

MagicUWE – A CASE Tool Plugin for Modeling Web Applications*

Marianne Busch¹ and Nora Koch^{1,2}

¹ Ludwig-Maximilians-Universität München, Germany

² Cirquent GmbH, Germany

busch@cip.ifi.lmu.de, kochn@pst.ifi.lmu.de

Abstract. Adequate tool support is a crucial factor of success for a software development method or technique. In this paper, we present the MagicUWE tool, that supports systematic design of web applications with the UML-based Web Engineering (UWE) approach. It provides assistance to the designer for the modeling activities using the UWE profile and the semi-automatic generation of models. MagicUWE is implemented as a plugin for the CASE tool MagicDraw. Focus of the development of the plugin was the usability, adaptability and extensibility of the tool.

1 Introduction

The Unified Modeling Language (UML)¹ provides a rich set of modeling elements and diagram types to cover the specification needs of all kinds of software. In particular, it offers an extension mechanism for the definition of domain specific modeling languages (so-called profiles). The UML-based Web Engineering (UWE) profile is such an extension and therefore, all UML CASE tools can be used to build UWE models. In order to provide an augmented assistance to the UWE modeler, we implemented a plugin for the MagicDraw² tool - called MagicUWE [1] [2]. In contrast to our previous tool ArgoUWE [3], it is based on the UML2 metamodel.

MagicUWE supports the UWE notation and the UWE development process, i.e. it comprises (1) extensions of the toolbar for comfortable use of UWE elements including shortcuts for some of them, (2) a specific menu to create UWE default packages and new diagrams for the different views of web applications (content, navigation, presentation and processes), and to execute model transformations, (3) additional context menus not only for the containment tree but also within navigation diagrams. MagicUWE is easy to install and easy to extend, and the usability is mainly given by intuitive icons, different types of menus and helpful hints.

* This work has been supported by the DFG Project MAEWA II, WI 841/7-2, and the EU FET-GC2 IP project SENSORIA, IST-2005-016004.

¹ OMG - Unified Modeling Language. <http://www.omg.org/docs/formal/09-02-02.pdf>

² MagicDraw. <http://www.magicdraw.com/>

2 Modeling with MagicUWE

In order to get a general idea of MagicUWEs main functionalities, the extensions of the toolbar, the menu and the context menus are introduced in this section. The plugin³ can be downloaded from the UWE website⁴, where a tutorial to UWE and a tabular reference of MagicUWE can be found.

Toolbar. MagicUWE allows selecting model elements for modeling web applications defined by stereotypes directly from the toolbar (see Fig. 1(a)) like navigation classes, and navigation links, index, and all kinds of presentation elements as page, presentation group and presentation properties, input and output user interface elements, etc. In particular, before drawing a presentation property, an element has to be chosen, which should afterwards contain the new property. The name of the new class and property can easily be typed into an input field provided by the plugin, as shown in Fig. 1(b). Properties can be nested and are graphically represented by nested class diagrams.

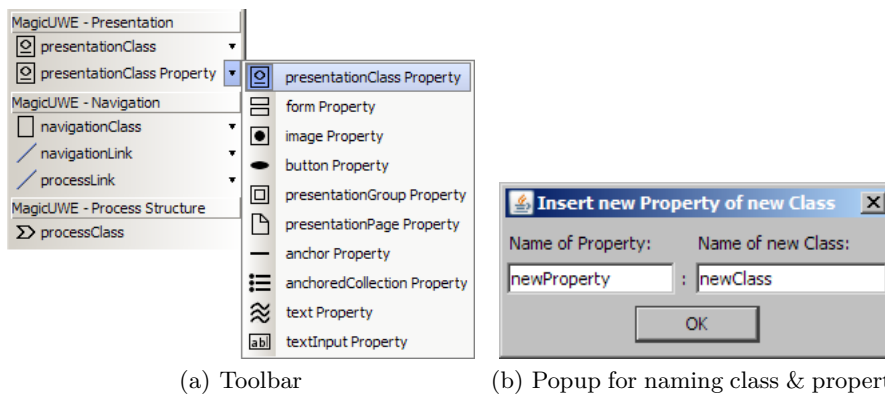


Fig. 1. MagicUWE: Insertion of property from toolbar

An example of the simplification of such a common operation, is the comparison to an insertion of an UWE property without MagicUWE. The user has to include the UWE profile⁵ manually and one stereotype from the UWE profile has to be added to a new class. The properties derived from this class, can be drawn only in elements, that have the **Suppress Structure** value set to **false** in the **Symbol(s) Properties**.

MagicUWE Menu. MagicDraw's main menu provides the functionalities to create the default UWE packages (which are also predefined in the MagicUWE

³ UWE - MagicUWE.

<http://www.pst.ifi.lmu.de/projekte/uwe/toolMagicUWE.html>

⁴ UWE - UML-based Web Engineering. <http://www.pst.ifi.lmu.de/projekte/uwe/>

⁵ UWE - Metamodel and Profile.

<http://www.pst.ifi.lmu.de/projekte/uwe/publicationsMetamodelAndProfile.html>

template) and to store new UWE diagrams in this hierarchy. On the one hand, it is possible to create new empty diagrams like in all UML tools. On the other hand, transformations are available in order to generate step by step drafts of diagrams based on another view of the web application:

- *Content-to-Navigation transformation*: Creates a new «navigationClass» class (in the navigation package) for each class in the content diagram, gives the new class the same name, adds «navigationLink» associations, and finally visualize them in a new navigation diagram.
- *Navigation-to-Presentation transformation*: Generates a new class for each class of a navigation diagram using a presentation specific stereotype. In this case, the «presentationClass» stereotype is added.
- *Navigation-to-ProcessStructure transformation*: Creates a new process structure diagram including a symbol of each class stereotyped «processClass» that is part of a navigation diagram. This is not a complex task, but a convenient method to avoid repetitive tasks to be performed by the modeler.
- *Navigation-to-ProcessFlow transformation*: Prepares a new process flow (an activity diagram) for each «processClass» of a navigation diagram. The new diagrams are labeled like the classes (with the appendix “Workflow”).

Context Menu. If more than one diagram should be transformed, the context menu of the containment tree is the most flexible solution, because several diagrams can be selected as source for one kind of transformation. Two more MagicUWE context menus are located over the classes and associations of navigation diagrams, as can be seen in Fig. 2. The checked menu makes it possible to switch the tagged values {isHome} and {isLandmark} to true or false.

The insertion of access primitives like index and query, is a challenge for the plugin, because it depends on the selected stereotype as well as on the number of associations the user selected before. A concrete example is the insertion of an index and a menu. An index is inserted, if an instance of a class should be selected, i.e. one end of a single association between navigation classes has the multiplicity ‘*’, whereas a menu connects one main class with several classes. A menu will be added with a composition between the menu and the main class.

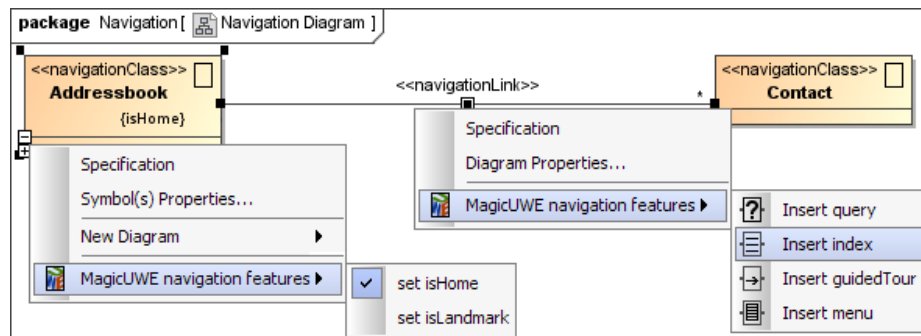


Fig. 2. MagicUWE: Context menus of navigation diagrams

3 Usability, Adaptability and Extensibility

The aim of MagicUWE is to make the design of web applications with UWE and MagicDraw really easy, therefore an intuitive user interface is essential. In order to support valid models, it is especially important to show advices, e.g. if presentation elements are drawn into a navigation diagram. This check is based on the type recognition of the current UWE diagram and compares predefined substrings with the name of the diagrams parent package.

MagicUWE's hint messages can be configured (or even disabled) separately for all UWE diagram types (e.g. the standard for the package of navigation diagrams is `packNavigation=nav`). Thus, adaptability is another main goal of MagicUWE. It is possible to change the name of the included UWE profile, to determine the shortcuts for the toolbar entries and to decide which presentation stereotypes should be displayed in the toolbar as property or as class. Further characteristics of usability are a convenient installer for MagicUWE and the integration of the plugin in the GUI of MagicDraw. In particular, the modeler can also create all UWE diagram types out of MagicDraw's "New Diagram" context menu and not only use the familiar toolbar with its *last used* functionality, but also quickly select the required item, recognisable by its stereotype symbol.

In order to cope with new features of UWE and the next versions of MagicDraw, MagicUWE is designed in a very modular way, the code is well documented and a complex build script is used to create and launch new versions of the plugin automatically. Therefore, the tool is easily extensible and adaptable, which is welcome in the continuous improvement process of the CASE tool.

4 Conclusions and Future Work

In this paper we presented MagicUWE – a CASE tool for designing and generating UWE models characterized by its usability. Implemented as a MagicDraw plugin, MagicUWE is a highly modular and easy to extend software. In fact, the development of MagicUWE is an ongoing project that currently focuses on the extension of the tool with modeling elements for rich internet applications [4] and on further validation mechanisms for model consistency.

References

1. Blagoev, P.: MagicDraw-Plugin for Modelling and Generating Web Applications. Master's thesis, Ludwig-Maximilians-Universität München (2007)
2. Busch, M.: Migration and Extension of the MagicDraw Plugin MagicUWE (German). Project report, Ludwig-Maximilians-Universität München (2009)
3. Knapp, A., Koch, N., Zhang, G., Hassler, H.-M.: Modeling Business Processes in Web Applications with ArgoUWE. In: Baar, T., Strohmeier, A., Moreira, A., Mellor, S.J. (eds.) UML 2004. LNCS, vol. 3273, pp. 69–83. Springer, Heidelberg (2004)
4. Koch, N., Pigerl, M., Zhang, G., Morosova, T.: Patterns for the Model-based Development of Rich Internet Applications. In: Proc. 9th Int. Conf. on Web Engineering (ICWE 2009). LNCS. Springer, Heidelberg (2009)