc de bits lecture max	Longueur maximale en bits d'une requête de bloc. Cette valeur s'applique uniquement à l'accès en lecture aux bits d'entrée et aux bobines de sortie.
Écriture du registre d'exploitation	Fonction Modbus pour les opérations d'écriture dans les registres de maintien. Veuillez choisir entre la fonction **06** (Prédéfinir registre seul) et la fonction **16** (Prédéfinir registres multiples). Si la fonction **06** est sélectionnée, le protocole utilisera toujours la fonction **06** pour l'écriture dans le contrôleur, même en écrivant dans plusieurs registres consécutifs. Si la fonction **16** est sélectionnée, le protocole utilisera toujours la fonction **16** pour l'écriture dans le contrôleur, même pour une requête d'écriture dans un registre seul, et le paramètre **Taille max bloc lecture** de la requête est configurée sur **2**. L'utilisation de la fonction **16** peut augmenter la performance de la communication.
Écrire bobines	Fonction Modbus pour les opérations d'écriture dans les bobines de sortie. Veuillez choisir entre la fonction **05** (Écrire bobine seule) et la fonction **15** (Écrire bobines multiples). Si la fonction Modbus **05** est sélectionnée, le protocole utilisera toujours la fonction **05** pour l'écriture dans le contrôleur, même en écrivant dans plusieurs bobines consécutives. Si la fonction Modbus **15** est sélectionnée, le protocole utilisera toujours la fonction **15** pour l'écriture dans le contrôleur, même pour une requête d'écriture dans une bobine seule. L'utilisation de la fonction **16** peut augmenter la performance de la communication.
Mode de transmission	• **RTU** : utilise le mode RTU • **ASCII** : utilise le mode ASCII Remarque : lorsque le réseau API est actif, tous les nœuds sont configurés sur le même mode de transmission.
Modèles d'API	Permet de choisir entre les différentes modèles d'API : • **Modicon Modbus (1-based)** : Implantation Modbus où toutes les ressources commencent avec décalage 1. • **Modicon Modbus (0-based)** : Implantation Modbus où toutes les ressources commencent avec décalage 0. • **Enron Modbus (1-based)** : étend l'implantation Modicon Modbus avec la zone de mémoire des registres 32 bits. • **Enron Modbus (0-based)** : étend l'implantation Generic Modbus avec la zone de mémoire des registres 32 bits. Remarque : la plage d'adresses utilisée dans les trames Modbus s'étend toujours de 0 à 65535 pour les registres d'exploitation et de 0 à 65535 pour les bobines.
Comm...	En cliquant dessus, vous affichez la boîte de dialogue de configuration des paramètres de communication.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

<table>
<tr><th>Élément</th><th>Description</th></tr>
<tr><td></td><td>


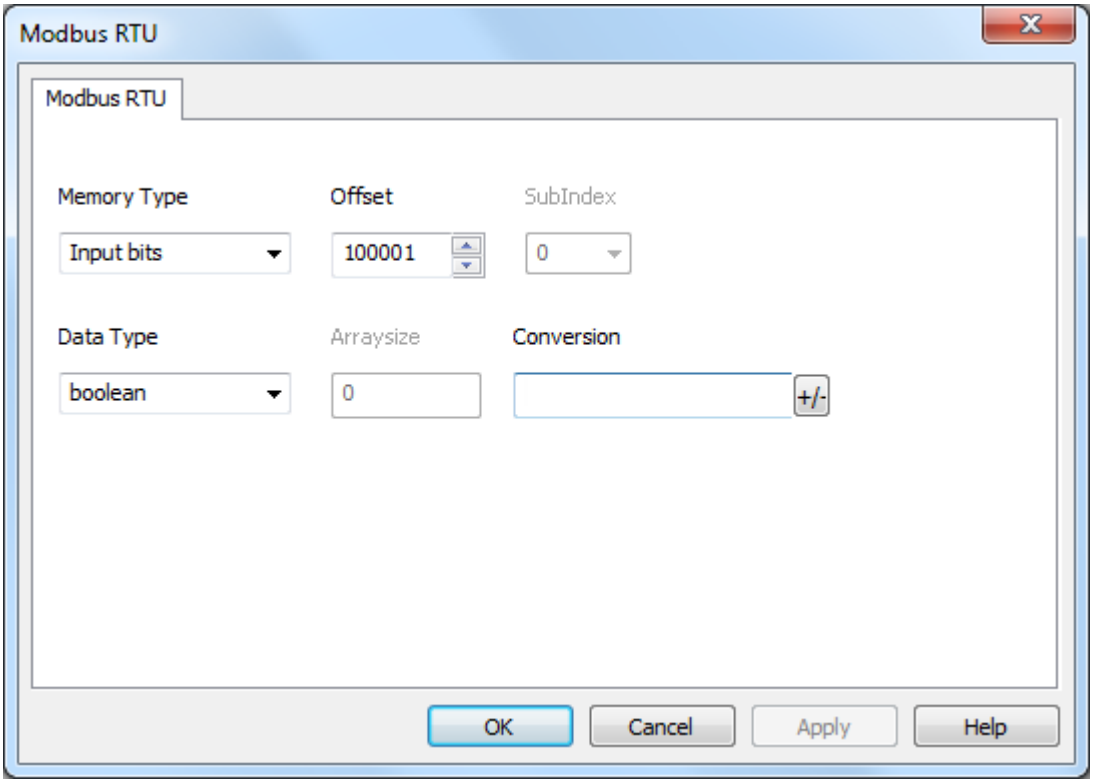
<table>
<tr><th>Élément</th><th>Paramètre</th></tr>
<tr><td>Port</td><td>Sélection du port série.
COM1 : port embarqué
COM2 : module exécutable en option branché sur fente n° 1 ou fente n° 2
COM3 : module exécutable en option branché sur fente n° 3 ou fente n° 4</td></tr>
<tr><td>Débit en bauds, parité, bits de données, bits d’arrêt</td><td>Paramètres de ligne série.</td></tr>
<tr><td>Mode</td><td>Mode de port série. modes disponibles :
RS-232.
RS-485 (2 fils)
RS-422 (4 fils)</td></tr>
</table>
</td></tr>
<tr><td>Réseau d’API</td><td>Il est possible de connecter plusieurs contrôleurs à un périphérique IHM. Pour configurer plusieurs connexions, sélectionnez Réseau d'API et cliquez sur Ajouter pour configurer chaque esclave</td></tr>
</table>

Paramètres de l’éditeur de variables

*Chemin : **VueProjet**> **Configuration** > double-cliquez sur **Variables***

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

1. Pour ajouter une variable, cliquez sur **+** : une nouvelle ligne est ajoutée.
2. Sélectionnez **Modbus RTU** dans la liste des protocoles : le dialogue de définition de variables s'affiche.



Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

<table>
<tr><th>Élément</th><th>Description</th></tr>
<tr><td>Type de mémoire</td><td>Ressource de Modbus où se situe la variable.
<table>
<tr><th>Type de mémoire</th><th>Description</th></tr>
<tr><td>Statut de la bobine</td><td>Bobines</td></tr>
<tr><td>Statut d’entrée</td><td>Entrée discrète</td></tr>
<tr><td>Registres d'entrée</td><td>Registres d'entrée</td></tr>
<tr><td>Registres de maintien</td><td>Registres de maintien</td></tr>
<tr><td>Registres à 32 bits</td><td>Zone de mémoire des registres à 32 bits.
Disponible uniquement pour les modèles d’API Enron Modbus.</td></tr>
<tr><td>ID d'écrasement de nœud</td><td rowspan="8">paramètre de protocole (voir Types de données spéciaux pour plus d'informations concernant ce mode)</td></tr>
<tr><td>Mode Modicon</td></tr>
<tr><td>Débit en bauds série</td></tr>
<tr><td>Parité série</td></tr>
<tr><td>Bits d’arrêt série</td></tr>
<tr><td>Mode série</td></tr>
<tr><td>Série terminée</td></tr>
</table></td></tr>
<tr><td>Offset</td><td>Adresse de décalage où se situe la variable.
Les adresses de décalage sont six chiffres composés par un préfixe de type de données à un chiffre + une adresse de ressource à cinq chiffres.
<table>
<tr><th>Type de mémoire</th><th>Studio
Plage de décalage</th><th>Modicon
Plage de décalage</th><th>Generic Modbus
Plage de décalage</th></tr>
<tr><td>Statut de la bobine</td><td>0 – 65535</td><td rowspan="5">1 – 65536</td><td rowspan="5">0 – 65535</td></tr>
<tr><td>Statut d’entrée</td><td>100000 – 165535</td></tr>
<tr><td>Registres d'entrée</td><td>300000 – 365535</td></tr>
<tr><td>Registres de maintien</td><td>400000 – 465535</td></tr>
<tr><td>Registres à 32 bits</td><td>0 – 65535</td></tr>
</table></td></tr>
<tr><td>Sousindex</td><td>Permet la sélection du décalage de ressource dans le registre.</td></tr>
</table>

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

<table>
<tr><th>Élément</th><th>Description</th></tr>
<tr><td>Type de données</td><td>
<table>
<tr><th>Type de données</th><th>Espace mémoire</th><th>Limites</th></tr>
<tr><td>booléen</td><td>Donnée à 1 bit</td><td>0 ... 1</td></tr>
<tr><td>octet</td><td>Donnée à 8 bits</td><td>-128 ... 127</td></tr>
<tr><td>court</td><td>Donnée à 16 bits</td><td>-32768 ... 32767</td></tr>
<tr><td>entier</td><td>Donnée à 32 bits</td><td>-2.1e9 ... 2.1e9</td></tr>
<tr><td>int64</td><td>Donnée à 64 bits</td><td>-9.2e18 ... 9.2e18</td></tr>
<tr><td>octet non signé</td><td>Donnée à 8 bits</td><td>0 ... 255</td></tr>
<tr><td>court non signé</td><td>Donnée à 16 bits</td><td>0 ... 65535</td></tr>
<tr><td>entier non signé</td><td>Donnée à 32 bits</td><td>0 ... 4.2e9</td></tr>
<tr><td>uint64</td><td>Donnée à 64 bits</td><td>0 ... 1.8e19</td></tr>
<tr><td>flottant</td><td>Type virgule flottante de 32 bits simple précision conforme IEEE</td><td>1.17e-38 ... 3.4e38</td></tr>
<tr><td>double</td><td>Type virgule flottante de 64 bits double précision conforme IEEE</td><td>2.2e-308 ... 1.79e308</td></tr>
<tr><td>chaîne</td><td colspan="2">Tableau des éléments contenant du code de caractères défini par l'encodage sélectionné</td></tr>
<tr><td>binaire</td><td colspan="2">Données binaires arbitraires</td></tr>
</table>

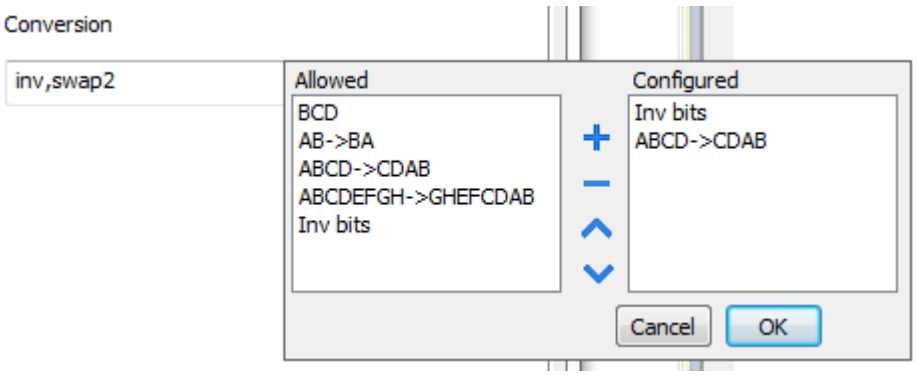
Remarque : pour définir des tableaux, sélectionnez l’un des formats de type de données suivi de crochets tels que « byte[] », « short[] », etc.
</td></tr>
<tr><td>Taille de tableau</td><td>

En cas de variable vectorielle, cette propriété représente le nombre d’éléments vectoriels.
En cas de variable de chaîne, cette propriété représente le nombre maximum d’octets disponible pour la variable de chaîne.

Remarque : le nombre d'octets correspond au nombre de caractères de chaîne si la propriété Encoding est définie sur UTF-8 ou Latin1 dans l'Éditeur de variables.
Si la propriété Encoding est définie sur UCS-2BE, UCS-2LE, UTF-16BE ou UTF-16LE, un caractère nécessite 2 octets.
</td></tr>
<tr><td>Conversion</td><td>Conversion à appliquer à la variable.</td></tr>
</table>

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Élément	Description



En fonction du type de données choisi, la liste **Autorisée** affiche un ou plusieurs types de conversion.

Valeur	Description
Inv bits	**inv** : inverse tous les bits de la variable. *Exemple :* 1001 → 0110 (format binaire) 9 → 6 (format décimal)
Inverser	**neg** : inverse la valeur de la variable. *Exemple :* 25.36 → -25.36
AB -> BA	**swapnibbles** : échange les quartets en un octet. *Exemple :* 15D4 → 514D (format hexadécimal) 5588 → 20813 (format décimal)
ABCD -> CDAB	**swap2** : échange les octets en un mot. *Exemple :* 9ACC → CC9A (format hexadécimal) 39628 → 52378 (format décimal)
ABCDEFGH -> GHEFCDAB	**swap4** : échange les octets en un mot double. *Exemple :* 32FCFF54 → 54FFFC32 (format hexadécimal) 855441236 → 1426062386 (format décimal)
ABC...NOP -> OPM...DAB	**swap8** : échange les octets en un mot long. Exemple : 142,366 → -893553517,588905 (format décimal) 0 10000000110

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Élément	Description
	<table><tr><th>Valeur</th><th>Description</th></tr><tr><td></td><td>0001110010111011011001000101101000011100101011000001 → 1 10000011100 10101010000101000101101101101100101101100000100111101 (format binaire)</td></tr><tr><td>**BCD**</td><td>**bcd** : sépare l'octet en deux quartets, les lit au format décimal (de 0 à 9) *Exemple :* 23 → 17 (format décimal) 0001 0111 = 23 0001 = 1 (premier quartet) 0111 = 7 (second quartet)</td></tr></table> Sélectionnez la conversion et cliquez sur le bouton +. La conversion sélectionnée s'affiche dans la liste **Configurée**. Si vous configurez d'autres conversions, celles-ci s'affichent dans l'ordre (du haut vers le bas de la liste **Configurée**). Utilisez les boutons de direction pour changer l'ordre des conversions configurées.

ID d'écrasement de nœud

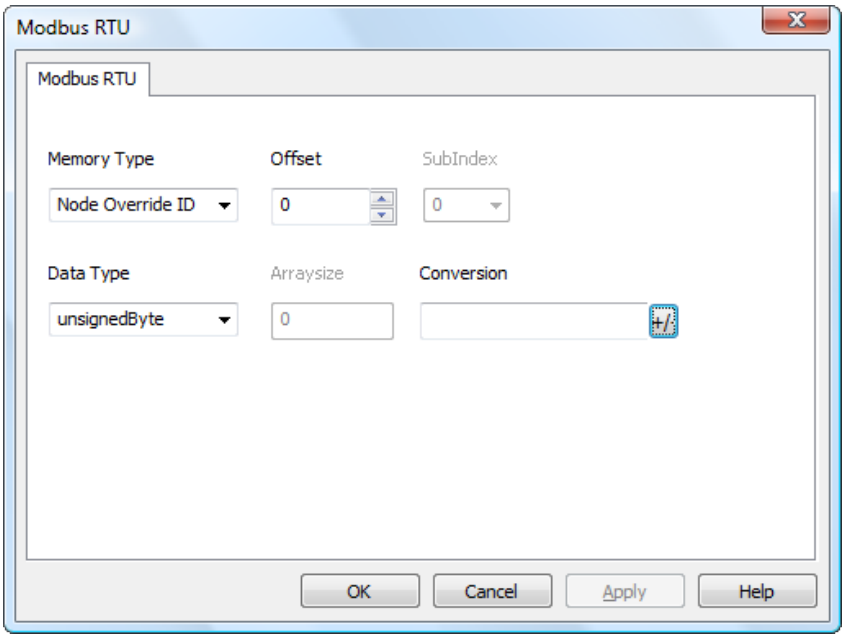
Le protocole propose le type de données spécial ID d'écrasement de nœud, qui vous permet de modifier l'ID de nœud de l'esclave au runtime. Ce type de mémoire est un octet non signé.

L'ID d'écrasement de nœud est initialisé avec la valeur de l'ID de nœud spécifiée dans le projet au moment de la programmation.

ID d'écrasement de nœud	Opération Modbus
0	La communication avec le contrôleur est coupée. En cas d'opération d'écriture, la requête sera transmise sans attente de réponse.
1 à 254	Interprété comme étant la valeur du nouvel ID de nœud et remplacé pour l'opération de runtime.
255	La communication avec le contrôleur est coupée, aucun message de requête ne sera généré.

Remarque : le maintien de la valeur de l'ID d'écrasement de nœud assignée au runtime est assuré par des cycles d'alimentation.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



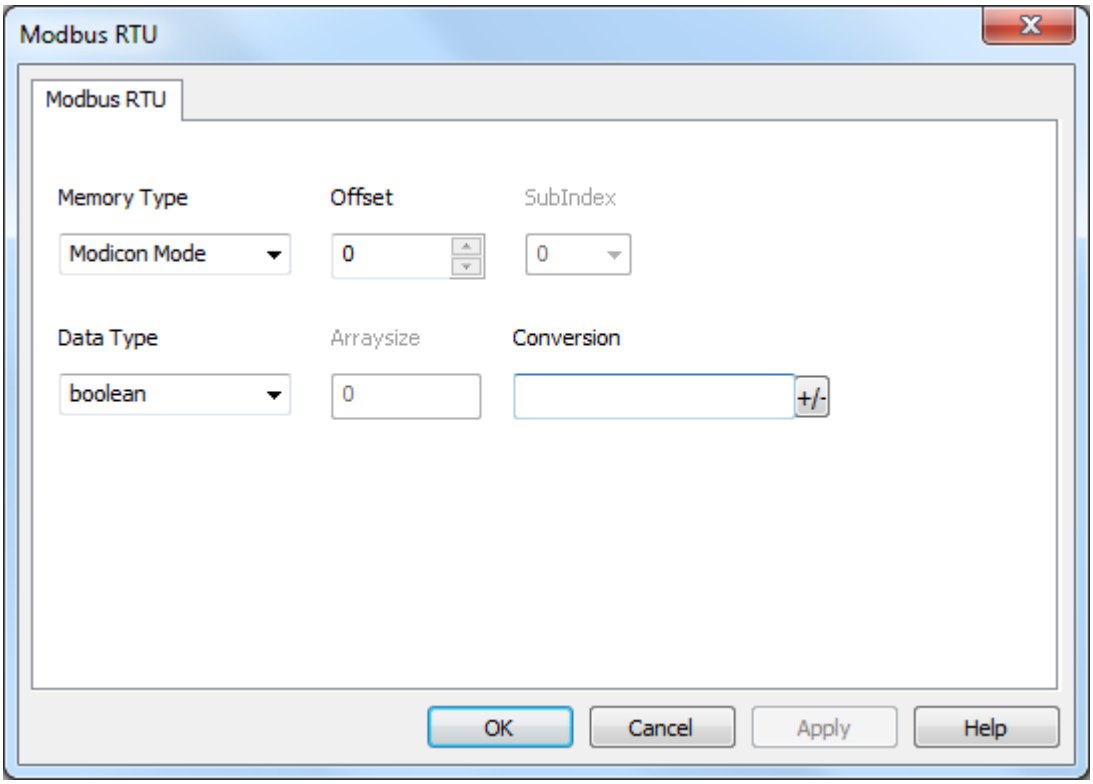
Mode Modicon

Ce protocole fournit un type de données spécial qui permet d'écraser les paramètres du mode Modicon au runtime.

Mode Modicon	Description
0	Generic Modbus (0-based). Les index de registre commencent à 0.
1	Modicon Modbus (1-based). Les index de registre commencent à 1.

Remarque : la valeur du paramètres de mode Modicon attribuée au runtime est conservée à travers les cycles d'alimentation.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



Écrasement des paramètres série

Ce protocole fournit des types de données spéciaux qui permettent d'écraser les paramètres série au runtime.

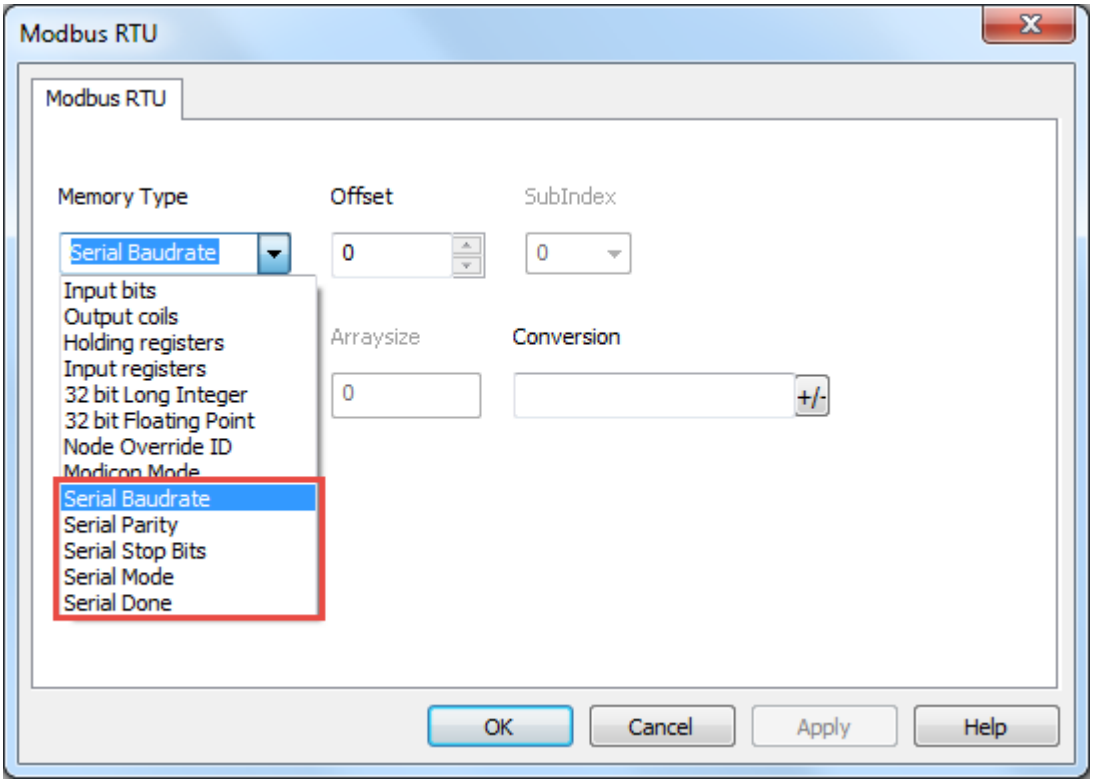
Paramètre	Description
Débit en bauds série	valeur 32 bits sans signature pour l'écrasement des débits en bauds. Les valeurs possibles sont 150, 300, 600, 1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200.
Parité série	valeur 8 bits sans signature pour l'écrasement des parités. Veuillez trouver les valeurs possibles dans la liste ci-dessous.

Valeur	Description
0	aucune parité
1	parité paire
2	parité impaire

Bits d'arrêt série	valeur 8 bits sans signature pour l'écrasement de bits d'arrêt. Les valeurs possibles sont 1 et 2.
Mode série	valeur 8 bits sans signature pour l'écrasement des modes série . Veuillez trouver les valeurs possibles dans la liste ci-dessous.

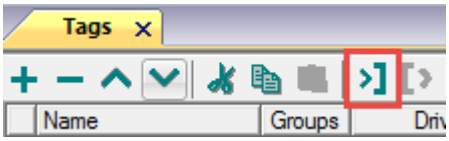
Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

<table>
<tr><th>Paramètre</th><th>Description</th></tr>
<tr><td></td><td>
<table>
<tr><th>Valeur</th><th>Description</th></tr>
<tr><td>0</td><td>Mode RS-232</td></tr>
<tr><td>1</td><td>Mode RS-485</td></tr>
<tr><td>2</td><td>Mode RS-422</td></tr>
</table>
</td></tr>
<tr><td>Série terminée</td><td>Configurez sur 1 pour écraser les paramètres de ligne de communication. Les paramètres sont traités ensemble uniquement lorsque vous configurez cette variable sur la valeur 1.</td></tr>
</table>



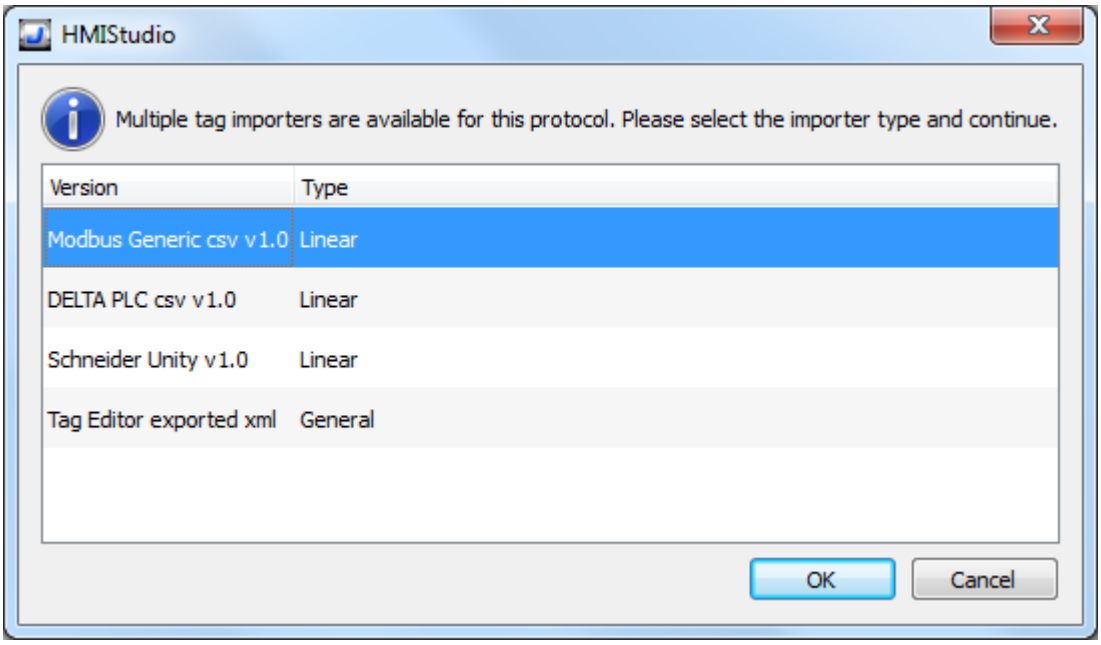
Importation de variables

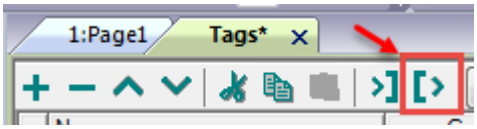
Sélectionnez le pilote dans l'éditeur de variables et cliquez sur le bouton **Importer variables** pour lancer le pilote d'importation.



La boîte de dialogue suivante affiche le type d'importation que vous pouvez sélectionner.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

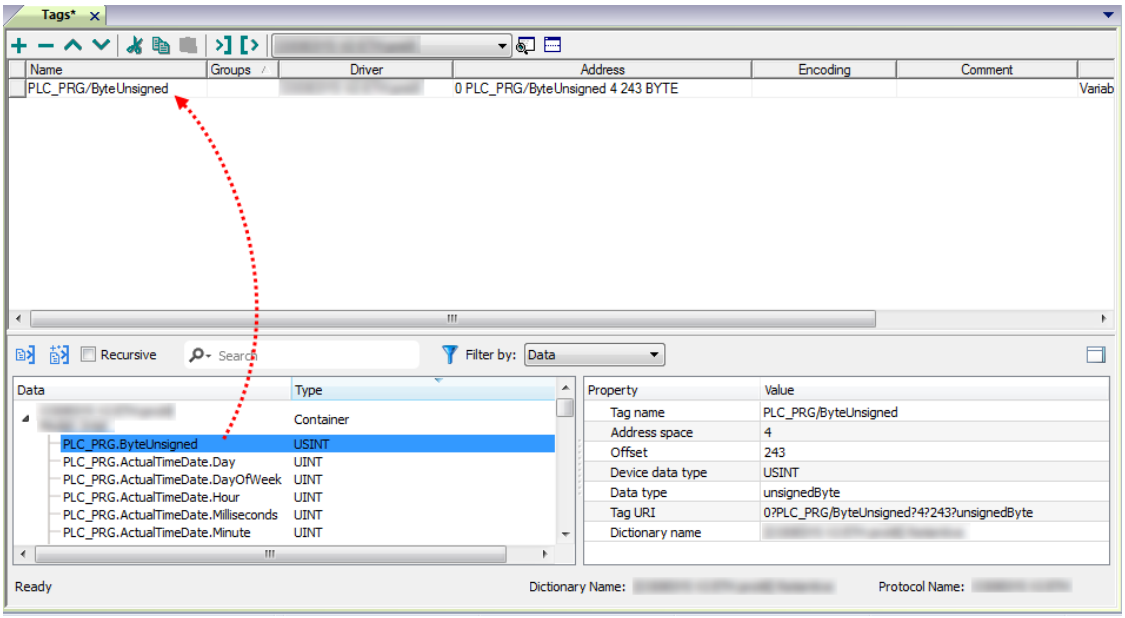


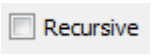
Type	Description
Modbus générique csv v1.0 **Linéaire**	Nécessite un fichier **.csv**. Toutes les variables s'affichent au même niveau.
DELTA PLC csv v1.0	Nécessite un fichier **.csv**. Toutes les variables s'affichent au même niveau.
Schneider Unity v1.0 **Linear**	Requiert un fichier **.uny**. Le fichier contenant des symboles doit être exporté au format **.txt** et plus tard rebaptisé **.uny**. Le pilote d'importation considère uniquement les variables situées à l'adresse fixe et ne tient pas compte des tableaux de chaînes. Tous les autres tableaux, sauf pour le type booléen, sont développés.
Fichier .xml exporté de l'éditeur de variables	Sélectionnez ce pilote d'importation pour lire un fichier XML générique exporté de l'éditeur de variables avec le bouton approprié. 

Lorsque vous avez sélectionné le pilote d'importation, localisez le fichier de symboles et cliquez sur **Ouvrir**.

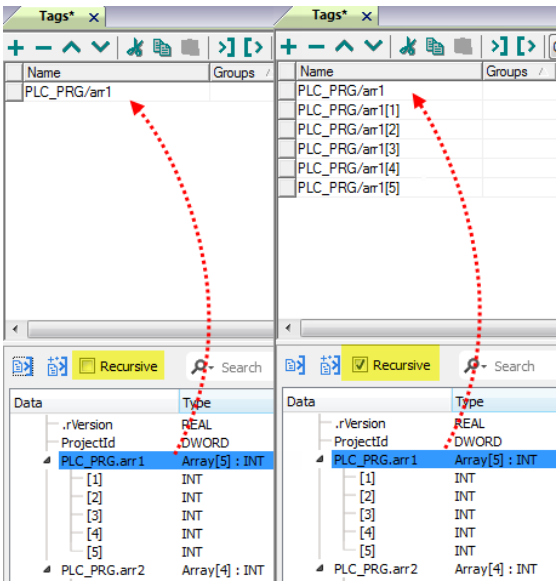
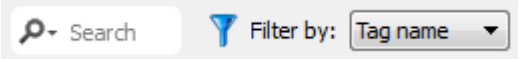
Les variables comprises dans le fichier de symboles sont listées dans le dictionnaire des variables. Le dictionnaire des variables est affiché en bas de l'écran.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



Élément de la barre d'outils	Description
	Importer variable(s). Sélectionnez les variables à importer et cliquez sur cette icône pour ajouter les variables du dictionnaire au projet.
	Mettre à jour variable(s). Cliquez sur cette icône pour mettre à jour les variables du projet grâce à un nouvel import du dictionnaire.
	Cochez cette case pour importer tous les sous-éléments d'une variable. Exemples de résultats avec la case cochée et non cochée :

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Élément de la barre d'outils	Description
	
	Recherche des variables dans le dictionnaire en filtrant en fonction de l'élément sélectionné dans la liste déroulante.

Structure de fichier csv Modbus Generic

Ce protocole prend en charge l'importation des informations de variable lorsque vous les lui fournissez au format **.csv**, conformément au format suivant :

```
NodeID, TagName, MemoryType, Address, DataFormat,...,[Comment]
```

Remarque : les champs entre crochets ainsi que les champs entre DataFormat et Comment sont facultatifs.

Champ	Description
NodeID	Nœud auquel appartient la variable
TagName	Description de la variable
Type de mémoire	• OUTP • INP • IREG • HREG
Adresse	Offset compatible avec la notation Modbus
FormatDeDonnées	Type de données dans la notation interne. Consultez la section « Concepts de programmation » du manuel principal.
Commentaire	Description supplémentaire facultative.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Exemple de fichier de variables

Exemple de ligne .csv :

```
2, Holding Register 1, HREG, 400001, unsignedShort,
```

Remarque : cette ligne est sans commentaire. Lorsque ce champ est manquant, le caractère de fin de ligne doit être une virgule.

Statut de communication

Vous pouvez afficher le statut de communication actuel grâce aux variables système. Voir la section « Variables système » du manuel principal.

Codes pris en charge pour ce pilote de communication :

Erreur	Cause	Action
Pas de réponse	Pas de retour dans le délai d'attente configuré.	Vérifiez que le contrôleur est connecté et correctement configuré pour accéder au réseau.
La réponse contient une adresse de nœud incorrecte	Le contrôleur a envoyé une réponse contenant une adresse de nœud non valide du périphérique.	-
Le message reçu est trop court	Le contrôleur a envoyé une réponse dans un format non valide du périphérique.	-
Écriture incorrecte confirmation des données	Le contrôleur a refusé une demande d'écriture.	Vérifiez la cohérence entre les données de projet et les ressources du contrôleur.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Modbus RTU Server

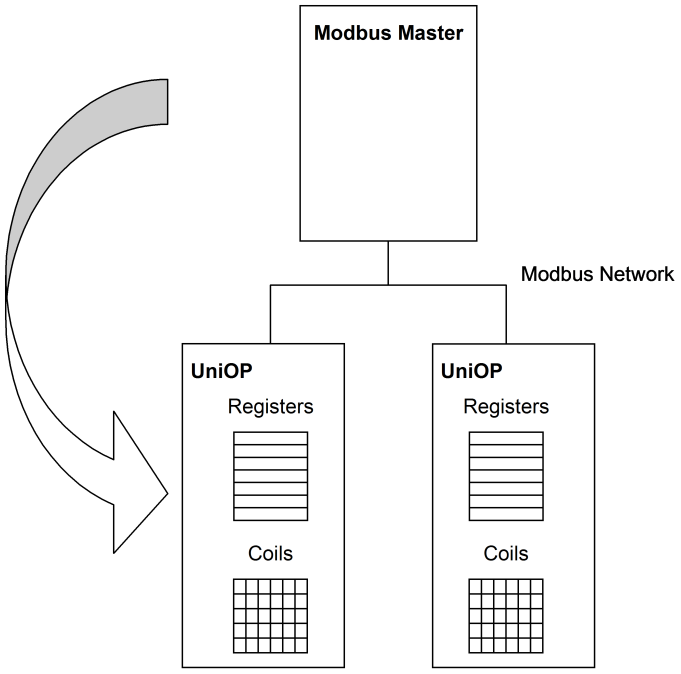
Le pilote de communication Modbus RTU Server permet de connecter le périphérique IHM en tant qu'esclave à un réseau Modbus RTU. L'échange d'informations utilise les messages de Modbus standard.

Cette approche permet de connecter des périphériques IHM à des systèmes SCADA par le protocole de communication Modbus RTU pris en charge de manière universelle.

Principe de fonctionnement

Ce pilote de communication implémente une unité esclave Modbus RTU dans le périphérique IHM. Un sous-ensemble de la plage complète des codes de fonction Modbus est pris en charge. Les codes de fonction disponibles permettent le transfert de données entre le maître et l'esclave.

Le diagramme suivant représente l'architecture de système.



Le périphérique IHM simule en fait l'interface de communication d'un API : les formats de données des bobines et des registres sont respectivement booléens et entiers à 16 bits.

Le périphérique peut à tout moment accéder aux données de sa mémoire interne. Le transfert de données de et vers le Maître Modbus est possible uniquement sur l'initiative du Maître lui-même.

Détails sur l'implantation

L'implantation du Modbus RTU esclave prend seulement en charge un sous-ensemble des code fonctions Modbus standard.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Code	Fonction	Description
01	Lecture du statut de la bobine	Lecture de bits multiples dans la zone du périphérique bobine.
03	Lecture des registres de maintien	Lecture de registres multiples du périphérique.
05	Forcer bobine seule	Force une bobine unique du périphérique à passer au statut « MARCHE » ou « ARRÊT ».
06	Prédéfinir registre seul	Prédéfinit une valeur dans un registre du périphérique.
08	Essai de diagnostic en boucle	Seule la sous-fonction 00 (Retour des données de requête) est prise en charge.
15	Forcer plusieurs bobines	Force plusieurs bobines du périphérique à passer au statut « MARCHE » ou « ARRÊT ».
16		Prédéfinit une valeur dans plusieurs registres du périphérique.
17	Rapport d'ID esclave	Retourne des informations de diagnostic du contrôleur sur l'adresse esclave.
23	Lecture/écriture de plusieurs registres	Lit et prédéfinit une valeur dans plusieurs registres du périphérique

Codes d'exception

Code	Description
01	**Fonction interdite**. le code de fonction de la requête n'est pas pris en charge
02	**Adresse de données interdite** . L'adresse des données reçue au cours de la requête dépasse la plage de données prédéfinie (voir **définition des variables** pour les détails des plages pour tous les types).
03	**Valeur de données interdite**. Une sous-fonction autre que 00 est spécifiée dans l'essai de diagnostic en boucle (Code 08),

Paramètres de l’éditeur de protocole

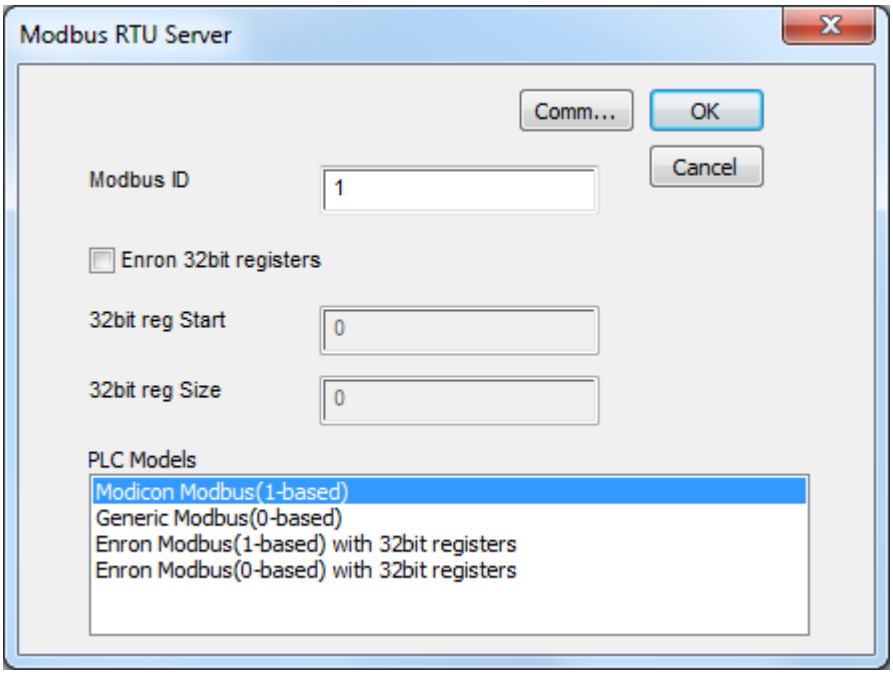
Ajouter un protocole

Pour configurer le protocole :

1. Dans le nœud **Config**, double-cliquez sur **Protocoles**.
2. Pour ajouter un pilote, cliquez sur **+** : une nouvelle ligne est ajoutée.
3. Sélectionnez le protocole dans la liste **API**.

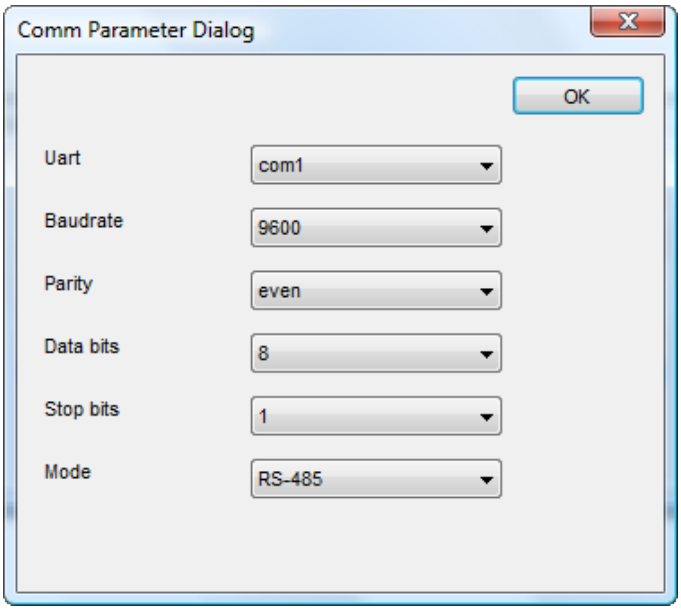
La boîte de dialogue de configuration du protocole s’affiche.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



Élément	Description
ID Modbus	ID de nœud Modbus. Chaque périphérique de serveur Modbus du réseau doit avoir son propre ID Modbus.
Registres Enron 32 bits	Si sélectionné, il permet de définir la première adresse de registre et le nombre de registres pour la zone de mémoire des registres à 32 bits. Remarque : les registres à 32 bit sont disponibles uniquement pour les modèles d'API **Enron Modbus**.
32bit reg Start **32bit reg Size**	Définition de la zone de mémoire des registres à 32 bits. La valeur **Start** représente la première adresse de registre. La valeur **Size** représente le nombre de registres. Remarque : la réponse à une requête de l'un des registres de cette zone sera 4 octets.
Modèles d'API	Permet de choisir entre les différentes modèles d'API : • **Modicon Modbus (1-based)** : Implantation Modbus où toutes les ressources commencent avec décalage 1. • **Modicon Modbus (0-based)** : Implantation Modbus où toutes les ressources commencent avec décalage 0. • **Enron Modbus (1-based)** : étend l'implantation Modicon Modbus avec la zone de mémoire des registres 32 bits.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Élément	Description
	• **Enron Modbus (0-based)** : étend l'implantation Generic Modbus avec la zone de mémoire des registres 32 bits. Remarque : la plage d'adresses utilisée dans les trames Modbus s'étend toujours de 0 à 65535 pour les registres d'exploitation et de 0 à 65535 pour les bobines.
Comm...	En cliquant dessus, vous affichez la boîte de dialogue de configuration des paramètres de communication. Veuillez configurer les paramètres selon les valeurs programmées dans le maître Modbus. 

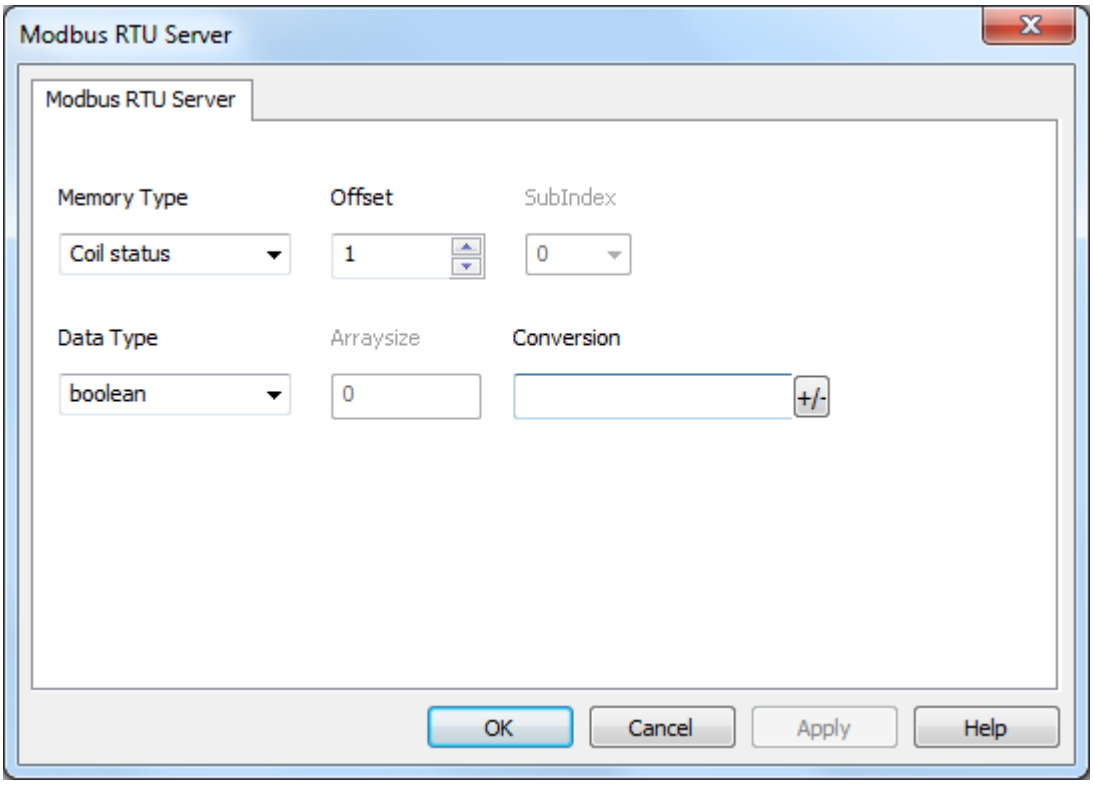
Élément	Description
UART	Sélection du port série. • **COM1** : port embarqué • **COM2** : module exécutable en option branché sur fente n° 1 ou fente n° 2 • **COM3** : module exécutable en option branché sur fente n° 3 ou fente n° 4
Débit en bauds, parité, bits de données, bits d'arrêt	Paramètres de ligne série.
Mode	Mode de port série. Options disponibles : • **RS-232** • **RS-485** (2 fils) • **RS-422** (4 fils)

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Paramètres de l’éditeur de variables

Chemin : ***VueProjet*** *>* ***Configuration*** *> double-cliquez sur* ***Variables***

1. Pour ajouter une variable, cliquez sur **+** : une nouvelle ligne est ajoutée.
2. Sélectionnez **Modbus RTU Server** dans la liste des protocoles : le dialogue de définition de variables s'affiche.

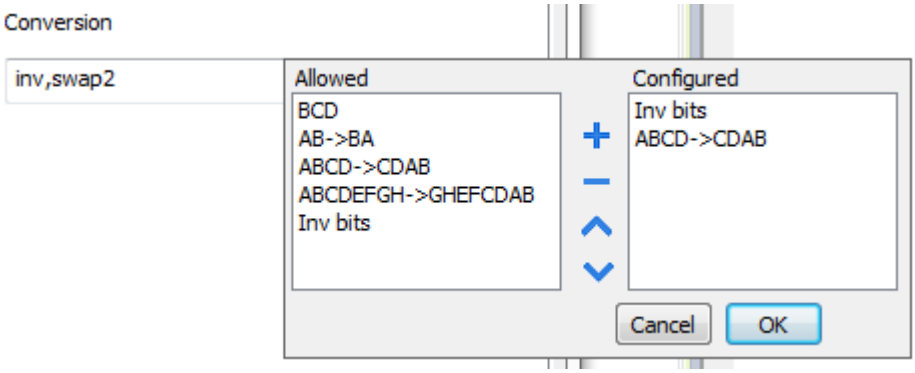


Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

<table>
<tr><th>Élément</th><th>Description</th></tr>
<tr><td>Type de mémoire</td><td>Ressource de Modbus où se situe la variable.
<table>
<tr><th>Type de mémoire</th><th>Ressource Modbus</th></tr>
<tr><td>Statut de la bobine</td><td>Bobines</td></tr>
<tr><td>Statut d’entrée</td><td>Entrée discrète</td></tr>
<tr><td>Registres d'entrée</td><td>Registres d'entrée</td></tr>
<tr><td>Registres de maintien</td><td>Registres de maintien</td></tr>
<tr><td>Registres à 32 bits</td><td>Zone de mémoire des registres à 32 bits.
Disponible uniquement pour les modèles d’API Enron Modbus.</td></tr>
<tr><td>ID d'écrasement de nœud</td><td rowspan="8">paramètre de protocole (voir Types de données spéciaux pour plus d'informations concernant ce mode)</td></tr>
<tr><td>Mode Modicon</td></tr>
<tr><td>Débit en bauds série</td></tr>
<tr><td>Parité série</td></tr>
<tr><td>Bits d’arrêt série</td></tr>
<tr><td>Mode série</td></tr>
<tr><td>Série terminée</td></tr>
</table></td></tr>
<tr><td>Offset</td><td>Adresse de décalage où se situe la variable.
Les adresses de décalage sont six chiffres composés par un préfixe de type de données à un chiffre + une adresse de ressource à cinq chiffres.
<table>
<tr><th>Type de mémoire</th><th>Studio
Plage de décalage</th><th>Modicon
Plage de décalage</th><th>Generic Modbus
Plage de décalage</th></tr>
<tr><td>Statut de la bobine</td><td>0 – 65535</td><td rowspan="5">1 – 65536</td><td rowspan="5">0 – 65535</td></tr>
<tr><td>Statut d’entrée</td><td>100000 – 165535</td></tr>
<tr><td>Registres d'entrée</td><td>300000 – 365535</td></tr>
<tr><td>Registres de maintien</td><td>400000 – 465535</td></tr>
<tr><td>Registres à 32 bits</td><td>0 – 65535</td></tr>
</table></td></tr>
<tr><td>Sousindex</td><td>Permet la sélection du décalage de ressource dans le registre.</td></tr>
</table>

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

<table>
<tr><th>Élément</th><th>Description</th></tr>
<tr><td>Type de données</td><td>
<table>
<tr><th>Type de données</th><th>Espace mémoire</th><th>Limites</th></tr>
<tr><td>booléen</td><td>Donnée à 1 bit</td><td>0 ... 1</td></tr>
<tr><td>octet</td><td>Donnée à 8 bits</td><td>-128 ... 127</td></tr>
<tr><td>court</td><td>Donnée à 16 bits</td><td>-32768 ... 32767</td></tr>
<tr><td>entier</td><td>Donnée à 32 bits</td><td>-2.1e9 ... 2.1e9</td></tr>
<tr><td>int64</td><td>Donnée à 64 bits</td><td>-9.2e18 ... 9.2e18</td></tr>
<tr><td>octet non signé</td><td>Donnée à 8 bits</td><td>0 ... 255</td></tr>
<tr><td>court non signé</td><td>Donnée à 16 bits</td><td>0 ... 65535</td></tr>
<tr><td>entier non signé</td><td>Donnée à 32 bits</td><td>0 ... 4.2e9</td></tr>
<tr><td>uint64</td><td>Donnée à 64 bits</td><td>0 ... 1.8e19</td></tr>
<tr><td>flottant</td><td>Type virgule flottante de 32 bits simple précision conforme IEEE</td><td>1.17e-38 ... 3.4e38</td></tr>
<tr><td>double</td><td>Type virgule flottante de 64 bits double précision conforme IEEE</td><td>2.2e-308 ... 1.79e308</td></tr>
<tr><td>chaîne</td><td colspan="2">Tableau des éléments contenant du code de caractères défini par l'encodage sélectionné</td></tr>
<tr><td>binaire</td><td colspan="2">Données binaires arbitraires</td></tr>
</table>

Remarque : pour définir des tableaux, sélectionnez l’un des formats de type de données suivi de crochets tels que « byte[] », « short[] », etc.
</td></tr>
<tr><td>Taille de tableau</td><td>Lors de la configuration du tableau ou des variables de chaîne, cette option définit la quantité d'éléments de réseau ou des caractères de la chaîne.</td></tr>
<tr><td>Conversion</td><td>Conversion à appliquer à la variable.
 

En fonction du type de données choisi, la liste Autorisée affiche un ou plusieurs types de</td></tr>
</table>

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Élément	Description

conversion.

Valeur	Description
Inv bits	**inv** : inverse tous les bits de la variable. *Exemple :* 1001 → 0110 (format binaire) 9 → 6 (format décimal)
Inverser	**neg** : inverse la valeur de la variable. *Exemple :* 25.36 → -25.36
AB -> BA	**swapnibbles** : échange les quartets en un octet. *Exemple :* 15D4 → 514D (format hexadécimal) 5588 → 20813 (format décimal)
ABCD -> CDAB	**swap2** : échange les octets en un mot. *Exemple :* 9ACC → CC9A (format hexadécimal) 39628 → 52378 (format décimal)
ABCDEFGH -> GHEFCDAB	**swap4** : échange les octets en un mot double. *Exemple :* 32FCFF54 → 54FFFC32 (format hexadécimal) 855441236 → 1426062386 (format décimal)
ABC...NOP -> OPM...DAB	**swap8** : échange les octets en un mot long. Exemple : 142,366 → -893553517,588905 (format décimal) 0 10000000110 00011100101110110110010001011010000111001010110000001 → 1 10000011100 10101010000101000101101101101100101101100000100111101 (format binaire)
BCD	**bcd** : sépare l'octet en deux quartets, les lit au format décimal (de 0 à 9) *Exemple :* 23 → 17 (format décimal) 0001 0111 = 23 0001 = 1 (premier quartet) 0111 = 7 (second quartet)

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Élément	Description
	Sélectionnez la conversion et cliquez sur le bouton +. La conversion sélectionnée s'affiche dans la liste **Configurée**. Si vous configurez d'autres conversions, celles-ci s'affichent dans l'ordre (du haut vers le bas de la liste **Configurée**). Utilisez les boutons de direction pour changer l'ordre des conversions configurées.

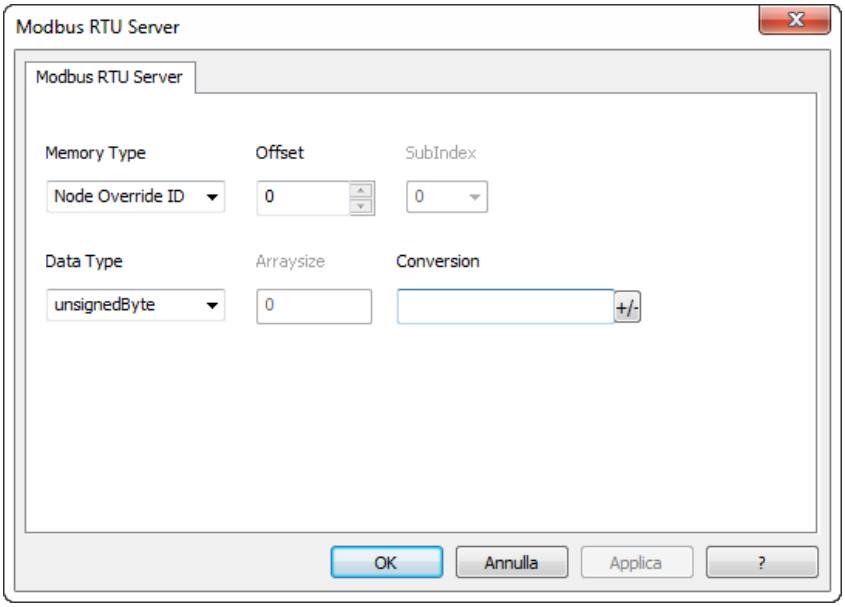
ID d'écrasement de nœud

Le protocole propose le type de données spécial ID d'écrasement de nœud, qui vous permet de modifier l'ID de nœud de l'esclave au runtime. Ce type de mémoire est un octet non signé.

L'ID d'écrasement de nœud est initialisé avec la valeur de l'ID de nœud spécifiée dans le projet au moment de la programmation.

ID d'écrasement de nœud	Opération Modbus
0	La communication avec l'esclave est coupée. En cas d'opération d'écriture, le périphérique ne répondra pas aux trames de requête.
1 à 255	Interprété comme étant la valeur du nouvel ID de nœud et remplacé pour l'opération de runtime.

Remarque : le maintien de la valeur de l'ID d'écrasement de nœud assignée au runtime est assuré par des cycles d'alimentation.



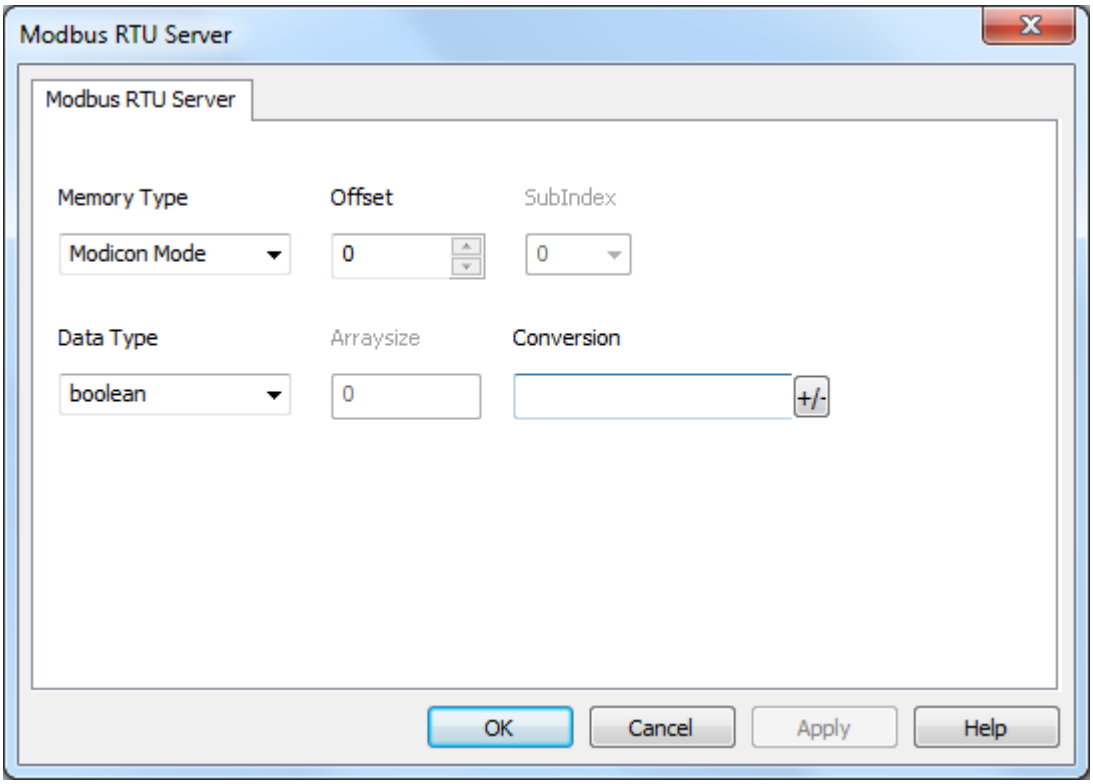
Mode Modicon

Ce protocole fournit un type de données spécial qui permet d'écraser les paramètres du mode Modicon au runtime.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Mode Modicon	Description
0	Generic Modbus (0-based). Les index de registre commencent à 0.
1	Generic Modbus (1-based). Les index de registre commencent à 1.

Remarque : la valeur du paramètres de mode Modicon attribuée au runtime est conservée à travers les cycles d'alimentation.



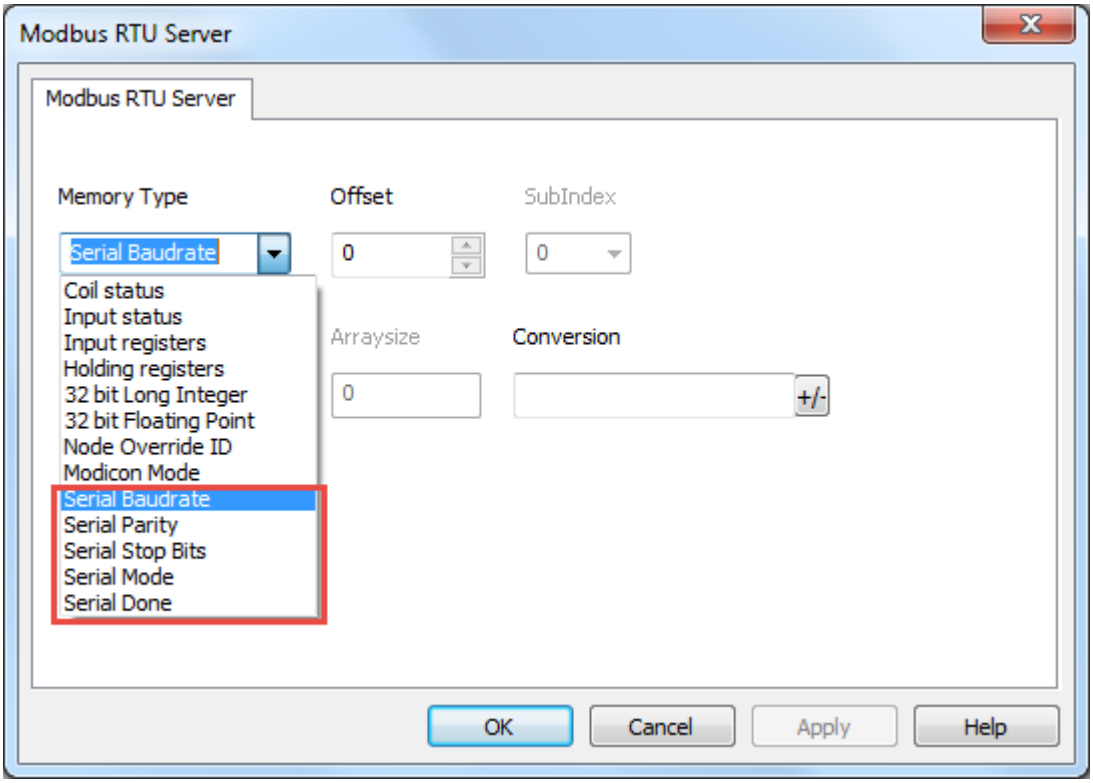
Écrasement des paramètres série

Ce protocole fournit des types de données spéciaux qui permettent d'écraser les paramètres série au runtime.

Paramètre	Description
Débit en bauds série	valeur 32 bits sans signature pour l'écrasement des débits en bauds. Les valeurs possibles sont 150, 300, 600, 1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200.
Parité série	valeur 8 bits sans signature pour l'écrasement des parités. Veuillez trouver les valeurs possibles dans la liste ci-dessous.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

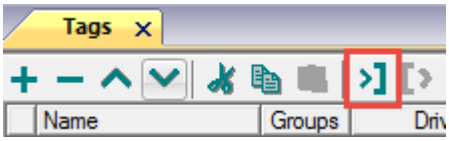
<table>
<tr><th>Paramètre</th><th>Description</th></tr>
<tr><td></td><td>
<table>
<tr><th>Valeur</th><th>Description</th></tr>
<tr><td>0</td><td>aucune parité</td></tr>
<tr><td>1</td><td>parité paire</td></tr>
<tr><td>2</td><td>parité impaire</td></tr>
</table>
</td></tr>
<tr><td>Bits d’arrêt série</td><td>valeur 8 bits sans signature pour l'écrasement de bits d'arrêt. Les valeurs possibles sont 1 et 2.</td></tr>
<tr><td>Mode série</td><td>valeur 8 bits sans signature pour l'écrasement des modes série . Veuillez trouver les valeurs possibles dans la liste ci-dessous.
<table>
<tr><th>Valeur</th><th>Description</th></tr>
<tr><td>0</td><td>Mode RS-232</td></tr>
<tr><td>1</td><td>Mode RS-485</td></tr>
<tr><td>2</td><td>Mode RS-422</td></tr>
</table>
</td></tr>
<tr><td>Série terminée</td><td>Configurez sur 1 pour écraser les paramètres de ligne de communication. Les paramètres sont traités ensemble uniquement lorsque vous configurez cette variable sur la valeur 1.</td></tr>
</table>



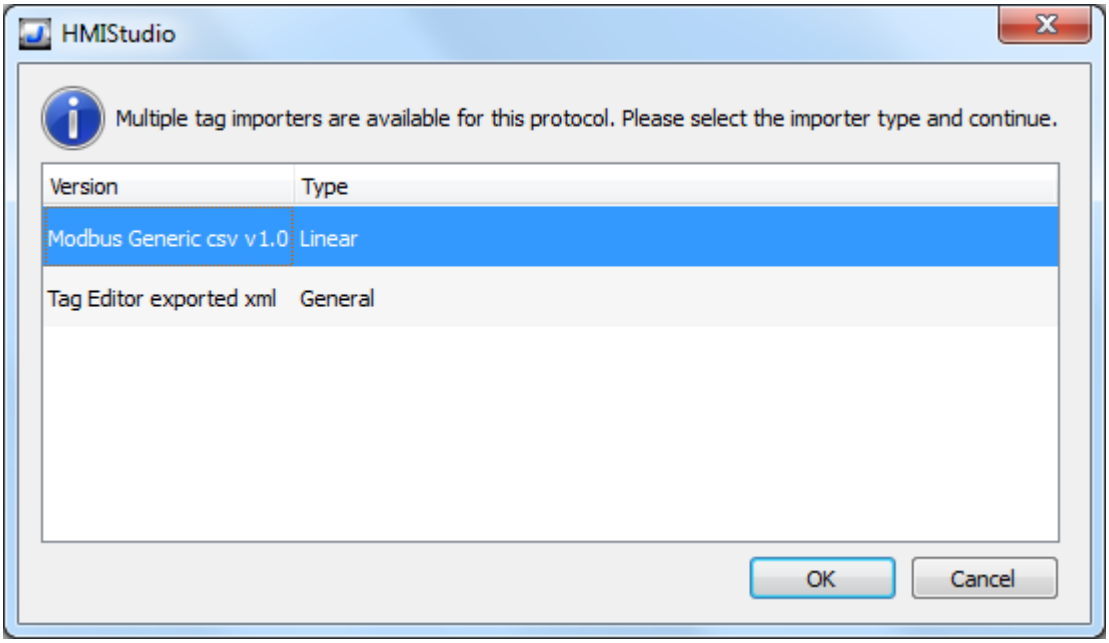
Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

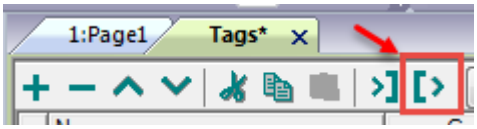
Importation de variables

Sélectionnez le pilote dans l'éditeur de variables et cliquez sur le bouton **Importer variables** pour lancer le pilote d'importation.



La boîte de dialogue suivante affiche le type d'importation que vous pouvez sélectionner.

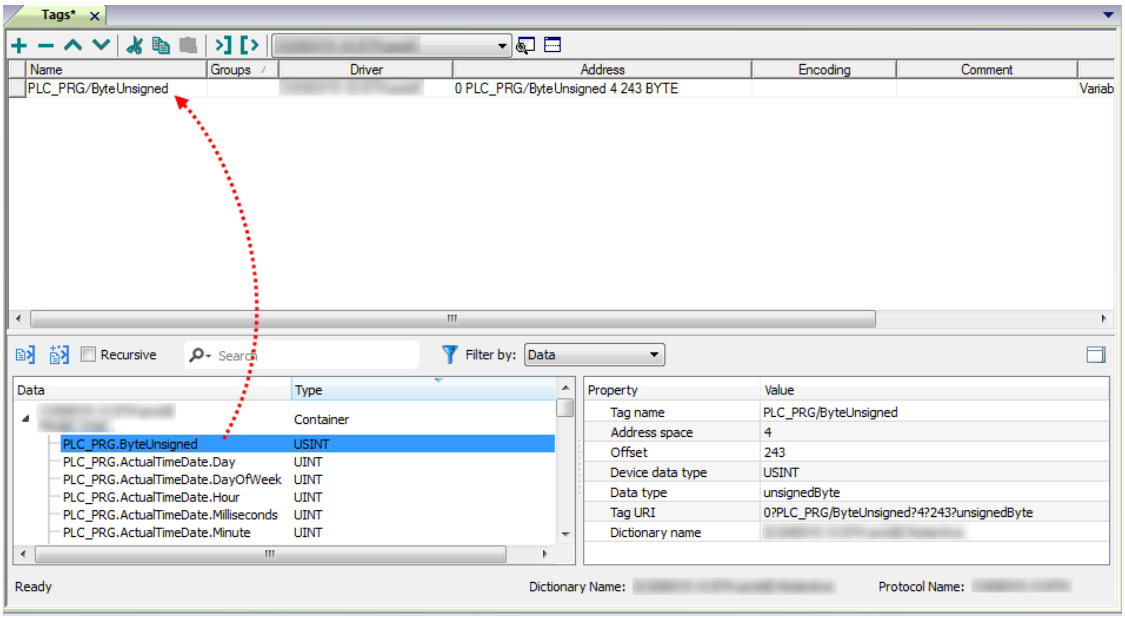


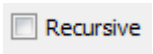
Type	Description
Modbus générique csv v1.0 Linéaire	Nécessite un fichier **.csv**. Toutes les variables s'affichent au même niveau.
Fichier .xml exporté de l'éditeur de variables	Sélectionnez ce pilote d'importation pour lire un fichier XML générique exporté de l'éditeur de variables avec le bouton approprié. 

Lorsque vous avez sélectionné le pilote d'importation, localisez le fichier de symboles et cliquez sur **Ouvrir**.

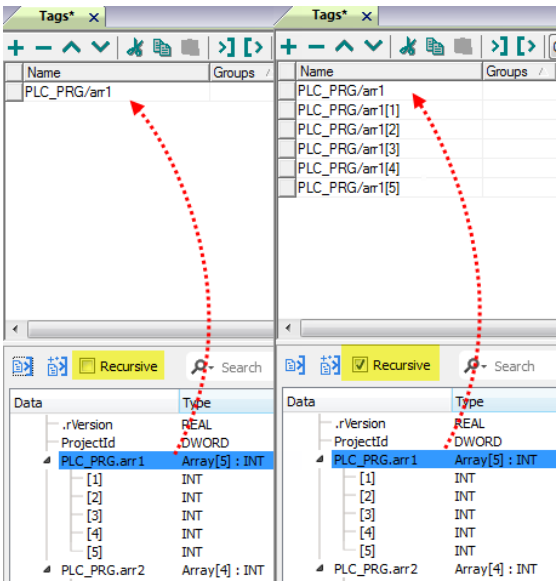
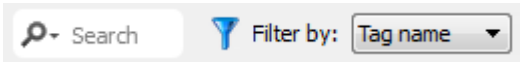
Les variables comprises dans le fichier de symboles sont listées dans le dictionnaire des variables. Le dictionnaire des variables est affiché en bas de l'écran.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



Élément de la barre d'outils	Description
	Importer variable(s). Sélectionnez les variables à importer et cliquez sur cette icône pour ajouter les variables du dictionnaire au projet.
	Mettre à jour variable(s). Cliquez sur cette icône pour mettre à jour les variables du projet grâce à un nouvel import du dictionnaire.
	Cochez cette case pour importer tous les sous-éléments d'une variable. Exemples de résultats avec la case cochée et non cochée :

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Élément de la barre d'outils	Description
	
	Recherche des variables dans le dictionnaire en filtrant en fonction de l'élément sélectionné dans la liste déroulante.

Structure de fichier csv Modbus Generic

Ce protocole prend en charge l'importation des informations de variable lorsque vous les lui fournissez au format **.csv**, conformément au format suivant :

```
NodeID, TagName, MemoryType, Address, DataFormat,...,[Comment]
```

Remarque : les champs entre crochets ainsi que les champs entre DataFormat et Comment sont facultatifs.

Champ	Description
NodeID	Nœud auquel appartient la variable
TagName	Description de la variable
Type de mémoire	• OUTP • INP • IREG • HREG
Adresse	Offset compatible avec la notation Modbus
FormatDeDonnées	Type de données dans la notation interne. Consultez la section « Concepts de programmation » du manuel principal.
Commentaire	Description supplémentaire facultative.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Exemple de fichier de variables

Exemple de ligne .csv :

```
2, Holding Register 1, HREG, 400001, unsignedShort,
```

Remarque : cette ligne est sans commentaire. Lorsque ce champ est manquant, le caractère de fin de ligne doit être une virgule.

Statut de communication

Le statut de communication actuel peut être affiché à l'aide de variables système. Ce protocole de communication se comporte comme un serveur et ne renvoie pas de message d'erreur de protocole spécifique.

Voir la section « Variables système » du manuel principal.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

TCP Modbus

Il est possible de connecter divers périphériques compatibles TCP Modbus à des périphériques IHM. Pour configurer votre périphérique TCP Modbus, consultez la documentation reçue avec ce périphérique.

La mise en œuvre du protocole s'effectue uniquement sous forme de client TCP Modbus.

Détails sur l'implantation

Cette mise en œuvre du TCP Modbus prend seulement en charge un sous-ensemble des codes fonction Modbus TCP standard.

Code	Fonction	Description
01	Lecture du statut de la bobine	Lecture de bits multiples dans la bobine de l'IHM.
02	Lecture du statut d'entrée	Lecture du statut « MARCHE/ARRÊT » des entrées TOR (référence 1x) de l'esclave.
03	Lecture des registres de maintien	Lecture de plusieurs registres.
04	Lecture des registres d'entrée	Lecture du contenu binaire des registres d'entrée (référence 3x) de l'esclave.
05	Forcer bobine seule	Force une bobine unique à passer au statut « MARCHE » ou « ARRÊT ».
06	Prédéfinir registre seul	Écriture d'une valeur dans un registre.
15	Écrire plusieurs bobines	Écriture du statut « MARCHE » ou « ARRÊT » pour chaque bobine d'une séquence de bobines.
16	Prédéfinir registres multiples	Écriture de valeurs dans un bloc de registres dans une séquence.

Paramètres de l'éditeur de protocole

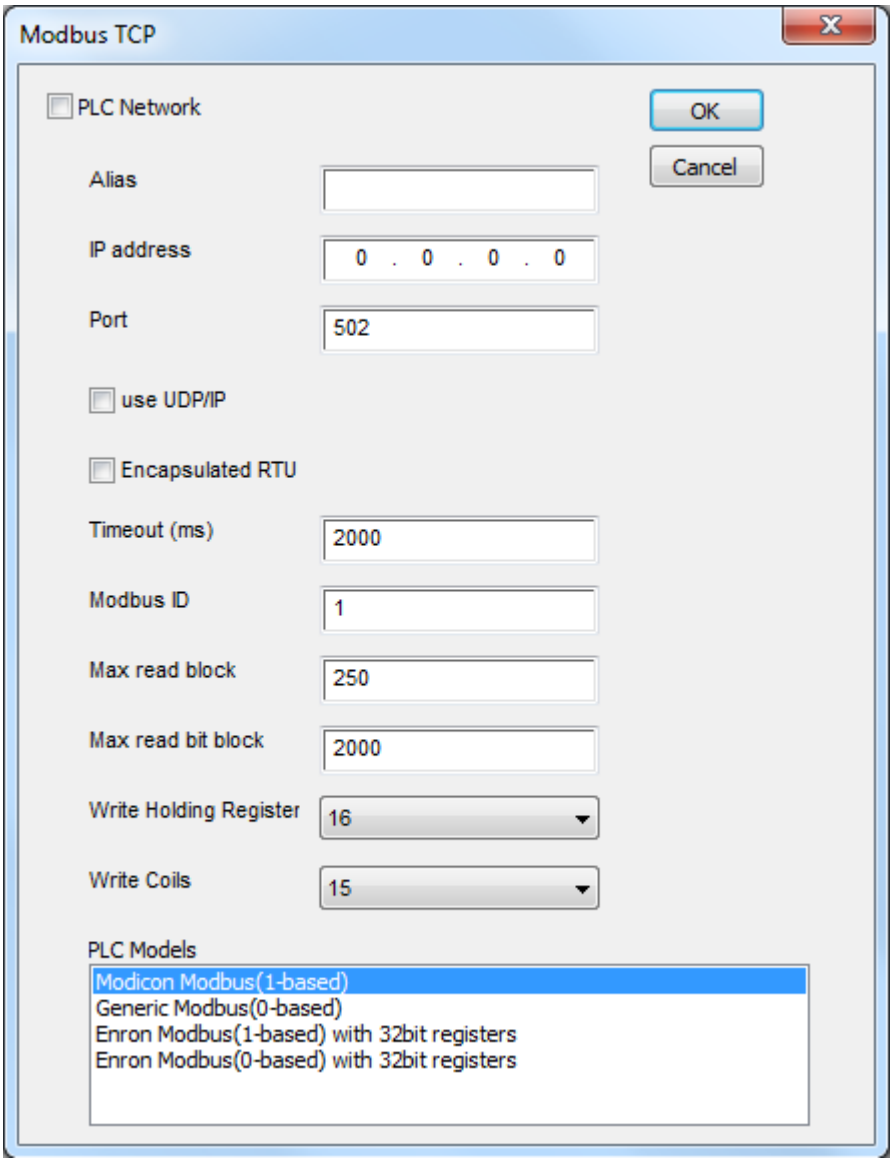
Ajouter un protocole

Pour configurer le protocole :

1. Dans le nœud **Config**, double-cliquez sur **Protocoles**.
2. Pour ajouter un pilote, cliquez sur **+** : une nouvelle ligne est ajoutée.
3. Sélectionnez le protocole dans la liste **API**.

La boîte de dialogue de configuration du protocole s'affiche.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

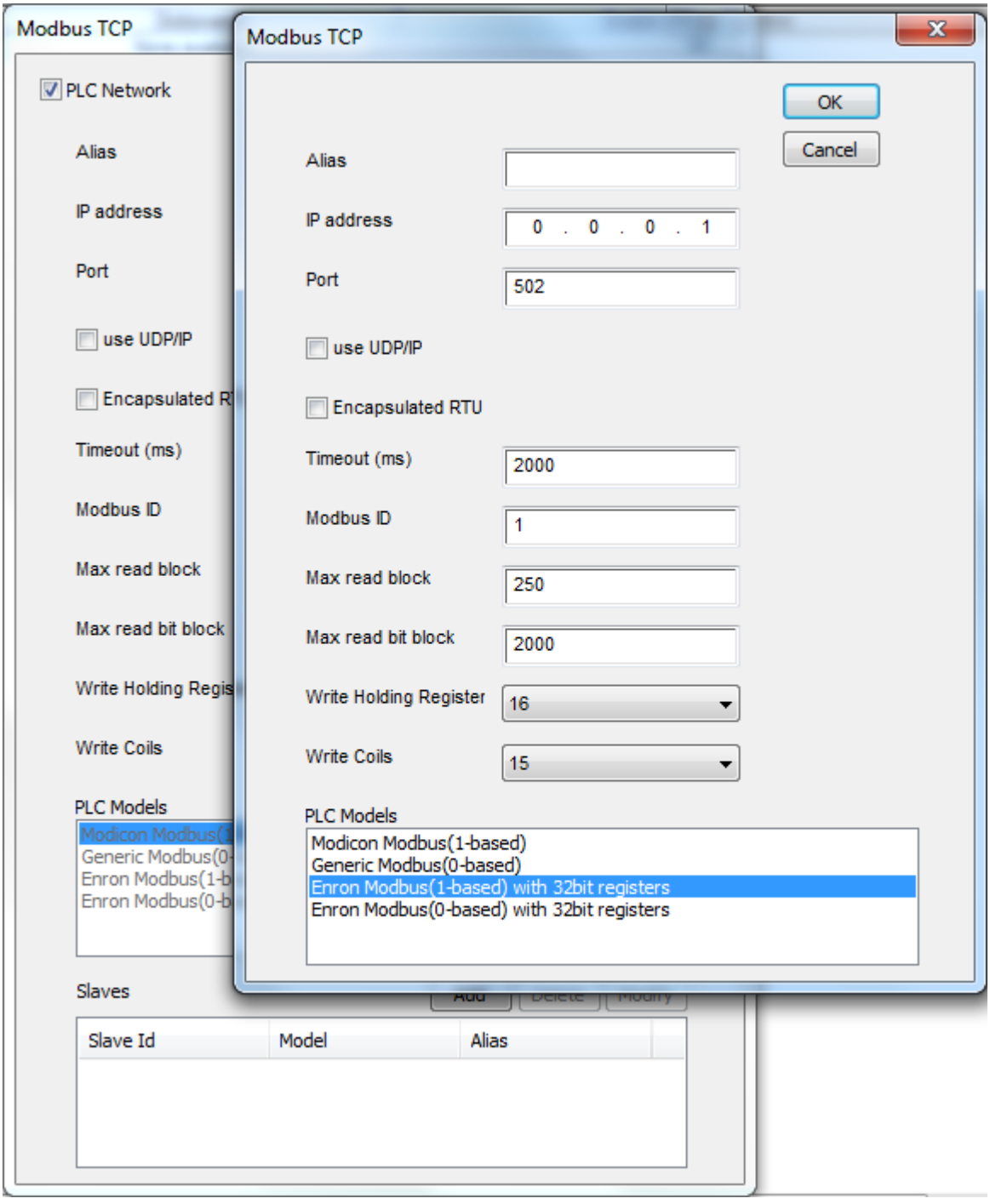


Élément	Description
Alias	Nom d'identification des nœuds dans les configurations réseau. Il est ajouté en préfixe à chaque nom de variable importé pour chaque nœud de réseau.
Adresse IP	Adresse du contrôleur.
Port	Numéro du port utilisé par le pilote Modbus TCP. La valeur par défaut est **502**. Vous pouvez la changer lorsque la communication se fait par des routeurs ou des passerelles Internet où que le numéro de port par défaut est déjà utilisé.
utiliser UDP/IP	Si cette option a été sélectionnée, le protocole utilisera des datagrammes UDP sans connexion.
RTU encapsulé	Si cette option a été sélectionnée, le protocole utilisera de préférence le protocole RTU en série via Ethernet au lieu du protocole Modbus TCP, indépendamment du fait si TCP ou UDP est utilisé.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Élément	Description
Délai d'attente (ms)	Délai d'attente (en millisecondes) entre deux tentatives en cas d'absence de réponse du périphérique esclave.
ID Modbus	Généralement utilisé en cas de communication par passerelles Ethernet vers série, puis interprété comme l'ID esclave. Sa valeur est simplement copiée dans le champ Identificateur d'unité de la trame de communication du Modbus TCP. Ceci doit correspondre à la configuration du serveur. Dans la plupart des cas, le serveur répond à Modbus ID 1, de sorte que ce paramètre peut être laissé sur 1.
Bloc lecture max	Longueur maximale en octets d'une requête de bloc de données. Cette valeur s'applique uniquement à l'accès en lecture aux Registres d'exploitation.
Bloc de bits lecture max	Longueur maximale en bits d'une requête de bloc. Cette valeur s'applique uniquement à l'accès en lecture aux bits d'entrée et aux bobines de sortie.
Écriture du registre d'exploitation	Fonction Modbus pour les opérations d'écriture dans les registres de maintien. Veuillez choisir entre la fonction **06** (Prédéfinir registre seul) et la fonction **16** (Prédéfinir registres multiples). Si **06** est sélectionnée, le protocole utilisera toujours la fonction **06** pour l'écriture dans le contrôleur, même en écrivant dans plusieurs registres consécutifs. Si **16** est sélectionnée, le protocole utilisera toujours la fonction **16** pour l'écriture dans le contrôleur, même pour une requête d'écriture dans un registre seul, et le paramètre **Taille max bloc lecture** de la requête est configurée sur **2**. L'utilisation de la fonction **16** peut augmenter la performance de la communication. Si **Auto** est sélectionné, le protocole utilisera la fonction **06** ou la fonction **16** en fonction du nombre de registres à écrire.
Écrire bobines	Fonction Modbus pour les opérations d'écriture dans les bobines de sortie. Veuillez choisir entre la fonction **05** (Écrire bobine seule) et la fonction **15** (Écrire bobines multiples). Si la fonction Modbus **05** est sélectionnée, le protocole utilisera toujours la fonction **05** pour l'écriture dans le contrôleur, même en écrivant dans plusieurs bobines consécutives. Si la fonction Modbus **15** est sélectionnée, le protocole utilisera toujours la fonction **15** pour l'écriture dans le contrôleur, même pour une requête d'écriture dans une bobine seule. L'utilisation de la fonction **16** peut augmenter la performance de la communication.
Modèles d'API	Permet de choisir entre les différentes modèles d'API : • **Modicon Modbus (1-based)** : Implantation Modbus où toutes les ressources commencent avec décalage 1. • **Modicon Modbus (0-based)** : Implantation Modbus où toutes les ressources commencent avec décalage 0. • **Enron Modbus (1-based)** : étend l'implantation Modicon Modbus avec la zone de mémoire des registres 32 bits. • **Enron Modbus (0-based)** : étend l'implantation Generic Modbus avec la zone de mémoire des registres 32 bits.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

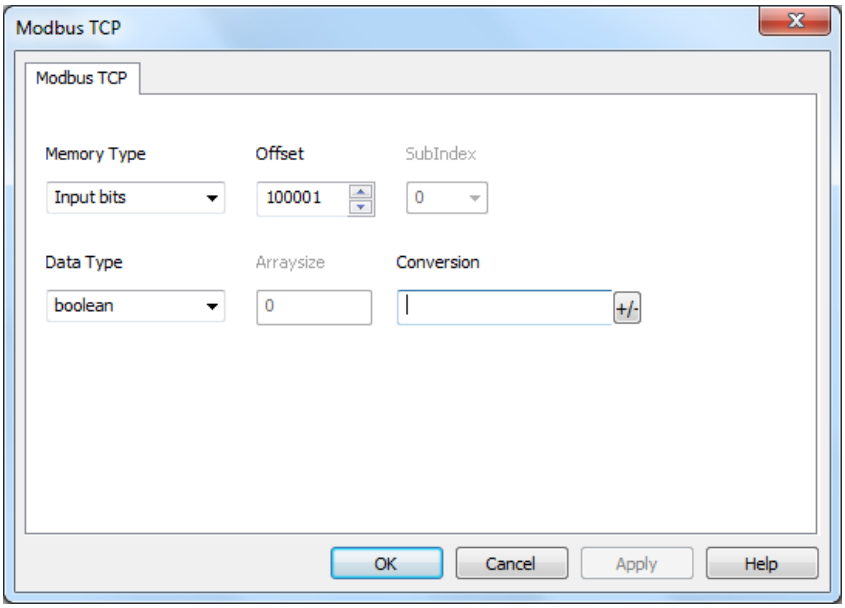
Élément	Description
	Remarque : la plage d'adresses utilisée dans les trames Modbus s'étend toujours de 0 à 65535 pour les registres d'exploitation et de 0 à 65535 pour les bobines.
Réseau d'API	Adresse IP de tous les contrôleurs en connexions multiples. Vous devez sélectionner la case à cocher **Réseau d'API** pour permettre les connexions multiples. 

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Paramètres de l’éditeur de variables

Chemin : ***VueProjet*** *>* ***Configuration*** *> double-cliquez sur* ***Variables***

1. Pour ajouter une variable, cliquez sur **+** : une nouvelle ligne est ajoutée.
2. Sélectionnez **Modbus TCP** dans la liste **Pilote** : le dialogue de définition de variables s'affiche.



Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

<table>
<tr><th>Élément</th><th>Description</th></tr>
<tr><td>Type de mémoire</td><td>Ressource de Modbus où se situe la variable.
<table>
<tr><th>Type de mémoire</th><th>Ressource Modbus</th></tr>
<tr><td>Statut de la bobine</td><td>Bobines</td></tr>
<tr><td>Statut d’entrée</td><td>Entrée discrète</td></tr>
<tr><td>Registres d'entrée</td><td>Registres d'entrée</td></tr>
<tr><td>Registres de maintien</td><td>Registres de maintien</td></tr>
<tr><td>Registres à 32 bits</td><td>Zone de mémoire des registres à 32 bits.
Disponible uniquement pour les modèles d’API Enron Modbus.</td></tr>
<tr><td>IP d'écrasement de nœud</td><td rowspan="4">paramètre de protocole (voir Types de données spéciaux pour plus d'informations concernant ce mode)</td></tr>
<tr><td>Port d'écrasement de nœud</td></tr>
<tr><td>ID d'écrasement de nœud</td></tr>
<tr><td>Mode Modicon</td></tr>
</table></td></tr>
<tr><td>Offset</td><td>Adresse de décalage où se situe la variable.
Les adresses de décalage sont six chiffres composés par un préfixe de type de données à un chiffre + une adresse de ressource à cinq chiffres.
<table>
<tr><th>Type de mémoire</th><th>Studio
Plage de décalage</th><th>Modicon
Plage de décalage</th><th>Generic Modbus
Plage de décalage</th></tr>
<tr><td>Statut de la bobine</td><td>0 – 65535</td><td rowspan="5">1 – 65536</td><td rowspan="5">0 – 65535</td></tr>
<tr><td>Statut d’entrée</td><td>100000 – 165535</td></tr>
<tr><td>Registres d'entrée</td><td>300000 – 365535</td></tr>
<tr><td>Registres de maintien</td><td>400000 – 465535</td></tr>
<tr><td>Registres à 32 bits</td><td>0 – 65535</td></tr>
</table></td></tr>
<tr><td>Sousindex</td><td>Permet la sélection du décalage de ressource dans le registre.</td></tr>
</table>

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

<table>
<tr><th>Élément</th><th>Description</th></tr>
<tr><td>Type de données</td><td>
<table>
<tr><th>Type de données</th><th>Espace mémoire</th><th>Limites</th></tr>
<tr><td>booléen</td><td>Donnée à 1 bit</td><td>0 ... 1</td></tr>
<tr><td>octet</td><td>Donnée à 8 bits</td><td>-128 ... 127</td></tr>
<tr><td>court</td><td>Donnée à 16 bits</td><td>-32768 ... 32767</td></tr>
<tr><td>entier</td><td>Donnée à 32 bits</td><td>-2.1e9 ... 2.1e9</td></tr>
<tr><td>int64</td><td>Donnée à 64 bits</td><td>-9.2e18 ... 9.2e18</td></tr>
<tr><td>octet non signé</td><td>Donnée à 8 bits</td><td>0 ... 255</td></tr>
<tr><td>court non signé</td><td>Donnée à 16 bits</td><td>0 ... 65535</td></tr>
<tr><td>entier non signé</td><td>Donnée à 32 bits</td><td>0 ... 4.2e9</td></tr>
<tr><td>uint64</td><td>Donnée à 64 bits</td><td>0 ... 1.8e19</td></tr>
<tr><td>flottant</td><td>Type virgule flottante de 32 bits simple précision conforme IEEE</td><td>1.17e-38 ... 3.4e38</td></tr>
<tr><td>double</td><td>Type virgule flottante de 64 bits double précision conforme IEEE</td><td>2.2e-308 ... 1.79e308</td></tr>
<tr><td>chaîne</td><td colspan="2">Tableau des éléments contenant du code de caractères défini par l'encodage sélectionné</td></tr>
<tr><td>binaire</td><td colspan="2">Données binaires arbitraires</td></tr>
</table>

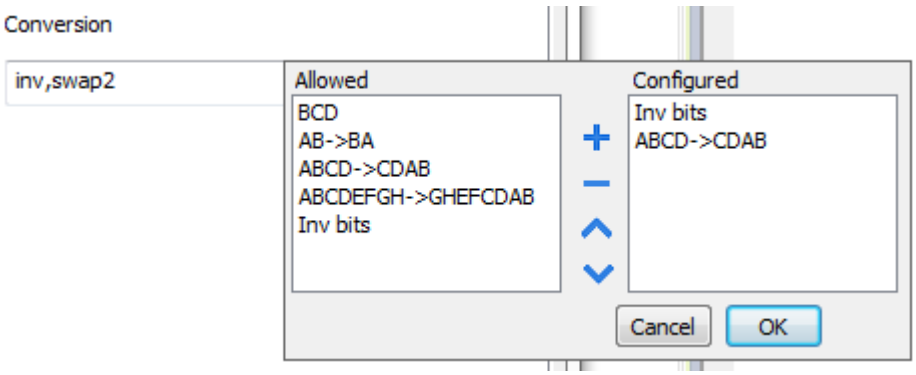
Remarque : pour définir des tableaux, sélectionnez l’un des formats de type de données suivi de crochets tels que « byte[] », « short[] », etc.
</td></tr>
<tr><td>Taille de tableau</td><td>

En cas de variable vectorielle, cette propriété représente le nombre d’éléments vectoriels.
En cas de variable de chaîne, cette propriété représente le nombre maximum d’octets disponible pour la variable de chaîne.

Remarque : le nombre d'octets correspond au nombre de caractères de chaîne si la propriété Encoding est définie sur UTF-8 ou Latin1 dans l'Éditeur de variables.
Si la propriété Encoding est définie sur UCS-2BE, UCS-2LE, UTF-16BE ou UTF-16LE, un caractère nécessite 2 octets.
</td></tr>
<tr><td>Conversion</td><td>Conversion à appliquer à la variable.</td></tr>
</table>

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Élément	Description



En fonction du type de données choisi, la liste **Autorisée** affiche un ou plusieurs types de conversion.

Valeur	Description
Inv bits	**inv** : inverse tous les bits de la variable. *Exemple :* 1001 → 0110 (format binaire) 9 → 6 (format décimal)
Inverser	**neg** : inverse la valeur de la variable. *Exemple :* 25.36 → -25.36
AB -> BA	**swapnibbles** : échange les quartets en un octet. *Exemple :* 15D4 → 514D (format hexadécimal) 5588 → 20813 (format décimal)
ABCD -> CDAB	**swap2** : échange les octets en un mot. *Exemple :* 9ACC → CC9A (format hexadécimal) 39628 → 52378 (format décimal)
ABCDEFGH -> GHEFCDAB	**swap4** : échange les octets en un mot double. *Exemple :* 32FCFF54 → 54FFFC32 (format hexadécimal) 855441236 → 1426062386 (format décimal)
ABC...NOP -> OPM...DAB	**swap8** : échange les octets en un mot long. Exemple : 142,366 → -893553517,588905 (format décimal) 0 10000000110 00011100101110110110010001011010000111001010110000001

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Élément	Description
	<table><tr><th>Valeur</th><th>Description</th></tr><tr><td></td><td>→ 1 10000011100 10101010000101000101101101101100101101100000100111101 (format binaire)</td></tr><tr><td>BCD</td><td>**bcd** : sépare l'octet en deux quartets, les lit au format décimal (de 0 à 9) *Exemple :* 23 → 17 (format décimal) 0001 0111 = 23 0001 = 1 (premier quartet) 0111 = 7 (second quartet)</td></tr></table> Sélectionnez la conversion et cliquez sur le bouton +. La conversion sélectionnée s'affiche dans la liste **Configurée**. Si vous configurez d'autres conversions, celles-ci s'affichent dans l'ordre (du haut vers le bas de la liste **Configurée**). Utilisez les boutons de direction pour changer l'ordre des conversions configurées.

IP d'écrasement de nœud

Le protocole propose le type de données spécial IP d'écrasement de nœud, qui vous permet de modifier l'adresse IP du contrôleur cible au moment du runtime.

Ce type de mémoire est un tableau de 4 octets non signés, un pour chaque octet de l'adresse IP.

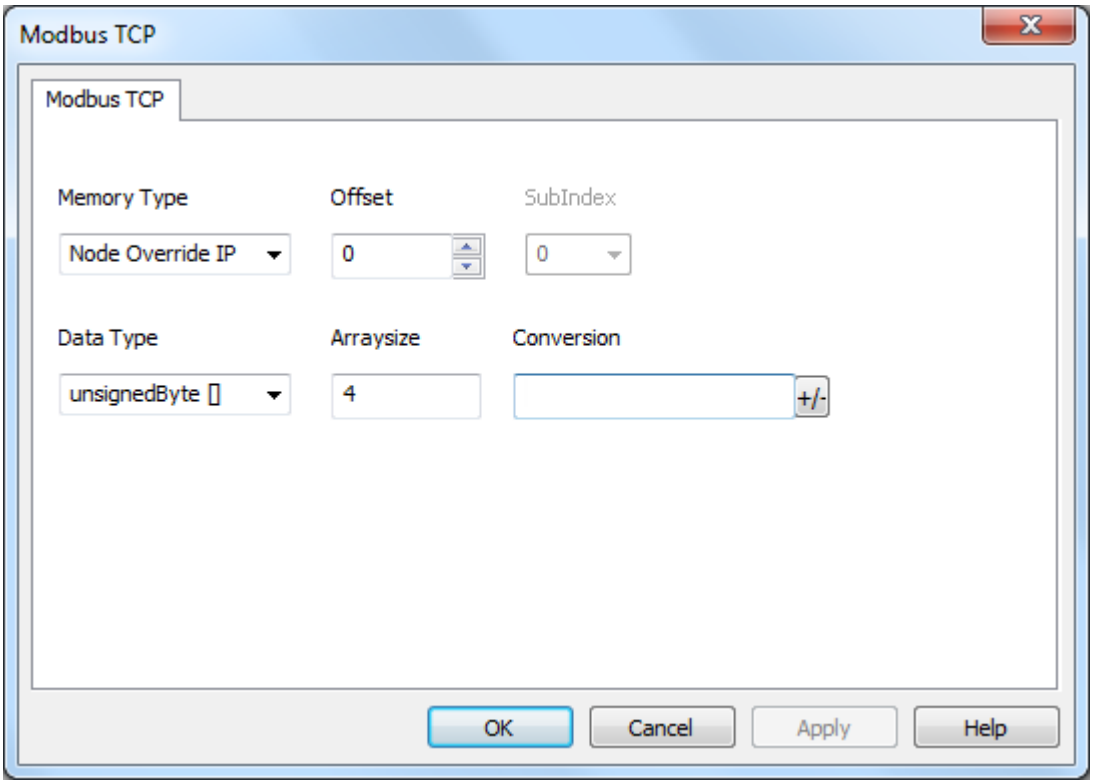
L'IP d'écrasement de nœud est initialisée avec la valeur de l'IP du contrôleur spécifiée dans le projet au moment de la programmation.

IP d'écrasement de nœud	Exécution de l'API
0.0.0.0	La communication avec le contrôleur est coupée, plus aucune trame de requête ne sera générée.
Autre que 0.0.0.0	Interprétée comme étant l'écrasement de l'IP de nœud, et remplace l'adresse IP cible au moment du runtime.

Lorsque le périphérique IHM est connecté à un réseau avec plus d'un nœud de contrôleur, chaque nœud dispose de sa propre variable d'IP d'écrasement de nœud.

Remarque : le maintien des valeurs de l'IP d'écrasement de nœud assignées au runtime est assuré par des cycles d'alimentation.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



Port d'écrasement de nœud

Le protocole propose le type de données spécial port d'écrasement de nœud, qui vous permet de modifier le port réseau du contrôleur cible au moment du runtime.

Ce type de mémoire correspond à un short non signé.

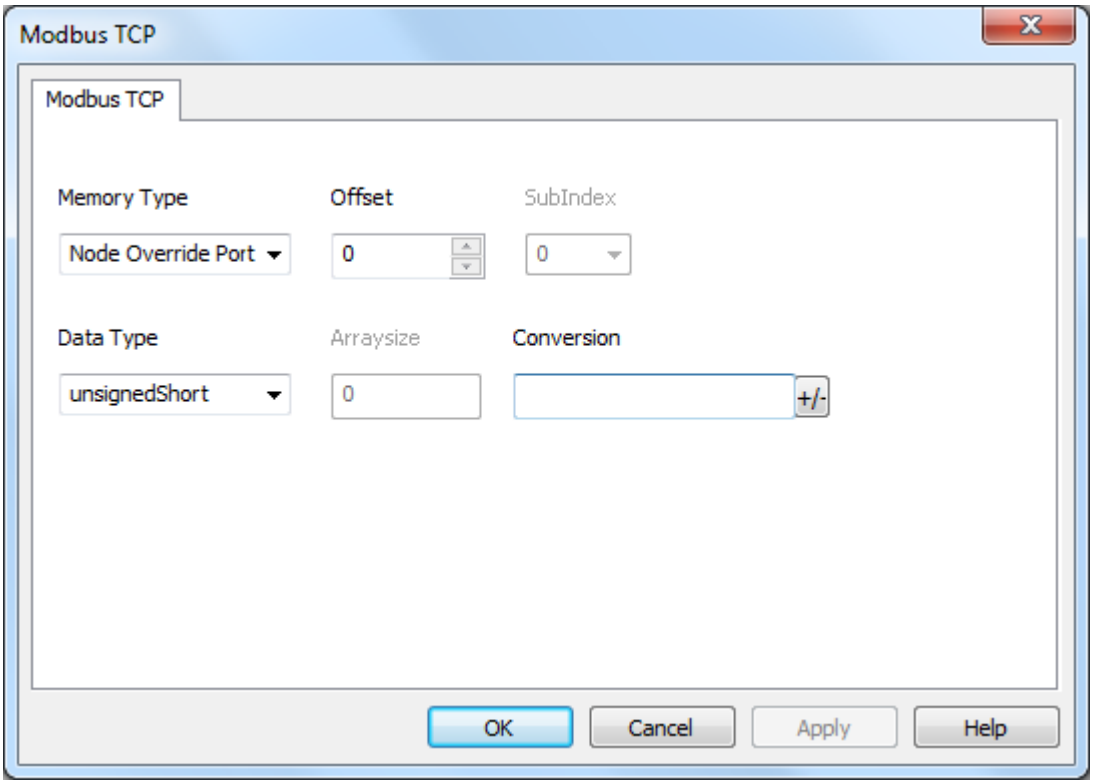
Le port d'écrasement de nœud est initialisé avec la valeur du port du contrôleur spécifiée dans le projet au moment de la programmation.

Port d'écrasement de nœud	Opération Modbus
0	La communication avec le contrôleur est coupée, plus aucune trame de requête ne sera générée.
Autre que 0	Interprété comme étant la valeur du nouveau port et remplacé pour l'opération de runtime.

Lorsque le périphérique IHM est connecté à un réseau avec plus d'un nœud de contrôleur, chaque nœud dispose de sa propre variable de port d'écrasement de nœud.

Remarque : le maintien des valeurs de port d'écrasement de nœud assignées au runtime est assuré par des cycles d'alimentation.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



ID d'écrasement de nœud

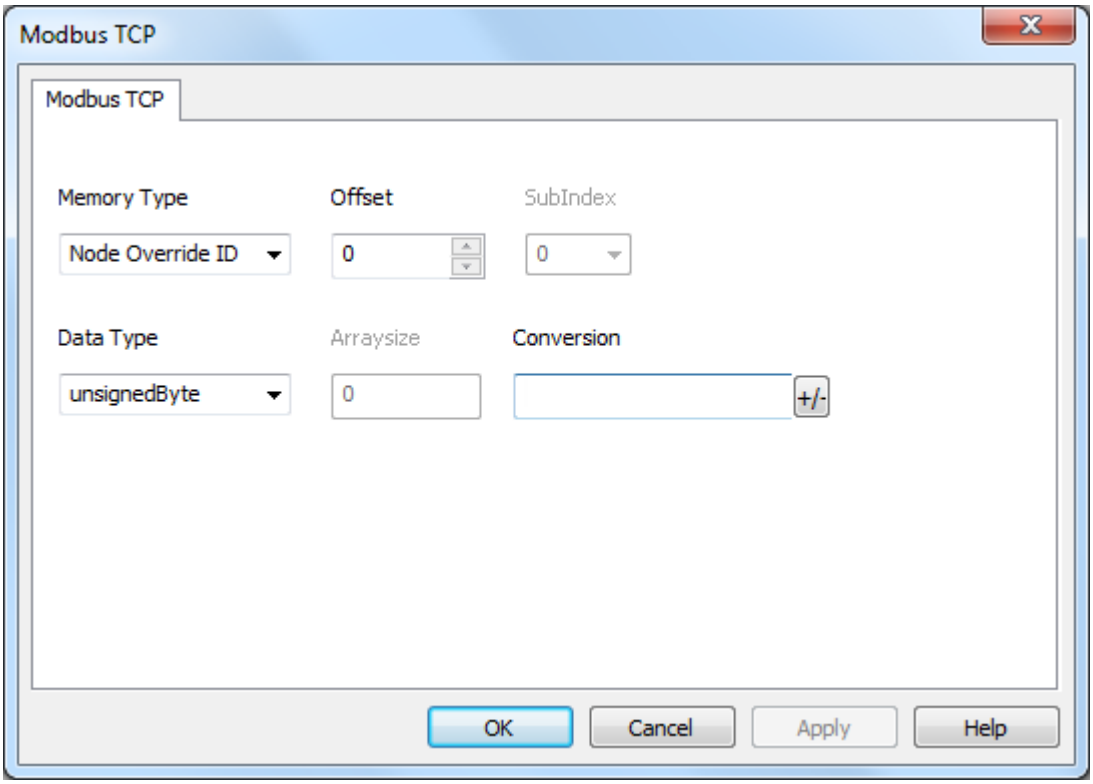
Le protocole propose le type de données spécial ID d'écrasement de nœud, qui vous permet de modifier l'ID de nœud de l'esclave au runtime. Ce type de mémoire est un octet non signé.

L'ID d'écrasement de nœud est initialisé avec la valeur de l'ID de nœud spécifiée dans le projet au moment de la programmation.

ID d'écrasement de nœud	Opération Modbus
0	La communication avec le contrôleur est coupée. En cas d'opération d'écriture, la requête sera transmise sans attente de réponse.
1 à 254	Interprété comme étant la valeur du nouvel ID de nœud et remplacé pour l'opération de runtime.
255	La communication avec le contrôleur est coupée, aucun message de requête ne sera généré.

Remarque : le maintien de la valeur de l'ID d'écrasement de nœud assignée au runtime est assuré par des cycles d'alimentation.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



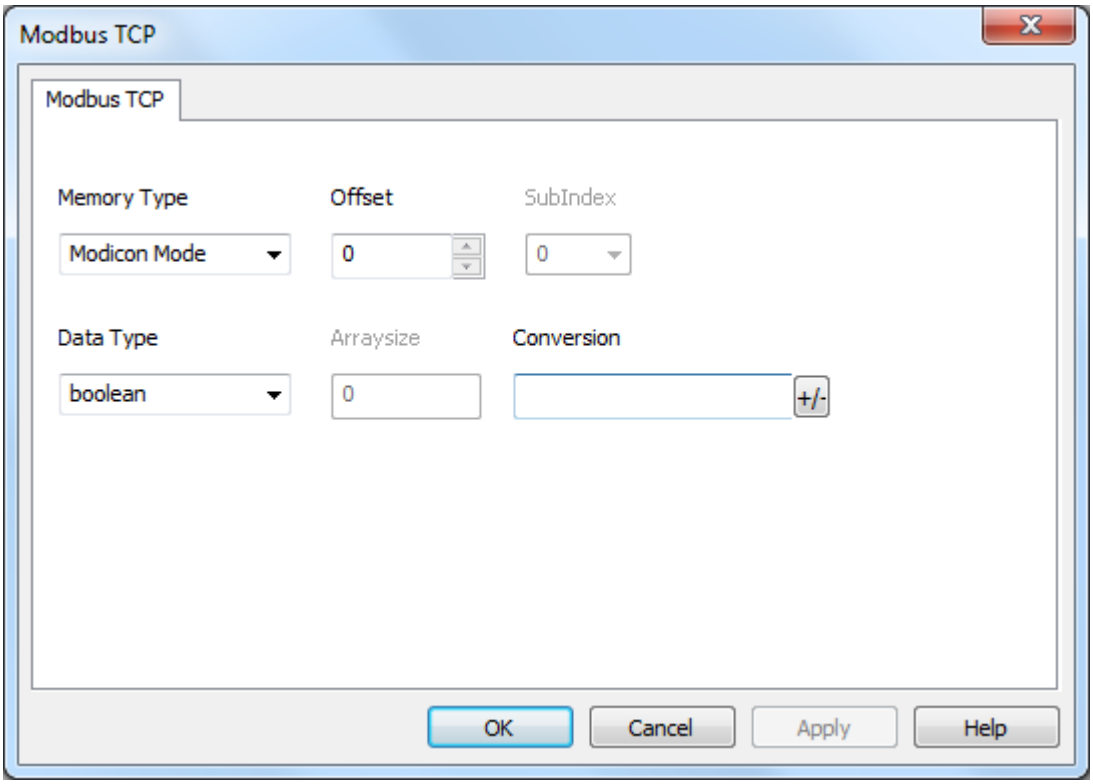
Mode Modicon

Ce protocole fournit un type de données spécial qui permet d'écraser les paramètres du mode Modicon au runtime.

Mode Modicon	Description
0	Generic Modbus (0-based). Les index de registre commencent à 0.
1	Modicon Modbus (1-based). Les index de registre commencent à 1.

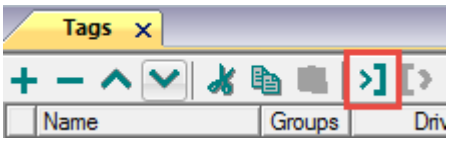
Remarque : la valeur du paramètres de mode Modicon attribuée au runtime est conservée à travers les cycles d'alimentation.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



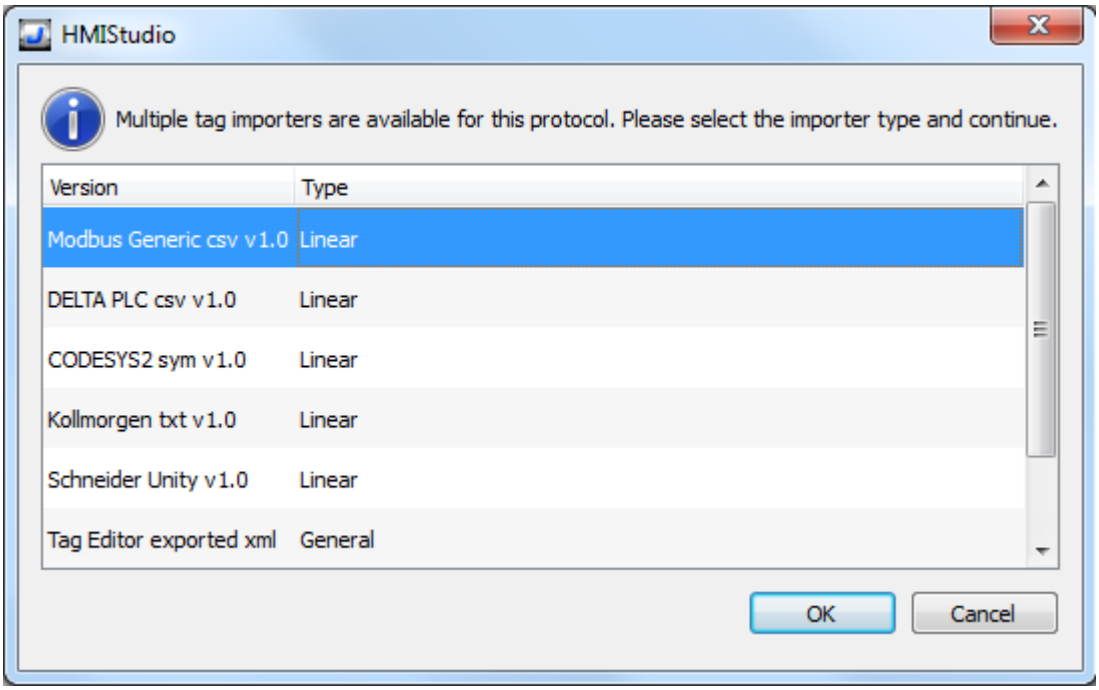
Importation de variables

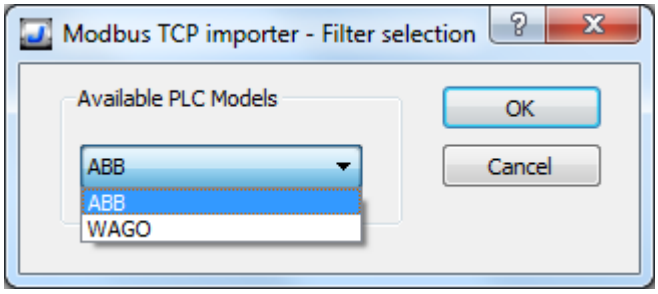
Sélectionnez le pilote dans l'éditeur de variables et cliquez sur le bouton **Importer variables** pour lancer le pilote d'importation.



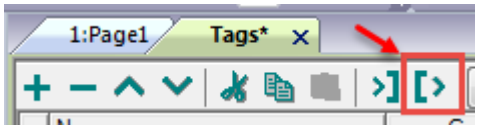
La boîte de dialogue suivante affiche le type d'importation que vous pouvez sélectionner.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



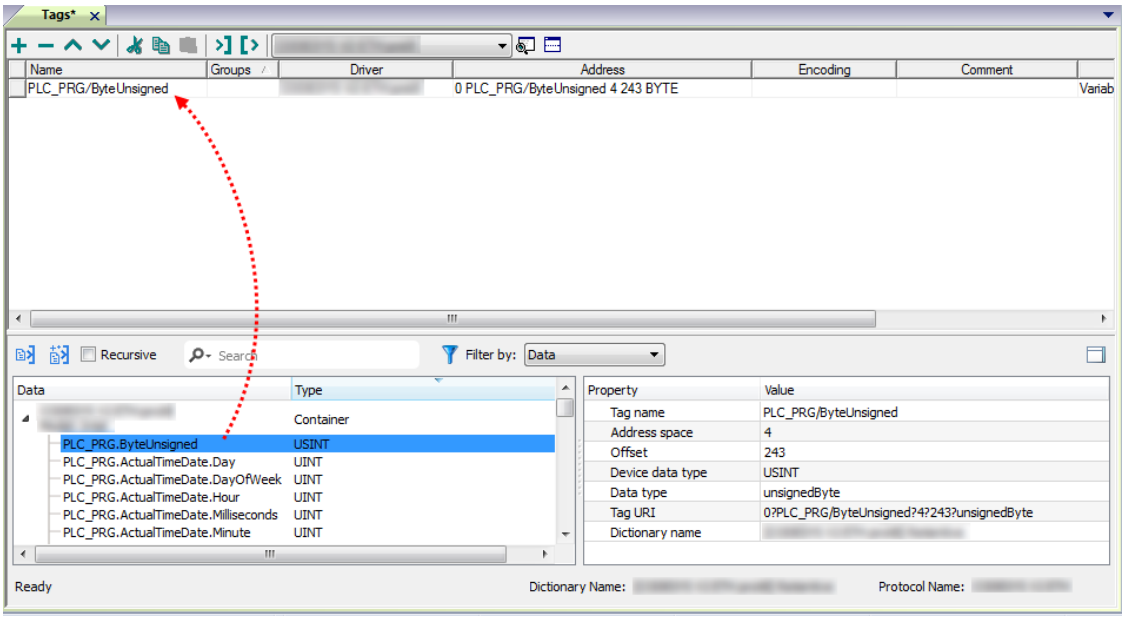
Type	Description
Modbus générique csv v1.0 **Linéaire**	Nécessite un fichier **.csv**. Toutes les variables s'affichent au même niveau.
DELTA PLC csv v1.0	Nécessite un fichier **.csv**. Toutes les variables s'affichent au même niveau.
CODESYS2 sym v1.0 **Linéaire**	Nécessite un fichier **.sym**. Toutes les variables s'affichent au même niveau. Une fois le fichier **.sym** sélectionné, le dialogue suivant s'affichera pour la sélection de modèle d'API. 
Kollmorgen txt v1.0 **Linéaire**	Nécessite un fichier **.txt**. Toutes les variables s'affichent au même niveau.
Schneider Unity v1.0 **Linear**	Requiert un fichier **.uny**.

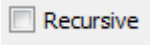
Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Type	Description
	Le fichier contenant des symboles doit être exporté au format **.txt** et plus tard rebaptisé **.uny**. Le pilote d'importation considère uniquement les variables situées à l'adresse fixe et ne tient pas compte des tableaux de chaînes. Tous les autres tableaux, sauf pour le type booléen, sont développés.
Fichier .xml exporté de l'éditeur de variables	Sélectionnez ce pilote d'importation pour lire un fichier XML générique exporté de l'éditeur de variables avec le bouton approprié. 

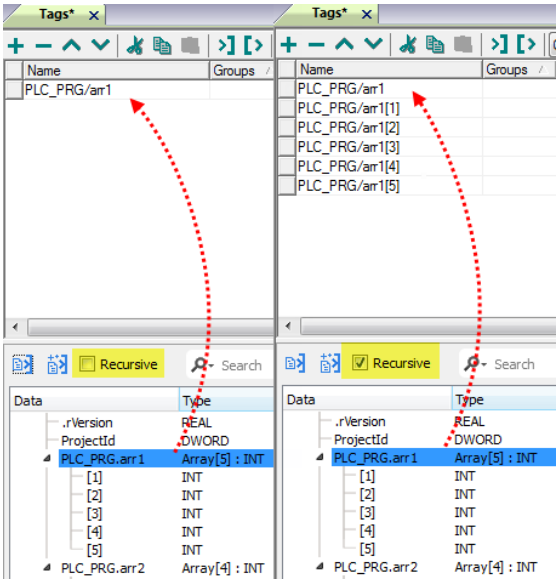
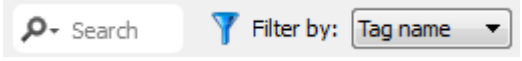
Lorsque vous avez sélectionné le pilote d'importation, localisez le fichier de symboles et cliquez sur **Ouvrir**.

Les variables comprises dans le fichier de symboles sont listées dans le dictionnaire des variables. Le dictionnaire des variables est affiché en bas de l'écran.



Élément de la barre d'outils	Description
	Importer variable(s). Sélectionnez les variables à importer et cliquez sur cette icône pour ajouter les variables du dictionnaire au projet.
	Mettre à jour variable(s). Cliquez sur cette icône pour mettre à jour les variables du projet grâce à un nouvel import du dictionnaire.
	Cochez cette case pour importer tous les sous-éléments

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Élément de la barre d'outils	Description
	d'une variable. Exemples de résultats avec la case cochée et non cochée : 
	Recherche des variables dans le dictionnaire en filtrant en fonction de l'élément sélectionné dans la liste déroulante.

Structure de fichier csv Modbus Generic

Ce protocole prend en charge l'importation des informations de variable lorsque vous les lui fournissez au format **.csv**, conformément au format suivant :

```
NodeID, TagName, MemoryType, Address, DataFormat,...,[Comment]
```

Remarque : les champs entre crochets ainsi que les champs entre DataFormat et Comment sont facultatifs.

Champ	Description
NodeID	Nœud auquel appartient la variable
TagName	Description de la variable
Type de mémoire	• OUTP • INP • IREG • HREG
Adresse	Offset compatible avec la notation Modbus

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Champ	Description
FormatDeDonnées	Type de données dans la notation interne. Consultez la section « Concepts de programmation » du manuel principal.
Commentaire	Description supplémentaire facultative.

Exemple de fichier de variables

Exemple de ligne .csv :

```
2, Holding Register 1, HREG, 400001, unsignedShort,
```

Remarque : cette ligne est sans commentaire. Lorsque ce champ est manquant, le caractère de fin de ligne doit être une virgule.

Statut de communication

Vous pouvez afficher l'état de communication actuel grâce aux variables système. Voir la section « Variables système » du manuel principal.

Codes pris en charge pour ce pilote de communication :

Erreur	Cause	Action
Pas de réponse	Pas de retour dans le délai d'attente configuré.	Vérifiez que le contrôleur est connecté et correctement configuré pour accéder au réseau.
La réponse contient une adresse de nœud incorrecte	Le contrôleur a envoyé une réponse contenant une adresse de nœud non valide du périphérique.	-
Le message reçu est trop court	Le contrôleur a envoyé une réponse dans un format non valide du périphérique.	-
Écriture incorrecte confirmation des données	Le contrôleur a refusé une demande d'écriture.	Vérifiez la cohérence entre les données de projet et les ressources du contrôleur.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Serveur Modbus TCP

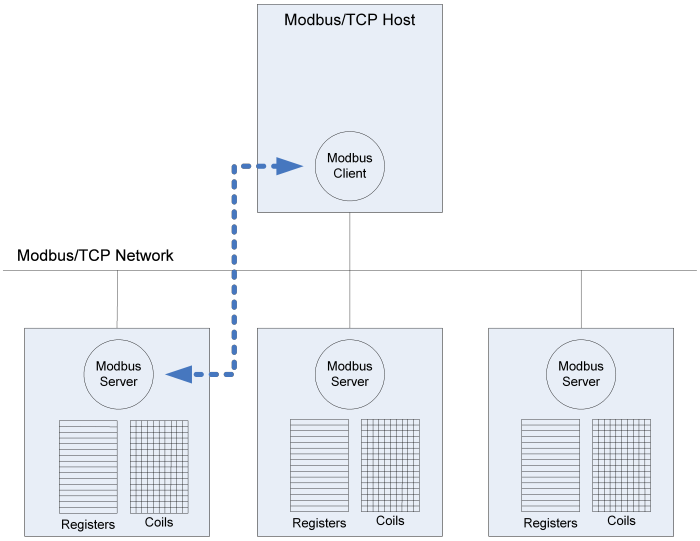
Le pilote de communication Modbus TCP Server permet de connecter le périphérique IHM en tant qu'esclave à un réseau TCP Modbus. il est possible que les clients Modbus TCP se connectent ensuite à plusieurs panneaux IHM agissant comme serveurs. Les messages normalisés TCP Modbus sont utilisés pour l'échange d'informations.

Cette approche permet de connecter des périphériques IHM à des systèmes SCADA à travers le protocole de communication TCP Modbus pris en charge universellement.

Principe de fonctionnement

Ce pilote de communication implémente une unité serveur Modbus TCP dans un périphérique IHM. Un sous-ensemble de la gamme complète des codes de fonction Modbus est pris en charge. Les codes de fonction disponibles permettent le transfert de données entre les clients sur le réseau TCP et le serveur. L'IHM agit comme un serveur dans le réseau. Il peut échanger des données avec jusqu'à 32 clients. Cela signifie que jusqu'à 32 clients peuvent être connectés à l'appareil IHM en même temps. Si toutes les 32 connexions disponibles sont utilisées, toute nouvelle tentative de connexion par un client sera refusée par le système.

Le diagramme suivant représente l'architecture de système.



Le périphérique simule l'interface de communication d'un API : les types de données des bobines et des registres sont respectivement booléens et entiers à 16 bits.

Le périphérique peut à tout moment accéder aux données de sa mémoire interne. Le transfert de données de et vers un client Modbus est possible uniquement sur l'initiative du client même.

Détails sur l'implantation

Cette mise en œuvre du serveur TCP Modbus prend seulement en charge un sous-ensemble des codes fonction Modbus standard.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Code	Fonction	Description
01	Lecture du statut de la bobine	Lecture de bits multiples dans la zone du périphérique bobine.
02	Lecture du statut d'entrée	Lecture de bits multiples dans la zone du périphérique bobine.
03	Lecture des registres de maintien	Lecture de registres multiples du périphérique.
04	Lecture des registres d'entrée	Lecture de registres multiples du périphérique.
05	Forcer bobine seule	Force une bobine unique du périphérique à passer en état « MARCHE » ou « ARRÊT ».
06	Prédéfinir registre seul	Prédéfinit une valeur dans un registre du périphérique.
15	Forcer plusieurs bobines	Force plusieurs bobines du périphérique à passer au statut « MARCHE » ou « ARRÊT ».
16	Prédéfinir registres multiples	Prédéfinit une valeur dans plusieurs registres du périphérique.
23	Lecture/écriture de plusieurs registres	Lit et prédéfinit une valeur dans plusieurs registres du périphérique

Remarque : pour les deux types d'API, les fonctions Lecture de l'état de la bobine et Lecture de l'état d'entrée accèdent toutes les deux à la même mémoire de bobine dans la mémoire du périphérique IHM. Les fonctions Lecture des registres de maintien et Lecture des registres d'entrée accèdent toutes les deux à la même zone de registre dans la mémoire du périphérique IHM.

Codes d'exception

Code	Description
01	**Fonction interdite**. le code de fonction de la requête n'est pas pris en charge
02	**Adresse de données interdite** . L'adresse des données reçue au cours de la requête dépasse la plage de données prédéfinie (voir **Paramètre d'édition variable** pour afficher les détails des plages pour tous les types).
03	**Valeur de données interdite**. Une sous-fonction autre que 00 est spécifiée dans l'essai de diagnostic en boucle (Code 08),

Paramètres de l'éditeur de protocole

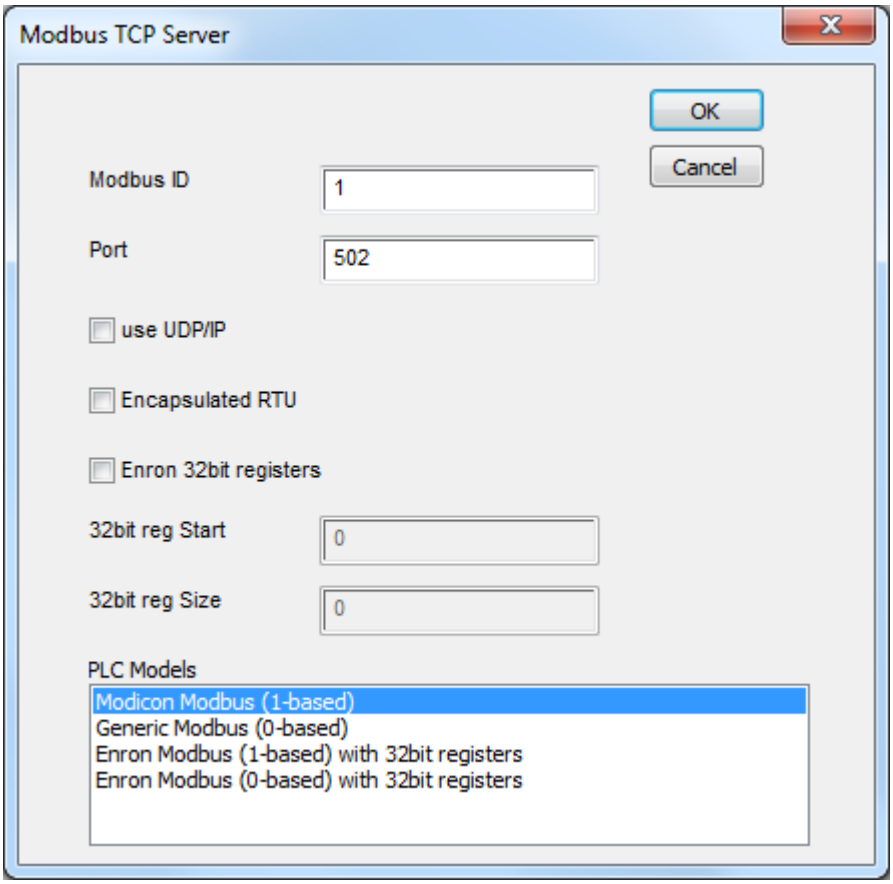
Ajouter un protocole

Pour configurer le protocole :

1. Dans le nœud **Config**, double-cliquez sur **Protocoles**.
2. Pour ajouter un pilote, cliquez sur **+** : une nouvelle ligne est ajoutée.
3. Sélectionnez le protocole dans la liste **API**.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

La boîte de dialogue de configuration du protocole s'affiche.



Élément	Description
ID Modbus	ID de nœud Modbus du périphérique IHM. Chaque périphérique de serveur Modbus du réseau doit avoir sa propre ID Modbus.
Port	Numéro du port utilisé par le protocole TCP Modbus. La valeur par défaut est **502**. Configurez la valeur selon le numéro de port utilisé par votre réseau TCP Modbus.
utiliser UDP/IP	Si cette option a été sélectionnée, le protocole utilisera des datagrammes UDP sans connexion.
RTU encapsulé	Si cette option a été sélectionnée, le protocole utilisera de préférence le protocole RTU en série via Ethernet au lieu du protocole Modbus TCP, indépendamment du fait si TCP ou UDP est utilisé.
Registres Enron 32 bits	Si sélectionné, il permet de définir la première adresse de registre et le nombre de registres pour la zone de mémoire des registres à 32 bits. Remarque : les registres à 32 bit sont disponibles uniquement pour les modèles d'API **Enron Modbus**.
32bit reg Start	Définition de la zone de mémoire des registres à 32 bits. La valeur **Start** représente la première adresse de registre. La valeur **Size** représente le nombre de registres.

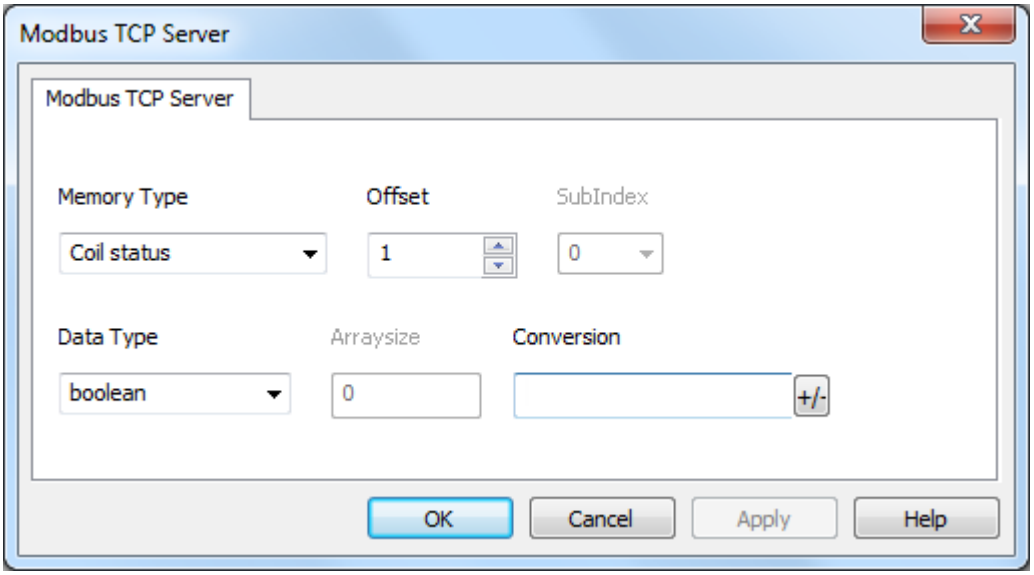
Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Élément	Description
32bit reg Size	Remarque : la réponse à une requête de l'un des registres de cette zone sera 4 octets.
Modèles d'API	Permet de choisir entre les différentes modèles d'API : • **Modicon Modbus (1-based)** : Implantation Modbus où toutes les ressources commencent avec décalage 1. • **Modicon Modbus (0-based)** : Implantation Modbus où toutes les ressources commencent avec décalage 0. • **Enron Modbus (1-based)** : étend l'implantation Modicon Modbus avec la zone de mémoire des registres 32 bits. • **Enron Modbus (0-based)** : étend l'implantation Generic Modbus avec la zone de mémoire des registres 32 bits. Remarque : la plage d'adresses utilisée dans les trames Modbus s'étend toujours de 0 à 65535 pour les registres d'exploitation et de 0 à 65535 pour les bobines.

Paramètres de l'éditeur de variables

*Chemin : **VueProjet**> **Configuration** > double-cliquez sur **Variables***

1. Pour ajouter une variable, cliquez sur **+** : une nouvelle ligne est ajoutée.
2. Sélectionnez **Modbus TCP Server** dans la liste des protocoles : le dialogue de définition de variables s'affiche.



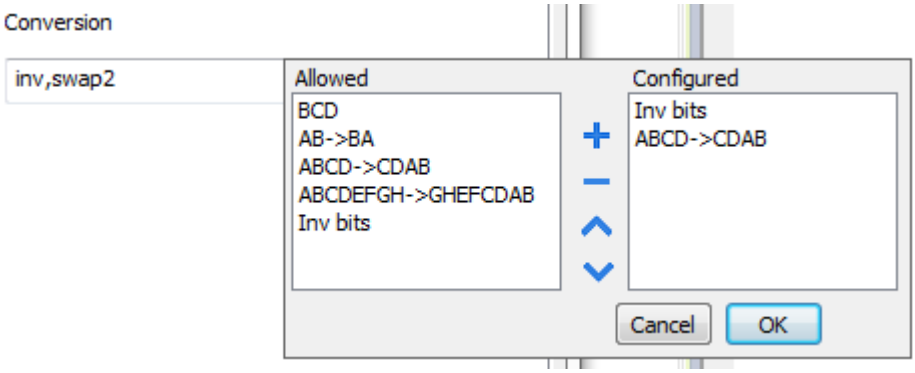
Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

<table>
<tr><th>Élément</th><th>Description</th></tr>
<tr><td>Type de mémoire</td><td>Ressource de Modbus où se situe la variable.
<table>
<tr><th>Type de mémoire</th><th>Ressource Modbus</th></tr>
<tr><td>Statut de la bobine</td><td>Bobines</td></tr>
<tr><td>Statut d’entrée</td><td>Entrée discrète</td></tr>
<tr><td>Registres d'entrée</td><td>Registres d'entrée</td></tr>
<tr><td>Registres de maintien</td><td>Registres de maintien</td></tr>
<tr><td>Registres à 32 bits</td><td>Zone de mémoire des registres à 32 bits.
Disponible uniquement pour les modèles d’API Enron Modbus.</td></tr>
<tr><td>Mode Modicon</td><td>paramètre de protocole (voir Types de données spéciaux pour plus d'informations concernant ce mode)</td></tr>
</table></td></tr>
<tr><td>Offset</td><td>Adresse de décalage où se situe la variable.
Les adresses de décalage sont six chiffres composés par un préfixe de type de données à un chiffre + une adresse de ressource à cinq chiffres.
<table>
<tr><th>Type de mémoire</th><th>Studio
Plage de décalage</th><th>Modicon
Plage de décalage</th><th>Generic Modbus
Plage de décalage</th></tr>
<tr><td>Statut de la bobine</td><td>0 – 65535</td><td rowspan="5">1 – 65536</td><td rowspan="5">0 – 65535</td></tr>
<tr><td>Statut d’entrée</td><td>100000 – 165535</td></tr>
<tr><td>Registres d'entrée</td><td>300000 – 365535</td></tr>
<tr><td>Registres de maintien</td><td>400000 – 465535</td></tr>
<tr><td>Registres à 32 bits</td><td>0 – 65535</td></tr>
</table></td></tr>
<tr><td>Sousindex</td><td>Permet la sélection du décalage de ressource dans le registre.</td></tr>
<tr><td>Type de données</td><td>
<table>
<tr><th>Type de données</th><th>Espace mémoire</th><th>Limites</th></tr>
<tr><td>booléen</td><td>Donnée à 1 bit</td><td>0 ... 1</td></tr>
<tr><td>octet</td><td>Donnée à 8 bits</td><td>-128 ... 127</td></tr>
<tr><td>court</td><td>Donnée à 16 bits</td><td>-32768 ... 32767</td></tr>
<tr><td>entier</td><td>Donnée à 32 bits</td><td>-2.1e9 ... 2.1e9</td></tr>
</table></td></tr>
</table>

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

<table>
<tr><th>Élément</th><th>Description</th></tr>
<tr><td></td><td>
<table>
<tr><th>Type de données</th><th>Espace mémoire</th><th>Limites</th></tr>
<tr><td>int64</td><td>Donnée à 64 bits</td><td>-9.2e18 ... 9.2e18</td></tr>
<tr><td>octet non signé</td><td>Donnée à 8 bits</td><td>0 ... 255</td></tr>
<tr><td>court non signé</td><td>Donnée à 16 bits</td><td>0 ... 65535</td></tr>
<tr><td>entier non signé</td><td>Donnée à 32 bits</td><td>0 ... 4.2e9</td></tr>
<tr><td>uint64</td><td>Donnée à 64 bits</td><td>0 ... 1.8e19</td></tr>
<tr><td>flottant</td><td>Type virgule flottante de 32 bits simple précision conforme IEEE</td><td>1.17e-38 ... 3.4e38</td></tr>
<tr><td>double</td><td>Type virgule flottante de 64 bits double précision conforme IEEE</td><td>2.2e-308 ... 1.79e308</td></tr>
<tr><td>chaîne</td><td colspan="2">Tableau des éléments contenant du code de caractères défini par l'encodage sélectionné</td></tr>
<tr><td>binaire</td><td colspan="2">Données binaires arbitraires</td></tr>
</table>
Remarque : pour définir des tableaux, sélectionnez l’un des formats de type de données suivi de crochets tels que « byte[] », « short[] », etc.
</td></tr>
<tr><td>Taille de tableau</td><td>

En cas de variable vectorielle, cette propriété représente le nombre d’éléments vectoriels.
En cas de variable de chaîne, cette propriété représente le nombre maximum d’octets disponible pour la variable de chaîne.

Remarque : le nombre d'octets correspond au nombre de caractères de chaîne si la propriété Encoding est définie sur UTF-8 ou Latin1 dans l'Éditeur de variables.
Si la propriété Encoding est définie sur UCS-2BE, UCS-2LE, UTF-16BE ou UTF-16LE, un caractère nécessite 2 octets.
</td></tr>
<tr><td>Conversion</td><td>
Conversion à appliquer à la variable.


En fonction du type de données choisi, la liste Autorisée affiche un ou plusieurs types de
</td></tr>
</table>

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Élément	Description
	conversion.

Valeur	Description
Inv bits	**inv** : inverse tous les bits de la variable. *Exemple :* 1001 → 0110 (format binaire) 9 → 6 (format décimal)
Inverser	**neg** : inverse la valeur de la variable. *Exemple :* 25.36 → -25.36
AB -> BA	**swapnibbles** : échange les quartets en un octet. *Exemple :* 15D4 → 514D (format hexadécimal) 5588 → 20813 (format décimal)
ABCD -> CDAB	**swap2** : échange les octets en un mot. *Exemple :* 9ACC → CC9A (format hexadécimal) 39628 → 52378 (format décimal)
ABCDEFGH -> GHEFCDAB	**swap4** : échange les octets en un mot double. *Exemple :* 32FCFF54 → 54FFFC32 (format hexadécimal) 855441236 → 1426062386 (format décimal)
ABC...NOP -> OPM...DAB	**swap8** : échange les octets en un mot long. Exemple : 142,366 → -893553517,588905 (format décimal) 0 10000000110 0001110010111011011001000101101000011100101011000001 → 1 10000011100 1010101000010100010110110110110010110110000100111101 (format binaire)
BCD	**bcd** : sépare l'octet en deux quartets, les lit au format décimal (de 0 à 9) *Exemple :* 23 → 17 (format décimal) 0001 0111 = 23 0001 = 1 (premier quartet) 0111 = 7 (second quartet)

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

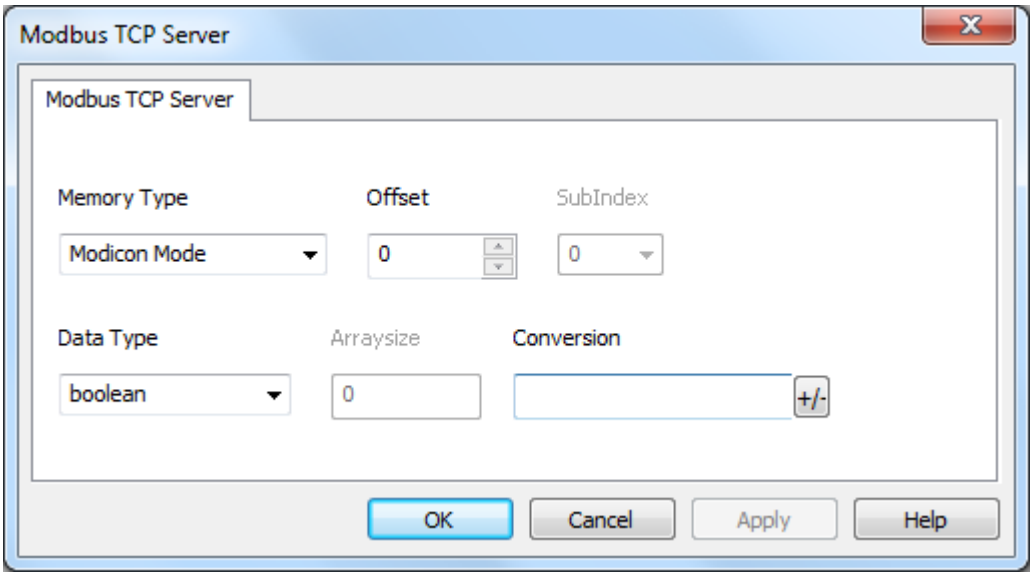
Élément	Description
	Sélectionnez la conversion et cliquez sur le bouton +. La conversion sélectionnée s'affiche dans la liste **Configurée**. Si vous configurez d'autres conversions, celles-ci s'affichent dans l'ordre (du haut vers le bas de la liste **Configurée**). Utilisez les boutons de direction pour changer l'ordre des conversions configurées.

Mode Modicon

Ce protocole fournit un type de données spécial qui permet d'écraser les paramètres du mode Modicon au runtime.

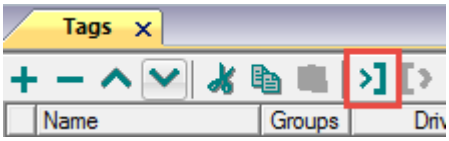
Mode Modicon	Description
0	Generic Modbus (0-based). Les index de registre commencent à 0.
1	Modicon Modbus (1-based). Les index de registre commencent à 1.

Remarque : la valeur du paramètres de mode Modicon attribuée au runtime est conservée à travers les cycles d'alimentation.



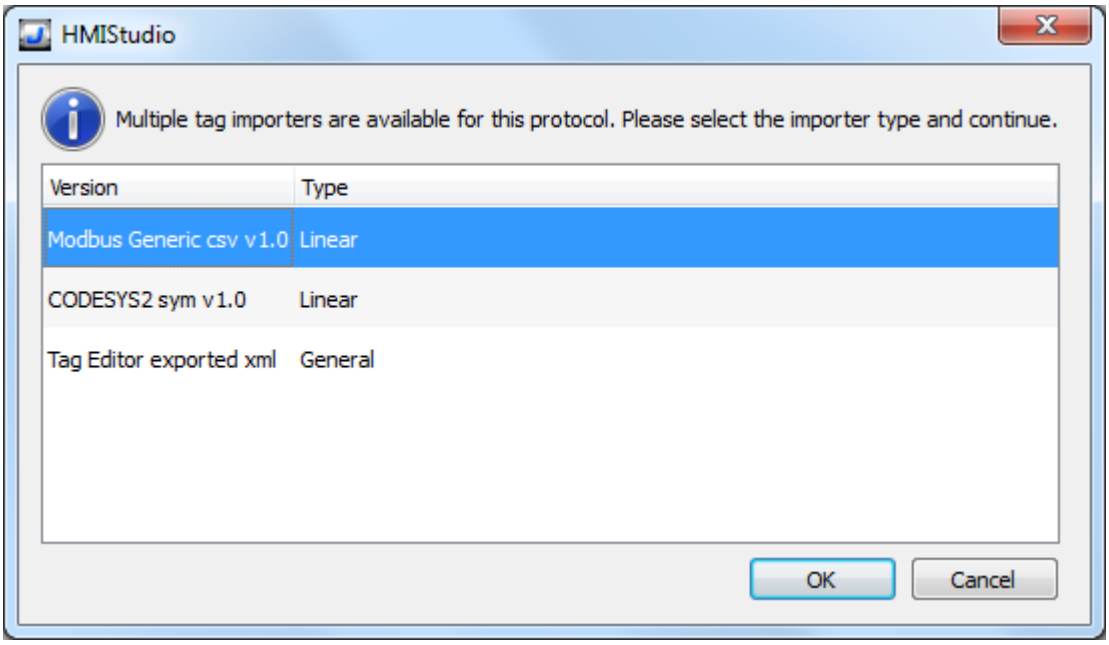
Importation de variables

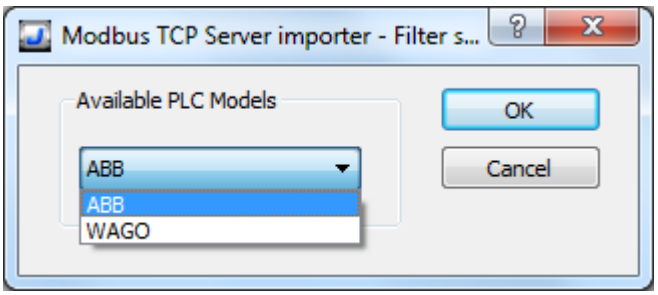
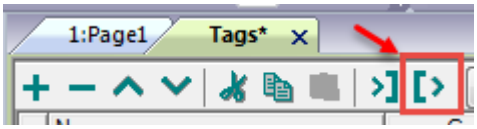
Sélectionnez le pilote dans l'éditeur de variables et cliquez sur le bouton **Importer variables** pour lancer le pilote d'importation.



La boîte de dialogue suivante affiche le type d'importation que vous pouvez sélectionner.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

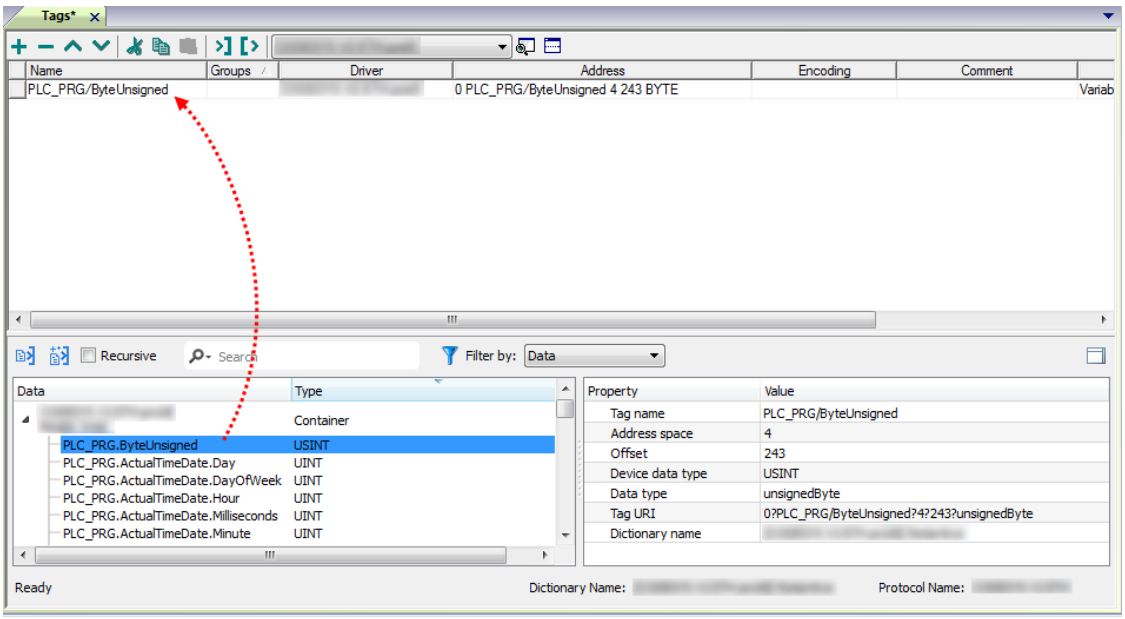


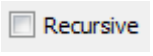
Importateur	Description
Modbus générique csv v1.0 Linéaire	Nécessite un fichier **.csv**. Toutes les variables s'affichent au même niveau.
CODESYS2 sym v1.0 Linéaire	Nécessite un fichier **.sym**. Toutes les variables s'affichent au même niveau. Une fois le fichier **.sym** sélectionné, le dialogue suivant s'affichera pour la sélection de modèle d'API. 
Fichier .xml exporté de l'éditeur de variables	Sélectionnez ce pilote d'importation pour lire un fichier XML générique exporté de l'éditeur de variables avec le bouton approprié. 

Lorsque vous avez sélectionné le pilote d'importation, localisez le fichier de symboles et cliquez sur **Ouvrir**.

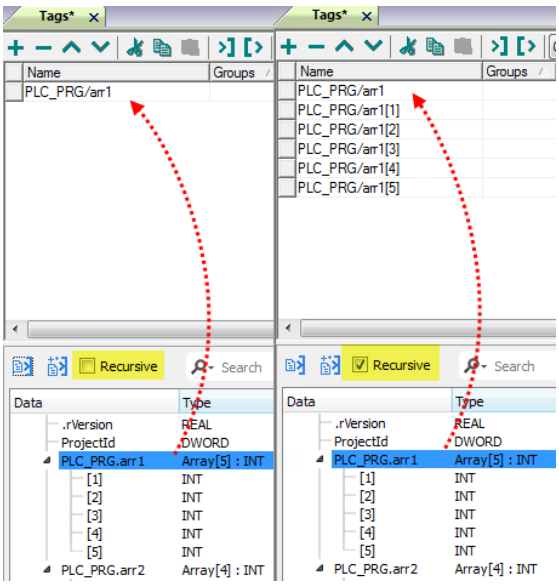
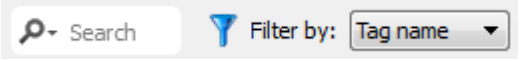
Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Les variables comprises dans le fichier de symboles sont listées dans le dictionnaire des variables. Le dictionnaire des variables est affiché en bas de l'écran.



Élément de la barre d'outils	Description
	Importer variable(s). Sélectionnez les variables à importer et cliquez sur cette icône pour ajouter les variables du dictionnaire au projet.
	Mettre à jour variable(s). Cliquez sur cette icône pour mettre à jour les variables du projet grâce à un nouvel import du dictionnaire.
	Cochez cette case pour importer tous les sous-éléments d'une variable. Exemples de résultats avec la case cochée et non cochée :

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Élément de la barre d'outils	Description
	
	Recherche des variables dans le dictionnaire en filtrant en fonction de l'élément sélectionné dans la liste déroulante.

Structure de fichier csv Modbus Generic

Ce protocole prend en charge l'importation des informations de variable lorsque vous les lui fournissez au format **.csv**, conformément au format suivant :

```
NodeID, TagName, MemoryType, Address, DataFormat,...,[Comment]
```

Remarque : les champs entre crochets ainsi que les champs entre DataFormat et Comment sont facultatifs.

Champ	Description
NodeID	Nœud auquel appartient la variable
TagName	Description de la variable
Type de mémoire	• OUTP • INP • IREG • HREG
Adresse	Offset compatible avec la notation Modbus
FormatDeDonnées	Type de données dans la notation interne. Consultez la section « Concepts de programmation » du manuel principal.
Commentaire	Description supplémentaire facultative.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Exemple de fichier de variables

Exemple de ligne .csv :

```
2, Holding Register 1, HREG, 400001, unsignedShort,
```

Remarque : cette ligne est sans commentaire. Lorsque ce champ est manquant, le caractère de fin de ligne doit être une virgule.

Statut de communication

Le périphérique IHM est un poste serveur dans le réseau TCP Modbus. La mise en œuvre actuelle du protocole ne rapporte pas de code d'erreur de communication autre que les codes standards d'erreur de communication pour le chargement correct du pilote.

Voir la section « Variables système » du manuel principal.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

OPC UA Client

Le pilote de communication OPC UA Client a été conçu pour connecter les périphériques IHM aux serveurs OPC UA.

Cette implantation du protocole s'effectue uniquement en tant que client.

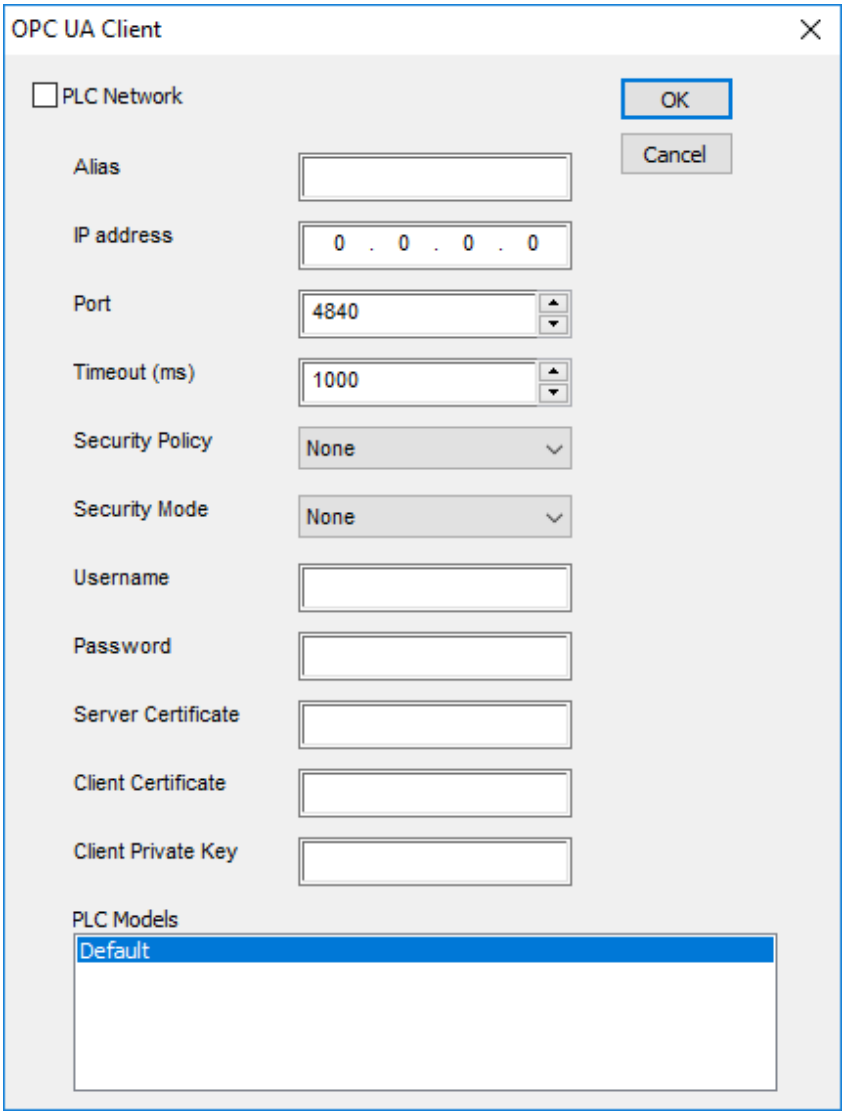
Paramètres de l'éditeur de protocole

Ajouter un protocole

Pour configurer le protocole :

1. Dans le nœud **Config**, double-cliquez sur **Protocoles**.
2. Pour ajouter un pilote, cliquez sur **+** : une nouvelle ligne est ajoutée.
3. Sélectionnez le protocole dans la liste **API**.

La boîte de dialogue de configuration du protocole s'affiche.



Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Élément	Description
Réseau d'API	Active l'accès aux contrôleurs multiples en réseau. Configurez les bonnes options pour chaque contrôleur.
Alias	Nom d'identification des nœuds dans les configurations réseau. Il est ajouté en préfixe à chaque nom de variable importé pour chaque nœud de réseau.
IP Address	Adresse IP du serveur.
Port	Numéro de port d'écoute du serveur.
Délai d'attente (ms)	Délai d'attente (en millisecondes) entre deux tentatives en cas d'absence de réponse du périphérique serveur.
Mode sécurité	Type d'authentification : • **None** : les certificats ne sont pas utilisés • **Sign** : certificats utilisés uniquement pour l'authentification avec le serveur. • **SignAndEncrypt** : certificats utilisés pour l'authentification avec le serveur et le cryptage des données.
Politique de scurité	Niveau de cryptage à utiliser (utilisé seulement quand le mode Sécurité est actif). • Basic256 • Basic256Sha256
Nom d'utilisateur Mot de passe	Authentification par identifiant et mot de passe
Server Certificate	Certificat du serveur OPC UA. Un certificat de serveur peut être téléchargé à l'aide de l'importateur de variables. Voir "Certificat du serveur OPC UA distant" Page 748
Certificat client	Certificat utilisé par le client OPC UA. Si vide, un certificat est généré automatiquement.
Clé privée du client	Clé utilisée par le client OPC UA. Si vide, un certificat est généré automatiquement.
Modèles d'API	Aucune option disponible.

Remarques :

- Avant de choisir les options de sécurité, sachez que tous les modes de sécurité ne sont pas nécessairement pris en charge par le serveur OPC UA. Assurez-vous d'utiliser le mode de sécurité pris en charge.
- Lorsque vous travaillez au sein d'un réseau privé, vous n'avez pas besoin de fournir des certificats de périphériques parce que vous faites confiance aux périphériques utilisés. Sur un réseau public, le certificat vous donnera une garantie de l'identité des périphériques.

Certificat externe

La version ASCII du certificat (généralement un fichier avec l'extension.pem) est nécessaire.

Éditez les fichiers de certificat, puis copiez et collez le texte intégral de votre certificat dans les champs du certificat.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Étape 1 : supprimer l'en-tête et le pied de la page

```
-----BEGIN CERTIFICATE-----
MIIDNjCCAh4CCQCJtJgjqDDUqjANBgkqhkiG9w0BAQsFADBdMQswCQYDVQQGEwJJ
VDEPMA0GA1UEBwwGVmVyb25hMRQwEgYDVQQKDAtDb21wYW55TmFtZTERMA8GA1UE
CwwIUiZEIFRlYW0xFDASBgNVBAMMC0hNSURldmljZUlEMB4XDTE4MDMyNjA5MTAz
OFoXDTI4MDMyMzA5MTAzOFowXTELMAkGA1UEBhMCSVQxDzANBgNVBAcMBlZlcm9u
YTEUMBIGA1UECgwLQ29tcGFueU5hbWUxETAPBgNVBAsMCFImRCBUZWFtMRQwEgYD
VQQDDAtITUlEZXZpY2VJRDCCASIwDQYJKoZIhvcNAQEBBQADggEPADCCAQoCggEB
ALONtzGwlrGv6cXH8i7sNWbwmx9Xo4tp20khnt/VJnDLoYHv7ZvV1vQYHom3/HiC
IaWV/uUvYnXaNBlxHnPsQPV0bEEg26Np01ne8jXEHY6bcMVK3XBV3eno3adOwHA5
vio0MmF6fPQVWTfyVb4/MrcfqUke1gWk3sFlFxEtxXlRLOwNK1+G7Wbnb3Oj4oPL
Ev60VN3DwisDzvivpW7Nv4RPjNK9XJ2DVI+/+KDCNNLlP8GpD0xB1iIpj1S8BwqZ
om1+SUs10IM1cfv/AfArZj9QaIo3c2uPwkLncqQxfDvmlC1fCfsRVxm5N3bmimwC
2F6hbkZksLp7ovCx/haKhfkCAwEAATANBgkqhkiG9w0BAQsFAAOCAQEALVjkNEa/
4OJnMZIVkSZZWGylHHGZ8rphcUPH4olbq7MkaHk7mKacYKqI/qorrIPhmKf7Y2x5
UcTN4Uff6NT0xjrMUg2Q6Lp+a/fBqOUvEebrtmd8NYbhjTs4iVYg3R/NBlgrfx9N
6IppO6OJoOhYXjwDZU0HADnSXVABeBxzAESvLVK7mxgXypdB1D+kgcC6hL9Xv4u5
melNI24LNkRiBT35Exlo2YTu4I9YHFelc5iILvC6DpUYHeSlIEKiNmccL2DDGEBZ
TscRZykvWRi1Xpm2WMzjbf9HE0XNRM8DTCkOscxcrYZrcTVpm0a0WH5OD2531LnF
XsH5sLPyOxtKFw==
-----END CERTIFICATE-----
```

Étape 2: supprimer tous les caractères sur nouvelle ligne

```
MIIDNjCCAh4CCQCJtJgjqDDUqjANBgkqhkiG9w0BAQsFADBdMQswCQYDVQQGEwJJVDEPMA0GA1.....
```

Étape 3: copier et coller la ligne de texte seule du certificat vers le dialogue du protocole

Script pour générer un certificat

Si vous voulez utiliser votre propre certificat, notez que le certificat doit inclure le paramètre nom alternatif « Subject Alternative Name » (SAN) comme l'exige la norme sur les serveurs à architecture unifiée OPC.

Voici un exemple de la façon de générer des fchiers de certificat à l'aide d'une bibliothèque publique OpenSSL-Win32 (Référence : https://www.openssl.org/)

```
@echo off
set OpenSSL="C:\Program Files (x86)\OpenSSL-Win32\bin\openssl.exe"
set NodeName=HMI-Client

rem Generate an RSA key
    %OpenSSL% genrsa -out client-key.pem 2048

rem Creating Certificate Signing Requests
    %OpenSSL% req -new -key client-key.pem -out client.csr -subj
"/ST=Italy/C=IT/L=Verona/O=CompanyName/OU=R&D Team/CN=OPCUAClient@%NodeName%

rem Creating Certificate (.pem)
    echo subjectAltName=URI:urn:%NodeName%:CompanyName:OPCUAClient > san.txt
    echo
keyUsage=digitalSignature,nonRepudiation,keyEncipherment,dataEncipherment,keyCertSign
>> san.txt
    echo extendedKeyUsage=critical,serverAuth,clientAuth >> san.txt
```

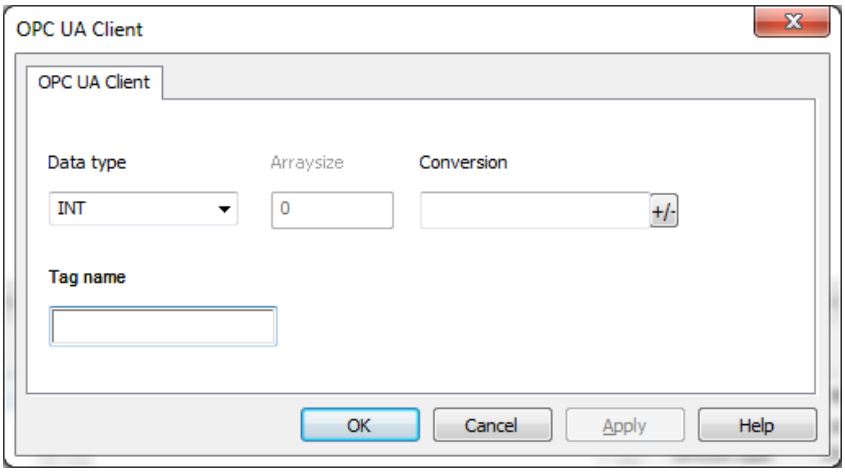
Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

```
    echo authorityKeyIdentifier=keyid,issuer >> san.txt
    echo basicConstraints=CA:TRUE >> san.txt
    %OpenSSL% x509 -req -days 3650 -in client.csr -signkey client-key.pem -out
client.crt -extfile san.txt

rem    Convert Certificate (.der)
    %OpenSSL% x509 -in client.crt -outform der -out client.der

rem Not necessary files
    del san.txt

pause
```

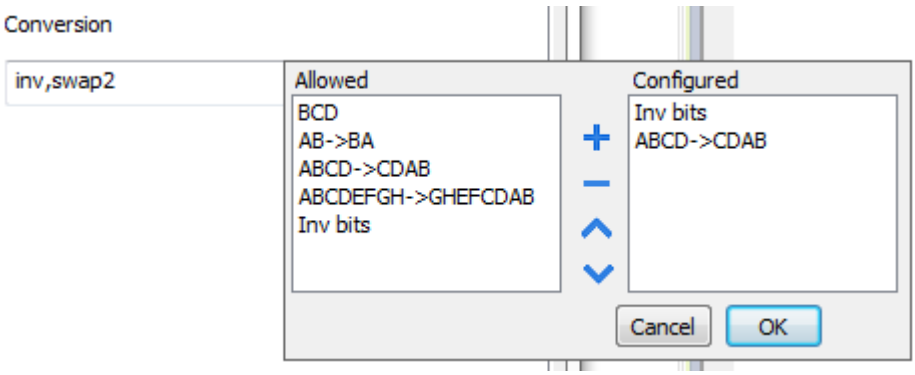
Paramètres de l'éditeur de variables

Chemin : ***VueProjet*** *>* ***Configuration*** *> double-cliquez sur* ***Variables***

1. Dans la liste des protocoles, sélectionnez **OPC UA Client**.
2. Pour ajouter une variable, cliquez sur **+** : la boîte de dialogue de définition de variable s'affiche.



Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Élément	Description
Type de données	Types de données disponibles : • **booléen** • **octet** • **court** • **entier** • **octet non signé** • **court non signé** • **entier non signé** • **flottant** • **double** • **heure** • **uint64** • **int64** • **chaîne** • **binaire** Consultez la section « Concepts de programmation » du manuel principal. Remarque : pour définir des tableaux, sélectionnez l'un des formats de type de données suivi de crochets.
Taille de tableau	• En cas de variable vectorielle, cette propriété représente le nombre d'éléments vectoriels. • En cas de variable de chaîne, cette propriété représente le nombre maximum d'octets disponible pour la variable de chaîne. Remarque : le nombre d'octets correspond au nombre de caractères de chaîne si la propriété Encoding est définie sur UTF-8 ou Latin1 dans l'Éditeur de variables. Si la propriété Encoding est définie sur UCS-2BE, UCS-2LE, UTF-16BE ou UTF-16LE, un caractère nécessite 2 octets.
Conversion	Conversion à appliquer à la variable.  En fonction du type de données choisi, la liste **Autorisée** affiche un ou plusieurs types de

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

<table>
<tr><th>Élément</th><th>Description</th></tr>
<tr><td></td><td>conversion.
<table>
<tr><th>Valeur</th><th>Description</th></tr>
<tr><td>Inv bits</td><td>inv : inverse tous les bits de la variable.

Exemple :
1001 → 0110 (format binaire)
9 → 6 (format décimal)</td></tr>
<tr><td>Inverser</td><td>neg : inverse la valeur de la variable.

Exemple :
25.36 → -25.36</td></tr>
<tr><td>AB -> BA</td><td>swapnibbles : échange les quartets en un octet.

Exemple :
15D4 → 514D (format hexadécimal)
5588 → 20813 (format décimal)</td></tr>
<tr><td>ABCD -> CDAB</td><td>swap2 : échange les octets en un mot.

Exemple :
9ACC → CC9A (format hexadécimal)
39628 → 52378 (format décimal)</td></tr>
<tr><td>ABCDEFGH -> GHEFCDAB</td><td>swap4 : échange les octets en un mot double.

Exemple :
32FCFF54 → 54FFFC32 (format hexadécimal)
855441236 → 1426062386 (format décimal)</td></tr>
<tr><td>ABC...NOP -> OPM...DAB</td><td>swap8 : échange les octets en un mot long.

Exemple :
142,366 → -893553517,588905 (format décimal)
0 10000000110
00011100101110110110010001011010000111001010110000001
→
1 10000011100
1010101000010100010110110110110010110110000100111101
(format binaire)</td></tr>
<tr><td>BCD</td><td>bcd : sépare l’octet en deux quartets, les lit au format décimal (de 0 à 9)

Exemple :
23 → 17 (format décimal)
0001 0111 = 23
0001 = 1 (premier quartet)
0111 = 7 (second quartet)</td></tr>
</table></td></tr>
</table>

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Élément	Description
	Sélectionnez la conversion et cliquez sur le bouton +. La conversion sélectionnée s'affiche dans la liste **Configurée**. Si vous configurez d'autres conversions, celles-ci s'affichent dans l'ordre (du haut vers le bas de la liste **Configurée**). Utilisez les boutons de direction pour changer l'ordre des conversions configurées.
Nom de variable	Nom de la variable à utiliser dans la communication.

Remarque : les propriétés des variables résultent du processus d'importation. Dans la plupart des cas, la création manuelle de nouvelles variables n'est pas nécessaire.

IP d'écrasement de nœud

Le protocole propose le type de données spécial IP d'écrasement de nœud, qui vous permet de modifier l'adresse IP du contrôleur cible au moment du runtime.

Ce type de mémoire est un tableau de 4 octets non signés, un pour chaque octet de l'adresse IP.

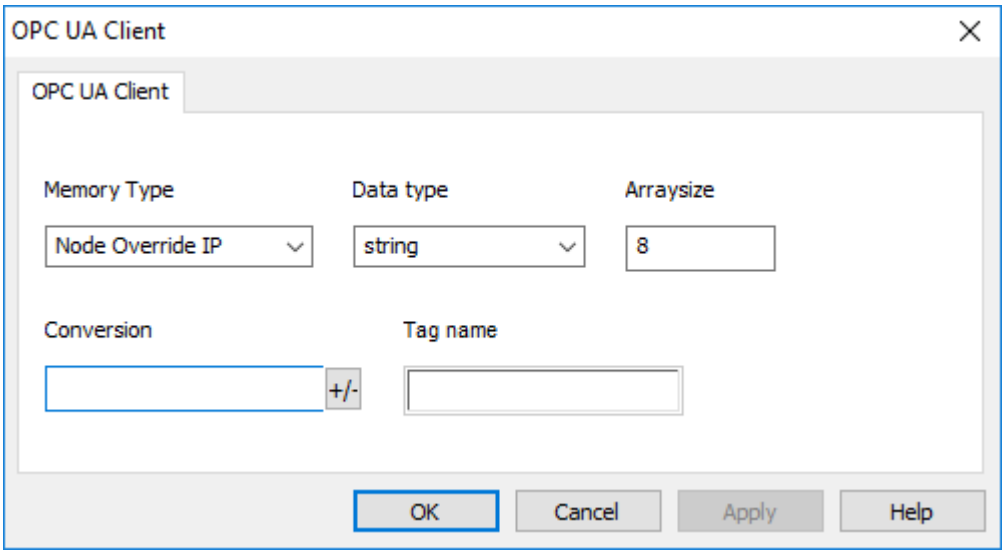
L'IP d'écrasement de nœud est initialisée avec la valeur de l'IP du contrôleur spécifiée dans le projet au moment de la programmation.

IP d'écrasement de nœud	Exécution de l'API
0.0.0.0	La communication avec le contrôleur est coupée, plus aucune trame de requête ne sera générée.
Autre que 0.0.0.0	Interprétée comme étant l'écrasement de l'IP de nœud, et remplace l'adresse IP cible au moment du runtime.

Lorsque le périphérique IHM est connecté à un réseau avec plus d'un nœud de contrôleur, chaque nœud dispose de sa propre variable d'IP d'écrasement de nœud.

Remarque : le maintien des valeurs de l'IP d'écrasement de nœud assignées au runtime est assuré par des cycles d'alimentation.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



Port d'écrasement de nœud

Le protocole propose le type de données spécial port d'écrasement de nœud, qui vous permet de modifier le port réseau du contrôleur cible au moment du runtime.

Ce type de mémoire correspond à un short non signé.

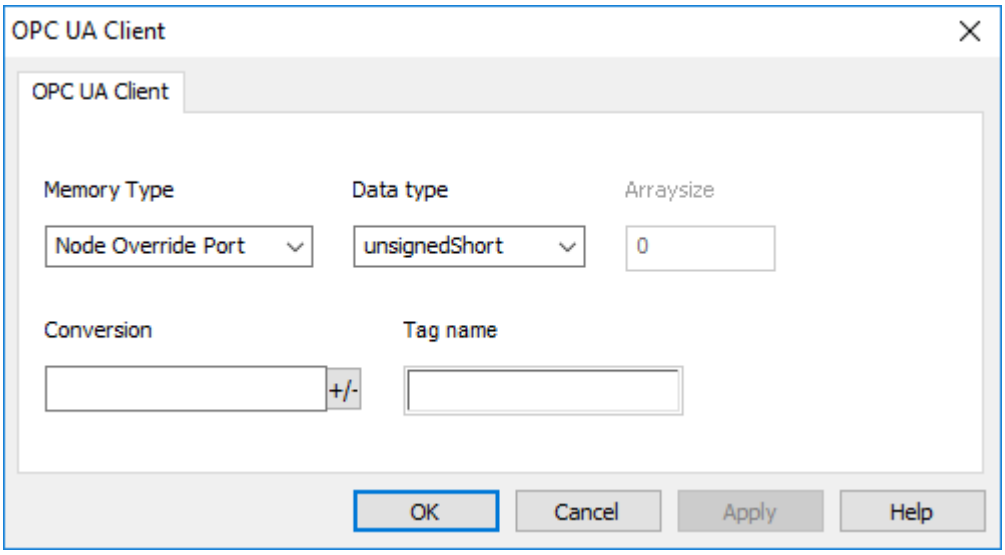
Le port d'écrasement de nœud est initialisé avec la valeur du port du contrôleur spécifiée dans le projet au moment de la programmation.

Port d'écrasement de nœud	Opération Modbus
0	La communication avec le contrôleur est coupée, plus aucune trame de requête ne sera générée.
Autre que 0	Interprété comme étant la valeur du nouveau port et remplacé pour l'opération de runtime.

Lorsque le périphérique IHM est connecté à un réseau avec plus d'un nœud de contrôleur, chaque nœud dispose de sa propre variable de port d'écrasement de nœud.

Remarque : le maintien des valeurs de port d'écrasement de nœud assignées au runtime est assuré par des cycles d'alimentation.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



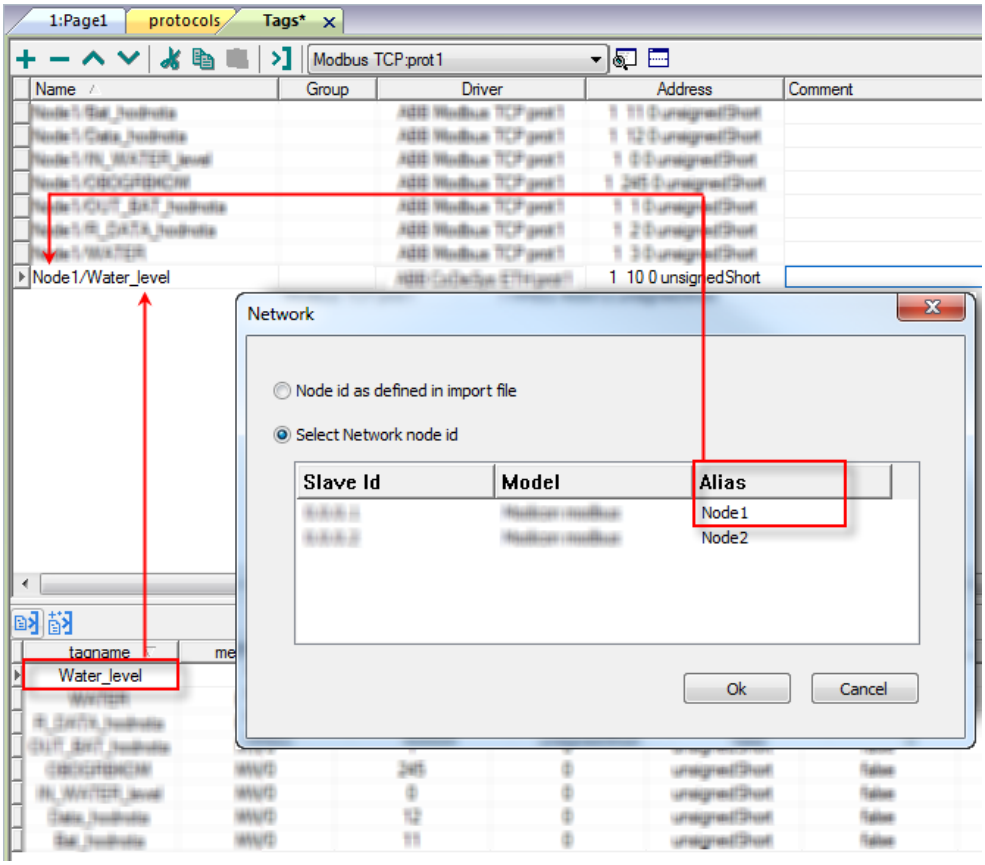
Ajouter un nom d’alias à un protocole

Les noms de variables doivent être uniques au niveau d’un projet. Vous pouvez cependant être amené à utiliser les mêmes noms de variables pour différents nœuds de contrôleur (par exemple lorsque le périphérique IHM est connecté à deux périphériques exécutant la même application).

Lorsque vous générez un protocole, vous pouvez ajouter un nom d’alias qui s’ajoutera aux noms de variables importés pour ce protocole.

Dans l’exemple, la connexion à un certain contrôleur porte le nom **Nœud1**. Lorsque vous importez les variables pour ce nœud, tous les noms de variables porteront le préfixe **Nœud1**, les rendant ainsi uniques au niveau projet/réseau.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



Remarque : vous pouvez donner un nom d'alias uniquement aux variables importées. Il n'est pas possible pour les variables ajoutées manuellement dans l'éditeur de variables de porter un préfixe d'alias dans leur nom.
La chaîne d'alias est associée à l'importation des variables. Une fois l'importation des variables effectuée, une modification de la chaîne d'alias n'aura aucun effet sur les noms figurant déjà dans le dictionnaire. Lorsque vous modifiez la chaîne d'alias et réimportez les variables, toutes les variables réimportées porteront la nouvelle chaîne de préfixe.

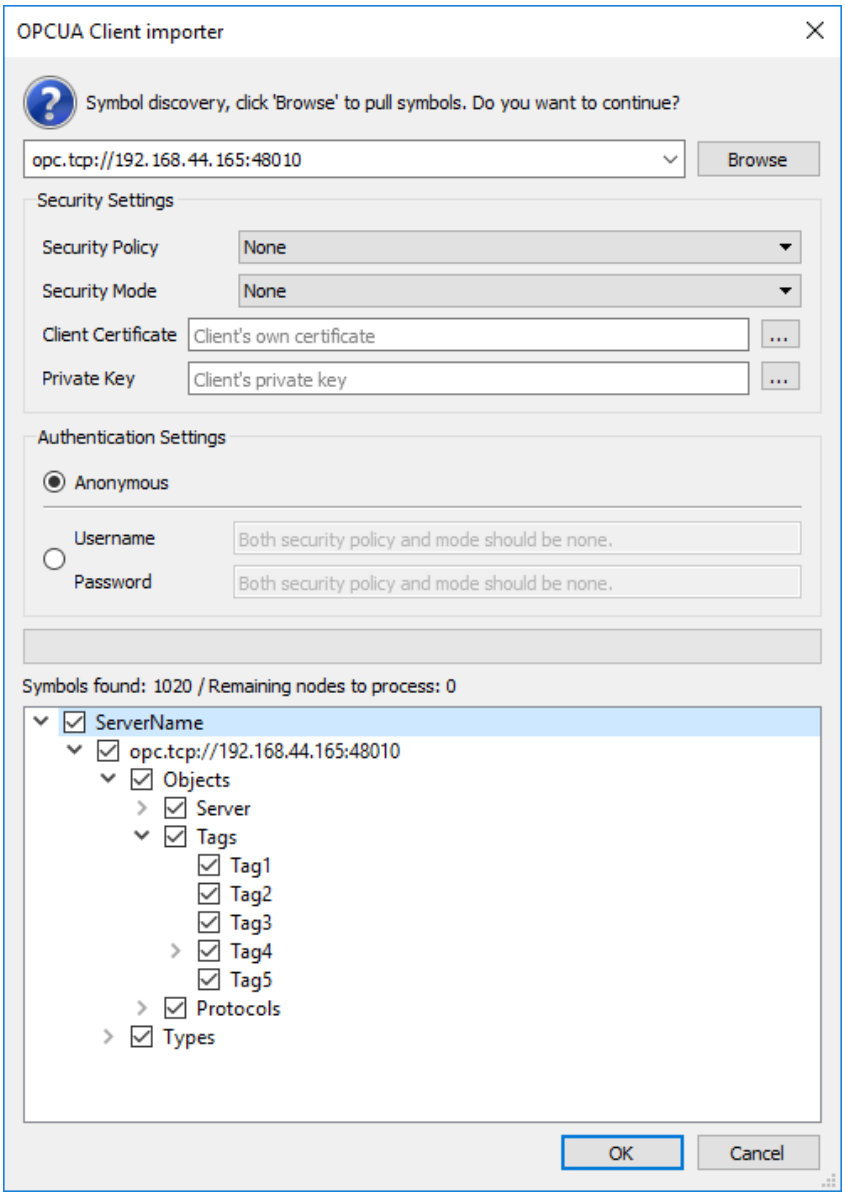
Importer des variables

Les variables du protocole client OPC UA doivent être importées des serveurs OPC UA.

*Chemin : **VueProjet**> **Configuration** > double-cliquez sur **Variables***

1. Sélectionnez **OPC UA Client** dans la liste des protocoles disponibles.
2. Cliquez sur **Importer variables.**
3. Sélectionnez **Pilote d'importation hiérarchique.**
4. Entrez l'adresse IP du serveur.
5. Choisissez les modes sécurité et authentification.
6. Cliquez sur **Browse** pour se connecter et récupérer le dictionnaire des variables du serveur OPC UA.
7. Le serveur OPC UA fournira son propre certificat. Vous devez accepter le certificat pour continuer et récupérer les données.
8. Une fois le processus de détection terminé, cliquez sur **OK** pour créer le dictionnaire de variables.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



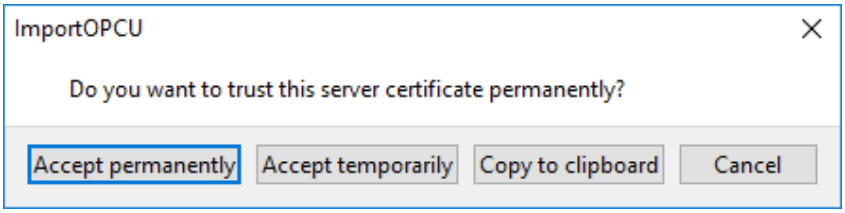
Élément	Description
Remote URI	Adresse su serveur OPC UA sous la forme : *opc:tcp:<IPAddress>:<Port>* Exemple : • opc.tcp://192.168.44.165:4840
Mode sécurité	Type d'authentification : • **None** : aucune authentification avec le serveur et aucun cryptage de données. • **Sign** : certificats utilisés uniquement pour l'authentification avec le serveur. • **SignAndEncrypt** : certificats utilisés pour l'authentification avec le serveur et le cryptage des données.
Politique de scurité	Niveau de cryptage à utiliser (utilisé seulement quand le mode Sécurité est actif).

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Élément	Description
	• Basic128Rsa15 • Basic256 • Basic256Sha256
Nom d'utilisateur **Mot de passe**	Authentification par identifiant et mot de passe
Certificat client	Certificat utilisé par le client OPC UA. Si vide, un certificat est généré automatiquement. Le certificat n'est utilisé par l'importateur que si le serveur le demande.
Clé privée du client	Clé utilisée par le client OPC UA. Si vide, un certificat est généré automatiquement.

Pour être autorisé à récupérer les données du serveur OPC UA, vous devez fournir les paramètres de sécurité requis. Le dialogue se remplira automatiquement avec les paramètres fournis par les paramètres de l'éditeur de protocole (vous pouvez simplement accepter les valeurs proposées)

Certificat du serveur OPC UA distant



Lorsque le serveur OPC UA fournit son propre certificat, vous avez les choix de :

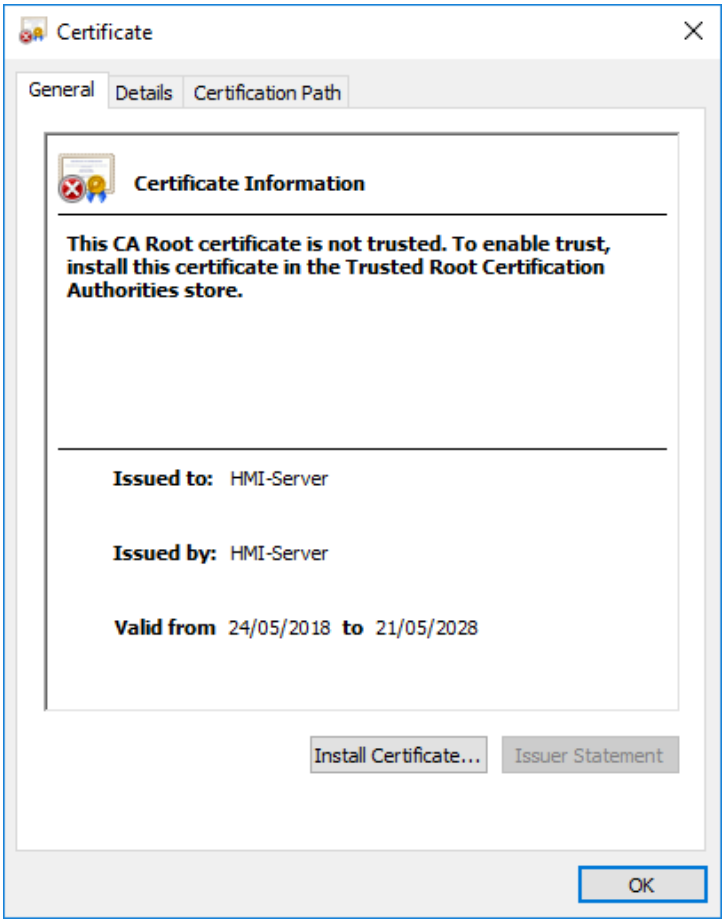
Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

- **Accepter temporairement**
  Le certificat n'est accepté que pour la session de travail en cours.
- **Accepter en permanence**
  Le certificat est accepté et copié sur l'ordinateur. Toute demande d'importation future pour le même serveur OPC UA sera automatiquement acceptée sans demande de confirmation.

  Le fichier de certificat sera copié dans le dossier :
  %AppData%\Roaming\ABB\studio\OPCUA\pki\trusted\certs

- **Copier dans le presse-papiers**
  Le format ASCII du certificat est copié dans le presse-papiers pour vous permettre de vérifier son authenticité, de l'enregistrer et de l'insérer dans la configuration du protocole (si nécessaire).

  Pour vérifier un certificat, utilisez un éditeur de texte pour le coller du presse-papiers dans un fichier texte avec l'extension.crt. Vous pouvez ensuite double-cliquer sur le fichier.crt pour permettre à Windows d'affcher les propriétés du certificat.



- **Cancel**
  Annuler l'importation

Statut de communication

Vous pouvez afficher le statut de communication actuel grâce aux variables système. Voir la section « Variables système » du manuel principal.

Codes pris en charge pour ce pilote de communication :

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Erreur	Description
Connecting <Error description>	Erreur durant la connexion
Connection while reading: <Error description>	Erreur rencontrée lors d'une connexion pour une opération de lecture
Bad status while reading: <Error description>	Erreur dans l'opération de lecture
Connection while writing: <Error description>	Erreur rencontrée lors d'une connexion pour une opération d'écriture
Bad status writing: <Error description>	Erreur dans l'opération d'écriture
Client OPC UA introuvable pour un ID de nœud donné	Mauvaise information d'ID de nœud

<Error description> peut être l'une des erreurs suivantes :

Erreur	Notes
BadTimeout	Erreur délai d'attente dépassé. Aucune réponse du serveur.
BadSecurityChecksFailed	Erreur durant l'échange de certificats. Se produit généralement lorsque le serveur ne considère pas le certificat client comme étant fiable.
BadCertificatexxxInvalid	Erreur dans le certificat client ou serveur.
BadNodeUnknown	La variable (nœud) n'existe pas.
BadAttributeNotFound	Tentative d'accès à un attribut non valide.
BadNotWritable	Tentative d'écriture dans un attribut en lecture seule.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Simatic S7 ETH

Le pilote de communication Simatic S7 ETH a été conçu pour communiquer avec les contrôleurs Simatic via une connexion Ethernet.

Le contrôleur Simatic doit être équipé soit d'un port Ethernet intégré, soit d'une interface Ethernet appropriée (intégrée ou avec un module).

La communication se base sur les fonctions PG/OP (protocole ISO-on-TCP).

Le présent document décrit les paramètres de pilote à appliquer dans le logiciel de programmation IDE et API S7.

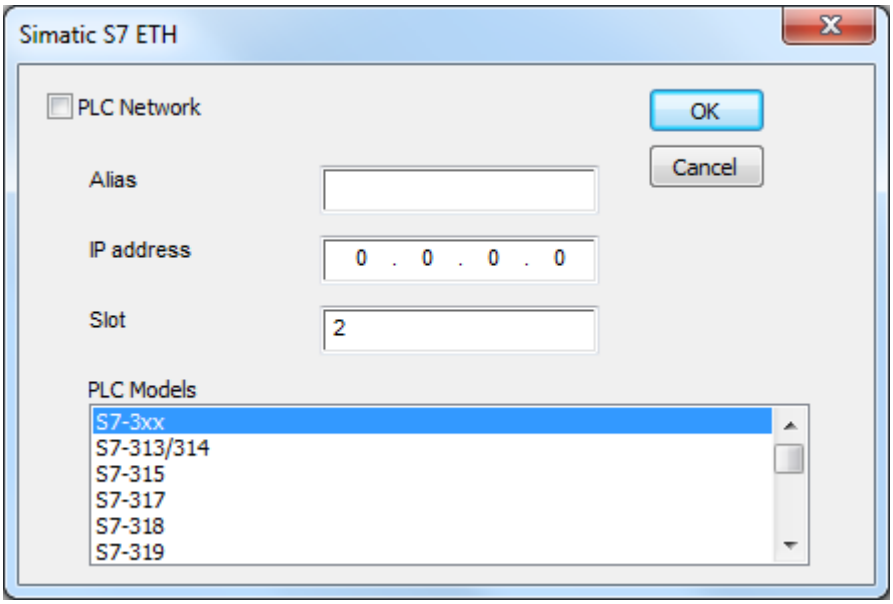
Paramètres de l'éditeur de protocole

Ajouter un protocole

Pour configurer le protocole :

1. Dans le nœud **Config**, double-cliquez sur **Protocoles**.
2. Pour ajouter un pilote, cliquez sur **+** : une nouvelle ligne est ajoutée.
3. Sélectionnez le protocole dans la liste **API**.

La boîte de dialogue de configuration du protocole s'affiche.



Élément	Description
Alias	Nom d'identification des nœuds dans les configurations réseau. Il est ajouté en préfixe à chaque nom de variable importé pour chaque nœud de réseau.
Adresse IP	Adresse IP Ethernet du contrôleur.
Fente	2Numéro du créneau sur lequel est monté le processeur. 2 pour S7-300. Cette valeur peut être plus grande pour des systèmes S7-400.

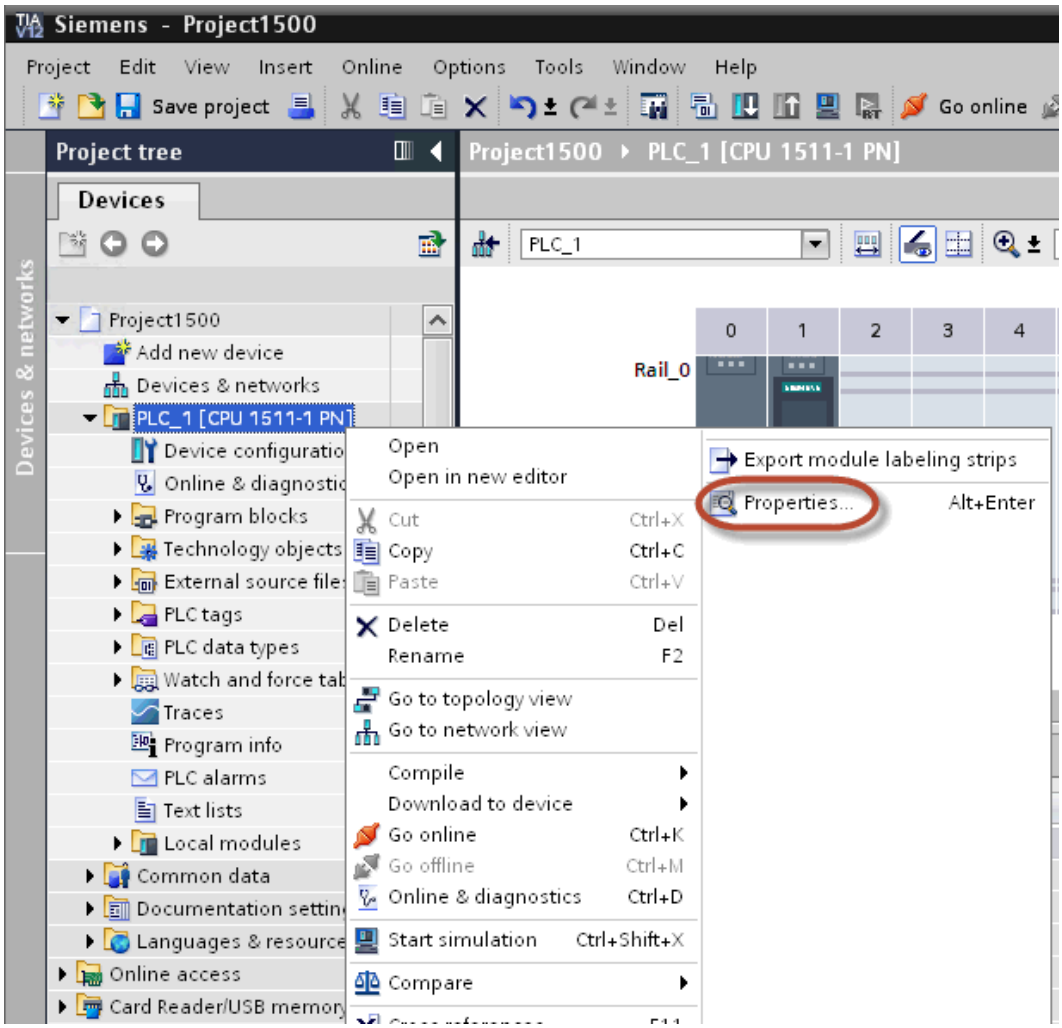
Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Élément	Description
Modèles d'API	Liste des modèles de contrôleurs compatibles. Assurez-vous de sélectionner le modèle d'API correct dans cette liste lors de la configuration du protocole.
Réseau d'API	Active l'accès aux contrôleurs multiples en réseau. Configurez la bonne option pour chaque contrôleur (esclave).

Configuration avec les API S7-1200 et S7-1500

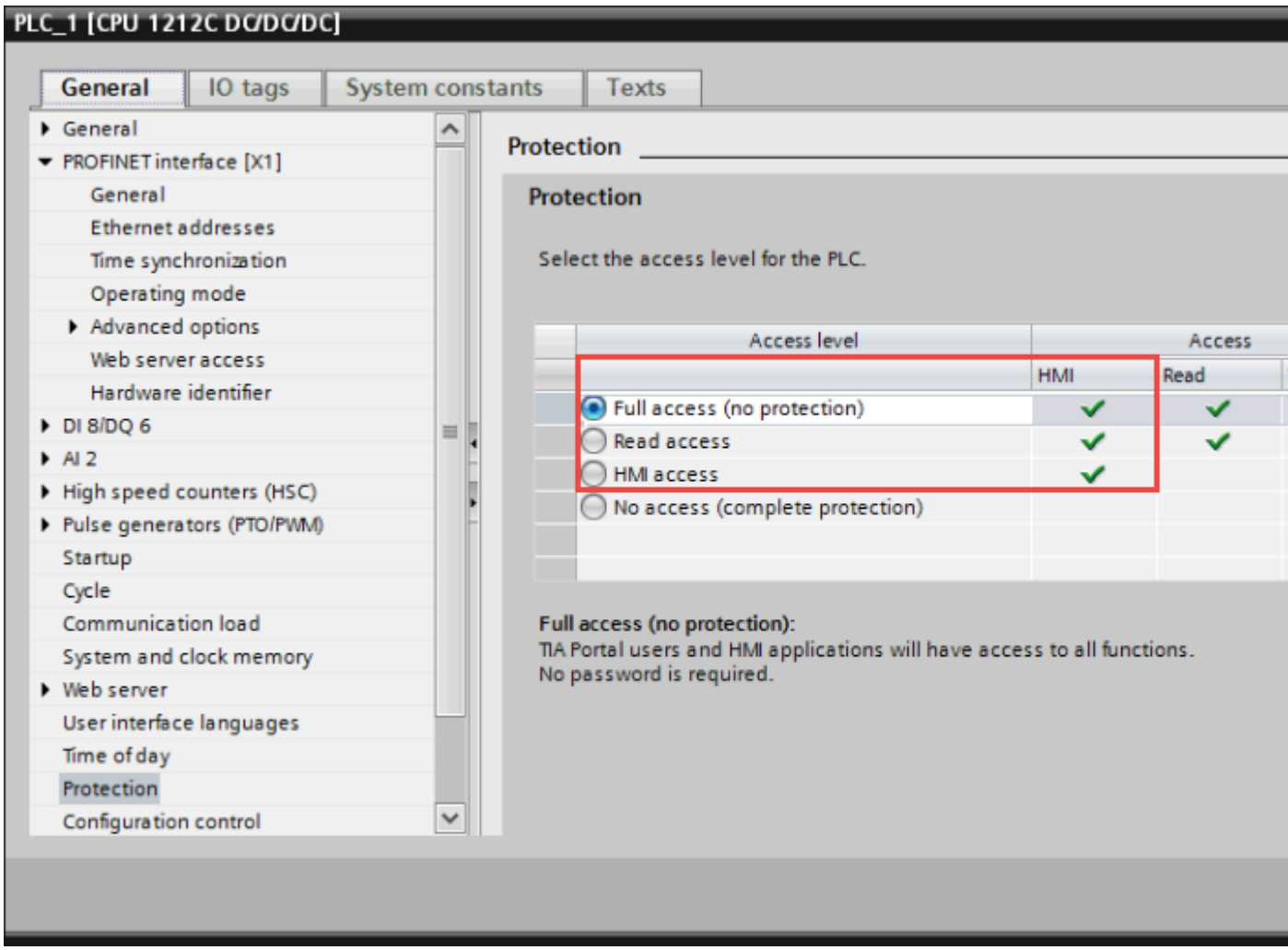
Les séries d'API S7-1200 (à partir de la version du logiciel 4.0) et S7-1500 de Siemens disposent d'un pare-feu intégré. Par défaut, le niveau de protection maximal est activé. Pour établir la communication entre ces modèles d'API, vous devez activer la communication S7 avec des périphériques tiers. Ce paramètre est disponible dans le logiciel de programmation du portail TIA.

1. Ouvrez le projet de l'API dans le portail TIA.
2. Sélectionnez l'API dans l'arborescence du projet et ouvrez les Propriétés de l'API.



Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

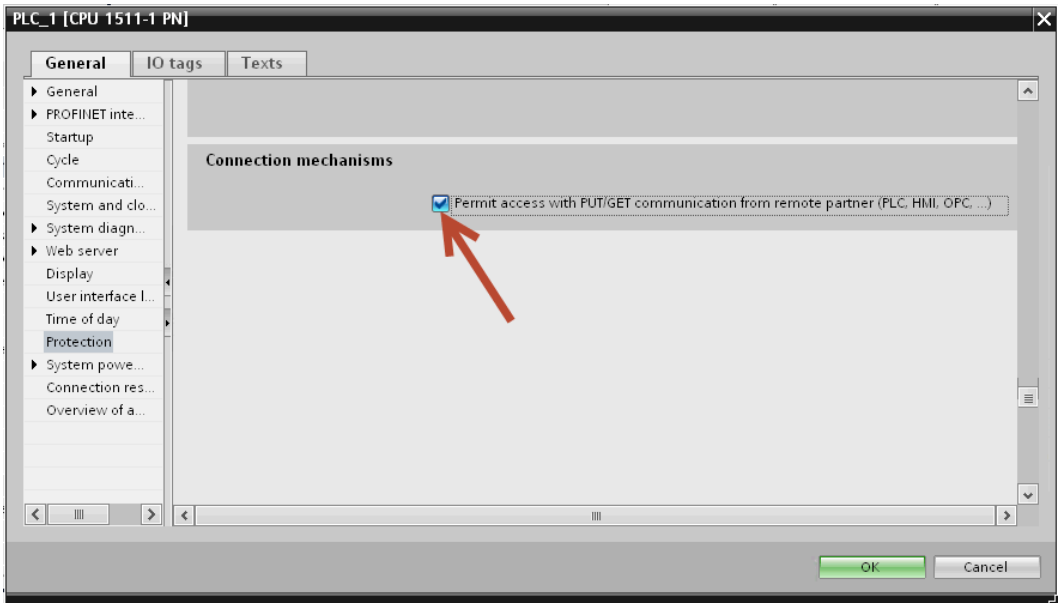
3. En général > La protection choisit une permission entre les trois premiers (assurez-vous que l'encoche soit présente sur la colonne IHM).



Remarque : si « Pas d'accès » est sélectionné, la communication avec le panneau ne sera pas établie.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

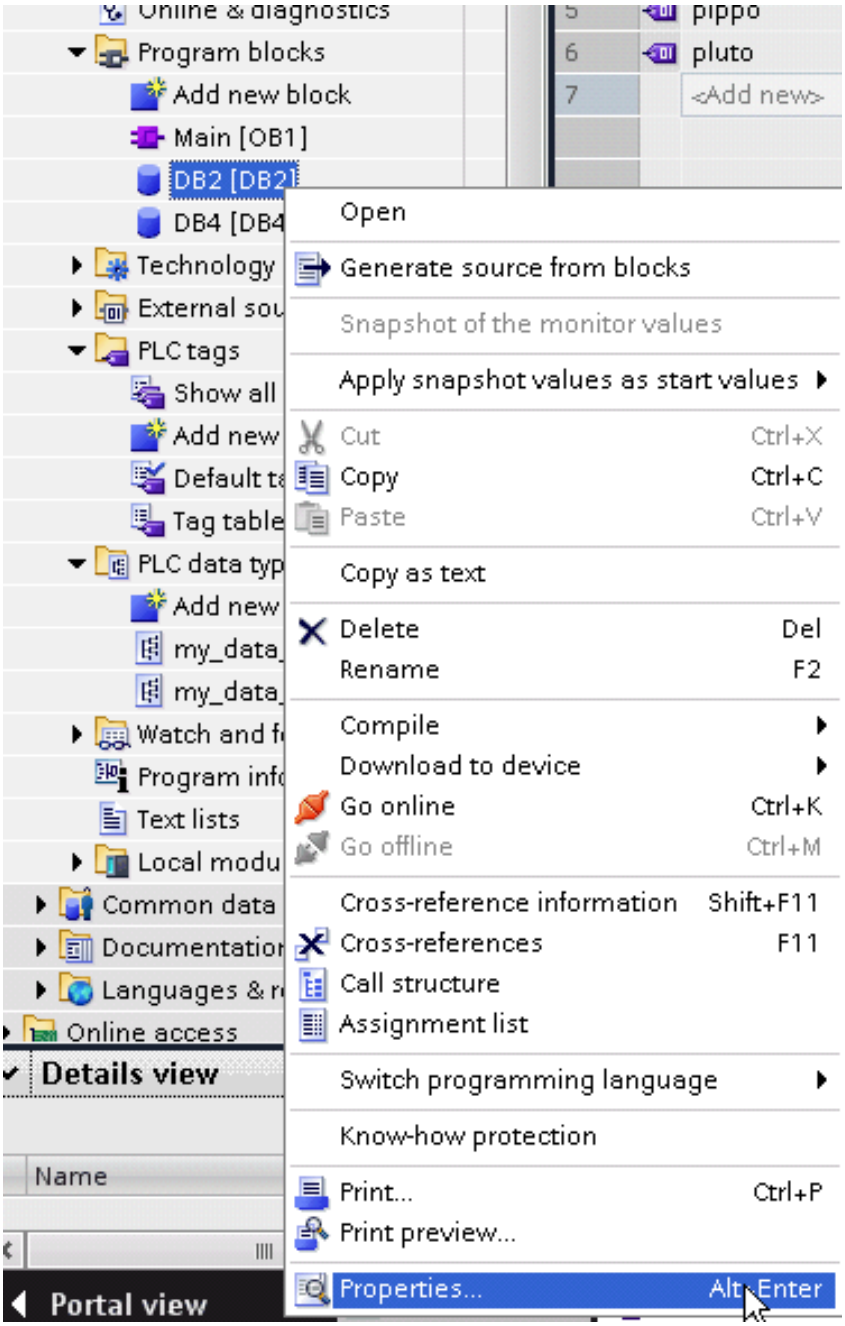
4. Faites défiler la page vers le bas et cochez la case « Autoriser l'accès via la communication PUT/GET par des partenaires distants ».



Remarque : si vous avez défini des variables dans « Blocs de programme », vous devez configurer le DB comme étant « Non optimisé ».

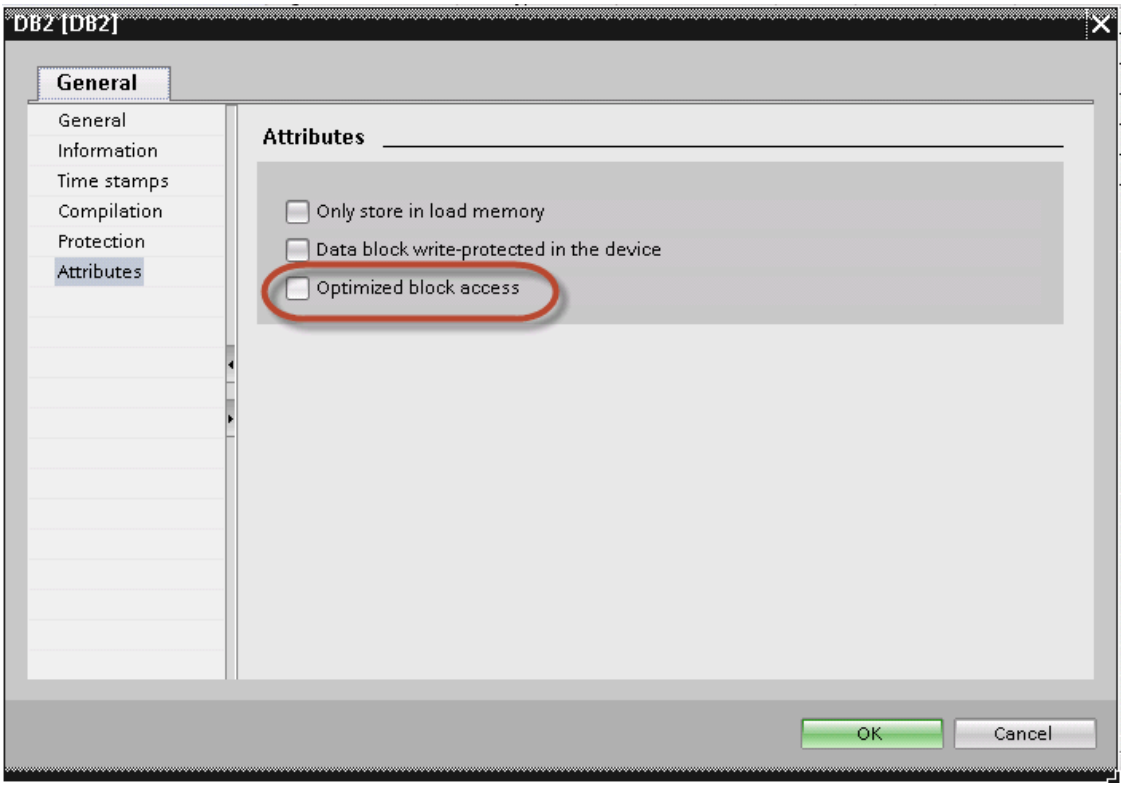
Pour vérifier ou modifier l'optimisation du DB, ouvrez les Propriétés du DB :

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



Dans l'onglet Général > Attributs, décochez la case « Accès au bloc optimisé » :

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



Si la case « Accès au bloc optimisé » n'est pas activée (grisée), cela peut signifier que le DB est un « DB d'instance » relié à un « FB à accès optimisé ».

Une fois le projet compilé, l'offset de variable s'affiche à côté du nom de la variable.

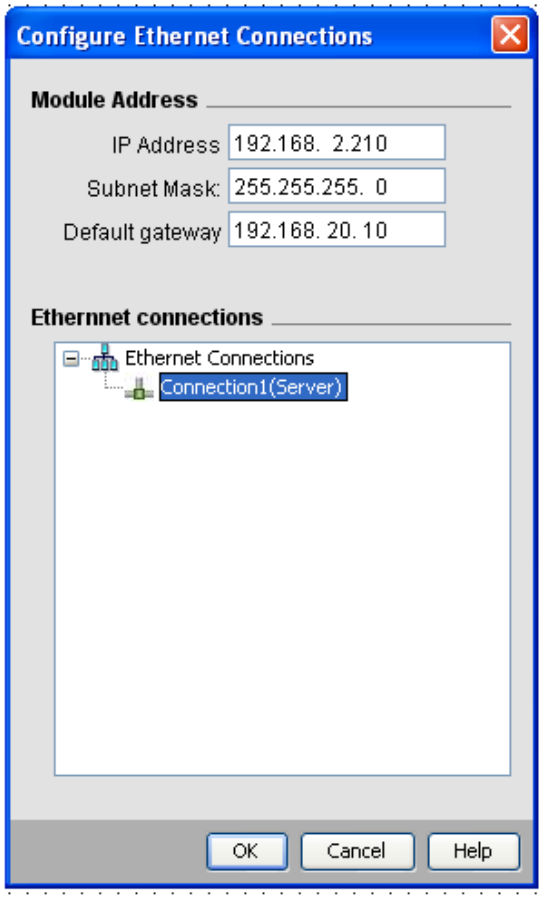
Ces paramètres s'appliquent au logiciel de programmation du portail TIA et aux gammes d'API S7-1200 (à partir de la version du logiciel 4.0) et S7-1500.

Configuration de API Logo!

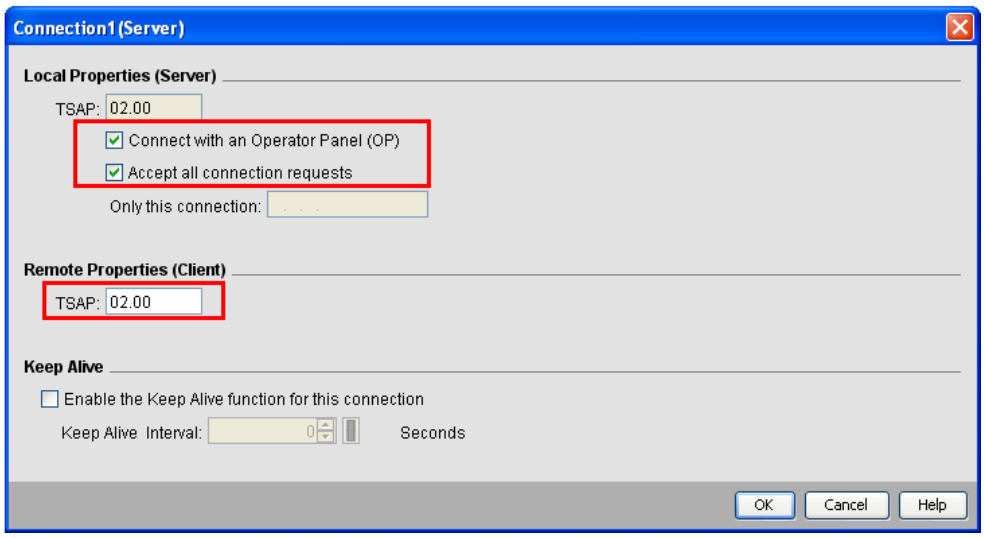
Pour configurer la communication avec API Logo! :

1. Ouvrez le projet Logo!Soft Comfort.
2. Sélectionnez **Outils** > **Connexions Ethernet** : la boîte de dialogue Configurer les connexions Ethernet s'affiche.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



3. Faites un clic droit sur **Connexions Ethernet** et ajoutez une connexion serveur.
4. Double-cliquez sur la nouvelle connexion : la boîte de dialogue des propriétés de cette connexion s'affiche.



5. Sélectionnez les options **Connexion au pupitre opérateur (OP)** et **Accepter toutes les demandes de connexion**.
6. Dans la section **Propriétés distantes (Client)**, configurez le **TSAP** sur 02.00.

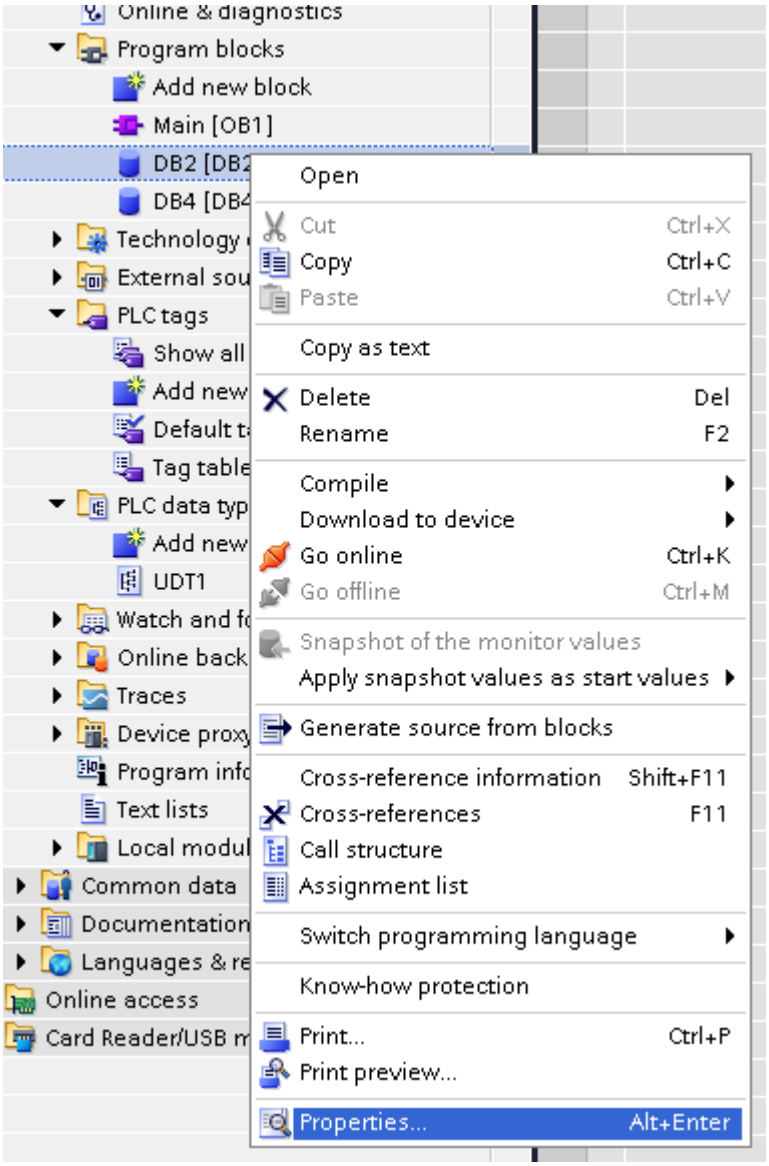
Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Export à l'aide du portail TIA v13, v14 ou plus récente

Exporter des blocs de programme

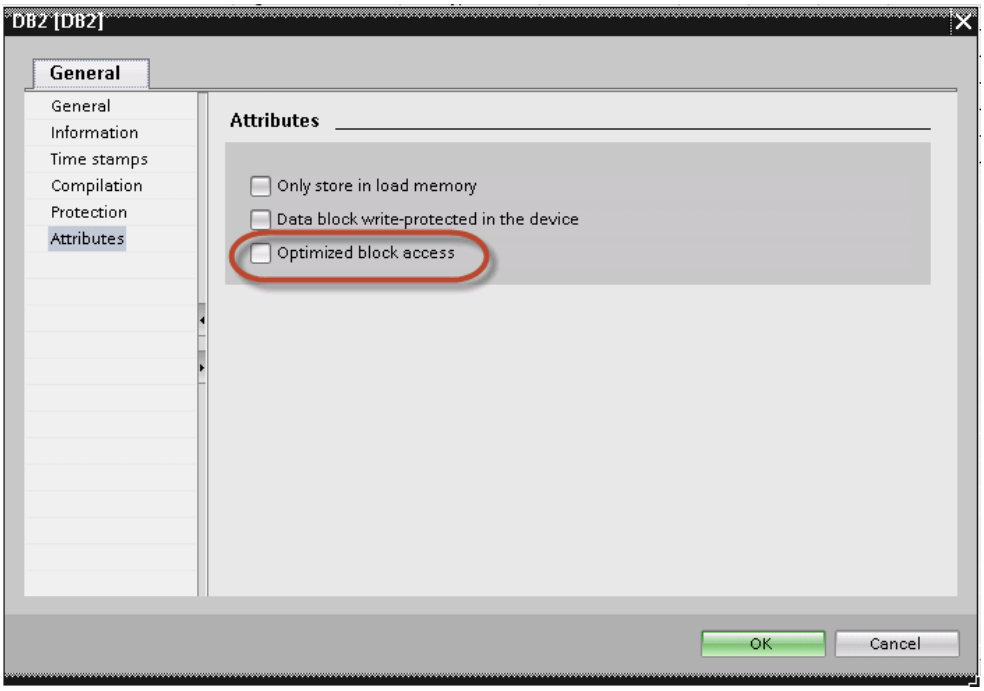
Ces fichiers se rapportent aux variables DB définies dans **Blocs de programme**.

1. Définissez le bloc de données comme étant **Non optimisé**.
2. Faites un clic droit sur le bloc de données et sélectionnez **Propriétés** :



3. Dans l'onglet **Général**, sélectionnez **Attributs** et désélectionnez **Accès au bloc optimisé**.

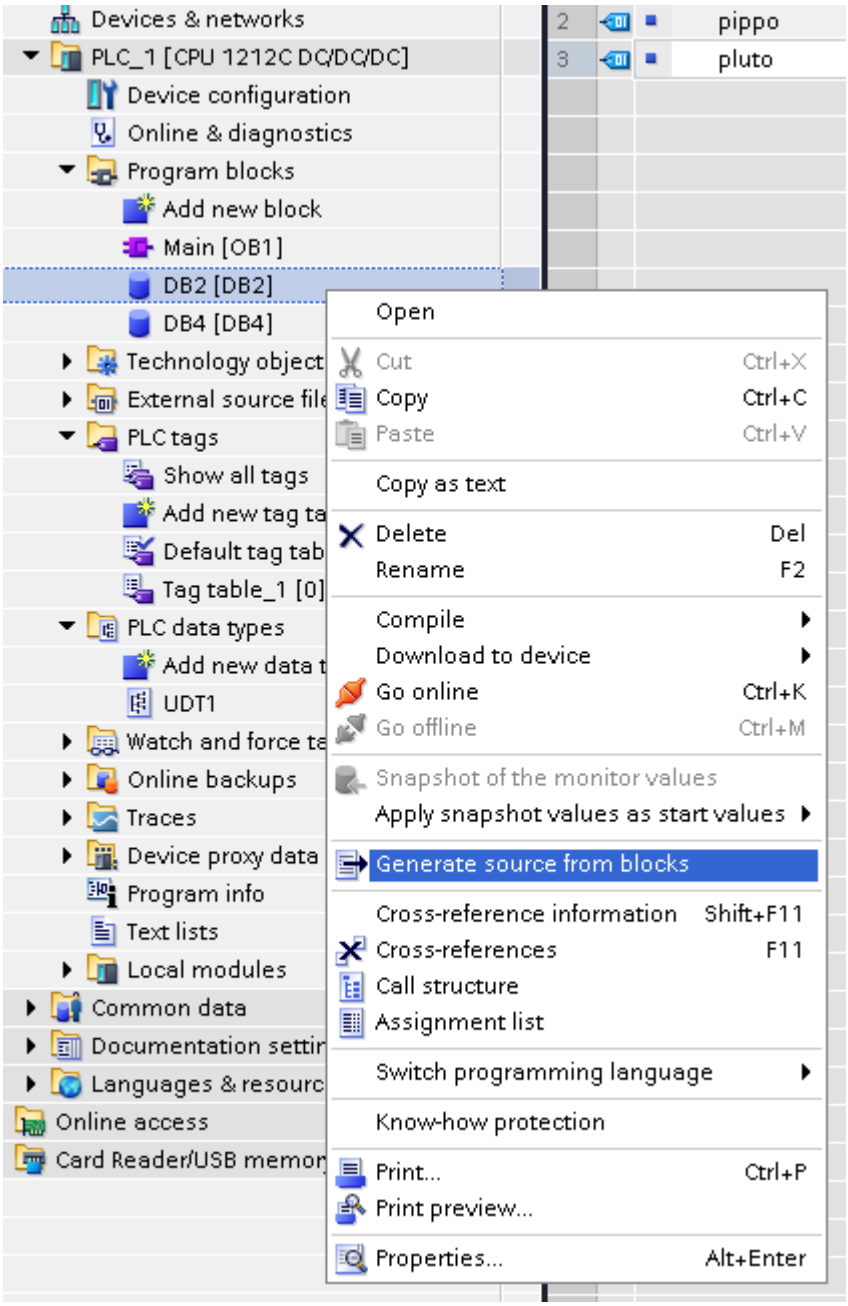
Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



Remarque : si l'option **Accès au bloc optimisé** n'est pas activée (case à cocher grisée), cela peut signifier que le bloc de données est un « DB d'instance » relié à un « FB à accès optimisé ».

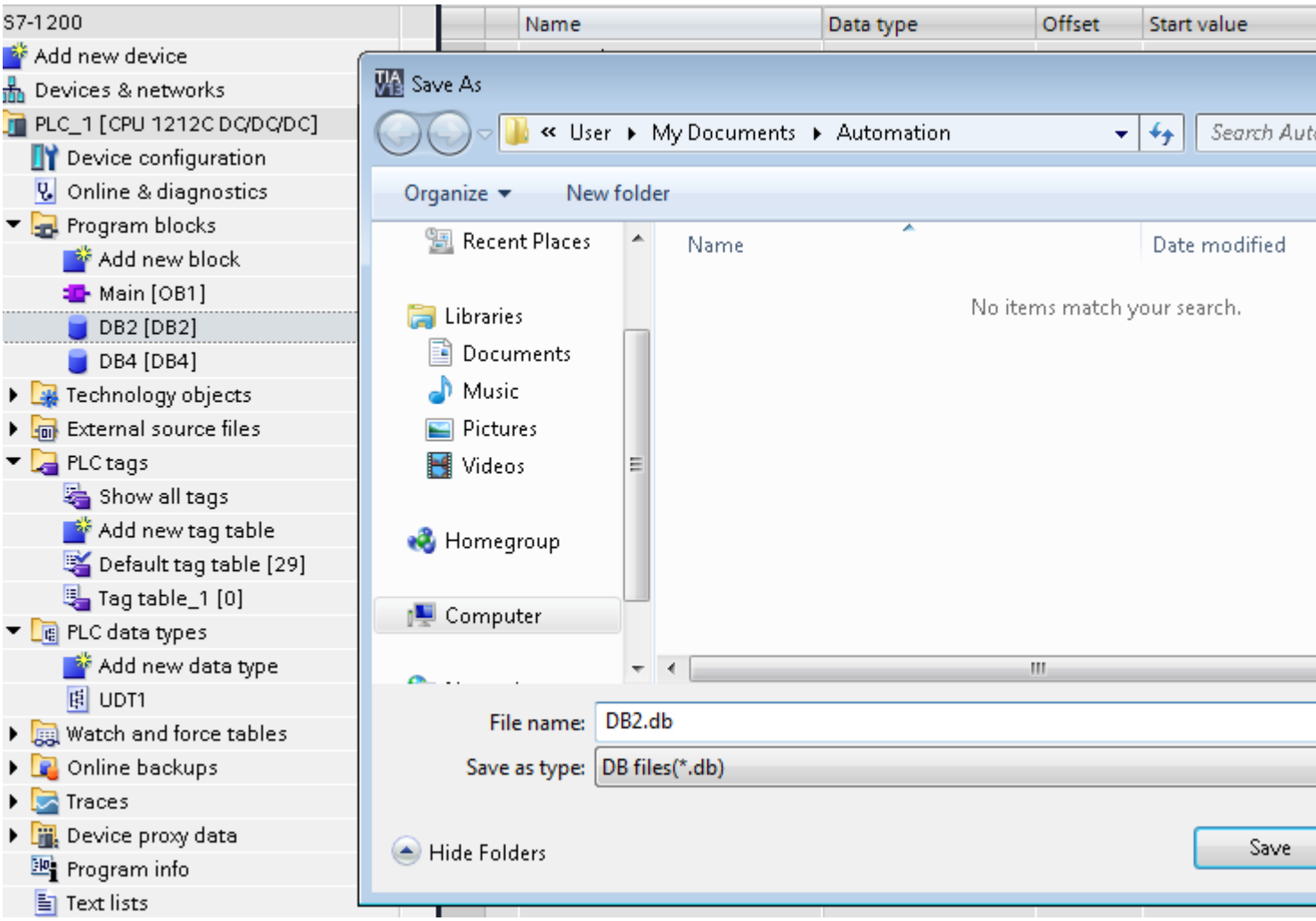
4. Faites un clic droit sur le bloc de données et sélectionnez **Générer les sources des blocs** :

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



5. Sauvegardez le fichier en DBxxx.tia, où xxx correspond au numéro de DB.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

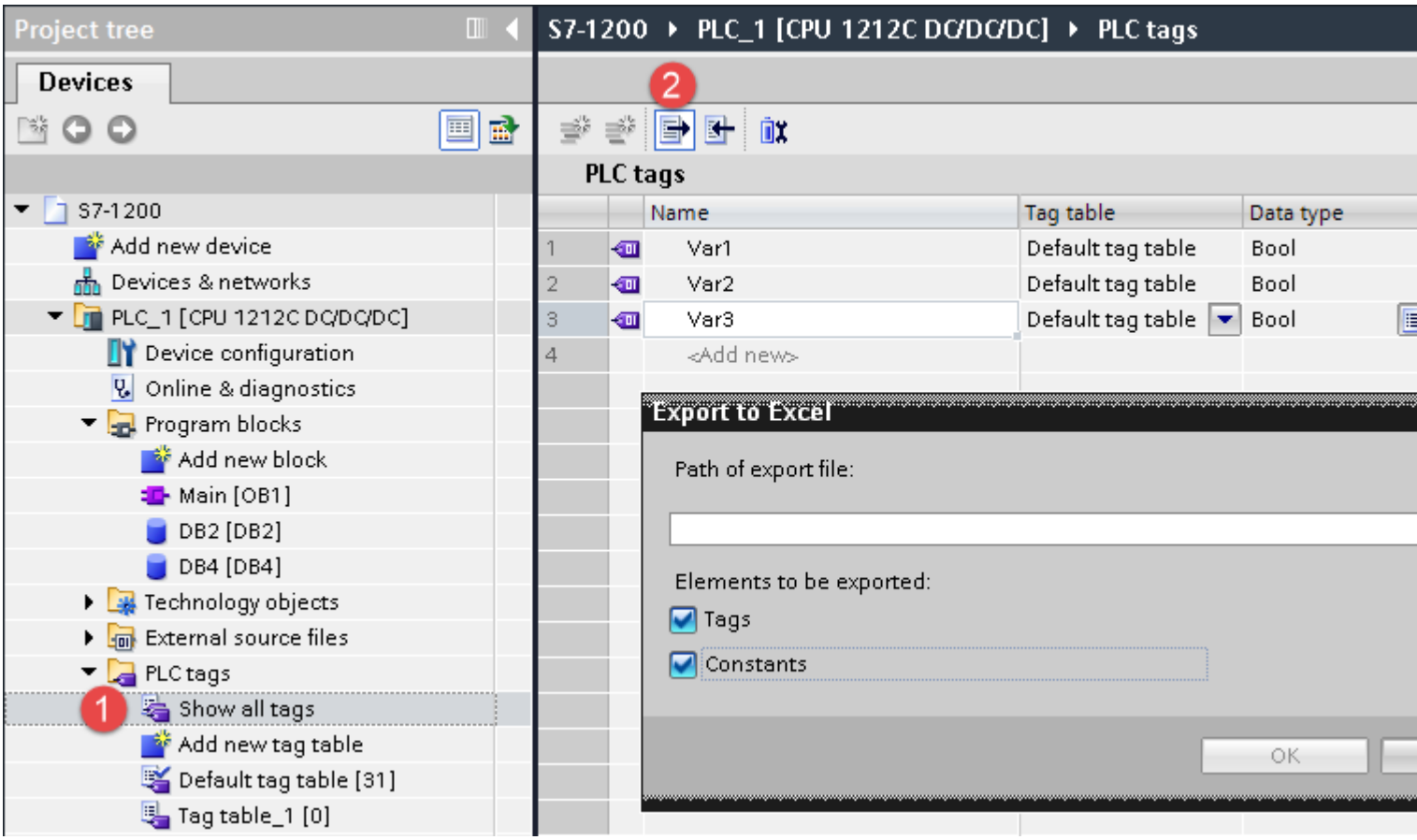


Exporter des variables API

Un fichier Excel fait référence à des variables API.

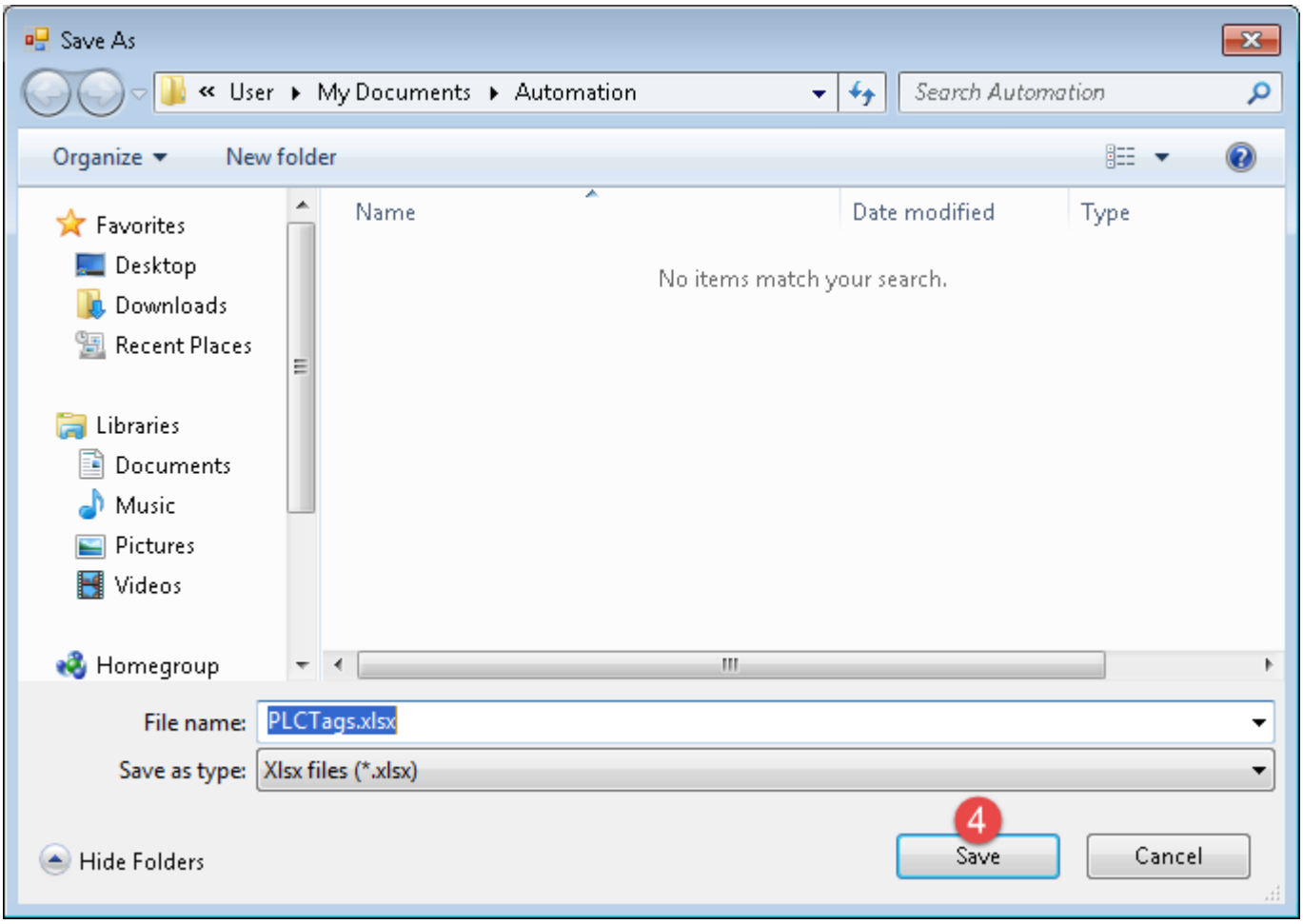
1. Double-cliquez sur **Afficher toutes les variables** : la table des variables s'affiche.
2. Cliquez sur le bouton **Exporter** et indiquez un chemin d'accès.
3. Donner un nom au fichier.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

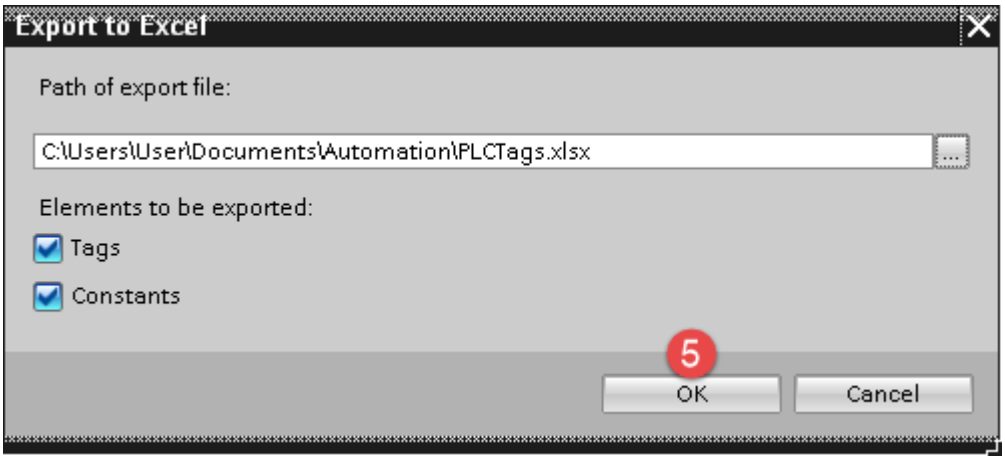


4. Cliquez sur **Enregistrer** pour confirmer.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



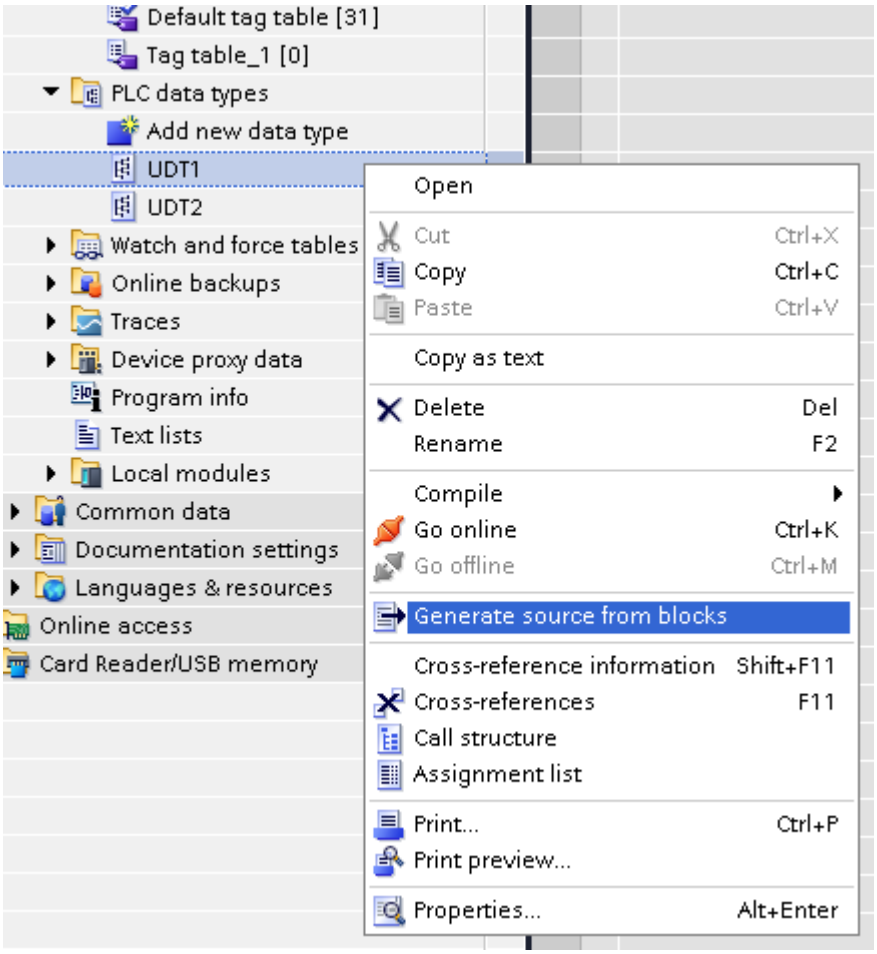
5. Cliquez sur **OK** pour exporter.



Exporter des types de données API

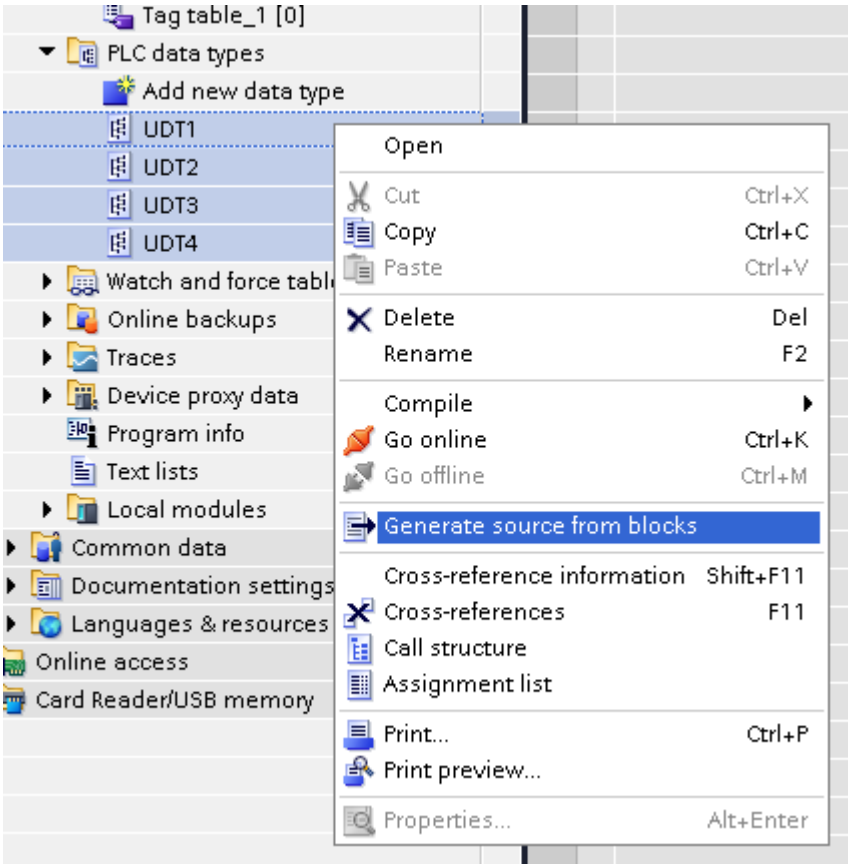
Pour générer le fichier, agrandissez l'élément **Types de données API** de l'arborescence du projet du portail TIA et faites un clic droit sur la structure définie par l'utilisateur. Cliquez ensuite sur **Générer un fichier source à partir des blocs**.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



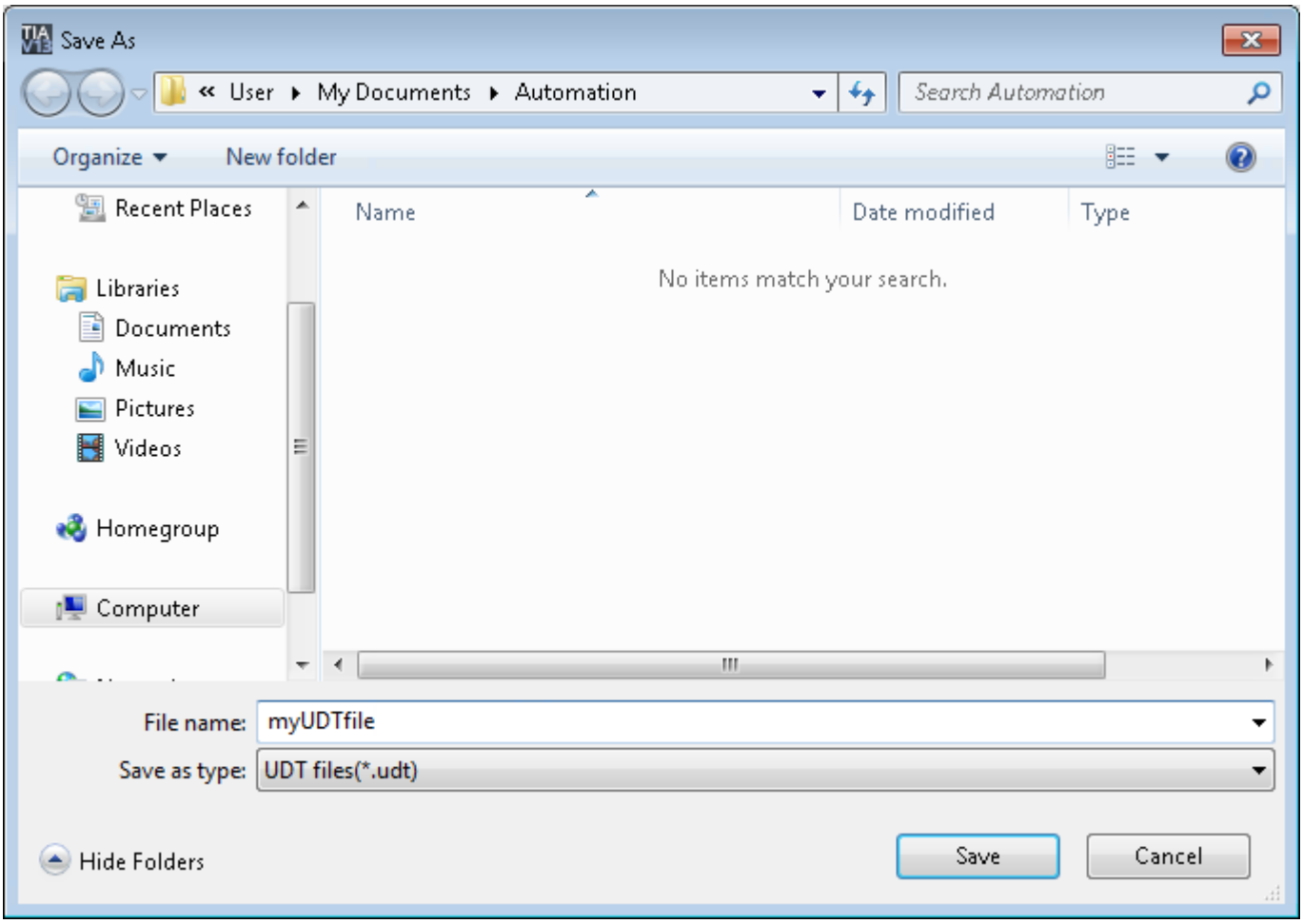
En cas de plusieurs types de données API définis dans le projet API, vous devez tous les sélectionner dans la liste **Types de données API**, faire un clic droit et sélectionner **Générer un fichier source à partir des blocs** pour créer le fichier .scl contenant tous les types de données API définis.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



À l'étape suivante, donnez un nom au fichier .scl et indiquez un chemin d'accès où l'enregistrer.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



Ce fichier contiendra tous les types de données API définis et vous pourrez l'utiliser pour importer des variables dans l'Éditeur de variables.

Consultez le chapitre **Importation de variables** pour plus d'informations.

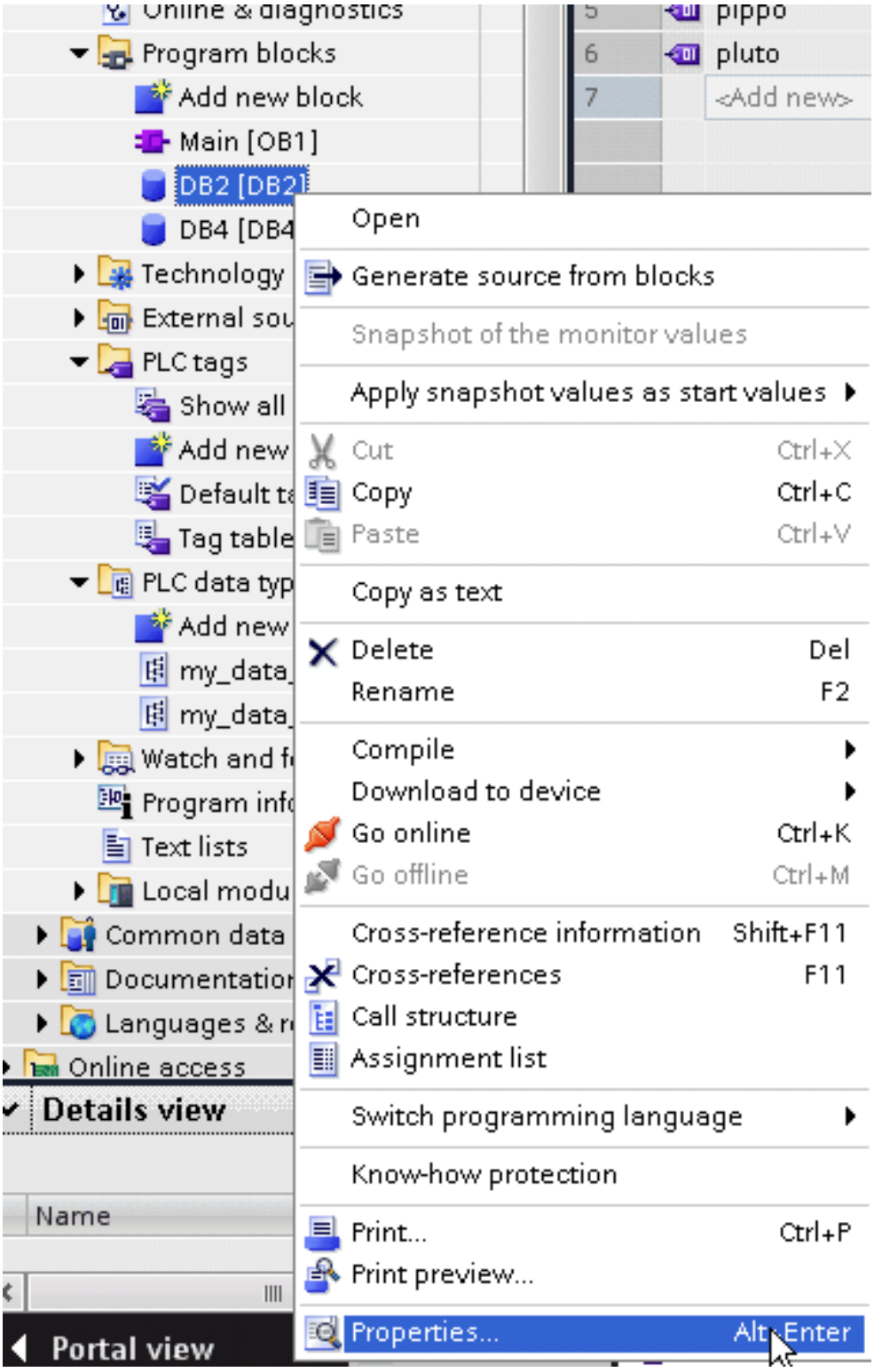
Export à l'aide du portail TIA v10, v11, v12

Exporter des blocs de programme

Ces fichiers se rapportent aux variables DB définies dans **Blocs de programme**.

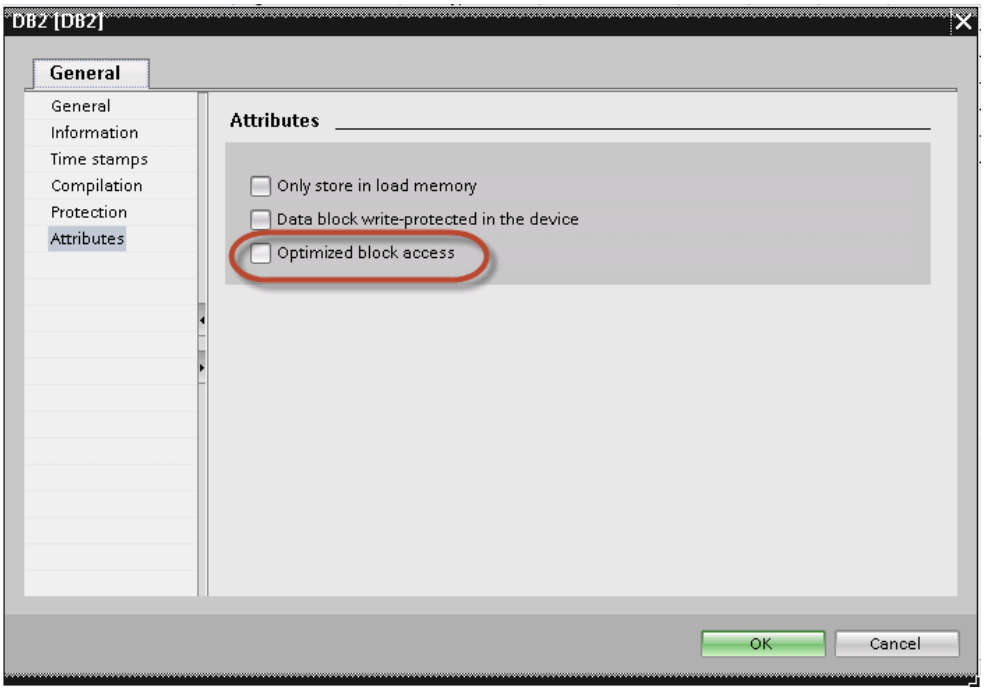
1. Définissez le bloc de données comme étant **Non optimisé**.
2. Faites un clic droit sur le bloc de données et sélectionnez **Propriétés** :

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

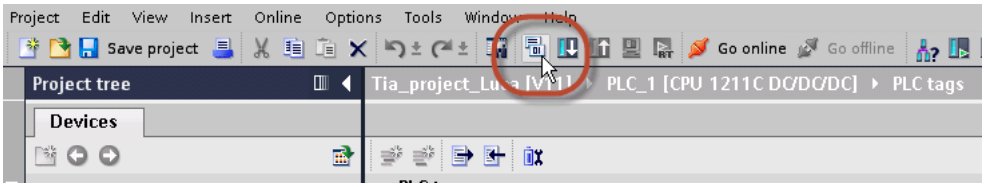


3. Dans l'onglet **Général**, sélectionnez **Attributs** et désélectionnez **Accès au bloc optimisé**.

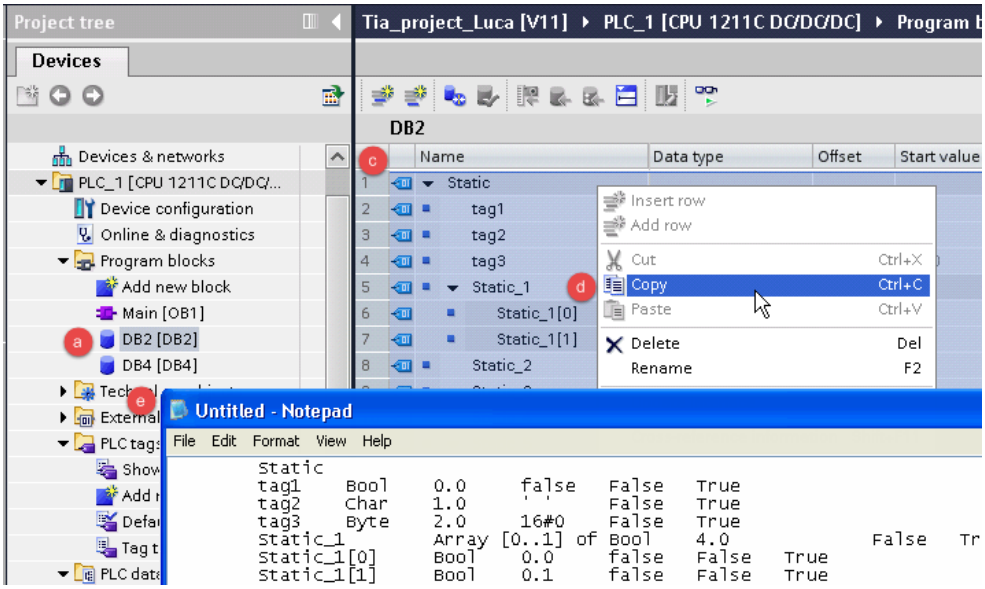
Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



Remarque : si l'option **Accès au bloc optimisé** n'est pas activée (case à cocher grisée), cela peut signifier que le bloc de données est un « DB d'instance » relié à un « FB à accès optimisé ».



4. Construisez le projet afin de vous assurer que le portail TIA calcule l'offset des variables.



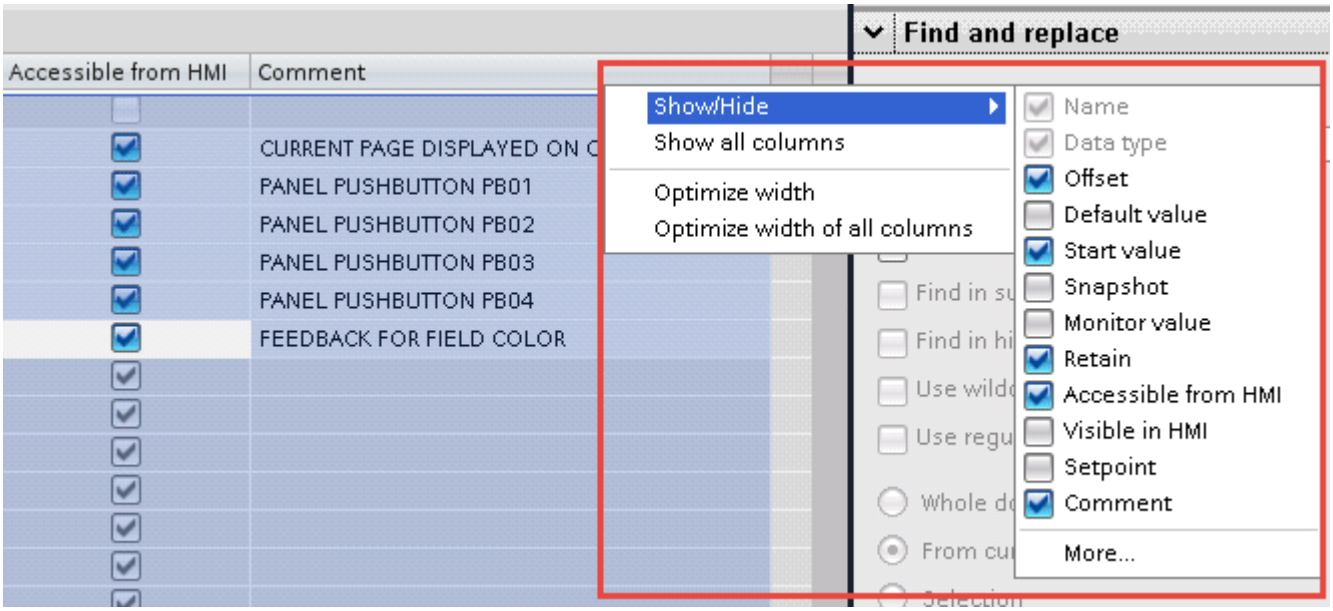
Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

5. Double-cliquez sur le nom d'un DB.
6. Agrandissez la zone de visualisation du bloc de programme sélectionné.
7. Sélectionnez toutes les lignes.
8. Faites un copier-coller dans n'importe quel éditeur de texte.
9. Enregistrer le fichier sous DBxxx.tia, où xxx correspond au numéro du DB.

Remarque : assurez-vous d'utiliser la fonction **Enregistrer sous**, sinon le fichier sera nommé DB2.tia.txt et ne sera pas visible depuis le pilote d'importation.

10. Répétez à partir de l'étape 5 pour tous les blocs de programme.

Remarque : assurez-vous que seules les colonnes suivantes sont affichées dans l'éditeur DB avant de copier toutes les données dans le fichier txt

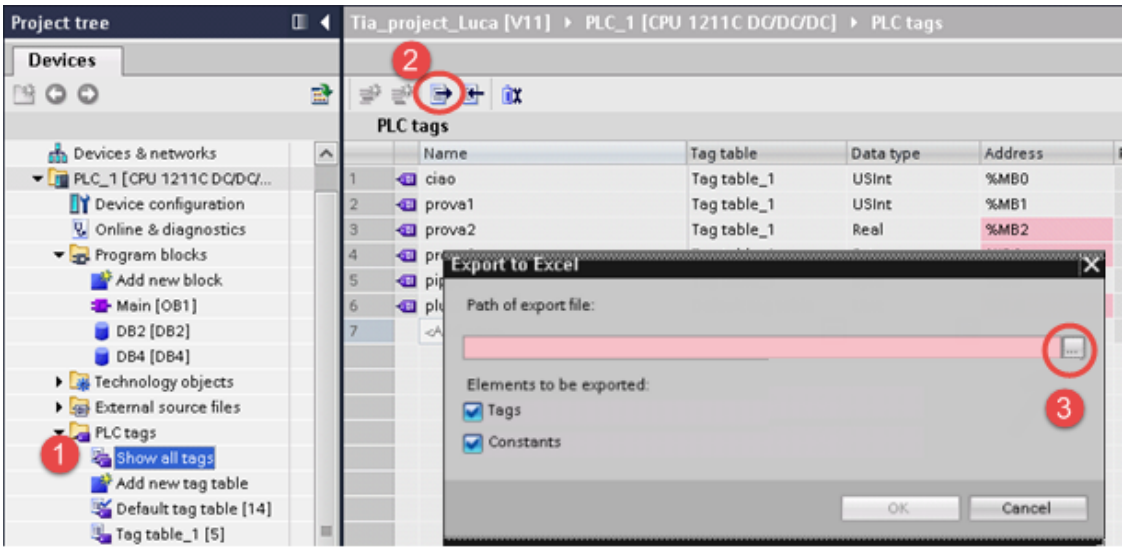


Exporter des variables API

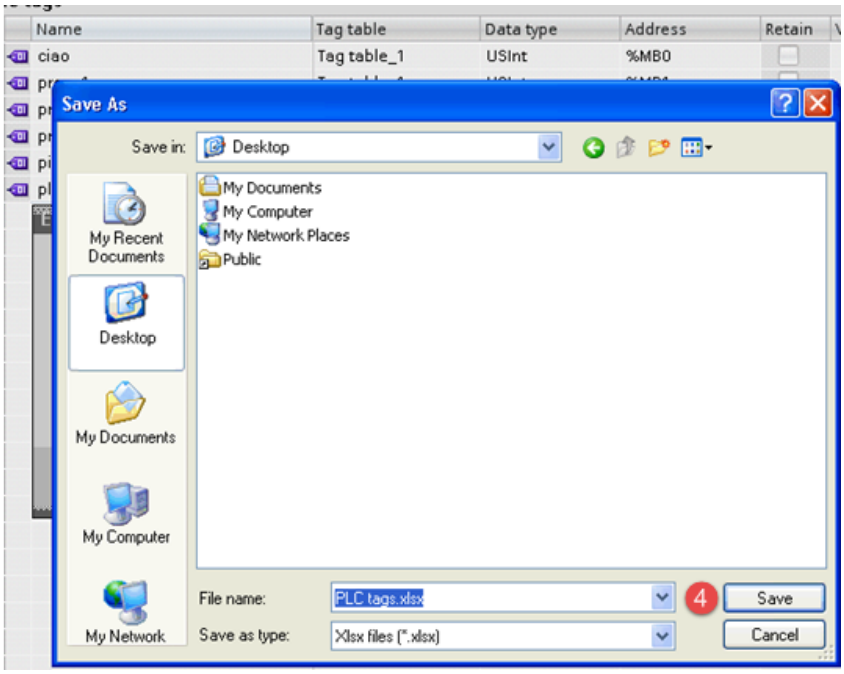
Un fichier Excel fait référence à des variables API.

1. Double-cliquez sur **Afficher toutes les variables** : la table des variables s'affiche.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

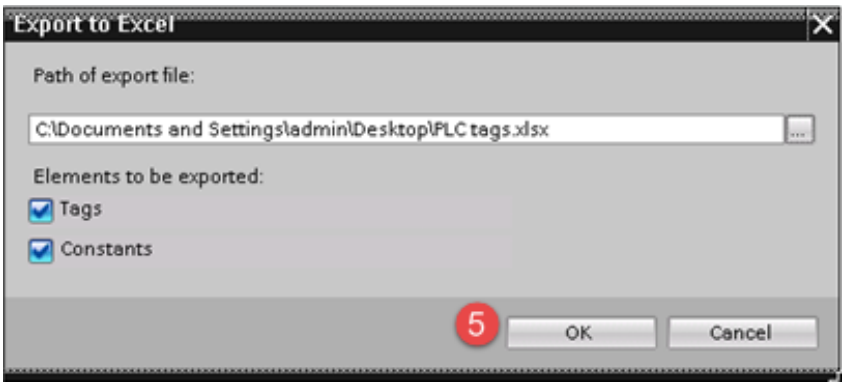


2. Cliquez sur le bouton **Exporter** et indiquez un chemin d'accès.
3. Donner un nom au fichier.
4. Cliquez sur **Enregistrer** pour confirmer.



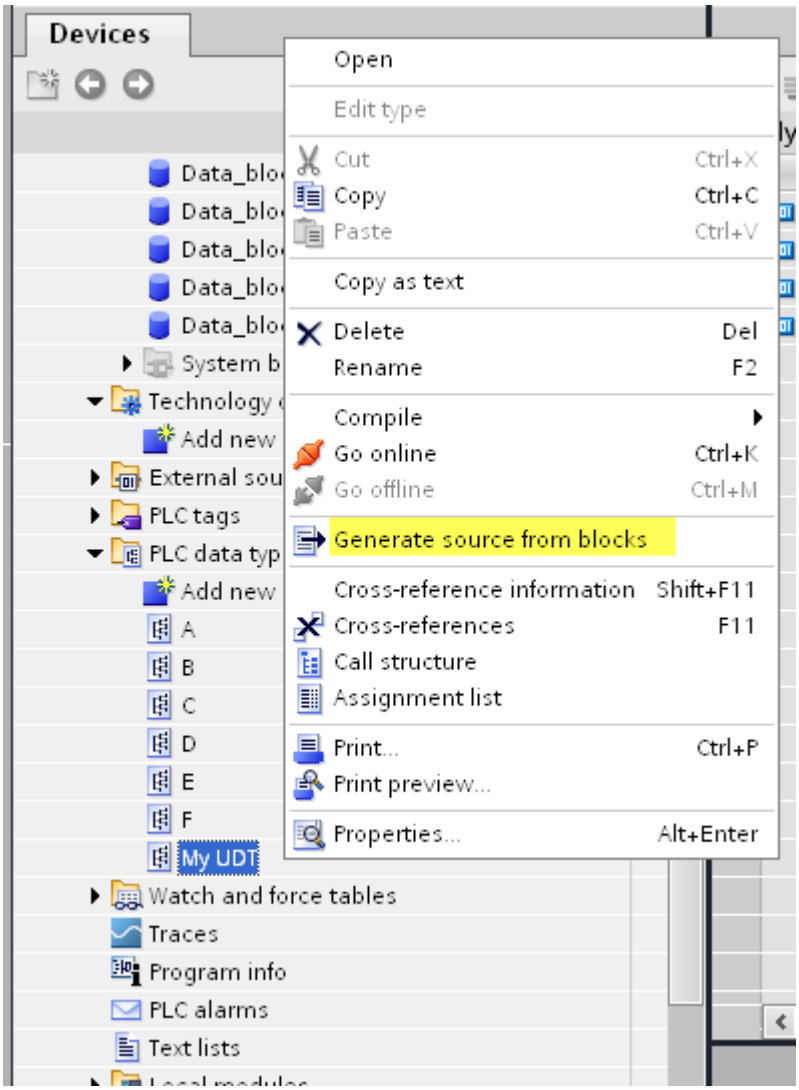
5. Cliquez sur **OK** pour exporter.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



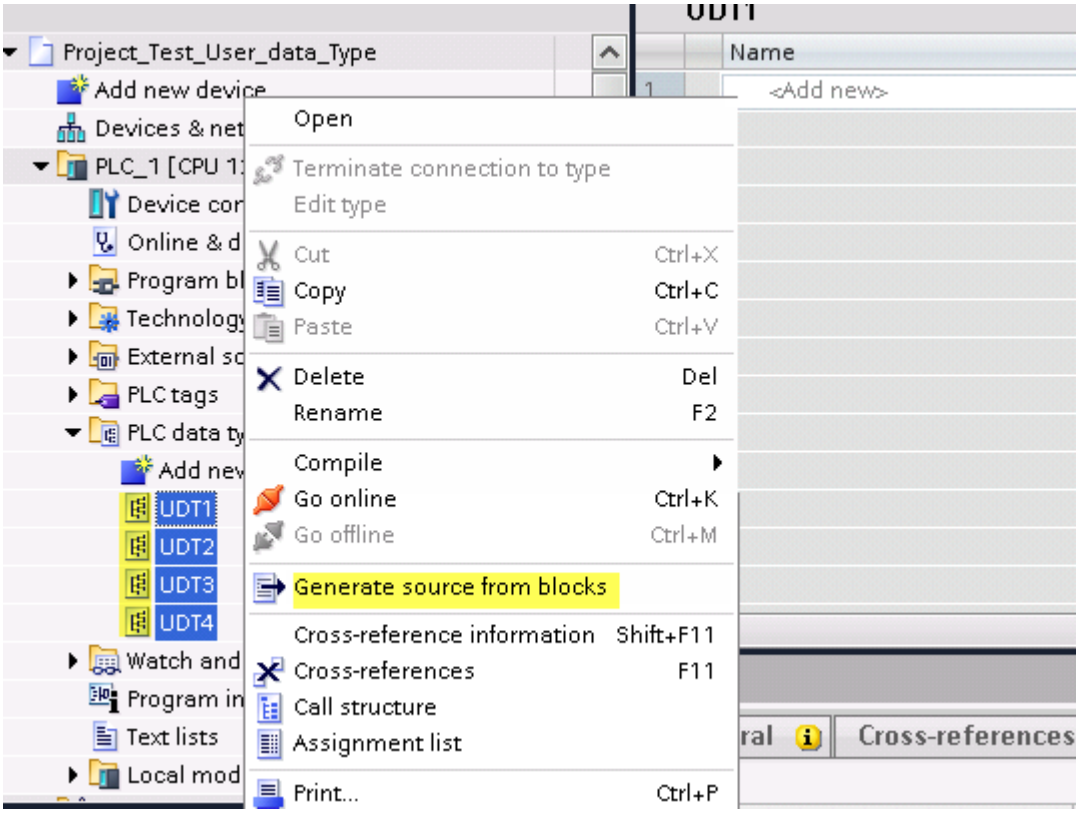
Exporter des types de données API

Pour générer le fichier, agrandissez l'élément **Types de données API** de l'arborescence du projet du portail TIA et faites un clic droit sur la structure définie par l'utilisateur. Cliquez ensuite sur **Générer un fichier source à partir des blocs**.



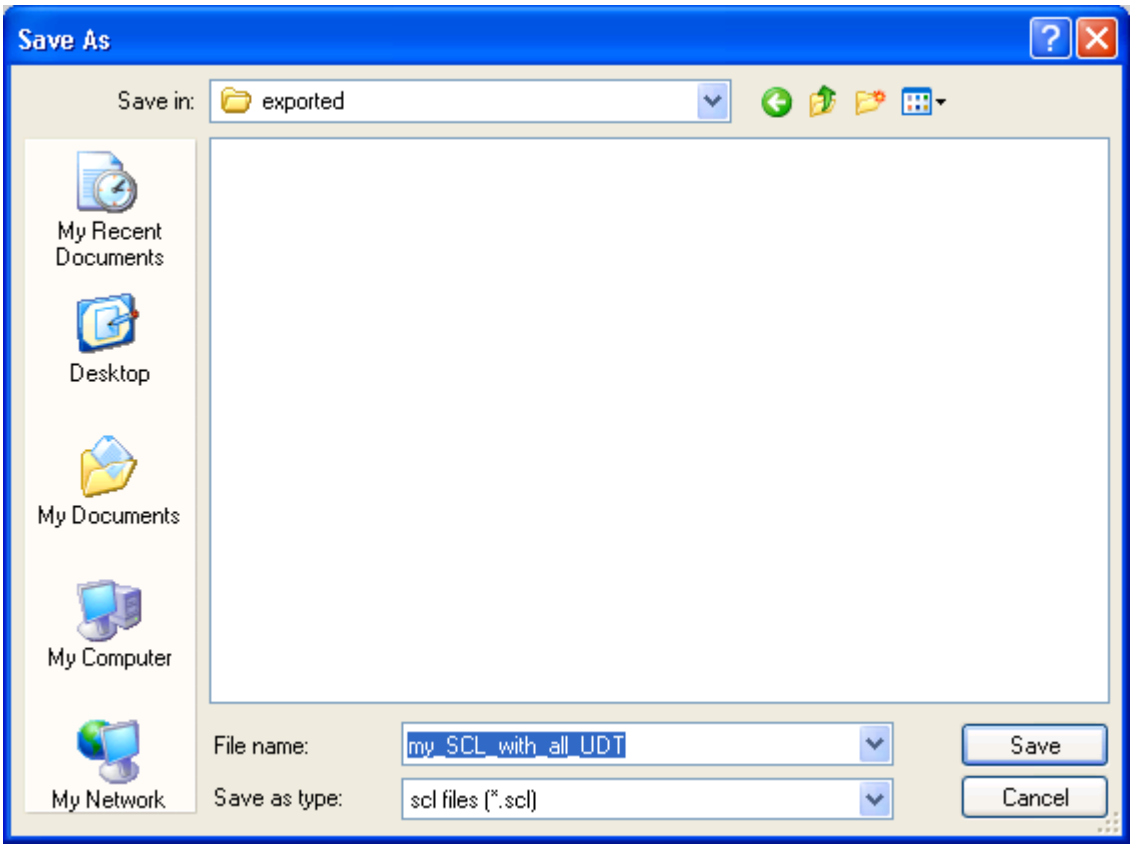
Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

En cas de plusieurs types de données API définis dans le projet API, vous devez tous les sélectionner dans la liste **Types de données API**, faire un clic droit et sélectionner **Générer un fichier source à partir des blocs** pour créer le fichier .scl contenant tous les types de données API définis.



À l'étape suivante, donnez un nom au fichier .scl et indiquez un chemin d'accès où l'enregistrer.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



Ce fichier contiendra tous les types de données API définis et vous pourrez l'utiliser pour importer des variables dans l'Éditeur de variables.

Consultez le chapitre **Importation de variables** pour plus d'informations.

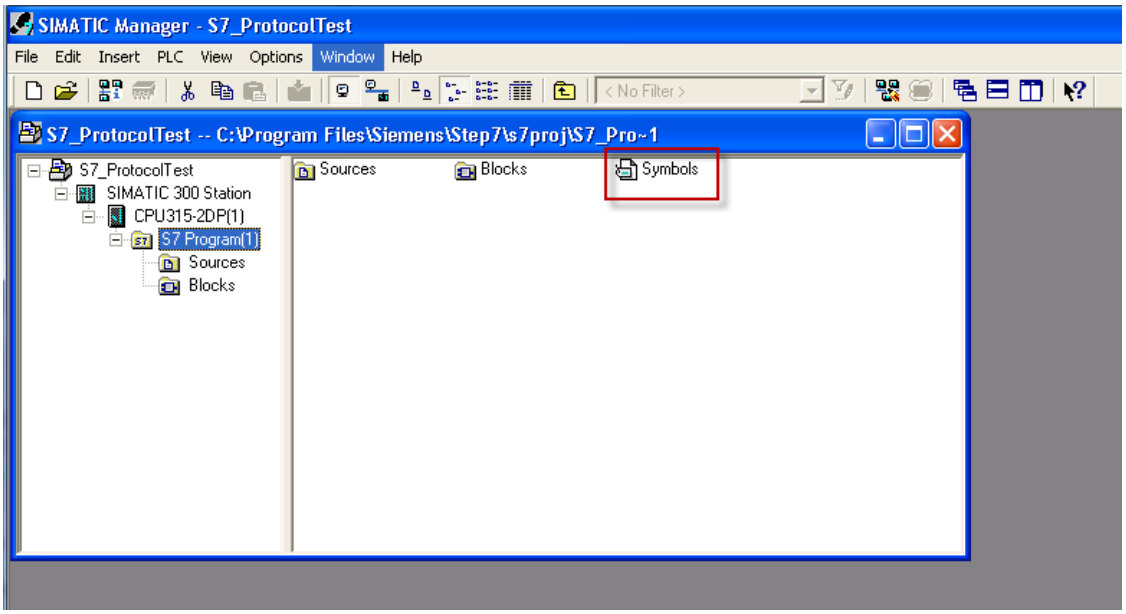
Export à l'aide de STEP7

L'importateur de variables S7 ETH accepte les fichiers de symboles (au format ASCII .asc) et les fichiers sources (dont l'extension est .awl) créés avec Simatic Step7. Vous pouvez préalablement exporter le fichier de symboles grâce à l'utilitaire de la table des mnémoniques de Step7.

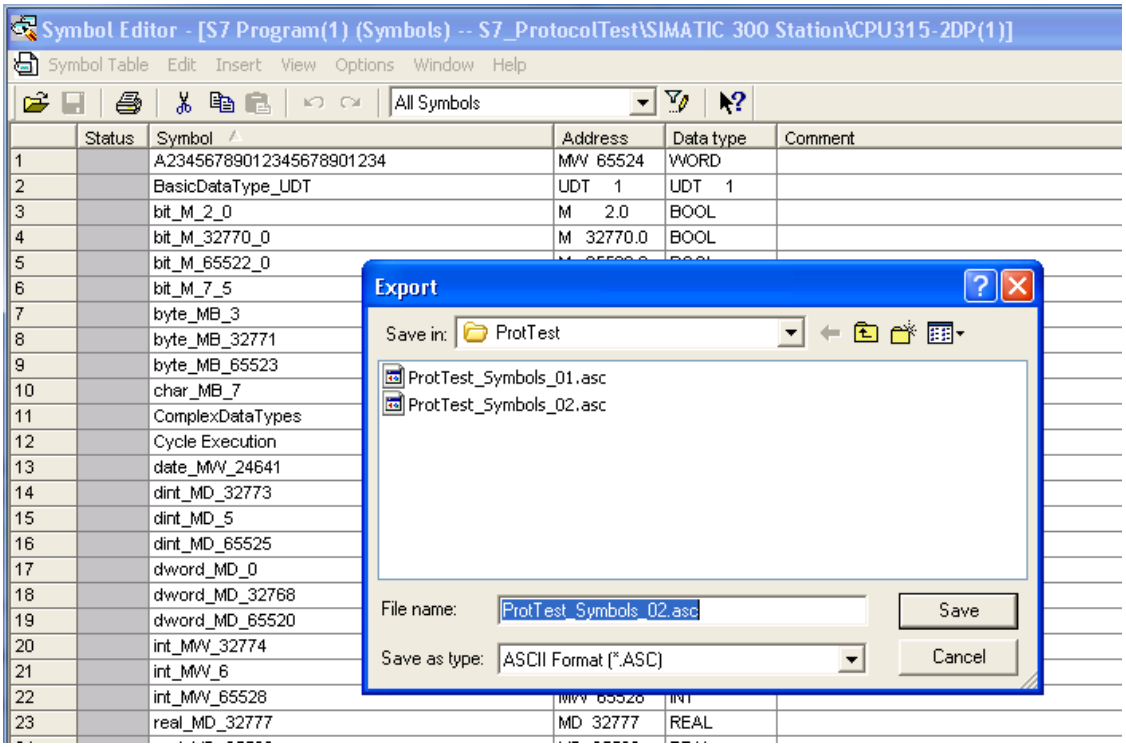
Export d'un tableau de symboles

Vous pouvez exporter les fichiers de symboles (.asc) depuis l'utilitaire de la table des mnémoniques.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



1. Dans l'éditeur de mnémoniques, sélectionnez le menu **Table des mnémoniques** et cliquez sur **Exporter**.
2. Donnez un nom à la table des mnémoniques et enregistrez-la au format de fichier ASCII.

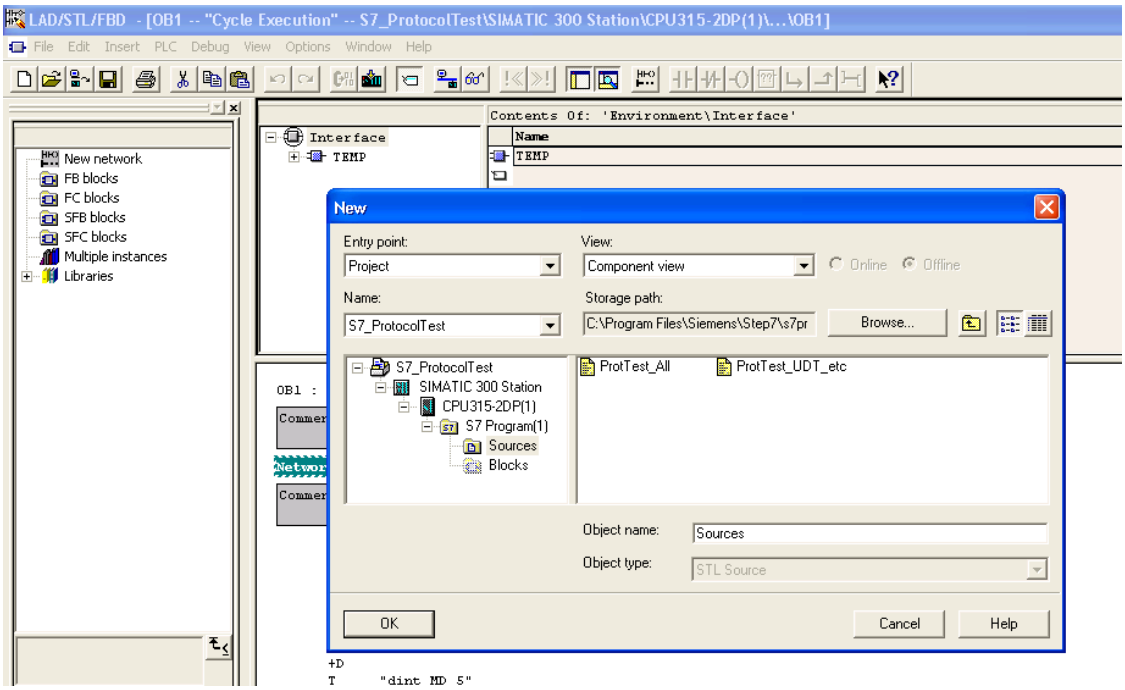


Export des sources

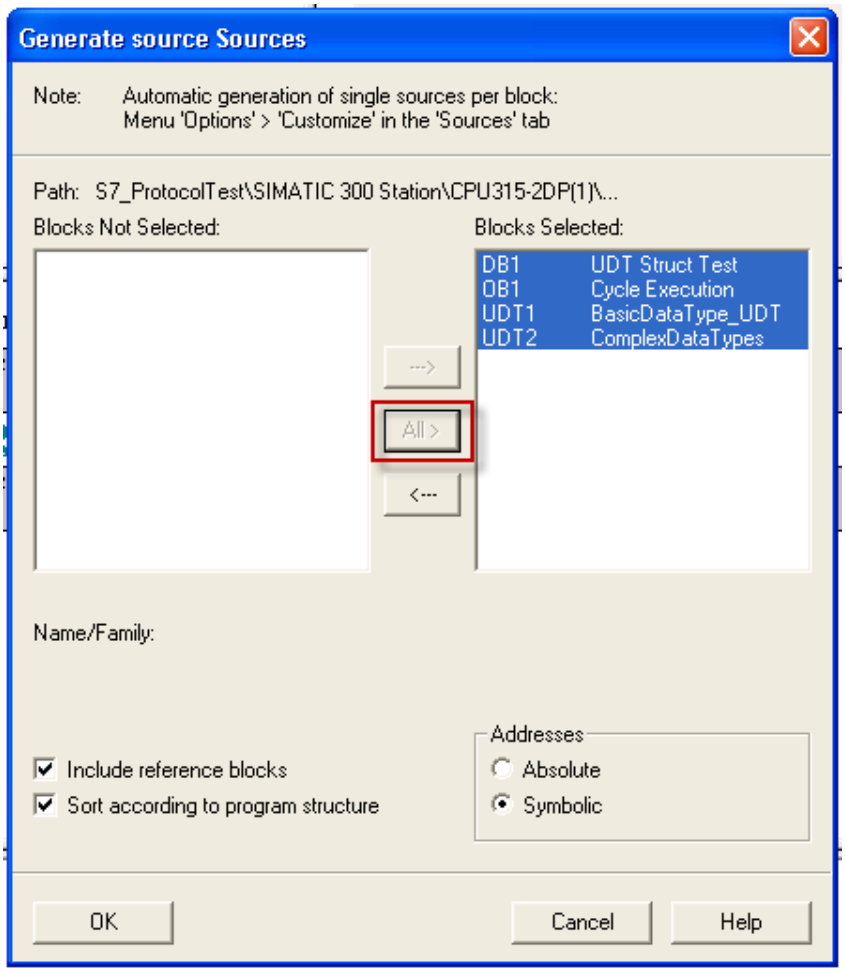
Vous pouvez générer ces fichiers en exportant du code source.

1. Ouvrez n'importe quel bloc de programme dans l'éditeur (« OB1 » dans l'exemple ci-dessous).
2. Dans le menu **Fichier**, cliquez sur **Générer un fichier source**. La boîte de dialogue suivante s'affiche :

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



1. Donnez-lui un nom (« Sources » dans l'exemple) et cliquez sur **OK** : la boîte de dialogue**Générer le fichier source Sources** s'affiche.

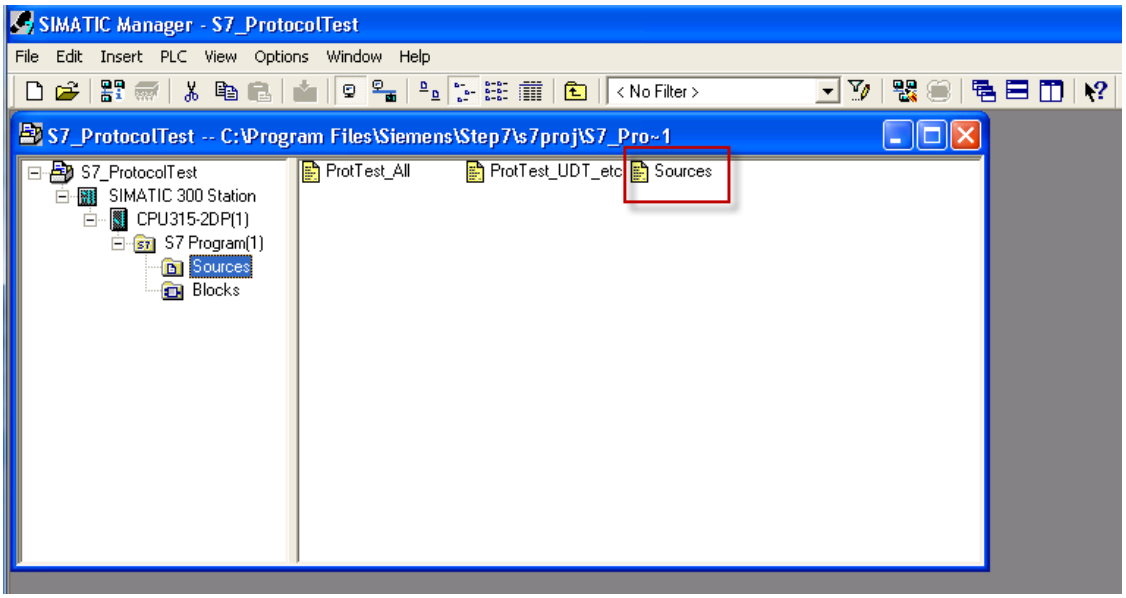


Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

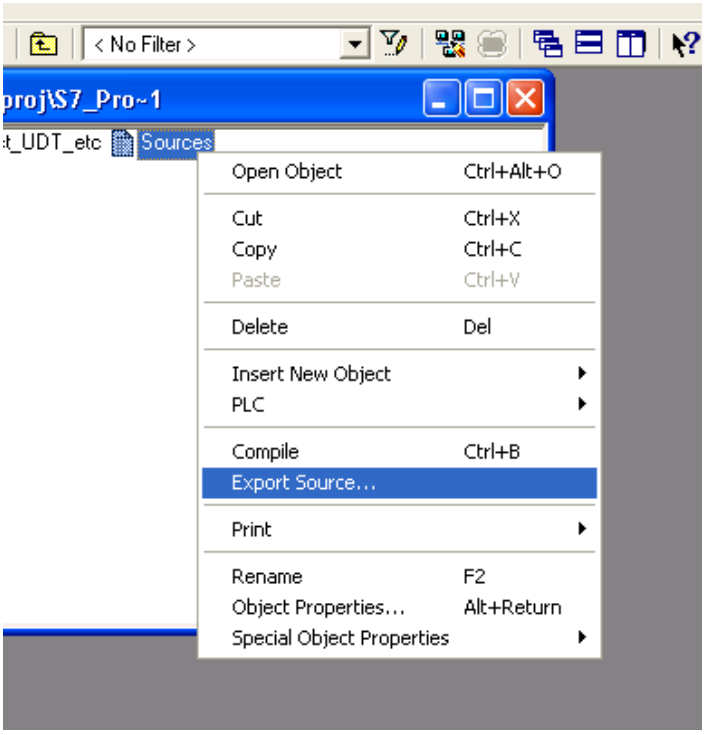
2. Cliquez sur **Tous >** afin de générer un fichier source pour tous les blocs.
3. Sélectionnez les options suivantes :

- **Inclure les blocs de référence**
- **Trier en fonction de la structure du programme**
- Adresse **Symbolique**

4. Cliquez sur **OK** pour confirmer : l'objet « Sources » est généré dans le projet Step7, comme dans l'exemple.



5. Faites un clic droit sur l'objet et sélectionnez **Exporter Sources.**

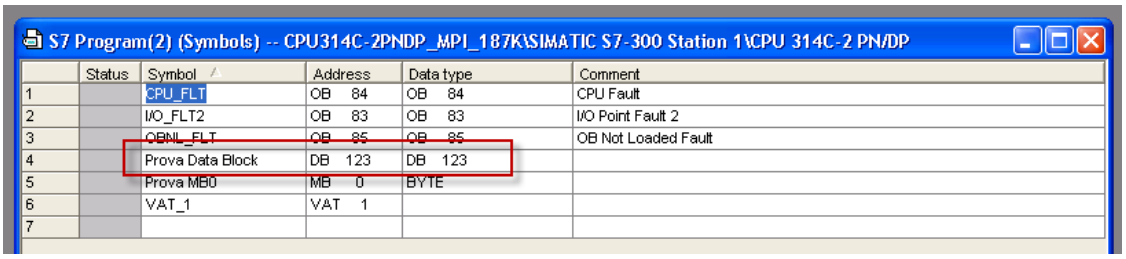


Vous pouvez importer le fichier .awl généré dans l'éditeur de variable.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Remarque : le fichier .awl contient des informations supplémentaires dont ne dispose pas le fichier .asc exporté à partir de la table des mnémoniques.

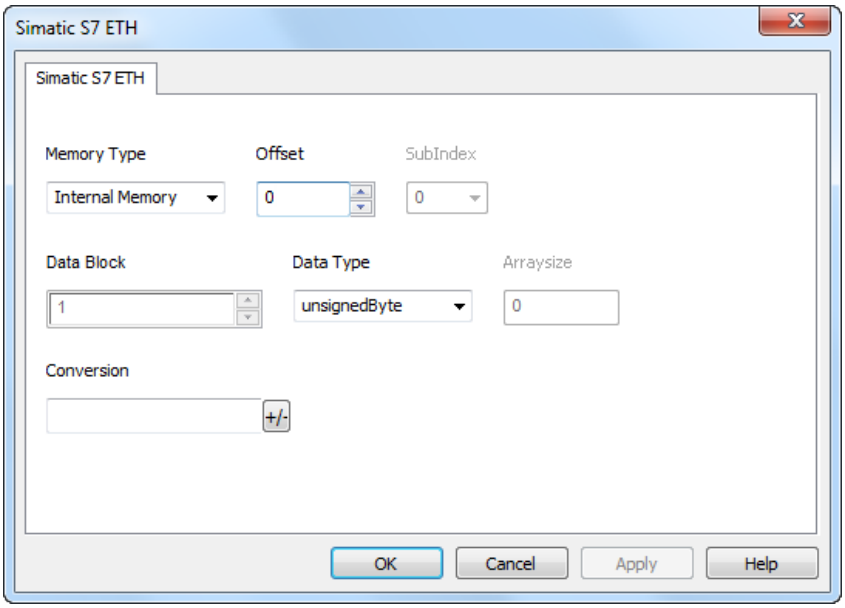
Assurez-vous que la table des mnémoniques contient une référence à tous les blocs de données. Vous pouvez importer les variables d'un bloc de données uniquement lorsque la table des mnémoniques contient une ligne mentionnant le nom du bloc de données et les commentaires le concernant.



Chaque entrée permet au filtre d'importation d'importer les variables se rapportant au bloc de données concerné.

Paramètres de l'éditeur de variables

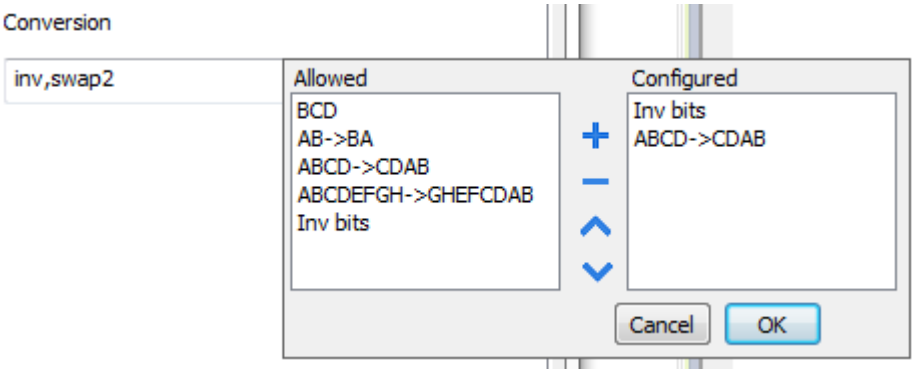
Dans l'éditeur de variables, sélectionnez « Simatic S7 ETH » dans la liste des protocoles définis, puis cliquez sur **+** pour ajouter une variable.



Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

<table>
<tr><th>Élément</th><th>Description</th></tr>
<tr><td>Type de mémoire</td><td>Zone de l’API où se situe la variable.<table><tr><th>Type de données</th><th>Type Simatic</th></tr><tr><td>Mémoire interne</td><td>M</td></tr><tr><td>Bloc de données</td><td>DB</td></tr><tr><td>Entrée</td><td>I (E)</td></tr><tr><td>Sortie</td><td>O (A)</td></tr><tr><td>Valeur de minuterie</td><td>T</td></tr><tr><td>Contre-valeur</td><td>C</td></tr></table></td></tr>
<tr><td>Offset</td><td>Adresse de décalage où se situe la variable.</td></tr>
<tr><td>Sous index</td><td>Décalage de ressource au sein du registre.</td></tr>
<tr><td>Bloc de données</td><td>Numéro de bloc de données pour le type de mémoire de bloc de données.</td></tr>
<tr><td>Type de données</td><td>Types de données disponibles :
booléen
octet
court
entier
octet non signé
court non signé
entier non signé
flottant
chaîne
Consultez la section « Concepts de programmation » du manuel principal.
Remarque : pour définir des tableaux, sélectionnez l’un des formats de type de données suivi de crochets.</td></tr>
</table>

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Élément	Description
Taille de tableau	• En cas de variable vectorielle, cette propriété représente le nombre d'éléments vectoriels. • En cas de variable de chaîne, cette propriété représente le nombre maximum d'octets disponible pour la variable de chaîne. Remarque : le nombre d'octets correspond au nombre de caractères de chaîne si la propriété Encoding est définie sur UTF-8 ou Latin1 dans l'Éditeur de variables. Si la propriété Encoding est définie sur UCS-2BE, UCS-2LE, UTF-16BE ou UTF-16LE, un caractère nécessite 2 octets.
Conversion	Conversion à appliquer à la variable.  En fonction du type de données choisi, la liste **Autorisée** affiche une ou plusieurs conversions, listées ci-dessous.

Valeur	Description
Inv bits	Inverse tous les bits de la variable. *Exemple :* 1001 → 0110 (format binaire) 9 → 6 (format décimal)
Inverser	Inverse la valeur de la variable. *Exemple :* 25.36 → -25.36
AB -> BA	Échange les quartets d'un octet. *Exemple :* 15D4 → 514D (format hexadécimal) 5588 → 20813 (format décimal)
ABCD -> CDAB	Échange les octets d'un mot. *Exemple :* 15D4 → 514D (format hexadécimal) 39628 → 52378 (format décimal)

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Élément	Description

Valeur	Description
ABCDEFGH -> GHEFCDAB	Échange les octets d'un mot double. *Exemple :* 32FCFF54 → 54FFFC32 (format hexadécimal) 855441236 → 1426062386 (format décimal)
ABC...NOP -> OPM...DAB	Échange les octets d'un mot long. Exemple : 142,366 → -893553517,588905 (format décimal) 0 10000000110 00011100101110110110010001011010000111001010110000001 → 1 10000011100 10101010000101000101101101101100101101100000100111101 (format binaire)
BCD	Sépare l'octet en deux quartets et les lit au format décimal (de 0 à 9) *Exemple :* 23 → 17 (format décimal) 0001 0111 = 23 0001 = 1 (premier quartet) 0111 = 7 (second quartet)
S5Timer(BCD)	Utilisé pour prendre en charge S5Timer. Pour plus d'informations, consultez la section **Type de données Simatic S5Timer**.
S5Timer(BIN)	Transformation héritée de S5Timer au format binaire.

Sélectionnez la conversion et cliquez sur le bouton plus. La conversion sélectionnée s'affiche dans la liste **Configurée**.

Si vous configurez d'autres conversions, celles-ci s'affichent dans l'ordre (du haut vers le bas de la liste **Configurée**).

Utilisez les boutons de direction pour changer l'ordre des conversions configurées.

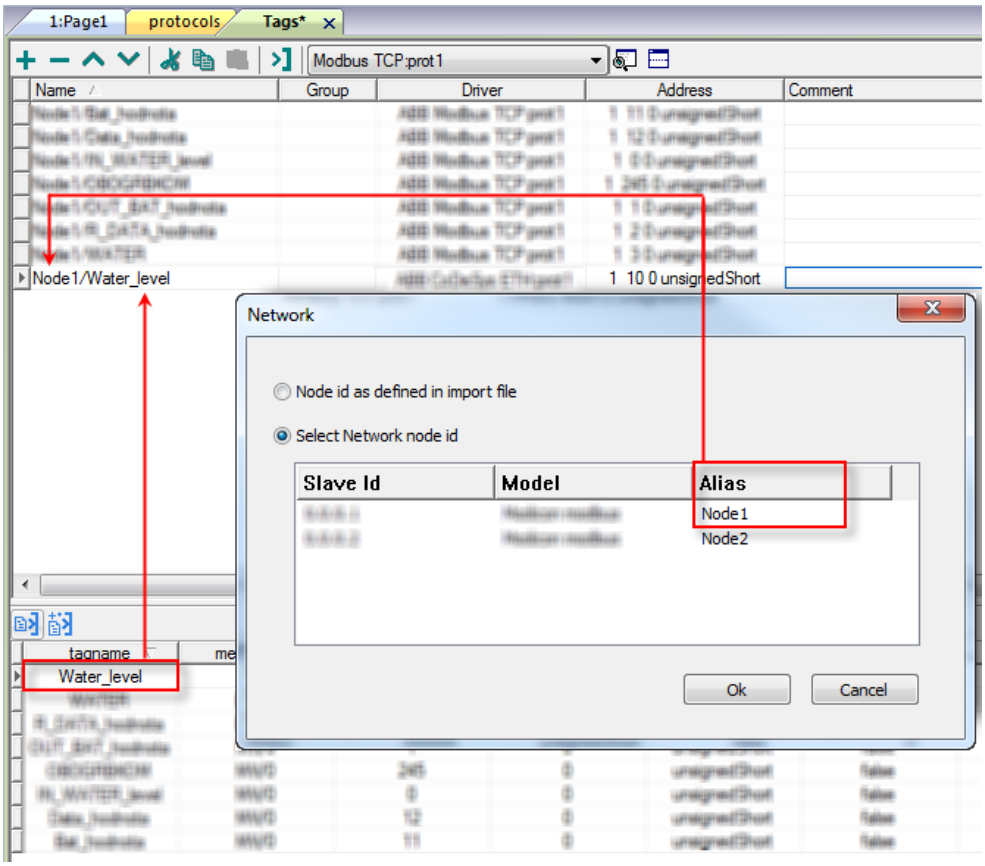
Ajouter un nom d'alias à un protocole

Les noms de variables doivent être uniques au niveau d'un projet. Vous pouvez cependant être amené à utiliser les mêmes noms de variables pour différents nœuds de contrôleur (par exemple lorsque le périphérique IHM est connecté à deux périphériques exécutant la même application).

Lorsque vous générez un protocole, vous pouvez ajouter un nom d'alias qui s'ajoutera aux noms de variables importés pour ce protocole.

Dans l'exemple, la connexion à un certain contrôleur porte le nom **Nœud1**. Lorsque vous importez les variables pour ce nœud, tous les noms de variables porteront le préfixe **Nœud1**, les rendant ainsi uniques au niveau projet/réseau.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



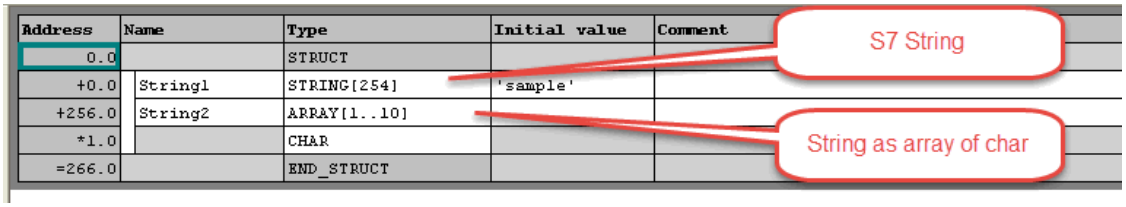
Remarque : vous pouvez donner un nom d'alias uniquement aux variables importées. Il n'est pas possible pour les variables ajoutées manuellement dans l'éditeur de variables de porter un préfixe d'alias dans leur nom.
La chaîne d'alias est associée à l'importation des variables. Une fois l'importation des variables effectuée, une modification de la chaîne d'alias n'aura aucun effet sur les noms figurant déjà dans le dictionnaire. Lorsque vous modifiez la chaîne d'alias et réimportez les variables, toutes les variables réimportées porteront la nouvelle chaîne de préfixe.

Type de données chaîne

Dans l'API Simatic S7, deux différents types de variables gèrent les variables chaîne :

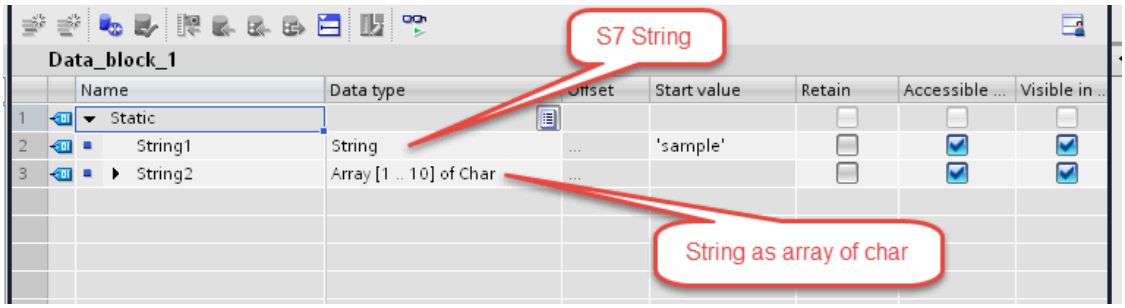
- sous forme de tableau [1..xx] de caractères,
- sous forme de chaîne[xx].

L'exemple suivant présente la déclaration de chaîne sous Step7 :



L'exemple suivant présente la déclaration de chaîne sous le portail TIA :

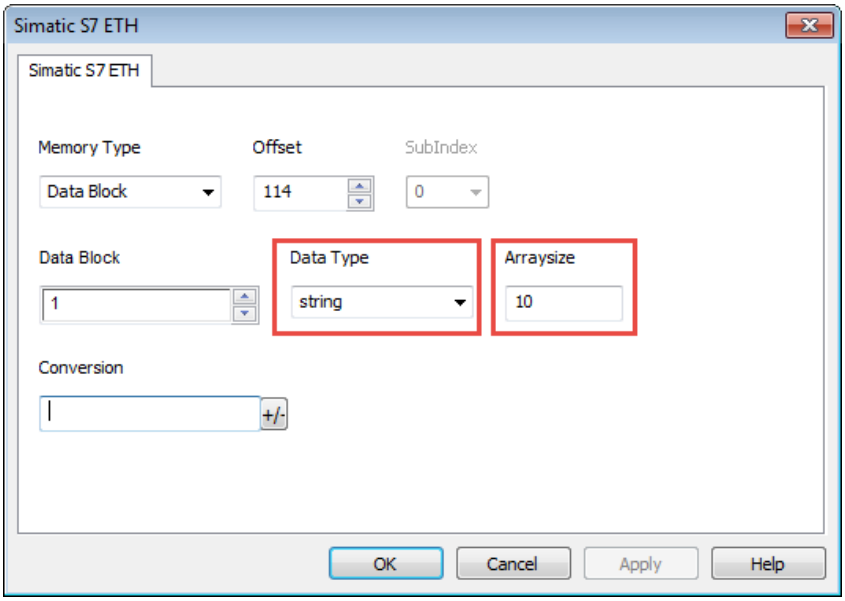
Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



Remarque : lorsque vous utilisez le type de données chaîne[xx], vous devez appliquer une conversion spécifique à la variable. Par contre, lorsque vous utilisez l'outil d'importation pour importer le dictionnaire de variables depuis le portail TIA ou Step7, la conversion des variables chaîne est automatique et aucune autre action ne vous sera demandée.

Pour ajouter une chaîne sous forme de tableau de caractères :

1. Appuyez sur **+** dans l'éditeur de variables.

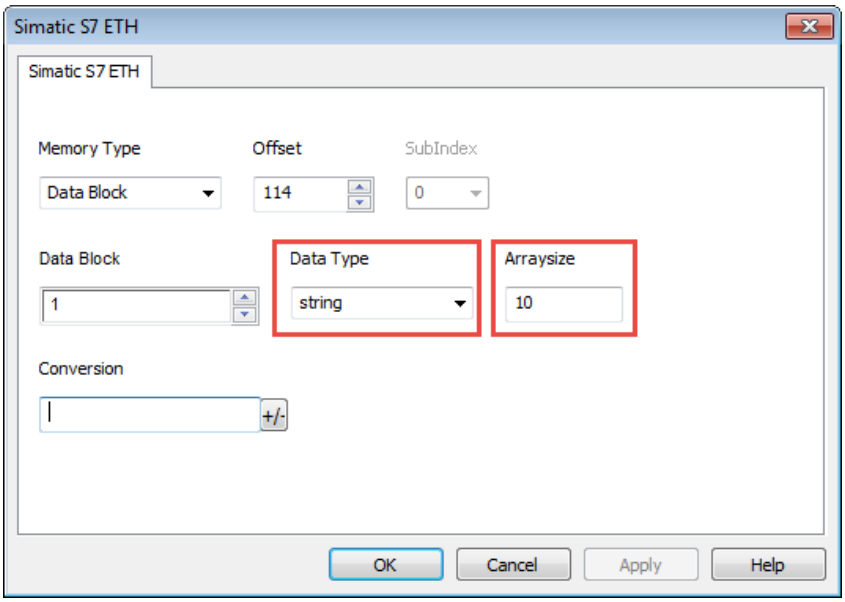


2. Sélectionnez **chaîne** comme **Type de données.**
3. Dans **Taille de tableau**, entrez la longueur de chaîne.
4. Cliquez sur **OK** pour confirmer.

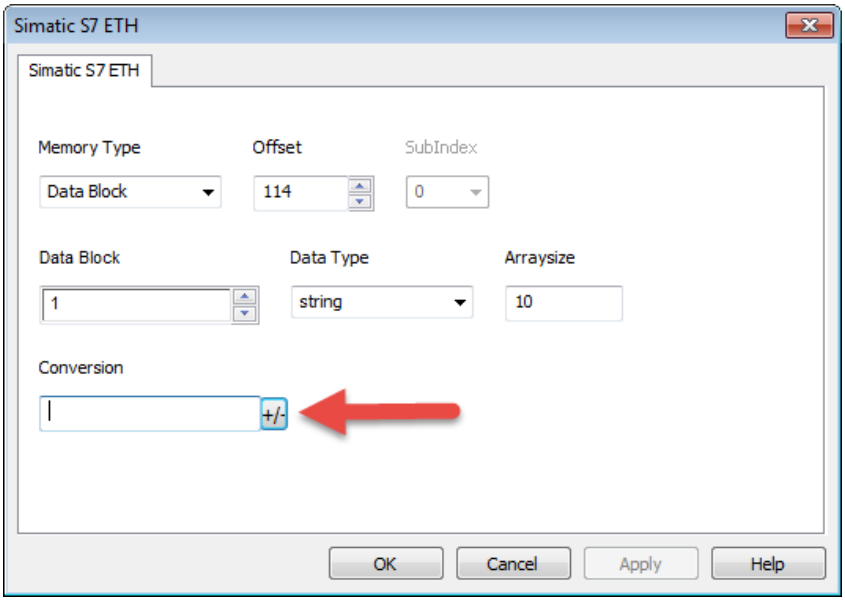
Pour ajouter un type de données chaîne :

1. Appuyez sur **+** dans l'éditeur de variables.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

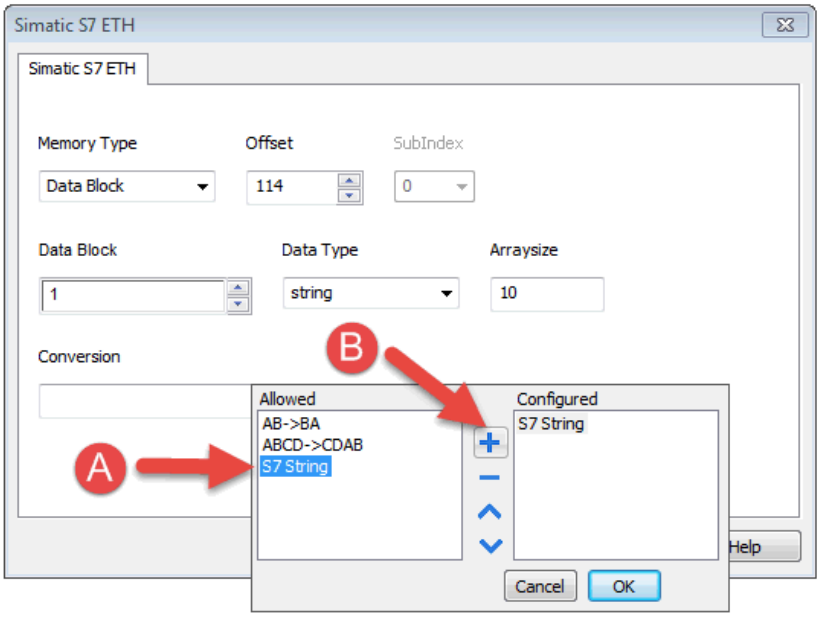


2. Sélectionnez **chaîne** comme **Type de données.**
3. Dans **Taille de tableau**, entrez la longueur de chaîne.
4. Cliquez sur **+/-** pour ouvrir la boîte de dialogue Conversion.



5. Dans cette boîte de dialogue, sélectionnez le type de conversion **Chaîne S7**.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

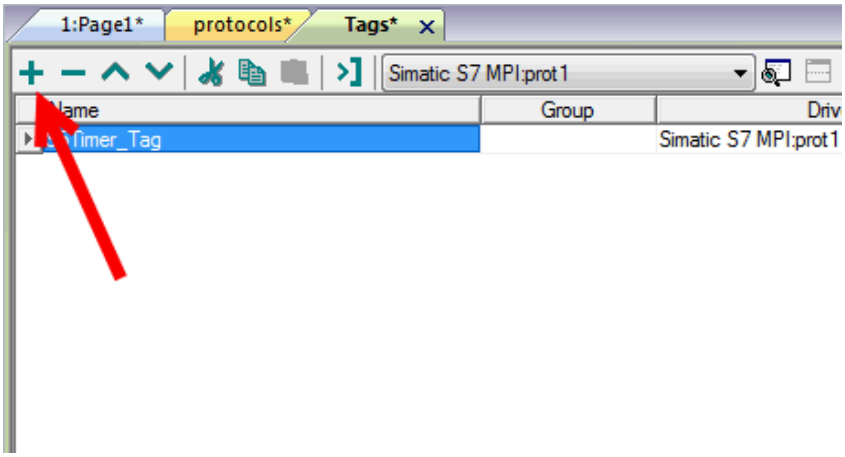


6. Cliquez sur **+** pour ajouter la conversion, qui s'affiche alors dans la liste **Configurée** sur la droite.
7. Cliquez sur **OK** pour confirmer.

Type de données Simatic S5Timer

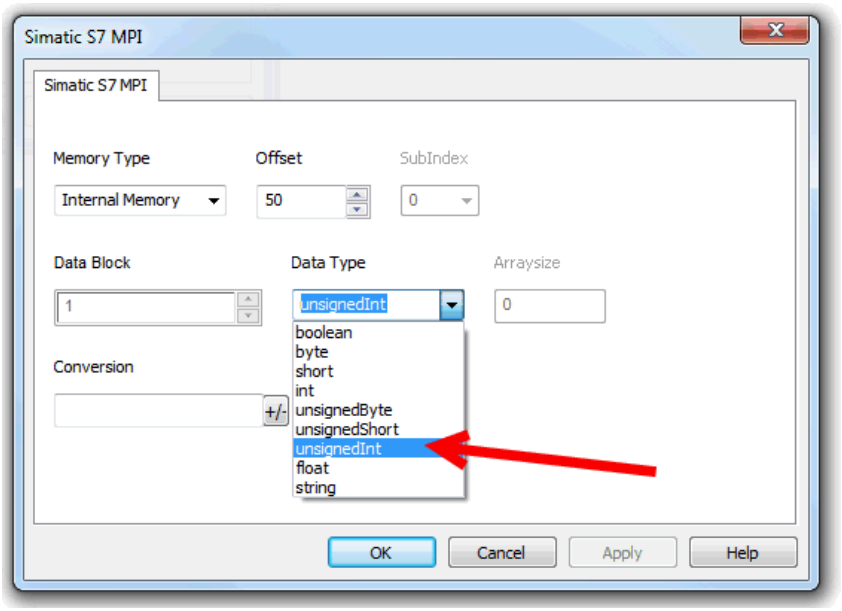
Les pilotes Simatic prennent en charge un type de données spécial : S5Timer.
Vous devez configurer la variable avec un type de données spécifique et appliquer une conversion à cette variable afin de pouvoir lire/écrire correctement une variable Simatic S5Timer.

1. Dans l'éditeur de variables, cliquez sur **+** pour ajouter une variable.

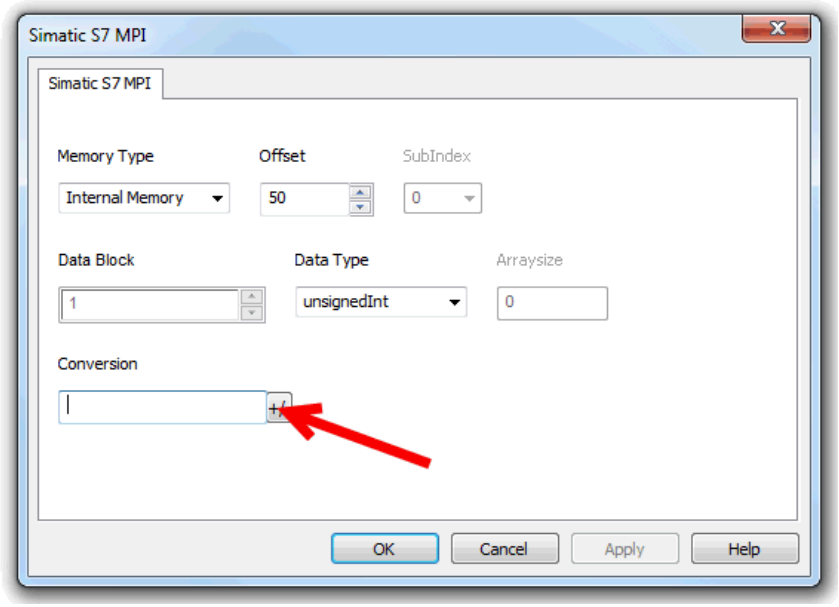


2. Sélectionnez **unsignedInt** comme **Type de données.**

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

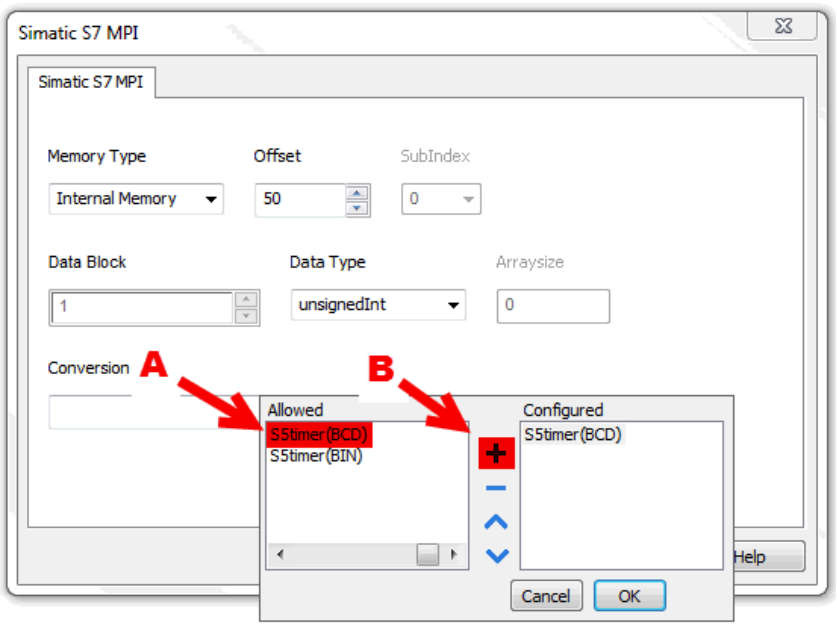


3. Cliquez sur **+/-** pour ouvrir la boîte de dialogue Conversion.



4. Dans cette boîte de dialogue, sélectionnez le type de conversion **S5timer(BCD)**.
5. Cliquez sur **+** pour ajouter la conversion, qui s'affiche alors dans la liste **Configurée** sur la droite.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



6. Cliquez sur **OK** pour confirmer.

IP d'écrasement de nœud

Le protocole propose le type de données spécial IP d'écrasement de nœud, qui vous permet de modifier l'adresse IP du contrôleur cible au moment du runtime.

Ce type de mémoire est un tableau de 4 octets non signés, un pour chaque octet de l'adresse IP.

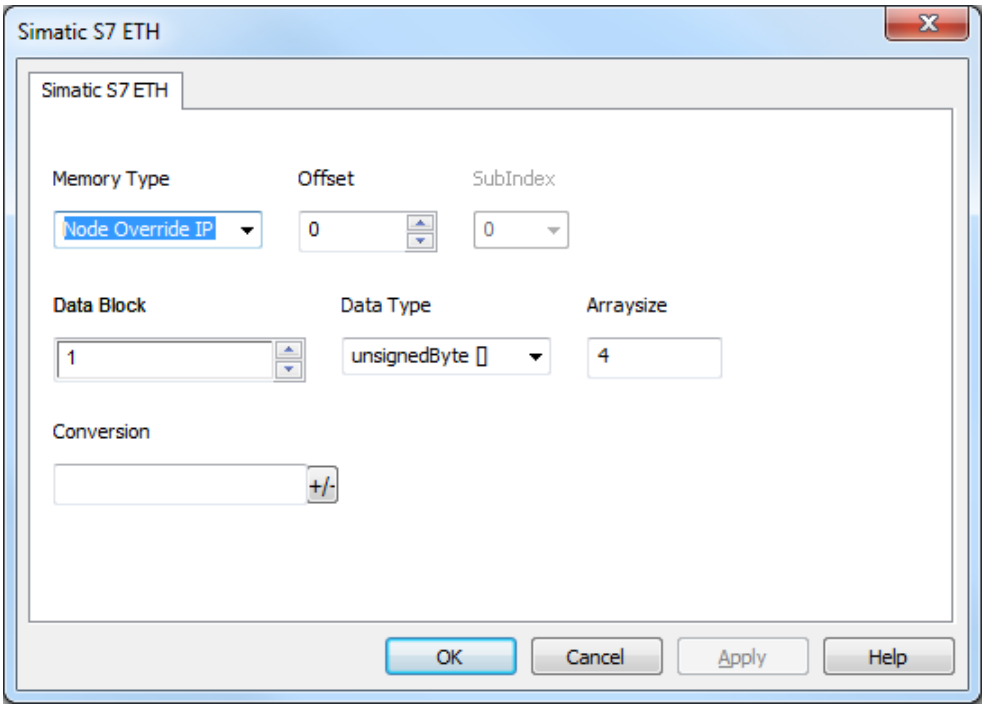
L'IP d'écrasement de nœud est initialisée avec la valeur de l'IP du contrôleur spécifiée dans le projet au moment de la programmation.

IP d'écrasement de nœud	Exécution de l'API
0.0.0.0	La communication avec le contrôleur est coupée, plus aucune trame de requête ne sera générée.
Autre que 0.0.0.0	Interprétée comme étant l'écrasement de l'IP de nœud, et remplace l'adresse IP cible au moment du runtime.

Lorsque le périphérique IHM est connecté à un réseau avec plus d'un nœud de contrôleur, chaque nœud dispose de sa propre variable d'IP d'écrasement de nœud.

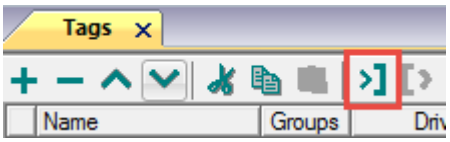
Remarque : le maintien des valeurs de l'IP d'écrasement de nœud assignées au runtime est assuré par des cycles d'alimentation.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

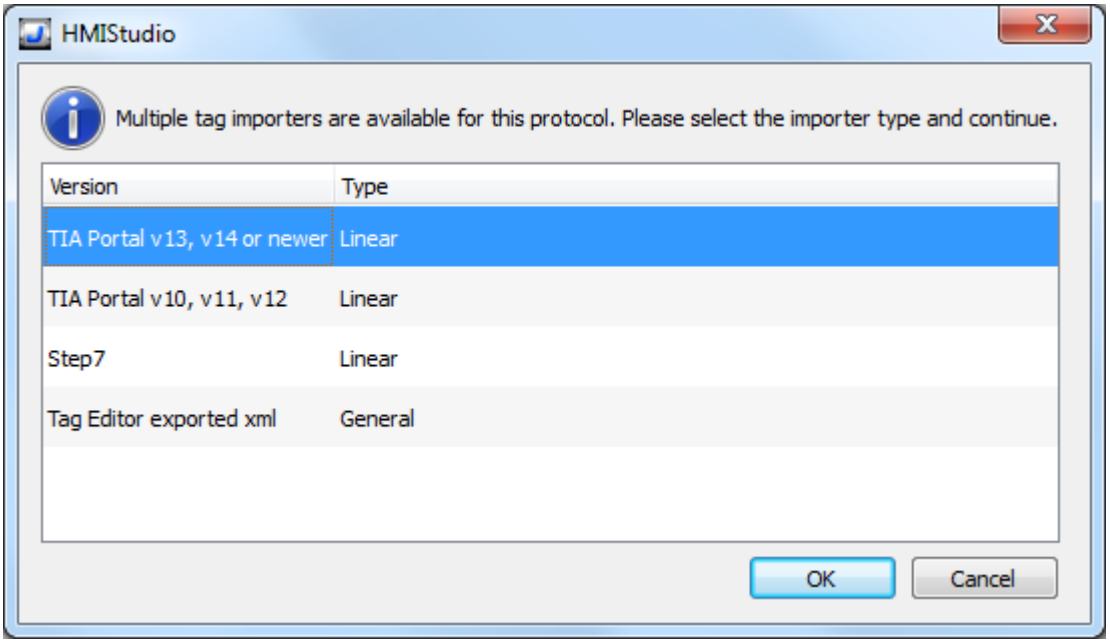


Importation de variables

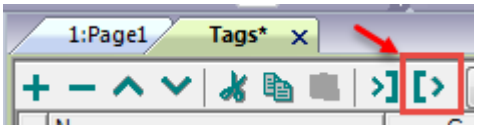
Sélectionnez le pilote dans l'éditeur de variables et cliquez sur le bouton **Importer variables** pour lancer le pilote d'importation.



La boîte de dialogue suivante affiche le type d'importation que vous pouvez sélectionner.



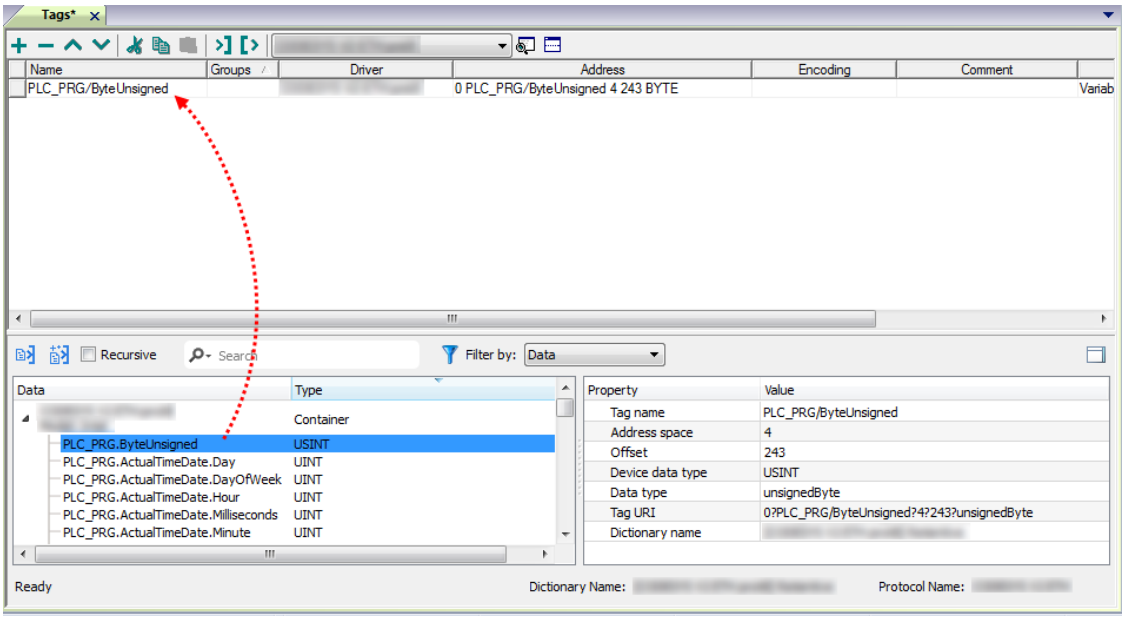
Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

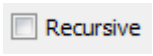
Importateur	Description
Portail TIA v13, v14 ou plus récente Linéaire	Permet d'importer : • Des blocs de programme à l'aide d'un fichier **.db** • Des variables API à l'aide d'un fichier **.xlsx** • Des types de données API à l'aide d'un fichier **.udt** Consultez **Export à l'aide du portail TIA v13, v14 ou plus récente** pour plus de détails. Toutes les variables s'affichent au même niveau.
Portail TIA v10, v11, v12 Linéaire	Permet d'importer : • Des blocs de programme à l'aide d'un fichier **.tia** • Des variables API à l'aide d'un fichier **.xlsx** • Des types de données API à l'aide d'un fichier **.scl** Consultez **Export à l'aide du portail TIA v10, v11, v12** pour plus de détails. Toutes les variables s'affichent au même niveau.
Step7 Linéaire	Permet d'importer : • Fichier de tableau de symboles **.asc** • Sources à l'aide d'un fichier **.awl** Consultez **Exporter à l'aide de STEP7** pour plus de détails. Toutes les variables s'affichent au même niveau.
Fichier .xml exporté de l'éditeur de variables	Sélectionnez ce pilote d'importation pour lire un fichier XML générique exporté de l'éditeur de variables avec le bouton approprié. 

Lorsque vous avez sélectionné le pilote d'importation, localisez le fichier de symboles et cliquez sur **Ouvrir**.

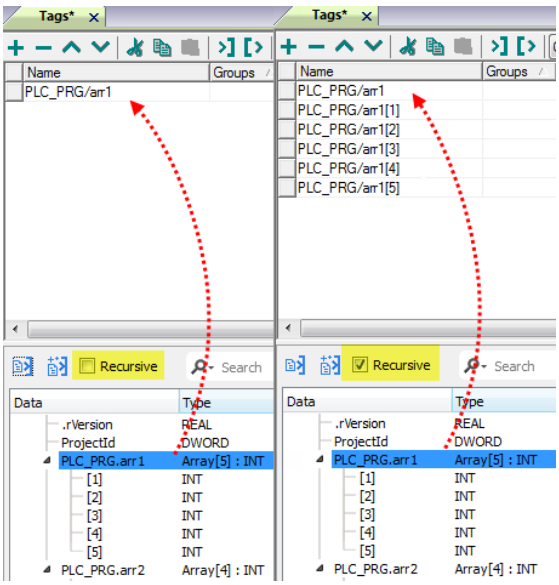
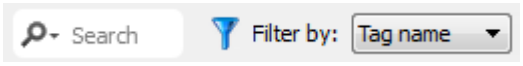
Les variables comprises dans le fichier de symboles sont listées dans le dictionnaire des variables. Le dictionnaire des variables est affiché en bas de l'écran.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



Élément de la barre d'outils	Description
	Importer variable(s). Sélectionnez les variables à importer et cliquez sur cette icône pour ajouter les variables du dictionnaire au projet.
	Mettre à jour variable(s). Cliquez sur cette icône pour mettre à jour les variables du projet grâce à un nouvel import du dictionnaire.
	Cochez cette case pour importer tous les sous-éléments d'une variable. Exemples de résultats avec la case cochée et non cochée :

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Élément de la barre d'outils	Description
	
	Recherche des variables dans le dictionnaire en filtrant en fonction de l'élément sélectionné dans la liste déroulante.

Statut de communication

Vous pouvez afficher l'état de communication actuel grâce aux variables système. Voir la section « Variables système » du manuel principal.

Codes pris en charge par ce pilote de communication :

Erreur	Cause	Action
NAK	Le contrôleur répond « Ne pas confirmer ».	-
Délai d'attente	La requête n'a obtenu aucune réponse dans le délai d'attente spécifié.	Vérifiez que le contrôleur est connecté et correctement configuré pour accéder au réseau.
Réponse non valide	Le contrôleur a envoyé une réponse ou un contenu non valide au périphérique.	Vérifiez la cohérence entre les données programmées dans le projet et les ressources du contrôleur.
Erreur générale	Erreur inconnue. Ne doit jamais être signalée.	Contactez l'assistance technique.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Variables système

Le pilote de communication des Variables système permet de créer des variables pointant sur les informations du système.

Reportez-vous au chapitre *System Variables > Protocol* du Manuel de l'utilisateur.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Paramètres de l’éditeur de protocole

Le pilote de communication des Variables système permet de créer des variables pointant sur les informations du système.

Reportez-vous au chapitre *System Variables > Protocol* du Manuel de l'utilisateur.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Variables

Le pilote de communication de variables permet de définir des variables qui pointe vers la mémoire interne IHM.

Les Variables ne sont pas rétentrices : lorsque le projet démarre, la valeur de départ de toutes les variables Tag est 0 (ou "" en cas de variable de chaîne).

Le pilote de communication de variables n'est pas compté comme protocole physique.
Se référer au **Tableau des fonctions et limites** de manuel principal dans la ligne « Nombre de protocoles physiques ».

Paramètres de l'éditeur de protocole

Ajouter un protocole

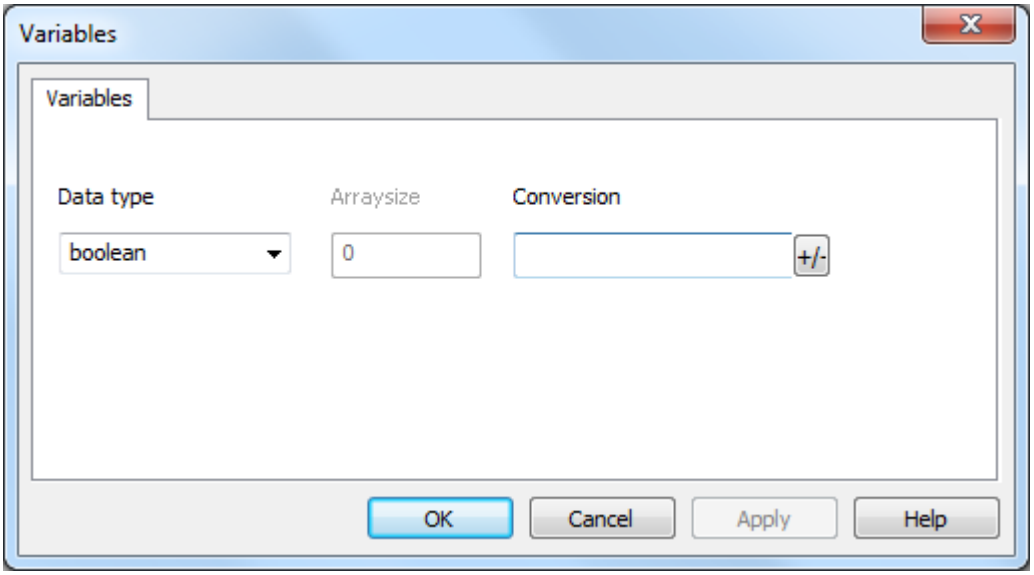
Pour configurer le protocole :

1. Dans le nœud **Config**, double-cliquez sur **Protocoles**.
2. Pour ajouter un pilote, cliquez sur **+** : une nouvelle ligne est ajoutée.
3. Sélectionnez le protocole **Variables** dans la liste **API**.

Paramètres de l'éditeur de variables

*Chemin : **VueProjet**> **Configuration** > double-cliquez sur **Variables***

1. Pour ajouter une variable, cliquez sur **+** : une nouvelle ligne est ajoutée.
2. Sélectionnez **Variables** dans la liste des protocoles : le dialogue de définition de variables s'affiche.



Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

<table>
<tr><th>Élément</th><th>Description</th></tr>
<tr><td>Type de données</td><td>
<table>
<tr><th>Type de données</th><th>Espace mémoire</th><th>Limites</th></tr>
<tr><td>booléen</td><td>Donnée à 1 bit</td><td>0 ... 1</td></tr>
<tr><td>octet</td><td>Donnée à 8 bits</td><td>-128 ... 127</td></tr>
<tr><td>court</td><td>Donnée à 16 bits</td><td>-32768 ... 32767</td></tr>
<tr><td>entier</td><td>Donnée à 32 bits</td><td>-2.1e9 ... 2.1e9</td></tr>
<tr><td>int64</td><td>Donnée à 64 bits</td><td>-9.2e18 ... 9.2e18</td></tr>
<tr><td>octet non signé</td><td>Donnée à 8 bits</td><td>0 ... 255</td></tr>
<tr><td>court non signé</td><td>Donnée à 16 bits</td><td>0 ... 65535</td></tr>
<tr><td>entier non signé</td><td>Donnée à 32 bits</td><td>0 ... 4.2e9</td></tr>
<tr><td>uint64</td><td>Donnée à 64 bits</td><td>0 ... 1.8e19</td></tr>
<tr><td>flottant</td><td>Type virgule flottante de 32 bits simple précision conforme IEEE</td><td>1.17e-38 ... 3.4e38</td></tr>
<tr><td>double</td><td>Type virgule flottante de 64 bits double précision conforme IEEE</td><td>2.2e-308 ... 1.79e308</td></tr>
<tr><td>chaîne</td><td colspan="2">Tableau des éléments contenant du code de caractères défini par l'encodage sélectionné</td></tr>
<tr><td>binaire</td><td colspan="2">Données binaires arbitraires</td></tr>
</table>

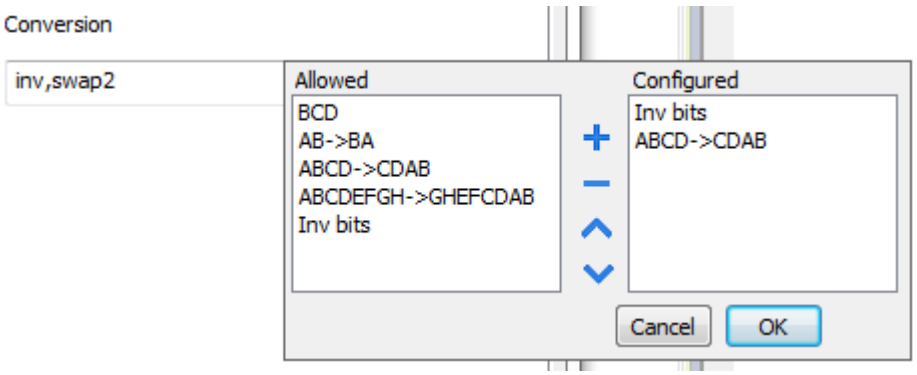
Remarque : pour définir des tableaux, sélectionnez l’un des formats de type de données suivi de crochets tels que « byte[] », « short[] », etc.
</td></tr>
<tr><td>Taille de tableau</td><td>

En cas de variable vectorielle, cette propriété représente le nombre d’éléments vectoriels.
En cas de variable de chaîne, cette propriété représente le nombre maximum d’octets disponible pour la variable de chaîne.

Remarque : le nombre d'octets correspond au nombre de caractères de chaîne si la propriété Encoding est définie sur UTF-8 ou Latin1 dans l'Éditeur de variables.
Si la propriété Encoding est définie sur UCS-2BE, UCS-2LE, UTF-16BE ou UTF-16LE, un caractère nécessite 2 octets.
</td></tr>
<tr><td>Conversion</td><td>Conversion à appliquer à la variable.</td></tr>
</table>

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Élément	Description



En fonction du type de données choisi, la liste **Autorisée** affiche un ou plusieurs types de conversion.

Valeur	Description
Inv bits	**inv** : inverse tous les bits de la variable. *Exemple :* 1001 → 0110 (format binaire) 9 → 6 (format décimal)
Inverser	**neg** : inverse la valeur de la variable. *Exemple :* 25.36 → -25.36
AB -> BA	**swapnibbles** : échange les quartets en un octet. *Exemple :* 15D4 → 514D (format hexadécimal) 5588 → 20813 (format décimal)
ABCD -> CDAB	**swap2** : échange les octets en un mot. *Exemple :* 9ACC → CC9A (format hexadécimal) 39628 → 52378 (format décimal)
ABCDEFGH -> GHEFCDAB	**swap4** : échange les octets en un mot double. *Exemple :* 32FCFF54 → 54FFFC32 (format hexadécimal) 855441236 → 1426062386 (format décimal)
ABC...NOP -> OPM...DAB	**swap8** : échange les octets en un mot long. Exemple : 142,366 → -893553517,588905 (format décimal) 0 10000000110

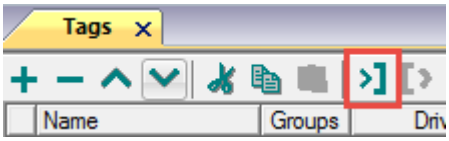
Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

<table>
<tr><th>Élément</th><th>Description</th></tr>
<tr><td></td><td>
<table>
<tr><th>Valeur</th><th>Description</th></tr>
<tr><td></td><td>0001110010111011011001000101101000011100101011000001
→
1 10000011100
10101010000101000101101101101100101101100001001111101
(format binaire)</td></tr>
<tr><td>BCD</td><td>bcd : sépare l’octet en deux quartets, les lit au format décimal (de 0 à 9)

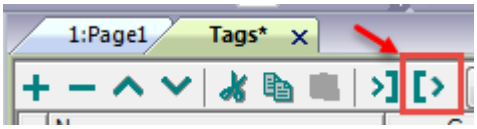
Exemple :
23 → 17 (format décimal)
0001 0111 = 23
0001 = 1 (premier quartet)
0111 = 7 (second quartet)</td></tr>
</table>
Sélectionnez la conversion et cliquez sur le bouton +. La conversion sélectionnée s’affiche dans la liste Configurée.
Si vous configurez d’autres conversions, celles-ci s’affichent dans l’ordre (du haut vers le bas de la liste Configurée).
Utilisez les boutons de direction pour changer l’ordre des conversions configurées.
</td></tr>
</table>

Importation de variables

Sélectionnez le pilote dans l'éditeur de variables et cliquez sur le bouton **Importer variables** pour lancer le pilote d’importation.



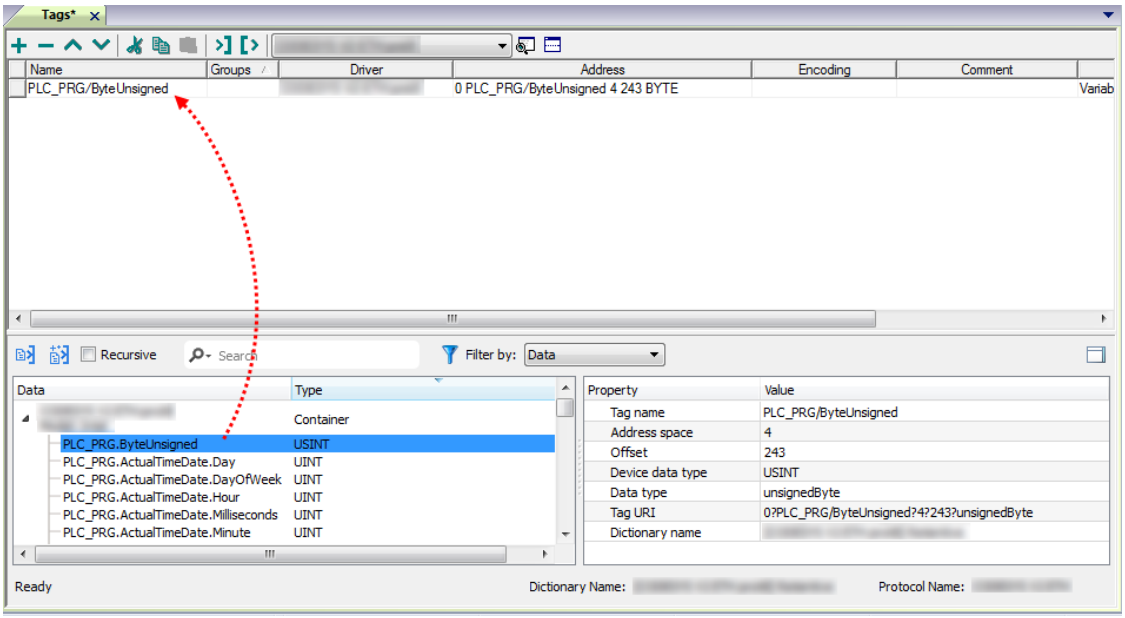
Le système aura besoin d’un fichier XML générique exporté de l'éditeur de variables avec le bouton approprié.

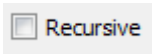


Lorsque vous avez sélectionné le pilote d’importation, localisez le fichier de symboles et cliquez sur **Ouvrir**.

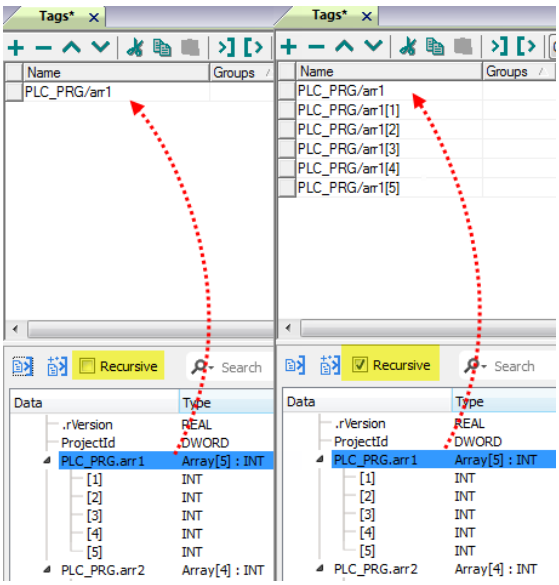
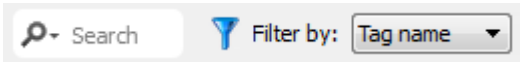
Les variables comprises dans le fichier de symboles sont listées dans le dictionnaire des variables. Le dictionnaire des variables est affiché en bas de l'écran.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21



Élément de la barre d'outils	Description
	Importer variable(s). Sélectionnez les variables à importer et cliquez sur cette icône pour ajouter les variables du dictionnaire au projet.
	Mettre à jour variable(s). Cliquez sur cette icône pour mettre à jour les variables du projet grâce à un nouvel import du dictionnaire.
	Cochez cette case pour importer tous les sous-éléments d'une variable. Exemples de résultats avec la case cochée et non cochée :

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

Élément de la barre d'outils	Description
	
	Recherche des variables dans le dictionnaire en filtrant en fonction de l'élément sélectionné dans la liste déroulante.

Droits d'auteur © 2011-2018 ABB. Tous droits réservés. Révision de la documentation 2.08 - Date de publication : 2018-06-21

ABB Automation Products GmbH
Eppelheimer Straße 82
69123 Heidelberg
Germany
Phone: +49 6221 701 1444
Fax: +49 6221 701 1382
E-Mail: plc.sales@de.abb.com

www.abb.com/plc
www.abb.com/automationbuilder

Note:
We reserve the right to make technical changes or modify the contents of this document without prior notice.
With regard to purchase orders, the agreed particulars shall prevail.
ABB AG does not accept any responsibility whatsoever for potential errors or possible lack of information in this document.

We reserve all rights in this document and in the subject matter and illustrations contained therein.
Any reproduction, disclosure to third parties or utilization of its contents – in whole or in parts – is forbidden without prior written consent of ABB AG.

© Copyright 2011 - 2018 ABB.
All rights reserved.

3ADR010277, 1, fr_FR