

D6.4.1-Ergonomics Guidelines document

ModelWriter

Text & Model-Synchronized Document Engineering Platform

Work Package: WP6

Task: T6.4 - User Interfaces Implementation

Edited by:

Marwa Rostren <marwa.rostren@obeo.fr> (OBEO)

Yvan Lussaud <yvan.lussaud@obeo.fr> (OBEO)

Date: 31-May-2015

Version: 1.0.0

Apart from the deliverables which are defined as public information in the Project Cooperation Agreement (PCA), unless otherwise specified by the consortium, this document will be treated as strictly confidential.

Document History

Version	Author(s)	Date	Remarks
0.1.0	Marwa Rostren	31-May-2015	Draft
1.0.0	Yvan Lussaud	31-May-2015	Initial Release
1.5.0	Moharram Challenger	27-Aug-2015	Adding some sketches and screenshots

Table of Contents

Contents

DOCUMENT HISTORY	2
1. INTRODUCTION.....	4
■ <i>Role of the deliverable</i>	4
■ <i>Structure of the document</i>	4
■ <i>Terms, abbreviations and definitions</i>	4
2. THE ERGONOMICS GUIDELINES	5
■ <i>The Writer Part</i>	5
■ <i>The Model Links Part</i>	7
3. WIP – ERGONOMICS GUIDELINE OF THE LOCATIONS VIEW	8

1. Introduction

|Role of the deliverable

This document is the first version of the ergonomics guidelines document of the writer part. It may be improved or up-dated depending on the further details and requirements we get from our industrial use case providers.

|Structure of the document

This document is organized as follows:

- Chapter 1 introduces the document.
- Chapter 2 describes the ergonomics guidelines
- Chapter 3 WIP-Ergonomics Guideline of the Locations view

|Terms, abbreviations and definitions

Abbreviation	Definition
RDF	Resource Description Framework
WP	Work Package
UC	Use Case
MW	ModelWriter

2. The Ergonomics Guidelines

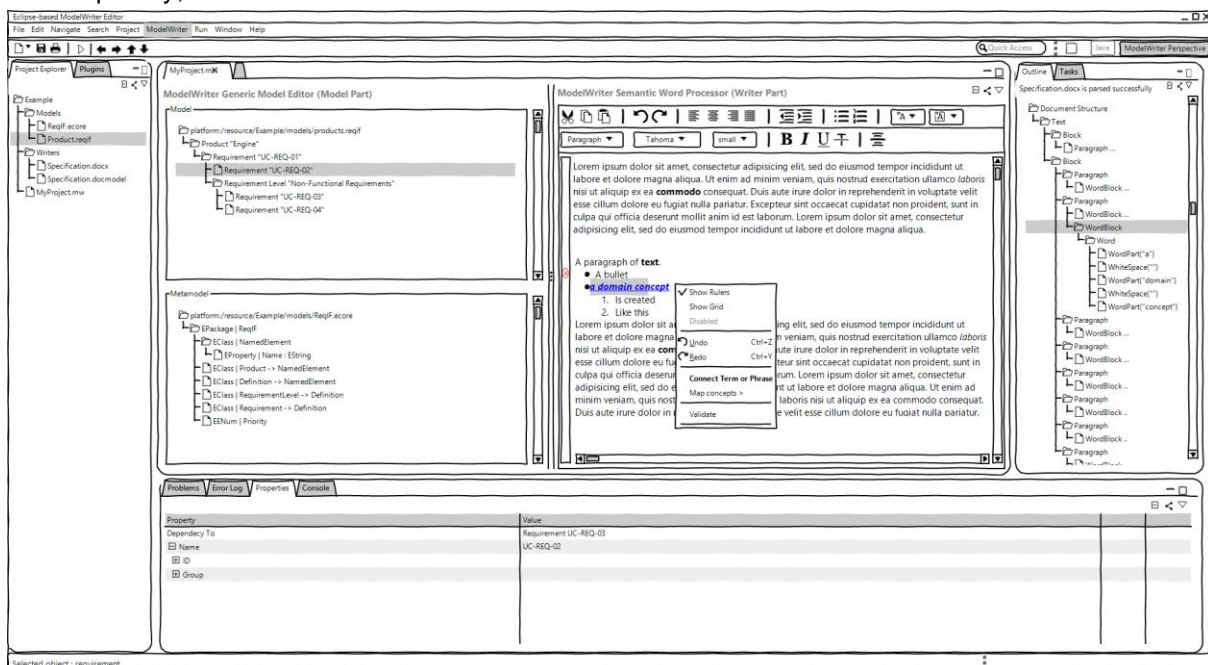
The Ergonomics Guidelines are about providing all the graphical user interfaces of the editor allowing the user to interact with the Knowledge Base content. In this section we define the list of ergonomics errors to prevent while implementing the editor.

Basing on lessons from past development of Intent, ModelWriter does not handle explicit mappings in documentations. This choice is made to prevent any efficiency problem while listening for the user modifications.

We assume that the editor will be used to:

- efficiently display the documentation content;
- efficiently handle listening modifications in the documentation
- cleverly allow the mapping of the documentation and the models;
- display the models existing links;
- keep up-to-date by reconciling and by synchronizing the links;
- handling actions like: export a documentation or a part of it, export a model, navigate from the editor to the linked models elements, etc.

Consequently, the MW editor will look as follows:



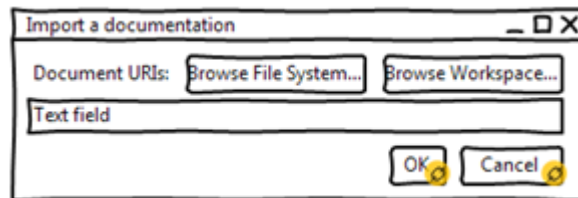
The left side of the editor, called the **Writer Part**, illustrates the document content regardless all formatting issues. The right side of the editor, called the **Model Links Part**, displays the models concepts to link and the models concepts already linked with the current selection in the document part. The next sections of this document detail each of these parts specification.

The Writer Part

The Writer part shall allow users to edit their document, to select either a word or a text part or a section or even multiple selection of one or more kind of those elements.

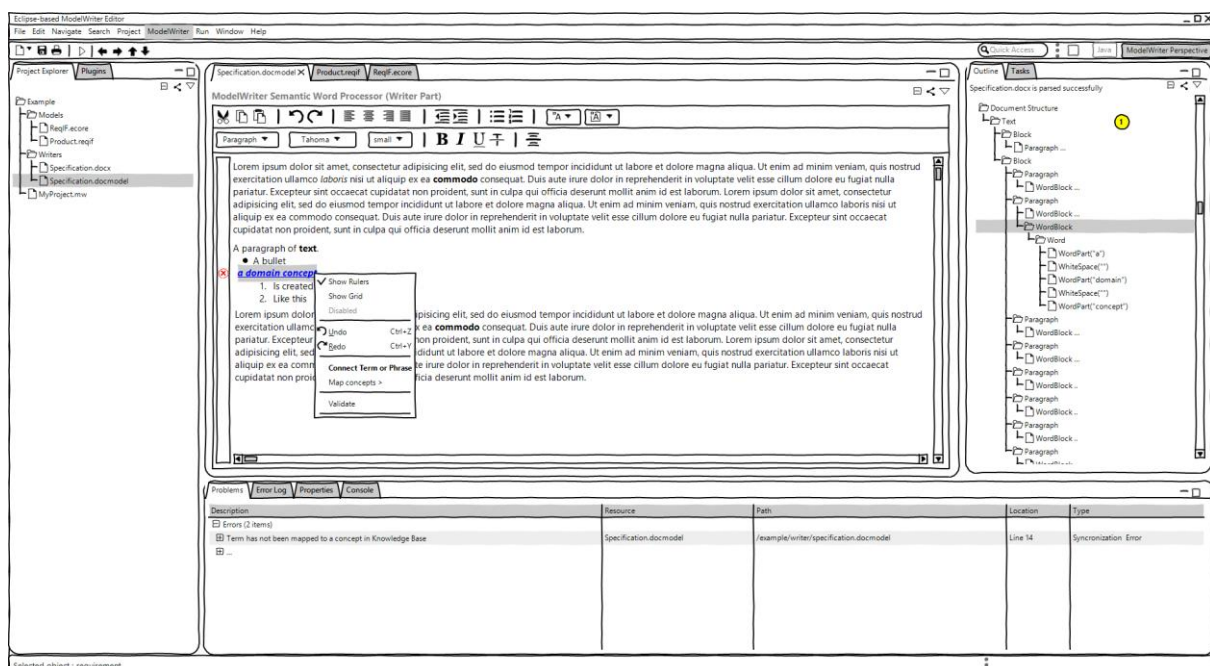
The **Writer Part** shall offer the next actions:

- The **Search** action which is available by using a **Search Area**. It helps users to seek either a specific word or a sentence by filtering the document content in the **Document Part**.
- The **Import** action which is available by selecting the **Import button**. It allows users to import a document either from the File System or from his Workspace. When the user chooses to import a new document, a new window appears to allow him browsing the File System or the Workspace as shown in the next screenshot.



- **Drag & Drop** (shall be available using the mouse) action which allows users to drag a document and drop it to open it in the writer part.
- The **Selection** action (shall be available using a keyboard shortcut) which is available by selecting either a word or a sentence or a section. It helps users to link the current selection to one or more model concept.

After a document import, the document part must contain the whole content of the imported document. The content will display the same original document content composition.



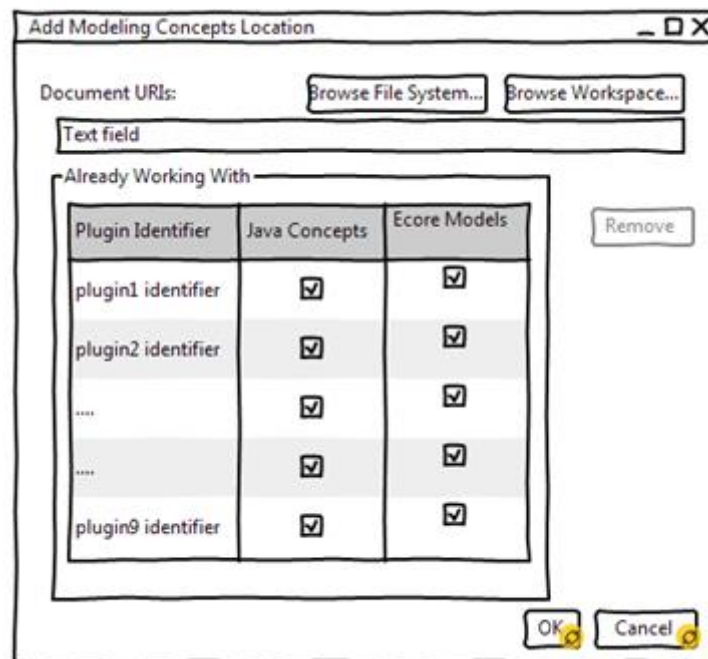
The writer part will have the capability of marking a term and then connecting a term to another term or a model element which is already marked.

The Model Links Part

The model links part must allow users to check for a text selection in the document part the already linked concepts **and** the list of the potential links.

The **Model Links Part** offers the next actions:

- the **Search** action which is available by using the **Search Area**. It helps users to seek a model concept.
- the **Add** action which is available by selecting the **Add button**. It allows users to add a plugin from the File System or from his Workspace to consider either the contained models or the contained Java concepts or both of them. When the user chooses to add a new plugin to work with, a new window appears to allow him browsing the File System or the Workspace as shown in the next screenshot. This window displays also the already working with plugins list.



- the **Remove** Working With plugin action which is available either by a single click on the Working With button or by single click on the **Add** button. It allows the user to retrieve a plugin with which he was working with from the list. This action will automatically delete all the already existing links between the document content and the concerned plugin concepts. The user must validate his choice before deleting the linked concepts.
- the **Check/Uncheck** concept to link action which is available by checking the box of a concept in the model links part. It helps users to link the current selection (in the document part) to one or more model concept (in the model links part).

3. WIP – Ergonomics Guideline of the Locations view

The locations view is used to filter and locate all the locations linked to a given selection in the editor.

Several actions and areas are available in this view:

1. The “Select knowledge base to work with” action which leads to define or to create the knowledge base that will be used.
2. The “Group by location” action which leads to group the displayed information basing on the location container of all linked locations.
3. The “Filtering” actions which help to show/hide valid, changed and/or deleted locations links.
4. The “Define scope” action which is needed to select resources in the scope of the currently edited resource.
5. The “Filter” area which can be used to display a subset of linked locations.
6. The “linked locations” area which displays all the linked locations to a given selection from the editor.

The whole set of linked locations must be displayed in the outline to give the user a global information about all the linked location in the editor.