EUD Mimic Toolset

TERMA
SPACE

EUD Mimic is a set of tools for designing and running mimics.

The EGOS User Desktop (EUD) Mimic Components are a set of Eclipse plug-ins integrated in the existing EUD software infrastructure (EUD version 2.x). The plug-ins are meant to provide a set of tools for designing and running synoptical representations of a system called mimic diagrams or mimics.

The EUD Mimic Components support the design, visualization, and operation of mimic diagrams:

- **The Mimic Designer Display (MD)** a standalone EUD-based graphical editor that provides functions for designing mimic diagrams.
- **The Mimic Viewer Display (MV)** a EUD-based display used for rendering and executing a mimic diagram created with the MD; the diagram is animated according to the data provided by the backend system. The MV also allows the visual commanding of the underlying system.

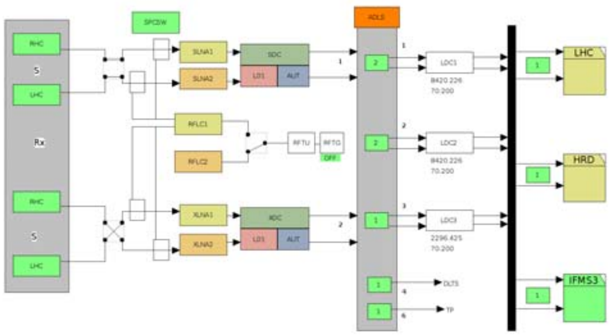

Figure 1 Example Mimic Diagram

Monitoring and Control of ESOC systems

Mimic diagrams are used to monitor and control systems within the ESOC ground segment through a visual representation of the components of the system. The representation is composed by a set of elements – "widgets", in this context. A widget can, for example, represent a component of a system, to display the value of a parameter of a system, or even the connection between two components.

Furthermore, via the interfaces provided by the EUD Framework, the EUD Mimic Components are able to receive parameter values from backend systems, such as the "real" ground station represented by a mimic. These parameters can be used to animate the mimic being displayed. For example, a mimic diagram can display the value of a parameter, or it can provide visual feedback on the alarm status of a parameter.

Finally, by using the EUD Operations Interface in combination with some widgets which are interactive, the user is able to command the connected backend system. For example, via a push button widget – which is a button that the user can click – it is possible to issue a command to the backend system, which could be a command to activate or deactivate some functionality of the "real" system.

The connection between the MV and the system being represented is implemented using the services provided by the EUD Framework; MV plug-ins can be integrated in any EUD-based application implementing a set of adapters for such EUD services.

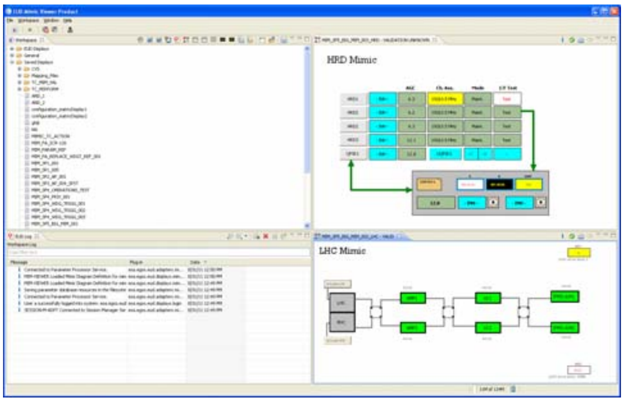

Figure 2 Mimic viewer

Since early releases, the Mimic Viewer was successfully integrated by the GSMC (ground Station Monitoring & Control) project.

Basic and Custom Widgets

Mimic Designer offers a basic set of widgets for designing Mimic diagrams. Widgets are organised in a palette and can be easily created and configured for representing the system under design.

The set of basic widget can be expanded with the definition of *custom widgets*. A custom widget is a complex widget composed of basic widgets that can be imported in the Mimic Designer palette and used any other widget. Custom widgets can be defined and imported in the designer palette at any time.

Additionally, there is a mechanism for managing collections of custom widgets called *libraries*. Like custom widgets, also libraries of custom widgets can be imported in the designer palette.

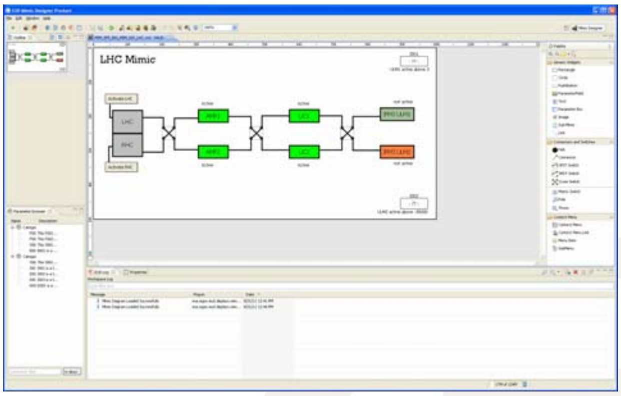

Figure 3 Mimic Designer

Action Procedures

An action procedure is a text written in Java Script language; mimic widgets can be assigned with one or more action procedure to accomplish different tasks.

The Action procedure can reference to any parameter (sample) available in the Backend system and trigger system M&C related actions, like:

- Changing the appearance properties of a widget.
- Opening an AND or a Graph Display.
- Acknowledgement of alarms.
- Controlling the backend system by the EUD Operations Interface, e.g. changing the value of a controlled variable.

There are two types of Action Procedures:

- Monitoring action procedures used to change the appearance of widgets in order to reflect parameters of the monitored systems;
- Control action procedures used to control the underlying systems by invoking operations (e.g. command a parameter) or to trigger actions on the EUD (e.g. open a new display).

Following properties for an action procedure can be configured in the Designer:

- Input Arguments: list of arguments the user has to provide before the action procedure is executed. Name, type, default values shall be provided for each input parameter. As soon as the action procedure is triggered, a window will be opened to allow the user to enter the parameters.
- Required Control Mode: the control mode (MONITOR, CONTROL) the user has to have for executing the action procedure. If the required mode is MONITOR, the action procedure can always be executed. If the required mode is CONTROL, only if the user has CONTROL role.
- Trigger Parameters: list of parameters; if value/status one of these parameters changes, the action procedure is executed
- Trigger Widgets: list of widgets, if the status property of one of these widgets changes, the action procedure is executed

Dynamic Path

Dynamic path highlighting is a concept for writing action procedures which is useful for changing the appearance of the connections between widgets according to certain conditions. It can be used, for example, to highlight the connection between two systems according to certain parameter values from the back-end. It is very versatile, as all the logic is present in the action procedures written by the user developing a mimic diagram.

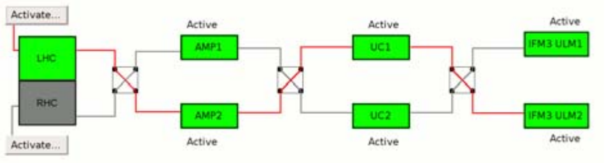

Figure 4 Example Dynamic Path

Parameter and Mimic Templates

The Mimic Designer supports the creation of the so called ‘Mimic Templates’, based on ‘Parameter Templates’ (PT), which are meant to facilitate the creation of mimic diagrams for subsystems of the same type by removing the need to specify the exact parameter references for each mimic diagram manually.

The basic idea is to allow the use of Parameter Templates instead of ‘real’ Parameter References for associated parameters, triggering parameters and inside action procedures. This allows creating a Mimic Template File. These files can be created and saved with the Mimic Designer as any regular Mimic Diagram File. The user can then create one or several Parameter Mapping Files that can be applied to the Mimic Template Files to create the final Mimic Diagram Files.