R. Keil-Slawik, H. Selke, G. Szwillus (Hrsg.): Mensch & Computer 2004: Allgegenwärtige Interaktion.
München: Oldenbourg Verlag. 2004, S. 307–308

Extending UML to GUI Modeling

Kai Blankenhorn, Wilhelm Walter

University of Applied Sciences Furtwangen

1 GUI Design and Software Development

Although the GUI is the most prominent part of a software system, GUI design, particularly GUI layout, i.e. dimensioning and arranging screen elements within a screen, is somewhat neglected by software development. While the interactional parts of user interfaces can be modeled using existing UML diagrams and elements, GUI layout cannot. Additionally, the work of graphic designers and its results are quite separated from software development; for instance, there is no standard way of specifying by which GUI elements the functionality of a particular use case is presented to the user.

2 UML Profile for GUI Layout

We have developed the UML Profile for GUI Layout, a UML 2.0 profile for modeling the layout of graphical user interfaces (Blankenhorn 2004). Its central idea is the ScreenArea stereotype, a rectangular area of screen space that serves a purpose. A ScreenArea can either contain other ScreenAreas or have one or more functionalities, which include static functionalities like images, text or headings, and activatable functionalities like links or forms. The notation of functionalities has been derived from designers' sketches so it can be recognized and interpreted by anyone familiar with that kind of artifact. ScreenAreas can be arranged into a Screen, which contains all elements of the GUI that are displayed at one point of time.

To model a user interface, the user nests and arranges ScreenAreas, creating a high-level mockup of a screen. In contrast to conventional UML diagrams, the layout of a diagram in our profile bears semantic meaning, as this is the only direct way to convey the layout of a screen. Layout information is stored using the recently specified UML 2.0 Diagram Interchange (DI) (OMG 2003b), which is an integral part of the UML 2.0 metamodel that adds size and position information to all UML elements that subclass *Element*. In explicitly storing layout information and sticking to UML's built-in mechanisms, our approach is different from earlier ones that have been developed prior to DI, most notably UWE (Koch 2001) and

OMMMA-L (Sauer & Engels 1999). Our approach is similar to the latter, which uses bounding boxes carrying functionality to partition screen space. Other than OMMMA-L, we use UML's extension mechanisms, creating a standards-conformant extension that can be adopted by tool vendors more easily.

3 Combining with UML Diagrams

Use cases can be associated with their *subject* Classifier, including classes and stereotypes thereof like ScreenArea. This can be used to associate use cases to ScreenAreas in order to model a realizing relationship between the two (OMG 2003a, 519), in which the associated ScreenArea provides the user interface for a part of the functionality of the system that is specified in the use case. As a combination of use cases and ScreenAreas, a use case diagram can be enriched by isolated ScreenAreas to give an overview of how the realization of Use-Cases is spread over the various Screens of the GUI. Secondly, a GUI layout diagram can be enriched by isolated representations of use cases to give an overview of which use cases are realized by a particular Screen.

ScreenAreas can be referred to from within an Activity by creating User, GUI and System partitions and modeling any information that is presented to the user in a ScreenArea as an ObjectNode from the system to the user in the GUI partition. Representing a ScreenArea with an object flow is possible because an ObjectNode "indicates an instance of a particular classifier, possibly in a particular state" (OMG 2003a, 349). Usually, ObjectNodes are used to model a flow of information from one activity node to another. They represent a set of data that is produced by the source node and then used by the target node. In our case, this set of data can be thought of as the information produced by the system and perceived by the user, conveyed by a ScreenArea. Thus, the roles specific ScreenAreas play in a sequence of interactions can be modeled, creating a more complete view of user interface interaction.

References

Blankenhorn, K. (2004): A UML Profile for GUI Layout. Master's Thesis, Department of Digital Media, University of Applied Sciences Furtwangen.

Koch, Nora (2001): Software Engineering for Adaptive Hypermedia Systems: Reference Model, Modeling Techniques and Development Process. PhD. Thesis. Ludwig-Maximilians-Universität München: UNIDRUCK Verlag.

OMG (Eds.) (2003a): UML 2.0 Superstructure Final Adopted Specification (ptc/03-08-02).

OMG (Eds.) (2003b): UML 2.0 Diagram Interchange Final Adopted Specification (ptc/03-09-01).

Sauer, S.; Engels, G. (1999): Extending UML for Modeling of Multimedia Applications. In: Hirakawa, M.; Mussio, P. (Eds.): Proc. 1999 IEEE Symposium on Visual Languages, September 13-16, 1999, Tokyo. IEEE Computer Society, pp. 80-87.