Report

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.

<table>
<thead>
<tr>
<th>Approval Process</th>
<th>Name</th>
<th>Action</th>
<th>Affiliation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Author</td>
<td>Utzel N.</td>
<td>21 Jun 2016:signed</td>
<td>IO/DG/COO/SCOD/CSD/CDC</td>
</tr>
<tr>
<td>Co-Authors</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Reviewers</td>
<td>Park M.</td>
<td>27 Jun 2016:recommended</td>
<td>IO/DG/COO/SCOD/CSD/CDC</td>
</tr>
<tr>
<td></td>
<td>Piccolo F.</td>
<td>27 Jun 2016:recommended</td>
<td>IO/DG/COO/SCOD/OMA</td>
</tr>
<tr>
<td>Approver</td>
<td>Wallander A.</td>
<td>28 Jun 2016:approved</td>
<td>IO/DG/COO/SCOD/CSD</td>
</tr>
</tbody>
</table>

Document Security: Internal Use

RO: Delong Joseph

Read Access

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 lin...
<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>Internal review comments integrated.</td>
</tr>
<tr>
<td>v1.1</td>
<td>Signed</td>
<td>05 Jan 2011</td>
<td>Footer error fixed</td>
</tr>
<tr>
<td>v1.2</td>
<td>Signed</td>
<td>01 Feb 2011</td>
<td>Reference to ITER Process for Human Machine Interface (HMI) Development (ITER D 3T9UK2)</td>
</tr>
<tr>
<td>v2.0</td>
<td>Approved</td>
<td>10 Feb 2011</td>
<td>PCDH V7 schema</td>
</tr>
<tr>
<td>v2.1</td>
<td>Approved</td>
<td>01 Feb 2013</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 (IHIFP) 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>v3.0</td>
<td>Approved</td>
<td>30 Jan 2015</td>
<td>Support of ITER Human Factor requirements for HMI development (QEDG6L)</td>
</tr>
<tr>
<td>v3.1</td>
<td>Disapproved</td>
<td>04 Feb 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>v3.2</td>
<td>Approved</td>
<td>18 Feb 2015</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>v3.3</td>
<td>Approved</td>
<td>13 Jul 2015</td>
<td>Missing section regarding the symbol animation</td>
</tr>
<tr>
<td>v3.4</td>
<td>Signed</td>
<td>09 Feb 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>v3.5</td>
<td>Approved</td>
<td>21 Jun 2016</td>
<td>PDF version generated with Office</td>
</tr>
</tbody>
</table>
Table of Contents

1 Introduction ........................................................................................................................................... 3
  1.1 PCDH Context .................................................................................................................................. 3
  1.2 Document Scope ................................................................................................................................. 3
  1.3 Related documents ............................................................................................................................... 3
  1.4 Definitions .......................................................................................................................................... 4
  1.5 HMI Style Guide ................................................................................................................................. 4
  1.6 HMI Toolkit ....................................................................................................................................... 4

2 Operator User Interface Layout ........................................................................................................... 5
  2.1 Size .................................................................................................................................................... 6
  2.2 Naming .............................................................................................................................................. 6
  2.3 Status Bar ......................................................................................................................................... 7
  2.4 Main Display Area - Mimic ............................................................................................................... 8
    2.4.1 Auxiliaries Status ....................................................................................................................... 8
    2.4.2 Mimic ......................................................................................................................................... 9
  2.5 Alarm Pane ...................................................................................................................................... 11
    2.5.1 Alarm Page ................................................................................................................................ 12
  2.6 Control Area / Faceplate Pane ......................................................................................................... 13
  2.7 Navigation Pane ............................................................................................................................... 17
  2.8 Alarms List ..................................................................................................................................... 20
  2.9 SDD Automatically Generated Operator Interface with Navigation ........................................... 21

3 Operator User Interface Detailed Design ........................................................................................... 22
  3.1 Fonts ............................................................................................................................................... 22
  3.2 Colours .......................................................................................................................................... 22
  3.3 Symbols .......................................................................................................................................... 23
  3.4 Symbol Animation ............................................................................................................................ 24
  3.5 Measurement Units and Precision ................................................................................................... 26
  3.6 Labelling ......................................................................................................................................... 26
  3.7 Mimics ........................................................................................................................................... 26
  3.8 Appearance ..................................................................................................................................... 31
  3.9 User Input ....................................................................................................................................... 32
  3.10 Feedback ....................................................................................................................................... 33
  3.11 Faceplates .................................................................................................................................... 33
  3.12 LED ............................................................................................................................................... 34
  3.13 Alarms .......................................................................................................................................... 35
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.14</td>
<td>Alarms in mimics</td>
<td>35</td>
</tr>
<tr>
<td>3.15</td>
<td>Additional symbol information (optional)</td>
<td>36</td>
</tr>
<tr>
<td>3.16</td>
<td>Scripting</td>
<td>38</td>
</tr>
<tr>
<td>4</td>
<td>Operator User Interface Validation</td>
<td>38</td>
</tr>
</tbody>
</table>
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.

Figure 1-1 Schema of PCDH documents

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

[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 monitor 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

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 includes 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.

The root element of the screen is the ‘Display’ widget with some predefined properties, such as the position and the size \(^2\) as illustrated on Figure 2-2.

```
<table>
<thead>
<tr>
<th>Property</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Position</td>
<td></td>
</tr>
<tr>
<td>Height</td>
<td>2080</td>
</tr>
<tr>
<td>Width</td>
<td>3830</td>
</tr>
<tr>
<td>X</td>
<td>0</td>
</tr>
<tr>
<td>Y</td>
<td>0</td>
</tr>
</tbody>
</table>
```

Figure 2-2 Display widget – Position properties

A macro \([\text{TITLE}]\) defines the mimic title as shown on Figure 2-3 and is used as widget name.

```
<table>
<thead>
<tr>
<th>Property</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Macros</td>
<td>{Parent Macros} {TITLE=UTIL OVERVIEW}</td>
</tr>
<tr>
<td>Name</td>
<td>${\text{TITLE}}$</td>
</tr>
<tr>
<td>Widget Type</td>
<td>Display</td>
</tr>
</tbody>
</table>
```

Figure 2-3 Title macro specification during edition

\(^2\) The HMI toolkit provides an automatic scaling tool to adapt Full HD screens to 4K resolution. Consult [RD4] for more information.
At runtime, the display tab reflects the widget name as illustrated on Figure 2-4.

![UTIL OVERVIEW](image)

Figure 2-4 Title macro instantiated at runtime for the tab name

### 2.1 Size

The size of each specific zone of the display canvas is fixed and not adjustable by the user. The following table gives the dimensions in percentage of the available screen size:

<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></td>
<td>100</td>
<td>5</td>
</tr>
<tr>
<td></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>85</td>
<td>85</td>
</tr>
<tr>
<td>Navigation Area</td>
<td></td>
<td>85</td>
<td>10</td>
</tr>
<tr>
<td></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 user available space for each container is presented on Figure 2-5.

![Display canvas zones sizing](image)

Figure 2-5 Display canvas zones sizing

### 2.2 Naming

By convention, the OPI file name is based on the CBS structure:

```
ITER-{CBS1}-{CBS2}-{CBS3}.opi
```

Example of the 15kV work site power monitoring screen name: `ITER-UTIL-S15.opi`
2.3 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 2-6, shows an implementation of the status bar.

Figure 2-6 Status bar

Global statuses and UTC time are provided in standard by CODAC as standalone screens to be linked to the user screen as illustrated on Figure 2-7.

Figure 2-7 Status Bar hierarchy of widgets

To integrate them, a linked folder `<project_name>/src/main/boy/templates` is provided in the I&C project that points to `/opt/codac/opi/boy/resources/templates`.

Finally, the title of the mimic is presented in the middle of the status bar, using a Label widget to display the title macro value defined in the root Display widget as shown on Figure 2-8.

Figure 2-8 Label widget – Text property to display the title using “IO Title” font style
2.4 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.

2.4.1 Auxiliaries Status

On the top left of the mimic, space is allocated to represent the status of auxiliaries. Figure 2-9, shows an implementation of the plant system auxiliaries status.

![Auxiliaries Status](image)

Figure 2-9 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 2-10.

![Auxiliaries grouping containers organisation](image)

Figure 2-10 Auxiliaries grouping containers organisation

---

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

In order to display independently a mimic on a wall panel, it is required to provide it as a standalone OPI screen to be linked into the canvas as illustrated on Figure 2-11. By convention, the name of the mimic OPI is based on the CBS structure as follow: [ITER-[CBS1]-[CBS2]-[CBS3]]_Mimic.opi

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 2-12.

- 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 2-13.
- 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 2-14 provides an overview on how the mimic is organised in containers with the definition of the tick box.

Figure 2-14 Check box widget to show/hide the information layer
2.5 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. Figure 2-15 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 2-15: Alarm pane integration in the canvas](image)

Figure 2-15 Alarm pane integration in the canvas

Figure 2-16 documents the priority icons and alarm message background and foreground colours.

<table>
<thead>
<tr>
<th>Alarm Type</th>
<th>Icons</th>
<th>Shelved alarms*</th>
<th>Colour</th>
</tr>
</thead>
<tbody>
<tr>
<td>Invalid/Unknown alarm</td>
<td>🔄</td>
<td>🔄</td>
<td>(255,0,255)</td>
</tr>
<tr>
<td>&quot;</td>
<td>🔄</td>
<td>🔄</td>
<td></td>
</tr>
<tr>
<td>&quot;</td>
<td>🔄</td>
<td>🔄</td>
<td></td>
</tr>
<tr>
<td>Major alarm</td>
<td>🔄</td>
<td>🔄</td>
<td>(250,15,14)</td>
</tr>
<tr>
<td>&quot;</td>
<td>🔄</td>
<td>🔄</td>
<td></td>
</tr>
<tr>
<td>Minor alarm</td>
<td>🔄</td>
<td>🔄</td>
<td>(255,255,0)</td>
</tr>
<tr>
<td>&quot;</td>
<td>🔄</td>
<td>🔄</td>
<td></td>
</tr>
<tr>
<td>&quot;</td>
<td>🔄</td>
<td>🔄</td>
<td></td>
</tr>
</tbody>
</table>

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

![Figure 2-16: Alarm coding](image)

Figure 2-16 Alarm coding
Figure 2-17 shows how the standard alarm pane templates/AlarmPane.opi is linked into the canvas in the OPI editor.

Figure 2-17 Alarm pane linked into the canvas

2.5.1 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 2-18 presents an alarm page with the alarm description and procedure, including all relevant data for analysis using trend widgets.

Figure 2-18 Alarm page
2.6 Control Area / Faceplate Pane

Each controllable component on a mimic shall have a ‘faceplate’ to provide information about the component and options to control it.

Figure 2-19, shows an implementation of a faceplate displayed when the user clicks on a valve symbol.

![Faceplate](image1)

Figure 2-19 Faceplate

The faceplate is displayed within a linking container provided by CODAC and integrated from `templates/ControlPane.opi` into the canvas as shown on Figure 2-20. A default legend is displayed.

![Control pane linked into the canvas](image2)

Figure 2-20 Control pane linked into the canvas
Then, an action is defined on the component widget, which reacts to user click to load a specified faceplate and to instantiate the component macro as illustrated on Figure 2-21. This action writes in a predefined variable \( \text{loc://$(DID)}\) the relative path to the faceplate OPI and specify the component name using the macro \( \{\text{COMPONENT}\}\).

In the following example, writing the value 

```
../faceplates/TecSystem opi COMP\n
```

into the PV name \( \text{loc://$(DID)}\) will ask the CODAC linking container to load 

```
../faceplates/TecSystem opi
```

and instantiate the macro \( \{\text{COMPONENT}\}\) with the name of the component \( \text{UTIL-S15-AG07:MUT9}\). In addition, the widget named \( \text{UTIL-S15-AG07:MUT9}\) will be highlighted on the screen.

![Figure 2-21 Action to load a faceplate on user click](image)

It is recommended to setup a dedicated folder for the faceplates within the I&C project under \( \text{src/main/boy} \) as shown on Figure 2-22. This folder can be organised with sub-folders such as `pictures` and `scripts`.

![Figure 2-22 Faceplates folder within the I&C project under src/main/boy](image)

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 templates folder, as reported on Figure 2-23. 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 2-24.

Scripts are also provided to illustrate user input validation and submission to the process.
Figure 2-23 Faceplate tabs – Status, Trends, Controls, Help

Figure 2-24 Analog and Digital Input Faceplates
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
2.7 **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 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 are alarm sensitive. They flash at 2Hz if a new alarm is active and need to be acknowledged by the operator on the corresponding CBS level.

Hovering the mouse over a navigation button, the tooltip will show the number of latched alarms on the corresponding alarm tree node as illustrated on Figure 2-25.

![Figure 2-25 Navigation button tooltip includes the number of latched alarms information](image)

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 2-26.

![Figure 2-26 Navigation button “Latched Alarm” context menu](image)

The following screenshots report examples of the navigation principles\(^4\).

---

\(^4\) Some navigation buttons are greyed out because they are currently not available. The mimics shown in the pictures are for illustrative purposes only.
Figure 2-27 ITER machine overview (CBS0)

Figure 2-28 UTIL overview (CBS1)
The navigation pane supports CBS4 and CBS5.

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.

Figure 2-29 UTIL overview (CBS2)

Figure 2-30 OPIs map
2.8 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 2-31 is made available via a right-click on any navigation button and using the option Actions -> Open in a new window the Alarms List.

![Figure 2-31 Alarms List](image-url)
2.9 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 2-32.

![Figure 2-32 Edition of an empty mimic for BUILD_B11-CA](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 2-33.

![Figure 2-33 Edition of Navigation-BUILD_B11-BA.xml](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`. 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:

- **IOTitle** and **IOSubtitle** for headers and critical operational information,
- **IOLabel** for operational information (e.g. labels, numerical data) presented in mimics and faceplates. This the default font for most of the widgets,
- **IOScale** for graph axis.

![Figure 3-1 Fonts](image)

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:

- **IOBackground** colour for the mimics\(^5\)
- **IOWord Foreground** colour for the text and line
- **IO Invalid Level Alarm**, **IO High Level Alarm** and **IO Medium Level Alarm** colours used in conjunction with the ‘background alarm sensitive = yes’ property
- **IOPV OFF** and **IOPV ON** colours for symbol in 0, 1 or more than 1 position.

\(^5\) 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
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)

By convention, the svg file name is “TTT full_name index.svg” with index [0..7] representing the state.
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.
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.

3.6 Labelling

Label widgets are predefined to be left-justified.

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
Figure 3-15 List example

Figure 3-16 Table example
The following requirements apply to the presentation of fields on screens:

- Entry fields shall be clearly indicated with a box or 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 an arrow when the user hovers over it.

It is possible to attach a tooltip to fields - by default, the tooltip displays the PV name, type, value, timestamp and severity.

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 sufficient 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.
- Coding shall be used when multiple functions are displayed in a single graph, particularly if curves approach and/or intersect one another.
- Line coding 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 noticeably thicker line style or in a different color. Thin coding shall be distinguishable from any other coding already used in the graph.

When the user must compare data represented by separate curves, the curves shall be displayed in one combined graph.

Figure 3-17 Fields example

Figure 3-18 Graphs example up to 8 traces
Trace colour shall use one of the eight predefined trace colours – \( \text{IO Trace 1} \) to \( \text{IO Trace 8} \), 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.

![Figure 3-22 Entry field initialised to readback value](image)

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

![Figure 3-23 Focus on an entry field](image)

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

![Figure 3-24 Invalid and valid input indication](image)

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

![Figure 3-25 User input error popup window](image)
5. After successful submission, the readback field will flash until it reaches the requested value.

![Position flashing](image1)
![Change complete](image2)

Figure 3-26 Current position flashing until it reaches the input

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](image3)

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).

![Diagram of alarm state transitions]

Figure 3-30 Major alarm state transition diagram example

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.

![Control valve example](https://boy/resources/Design-Guide/industrial-control)

Figure 3-31 Control valve example under boy/resources/Design-Guide/industrial-control

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

![Control valve example](https://boy/resources/Design-Guide/industrial-control)

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>Medium</td>
</tr>
<tr>
<td>Data OK -</td>
<td>None</td>
<td>None</td>
<td>None</td>
<td>Lowest</td>
</tr>
</tbody>
</table>

![Control valve example](https://boy/resources/Design-Guide/industrial-control)

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

![Control valve example](https://boy/resources/Design-Guide/industrial-control)

Figure 3-33 Control valve example 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 boy/resources/Design-Guide/industrial-control/Field;
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.

![Figure 3-34 Control valve example with a PIS Interlock and with a conventional interlock](image)

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.

![Figure 3-35 Control valve example in manual and force mode](image)
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 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)

Figure 4-1 Operator User Interface Validation Results