# HMI Style Guide and Toolkit

This guide provides simple and practical guidance to plant system Instrumentation and Control (I&C) responsible officers and designers on how to design, develop, procure, operate and maintain an effective plant system operator user interface.

## Approval Process

<table>
<thead>
<tr>
<th>Name</th>
<th>Action</th>
<th>Affiliation</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Author</strong></td>
<td>Utzel N.</td>
<td>15 Feb 2018:signed</td>
</tr>
<tr>
<td><strong>Co-Authors</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Reviewers</strong></td>
<td>Park M.</td>
<td>19 Feb 2018:recommended</td>
</tr>
<tr>
<td></td>
<td>Tarasewicz G.</td>
<td>19 Feb 2018:recommended</td>
</tr>
<tr>
<td><strong>Approver</strong></td>
<td>Wallander A.</td>
<td>22 Feb 2018:approved</td>
</tr>
</tbody>
</table>

## Document Security: Internal Use

- RO: Park Mikyung
- LG: CODAC team
- AD: ITER
- AD: External Collaborators
- AD: IO_Director-General
- AD: EMAB
- AD: OBS - CODAC Section (CDC)
- AD:Auditors
- AD: ITER Management Assessor, project administrator, RO
- AD: OBS - Control System Division (CSD) - EXT
- AD: OBS - CODAC Section (CDC) - EXT
- LG: Cryogenic Section ili...
<table>
<thead>
<tr>
<th>Version</th>
<th>Latest Status</th>
<th>Issue Date</th>
<th>Description of Change</th>
</tr>
</thead>
<tbody>
<tr>
<td>v1.0</td>
<td>Approved</td>
<td>09 Dec 2010</td>
<td></td>
</tr>
<tr>
<td>v1.1</td>
<td>Signed</td>
<td>05 Jan 2011</td>
<td>Internal review comments integrated.</td>
</tr>
<tr>
<td>v1.2</td>
<td>Signed</td>
<td>01 Feb 2011</td>
<td>Footer error fixed</td>
</tr>
<tr>
<td>v2.0</td>
<td>Approved</td>
<td>10 Feb 2011</td>
<td>Reference to ITER Process for Human Machine Interface (HMI) Development (ITER D 3T9UK2)</td>
</tr>
<tr>
<td>v2.1</td>
<td>Approved</td>
<td>01 Feb 2013</td>
<td>PCDH V7 schema</td>
</tr>
<tr>
<td>v3.0</td>
<td>Signed</td>
<td>30 Jan 2015</td>
<td>This document sets out the designs for human-machine interfaces (HMI) that ITER personnel will use to monitor and control the ITER systems during phases of commissioning, operation, and maintenance. The ITER Human Factors Integration Plan (IHFIP) identified the need for a HMI Development Process that included this HMI design guide to achieve high user reliability, efficiency, and effectiveness by enforcing high standards for human-system interface design consistently across ITER.</td>
</tr>
<tr>
<td>v3.1</td>
<td>Disapproved</td>
<td>04 Feb 2015</td>
<td>Implemented the comment from Nadine on missing Pop-up boxes images.</td>
</tr>
<tr>
<td>v3.2</td>
<td>Approved</td>
<td>18 Feb 2015</td>
<td>Support of ITER Human Factor requirements for HMI development (QEDG6L)</td>
</tr>
<tr>
<td>v3.3</td>
<td>Approved</td>
<td>13 Jul 2015</td>
<td>Added requirements for area dimensions, font sizes and trace colours. New navigation pane. User input validation and feedback. OPI validation tool.</td>
</tr>
<tr>
<td>v3.4</td>
<td>Signed</td>
<td>09 Feb 2016</td>
<td>CODAC Core System 5.2 integration. New alarm pane and alarm coding. New sections about alarms. Update of the user input section. New section about additional symbol information.</td>
</tr>
<tr>
<td>v3.5</td>
<td>Approved</td>
<td>10 Feb 2016</td>
<td>Missing section regarding the symbol animation</td>
</tr>
<tr>
<td>v3.6</td>
<td>Signed</td>
<td>21 Jun 2016</td>
<td>New features of 5.3 release including the alarms list, LED standardisation and standard faceplate for analog and digital input and output</td>
</tr>
<tr>
<td>v3.7</td>
<td>Approved</td>
<td>21 Jun 2016</td>
<td>PDF version generated with Office</td>
</tr>
<tr>
<td>v3.8</td>
<td>Approved</td>
<td>15 Feb 2018</td>
<td>Updated for 6.0 release</td>
</tr>
</tbody>
</table>
3.11 Faceplates..............................................................35
3.12 LED .................................................................36
3.13 Alarms.................................................................37
3.14 Alarms in mimics ................................................37
3.15 Additional symbol information (optional) ............38
3.16 Scripting.............................................................40
4 Operator User Interface Validation..........................40
1 Introduction

1.1 PCDH Context

The Plant Control Design Handbook (PCDH) [RD1] defines methodology, standards, specifications and interfaces applicable to ITER plant systems Instrumentation & Control (I&C) system life cycle. I&C standards are essential for ITER to:

- Integrate all plant systems into one integrated control system.
- Maintain all plant systems after delivery acceptance.
- Contain cost by economy of scale.

PCDH comprises a core document which presents the plant system I&C life cycle and recaps the main rules to be applied to the plant system I&Cs for conventional controls, interlocks and safety controls. Some I&C topics will be explained in greater detail in dedicated documents associated with PCDH as presented in Figure 1-1. This document is one of them.

PCDH core and satellite documents: v7

![Figure 1-1 Schema of PCDH documents](image)

1.2 Document Scope

This guide provides simple and practical guidance to plant system Instrumentation and Control (I&C) responsible officers and designers on how to design, develop, procure, operate and maintain an effective plant system operator user interface.

1.3 Related documents

- [RD1] Plant Control Design Handbook (PCDH). (27LH2v)
- [RD2] ITER Process for Human Machine Interface (HMI) Development (3T9UK2)
- [RD3] ITER Human Factor requirements for HMI development (QEDG6L)
- [RD4] CODAC Core System User Manual (43PSH9)
1.4 Definitions
Definitions as represented on Figure 1-2 include:
- VDU workstation or “station” is composed of at least 3 VDUs.
- VDU or “monitor” 16/9 has a resolution of 3840 x 2160 - diagonally viewable size of 24 inches.
- Full screen display canvas within CS-Studio environment has a resolution of 3830 x 2080.
- A mimic resolution within the display canvas is limited to 3236 x 1760.
- A faceplate resolution within the display canvas is limited to 575 x 943.

![Figure 1-2 HMI basic definitions](image)

1.5 HMI Style Guide
HMI Style Guide includes general design principles for the displays derived from [RD3]. It describes the displays layout and organisation.

1.6 HMI Toolkit
An HMI toolkit\(^1\) is delivered with CODAC Core System to edit and run displays used to control and monitor systems on the ITER site during and throughout commissioning, operation and maintenance phases. The toolkit is based on CS-Studio (CSS) set of tools that include an operator interface (BOY), an alarm system (BEAST) and an archive system (BEAUTY).

The HMI toolkit provides standardised colours and fonts, a collection of graphical objects – widgets – that the user can drag and drop from the palette to the display canvas, trend widgets, an industrial symbol library (electrical breaker, valve, pump…) and many templates.

\(^1\) This document refers to the HMI toolkit 4K/Quad HD version. Please consult [RD4] to learn how to switch from Full HD to 4K definition.
2 Operator User Interface Layout

The entire screen area visible to the user is the ‘display canvas’ and has the following specific zones:

- Status Bar
- Main Display Area - mimic
- Alarm Pane
- Control Area / Faceplate Pane
- Navigation Pane

Figure 2-1, shows the display canvas layout running on one-terminal using CS-Studio BOY.

<table>
<thead>
<tr>
<th>Area Title</th>
<th>Subarea Title</th>
<th>Width[%]</th>
<th>Height[%]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Status Bar</td>
<td>Global Statuses</td>
<td>30</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>Mimic Title</td>
<td>55</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>UTC Time</td>
<td>15</td>
<td>5</td>
</tr>
<tr>
<td>Mimic Area</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Navigation Area</td>
<td>General Navigation</td>
<td>20</td>
<td>10</td>
</tr>
<tr>
<td></td>
<td>Mimic Navigation</td>
<td>65</td>
<td>10</td>
</tr>
<tr>
<td>Alarm Pane</td>
<td></td>
<td>15</td>
<td>50</td>
</tr>
<tr>
<td>Faceplate Area</td>
<td></td>
<td>15</td>
<td>45</td>
</tr>
</tbody>
</table>

With the HMI toolkit provided by CODAC Core System, some extra space is required for the window, view and border management. The available space for each zone is presented on Figure 2-2.
2.2 User Zones

In the display canvas, some zones are provided by the CODAC HMI toolkit such as the status bar, the navigation area, the alarm pane and the control pane. The user on the other hand has to develop the main display area by providing mimics loaded by the navigation pane and design faceplates to be displayed in the control pane as illustrated on Figure 2-3.
2.3 Project Organisation

In the development phase the mimics and faceplates have to be developed under dedicated folders in `<project_name>/src/main/boy` as illustrated on Figure 2-4.

Figure 2-4 Project folder organisation

Explanation of the main folders is the following:

<table>
<thead>
<tr>
<th>Project Structure</th>
<th>Explanation</th>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>boy</td>
<td>Shortcut to src/main/boy</td>
<td>Alarm pane</td>
</tr>
<tr>
<td>src</td>
<td></td>
<td></td>
</tr>
<tr>
<td>main</td>
<td></td>
<td></td>
</tr>
<tr>
<td>beast</td>
<td>Alarm configuration</td>
<td>Alarm server / alarm pane</td>
</tr>
<tr>
<td>beauty</td>
<td>Archive configuration</td>
<td>Archive engine / plotter</td>
</tr>
<tr>
<td>boy</td>
<td>Operator Interface</td>
<td></td>
</tr>
<tr>
<td>SymbolLibrary</td>
<td>Shortcut to the HMI toolkit</td>
<td>Mimics, alarm pages and faceplate</td>
</tr>
<tr>
<td>alarmpages</td>
<td>Alarm pages</td>
<td>Alarm pane</td>
</tr>
<tr>
<td>faceplates</td>
<td>Faceplates</td>
<td>Control pane</td>
</tr>
<tr>
<td>mimics</td>
<td>Mimics</td>
<td></td>
</tr>
<tr>
<td>navigation</td>
<td>Navigation configuration files</td>
<td>Navigation pane</td>
</tr>
<tr>
<td>templates</td>
<td>Shortcut to the HMI toolkit</td>
<td></td>
</tr>
<tr>
<td>ITER.opi</td>
<td>Shortcut to the HMI toolkit</td>
<td>To run the operator interface</td>
</tr>
<tr>
<td>ITERAlarm.opi</td>
<td>Shortcut to the HMI toolkit</td>
<td></td>
</tr>
</tbody>
</table>
2.4 Status Bar

The primary purpose of the Status Bar is to provide the user with a high-level overview of ITER’s overall status and to “set the context” for the station. The status bar includes:

- Global Statuses (CODAC, Central Safety System, Central Interlock System…)
- Mimic Title
- Coordinated Universal Time (UTC) 24-hour time with seconds
- ITER logo

Figure 2-5, shows an implementation of the status bar.

Global statuses and UTC time are provided by CODAC HMI toolkit and integrated automatically in the display canvas.

The title of the mimic is retrieved from the navigation XML configuration file using the provided description tag converted to UPPER case as illustrated on Figure 2-6.

It is also possible the setup the title of the mimic using the macro TITLE when configuring an action button to open a mimic directly.

Note: macros are loaded when the display canvas is opened. So any macro changes after that are not reflected immediately.
2.5 Main Display Area - Mimic

Mimics present representations of the ITER plant systems, sub-systems and any other data and information necessary for user tasks. Mimics provide some control functions that allow users to interact with the plant systems.

2.5.1 Auxiliaries Status

On the top left of the mimic, space is allocated to represent the status of auxiliaries of this particular level. Figure 2-7, shows an implementation of the plant system auxiliaries status.

A status includes usually a LED widget that shows the status of the auxiliary:

- healthy
- not healthy minor
- not healthy major

And a Label widget that provides the auxiliary text information using “IO Label” font. It is recommended to group the LED and Label widgets in a container as illustrated on Figure 2-8.

---

2 The fonts “IO Label” and “IO Normal” are equivalent. “IO Label” is the default one and “IO Normal” is maintained for compatibility reason.
2.5.2 Mimic

Each mimic is a standalone OPI screen linked into the display canvas as illustrated on Figure 2-9. By
convention, the name of the mimic OPI is based on the CBS structure as follow: [ITER-]CBS1]-
[CBS2]-[CBS3]_Mimic.opi

The size of the mimic is 3236 x 1760.

The mimic itself has to be organised in different layers as illustrated on Figure 2-10.

- Information layer

Dynamic data shall be presented in the information layer and shall be easily distinguishable from the
background and static data layers. This layer shall have a show/hide function for information that is not
always needed for operator tasks e.g. component labels and to de-clutter mimics. Take the
templates/Empty_mimic.opi as an example on how to define the show/hide tick box and the rule defined
on the Information Layer container as illustrated on Figure 2-11.
- **Priority layer**
  Alarm information shall be presented in a priority layer where any changes are immediately brought to the operator's attention. This layer may also be used for other more urgent or higher-priority information e.g. plant equipment is out for maintenance.

- **Static data layer:**
  The static data shall enable the operator to interpret the meaning of the displayed information (e.g. showing the geography and/or structure of a process).

Finally, the check box to show/hide the Information layer has to be added to the mimics on the top right corner. Figure 2-12 provides an overview on how the mimic is organised in containers with the definition of the tick box.

![Figure 2-12 Check box widget to show/hide the information layer](image-url)
2.6 Alarm Pane

The alarm pane reflects the alarm status of the controlled system displayed on the mimic. It is a standard component provided by CODAC HMI toolkit. Figure 2-13 shows the integration of the alarm pane at runtime:

- Each alarm is presented in a single row
- To the left of each alarm there is an acknowledgement tick box
- Each alarm states (left to right):
  - Alarm priority symbol
  - Alarm description
  - Time
- The background colour of the alarm message reflects the priority of the alarm
- The colour of the alarm priority symbol reflects the priority of the alarm
- The alarm priority symbol flashes at 2Hz until acknowledged by the operator

![Figure 2-13 Alarm pane integration in the canvas](image1)

Figure 2-13 Alarm pane integration in the canvas

Figure 2-14 documents the priority icons and alarm message background and foreground colours.

<table>
<thead>
<tr>
<th>Icons</th>
<th>Shelved alarms*</th>
<th>(R, G, B)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Invalid/Unknown alarm</td>
<td>⚠️</td>
<td>⚠️</td>
</tr>
<tr>
<td>&quot; recovered</td>
<td>⚠️</td>
<td></td>
</tr>
<tr>
<td>&quot; acknowledged</td>
<td>⚠️</td>
<td></td>
</tr>
<tr>
<td>Major alarm</td>
<td>⚠️</td>
<td></td>
</tr>
<tr>
<td>&quot; recovered</td>
<td>⚠️</td>
<td></td>
</tr>
<tr>
<td>&quot; acknowledged</td>
<td>⚠️</td>
<td></td>
</tr>
<tr>
<td>Minor alarm</td>
<td>⚠️</td>
<td></td>
</tr>
<tr>
<td>&quot; recovered</td>
<td>⚠️</td>
<td></td>
</tr>
<tr>
<td>&quot; acknowledged</td>
<td>⚠️</td>
<td></td>
</tr>
</tbody>
</table>

* Shelved – the operator has prevented a nuisance alarm from being displayed for a limited time

![Figure 2-14 Alarm coding](image2)

Figure 2-14 Alarm coding
The alarm pane displayed a reduced list of alarms up to 20 based on a filter specified in the navigation configuration file with the tag `alarm_root` as illustrated on Figure 2-15.

```
<xml version="1.0" encoding="UTF-8" standalone="no"/>
<!DOCTYPE BOY Files Navigation CODAC-VERSION: 5.2.0 WARNING: This is an SDD generated file. Contact codac-support for any query. -->

<CBS name="AG91" description="B20 High Voltage (HV)" enabled="true" opl_file="/UTIL-S15/boy/ITER-UTIL-S15-AG91.opi LEVEL=ITER-UTIL-S15-AG91" deprecated="false" alarm_root="/UTIL/UTIL-S15/UTIL-S15-AG91"
xmns:xi="http://www.w3.org/2001/XInclude">
</CBS>
```

Figure 2-15 Alarm pane filter specification

It is also possible the setup the alarm filter using the macro `ALARM_ROOT` when configuring an action button to open a mimic directly.

### 2.6.1 Alarm Page

The function of the Alarm Pane is to attract immediate attention of the user to the specific alarm. Making a right-click on an alarm from the alarm pane and selecting the related display, will open a dedicated screen for alarm handling and investigation.

Figure 2-16 presents an alarm page with the alarm description and procedure, including all relevant data for analysis using trend widgets.

By convention, the name of the alarm page OPI is based on the CBS structure as follow: `ITER-{CBS1}[-{CBS2}][-{CBS3}]-{ALARM_PV}_AlarmPage.opi`. In the case of an alarm page common to a group of alarm PVs then the name of the alarm page OPI is as follow:

`ITER-{CBS1}[-{CBS2}][-{CBS3}]-{ALARM}_AlarmPage.opi`

where `{ALARM}` represents a generic name for the alarm. Example of the control voltage defect alarm page name: `ITER-UTIL-S15-BG07-Control_Voltage_Defect_AlarmPage.opi`. 
The action to open an alarm page is defined in the BEAST alarm configuration xml file as illustrated on Figure 2-17.

The control pane is a standard component provided by CODAC HMI toolkit. Figure 2-18 shows the integration of the control pane at runtime. When the user clicks on a valve symbol its associated faceplate is displayed.

Each controllable component on a mimic shall have a ‘faceplate’ to provide information about the component and options to control it. Faceplates are designed using a tab container with:

- **Status** tab: read-only fields providing information on the status of the component
- **Trends** tab: trend widgets displaying historic and live data
- **Controls** tab: input fields controlling the component
- **Help** tab: containing information about the specific component. Links to datasheet, documentation and information about the physical location of the component are available from this tab.
Some faceplates have been made available under the toolkit templates folder, as reported on Figure 2-19.

**Figure 2-19 Faceplate**

AnalogComponent.opi tabs – Status, Trends, Controls, Help

In particular, standard faceplates are provided for analog and digital input and output in order to display PV and alarm details as shown on Figure 2-20.

**Figure 2-20 Analog and Digital Input Faceplates**

Scripts are also provided to illustrate user input validation and submission to the process.
Fonts to be used within the faceplate are:

<table>
<thead>
<tr>
<th>Subarea</th>
<th>Font Family</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name of the component</td>
<td>IO Label</td>
</tr>
<tr>
<td>Title of the Tab</td>
<td>IO Subtitle</td>
</tr>
<tr>
<td>Title of Container (Grouping Container)</td>
<td>IO Title</td>
</tr>
<tr>
<td>Values and Label</td>
<td>IO Label</td>
</tr>
<tr>
<td>Input Box</td>
<td>IO Label</td>
</tr>
<tr>
<td>Text in Buttons</td>
<td>IO Label</td>
</tr>
<tr>
<td>Trend Title</td>
<td>IO SubTitle</td>
</tr>
<tr>
<td>Axis values in Trends</td>
<td>IO Scale 3</td>
</tr>
<tr>
<td>Axis Title in Trends</td>
<td>IO Label 3</td>
</tr>
</tbody>
</table>

Table 1: Font style used in faceplates

Then, an action is defined on the component widget within the mimic, which reacts to user click to load a specified faceplate and to instantiate the component macro as illustrated on Figure 2-21.

```
then, an action is defined on the component widget within the mimic, which reacts to user click to load an specified faceplate and to instantiate the component macro as illustrated on Figure 2-21.
```

```
loc://$(DID)_FACEPLATE(""
```

This action writes in a predefined variable the relative path to the faceplate OPI and specify the component name using the macro.

```
loc://$(DID)_FACEPLATE(""
```

```
loc://$(DID)_FACEPLATE(""
```

```
loc://$(DID)_FACEPLATE(""
```
In the previous example, writing the value 
\texttt{../faceplates/TecSystem.opi COMPONENT=UTIL-S15-AG07:MUT9} into the PV name \texttt{loc://$(DID) FACEPLATE("")} will ask the control pane to load 
\texttt{../faceplates/TecSystem.opi} and instantiate the macro \texttt{${COMPONENT}} with the name of the component \texttt{UTIL-S15-AG07:MUT9}. In addition, the widget named \texttt{UTIL-S15-AG07:MUT9} will be highlighted on the screen.

Finally the macro \texttt{${COMPONENT}} is used in the faceplate as illustrated on Figure 2-22 as faceplate title and as a generic part of the PV names.

![Figure 2-22 Faceplate title and PV name using predefined macro $(COMPONENT)](image)
2.8 Navigation Pane

The mimics for plant system users are arranged in a hierarchical structure to allow the user to navigate through the plant system to specific mimics based on the Control Breakdown Structure (CBS), and this in 5 clicks maximum.

The navigation pane is part of CODAC HMI toolkit and is divided into two areas:

- General Navigation area with the current location of the mimic and home/up buttons
- Mimic Navigation which contains the link to other relevant displays.

The foreground colour of the navigation buttons is alarm sensitive as illustrated on Figure 2-23. The navigation buttons flash at 2Hz if a new alarm is active on a specific CBS level and need to be acknowledged by the operator. In the following example there are major alarms on UTIL → S15 → BG07 and BG01.

Hovering the mouse over a navigation button, the tooltip will show the number of latched alarms on the corresponding alarm tree node. Making a right-click on a navigation button will allow the operator to acknowledge directly the alarm node and subsequent alarms as illustrated on Figure 2-24.

The configuration of the navigation is done through XML files under the navigation folder as illustrated on Figure 2-25. By convention, the navigation XML file name is based on the CBS structure: Navigation_[CBS1]_[CBS2]_[CBS3]].xml.
The whole navigation is a tree composed of the inclusion of the different XML files. Looking at UTIL-S15 example as reproduced on Figure 2-26, the XML syntax is quite simple and defines:

- **name** tag: the navigation button label
- **description** tag: the tooltip displayed when hovering the mouse over the navigation button. The description is also used to set mimic title in UPPER case
- **enabled** tag: button enable (true) or disable (false) property
- **opi_file** tag: mimics to be loaded by the navigation pane. A macro LEVEL is also specify as part of the opi_file tag which gives the hierarchy of the button and is used in the general navigation area. Other user macros can be specified here
- **alarm_root** tag: alarm filter for the alarm pane

Then all sub-levels of the navigation are included and follow the same syntax.

```
<ITER opi *=Navigation_UTIL-S15-AG91.xml *=Navigation_UTIL-S15.xml !=
  <!-- BOY Files Navigation -->
</CBS>

<xi:include href="/UTIL-S15/boy/navigation/Navigation_UTIL-S15-0000.xml">
  <xi:fallback />
</xi:include>

<xi:include href="/UTIL-S15/boy/navigation/Navigation_UTIL-S15-AG07.xml">
  <xi:fallback />
</xi:include>

<xi:include href="/UTIL-S15/boy/navigation/Navigation_UTIL-S15-BG07.xml">
  <xi:fallback />
</xi:include>

<xi:include href="/UTIL-S15/boy/navigation/Navigation_UTIL-S15-AG91.xml">
  <xi:fallback />
</xi:include>

<xi:include href="/UTIL-S15/boy/navigation/Navigation_UTIL-S15-BG91.xml">
  <xi:fallback />
</xi:include>

<xi:include href="/UTIL-S15/boy/navigation/Navigation_UTIL-S15-AG92.xml">
  <xi:fallback />
</xi:include>

<xi:include href="/UTIL-S15/boy/navigation/Navigation_UTIL-S15-BG92.xml">
  <xi:fallback />
</xi:include>
```

Figure 2-26 Navigation XML syntax

They will be displayed in the mimic navigation part following the provided order.
The following screenshots report examples of the navigation principles.

![ITER machine overview (CBS0)](image)

**Figure 2-27** ITER machine overview (CBS0)

- **Current position**: Home
- **Delta buttons**: CBS1
- **Alarms filtered for**: UTIL
- **CBS2 buttons**

---

3 Some navigation buttons are greyed out because they are currently not available. The mimics shown in the pictures are for illustrative purposes only.
The navigation pane supports CBS4 and CBS5.

2.8.1 OPIs Map

An automatic CBS map is made available via a right-click on any navigation button and using the option Actions -> Open in a new tab the OPIs Map. The scrollable map as shown on Figure 2-30 lists the breakdown of the user interface and by clicking on any dropdown button, it is possible to open directly the related screen.
### 2.9 Navigation Within Mimics

Even though the navigation pane is the main navigation tool, it is possible to add within a mimic navigation button as illustrated on Figure 2-31.

![Image 2-31 Navigation integrated in a mimic using the dedicated symbol](image)

These buttons could be used to browse North, South, East or West as “extensions” of the current mimic, i.e. the level and alarm filter remain the same.

![Image 2-32 Panoramic navigation buttons](image)

To load a specific mimic, the action Open OPI has to be configured as illustrated on Figure 2-33 with the file path set to `../ITER.opi` and the macro `MIMIC_FILE` set to the mimic file without extension.

![Image 2-33 Open OPI action to load a specific mimic](image)

If needed additional macros can be specified. In particular, if the mimic to be loaded belongs to another CBS level, then `ALARM_ROOT` and `LEVEL` have to be set in addition to `TITLE`.
2.10 Alarms List

The Alarms List provides the operator with a list of all latched alarms and the functionality to navigate and view additional information about alarms.

*A latched alarm is the first highest occurrence of an un-acknowledged alarm. If an alarm goes from minor to major state, the latched alarm will correspond to the major state, time and value. If an alarm flickers between cleared and alarm states, the latched alarm will correspond to the first time the alarm was triggered.*

Each entry in an alarm list shows:

- Tick box to acknowledge the alarm,
- Alarm priority symbol,
- Alarm description,
- Date and time,
- Current severity and status. This information could be [OK – NO ALARM] if the condition that triggered the alarm has returned to normal. Such cleared alarm will remain in the list until acknowledged by the operator (latched behaviour) but will be displayed in reverse video,
- Alarm PV,
- Value of the setting transgressed,
- Alarm ID.

The background colour of the alarm message reflects the priority of the latched alarm and the alarm priority symbol flashes at 2Hz until acknowledged by the operator.

The alarms list as shown on Figure 2-34 is made available via a right-click on any navigation button and using the option **Actions** -> **Open in a new window the Alarms List**.
2.11 SDD Automatically Generated Operator Interface with Navigation

For an I&C project, SDD is preparing the production of the operator interface by generating for each CBS level the canvas and an empty mimic under `src/main/boy` that can be imported in cs-studio for edition as shown in Figure 2-35.

![Figure 2-35](image)

The navigation based on the CBS breakdown is also generated automatically under `src/main/boy/navigation`. With a navigation configuration file for each CBS level which can be edited and extended by the user as shown in Figure 2-36.

![Figure 2-36](image)

Finally, SDD is generating an alarm page canvas `ITER-{ALARM_PV}_Alarm.opi` and its editable content `ITER-{ALARM_PV}_AlarmPage.opi` under `src/main/boy/alarmpages`. The action to open it from the alarm pane or the alarms list via a right-click on the alarm message is automatically integrated in the alarm configuration.
3 Operator User Interface Detailed Design

The HMI toolkit includes fonts and colours definition, templates and examples of all necessary graphic widgets and elements to implement an operator user interface that meets the style guide requirements. During the detailed design, the developer will drag and drop widgets from the palette to the canvas, adjust few properties such as the process variable name and keep the standardised settings in order to achieve a consistent look and feel of the interface.

3.1 Fonts

Styles are predefined and cannot be adjusted by the user. Most of the widgets come with the font already configured. Styles that are permitted for the different types of information to be presented to the user are the following and are illustrated on Figure 3-1:

- **IO Title** and **IO Subtitle** for headers and critical operational information,
- **IO Label** for operational information (e.g. labels, numerical data) presented in mimics and faceplates. This the default font for most of the widgets,
- **IO Scale** for graph axis.

Figure 3-1 Fonts

3.2 Colours

Colours are predefined and cannot be adjusted by the user. Most of the widgets come with the background and foreground colours already configured. As illustrated on Figure 3-2, the main colours are the following:

- **IO Background** colour for the mimics
- **IO Foreground** colour for the text and line
- **IO Invalid Level Alarm** and **IO High Level Alarm** colours used in conjunction with the ‘background alarm sensitive = yes’ property
- **IO PV OFF** and **IO PV ON** colours for symbol in 0, 1 or more than 1 position.

---

4 The Menu bar, alarm pane, control area and navigation pane use a different background colour to enlighten them. This “IO Area Background” colour shall not be used for another purpose.
3.3 Symbols

Symbols are predefined and cannot be redesigned. Browse the installed library to select the required component when inserting a Boolean or multistate symbol on the canvas.

Screens describing the library are provided in the HMI toolkit as illustrated on Figure 3-3 as well as electrical and PID diagram examples - Figure 3-4 and Figure 3-5.
3.4 Symbol Animation

The enumerated component state is defined as:

- **2-way Valve state**
  - Closed = 0
  - Open = 1
  - Opening = 2
  - Closing = 3
  - Half-open = 4

- **3-way Valve state**
  - Closed = 0
  - Open A = 1
  - Opening A = 2
  - Open B = 3
  - Opening B = 4
  - Open A+B = 5
  - Opening A+B = 6
  - Closing = 7

- **Pump state**
  - Stopped = 0
  - Running = 1
  - Starting = 2
  - Stopping = 3
  - Travel = 4

By convention, the svg file name is “TTT full_name index.svg” with index [0..7] representing the state.
<table>
<thead>
<tr>
<th>State</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 – CLOSED</td>
<td>Symbol – Black Symbol fill – White</td>
</tr>
<tr>
<td>1 – OPEN</td>
<td>Symbol – Black Symbol fill – Dark grey as per the position the component is moving to Flash rate of symbol: 2Hz until movement is complete</td>
</tr>
<tr>
<td>2 – OPENING (flash)</td>
<td>Symbol – Black Symbol fill – Dark grey as per the position the component is changing to Flash rate of symbol: 2Hz until movement is complete</td>
</tr>
<tr>
<td>3 – CLOSING (flash)</td>
<td>Symbol – Black Symbol fill – left side dark grey and right side white.</td>
</tr>
<tr>
<td>4 – HALF-OPEN</td>
<td>Symbol – Black Symbol fill – left side dark grey and right side white.</td>
</tr>
</tbody>
</table>

Figure 3-6 “VC – On-Off 2-way Pneumatic Valve” example

<table>
<thead>
<tr>
<th>State</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 – CLOSED</td>
<td>Symbol – Black Symbol fill – White</td>
</tr>
<tr>
<td>1 – OPEN A</td>
<td>Symbol fill – Dark grey on the left Flash rate of symbol: 2Hz until movement is complete</td>
</tr>
<tr>
<td>2 – OPENING A</td>
<td>Symbol fill – Dark grey on the right Flash rate of symbol: 2Hz until movement is complete</td>
</tr>
<tr>
<td>3 – OPEN B</td>
<td>Symbol fill – Dark grey on the right Flash rate of symbol: 2Hz until movement is complete</td>
</tr>
<tr>
<td>4 – OPENING B</td>
<td>Symbol – Black Symbol fill – Dark grey Flash rate of symbol: 2Hz until movement is complete</td>
</tr>
<tr>
<td>5 – OPEN A+B</td>
<td>Symbol – Black Symbol fill – Dark grey Flash rate of symbol: 2Hz until movement is complete</td>
</tr>
<tr>
<td>6 – OPENING A+B</td>
<td>Symbol – Black Symbol fill – Dark grey Flash rate of symbol: 2Hz until movement is complete</td>
</tr>
<tr>
<td>7 – CLOSING</td>
<td>Symbol – Black Symbol fill – Dark grey Flash rate of symbol: 2Hz until movement is complete</td>
</tr>
</tbody>
</table>

Figure 3-7 “VG –On-Off 3-way Bottom Inlet Valve” example

<table>
<thead>
<tr>
<th>State</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 – CLOSED</td>
<td>Symbol – Black Symbol fill – White</td>
</tr>
<tr>
<td>1 – OPEN A</td>
<td>Symbol fill – Dark grey on the left Flash rate of symbol: 2Hz until movement is complete</td>
</tr>
<tr>
<td>2 – OPENING A</td>
<td>Symbol fill – Dark grey on the line Flash rate of symbol: 2Hz until movement is complete</td>
</tr>
<tr>
<td>3 – OPEN B</td>
<td>Symbol fill – Dark grey on the line Flash rate of symbol: 2Hz until movement is complete</td>
</tr>
<tr>
<td>4 – OPENING B</td>
<td>Symbol – Black Symbol fill – Dark grey Flash rate of symbol: 2Hz until movement is complete</td>
</tr>
<tr>
<td>5 – OPEN A+B</td>
<td>Symbol – Black Symbol fill – Dark grey Flash rate of symbol: 2Hz until movement is complete</td>
</tr>
<tr>
<td>6 – OPENING A+B</td>
<td>Symbol – Black Symbol fill – Dark grey Flash rate of symbol: 2Hz until movement is complete</td>
</tr>
<tr>
<td>7 – CLOSING</td>
<td>Symbol – Black Symbol fill – Dark grey Flash rate of symbol: 2Hz until movement is complete</td>
</tr>
</tbody>
</table>

Note that equipment not monitored by the control system should not be shown on operator HMIs. However, there may be instances when there is the operational need to have these components presented.

Figure 3-8 “VG –On-Off 3-way Side Inlet Valve” example

Figure 3-9 “VG Non-Monitored Valve” example
3.5 Measurement Units and Precision
Measurement units are standardised. Widgets used to display process variable value, retrieve and show the unit as well as the precision from the EPICS PV. No measurement unit and no precision should be hardcoded in the HMI.

Figure 3-11 Measurement unit example

3.6 Labelling
Label widgets are predefined to be left-justified.

Figure 3-12 Labelling example

3.7 Mimics
Information may be presented on mimics in the following formats:

- Schematic diagrams as illustrated on Figure 3-13 – cf. previous Figure 3-4 and Figure 3-5
- Isometric diagrams using an image widget to display a SVG image in the background as illustrated on Figure 3-14
- Lists using label widgets
- Tables designed using containers with ridged border style as illustrated on Figure 3-16
- Fields using text update widget for read-only field and text input for entry field. Each widget has a different representation as illustrated on Figure 3-17
- Graphs
Figure 3-13 Schematics diagram example

Figure 3-14 Isometric diagram example
Lists are suitable for presenting a series of related items or user instructions. The following requirements apply to the presentation of lists on menus:
- Lists shall be organised in a logical order for the user's task.
- Where multiple lists are needed, these shall be visually distinct from one another.
- All text lists shall be left justified and read left to right.
- Numeric lists shall be right justified.
- Listed items that are numerically labelled shall start at "1", not "0".

Figure 3-15 List example

Tables are suitable for tasks that require detailed comparisons of ordered sets of data. The information most relevant to the user or with the highest priority shall be displayed in the leftmost column, and associated but less significant material in columns further to the right.

Figure 3-16 Table example
The following requirements apply to the presentation of fields on panels:
- Entry fields shall be clearly indicated with a box outline.
- Entry fields shall be visually distinct from read-only fields. Entry fields shall be white with a black border and when the user hovers over it the cursor shall change from an arrow to an 'I'. Read-only fields shall have no box or border around them and the cursor should remain as an arrow when the user hovers over it.
- It is possible to attach a tooltip to fields - by default, the tooltip displays the FV name, type, value, timestamp and severity.

Figure 3-17 Fields example

A graph is a display that represents the variation of a variable in comparison with that of one or more other variables.
- Graphs shall convey enough information to allow the user to interpret the data without referring to additional information sources.
- When multiple curves are included in a single graph, each curve shall be identified in a legend.
- Legend shall be used when multiple functions are displayed in a single graph, particularly if curves approach and/or intersect one another.
- Line colour shall be used consistently across graphs.
- In graphs that display multiple curves, if one curve represents data of particular significance, then that curve should be highlighted. If one curve represents critical/important data, for example, that curve might be displayed with a nationally thicker line stroke or in a different colour. This colouring shall be distinguishable from any other coding already used in the graph.
- When the user next compares data represented by separate curves, the curves shall be displayed in one combined graph.

Figure 3-18 Graphs example up to 8 traces
Trace colour shall use one of the eight predefined trace colours – IO Trace 1 to IO Trace 8, as illustrated on Figure 3-19.

![Figure 3-19 Traces coding](image)

By default, an XY Graph has a title and shows the legend as represented on Figure 3-20.

![Figure 3-20 XY Graph widget standard Layout](image)

Fonts to be used within the XY Graph are:

<table>
<thead>
<tr>
<th>Subarea</th>
<th>Font Family</th>
</tr>
</thead>
<tbody>
<tr>
<td>Title</td>
<td>IO Subtitle</td>
</tr>
<tr>
<td>Axis values in Trends</td>
<td>IO Scale</td>
</tr>
<tr>
<td>Axis Title in Trends</td>
<td>IO Label</td>
</tr>
</tbody>
</table>

Table 2: Font style used in faceplates

Figure 3-21 provides the look and feel of all the available widgets in the HMI toolkit, including:

- Graphics: label, image, ellipse, rectangle, polyline…
- Monitors: text update, LED, progress bar, tank…
- Controls: text input, action button, Boolean button, choice button, combo box and check box…
3.8 Appearance

All mimics shall have a consistent appearance.

- Mimic size shall be fixed and not adjustable by the user (3236 x 1760).
- Displayed information shall be presented from left to right and begin at the top.
- Information and controls used in emergency conditions shall be grouped separately to those used for normal operations.
- All necessary information required for handling the worst case scenario shall be presented on a single page.
- The density of information on each page shall not appear overly cluttered to the user. The density of information on pages used for critical task sequences shall be lower than on other information pages.
- The use of scrolling on mimics is not permitted.
- Corresponding information on diagrams shall be placed in the same relative position in all similar instances.
- Information and controls shall be grouped in relation to the user’s task and their objectives, and relate to operating procedures.
- Options shall be grouped in conventional or natural grouping according to the user’s task.

For example:

- Information and controls shall be presented in the sequence of how they will be used.
- Information and controls used most frequently shall be positioned at the top of the display.
- Highest priority information and controls shall be placed in the most accessible position within groups and on pages.

For rapid and accurate searching, groups of controls and/or information shall not exceed 6 lines in vertical length top to bottom and 12 characters in width.

Controls shall be placed below indications. Where this is not practicable, the control shall be placed on the right of the indication.
3.9 User Input

Lists, tables and graphs shall present read-only information.

The system shall automatically input data that is already available or provide defaults by reading feedback from the process as shown in Figure 3-22.

The operator shall complete the following steps to change a process variable:

1. Click in the entry field: the mouse pointer changes as well as the background colour

2. Enter a new value and validate it by pressing the Enter key or by losing the focus. According to the result of the validation script, the input data will be displayed in invalid or valid colour and the tooltip will indicate the status of the input

3. Be able to modify an input prior submission to the process

4. Submit the new value. Another validation is done before sending the value to the process. If the action cannot be performed, a popup window is displayed
5. After successful submission, the readback field will flash until it reaches the requested value.

![Figure 3-25 User input error popup window](image)

It is not recommended to hide or disable the **Submit** and **Cancel** buttons during this process.

Confirmation and “*Are you sure?*” dialogues are not required for simple control actions that will have no impact or compromise personnel safety, machine availability, or asset protection. System self-correcting features that detect and automatically correct user errors are not permitted for control actions.

Where possible, users shall not be responsible for control actions that result in destructive consequences.

Examples of scripts attached to input validation and submit/cancel buttons are provided in templates/faceplates/scripts. It is important to note that the execution of the input submission script is skipped for the trigger PVs’ first connection.

### 3.10 Feedback

The system shall have sufficient response time to provide feedback about the outcome of mimic control operations during or immediately following those operations.

Following a control action, the mimic shall provide the user with the following feedback:

- That the system has received the variable change command, has processed it and accepted the command.
- The variable is in the process of changing. The form of feedback needs to indicate what is changing, e.g. if the valve is closing or opening. The status of valves, open, closed or travelling needs to be visible.
- Confirmation that the change is complete and the system state is as requested.

### 3.11 Faceplates

Each faceplate shall be uniquely labelled. This is done by adding a Label widget on top of the tab container to display the macro value `$COMPONENT` using the `IO Title` font style.

![Figure 3-27 Position and style of the faceplate label](image)

All faceplates shall have a consistent appearance.

A faceplate shall remain open until the operator click on another controllable component or navigate to another screen.
3.12 LED

The LED widget default properties and behaviour have been standardised:

- Off state colour = light brown
- On state colour with no alarm = Green
- On state colour with minor alarm = Yellow
- On state colour with major alarm = Red
- On/Off disconnected state colour = Magenta

Two standard rules have been attached to the LED widget as illustrated on Figure 3-28.

Figure 3-28 Standard rules for the LED ON and OFF properties

Figure 3-29 gives the details of the Alarm LED ON State Color rule definition.

Figure 3-29 LED On Color property rule definition

Tip: To add the rules on existing OPI, just validate the OPI and make a quick fix.
3.13 Alarms

The system shall not present an “alarm OK” message to the operator in the alarm pane or the alarms list.

The system shall provide an ‘invalid’ alarm state for instances when there is a signal fault i.e. there is an error in the feedback sent back from the plant. Depending on its consequence, this alarm state may be overridden and be presented as minor or major alarms to the operator, as by design the invalid severity is the highest one.

The following alarm states are used consistently across all displays:

- Acknowledged – when the operator has indicated awareness of the presence of an alarm
- Unacknowledged – when the alarm is raised but the operator has not indicated awareness of its presence
- Cleared – when the condition that triggered the alarm has returned to normal
- Shelved – the operator has prevented a nuisance alarm from being displayed for a limited time
- Suppressed – alarm is suppressed when logical criteria are applied to determine that the alarm shall not occur, even though the base alarm condition is present.

If an alarm clears before being acknowledged by the operator, the alarm remains in the alarm table – latched behaviour - but is presented in reverse-video (text font colour becomes background colour, and background colour is used for the alarm font).

![Figure 3-30 Major alarm state transition diagram example](image)

3.14 Alarms in mimics

Alarm indications will be shown on mimics to indicate the component that is in an alarm state. Functionality to acknowledge alarms from the mimic is provided and the ability to see more information in the component’s faceplate shall be developed.

All active alarms shall be shown in process diagram displays - this includes those that have been automatically suppressed or manually shelved (since they may constitute relevant status information for the operator). EPICS alarm information will be used for this purpose and not BEAST latched alarms.

Each alarm on Process Diagrams will show their priority - the widget background reflects the priority of the alarm - and state – the background flashes at 2Hz until the alarm is acknowledged by the operator. Only individual BEAST alarm can be acknowledged from mimics, by right-click on the widget/symbol.

Alarm synthesis is provided per CBS level and used to animate the navigation buttons foreground colour which reflects the highest alarm severity on a given CBS, from invalid, major to minor.
3.15 Additional symbol information (optional)

Additional information regarding the status of the component, the data quality, the alarm information and the control mode can be displayed around the symbol as illustrated on Figure 3-31:

- The tag name of the component shall be placed above the component symbol, while numerical indicators, such as measurements and local statuses, are located below the symbol. Numerical indicators shall always be displayed with the relevant engineering unit.
- The top left corner is allocated to the data quality indication.
- The top right corner is allocated to alarm indication and presence of interlocks.
- The bottom right corner is allocated to the current control mode of the component.

![Figure 3-31 Control valve example](https://example.com/)

Figure 3-31 Control valve example under [resource link]

Figure 3-32 shows the layout at runtime without and with the Information Layer visible.

![Figure 3-32 Control valve example](https://example.com/)

Figure 3-32 Control valve example without and with the Information Layer visible

Data quality indication can have the following values:

<table>
<thead>
<tr>
<th>Status</th>
<th>Letter</th>
<th>Font Colour</th>
<th>Background Colour</th>
<th>Priority</th>
</tr>
</thead>
<tbody>
<tr>
<td>I/O Error, I/O card error, etc.</td>
<td>E</td>
<td>IO Label</td>
<td>IO Invalid Level Alarm</td>
<td>Highest</td>
</tr>
<tr>
<td>I/O Simulated – when HW I/Os measured values are substituted with simulated values</td>
<td>S</td>
<td>IO Label</td>
<td>IO Medium Level Alarm</td>
<td></td>
</tr>
<tr>
<td>Data OK -</td>
<td>None</td>
<td>None</td>
<td>None</td>
<td>Lowest</td>
</tr>
</tbody>
</table>

![Figure 3-33 Control valve example](https://example.com/)

Figure 3-33 shows a control valve with an I/O error and with simulated data.

The SDD EPICS template as well as the HMI object and faceplate of this example can be found under [resource link].
Alarm indicator can have the following values:

<table>
<thead>
<tr>
<th>Status</th>
<th>Letter</th>
<th>Font Colour</th>
<th>Background Colour</th>
<th>Priority</th>
</tr>
</thead>
<tbody>
<tr>
<td>PIS Interlock</td>
<td>T</td>
<td>IO Label</td>
<td>IO High Level Alarm</td>
<td>Highest</td>
</tr>
<tr>
<td>Conventional Control Interlock</td>
<td>I</td>
<td>IO Label</td>
<td>IO High Level Alarm</td>
<td></td>
</tr>
<tr>
<td>Analogue Alarm severity: Major</td>
<td>HH</td>
<td>IO Label</td>
<td>IO High Level Alarm</td>
<td></td>
</tr>
<tr>
<td></td>
<td>H</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>L</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>LL</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Analogue Alarm severity: Minor</td>
<td>HH</td>
<td>IO Label</td>
<td>IO Medium Level Alarm</td>
<td></td>
</tr>
<tr>
<td></td>
<td>H</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>L</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>LL</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Digital Alarm</td>
<td>A</td>
<td>IO Label</td>
<td>IO High Level Alarm</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>IO Medium Level Alarm</td>
<td></td>
</tr>
<tr>
<td>No Alarm</td>
<td>None</td>
<td>None</td>
<td>None</td>
<td>Lowest</td>
</tr>
</tbody>
</table>

Figure 3-34 shows an illustration of a control valve with a PIS interlock and one with a conventional control interlock. The background of the component reflects the status of the computed highest alarm.

Control Mode indicator can have the following values:

<table>
<thead>
<tr>
<th>Mode</th>
<th>Letter</th>
<th>Font Colour</th>
<th>Background Colour</th>
<th>Priority</th>
</tr>
</thead>
<tbody>
<tr>
<td>Force – when the set-point of the device is forced by the operator in presence of Conventional Control Interlock</td>
<td>F</td>
<td>IO Label</td>
<td>None</td>
<td>Highest</td>
</tr>
<tr>
<td>Local – when the device is under local hardware control (close to the plant system)</td>
<td>L</td>
<td>IO Label</td>
<td>None</td>
<td></td>
</tr>
<tr>
<td>Manual – when the set-point of the device is set by the operator via the Mimic – No override of the Conventional Control Interlock allowed in this operation mode</td>
<td>M</td>
<td>IO Label</td>
<td>None</td>
<td></td>
</tr>
<tr>
<td>Auto</td>
<td>None</td>
<td>None</td>
<td>None</td>
<td>Lowest</td>
</tr>
</tbody>
</table>

Figure 3-35 shows a control valve in manual mode and one in force mode.
3.16 Scripting

Each graphical widget comes with a predefined behaviour. For instance, the monitor widgets are background alarm sensitive which means that the background colour reflects the alarm severity and flashes at 2Hz until the alarm is acknowledged by the operator.

For complex mechanism, CODAC HMI toolkit provides containers which manage user interaction – navigation pane, alarm pane and control pane.

In order to minimise the maintenance effort, the customization of widgets shall be limited. In case the predefined behaviour is not acceptable, defining rules shall be the preferred solution. The usage of script shall be exceptional – input validation for example, and Javascript the chosen programming language. Embedded scripts shall be avoided.

4 Operator User Interface Validation

OPI Validation tool built on the Eclipse’s validation framework provides means to validate and quickly fix existing OPI files with regards to ITER OPI schema. Validation can be invoked manually on an OPI file or group of OPI files by invoking the Validate contextual menu item via a right-click. Manually invoking validation will run validation in the background.

When validation is completed, a validation results dialog is displayed, and any problems found will be shown in the Problems view as illustrated on Figure 4-1.

To open the Problems view, select Problems from the menu bar Window -> Show View -> Other... -> General -> Problems.

Double-clicking on an error/warning/info message in this view opens the respective editor (OPI Editor or OPI Runtime) and highlights the failing widget.

Finally, from the Problems view, fixes can be applied on one or many error/warning messages by invoking the Quick fix contextual menu item. A confirmation is asked for making a backup before applying the fix.

After the quick fix is completed, the modified OPI files are revalidated.

![Figure 4-1 Operator User Interface Validation Results](image)