

A Methodology for Model-Driven Development of Crisis Management Applications using Solverational

Andreas Petter, Alexander Behring, Max Mühlhäuser

{a_petter,behring,max}@tk.informatik.tu-darmstadt.de

This paper contributes a user study involving real firemen to support our assumptions about the application of context-aware adaption to the software used by crisis management teams and a methodology to develop models and applications based on our model-to-model transformation language called “Solverational”. This methodology provides a way to develop context-aware applications using Solverational. We apply the methodology to the development of crisis management tools.

A user study motivates the use of context-aware computing for crisis management applications. Our study involved real firemen, which are usually hard to convince to test new software. It focuses on the information manager who plays a major role in the messaging applications of crisis management teams. Because provisions of local laws require the information manager to be the only entity to distribute messages within the team, his work may be a bottleneck for information flow which is critical in the event of a crisis. As a result of the user study messaging applications should adapt to the roles of their users and especially the information manager.

We developed a *methodology, which uses a model-to-model transformation language with constraint solving* at its heart. The methodology provides a way to develop models and applications based on our model-to-model transformation language called Solverational and guides developers in developing context-aware applications.

Our model-to-model transformation language called “Solverational” is based on the QVT Relations standard, but adds constraint solving to it. QVT Relations is a declarative model-to-model transformation language which has been developed by the OMG. However, since QVT Relations does not support constraints (other than equalities, which could arguably be called constraints), Solverational is more expressive. Using these constraints user interface models can be transformed according to context constraints. We present the methodology to develop context-aware applications with Solverational and developed constraints for the development of platform adaptations. These constraints enable target models for different display resolutions. These models come to live using a model interpreter.

We applied the *methodology to the development of crisis management applications*. There, the information manager uses a messaging application to distribute messages within the crisis management team. We use our tools to adapt these messaging applications. In the future we will integrate several adaptations into the transformation, most notably role based adaptation, as directly motivated by the user study.