Multi-Agent Model to Multi-Process Transformation

A housing market case study

Gerhard Zimmermann

University of Kaiserslautern

Keywords: Agent technology, User activity modeling, User activity simulation, Software engineering, Code generation, Software process

Abstract: Simulation is a means to help urban planners and investors to optimize inhabitant satisfaction and return on investment. An example is the optimal match between household preferences and property profiles. The problem is that not enough knowledge exists yet about dynamic user activity models to build reliable and realistic simulators. Therefore, we propose a modeling and software technique that produces simulator prototypes very efficiently for the development, test, and evaluation of many different user activity models, using executable models, code generation, and a domain specific software process. As a specific feature, the model is based on many agents acting independently from each other and that are mapped in several refinement steps into the same number of concurrent processes. The housing example is used as a case study to explain the process and show performance results.

1. INTRODUCTION

User activities in urban and building environments gain more and more interest in architectural design and evaluation tasks. Since user activities are non-deterministic and since model abstractions of these activities are not yet well understood or validated, tools are necessary to develop and test user activity models and to apply such models in design decision processes. Computer simulation is a means to implement such models and to execute, observe, and analyze user activities with different initial and boundary conditions.

Although a number of simulators exist for applications such as motorized and pedestrian traffic, models and simulators for other kinds of user behavior...
in urban and building environments still have to be developed and tested. One problem is that of finding the appropriate models for the desired applications. We specifically concentrate on dynamic models of many individual users instead of stochastic models of user groups because traffic simulations have shown that human behavior is modeled more realistically that way. Appropriate in this context means that because of the high complexity of such models the abstraction of the reality should be as simple as possible without missing essential details. Therefore, many experiments with different abstractions are necessary to arrive at models with the right amount of details.

This leads to the next problem, the implementation, execution, and variation of models. Advances in computing and software techniques in the last decade have provided us with a number of options to solve this problem as efficiently as possible. Efficiency is very important because during the development phase of models a large number of different experiments are necessary. Therefore, we propose and demonstrate a modeling and software prototyping process that is based on executable models and code generation tools. Executable models are formal models which include dynamic behavior and can be either interpreted as or compiled into executable computer programs. We will use a case study to demonstrate the process and show efficiency data.

2. STATE OF THE ART

Our work is based on three main areas: user activity modeling, simulation, and software engineering. We cannot cover all of them completely and will restrict ourselves to some representative examples.

Static user activity modeling in building environments was introduced by Eastman and Siabiris (1995) and by Eckholm (2001). In the IFC models user activities are not yet included. We also presented an effort to integrate user activities into a comprehensive building system model (Zimmermann, 2003) as a basis for extending it to user dynamics.

Models exist or are under development for pedestrian movements (Kukla et al. 2002) or in shopping centers (Borgers and Timmermans, 2005). The models are either based on cellular automata with a very restricted horizon and no differentiation of individuals or on agent systems with a much larger horizon and the possibility of giving different properties to each individual. Still, cellular automata exhibit much more realistic pedestrian behaviors than stochastic models and because of the regularity, synchrony, and simplicity can be executed with very large numbers of individuals. Agent models can implement much more complex behaviors and are in principle asynchronous. The possible size of such simulations has to be seen.