

## GUI\_PatternDefnEdit – Pattern Definition Edit

### Revision history

2017-03-19	Robin Lamacraft	Original draft
2018-06-02	Rod Thompson	Amend Requires Services 2 & 3
2018-06-08	Rod Thompson	Replace 'screen' with 'window' Replace BR_PanelConfig with BR_WindowConfig

### SCOPE

This GUI module displays the list of properties of a Pattern Definition instance for creating, editing and deleting. An HRE Pattern Definition instance defines the properties and provides defaults for other properties for a Node Entity instance of a designated Node Entity type. It displays the current properties of a Pattern Definition. The Pattern Definitions are aligned with the Node Entities within HRE. Over time this list will be expanded by the installation of plugins. Some Node Entities will only have one Pattern Definition while others like Events and Tasks will have many. [BLUE are later additions.](#)

#### Scope Variations:

Node Entity Type	Node Entity Differences
ITEM	(hidden)
→ External File	
→ Image	
→ Internal Text	
→ Another Item Type	(like Artworks)
→ Biological	(hidden)
→→ Person	
→→ Another Bio Type	(like Plants)
GROUP (hidden)	(hidden)
→ Event	
→ Task	
→ Location	
→ Repository	
→ Source	
→ Another Group Type	(like Organizations)

### LOOK AND FEEL

The window has 4 sections whose vertical height is adjustable by moveable horizontal dividers:

- Heading section shows:
  - The Node Entity Type (focus preset – selection available).
- A collection of command buttons?
  - “Configure” to select the layout for this window or create a new layout
  - “Save” to save the changes
  - “Reject” to reject the changes.
- Editing Toolbar section: This is customized depending on the Application Language. At the top there is collection of command buttons:
  - “Add New” – to create a new Pattern Definition instance from scratch
  - “Add Clone” – to create new Pattern Definition instance from currently selected
  - “Edit” – to edit or view the details of the selected Pattern Definition instance
  - “Delete” – to delete the selected Pattern Definition instance.

- Data Editing section: This is displayed as a scrollable property editor GUI Element whose fields are configured according to the Pattern Purpose and the link end Node Entity types. This Property Editor has 3 columns:
  - Column 1: Name of Field (label field) presented as:
    - Normal as the user initial value
    - Underline when the value has been edited in this view
    - Labels enclosed in “[” and “]” indicate are read only values.
  - Column 2: Property Value (text field)
  - Column 3: A button to open a special editor if that setting has a format that should not be in-place edited. Typically, this may be GUI\_UserTranslationEdit module.

#### Property Editing Variations:

Node Entity Type	Indicative List of Properties
<b>All Entity Types</b>	<ul style="list-style-type: none"> <li>▪ Pattern [by selection] (each pattern allocated Internal ID)</li> <li>▪ Pattern Label [edit]</li> <li>▪ Pattern Abbrev [edit]</li> <li>▪ Pattern Description [edit]</li> <li>▪ Apply Theme [edit]</li> <li>▪ User Data Quality default values [edit]</li> <li>▪ User Hidden [edit]</li> </ul>
<b>Event</b>	<ul style="list-style-type: none"> <li>▪ Can Reproduce = NO [read only]</li> <li>▪ Is Divisible = YES [read only]</li> <li>▪ Pattern Detail ID (Entity Type specific record)</li> <li>▪ Pattern Validation ID (Entity Type specific record)</li> </ul>
<b>External File</b>	<ul style="list-style-type: none"> <li>▪ Can Reproduce = NO [read only]</li> <li>▪ Is Divisible = NO [read only]</li> <li>▪ Pattern Detail ID (Entity Type specific record)</li> <li>▪ Pattern Validation (Entity Type specific record)</li> </ul>
<b>Image</b>	<ul style="list-style-type: none"> <li>▪ Can Reproduce = NO [read only]</li> <li>▪ Is Divisible = NO [read only]</li> <li>▪ Pattern Detail ID (Entity Type specific record)</li> <li>▪ Pattern Validation (Entity Type specific record)</li> </ul>
<b>Internal Text</b>	<ul style="list-style-type: none"> <li>▪ Can Reproduce = NO [read only]</li> <li>▪ Is Divisible = NO [read only]</li> <li>▪ Pattern Detail ID (Entity Type specific record)</li> <li>▪ Pattern Validation (Entity Type specific record)</li> </ul>
<b>Location</b>	<ul style="list-style-type: none"> <li>▪ Can Reproduce = NO [read only]</li> <li>▪ Is Divisible = YES [read only]</li> <li>▪ Pattern Detail ID (Entity Type specific record)</li> <li>▪ Pattern Validation (Entity Type specific record)</li> </ul>
<b>Person</b>	<ul style="list-style-type: none"> <li>▪ Can Reproduce = YES [read only]</li> <li>▪ Is Divisible = NO [read only]</li> <li>▪ Pattern Detail ID (Entity Type specific record)</li> <li>▪ Pattern Validation (Entity Type specific record)</li> </ul>
<b>Repository</b>	<ul style="list-style-type: none"> <li>▪ Can Reproduce = NO [read only]</li> <li>▪ Is Divisible = YES [read only]</li> <li>▪ Pattern Detail ID (Entity Type specific record)</li> <li>▪ Pattern Validation (Entity Type specific record)</li> </ul>
<b>Source</b>	<ul style="list-style-type: none"> <li>▪ Can Reproduce = NO [read only]</li> <li>▪ Is Divisible = YES [read only]</li> <li>▪ Pattern Detail ID (Entity Type specific record)</li> </ul>

	<ul style="list-style-type: none"> <li>▪ Pattern Validation (Entity Type specific record)</li> </ul>
<b>Task</b>	<ul style="list-style-type: none"> <li>▪ Can Reproduce = NO [read only]</li> <li>▪ Is Divisible = YES [read only]</li> <li>▪ Pattern Detail ID (Entity Type specific record)</li> <li>▪ Pattern Validation (Entity Type specific record)</li> </ul>

[Needs a mockup diagram here]

## ACTIONS

The fundamental operations are:

1. Open the window according to its saved Window Layout (BR\_WindowConfig)
2. Populate the heading section.
3. Perform actions of command buttons.
4. Save values in the appropriate database.

## USED BY:

1. GUI\_PatternDefnSelect.
2. Translation keyboard short cut.

## DATA CONTROLLED BY THIS MODULE:

1. None.

## REQUIRED DATA CONTROLLED BY OTHER MODULES:

1. HRE-ID.

## REQUIRED SERVICES

1. GUI\_FieldDefinitionEdit
2. GUI\_PatternDefnDetail
3. GUI\_PatternDefnValidate
4. BR\_Pattern
5. BR\_Setting
6. BR\_WindowConfig.

## APPLICATION PROGRAMMING INTERFACE (API)

1. Need Details.

## EVENT ACTIONS

1. Need details of event (keyboard or mouse) and the description of the action.

## WARNING CONDITIONS

1. Need details of the condition that raised the warning, example message and possible next steps.

## ERROR CONDITIONS

1. Need to record the condition that raised the error, example message and possible next steps.