User Manual

Integrated Engineering Tool

IET600 Ver. 5.3 Feature Pack 5
Copyright

This document and parts thereof must not be reproduced or copied without written permission from ABB, and the contents thereof must not be imparted to a third party, nor used for any unauthorized purpose.

The software or hardware described in this document is furnished under a license and may be used, copied, or disclosed only in accordance with the terms of such license.

Trademarks

ABB is a registered trademark of ABB Group. All other brand or product names mentioned in this document may be trademarks or registered trademarks of their respective holders.

Guarantee

Please inquire about the terms of guarantee from your nearest ABB representative.

Important notice

Experience has shown that reliable operation of our products is assured, providing the information and recommendations contained in these Operating Instructions are adhered to.

It is scarcely possible for the instructions to cover every eventuality that can occur when using technical devices and systems. We would therefore request the user to notify us directly or our agent of any unusual observations or instances, in which these instructions provide no or insufficient information.

In addition to these instructions, any applicable local regulations and safety procedures must always be strictly observed both when connecting up and commissioning this equipment.

Any work such as insertion or removal of soldered jumpers or setting resistors, which may be necessary, may only be performed by appropriately qualified personnel.

We expressly accept no responsibility or liability for any damage, which may result from operation of this equipment, even if no reference is made to the particular situation in the Operating Instructions.
Disclaimer

This data, examples and diagrams in this manual are included solely for the concept or product description and are not to be deemed as a statement of guarantee properties. All persons responsible for applying the equipment addressed in this manual must satisfy themselves that each intended application is suitable and acceptable, including that any applicable safety or other operational requirements are complied with. In particular, any risks in applications where a system failure and/or product failure would create a risk for harm to property or persons (including but not limited to personal injuries or death) shall be the sole responsibility of the person or entity, applying the equipment, and those so responsible are hereby requested to ensure that all measures are taken to exclude or mitigate such risks.

This document has been carefully checked by ABB but deviations cannot be completely ruled out. In case any errors are detected, the reader is kindly requested to notify the manufacturer. Other than under explicit contractual commitments in no event shall ABB be responsible or liable for any loss or damage resulting from the use of this manual or the application of the equipment.
Table of contents

1 Preface ........................................................................................................... 2
  1.1 IET600, a System Engineering Tool ......................................................... 2
  1.2 Organization of the Manual ....................................................................... 3
  1.3 Use of This Manual for Lookups ............................................................... 4

2 Important concepts in IET600 .................................................................. 5
  2.1 Parallelization of Tasks ............................................................................. 5
  2.2 Naming of Objects ...................................................................................... 5
    2.2.1 Internal Names ................................................................................... 6
    2.2.2 Customer Names ................................................................................ 6

3 IET600 User Interface ................................................................................. 7
  3.1 GUI Overview ............................................................................................. 7
    3.1.1 Main Menu ......................................................................................... 8
    3.1.2 Navigation Trees ................................................................................ 9
    3.1.3 Editors ............................................................................................... 10
    3.1.4 Properties ........................................................................................... 12
    3.1.5 Logging ............................................................................................... 13
    3.1.6 Status Bar .......................................................................................... 14
    3.2 ‘Table’-based Editors ............................................................................. 15
    3.2.1 Editor Elements ................................................................................. 15
    3.2.2 Context Menus .................................................................................... 16
    3.2.3 Show/Hide Columns .......................................................................... 17
    3.2.4 Sort on Columns ............................................................................... 18
    3.2.5 Three State Checkboxes ................................................................. 19
    3.2.6 Set and Clear Filters .......................................................................... 19
  3.3 GUI and Engineering Workflow ................................................................. 22
    3.3.1 Execute a Task via Main Menu ......................................................... 22
    3.3.2 Execute a Task via Navigation Tree and Editor ................................ 22
  3.4 Options ...................................................................................................... 23
    3.4.1 Options -> Appearance .................................................................... 23
    3.4.2 Options -> User Settings .................................................................... 24
    3.4.3 Options -> Updates ........................................................................... 24
    3.4.4 Options -> About ............................................................................... 25

4 IET600 Data Organization ......................................................................... 26
  4.1 Overview .................................................................................................... 26
    4.1.1 Project Data ....................................................................................... 26
    4.1.2 Project Backup .................................................................................. 27
    4.1.3 Project Archives ............................................................................... 27
    4.1.4 Program and User Configuration Data ............................................. 27
  4.2 Managing IET Projects ............................................................................ 28
    4.2.1 The ‘Projects’ Dialog ......................................................................... 28
    4.2.2 Create a New Project ......................................................................... 29
    4.2.3 Open an Existing Project ................................................................. 30
    4.2.4 Save Project Data. ............................................................................ 31
Table of contents

6.4    Creating a Substation by Importing SCD/SSD File .......................... 56

7     IED Engineering ................................................................. 57
    7.1     Introduction ........................................................................ 57
    7.1.1   IED Roles ........................................................................... 57
    7.1.2   IED Status Editor .............................................................. 58
    7.1.3   IED Settings Editor ........................................................... 59
    7.1.4   LN Data Editor ................................................................... 61
    7.1.5   IED Services Editor ......................................................... 62
    7.1.6   IED Capabilities Editor ................................................... 63
    7.2     Creating and Configuring IEDs ............................................. 65
    7.2.1   Overview ............................................................................ 65
    7.2.2   Create New IED .................................................................... 65
    7.2.3   Create New Protection/Control IED ................................. 67
    7.2.4   Create New Switch ............................................................ 68
    7.2.5   Create New HMI ................................................................. 69
    7.2.6   Configure IEDs ..................................................................... 71
    7.3     Updating IEDs ...................................................................... 72
    7.3.1   Overview ............................................................................ 72
    7.3.2   Compare IEDs ..................................................................... 72
    7.3.3   Update one IED ................................................................. 74
    7.3.4   Update Several IEDs Together .......................................... 76
    7.4     Deleting IEDs ...................................................................... 77
    7.4.1   Overview ............................................................................ 77
    7.4.2   Delete an IED ...................................................................... 77
    7.4.3   Delete the Contents of an IED .......................................... 78
    7.5     Mapping LNs to Substation Elements (“LN Mapping”) .......... 79
    7.5.1   Overview ............................................................................ 79
    7.5.2   LN Mapping Restrictions ................................................ 80
    7.5.3   Map LNs in Bays ................................................................. 81
    7.6     Disturbance Recording Configuration ............................... 82
    7.6.1   Introduction ....................................................................... 82
    7.6.2   Configuration ..................................................................... 82
    7.6.3   Disturbance Recording Directory Recommendations .......... 84
    7.6.4   Export ............................................................................... 85
    7.7     Dynamic Datasets ............................................................... 85
    7.8     SNMP Configuration Usage ............................................ 85
    7.8.1   SNMP Properties Configuration ...................................... 86
    7.8.2   SNMP Configuration of Signals ...................................... 87
    7.9     PCM Template Usage ........................................................ 88
    7.9.1   PCM Template Configuration ......................................... 88

8     IEC 61850 Subnetwork Engineering ................................. 89
    8.1     Introduction ....................................................................... 89
    8.2     Subnetwork Conditions and Limitations ....................... 89
    8.2.1   IEC 61850 Subnetworks ................................................ 89
    8.2.2   Modelling of Other Protocols ........................................ 89
    8.3     Subnetwork Configuration ............................................... 90
    8.3.1   Create a Subnetwork ....................................................... 90
    8.3.2   Connect IEDs to a Subnetwork in the Subnetwork Editor ... 91
    8.3.3   Connect IEDs to a Subnetwork in the Communication Tree .. 93
# Table of contents

## Communication Configuration Engineering

<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>9.1</td>
<td>Introduction</td>
<td>95</td>
</tr>
<tr>
<td>9.2</td>
<td>The ‘CCD’ Editor</td>
<td>95</td>
</tr>
<tr>
<td>9.2.1</td>
<td>Diagram View</td>
<td>96</td>
</tr>
<tr>
<td>9.2.2</td>
<td>Table View</td>
<td>97</td>
</tr>
<tr>
<td>9.2.3</td>
<td>Navigate Within Views</td>
<td>97</td>
</tr>
<tr>
<td>9.2.4</td>
<td>Toolbar</td>
<td>98</td>
</tr>
<tr>
<td>9.2.5</td>
<td>IED Shapes</td>
<td>98</td>
</tr>
<tr>
<td>9.2.6</td>
<td>Detail View for Selected IED</td>
<td>99</td>
</tr>
<tr>
<td>9.2.7</td>
<td>Fade Unselected Items</td>
<td>99</td>
</tr>
<tr>
<td>9.2.8</td>
<td>Zoom Diagram</td>
<td>101</td>
</tr>
<tr>
<td>9.2.9</td>
<td>Magnify a Section</td>
<td>102</td>
</tr>
<tr>
<td>9.2.10</td>
<td>Show Bird’s Eye View</td>
<td>102</td>
</tr>
<tr>
<td>9.2.11</td>
<td>Draw Rectangle Shape</td>
<td>103</td>
</tr>
<tr>
<td>9.2.12</td>
<td>Draw Annotation Shape</td>
<td>104</td>
</tr>
<tr>
<td>9.2.13</td>
<td>Print Out Diagram</td>
<td>105</td>
</tr>
<tr>
<td>9.2.14</td>
<td>Export Image of Diagram</td>
<td>106</td>
</tr>
<tr>
<td>9.2.15</td>
<td>Shortcut Keys</td>
<td>106</td>
</tr>
<tr>
<td>9.3</td>
<td>Engineering Workflow</td>
<td>107</td>
</tr>
<tr>
<td>9.4</td>
<td>Creating and Configuring IEDs</td>
<td>108</td>
</tr>
<tr>
<td>9.4.1</td>
<td>Configure Communication Properties</td>
<td>108</td>
</tr>
<tr>
<td>9.5</td>
<td>Placing IEDs</td>
<td>109</td>
</tr>
<tr>
<td>9.5.1</td>
<td>Move an IED Manually</td>
<td>110</td>
</tr>
<tr>
<td>9.5.2</td>
<td>Auto-Arranging IEDs</td>
<td>110</td>
</tr>
<tr>
<td>9.6</td>
<td>Connecting IEDs</td>
<td>111</td>
</tr>
<tr>
<td>9.6.1</td>
<td>Add Connection</td>
<td>111</td>
</tr>
<tr>
<td>9.6.2</td>
<td>Remove Connection</td>
<td>111</td>
</tr>
<tr>
<td>9.6.3</td>
<td>Reconnect Connection</td>
<td>112</td>
</tr>
<tr>
<td>9.6.4</td>
<td>Reroute Connection Manually</td>
<td>112</td>
</tr>
<tr>
<td>9.6.5</td>
<td>Add / Remove Connection Via-Point</td>
<td>112</td>
</tr>
</tbody>
</table>

## Dataflow Engineering

<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.1</td>
<td>Introduction</td>
<td>113</td>
</tr>
<tr>
<td>10.1.1</td>
<td>Overview</td>
<td>113</td>
</tr>
<tr>
<td>10.1.2</td>
<td>Engineering Approach</td>
<td>113</td>
</tr>
<tr>
<td>10.1.3</td>
<td>Common Problems in Dataflow Engineering</td>
<td>114</td>
</tr>
<tr>
<td>10.2</td>
<td>Dataset Configuration</td>
<td>115</td>
</tr>
<tr>
<td>10.2.1</td>
<td>Datasets Editor</td>
<td>115</td>
</tr>
<tr>
<td>10.2.2</td>
<td>Create Datasets</td>
<td>116</td>
</tr>
<tr>
<td>10.2.3</td>
<td>Edit Datasets</td>
<td>117</td>
</tr>
<tr>
<td>10.2.4</td>
<td>Add Dataset Entries</td>
<td>117</td>
</tr>
<tr>
<td>10.2.5</td>
<td>Delete Dataset Entries</td>
<td>118</td>
</tr>
<tr>
<td>10.3</td>
<td>RCB Data Configuration</td>
<td>119</td>
</tr>
<tr>
<td>10.3.1</td>
<td>RCB Data Editor</td>
<td>119</td>
</tr>
<tr>
<td>10.3.2</td>
<td>Create RCBs</td>
<td>121</td>
</tr>
<tr>
<td>10.3.3</td>
<td>Delete RCBs</td>
<td>123</td>
</tr>
<tr>
<td>10.3.4</td>
<td>Edit RCBs</td>
<td>123</td>
</tr>
<tr>
<td>10.3.5</td>
<td>Reuse “Deleted” RCBs</td>
<td>124</td>
</tr>
<tr>
<td>Section</td>
<td>Page</td>
<td></td>
</tr>
<tr>
<td>------------------------------------------------------------------------</td>
<td>------</td>
<td></td>
</tr>
<tr>
<td>10.4 RCB Client Configuration</td>
<td>124</td>
<td></td>
</tr>
<tr>
<td>10.4.1 Prerequisites</td>
<td>124</td>
<td></td>
</tr>
<tr>
<td>10.4.2 RCB Client Editor</td>
<td>124</td>
<td></td>
</tr>
<tr>
<td>10.4.3 Basic Workflow for RCB Client Engineering</td>
<td>125</td>
<td></td>
</tr>
<tr>
<td>10.4.4 Configure Default Clients</td>
<td>126</td>
<td></td>
</tr>
<tr>
<td>10.4.5 Configure RCB Clients</td>
<td>126</td>
<td></td>
</tr>
<tr>
<td>10.5 LCB Data Configuration</td>
<td>128</td>
<td></td>
</tr>
<tr>
<td>10.5.1 LCB Data Editor</td>
<td>128</td>
<td></td>
</tr>
<tr>
<td>10.5.2 Create LCBs</td>
<td>129</td>
<td></td>
</tr>
<tr>
<td>10.5.3 Delete LCBs</td>
<td>130</td>
<td></td>
</tr>
<tr>
<td>10.5.4 Edit LCBs</td>
<td>130</td>
<td></td>
</tr>
<tr>
<td>10.6 GCB Data Configuration</td>
<td>131</td>
<td></td>
</tr>
<tr>
<td>10.6.1 GCB Data Editor</td>
<td>131</td>
<td></td>
</tr>
<tr>
<td>10.6.2 Create GCBs</td>
<td>132</td>
<td></td>
</tr>
<tr>
<td>10.6.3 Delete GCBs</td>
<td>133</td>
<td></td>
</tr>
<tr>
<td>10.6.4 Edit GCBs</td>
<td>134</td>
<td></td>
</tr>
<tr>
<td>10.6.5 Reuse &quot;Deleted&quot; GCBs</td>
<td>134</td>
<td></td>
</tr>
<tr>
<td>10.7 GCB Client Configuration</td>
<td>134</td>
<td></td>
</tr>
<tr>
<td>10.7.1 Prerequisites</td>
<td>134</td>
<td></td>
</tr>
<tr>
<td>10.7.2 GCB Client Editor</td>
<td>135</td>
<td></td>
</tr>
<tr>
<td>10.7.3 Configure GCB Clients</td>
<td>135</td>
<td></td>
</tr>
<tr>
<td>10.7.4 Configure filter for GCB Clients</td>
<td>136</td>
<td></td>
</tr>
<tr>
<td>10.8 SVCB Data Configuration</td>
<td>137</td>
<td></td>
</tr>
<tr>
<td>10.8.1 SVCB Data Editor</td>
<td>137</td>
<td></td>
</tr>
<tr>
<td>10.8.2 Create SVCBs</td>
<td>139</td>
<td></td>
</tr>
<tr>
<td>10.8.3 Delete SVCBs</td>
<td>140</td>
<td></td>
</tr>
<tr>
<td>10.8.4 Edit SVCBs</td>
<td>140</td>
<td></td>
</tr>
<tr>
<td>10.8.5 Reuse &quot;Deleted&quot; SVCBs</td>
<td>140</td>
<td></td>
</tr>
<tr>
<td>10.9 SVCB Client Configuration</td>
<td>141</td>
<td></td>
</tr>
<tr>
<td>10.9.1 Prerequisites</td>
<td>141</td>
<td></td>
</tr>
<tr>
<td>10.9.2 SVCB Client Editor</td>
<td>141</td>
<td></td>
</tr>
<tr>
<td>10.9.3 Configure SVCB Clients</td>
<td>142</td>
<td></td>
</tr>
<tr>
<td>10.9.4 Configure filter for SVCB Clients</td>
<td>142</td>
<td></td>
</tr>
<tr>
<td>10.10 ‘Inputs’ Configuration</td>
<td>143</td>
<td></td>
</tr>
<tr>
<td>10.10.1 Overview</td>
<td>143</td>
<td></td>
</tr>
<tr>
<td>10.10.2 Inputs Editor</td>
<td>144</td>
<td></td>
</tr>
<tr>
<td>10.10.3 Create Inputs</td>
<td>145</td>
<td></td>
</tr>
<tr>
<td>10.10.4 Delete Inputs</td>
<td>145</td>
<td></td>
</tr>
<tr>
<td>10.10.5 Remapping Inputs to Different Parent LNs</td>
<td>146</td>
<td></td>
</tr>
<tr>
<td>10.10.6 Configure Predefined Inputs (Late Binding)</td>
<td>148</td>
<td></td>
</tr>
<tr>
<td>10.10.6.1 Configure Mapping to Predefined DATA Attributes (DAs)</td>
<td>148</td>
<td></td>
</tr>
<tr>
<td>10.10.6.2 Configure Mapping to Predefined DATA Objects (DOs)</td>
<td>149</td>
<td></td>
</tr>
<tr>
<td>10.11 Dataflow Consistency Check</td>
<td>151</td>
<td></td>
</tr>
<tr>
<td>10.12 Dataflow Communication Signal List</td>
<td>152</td>
<td></td>
</tr>
<tr>
<td>11 Customer Name Engineering</td>
<td>153</td>
<td></td>
</tr>
<tr>
<td>11.1 Overview</td>
<td>153</td>
<td></td>
</tr>
<tr>
<td>11.2 The ‘Naming’ Editor</td>
<td>154</td>
<td></td>
</tr>
<tr>
<td>11.2.1 Rename IED, Internal Name</td>
<td>154</td>
<td></td>
</tr>
<tr>
<td>11.2.2 Rename Bay, Internal Name</td>
<td>155</td>
<td></td>
</tr>
<tr>
<td>11.2.3 Rename Substation or Voltage Level, Internal Name</td>
<td>155</td>
<td></td>
</tr>
</tbody>
</table>
# Table of contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>11.2.4</td>
<td>Hierarchical Customer Names</td>
<td>156</td>
</tr>
<tr>
<td>11.2.5</td>
<td>Non-hierarchical Customer Names</td>
<td>156</td>
</tr>
<tr>
<td>11.2.6</td>
<td>Customer Path/Name Display</td>
<td>156</td>
</tr>
<tr>
<td>11.3</td>
<td>LD Naming</td>
<td>156</td>
</tr>
<tr>
<td>11.3.1</td>
<td>Customization of Logical Device Names</td>
<td>157</td>
</tr>
<tr>
<td>11.3.2</td>
<td>Configuring Control Block Clients</td>
<td>157</td>
</tr>
</tbody>
</table>

## 12 HMI Engineering

<table>
<thead>
<tr>
<th>Subsection</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.1</td>
<td>Introduction</td>
<td>158</td>
</tr>
<tr>
<td>12.1.1</td>
<td>User Interface</td>
<td>158</td>
</tr>
<tr>
<td>12.1.2</td>
<td>Engineering Workflow</td>
<td>158</td>
</tr>
<tr>
<td>12.2</td>
<td>HMI Creation and Configuration</td>
<td>159</td>
</tr>
<tr>
<td>12.2.1</td>
<td>Overview</td>
<td>159</td>
</tr>
<tr>
<td>12.2.2</td>
<td>Preliminary: Configure Project OI</td>
<td>161</td>
</tr>
<tr>
<td>12.2.3</td>
<td>Create an HMI</td>
<td>161</td>
</tr>
<tr>
<td>12.2.4</td>
<td>Create a Hot-Standy (HSB) HMI System</td>
<td>162</td>
</tr>
<tr>
<td>12.3</td>
<td>Signal Creation</td>
<td>164</td>
</tr>
<tr>
<td>12.3.1</td>
<td>Overview</td>
<td>164</td>
</tr>
<tr>
<td>12.3.2</td>
<td>Copy HMI Signals from another IED</td>
<td>164</td>
</tr>
<tr>
<td>12.3.3</td>
<td>Insert HMI Signals from Signal Templates</td>
<td>165</td>
</tr>
<tr>
<td>12.3.4</td>
<td>Import a MicroSCADA Database File</td>
<td>165</td>
</tr>
<tr>
<td>12.3.5</td>
<td>Create HMI Signals manually</td>
<td>166</td>
</tr>
<tr>
<td>12.4</td>
<td>Signal Configuration</td>
<td>167</td>
</tr>
<tr>
<td>12.4.1</td>
<td>Overview</td>
<td>167</td>
</tr>
<tr>
<td>12.4.2</td>
<td>HMI Data Editor</td>
<td>167</td>
</tr>
<tr>
<td>12.4.3</td>
<td>Generic HMI Signals</td>
<td>168</td>
</tr>
<tr>
<td>12.4.4</td>
<td>HMI Text Rules</td>
<td>169</td>
</tr>
<tr>
<td>12.4.5</td>
<td>Mapping Signals to IEC 61850 Attributes</td>
<td>173</td>
</tr>
<tr>
<td>12.4.6</td>
<td>Mapping Signals to SNMP OIDs</td>
<td>173</td>
</tr>
<tr>
<td>12.4.7</td>
<td>HMI Data Consistency Check</td>
<td>174</td>
</tr>
<tr>
<td>12.5</td>
<td>Signal Client Configuration</td>
<td>176</td>
</tr>
<tr>
<td>12.5.1</td>
<td>Overview</td>
<td>176</td>
</tr>
<tr>
<td>12.5.2</td>
<td>The 'Signal Clients' Editor</td>
<td>176</td>
</tr>
<tr>
<td>12.5.3</td>
<td>Select Signals to be sent to HMI Manually</td>
<td>176</td>
</tr>
<tr>
<td>12.5.4</td>
<td>Copy Full Signal Client HMI Configuration from Existing HMI</td>
<td>177</td>
</tr>
<tr>
<td>12.6</td>
<td>Configuration of Signals of Non-IEC-61850 IEDs</td>
<td>178</td>
</tr>
<tr>
<td>12.6.1</td>
<td>Introduction</td>
<td>178</td>
</tr>
<tr>
<td>12.6.2</td>
<td>Supported Protocols</td>
<td>178</td>
</tr>
<tr>
<td>12.6.3</td>
<td>Engineering Workflow</td>
<td>179</td>
</tr>
<tr>
<td>12.6.4</td>
<td>Configure 'Mapped To' Attribute of Signals</td>
<td>179</td>
</tr>
<tr>
<td>12.6.4.1</td>
<td>Pseudo Mapping to Resolve References</td>
<td>179</td>
</tr>
<tr>
<td>12.6.5</td>
<td>Configure Signal Addresses</td>
<td>180</td>
</tr>
<tr>
<td>12.7</td>
<td>HMI Status Text Configuration</td>
<td>180</td>
</tr>
<tr>
<td>12.7.1</td>
<td>Overview</td>
<td>180</td>
</tr>
<tr>
<td>12.7.2</td>
<td>View/Edit Status Texts</td>
<td>181</td>
</tr>
<tr>
<td>12.8</td>
<td>Merge HMI Data in MicroSCADA</td>
<td>181</td>
</tr>
<tr>
<td>12.8.1</td>
<td>Introduction</td>
<td>181</td>
</tr>
<tr>
<td>12.8.2</td>
<td>Engineering Procedure and Restrictions</td>
<td>182</td>
</tr>
<tr>
<td>12.8.3</td>
<td>Preparation of Projects</td>
<td>182</td>
</tr>
<tr>
<td>12.9</td>
<td>Working With User Defined Signal Templates</td>
<td>183</td>
</tr>
<tr>
<td>12.9.1</td>
<td>Overview</td>
<td>183</td>
</tr>
</tbody>
</table>
### Table of contents

12.9.2 Create new Signal Template ......................................................... 183
12.9.3 Insert Signal Template ................................................................. 184
12.9.4 Maintaining Templates ................................................................. 184

13

**HMI Mirroring** ................................................................. 186

13.1 Overview ................................................................................. 186
13.1.1 User Interface ................................................................. 186
13.2 Configure Clients and Servers .................................................. 186
13.2.1 Configure an HMI to be a Mirroring Client ................. 187
13.2.2 Configure an HMI to be a Mirroring Server for one Specific Client ................................................................. 187
13.3 Configure Signals to be Mirrored ........................................... 187
13.3.1 Mirrored vs. Copied Signals ............................................. 191
13.3.2 Configuring Copied Signals ............................................. 192
13.3.3 Signals Which Cannot be Mirrored .................................. 192
13.4 Edit Mirrored signals ......................................................... 193
13.4.1 Inheritance ................................................................. 193
13.4.2 Blocked Attributes ............................................................. 193
13.5 Common Mirroring Configurations ....................................... 193
13.5.1 Mutual Mirroring in the Same Project ......................... 193
13.5.2 Mirroring with one NCC Connection ............................. 194
13.5.3 Mirroring with two NCC Connections ............................. 195
13.5.4 Mirroring to a Control Center ................................. 195
13.5.5 Mirroring Across Two Projects With one Common NCC Gateway ................................................................. 196
13.5.6 Mirroring Across Two Projects With Restricted Data Exchange ................................................................. 196

14

**NCC Gateway Engineering** ......................................................... 198

14.1 Introduction ............................................................................. 198
14.1.1 Supported Gateway Products ............................................ 198
14.1.2 Supported Gateway Protocols ............................................. 198
14.1.3 User Interface ................................................................. 199
14.1.4 Prerequisites ................................................................. 199
14.2 Engineering Workflow ......................................................... 199
14.3 Creating and Configuring Gateways ........................................ 200
14.3.1 Overview ............................................................................. 200
14.3.2 Create a Stand-Alone Gateway ........................................... 201
14.3.3 Create a Gateway combined with HMI ............................... 201
14.3.4 Create a Hot-Standby (HSB) Gateway System ................. 202
14.4 Creating NCCs ................................................................. 203
14.4.1 Overview ............................................................................. 203
14.4.2 Create a Stand-Alone NCC .................................................. 204
14.4.3 Create a Hot-Standby (HSB) NCC System ......................... 204
14.5 Connecting Gateways to NCCs .............................................. 205
14.5.1 Overview ............................................................................. 205
14.5.2 Connect Gateway to NCC .................................................. 206
14.5.3 IEC 101/104 Special Parameters ....................................... 206
14.6 Selecting Signals for NCCs ..................................................... 208
14.6.1 Overview ............................................................................. 208
14.6.2 The 'Signal Clients' Editor .................................................. 208
14.6.3 Select Signals to Be Sent to NCC ............................................. 209
14.7 Grouping Signals ................................................................. 209
14.7.1 Overview ........................................................................... 209
14.7.2 Create an Indication Group ................................................. 210
14.7.3 Create a Command Group ................................................... 211
14.8 Copying a Configuration from an Existing NCC ......................... 212
14.8.1 Copy Full Signal Client NCC Configuration from Existing NCC 212
14.9 Configure NCC Data ................................................................. 213
14.9.1 Overview ........................................................................... 213
14.9.2 NCC Data Editor ................................................................. 213
14.9.3 Configure NCC Signal Properties ......................................... 214
14.10 Merge NCC Data in MicroSCADA ............................................ 215
14.10.1 Introduction ...................................................................... 215
14.10.2 Engineering Procedure and Restrictions ............................ 216
14.10.3 Preparation of Projects ...................................................... 216
14.10.4 Calculate the NCC Gateway Group Offset ......................... 217
14.10.5 Configure the NCC Gateway Group Offset ......................... 217

15 Bay-level Gateway Engineering ................................................. 220
15.1 Overview .............................................................................. 220
15.2 SPA-ZC40x .......................................................................... 220

16 Data Import/Export ................................................................. 221
16.1 ICD/IID File Exchange within a Project ................................. 221
16.1.1 Import an ICD/IID File to Update IEDs .............................. 221
16.1.2 Export an ICD/IID File ...................................................... 221
16.2 SCD File Exchange Within a Project ..................................... 222
16.2.1 Import an SCD File to Create a Project ......................... 222
16.2.2 Import an SCD File to Update IEDs ............................... 223
16.2.3 Export an SCD File ......................................................... 223
16.2.4 Export an SCD File, Mixed Systems ............................... 224
16.3 SSD File Import ................................................................. 226
16.3.1 Overview ...................................................................... 226
16.3.2 Substation Specification .................................................. 227
16.3.3 LN Mapping Specification ............................................... 227
16.3.4 Data Type Templates Specification ............................... 227
16.3.5 IED Specification ........................................................... 228
16.3.6 Subnetwork Specification ............................................... 228
16.3.7 Import an SSD File ......................................................... 228
16.4 SED File Exchange Between Projects ................................. 230
16.4.1 Introduction .................................................................. 230
16.4.2 Basic Mechanism ............................................................. 230
16.4.3 Pre-Conditions for SED Data Exchange ........................ 231
16.4.4 IET600 behavior in SED Data Exchange ......................... 232
16.4.5 Good Engineering Practices in SED Data Exchange .......... 232
16.4.6 Example Transformer Interlocking via GOOSE ............... 233
16.4.7 Step 1: Export of an SED File in Project A ..................... 234
16.4.8 Step 2: Import of SED File into Project B ....................... 237
16.4.9 Step 3: Add Dataflow to Foreign IEDs in Project B ........... 238
16.4.10 Step 4: Re-Export Foreign IEDs and own IEDs from Project B 239
16.4.11 Step 5: Import of SED File into Project A ..................... 240

IET600 Integrated Engineering Tool
User Manual
# Table of contents

16.4.12 Step 6: Add Dataflow to Foreign IEDs in Project A .................. 240  
16.4.13 Step 7: Re-export Foreign IEDs from Project A ................. 241  
16.4.14 Step 8: Re-import of SED File into Project B ................. 242  
16.4.15 IET600 SED File Content ........................................... 244  
16.5 HMI Status Texts Exchange ........................................... 245  
16.5.1 Format of Status Text Files ......................................... 245  
16.5.2 Import Status Texts .................................................. 246  
16.5.3 Export Status Texts .................................................. 246  
16.6 HMI Signal Information Exchange ................................... 247  
16.6.1 Overview .................................................................. 247  
16.6.2 Exchange File Formats .............................................. 247  
16.6.3 Import Behavior of IET600 ......................................... 248  
16.6.4 Import Considerations .............................................. 249  
16.6.5 Import a MicroSCADA 'lof' File .................................. 249  
16.6.6 Import a MicroSCADA 'sfx' File .................................. 251  
16.6.7 Import a MicroSCADA 'csv' File with Process Object Data ...... 254  
16.6.8 Import a MicroSCADA 'sasmc' File ............................... 257  
16.6.9 Import Signal Overview ............................................. 259  
16.6.10 MicroSCADA Comparison Preferences dialog .................. 261  
16.6.11 Export a MicroSCADA 'lof' File .................................. 261  
16.6.12 Export a MicroSCADA 'sfx' File .................................. 263  
16.6.13 Export a MicroSCADA 'csv' file with Process Object Data ...... 264  
16.6.14 Export a MicroSCADA 'sasmc' file ............................... 264  
16.7 NCC Gateway Signal Information Exchange ....................... 267  
16.7.1 Overview .................................................................. 267  
16.7.2 Export COM500 Configuration File (XRF) ....................... 267  
16.7.3 Export COM500 Configuration File (XRF) ....................... 268  
16.7.4 Export Configuration Data for Other NCC Gateways ........... 268  

**17** Customer Signal Lists ........................................................................ 269  

17.1 Introduction .................................................................................. 269  
17.2 Customizing a Template .................................................................. 269  
17.2.1 Copy the Default Template into Your Project Folder ............... 270  
17.2.2 Customize the Template ..................................................... 270  
17.3 Customizing Filtering, Grouping, Sorting and Formatting .......... 271  
17.3.1 Open the 'Export Customer Signal List' Dialog ....................... 271  
17.3.2 Configure Filtering .......................................................... 273  
17.3.3 Configure Grouping ......................................................... 276  
17.3.4 Configure Sorting ............................................................ 277  
17.3.5 Configure Formatting ....................................................... 279  
17.4 Export a Customer Signal List ..................................................... 283  
17.4.1 First Export ..................................................................... 283  
17.4.2 Update an Existing Customer Signal List ............................. 285  
17.4.3 Editing exported Customer Signal List ............................... 286  
17.5 Import a Customer Signal List .................................................... 287  

**18** Protecting IEDs Against Changes ..................................................... 289  

18.1 Introduction .................................................................................. 289  
18.2 Applying Protection ..................................................................... 289  
18.2.1 General Principles ............................................................. 289  
18.2.2 Configuring and Editing Protection ...................................... 290
# Table of contents

18.2.3 Displaying Protection ......................................................... 291
18.2.4 RCB Dataflow Protection .................................................... 291
18.2.5 GCB Dataflow Protection .................................................... 291
18.2.6 SVC Dataflow Protection .................................................... 292
18.2.7 HMI Engineering Protection ................................................. 292
18.3 Engineering Workflow and Protection ......................................... 293
18.3.1 GCB Dataflow Protection After FAT ......................................... 293
18.3.2 GCB Dataflow Protection of Third-Party Engineered IEDs .......... 293
18.3.3 Commissioning Bays .......................................................... 294
18.3.4 Protecting HMIs While Engineering a Gateway ..................... 294

19 Migration of Engineering Data from Older Projects .. 295
19.1 Overview .............................................................................. 295
19.2 Migration Effort Considerations ............................................. 295
19.2.1 Scenarios That Will Profit from Migration to IET600 ............... 295
19.2.2 Scenarios to be Evaluated for Migration to IET600 ............... 295
19.2.3 Scenarios not Very Suitable for Migration to IET600 .......... 296
19.3 Migration of Systems of Different Types .................................. 296
19.3.1 Existing IEC 61850-based System with MicroSCADA Station Level ....................................................... 296
19.3.2 Existing Non-IEC 61850 System with MicroSCADA Station Level .......................................................... 296
19.3.3 Integration of Non-61850 IEDs into a 61850 Substation .......... 297
19.4 Migration Step Details .......................................................... 297
19.4.1 Prerequisites ................................................................. 297
19.4.2 Import SCD file ................................................................. 298
19.4.3 Create HMIs and Gateways .................................................. 298
19.4.4 Create NCCs ............................................................... 298
19.4.5 Import HMI and COM500 Configuration .............................. 299

20 Diagnostics ........................................................................... 300

21 Annex: Naming Conventions ..................................................... 304
21.1 IEC 81346 .................................................................. 304
21.1.1 Introduction ................................................................. 304
21.1.2 General Conventions in IEC 81346 .................................. 304
21.1.3 Hierarchies in IEC 81346 and IEC 61850 .......................... 304
21.2 Default Substation Naming .................................................... 305
21.2.1 Typical Bay, Line .......................................................... 305
21.2.2 Typical Bay, Transformer ................................................. 306
21.2.3 Typical Bay, Bus Coupler ................................................. 306
21.2.4 Typical Diameter, Line/Transformer .................................. 307

22 Annex: Switch Configuration ..................................................... 308
22.1 The Switch Data Model ......................................................... 308
22.2 MAUs .............................................................................. 308
22.2.1 Short description ............................................................. 308
22.2.2 References ............................................................... 309
### Table of contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>23</td>
<td>Annex: MicroSCADA OI Settings</td>
<td>310</td>
</tr>
<tr>
<td>23.1</td>
<td>Setting the OI Attribute</td>
<td>310</td>
</tr>
<tr>
<td>23.2</td>
<td>OI Settings on Project</td>
<td>310</td>
</tr>
<tr>
<td>23.2.1</td>
<td>Default Settings</td>
<td>311</td>
</tr>
<tr>
<td>23.2.2</td>
<td>Configuration</td>
<td>311</td>
</tr>
<tr>
<td>23.3</td>
<td>OI Settings for MicroSCADA</td>
<td>312</td>
</tr>
<tr>
<td>23.3.1</td>
<td>Configuration</td>
<td>312</td>
</tr>
<tr>
<td>23.4</td>
<td>Known Limitations</td>
<td>312</td>
</tr>
<tr>
<td>24</td>
<td>Annex: Mirroring Command Group Logic</td>
<td>313</td>
</tr>
<tr>
<td>24.1</td>
<td>Introduction</td>
<td>313</td>
</tr>
<tr>
<td>24.2</td>
<td>Predefined Mirroring Group</td>
<td>313</td>
</tr>
<tr>
<td>25</td>
<td>Annex: NCC Command Group Logic</td>
<td>315</td>
</tr>
<tr>
<td>25.1</td>
<td>Introduction</td>
<td>315</td>
</tr>
<tr>
<td>25.2</td>
<td>Predefined Command Groups</td>
<td>315</td>
</tr>
<tr>
<td>25.2.1</td>
<td>Group of 5 Fixed Commands with 1 Fixed Feedback Indication</td>
<td>316</td>
</tr>
<tr>
<td>25.2.2</td>
<td>Group of 1 Fixed Command with 1 Fixed Feedback Indication</td>
<td>317</td>
</tr>
<tr>
<td>25.2.3</td>
<td>Group of 2 Fixed Commands with no Feedback Indication</td>
<td>317</td>
</tr>
<tr>
<td>25.2.4</td>
<td>Group of 4 Fixed Commands with 1 Fixed Feedback Indication</td>
<td>318</td>
</tr>
<tr>
<td>25.3</td>
<td>Flexible Command Groups</td>
<td>318</td>
</tr>
<tr>
<td>26</td>
<td>Annex: IED Capabilities</td>
<td>319</td>
</tr>
<tr>
<td>26.1</td>
<td>Introduction</td>
<td>319</td>
</tr>
<tr>
<td>26.2</td>
<td>Capabilities Description</td>
<td>319</td>
</tr>
<tr>
<td>26.3</td>
<td>Common Corrections with IED Capabilities</td>
<td>326</td>
</tr>
<tr>
<td>26.3.1</td>
<td>Service ConfReportControl, Max attribute</td>
<td>326</td>
</tr>
<tr>
<td>26.3.2</td>
<td>Service ConfDataset, maxAttributes attribute</td>
<td>327</td>
</tr>
<tr>
<td>26.3.3</td>
<td>Service ConfDataset, Modify attribute</td>
<td>327</td>
</tr>
<tr>
<td>26.3.4</td>
<td>Client Services</td>
<td>327</td>
</tr>
<tr>
<td>26.3.5</td>
<td>Dataflow Cannot be Created or Edited at All</td>
<td>328</td>
</tr>
<tr>
<td>27</td>
<td>Annex: IET600 and IEC 61850 Edition Handling</td>
<td>330</td>
</tr>
<tr>
<td>27.1</td>
<td>IEC 61850 Classification</td>
<td>330</td>
</tr>
<tr>
<td>27.1.1</td>
<td>IEC 61850 Edition, Version and Revision</td>
<td>330</td>
</tr>
<tr>
<td>27.1.2</td>
<td>IEC 61850 Namespaces</td>
<td>330</td>
</tr>
<tr>
<td>27.1.3</td>
<td>SCL Elements Conflicting With SCL Namespace Version 2003</td>
<td>331</td>
</tr>
<tr>
<td>27.1.4</td>
<td>IED DATA Conflicting With Namespace Version 2003</td>
<td>332</td>
</tr>
<tr>
<td>27.2</td>
<td>IEC 61850 Compatibility Issues</td>
<td>333</td>
</tr>
<tr>
<td>27.2.1</td>
<td>SCL Namespace compatibility Problems</td>
<td>333</td>
</tr>
<tr>
<td>27.2.2</td>
<td>Name Length Problems</td>
<td>333</td>
</tr>
<tr>
<td>27.2.3</td>
<td>Dataflow Compatibility Problems</td>
<td>333</td>
</tr>
<tr>
<td>27.2.4</td>
<td>CDC compatibility problems</td>
<td>333</td>
</tr>
<tr>
<td>27.3</td>
<td>Mixed Systems, Engineering Scenarios</td>
<td>334</td>
</tr>
<tr>
<td>27.3.1</td>
<td>Extension of an Existing Substation with Edition 1 IEDs</td>
<td>334</td>
</tr>
<tr>
<td>27.3.2</td>
<td>Mixed Substation Specified by Customer</td>
<td>335</td>
</tr>
<tr>
<td>27.4</td>
<td>IET600 Mixed System Engineering</td>
<td>336</td>
</tr>
<tr>
<td>27.4.1</td>
<td>Put IEDs of Different Versions Into Different Subnetworks</td>
<td>336</td>
</tr>
</tbody>
</table>
# Table of contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>27.4.2</td>
<td>SCD Export</td>
<td>337</td>
</tr>
<tr>
<td>27.4.3</td>
<td>SCD files Version 2007A</td>
<td>338</td>
</tr>
<tr>
<td>27.5</td>
<td>IET600 Versioning of IEC 61850 Items</td>
<td>339</td>
</tr>
<tr>
<td>27.5.1</td>
<td>Versioning Based on IED DATA Namespaces</td>
<td>339</td>
</tr>
<tr>
<td>27.5.2</td>
<td>Versioning Based on XML Namespace</td>
<td>339</td>
</tr>
<tr>
<td>27.5.3</td>
<td>Versioning of Switch IEDs</td>
<td>339</td>
</tr>
<tr>
<td>28</td>
<td>Annex: Inputs Predefined from IED (Late Binding)</td>
<td>341</td>
</tr>
<tr>
<td>28.1</td>
<td>Introduction</td>
<td>341</td>
</tr>
<tr>
<td>28.2</td>
<td>Late Binding SCL Attributes</td>
<td>341</td>
</tr>
<tr>
<td>28.3</td>
<td>Late Binding Workflow</td>
<td>342</td>
</tr>
<tr>
<td>28.3.1</td>
<td>IEC 61850 Intended Workflow</td>
<td>342</td>
</tr>
<tr>
<td>28.3.2</td>
<td>IEC 61850 Late Binding Mapping Rules</td>
<td>343</td>
</tr>
<tr>
<td>28.4</td>
<td>Configuration of Late Binding in the System Configuration Tool</td>
<td>344</td>
</tr>
<tr>
<td>28.4.1</td>
<td>Implementation in IET600</td>
<td>344</td>
</tr>
<tr>
<td>29</td>
<td>References</td>
<td>345</td>
</tr>
<tr>
<td>30</td>
<td>Glossary of Terms</td>
<td>346</td>
</tr>
</tbody>
</table>
1 Preface

1.1 IET600, a System Engineering Tool

ABB’s Integrated Engineering Toolbox (IET600) is a system engineering tool, designed to engineer Substation Automation systems (with emphasis on but not exclusively for IEC-61850-based Systems).

IET600 provides:

- Engineering of IEC-61850-Systems (IET600 is a System Configuration tool as defined by IEC 61850):
  - Simple integration of IEC-61850-based ABB and third-party devices that follow the IEC 61850 standard.
  - High scalability from small substations to large switchyards.
  - Use of the standardized IEC 61850 Substation Configuration Language (SCL) to exchange data with other engineering tools.
  - Creation of Single Line Diagrams and Communication Configuration Diagrams.
  - IEC-61850-compatible extensions to allow an enhanced modelling of complex communication configurations.
- Engineering of ABB MicroSCADA systems (SYS600/C, COM500).
- Database engineering for multiple systems (including hot-standby systems and database mirroring).
- COM500 Gateway engineering.
1.2 Organization of the Manual

This manual is organized as follows:

- Chapter 2 introduces some new concepts which make engineering considerably more efficient. Read this to understand and take full advantage of the efficiency-enhancing features of IET600.
- Chapter 3 introduces the IET600 user interface. Read this to understand how to access and edit your data in IET600 and to learn about GUI features available in most of the editors.
- Chapter 4 gives a short overview of how IET600 organizes and stores data. Read this to get an overview how to backup and restore your data and how to exchange IET600 project data.
- Chapter 5 describes different engineering workflows, read this to understand which tasks can be executed in parallel to gain time during engineering and to decide which workflow is suitable for your project.
- Chapters 6 to 10 describe the engineering steps related to IEC 61850 engineering. Read any of these chapters to get detailed information about using IET600 as an IEC 61850 System configuration tool.
- Chapter 11 describes the engineering of customer names, relevant for both IEC 61850 engineering and MicroSCADA database engineering.
- Chapters 12 to 14 describe the engineering steps related to MicroSCADA database engineering. Read any of these chapters to get detailed information about using IET600 as a MicroSCADA database configuration tool.
- Chapter 15 deals with a few special situations regarding Bay-level Gateways.
- Chapter 16 describes the Data Exchange between IET600 and other tools with different data formats.
- Chapter 17 describes how to customize and export Customer Signal Lists.
- Chapter 18 describes how IEDs may be protected against further changes; a feature useful in Commissioning.
- Chapter 19 describes how to proceed when migrating a project from another environment to IET600. Read this chapter to understand how old projects can be migrated and to decide which workflow is suited best to your specific migration.
- Chapter 20 describes how to analyze and detect inconsistencies inside the project.
- Chapters 21 to 28 are Annexes that provide in-depth information about some areas.
1.3 Use of This Manual for Lookups

This manual has grown thick enough that it becomes laborious to read it completely. Many users will consult the manual once they face a problem. In such a case there are some ways to efficiently find the information you are looking for.

1. Search for the word of what you actually have problems with (e.g. a menu item, an IED Service etc.), e.g. “ConfReportControl.Max”
2. Search for a more generic word, but still related to what you actually have problems with, e.g. “ConfReportControl”.
3. Search for a word related to the engineering step you want to perform, e.g. “RCB Client”
4. Look at the index to find a title related to the engineering step you want to perform.
Chapter 2

Important concepts in IET600

2.1 Parallelization of Tasks

System engineering is often considered as a linear process starting at base design and ending with commissioning of the system. In practice, however, situations frequently occur where engineering tasks need to be done in parallel and merged later. Common examples of such situations include:

- Station-level engineering is required early in the project, before the bay-level engineering has begun (e.g., a preliminary SAS database is needed for SCADA engineering or a signal list for discussions with the customer).
- Different project parts are engineered by different people (e.g. bay-level engineering and station-level engineering are often assigned to two different engineering groups working in parallel to reduce project execution time).
- When changes and corrections are needed, resulting in re-engineering of some parts of the project (for example, a commissioning engineer needs to implement corrections after an acceptance test).

IET600 supports parallel workflows by decoupling engineering tasks within the tool itself, allowing the system engineer to work flexibly and merge these various parts at a later stage. Engineering of the station level is decoupled from the IEC 61850 communication engineering, reducing the impact of changes considerably.

2.2 Naming of Objects

Technical projects very often have the difficulty that the manufacturer has one naming convention and the customer has another one. The customer needs his naming scheme for efficient operation; the manufacturer needs his own scheme for efficient commissioning and also to be able to provide efficient support and in case of expansion projects later.

Additionally, the IEC 61850 standard imposes certain restrictions for some objects (e.g. for IED names), while allowing a lot of freedom in other respects (e.g. objects in the Substation section).

To provide a solution to both parties, IET600 provides a dual naming scheme. For all objects and for signal texts, IET600 provides an “Internal Name” or “Internal Text” (which belongs to ABB) and a “Customer Name” or “Customer Text” (which belongs to the customer). User Interface elements can be configured to display either one or the other, or sometimes both, if required.
2.2.1 Internal Names

Internal names of any object may contain only characters A-Z and a-z and digits 0-9. Furthermore, ABB adheres to the naming standard as defined in IEC 81346 which imposes further restrictions (for a more detailed description, see chapter 21, Annex: Naming Conventions). This convention is applied for the naming of all Substation objects as well as for IEDs and IEC 61850 Subnetworks.

These internal object names are used as “Name” attributes in the IEC 61850 data model. Therefore, they typically need to be unique (e.g. IED names in a project need to be unique, Bay names within a Voltage Level need to be unique etc.).

2.2.2 Customer Names

Customer names of any object, on the other hand, may contain any Unicode characters and have no restrictions whatsoever. They need not be hierarchical (however, a hierarchical naming scheme is recommended whenever possible as it allows many advantages e.g. in renaming.)

These Customer object names are exported as “desc” (Description) attributes in the IEC 61850 data model.

Substation automation is sometimes just a part of a bigger plant (e.g. a power generation station or a refinery) which imposes other naming conventions. If those naming conventions are not compatible with IEC 81346, the Customer name needs to be used to implement those names.
3 IET600 User Interface

This chapter gives an overview of the IET600 graphical user interface (GUI), describes its different parts and describes general functionality that is provided throughout IET600. For a specific description of a functionality or editor, please read chapters 4.2 to 18.

3.1 GUI Overview

The IET600 user interface has the following distinct sections:

1. Main menu
2. Editors
3. Navigation Trees
4. Properties
5. Logging
6. Status bar
### Main Menu

The Main menu gives the user an overview of engineering tasks. The Main menu is organized in a series of tabs, each related to a particular task:

<table>
<thead>
<tr>
<th>Tab</th>
<th>Menu items</th>
</tr>
</thead>
<tbody>
<tr>
<td>Home</td>
<td>Display-related menu items (adjust windows), often used functions (SCD Export/import)</td>
</tr>
<tr>
<td>Substation</td>
<td>Menu items related to Substation configuration (create SS, Voltage Levels etc.)</td>
</tr>
<tr>
<td>IEDs</td>
<td>Menu items related to IED configuration (import, create and configure IEDs)</td>
</tr>
<tr>
<td>Communication</td>
<td>Menu items related to Dataflow configuration (create Subnetworks, create Datasets and Control Blocks, configure Control Block Clients)</td>
</tr>
<tr>
<td>HMI</td>
<td>Menu items related to HMI, Gateway and NCC configuration (creation, import/export of lists related to their configuration).</td>
</tr>
</tbody>
</table>

Using the Main menu is intuitive: select the tab related to the task you want to perform, and then click on the menu item:

Some tabs are “context-dependent”, i.e. they are shown only, if a certain Editor is selected:

<table>
<thead>
<tr>
<th>Tab</th>
<th>Tasks specific for the Editor</th>
</tr>
</thead>
<tbody>
<tr>
<td>Single Line Diagram</td>
<td>Tasks specific for the SLD Editor</td>
</tr>
<tr>
<td>CCD</td>
<td>Tasks specific for the CCD Editor</td>
</tr>
</tbody>
</table>

When selecting any function in the Main menu, IET600 will automatically select a Navigation Tree and an Editor best suited to this task. This is intended to support a user who is not too familiar with the engineering procedure. Experienced engineers will usually prefer to select their own Navigation Tree and editor and perform tasks from the context menus of elements.

Some functionality, mainly import and export of different files, are only available from the Main menu.
3.1.2 Navigation Trees

Navigation Trees display objects in different hierarchies. Selecting a particular object in a Navigation Tree shows data of this element and lower elements in the currently selected editor.

The following Navigation Trees are available:

<table>
<thead>
<tr>
<th>Substation</th>
<th>displays the substation topology, primary equipment and IEDs; roughly based on the Substation section as defined by IEC 61850-6</th>
</tr>
</thead>
<tbody>
<tr>
<td>IEDs</td>
<td>displays all IEDs in a flat structure</td>
</tr>
<tr>
<td>Communication</td>
<td>displays the Subnetworks and their attached IEDs; based on the Communication section as defined by IEC 61850-6</td>
</tr>
<tr>
<td>HMIs</td>
<td>displays HMIs, Gateways, NCCs and their relations; displays also mirroring configurations</td>
</tr>
</tbody>
</table>

The following Navigation Tree is also available, but hidden by default, as it is rarely needed:

| Data Type Templates         | displays the Data Type Templates of all IEDs  
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>The Data Type Templates</td>
<td>contain the definition of the exact structure of specific Logical Nodes (LNs) of specific IEDs. They may be needed in rare cases to find errors, but their analysis requires in-depth knowledge of IEC 61850 LN classification.</td>
</tr>
</tbody>
</table>
The following possibilities are available:

3.1.3 Editors

The editors are the place where the user actually configures the data of a project. An editor will display the data related to the element selected in the Navigation Tree and all its lower elements. As not all elements are suitable for all editors, an editor will display an appropriate message, if it cannot display the element selected in the Navigation Tree.

To use a specific editor, select its tab:
To change to another editor, click on its tab.
Alternatively, right-click on any Tab -> a popup menu with available editors will show:

Select the editor tab which you want to use -> this editor will open. This is a useful way to navigate rapidly between editors at the beginning and at the end of the tabs.

Each editor provides detailed views related to various engineering tasks:

<table>
<thead>
<tr>
<th>Editor</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>IED Status</td>
<td>Display IED status information (Import, Export, Versions, Engineering protection), configure engineering protection.</td>
</tr>
<tr>
<td>IED Settings</td>
<td>Configure IED-related information (Disturbance Recording, PCM600 Templates, SNMP Access Parameters).</td>
</tr>
<tr>
<td>SLD</td>
<td>Edit Single Line Diagram, configure LN Mapping.</td>
</tr>
<tr>
<td>CCD</td>
<td>Edit Communication Configuration Diagram, configure physical communication information.</td>
</tr>
<tr>
<td>Naming</td>
<td>Configure Internal and Customer Names for Substation objects, IEDs, Subnetworks etc.</td>
</tr>
<tr>
<td>Subnetworks</td>
<td>Configure IED-to-Subnetwork relations and communication address information.</td>
</tr>
<tr>
<td>Signal Clients</td>
<td>Map HMI signals from IEDs to HMI and GW clients, configure mirrored signals.</td>
</tr>
<tr>
<td>HMI Data</td>
<td>Configure HMI signals.</td>
</tr>
</tbody>
</table>
### Editor | Function
--- | ---
NCC Data | Configure NCC signals.
Datasets | Configure Datasets and their content.
RCB Data | Configure RCBs.
RCB Clients | Map client IEDs to RCBs.
LCB Data | Configure RCBs (hidden by default).
GCB Data | Configure GCBs.
GCB Clients | Map client IEDs to GCBs.
SVCB Data | Configure SVCBs.
SVCB Clients | Map client IEDs to SVCBs.
Inputs | View and edit IED inputs.
LN Data | View IEC 61850 Data one IED at a time, edit selected attributes.
IED Services | View IEC 61850 Services related to Dataflow and their relation to IED Capabilities (hidden by default).
IED Capabilities | View IED Capabilities (hidden by default).

IET600 can be used as a pure System Configuration Tool for IEC 61850 Dataflow; or it can be used to configure HMI- and NCC-related functionality. To optimally adapt to the intended use, it is possible to show/hide the following editors:

- Signal Client
- HMI Data
- NCC Offset
- NCC Data

This can be done from the Main menu -> 'HMI' tab -> 'Show/Hide HMI Editors':

#### 3.1.4 Properties

This pane shows detailed properties for the element selected in the Navigation Tree. Most of the properties can be edited more easily in the editor designed for this task. Some few properties need to be edited in the ‘Properties’ pane.
3.1.5 Logging

The Logging section displays a log of events to support analysis and trouble-shooting of the project during engineering. It contains two tabs to display logged events:

‘Output’
‘Error list’

The ‘Output’ list has the following features:

- It serves as a permanent log, contains all log messages time-tagged and in chronological order.
- All messages that are logged into the Output tab are also written into a user-specific log file (which is located in C:\Users\<user name>\AppData\Roaming\ABB\IET600 5.3.19\Logfiles and can be accessed directly from the context menu, see below)
- It has a context menu with the following menu items:
  - ‘Clear Output’ clears the output list.
  - ‘Copy selected text’ / ‘Copy all text’ copies selection or whole output content to the copy buffer.
  - ‘Word wrap’ wraps lines that do not fit in the editor.
  - ‘View logfile’ opens the log file mentioned above.

The ‘Error List’ tab has the following features:

- It contains errors, warnings and infos related to an action performed.
- It is cleared automatically before every bigger action so that the user should see only the logs related to the most recent action.
- It should therefore be consulted immediately after every action performed.
- It allows filtering on errors, warnings or infos or any combination (default is to show all of them).
3.1.6 Status Bar

As in most applications the Status bar provides some additional information. Relevant in IET600 are:

- A **Progress bar** shows the action description and progression of actions that take a longer time, e.g. opening or saving a project, or updating an IED from an SCL file.

- A ‘**Selection’ indicator** shows (in table-based editors only) the visible and the total rows (thereby providing an indicator about filters)
  It also shows the current selection: rows, cells or columns, and how many.

- An ‘**Update’ indicator** shows that an update for your IET600 is available.

A click on the indicator will open a dialog from which you can install the update directly (this dialog also pops up if you start IET600 and an available Update is discovered):

To prevent this dialog from popping up again, you may select ‘Don’t notify me again for this version’; this may be useful if you are in a critical phase of a project and do not want to update right now. The update indicator will still be be available

Alternatively, you can update IET600 from Options -> Updates (as described in chapter 3.4.3).
3.2 'Table'-based Editors

Most editors in IET600 show data in the form of tables. Each editor provides the following:

- a row context menu
- a column context menu
- a cell context menu
- optionally: column tool tips

IET600 provides the following general functionality for table-based editors:

- show/hide columns
- sorting on (multiple) columns
- selecting whole columns
- filtering the data based on filter criteria in multiple columns

3.2.1 Editor Elements

In the remainder of the document, we will use some terms to designate specific elements of an editor. The following overview shows these elements and their respective names:
3.2.2 Context Menus

As with context menus in general, you can open them by right-clicking into the appropriate area of the table:

- Right-clicking on the column header shows the column context menu.
- Right-clicking into any row header shows the row context menu (if no rows are shown, right-clicking anywhere into the empty space below the column headers will also bring up the row context menu).
- Right-clicking into any cell shows the row context menu.

The column context menu will usually contain the following menu items (individual editors may contain additional menu items or modify them):

- ‘Set to “Sort” mode’/‘Set to “Select” mode’ determines whether clicking on the column header will sort this column or select the column. Default behavior is “Sort”.
- ‘Show only rows with warnings’ will set a filter so that only rows with warnings (i.e. rows that contain a red error mark) are shown. This is helpful in some editors to systematically go through all rows that contain any error. It is different from the column filter that shows rows with errors in one specific column. If this menu item is disabled, it means that there are no rows with warnings in the current editor for the element currently selected in the Navigation Tree.
- ‘Reset all filters’ clears all existing filters, including the ‘Show only rows with warnings’ (the same functionality as clicking on the button in the Filter row header). The menu item is disabled, if there are no filters currently set in an editor.
- ‘Scroll to column’ allows rapidly finding a column by scrolling to it, and by making it visible if it was hidden. This is particularly useful in editors with many columns (e.g. ‘HMI Data’) or rarely used hidden columns (e.g. RCB editor).
This function is not available for dynamic columns, e.g. client columns in the miscellaneous ‘XXX Clients' editors.

- ‘Reset Table layout’ sets back column sequence, visibility, width and sorting to a default, clear any grouping and clears all filters.

The **row context menu** will usually contain the menu items ‘Insert New Row’ (i.e. create a new element) and ‘Delete Rows’ (remove one or several elements). These menu items are not available in editors that do not allow creation or deletion of objects, e.g. the ‘IED Status’ editor.

The **cell context menu** will usually contain the menu items ‘Copy’, ‘Cut’ and ‘Paste’ which are probably familiar to anybody. Additionally it contains the following menu items:

- ‘Fill Down’ is designed to rapidly fill a number of cells in one single column with the same values. Mark the cells you want to fill first, then right-click to get the context menu and select ‘Fill Down’.

  The value used to fill the cells is the one of the first cell selected. Columns with numerical values have the additional feature that IET600 will take the difference of the first two cells and propagate this difference to all other cells. To fill a numerical column with identical values, ensure that the first two cell values are the same.

  To rapidly fill down a big number of cells (several thousands), mark the first cell and then use Ctrl+Shift+PgDn (or Ctrl+Shift+PgUp), this is considerably faster than using mouse dragging to mark cells.

- ‘Find and replace’ calls a dialog that allows finding all cells in a column containing a specific value and optionally replacing it with another user-defined value.

  ‘Find and replace’ works only on the rows in the editor, not on the whole project.

- ‘Export to Excel’ allows exporting the contents visible in a table-based editor into an Excel sheet. A re-import is not possible.

### 3.2.3 Show/Hide Columns

To show or hide columns:
1. click on the top left icon of the editor:

![Editor Icon]

2. The ‘Field Chooser’ dialog will appear, select the columns to be shown or hidden by checking/unchecking their respective selection boxes:

![Field Chooser]

3.2.4 Sort on Columns

IET600 allows to sort on single or multiple columns. The current sorting is indicated by small arrows in the header, the example shows an editor which is sorted by ascending ‘Station Text’ and then ascending ‘Signal Text’

- To sort on a single column, left-click on its header -> any existing sorting will be cleared and the data in the editor will be sorted ascending by that column.
- To reverse the sort order of a single-sorted column, left-click on a sorted column -> its sort order will reverse.
• To sort on multiple columns, left-click on the header of the first column on which you want to sort (this will clear any existing sorting)

Then press SHIFT and left-click on additional columns in the sequence that the sorting shall be applied (once to add a column with ascending sort, twice for descending) -> the data in the editor will be sorted ascending by that column.

You can add columns any time by pressing SHIFT and left-clicking. However, the editor does not provide any feedback about the sort order of the already selected columns (e.g. in the above example, you do not know whether it is sorted by ascending ‘Station Text’ and then ascending ‘Signal Text’ or vice versa).

• To reverse the sort order of a multiple-sorted column, you cannot simply left-click on a sorted column. To do so will not only reverse the sort order of this column, but also put it at the end of the sorting sequence. E.g. the above example is sorted by ascending ‘Station Text’ and then ascending ‘Signal Text’. If you press SHIFT and left-click ‘Station Text’, the new sort order will be ascending ‘Signal Text’ and then descending ‘Station Text’.

• You cannot clear a sorting; you can only sort on another column.

### 3.2.5 Three State Checkboxes

In various places, IET600 is using 3-state checkboxes, which can have the values:

<table>
<thead>
<tr>
<th>undefined</th>
<th>unchecked</th>
<th>checked</th>
</tr>
</thead>
</table>

When clicking on a checkbox, it toggles its state from 'checked' to 'unchecked' and vice versa.

To bring it into the 'undefined' state, select the checkbox and press the 'Delete' key.

### 3.2.6 Set and Clear Filters

There are two different, mutually exclusive ways in IET600 how to access filters. You can toggle between these two ways of accessing filters in Options -> Appearance (for a detailed description, see chapter 3.4.1, Options -> Appearance).

1. via Filter Symbol in the column header:

The Filter Symbol allows you to select a Filter from a combo box that appears when you click on it. To create a complex filter, select “Custom” and specify your Filter conditions.
2. via Filter Row:

The Filter Row allows a more selective and rapid approach:

- The ‘Filter logic’ button on the left determines how the following condition is evaluated. The default is “Starts with”, click on the button to change this value.

- The adjoining text/combo box allows you to type the filter condition directly. Alternatively, you can also select any of the options available from the combo box.

- The ‘Clear filter’ button on the right allows you to clear an existing filter for one column.

- The ‘Clear all filters’ button in the row header of the filter row allows you to clear all existing filters for all columns (the same can be done with the ‘Reset all filters’ menu item from the column context menu).

The Combo box offers the following choices

<table>
<thead>
<tr>
<th>Choice</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>(All)</td>
<td>shows all rows (reset filter)</td>
</tr>
<tr>
<td>(Custom)</td>
<td>opens a dialog to configure customized filters</td>
</tr>
<tr>
<td>(Blanks):</td>
<td>shows all rows with no content in this column</td>
</tr>
<tr>
<td>(NonBlanks):</td>
<td>shows all rows with content in this column</td>
</tr>
<tr>
<td>(Errors)</td>
<td>shows all rows with an error in this column</td>
</tr>
<tr>
<td>(NonErrors)</td>
<td>shows all rows with no errors in this column</td>
</tr>
<tr>
<td>&lt; value &gt;</td>
<td>shows all rows with matching value (the list includes all different values found in that column)</td>
</tr>
</tbody>
</table>

As a special filter, ‘Show only rows with warnings’ is available from the column context menu. It will cause all rows with warnings (i.e. rows that contain a red error mark) to be shown (which is different from the column filter that shows rows with errors in one specific column).

A quick way of checking for set filters is the ‘Reset all filters’ menu item from the column context menu: if it is not disabled than a filter of some sort is active.

If you hide a column that has a filter set, this filter still applies after you have hidden the column.
Filters are saved when you close a project or close IET600. So if you set a filter in one project, then close it and open another project, the filter will apply to the data of the new project (which may be desirable or not).
3.3 GUI and Engineering Workflow

For an individual engineering step, you have two ways to execute a task:

- via Main menu, a method suited to users less familiar with IET600
- by choosing Navigation Tree, edited element and editor yourself, a method suited to users familiar with IET600 and the engineering process

As a rough rule for users yet unfamiliar with IET600, tasks in a basic engineering process are arranged in the Main menu tabs from left (steps early in the engineering process) to right (steps late in the engineering process).

After a while you will want to take advantage of the parallelization of tasks. Consult chapter 5 (Engineering Workflow) for a more sophisticated overview of possible engineering workflows.

3.3.1 Execute a Task via Main Menu

As a rough rule, tasks in a basic engineering process are arranged in the Main menu tabs from left (steps early in the engineering process) to right (steps late in the engineering process).

1. Select the Main menu tab that fits your engineering step best.
2. Select the task you want to perform in this tab. IET600 will then automatically select a Navigation Tree and an Editor that is suited to your task.

After a while, you will get the feeling which Navigation Trees and editors are best used for what task and you can proceed as described below.

After a while you will also want to take advantage of the parallelization of tasks. Consult chapter 5 (Engineering Workflow) for a more sophisticated overview of possible engineering workflows.

3.3.2 Execute a Task via Navigation Tree and Editor

Select the Navigation Tree that suits your task:

- ‘Substation’ for most editing
- ‘IEDs’ for task related specifically to one IED
- ‘Communication’ for some communication-related tasks
- ‘HMIs’ to create/reconfigure HMIs, Gateways and NCCs

Select the element in the Navigation Tree you want to edit; consider the following:

- A higher element allows you to compare data from different sub-elements in the Editor (e.g. selecting the project will display the data of all IEDs, allowing you to compare already configured IEDs with not yet configured ones).
• A lower element shows more specific data; the editors will show the data faster and there is less danger that you accidently edit a wrong element. Select the editor appropriate to your task.

3.4 Options

To access the user options, click the round button on the top-left corner of IET600, then click the 'Options' button:

The 'Options' dialog will appear, offering you several Sub-Dials where you can configure settings.

3.4.1 Options -> Appearance

This section contains options that affect the appearance of IET600. They are more or less self-explanatory.

As application language, only “English” is currently available.
3.4.2 Options -> User Settings

This section allows to set user-specific file paths and activate a history, if required. For a more detailed explanation of these paths, please consult chapter 4 (IET600 Data Organization).

3.4.3 Options -> Updates

This section allows to check for IET600 Updates. Click on the ‘Check for Updates’ button. If an update is available, you can download it and install it at your convenience.
For a more detailed explanation about IET600 updates, see document “IET600 Installation Manual”, chapter “Updating IET600”.

3.4.4 Options -> About

Click on the ‘About’ button to get additional information about the IET600 version and the version of its components:
IET600 Data Organization

4.1 Overview

In IET600, you have several different kinds of data:

- Project Data which include the actual project (an MSSQL database), files imported into IET600 and files exported from IET600.
- Project Backups, each file containing the MSSQL database of one project.
- Project Archives, each archive containing a database Backup and configuration files of a project.
- Configuration data.

4.1.1 Project Data

IET600 uses Microsoft SQL Server (MSSQL) to store its data.

MSSQL has some system databases which handle access rights and register the databases of the individual projects. The location of those system databases is determined during installation and cannot be changed. The default path is C:\Data\IETDatabases. As a user, you should not need to have anything to do with this System Database.

The project data are stored below a user-definable path, the default is C:\Data\IETProjects. If you change this path, new projects will be created in the new place; however the existing projects cannot simply be copied or moved to the new path, because their path is registered in the System Database.

Both paths must be on a local drive and must not be compressed or encrypted (this is required by MSSQL).

The data of an individual project are stored below this path, in a folder C:\Data\IETProjects\[project name\_SQL2014]. This folder has again 3 subfolders:

- Data (containing the MSSQL project database, it should not be used for anything else).
- Export (containing a copy of all exported files in a number of pre-defined subfolders).
- Import (containing a copy of all imported files, again in a number of pre-defined subfolders identical to those in the export folder).
4.1.2 Project Backup

IET600 allows to import and export the MSSQL project database, to either create a backup of a project in a defined state, or to exchange it with another IET600 installation. The detailed procedure is described in chapter 4.2.5 (Import and Export Projects).

A project export is a zipped MSSQL database backup. To distinguish it from the multitude of other zip files, IET600 gives it an “.ietprj” file extension.

The default folder for such files is \Data\IETBackup. However, it is strongly recommended to set this path to a location/hard disk different from the drive where you actual project data are located (typically \Data\IETProjects), in order not to lose both project data and backup if the local hard disk fails. A good choice is a USB or network drive. Be sure, however, to select a location with sufficient space (a medium-sized project backup has a size of about 12-15 MB; big projects may have 30-40 or sometimes even 100 MB).

If you import or export a database, you can select any path. However, IET600 always keeps a copy in the project backup folder, thereby eliminating the need for you to make such a copy manually.

4.1.3 Project Archives

At certain stages of a project, you may want to create a backup not only of the database, but of all project-related files. A typical example would be, if you hand over your project to another engineering crew, or to create a backup for a project milestone for legal reasons. The detailed procedure is described in chapter 4.2.6 (Archive a Project).

A project archive is a zipped file that contains the most recent backup of a project and the import and export folders in exactly the same structure as in the project.

The default folder for such files is \Data\IETArchives. It is recommended to set this path to a location/hard disk different from the drive where you actual project data are located (typically \Data\IETProjects). A good choice during the engineering phase is a Network path with the “official” project folder; during testing and commissioning, a USB drive may be needed. Be sure, however, to select a location with sufficient space (a medium-sized project archive has a size of about 20 MB; big projects may have 40-50 or sometimes even more than 100 MB).

The default archive path is configurable; however, you can select any path.

4.1.4 Program and User Configuration Data

These are stored in the default paths that Microsoft has defined for such Data. User-specific data are stored under the pre-defined user application data path; under Windows 7, this is:

C:\Users\[username]\AppData\Roaming\ABB\IET600\[version number]
Data saved here are e.g. IET600 layout, editor configurations etc. Each user can have his/her separate configuration. However, when you install a new version, the configuration will currently not be automatically taken over from the older versions.

**Common configuration data** of IET600 are saved under the pre-defined common application data path; under Windows 7, this is:

C:\ProgramData\ABB\IET600\[version number]

Normally, these data files are installed by IET600 and there should not be any need to change them. Under Windows 7, normal users have only read rights to this path; only administrators can change these files.

### 4.2 Managing IET Projects

#### 4.2.1 The ‘Projects’ Dialog

To get an overview and perform all kinds of project-related functions, IET600 provides the ‘Projects’ Dialog:

<table>
<thead>
<tr>
<th>Project</th>
<th>Status</th>
<th>Tool Version</th>
<th>Last Modified (Local Time)</th>
<th>Last Modified (Remote Representation)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Default</td>
<td>In progress</td>
<td>5.3.256</td>
<td>2021-10-23 13:50:43</td>
<td></td>
</tr>
<tr>
<td>Example</td>
<td>Closed</td>
<td>5.3.255</td>
<td>2021-10-23 13:50:43</td>
<td></td>
</tr>
<tr>
<td>Example 2</td>
<td>In progress</td>
<td>5.3.255</td>
<td>2021-10-23 13:50:43</td>
<td></td>
</tr>
<tr>
<td>Example 3</td>
<td>Closed</td>
<td>5.3.255</td>
<td>2021-10-23 13:50:43</td>
<td></td>
</tr>
</tbody>
</table>

The Dialog provides the following information:

- an overview of all the IET600 projects
- the status of each project
- the IET600 version needed to open the project
- the time the project was last modified

Additionally it provides access to the following functions:

- create a new project
- open an existing project
- rename a project
- delete a project
- import/export (or backup/restore) a project database
- archive a project
To open the ‘Projects’ Dialog, click the round button on the top-left corner of IET600, then select ‘Manage Projects’:

![Image of Manage Projects dialog]

### 4.2.2 Create a New Project

1. Click the round button on the top-left corner of IET600, then select ‘Manage Projects’ -> the ‘Projects’ dialog opens

![Image of project list]

2. To create a new project, click on the ‘New’ button. The following dialog will open:

![Image of create project dialog]
Chapter 4

Enter the desired project name and click ‘OK’ -> a new project will be created:

The small indicator to the right of the project icon indicates the IEC 61850 version of the project, it can be “Ed.1” (per default) or “Ed.2”. The “Ed.1” or “Ed.2” refers to the IEC 61850 edition to which the project is compatible.

How the version of a project affects the engineering possibilities and compatibility with other engineering tools is a rather complex issue, unfortunately. To learn how to change the version of a project, please consult chapter 4.2.8 (Change the Version of a Project).

To learn more about the relatively complex way of how IET600 handles this versioning, consult chapter 27 (Annex: IET600 and IEC 61850 Edition Handling)

4.2.3 Open an Existing Project

1. Click the round button on the top-left corner of IET600, then select ‘Manage Projects’ -> the ‘Projects’ dialog opens

2. Select the row with the project you want to open, and then click ‘Open’. The project will open in IET600

Please take the following points into account:

- If the project is already opened, you cannot open it again.
- If a project has a status ‘Migration required’, it will be migrated to a newer tool version. Before doing so, a backup will automatically be created below the Backup Folder `C:\Data\IETBackup\AutomaticBackup`. After migration, it cannot be opened with older IET600 versions; however, you may import the older version again from the `AutomaticBackup` folder mentioned above.

- You can open only two projects at the same time. The second project will be opened in ‘read-only’ mode which means that data in this project cannot be modified during this session. However, you may copy data from the second project (e.g. entire Bays, IEDs, signals, etc.) into the first project.

### 4.2.4 Save Project Data

Note that IET600 does not save automatically. You must press the save button to save current state of the project. Tip: save your work regularly to shorten saving times.

### 4.2.5 Import and Export Projects

A project can be easily exported and imported in order to either transfer the project to other users or to create a backup. The import/export functions can be accessed from the ‘Projects’ dialog using the ‘Import’ and ‘Export’ buttons. The exported project will be stored in an “.ietprj2” file, which can be imported by other users.

When importing a project, you may need to be aware of the project Identifier:

- If you import a project in a different version (e.g. for comparison), you should keep the identifier, as it still represents the same project.

- If you import one project to create a second project from it (e.g. if you create a second project for a customer and want to use an old project of the same customer as a template), you should set the checkbox ‘Create new identifiers’.

same project  

different project
IET600 has the above as the default behavior: if you keep the same project name, just with (01), (02) etc. added, it will by default keep the identifier. If you use another name, it will automatically set the checkbox ‘Create new identifiers’. However, you can override this, if you are in a special situation.

The identifier serves as a means to determine which IEDs are owned by which projects. This is important when configuring Dataflow between projects or Substations, e.g. as described in chapter 16.4 (SED File Exchange Between Projects).

4.2.6 Archive a Project

You can only archive an open project.

1. You can only archive an open project. Open the project first, as described in chapter 4.2.3 above.

2. Open the ‘Projects’ dialog again by clicking the round button on the top-left corner of IET600 and selecting ‘Manage Projects’:

3. Select the row with the opened project -> the Archive button will become enabled.
Chapter 4

4. Click on the ‘Archive’ button -> the following dialog will open:

You have now several options which differ essentially in the number and type of files which are included in the archive.

- **Create an archive with standard files:**
  
  This option is intended to create an Archive that can be passed to another person who will take over engineering.

  In this option, IET600 will search the ‘Backup’ ‘Import’ and Export’ folders of your project for the following:
  
  - the most recent project Backup.
  - an SCD file of the project and all files imported into IEDs.
  - configuration files for all Gateways and HMIs.

  If some of these files are not found, IET600 will indicate this in the ‘Status’ column. It is up to the user to decide whether to create an Archive without these files or to stop and create these files first.

  Per default, all files found are selected for archiving; you can decide to exclude files by unchecking them.

- **Create an Archive with all files:**

  This is intended for projects with an unusual configuration, where the standard file option above does not cover all essential files.

  In this option, IET600 will search the ‘Backup’ ‘Import’ and Export’ folders of your project for the following:
  
  - the most recent project Backup.
  - all files contained in the ‘Import’ and Export’ folders.
  - configuration files for all Gateways and HMIs.

  If some of the standard files are not found, IET600 will indicate this in the ‘Status’ column. It is up to the user to decide whether to create an Archive without these files or to stop and create these files first.
Per default, all files found are selected for archiving; you can decide to exclude files by unchecking them.

- Create a Milestone-Archive:
  
  This is intended to document the project state at certain defined project milestones. As described above, you can include either only standard files or all files in the Milestone archive.

  If you try to create a Milestone Archive and some standard files are missing, IET600 will warn you and ask you for confirmation, before creating an incomplete Milestone Archive.

5. You may adapt the destination folder for the Archive and the Archive file name, if required.

6. To finally create the Archive, click on ‘Export’. To close the dialog at any time, click on ‘Close’

### 4.2.7 Unarchive a Project

This is currently not provided by IET600.

If you want to restore an archived project, proceed as follows:

1. Extract the archive to a temporary folder.

2. Import the *.ietprj file contained in the root of the archive. This will create a folder for the project.

3. Copy the complete ‘Export’ and ‘Import’ folders contained in the archive into the project folder.

### 4.2.8 Change the Version of a Project

If you create a new project, it will have “Ed.1” as version, indicated by a small text to the right of the project icon in any of the Navigation Trees. This “Ed.1” refers to the IEC 61850 edition to which the project is compatible.

- A higher project version may be required, if an IED DATA model requires it (this, it will be handled on ICD/IID file import). This is

- Few functionalities in the domain of IET600 (e.g. in the ‘Substation’ Tree, the usage of a Function below a Bay) also require a higher project version.

- For compatibility reasons with an older IED configuration tool, you may want to downgrade your version to a lower version number. This, however, will often not be possible.

Before changing a version of a project, always consult chapter 27 (Annex: IET600 and IEC 61850 Edition Handling). It describes in more detail the IET600 project versioning, its relation to the IEC 61850 standard versions and its implications and restrictions.
To change the project version, click on the project element in any Navigation Tree. The ‘Properties’ window will show the following:

Use the combo box to select another version.

If you try to downgrade the version, this may not be possible. The following dialog will inform you about the reasons:

### 4.2.9 Record Comments on Project Changes

IET600 allows the user to add and modify comments for modifications in the project, which will be stored in a text file inside the project directory. The timestamp and the user name will be used for identification.

### 4.3 Upgrade Projects

#### 4.3.1 Introduction

IET600 uses MSSQL as a database to store its projects; MSSQL is installed in the ABB IET600 Prerequisites setup.
Until IET600 5.3.2xx MSSQL Server® 2008 Express was used. Any earlier IET600 version used the same MSSQL version, therefore there was no need for any upgrade.

To support Windows 10 (which IET600 5.3 FP3 supports), MSSQL Server® 2014 Express is required. It will be installed in parallel to the older MSSQL Server® 2008 Express.

New projects created with IET600 5.3 FP3 will therefore be residing on a different database than the old ones. Unlike in earlier updates, you will not see any projects from older IET600 versions, but must take an additional step to transfer them to the new database, from where they can then be migrated and opened like in earlier IET600 Updates.

The Installation procedure is described in the Installation Manual (chapter “Upgrading from IET600 5.3 FP2 and earlier”), the upgrading of older projects from within IET600 will be described in detail in the following chapters.

### 4.3.2 Preliminary Remarks

You may use both MSSQL Server® versions in parallel, if you need to work with both IET600 5.3 FP2 (or older), and IET600 5.3 FP3 (or newer) on the same PC. Also for the upgrade described here, you need both versions.

To prevent conflicts between the two MSSQL server products, several names and folders have been changed in IET600 5.3 FP3. The following table gives an overview of the differences:

<table>
<thead>
<tr>
<th>IET600 Prerequisites</th>
<th>IET600 5.3 FP2 or older</th>
<th>IET600 5.3 FP3 or newer</th>
</tr>
</thead>
<tbody>
<tr>
<td>MSSQL Server Product</td>
<td>MSSQL Server® 2008 Express</td>
<td>MSSQL Server® 2014 SP2 Express</td>
</tr>
<tr>
<td>MSSQL Server Instance Name</td>
<td>IETSERVER</td>
<td>IETSERVER2</td>
</tr>
<tr>
<td>Project Subfolder</td>
<td>\IETProjects\project</td>
<td>\IETProjects\project_SQL2014</td>
</tr>
<tr>
<td>Project Backup File Extension</td>
<td>.ietprj</td>
<td>.ietprj2</td>
</tr>
</tbody>
</table>

### 4.3.3 Upgrade Preparation

1. Close all older IET600 programs (they access the old IETSERVER instance which prevents an upgrade).

2. Close any program which might access any files of projects you want to migrate. E.g. make sure you do not have any Customer Lists open in Excel; and
also make sure, you do not have any project folders or files open in Windows Explorer.

3. Both MSSQL Instances, IETSERVER and IETSERVER2 must be running. If you could install IET600 5.3 FP3, this can be assumed to be the case (consult the Installation Guide, chapter “Upgrade IET600 Installation” how to check this).

4. (Before any action, the old database will be exported into the <Backup>\AutomaticBackup folder; therefore you do not specifically need to do an ‘Export’ or ‘Archive’ in an older IET600 version.)

4.3.4 Upgrade Projects

To upgrade projects from IET600 5.3 FP2 or older to IET600 5.3 FP3 requires first that the respective project databases will be shifted from the MSSQL Server® 2008/ IETSERVER instance to the MSSQL Server® 2014/IETSERVER2. This can be done from within IET600.

1. Open the ‘Manage Projects’ dialog:

   If you have projects to migrate, an additional ‘Upgrade’ button is visible on this dialog. This button is only visible if the following conditions are fulfilled:

   - IET600 Prerequisites 5.1.x or older are installed
   - The ‘SQL Server (IETSERVER)’ service is running
   - Its MSSQL server instance contains at least one IET600 project database.

   If you open the ‘Manage Projects’ dialog and you have no migrated projects yet, a dialog will provide some guidance (if the above upgrade conditions are fulfilled).

   ![Manage Projects Dialog](image)

   This dialog will appear only as long as you did not upgrade any projects; after migrating at least one, it is assumed that you know the way to do it for additional projects.
Please be aware than on an older Hard Disk, it may take a while to scan the SQL Server for projects and display them. You can recognize this by the fact that all buttons except ‘Exit Dialog’ will be greyed out while scanning is not complete.

2. Click on Upgrade -> the actual ‘Upgrade’ Dialog opens:

3. Select one or several projects you want to upgrade and then click ‘Upgrade’. The upgrade progress will be indicated by a progress bar in the bottom of the dialog. After the upgrade has successfully finished, those projects will disappear from the list.

(What happens during this upgrade is that the project folder is renamed and the project database is transferred to the new IETSERVER2 instance).

You may upgrade some more projects by repeating the above steps.

4. If you have finished, click on ‘Exit Dialog’ (you can still upgrade additional projects at a later time, if so required).

All upgraded projects should now be visible in the ‘Manage Projects’ dialog. You can now continue as usual by opening the upgraded projects from the ‘Manage Projects’ Dialog.

4.3.5 Upgrade by Importing IETPRJ Files

In IET600 5.3 FP3, you can import any *.ietprj or *.ietprj2 files, irrespective whether it was created with a version IET600 5.3 FP3 and newer or IET600 5.3 FP2 and older.

Simply go to the ‘Manage Projects’ dialog, click ‘Import’ and select the file you want to import, as you are probably used to already. The project will be imported as usual, any necessary upgrade happens in the background and needs no action by the user.
5 Engineering Workflow

5.1 Overview

The engineering process for an IEC 61850-based substation automation system usually involves the following steps:

- Substation Specification (Single Line Diagram)
- Signal engineering
- IED Data engineering
- Communication configuration (CCD)
- Dataflow configuration

Decoupling of the system specification tasks is a central concept in IET600 which provides flexibility to decide when and how to complete the various engineering tasks. In other words, the tool does not impose a strict engineering workflow, but allows work on the various steps to be done in parallel and provides the support to unite the configuration at a later stage of the project.
5.2 Engineering of a New Project

5.2.1 Overview

In IET600, it is possible to start engineering at several places in parallel. The following overview shows how different engineering steps are linked.

The recommended engineering process can be divided into the following steps:

1. Basic Substation configuration (SLD and IEDs, the latter without any content at the moment).
2. Basic Signal configuration
3. Full Signal Configuration.
4. Communication Configuration (CCD)
5. Dataflow configuration.
Engineering steps 2 to 5 above can partially be executed in parallel, the diagram shows the dependencies.

5.2.2 Basic Substation Configuration

1. Create the whole substation structure (Substation, Voltage Level and Bays), create Equipment in Bays.
2. Draw the Single Line Diagram.
3. Configure customer names for all the above elements.
4. Create the IEDs in Bays.
   
   At this stage, you have a Single Line Diagram available.
   
   To actually draw the Single Line Diagram is optional. However, the equipment and its internal and customer names are needed for a correct signal configuration.
   
   Experience shows that drawing the Single Line Diagram requires little additional effort and considerably enhances the ability to orient oneself in the substation. It is therefore highly recommended.

5.2.3 Basic Signal configuration

1. Configure customer names for the IEDs.
2. Add signals to the IEDs and configure them.
3. Do the LN mapping
   
   At this stage, you are ready to export Signals Lists for approval by the customer.

5.2.4 Full Signal configuration

1. Create all IEDs that were still missing (HMIs, Gateways, NCCs, Switches and other communication equipment).
2. Add monitoring signals to those IEDs and configure them.
3. Do the LN mapping for those signals, if required.
4. Map the signals to HMIs and Gateways, as required
   
   At this stage, you are ready to export complete Signal Lists and configuration files for HMIs and Gateways.
   
   Depending on your HMI/Gateway configuration tool, you may also need the SCD file (see chapter 5.2.6, Dataflow Configuration)
5.2.5 Communication configuration

This configuration is not mandatory, i.e. you can fully engineer a substation, even if you skip this section. However, a Communication Configuration Diagram (CCD) does not take a lot of time to configure and can be of great use in the following situations:

- discussion with the customer, particularly for redundancy configuration
- problem tracing in the commissioning phase

It is therefore recommended to do it; the time is usually well spent. As a precondition, you should have most of your IEDs created, and any redundancy configuration should be clear.

1. Configure the physical communication ports for all IEDs.
2. Arrange the IEDs in the CCD.
3. Connect the IEDs in the CCD.
4. In complex communication configurations (e.g. redundancy), you may want to add Access Points to IEDs and map them to physical ports, if you do not have the IEC 61850 Data model of these IEDs available.
   At this stage, you have a fully configured physical communication and can print a CCD.

5.2.6 Dataflow Configuration

1. Create Subnetworks and configure them for the appropriate protocols.
2. Connect the IEDs to the appropriate Subnetworks
3. If you have not done so till now, import the IEC 61850 Data model to the IEDs
4. Configure the Dataflow:
   - RCBs with their Datasets
   - LCBs with their Datasets, if configurable
   - GCBs with their Datasets
   - SVCBs with their Datasets, if configurable
5. Configure the Dataflow clients.

After this step is performed, you can export an SCD file.
5.3 Engineering of a Retrofit Project

5.3.1 Overview

In a retrofit project, the engineer will start with an SCD file and some HMI/Gateway configuration files. After these are imported to provide the existing configuration, he may add new items as requested (the example below assumes that a Bay is added to an existing Substation).

The recommended engineering process can be divided into the following steps:

1. Import of the existing configuration
2. Add new elements (Bays, IEDs)
3. Signal configuration of new elements
4. Communication configuration of new elements
5. Dataflow configuration

Steps 3 to 5 above can partially be executed in parallel, the diagram shows the dependencies.

5.3.2 Import of Existing Configuration

1. Import SCD file. The SCD file will provide IED data models and Dataflow configuration. It may additionally provide LSD and CCD information and LN mapping; however, this information tends to be incomplete and needs to be carefully checked.

2. Import HMI/Gateway configuration files. IET600 will try to associate signals to individual IEDs based on Signal information; signals that cannot be reliably associated will remain on the HMI to which the configuration file belongs. IET600 will also recreate an existing NCC configuration, if the proper configuration files are provided.

5.3.3 Add New Elements

1. Add new structures as needed (usually Bays), add equipment as needed.
2. Complete the Single Line Diagram.
3. Configure customer names for all the above elements.
4. Add IEDs as required.
   At this stage, you have the revised Single Line Diagram available.

5.3.4 Signal Configuration of New Elements

1. Configure customer names for the added IEDs.
2. Add signals to the added IEDs and configure them.
3. Do the LN mapping for the added signals
4. Map the signals to HMIs and Gateways, as required
   At this stage, you are ready to export complete Signal Lists and configuration files for HMIs and Gateways.
   Depending on your HMI/Gateway configuration tool, you may also need the SCD file (see chapter 5.3.6, Dataflow Configuration)
5.3.5 Communication Configuration of New Elements

This configuration is not mandatory. If the existing Substation already provided a CCD, you will probably also want to configure the added elements.

1. Configure the physical communication ports for the added IEDs.
2. Arrange the added IEDs in the CCD.
3. Connect the added IEDs in the CCD.
4. In complex communication configurations (e.g. redundancy), you may want to add Access Points to IEDs and map them to physical ports, if you do not have the IEC 61850 Data model of these IEDs available.

5.3.6 Dataflow Configuration

1. (Create additional Subnetworks, if required; most likely you will re-use existing Subnetworks).
2. Connect the added IEDs to the appropriate Subnetworks.
3. If you have not done so till now, import the IEC 61850 Data model to the added IEDs.
4. Configure the Dataflow of the added IEDs:
   - RCBs with their Datasets
   - LCBs with their Datasets, if configurable
   - GCBs with their Datasets
   - SVCBs with their Datasets, if configurable
5. Configure the Dataflow clients.

After this step is performed, you can export an SCD file.
5.4 Engineering of Dataflow Only

5.4.1 Overview

In some Substation Automation projects, it happens that one party delivers protection and control IEDs, another party delivers the HMI/Gateway. In this case, the first party is expected to configure not only the IED data model, but also the Dataflow; the second party is responsible for the configuration of Signals, HMI and/or Gateway.

In this case, the engineer will start with an SCD file from a configuration tool like PCM600, which contains all IEDs and may contain some Substation configuration, some LN mapping, some Subnetwork and Dataflow configuration.

The engineer will use IET600 to calculate the complete Dataflow. If required, he may also provide LN mapping.

The recommended engineering process can be divided into the following steps:

1. Import of the existing configuration (via SCD file).
2. Update of SLD configuration and LN mapping, if required.
3. Dataflow configuration.

5.4.2 Import of Existing Configuration

1. Import SCD file. The SCD file will provide IED data models and may provide a default Dataflow configuration. It may or may not contain Subnetworks. It may additionally provide SLD and LN mapping; however, this information tends to be incomplete and needs to be carefully checked if it is to be used.

5.4.3 Update of SLD Configuration and LN Mapping

(optional)

1. Add Substation structures and equipment as needed.
2. Provide LN mapping as needed.
   At this stage, you can export a Single Line Diagram, if needed.

5.4.4 Dataflow Configuration

1. Create Subnetworks as required; most likely you will re-use existing Subnetworks).
2. Add Client IEDs (HMIs/Gateways), if they are not provided by the SCD already.
3. Connect IEDs to the appropriate Subnetworks.
4. Configure the Dataflow of the IEDs:
   - RCBs with their Datasets
   - LCBs with their Datasets, if configurable
   - GCBs with their Datasets
   - SVCBs with their Datasets, if configurable
5. Configure the Dataflow clients.
   After this step is performed, you can export an SCD file.
6 Substation Specification

6.1 IEC 61850 Modelling of a Substation

IEC 61850 allows you to build a Substation tree so that you can model your Substation Hierarchy.

The main purpose of the Substation tree elements is that Logical Nodes (LNs) of an IED can be mapped to such elements. This mapping can then be used e.g. for identification (CSWI1 belonging to QA1, CSWI2 to QB1 etc.) or to automatically configure texts in signals.

IEC 61850 distinguishes between 'containers' and 'equipment'. Containers are e.g. a Substation, a Voltage Level or a Bay, Equipment is e.g. a Circuit Breaker, a Voltage Transformer or a Tap Changer.

6.1.1 Substation ‘Containers’

Originally, IEC 61850 (Ed. 1) allowed you to have 3 levels of containers only:

- Substation -> Voltage Level -> Bay.
- Substation -> ‘Function’ -> ‘Subfunction’ (to model auxiliary functionality).

The three levels proved insufficient, so that IEC 61850 (Ed. 2) expanded this container model:

- ‘Functions’ could now be used below Voltage Levels and Bays also, and ‘Subfunctions’ could contain other ‘Subfunctions’.

A Diameter e.g. can now be modeled as Bay and its Sections can be modelled as Functions (Annex 21.2.4 provides an example how to model a Diameter). Be aware that this modelling requires IEC 61850 (Ed. 2), as Functions below Bays were not allowed in Ed. 1. See Annex: IET600 and IEC 61850 Edition Handling for more details about edition handling.

6.1.2 Substation Equipment

IEC 61850 (Ed. 1) split ‘Equipment’ into 3 types:

- Power Transformers (a separate category due to their complexity)
- Conducting Equipment (electrical equipment such as Circuit Breakers, Disconnectors etc.).
- General Equipment (non-electrical equipment such as Pumps, Valves etc.)
Additionally you could use a ‘Sub-Equipment’ to further divide Equipment (typically used to model phases of an Equipment, e.g. a Circuit Breaker.

Again, IEC 61850 (Ed. 2) expanded this model:

- Equipment could now contain an ‘Equipment Function’ which in turn could contain an ‘Equipment Subfunction’ which could contain other ‘Equipment Subfunctions’.

This is mainly intended to allow the modeling of redundancy of some sort; e.g. rather than modeling protection trips all to the same Circuit Breaker, an engineer could model two Equipment Functions ‘Main 1’ and ‘Main 2’ so that LNs of the respective IEDs ‘Protection Main 1’ and ‘Protection Main 2’ can be mapped to those Equipment functions rather than the equipment itself.

6.2 Creating Substation ‘Containers’

6.2.1 Overview

IET600 provides 2 different methods to configure containers:

1. All container types can be created from the context menu of the respective items in the ‘Substation’ Navigation Tree. The choice is limited to those container types that can be created for a specific parent (e.g. below a Substation, you can create only a ‘Voltage Level’ or a ‘Function’).

2. The most commonly used containers (Substation, Voltage Levels and Bays) can be created from the Main menu:

6.2.2 Create a Substation

1. Select the Substation tab in the Main menu and click ‘Create Substation’:

   Alternatively, from the context menu of the project in the ‘Substation’ Navigation Tree, select ‘Create new Substation’.
2. Enter a name for the Substation. The name must be unique within the project.

3. Click OK -> a new Substation is created below the project in the ‘Substation’ Navigation Tree

### 6.2.3 Create a Voltage Level

1. Select the Substation tab in the Main menu and click 'Create Voltage Level':

   Alternatively, from the context menu of a Substation in the ‘Substation’ Navigation Tree, select 'Create new Voltage Level'.

2. Enter a name for the Voltage Level. The name must be unique within the substation.

3. Click ‘OK’ -> a new Voltage Level is created below the Substation in the ‘Substation’ Navigation Tree
6.2.4 Create a Bay

1. Select a Voltage Level, right click and select ‘Create New Bay’ or click on the Bay function in the menu to open the Bay dialog.

2. Enter a name for the new Bay. The name must be unique within the Voltage Level.

3. Click ‘OK’ -> a new Bay is created below the Voltage Level in the ‘Substation’ Navigation Tree.

To configure Bay Equipment and Topology:
1. Select the Bay you want to configure in the Substation Tree.

2. Add/configure Equipment as described in chapter 6.3 (Creating Equipment: The ‘SLD’ Editor)

To configure a whole Voltage Level
1. Repeat the above step for all Bays in a Voltage Level.

2. Select the Voltage Level you want to configure in the Substation Tree.

3. Create Busbars as needed

4. Use the ‘Layout Bays’ or ‘Align Bays’ from the Single Line Diagram tab in the Main menu to do an automatic initial layout.

5. Adapt the layout manually, if needed.

6.2.5 Create a ‘Function’

1. Select a Substation, Voltage Level or Bay, right click and select ‘Create New Function’ -> this will open a dialog.
2. Enter a name for the new ‘Function’. The name must be unique within the parent Substation, Voltage Level or Bay.

3. Click ‘OK’ -> a new Function is created below the parent Substation, Voltage Level or Bay in the ‘Substation’ Navigation Tree.

6.2.6 Create a ‘Subfunction’

1. Select a Function or Subfunction, right click and select ‘Create New Subfunction’ -> this will open a dialog.

2. Enter a name for the new ‘Subfunction’. The name must be unique within the parent Function or Subfunction.

3. Click ‘OK’ -> a new Subfunction is created below the parent Function or Subfunction in the ‘Substation’ Navigation Tree.

6.2.7 Create a ‘Subequipment’

1. Select an Equipment, right click and select ‘Create New Subequipment’ -> this will open a dialog.

2. A Subequipment should normally be used for Phases. Select the Phase, if applicable -> a default name will be inserted
3. If necessary, modify the name for the new ‘Subequipment’. The name must be unique within the Equipment.

4. Click ‘OK’ -> a new Subequipment is created below the Equipment in the ‘Substation’ Navigation Tree.

6.2.8 Create an ‘Equipment Function’

Please do not use ‘Equipment Function’ to model phases of an Equipment, use ‘Subequipment’ for this purpose.

1. Select an Equipment, right click and select ‘Create New Equipment Function’ -> this will open a dialog.

2. Enter a name for the new ‘Equipment Function’. The name must be unique within the Equipment.

3. Click ‘OK’ -> a new Equipment Function is created below the Equipment in the ‘Substation’ Navigation Tree.

6.2.9 Create an ‘Equipment Subfunction’

1. Select an Equipment Function or Equipment Subfunction, right click and select ‘Create New Equipment Subfunction’ -> this will open a dialog.
2. Enter a name for the new ‘Equipment Subfunction’. The name must be unique within the parent Equipment Function or Equipment Subfunction.

3. Click ‘OK’ -> a new Subfunction is created below the Equipment Function or Equipment Subfunction in the ‘Substation’ Navigation Tree.

6.3 Creating Equipment: The ‘SLD’ Editor

6.3.1 Overview

IET600 provides the ‘SLD’ Editor to create and connect equipment. Some restrictions apply, however:

- Only equipment commonly used in Substation engineering can be created, not all equipment types specified by IEC 61850 are available.
- In IET600, equipment can only be created within Bays, although IEC 61850 allows so-called ‘General Equipment’ and Transformers to be allocated to other containers as well.

Alternatively, equipment can be created by importing an SCD or SSD file to initially create a project. IET600 will also import and display equipment from types which are not available from the ‘SLD’ editor and will also show equipment allocated to Voltage Levels and Substations in the tree according to their allocation. However such equipment cannot be re-allocated.
6.3.2  Create a Single Line Diagram

The editor provides shapes for primary equipment and shapes to connect equipment. Moreover, the Bay equipment can be connected to the Busbar to define a complete Single Line Diagram for a voltage level.

To draw the single line diagram for a Bay:
1. Select the Bay in the ‘Substations’ Navigation Tree
2. Click on the primary equipment you want to insert (e.g. a circuit breaker)
3. Move cursor to the Bay drawing area (orange box) and click where you want to insert your equipment.
4. Repeat steps 1 to 3 to draw all primary equipment
5. Connect the primary equipment with lines

To view the shape names, move your cursor above the shape and the Tooltip box will appear.

Alternatively, you can right click on the connectivity node and select ‘Attach Conducting Equipment’ to connect to available shapes.

Multiple selection is supported for equipment:
1. Select Arrow shape
2. To select multiple equipment, press the Ctrl key on the keyboard and
   - either click sequentially on each equipment you want to include in the selection.
   - or select an area with mouse drag’n’drop -> all equipment fully or partially within this area is selected.
3. Use keyboard shortcuts Ctrl+C to copy and Ctrl+V to paste.

To unselect, click on an equipment not included in the multiple selection or on a Bay. You cannot selectively remove a single equipment out of a multiple selection.

When using the SLD editor, the Main menu shows a context-dependent menu ‘Single Line Diagram’ that provides some additional features:

- IEC/ANSI/Custom: changes the symbols according to the selected standard
- Align Bays: aligns multiple Bays in the same drawing (e.g. voltage level) to avoid overlapping Bays
- Layout Bays: aligns multiple Bays in the same drawing (e.g. voltage level) and orders Bays in alpha-numerical order from left to right

A diagram of the whole substation is not available.

6.4 Creating a Substation by Importing SCD/SSD File

To create the substation structure by importing an SCD or SSD file, please refer to chapters 16.2.1 (Import an SCD File to Create a Project) or 16.3 (SSD File Import). This can only be done in an empty project which does not contain a Substation already.

To create IEDs in a Bay, Voltage Level or Substation, please refer to chapter 7.2 (Creating and Configuring IEDs).
7  IED Engineering

7.1  Introduction

In IET600, you have several ways to engineer IEDs:
Create an IED manually, and then add/edit their data.
Copy a configured IED from the same or another project.

7.1.1  IED Roles

Each IED must have at least one role (primary role). In IET600, the following roles can be configured:

- Archive
- *HMI / Gateway*
- IED (Protection / Control IEDs)
- Merging Unit
- NCC
- Printer
- Router
- Security Device
- *Switch / Hub *
- Time Server
- Other

* Some few roles can be combined with other roles. For example it is allowed to create a Gateway IED with an additional role HMI. A combination of roles is only possible for related roles. Related roles are listed in the same line and in italic type. An already created IED can be reconfigured with a related role only.
### 7.1.2 IED Status Editor

The IED Status editor provides an overview of all IEDs in the project, including revision history information. The following properties are available:

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Modified</td>
<td>Indicates if changes to the IED have been made since the last SCD export. The flag is reset with SCD export.</td>
</tr>
<tr>
<td>IED Name</td>
<td>IED name</td>
</tr>
<tr>
<td>IED Type</td>
<td>IED type (i.e. product family) from imported ICD/IID file (read-only)</td>
</tr>
<tr>
<td>IED Version</td>
<td>The SCL version as which this IED is treated (for details, see chapter 27.1, IEC 61850 Classification)</td>
</tr>
<tr>
<td>Blocked due to SCL Version Conflict</td>
<td>Dataflow Engineering is restricted due to IEC 61850 version conflicts (for details, see chapter 27.4 IET600 Mixed System Engineering)</td>
</tr>
<tr>
<td>Customer Name</td>
<td>Customer specific name</td>
</tr>
<tr>
<td>Manufacturer</td>
<td>Manufacturer name from imported ICD/IID file (read-only)</td>
</tr>
<tr>
<td>Foreign IED</td>
<td></td>
</tr>
<tr>
<td>IEC 61850 Engineering Rights</td>
<td>Information related to SED file transfer (for details, s. chapter 16.4, SED File Exchange Between Projects)</td>
</tr>
<tr>
<td>Pending SED Transfer</td>
<td></td>
</tr>
<tr>
<td>Checked Out</td>
<td></td>
</tr>
<tr>
<td>Protected</td>
<td>Summary protection flag, indicating that some protection is active for an IED (s. below)</td>
</tr>
<tr>
<td>Protect RCB Dataflow</td>
<td></td>
</tr>
<tr>
<td>Protect GCB Dataflow</td>
<td></td>
</tr>
<tr>
<td>Protect SVCB Dataflow</td>
<td>Protect IED against specific configuration changes (for details, see chapter 18, Protecting IEDs Against Changes).</td>
</tr>
<tr>
<td>HMI Engineering Right</td>
<td></td>
</tr>
<tr>
<td>SCL Import File Name</td>
<td>Path and file name of imported ICD/IID file (read-only)</td>
</tr>
<tr>
<td>SCL Import Date</td>
<td>Date and time of last ICD/IID file import (read-only)</td>
</tr>
<tr>
<td>Property</td>
<td>Description</td>
</tr>
<tr>
<td>--------------------------</td>
<td>--------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>SCL Export Date</td>
<td>Date and time of last SCD file export (read-only)</td>
</tr>
<tr>
<td>Configuration Version</td>
<td>IED Configuration Version from imported ICD/IID file (read-only)</td>
</tr>
<tr>
<td>Software Revision</td>
<td>IED Software revision from imported ICD/IID file (read-only)</td>
</tr>
<tr>
<td>Configuration Revision</td>
<td>Configuration Revision from imported ICD/IID file (read-only)</td>
</tr>
<tr>
<td></td>
<td>Only the 1st LD is shown in the table, use Tool Tip to see all, if IED contains several LDs.</td>
</tr>
<tr>
<td>Primary Role</td>
<td>(read-only)</td>
</tr>
<tr>
<td>HMI Signal Count</td>
<td>Number of HMI Signals per IED, as defined in HMI Data editor (read-only)</td>
</tr>
<tr>
<td>Capability Definition</td>
<td>Reference to Manufacturer and IED Type in the IED Capabilities editor (read-only)</td>
</tr>
</tbody>
</table>

7.1.3 IED Settings Editor

The IED Configuration editor provides a list of IED properties that can be configured. These are mainly Disturbance Recording parameters which can currently only be used by ABB OPC Server clients.

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>IED Name</td>
<td>IED name</td>
</tr>
<tr>
<td>IED Type</td>
<td>IED product family (from imported CID/ICD/IID file)</td>
</tr>
<tr>
<td>IED Version</td>
<td>The SCL version as which this IED is treated (for details, s. chapter 27.1, IEC 61850 Classification)</td>
</tr>
<tr>
<td>Customer Name</td>
<td>Customer specific name</td>
</tr>
<tr>
<td>Manufacturer</td>
<td>Manufacturer name from imported ICD/IID file</td>
</tr>
<tr>
<td>Disturbance Recording</td>
<td>true, if Disturbance Recording should be enabled. (see chapter 7.6 for more details)</td>
</tr>
<tr>
<td>enabled</td>
<td></td>
</tr>
<tr>
<td>Property</td>
<td>Description</td>
</tr>
<tr>
<td>----------------------------------------------</td>
<td>-------------------------------------------------------------------------------------------------------</td>
</tr>
</tbody>
</table>
| Disturbance Recording Delete Recordings After Upload | true, if Disturbance Recordings on the IED will be deleted after a successful upload.  
(see chapter 7.6 for more details) |
| Disturbance Recording 61850 Default Directory | true, if the directory, where the Disturbance Recordings are stored, is as defined in IEC 61850.  
(see chapter 7.6 for more details) |
| Disturbance Recording Directory              | The directory, where the Disturbance Recordings are stored.  
(see chapter 7.6 for more details) |
| Disturbance Recording Polling Period          | On the Server IED, the minimum allowed polling period in seconds (to avoid performance problems due to a misconfigured client)  
On the Client IED the actual polling period in seconds.  
(see chapter 7.6 for more details) |
| Disturbance Recording MMS Upload Capability   | true, if Disturbance Recordings can be uploaded via MMS.  
(see chapter 7.6 for more details) |
| Disturbance Recording FTP Upload Capability    | true, if Disturbance Recordings can be uploaded via FTP.  
(see chapter 7.6 for more details) |
| Disturbance Recording FTPS Upload Capability   | true, if Disturbance Recordings can be uploaded via FTPS.  
(see chapter 7.6 for more details) |
| Dynamic Datasets                              | Allows to configure whether an 8-MMS Client IED shall try to configure Dynamic Datasets on an 8-MMS Server IED.  
(see chapter 7.7 for more details) |
| PCM Template                                  | Reference to PCM600 IED Configuration Template.  
(see chapter 7.8 for more details) |
| PCM Object Type                               | Reference to PCM600 Object/IED Type.  
(see chapter 7.8 for more details) |
<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SNMP Write Community</td>
<td>String specifying the SNMP Community allowed to write via SNMP.</td>
</tr>
<tr>
<td></td>
<td>(see chapter 7.8 for more details)</td>
</tr>
<tr>
<td>SNMP Read Community</td>
<td>String specifying the SNMP Community allowed to read via SNMP.</td>
</tr>
<tr>
<td></td>
<td>(see chapter 7.8 for more details)</td>
</tr>
<tr>
<td>SNMP Poll Rate</td>
<td>Seconds between polling IEDs via SNMP.</td>
</tr>
<tr>
<td></td>
<td>(see chapter 7.8 for more details)</td>
</tr>
<tr>
<td>SNMP Timeout</td>
<td>Seconds for Timeout if an SNMP Server does not respond to a request from an SNMP Client.</td>
</tr>
<tr>
<td></td>
<td>(see chapter 7.8 for more details)</td>
</tr>
<tr>
<td>SNMP Version</td>
<td>SNMP Protocol Version to communicate.</td>
</tr>
<tr>
<td></td>
<td>(see chapter 7.8 for more details)</td>
</tr>
<tr>
<td>SNMP Authentication Protocol</td>
<td>Algorithm for authentication of client to server.</td>
</tr>
<tr>
<td></td>
<td>(see chapter 7.8 for more details)</td>
</tr>
<tr>
<td>SNMP Encryption Algo-</td>
<td>Algorithm for encryption of communication between client and server.</td>
</tr>
<tr>
<td>rithm</td>
<td>(see chapter 7.8 for more details)</td>
</tr>
<tr>
<td>SNMP User Name</td>
<td>User name used for authentication.</td>
</tr>
<tr>
<td></td>
<td>(see chapter 7.8 for more details)</td>
</tr>
</tbody>
</table>

### 7.1.4 LN Data Editor

The LN Data editor provides a view on the IEC 61850 model of the IED, including the following attributes.

- IED
- LD
- LN
- Path
- Value
- Short Address
- Description
Note that most of these attributes are “read-only”, as they must be defined by the IED configuration tool in the ICD/IID file. Some attributes are editable; however these shall only be used for engineering of COMS81 gateways.

### 7.1.5 IED Services Editor

IEC 61850 specifies IED services, with which an IED can declare its own capabilities and limitations. The understanding of IED services, however, is made somewhat complex by three facts:

- **The default values, if a service is missing, are not consistent.**
  
  E.g., service FileHandling: a missing service indicates that IED File handling services are not supported, an existing service FileHandling without sub-services means that file transfer via MMS is supported while file transfer via FTP or FTPS is not supported.

- **IEC 61850 allows to specify services on either IED or Access Point(s) to allow different services to be available via different Access Points.**

- **Due to incomplete service specifications (at least in IEC 61850 Ed.1), IET600 introduced its ‘IED Capabilities’ which sometimes just provide a default value, if the service is missing, sometimes they override an existing service.**

The ‘IED Services’ editor is intended for persons which are familiar with the IEC 61850 services as it does not explain the services, but only makes the above-mentioned complexity transparent. If used by a non-expert, it may confuse more than it clarifies.

The IED Services editor is hidden by default. To make it visible, open the Main menu (top left circle), select ‘Options’, ‘Appearance’ and enable the checkbox ‘Show IED Services Tab’:
The ‘IED Services’ Editor then shows:

- default values in case a service is missing
- which services are defined by the IED and whether those services are defined on IED or Access Point level.
- whether IET600 provides any overrides or default values (see chapter 7.1.6 below).
- the final result of all the above (and whether the result applies on IED or Access Point level).

By filtering on the ‘Service Name’ column, you can compare a service across several IEDs.

### 7.1.6 IED Capabilities Editor

In IEC 61850, so-called services, which are part of an ICD/IID/SCD file, define certain capabilities of an IED. These services – for historical reasons – may not always be correctly configured by the IED configuration tool or correctly interpreted by IET600. Therefore, the IED Capabilities editor allows the user to define default values or override behavior for the purpose of correct engineering in IET600.

For more information and to address some known problems which can be solved with IED capabilities, please consult chapter 26 (Annex: IED Capabilities).

Please be very careful when editing anything in the IED Capabilities editor. The default options are set according to an ABB-tested engineering process for defined ABB devices and reflect product-specific know-how and engineering experience. To change these settings, expert knowledge is required; and extreme care must be taken to test and verify results of changed settings, as such changes may otherwise result in improper functioning of a system.
Note that the changes in the IED Capabilities tab will not be written into the SCL IED Services of the exported SCD file; they are only used within IET600.

To enable the IED Capabilities editor, open the Main menu (top left circle) and select Options and enable the checkbox ‘Show IED Capabilities Tab’:
7.2 Creating and Configuring IEDs

7.2.1 Overview

You can create new IEDs in several locations:

- In the IED Tree below a project.
- In the Substation Tree, below a Substation, Voltage Level or Bay.
- In the HMI Tree (station-level IEDs only, like MicroSCADA SYS600 HMIs and COM500 NCC Gateways).

The next chapter describes how to create a generic new IED. Subsequent chapters describe how to create new IEDs with a predefined role. There are also some special properties which are available dependent the role (e.g. OPC servers for HMIs).

7.2.2 Create New IED

An IED can be created using either the Substation menu or in the Substation or IEDs navigation tree using the context menu.

1. In the Substation or IEDs menu, select the ‘IED’ button:

2. Alternatively, you can create an IED from the context menu in:
   - a ‘Substation’, ‘Voltage Level’ or ‘Bay’ element in the ‘Substation’ tab, or
   - the ‘Project’ element in the IED tree
Right-click the element -> the context menu for that element appears. Select ‘Create New IED’ -> ‘Other…’

3. This opens the dialog ‘Create New IED’.

The following steps describe the mandatory input fields:

- First, select the location for the new IED on the left hand side of the dialog. The location is the element under which the IED will be created.

- Depending on the previously selected location, IET600 proposes a name, which you can adapt according to your needs. Invalid names are indicated by an error mark (invalid names are usually either duplicate names or contain characters that are not allowed for a particular element).

- Select one of the available roles as primary role. If you started the dialog with a predefined role you cannot change the role here anymore.
Additionally, the dialog allows you to configure some further settings:

- The IED Type is set to “GENERIC” by default. You can adapt it according your needs.
- You can select the number of ports and configure individual ports, select the first tab, ‘Ports’. (Ports are required to design the communication configuration diagram. For more detailed information about port configuration, please consult chapter 9.4.1, Configure Communication Properties).
- For some IEDs, (e.g. MicroSCADA, Switches), you can configure Access Points. To do so, select the second tab. You can add, remove or rename access points. Note: At least one access point is required.
- To reduce the engineering time, IET600 allows you to use templates which contain predefined settings for the communication parameters. They are only available if the primary role is already set. Also there aren’t templates for all roles. To apply a template perform following steps:
  - In the section ‘Available Templates’, select a template.
  - Click the arrow button to transfer the template settings to the IED. Note: Applying the template will overwrite all current communication parameters. You can undo this action only by exiting the dialog with ‘Cancel’.

Depending on the selected primary role, there are other settings available. For more information see the following chapters which describe how to create IED with specific roles.

Click ‘OK’ to create the new IED or ‘Cancel’ to close the dialog without creating an IED.

If you created a new IED, the Navigation Tree will change to either the ‘Substation’ or ‘IEDs’ Tree.

### 7.2.3 Create New Protection/Control IED

This section describes the specific part of configuring a Protection or Control IED and is based on the general description for creating an IED (see 7.2.2, Create New IED):

1. Select the ‘Substation’ or ‘IEDs’ Navigation Tree, and select an element below which you want to create an IED. This could be:
   - a ‘Substation’, ‘Voltage Level’ or ‘Bay’ element in the ‘Substation’ Tree
   - the ‘Project’ element in the ‘IEDs’ tree
Right-click the element -> the context menu for that element appears. Select ‘Create New IED’ -> ‘Protection/Control IED…’:

2. This opens the dialog ‘Create new IED’.

3. To configure the general part, please see chapter 7.2.2, Create New IED.

4. In the section ‘Role Parameters’ -> tab ‘IED’, you can define if the IED is for protection and/or control.

7.2.4 Create New Switch

If you create a switch, IET600 will currently automatically generate its IEC-61850-model.

This section describes the specific part of configuring a switch and is based on the general description for creating an IED (see chapter 7.2.2, Create New IED):

1. Select the ‘Substation’ or ‘IEDs’ Navigation Tree, and select an element below which you want to create an IED. This could be either:
   - a ‘Substation’, ‘Voltage Level’ or ‘Bay’ element in the ‘Substation’ Tree
   - the ‘Project’ element in the ‘IEDs’ tree
Right-click the element -> the context menu for that element appears. Select 'Create New IED' -> 'Switch...':

The dialog 'Create new IED' will open.

2. To configure the general part please see chapter 7.2.2 (Create New IED).

3. In the section 'Role Parameters -> Switch' you can define the 'RSTP priority'. Possible values are root switch, backup switch or undefined.

4. When creating a switch, the 'Ports' table in the dialog contains the additional columns 'MAU Type*' and 'Auto negotiation'. If 'Auto negotiation' is disabled a 'MAU Type*' should be selected. Configure them as needed.

"MAU" is an abbreviation for Medium Attachment Unit; it defines certain capabilities of a port. For a detailed explanation, please consult chapter 22, "Annex: Switch Configuration"22

7.2.5 Create New HMI

This section describes the specific part of configuring an HMI and is based on the general description for creating an IED (see chapter 7.2.2, "Create New IED"):

1. Select the 'Substation' or 'IEDs' Navigation Tree, and select an element below which you want to create an IED. This could be either:
   - a 'Substation', 'Voltage Level' or 'Bay' element in the 'Substation' Tree
   - the 'Project' element in the 'IEDs' tree
Right-click the element -> the context menu for that element appears. Select ‘Create New IED’ -> ‘HMI...’:

2. This opens the ‘Create new IED’ dialog.

3. To configure the general part please see chapter 7.2.2 (“Create New IED”).

4. In the section ‘Role Parameters -> HMI’ you can specify one or several OPC servers which run on the HMI you are creating. To create or remove an OPC server right-click in the OPC server table and choose ‘Insert new OPC server’ resp. ‘Delete OPC server(s)’:

For each created OPC server, a new access point on the HMI will be created automatically. Like other access points this access points also appears in the “Access Point” tab.
7.2.6 Configure IEDs

Already created IED can be configured:

1. Select the IED you want to configure in:
   - 'Substation' Navigation tree
   - 'IEDs' Navigation tree
   - 'HMIs' Navigation tree

2. Right-click the IED -> the context appears. Select 'Configure IED…':

3. This opens the 'Configure IED' dialog:
4. You can now reconfigure the IED. In contrast to the ‘Create IED’ procedure (see Chapter 7.2.2, Create New IED), some input fields are disabled or have more restricted selection range:
   - The IED location is fixed.
   - The primary role and also further roles of the IED cannot be changed (for more detailed information see chapter 7.1.1, IED Roles).
   - Additional Role parameters can be set, e.g. in the above example, you can check the ‘Protection’ flag to define an IED specifically as ‘Protection IED’
5. Click ‘OK’ to adopt the changes or ‘Cancel’ to reject all changes.

7.3 Updating IEDs

7.3.1 Overview

Updating an IED involves importing the ICD or IID file for a given IED. IEDs can be updated:
- from the Main menu -> IEDs tab -> Update IEDs
- from the context menu of an IED (single IED only)
- from the context menu of an element that can have IEDs below it, e.g. Substation, Voltage Level or Bay

7.3.2 Compare IEDs

The ‘Compare’ dialog provides additional information when re-importing SCL data. It is available when updating or re-importing SCL files into the project (e.g. Update IED). The dialog provides:
- preview of the content in import file
- compare differences between data in project and in the import file

The dialog is opened by clicking on the Compare column during Update IED, as shown below:
On the left-hand side of the dialog is the IED model in the import file and on the right-hand side is the existing IED model in the project. By selecting a node in either window, the details of the differences are shown in the bottom window.
7.3.3 Update one IED

1. Right-click on the IED you want to update -> the IED context menu appears:

2. Select ‘Update IED’ -> the normal dialog for selecting a file appears.

3. Select any valid SCL file (SCD-, ICD- or IID-file) and click ‘Open’.

4. The following dialog appears:

If the file contains only one IED (ICD- or IID-file), the source IED (in the file) will automatically be mapped to the destination IED (in IET600). If the file contains several IEDs, IET600 will try to automatically match IEDs with the same name. For those IEDs in IET600 where no IED with matching name is found, you need to select the source IED manually.

- You may select the same source IED to be imported into several destination IEDs; in some situations, this may be helpful.
- You can also override any automatic mapping done by IET600.

The ‘Compare’ column indicates whether the IET600 IED and the IED in the file are equal (==) or different (<>). You can click on the symbol, which will open the Compare dialog as described in the previous section.

If IET600 detects something which might be due to an error (e.g. if the source IED has a different manufacturer or IED Type, this might indicate that the wrong ICD/IID file has been selected), it shows the ‘Error’ indicator in the ‘Update’
Column and unchecks this column. Putting the mouse over the error indicator shows possible errors:

You may now either 'Cancel', compare both IEDs for differences, or simply override the warning by checking 'Update' manually.

5. To check the differences between the Source and the Destination IED, click the button in the 'Compare' column. The following dialog will open:

Check the differences. 'Close' will return you to the 'Update IED(s)' dialog.

6. To import the IED from the file, ensure that 'Update' is checked, then click 'OK'. The IED will be imported without further questions.

To close the dialog without importing anything (e.g. if you found unexpected differences when comparing IEDs as described above), click 'Cancel'
7.3.4 Update Several IEDs Together

1. Right-click on a Project, Substation, Voltage Level or Bay -> the context menu appears:

2. Select ‘Update IEDs…’ -> the normal dialog for selecting a file appears.

3. Select any valid SCL file (SCD-, ICD- or IID-file) and click ‘Open’.

The following dialog appears:

IET600 will try to automatically match IEDs with the same name in IET600 and in the file. For those IEDs in IET600 where no IED with matching name is found, you need to select the source IED manually.

- You may select the same source IED to be imported into several destination IEDs; in some situations, this may be helpful.
- You can also override any automatic mapping done by IET600.

The ‘Compare’ column indicates whether the IET600 IED and the IED in the file are equal (==) or different (<>). You can click on the symbol, which will open the Compare dialog as described in the previous chapter.
Chapter 7

The ‘Update’ column allows you to select/deselect individual IEDs for updating. Only IEDs marked with ☑ will be included in the update.

The ‘Error’ indicator in the ‘Update’ column indicates potential import problems which you can evaluate as described in the previous chapter.

Be aware that any error will unselect the IED for an ‘Update’ to avoid accidentally overwriting the current configuration with a faulty one.

4. Click ‘OK’ in the ‘Update IEDs’ dialog to update the selected IEDs. Click ‘Cancel’ to close the dialog without updating any IEDs.

7.4 Deleting IEDs

7.4.1 Overview

Basically, you can delete any IED from its context menu. You can either delete the IED totally or you can remove its content, leaving the IED empty.

7.4.2 Delete an IED

1. Right-click on the IED you want to delete -> the context menu appears:
2. Select ‘Delete’ -> a confirmation dialog appears:

3. Confirm deletion with ‘Yes’ or cancel it with ‘No’.

7.4.3 Delete the Contents of an IED

1. Right-click on the IED whose contents you want to delete -> the context menu appears:

2. Select ‘Delete Content’ -> a confirmation dialog appears:

3. Confirm deletion with ‘Yes’ or cancel it with ‘No’.
7.5 Mapping LNs to Substation Elements ("LN Mapping")

7.5.1 Overview

According to part 6 of the IEC 61850 standard, the SCD file shall describe not only the system specification in terms of the single line diagram and configured IEDs in terms of logical nodes (LNs), but also the relation between these LNs to parts and equipment of the single line. The LN Mapping table in IET600 serves as editor to configure this relation. The table is available in the Single Line Diagram (SLD) editor.

Since the LN mapping information from the ICD/IID file is not available in IET600, it must be entered manually. Note that this information must be synchronized with the functional/LN mapping used in the IED configuration tool (e.g. PCM600).

By default, the LN Mapping editor is displayed when displaying a Bay, but hidden when displaying a Voltage Level. To override this behavior, use the 'Show/Hide LN Mapping Editor' menu item in the 'Single Line Diagram' menu:

The LN mapping table is shown on the right hand side of the SLD editor, as shown below:

Each row in the table refers to a LN and its properties. The following properties are shown:

- **Mapping Target** – reference to the Bay and primary equipment
- **IED Name, LD Name, LN Name, LN Prefix, LN Class, Instance** – LN properties
- **Status** – indicates the source of the LN:
- ‘resolved’ indicates a LN from an IED in the project,
- ‘virtual’ indicates that a LN has been defined from an HMI signal and is not yet mapped to an IED

The last point refers to a concept of decoupling between the IEC-61850-model of an IED and its HMI signals, which is described in the chapter on HMI Engineering.

To map LNs from an IED which belongs to another Bay, select the field ‘Add LNs from IED’ and select another IED in the project.

To rapidly map all LNs to the Bay and then map some few LNs individually to Equipment (typically the CSWIs and some measurement LNs), use ‘Fill Down’ from the cell context menu.

### 7.5.2 LN Mapping Restrictions

IET600 places some restrictions to the LN Mapping Editor. These restrictions were implemented to improve the quality of the substation section in the SCD file. Some common restrictions are:

- LN type CSWI can only be mapped to equipment of type Circuit Breaker (CBR) or Disconnector (DIS); each Circuit Breaker (CBR) or Disconnector (DIS) can have only one CSWI allocated.

- LN type XSWI can only be mapped to equipment of type Disconnector (DIS); each Disconnector (DIS) can have 1..4 XSWI allocated
  
  This is to allow the modelling of phases; however, it is better practice to model the phases and attach 1 XSWI to each phase and 1 - the common XSWI calculated from the phases – to the DIS itself.

- LN type XCBR can only be mapped to equipment of type Circuit Breaker (CBR); each Circuit Breaker (CBR) can have 1..4 XCBR allocated
  
  This is to allow the modelling of phases; however, it is better practice to model the phases and attach 1 XCBR to each phase and 1 - the common XCBR calculated from the phases – to the CBR itself.

- RSYN can only be mapped to equipment of the type CBR; each CBR can have 1..5 RSYN allocated (for max. 2 control and 3 protection IEDs)

- LN type CILO can only be mapped to equipment of the type Circuit Breaker (CBR) or Disconnector (DIS); each Circuit Breaker (CBR) or Disconnector (DIS) can have several CILO allocated.
  
  This is to allow a distributed interlocking logic where different CILOs may be used to provide the complete interlocking logic.

- LN type MMXU can only be mapped to equipment of the type Current Transformer (CT) or Voltage Transformer (VT).

- LN type LLN0 and LPHD can only be mapped to the Bay
7.5.3 Map LNs in Bays

1. Select the 'Substation' tab in the Navigation Tree

   In the Navigation Tree, select the element you want in the configuration Tree (Voltage Level or Bay).

2. Select the 'SLD' editor.

   By default, the LN Mapping editor is displayed when displaying a Bay, but hidden when displaying a Voltage Level. If it is hidden, make it visible from the tab Single Line Diagram -> Show/Hide LN Mapping Editor.

   On the right of the single line diagram is the LN Mapping Editor. The first column shows the mapping target and the remaining columns to the right show the available LNs and their properties. The first column allows you to select a Substation item (Bay or Equipment) to map to that LN.

   Note: to select the LN Mapping target, one can either type in the target, select it from a drop down menu, or use the 'Fill Down' function to fill multiple rows automatically. IET600 will apply the LN mapping restrictions as described in the previous section.
7.6 Disturbance Recording Configuration

7.6.1 Introduction

The configuration for Disturbance Recording can currently only be used by MicroSCADA via '.sasmsc' file. If you need any other program to upload Disturbance Recording files from an IED for further processing, a configuration in IET600 will currently not be helpful.

In the following description, the following terms will be used:

- Disturbance Recording Server for an IED that creates Disturbance Record files (typically a Protection IED).
- Disturbance Recording Collector for an IED that uploads Disturbance Recording files from individual Disturbance Recording Servers and collects them in a defined place in the file system.

7.6.2 Configuration

The configuration for Disturbance Recording is done in the 'IED Settings' Editor. Disturbance Recording can only be configured for:

- IEDs classified as “Protection IEDs” (see chapter 7.2.6, “Configure IEDs” for more details), these IEDs act as Disturbance Recording Servers.
- OPC Servers, these IEDs act as Disturbance Recording Collectors.

If all values are greyed out, preventing you from configuring individual settings, please check whether one of the above conditions is fulfilled.

<table>
<thead>
<tr>
<th>Column</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>Disturbance Recording Enabled</td>
<td>Checking this flag on a Disturbance Recording Server (a Protection IED) will indicate to a client that it should try to upload Disturbance Records from this IED. The flag is enabled only. Checking this flag on a Disturbance Recording Collector (OPC Server) will indicate that it is an active client.</td>
</tr>
<tr>
<td>Disturbance Recording Delete Recording After Upload</td>
<td>Checking this flag will cause a Disturbance Recording Collector to try and remove Disturbance Records from an IED after a successful upload. When having several Disturbance Recording Collectors doing uploads from an IED, it is important to set this flag correctly (normally only one Collector should have this flag set).</td>
</tr>
</tbody>
</table>
### Disturbance Recording Use IEC 61850 Default Directory

IEC 61850 specifies a fixed directory on a protection IED, from where Disturbance Records can be retrieved by a Disturbance Recording Collector: the IED must have a Logical Device containing an LN of class RDRE; the path is then [LDName]\COMTRADE.

Checking this flag on a Disturbance Recording Server will cause the Disturbance Recording Directory to be set to this fixed value.

### Disturbance Recording Directory

The directory where the Disturbance Recording files are stored.

On a Disturbance Recording Server, it is the directory from where the Disturbance Recording Collector will try to upload files. This value is read-only, if the ‘Disturbance Recording Use IEC 61850 Default Directory’ flag is checked. See chapter 7.6.3 below for detailed recommendations how to set this path.

On a Disturbance Recording Collector (currently only supported for ABB OPC Servers), it is the root directory where uploaded Disturbance Record files will be saved, (the substructure may vary between different Disturbance Recording Collectors).

### Disturbance Recording Polling Period

On a Disturbance Recording Server, the minimum sensible polling period. Used for Legacy purposes; mostly an event-triggered upload is used.

On a Disturbance Recording Collector, the actual polling period, should not be smaller than the largest minimum polling period of any Disturbance Recording Server.

Used for legacy purposes; today, mostly an event-triggered upload is used.

### Disturbance Recording MMS Upload Capability

A flag indicating whether Disturbance Records can be uploaded via 8-MMS protocol (IEC 61850).

For IEC 61850 Ed.2 IEDs, this is read-only, as it is defined in the Services of the IED.

For IEC 61850 Ed.1 IEDs, the default value is taken from the IED Capabilities. If several Capabilities are selected, 8-MMS will be preferred over FTP over FTPS.
### Disturbance Recording Directory Recommendations

Be aware that the handling of the Disturbance Recording Directory on a Disturbance Recording Server is currently not yet consistent. As a guidance we suggest to use the following paths (without the ""):

- For IED611 series, use “COMTRADE\" as path (an empty path will upload unexpected files).
- For IED615 series, use “COMTRADE\" as path (an empty path will upload unexpected files).
- For IED620 series, use “COMTRADE\" as path (an empty path will upload unexpected files).
- For most other ABB IEDs, use “*” (the IED will automatically use its internally known path).
- For many 3rd party IEDs, try “*” (the IED will automatically use its internally known path). If this does not work, consult the manual.
- For Ed.2-compliant IEDs, you may try “Use IEC 61850 Default Directory”; however, the current experience is that this fails fairly often; also this is not useful when using protocols other than 8-MMS for upload. Always test this setting, and use “*”, if it does not work.
7.6.4 Export

The engineer does not need to export this information separately, as it will automatically be included in an `.sasmsc` file, provided that the following conditions are met:

- at least one OPC Server belonging to the MicroSCADA must be configured as a Disturbance Recording Collector by setting its ‘Disturbance Recording Enabled’ flag to true.

7.7 Dynamic Datasets

This flag is currently just used for the OPC Server as an RCB Client, to specify whether it should initialize dynamic Datasets from an 8-MMS Server IED (the information is included in the *.sasmsc file only, see chapter 16.6.8, "Import a MicroSCADA ‘sasmsc’ File").

| Dynamic Datasets | A flag indicating whether Dynamic Datasets should be queried for by the MicroSCADA OPC Server |

If the IED not have a server, only a Client Access Point, the flag is in intermediate state. It should be left as such.

In any other situation, the flag is set to “false”. In this case, the MicroSCADA OPC Server will not query for Dynamic Datasets (as this is the most common situation). You may set the flag to “true” which will cause the MicroSCADA OPC Server to query the Server IED. This behavior can be enforced even though the Server IED may not provide Dynamic Datasets, it is the responsibility of the engineer to configure this correctly.

7.8 SNMP Configuration Usage

The SNMP Configuration can currently only be used by MicroSCADA and its new SNMP-OPC-Gateway via `.sasmsc` file.

If you need any other program to supervise IEDs via SNMP (including the Obermeier OPC-SNMP Gateway), a configuration in IET600 will currently not be helpful.

The parameters described below are used to tell the SNMP-OPC-Gateway how to communicate with the corresponding Server IEDs.
### 7.8.1 SNMP Properties Configuration

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SNMP Write Community</td>
<td>String specifying the password to allow writing via SNMP. In SNMPv3, proper authentication is used. In older SNMP versions, this string is equivalent to a cleartext(!) password: the client must send this string along with any request, the server will only respond if the string/password is known to it. In IET600, the string is defined as the one that the client will send to the Server IED. The Server IED cannot currently be configured with IET600, the engineer must therefore find out what string the Server IED expects and configure it appropriately. The string “private” is a default setting of most SNMP Servers which allow to write; it does not mean that writing is disabled (!). It is good practice for each network not to use such default strings; however, be aware that the string is readable in plain text in the IET600 configuration database, the SNMP-OPC-Gateway configuration file and on the network.</td>
</tr>
<tr>
<td>SNMP Read Community</td>
<td>String specifying the password to allow reading via SNMP. In SNMPv3, proper authentication is used. In older SNMP versions, this string is equivalent to a cleartext(!) password: the client must send this string along with any request, the server will only respond if the string is known to it. In IET600, the string is defined as the one that the client will send to the Server IED. The Server IED cannot currently be configured with IET600, the engineer must therefore find out what string the Server IED expects and configure it appropriately. The string “public” is a default setting of most SNMP Servers which allow to write; it does not mean that writing is disabled (!). It is good practice for each network not to use such default strings; however, be aware that the string is readable in plain text in the IET600 configuration database, the SNMP-OPC-Gateway configuration file and on the network.</td>
</tr>
<tr>
<td>SNMP Poll Rate</td>
<td>Minimum seconds between two polls from the SNMP Client to the SNMP Server.</td>
</tr>
</tbody>
</table>
### Property Description

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SNMP Timeout</td>
<td>Seconds for Timeout if an SNMP Server does not respond to a request from an SNMP Client.</td>
</tr>
<tr>
<td>SNMP Version</td>
<td>SNMP Protocol Version to communicate. Can be SNMPv1, SNMPv2c or SNMPv3</td>
</tr>
<tr>
<td>SNMP Authentication Protocol</td>
<td>Algorithm for authentication of client to server.</td>
</tr>
<tr>
<td>SNMP Encryption Algorithm</td>
<td>Algorithm for encryption of communication between client and server.</td>
</tr>
<tr>
<td>SNMP User Name</td>
<td>User name used for authentication</td>
</tr>
<tr>
<td></td>
<td>(As IET600 does not provide suitable protection – such as encryption of data – and as it is not compatible with current security practice to have plaintext passwords, it is not possible to enter the password for the user here).</td>
</tr>
</tbody>
</table>

#### 7.8.2 SNMP Configuration of Signals

The above-mentioned attributes apply to the SNMP Client. Additionally, the individual Signals need to be configured with OIDs, see chapter 12.4.6 (Mapping Signals to SNMP OIDs).
7.9 PCM Template Usage

IET600 provides two attributes which can be used to refer to templates to be used in PCM when exchanging SCL Data with PCM600. The usage of such Templates and object types is entirely determined in PCM600; IET600 only provides a way to specify these templates as names for each IED.

7.9.1 PCM Template Configuration

The configuration for PCM Templates is done in the 'IED Settings' Editor. The following two columns are provided:

<table>
<thead>
<tr>
<th>PCM Template</th>
<th>Can be used to refer to a matching template in PCM. Set or change only in collaboration with the person doing PCM600 engineering.</th>
</tr>
</thead>
<tbody>
<tr>
<td>PCM Object Type</td>
<td>Can be used to refer to a matching object type in PCM. Set or change only in collaboration with the person doing PCM600 engineering.</td>
</tr>
</tbody>
</table>
8 IEC 61850 Subnetwork Engineering

8.1 Introduction

A ‘Subnetwork’ is an artificial item defined by the IEC 61850 standard. It serves basically to define a collection of IEDs which all talk the same protocol. The protocol is defined on the Subnetwork level, IEDs attached to the Subnetwork are assumed to talk the corresponding protocol.

Therefore a Subnetwork is a logical grouping of IEDs talking the same protocol. It must not be confused with the TCP/IP Subnet or with the physical network.

8.2 Subnetwork Conditions and Limitations

8.2.1 IEC 61850 Subnetworks

IEC 61850 requires that all IEDs using any protocol defined in IEC 61850 (8-MMS, GOOSE or Sampled Values) are attached to the same Subnetwork.

IEC 61850 historically did not distinguish between those protocols. One needs to look at the IED services to find out whether an IED is capable of talking 8-MMS, GOOSE and/or Sampled Values.

8.2.2 Modelling of Other Protocols

An IED may be capable of different protocols (e.g. 8-MMS/GOOSE for IEC 61850, FTP for Disturbance Recording upload and SNMP for Network supervision).

However, the number of protocols which can be modeled via Subnetworks is limited by its Access Points. It is quite common that an IED, although capable of several protocols, has only 1 Access Point. Its protocol capabilities (e.g. as described in the example above) cannot be modelled fully, as the System Configuration Tool is not allowed to add, modify or remove Access Points.
8.3 Subnetwork Configuration

The 'Communication' Navigation Tree displays the Subnetworks and IEDs (or – more precisely - IED access points) assigned to the Subnetwork. The 'Subnetworks' editor shows engineering information related to the IED Access points, including:

- Subnetwork name
- IP address
- IP subnet mask

The following section will describe how to build this communication structure and define the communication properties in IET600.

8.3.1 Create a Subnetwork

1. Open the 'Communication' menu tab and select the 'Subnetworks' button:

2. Alternatively, select the 'Communication' tab in the Navigation Tree, right-click on the project and select 'Create New Subnetwork':

![Create New Subnetwork](image1.png)

![Create New Subnetwork](image2.png)
The following dialog appears:

(A Subnetwork can only be created in the project which is pre-selected).

The tool proposes a name, which you can adapt according to your wishes. Invalid names are indicated by an error mark (Invalid names typically are either duplicate names or contain characters that are not allowed for a particular element).

3. Configure the protocol type for this Subnetwork from the available selection.

4. Click 'OK' to create the new Subnetwork or 'Cancel' to close the dialog without creating a Subnetwork.

5. If you created a new Subnetwork, the Navigation Tree will change to 'Communication' and the Subnetwork will appear there. The editor will not change.

**8.3.2 Connect IEDs to a Subnetwork in the Subnetwork Editor**

Select a Navigation Tree where the Subnetwork Editor is visible ('Substation', 'IEDs' or 'Communication' tab); select an appropriate element (usually the 'Communications' tab and the Project or a Subnetwork is a good choice).
1. Select the 'Subnetwork' editor:

<table>
<thead>
<tr>
<th>IED Name</th>
<th>AP</th>
<th>IP Address</th>
<th>IP Subnet</th>
<th>MACAddress</th>
<th>Unit Number</th>
<th>Subnetwork</th>
<th>Status</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>IED1</td>
<td>AP1</td>
<td>192.168.1.10</td>
<td>192.168.1.0</td>
<td>0x12345678</td>
<td>1</td>
<td>Subnetwork1</td>
<td>Active</td>
<td>Connected</td>
</tr>
<tr>
<td>IED2</td>
<td>AP2</td>
<td>192.168.2.10</td>
<td>192.168.2.0</td>
<td>0x12345678</td>
<td>2</td>
<td>Subnetwork2</td>
<td>Active</td>
<td>Connected</td>
</tr>
</tbody>
</table>

2. From the combo box with available Subnetworks, select the Subnetwork to which you want to add the IED/Access Point. To disconnect it, select the empty row.

After exiting the Combo Box, the Communication Tree will be updated, the IED (or, more precisely, its 'Connected Access Point' can be seen under the new Subnetwork).

An IED can have several Access Points which can be connected to different Subnetworks. It is not allowed, however, to have a Server IED with a Server with two Access Points and connect them both to the same Subnetwork.

Use the 'Fill Down' mechanism to rapidly connect multiple IEDs/Access Points to a Subnetwork. Use an appropriate sorting mechanism to avoid mapping two Access Points of the same IED to the same Subnetwork, e.g. by sorting first on the Access Point and the on the IED.

IEDs of different IEC 61850 versions should not be put to the same subnetwork as this causes compatibility problems. For more details, please read chapter 27.4 (IET600 Mixed System Engineering)
8.3.3 Connect IEDs to a Subnetwork in the Communication Tree


2. Drag’n’drop an IED (or, more precisely, its “Connected Access Point”) to the Subnetwork you want to connect it to. Drag’n’drop it to the Project node to disconnect it from any Subnetwork.

IEDs of different IEC 61850 versions should not be put to the same subnetwork as this causes compatibility problems. The indicator next to the subnetwork icon gives a hint:

<table>
<thead>
<tr>
<th>Indicator</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>E1:</td>
<td>contains Editions 1 IEDs only.</td>
</tr>
<tr>
<td>E2:</td>
<td>contains Editions 2 IEDs only.</td>
</tr>
<tr>
<td>M:</td>
<td>‘mixed’, contains both Editions 1 and Editions 2 IEDs; this should be avoided.</td>
</tr>
<tr>
<td>(empty):</td>
<td>non-61850 subnetwork (e.g. NCC, SNMP)</td>
</tr>
</tbody>
</table>

For more details, please consult chapter 27.4 (IET600 Mixed System Engineering).
8.3.4 Edit IP addresses

1. Select the 'Subnetwork' Editor:

Enter the IP Addresses directly in the column 'IP Address'. Duplicate addresses will be marked with a red circle (Exception: OPC servers can have the same addresses as their related HMI).

If you need to edit additional address information (e.g. OSI addresses), click on the button in the 'Edit Addresses' column. A dialog will pop up which will allow editing additional address attributes. This should not normally be needed, though.
9 Communication Configuration Engineering

9.1 Introduction

The Communication Configuration Diagram (CCD) allows you to configure the physical communication topology of a project, i.e.:

- to arrange all IEDs of a whole project in one diagram
- to define physical ports of IEDs and their properties (e.g. redundancy type, plug type or position).
- to create connections between ports of IEDs, thereby designing the communication topology.

9.2 The ‘CCD’ Editor

The CCD is divided into two views and additionally contains a toolbar.

The ratio between the diagram view and the table view is adjustable by dragging the divider placed between the two views to the left respectively to the right.
Optionally the table view can be collapsed by clicking the button centered the divider element (see illustration above).

### 9.2.1 Diagram View

The diagram view is intended to design the topology by creating connections between IEDs.

The diagram view itself also is divided into 2 areas:

- the main area (colored in white) at the right side.
- the area for unplaced IEDs (colored in blue) at the left side. This area is only visible if one or more unplaced IEDs are available.

Additionally, the diagram has sections. These sections are intended to support the design of a clearly arranged topology; and they are also used for the auto-arrange mechanism (see chapter 9.5.2, Auto-Arranging IEDs).

The height of the diagram sections can be adjusted by dragging the section dividers upwards or downwards:

To hide the sections, unselect 'Show diagram sections' in the diagram context menu:
9.2.2 **Table View**

This view contains two tables. The table in the first tab lists all IEDs from the whole project and their main communication properties. The table in the second tab lists the ports of all IEDs.

In contrast to the IED configuration dialog (see chapter 7.2.6, Configure IEDs) which is intended to configure a single IED, the table view allows editing properties from several IEDs together (e.g. changing the Plug type of several ports with help of the Fill-down mechanism).

9.2.3 **Navigate Within Views**

In general, the selection in the diagram view and in the two table views are always synchronized. Selecting one or several items in the diagram view automatically selects the same items in both table views:

<table>
<thead>
<tr>
<th>Action in Diagram View</th>
<th>IED Table View</th>
<th>Port Table View</th>
</tr>
</thead>
<tbody>
<tr>
<td>Select an IED</td>
<td>IED row will be selected.</td>
<td>All ports of IED will be selected.</td>
</tr>
<tr>
<td>Select a Connection</td>
<td>IEDs on which the connection is attached will be marked.</td>
<td>Ports on which the connection is attached will be marked.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Action in IED Table View</th>
<th>Diagram View</th>
<th>Port Table View</th>
</tr>
</thead>
<tbody>
<tr>
<td>Select an IED</td>
<td>Selected IED will be marked.</td>
<td>All ports of selected IED will be marked.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Action in Port Table View</th>
<th>Diagram View</th>
<th>IED Table View</th>
</tr>
</thead>
<tbody>
<tr>
<td>Select a Port</td>
<td>IED to which the port belongs and connection which is attached to the port will be selected.</td>
<td>IED to which the port belongs will be selected.</td>
</tr>
</tbody>
</table>

All actions described above can be performed on one or several elements. Also the two actions in the Diagram View can be combined.
9.2.4 Toolbar

The toolbar contains several auxiliary tools for dealing with the communication configuration diagram, such as:

- zooming / panning
- printing (including print configuration)
- exporting an image of the diagram
- creating rectangle shapes and annotation in the diagram
- editing diagram properties (diagram size, format)

By default, the toolbar only displays the last used ‘print out’ and ‘create diagram shape’ option. To open the full menu of this tools move the mouse over the button without clicking it and then select the desired option.

9.2.5 IED Shapes

In the Communication Configuration Diagram, the IEDs are displayed with different shapes depending on their primary role:

<table>
<thead>
<tr>
<th>IED Shape</th>
<th>Associated Roles</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image1.png" alt="Image 1" /></td>
<td>IED (protection / control), Security Device, Archive</td>
</tr>
<tr>
<td><img src="image2.png" alt="Image 2" /></td>
<td>Switch, Hub, Router (shape resizes according to port count)</td>
</tr>
<tr>
<td><img src="image3.png" alt="Image 3" /></td>
<td>HMI (including OPC servers)</td>
</tr>
<tr>
<td><img src="image4.png" alt="Image 4" /></td>
<td>Gateway, NCC</td>
</tr>
<tr>
<td><img src="image5.png" alt="Image 5" /></td>
<td>Printer</td>
</tr>
<tr>
<td><img src="image6.png" alt="Image 6" /></td>
<td>Time Server</td>
</tr>
</tbody>
</table>
Some IED roles support redundant ports. To distinguish redundant ports from single ports they are shown in different colors:

- Single ports (and ports of switches) are black.
- Port A in a redundant pair of ports is green.
- Port B in a redundant pair of ports is red.

### 9.2.6 Detail View for Selected IED

If the user zooms out fully to show the whole substation in an overview, the IEDs are usually too small to display detailed information; therefore such information is hidden and only becomes visible at higher zoom levels.

To access detail information, select a single IED -> a ‘Details’ view appears in the lower right corner of the diagram:

The ‘Details’ view is independent of the actual zoom level. It disappears if either the single IED in the diagram is deselected, or if more than one IED is selected.

You can suppress this automatic popping up with a click on the ‘Hide detail view’ button.

### 9.2.7 Fade Unselected Items

If there are a lot of IEDs and connections between them, it is difficult to determine a single connection or a port on which this connection is attached.
To better be able to see one item and its connections, it is possible to dim unselected items (IEDs, Connections and other diagram elements). To do this, use one of the following possibilities:

- select the ‘Communication Configuration Diagram’ tab in the Main menu and enable the checkbox ‘Fade unselected items’.
- right-click in the diagram area and select ‘Fade unselected items’

If one or more elements are selected, only the selected items and all items connected to it will be displayed normally, all other elements will appear dimmed:

In the example above, one IED is selected. Two other IEDs connected to it are displayed normally, all other IEDs appear dimmed.

If no element is selected, all elements get visualized normally again.
9.2.8 Zoom Diagram

The diagram view provides a ‘zoom in’ / ‘zoom out’ functionality. This can be done via:

- The context menu in the diagram view:

- Keyboard shortcuts
  - Ctrl + Mouse wheel up to zoom in, Ctrl + Mouse wheel down to zoom out
  - Ctrl + '+' to zoom in; Ctrl + '-' to zoom out

- The toolbox menu (zoom in / zoom out button or slider):

- The Main menu:

To quickly zoom out as much as to see the complete diagram, use the option ‘Zoom to fit’ (Ctrl + 0) in the context menu or in the toolbox.
9.2.9 Magnify a Section

As an alternative way to zoom the whole diagram, it is also possible use a magnifying glass to **zoom in** the area around the mouse cursor. To activate the magnifying glass press and hold the function key ‘F2’. To deactivate the glass just release ‘F2’. Alternatively the magnifying glass button in the toolbar can be used:

To lock the magnifier glass press Ctrl + F2. The glass remains now until the function key F2 is clicked again or the ‘Esc’ key is pressed.

Note: As in normal mode you can also move IEDs, create and modify connections etc. while the magnifier glass is active.

9.2.10 Show Bird’s Eye View

The CCD toolbox offers a ‘bird’s eye view’, a small overview of the whole diagram independent the actual zoom level. The currently visible area of the main diagram is highlighted. With the Bird’s eye view you can rapidly navigate to another area of the diagram without having to zoom out and back in again.
The bird’s eye view can be displayed by clicking the corresponding Toolbar button:

9.2.11 Draw Rectangle Shape

To create a new rectangle (only possible inside the main diagram area):

1. Click the ‘Create new rectangle shape’ button (the cursor change its appearance to a crosshair cursor):

2. Press and hold the left mouse key in the desired left top corner from the rectangle.

3. Move the cursor to the right bottom of the rectangle and release the button.

To modify the size, select the rectangle and use the border marks to resize. To modify the position, select the rectangle at the center mark and move it to the target location.

To delete a rectangle shape select it and press the keyboard key ‘Delete’.
To configure rectangle properties, like fill color, border color, etc., open the rectangle editor:

- via the rectangle context menu item ‘Configure rectangle…’
- via shortcut Ctrl + E (exactly one rectangle must be selected)

![Rectangle Configuration](image)

### 9.2.12 Draw Annotation Shape

To create a new annotation (only possible inside the main diagram area):

1. Click the ‘Create new annotation shape’ button - the cursor change its appearance to a text cursor:

![Create New Annotation Shape](image)

2. Left click at the position in the main diagram you want to create the annotation.

3. Write the text with help of following special keys:
   - Press ‘Alt + Enter’ for a line break.
   - Press ‘Enter’ to finish the action.
   - Press ‘Esc’ to cancel the action (the annotation will be removed again).

To edit an annotation, double-click it again to switch to the edit mode again.

To modify the position of an annotation, select it and drag & drop it to the target location.

To delete an annotation shape, select it and press the keyboard key ‘Delete’ (also if the annotation is empty it will be deleted automatically).
To configure the properties like size, color, font type, etc., open the annotation editor:

- via the annotation context menu item ‘Configure annotation…’.
- via shortcut Ctrl + E (exactly one annotation must be selected).

### 9.2.13 Print Out Diagram

IET600 offers two alternative print options which can be triggered via toolbar menu or with keyboard shortcuts:

1. Print out the entire diagram (Ctrl + P).
2. Print out the currently visible diagram area (Ctrl + Shift + P).

The following diagram settings will also be used for the print out (set these properties before printing out the diagram):

- IED details (visibility depends on zoom level)
- IED, connection, rectangle and annotation visibility (normal, faded or highlighted)
- Diagram grid
- Diagram section lines
9.2.14 Export Image of Diagram

The diagram view can also be exported as an image. The image can either be saved to a path or be copied to the clipboard. Additionally it can be chosen if just the visible area of the diagram or the complete diagram is exported:

1. Save an image of the diagram to a specific path (complete diagram, or visible area only)
2. Copy an image of the diagram to the Clipboard (complete diagram, or visible area only)

The following diagram settings will be used for the image export (set these properties before exporting):

- IED details (visibility depends on zoom level)
- IED, connection, rectangle and annotation visibility (normal, faded or highlighted)
- Diagram grid
- Diagram section lines

9.2.15 Shortcut Keys

In the Communication Configuration Diagram are several shortcuts available:

<table>
<thead>
<tr>
<th>Key Combination</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>F2</td>
<td>Press to show magnifier glass, release to hide it</td>
</tr>
<tr>
<td>Ctrl + F2</td>
<td>Shows magnifier glass permanently (to hide it, press F2).</td>
</tr>
<tr>
<td>Ctrl + E</td>
<td>Configure selected diagram item (IED, Annotation or Rectangle)</td>
</tr>
<tr>
<td>Ctrl + 0</td>
<td>Adjust the zoom level to fit the whole diagram into the visible area.</td>
</tr>
</tbody>
</table>
9.3 Engineering Workflow

The following steps suggest a possible way to create the physical communication diagram:

1. Create IEDs
   - Typically, the IEDs are initially created without IEC 61850 data model.

2. Configure IEDs
   - To connect IEDs it is required to configure the communication model first. At least, the physical communication ports and their properties (redundancy, plug type, etc.) must be defined.

3. Place IEDs
   - Created IEDs are not positioned initially. For a quick positioning, IET600 offers several different ways for automatic placement. After this, you can rearrange them manually according your preferences.

4. Connect IEDs
   - Finally the IEDs can be connected physically via their Ports.

You can do the above-mentioned configuration steps at different times for different IEDs. For example, in a first cycle, you may want to design network and station level topology. (Substation-level Switches, HMIs, Gateways, Printers, Clocks etc.). In a second cycle you may then design bay-level IEDs (Protection and Control IEDs, Merging Units).
9.4 Creating and Configuring IEDs

To read about how to create and configure an IED in detail, consult chapter 7.2 (Creating and Configuring IEDs).

You can open the ‘Configure IED’ dialog described in the chapter from the CCD directly:

- via the IED context menu item ‘Configure IED…’
- via shortcut Ctrl + E (exactly one IED must be selected)

9.4.1 Configure Communication Properties

Open the ‘Create new IED’ or ‘Configure IED’ dialog. In the ‘Communication Parameters section’ all kind of communication properties can be defined:

1. Define the number of (physical) ports.

2. If two ports are redundant, define the second port as redundant to the first port by selecting the number of the first port in the ‘Redundant to’ column of the second port:
   - If no port is selected, the port is not redundant.
   - Be aware that in many redundancy protocols, first and second port have defined meanings (e.g. in HSR the first port of one IED needs to be connected to the second port of the next): define redundant ports with this in mind.
   - A port can be redundant to exactly one other port. In the drop down menu, only those ports, which do not already belong to such a redundant pair, are available for selection.
   - Not all roles support redundant ports (the column is disabled in this case).
3. If one or several port pairs are defined as redundant, select the redundancy protocol of the second port for each pair. The selected protocol will automatically be applied to the first port also (cell is read only).

4. Define one or several access points for each port:
   - A physical port or a redundant port pair can have several Access Points connected to them.
   - One Access Point can belong only to one port or one redundant port pair.
   - For a redundant port pair, configure the Access Points on the second port; The selected Access Point(s) will automatically be applied to the first port also (cell is read only).

5. Choose the plug type for each port.

6. Define where the ports are positioned. They can be basically be positioned left, right, top or bottom; some primary roles restrict this choice (e.g. in an HMI, ports can only be configured at the bottom due to the way its interior is displayed).

7. The order in which the ports appear in the diagram can also be changed for each side. This can be used to avoid crossing connections.

9.5 Placing IEDs

By default, IEDs are not placed in the Communication Configuration Diagram but in an extra section on the left side of the main diagram (blue colored).

Depending on their primary role, IEDs will be placed in different sections:

<table>
<thead>
<tr>
<th>Section</th>
<th>IED Roles</th>
</tr>
</thead>
<tbody>
<tr>
<td>NCC</td>
<td>NCC</td>
</tr>
<tr>
<td>Station</td>
<td>HMI, Gateways, Hub, Printer, Router, Security Device, Time Server, Archive</td>
</tr>
<tr>
<td>Network</td>
<td>Switch (except bay-level switch*)</td>
</tr>
<tr>
<td>Bay</td>
<td>IED (protection / control), Bay level Switch*</td>
</tr>
<tr>
<td>Merging Unit</td>
<td>Merging Unit</td>
</tr>
</tbody>
</table>

* A switch is handled as bay-level switch if it is placed beneath a Bay in the substations tree.
Chapter 9

IEDs can be placed and moved manually by drag & drop or also automatically. The 'auto arrange' functions (see detail description below) are mainly intended for a first positioning of one or several IEDs. After this, you can manually move the IEDs according your preferences.

9.5.1 Move an IED Manually

To place an IED manually:

1. Left-click the IED you want to move.
2. Press and hold the left mouse button and move the IED to new location.
3. Release the mouse button to move the IED definitely; or press the 'Esc' Key before releasing the mouse button to abort the move procedure (this sets IED back to the original position)

The move procedure will be automatically aborted if any of the following conditions is fulfilled while releasing the mouse:

- Moved IED overlaps any other IED.
- Moved IED overlaps any connection which isn’t attached to this IED.
- Moved IED is not fully inside the diagram sheet.

All connections attached to the moved IED will be rerouted automatically.

9.5.2 Auto-Arranging IEDs

The ‘Auto Arrange' functionality allows a rapid initial placement of a considerable number of IEDs.

There are three variations of auto-arrange that can be called via the context menu of the communication configuration diagram:

- Arrange all IEDs
- Arrange selected IEDs (only enabled if at least one IED is selected)
- Arrange all unplaced IEDs (only enabled if at least one IED is still not placed in the diagram)

‘Auto Arrange' uses the following rules for the placement of IEDs:

- IEDs allocated within one Bay are placed as a group close together.
These groups are then placed in an alphabetical order according to the Bay name.

IEDs are placed in sections depending on their roles (which primary role in which section is described in chapter 9.6 Connecting IEDs)

IET600 allows creating physical connections between two IED ports. There are some restrictions:

- A connection must be attached always to exactly two ports. It is not possible to create “dangling” connections (i.e. connections attached to only one or no port).
  
  (Deleting an IED will also automatically delete all links attached to a port of this IED; thereby preventing “dangling” connections).

- A connection must always be between two different IEDs, connections within an IED are not allowed.

- Connection segments are always horizontal or vertical, creating a diagonal connection or a diagonal connection segment is not possible.

Note: Moving an IED provokes an automatic rerouting of all connections which are attached to this IED. All modifications that have been made manually will be lost.

9.6.1 Add Connection

1. Move the mouse cursor over the first port you want to connect - the cursor changes its appearance to a ‘pointing hand’.

2. Press and hold the left mouse button and move the mouse cursor to the second port. A ‘pointing hand’ cursor symbolizes a port which can be connected, otherwise a ‘prohibited’ cursor appears.

3. Release the left mouse button while mouse cursor is over the second port.

9.6.2 Remove Connection

1. Select one or several connections.

2. Press the ‘Delete’ Button.

3. A confirmation dialog will appear. Select ‘yes’ to delete the connection(s) or ‘no’ to cancel the action.
9.6.3 Reconnect Connection

1. Select a single connection.
2. Move the mouse cursor over the port from which you want to detach the connection -> cursor changes to a four-way arrow symbol.
3. Press and hold the left mouse button and drag the connection to another free port.
4. Release the left mouse button.

9.6.4 Reroute Connection Manually

Connections can be rerouted manually:

1. Select a single connection.
2. Move the mouse cursor over a via-point of the selected connection (cursor change to four way arrow symbol).
3. Press and hold the left mouse button and shift the via-point to desired position.

As start point or end point of a connection cannot be rerouted, moving a via-point next to a start point or end point is also restricted:

1. Via-point cannot be moved horizontally or vertically
2. No via-point available, connection cannot be rerouted
3. Via-points can be moved only in horizontal direction

For rerouting a connection, add first one or several via points (see chapter 9.6.5, Add / Remove Connection Via-Point).

9.6.5 Add / Remove Connection Via-Point

To add a via-point:

1. Select a connection.
2. Double click the connection where you want to add a via-point.

To remove a via-point (Note: only via points which are in line with previous and next via point can be removed):

3. Select a connection.
4. Double click the via-point you want delete to.
10 Dataflow Engineering

10.1 Introduction

10.1.1 Overview

Dataflow is a general term for the functionality and configuration that is needed to exchange data between different IEDs. To configure the dataflow, the data model of the participating IEDs and the communication (the connection between the IEDs) must be more or less defined.

In IET600, there are several ways to engineer Datasets, RCBs and GCBs:

- create Datasets, RCBs and GCBs manually, and then add/edit their data.
- copy an IED with configured Datasets, RCBs or GCBs from the same or another project.
- import a configured Dataset, RCB or GCB from an IED SCL (CID/IID) file.

10.1.2 Engineering Approach

The IEC 61850 standard distinguishes:

A Dataset that specifies what data are sent. It can basically be understood as a defined sequence of data items that can be sent as a whole packet only.

A Control Block that specifies how the data are sent. IEC 61850 defines:

- Report Control Blocks (RCBs)
- GOOSE Control Blocks (GCBs)
- Sampled Value Control Blocks (SVCBs)
- GSSE Control Blocks (obsolete, should not be used)
- Log Control Blocks
- Setting Group Control Blocks

IET600 stresses a slightly different approach. From an engineering point of view, it is helpful to distinguish:

Vertical communication from bay-level IEDs to HMIs or NCC Gateways. The vertical communication includes:
• configuring Datasets
• configuring Report Control Blocks (RCB)
• configuring Clients for the RCB in a matrix

Horizontal communication, essentially to exchange information between IEDs. The horizontal communication includes:
• configuring Datasets
• configuring Goose Control Blocks (GCB)
• configuring receivers for GCB in a matrix

Analogue data communication (Sampled Values), essentially between IEDs transforming analogue measurement values into data streams, and IEDs. The analogue data communication includes:
• configuring Datasets.
• configuring Sample Value Control Blocks (SVCB).
• configuring receivers for SVCB in a matrix.

10.1.3 Common Problems in Dataflow Engineering

In IEC 61850, so-called services define how Dataflow can be edited in an IED. These services – for historical reasons – may not always be correctly configured by the IED configuration tool or correctly interpreted by IET600. IET600 behaves very conservatively and often blocks creating or editing Dataflow elements in such cases.

To learn how to get information about such services, please consult chapter 7.1.5 (IED Services Editor).

To learn how you may adapt IET600 in such cases to get the desired result, please consult chapter 7.1.6 (IED Capabilities Editor).
10.2 Dataset Configuration

10.2.1 Datasets Editor

The Datasets editor provides the following views:

- all available Datasets and their properties
- Dataset entries of selected Dataset
- a selection editor to allow adding/removing Dataset entries

The following Dataset properties are shown:

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type</td>
<td>IED Type</td>
</tr>
<tr>
<td>IED</td>
<td>IED Name</td>
</tr>
<tr>
<td>AP</td>
<td>Access Point name</td>
</tr>
<tr>
<td>Srv</td>
<td>Server name</td>
</tr>
<tr>
<td>LD</td>
<td>Logical Device where RCB is defined</td>
</tr>
<tr>
<td>LN</td>
<td>Logical Node where RCB is defined</td>
</tr>
</tbody>
</table>
### Create Datasets

1. Select an appropriate IED in the ‘Substation’, ‘IEDs’ or ‘Communication’ Navigation Tree where the Dataset shall be created. To have the full choice of IEDs, select the project.

   Alternatively, this dialog can also be opened from the ‘Communication’ tab in the Main menu, using the ‘Create Dataset(s)’ menu item:

2. Select the ‘Datasets’ Editor

3. Select a row and right-click to select ‘Insert New Row’ from the context menu.

4. A dialog ‘Create New Datasets’ is opened. The Tree in the dialog will show all IEDs below the element you selected. Navigate to the IED and then to the LN where you want to create the new Dataset. For ABB IEDs, Datasets are normally allocated to L00/LLN0.
5. Once you select an LN, the tool proposes a name, which you can adapt according to your wishes. Invalid names are indicated by an error mark (Invalid names typically are either duplicate names or contain characters that are not allowed for a particular element).

6. Click ‘OK’ to create the new Dataset or ‘Cancel’ to close the dialog without creating a Dataset.

10.2.3 Edit Datasets

In the Dataset Editor, you can edit the name and the description of a Dataset. If you change the name, attached RCBs or GCBs will automatically be updated.

To attach a Dataset to a Control Block is currently not possible in the Dataset Editor; you have to use the RCB Editor or GCB Editor, respectively.

10.2.4 Add Dataset Entries

To add Dataset Entries, you need to use the lower part of the editor.

1. Select the Dataset to which you want to add entries. The first list below shows the available LDs:

2. Select LD -> a choice of LNs appears. Continue selecting LN, Data Objects (DOIs) and Data attributes (DAIs) as appropriate (it is recommended to use Data Objects for RCB-Datasets and Data Attributes for GCB-Datasets).

3. If the functional constraint is not unique yet by your selection, select it now. You can only select FCs that make sense for spontaneously sent data (e.g. ST or MX).

In the ‘Result List’, you see all the attributes that will be added to the Dataset. If it is empty, your selection is not valid yet (typically when the Data Object has
attributes with several Functional Constraints - e.g. CF and ST - and you have not selected the FC yet).

4. After verifying the attributes in the result box, click ‘Append’ or ‘Insert’ to add your selection to the Dataset entries.

As long as the error indicator appears, something is not (yet) correct in the selection. The tool tip of the error indicator provides additional information.

It is not possible to add the same data twice to the same attribute, e.g. you cannot add CMMXU1.A[MX] twice.

You can still add CMMXU1.A.phsA[MX] after having added CMMXU1.A[MX], although they partially contain the same data. Such a configuration usually does not make sense (some few clients even refuse to work correctly when receiving such Datasets); it is the responsibility of the engineer to avoid such configurations.

The maximum number of Dataset Entries in a Dataset is limited. The current and the allowed number of entries are indicated in the Editor. Also the total number of Dataset entries for the IED is displayed (for statistics only).

The maximum number of Dataset Entries is specified by a service in the SCL file, ConfDataset.maxAttributes. Historically, some IEDs were counting entries, others were counting attributes. To visually indicate which way a particular IED is counting, IET600 displays the value used by the IED in black and the other way greyed out. E.g. in the above example, the IED is counting entries.

For a more detailed explanation, please see chapter 26.3.2, Annex: IED Capabilities, Service ConfDataset, maxAttributes attribute.

10.2.5 Delete Dataset Entries

To delete a Dataset Entry, you have two possibilities:
1. Select the entries you want to delete in the Dataset Entry part of the editor. Right-click and select 'Delete Row(s)’ or press the ‘Delete’ key. A confirmation dialog appears, confirm the Deletion or cancel it. This is the only way to delete multiple Dataset entries in one action.

2. Select a single entry you want to delete in the Dataset Entry part of the editor. As long as the error indicator appears, something is not (yet) correct. The tool tip of the error indicator provides additional information.

3. Click the Button 'Remove' in the lower part of the editor. The entry is removed from the Dataset entries without further warning, but it is still preserved in the bottom part of the editor. You can append it again or insert it into some other place in the Dataset entry list.

10.3 RCB Data Configuration

10.3.1 RCB Data Editor

The RCB Data editor defines the parameters for the reporting service, according to IEC 61850 part 7-2.

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>IED Type</td>
<td>IED Type</td>
</tr>
<tr>
<td>IED</td>
<td>IED Name</td>
</tr>
<tr>
<td>Property</td>
<td>Description</td>
</tr>
<tr>
<td>------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>AP</td>
<td>Access Point name</td>
</tr>
<tr>
<td>Srv</td>
<td>Server name</td>
</tr>
<tr>
<td>LD</td>
<td>Logical Device where RCB is defined</td>
</tr>
<tr>
<td>LN</td>
<td>Logical Node where RCB is defined</td>
</tr>
<tr>
<td>RCB</td>
<td>RCB name</td>
</tr>
<tr>
<td>Identifier (rptID)</td>
<td>RptID uniquely identifies a RCB per client on each IED. By default it is calculated from the full identification (path name) of the RCB.</td>
</tr>
<tr>
<td>Status</td>
<td>Describes how the RCB was created. This attribute is automatically generated by IET.</td>
</tr>
<tr>
<td>Attached Dataset</td>
<td>Select the name of the Dataset, as specified in the Dataset Editor.</td>
</tr>
<tr>
<td>Conf.Rev.</td>
<td>Version number will be incremented on changes of the Dataset related to the control block. Changes are the deletion or insertion of a member of the Dataset, reordering of the member, or changing the Dataset reference of the control block. Changes will increment this value by 100.</td>
</tr>
<tr>
<td>Buffered</td>
<td>In general, only buffered reporting is used for all kinds of data, except for measurements.</td>
</tr>
<tr>
<td>Buffer Time (ms)</td>
<td>Defines the amount of time that the report is delayed (after a change) to wait for further changes before it is sent.</td>
</tr>
<tr>
<td>Enabled Clients</td>
<td>The number of instances or clients mapped to the RCB. This value is automatically calculated by IET based on the number of clients mapped in the RCB Clients editor. By default, there is always 1 additional RCB instance, unless number of clients = Max Enabled Clients.</td>
</tr>
<tr>
<td>DChg</td>
<td>TRUE specifies the condition to trigger reporting on Data Change, i.e. on value change</td>
</tr>
<tr>
<td>QChg</td>
<td>TRUE specifies the condition to trigger reporting on Quality Change.</td>
</tr>
<tr>
<td>DUpt</td>
<td>TRUE specifies the condition to trigger reporting on Data Update, i.e. whenever a value is updated, irrespective of whether it has changed or not.</td>
</tr>
</tbody>
</table>
### Property Description

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cyclic</td>
<td>TRUE specifies that the reporting for monitored data is <em>Cyclic</em>, it will be triggered every Cycle Time.</td>
</tr>
<tr>
<td>Cycle Time (ms)</td>
<td>Cycle time for cyclic-triggered reporting</td>
</tr>
<tr>
<td>GI</td>
<td>TRUE indicates that RCB/Dataset can be queried via General Interrogation.</td>
</tr>
<tr>
<td>SeqNum</td>
<td>TRUE indicates the chronological order of reports is used in RCBs</td>
</tr>
<tr>
<td>BufOvfl</td>
<td>TRUE indicates that a flag to indicate Buffer Overflow for the BRCB is used.</td>
</tr>
<tr>
<td>Dataset</td>
<td>TRUE indicates usage</td>
</tr>
<tr>
<td>Entry ID</td>
<td>TRUE indicates usage</td>
</tr>
<tr>
<td>Entry Time</td>
<td>TRUE indicates usage</td>
</tr>
<tr>
<td>DataRef</td>
<td>TRUE indicates usage</td>
</tr>
<tr>
<td>Reason Code</td>
<td>TRUE indicates usage</td>
</tr>
<tr>
<td>Config Rev.</td>
<td>TRUE indicates usage</td>
</tr>
</tbody>
</table>

### 10.3.2 Create RCBs

If you know where you want to create your RCB, you can preselect an appropriate element in the 'Substation', 'IEDs' or 'Communication' Navigation Tree. To have the full choice of IEDs, select the project.
1. Open the RCB Data editor and select a row in the RCB list, right-click and select 'Insert New Row' from the context menu.

2. Alternatively, select the 'Communication' menu tab (the 'Communication' Navigation tab will automatically be selected) and select 'Create RCB(s)'.

3. The dialog 'Create New RCB' opens. The Tree in the dialog will show all IEDs below the element you selected. Navigate to the IED and then to the LN where you want to create the new RCB. Note that for ABB IEDs, the RCBs are allocated to the LD0/LLN0 logical node.

4. Once you select an LN, the tool proposes a name, which you can adapt according to your wishes. Invalid names are indicated by an error mark (Invalid names typically are either duplicate names or contain characters that are not allowed for a particular element).
5. Click ‘OK’ to create the new RCB or ‘Cancel’ to close the dialog without creating an RCB.

If you created a new RCB from the Main Menu, the Navigation Tree will remain where it is and the Editor will change to ‘RCB Data’. The new RCB will have a standard configuration, which you may edit, if needed.

10.3.3 Delete RCBs

An RCB once created cannot be renamed or totally removed anymore. The reason is that, once it is exported, the RCB and its last configuration revision value (Column ‘Conf.Rev’, ‘configRev’ attribute in IEC 61850) needs to be remembered. If this is not done, an RCB with the same name and the same configuration revision number, but different content and/or behavior could be created at a later time. This would lead to confusion and even potentially dangerous situations.

So an RCB is “deleted” by setting its status to ‘deleted’. Deleted RCBs will remain in the tool forever, but will not be exported in SCD/ICD/IID files, and will not count when evaluating limits for RCBs.

1. Select the RCB Editor
2. Select any element that has RCBs below it in the Navigation Tree (the IED where you want to create it, is usually a good choice).
3. Select one or several rows with the RCBs intended for deletion.
4. Right-click and select ‘Delete Row(s)’ or press the ‘Delete’ key. A confirmation dialog appears, confirm the deletion or cancel it.

The GCBs will remain, but their Datasets will be removed and their status will go to ‘deleted’.

10.3.4 Edit RCBs

The RCB configuration can be edited in the RCB editor; however, this should normally not be necessary. There are several specialties to be observed:

- An RCB cannot be renamed. To rename an RCB, delete it and create a new RCB with the intended name.
- Removing a Dataset from an RCB will automatically put the RCB into a ‘deleted’ state.

Configuration changes will cause its configuration revision value (Column ‘Conf.Rev’, ‘configRev’ attribute in IEC 61850) to be increased to the next multiple of 100. This will happen only once between import and export, not with every change.
10.3.5 **Reuse “Deleted” RCBs**

A deleted RCB can be reused by simply attaching a Dataset to it. Instead of creating a new RCB, it is recommended to reuse “deleted” RCBs first.

10.4 **RCB Client Configuration**

10.4.1 **Prerequisites**

To do a sensible RCB client configuration, the potential clients and their communication configuration should be known (typically HMIs and NCC Gateways). Therefore add these IEDs (see chapter 12.2, HMI Creation and Configuration) and configure them to the Subnetworks before doing the RCB client configuration.

Bay-level IEDs can be added and configured easily also in later stages of the project.

10.4.2 **RCB Client Editor**

The rows of this editor show IEDs (or their Access Points, respectively) and RCBs. The rows depend on the element selected in the Navigation Tree.

The columns show the available Client IEDs (and their Access Points). Valid clients are IEDs that have IHMI-, ITCI-, ITMI- or IARC-LNs. The columns do not depend on the element selected in the Navigation Tree (i.e. all Client IEDs are always shown). The client IED names are vertical to save space. To avoid neck strains, use the Tooltip to see the text horizontally.
If a Client IED is not on the same Subnetwork as a Server IED or RCB, it cannot be configured as a Client. In the Editor, this shows as greyed-out cells that cannot be edited.

For those cells that can be edited, you can use ‘+’ or ‘*’ or ‘x’ to add an additional client to the existing ones. You can use numbers to change the client sequence or add clients. You can use ‘-’, the space bar or ‘Delete’ to delete existing clients. The result will always show the clients as sequential numbers from 1 to the highest configured client.

The reason for this is that the IEC 61850 standard does not allow you to reserve slots for later clients (e.g. to configure a sequence 1, 2, 4 with the intention to add a client 3 later). If you want to do that, you need to create a dummy client.

On the right, the editor has a set of Radio Buttons that allow fast filter setting (show either Default Clients or RCBs or both together; the latter is the default).

It also has some Buttons to allow fast, semi-automatic configuration of Default clients and RCB clients. The buttons operate only on the rows displayed in the editor, so by cleverly choosing objects in the Navigation tree and/or setting filters, one can do complex configurations for a big substation quite rapidly.

**10.4.3 Basic Workflow for RCB Client Engineering**

Configuring RCB Clients consists of two or three steps:

1. Configure the Default clients for each IED (or, more precisely, for its Access Points).
2. Propagate these defaults to all RCBs of the same IED/Access Point.
3. Configure individual clients for RCBs, if needed.
10.4.4 Configure Default Clients

For each IED, or rather, for each Access Point of an IED, Default clients should be configured. These “Default clients” can then easily be propagated to all RCBs in that IED/Access Point.

In the Navigation Tree, select any element that has IEDs/Access Points below it in the Navigation Tree (usually a Subnetwork, a Voltage Level or the whole project is a good choice).

1. Select the RCB Client Editor.

   If you have IEDs with more than one Access Points, make the column 'AP' visible in the RCB Client Editor.

   To have a better overview, you may choose to select ‘Show IEDs Only’.

2. Configure the IEDs/Access Points with Default Clients:
   - ‘Clear All’ removes all Default Clients.
   - ‘Configure Empty’ automatically fills out only Default Clients for those IEDs/Access Points that have no clients configured yet.
   - ‘Configure All’ deletes all existing Default clients and automatically fills them out afterwards.

3. Use the mechanisms described above (chapter 10.4.2, “RCB Client Editor”), if you need to individually configure RCBs.

10.4.5 Configure RCB Clients

Precondition: you must have Default clients configured, otherwise the automatic RCB client configuration will not work.
Chapter 10

In the Navigation Tree, select any element that has RCBs below it in the Navigation Tree (usually a Subnetwork, a Voltage Level or the whole project is a good choice).

1. Select the RCB Client Editor.

   If you have IEDs with more than one Access Points, make the column ‘AP’ visible.

   To have a better overview, you may choose to select ‘Show RCBs Only’.

2. Configure the RCBs with Clients.
   - ‘Clear All’ removes all RCB Clients in the editor.
   - ‘Configure Empty’ copies the Default Client Configuration of this IED/Access Point to its RCBs (only for those RCBs that have no clients configured yet).
   - ‘Configure All’ deletes the existing RCB clients and then copies the Default Client Configuration of this IED/Access Point to its RCBs.

   Use the mechanisms described in the previous chapter if you need to individually configure the clients of one or few RCBs.
### 10.5 LCB Data Configuration

#### 10.5.1 LCB Data Editor

The LCB Data editor defines the parameters for the reporting service, according to IEC 61850 part 7-2.

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>IED Type</td>
<td>IED Type</td>
</tr>
<tr>
<td>IED</td>
<td>IED Name</td>
</tr>
<tr>
<td>AP</td>
<td>Access Point name</td>
</tr>
<tr>
<td>Server</td>
<td>Server name</td>
</tr>
<tr>
<td>LD</td>
<td>Logical Device where LCB is defined</td>
</tr>
<tr>
<td>LN</td>
<td>Logical Node where LCB is defined</td>
</tr>
<tr>
<td>LCB</td>
<td>LCB name</td>
</tr>
<tr>
<td>Status</td>
<td>Describes how the LCB was created. This attribute is automatically generated by IET.</td>
</tr>
<tr>
<td>Attached Dataset</td>
<td>Select the name of the Dataset, as specified in the Dataset Editor.</td>
</tr>
<tr>
<td>Log Enabled</td>
<td>Enables the LCB to write its data into the specified Log.</td>
</tr>
<tr>
<td>Log Name</td>
<td>The name of the Log into which the LCB is writing.</td>
</tr>
<tr>
<td>Reason Code</td>
<td>Include the reason code in the log, if TRUE</td>
</tr>
<tr>
<td>DChg</td>
<td>TRUE specifies the condition to trigger logging on Data Change, i.e. on value change</td>
</tr>
<tr>
<td>QChg</td>
<td>TRUE specifies the condition to trigger logging on Quality Change.</td>
</tr>
<tr>
<td>DUd</td>
<td>TRUE specifies the condition to trigger logging on Data Update, i.e. whenever a value is updated, irrespective of whether it has changed or not.</td>
</tr>
<tr>
<td>Cyclic</td>
<td>TRUE specifies that the logging for monitored data is Cyclic, it will be triggered every Cycle Time.</td>
</tr>
<tr>
<td>Cycle Time (ms)</td>
<td>Cycle time for cyclic-triggered logging</td>
</tr>
</tbody>
</table>
10.5.2 Create LCBs

If you know, where you want to create your LCB, you can preselect an appropriate element in the 'Substation', 'IEDs' or 'Communication' Navigation Tree. To have the full choice of IEDs, select the project.

1. Open the LCB Data editor and select a row in the LCB list, right-click and select 'Insert New Row' from the context menu.

2. Alternatively, select the 'Communication' menu tab (the 'Communication' Navigation tab will automatically be selected) and select 'Create LCB(s)'.

3. The dialog 'Create New LCB' opens. The Tree in the dialog will show all IEDs below the element you selected. Navigate to the IED and then to the LN where you want to create the new LCB. Note that for ABB IEDs, the LCBs are allocated to the LD0/LLN0 logical node.
Once you select an LN, the tool proposes a name, which you can adapt according to your wishes. Invalid names are indicated by an error mark (Invalid names typically are either duplicate names or contain characters that are not allowed for a particular element).

4. Click 'OK' to create the new LCB or 'Cancel' to close the dialog without creating an LCB.

If you created a new LCB, the Navigation Tree will remain where it is and the Editor will change to 'LCB Data'. The new LCB will have a standard configuration, which you may edit, if needed.

10.5.3 Delete LCBs

1. Select the 'LCB Data' Editor
2. Select any element that has LCBs below it in the Navigation Tree (the IED where you want to create it, is usually a good choice).
3. Select one or several rows with the LCBs intended for deletion.
4. Right-click and select 'Delete Row(s)' or press the 'Delete' key. A confirmation dialog appears, confirm the deletion or cancel it.

Unlike other Control Blocks, LCB(s) will actually be deleted; they are not just marked as "Deleted".

10.5.4 Edit LCBs

The LCB configuration can be edited in the LCB editor; however, this should normally not be necessary. To be observed:

- An LCB cannot be renamed. To rename an LCB, delete it and create a new LCB with the intended name.
10.6 GCB Data Configuration

10.6.1 GCB Data Editor

The GCB Data editor defines the parameters for the reporting service, according to IEC 61850 part 7-2.

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>IED Type</td>
<td>IED Type</td>
</tr>
<tr>
<td>IED</td>
<td>IED Name</td>
</tr>
<tr>
<td>AP</td>
<td>Access Point name</td>
</tr>
<tr>
<td>Srv</td>
<td>Server name</td>
</tr>
<tr>
<td>LD</td>
<td>Logical Device where GCB is defined</td>
</tr>
<tr>
<td>LN</td>
<td>Logical Node where GCB is defined</td>
</tr>
<tr>
<td>GCB</td>
<td>GCB name</td>
</tr>
<tr>
<td>Application (appId)</td>
<td>Identifies purpose or application of the GOOSE message. It should be unique for an application within the Subnetwork.</td>
</tr>
<tr>
<td>Status</td>
<td>Describes how the GCB was created. This attribute is automatically generated by IET.</td>
</tr>
<tr>
<td>Attached Dataset</td>
<td>Select the name of the Dataset, as specified in the Dataset Editor.</td>
</tr>
<tr>
<td>t(min) (ms)</td>
<td>Minimum cycle time for sending messages (e.g. after a change in Dataset)</td>
</tr>
<tr>
<td>t(max) (ms)</td>
<td>Max cycle time for sending messages (e.g. used to detect missing or delayed messages)</td>
</tr>
<tr>
<td>Conf.Rev.</td>
<td>Version number will be incremented on changes of the Dataset related to the control block. Changes are the deletion or insertion of a member of the Dataset, reordering of the member, or changing the Dataset reference of the control block. Changes will increment this value by 10000.</td>
</tr>
<tr>
<td>GCB Type</td>
<td>GOOSE</td>
</tr>
</tbody>
</table>
### Property Description

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>MAC Address</td>
<td>Identifies source and allow filtering of the GOOSE message. It should be unique for an application within the Subnetwork.</td>
</tr>
<tr>
<td>APP-ID</td>
<td>Identifies source and allow filtering of the GOOSE message. It should be unique for an application within the Subnetwork.</td>
</tr>
<tr>
<td>VLAN-ID</td>
<td>Controls the dataflow and real-time behavior of the messages within a VLAN network. HEX value between 000-FFF is used, where value 000 (default) indicates that only the VLAN user priority is used.</td>
</tr>
<tr>
<td>VLAN Priority</td>
<td>Controls the dataflow and real-time behavior of the messages within a VLAN network. Priority value between 0-7 is used, where higher numbers indicate higher priority message. 4 is the recommended value for GOOSE.</td>
</tr>
</tbody>
</table>

#### 10.6.2 Create GCBs

If you know, where you want to create your GCB, you can preselect an appropriate element in the 'Substation', 'IEDs' or 'Communication' Navigation Tree. To have the full choice of IEDs, select the project.

1. Open the GCB Data editor and select a row in the GCB list, right-click and select 'Insert New Row' from the context menu.
2. Alternatively, you can select the ‘Communication’ menu tab (the ‘Communication’ Navigation tab will automatically be selected) and click on ‘Create GCB(s)’.

3. The Dialog ‘Create New GCB’ appears. The Tree in the dialog will show all IEDs below the element you selected. Navigate to the IED and then to the LLN0 where you want to create the new GCB.

   If there are several Access Points available, through which the GCB can be sent, select the one you want in the ‘Access Point’ combo box.

   Note that for ABB IEDs, the GCBs are allocated to the LD0/LLN0 logical node.

   Once you select an LN (e.g. LD0/LLN0), the tool proposes a name, which you can adapt according to your wishes. Invalid names are indicated by an error mark (Invalid names typically are either duplicate names or contain characters that are not allowed for a particular element).

4. Click ‘OK’ to create the new GCB or ‘Cancel’ to close the dialog without creating a GCB.

   If you created a new GCB, the Navigation Tree will remain where it is and the Editor will change to ‘GCB Data’. The new GCB will have a standard configuration, which you may edit, if needed.

**10.6.3 Delete GCBs**

A GCB once created cannot be renamed or totally removed anymore. The reason is that, once it is exported, the GCB and its last version number needs to be remembered. If this is not done, a GCB with the same name and the same version number, but different content and/or behavior could be created at a later time. This would lead to confusion and even potentially dangerous situations.

So a GCB is “deleted” by setting its status to ‘deleted’. Deleted GCBs will remain in the tool forever, but will not be exported in SCD/ICD/IID files, and will not count when evaluating limits for GCBs.
1. Select the GCB Editor
2. Select any element that has GCBs below it in the Navigation Tree (the IED where you want to create it, is usually a good choice).
3. Select one or several rows with the GCBs intended for deletion.
4. Right-click and select ‘Delete Row(s)’ or press the ‘Delete’ key. A confirmation dialog appears, confirm the deletion or cancel it.
   The GCBs will remain, but their Datasets will be removed and their status will go to ‘deleted’.

10.6.4 Edit GCBs

The GCB configuration can be edited in the GCB editor; normally this should not be necessary however. There are several specialties to be observed:

A GCB cannot be renamed. To rename a GCB, delete it and create a new GCB with the intended name.

Removing a Dataset from a GCB will automatically put the GCB into a ‘deleted’ state.

Configuration changes will cause its configuration revision value (Column ‘Conf.Rev’, ‘configRev’ attribute in IEC 61850) to be increased to the next multiple of 100. This will happen only once between import and export, not with every change.

Although it is not demanded by the standard, some vendors do not allow address combinations of the same MAC address with a different AppID or vice versa. It is recommended to follow an addressing scheme that always changes both together.

10.6.5 Reuse “Deleted” GCBs

A deleted GCB can be reused by simply attaching a Dataset to it. Instead of creating a new GCB, it is recommended to reuse “deleted” GCBs first.

10.7 GCB Client Configuration

10.7.1 Prerequisites

To do a sensible GCB client configuration, most of the Bay-level IEDs should be known and should be attached to the correct Subnetworks.
10.7.2 GCB Client Editor

The rows of this editor show GCBs (“Senders”). The rows depend on the element selected in the Navigation Tree.

The columns show the IEDs available as GOOSE clients (“ Receivers”). To save space the client IED names are vertical. To avoid neck strains, use the Tooltip to see the text horizontally.

If the Client IED is not on the same Subnetwork as the GCB sender, it cannot be configured as a GCB Client. In the editor, the respective cells are disabled and cannot be edited.

Also an IED cannot be configured to send GOOSE to itself; again the respective cells are disabled.

Upon adding or removing clients, the corresponding input sections are updated automatically (excluding those that have their intAddrs set; the intAddrs serve as a reference to IED Configuration Tools, e.g. to the Signal Matrix in PCM600).

10.7.3 Configure GCB Clients

In the Navigation Tree, select any element that has GCBs below it in the Navigation Tree (usually a Subnetwork, a Voltage Level or the whole project is a good choice).

1. Select the GCB Client Editor. The rows will display available GCBs and the columns will show available GCB clients.

2. If you have IEDs with more than one Access Points, make the column ‘AP’ visible.

3. Configure the GCBs with Clients:
   - Add a GCB client individually by selecting one cell and entering ‘+’ or ‘*’ or ‘x’.
• Remove a GCB client individually by selecting one cell and entering ‘-’ or pressing ‘Delete’.

4. Right-click and select ‘Remove GCB Clients in selection’ to remove all existing connections in the selected cells, rows or columns (and all related inputs).

5. Right-click and select ‘Connect GCBs to Clients in selection’. This will connect all GCBs in the selected cells, rows or columns, if allowed by the Subnetwork configuration.

6. Use the mechanisms described above in the previous chapter if you need to individually configure clients.

10.7.4 Configure filter for GCB Clients

IET600 allows the user to add a regular expression, which will filter the clients based on their name.
Chapter 10

10.8 SVCB Data Configuration

10.8.1 SVCB Data Editor

The SVCB Data editor defines the parameters for the reporting service, according to IEC 61850 part 7-2.

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>IED Type</td>
<td>IED Type</td>
</tr>
<tr>
<td>IED</td>
<td>IED Name</td>
</tr>
<tr>
<td>AP</td>
<td>Access Point name</td>
</tr>
<tr>
<td>Srv</td>
<td>Server name</td>
</tr>
<tr>
<td>LD</td>
<td>Logical Device where GCB is defined</td>
</tr>
<tr>
<td>LN</td>
<td>Logical Node where GCB is defined</td>
</tr>
<tr>
<td>SVCB</td>
<td>SVCB name</td>
</tr>
<tr>
<td>Status</td>
<td>Describes how the SVCB was created. This attribute is automatically generated by IET.</td>
</tr>
<tr>
<td>Attached Dataset</td>
<td>Select the name of the Dataset, as specified in the Dataset Editor.</td>
</tr>
<tr>
<td>Property</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>SvId</td>
<td>No default value, must be unique within the Subnetwork. If it is editable, it is recommended to make unique within substation to avoid reconfiguration when changing Subnetworks.</td>
</tr>
<tr>
<td>Multicast</td>
<td>FALSE indicates that this is a Unicast control block, a maximum of one client IED shall be assigned to the instance</td>
</tr>
<tr>
<td>Sample Mode</td>
<td>Specifies if the sample rate is defined either in 1) “SmpPerPeriod” in units of samples per nominal period, 2) “SmpPerSec” samples per second, or 3) “SmpPerSmp” seconds per sample. If missing, default is 1)</td>
</tr>
<tr>
<td>Sample Rate</td>
<td>The value shall be interpreted depending on the value of Sample Mode</td>
</tr>
<tr>
<td>Reserved</td>
<td>TRUE indicates that SVCB is currently exclusively reserved for the client that has set the value to TRUE. Other clients shall not be allowed to set any attribute of that SVCB</td>
</tr>
<tr>
<td>Conf.Rev.</td>
<td>Version number will be incremented on changes of the Dataset related to the control block. Changes are the deletion or insertion of a member of the Dataset, reordering of the member, or changing the Dataset reference of the control block, or any change of a value of an attribute of MSVCB. Changes will increment this value by 10000.</td>
</tr>
<tr>
<td>NoAsdu</td>
<td></td>
</tr>
<tr>
<td>MAC Address</td>
<td>Identifies source and allow filtering of the SV message. It should be unique for an application within the Subnetwork.</td>
</tr>
<tr>
<td>APP-ID</td>
<td>Identifies source and allow filtering of the SV message. It should be unique for an application within the Subnetwork.</td>
</tr>
<tr>
<td>VLAN-ID</td>
<td>Controls the dataflow and real-time behavior of the messages within a VLAN network. HEX value between 000-FFF is used, where value 000 (default) indicates that only the VLAN user priority is used. If not used it shall be set to 0</td>
</tr>
<tr>
<td>VLAN Priority</td>
<td>Controls the dataflow and real-time behavior of the messages within a VLAN network. Priority value between 0-7 is used, where higher numbers indicate higher priority message.</td>
</tr>
</tbody>
</table>
10.8.2 Create SVCBs

1. If you know, where you want to create your SVCB, you can preselect an appropriate element in the ‘Substation’, ‘IEDs’ or ‘Communication’ Navigation Tree. To have the full choice of IEDs, select the project.

2. Open the SVCB Data editor and select a row in the SVCB list, right-click and select ‘Insert new MSVCB’ or ‘Insert new USVCB’ from the context menu.

3. Alternatively, you can select the ‘Communication’ menu tab (the ‘Communication’ Navigation tab will automatically be selected) and click on ‘Create MSVCB(s)’ or ‘Create USVCB(s)’ respectively.

4. A new row is created in the editor. Select a Dataset to the SVCB in the ‘Attached Dataset’ column.

5. Modify the remaining properties as required.
10.8.3 Delete SVCBs

A SVCB once created cannot be renamed or totally removed entirely from the project to maintain a consistent revision history of that SVCB. Once a SVCB has been exported, its last version number needs to be remembered by the system tool. If this is not done, a SVCB with the same name and the same version number, but different content and/or behavior, could be created at a later time, which would lead to confusion and even potentially dangerous situations.

So a SVCB is “deleted” by setting its status to ‘deleted’. Deleted SVCBs will remain in the tool forever, but will not be exported in SCD/ICD/IID files, and will not count when evaluating limits for SVCBs.

1. Select the SVCB Data editor
2. Select any element that has SVCBs below it in the Navigation Tree
3. Select one or several rows with the SVCBs intended for deletion.
4. Right-click and select ‘Delete Row(s)’ or press the ‘Delete’ key. A confirmation dialog appears, confirm the deletion or cancel it.

The SVCBs will remain, but their Datasets will be removed and their status will go to ‘deleted’.

10.8.4 Edit SVCBs

The SVCB configuration can be edited in the SVCB editor; normally this should not be necessary however. There are several specialties to be observed:

An SVCB cannot be renamed. To rename an SVCB, delete it and create a new SVCB with the intended name.

Removing a Dataset from an SVCB will automatically put the SVCB into a ‘deleted’ state.

Configuration changes will cause the configRev value to be increased to the next multiple of 100. This will happen only once between import and export, not with every change.

10.8.5 Reuse “Deleted” SVCBs

A “deleted” SVCB can be reused by simply attaching a Dataset to it. Instead of creating a new SVCB, it is recommended to reuse deleted SVCBs first.
10.9 SVCB Client Configuration

10.9.1 Prerequisites
To do a sensible SVCB client configuration, most of the Bay-level IEDs should be known and should be attached to the correct Subnetworks.

10.9.2 SVCB Client Editor
The rows of the SVCB Client editor show SVCBs from the selected element in the Navigation Tree. The last columns in the editor shows the IEDs available as SVCB clients. To save space the client IED names are vertical. Use the Tooltip to see the text horizontally.

If the Client IED is not on the same Subnetwork as the SVCB sender, it cannot be configured as a Client. In the Editor, this shows as greyed-out cells that cannot be edited.

Also an IED cannot be configured to send SVs to itself, this shows as greyed-out cells too.

To send a SVCB to a receiving IED, enter ‘+’ or ‘*’ or ‘x’ in the IED column at the end of the SVCB row. You can use ‘-‘, the space bar or ‘Delete’ to remove an existing clients.

Upon adding or removing clients, the corresponding input sections are updated automatically (excluding those that have their intAddrs set; the intAddrs serve as a reference to IED Configuration Tools, e.g. to the Signal Matrix in PCM600).
10.9.3 Configure SVCB Clients

In the Navigation Tree, select any element that has SVCBs below (usually a Subnetwork, a Voltage Level or the whole project is a good choice).

1. Select the SVCB Client Editor.
2. If you have IEDs with more than one Access Points, make the column ‘AP’ visible.
3. Configure the SVCBs Clients:
   - Add an SVCB client individually by selecting one cell and entering ‘+’ or ‘*’ or ‘x’.
   - Remove an SVCB client individually by selecting one cell and entering ‘-’ or pressing ‘Delete’.
4. Right-click and select ‘Remove SVCB Clients in selection’ to remove all existing connections in the selected cells, rows or columns (and all related inputs).
5. Right-click and select ‘Connect SVCBs to Clients in selection’. This will connect all GCBs in the selected cells, rows or columns, if allowed by the Subnetwork configuration.

10.9.4 Configure filter for SVCB Clients

IET600 allows the user to add a regular expression, which will filter the clients based on their name.
10.10 ‘Inputs’ Configuration

‘Inputs’ is understood here in the sense of IEC 61850: as references to Data Attributes which an IED is receiving via Dataflow Communication.

10.10.1 Overview

There are two ways how inputs are currently used.

1. By the System Tool: if the Target IED allows, IET600 automatically creates inputs for all DATA sent via GCB or SVCB on the Client IED. (IET600 does currently not create inputs for RCB Clients).

   In this case IET600 configures all inputs in the LLN0 of the receiving server. If inputs on another LN are requested by the IED Configuration Tool, they need to be manually remapped.

2. An IED can specify inputs to indicate what information is needed. This approach is called “Late Binding” (for details how this specification is done, please read “Annex: Inputs Predefined from IED (Late Binding)”). The System engineer has then the duty to map attributes from the desired source and configure them to be sent via specified Control Block service (GCB, SVCB, RCB).

   This is yet unexplored territory and may cause considerable manual work. One engineering workflow where this can be potentially useful is between an SVCB Publisher (e.g. SAM600) and its consumer (e.g. REx670).
### Inputs Editor

The ‘Inputs’ Editor displays Inputs and allows a few things to be edited.

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Target IED</td>
<td>IED where the Input is located</td>
</tr>
<tr>
<td>AP</td>
<td>Access Point where the Input is located (default hidden)</td>
</tr>
<tr>
<td>Server</td>
<td>Server where the Input is located (default hidden)</td>
</tr>
<tr>
<td>LD</td>
<td>LD where the Input is located</td>
</tr>
<tr>
<td>LN</td>
<td>LN where the Input is located</td>
</tr>
<tr>
<td>pServT</td>
<td>predefined Communication Service Type for Late Binding (for an explanation, see “Annex: Inputs Predefined from IED (Late Binding)”</td>
</tr>
<tr>
<td>pLN</td>
<td>predefined Source LN (for an explanation, see “Annex: Inputs Predefined from IED (Late Binding)”</td>
</tr>
<tr>
<td>pDO</td>
<td>predefined Source DATA Object / Common Data (for an explanation, see “Annex: Inputs Predefined from IED (Late Binding)”</td>
</tr>
<tr>
<td>pDA</td>
<td>predefined Source DATA Attribute (for an explanation, see “Annex: Inputs Predefined from IED (Late Binding)”</td>
</tr>
<tr>
<td>Input Source</td>
<td>IEC-61850-Reference to the Source of the Input, typically a Data Attribute on another IED. IEC 61850 allows also Data Objects to be defined as Input Sources, but this is rare.</td>
</tr>
<tr>
<td>Service</td>
<td>The Dataflow Service, by which this information is received (GOOSE, SMV or Reporting)</td>
</tr>
<tr>
<td>Source Control Block</td>
<td>IEC-61805-Reference to the Control Block used to transmit this information.</td>
</tr>
<tr>
<td>Internal Address</td>
<td>Address which an IED or its configuration tool can give indicating internal address to which this information is written to. Main function of the internal address in IET600 is to indicate that the information supplied by an Input section is actually used by the target IED.</td>
</tr>
</tbody>
</table>
### Property | Description
--- | ---
 Status | ‘Resolved’ indicates that the Data Object or Data attribute referenced by ‘Input Source’ exists in the DATA model of the source IED; and it is contained in a Dataset and the Target IED is client of this Dataset.  

‘Unresolved Configurable’ indicates that either the Data Object or Data attribute referenced by ‘Input Source’ does not exist in the DATA model of the source IED, or that it is not contained in a Dataset or that the Target IED is not client of a Dataset with this attribute. The Input does not have an Internal Address which allows it to be reconfigured or removed.  

‘Unresolved Fixed’ indicates an Input with an unresolved reference (as described above), and which contains an Internal Address, which prevents it from being reconfigured or removed. If the Input Source of such an Input is empty, this Input might require Manual Mapping, see chapter 10.10.6 below)  

Description | A user-editable attribute where an engineer could provide some additional information about this attribute

---

### 10.10.3 Create Inputs

Inputs are created automatically on the client IED, if GCB or SVCB Clients are configured unless the client IED specifically forbids this (Service ClientServices.nolctBinding = true, introduced in Ed. 2).

### 10.10.4 Delete Inputs

Inputs with Status ‘Resolved’ (resulting from a GCB or SVCB Client mapping) cannot be deleted manually, remove the associated GCB or SVCB client mapping to delete those inputs.

Inputs with Status ‘Unresolved configurable’ can be deleted from the row context menu.

Inputs with Status ‘Unresolved fixed’ should normally not be deleted:
- an Internal Address indicates that this Input is actually required by the IED for its correct functioning.
- Strictly according to IEC 61850, a System Configuration Tool is not allowed to delete Inputs with an Internal Address. However, sometimes, a deletion of such Inputs may be required to clean up an old configuration.

To delete ‘Unresolved configurable’ or ‘Unresolved fixed’ inputs:

1. Select the ‘Inputs’ Editor.
2. Select any element that has IEDs below it in the Navigation Tree (the IED with the Inputs in question is usually a good choice).

3. Select one or several rows with the ‘Inputs’ intended for deletion (in IEDs with lots of Inputs, you may want to set the filter to either an appropriate Status or to show only rows with errors).

4. Right-click and select ‘Delete Row(s)’ or press the ‘Delete’ key.

If any of the rows to be deleted contains an Internal Address (i.e. Status is ‘Unresolved fixed’), the following confirmation dialog appears:

If you are sure about the deletion (confirmation from the engineer responsible for the IED configuration is strongly recommended), continue with ‘Yes’.

5. The selected Inputs will be deleted.

### 10.10.5 Remapping Inputs to Different Parent LNs

Sometimes, a receiver IED requires an Input mapped to a particular LN different from the ABB standardized mapping to LLN0.

For Ed. 2 IEDs, this can be done by using the Late Binding Parameters as described in chapter 10.10.6 below. For older IEDs, this information will have to be provided by the engineer configuring the Target IED. Both procedures are cumbersome and should be avoided as much as possible.

To map Inputs to a different LN:
1. Select the ‘Inputs’ Editor.

2. Select one or several Inputs which you want to map to a different Parent LN.

3. Right-click in the row selector column of the editor -> the row context menu of the editor appears.

4. Select ‘Change Parent LN’ -> the following Dialog appears:

5. Select the LN which you want as parent for the Input, click ‘OK’ to remap all the selected Inputs to this LN.

While IET600 does its best to conserve Input configurations, any action deleting the Inputs will also remove any customized mapping of Parent LNs. As there can be considerable work behind such customized mapping, ensure that you create Back-ups of such configurations.

Any Inputs containing an Internal Address will not be deleted and are therefore safe against such accidental deletions. Therefore it is best to:

- do not do such Input re-mapping unless you absolutely have to.
• if you have an IED which absolutely needs it, require the IED Configuration Tool to complete the turnaround by supplying you an IID file with Inputs completed with Internal Addresses.

10.10.6 Configure Predefined Inputs (Late Binding)

For an explanation how predefined Inputs are intended to be used, please read “Annex: Inputs Predefined from IED (Late Binding)”. This chapter just describes the practical steps to configure predefined Inputs.

If an IED contains some predefined Inputs, they will appear in the ‘Inputs’ editor with empty ‘Input Source’ cell. Accordingly, a warning is shown.

Per default, all the columns that show the (predefined) Attributes are hidden, you might want to show them via the ‘Field Chooser’ Dialog (click on top left icon to open it).

10.10.6.1 Configure Mapping to Predefined DATA Attributes (DAs)

To provide the mapping to a predefined DATA Attribute, do the following steps:
1. Open the ‘Map IEC61850 DATA’ dialog by double-clicking into an ‘Input Source’ Cell:

![Image of the 'Map IEC61850 DATA' dialog]

In the bottom, the Dialog will show
- the predefined items (Service Type, LN Class, DO and DA)
- the selected DATA attribute.
- whether and in which Control Block the DATA Object or DATA Attribute is available (if this remains empty, you may still select an item; however, the ‘Inputs’ Editor will show this as a warning to remind you to add this at a later time)

The dialog will allow any choice of IEDs and LDs, as they cannot be predefined.
The dialog will show only DATA Objects and DATA Attributes matching the predefined specifications.

2. Select the appropriate attribute and click ‘Update’

![Image of selecting attribute and updating]

The ‘Map IEC61850 DATA’ dialog will close and the selected attribute will be added to the ‘Input Source’ field in the ‘Inputs’ Editor

10.10.6.2. Configure Mapping to Predefined DATA Objects (DOs)

To provide the mapping to a predefined DO, do the following steps:
1. Open the ‘Map IEC61850 DATA’ dialog by double-clicking into an 'Input Source' Cell:

In the bottom, the Dialog will show

- the predefined items (Service Type, LN Class and DO)
- the selected DATA object or attribute.
- whether and in which Control Block the DATA Object or DATA Attribute is available (if this remains empty, you may still select an item; however, the 'Inputs' Editor will show this as a warning to remind you to add this at a later time)

The dialog will allow any choice of IEDs and LDs, as they cannot be predefined.

The dialog will show only DATA Objects and DATA Attributes matching the predefined specifications.

(Please remember that DATA Objects with another name but with the same Common Data Class are also eligible and will therefore appear in the list. However, preference should always be given to DOs with a matching name).

2. IET600 does not allow you to map DOs directly. Instead of selecting a DATA Object, you can must create additional Inputs for some or all attributes of the predefined DATA Object: Select one appropriate DATA Attribute and click 'Add New Input'; repeat till you have mapped all attributes you wish.

With "Update", you may add an attribute to the current Input; however, it is recommended to leave this in its original configuration, till both sides agree which attributes to use.
You may use the FC combo box to sensibly limit the amount of Data Points to be selected; “ST” and “MX” are normally sensible choices:

It is also sensible to limit the amount of DATA Attributes configured as much as possible (in discussion with the engineer who has configured the predefined inputs).

Please take note that the same intAddr will be copied to all DATA Attributes. This behavior is mandated by IEC 61850; however it may or may not be correct for the IED Configuration Tool. It is up to the IED Configuration Tool to change those addresses appropriately.

10.11 Dataflow Consistency Check

The Dataflow Consistency Check provides a check for problems in the Dataflow configuration in an SCD file. The check is always performed on the whole project; it is automatically called before exporting an SCD file, but can also be called manually from the ‘Communication’ tab in the Main menu.

To check the consistency manually:

1. Select the ‘Communication’ menu tab (the ‘Communication’ Navigation tab will automatically be selected) and select ‘Check Dataflow Consistency’.

2. A dialog will appear, indicating a summary of found problems, e.g.:

   Details about the issues can be found in the Log.
The consistency check groups inconsistencies into errors and warnings:

- **Errors** will cause invalid SCD files or immediate problems when downloading such a configuration to the IEDs.
- **Warnings** may be allowed during transitional engineering phases, they may or may not cause problems when downloading such a configuration to the IEDs, but must be resolved before finally commissioning the system.

As the Dataflow consistency check concerns different items in different editors, the problematic items are not marked in the editors, but are logged.

The following issues are checked:

- Duplicate IP addresses (error).
- Datasets without any Dataset Entries (error, these will be removed on export).
- Datasets allocated to more than one Control Blocks of different types (e.g. RCB and GCB) (error)
- Connected Access Points not connected to a Subnetwork (warning).
- Control Blocks without any clients (warning).
- Datasets not allocated to any Control Block (warning).
- Datasets allocated to more than one Control Blocks of different types (e.g. RCB and GCB).
- IEDs which are mapped as GCB or SVCB clients, but do not have the corresponding inputs configured (warning).
- Missing MAC Addresses or APP-IDs for GCBs (warning).
- Missing MAC Addresses or SmvIDs for SVCBs (warning).
- Duplicate SmvIDs for SVCBs (the IEC 61850 standard enforces uniqueness only in Subnetworks, IET600 checks for uniqueness in the whole project; this may sometimes cause false positives) (warning).
- GCBs or SVCBs with the same MAC address but different APP-ID/SmvIDs or vice versa (this is basically allowed by the IEC 61850 standard, but IEDs of some vendors have a problem with it) (warning).

Note that IET600 will still allow you to export an invalid SCD file in spite of errors or warnings.

## 10.12 Dataflow Communication Signal List

IET600 can generate Excel document, which contains the configuration of the dataflow communication among RCBs, GCBs and SVCBs and their clients. The datasets are also included with their entries.
11 Customer Name Engineering

11.1 Overview

For a general explanation how “Internal Name” and “Customer Name” are intended to be used, please read chapter 2.2 (Naming of Objects). This chapter describes the functionality of the ‘Naming’ Editor.

In IET600, the following elements can have customer names:

- Substations, Voltage Levels, Bays, Functions and Subfunctions
- Equipment and Sub-Equipment
- IEDs
- Subnetworks
- Logical Devices (IEC 61850 Ed.2 IEDs may allow modifying the LD Name attribute. IET600 treats this attribute as the customer name of an LD). As this has some additional implications, see chapter 11.3, LD Naming for a detailed explanation)
11.2 The 'Naming' Editor

The central components of this editor are the two columns 'Internal Name' and 'Customer Name'. The remaining columns are mainly for information; additionally, by clever sorting and/or filtering, they support rapid (re)naming of many items.

11.2.1 Rename IED, Internal Name

Renaming the internal name of an IED will cause the updating of several related items (e.g. GCB default Application or appID, RCB default rptID, Input source references in other IEDs etc.)

Changing an IED internal name will affect the communication and require a reloading of all clients of the affected IED. Once a project is in the testing/commissioning phase, this is therefore not recommended.
11.2.2 Rename Bay, Internal Name

If you rename a Bay name and the Bay contains IEDs with the Bay name as a part of their name (e.g. IED A1D1Q01KF1 in Bay Q01), IET offers you the possibility to rename all IEDs in that Bay accordingly. The following Dialog will appear:

It is strongly recommended to rename the IEDs along with the Bay. However, communication items will be affected (see chapter 11.2.1 above).

If you do not follow a hierarchical naming scheme, the dialog will not appear and the Bay will be renamed without further confirmation.

11.2.3 Rename Substation or Voltage Level, Internal Name

If you rename a Substation or Voltage Level and it contains IEDs with the Substation or Voltage Level name as a part of their name (e.g. IED A1D1Q01KF1 in Substation A1), IET offers you the possibility to rename all IEDs in that Voltage Level accordingly.

As when renaming Bays, a dialog will appear and you can confirm or refuse to rename the IEDs along with the Substation or Voltage Level:

It is strongly recommended to rename the IEDs along with the Bay; however, communication items will be affected (see chapter 11.2.1 above).

As a great number of IEDs and all their related communication items need to be renamed (especially when renaming a Substation with all IEDs contained in it), this may take several minutes in a big Substation.

If following the IEC 81346 naming scheme, Communication Subnetworks will also contain the Substation Name. However, they are not automatically renamed together with the Substation. To rename these Communication Subnetworks efficiently, sort on ‘Type’ and ‘Internal Name’ columns, then use ‘Search and Replace’.
11.2.4 Hierarchical Customer Names

A typical example is a customer which was using the older variant IEC 61346 for naming, which is also hierarchical. In this case, the naming structure is the same, only the standard names are different (e.g. the Circuit Breaker of a Bay is named Q0 instead of QA1, Disconnector Q1 instead of QB1 etc.)

In such cases, Customer names can be given (or renamed) very efficiently by sorting on ‘Type’ and ‘Internal Name’ columns and then using either ‘Search and Replace’ or ‘Fill Down’ to give a defined Customer Name for all equipment with a specific Internal Name.

11.2.5 Non-hierarchical Customer Names

Certain customers have specifications for their names that are not hierarchical (a typical example is the Power System Designation System, used in Power Generation Plants).

The “Customer Path/Name” in such cases is usually not needed (the hierarchy is implicitly contained in the naming); therefore the engineer may want to hide the respective column to avoid confusion.

11.2.6 Customer Path/Name Display

If an external name for any item is missing, its internal name will be shown in brackets, e.g. Sils.220 kV.(Q01). This is done to provide sensible paths for equipment, even if higher levels are not named yet (e.g. Sils.220 kV.(Q01).CT1 instead of Sils.220 kV.CT1 which would not allow identifying the CT in question properly).

11.3 LD Naming

IEC 61850 Section 8-1 – defining the online communication – always refers to an LD Name as part of a reference of any IEC61850 DATA object.

As a default value, this LD Name is a concatenation of IED name and Logical Device Name (LDInst in the SCL file). As an example, a Logical Device ‘LD0’ in an IED ‘AA1C1Q01KF1’ would have the LD Name ‘AA1C1Q01KF1LD0’

In some situations, it was considered useful to allow this LD Name to be customized. So in the above example, a customer might want to rename this into ‘Main1_Control’, i.e. give it a name which is more appropriate in his own environment. So IEC 61850 Ed. 2 introduced the appropriate SCL element and IEC 61850 Ed. 2.1 introduces the appropriate services that allow an IED to declare whether it can handle LD renaming or not.

LD Naming has two implications:

1. An IED must allow the customization of this LD Name.
2. An IED which is a communication client of an IED with customized LD Names must understand it.

### 11.3.1 Customization of Logical Device Names

As currently, not many IEDs are allowing such a customization, per default, the corresponding rows are hidden in the ‘Naming’ Editor. To make them visible, check the Menu item ‘Show LD’ in the ‘Column’ Context menu of the ‘Naming editor’:

![Diagram of Naming Editor]

You will need to do this every time you open IET600.

The Logical Devices will then also appear as rows. If an IED does not allow its LD Names to be customized, they will appear greyed out in the editor (as in the example above). To allow it, the IED must have the Service ‘ConfLdName’ defined.

### 11.3.2 Configuring Control Block Clients

If a Server IED has actually changed LD Names, only IEDs that have ClientServices.supportsLdName defined and set to “true” can be configured as Clients for Control Blocks of the Server IED.

However, if the Server IED does allow a customization, but does not have any customized names, an IED can be configured as Client even though it may not be able to handle customized LD Names.
Chapter 12

12 HMI Engineering

12.1 Introduction

HMI data are basically organized as “signals”; a signal can be understood as what appears on one line in an Event List or Alarm List.

12.1.1 User Interface

To do HMI engineering, you will use the following editors:

- ‘Signal Clients’: to configure signals to different HMIs (direct or via Mirroring) and NCCs.
- ‘HMI’ Data to configure attributes of signals.
- ‘NCC Offset’ and ‘NCC Data’ to configure attributes for sending data to NCCs.

If these Editors are not visible as Tabs, read chapter 3.1.3 (Editors), the last section explains how to make them visible.

12.1.2 Engineering Workflow

There are several ways to engineer HMI Signals:

- by importing or copying Bays or IEDs (from a Library, another project etc.). HMI signals will be imported/copied along with the Bay/IED. Inheritance mechanisms will apply as usual.
- by importing a MicroSCADA ‘lof’ file or a MicroSCADA ‘csv’ export of Process Objects.
- by importing individual preconfigured signals from a Library
- by inserting HMI Signals from user-defined Signal Templates
- by copying signals from another project.
- by creating individual signals and configuring them.

Efficient Substation engineering will be based on several or all possibilities mentioned above.

HMI signals have some attributes configured via formula; the evaluation of these formula depends on a LN Mapping. IET600 allows you to do LN Mapping even if there is no IED DATA configuration yet which has distinct advantages:
• Signals can be configured (and exchanged with the HMI configuration tool), before actually knowing the DATA structure of an IED.
• IED DATA can be deleted or reconfigured without losing the respective HMI information.
• Station-level engineering and bay-level engineering can therefore be done in parallel and merged at a relatively late engineering stage.

To allow this, IET600 treats the attribute ‘Mapped To’ in the HMI Data editor as a “virtual” reference to an LN. This virtual reference will then be resolved when the IED DATA model is imported.

The status of an HMI signal reference can be seen in the column ‘Mapping Resolved’ in the HMI Data editor.

It can also be seen in the SLD editor in the LN Mapping table under the column ‘Status’. The status can have the following values:
• “resolved” for mappings which reference an actual IED DATA model (irrespective of any signals)
• “virtual” for mappings which have a reference to an HMI signal, but not to an actual IED DATA model
• “unresolved” for LN references which has not been mapped to an IED nor to an HMI signal. This situation may occur if a signal and its corresponding LN in the IED model have been deleted; another possibility is via Import of an SSD file with LN Mappings.

12.2 HMI Creation and Configuration

12.2.1 Overview

IET600 allows engineering multiple HMI Clients and configuring different signals or different signal configurations for each HMI.

IET600 allows also the configuration of Hot-Standby systems. They consist of a Hot-Standby element with two related HMIs. The signals of all HMIs participating in a Hot-Standby system are identical, while the communication configuration of each HMI is different.

There are 3 types of HMI IEDs
• HMIs (a physical IED).
• Hot-Standbys (a cluster of several HMIs that are redundant and have identical HMI Signals, but different communication configurations).

OPC servers are modeled as IEDs only for historical reasons, in reality they are just pieces of software that run in the HMI. They are displayed in the HMI tree as children of their HMI, and still appear as IEDs in the IED tree.

As multiple HMIs are possible, it is important to be aware how the HMI signals are presented:

• A Bay-Level IED in the IED tree may send different signals to different HMIs. The IED has a collection of signals for each HMI it is sending signals to. When selecting a Bay-Level IED in the IED tree; the user needs to choose one HMI to display the signals that this particular Bay-Level IED is sending to this particular HMI. So in this case, it is a server’s or sender’s view (which process signals are sent from this IED).

• When selecting an HMI in the IED or HMI Navigation tree, you see all the process signals that belong to this particular HMI. This is a client’s or receivers view (which process signals are received by this particular HMI).

The following icons are used in the HMI navigation tree to represent the various HMI and Gateway configurations:

<table>
<thead>
<tr>
<th>Icon</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>![Icon]</td>
<td>Stand-alone HMI</td>
</tr>
<tr>
<td>![Icon]</td>
<td>Stand-alone HMI, linked</td>
</tr>
<tr>
<td>![Icon]</td>
<td>Hot-Standby HMI</td>
</tr>
</tbody>
</table>
The next chapters will describe how to configure the HMI clients in IET600. Gateways and NCC configuration will be described in subsequent chapters.

12.2.2 Preliminary: Configure Project OI

If you work with MicroSCADA and use OI Settings different from the default, adapt the OI Settings before creating any MicroSCADA HMIs. For details, see chapter 23 (Annex: MicroSCADA OI Settings).

12.2.3 Create an HMI

An HMI can be created from the ‘HMI’ tab in the Main menu.
It can also be created from the context menu of several Navigation Tree objects:

The details are described in chapter 7.2.5 (Create New HMI). The configuration of OPC servers running on HMIs of defined types (e.g. MicroSCADA) is also described there.

Although not mandatory, it is recommended to add some HMI parameters now, e.g. configure the MicroSCADA OI length in the ‘Properties’ Window.

### 12.2.4 Create a Hot-Standby (HSB) HMI System

A Hot-Standby system can be created from the ‘HMI’ tab in the Main menu.

An HMI can be created from the HMI tab in the Main menu.

It can also be created from the context menu of the ‘HMIs’ Navigation Tree:

1. Select the ‘HMI’ tab in the Navigation Panel, right-click the ‘HMIs’ element -> the context menu for that element appears.
2. Select ‘Create new HSB …’

3. The Dialog ‘Create New Hot Stand By’ appears:

4. IET600 proposes a name, which you can adapt according to your wishes. Invalid names are indicated by an error mark.

5. Click ‘OK’ to create the new HSB or ‘Cancel’ to close the dialog without creating an IED.

6. Drag’n’drop two existing HMIs on the newly created Hot-Standby element.
   If the HMIs were created earlier and already contain HMI signals, the HMI signals of the first HMI will be used for the HSB, the signals of the subsequent HMIs will be thrown away).
Chapter 12

12.3 Signal Creation

12.3.1 Overview

HMI signals are created and configured in the HMI Data editor, as described in chapter 12.4.2: HMI Data Editor. There are several ways to create HMI signals:

- Copy HMI signals from other IEDs or other HMIs
- Insert HMI Signals from user-defined Signal Templates
- Import a MicroSCADA Loadfile (‘lof’ file) or ‘csv’ file
- Create HMI signals manually

The decision which method to use to create the HMI signals depend on the project. For example, if one is starting a new project with mostly ABB devices, importing from the Library is likely the best method. On the other hand, if similar projects or IEDs are available, then reusing the previously engineered data by copying the HMI signals from another IED or HMI could be faster.

12.3.2 Copy HMI Signals from another IED

This is basically done as a simple copy (the selected signals from one IED) and paste (these signals to another IED).

2. Select the IED whose signals you want to copy.
3. Select all the rows with the signals you want to copy (from the leftmost "row selector" column of the editor).

4. Right-click in the row selector column of the editor -> the row context menu of the editor appears.

5. Select ‘Copy row(s)’ -> all the selected rows will be copied.

6. Select the another IED (destination for signals) in the Navigation Tree

7. Right-click in the row header of the 'HMI' editor -> the row context menu appears.

8. Select ‘Insert copied rows’

The signals are inserted to this IED.

Note that when copying and pasting a complete IED (or any element containing IEDs, e.g. a Bay), its HMI signals are copied along.

12.3.3 Insert HMI Signals from Signal Templates

Please consult chapter 12.9 (Working With User Defined Signal Templates) to learn how to prepare and use Signal Templates.

12.3.4 Import a MicroSCADA Database File

A MicroSCADA database file can be in different formats: ‘lof’ file or ‘csv’ file

To engineer a new project from scratch, it is best to work with HMI signals from a similar project or from the Library.

For retrofits or upgrades of old HMIs however, signals from the existing HMI have to be used. An import of such signals is currently only possible for MicroSCADA.
Chapter 12

- for MicroSCADA versions 8.x, it is recommended to export a ‘lof’ file from MicroSCADA and use this for import into IET600.
- for MicroSCADA versions 9.x and newer, it is recommended to use a ‘csv’ export from the Database (described below).

Only if neither is feasible, use a ‘lof’-file export from an IET 4.x project. Be aware, though, that this may not represent the actual state of the MicroSCADA process database, as customers may have made modifications directly in MicroSCADA.

A database file import takes advantage of existing IN attributes to map imported HMI signals to current IED data. To take full advantage of this mapping mechanism, the Substation and bay-level IED data should already exist (e.g. from an SCD import) and LN mapping should be done before a database file import.

For detailed steps of the import, see chapter 16.6.5 (Import a MicroSCADA ‘lof’ File).

12.3.5 Create HMI Signals manually

HMI / MicroSCADA signals can also be created manually in IET600.

1. Select an IED in any of the Navigation Trees (except the ‘HMI’ Navigation Tree) and open the ‘HMI Data’ editor.
2. Right-click in the row selector column of the editor -> the row context menu of the editor appears.
4. A new row is created with default HMI / MicroSCADA properties.
12.4 Signal Configuration

12.4.1 Overview

An HMI signal has two parts: a generic part that is useful for any HMI and a MicroSCADA-specific part that is useful for MicroSCADA engineering only. Currently, an export is possible only for MicroSCADA data (see chapter 16.6, HMI Signal Information); in future, an export of generic HMI data for configuration of other HMIs like 800xA or RTU560 is planned.

In the HMI Signals Editor, you can select whether you want to see the generic part or the MicroSCADA part or both by selecting the checkboxes shown above the grid.

The HMI data engineering consists of configuring the individual HMI (MicroSCADA) signal properties. These properties can be set in the HMI Data editor.

Moreover, HMI signals are attached to IEDs in the project. This is achieved via a mapping to the IEC 61850 model of the IED, as described in chapter 12.4.5: Mapping Signals to IEC 61850.

12.4.2 HMI Data Editor

The HMI Data Editor provides the signal specification or MicroSCADA properties of each signal, in particular:

- Generic HMI signals – a generic description of the signal and its behavior
- MicroSCADA signals – MicroSCADA Process Object attributes
- IEC 61850 Mapping – mapping of signals to the IEC 61850 model of an IED
• HMI Rules – rule-based configuration for several HMI attributes
• Signal and status text in multiple languages

To view all available columns in the HMI Data editor, click on the Column Chooser icon to view all available columns.

12.4.3 Generic HMI Signals

To view the generic HMI signals in the HMI Data editor, select ‘Show generic part’ in the header. Most of these generic HMI signals correlate to specific MicroSCADA attributes, as shown in the table below:

<table>
<thead>
<tr>
<th>Generic Signal Attributes</th>
<th>Equivalent MicroSCADA Attribute</th>
</tr>
</thead>
<tbody>
<tr>
<td>In Use</td>
<td>IU</td>
</tr>
<tr>
<td>Station Text</td>
<td>n/a*</td>
</tr>
<tr>
<td>Signal Text (Internal)</td>
<td>OX</td>
</tr>
<tr>
<td>Status Text (Internal)</td>
<td>(Derived from EH)</td>
</tr>
<tr>
<td>Internal Signal</td>
<td>n/a*</td>
</tr>
<tr>
<td>Event List</td>
<td>HE</td>
</tr>
<tr>
<td>Event List Update Condition</td>
<td>HA</td>
</tr>
<tr>
<td>Alarm List</td>
<td>AL</td>
</tr>
<tr>
<td>Alarm Class</td>
<td>AC</td>
</tr>
<tr>
<td>Alarm Delay</td>
<td>AD</td>
</tr>
<tr>
<td>Printer</td>
<td>n/a*</td>
</tr>
<tr>
<td>Printer Update Condition</td>
<td>PA</td>
</tr>
<tr>
<td>Comment</td>
<td>n/a*</td>
</tr>
</tbody>
</table>

Signals marked with n/a* indicate that these generic signal attributes do not have a direct relation to a MicroSCADA attribute. They are however useful for identifying the signal within the project. For example, ‘Comment’ can be used to store some additional information for a signal.
12.4.4 **HMI Text Rules**

IET600 provides flexible rule-based configuration of the following HMI properties:

- **LNRule**: rule-based configuration of MicroSCADA LN attribute
- **OIRule**: rule-based configuration of MicroSCADA OI attribute
- **INRule**: rule-based configuration of MicroSCADA IN attribute
- **ONRule**: rule-based configuration of MicroSCADA ON attribute
- **Station Text Rule**: rule-based configuration of generic signal
- **Signal Text (Customer)**: rule-based configuration of MicroSCADA OX attribute

You can use the following variables to assemble a station text:

(for a detailed explanation of Function, Subfunction etc., please consult chapter 6.1, IEC 61850 Modelling of a Substation.)

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SS</td>
<td>Substation internal name</td>
</tr>
<tr>
<td>SS_C</td>
<td>Substation customer name</td>
</tr>
<tr>
<td>VL</td>
<td>Voltage Level internal name</td>
</tr>
<tr>
<td>VL_C</td>
<td>Voltage Level customer name</td>
</tr>
<tr>
<td>BAY</td>
<td>Bay internal name</td>
</tr>
<tr>
<td>BAY_C</td>
<td>Bay customer name</td>
</tr>
<tr>
<td>FN</td>
<td>Function internal name</td>
</tr>
<tr>
<td>FN_C</td>
<td>Function customer name</td>
</tr>
<tr>
<td>SFN</td>
<td>Subfunction internal name. In case of recursive Subfunctions, this variable includes the name of all Subfunctions, not of only one level.</td>
</tr>
<tr>
<td>SFN_x</td>
<td>Subfunction internal name. This variable includes the name of the Subfunction of only one level x.</td>
</tr>
<tr>
<td>SFN_C</td>
<td>Subfunction customer name. In case of recursive Subfunctions, this variable includes the customer name of all Subfunctions, not of only one level.</td>
</tr>
<tr>
<td>SFN_C_x</td>
<td>Subfunction customer name. This variable includes the name of the Subfunction of only one level x.</td>
</tr>
<tr>
<td>EQ</td>
<td>Equipment internal name</td>
</tr>
<tr>
<td>Variable</td>
<td>Description</td>
</tr>
<tr>
<td>-------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>EQ_C</td>
<td>Equipment customer name</td>
</tr>
<tr>
<td>SEQ</td>
<td>Sub-Equipment internal name</td>
</tr>
<tr>
<td>SEQ_C</td>
<td>Sub-Equipment customer name</td>
</tr>
<tr>
<td>EQFN</td>
<td>Equipment Function internal name</td>
</tr>
<tr>
<td>EQFN_C</td>
<td>Equipment Function customer name</td>
</tr>
<tr>
<td>EQSFN</td>
<td>Equipment Subfunction internal name. In case of recursive Equipment Subfuns, this variable includes the name of all Equipment Subfunctions, not of only one level.</td>
</tr>
<tr>
<td>EQSFN_x</td>
<td>Equipment Subfunction internal name. This variable includes the name of the Subfunction of only one level x.</td>
</tr>
<tr>
<td>EQSFN_C</td>
<td>Equipment Subfunction customer name. In case of recursive Equipment Subfuns, this variable includes the customer name of all Equipment Subfunctions, not of only one level.</td>
</tr>
<tr>
<td>EQSFN_x_C</td>
<td>Equipment Subfunction customer name. This variable includes the name of the Subfunction of only one level x.</td>
</tr>
<tr>
<td>SN</td>
<td>Subnetwork internal name</td>
</tr>
<tr>
<td>AP</td>
<td>Access Point internal name</td>
</tr>
<tr>
<td>IED</td>
<td>IED internal name</td>
</tr>
<tr>
<td>IED_C</td>
<td>IED customer name</td>
</tr>
<tr>
<td>IED_S</td>
<td>IED internal short name (the non-hierarchical part of a hierarchical IED name, e.g. KF1 in case of AA1C1Q01KF1)</td>
</tr>
<tr>
<td>IED_S_C</td>
<td>IED customer short name (the non-hierarchical part of a hierarchical IED customer name) (rarely useful)</td>
</tr>
<tr>
<td>LDInst</td>
<td>IEC 61850 Logical Device instance</td>
</tr>
<tr>
<td>LN</td>
<td>IEC 61850 Logical Node name</td>
</tr>
<tr>
<td>DOI</td>
<td>IEC 61850 Data object name</td>
</tr>
<tr>
<td>DARef</td>
<td>IEC 61850 Data attribute reference</td>
</tr>
</tbody>
</table>
Chapter 12

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DARef_IN</td>
<td>IEC 61850 Data attribute reference, stripped by “.f” and “.i” (no distinction between float and integer values), to be used only for INRule</td>
</tr>
<tr>
<td>DARef_ON</td>
<td>IEC 61850 Data attribute reference, stripped by “.f” and “.i”, and “.” replaced by “\”), to be used only for ONRule</td>
</tr>
</tbody>
</table>

In IET600 is possible to define custom variables using the HMI Text Rule Variable Editor, which can be activated from the HMI ribbon.

The above variables can be constructed into an expression which defines the text rule, with the help of the following syntaxes:

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{expression}</td>
<td>Contains expression</td>
</tr>
<tr>
<td>@variable@</td>
<td>Contains variable</td>
</tr>
<tr>
<td>variable!</td>
<td>Must include variable (mandatory)</td>
</tr>
<tr>
<td></td>
<td>Field Separator in OI Rules</td>
</tr>
</tbody>
</table>

For example, for the station shown below, the rule evaluates as follows:
You may also use normal text strings in the above expressions. For example, if a string "Demo" is added to the end of the LNRule:

`LNRule : {@SS}@{@VL}@{@BAY}@{@FN}@{@SFN}@{@EQ}@Demo
LN: AA1C1Q01BI2Demo`

Be aware whether you include separators in the brackets or not, the evaluation is different:

`{.@EQ_C}@` The dot will be the prefix of variable IED, e.g. ".REC670". If the variable does not exist, the dot will not be visible.

`{@BAY_C}.{@EQ_C}@` The dot will always be shown, even if one or both variables don’t exist.

If IET600 cannot resolve a mandatory rule (using “!” syntax), cell and row will be marked with an error. These errors must be resolved by the user, either by satisfying the rule or by changing it.

A rule can be set back to the default value from the context menu item ‘Set to Default rule(s)’. This context menu is available from cell, row and column context menu, so you can easily set reset either a single cell, all rules of a signal (row) or all rules of a certain type (e.g. Station Text Rule).
12.4.5 Mapping Signals to IEC 61850 Attributes

The ‘Mapped To’ property for each HMI signal provides the link or association to the IEC 61850 model of an IED. When an LN has been mapped to the primary equipment as described in chapter 7.5, “Mapping LNs to Substation Elements (‘LN Mapping’)” – IET600 can automatically assign several HMI and MicroSCADA properties according to this mapping. Of course, to map the signal, you must have the IEC 61850 DATA model of the IED available.

Click on the ‘Mapped To’ cell to open the ‘Map IEC 61850 Data Attribute’ dialog:

The ‘Map IEC 61850 Data Attribute’ dialog shows the IEC 61850 model of the IED and allows you to select which attribute to allocate to the signal.

Please note:

- If the dialog does not appear and the cell just allows you to enter text, the IED does not have an IEC 61850 DATA model.
- In rare cases (e.g. Switches created in IET600), an IED may have an IEC 61850 model, but will still communicate via a different protocol. In this case, the dialog will open, but ‘OK’ will be disabled. In this case the ‘Application’ column for this signal will show “SNMP” rather than “IEC 61850”.

12.4.6 Mapping Signals to SNMP OIDs

(OID mappings are only useful if you use a recent SNMP-OPC-Gateway on a MicroSCADA system. Older configurations which work with a 3rd party SNMP-OPC-Gateway will not profit from this mapping).

The SNMP equivalent of a Signal Address is the SNMP OID (Object Identifier). Essentially, it is a sequence of numbers, separated by dots. While few OIDs are standardized, most communication equipment uses a private model. The equipment
vendor will normally provide those addresses. For some more information see https://en.wikipedia.org/wiki/Object_identifier.

For SNMP Signals, the ‘Mapped To’ must contain the OID (or optionally some static text); it cannot be configured via ‘Mapped To’ dialog, but must be entered manually from a vendor-specified list of OIDs.

**12.4.7 HMI Data Consistency Check**

The HMI Data Consistency Check provides a check for problems in the configuration file exported to MicroSCADA. It is automatically called before exporting such a file, but can also be called manually from the ‘HMI Data’ editor.

The checks performed depend on the element selected in the Navigation Tree:

- If an element in any but the ‘HMIs’ Navigation Tree is selected, checks will be performed for all related IEDs on a ‘per-IED’ base. So if this check shows no errors, it means, that an IED has no consistency errors such as duplicate LN/IX in itself.
- If in the ‘HMIs’ Navigation Tree, a single HMI or Gateway is selected, the check is performed on that HMI for all signals that it is receiving (which is equivalent to the check performed before a ‘lof’ file export). This is more reliable as e.g. two IEDs can be internally consistent, but if they have the same UN, there can still be duplicate UN/OA/OB combinations.

Early in the project, you will probably perform consistency checks per IED or per Bay. Once these are ok, you will probably check the station-level IEDs, such as switches, HMIs and Gateways (with their respective supervision signals). Finally you will perform a final check on each HMI before exporting its ‘lof’ file.

To check the consistency:

1. Select an element in any Navigation Tree, taking the above criteria into account.
2. Select the HMI Data editor and click on ‘Check Consistency’ button.
Chapter 12

The consistency check groups inconsistencies into errors, warnings and infos:

- Errors will cause an error when importing the ‘lof’ file into MicroSCADA, they need to be corrected before exporting a ‘lof’ file.
- Warnings are caused e.g. by incomplete translation of texts, they may be allowed during transitional engineering phases and will not cause import errors, but must be resolved before finally commissioning the system.

Rows and cells being the cause of a warning will be marked in the 'HMI Data' editor.

The following issues are checked:

- Duplicate LN: IX combinations or missing LN or IX values (error).
- Duplicate UN:OA:OB or missing UN, OA or OB values (error).
- Invalid OB values (error).
- duplicate ON values (error)
- invalid AN values (error)
- invalid DX attributes (warning)
- missing Status texts in internal and customer language (warning)
- aspect references which are not contained in a Dataset (info)
- concatenated AN values (info)
- LNs in the IEC-61850 DATA model that have no signals attached (info)

Please note that IET600 will still allow you to export a MicroSCADA ‘lof’ file in spite of errors or warnings.
12.5 Signal Client Configuration

12.5.1 Overview

Once HMI signals have been created for a particular IED, they can now be “mapped” to the available HMI clients. Mapping to the HMI clients allows IET600 to generate the corresponding configuration files for that HMI, for example ‘lof’ files for MicroSCADA.

12.5.2 The 'Signal Clients' Editor

The 'Signal Clients' Editor allows the mapping of the available signals in the project to a specific station-level client, which could be an HMI or Gateway.

This editor is also used for mapping signals and groups to an NCC, as described in chapter 14.6.2 (The 'Signal Clients' Editor).

12.5.3 Select Signals to be Sent to HMI Manually

Use the Signal Clients editor to define the signals to be sent to the HMI client.

1. Select an IED and open the Signal Clients editor
2. Find signal row(s) which shall be sent to the HMI (these signals should be generated in the HMI configuration file)
3. Find the column on the right side labeled with the HMI name and mark the cell with an "x" or double click the cell.

In most cases, one would send all of the signals to the HMI. To do this, use the ‘Fill Down’ function as described in chapter 3.2.2 (Context Menus).

### 12.5.4 Copy Full Signal Client HMI Configuration from Existing HMI

It is possible to transfer the full configuration from an existing, configured HMI to another HMI.

1. In the Signal Client Editor, right-click in the Column Header of the Destination HMI, then click on ‘Copy Configuration from’ -> a submenu with available HMIs will appear.

(If the ‘Copy configuration from’ context menu is disabled, the right-click was in an unsuitable place)

2. Click on the desired source HMI in this context menu -> one of the following dialog will appears

If the data can potentially be copied, a dialog will ask you for confirmation, as the current configuration of the destination HMI is about to be overwritten:
If the destination HMI is also a Gateway and has some existing NCC configuration, IET600 will refuse to copy the HMI configuration, as there are too many things that may go wrong. The dialog will give you advice how to proceed:

3. If copying is possible and you confirm with 'Yes', the HMI configuration will be copied to your destination HMI.

   If the copying is successful, no further confirmation is given (in case of an error, the standard warning dialog appears telling you to check for warnings in the log). You can easily check the result by selecting the 'HMIs' element in the 'HMIs' Navigation tree and then sort on either HMI column: the marks in source and destination columns must match.

12.6 Configuration of Signals of Non-IEC-61850 IEDs

12.6.1 Introduction

   For IEC-61850-Signals, the ‘Mapped To’ is the equivalent of addresses for other protocols within a Substation.

   For other protocols, only MicroSCADA attributes for entering such addresses exist (UN/OA/OB), no generic attributes. To configure other HMIs and/or Gateways is currently not supported.

12.6.2 Supported Protocols

   For MicroSCADA, support is provided for the following protocols:
   - IEC 101/104
   - DNP 3.0
   - Modbus
   - SPA
   - LON
   - SNMP
12.6.3 **Engineering Workflow**

The workflow is very similar to working with the IEC-61850-Protocol, except that there is not data model of the IED.

1. Create IEDs
2. Create a Subnetwork and configure it for the protocol you need.
3. Attach the IEDs to this Subnetwork
4. Import signals
5. Configure ‘Mapped To’ attribute of signals.
6. Provide LN Mapping
7. Configure the signal addresses (OA/OB) according to the protocol specifications
8. Configure signals and signal groups for NCCs (see chapter 14.7, Grouping Signals)

Below, you will find a description only for steps 5 and 7, which are slightly different from engineering signals in an IEC 61850 IED.

12.6.4 **Configure ‘Mapped To’ Attribute of Signals**

This attribute has also a function in non-IEC-61850 protocols

- in SNMP it contains the OID (and optionally some additional information needed for SNMP Signal configuration).
- for internal Signals, it can be used for LN mapping which in turn serves to automatically configure all attributes that can be calculated with formula (see chapter 12.4.4, HMI Text Rules).

12.6.4.1 **Pseudo Mapping to Resolve References**

If no mapping is configured, then the structure to which the IED is attached, is taken as default. Therefore, if an IED is attached to a Bay, formulas which reference Substation, Voltage Level or Bay will resolve even without ‘Mapped To’ Attribute. If an IED is attached to a Substation, only references to the Substation will resolve.

For any signal that needs to be attached to another structure, a Pseudo-LN must be configured, e.g. AA2C1Q01KF1LD0.CSWI1 for a signal belonging to a circuit breaker or disconnector. This Pseudo-LN will then appear in the LN mapping table and can be mapped to a primary equipment, e.g. AA2.C1.Q01.QA1. All formulas referencing this structure (QA1 in the above example) should now properly resolve.

Alternatively, the LN mapping can be skipped and the information written hard-coded to the rule attributes, as described in chapter 12.4.4, HMI Text Rules. However, this approach needs usually more work and does not guarantee automatic update in case of name changes. It is recommended to limit this approach to substations with no IEC-61850 IEDs, and few signals to be edited. In all other cases, a configuration of the ‘Mapped To’ attribute will be more efficient.
12.6.5 Configure Signal Addresses

Older implementation of protocols often reserved specific address ranges for specific data types. Newer protocols like IEC 101/104 often provide one address per information, irrespective of whether it is a simple bit (binary value) or a larger value (e.g. 16bit analogue value).

MicroSCADA additionally also enforces certain address ranges for certain protocols even though the protocol itself may not require this.

To make this configuration as simple as possible, IET600 provides 3 columns:

1. OA Offset, a read-only column, is the offset for the given data type and protocol. It may be empty which is equal to an offset of 0.

2. OA Index is the address within a certain offset range. This attribute is for convenient configuration by the user so that he need not bother about offsets for each protocol. This is the column where the address should normally be configured.

3. OA is calculated as (OA Offset + OA Index), it is the final address as it appears in MicroSCADA data exchange files and also in the actual communication. This value is important when debugging data exchange files and analyzing online communication of MicroSCADA.

12.7 HMI Status Text Configuration

12.7.1 Overview

In IET600, the term “Status Texts” is used to describe the part of a signal that describes the process state, e.g. “on/off” or “intermediate/open/close/faulty”.

To imitate the mechanism of most current HMIs, Status Texts have been implemented as follows:

Each signal has a reference to a status text, the “key”. E.g. the MicroSCADA EH attribute references Status Texts.

Depending on the signal type, the signal can take different values (2 in case of a binary value or binary command, 4 in case of a double binary, an irregular number in case of analogue values). So, Status Texts come in groups; the number of items in such a group is determined by the signal type it refers to.

A collection of Status Texts exists in the project, independent of HMI Signals. If HMI signals are created or imported, the Status Texts are connected via its key, if it exists in the import. Otherwise they have to be configured manually.

So sometime very early in the project, before configuring any HMI signals, you need to import such a Status Text collection.

To import and export the HMI Status text from IET to MicroSCADA, see chapter 16.5.2 and chapter 16.5.3, respectively.
12.7.2 View/Edit Status Texts

Imported Status Text from MicroSCADA can be edited and re-exported from IET600 and imported into the MicroSCADA application, thereby ensuring that changes to the Status Text during IET600 engineering will be consistent to the MicroSCADA project.

To view/edit the Status Texts which have been imported into IET:

1. Select the HMI menu and under Configure Data, click “Status Text”:

2. The following dialog appears:

12.8 Merge HMI Data in MicroSCADA

The engineering procedure described in this section applies only to IET600 5.3 FP1 or later and HMIs/Gateways based on MicroSCADA 9.4 FP1 or later. It does not apply to other Gateways.

12.8.1 Introduction

Sometimes, a Substation needs to be split into 2 IET600 projects for performance or project management reasons. A typical example is a project of a Substation...
which contains an HV part, operated by one utility, and an MV part, operated by another utility, but where all Data should be collected in the HV HMI in the end.

### 12.8.2 Engineering Procedure and Restrictions

Basically, both projects are engineered independently. The data are transferred to MicroSCADA via 2 ‘sasmc’ file, where the ‘MIET’ Tool is used for import; both ‘sasmc’ files are imported independently of each other.

![Diagram](image)

It is important that the two ‘lcf’ files in both ‘sasmc’ files do not contain any duplicate signals, i.e. signals with the same LN and IX: the importing MIET is not designed to handle such situations. If this condition is not adhered to, signal attributes may unexpectedly overwritten; or signals may accidentally and unexpectedly be added or deleted. In worst case, this can lead to an inconsistent Process Object Database in MicroSCADA.

### 12.8.3 Preparation of Projects

1. In both MV and HV project, create the HV MicroSCADA with its name.

2. From the very beginning of the project, define an appropriate naming scheme to avoid duplicate LN:IX combinations for signals in the HV project and in the MV project.

   When the two projects concern different voltage levels, using the ABB naming conventions (where the Voltage Level appears in the LN) will ensure unique LNs for all signals that are related to primary objects and IED supervision signals for IEDs on Bay and Voltage Level.

   However, special care must be taken for signals on Substation Level; these should be mapped to the HV HMI only in the HV project, but not in the MV project. It is the responsibility of the engineer to ensure uniqueness.

3. For the IEC 61850 Dataflow configuration, use OPC Servers Subnetworks with different names in MV and HV projects.

   As an example, use OPC Servers AA1OPC1, AA1OPC2 etc. and Subnetworks AA1WF1, AA1WF2 etc. in the MV substation; OPC Servers AA1OPC5, AA1OPC6 etc. and Subnetworks AA1WF5, AA1WF6 etc. in the HV substation.
12.9 Working With User Defined Signal Templates

12.9.1 Overview

To increase the signal engineering efficiency when creating HMI signals manually, IET600 allows to save signals as template for later reuse. The signal templates belong to the user and can be used in any project.

12.9.2 Create new Signal Template

1. Select an IED and open the HMI Data editor.
2. Select all the rows with the signals you want to save as template (from the leftmost ‘row selector’ column of the editor).
3. Right-click in the row selector column of the editor -> the context menu for the editor rows appears.
4. Select ‘Signal Templates > ‘Save As User Template...’.

A ‘Save As’ dialog will appear, save the signal templates under a meaningful filename.

The root folder is defined by IET600 and cannot be changed. You can create sub-folders to organize your templates in a structured way.

The template contains for each signal the same configuration as the original signal with the exception of attributes that make only sense in the current context. These are the rule results (Station Text, LN, IN, OI, ON) and address information (OA, OB, UN, TI, RI).

The template contains the signal texts and status texts for the internal language and all translations that are available in the current project.
12.9.3 Insert Signal Template

1. Select an IED and open the HMI Data editor.
2. Right-click in the row selector column of the editor -> the context menu for the editor rows appears.
3. Select ‘Signal Templates > Insert User Template’. If this menu item is inactive than your template library does not contain any template to insert.
4. Browse the available templates provided by the context menu and select the one to insert. The hierarchy of the context menus reflects the folder hierarchy of the templates.

New rows are created for the signals contained in the template.

IET600 provides a set of very basic default templates for IEC-61850-signals as shown above. However, if you have already some signals available, it is probably a better idea to create your own Signals Templates from existing signals as described in chapter 12.9.2 (Create new Signal Template) above.

If the current project does not already contain the signal texts or status texts for the customer language, they are copied from the template, if available.

12.9.4 Maintaining Templates

To modify existing signal templates, insert the template into an IED, modify the signals and save them as template using the old name. This will overwrite the existing template with the new one.

Removing or renaming of templates can be solved by windows file system operations.
The templates are stored in the IET600 users application data folder, e.g. ‘C:Users\[username]\AppData\Roaming\ABB\IET600’ in the sub-folder ‘VersionIndependent\SignalTemplates\1.0.1.0’.

The easiest way to retrieve the actual template location is from the ‘Save As’ dialog used to save the templates.

Templates can be transferred between different users (or the same user on a different PC) by manually copying the files.
13 HMI Mirroring

13.1 Overview

Many HMIs have the capability to forward their signals to another HMI. Examples are MicroSCADA or OPC Client/Server combinations. This capability is mostly referred to as “Mirroring.”

The term “Mirroring” is slightly misleading: it is not a mirroring between two devices on an equal footing which continuously synchronize their data, but a transfer of process values from a dedicated server (sender, host) to a dedicated client (receiver, image). As the term “Mirroring” has become common-place for this type of data transfer, it is used here; we will refer to Mirroring Clients and Mirroring Servers to indicate their roles in the configuration.

13.1.1 User Interface

To configure mirroring of HMI signals, you will use the following editors:

- ‘Signal Clients’: to configure signals to different HMIs (direct or via Mirroring) and NCCs.
- ‘HMI Data’ to configure attributes of mirrored signals, if required.

If these Editors are not visible as Tabs, read chapter 3.1.3 (Editors), the last section explains how to make them visible.

13.2 Configure Clients and Servers

To configure mirroring, 2 basic steps are needed:

1. Configure clients and servers in the ‘HMIs’ Navigation Tree.
   1. Configure HMIs as Mirroring Clients.
   2. For each Mirroring Client, configure its Mirroring Servers.

2. For each Client/Server(s)-Combination: configure the signals to be mirrored (i.e. sent from client to server) in the ‘Signal Client’ Editor.
13.2.1 Configure an HMI to be a Mirroring Client

You can only configure mirroring for already existing HMIs and/or Gateways.

Prerequisites:
- existing HMIs that are suitable for Mirroring (e.g. MicroSCADA).

Procedure:
1. Select the ‘HMIs’ tab in the Navigation pane
2. Drag’n’drop an HMI from the ‘HMIs’ element to the Mirroring element.

13.2.2 Configure an HMI to be a Mirroring Server for one Specific Client

Prerequisites:
- existing HMIs that are suitable for Mirroring (e.g. MicroSCADA).
- at least one mirroring client configured.

Procedure:
1. Select the HMIs tab in the Navigation pane.
2. Drag’n’drop an HMI from the ‘HMIs’ element to a mirroring client.

A mirroring-capable HMI can be server to several clients.

13.3 Configure Signals to be Mirrored

Prerequisites:
- existing HMIs that are suitable for Mirroring (e.g. MicroSCADA).
- at least one mirroring client configured.
- at least one server configured for the mirroring client.
- for MicroSCADA: UN configured, OA/OB configured.
Chapter 13

Procedure:

1. Select the 'HMIs' tab in the Navigation pane.

2. Select any element in the mirroring branch:

   The Example shows a cross-configuration as described in chapter 13.5.1 (Mutual Mirroring):

   A Hot-Standby MicroSCADA (HSB 12/13) is HMI on the HV side, two individual MicroSCADAs (AA1KF14 and AA1KF15) are configured as HMIs on the MV side.

   HSB 12/13 is configured as client with AA1KF14 and AA1KF15 as its mirroring servers.

   AA1KF14 is configured as mirroring client with HSB 12/13 as its mirroring server.

   AA1KF15 is configured as mirroring client with HSB 12/13 as its mirroring server.

3. Select the 'Signal Clients' Editor. Depending what kind of element you have selected, you will see the following:

   If you select a Mirroring Client (e.g. HSB 12/13 as client in the above example), you will see the signals of all IEDs configured as mirroring servers for this client, and you will see only the client itself as column:

   As the same process data can be made available via different mirroring servers, an additional column 'Source HMI' is shown.

   This selection is recommended for doing the initial configuration of a client.
If you select a Mirroring Server (e.g. HSB 12/13 as a server below AA2KF14 as client in the above example), you will see all its signals as rows, and you will see all its possible Mirroring Clients (AA1KF14 and AA1KF15 in the above example) as columns:

As you have selected only one server, only signals of this server will be available; ‘Source HMI’ will show the name of this server for all signals.

This selection is useful mostly to compare mirroring to 2 different clients (e.g. in the above example, AA2K14 and AA2K15 are supposed to be redundant, although not configured as a Hot-Standby-System and therefore to have identical configurations)
Chapter 13

if you select the ‘Mirroring Islands’ element, you will see all signals from all possible servers as rows, and all possible Mirroring Clients as columns:

This selection is recommended to check the overall configuration and reconfigure individual signals at a late stage of the project.

4. Configure signals to clients:

Signals configured to be mirrored will be indicated with an “m” (mirrored) or “c” (copied), to distinguish them from signals directly from the process which are marked with “x”. For the distinction between mirrored and copied signals, see chapter 13.3.1 (Mirrored vs. Copied Signals).

- to add individual signals to a client, double-click an empty cell, or enter “x” or “m” or “c”. After successfully mirroring a signal, the cell will indicate either “m” for mirrored or “c” for copied signals.

- to remove individual signals from a client, double-click an “m”/“c” cell, select a cell and press ‘Delete’, delete the cell content or enter [Space] or ‘~’.

Normal mechanisms like ‘Fill Down’ or marking multiple cells and pressing ‘Delete’ are available to rapidly configure multiple signals.

By additionally using filters, you can easily map a group of specific signals to a client. E.g. in the above example, a typical configuration would be to configure switch positions and protection trips from an MV transformer Bay to the HMI of the HV and vice versa. So by filtering the station-dependent text on “Trafo” and the signal text on “Switch position”, you see only the relevant signals and can configure them with ‘Fill Down’. 
5. Control HMI configuration:

To finally check an HMI configuration, select the HMI (e.g. HSB 12/13) in the ‘HMIs’ tab below the ‘HMIs’ element. You will see all directly configured ("x"), mirrored ("m") and copied ("c") signals:

<table>
<thead>
<tr>
<th>Source ID</th>
<th>Process Object Type</th>
<th>Station Target</th>
<th>Signal Text (column1)</th>
<th>StatVar Text (column2)</th>
<th>UN</th>
<th>OE</th>
<th>OE1</th>
<th>OE2</th>
</tr>
</thead>
<tbody>
<tr>
<td>AA2101HFM</td>
<td>Analog Input</td>
<td>86 Local Server</td>
<td>49</td>
<td>AddCurrent of command</td>
<td>SOD</td>
<td>D4</td>
<td>10</td>
<td>x</td>
</tr>
<tr>
<td>AA2101HFM</td>
<td>Analog Input</td>
<td>86 Local Server</td>
<td>49</td>
<td>DevCtrlControl</td>
<td>SOD</td>
<td>D4</td>
<td>10</td>
<td>x</td>
</tr>
<tr>
<td>AA2101HFM</td>
<td>Digital Binary Input</td>
<td>86 Local Server</td>
<td>49</td>
<td>Position Indication</td>
<td>Info</td>
<td>D4</td>
<td>10</td>
<td>x</td>
</tr>
<tr>
<td>AA2101HFM</td>
<td>Binary Input</td>
<td>86 Local Server</td>
<td>49</td>
<td>Block for opening</td>
<td>CMo</td>
<td>D4</td>
<td>41</td>
<td>x</td>
</tr>
<tr>
<td>AA2101HFM</td>
<td>Binary Input</td>
<td>86 Local Server</td>
<td>49</td>
<td>Block Thresholding</td>
<td>CMo</td>
<td>D4</td>
<td>42</td>
<td>x</td>
</tr>
<tr>
<td>AA2101HFM</td>
<td>Analog Input</td>
<td>86 Local Server</td>
<td>49</td>
<td>DevCtrlControl</td>
<td>Info</td>
<td>D4</td>
<td>10</td>
<td>x</td>
</tr>
<tr>
<td>AA2101HFM</td>
<td>Analog Input</td>
<td>86 Local Server</td>
<td>49</td>
<td>Position Indication</td>
<td>Info</td>
<td>D4</td>
<td>41</td>
<td>x</td>
</tr>
<tr>
<td>AA2101HFM</td>
<td>Binary Input</td>
<td>86 Local Server</td>
<td>49</td>
<td>Block for opening</td>
<td>CMo</td>
<td>D4</td>
<td>41</td>
<td>x</td>
</tr>
<tr>
<td>AA2101HFM</td>
<td>Binary Input</td>
<td>86 Local Server</td>
<td>49</td>
<td>Block Thresholding</td>
<td>CMo</td>
<td>D4</td>
<td>42</td>
<td>x</td>
</tr>
<tr>
<td>AA2101HFM</td>
<td>Analog Input</td>
<td>86 Local Server</td>
<td>49</td>
<td>DevCtrlControl</td>
<td>Info</td>
<td>D4</td>
<td>41</td>
<td>x</td>
</tr>
<tr>
<td>AA2101HFM</td>
<td>Analog Input</td>
<td>86 Local Server</td>
<td>49</td>
<td>Position Indication</td>
<td>Info</td>
<td>D4</td>
<td>41</td>
<td>x</td>
</tr>
<tr>
<td>AA2101HFM</td>
<td>Binary Input</td>
<td>86 Local Server</td>
<td>49</td>
<td>Block for opening</td>
<td>CMo</td>
<td>D4</td>
<td>41</td>
<td>x</td>
</tr>
<tr>
<td>AA2101HFM</td>
<td>Binary Input</td>
<td>86 Local Server</td>
<td>49</td>
<td>Block Thresholding</td>
<td>CMo</td>
<td>D4</td>
<td>42</td>
<td>x</td>
</tr>
</tbody>
</table>

This selection gives you an overview of all signals to a specific HMI (whether direct or mirrored) and it shows you which signals will be exported into the configuration file for the HMI (e.g. ‘.lof’ file for MicroSCADA).

As described in chapter 12.5.3 (Select Signals to be Sent to HMI Manually), you can reconfigure direct signals in this selection. However, you cannot edit mirrored signals; to add/remove a mirrored signal, select the appropriate element in the ‘Mirroring Islands’ section and configure it there as described above.

### 13.3.1 Mirrored vs. Copied Signals

For Process Signals (SS=2), MicroSCADA provides an automatic mechanism to transfer the signal status from the mirroring server to the mirroring client; no special configuration is needed.

However, internal signals (SS=1 or 3) are not automatically mirrored; they need to trigger a command procedure which must then write the signal status from mirroring server to mirroring client.

The decision whether a signal is mirrored rather than copied is made based on the MicroSCADA attributes SS, UN and OA: SS must be 2 and UN and OA must not be 0.

If a signal is copied rather than mirrored although you think it should be mirrored, check in the ‘HMI Data’ editor whether UN and OA are configured.
13.3.2 Configuring Copied Signals

IET600 automatically reconfigures the signal on the server as follows:

1. **AN (Event Channel Name):** "MIRR_COMMON" is added.

2. **AA (Action Activation):** is set to 1 (New Value), except if its value was at 2 (Update) before.

3. **AF (Action on First Update):** is set to 1 if SS=2 (automatic) or SS=3 (fictive) to ensure that the Event Channel is also triggered on startup.

When "unmirroring" a copied signal, IET600 automatically reconfigures the signal on the server as follows:

1. **AN (Event Channel Name):** "MIRR_COMMON" is removed.

2. **AA (Action Activation):** is set to 0 only if the Event Channel AN is empty, otherwise, it is left at its value of 1 or 2. In rare cases, this may need to be changed manually by the engineer, if as its former value before mirroring the signal was at some other value.

3. **AF (Action on First Update):** remains where it is, as its former value before mirroring the signal is unknown. In rare cases, this may need to be changed manually by the engineer.

Although, there are situations where such copied signals need to be mirrored via MIRR_COMMON command procedure. IET600 attaches it via Event Channel to all such signals. The logic which signals to mirror and which not needs to be implemented in this command procedure.

13.3.3 Signals Which Cannot be Mirrored

If a cell is greyed out, you cannot mirror the signal from the source HMI to the destination HMI. There are several reasons why this might be the case:

- The source HMI is not configured to be a server to the client HMI representing the column.

  To mirror signals, you need to explicitly configure clients and their servers. This helps to prevent misconfigurations in complex situations.

- The signal is already mapped to the destination HMI directly or mirrored via another HMI.

  To avoid circular and duplicate mapping, a signal can only be configured via exactly one way from the process to an HMI. So if, in the above example (where AA2KF14 and AA2KF15 are configured exactly the same way), you configure a signal, e.g. ‘Sils.Trafo 1.Q0.Position indication’ to HSB 12/13 via AA2KF14, the same signal cannot be configured anymore via AA2KF15. If you decide that it would better be configured via AA2KF15, you must remove the existing configuration via AA2KF14 first.
13.4  **Edit Mirrored signals**

The editing of mirroring signals is done in the HMI Data editor and is basically the same as for other signals. Some few attributes

13.4.1  **Inheritance**

Hierarchy: signal attributes are propagated along the arrows. Modifying an attribute at any level will propagate to the levels below, but break the propagation from the levels above:

Instance HMI direct (e.g. HMI HV):

```
|   |   |   |   |   |
```

Instance HMI mirrored (e.g. HMI MV):

```
|   |   |   |   |   |
```

Instance HMI mirrored, 2nd stage etc.:

```
|   |   |   |   |   |
```

This rule can be used to configure signals differently on the mirrored HMI. E.g. if in the above example, the MV owner insists on signal texts different from those on the HV HMI, the signal text on the mirrored instance on the HMI MV can be changed without affecting the signal text on the direct instance on HMI HV.

13.4.2  **Blocked Attributes**

OA/OB attributes are relevant for mirroring, they cannot be modified in mirrored signals.

13.5  **Common Mirroring Configurations**

13.5.1  **Mutual Mirroring in the Same Project**

This situation often occurs in big substations with redundancies. If you have too many clients (HMIs or Gateways), you may not be able to configure them directly to RCBs/Datasets. Mirroring is then a solution which allows you to overcome this limitation in an indirect way.

Another fairly common occurrence is in Substations with different owners/operators for HV and MV. The HV owner often does not want that the MV owner accesses his IEDs directly and vice versa; they want to use their own HMI as a filter that passes on only restrictively defined data to the other owner/operator.
In both cases, a possible solution is:

- configure HMI HV as mirroring client, configure HMI MV as server to this client.
- configure HMI MV as mirroring client, configure HMI HV as server to this client.

### 13.5.2 Mirroring with one NCC Connection

Additionally to the situation described above in chapter 13.5.1, one HMI (typically the HMI HV may have an NCC connection.

A possible solution is:

- configure HMI HV as mirroring client, configure HMI MV as server to this client.
- mirror all signals needed for the NCC to the HMI HV.
- configure the NCC signals as usual in the Signal Client Editor (signals from the HMI MV will appear similar to the process signals and can be configured, grouped etc. identical to those).
13.5.3 Mirroring with two NCC Connections

This is a combination of the situations described in chapters 13.5.1 and 13.5.2 above: HMI HV and HMI MV exchange both data via mirroring, additionally both HMIs have an NCC connection.

A possible solution is:

- configure HMI HV as mirroring client, configure HMI MV as server to this client.
- configure HMI MV as mirroring client, configure HMI HV as server to this client.
- mirror all necessary signals.
- configure the NCC signals as usual in the Signal Client Editor (signals from the mirrored HMIs will appear similar to the process signals and can be configured, grouped etc. identical to those).

13.5.4 Mirroring to a Control Center

If a customer has several substations with MicroSCADA and a control center on a higher level which is also a MicroSCADA HMI, this can be configured more easily via mirroring than via a generic protocol like IEC101/104.

An efficient solution is:

- configure the higher-level control center (HMI NCC) as mirroring client.
- configure Substation HMIs (HMI SS1-5) as server to this client.
13.5.5 Mirroring Across Two Projects With one Common NCC Gateway

In big substations, due to size/performance restrictions, the Substation may be split into two IET projects; often the MV part and HV part are even engineered by two different engineering departments.

However, there may be one common NCC connection with one common NCC Gateway (in this example shown as a combined HMI HV and Gateway). To configure this NCC Gateway correctly, all signals need to be available on that Gateway.

One possible way to do this is to mirror the HMI MV data into the HMI HV and then do the complete NCC configuration on the HMI HV/Gateway.

The solution is:

- configure HMI HV as mirroring client in Project HV.
- configure HMI MV as mirroring server to this client.
- transfer the complete HMI MV data via ‘lof’ file from the MV project to the HV project and import them into the “mirrored” HMI MV there (this needs to be repeated whenever signals are added to the HMI MV that need to be sent to the NCC).
- (use the facility to put the internal signal text into the OX attribute - described in chapter 16.6.5 - to transfer also the internal signal text to the destination HMI)
- (In case of repeated ‘lof’ transfer: if you import the ‘lof’ file over the old configuration, signals will be added and updated, but not deleted.)
- mirror the desired signals from HMI MV to HMI HV so that they e.g. can be sent to the NCC

13.5.6 Mirroring Across Two Projects With Restricted Data Exchange

Let us assume a situation similar to the one described in chapter 13.5.5 above, but additionally the owner of the MV substation forbids us to pass the full HMI configuration to the other owner/operator for security reason. He insists that he wants
to do a pre-selection of signals, and only those preselected signals are allowed to be passed on to the other owner/operator.

Here, the best solution is to apply mirroring in both projects:

- configure HMI HV* as mirroring client in Project MV.
- configure HMI MV as mirroring server to this client.
- configure HMI HV as mirroring client in Project HV.
- configure HMI MV* as mirroring server to this client.
- (Note the naming of HMI MV* and HMI HV*, both HMIs should have an identical signal configuration on both sides; but from a customer/engineering point of view it is probably more suitable to name them as proposed here).
- mirror the desired, restricted signals from HMI MV to HMI HV* in the MV project (this serves only to restrict the signals transferred to the other HMI).
- transfer the HMI HV* data via ‘lof’ file from the MV project to the HV project and import them into the HMI HV* there.
- (use the facility to put the internal signal text into the OX attribute - described in chapter 16.6.5 - to transfer also the internal signal text to the destination HMI)
- (In case of repeated ‘lof’ transfer: if you import the ‘lof’ file over the old configuration, signals will be added and updated, but not deleted.)
- mirror the desired signals from HMI MV* to HMI HV so that they e.g. can be sent to the NCC.
14 NCC Gateway Engineering

14.1 Introduction

NCC Gateway engineering refers to the engineering needed to exchange data between a Gateway in a substation and a client outside of the substation, typically a regional or national Network Control Center (NCC).

14.1.1 Supported Gateway Products

Currently, IET600 provides full support to engineer MicroSCADA-based Gateways (COM500; SYS600 MicroSCADA Pro and SYS600C with integrated Gateway).

IET600 provides limited support for COM581 and RTU560 Gateways:
1. Signals can be mapped to a Gateway.
2. Specific command signals can be mapped to a Gateway to indicate that this command is to be configured for NCC operation (the configuration itself is to be done in the Gateway Configuration Tool).
3. No alarm groups can be configured.
4. No addresses can be configured and passed to the Gateway Configuration Tool.

14.1.2 Supported Gateway Protocols

The following protocols between Gateway and NCC are currently supported:

- IEC101
- IEC104
- DNP 3.0

14.1.3 User Interface

To do Gateway engineering, you will use the following editors:

- ‘Signal Clients’: to configure signals to different NCCs, either directly or grouped.
- ‘NCC Offset’ and ‘NCC Data’ to configure attributes for sending signals to NCCs.
14.1.4 **Prerequisites**

NCC engineering is a general term for the functionality needed to exchange data between the station computer or gateway to clients outside of the substation, such as the Network Control Center (NCC). To configure this data exchange, the data model of the IEDs in the system and the communication (between the IED and station computer/gateway) must be defined.

14.2 **Engineering Workflow**

To configure this data exchange, the data model of the IEDs in the system and the communication (between the IED and station computer/gateway) must be defined.

Note that all gateways (except COM500) are treated as IEDs and require an import of an ICD file to generate the IEC 61850 structure.

The following steps are required for engineering of the gateway for one NCC connection:

1. define Gateway devices and data connection to NCCs (e.g. single connections, hot standby configuration)
2. define data points or signals which are to be sent to the NCC, including conditions for transmission
3. define grouping of data signals
4. configure protocol-specific information (addressing information, data type mapping, etc.)
5. export signal list (if needed for customer discussions or verification)
6. export device-specific configuration files

One important goal of NCC engineering is to provide the customer with a list of data points which will be delivered to the NCC. This list is normally required at an early stage in the project and should be consistent with the final engineered system.

For more than one NCC, IET provides support to copy or duplicate the configuration data from one NCC to reuse for other NCC connections (e.g. 2 redundant gateways with identical configuration).
14.3 Creating and Configuring Gateways

14.3.1 Overview

The Gateway or Proxy device must be defined in the HMI Navigation pane. There are two types of Gateways:

- A stand-alone Gateway
- A Gateway with integrated HMI

Note that the communication configuration of the Gateway (e.g. the Subnetwork to which the Gateway belongs) may need to be defined separately in the Communication navigation tab.

Two types of Gateway nodes are supported:

- a single Gateway
- a hot standby (HSB) Gateway

A Hot Standby (HSB) configuration is a cluster of several Gateways or NCCs that are redundant and have identical HMI Signals, but different communication configurations. Devices which are defined in a HSB configuration will be treated as one device in the NCC Data Editor.

The following icons are used in the HMI navigation tree to represent the various HMI and Gateway types:

<table>
<thead>
<tr>
<th>Icon</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image1" alt="Icon" /></td>
<td>Stand-alone HMI</td>
</tr>
<tr>
<td><img src="image2" alt="Icon" /></td>
<td>Stand-alone HMI, linked</td>
</tr>
<tr>
<td><img src="image3" alt="Icon" /></td>
<td>Hot-Standby HMI</td>
</tr>
<tr>
<td><img src="image4" alt="Icon" /></td>
<td>Hot-Standby HMI, linked</td>
</tr>
<tr>
<td><img src="image5" alt="Icon" /></td>
<td>Stand-alone Gateway</td>
</tr>
<tr>
<td><img src="image6" alt="Icon" /></td>
<td>Stand-alone Gateway, linked</td>
</tr>
<tr>
<td><img src="image7" alt="Icon" /></td>
<td>Hot-Standby Gateway</td>
</tr>
<tr>
<td><img src="image8" alt="Icon" /></td>
<td>Hot-Standby Gateway, linked</td>
</tr>
<tr>
<td><img src="image9" alt="Icon" /></td>
<td>Stand-alone NCC</td>
</tr>
</tbody>
</table>
14.3.2 Create a Stand-Alone Gateway

A Gateway can be created from the HMI tab in the Main menu.

It can also be created from the context menu of several Navigation Tree objects:

The details are as described in chapter 7.2.5 (Create New HMI). The configuration of OPC servers running on HMIIs of defined types (e.g. MicroSCADA) is also described there.

14.3.3 Create a Gateway combined with HMI

Combined HMI/GW IEDs need to be defined under both the ‘HMIIs’ and the ‘GWs / Proxies’ element of the HMI navigation tab.

1. Select the ‘HMIIs’ Navigation Tree and select an existing HMI or Hot Standby (HSB) HMI device
2. Drag and drop the HMI device (e.g. HSB12/13) onto the 'GWs / Proxies' element

3. The HSB 12/13 device is now defined as a Hot-Standby Gateway.

14.3.4 Create a Hot-Standby (HSB) Gateway System

1. Select the 'HMIs' Navigation Tree and select the 'GWs / Proxies' element
2. Right-click it to open the context menu and select 'Create new HSB…'
3. Specify the name of the new HSB → click OK.

4. The new HSB Gateway is now shown under the 'GWs / Proxies' node.

5. Create two stand-alone Gateways as described in the previous sections.

6. Select the first Gateway device → drag and drop it into the HSB node.

7. Select the second Gateway device → drag and drop it into the HSB node.

14.4 Creating NCCs

14.4.1 Overview

An NCC device defines a remote client outside of the substation perimeter. Consequently, the NCC node in the HMI navigation pane provides the client view within each Editor. For each NCC node, one needs to define:

- the Gateway server(s) providing data to a particular NCC

Two types of NCC nodes are supported:

- a single NCC
- a hot standby (HSB) NCC

A Hot Standby (HSB) configuration is a cluster of two Gateways or NCCs that are redundant and have identical HMI Signals, but different communication configurations. Devices which are defined in a HSB configuration will be treated as one device in the NCC Data Editor.
14.4.2 Create a Stand-Alone NCC

A Gateway can be created from the HMI tab in the Main menu.

It can also be created from the context menu of several Navigation Tree objects:

14.4.3 Create a Hot-Standby (HSB) NCC System

1. Select the ‘HMIs’ Navigation Tree and select the ‘NCCs’ element
2. Right-click it to open the context menu and select ‘Create new HSB...’
3. Enter the HSB name in the dialog and click OK

4. The HSB device is now created under the 'NCCs' element

5. Create two stand-alone NCCs as described in the previous section

6. Select the first NCC → drag and drop it onto the HSB element

7. Select the second NCC → drag and drop it onto the HSB element

14.5 Connecting Gateways to NCCs

14.5.1 Overview

Once the NCCs and Gateway devices have been created, the logical connection between these devices can be defined.

An NCC can be connected to one Gateway only. A Gateway, however, can be connected to one or several NCCs.

Possible Gateway and NCC configurations include:

- Single Gateway to single NCC
- Single Gateway to multiple independent NCCs
- Single Gateway to HSB NCCs
- HSB Gateway to single NCC
- HSB Gateway to multiple independent NCCs
- HSB Gateway to HSB NCCs

Connecting the Gateway to the NCCs in the HMI navigation tab specifies which data is available to the NCCs, or in other words which Gateways send data to which NCCs. However, it does not specify the physical or communication configuration of the Gateway and NCCs (e.g. if they belong to the same network). The communication configuration has to be defined separately in the Communication navigation tab.
14.5.2 Connect Gateway to NCC

1. Select a single or Hot Standby Gateway → drag and drop it into a single or NCC Hot-Standby NCC

![Diagram of Gateway and NCC connection]

2. For the simple case that you connect a single Gateway to a single NCC, IET600 asks you immediately after dropping the Gateway to specify the physical connection properties of the link between the Gateway access point and the NCC access point.

![Diagram of Gateway and NCC connection properties setup]

3. The Gateway now appears under the NCC node.

If you have a more complicated situation, e.g. a Hot-Standby Gateway to a Hot-Standby NCC, you need to configure their Subnetworks yourself in the Subnetworks editor. It is best done immediately after creating the Gateway-to-NCC links in the 'HMIs' tab

14.5.3 IEC 101/104 Special Parameters

IEC101/104 protocol requires to specify the size of the Common Address of ASDU (1 or 2 bytes) and of the Information Object Address (2 or 3 bytes).
As default, IET600 assumes a Common Address of ASDU size of 1 byte (allowing addresses of 1-255) and an Information Object Address size of 2 bytes (allowing addresses of 1-65535).

Historically, these address ranges were sufficient in most projects. However, more and more customers require the full range of addresses. To reconfigure these address sizes, please proceed as follows:

1. Select the NCC IED (in the Substation, IEDs, or Communication Trees).
2. Open the ‘Subnetworks’ Editor.
3. Click on ‘Edit Addresses’ in the row with the Access Point of the NCC that needs the extended address range -> the following dialog opens:

Right-click in the row header to get the row context menu and click ‘Insert new Row’.

4. Add the following as needed (exact spelling is required):
   - To set Common Address of ASDU to 2 byte size:
     Address Type: CommonAddressOctets
     Address: 2
   - To set Information Object Address to 3 byte size:
     Address Type: InformationObjectAddressOctets
     Address: 3

The following dialog shows the above configuration:
14.6 Selecting Signals for NCCs

14.6.1 Overview

Once the Gateway and NCC devices nodes have been created, the next step is to configure the specific signals to be sent to the remote client in the 'Signal Clients' editor.

14.6.2 The 'Signal Clients' Editor

The 'Signal Clients' editor shows the station signals which are available to the Gateway and connected NCCs. It provides the “server” view of the Gateway and shows:

- as rows: signals which are available from the process
- as columns: Gateway(s) and their associated NCCs, and columns for Signal selection and grouping for each NCC:
  - Sel (Select): type ‘x’ to assign a signal to a Gateway and NCC.
  - Grp (Group): enter any identifying string to group signals. Signals with the same string in the same NCC belong to the same group.
  - GrpI (Group Index): An automatically generated incremental number for each member of the group. A positive number indicates a signal going to the process (downlink) and a negative number indicates a signal going to the NCC (uplink).

<table>
<thead>
<tr>
<th>Process Object Type</th>
<th>Station State</th>
<th>Signal Type (State)</th>
<th>Signal Index</th>
<th>Gateway</th>
<th>NCC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Analog Output</td>
<td>On</td>
<td>On</td>
<td>1</td>
<td>Galaxy</td>
<td>4</td>
</tr>
<tr>
<td>Analog Output</td>
<td>Off</td>
<td>Off</td>
<td>2</td>
<td>Earth</td>
<td>5</td>
</tr>
<tr>
<td>Digital Input</td>
<td>Open</td>
<td>Open</td>
<td>3</td>
<td>Venus</td>
<td>6</td>
</tr>
<tr>
<td>Digital Input</td>
<td>Close</td>
<td>Close</td>
<td>4</td>
<td>Jupiter</td>
<td>7</td>
</tr>
</tbody>
</table>

Signal identification

Client Mapping
To view all available columns in the Signal Clients editor, click on the Column Chooser icon to view all available columns.

The Signal Clients editor is empty when an NCC element is selected in the ‘HMIs’ navigation Tree.

14.6.3 Select Signals to Be Sent to NCC

   - If you select any element in the ‘Substation’ ‘IEDs’ or ‘Communication’ Tree, the signals of all IEDs belonging to the selected element will be shown. This is useful if you want to configure (or check the configuration of) all signals of a Bay to different Gateways/NCCs. If you select the project in this tree, you will see all available signals and all available Gateways and HMIs. This is useful to give you an overview over the complete configuration.
   - If you select a Gateway in the ‘HMIs’ Tree, all signals of the project will be shown, and only the selected Gateway. This is useful if you want to check the configuration of a single Gateway and its NCCs.

2. Place an ‘x’ in the matrix under the column heading ‘Sel’ to assign a station signal to a Gateway and NCC

3. These signals are now configured to be sent from the original IED to the Gateway and further to the NCC.

14.7 Grouping Signals

The grouping of signals is currently possible only for MicroSCADA Gateways.

Other Gateways, like COM581 and RTU560 can have only direct signals mapped. As a workaround for command signals (which need to be processed on the Gateway), specific signals can be mapped to the Gateway. The proper configuration of these signals needs to be done in the respective IED configuration tool.

14.7.1 Overview

IET600 distinguishes 2 types of groups
   - Indication groups (needed to configure group alarms)
   - Command groups (needed, because typically commands need to be processed on the Gateway, so all involved signals, including an optional feedback indication are grouped together)
Each group is identified by its name which must be unique within an NCC connection. All signals with the same name in a ‘Grp’ column belong to the same group.

Within a group, IET600 distinguishes between:

- **Downlink Signals** (i.e. Signals exchanged between local IEDs and the Gateway, their GrpInd is positive).
- **Uplink Signals** (i.e. Signals exchanged between the Gateway and the NCC, their GrpInd is negative).

When configuring Groups, the uplink signals are created automatically when the first Group member is created; they are deleted automatically when the last Group member is deleted.

### 14.7.2 Create an Indication Group

1. Select a suitable object in one of the Navigation trees.

   Usually, a good idea is to first select a Bay to create/configure groups of signals within this Bay (a typical example are alarms of switching equipment in a Bay).

   When all groups within Bays are done, then select a Voltage Level or Substation to configure group signals that span across several Bays (a typical example are communication equipment alarms).

2. Select the first signal which you want to add to group.

3. Enter a group name under ‘Grp’ column (e.g. “Al1”) which does not exist yet.

   An uplink signal (e.g. Group Alarm 1 for NCC NCC51) is automatically generated by IET600.

   Please note that the visibility of the uplink signal depends on the object selected in the Navigation tree. While the other signals belong to a bay-level IED, the uplink signal belongs to the Gateway. It will therefore only appear, if you
have selected the Gateway in any of the Navigation Trees. The only way to see all signals of a group is to select an appropriate object in the ‘HMI’ Tree.

4. Add more signals to this group by putting the same group name (e.g. “Al1”) into the group column for other signals. They will be added to this group.

5. (the ‘GrpInd’ column shows the index of the signal within the group, for the uplink signal, a negative index is used).

Please note that a signal does not need to be selected in the ‘Sel’ column to be part of a group. Selecting it in the ‘Sel’ column will cause the signal to be sent to the NCC directly.

Please note that the uplink signal is initialized in a way that limits group signals to 240 (this is a limitation of the ‘lof’ file used for data exchange with MicroSCADA). If you are configuring more group signals into a group (which IET600 will allow you to do), you will need to initialize the uplink signal in MicroSCADA with an appropriate command procedure, otherwise the signals with GrpInd > 240 will not trigger the Group Alarm.

### 14.7.3 Create a Command Group

For some command groups, a specific number of signals is required for proper functioning. IET600 supports this by collecting all required signals and creating a group, whenever just one signal is selected.

The logic for collecting these groups depends on the gateway and on the protocol between Gateway and bay-level IEDs. For a more detailed description, please consult chapter 25, “Annex: NCC Command Group Logic”.

To create a command group:

1. Select a signal suitable as command (one of the signal types listed above (e.g. Position indication) and enter a group name in the ‘Grp’ column (e.g. “CC1”)
2. IET600 will automatically find the remaining signals in the group and assign these also to the same group (e.g. "CC1").

If there is no suitable logic for assembling the group, it will behave like an indication group: The group will be created with one command, you can then add additional command signals to this group.

Please note that the visibility of the uplink signal depends on the object selected in the Navigation tree. While the other signals belong to a bay-level IED, the uplink signal belongs to the Gateway. It will therefore only appear, if you have selected the Gateway in any of the Navigation Trees. The only way to see all signals of a group is to select an appropriate object in the ‘HMIs’ Tree.

14.8 Copying a Configuration from an Existing NCC

14.8.1 Copy Full Signal Client NCC Configuration from Existing NCC

It is possible to transfer the full configuration from an NCC to another NCC, provided the NCC Gateway is the same type (e.g. from RTU560 to RTU 560 or from COM500 to COM500).

1. In the Gateway column headers, right-click in the NCC part of the destination NCC, then click on ‘Copy Configuration from’ -> a submenu with available NCC configurations will appear (if a menu item is greyed out, it cannot be copied; e.g. because the Gateway is of a different type that has an incompatible configuration).

(If the ‘Copy configuration from’ context menu is disabled, the right-click was in an unsuitable place -> make sure that you clicked in the NCC part as indicated)
2. Click on the desired source NCC in this context menu -> a dialog will appear, asking you for confirmation (be aware that the current configuration of the destination NCC will be overwritten).

![Confirmation dialog]

3. If you confirm with ‘Yes’, the NCC configuration will be copied to your destination Gateway/NCC. If the destination Gateway is also configured as an HMI, existing HMI signals will be preserved; however, some HMI signals may be added, if required by the Gateway configuration (identical to when you configure NCC signals manually on a combined HMI/Gateway).

4. If the copying is successful, no further confirmation is given (in case of an error, the standard warning dialog appears telling you to check for warnings in the log). You can easily check the result by selecting the ‘GW/Proxies’ element in the ‘HMIs’ Navigation tree and then sort on either ‘Sel’ or ‘Grp’/’GrpInd’ column: the configurations in source and destination columns must match.

### 14.9 Configure NCC Data

#### 14.9.1 Overview

The NCC Data editor provides the functionality to configure the signal specific parameters at the NCC.

#### 14.9.2 NCC Data Editor

The NCC Data Editor is available from the HMI Navigation pane. The editor provides a filtered view of the station signals which are available to the specified NCC only.

Note that the NCC Data Editor provides the “client” view to the Gateway. It shows the signals which are mapped to the specific NCC only.
The editor shows on the left hand side the station signals which have been selected in the Signal Clients editor. On the right hand side, the editor shows a matrix for configuration of the NCC signals.

To view all available columns in the NCC Data editor, click on the Column Chooser icon to view all available columns.

### 14.9.3 Configure NCC Signal Properties

The following sub-headings are available on the NCC Signal Properties section:

- **Address**
- **Signal Handling**
- **Command Type**
- **Transmission Class**
- **Scale Name**
- **Send Trigger** (default: OnChange, currently not used)

Note that only the white cells in the NCC signal properties columns are editable fields, meaning that one can manually specify the attribute in the field. The grey cells are automatically defined by the tool.
14.10 Merge NCC Data in MicroSCADA

The engineering procedure described in this section applies only to IET600 5.3 FP1 or later and HMIs/Gateways based on MicroSCADA 9.4 FP1 or later. It does not apply to other Gateways.

14.10.1 Introduction

Sometimes, a Substation contract is split into 2 IET600 projects. A typical example is a project of a Substation which contains an HV part, operated by one utility, and an MV part, operated by another utility. Often, these two projects have different timelines, owners etc. and are given to the vendor as two different contracts.

We will assume the above situation. Additionally, we assume that an NCC connection is engineered from a MicroSCADA in the HV substation, but which should also include information from the MV part.

There are two ways to proceed with this:

1. Use mirroring as described in chapters 13.5.5 (Mirroring Across Two Projects With one Common NCC Gateway) or 13.5.6 (Mirroring Across Two Projects With Restricted Data Exchange).

   This is recommended under the following conditions:
   • the engineers do not have extended experience in MicroSCADA engineering
   • at least the project on the MV side is fairly stable, no or very few changes are expected to happen.

   This is the more common situation, as typically, the NCC engineering is done late in the project, when engineering on Substation level is more or less finished. This procedure is safer, because the owner of the NCC connection has full control over the NCC engineering.

2. Use the engineering procedure described in the following chapter, where both projects are engineered independently, but are merged only in MicroSCADA.

   This is recommended only under the following circumstances:
   • the naming of both Substation parts (MV and HV) are free of conflicts.
   • the projects have many ongoing changes in HMI and/or NCC Gateway configuration.
   • both responsible project engineers cooperate well and have extended experience in MicroSCADA engineering.
   • no Alarm Groups containing signals from both MV and HV project are required.

   This procedure may be more efficient in certain situations than the one above, but the risk of unexpected conflicts is higher. It requires experienced engineers and a good coordination between the projects to mitigate this risk to acceptable levels.
14.10.2 Engineering Procedure and Restrictions

Basically, both projects are engineered independently. The data are transferred to MicroSCADA via 2 ‘sasmsc’ file, where the ‘MIET’ Tool is used for import; both ‘sasmsc’ files are imported independently of each other.

It is important that the two ‘lof’ files in both ‘sasmsc’ files do not contain any duplicate signals, i.e. signals with the same LN and IX: the importing MIET is not designed to handle such situations. If this condition is not adhered to, signal attributes may unexpectedly overwritten; or signals may accidentally and unexpectedly be added or deleted. In worst case, this can lead to an inconsistent Process Object Database or NCC configuration in MicroSCADA.

14.10.3 Preparation of Projects

1. In both MV and HV project, create the HV MicroSCADA with its name.

2. From the very beginning of the project, define an appropriate naming scheme to avoid duplicate LN:IX combinations for signals in the HV project and in the MV project.

   When the two projects concern different voltage levels, using the ABB naming conventions (where the Voltage Level appears in the LN) will ensure unique LNs this for all signals that are related to primary objects and IED supervision signals for IEDs on Bay and Voltage Level.

   However, special care must be taken for signals on Substation Level; these should be mapped to the HV HMI only in the HV project, but not in the MV project. It is the responsibility of the engineer to ensure uniqueness.

3. For the IEC 61850 Dataflow configuration, use OPC Servers Subnetworks with different names in MV and HV projects.

   As an example, use OPC Servers AA1OPC1, AA1OPC2 etc. and Subnetworks AA1WF1, AA1WF2 etc. in the MV substation; OPC Servers AA1OPC5, AA1OPC6 etc. and Subnetworks AA1WF5, AA1WF6 etc. in the HV substation.
14.10.4 Calculate the NCC Gateway Group Offset

While you can change the offset later in the project, it is recommended to set it correctly from the very beginning. To calculate it, estimate the number of required NCC Command Groups and NCC Alarm Groups in one project to calculate the offset for the other project:

1. For commands, add all the planned commands for all NCCs that will be connected to this Gateway. E.g. if in project HV, you have 2 NCC connections, and 1 NCC will have 72 commands (e.g. 10 Bays with 7 remotely operated Switches each + 2 Tap Changer Commands), the other 70 (e.g. 10 Bays with 7 remotely operated Switches, but no Tap Changer Operation), the minimal offset to be set in the MV project is 72.

2. For Alarm Groups, take the biggest number of alarm groups to be sent to a single NCC (as each NCC has signals with a different LN; groups need not be added across all NCCs). E.g. if in project HV, you have 2 NCC connections, and 1 NCC will have 35 Alarm Groups, the other 30, the minimal required would be 35.

3. Take the bigger of those two numbers (in this case 72) and add a considerable safety margin to allow project changes and later expansion without having to change the offset. In this example, probably an offset of 200 would be suitable.

14.10.5 Configure the NCC Gateway Group Offset

Use this offset only, when using ‘sasmc’ files for Data Exchange in MicroSCADA, not when using ‘lof’ and ‘xref’ files.

After calculating the offset, as described above, configure this offset in one project, e.g. the MV project. Leave it at 0 (the default value) in the other project.
1. In the Navigation Tree ‘HMI’s’, in the ‘GWs/Proxies’ section, select a MicroSCADA Gateway or MicroSCADA Hot-Standby system and right-click it -> the context menu for the element appears:

2. Select ‘Configure NCC Gateway Group Offset’ (if the menu is not visible, you have selected a non-MicroSCADA Gateway) -> the following dialog appears:

![Configuration dialog screenshot]

Enter an offset that is big enough to prevent overlapping of any NCC-relevant items.

3. After setting this offset, IET600 will automatically recalculate some attributes which are automatically given when configuring NCC groups:

- **NCC Alarm Group Uplink Signals** will have the IX recalculated, beginning with the offset + 1 and continuing sequentially:
  - BNCCy_GRP:P1 -> BNCCy_GRP:P2001
  - BNCCy_GRP:P2 -> BNCCy_GRP:P2002
  - BNCCy_GRP_AS:P2 -> BNCCy_GRP_AS:P2002
  etc.

- **NCC Command Group Uplink Signals** will have their LN recalculated, beginning with the offset + 1 and continuing sequentially:
  - BNCC__00001:P10 -> BNCC__02001:P10
  - BNCC__00001:P11 -> BNCC__02001:P11
  - BNCC__00001:P12 -> BNCC__02001:P12
  - BNCC__00002:P10 -> BNCC__02002:P10
  etc.

- **NCC Command Group TI attributes** will have the offset added.
• NCC Indication Group TI attributes will not be modified, as they refer either to the number of signals (on the Uplink Signal) or to the position within a bit string (on the Downlink Signals).

This can be verified in checking the appropriate attributes in the 'HMI Data' Editor.

4. Additionally, on the export of the Alarm Groups into [HMI]_xrefind.txt in the ‘sasmc’ file, the Ag attributes will not start at 1, but at offset + 1 (this is not visible in IET600 directly, but is only calculated upon export).
15 Bay-level Gateway Engineering

15.1 Overview

IET600 supports the integration of all IEC 61850 bay-level gateways using the ICD/IID file of the gateway (Bottom-up). The ICD/IID file contains the IEC 61850 model of the gateway itself as well as an IEC 61850 representation of the connected devices. This file must be generated by the gateway configuration tool and will be imported into IET600 for system engineering.

15.2 SPA-ZC40x

The CET for SPA-ZC40x is the configuration tool for the SPA/IEC 61850 gateway. Use CET to configure and download the configuration of the SPA-ZC40x gateway and create a CID file. Please refer to the CET for SPA-ZC40x manual for further information.
16 Data Import/Export

16.1 ICD/IID File Exchange within a Project

16.1.1 Import an ICD/IID File to Update IEDs

This can be done only from the context menu in the Navigation Trees, not from the Main menu.

Right-click on a Project, Substation, Voltage Level, Bay or IED -> the context menu appears:

Choose ‘Update IED’ (from an IED context menu) or ‘Update IEDs’ (from all other context menus).

Proceed as described in chapter 7.3 (Updating IEDs).

16.1.2 Export an ICD/IID File

In the normal engineering workflow, this should never be needed, as per IEC 61850, a System Tool should only export an SCD file including the complete configuration with all IEDs, it should not export single IEDs.

In some few cases, legacy IED configuration tools had problems in importing a full SCD file, due to XML errors, too big SCD files etc. Also for some support questions, it may be helpful to export only one IED instead of a big SCD file. Only in such special cases, these IID/ICD files should be used; they are to be understood as a workaround.

This export can be done only from the context menu of IEDs in in Navigation Trees.

1. Right-click on an IED -> the context menu appears:
2. Choose ‘Export to SCL file...’ -> a ‘File Save’ Dialog appears.
3. Optionally adapt the filename.
4. Save the file.

The exported file will contain 1 IED with its DATA model and Dataflow; and it will also contain all related Subnetworks with their Connected Access Points. A Substation Section will not be exported.
16.2 SCD File Exchange Within a Project

16.2.1 Import an SCD File to Create a Project

You can import a whole SCD file (including Substation and Communication structure) only once per project. The idea is that an SCD file can be used as an initial starting point for a retrofit or extension project.

After the first import, any additional SCD import will treat the SCD file as a collection of IEDs and allow you to selectively import some or all IEDs.

An SCD file can be imported either from the Home menu or from the navigation tab.

1. Open a new project
2. Select the ‘Home’ menu tab and click on ‘Import’
3. Alternatively, from the context menu of a project in any Navigation Tree, select ‘Import SCD File’.
4. The normal dialog for selecting a file appears. Select an appropriate SCD file and click ‘Open’.

The contents of the file will be imported without further confirmation.
5. Check the contents of the project after the import

- If you choose the wrong SCD file or you are not satisfied otherwise, close the project without saving. You will not be able to import another SCD file in this project, therefore it would be no use to attempt to delete e.g. the Substation structure in the hope to import it again from another SCD file.
• If you choose an ICD/IID file (probably by accident; an ICD/IID file will not make much sense in this situation), you will probably find only one or several IEDs, but no Substation structure. Again, close the project without saving; do not try to attempt any other way to correct it.

16.2.2 Import an SCD File to Update IEDs

This can be done only from the context menu of a project, not from the Main menu
1. Right-click on a Project, Substation, Voltage Level, Bay or IED -> the context menu appears:
2. Choose ‘Update IED’ (from an IED context menu) or ‘Update IEDs’ (from all other context menus).
3. Proceed as described in chapter 7.3 (Updating IEDs)

16.2.3 Export an SCD File

(This chapter describes and SCD file export for projects which contains either Edition 1 or Edition 2 IEDs only. For projects which contain both – so-called “mixed” systems – please read chapter 16.2.4 below).

An SCD file can be exported either from the Home menu or from the navigation tab.
1. Select the ‘Home’ menu tab and click on ‘Export’
2. Alternatively, right-click on the project in any Navigation Tree and select ‘Export SCD File’ from the context menu:
3. If the project is not mixed, a normal 'File save' dialog appears. Select the directory where you want to save the file and give it an appropriate name.

(If any SPAZC40x Gateways exist in the project, their Loadfiles will automatically be exported on an SCD file export, they will be found in a subfolder named “SPAZC40x” to the folder the SCD file is exported to.).

Under ‘Save as Type’ in the ‘File Save’ Dialog, you may have several options available:

- ‘SCL version 2003’ refers to an Edition 1-compatible SCL format (only available for Ed.1 projects).
- ‘SCL version 2007 Rev.B’ including Switch model will also export the DATA model of Switches, although it is neither Edition-1 nor Edition-2-compliant. This is only recommended if you need the Switch model to be included in the SCD file.


16.2.4 Export an SCD File, Mixed Systems

(This chapter describes and SCD file export for projects which contains both Edition 1 or Edition 2 IEDs, so-called “mixed” systems. For projects containing only either Edition 1 or Edition 2 IEDs, please read chapter 16.2.3 above). An SCD file can be exported either from the Home menu or from the navigation tab.

Preliminary: check the allocation of IEDs to Subnetworks in the Communication Navigation Tree:

- IEC 61850 Subnetworks with only Edition 1 IEDs will show a small ‘E1’ in the icon.
- IEC 61850 Subnetworks with only Edition 2 IEDs will show a small ‘E2’ in the icon.
- IEC 61850 Subnetworks with mixed Edition 1 and Edition 2 IEDs will show a small ‘M’ in the icon. These Subnetworks can only be exported into Edition 2 files and have some restrictions; in a fully engineered project, they should be avoided as much as possible for compatibility reasons.
- Non- IEC 61850 Subnetworks will not have any marker.
1. Select the ‘Home’ menu tab and click on ‘Export’

![Image of the 'Home' menu tab and 'Export' option]

2. Alternatively, right-click on the project in any Navigation Tree and select ‘Export SCD File’ from the context menu:

![Image of the context menu with 'Export SCD File' highlighted]

3. The following dialog appears:

![Image of the export dialog]

You have several options to select:

- Export the complete SCD file (its default name is <project name>.scd). This SCD file will contain all IEDs in the project. This option is best when exchanging SCD files with up-to-date Edition 2 IED configuration Tools (e.g. PCM600 2.7).

- Export Edition 2 IEDs only (SCL Version 2007B, therefore its default name is <project name> - 2007B.scd). This file will contain all IEDs connected to Edition 2 and mixed Subnetworks.
Chapter 16

This options should be used for SCD file transfer to Edition 2 IED configuration tools which do not accept user-defined CDCs, if your project contains IEDs with such CDCs (e.g. old ABB IEDs had an LPHD with a user-defined CDC “RSS” for redundant port supervision).

- Export Edition 1 IEDs only (SCL Version 2003A, therefore its default name is <project name> - 2003A.scd). This file will contain all IEDs connected to Edition 1 Subnetworks.

This options should be used for SCD file transfer to Edition 1 IED configuration tools, unless they can import the complete SCD file without any error messages.

- Additionally, for Edition 2 exports, you can check the “Include Switch Model” flag. As the default Switch model is not fully Edition 2 compliant, Switches by default will be exported as IEDs with Access Points only (including address information). If the flag is checked, the full DATA model of IEC-61850 Switches will be included in the SCD file.

This options should only be used for SCD file transfer to tools which need the Switches, e.g. HMIs/Gateways which supervise these Switches.

Any partial export will contain only Subnetworks for the IEC 61850 protocol. This is done to ensure maximum compatibility with IED configuration tools. however, to include non-61850 protocols like SNMP, IEC104 etc., you must export the complete SCD file.

4. Once you have made your choice, click 'Export' -> the appropriate files will be exported. Click 'Cancel' to close the dialog without exporting anything.

(If any SPAZC40x Gateways exist in the project, their Loadfiles will automatically be exported on an SCD file export, they will be found in a subfolder named “SPAZC40x” to the folder the SCD file is exported to.).


16.3 SSD File Import

16.3.1 Overview

IEC 61850 allows a customer to specify a Substation via so-called SSD file (System Specification Description). This was not used very much until now. However, some customers have discovered this format and have started to provide specifications with an SSD file.

An SSD file can specify a Substation on at least 5 different levels of detail; each level must include lower-level specifications:

1. Substation.
2. LN Mapping.
3. Data Type Templates (defining the details of an LN).
4. IEDs.
5. Communication.

You can import an SSD file only once per project. The idea is that an SSD file can be used as an initial starting point for a retrofit or extension project.

16.3.2 Substation Specification

This is the lowest level. It essentially contains the information that a customer would otherwise provide in a Single Line Diagram.

This can easily be imported into IET600. However, often the Substation Topology is not correctly designed, the coordinate system does not match the one of IET600 etc.; usually the engineer needs to invest some time to rearrange items.

16.3.3 LN Mapping Specification

Additionally to the specification of Substation containers and equipment, an SSD file can contain LN Mapping.

This can be used to define required functionality; e.g.:
- by allocating a PDIS to a Bay or a Circuit Breaker, the customer specifies a Distance Protection.
- by allocating CSWI, CILO and XSWI, the customer specifies a remotely operated Disconnector, while by allocation only an XSWI, he could specify a switch which can only be hand-operated.

This allocation can be used in IET600. However, as the unspecific LNs in the file may not match the LNs of specific IEDs (e.g. CSWI1 in the SSD against SCSWI1 in an ABB REC670 IED), manual work by the engineer is typically required to reuse this mapping.

16.3.4 Data Type Templates Specification

While LN mapping can be used to specify certain LNs, an LN itself is still not very well defined. As many LNs contain very few mandatory, but many optional DOs and DAs, the implementation may vary considerably from vendor to vendor.

By specifying Data Type Templates, a customer can define that he also wants certain optional DOs and DAs for his project. The vendor must then ensure that his LNs match the specification of the customer.

IET600 does not provide support for this comparison. A comparison in this case is difficult: the full structure of the Data Type Templates needs to be manually checked. This can only be done by persons thoroughly familiar with IEC 61850 and the Data Type Template Structure; it is not recommended for the average engineer.
16.3.5 IED Specification

By not only specifying Data Type Templates, but IEDs with a DATA structure, a customer can also specify which functionality he wants to reside in which IED. A typical application of that would be the case of two protection IEDs Main 1 and Main 2, where the customer could specify which IED contains which protection functionality.

As with Data Type Templates, IET600 does not provide support for this comparison. In this case, however, there is some possibility of comparison: import the SSD file into one IET600 project and some actual IEDs into another project, and then compare their details manually, e.g. in the ‘LN Data’ Editor.

16.3.6 Subnetwork Specification

Using this last level, a customer could e.g. also specify communication addresses, e.g. IP Addresses.

16.3.7 Import an SSD File

You can import an SSD file only once per project. The idea is that an SSD file can be used as an initial starting point for a retrofit or extension project.

An SCD file can be imported either from the Home menu or from the navigation tab.

1. Open a new project
2. Select the ‘Home’ menu tab and click on ‘SSD Import’

3. Alternatively, from the context menu of a project in any Navigation Tree, select ‘Import SSD File’.
Chapter 16

The ‘Import SSD File’ dialog appears. Select an SSD file to open:

4. Once you have selected the SSD file, IET600 will analyze it and show you a summary of its contents. Select those levels you want to import (selecting a higher level will always include lower levels):

5. If you have made your choice, click ‘Import’ -> the selected items will be imported into your new project.
16.4 SED File Exchange Between Projects

16.4.1 Introduction

In certain situations, Dataflow configuration between two projects is needed; which means that two System Configuration tools must exchange IED data and modify their Dataflow in a way that preserves consistency.

To allow this, IEC 61850 defines a Data exchange between two System Configuration tools via so-called SED file (SED stands for System Exchange Description).

To ensure that only one System Configuration tool can edit Dataflow (thereby ensuring consistency), IEC 61850 specifies two attributes for each IED:

1. “Engineering rights” which can be:
   - full (this is the normal state, DATA and Dataflow can be changed)
   - Dataflow (only Dataflow can be changed)
   - fix (neither DATA nor Dataflow can be changed)

2. “Owner”, which determines to which project an IED belongs.

The workflow is a bit complex. We will describe the following:

- Basic workflow for SED exchange (chapter 16.4.2)
- Pre-conditions for successful SED exchange (chapter 16.4.3)
- IET600 specific behavior and good engineering practices regarding SED Exchange (chapters 16.4.4 and 16.4.5)
- Example with detailed step-by-step guidance (chapters 16.4.6 - 16.4.14)
- IET600 SED file contents (chapter 16.4.15)

16.4.2 Basic Mechanism

Based on the engineering rights and ownership defined in IEC 61850, IET600 implements a check-out/check-in mechanism:

<table>
<thead>
<tr>
<th></th>
<th>Project A</th>
<th>Project B</th>
</tr>
</thead>
<tbody>
<tr>
<td>before Check-Out</td>
<td>IED A, full</td>
<td>SED File</td>
</tr>
<tr>
<td>during Check-Out</td>
<td>IED A, fix</td>
<td>IED A, Dataflow</td>
</tr>
<tr>
<td>after Check-In</td>
<td>IED A, full</td>
<td>IED A, fix</td>
</tr>
</tbody>
</table>

IED owned by the project:
- Project A owns an IED A which has “full” engineering rights.
Chapter 16

• Project A checks out IED A by exporting it into an SED file with “Dataflow” engineering rights.
• After checking it out, IED A has “fix” engineering rights in Project A, i.e. you cannot change its DATA (via ‘Update IED’) or its Dataflow.
• After checking it back in (by importing an SED file containing these same IEDs), it has again “full” engineering rights.

Foreign IED:
• Project B gets IED A with “Dataflow” engineering rights by importing an SED file.
• Project B changes the Dataflow in IED A.
• Project B checks out IED A by exporting it into an SED file, afterwards IED A has “fix” engineering rights in Project B.

16.4.3 Pre-Conditions for SED Data Exchange.

An SED Data Exchange has some pre-conditions which should be checked before planning to use it in the engineering process:

1. Ensure that there are no IED name conflicts.
   If a project has an IED with the same name as an IED to be transferred from another project, then an SED exchange is not possible!
   This cannot be resolved by changing an IED name on the fly just for the data exchange, because the IED name will also appear in real-time communication. If possible, the name of one of the conflicting IEDs must be changed in its own project. However, if this happens late in the project, or – even worse – when configuring a Data exchange between operating substations, this may result in a lot of unexpected reconfiguration work.
   (When using the ABB naming scheme, substations should therefore not always be named “AA1”, but “AA1”, “AA2” etc.; at least within the substations of one customer or area. Otherwise, a naming conflict is very likely).

2. Agree about a common Subnetwork for the Dataflow exchange between the projects.
   To configure Dataflow, both sending and receiving IEDs must be on the same Subnetwork. As IEDs can only be connected to Subnetworks via Access Points, the number of Access Points in the IEDs participating in the SED exchange limits possible (re-)configurations.

3. Ensure that there are no IP-Address and MAC-Address conflicts.
   For IT security reasons and to keep the amount of data transferred between projects small, not all IP addresses or MAC addresses of a project will be provided in the SED file.
   The engineer has to ensure that there are no conflicts. In case of conflicts, however, a reconfiguration is typically quite simple.
16.4.4 IET600 behavior in SED Data Exchange.

IEC 61850 does not specify an SED change completely. To guarantee a safe exchange, IET600 implements some additional logic for SED Data exchange:

1. You can initiate only one export session at a time.
   IEC 61850 would theoretically allow to open a Session X with Project B and a Session Y with Project C at the same time. IET600 however, forces you to finish one session before opening another.

2. When returning foreign IEDs which you have imported at an earlier time, you always need to return all of them. IET600 does not allow you to return some and keep some others to do some more changes.

3. IET600 currently does not import the Substation Section of a foreign Substation on SED import.

4. Already existing Dataflow items of foreign IEDs cannot be edited, only new Datasets and Control Blocks can be added/edited.

5. IET600 allows to add own IEDs as clients for Control Blocks, however, you cannot change the Control Block Parameters (e.g. addresses). If you need to do so, you need to add another Control Block (which may use an existing Dataset, if it already contains the right data entries).

6. IET600 allows you, after user confirmation, to delete foreign IEDs, even though they have “fix” engineering rights.
   If such an IED has some Dataflow configured to your own IEDs, take any warning dialog very seriously! Depending on the functionality of the configured Dataflow, serious malfunction up to faulty or missing protection trips may result, if you reconfigure your own IEDs without doing the necessary re-engineering of the foreign IED which you are deleting.
   Also, once you have deleted such an IED, you have lost the information what functionality its Dataflow was providing; therefore you must note down all information relevant for the necessary re-engineering before you delete the IED.

16.4.5 Good Engineering Practices in SED Data Exchange.

During SED export, the editing of your own IEDs is restricted. Also foreign IEDs, once imported, cannot be deleted.

It is therefore strongly recommended to:

1. keep the period of SED Exchange as short as possible (e.g. do not initiate an SED exchange, if the engineer from the other project tells you he will do the changes only in two weeks from now: your engineering might be severely hindered by this.)

2. backup your database before starting the SED exchange (and preferably after each exchange as well).
16.4.6 Example Transformer Interlocking via GOOSE

Let's assume the following situation (the exact procedure will be described step by step in the following chapters 16.4.7 - 16.4.14):

- a substation, where Vendor A engineers the HV part and Vendor B the MV part.
- the customer wants both vendors to implement switch interlocking across two Transformers with GOOSE.
- the participating IEDs are IED A1 and A2 (the Control IEDs on the HV side) and IEDs B1 and B2 (the Control IEDs on the MV side).

The picture below demonstrates the workflow and shows the engineering rights in the projects after each step (only one IED per project shown):

1. Engineer A exports IED A1 and A2 with Dataflow engineering rights to Project B.
2. Engineer B imports the SED file into Project B. During import, the IEDs will be added to the Subnetwork as defined in the SED file; if the Subnetwork does not exist, it will be created.
3. Engineer B adds a Dataset/GCB to IED A1 and A2 each with the needed interlocking information. He checks whether IEDs A1 and A2 are on the same Subnetwork as IEDs B1 and B2; then adds his own IEDs B1 and B2 as clients to the new Dataset/GCBs.
4. Engineer B exports an SED file with IEDs A1 and A2 (engineering rights not important, IET600 always returns foreign IEDs with the same rights as it has received them) and IEDs B1 and B2 with Dataflow engineering rights.
5. Engineer A imports the SED file into Project A. During import, the IEDs will be added to the Subnetwork as defined in the SED file; if the Subnetwork does not exist, it will be created.
6. Engineer A adds a Dataset/GCB to IEDs B1 and B2 with the needed interlocking information. He configures IEDs B1 and B2 to the same Subnetwork as IED A and adds his own IED A as client to the new Dataset/GCB.
7. Engineer A exports an SED file with IED A and IED B (IED A is exported again to ensure that engineer B has the up-to-date version of it; it may have changed between the last SED import and this export).
8. Engineer B imports the SED file into Project B.
In the end, each project must have its own IEDs with “full” rights and foreign IEDs with “fix” rights, no IEDs in either project must be “Checked out” any more.

16.4.7 **Step 1: Export of an SED File in Project A**

Before starting the export, you should be in a situation that you can save the project after the export (e.g. no changes which you want to throw away). To close the project without saving it after an SED export will render the SED file invalid.

An SED export will initiate a “pending SED Transfer” with the receiving project. While this transfer is pending, you cannot exchange another SED file with a third project.

1. In the Main menu, select ‘Communication’ -> ‘SED Exchange’ -> ‘SED Export’
2. A dialog will open.

   As an IED can be connected via different Access Points to several Subnetworks, select the one that connects it to the Subnetwork to which you agreed during the preparation, because that Subnetwork (and only that Subnetwork) will also be exported into the SED file.

Select the IEDs/Access Points for export and configure the engineering rights that you want to give to the receiving project (“fix” if the other project is using...
your IED just as a RCB or GCB client; “dataflow”, if the other project needs to create Datasets/Control Blocks in your IED.

3. Select file path and name for export -> select ‘Export’ -> an SED file will be created:

If you have already an open SED session, you will be prevented to export another SED file and the following dialog will appear:
4. In Project A, the IEDs transferred to Project B with Dataflow rights, have now "fix" rights in Project A:

- 'Update IED' is not possible
- Dataflow changes, including adding clients, is not possible.
- Signal engineering, however, can be done as normal.

In the IED Status Editor, you can check the SED exchange in more detail by looking at the marked columns:

<table>
<thead>
<tr>
<th>Modified</th>
<th>IED Name</th>
<th>IED Type</th>
<th>Foreign</th>
<th>IED</th>
<th>RIC</th>
<th>Engineering</th>
<th>Rights</th>
<th>Preceding</th>
<th>SED Transfer</th>
<th>Checked</th>
<th>Protected</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>Full</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Full</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Full</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Full</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Full</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Full</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Full</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Full</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Full</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Full</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Full</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Full</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Full</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Full</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Full</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Full</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Full</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Full</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Full</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Full</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Full</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

[Drag a column header here to group by that column.]

In the IED Status Editor, you can check the SED exchange in more detail by looking at the marked columns:
In the IED Navigation Tree, the exported IEDs are displayed with a red lock to indicate their checked-out state:

It is recommended to save the project immediately after SED export.

If you try to close the project without saving it, a warning dialog will appear:

16.4.8 Step 2: Import of SED File into Project B

Before importing an SED file, it is strongly recommended to back up your project. After importing foreign IEDs and saving the project, it is not possible to remove those foreign IEDs anymore; the only way to do so is to restore a backup from before the import.

You should be in a situation that you can close the project after the import with or without saving.

An SED import will initiate a “pending SED Transfer” with project A. While this transfer is pending, you cannot exchange an SED file with any other project.

1. In the Main menu, select ‘Communication’ -> ‘SED Exchange’ -> ‘SED Import’

2. A standard ‘File open’ dialog appears. Select the SED file which you want to import -> the SED file will be imported.

If you already have previously imported some IEDs, you will not be able to import the SED file, the following warning dialog will appear:

You will need to return those IEDs first (see chapter 16.4.10, “Step 4: Re-Export Foreign IEDs and own IEDs from Project B”), before you can import another SED file.

3. In Project B, the imported IEDs transferred from Project A appear in the IED Status Editor as foreign IEDs and have either “dataflow” rights (those you need to
edit) or “fix” rights (those that have been transferred to resolve references in the SED file):

In the Navigation Trees, the foreign IEDs appear dimmed and are displayed with a blue lock:

Again, not only the two IEDs selected before, but also additional IEDs (those that were configured as RCB Clients of any RCBs of the selected IEDs) can be seen. However, the Navigation Tree does not show any difference between IEDs with “dataflow” or “fix” rights.

After SED import, check that you received the IEDs you need for the agreed-upon configuration. Also check the Subnetwork configuration and ensure that it matches your expectations.

If all your expectation are met, save the project now. If not, then exit the project without saving and request a correct SED file from the project A. If you save the Project, you will not be able to delete any foreign IEDs anymore.

### 16.4.9 Step 3: Add Dataflow to Foreign IEDs in Project B

Engineer B can now add Datasets and Control Blocks as required to the imported IEDs and add clients to them. This is not different from adding Datasets/Control Blocks to own IEDs as described in chapter 10 (“Dataflow Engineering”).
Step 4: Re-Export Foreign IEDs and own IEDs from Project B

Again, before starting the export, you should be in a situation that you can save the project after the export (e.g. no changes which you want to throw away). To close the project without saving it after an SED export will render the SED file invalid.

1. In the Main menu, select ‘Communication’ -> ‘SED Exchange’ -> ‘SED Export’

2. A dialog will open.

Those foreign IEDs which have a “Pending SED Transfer” and “dataflow” rights will automatically be selected for Export; they cannot be unselected.

If required, select some own IEDs/Access Points for export and configure the engineering rights that you want to give to the receiving project (“fix” if the other project is using your IED just as a RCB or GCB client; “dataflow”, if the other project needs to create Datasets/Control Blocks in your IED.

3. Select file path and name for export -> select ‘Export’ -> an SED file will be created.
### 16.4.11 Step 5: Import of SED File into Project A

You should be in a situation that you can close the project after the import with or without saving.

1. In the Main menu, select ‘Communication’ -> ‘SED Exchange’ -> ‘SED Import’
2. A standard ‘File open’ dialog appears. Select the SED file which you want to import - the SED file will be imported.
3. In Project A, the re-imported own IEDs will have engineering rights “full”, they are not checked out anymore.

The newly imported foreign IEDs from Project B have either “dataflow” rights (those you need to edit) or “fix” rights (those that have been transferred to resolve references in the SED file):

![IED dataflow rights](image)

Check carefully whether the imported IEDs match your expectations. If so, save the project now. If not, then exit the project without saving. If you save the project, you will not be able to delete any foreign IEDs anymore.

### 16.4.12 Step 6: Add Dataflow to Foreign IEDs in Project A

Engineer A can now add Datasets and Control Blocks as required to the imported IEDs and add clients to them.

This is not different from adding Datasets/Control Blocks to own IEDs as described in chapter 10 (Dataflow Engineering).
16.4.13 Step 7: Re-export Foreign IEDs from Project A

Again, before starting the export, you should be in a situation that you can save the project after the export (e.g. no changes which you want to throw away). To close the project without saving it after an SED export will render the SED file invalid.

1. In the Main menu, select ‘Communication’ -> ‘SED Exchange’ -> ‘SED Export’
2. A dialog will open.
   Those foreign IEDs which have a “Pending SED Transfer” and “dataflow” rights will automatically be selected for Export; they cannot be unselected
   Your own IEDs A1 and A2 which you exported to project B earlier in step 1, should have their final configuration. However, if you have done some changes of their DATA model, you might want to pass them to project B again by selecting them and set the engineering rights to “fix” for this export.
   (The figure below shows only the IEDs to be exported, the other IEDs are filtered out)

3. Select file path and name for export -> select ‘Export’ -> an SED file will be created:
Chapter 16

4. After this export, check in the IED Status editor.
   No IEDs should have “Pending SED transfer” or be “Checked Out”.
   All your own IEDs should have “full” IEC 61850 engineering rights.
   All foreign IEDs should have “Foreign IEDs” and “Protected” checked and IEC 61850 engineering rights “fix”.

   If this is correct, the SED exchange is finished as far as project A is concerned.

16.4.14 Step 8: Re-Import of SED File into Project B

You should be in a situation that you can close the project after the import with or without saving.

1. In the Main menu, select ‘Communication’ -> ‘SED Exchange’ -> ‘SED Import’

2. A standard ‘File open’ dialog appears. Select the SED file which you want to import -> the SED file will be imported.

3. After this re-import, check in the IED Status editor:
   No IEDs should have “Pending SED transfer” or “Checked Out” checked anymore.
   All your own IEDs should have “full” IEC 61850 engineering rights.
   All foreign IEDs should have “Foreign IEDs” and “Protected” checked and IEC 61850 engineering rights “fix”.

If this is correct, the SED exchange is finished as far as project A is concerned.
No additional foreign IEDs should appear anymore.

Check carefully whether the imported IEDs match these expectations. If so, save the project now, the SED exchange is finished as far as project B is concerned. If not, then exit the project without saving.
16.4.15 IET600 SED File Content

IET600 tries to keep the amount of data exchanged via SED file as small as possible:

- a project should not be cluttered with unnecessary foreign IEDs, but should contain only IEDs relevant for the desired communication.
- Communication data are IT-security-relevant, therefore only the needed communication data should be exchanged

However, IEC 61850 also demands that all references in an SED file are resolved, i.e. an IED must e.g. contain IEDs that appear as Control Block clients.

To fulfill both above-mentioned requirements, an SED file exported by IET600 contains the following items:

- selected IEDs with their full DATA, Dataset and Control Block configuration, except that GCB clients and SVCB clients will be removed (unless the client IEDs are also selected to appear in the SED file).
- all RCB clients referenced by any of the above IEDs (they cannot be removed, because the sequence of RCB clients is relevant and needed, if the receiving project wants to add some of its own IEDs as RCB clients); however, their DATA model will be stripped down to an absolute minimum.
- all Subnetworks, to which the selected IEDs/Access Points are connected.
- all IP Addresses for the selected IEDs/Access Points connected to those Subnetwork.
- GCB and SVCB address information for GCBs/SVCBs connected to this Subnetwork (required for a valid and consistent configuration of additional clients for these Control Blocks).
- a Substation section with all Bay and their respective Equipment, if the IED belongs to this Bay.
16.5 HMI Status Texts Exchange

16.5.1 Format of Status Text Files

Status Texts can be exchanged with other tools via text files in ‘csv’-format. Translations into different languages are kept in different files, so each file contains texts of only one single language.

The file name must have the format [filename].[language code].csv with the following specifications:

[filename] can currently be freely chosen; it is recommended to use “StatusTexts” to distinguish it from other files following a similar naming pattern (e.g. “SignalTexts”).

[language code] must be a two-letter language code according to ISO 639-1. The tool will recognize capital letters “EN” instead of “en” although it is not according to ISO 639-1.

“csv” as extension.

One line of the file must contain the following information:

1. A key to identify a set of Status Texts.
2. The type of the signal (Ai for “analogue input”, AO for analogue output etc.)
3. A set of process values. They can be specified in two different ways:
   • a plain number represents a set of continuous values from 0 to the number minus one, e.g. a double binary value will have a “4”, representing the consecutive values 0, 1, 2 and 3
   • a set of numbers in vector format to represent an irregular sequence of numbers, e.g. “(0, 5, 6, 10, 100)”
4. A sequence of status texts, their count must match the count of the process values (e.g. if the process values are specified by a number “4”, there must be 4 status texts “intermediate”, “on”, “off”, “faulty”).

Items 1, 2, 3, and each text under 4 must be separated by a “;”. The line must not be terminated with a “;”.

Texts must not start or end with delimiters such as “ “ or ‘’ (such characters will be interpreted as belonging to the texts, not as delimiters)

The file must also contain a header with defined information. Any file not following the above specifications cannot be imported.

A default file with English status texts, as well as the command procedures scripts to import and export these texts into MicroSCADA, are provided with the IET600 baseline installation. These files can be found at:

C:\Program Files\ABB\IET 5.x.x\SADesigner\ApplicationTemplates\HMI\StatusTexts.
16.5.2 Import Status Texts

Status Texts can only be imported from the Main menu:

1. Select the 'HMI' tab in the Main menu.
2. Click 'Status Texts' in the 'Import' group -> the 'Import Status Texts' dialog appears (the default path is the subdirectory “Import” in your local project path).
3. Select one or more Status Text files (typically named “StatusText.[language code].csv”.
   The part [language code].csv is mandatory; a file with a different name cannot be imported.
   You can select several files in different languages to import Status Texts in several languages at the same time.
4. Click 'Open' -> the files are imported.

If the import succeeds, you will not get any feedback directly. If something goes wrong, an error message will appear.

If you have HMI signals without Status Texts but with valid Status Text keys (e.g. EH attribute for MicroSCADA signals), Status Texts will be added in the HMI Signal Editor immediately after import.

A Re-Import of a Status Texts file will add new Status Texts and will overwrite Status Texts that have changed, but it will not delete existing Status Texts that may not exist in the imported file.

16.5.3 Export Status Texts

Status Texts can only be exported from the Main menu:

1. Select the 'HMI' tab in the Main menu.
2. Click 'Status Texts' in the 'Export' group -> an 'Export Status Texts' dialog appears (the default export path is the subdirectory “Export” in your local project path).
3. Choose a basic name (“StatusTexts” is recommended and may become mandatory at some point in the future).
   If you have status texts in different languages, they will be exported as “StatusTexts.en.csv”, “StatusTexts.de.csv”, “StatusTexts.fi.csv” etc. Files with those names in the export directory will be overwritten,
4. Click 'Save' -> the files are exported.

If the export succeeds, you will not get any feedback directly. If something goes wrong, an error message will appear.
16.6 HMI Signal Information Exchange

16.6.1 Overview

There is no standard describing how to exchange Signal Texts and other HMI-related information via SCD file. So the tendency is to exchange these data in additional files, with references to IEC-61850 objects.

Currently, HMI information can only be exchanged with MicroSCADA in specific formats:

- ‘lof’ file
- ‘sfx’ file
- ‘csv’ file in a MicroSCADA-specific format.
- ‘sasmsc’ file (available for MicroSCADA 9.4, FP1 and higher)

For information about additionally exchanging Gateway/NCC information, please refer to chapter 16.7 (NCC Gateway Signal Information Exchange).

16.6.2 Exchange File Formats

‘lof’ stands for Loadfile. It is a MicroSCADA-specific format and contains information for the MicroSCADA Process Object Database. The data are saved in a (badly) human-readable text file. It is currently the default format to export HMI Data from IET600 to MicroSCADA. Its usage is recommended in the following situations:

- In a retrofit project, if you do not have an IET600 project, but only a running MicroSCADA 8.x (or an up-to-date ‘lof’ file) with or without an SCD. This ‘lof’ file can be used to import the existing signals from MicroSCADA into IET600. Before the import, you should have configured your IEDs and done the LN mapping (this will be used by IET600 to automatically map signals on import).
- To exchange data with MicroSCADA 8.x. In this case you are sure to have valid IN attributes for the mapping of the data.

‘sfx’ stands for Safe Exchange. The format is basically the same as for ‘lof’ files, but an ‘sfx’ file contains only attributes that are considered “safe” (see chapter 18.2.7, HMI Engineering Protection, for more detailed information). Its usage is recommended in the following situations:

- In late project phases, e.g. commissioning, to be able to respond to customer requirements like changed texts, event listings etc. while ensuring that no communication-relevant configuration is changed.

‘csv’ is a generic format; however, MicroSCADA exports signals in a hierarchical file structure with one file being the master file containing the file names of the other files, and additional ‘csv’ files – one per signal type – containing the actual signal information. Its usage is recommended in the following situations:
In a retrofit project, if you do not have an IET600 project, but only a running MicroSCADA 9.x (or up-to-date ‘csv’ files) with or without an SCD. These ‘csv’ files can be used to import the existing signals from MicroSCADA into IET600. Before the import, you should have configured your IEDs and done the LN mapping (this will be used by IET600 to automatically map signals on import).

To import HMI Data from MicroSCADA 9.x into IET600. In this case you are sure to have valid IN attributes for the mapping of the data.

‘sasmsc’ stands for SAS MicroSCADA. It is a format to export the complete data required to configure OPC Servers, the MicroSCADA Process Object Database, OPC DA Clients and NCC signals for one MicroSCADA-based HMI or Gateway within one file.

The ‘sasmsc’ file is a ZIP-format file which contains internally the following files:

- A System Identifier file which contains basic information about the System to be configured and the content of the ‘sasmsc’ file.
- optionally an SCD file (required for OPC Server configuration).
- ‘lof’ file for one HMI (required for MicroSCADA Process Object and OPC DA Client configuration).
- ‘xref’ files for one Gateway (if it is a MicroSCADA with Gateway functionality).
- Status values Text files for the ABB Standard and customer languages.
- Additional files providing information for OPC Server configuration (e.g. Disturbance Recording).

Its usage is recommended in the following situations:

To exchange data with MicroSCADA 9.4 FP1 and higher.

IET600 currently allows only the export of ‘sasmsc’ files. To transfer data from MicroSCADA to IET600, another of the above-mentioned formats needs to be used.

### 16.6.3 Import Behavior of IET600

On import, the user must always specify the Target HMIs, to which these files with HMI information belong.

IET600 will use the MicroSCADA IN attribute (which contains typically the IED name) to attach the imported signals to IEDs and then map them to the Target IED.

If an IN does not exist or if there is no IED with the same name as in the IN reference, the signals are attached to the target HMI directly (this is typically the case for so-called “internal” signals). The engineer may need to shift them from the target HMI to the proper IED manually.

The MicroSCADA Database file import can use files of type ‘lof’ (with or without ‘xref’ files), ‘sfx’, ‘csv’ or ‘sasmsc’ (with or without ‘xref’ and SCD files)

‘lof’ files and ‘csv’ files will always update existing signals and add new signals; no signals will be removed. If a Gateway/NCC configuration exists, ‘xref’ files should always be imported together with the corresponding ‘lof’ files to ensure consistency.
‘sfx’ files contain only attributes that are considered “safe” (see chapter 18.2.7, HMI Engineering Protection, for detailed information about what is considered to be safe in this context). An ‘sfx’ import will only update existing signals, it will neither add nor remove any signals.

‘sasmsc’ files will act like a ‘lof’ file import; additionally, the Disturbance Recording information will be updated from information contained in the ‘sasmsc’ file. Any SCD file contained in the ‘sasmsc’ file is disregarded, as the danger of interference with the current configuration is too high (accordingly, it is the responsibility of the engineer to ensure that the IET600 SCD DATA model is matching the one in the SCD file).

16.6.4 Import Considerations

During normal engineering, HMI Data engineering should always be done in IET600 and exported to MicroSCADA.

However, there are certain specific situations where a MicroSCADA file import is necessary:

At the start of a project, if you do not have an IET600 project, but only a running MicroSCADA 9.x with or without an SCD. Before the import, you should have configured your IEDs and done the LN mapping (this will be used by IET600 to automatically map signals on import).

To exchange data with MicroSCADA 8.x. In this case you are sure to have valid IN attributes for the mapping of the data.

Late in project engineering, to transfer changes of ‘safe’ attributes which have been made directly in MicroSCADA back to IET600.

16.6.5 Import a MicroSCADA ‘lof’ File

A MicroSCADA Database file import can only be initiated from the Main menu.

1. Select the ‘HMI’ menu tab -> the ‘HMIs’ Navigation Tree is selected automatically.

2. Click ‘MicroSCADA Database File(s)’ in the ‘Import’ group of the Main menu.
3. The 'Import MicroSCADA Database File(s)' dialog appears. Click on the file selector button to the right to choose a file for import:

![Import MicroSCADA Database File(s) dialog](image1)

4. A ‘File Open’ dialog appears.

5. Select a ‘lof’ file.

![File Open dialog](image2)

6. Select the HMI into which the ‘lof’ file will be imported.

![Select HMI](image3)
In case of a ‘lof’ import, you can also decide to import ‘Only Safe Attributes’ to simulate an ‘sfx’ import. This can be used in connection with MicroSCADA versions older than 9.3 which cannot import/export the ‘sfx’ format yet.

The Tool Tip of this check box will show you which attributes are considered safe.

7. Click ‘Preferences’ button to define MicroSCADA comparison preferences (see chapter 16.6.10 for more details).

8. To preview the signals to be imported, click the ‘Signal Overview’ button:

Signals which are different will be marked; selecting such a signal will display the differences (see chapter 16.6.9 for more details).

9. Click ‘Import/Update Signals’ to start the file import (only enabled in the ‘Signal Overview’).

10. The dialog will inform you how many signals have been imported to each HMI.

11. Repeat 3, 4, 5 and 6 for more imports from other files, if needed.

12. Click ‘Close Dialog’ to close the dialog.

16.6.6 Import a MicroSCADA ‘sfx’ File

A MicroSCADA Database file import can only be initiated from the Main menu.
Chapter 16

1. Select the 'HMI' menu tab - the 'HMIs' Navigation Tree is selected automatically.

2. Click 'MicroSCADA Database File(s)' in the 'Import' group of the Main menu.

3. The 'Import MicroSCADA Database File(s)' dialog appears. Click on the file selector button to the right to choose a file for import:

4. A 'File Open' dialog appears.

5. Select an 'sfx' file.
6. Select the HMI into which the 'sfx' file will be imported.

In this case, the 'Only Safe Attributes' is mandatorily checked. The Tool Tip of 'Only Safe Attributes' check box will show you which attributes are considered safe.

The 'Import XRF Files is mandatorily unchecked, as anything related to XRF files is not considered a 'safe' import.

7. Click 'Preferences' button to define MicroSCADA comparison preferences (see chapter 16.6.10 for more details).
8. To import the signals, or to preview the signals to be imported, click the ‘Signal Overview’ button:

Signals which are different will be marked; selecting such a signal will display the differences (see chapter 16.6.9 for more details).

9. Click ‘Import/Update Signals’ to start the file import (only enabled in the ‘Signal Overview’).

10. The dialog will inform you how many signals have been imported to each HMI.

11. Repeat 3, 4, 5 and 6 for more imports from other files, if needed.

12. Click ‘Close Dialog’ to close the dialog.

16.6.7 Import a MicroSCADA ‘csv’ File with Process Object Data

The import of MicroSCADA ‘csv’ files is limited to those containing the configuration of the MicroSCADA Process Object Database.

A MicroSCADA Database file import can only be initiated from the Main menu.

1. Select the ‘HMI’ menu tab -> the ‘HMI’ Navigation Tree is selected automatically.

2. Click ‘MicroSCADA Database File(s)’ in the ‘Import’ group of the Main menu.
3. The 'Import MicroSCADA Database File(s)' dialog appears. Click on the file selector button to the right to choose a file for import:

Select ‘csv’ file, then select the ‘csv’ Master file, ("TEST.CSV" in this example, the dependent files must be in the same folder):
To successfully import such 'csv' files into IET600, ensure that the following settings are used to export the 'csv' files from MicroSCADA:

To ensure consistency, make sure that all attributes are selected for export (configurable under the Menu of this dialog -> 'csv' attributes -> if you have any other configuration, a 'Reset' should achieve the desired result).

The Attribute Name Header is mandatory for data exchange between MicroSCADA and IET600. A comma as delimiter is not supported. OI split is not supported.

4. Select the HMI into which the 'csv' file will be imported.

In case of a 'csv' import, you can also decide to import 'Only Safe Attributes' to simulate an 'sfx' import. This can be used in connection with MicroSCADA versions older than 9.3 which cannot import/export the 'sfx' format yet. (see chapter 18.2.7, HMI Engineering Protection for detailed information about what is considered to be safe in this context).
5. To preview the signals to be imported, click the ‘Signal Overview’ button:

Signals which are different will be marked; selecting such a signal will display the differences (see chapter 16.6.9 for more details).

6. Click ‘Import/Update Signals’ to start the file import (only enabled in the ‘Signal Overview’).

7. The dialog will inform you how many signals have been imported to each HMI.

8. Repeat 3, 4, 5 and 6 for more imports from other files, if needed.

9. Click ‘Close Dialog’ to close the dialog.

16.6.8 Import a MicroSCADA ‘sasmsc’ File

A ‘sasmsc’ file may contain an SCD file. However, the ‘sasmsc’ import will disregard this SCD file because MicroSCADA should not modify it in any way. If you need the information in this file, you need to extract the SCD file from the ‘sasmsc file and process it manually.

A MicroSCADA Database file import can only be initiated from the Main menu.

1. Select the ‘HMI’ menu tab -> the ‘HMIs’ Navigation Tree is selected automatically.

2. Click ‘MicroSCADA Database File(s)’ in the ‘Import’ group of the Main menu.
3. The 'Import MicroSCADA Database File(s)' dialog appears. Click on the file selector button to the right to choose a file for import:

![Import MicroSCADA Database File(s) dialog](image1.png)

4. A 'File Open' dialog appears.

5. Select an 'sasmsc' file:

![Select an sasmsc file](image2.png)

6. Select the HMI into which the 'sasmsc' file will be imported.

![Select the HMI](image3.png)

In this case, the 'Only Safe Attributes' is mandatorily unchecked.
7. Click ‘Preferences’ button to define MicroSCADA comparison preferences (see chapter 16.6.10 for more details).

8. To preview the signals to be imported, click the ‘Signal Overview’ button:

 Signals which are different will be marked; selecting such a signal will display the differences (see chapter 16.6.9 for more details).

9. Click ‘Import/Update Signals’ to start the file import (only enabled in the ‘Signal Overview’).

10. The dialog will inform you how many signals have been imported to each HMI.

11. Repeat 3, 4, 5 and 6 for more imports from other files, if needed.

12. Click ‘Close Dialog’ to close the dialog.

16.6.9 Import Signal Overview

Signal Overview shows whether imported signal and signal in project are equal. Every row in a list of signals can have in several states:

- signals do not have any differences (‘Comparison’ column has green background)
- imported signal does not exist in a project (‘Comparison’ column has gray background)
Chapter 16

- signals have some differences ("Comparison" column has red background)
- signals have some differences, but all different attributes are excluded from comparison ("Comparison" column has yellow background. "Import/Update" check box is in undefined state. No data is presented in Difference view)
- signal attributes are different, but some of them are excluded from comparison ("Comparison" column has red background. In Difference view presented only attributes which are included into comparison)

Attributes excluded from comparison are not imported/updated into a project.

For multi selected rows Difference View shows green cells for attributes without difference and red for different ones:

Also there is a possibility to create Excel document by exporting Difference View from context menu:
16.6.10 MicroSCADA Comparison Preferences dialog

A dialog gives a possibility to choose which MicroSCADA attributes include to comparison.

User can check/uncheck ‘Include Addresses’ to select address attributes at once and define own set of attributes to compare.

After pressing ‘Close Dialog’ button, Signal Overview list is automatically regenerated according to current preferences.

16.6.11 Export a MicroSCADA ‘Iof’ File

‘Iof’ files can only be exported from the Main menu.

1. Select the ‘HMI’ menu tab -> the ‘HMIs’ Navigation Tree is selected automatically.

2. Click ‘MicroSCADA Loadfile(s)’ in the ‘Export’ group of the Main menu.
3. The 'Export MicroSCADA DB Loadfile' dialog appears:

4. Select the HMIs whose HMI signals you want exported. For each stand-alone or each Hot Standby System, one 'lof' file will be created.

For each file, the last known export folder and file name are preselected (the default is \Projects Root Folder\<Project Name>\Export\MicroSCADA\<HMI Name>). To change an individual path and/or name click on the button in the 'Edit Filepath' Column in the same row. To rapidly change several folder to the same value, change one, then right-click into the changed cell and select 'Apply Export Folder Path to all/selected HMIs' from the cell context menu.

5. Check 'lof/xref' as File Format.

Check 'ABB Standard' only if you want the OX attribute to contain the internal rather than the customer signal text. This is mainly relevant when configuring mirroring between two projects (see chapters 13.5.5 and 13.5.6).

Check 'Backup old Loadfile' if you want IET600 to create backups of older files with the same name.

(If the MicroSCADA is a Gateway and has NCC connections, XREF files are exported automatically to the same folder as the MicroSCADA ‘lof’ files)

6. Before the file is actually exported, a consistency check is performed and the result is presented in a dialog:

Errors will probably make an import into MicroSCADA impossible; you should not continue the export, but correct the errors first.
Warnings are acceptable in earlier phases of engineering (if e.g. the engineering is not completed yet, but a MicroSCADA Process Object Database is already needed for picture engineering), but should be rectified before testing.

7. The dialog will stay open to allow you to export additional HMI configurations. After finishing all exports, close the dialog.

16.6.12 Export a MicroSCADA ‘sfx’ File

SFX files can only be exported from the Main menu.

1. Select the ‘HMI’ menu tab -> the ‘HMIs’ Navigation Tree is selected automatically.
2. Click ‘MicroSCADA Loadfile(s)’ in the ‘Export’ group of the Main menu.

3. The ‘Export MicroSCADA DB Loadfile’ dialog appears:

4. Select the HMIs whose HMI signals you want exported. For each stand-alone or each Hot Standby System, one ‘sfx’ file will be created.

   For each file, the last known export folder and file name are preselected (the default is `<Projects Root Folder>\<Project Name>\Export\MicroSCADA\<HMI Name>`). To change an individual path and/or name click on the button in the ‘Edit Filepath’ Column in the same row. To rapidly change several folder to the same value, change one, then right-click into the changed cell and select ‘Apply Export Folder Path to all/selected HMIs’ from the cell context menu.

5. Check ‘sfx’ as File Format.

   Check ‘ABB Standard’ only if you want the OX attribute to contain the internal rather than the customer signal text. This is mainly relevant when configuring mirroring between two projects (see chapters 13.5.5 and 13.5.6).

   Check ‘Backup old Loadfile’ if you want IET600 to create backups of older files with the same name.

6. Before the file is actually exported, a consistency check is performed and the result is presented in a dialog:
Chapter 16

16.6.13 Export a MicroSCADA ‘csv’ file with Process Object Data

This is not possible. To transfer data from IET600 to MicroSCADA, only ‘lof’ files (or ‘sasmsc’ files which include the ‘lof’ files) are available.

16.6.14 Export a MicroSCADA ‘sasmsc’ file

Caution: When exporting an ‘sasmsc’ file, a new SCD file is generated within the ‘sasmsc’ file. If this does not match the SCD file exported separately and passed to the IED Configuration tool(s), this can result in communication problems which may be difficult to detect!

When importing the file into MicroSCADA, the IET600 customer name is checked against the computer name of the Target System -> as a prerequisite to the export, please provide the customer name for those MicroSCADA System.

‘sasmsc’ files can only be exported from the Main menu.

1. Select the ‘HMI’ menu tab -> the ‘HMIs’ Navigation Tree is selected automatically.

2. Click ‘MicroSCADA Loadfile(s)’ in the ‘Export’ group of the Main menu.
3. The ‘Export MicroSCADA DB Loadfile’ dialog appears:

4. Select the HMIs whose HMI signals you want exported. For each stand-alone or each Hot Standby System, one ‘sasmsc’ file will be created.

   For each file, the last known export folder and file name are preselected (the default is \Projects Root Folder\ <Project Name>\Export\MicroSCADA\<HMI Name>). To change an individual path and/or name click on the button in the ‘Edit Filepath’ Column in the same row. To rapidly change several folder to the same value, change one, then right-click into the changed cell and select ‘Apply Export Folder Path to all/selected HMIs’ from the cell context menu.

5. Check ‘sasmsc with SCD’ as File Format, if you want to configure OPC Servers as well; this will be the case in early project stages, and if the IEC-61850-Dataflow is changed.

   Alternatively, check ‘sasmsc without SCD’ as File Format, if you want to leave the OPC Servers untouched. This is e.g. recommended if you are configuring an NCC connection and want to make sure that the local configuration between bay-level IEDs and OPC servers remains untouched. OPC DA Client configuration files may be recalculated, depending on changes in the Process Object Database.

   Check ‘ABB Standard’ only if you want the OX attribute to contain the internal rather than the customer signal text. This is mainly relevant when configuring mirroring between two projects (see chapters 13.5.5 and 13.5.6).

   Check ‘Backup old Loadfile’ if you want IET600 to create backups of older files with the same name.

6. Before the file is actually exported, a consistency check is performed and the result is presented in a dialog:
Errors will probably make an import into the OPC Server Configurator or MicroSCADA impossible; you should not continue the export, but correct the errors first.

Warnings are acceptable in earlier phases of engineering (if e.g. the engineering is not completed yet, but a preliminary OPC Server configuration and/or a MicroSCADA Process Object Database is already needed for picture engineering), but should be rectified before testing.

7. The dialog will stay open to allow you to export additional HMI configurations. After finishing all exports, close the dialog.
16.7 NCC Gateway Signal Information Exchange

16.7.1 Overview

Configuration exchange with the NCC Gateway is only supported for MicroSCADA COM500 or SYS600C gateways. The following files are exchanged between IET600 and the COM500 configuration tool:

- COM_XRIND.XRF
- COM_XCMD1.TXT

The above files are imported and exported together with the MicroSCADA Loadfile(s), as described in the previous section.

Note that while the Loadfiles can have different names per HMI, the name of these cross-reference files are fixed and cannot be changed. Therefore you cannot export several HMIs into one folder, as the cross-reference files would overwrite each other.

16.7.2 Import COM500 Configuration File (XRF)

The COM500 configuration files are always imported with the corresponding MicroSCADA 'lof' file.

An important task to be done before importing the COM500 files is to prepare the NCCs in the 'HMI's Navigation Tree. The NCCs shall be defined in the order of the NCC line numbers in the COM500 file. Furthermore, the UN numbers of the NCCs must also be specified according to the file. IET600 will use these two references to correctly map the COM500 configuration to the correct NCC during the import.

The COM500 files must be stored in the same folder as the MicroSCADA 'lof' file. So e.g. to import this configuration into an HMI, the following files need to be present in the folder:

- AA2KF14.LOF
- COM_XRIND.XRF
- COM_XCMD1.TXT

or

- AA2KF14.CSV ('csv' master file)
- AA2KF14_IX_BI.CSV
- ... (more 'csv' files)
- COM_XRIND.XRF
- COM_XCMD1.TXT

To import the MicroSCADA 'lof' files, follow the steps described in chapter 16.6.5 (Import a MicroSCADA 'lof' File). In the import dialog, select the check box 'Import XRF files'.
Note that this checkbox will only appear, if the two XRF files are found in the same folder as the selected ‘lof’ or ‘csv’ master file:

16.7.3 Export COM500 Configuration File (XRF)

To export the COM500 signal configuration files, select ‘Export MicroSCADA Load-file(s)’ from the Main menu, ‘HMI’ tab (as described in the chapter 16.6.11, “Export a MicroSCADA ‘lof’ File”). The COM500 files will be exported automatically to the same folder as the MicroSCADA Loadfiles.

Alternatively, for MicroSCADA 9.4 FP1 and newer, you can export an ‘.sasmsc’ file (as described in chapter 16.6.14, “Export a MicroSCADA ‘sasmsc’ file”). If the MicroSCADA is a Gateway, the XREF files will automatically be included in the ‘.sasmsc’ file.

16.7.4 Export Configuration Data for Other NCC Gateways

Currently, only the SCD file is available for other NCC Gateways. No specific signal configuration files are available for other Gateways at this time.
17 Customer Signal Lists

17.1 Introduction

In any project, signal lists of some form need to be exported for discussion with the customer and as documentation of a project. Unfortunately, each customer has his/her own idea of how these lists should look like. IET600 allows you to customize such lists to a considerable extent to accommodate customer’s requirements.

- IET600 provides a Default Template as a starting point with columns for all possible attributes that can be exported.
- Either modify this default template, or – better – create a copy of this default template, give it a sensible name and to the changes to this renamed template.
- You can customize a template by
  - removing columns as suitable to adapt the exported attributes to customer requirements.
  - defining grouping, filtering and sorting criteria to filter, group and sort the rows according to customer criteria.
  - saving one or several such configurations as customized templates in your project.
- You can export your data based on those templates and additionally customize the exported workbook by adding formatting, e.g. by coloring group headers, changing the font and making it bold etc.
- You can re-export your project data into the same Excel Workbook without losing such customized formatting.
- You can ask the customer to make changes in those Excel Workbooks and import those changes into IET600.

17.2 Customizing a Template

Before customizing a Template, please be aware of the following:

To allow a re-import of such customer lists into IET600, each column must be defined as a ‘Named Range’ in Excel with a specific name. Therefore:
You can change the text or colors of the headers, rearrange columns etc. Export and re-import will work correctly, as long as the ‘Named Ranges’ are defined correctly.

If you have accidentally removed a column which is designed for re-import, do not try to repair this by copy/pasting an existing column and adjusting its header; but rather start with the original template again.

You can add additional columns with information which is not expected to be re-imported into IET600, e.g. engineering/customer comments etc. Such columns can also be removed later. To avoid confusion, it is recommended to mark these columns in a way which clearly distinguishes them from the columns with ‘Named Ranges’.

### 17.2.1 Copy the Default Template into Your Project Folder

1. The Default Template provided by IET600 is located in the Program Directory of IET600 (in Windows 7 typically “C:\Program Files (x86)\ABB\IET600 5.3.xxx\SADesigner\ApplicationTemplates\Reports\TemplateSignalReport.xlsm”)

   Copy this Default Template (TemplateSignalReport.xlsm) into the ‘Reports’ folder in your project, i.e. “<Project Folder>\ ApplicationTemplates\Reports\TemplateSignalReport.xlsm”.

   (The fastest way of doing this is to open the ‘Export Customer Signal List’ dialog once, this will automatically copy the template into your project folder.)

2. If you intend to use more than one template, it is recommended to rename this template into something more meaningful, e.g. “<Project> Report - HMI Signal List.xlsm”

### 17.2.2 Customize the Template

1. Open your Project Default Template (as prepared in chapter 17.2.1 above)

2. Configure the ‘Title’ worksheet with general information about your project.

   As this general information will probably be the same for all Customer Signal Lists for one particular customer, you might want to save it now into TemplateSignalReport.xlsm before deriving additional customized Templates from it.

3. Remove those columns you do not want in your customized Template.

4. Rearrange the remaining columns according to your needs (by using cut/paste for columns).

   Caution: Excel does not allow to cut/paste columns, if they contain horizontally merged cells (e.g. the first header row in the default template). In such cases, you need to “unmerge” the cells, move the columns and the “remerge” the cells as desired.
5. Customize the column headers according to your needs (color, font, header text etc.)

6. Save the customized Template under a sensible name, e.g. “Signal Text - Event List - Alarm List Configuration” into <Project Folder>\ ApplicationTemplates\Reports.

Repeat the above for as many customized lists you want. E.g. you might want to configure Templates for the following Signal Lists:

- a Signal List with all MicroSCADA attributes for MicroSCADA documentation.
- a Signal List with attributes that the customer might want to edit (e.g. Signal Texts, Alarm and Event Listing, Printout).
- an NCC Signal List with communication information for each NCC.

Of course, you can create customized Templates also at any time later in your project.

17.3 Customizing Filtering, Grouping, Sorting and Formatting

The configuration of filters, groups and sorting are saved into the particular template you use for creating your signal list.

The configuration of these items is done in the ‘Export Customer Signal List’ dialog. While you can create the configuration without exporting an actual Signal List, you will probably want to see the result of your configuration by trying an export immediately; this is the reason why those two functionalities have been combined in the same dialog.

17.3.1 Open the ‘Export Customer Signal List’ Dialog

This dialog can only be opened from the Main menu.

1. Select the ‘HMI’ menu tab.
2. Click ‘Customer Signal List’ in the ‘Export’ group of the Main menu.
3. The 'Export Customer Signal List' dialog appears:

4. As default template, the “TemplateSignalReport.xlsm” is selected. Select the template which you want to use / configure. You can select either Excel Workbook (.xlsx) or Excel Macro-Enabled Workbook (.xlsm).

5. At any time, if you want to save the template configuration, use “Save Rules”. Any changes in filtering, grouping, sorting or formatting will be written into the selected template.
If you want to remove all changes since opening the template, use “Reset Rules”:

17.3.2 Configure Filtering

Currently, the Signal Groups and the MicroSCADA attributes can be used to create filter conditions.

After opening the ‘Export Customer Signal List’ as described in chapter 0, add filters as required:

1. Select the Template whose filters you want to add or edit. Caution: initially always the default template is selected.
2. Make sure the ‘Filtering’ tab is selected (default).
3. Right-click into the empty space in the ‘Filtering’ Tab -> a row context menu appears:


5. Right-click into the row header in any existing filter condition row and use ‘Insert New Row’ to add filter conditions as required.
The conditions within one filter condition row will be AND-combined, the conditions of several rows will be OR-combined. The tool tip of any cell will show the result of the whole filter group configuration:

6. Right-click into the row header in any existing filter condition row and use ‘Delete Row(s)’ to remove filter conditions as required.

Removing the last condition of a group will also remove the group itself.

7. You can add more filter groups, if you want. However, only one filter group at a time can be active.

This can be used, if you want to export signals according to different criteria into different lists, but based on the same template. In the above example, you could create a second filter condition for all alarm signals and export it from the same template into a different Customer Signal List.

8. You can move individual rows up and down, this is purely for convenience and does not affect the filtering logic. Right-click into the row header in any existing filter condition row and use ‘Move Row(s) Up’ or ‘Move Rows(s) Down’ to remove filter conditions as required.

9. If you want to save your current configuration before continuing, click on ‘Save Rules’. If you want to reset the rules to their original state after loading the template, click on ‘Reset Rules’.
17.3.3 Configure Grouping

‘Grouping’ in IET600 has a double function:

- It brings the signals selected for a group to the beginning of the Customer Signal List, before other signals. This is typically used to adapt to customer-specific requirements, e.g. a customer may want to see all measurements or all switch positions on top of the list.

- It inserts ‘Grouping’ Headers before each group. This is typically used for additional text and formatting.

Caution: Grouping in IET600 is not behaving like the ‘group by’ directive in SQL statements, this may be a bit misleading for persons with experience in SQL.

Currently, the Signal Groups and the MicroSCADA attributes can be used to create grouping conditions.

After opening the ‘Export Customer Signal List’ as described in chapter 0, add groupings as required:

1. Select the Template whose filters you want to add or edit. Caution: initially always the default template is selected.
2. Select the Grouping tab.
3. Right-click into the empty space in the ‘Grouping’ Tab -> a row context menu appears:

4. Select ‘Insert New Row’ -> a new ‘Grouping’ group is added. Give this group a descriptive name in the ‘Group’ column Expand its child rows to define grouping conditions.

5. Right-click into the row header in any existing grouping condition row and use ‘Insert New Row’ to add grouping conditions as required.
The conditions within one grouping condition row will be AND-combined, the conditions of several rows will be OR-combined. The tool tip of any cell will show the result of the whole ‘Grouping’ group configuration:

The example shows a grouping that will have all Protection signals that cause an Alarm in the first group, all Protection signals that do not cause an alarm in the second group, all measurements in the 3rd group and all other signals will automatically be configured in an additional group which appears after all defined groups.

6. Right-click into the row header in any existing grouping condition row and use ‘Delete Row(s)’ to remove grouping conditions as required.

Removing the last condition of a group will also remove the group itself.

7. You can move individual rows up and down, this will affect the sequence in which the groups are exported. Right-click into the row header in any existing filter condition row and use ‘Move Row(s) Up’ or ‘Move Rows(s) Down’ to remove filter conditions as required.

8. As the example shows, you can configure as many filter groups as required, if you want.

If you uncheck the ‘Active’ flag of a group, the signals specified by this group will appear in the ‘collect all other signals’ group at the end, they will not be missing. This may make sense when experimenting with groupings to fulfil customer requirements. After finalizing the Grouping, it probably does not make sense to change the Grouping activations between repeated exports.

17.3.4 Configure Sorting

Currently, the Signal Groups and the MicroSCADA attributes can be used to create sorting conditions.
After opening the ‘Export Customer Signal List’ as described in chapter 0, and configuring filtering and grouping, configure the sorting as required:

1. Select the Template whose filters you want to add or edit. Caution: initially always the default template is selected.

2. Select the ‘Sorting’ tab.

3. Right-click into the empty space in the ‘Sorting’ Tab -> a row context menu appears:

4. Select ‘Insert New Row’ -> a new sorting criterion is added. Add and configure as many rows as you need to specify your sorting criteria:
The example shows a typical sorting that uses the Substation Hierarchy and then Signal Categories / Groups to provide sorting.

Caution: Grouping has preference over sorting: First the Grouping criteria will be used, and then the sorting criteria will be applied to signals within each group.

5. Right-click into the row header in any existing sorting criterion row and use 'Delete Row(s)' to remove rows as required.

6. You can move individual rows up and down, this is purely for convenience and does not affect the filtering logic. Right-click into the row header in any existing filter condition row and use 'Move Row(s) Up' or 'Move Rows(s) Down' to remove filter conditions as required.

17.3.5 Configure Formatting

‘Formatting’ collects some miscellaneous items which all have to do how data are displayed in the exported Customer Signal List. It includes:

- whether Alarm Groups and Command Groups are shown as groups
- whether NCC Addresses are displayed as structured or unstructured addresses (or both).
- the formatting of Title Rows (Color, Font etc.) to make them stand out more distinctly in the Customer List.

After opening the ‘Export Customer Signal List’ as described in chapter 0, and configuring grouping, configure the formatting as required:

1. Select the Template whose filters you want to add or edit. Caution: initially always the default template is selected.

2. Select the ‘Formatting’ tab.

3. Configure Data Grouping as required:
Chapter 17

If this is set to true, Alarm Groups and Command Groups will show as collapsible groups in the Customer Signal List:

4. Configure NCC Address format as required:
5. Configure Title Formatting (a default format for all group titles is preconfigured).

You can add additional formatings for the title of each group. To do so, right-click into the row header of this section and choose 'Insert New Row':

6. After inserting a row, you must configure for which group title this formatting shall be valid. You can select groups which you have defined in the ‘Grouping’ Tab.

In the example, the title for the group ‘Protection Alarms’ is configured to have a red font color; all other titles will use the default (black font color).
7. Right-click into the row header in any existing sorting criterion row and use 'Delete Row(s)' to remove rows as required. You cannot remove the row ‘Default for All Title Rows’.

8. You can move individual rows up and down, this is purely for convenience and does not affect the filtering logic. Right-click into the row header in any existing filter condition row and use ‘Move Row(s) Up’ or ‘Move Rows(s) Down’ to remove filter conditions as required.
17.4 Export a Customer Signal List

17.4.1 First Export

A Customer Signal List can only be exported from the Main menu.

1. Select the ‘HMI’ menu tab -> the Navigation Panel switches to the ‘HMIs’ tab.

3. Select a suitable Template (which contains the filtering, grouping and sorting configuration) for the export.

   Caution: for a later re-export of the same Customer Signal List (see chapter 17.4.2 below), it is important that you remember which template you used for a particular list. Ensure that you have some way of knowing which Template belongs to which Customer Signal List.

4. Select the HMI or Hot Standby System whose data you want to export (for each HMI, one Excel file will be generated).

   Depending on the selected HMI, IET600 proposes a filename. You may change it according to your needs by clicking on the ‘…’ button.

   Caution: changing the HMI will reset this filename to the default filename for the new HMI export.
5. If the selected HMI is a combined HMI/Gateway, select which NCCs you want to include in the export (per default, all are selected). If you select at least one NCC, you can decide to export ‘All Signals’ or ‘Only Signals to NCC(s)’.

![Image of Export Customer Signal List]

6. You can define how data are distributed to Excel Sheets by selecting an appropriate container:

![Image of Export Customer Signal List]

The most common choice is probably ‘Bay’ which will export one sheet per bay and an additional sheet for all signals that are not allocated to a Bay.

If you want all signals in one single sheet, you might want to select ‘Substation’.
Other criteria are available to adapt to more special customer requirements. As you can configure the MicroSCADA Object Identifiers according to your needs, you can use its parts to create very specific containers, if this is required.

7. Depending on the selection of a container, the corresponding objects will appear now for individual selection (Bays in the example).

Only those containers with signals mapped to the selected HMI will appear. By default, all objects are pre-selected.

You can de-select those containers you do not want. E.g. early in the project, you may want to export only few Bays (e.g. 1 Line, 1 Trafo and 1 Bus Coupler Bay) for discussion with the customer. For documentation, you will want to select all containers.

Caution: changing the HMI or the container will cause this list to change and the selection to revert to the default (all containers selected).

If necessary, you may activate or deactivate ‘Filtering’ groups.

8. Click on ‘Export’ to finally export the data of the selected HMI.

9. Repeat steps 4, 5 and 6 for additional HMIs, if needed.

10. Click ‘Close’ to finally close the dialog.

You may now apply additional formatting in the exported Excel Sheet (e.g. color rows and individual cells).

17.4.2 Update an Existing Customer Signal List

Based on the Template and an existing Customer Signal List, IET600 allows you to re-export the signals while keeping the formatting information.

The logic is as follows:
• The column configuration is taken from the Template. You can move columns around or delete columns from the Template which the customer does not require.

• For signals which existed in the previous Customer Signal List, the formatting is preserved and any changed attributes will be exported with the new values. IET600 will identify signals correctly, even if grouping and/or sorting have changed.

• New signals will be inserted according to grouping and sorting rules, with default formatting (black text on white background).

• Deleted or filtered-out signals will be removed from the list, their formatting is lost.

Caution: If you are experimenting with filter settings, it is recommended to have a backup of the old Customer Signal List: if you accidentally misconfigure the filter so that no signals are exported, all your formatting will be lost irrevocably.

The configuration of a re-export is identical to the first export as described above.

1. Do steps 1 to 7 above

2. If IET600 detects that a Customer Signal List with the same name exists as you have configured in step 4, it will offer you to either update the existing list or overwrite it by recreating the list from scratch:

Caution: if you answer with 'No', the existing list will be overwritten and all its row and group formatting information will be lost!

A re-export is only possible, if you choose the same container as in the old Customer Signal List. If by accident, you choose a different container, IET600 will stop the export and warn you accordingly:

17.4.3 Editing exported Customer Signal List

There is a possibility to create new signals in Excel file.

1. Select ‘Review’ menu tab->’Unprotect Sheet’.

2. Create a new signal:
2.1. From scratch:
   2.1.1. Select a row below which you want to create a new signal.
   2.1.2. Select HMI menu tab -> 'Create signal'.

2.2. With copy/paste:
   2.2.1. Select single or multiple rows and copy them.
   2.2.2. Select a row below which you want to insert copied signals.
   2.2.3. Select HMI menu tab -> 'Insert signals'.

After inserting new row(s) you can update all needed attributes. Consider that if there is no value for MicroSCADA attribute – default value will be set while import.

17.5 Import a Customer Signal List

A Customer Signal List can only be imported from the Main menu.
1. Select the 'HMI' menu tab -> the Navigation Panel switches to the 'HMIs' tab.
2. Click 'Customer Signal List' in the 'Import' group of the Main menu.
3. The 'Import Customer Signal List' dialog appears. Click the ‘…’ button to select a Customer Signal List for import:

The Customer Signal List knows to which HMI it belongs and will select the corresponding HMI automatically.

If it is a combined HMI / Gateway you can decide to unselect NCCs. As a general rule, it is recommended to import the complete information for consistency reasons (this is also the default: all NCCs are pre-selected).

4. Per default, all containers are preselected (in the example, on export the containers have been defined as Bays; each such container is represented as a sheet in Excel).

If you are sure that changes affect one container only, you may decide to unselect the other containers for efficiency reasons (e.g. when expanding an existing Substation with an additional Bay); however, a better approach would be to reduce the number of containers already on export.

5. Select ‘Import’ to finally import the data.

6. Repeat steps 3, 4 and 5 to import data from other Customer Signal Lists to other HMIs.

7. Click ‘Close’ to finally close the dialog.
18 Protecting IEDs Against Changes

18.1 Introduction

IET600 has many mechanisms to provide data consistency across a project. While this is very useful in early engineering stages in the project, it can cause problems in late stages of commissioning: if a customer requests a change, this change may automatically propagate to other IEDs, thereby potentially changing already commissioned IEDs.

Therefore a way of protecting such IEDs against inadvertent changes is needed. However, just protecting a whole IED would limit engineering too much. IET600 therefore provides several protections:

- Protecting RCB Dataflow
- Protecting GCB Dataflow
- Protecting SVCB Dataflow
- Limiting HMI engineering (HMI Engineering Rights)

These protections can be set individually for each IED.

Additionally, IEDs may be protected against changes due to an SED file Exchange. This is additional to the user-configurable protection described here; its details are described in chapter 16.4 (SED File Exchange Between Projects).

18.2 Applying Protection

18.2.1 General Principles

The basic idea behind Dataflow protection is to prevent the engineer from doing a change that requires a reloading of the protected IED. Therefore, all Dataflow protection of an IED protects basically against the following changes:

- the reconfiguration of the Dataflow items on the IED itself, i.e. Datasets and Control Blocks
- if the IED is a client to other IEDs, the reconfiguration of Dataflow items on all these server IEDs. E.g. if an IED A is GCB Dataflow protected and is a GCB client to IED B, the subscribed GCBs on IED B can also not be modified. This is necessary: although in an SCD file, the Data model of IED A is not modified and looks as before, it needs to be reloaded as its communication stack needs to be reconfigured to the new situation.
• importing an ICD/IID file to an IED: IET600 allows to import IEDs with a changed DATA model in spite of existing Control Blocks and Datasets; however, this can potentially alter Datasets or require the reconfiguration of Dataflow items; therefore it is not allowed on dataflow-protected IEDs.

• changes of any names that may affect communication, such as the IED name.

The basic idea behind reducing HMI Engineering rights is to prevent the engineer from doing a change that requires a reloading of its clients as configured in the ‘Signal Clients Editor’ (HMIs or Gateways). Reducing HMI Engineering rights of an IED protects against the following changes:

• Signals of an IED with HMI Engineering Rights ‘Fix’ cannot be changed at all; neither can its attributes be edited, nor can their configuration in the ‘Signal Clients’ or ‘NCC Data’ Editors be changed.

• Signals of an IED with HMI Engineering Rights ‘Safe Only’ allow the change of few selected attributes in the ‘HMI Data’ Editor. ‘Safe’ attributes essentially are those that do not affect communication in any way (for a detailed listing, see 18.2.7).

### 18.2.2 Configuring and Editing Protection

The protection settings are configured in the ‘IED Status’ Editor. The example shows a situation where SVCB and GCB configuration are protected for a whole voltage level, while RCB configuration is protected for two commissioned Bays Q01 and Q02; for the same Bays, HMI Engineering Rights have been set to “Safe Only”
18.2.3 Displaying Protection

To indicate protection, IEDs that have any protection set, are indicated with a blue lock in all trees.

(However, IEDs that are e.g. clients to a protected and whose engineering is therefore also restricted, do not show such an indication in the tree).

If a protection forbids editing, the appropriate rows, columns or cells in grid-based editors are set to read-only and therefore appear greyed out.

18.2.4 RCB Dataflow Protection

Protecting an IED against RCB Dataflow changes does the following:

- The IED DATA model cannot be changed (by import of an ICD/IID file).
- The IED name cannot be changed.
- All RCBs and their related Datasets in this IED cannot be changed anymore.
- All RCBs clients of these RCBs cannot be changed anymore.
- All RCB client mappings from another IED/RCB that have the protected IED as a client of a smaller index cannot be changed anymore (an example: if IED A has an RCB A which has protected HMI X as client 3, then not only client 3 cannot be changed, but also clients 1 and 2: deleting client 2 would shift HMI X/client 3 to client 2 position, which must be prevented by protection; adding another client 3 would shift HMI X/client 3 to position 4 which again must be prevented by protection.)

The above recommends that RCB clients (typically HMIS and Gateways) should be protected only very late in commissioning, as their protection will prevent almost all editing of RCB Dataflow.

- The IED cannot be removed from/added to another Subnetwork, as this would affect the RCB Client mapping.

18.2.5 GCB Dataflow Protection

Protecting an IED against GCB Dataflow changes does the following:

- The IED DATA model cannot be changed (by import of an ICD/IID file).
• The IED name cannot be changed.
• All GCBs and their related Datasets in this IED cannot be changed anymore.
• All GCBs clients of these GCBs cannot be changed anymore.
• All GCB client mappings from another IED/GCB that have the protected IED as a client cannot be changed anymore.

The above means that protecting an IED which is GCB client to many other IEDs/GCBs will prevent most client reconfigurations. It should typically be done when the functionality provided by GCBs is tested and approved by the customer.

It is a known issue that a reconfiguration of GCBs typically requires a reloading of many IEDs. This can only be helped partly by separating different functions into different GCBs and zones (e.g. separate Interlocking from Protection, or Bus Coupler Interlocking from Transformer interlocking), partly by ensuring GCB functionality is working correctly during or soon after Factory Acceptance Test.

• The IED cannot be removed from/added to another Subnetwork, as this would affect the GCB Client mapping.

18.2.6 SVCB Dataflow Protection

Currently, most IEDs providing Sampled Value Streams via SVCBs have a fixed, pre-defined SVCB Dataflow configuration, where only the clients need to be added. As these are also typically known early in the project, SVCB protection offers less during commissioning than the other Dataflow protections. Typically this protection can be done together with SVCB Dataflow protection. It probably should not be done before other protections, as it is easy to reconfigure (little gain in protection) but may prevent many changes like IED reimport of an IID file (severe hindrance).

Protecting an IED against SVCB Dataflow changes does the following:

• The IED DATA model cannot be changed (by import of an ICD/IID file).
• The IED name cannot be changed.
• All SVCBs and their related Datasets in this IED cannot be changed anymore (however they are mostly pre-defined anyway).
• All SVCB clients of these SVCBS cannot be changed anymore.
• All SVCB client mappings from another IED/SVCB that have the protected IED as a client cannot be changed anymore. This is useful when commissioning the client IEDs of a Sampled-Value-Stream (a protection or control IED): turning on the SVCB protection of this client IED after commissioning prevents client changes that might affect its functionality (e.g. Protection or Synchronization)
• The IED cannot be removed from/added to another Subnetwork, as this would affect the SVCB Client mapping.

18.2.7 HMI Engineering Protection

HMI Signals have three levels of rights:

• “Full” allows all changes.
• “Fix” allows no changes at all.
“Safe only” allows limited changes: Customers want to be able to easily change certain attributes affecting the display of signals, such as event listing, printout and alarming. However, they also want to be sure that they do not accidentally change something that affects communication (and therefore requires reloading the HMI Database and/or retesting). “Safe Only” therefore allows the change of a limited set of attributes, while preventing adding or removing signals or changing address parameters etc.

“Safe Only” allows modification of the following attributes:

- Station text, Signal text and Status text (MicroSCADA attributes OI, OX, EH)
- Event Listing, Event Update Condition (MicroSCADA attributes EE, HA, HE, HF, HL)
- Printer, Print Update Condition (MicroSCADA attributes PA, PH, PU, LD)
- Alarm List, Alarm Class, Alarm Delay (MicroSCADA attributes AG, LA, AC, AD)
- MicroSCADA attribute CX.

18.3 Engineering Workflow and Protection

During the engineering of a project, the different protections will probably be applied during different phases of the project. In this chapter, a basic way when to protect which items is suggested.

18.3.1 GCB Dataflow Protection After FAT

Usually, any bay-to-bay logic (Bus Coupler interlocking, Double Command Blocking, Breaker Failure Protection etc.) should be defined during engineering; an adaptation may happen during Factory Acceptance Test. Once the FUP of an IED and the interlocking/protection logic is finalized, i.e. in the early phases of commissioning, probably all IED should be GCB protected to prevent accidental changes.

However, any re-import of the IED DATA model (ICD/IID-Import) is blocked by GCB protection. You may need to selectively unblock an IED to reload its DATA model.

18.3.2 GCB Dataflow Protection of Third-Party Engineered IEDs

A fairly common situation is a Substation with an MV part, an HV part and a common HMI. Often, the MV part is provided by one party; HV and HMI by another party. The HV engineer may then be required to do RCB Dataflow engineering on the MV IEDs, but may not be allowed to change their DATA model or (safety-relevant) GCB Dataflow configuration.

With the provided Protection, this can easily be enforced by putting all MV IEDs under GCB Dataflow protection.
18.3.3 Commissioning Bays

After successfully commissioning a Bay, i.e. testing protection and control functionality and ensuring the appropriate signals are correctly sent to the HMI, it makes sense to apply RCB Dataflow protection to its IEDs and reduce the HMI Engineering rights.

If you are fairly sure that you do not change any signal attributes, you can set the HMI Engineering rights to “Fix”.

However, it often happens that some attributes need to change very late in commissioning. E.g. after commissioning the 5th Bay, the customer may decide to change the Signal Text of a protection signal, and he wants to change it for all signals consistently, even those from the already commissioned Bays. In this case, set the HMI Engineering rights to “Safe Only”. Attributes that would affect communication are blocked, while attributes that affect text, printout, event and alarm listing can still be edited. This is the recommended setting after commissioning Bays.

18.3.4 Protecting HMIs While Engineering a Gateway

It is typically not recommended to apply RCB Dataflow protection and/or limit the HMI engineering rights on clients (HMIs or Gateways), as this will block changes on their servers as well.

However, in special situations this may make sense, e.g. when adding a Gateway/NCC connection to an existing substation which is already commissioned or even in operation.

In this case, one can protect the Dataflow of all IEDs and set the engineering rights of an existing HMI to “Fix”, while leaving the HMI engineering rights of the IEDs and of the Gateway on “Full”. This allows to do Signal client configuration and NCC configuration for the Gateway while ensuring that the HMI configuration cannot be changed. E.g. signal texts can then only be changed on the Gateway, not on the IED (as this would affect the HMI as well).
19 Migration of Engineering Data from Older Projects

19.1 Overview

Till now, we have assumed that we start a new project from scratch. This chapter describes how to proceed, if we need to work with an existing substation that has been created using older ABB tools.

The following Migration Scenarios are described

- Migration an existing IET600-61850-system with MicroSCADA as Station HMI.
- Existing LON/SPA system with MicroSCADA station level.

19.2 Migration Effort Considerations

19.2.1 Scenarios That Will Profit from Migration to IET600

The following cases will strongly profit from the Basic Migration:

- A customer with several similar projects and quite a defined data structure that is similar from project to project.
- A project that is following ABB engineering standards closely.

IET600 offers many automatic calculations, based on ABB engineering standards. These automatic calculations will make migration fairly easy.

19.2.2 Scenarios to be Evaluated for Migration to IET600

- An extension of a non-IEC 61850 based Substation with an IEC-61850-based Bay.

Here the new Bay, including its HMI part, can be engineered with IET600. Also, potentially, improvements of older HMI process data can be done. On the other hand, IET600 cannot generate configuration files for e.g. LON/SPA communication and also not configure HMI data for these communication protocols.

- A replacement of an HMI.
In this case, IET600 can be profitably used for modification of HMI data, as long as those parts relating to non-IEC-61850 communication can be taken from the old HMI without modification and no reconfiguration of a COM500 NCC gateway is needed.

19.2.3 Scenarios not Very Suitable for Migration to IET600

- An extension of a non-IEC 61850 based Substation with a non-IEC-61850-based Bay.
  IET600 does not offer any facilities for any communication data keeping and/or editing in this scenario.
- A small extension that affects a COM500 NCC gateway.
  In this case the work to transfer all the data to do a consistent and automated COM500 engineering would probably far outweigh any gain.

19.3 Migration of Systems of Different Types

19.3.1 Existing IEC 61850-based System with MicroSCADA Station Level

We assume an old system based on IEC 61850 with a valid and up-to-date SCD configuration and a MicroSCADA HMI and COM500i gateway.

The migration steps are.
1. Export the current MicroSCADA configuration with the most recent Data Exchange format available (typically ‘lof’ and XREF files).
2. Create a new project in IET600 5.x.
3. Import an SCD file with substation, communication and IED information.
4. Create HMIs devices for HMI signal import.
5. Create Gateways and NCCs for COM500 configuration import.
6. Import HMI signals (e.g. MicroSCADA ‘lof’ files).
7. Import COM500 configuration.

For a detailed description of the steps, please see chapter 19.4 ("Migration Step Details")

19.3.2 Existing Non-IEC 61850 System with MicroSCADA Station Level

Direct migration of LON, SPA, IEC 101/103/104 etc. configuration from a legacy IET 3.x/4.x project is not supported.
We assume an old system based on any non-IEC 61850 configuration with a MicroSCADA HMI and COM500i gateway.

1. Create a new project in IET600 5.x.
2. Create HMIs devices for HMI signal import.
3. If applicable: Create Gateways and NCCs for COM500 configuration import. Configure the NCCs with the correct Line numbers as described in
4. Optional:
   - create IEDs
   - create Subnetworks for the Protocols involved
   - attach IEDs to these Subnetworks
   - from the MicroSCADA System Configuration (Editor, sys_bascon.com file), enter the MicroSCADA Unit numbers to the IEDs/Access Points in the ‘Subnetwork Editor’.
5. Import HMI signals (e.g. MicroSCADA ‘lof’ files).
   With correctly configured UN Numbers in step 4 above, the signals will be allocated to their respective IEDs. Signals without UNs will be allocated to the HMI itself. Manual allocation for some signals may be required.
6. If applicable: Import COM500 XREF files

19.3.3 Integration of Non-61850 IEDs into a 61850 Substation

Non-IEC 61850 IEDs can be integrated into an IEC 61850-based Substation Automation system using a Gateway.

In this scenario, existing LON and other third party protocol IEDs are connected to a Gateway which translates them into IEC 61850. Whatever is available as signal and communication configuration is used as input to the Gateway Configuration Tool. This tool produces an ICD/IID file as output which can then be imported into an IET600 project.

19.4 Migration Step Details

19.4.1 Prerequisites

A valid and up-to-date SCD file of the Substation must be available.

The MicroSCADA process database must be exported from MicroSCADA in a suitable format (or valid and up-to-date export files must be available).

If applicable: the COM500 configuration must be exported from MicroSCADA in a suitable format (or valid and up-to-date export files must be available).
19.4.2 Import SCD file

1. Create a new project as described in chapter 4.2.2 (“Create a New Project").
2. Import the SCD file to this project as described in chapter 16.2.1 (“Import an SCD File to Create a Project").

19.4.3 Create HMIs and Gateways

The SCD file from IET 4.x will typically already contain HMI devices. However, not all attributes for the HMI devices are available in this SCD file, for example the MicroSCADA OI settings. Therefore, these imported HMI devices must be deleted and replaced by HMI objects from IET600 5.x, as described below:

1. Go to the HMIs Navigation Tree. You will most likely find an element ‘Unallocated IEDs’ with some IEDs below it.
2. Drag’n’drop unallocated HMIs or Gateways to the ‘HMIs’ or ‘GWs / Proxies’ element.
3. Drag’n’drop unallocated OPC servers to the HMIs to which they belong.
4. Check and/or redefine the OI length of the HMIs according to the system configuration (for details, see chapter 23, Annex: MicroSCADA OI Settings).

19.4.4 Create NCCs

If COM500 engineering is used, then the NCC nodes must also be created. NCC nodes are not defined in the SCD file from IET 4.x and shall therefore be defined manually in the new IET600 project:

1. Go to the HMIs Navigation Tree and select the NCCs node
2. Create new NCC or new HSB as described in chapter 14.4 (“Creating NCCs”)
3. Connect the Gateways to the NCCs as described in chapter 14.5 (“Connecting Gateways to NCCs”)
4. Define the NCC line numbers and addresses according to the system configuration

NCC line numbers and addresses cannot be extracted from the files available for import (SCD, 'lof' or XRF files). The COM500 X-Ref tool can export a “COM_XRNCC.XRF” file which contains information about NCC addresses and line numbers. The line number must be entered manually, but they must be entered manually in IET600. Also, the NCC names in the SCD file and the COM500 x-Ref tool can differ.
19.4.5 Import HMI and COM500 Configuration

Import the MicroSCADA Signals for each HMI as described in chapters 16.6.5 or 16.6.7.

The signals in these files will be allocated to IEDs based on the IN attribute (if available) or based on the UN attribute as a fallback.

The COM500 configuration files can be imported together with the corresponding MicroSCADA Loadfiles as described in chapter 16.7.2 ("Import COM500 Configuration File (XRF)").

Check that the imported signals have appeared under the individual bay-level IEDs.
IET600 has a feature to analyze and detect inconsistencies inside the project. In order to use this feature the Analyzer must be first activated.

If problems are detected, then in IET600 will pop up a “Diagnostics Manager” dialog, where all issues are stated. The user can find the reason for the issue and what can solve it. Detailed information is logged in the lowest part of the dialog. The same messages are present inside the IET600 Logging and can be investigated after closing the “Diagnostics Manager” dialog.

In some case, IET600 can automatically resolve the issues. If it is possible to resolve a problem, the “Select” checkbox is active and it can be ticked at the beginning of the row. To resolve problems, select all the rows and press the “Resolve” button. If the resolve operation was successful, the status will be updated and additional information is logged.
The detection of problems in IET600 Analyzer is based on predefined rules and for the moment covers just a small group of well-known issues. Often they are related to misconfiguration of IED Services and Capabilities. The settings for IED Services and IED Service capabilities can be observed in the “IED Services” and “IED Capabilities” editors. The following issues can be detected:

<table>
<thead>
<tr>
<th>Problem</th>
<th>Description</th>
<th>Auto Correction</th>
</tr>
</thead>
<tbody>
<tr>
<td>IEC61850 SCL model configuration may be invalid or corrupted.</td>
<td>During operations with SCL files, the SCL model in IET600 is validated. The detected issues will be logged. The Analyzer does not differentiate SCL error types and it will observe just for any error related to SCL model. <strong>Solution:</strong> Check the error(s), if applicable, correct them and reload the configuration file(s).</td>
<td>NO</td>
</tr>
<tr>
<td>CB(s) or Dataset(s) cannot be modified.</td>
<td>IED Services for CB or Dataset are set to ‘Fix’ or the IED Service capability &quot;Enforce Fix Service Settings&quot; are set to TRUE. <strong>Solution:</strong> Set the IED capability &quot;Enforce Fix Service Settings&quot; to FALSE for this device type. Thus, it will be ignored this rule to protect the CB(s) and Dataset(s) introduced with tissues in Ed.2.</td>
<td>YES</td>
</tr>
<tr>
<td>Problem</td>
<td>Description</td>
<td>Auto Correction</td>
</tr>
<tr>
<td>---------</td>
<td>-------------</td>
<td>----------------</td>
</tr>
<tr>
<td>Client LN(s) cannot be assigned to buffered/unbuffered RCB(s).</td>
<td>The IED Service &quot;ClientServices.bufReport&quot; or &quot;ClientServices.unbufReport&quot; is missing. <strong>Solution:</strong> Set the corresponding IED capability &quot;Default For Client Service Buffered RCB&quot; to TRUE or &quot;Default For Client Service Unbuffered RCB&quot; to TRUE for this device type.</td>
<td>Yes</td>
</tr>
<tr>
<td>Client IED(s) cannot be assigned to GCB(s)/SVCB(s).</td>
<td>The IED Service &quot;ClientServices.goose&quot; or &quot;ClientServices.sv&quot; is missing. <strong>Solution:</strong> Set the corresponding IED capability &quot;Default For Client Service Goose&quot; to TRUE or &quot;Default For Client Service Sampled Values&quot; to TRUE for this device type.</td>
<td>YES</td>
</tr>
<tr>
<td>GCB(s) and client IED(s) cannot be configured.</td>
<td>No IED Service &quot;GOOSE.max&quot;. <strong>Solution:</strong> Check the IED Service &quot;GOOSE.max&quot; and if appropriate, define a value for the IED Capability &quot;Override For Service Goose Max&quot;.</td>
<td>NO</td>
</tr>
<tr>
<td>Cannot create Dataset(s).</td>
<td>The value for IED Service &quot;ConfDataset.max&quot; is reached. <strong>Solution:</strong> Check the IED Service &quot;ConfDataset.max&quot; and if appropriate, define a value for the IED Capability &quot;Override For Service Conf Dataset Max&quot;.</td>
<td>NO</td>
</tr>
<tr>
<td>Problem</td>
<td>Description</td>
<td>Solution</td>
</tr>
<tr>
<td>----------------------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>-----------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Cannot create RCB Clients(s).</td>
<td>No Service &quot;DynAssociation.max&quot;.</td>
<td><strong>Solution</strong>: Check the IED Service &quot;DynAssociation.max&quot; and if appropriate, define a value for the IED Capability &quot;Default Max Concurrent Report Clients&quot;.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The value of the RCB Property &quot;RptEnabled.max&quot; is reached.</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Solution</strong>: If the RCB property &quot;RptEnabled.max&quot; can be modified, set the IED Capability &quot;Can Modify Report Enabled Max&quot; to TRUE.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The value of the IED Service &quot;ConfReportControl.max&quot; is reached.</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Solution</strong>: Check the IED Service &quot;ConfReportControl.max&quot; and if appropriate, define a value for the IED Capability &quot;Override For Service ConfReportControl Max&quot;.</td>
</tr>
<tr>
<td>Dataset(s) cannot be sent to the Client(s).</td>
<td>Ethernet frame of the Dataset(s) is bigger than 1527 octets.</td>
<td><strong>Solution</strong>: Reduce the side of the Ethernet frame by making the Dataset(s) smaller.</td>
</tr>
</tbody>
</table>
21 Annex: Naming Conventions

21.1 IEC 81346

21.1.1 Introduction

IEC 81346 defines a comprehensive system to name all kinds of objects in a technical environment.

One crucial requirement of IEC 81346 is that it is hierarchical. Each level is a combination of characters followed by a combination of ciphers. This allows to figure out a hierarchy from any name, independent of separators.

To be able to figure out this hierarchy is important in IET600 regarding IED names: many renaming rules (e.g. renaming an IED and all its references when renaming a Bay) depend on correctly evaluating this hierarchy.

21.1.2 General Conventions in IEC 81346

The internal names proposed by IET follow IEC 81346 as closely as possible. However, as IEC 61850 naming conventions are much less strict (e.g. all Unicode Characters are allowed for naming elements in the Substation Tree); customers may require to use names not consistent with IEC 81346. It is recommended to use the ‘Customer Name’ for such names, where you are free to use any character combination.

21.1.3 Hierarchies in IEC 81346 and IEC 61850

<table>
<thead>
<tr>
<th>Level</th>
<th>IEC 81346</th>
<th>IEC 61850</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Plant level</td>
<td>Substation</td>
</tr>
<tr>
<td>2</td>
<td>Subplant level</td>
<td>Voltage level, Function</td>
</tr>
<tr>
<td>3, 4, 5</td>
<td>Further division</td>
<td>Bay, Function, Subfunction</td>
</tr>
<tr>
<td>3, 4, 5</td>
<td>Apparatus, Components</td>
<td>Equipment, Subequipment</td>
</tr>
</tbody>
</table>
21.2 Default Substation Naming

The following chapters show some standard feeder configurations and the structuring and naming of primary equipment according to IEC 81346.

For a detailed description how IEC 81346 is used by ABB Substation Automation (including naming of secondary equipment), please refer to [1]. The document can be made available to external customers as well.

21.2.1 Typical Bay, Line

The voltage transformer BA1/2 may also sit at edge QB9/QC9
21.2.2 Typical Bay, Transformer

21.2.3 Typical Bay, Bus Coupler
21.2.4 Typical Diameter, Line/Transformer

The voltage transformer BA1/2 may also sit at edge QB9/QC9.
22 Annex: Switch Configuration

22.1 The Switch Data Model

IEC/TR 61850-90-4 specifies a DATA model for Switches. It is relatively complex, consisting of the following LN Classes:

<table>
<thead>
<tr>
<th>LN</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>LPCP</td>
<td>represents the physical properties of a port.</td>
</tr>
<tr>
<td>1 LPCP per physical port</td>
<td></td>
</tr>
<tr>
<td>LPLD</td>
<td>represents the port link discovery properties of a port (e.g. the TCP/IP address of its remote partner etc.).</td>
</tr>
<tr>
<td>1 LPLD per physical port, references 1 LPCP</td>
<td></td>
</tr>
<tr>
<td>LBSP</td>
<td>represents the RSTP properties of 1 port (e.g. “Edge” or “Trunk”).</td>
</tr>
<tr>
<td>1 LBSP per physical port, references 1 LPCP.</td>
<td></td>
</tr>
<tr>
<td>LCCH</td>
<td>represents the channel properties of a port or a redundant port pair.</td>
</tr>
<tr>
<td>1 LCCH per port or port pair, references 1 or 2 LPCP.</td>
<td></td>
</tr>
<tr>
<td>LCCF</td>
<td>represents the filtering properties of 1 port or 1 redundant port pair (e.g. MAC Address filters or VLAN filters).</td>
</tr>
<tr>
<td>1 LCCF per port or port pair, references 1 LCCH.</td>
<td></td>
</tr>
<tr>
<td>LBRI</td>
<td>represents the connection of all ports (typically called a “Bridge”) and its related properties (e.g. RSTP weight of this node etc.).</td>
</tr>
<tr>
<td>1 LBRI per Switch, references all LBSPs.</td>
<td></td>
</tr>
</tbody>
</table>

22.2 MAUs

22.2.1 Short description

A Medium Attachment Unit (MAU) is a technical term for what a normal user would call a ‘Port’ of a switch. However, MAU includes more information. As an example:
The xxxxBase-YY specifications are standards defined in IEEE 802.3xx. These specifications do not always contain exactly the same information, however. E.g.:

- 1000Base-T defines an electrical connection, a plug type (RJ45), the wiring in the cable and a maximum length of the cable (100 m).
- 1000BASE-LX defines a fiber-optic connection, the wavelength of the laser used for transmission (1,270–1,355 nm), two possible fiber types (multi-mode and single-mode), maximum cable lengths (550 m for multi-mode, 5 km for single-mode fibers), but no plug types.
- 1000BASE-LX10 defines a fiber-optic connection, the wavelength of the laser used for transmission (1,270–1,355 nm), two possible fiber types (multi-mode and single-mode), maximum cable length (10 km for single-mode fibers), but no plug types.

Currently, IET600 uses the MAU type only for Switch specifications, as they are a part of the IEC/TR 61850-90-4 DATA model.

### References


23 Annexe: MicroSCADA OI Settings

The OI Settings for MicroSCADA can be done on the Project and on each MicroSCADA individually.

23.1 Setting the OI Attribute

MicroSCADA splits the OI field into a maximum of 5 parts. All fields together including separators must be <= 63.

If you exceed this length, IET600 will give you only a very discreet warning; check this carefully, if you edit these settings:

[Image of MicroSCADA settings]

There is currently no transfer of this setting between IET600 and MicroSCADA. It is the responsibility of the engineer to synchronize the OI settings in both IET600 and MicroSCADA.

23.2 OI Settings on Project

These settings are used for 2 purposes:

- To calculate OIs from formulas for signals on IEDs, which have not been mapped to any MicroSCADA yet.
- As default values when creating a new MicroSCADA HMI (if you have a project with different settings, but which are the same for all MicroSCADAs, it is therefore most efficient to adjust the Project OI Settings accordingly).
- To calculate OIs when using HMI Mirroring.
23.2.1 Default Settings

The default settings for a newly created Project are:

<table>
<thead>
<tr>
<th>Field</th>
<th>Field Mapping</th>
<th>Field Length</th>
<th>OI Separator Length</th>
</tr>
</thead>
<tbody>
<tr>
<td>1st</td>
<td>Station Name</td>
<td>9</td>
<td>1</td>
</tr>
<tr>
<td>2nd</td>
<td>Bay Name</td>
<td>14</td>
<td>1</td>
</tr>
<tr>
<td>3rd</td>
<td>Device Name</td>
<td>5</td>
<td>1</td>
</tr>
<tr>
<td>4th</td>
<td>Optional Field 4</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>5th</td>
<td>Optional Field 5</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

The Field Mapping is not used in any IET600 logic, it just serves as a reminder what MicroSCADA expects in Fields 1, 2, 3… The OIRules in the ‘HMI Data’ Editor of IET600 may need to be adjusted, if some other configuration than the default Field Mappings are used.

23.2.2 Configuration

1. Select the ‘Project’ node in any Navigation Tree
2. Search for the property ‘MicroSCADA Object Identifier Settings’
3. Open it, configure the individual properties.

If you have several MicroSCADAs with different OI Settings for each MicroSCADA, it is recommended to set each of the Project OI Field Lengths to the biggest any MicroSCADA has in the respective field, e.g.:

<table>
<thead>
<tr>
<th></th>
<th>HMI 1</th>
<th>HMI 2</th>
<th>Project</th>
</tr>
</thead>
<tbody>
<tr>
<td>1st Field Length</td>
<td>9</td>
<td>12</td>
<td>12</td>
</tr>
<tr>
<td>2nd Field Length</td>
<td>14</td>
<td>8</td>
<td>14</td>
</tr>
<tr>
<td>3rd Field Length</td>
<td>5</td>
<td>10</td>
<td>10</td>
</tr>
<tr>
<td>4th Field Length</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>5th Field Length</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

It is possible in this case, that the total length of the Project OI will exceed 63 characters. This may cause problems, unless the setting of each MicroSCADA HMI does not exceed 63 characters. Remember in such a case to adjust the OI Settings immediately after creating a new HMI.

When using HMI Mirroring, the Project OIs are used to calculate the OIs for mirrored signals. If, in the above example, e.g. the project settings were not adjusted (e.g.
leaving 1st OI at 9), the 1st OI for HMI 2 will be truncated to a length of 9 although the 1st OI Setting at HMI 2 is correctly set at 12.

23.3 OI Settings for MicroSCADA

23.3.1 Configuration

1. Select a MicroSCADA IED in the ‘Substation’ or ‘IEDs’ Navigation Tree; or an HMI or Gateway in the ‘HMIs’ Navigation Tree
2. Search for the property ‘MicroSCADA Object Identifier Settings’
3. Open it, configure the individual properties.

When creating an HMI, the OI Settings from the Project are copied to the HMI as initial values.

23.4 Known Limitations

IET600 does not support ‘Field Mapping’, i.e. meanings of fields. E.g. if on Project Level, the first 3 parts are given the meaning of Station – Bay – Device, and on an individual HMI, the first 3 parts are given the meaning of Station – Device – Bay, IET600 will not magically reverse Device and Bay part; the calculated OI will be identical. Also there is no automated logic that would adapt OIRules in the ‘HMI Data’ Editor, if ‘Field Mappings’ are changed; this is up to the engineer.

IET600 currently does not support to split the OI attributes into 5 separate fields as has been introduced in more recent MicroSCADA versions.
Chapter 24

Annex: Mirroring Command Group Logic

24.1 Introduction

MicroSCADA requires a specific number of signals to be present for the proper handling of a command from the MicroSCADA HMI.

When mirroring a command, it is important that all signals needed for this proper command handling are mirrored, as the MicroSCADA-internal handling of switching operations is based on signals with defined LNs/IXs to have a specific meaning.

IET600 provides a logic that automatically mirrors all required signals, if any single command is selected.

24.2 Predefined Mirroring Group

Mirroring Groups are applicable only for MicroSCADA; the MicroSCADA-internal logic for handling switching operations is based on these signals with defined IXs (where each IX also has a defined meaning). It depends on the MicroSCADA LN/IX attribute.

The logic for assembling such a Mirroring Command Group depends on the MicroSCADA LN and IX attributes:

1. All Signals must have the same LN.
2. The Signals with the following IX attributes and meaning are selected for the group:

   Commands:
   - 11 (Command select open)
   - 12 (Command select close)
   - 13 (Command execute open)
   - 14 (Command execute close)
   - 25 (Command cancel)

   Indications:
   - 10 (Position indication)
   - 15 (Device control blocked)
   - 16 (Open interlocked)
   - 17 (Close interlocked)
   - 19 (Selection on monitor)
   - 20 (Command enabled)
• 22 (Synchrocheck) (only for Circuit Breakers)
• 41 (Block for opening)
• 42 (Block for closing)
• 55 (AddCause of command)
Annex: NCC Command Group Logic

25.1 Introduction

MicroSCADA requires a specific number of signals to be present for the proper handling of a command that is sent from the NCC and must be passed on to a local IED.

The logic to build these command groups depends not so much on the protocol between NCC and MicroSCADA (as a Gateway), but on the Protocol between MicroSCADA and the local IED.

For some known and predefined command groups, IET600 automatically builds the whole group if just one signal is pre-selected.

If no such group of signals can be found, IET600 will allow you to create a flexible command group.

25.2 Predefined Command Groups

Specific groups are pre-defined for the following protocols between MicroSCADA and local IEDs in a Substation:

1. IEC 61850:
   - group of 5 fixed commands and 1 fixed feedback indication for circuit breaker, disconnector and earth switch operation.
   - group of 1 command and 1 optional feedback indication for all other commands (e.g. tap changer operation).

2. IEC 101/104
   - group of 1 fixed commands and 1 fixed feedback indication for circuit breaker, disconnector and earth switch operation.
   - group of 2 fixed commands and no feedback indication for circuit breaker, disconnector and earth switch operation.

3. DNP 3.0
   - group of 1 fixed commands and 1 fixed feedback indication for circuit breaker, disconnector and earth switch operation.
   - group of 2 fixed commands and no feedback indication for circuit breaker, disconnector and earth switch operation.

4. LON
• group of 1 fixed commands and 1 fixed feedback indication for circuit breaker, disconnector and earth switch operation.
• group of 4 fixed commands and 1 fixed feedback indication for circuit breaker, disconnector and earth switch operation.

The logic for building these groups will be described in the following chapters.

25.2.1 Group of 5 Fixed Commands with 1 Fixed Feedback Indication

This group is used for operation of switches in IEDs connected to the Gateway with protocol IEC 61850.

The logic for assembling this group depends on the MicroSCADA IN attribute or on internal signal texts. The Signal Text logic is a backup for the IN logic, as the IN is not always available (e.g. in mirroring configurations).

The IN-based logic is:
1. All Signals must be mapped to a Logical Node with ‘CSWI’ or ‘GGIO’ as LNClass.
2. All Signals must reference the same Logical Node.
3. Exactly one signal in the same LN must meet the following conditions:
   • IN ends with “ctlSelOff”
   • IN ends with “ctlSelOn”
   • IN ends with “ctlOperOff”
   • IN ends with “ctlOperOn”
   • IN ends with “ctlCan”
   • IN contains “Pos.stVal”

The Signal-text-based logic is:
1. All Signals must have the same MicroSCADA LN.
2. All Signals must have “Analogue Output” as MicroSCADA Process object type (PT).
3. Exactly one signal with the same MicroSCADA LN must meet the following conditions:
   • Internal Signal text contains both “select” and “open”
   • Internal Signal text contains both “select” and “close”
   • Internal Signal text contains both “execute” and “open”
   • Internal Signal text contains both “execute” and “open”
   • Internal Signal text contains “cancel”
   • Internal Signal text contains “position” and MicroSCADA IX = 10
     (here, the MicroSCADA IX is intended to prevent conflicts with duplication in redundancy configurations, which typically contain several “position” signals on indexes 10, 110 and 210)

(Sequence and capitalization do not matter, e.g. “Command open select” and “Select open” would both fulfill the first condition above).
25.2.2 Group of 1 Fixed Command with 1 Fixed Feedback Indication

This group is used for operation of switches in IEDs connected to the Gateway with protocols IEC 101/104, DNP 3.0 or LON.

The logic for assembling this group depends on the MicroSCADA LN and IX. It is:

1. All Signals must have the same MicroSCADA LN.
2. Exactly one signal in the same LN must meet the following conditions:
   - IX is 13 and is of any "output" Process object type (this signal will be taken as command)
   - IX is 10 and is of Process object type “Double Binary Indication” (this signal will be taken as feedback indication)
3. No signal in the same LN is allowed to meet the following conditions:
   - IX is 11
   - IX is 12
   - IX is 14

(So a group of signals with LN C1Q01QA1 containing indexes 10, 13 and 20 would allow to form such a group, whereupon a group of signals with LN C1Q01QA1 containing indexes 10, 11, 12 and 13 would prohibit the creation of such a group)

25.2.3 Group of 2 Fixed Commands with no Feedback Indication

This group is used for operation of switches in IEDs connected to the Gateway with protocols IEC 101/104.

The logic for assembling this group depends on the MicroSCADA LN and IX. It is:

1. All Signals must have the same MicroSCADA LN.
2. All Signals must have the same Process object type which must be of an "output" type.
3. Exactly one signal in the same LN must meet the following conditions:
   - IX is 11
   - IX is 12 (this signal will be taken as feedback indication)
4. No signal in the same LN is allowed to meet the following conditions:
   - IX is 10
   - IX is 13
   - IX is 14

(So a group of signals with LN C1Q01QA1 containing indexes 11, 12 and 20 would allow to form such a group, whereupon a group of signals with LN C1Q01QA1 containing indexes 10, 11, 12 and 13 would prohibit the creation of such a group)
25.2.4 Group of 4 Fixed Commands with 1 Fixed Feedback Indication

This group is used for operation of switches in IEDs connected to the Gateway with protocol LON.

The logic for assembling this group depends on the MicroSCADA LN and IX. It is:

1. All Signals must have the same MicroSCADA LN.
2. All Signals must have the same Process object type which must be of an “output” type, except the one with IX = 10.
3. Exactly one signal in the same LN must meet the following conditions:
   - IX is 10 and is of Process object type “Double Binary Indication” (this signal will be taken as feedback indication)
   - IX is 11
   - IX is 12
   - IX is 13
   - IX is 14

(So a group of signals with LN C1Q01QA1 containing indexes 10, 11, 12, 13 and 14 would allow to form such a group, whereupon a group of signals with LN C1Q01QA1 containing indexes 10, 11, 12 and 13 would prohibit the creation of such a group).

25.3 Flexible Command Groups

A flexible command group is created, if none of the predefined command groups defined above can be found.

To a flexible command group, an unlimited number of unspecified commands can be added. However, a flexible command group may contain either 0 or 1, but not several feedback indications.
Chapter 26

Annex: IED Capabilities

26.1 Introduction

When engineering Dataflow with IEC-61850-based configuration tools, some problems appeared:

1. IED services were not always properly defined.
2. The IED services defined in IEC 61850 were interpreted differently by different vendors.
3. Some IED services/restrictions were not defined in IEC 61850.

To deal with these issues, IET600 introduced ‘IED Capabilities’ which allowed to either define default values for service that were often missing or define override values for services which are known to often be defined wrongly.

Most of the problems/ambiguities have been corrected and missing services added in IEC 61850, Ed.2; their proper usage will hopefully render the IED capabilities described in this chapter obsolete at some stage. However, as there are still many ICD files with incomplete information, IET600 still provides these capabilities to allow proper engineering with such files. Chapter 26.2 describes the capabilities provided by IET600 in detail, chapter 26.3 shows how to resolve some known problems via IED capabilities.

26.2 Capabilities Description

The IED Capabilities editor is shown in the IEDs tab and shows the following properties:

1. General IED properties that serve as identification of a set of IED Capabilities:

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Manufacturer</td>
<td>IED manufacturer</td>
</tr>
<tr>
<td>IED Type</td>
<td>IED Type / Family</td>
</tr>
<tr>
<td>Configuration Version</td>
<td>IED configuration version, allows to define capabilities for a specific version of an IED</td>
</tr>
</tbody>
</table>
To apply IED capabilities to a specific IED, IET600 first looks for a capability that matches the IED in question by Manufacturer, IED Type and Configuration Version; then by Manufacturer and IED Type only, then by Manufacturer only. If still no matching template is found, then the Default capability (without Manufacturer or IED Type or Configuration Version) is applied.

Please be aware that the Default Capability is very conservative, it does not allow editing of Dataflow items. This has historical reasons (most “unknown” IEDs did not allow the editing of its Dataflow), it may not apply in every case.

2. Capabilities related to RCB Client configuration:

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Update Report Client Mapping</td>
<td>TRUE will cause the Report Clients to be set from default clients automatically, e.g. when importing an ICD/IID file with RCBs but without clients.</td>
</tr>
<tr>
<td>Can Modify Report Enabled Max</td>
<td>TRUE if Report Enabled Max attribute can be modified.</td>
</tr>
</tbody>
</table>

3. Capabilities related to serve as initialization values for IET600-specific attributes:

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Default Report Control Block Status</td>
<td>Default status of RCB when imported from SCL file. “IED-defined, read-only” prevents any changes to the imported RCB. “IED-defined, configurable” allows changes to be made to the imported RCB.</td>
</tr>
<tr>
<td>Default Goose Control Block Status</td>
<td>Default status of GCB when imported from SCL file. “IED-defined, read-only” prevents any changes to the imported GCB. “IED-defined, configurable” allows changes to be made to the imported GCB.</td>
</tr>
<tr>
<td>Default Sampled Values Control Block Status</td>
<td>Default status of SVCB when imported from SCL file. “IED-defined, read-only” prevents any changes to the imported SVCB. “IED-defined, configurable” allows changes to be made to the imported SVCB.</td>
</tr>
<tr>
<td>Default Dataset Status</td>
<td>Default status of Dataset when imported from SCL file. “IED-defined, read-only” prevents any changes to the imported Dataset. “IED-defined, configurable” allows changes to be made to the imported Dataset.</td>
</tr>
</tbody>
</table>
Declaring any of the above items as “IED-defined, read-only” will prevent their editing, even though the IED services might allow editing.

Declaring any of the above items as “IED-defined, configurable” will allow their editing only insofar, as it is allowed by the IED services also.

4. Capabilities related to limitations not defined via SCL file

While IEDs may specify a maximum number of instances per RCB in the SCL file, this does not correspond to the number of concurrent connections which are possible. This property provides an additional parameter to take such limits into account:

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Default Max Concurrent Report Clients</td>
<td>Equivalent to the IED Service DynAssociation.Max, any value in the ICD/IID file will be observed. In Ed. 1 the service exists, but as a Boolean, not as a number. Therefore it may be set here. Maximum number of RCB clients that can concurrently connect to an IED, therefore indirectly the maximum number of allowed RCB clients/instances.</td>
</tr>
</tbody>
</table>

In IEC 61850, Ed.2, the service “DynAssociation.Max” has been given the meaning covered by this capability. If service DynAssociation.Max is specified, it takes precedence over the above parameter.

5. Capabilities that serve to set default values or override values of an SCL file in project

The override properties serve to override values contained in an SCL file. If these values are not defined, values from the SCL file will be observed.

The default values serve only to set default values, if none are defined by IEC 61850. This is typically the case for IED Services which were introduced only in Ed.2; for Ed.1 IEDs, no default values can safely be assumed, therefore they may need to be configured here.

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Override for Service ConfDataset Max</td>
<td>Change the maximum number of Datasets from the value specified in the SCL file. If not defined, values from the SCL file will be observed.</td>
</tr>
<tr>
<td>Override for Service ConfDataset Max Attributes</td>
<td>Change the maximum number of Dataset attributes from the value specified in the SCL file. If not defined, values from the SCL file will be observed.</td>
</tr>
<tr>
<td>Property</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Override for Service ConfDataset Modify</td>
<td>TRUE means that a preconfigured Dataset may be modified.</td>
</tr>
<tr>
<td>Override for Service ConfReportControl BufMode</td>
<td>Specifies the buffer modes (buffered, unbuffered, both) allowed to configure for new control block types. If not defined, values from the SCL file will be observed.</td>
</tr>
</tbody>
</table>
| Default for Service ConfReportControl BufConf | Any value of the IED Service in the ICD/IID file will be observed. In Ed. 1 the service does not exists and there is no default value, therefore it may be set here.  
Checked means that the buffered attribute of preconfigure report control blocks can be changed via SCL  
Unchecked means that they cannot be changed.                                                                 |
| Override for Service ConfReportControl Max    | Maximum number of instantiated report control blocks. If this number is equal to the number of preconfigured instances, then no new instances can be created.                                                |
| Override for Service Goose Max                | Maximum number of GOOSE control blocks, which can be configurable for publishing (max=0 means that the device is only a GOOSE client)                                                                   |
| Default for Client Service Goose              | Any value of the IED Service in the ICD/IID file will be observed. In Ed. 1 the service does not exists and there is no default value, therefore it may be set here.  
Checked means that an IED can be configured as a GOOSE Client.                                                                 |
| Default for Client Service SampledValues      | Any value of the IED Service in the ICD/IID file will be observed. In Ed. 1 the service does not exists and there is no default value, therefore it may be set here.  
Checked means that an IED can be configured as an SVCB Client.                                                                 |
| Default for Client Service Buffered RCB       | Any value of the IED Service in the ICD/IID file will be observed. In Ed. 1 the service does not exists and there is no default value, therefore it may be set here.  
Checked means that an IED can be configured as a Client for buffered RCBs.                                                                 |
### Property Description

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
</table>
| Default for Client Service Unbuffered RCB | Any value of the IED Service in the ICD/IID file will be observed. In Ed. 1 the service does not exist and there is no default value, therefore it may be set here.  
Checked means that an IED can be configured as a Client for unbuffered RCBs. |
| Enforce Fix Service Settings | A Tissue for IEC 61850 has specified a more precise logic in which configurations Datasets and Control Blocks may or may not be deleted. One important rule was that deletion is not possible if any of the Control Block handling attributes is “fix” (i.e. they cannot be edited by the System Configuration Tool).  
Unfortunately, many IED Configuration Tools have not backported this logic to Ed. 1 IEDs, which may cause IET600 to refuse to delete Datasets and CBs for such IEDs. ‘Override for Fix Service Settings’ allows to override this behavior.  
Related to ‘xxxSettings’ IED Services (ReportSettings, GSESettings, etc.) and their attributes.  
TRUE - CBs cannot be deleted or added and datasets and their references cannot be modified, independently from IED Services ‘xxxSettings’ configuration.  
FALSE - CBs can be deleted or added and datasets and their references can be modified, independently from IED Services ‘xxxSettings’ configuration.  
UNDEFINED - The rule below will be taken into account in order to delete or add CBs and datasets.  
‘xxxSettings’ IED Services rule:  
According to a rule inside the IEC61850 standard, if one of the ‘xxxSettings’ attribute has ‘Fix’ value, then CBs cannot be deleted or added.  
Additionally, if some ‘xxxSettings.datSet’ is ‘Fix’, the data set and its references cannot be modified.  
If ‘Enforce Fix Service Settings’ has some value, then the ‘xxxSettings’ configurations will be ignored. |
6. Capabilities related to Disturbance Recording.

These properties serve to provide default values. Values from the SCL file take precedence over these default values.

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Default Disturbance Recorder Capability</td>
<td>Allows to specify whether an IED provides Disturbance Recording (normally, an IED should indicate this by providing an RDRE LN in its DATA model, this is mainly intended for legacy IEDs)</td>
</tr>
</tbody>
</table>

7. Capabilities related to File Transfer

These properties serve to provide default values if respective elements are missing in the. Values from the SCL file take precedence over these default values.

This is mainly used in connection with Disturbance Recording to specify the possible File Transfer protocols.

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Default for MMS File Transfer Capability</td>
<td>Any value of the IED Service in the ICD/IID file will be observed. In Ed. 1 the service does not exists and there is no default value, therefore it may be set here. Allows to specify whether an IED supports File Transfer via MMS</td>
</tr>
<tr>
<td>Default for FTP File Transfer Capability</td>
<td>Any value of the IED Service in the ICD/IID file will be observed. In Ed. 1 the service does not exists and there is no default value, therefore it may be set here. Allows to specify whether an IED supports File Transfer via FTP</td>
</tr>
<tr>
<td>Default for FTPS File Transfer Capability</td>
<td>Any value of the IED Service in the ICD/IID file will be observed. In Ed. 1 the service does not exists and there is no default value, therefore it may be set here. Allows to specify whether an IED supports File Transfer via FTP</td>
</tr>
</tbody>
</table>

8. Miscellaneous Capabilities

These properties serve to provide default values to configure some behavior which is not adequately represented by IED services.
<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enforce Single MMS message</td>
<td>Allows to enforce that an IED sends RCB/Datasets in a single telegram (equivalent to the parameter “Send single message MMS writes” in the 61850-OPC-Gateway configuration tool).</td>
</tr>
<tr>
<td>Indexed RCBs</td>
<td>Allows to specify whether an IED understands a request for indexed RCBs (some Ed.1 IEDs require specifically that the client ask for a non-indexed RCB; Ed.2 specifies that all IEDs must understand indexed RCBs).</td>
</tr>
</tbody>
</table>
| Dynamic Datasets              | Allows to specify whether a 61850-OPC-Gateway Client shall configure dynamic Datasets on a Server IED (up to the limit that the services of the Server IED allows).  
Main usage is to set this property to “false” to tell a 61850-OPC-Gateway Client not to use dynamic Datasets even though the Server IED may be capable to provide them. |
26.3 Common Corrections with IED Capabilities

There are typical cases where IED capabilities are needed, they are described here.

26.3.1 Service ConfReportControl, Max attribute

Due to a vague wording in the IEC 61850, this attribute has been misinterpreted to be the maximum number of possible Report Control Blocks, rather than being the maximum number of possible Report Control Block instances.

A typical example is an IED with 8 fixed RCBs where IET600 is supposed to add clients only. To prevent an addition of RCBs, the IED manufacturer delivers an IED file with ConfReportControl.Max = 8; each RCB has RptEnabled.Max = 4. The intention is to allow up to 4 clients for each RCB, allowing a total number of 32 clients (and therefore 32 RCB instances). However, if you try to configure clients in the RCB Client Editor, IET600 will allow you to configure a total of 8 clients only and will block you from configuring more (which is not the intention of the IED manufacturer).

To allow proper configuration of such an IED, define a new IED Capability for its Manufacturer and Type and provide an override value for the Service ConfReportControl, Max as shown below:
26.3.2 Service ConfDataset, maxAttributes attribute

The interpretation of this attribute has changed due to a clarification in the IEC 61580 standard.

Up to IET600 5.2, this value was interpreted to mean the maximum numbers of Data Attributes (DAIs) that could be contained in a Dataset. So an entry CSWI.Pos[ST] has 3 attributes (stVal, q and t) and therefore contributed 3 attributes to the total.

However, after clarification in IEC 61850, this must be interpreted to mean Dataset entries. So a CSWI.Pos[ST] is one entry, irrespective of how many attributes it has, and therefore contributes 1 entry to the total.

To stay compatible with existing, already engineered projects, IET600 FP1 uses the following logic:

- in all ABB Ed.1 IEDs, the value is interpreted to mean attributes. This corresponds to the behavior of the IED Configuration tool PCM600, which keeps existing maxAttributes values for Ed.1 IEDs and counts attributes for Ed.1 IEDs.
- in all ABB Ed.2 IEDs, the value is interpreted to mean Dataset entries. This corresponds to the behavior of the IED Configuration tool PCM600, which adjusts the maxAttributes values for Ed.2 IEDs accordingly and counts entries for Ed.2 IEDs.
- in all non-ABB IEDs, the value is interpreted to mean Dataset entries. This conforms to the interpretation of most, if not all, 3rd-party vendors.

If any IED does not fit the above logic, an engineer can use the IED capabilities and set “Override for Service ConfDataSet Max Attributes” to an appropriate value.

26.3.3 Service ConfDataset, Modify attribute

The specification of this service in IEC 61850 says: “TRUE means that preconfigured datasets may be modified”. Unfortunately, in an SCD file, there is no way to distinguish a preconfigured Dataset from one that has been generated at some stage during the engineering process.

Some vendors that do not provide preconfigured Datasets, set ConfDataset.Modify to false with the idea that as there are no preconfigured datasets, they cannot be modified either. IET600 however, interprets this flag as meaning that no Dataset can be modified, not even the ones configured in IET600 and will give an appropriate error message when the user tries to do so. In such cases, it is necessary to configure an IED Capability for this IED Type and Vendor with “Override for Service ConfDataset Modify” set to TRUE.

26.3.4 Client Services

Client Services have been introduced in IEC 61850 Ed.2 to allow to specify precisely for which Control Blocks an IED can serve as a client. Default value in case of a missing service is that an IED cannot be a client to the particular Control Block in question.
Older IET600 versions had a fixedly programmed logic that all IEDs containing an IHMI, ITCI or ITMI node were assumed to be possible RCB clients. In IET600, this is not the case anymore; such clients need either to have appropriate services or appropriate overrides configured.

When creating a new project in IET600, IED capabilities will be configured to take missing services into account:

However, if you migrate a project created in an older version to IET600 5.3, you may experience that you suddenly cannot configure RCB clients as you used to. This is because these client overrides in the IED capabilities are missing; please take the above picture as a template how to configure them to avoid problems in RCB Client configuration.

### 26.3.5 Dataflow Cannot be Created or Edited At All

This will most commonly happen when you try to create or edit Dataflow of an IED whose type is not yet know to IET600. In this case, IET600 uses the Default Capability which sets all Dataflow items to read-only; to prevent accidental misconfiguration.
To allow proper configuration of such an IED, look up the Manufacturer and IED Type in the ‘IED Status Editor’. Then define a new IED Capability for this Manufacturer and IED Type, and set the capabilities as shown below:

This is the most neutral configuration; if all IED services are defined correctly (which is possible only for an Ed.2 IED), Dataflow configuration should be possible now as intended by the IED manufacturer.

If Dataflow configuration still does not work as intended, you will most likely get a hint which service is at fault in a Log entry or Dialog message. You may try to change some IED capabilities to get the desired result. However, please be very careful when doing this: consult Manufacturer documentation and test and verify results of such changes with extreme care, as they may otherwise result in improper functioning of a system.
Annex: IET600 and IEC 61850 Edition Handling

27.1 IEC 61850 Classification

27.1.1 IEC 61850 Edition, Version and Revision

To classify IEC 61850 releases, the standard has defined a version and a revision.

- “Version” refers to the year of publication of the specification.
- “Revision” refers to a revision indicator. This indicator is required to differentiate revisions of the standard between two versions. If it is missing in any versioning, “A” is assumed.

Edition 2 has gone through such a revision process, an Edition 2.1 is planned, but not formally released yet at the time of the release of this manual.

The following table provides an overview:

<table>
<thead>
<tr>
<th>Edition</th>
<th>Year of release</th>
<th>Version</th>
<th>Revision</th>
</tr>
</thead>
<tbody>
<tr>
<td>Edition 1</td>
<td>2003</td>
<td>2003</td>
<td>A</td>
</tr>
<tr>
<td>Edition 2</td>
<td>2007</td>
<td>2007</td>
<td>A</td>
</tr>
<tr>
<td>Edition 2.1</td>
<td>(not yet released)</td>
<td>2007</td>
<td>B</td>
</tr>
</tbody>
</table>

27.1.2 IEC 61850 Namespaces

A “namespace” in IEC 61850 terminology is uniquely defined by a version/revision combination.

Regarding the configuration of a Substation, IEC 61850 has 2 particularly relevant parts:

- IEC 61850-6, which defines the “System Configuration description Language” or SCL schema (which is an XML namespace).
  The namespace of an SCL file is defined by the “version” and “revision” attributes of the topmost ‘SCL’ element (not to be confused with “version” and “revision” attributes of the ‘Header’ Element, a sub-element to the ‘SCL’ element).
- IEC 61850-7-1/2/3/4, which defines the namespaces of the DATA of an IED, i.e. the definitions of Common Data Classes, Data objects, Logical Nodes etc.
The DATA namespace of an IED is declared in the DATA attribute “[first Logical Device].LLN0.NamPlt.IdNs”

It is a bit confusing, but important to understand that the above-mentioned two namespaces are completely independent of each other.

It is possible to have:

- an IED which contains a CDC with a new Attribute Type that requires IED Namespace 2007B, but which is exported into an ICD file which declares the SCL namespace as 2003A.
- an IED with a ServerAt element that requires an Export into an ICD file with XML Namespace 2007A, but the IED DATA model itself is namespace 2007B

### 27.1.3 SCL Elements Conflicting With SCL Namespace Version 2003

The following elements require an SCL file to declare a namespace with a version of 2007:

- Functions that are attached to Voltage Levels or Bays (in the Edition 1 XML scheme, a Function could only be attached to a Substation). This is a helpful element to provide an additional layer of elements in the Substation Tree and can make modeling e.g. of a 1.5-CB-Bay easier. However, this can often be worked around by using a clever naming scheme to avoid upgrading a project to a newer version.

- ServerAt element which makes it possible that a server can supply its data via several Access Points; this did not exist in the in the Edition 1 XML scheme. This will probably be required by bay-level protection and control IEDs which receive measurement data via sampled value streams (e.g. from an ABB SAM600 IED) on a process bus and at the same time participate in GOOSE and RCB communication on the station bus. If an IED configuration tool specifies a ‘ServerAt’ element in an ICD file, there is no workaround in IET600; a project with such IEDs must have version 2007.

- IEC 61850 Edition 2 allows to define IED capabilities more precisely by adding new services. It also allows to specify services on the level of an Access Point, not only for the IED as a whole. This is helpful for Dataflow configuration. If an IED configuration tool specifies such services in an ICD file, there is no workaround in IET600; a project with such IEDs must have version 2007.

- SED Data exchange. The SED file and its related attributes are specified only in IEC 61850 Edition 2.

As a workaround, one could exchange IEDs with an IEC 61850 Edition 1 project in the form of ICD or SCD files. However, an SED data exchange has a Dataflow configuration as purpose. When using the above workaround, all participating IEDs should be of the same version/revision to avoid compatibility problems. Also, the above workaround does not provide a checkout/protection mechanism; it therefore requires an experienced engineer to prevent incorrect or erroneous configuration changes.
27.1.4 IED DATA Conflicting With Namespace Version 2003

The following elements require an IED to declare its DATA namespace with a version of 2007:

- Usage of any DATA element with new Attribute data types defined only in IEC 61580, Edition 2.
- DATA attributes with new Functional Constraints defined only in IEC 61580, Edition 2.
- Usage of DATA item names with lengths that exceed allowed maximum lengths as defined in IEC 61580, Edition 1 (the allowed lengths for miscellaneous names have been extended in IEC 61580, Edition 2). However, this can often be worked around by using a naming convention for those items specified in IET600 (e.g. IEDs, Datasets, Control Blocks) which respects the limits of Edition 1 and should not be a problem in practice.
27.2  IEC 61850 Compatibility Issues

27.2.1  SCL Namespace compatibility Problems.

Several Edition 1 configuration tools have implemented a strict XML checking when importing an SCL file and reject an import, if this strict check fails. Such tools may reject an SCL file with a 2007 XML schema, because it contains elements unknown to the Edition 1 configuration tool.

As the migration of all tools of all vendors to IEC 61580, Edition 2 will take some time, one has to be aware of this issue in retrofit projects.

From Edition 2 (2007A) onwards, it has been defined that a configuration tool must not reject a file based on XML Schema checking only; it must be able to simply skip unknown elements. The intention is to provide backward and forward compatibility at least regarding the SCL files. However, this still may cause engineering problems. E.g. if an IED version 2007B defines IED Services to restrict Dataflow engineering which a System Tool version 2007A does not know, a misconfiguration may result.

27.2.2  Name Length Problems

Edition 2 has extended the length allowed for most names (DATA items, Dataflow items, IED names etc.).

Edition 1 tools can be expected to either reject or misinterpret files with longer names than permitted by Edition 1. This makes e.g. Ethernet Switch models according to IEC/TR 61850-90-4 non-compliant with Edition 1.

27.2.3  Dataflow Compatibility Problems

IEDs which use new Attribute types according to the Edition 2 specification cannot have Edition 1 IEDs as clients, as those clients will not know about the lengths of the new data types and therefore cannot analyze and interpret a Dataset properly.

27.2.4  CDC compatibility problems

Edition 1 of IEC 61850 allowed user-defined Common Data Classes (CDCs). Edition 2 does not allow this anymore. This is a serious forward compatibility break: an Edition 1 IED may declare a user-defined CDC; an Edition 2 IED (or its IED configuration tool) which is configured to receive Dataflow based on such CDCs may refuse import or proper configuration in such a case.

Ethernet Switch models according to IEC/TR 61850-90-4 may use a user-defined CDC "VLN" which makes it non-compliant with Edition 2.
27.3 Mixed Systems, Engineering Scenarios

Below, some engineering scenarios and potential solutions and restrictions will be described:

1. Extension of an existing Substation that was created as IEC 61580, Edition 1 project.
2. Creating a new Substation, which needs to integrate some 3rd-Party IEDs that are not yet available in Edition 2.
3. Engineering a Mixed Substation, where Edition 1 and Edition 2 IEDs are required by the customer.

27.3.1 Extension of an Existing Substation with Edition 1 IEDs

One approach when extending an existing substation is: not to use any Edition 2 IEDs at all, but use Edition 1 IEDs, preferably of the same generation that are already used in the existing substation.

This has several advantages:

- The customer does not need to update all configuration tools and migrate existing substation configurations to a new version.
- New IEDs are identical to old IEDs, the customer does not need to keep IEDs of different versions as spare parts.

Following this approach, an existing IET600 project can be engineered without restrictions.
A customer might already be using Edition 1 IEDs of a certain manufacturer and type in his other substations and wants to use the same ones to keep maintenance and spare parts efficient; but on the other hand he might want to use some new technology available only in Edition 2 IEDs. Therefore to fulfill his specification, a Substation with both Edition 1 and Edition 2 IEDs needs to be engineered.

Another situation which frequently occurs, is a new project where the customer requires some protection IEDs from a 3rd-Party vendor (e.g. Main 1 and Main 2 protection from 2 different vendors; or e.g. Distance protection with remote trip, which requires specific IEDs of the same vendor and type at both ends). In this case, the control IEDs are typically from one vendor and of the same Edition.

Theoretically, it would be possible to configure GCBs (or SVCBs) between Edition 1 and Edition 2 IEDs. In practice, however, GCB engineering requires an SCD file exchange between System Configuration Tool and IED configuration tools of both Edition 1 and Edition 2. Experience shows that in many cases, an Edition 1 IED configuration tool will not be able to import an SCD file with Edition 2 IEDs (which is necessary, if clients are mixed).

To avoid such situations (which are often discovered relatively late in the engineering process), IET600 currently does not support this scenario of freely mixing Edition 1 and Edition 2 IEDs.

The IET600 approach in this case is to separate Edition 1 IEDs and Edition 2 IEDs into different Subnetworks (for engineering details, see chapter 27.5, IET600). This has some restrictions, which need to be taken into account when engineering the Substation:

1. no GCBs can be exchanged between Edition 1 and Edition 2 IEDs (for GCB Exchange, all participating IEDs need to be on the same Subnetwork). Any remote or backup tripping, or interlocking, which requires the participation of IEDs of both Editions will therefore need to be implemented hard-wired. In case of backup tripping, this will typically not be difficult, in case of interlocking, the effort might be considerable.

2. no SVCBs can be exchanged between Edition 1 and Edition 2 IEDs. However, as IEDs capable of Sampled Value Streams are currently rare, and are mostly implemented in new Substations with Edition 2 IEDs, this will in practice rarely lead to problems.

3. No RCBs can be exchanged between Edition 1 and Edition 2 IEDs. This can be often worked around by configuring two communication stacks on the RCB Client (typically an HMI or Gateway) as shown in the picture. One network stack is connected to the Edition 1 Subnetwork and one is connected to the Edition 2 Subnetwork. This is a possible configuration for HMIs or Gateways typically
used by ABB (MicroSCADA with OPC Servers or RTU560s with two different physical client ports).

27.4 IET600 Mixed System Engineering

Basically the main issue for mixed systems is to create SCD files which all concerned tools can import and interpret. To solve this issue, IET600 allows to export several SCD files, either as complete SCD file, or as a subset (all Edition 1 IEDs into one SCD file and all Edition 2 SCDs into one SCD file). This subset SCD export, however, is only possible, if there is no Dataflow exchange between these IEDs; otherwise, the communication cannot be configured correctly.

In IET600, there are two different steps to successfully manage mixed systems:

- during engineering, put IEDs of different versions into different Subnetworks.
- when exporting SCD files, choose the files necessary for successful exchange with the IED configuration tool.

These steps will be explained in detail in the following chapters.

27.4.1 Put IEDs of Different Versions Into Different Subnetworks

When engineering, allocate IEDs of different versions into different Subnetworks. The icon of a Subnetwork indicates how it is classified:

- If the Subnetwork Protocol is not IEC 61850, the Subnetwork icon will be missing (e.g. AA3WF912, an NCC connection with an NCC 104 protocol).
- If all IEDs on the Subnetwork evaluate to Edition 1, the Subnetwork icon ‘E1’ will be shown.
- If all IED on an IEC 61850 subnetwork evaluates to Edition 2, the Subnetwork icon ‘E2’ will be shown.
- If some IEDs on an IEC 61850 subnetwork evaluate to Edition 1 and some to Edition 2, the Subnetwork icon ‘M’ for ‘mixed’ will be shown.
27.4.2 SCD Export

IET600, based on the Subnetwork classification then allows to export different SCD files:

1. an overall SCD file, containing everything in the project.
   This file will contain all IEDs, the full Substation Section and the full communication section.

2. an SCD file of all Edition 1 Subnetworks only (a subset of the overall SCD file).
   Purpose of this file is that Edition 1 IED configuration tools should be able to import this file. Therefore it should contain nothing which is not Edition 1 compatible.
   This file will contain all IEDs in Subnetworks classified as Edition 1, the Data Type Template of those IEDs, the Subnetworks, and the Substation Section (stripped of LNodes referenced by IEDs not contained in the SCD file; and stripped of items which are not Edition-1-compliant, e.g. Functions below a Voltage Level or a Bay, but containing all Voltage Levels and Bays, independent of the exported IEDs).

3. an SCD file of all Edition 2 Subnetworks only (a subset of the overall SCD file).
   Purpose of this file is that Edition 2 IED configuration tools can import and use it. Therefore, it should contain nothing which is not Edition 2 compatible (customer-defined CDCs).
   This file will contain all IEDs in Subnetworks classified as Edition 2 or ‘mixed’, the Data Type Templates of those IEDs, the Subnetworks and the Substation Section (stripped of LNodes referenced by IEDs not contained in the SCD file, but containing all Voltage Levels and Bays, independent of the exported IEDs).

A Switch DATA model is neither Edition 1 nor fully Edition 2 compliant. However, IET600 creates a Switch DATA model for saving configuration information, even if the Switch is in the end communicating with SNMP rather than IEC 61850.

Due to these compatibility issues, a Switch is typically only exported as IED with its Access Points and Connected Access Points (including addressing information). When exporting an Edition 2 file, the user may explicitly check ‘Include Switch Model’ to also export the Switch DATA model. This is currently only useful, if a Switch is actually communicating with IEC 61850 with e.g. the Station HMI for supervision In such cases, ITT600 Explorer may also use the SCD file with the full Switch model for analyzing the communication.

Any partial export will contain only Subnetworks for the IEC 61850 protocol. This is done to ensure maximum compatibility with IED configuration tools. however, to
include non-61850 protocols like SNMP, IEC104 etc., you must export the complete SCD file.

27.4.3 SCD files Version 2007A

IET600 does not provide SCL files according to version 2007A.

This should not normally cause a compatibility problem for SCL file import and export: IEC 61850 demands from version 2007A onwards that an older configuration tool must not refuse the import of a newer SCL version because it does not understand its elements.

However, an IED configuration tool may not be able to display or use newer features; also it may not understand newer DATA objects from the DATA model. If you have such a combination (an IED and its configuration tool of Version 2007A, communicating with IEDs of version 2007B and working with IET600 as a System configuration tool exporting files according to 2007B), verify in advance whether this combination has been tested and is known to work.
27.5 IET600 Versioning of IEC 61850 Items

Due to compatibility issues, as mentioned in chapter 27.2 (IEC 61850 Compatibility Issues), IET600 needs to evaluate the versions of IEC 61850 items. IET600 needs this versions to then put a version to the project and to Subnetworks.

The project version is used to export SCL files accordingly. So an “Edition 1” project will export an SCL file 2003A by default, an “Edition 2.1” project will export an SCL file 2007B.

The Subnetwork version is used in case of mixed projects, to allow the export of partial SCD files which should be Edition 1 or Edition 2 compliant respectively.

The following chapters describe the logic how IET600 determines project and Subnetwork versions.

27.5.1 Versioning Based on IED DATA Namespaces

A project or Subnetwork which has at least one IED with a Version 2007 Revision B will be classified as “Edition 2” or “2007B”.

An IED, in turn is classified as “2007B” by IET600, if:

- the ICD/IID file has an appropriate version/revision (2007A or 2007B)
  e.g. if it is declaring services according to 2007A or 2007B, or using a ServerAt element
- or if the IED declares its DATA model namespace to be 2007A or 2007B, which means that all its communication services are behaving according to the specified version/revision.
  The relevant attribute to determine this is “[first Logical Device].LLN0.NamPlt.IdNs”.

27.5.2 Versioning Based on XML Namespace

A project which has all IEDs with a DATA namespace 2003A, but uses a function below a Bay would also be classified as “Edition 2” or “2007B”.

However, as the gains of using this functionality are minor and the restrictions are considerable (particularly the loss of GCB/SVCB dataflow between Edition 1 and Edition 2 IEDs), any functionality originating in IET600 which requires a higher version/revision, is disabled. To enable it, the user needs to reclassify a project manually as 2007B. When trying to do so, IET600 will issue an appropriate warning and ask the user to confirm the version change.

27.5.3 Versioning of Switch IEDs

IET600 relies on the DATA model of a Switch to save certain configuration information. However, as the Switch model is neither Edition 1 nor fully Edition 2
compatible, its versioning is faked by IET600 to be Edition 1 (to avoid incrementing a project version just because of Switches and creating a mixed system with all its restrictions).
Annex: Inputs Predefined from IED (Late Binding)

28.1 Introduction

While an IED Configuration Tool could always create an Input section (LN -> Inputs -> extRef) with an internal Address (intAddr), there was no standardized way to indicate what information should be delivered to this input. At best, the description (desc) attribute contained some text so that the user could look for the appropriate DATA item in another IED and map it.

In Edition 2007B2, some attributes were added to the SCL schema, so that an IED Configuration Tool could specify the meaning of the expected Input (extRef) in a more precise way. This whole way of working was termed “Late Binding”; i.e. the IED specifies something and later, the System Configuration Tool is obliged to bind the appropriate DATA item from another IED to it.

28.2 Late Binding SCL Attributes

The following attributes can be specified by an IED Configuration Tool:

<table>
<thead>
<tr>
<th>ExtRef Attribute</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>intAddr</td>
<td>must be set to a non-empty value</td>
</tr>
<tr>
<td>pDO</td>
<td>“predefined DATA Object”; specifies the name of the CDC (Common Data Class) of the DO (DATA Object) which is expected.</td>
</tr>
<tr>
<td>pDA</td>
<td>“predefined DATA Attribute”; specifies the name of the DA (DATA attribute) which is expected.</td>
</tr>
<tr>
<td>pLN</td>
<td>“predefined LN”; specifies the name of the LN Class which is expected</td>
</tr>
<tr>
<td>pServT</td>
<td>“predefined Service Type”; specifies the Service Type (RCB, GCB etc.) which is expected to supply the information.</td>
</tr>
</tbody>
</table>

The System Configuration Tool may not delete an Input ExtRef with a non-empty intAddr.
The System Configuration Tool may not add any Input ExtRefs on its own, if Client-Services.NoIctBinding = true; in this case, only Input ExtRefs specified by the IED Configuration Tool may be configured by the System Tool.

The System Tool may not modify any of the p-Attributes, they are owned entirely by the IED Configuration tool (however, it may delete whole ExtRefs, if they do not contain any intAddr).

28.3 Late Binding Workflow

28.3.1 IEC 61850 Intended Workflow

This chapter specifies the workflow as intended in IEC 61850 for Late Binding:

1. The IED Configuration Tool creates ExtRefs for signals it expects as input (for its control or protection logic) and provides attributes for the expected configuration (p-Attributes).
   
   It puts an intAddr to prevent the System Configuration Tool from deleting such ExtRefs.
   
   It configures as a minimum a pDO, optionally a pDA and / or a pLN to specify what type of signal is expected.
   
   It configures a pServT, if it expects the information sent via a particular communication type (e.g. GOOSE).
   
   It may put a user-understandable description into the 'desc' attribute (as the above defined attributes do not allow a complete specification of the meaning of the expected information).

2. It exports its ICD/IID file with those ExtRefs.

3. The System Configuration Tool imports the ICD/IID file.

4. The System Configuration Tool must configure all such ExtRefs which contain an intAddr; the p-Attribute data serve as a guidance what to map.

5. The System Configuration Tool must not modify the intAddr or any p-Attributes, they belong entirely to the IED Configuration Tool.

6. The System Configuration Tool may configure additional Inputs.

7. The System Configuration Tool exports the SCD file.

8. The IED Configuration Tool imports the SCD file and has now all the information needed to configure its communication.

The above sequence can be repeated, if the IED Configuration Tool requests additional ExtRefs (e.g. when changing the protection or control logic).
28.3.2 IEC 61850 Late Binding Mapping Rules

If only a pDO is specified:

- It is mandatory that a DATA item with the same Common Data Class (!) but not necessarily the same DO is mapped. So e.g. if a “Pos” is specified, the mapping of either CSWI.Pos or GGIO.DPCSOxx is allowed as both have the same Common Data Class “Double Point Controllable” (DPC).

This allows some flexibility, if the sender IED uses some alternative way of implementation (e.g. if a Breaker command is implemented as GGIO rather than as CSWI as in the above-mentioned example).

However, it is the responsibility of the user to map a DO with the same meaning. E.g. according to the above rule, if pDO = “Health” is specified, the user would be allowed to map “Beh” of the same LN also. This does not make sense, however: while both have the same Common Data Class (Single Point Status), their meaning is quite different. Therefore, if it is possible to give a DO as specified by the pDO, this should be done; mapping an item with different DO, but same Common Data Class should be a fallback solution only.

- The System Configuration Tool may create ExtRefs for each individual DA of the specified pDO, rather than just specifying the DO. This is actually the preferred way: the receiver IED does not necessarily know what optional DAs the sender implements; therefore specifying Inputs for all the DAs rather than just the DO makes it transparent to the receiver what he will be receiving.

In this case, the p-Attributes and intAddr must be copied to all entries created by the System Configuration Tool.

If a pDO and a pDA are specified:

- The DA of a DO with the same Common Data Class (!) needs to be mapped. Again, it is allowed to use another DO, provided Common Data Class and DA name are the same.

So e.g. if “Pos” and “stVal” are specified, the mapping of either CSWI.Pos.stVal or GGIO.DPCSOxx.stVal is allowed as both have the same Common Data Class “Double Point Controllable” (DPC).

If a pLN and a pDO are specified:

- A DATA item with the same LN Class and same Common Data Class shall be specified.

To use not the specified, but an alternative DO makes sense only in special situations, e.g. if GGIO.DPCSO01 is specified, but from some other information, the user knows that the information with the required meaning is coming from GGIO.DPCSO02.

Other than that, it should not happen that an LN Class has several DOs (with the same Common Data Class) with the same meaning, therefore the common case will be to map a DATA item with the same LN Class and DO as specified in the P-Attributes.

If a pLN and a pDO and a pDA are specified:

- A DATA item with the same LN Class, same Common Data Class and same DA shall be specified.
The same logic as specified for pLN and pDO applies.

28.4 Configuration of Late Binding in the System Configuration Tool

28.4.1 Implementation in IET600

As it is not very clear yet how efficiently this may be used, IET600 implements only the mandatory subset of functionality:

- If p-Attributes are provided, you can map an existing DATA Object or attribute to an ExtRef according to the mapping rules given above.
- IET600 currently does not support a split of an attribute specified on pDO level into several ExtRefs on DA level (although this would be the preferred workflow).
References

# Glossary of Terms

<table>
<thead>
<tr>
<th>Term</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CID-file</td>
<td>Configured IED description. According to IEC 61850, this file transports data from the IED configuration tool to the IED. In the current engineering context, it is also used to designate a file containing a configured IED description to be imported into IET NG. In IEC 61850, V2, there is another file specified for this process, the IID-file.</td>
</tr>
<tr>
<td>DATA</td>
<td>(spelt with capital letters), refers to the IEC 61850 DATA model, i.e. Logical Nodes, DATA objects and DATA attributes. In this document, DATA is sometimes used in a sense that also include Logical Devices and Servers.</td>
</tr>
<tr>
<td>Dataflow</td>
<td>refers to the IEC-61850-specific items that define what data are exchanged (i.e. Datasets) and how they are exchanged; (i.e. all kinds of Control Blocks, like RCBs, GCBs and SVCBs)</td>
</tr>
<tr>
<td>Data Type Template</td>
<td>In many Logical Nodes, there are very similar data repeated again and again. To save space, IEC 61850 specifies not to store such data for each LN individually, but via a rather sophisticated system of references to minimize the length of files. This whole system of references and data is contained in the so-called Data Type Templates.</td>
</tr>
<tr>
<td>GCB</td>
<td>Goose Control Block (as defined in IEC 61850)</td>
</tr>
<tr>
<td>GUI</td>
<td>Graphical User Interface</td>
</tr>
<tr>
<td>HMI</td>
<td>Human-Machine interface</td>
</tr>
<tr>
<td>Term</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>IED</td>
<td>Intelligent Electronic Device, general term in IEC 61850 for everything programmable and electronic. In the sense of IEC 61850, e.g. HMIs or NCC gateways are also IEDs and appear as such in an SCL file.</td>
</tr>
<tr>
<td></td>
<td>In this document, the term ‘IED’ is mostly used in a narrower sense, designating Protection and Control IEDs; HMIs Gateways etc. are labeled as such.</td>
</tr>
<tr>
<td>IED Configuration Tool</td>
<td>a manufacturer- or IED-specific tool that must be able to provide IED-specific settings and generate IED-specific configuration files (ICD-, IID-files). It must be able to import SCD-files.</td>
</tr>
<tr>
<td>ICD file</td>
<td>IED Capability Description file. According to IEC 61850, this file must contain exactly one IED whose capabilities are described:</td>
</tr>
<tr>
<td></td>
<td>• the IED name must be TEMPLATE.</td>
</tr>
<tr>
<td></td>
<td>• the file must contain the needed data type templates inclusive logical node type definitions</td>
</tr>
<tr>
<td></td>
<td>• the file may contain an optional substation section, where the substation name shall be TEMPLATE.</td>
</tr>
<tr>
<td></td>
<td>The substation section of this file may contain Logical Node references. Any substation, in which this IED shall be used, must match an appropriate substation topology part</td>
</tr>
<tr>
<td>IID-file</td>
<td>Instantiated IED Description file (IEC 61850, V2). Describes the project-specific configuration of an IED.</td>
</tr>
<tr>
<td>LN</td>
<td>Logical Node, in IEC 61850 a container for all data related to one specific function (e.g. PTOC for time-delayed overcurrent protection, or CSWI for Switch control). LNs are product-specific.</td>
</tr>
<tr>
<td>Term</td>
<td>Description</td>
</tr>
<tr>
<td>-------</td>
<td>-------------</td>
</tr>
<tr>
<td>LNode</td>
<td>Logical Node Reference in a Substation Tree element. It can either be used as a standalone name to specify some desired functionality in an SSD-file, or it can connect an LN in an IED to an object of the Substation Tree. The latter is helpful to:  &lt;ul&gt;  &lt;li&gt;attach a functionality to a substation object (e.g. a protection function PTOC to a Bay or a control function CSWI to a specific circuit breaker or switch).  &lt;/li&gt;  &lt;li&gt;distinguish functionalities (e.g. Protection Trip from a Main Protection Device or from a Backup Protection device).  &lt;/li&gt;  &lt;li&gt;support automated naming in the HMI according to the Substation Tree.  &lt;/li&gt;  &lt;/ul&gt;</td>
</tr>
<tr>
<td>NCC</td>
<td>Network Control Center, general term for a place that controls substations remotely.</td>
</tr>
<tr>
<td>RCB</td>
<td>Report Control Block (as defined in IEC 61850)</td>
</tr>
<tr>
<td>SCD file</td>
<td>System Configuration Description file. According to IEC 61850, this file contains all IEDs, a communication configuration section and a substation description section.</td>
</tr>
</tbody>
</table>
| SCL file | A file written in System Configuration description Language. SCL is a specific XML namespace defined in IEC 61850.  
This term is typically used to include all more specifically defined files (SCD, SSD, IID etc.). |
| SED file | System Exchange Description file (IEC 61850, Ed.2). A file designed to exchange IEC 61850 data between two different projects/substations/System Configuration tools. |
| Signal | corresponds to one line in an event or alarm list; or one process information.  
IEC 61850 does not really know data points, but object structures. For the purpose of this document and when talking with a customer, a data point is roughly equivalent to a DAI, except that quality and timestamp are separate DAIs in IEC 61850, but included in a data point. |
<table>
<thead>
<tr>
<th>Term</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SSD file</td>
<td>System Specification Description file. According to IEC 61850,</td>
</tr>
<tr>
<td></td>
<td>• the file must contain a substation description section  \</td>
</tr>
<tr>
<td></td>
<td>• the file may contain Logical Node references and, if so, then must contain the needed data type templates and logical node type definitions.</td>
</tr>
<tr>
<td>Substation Tree</td>
<td>A tree with one Substation as its root, containing all the primary equipment for this Substation (e.g. Circuit Breakers, Disconnectors etc.) in a hierarchy defined by IEC 61850 (Voltage Levels, Bays, Functions, Subfunctions).</td>
</tr>
<tr>
<td>SV</td>
<td>Sampled Value (as defined in IEC 61850)</td>
</tr>
<tr>
<td>SVCB</td>
<td>Sampled Value Control Block (as defined in IEC 61850)</td>
</tr>
<tr>
<td>System Configuration Tool</td>
<td>an IED independent system level tool that must be able to import IED-specific configuration files (ICD-, IID-files). It must be able to export SCD-files. It should also be able to read an SSD file. According to IEC 61850, Ed.2, it needs to be able to handle SED-files, also.</td>
</tr>
</tbody>
</table>