

# A Model-Driven Engineering Approach for Immersive Mixed-Reality Environments

Marija Nakevska<sup>1</sup>, Jasen Markovski<sup>2,\*</sup>, and Matthias Rauterberg<sup>1</sup>

<sup>1</sup> Department of Industrial Design,

<sup>2</sup> Department of Mechanical Engineering,

Eindhoven University of Technology,

P.O. Box 513, 5600 MB Eindhoven,

The Netherlands

{m.nakevska, j.markovski, g.w.m.rauterberg}@tue.nl

**Abstract.** We propose a model-based engineering approach for development of immersive mixed-reality environments based on supervisory control theory that provides for automated software synthesis. The proposed approach greatly improves the consistency of the design process by employing models as means of communication, whereas supervisory control synthesis caters for system flexibility and evolvability.

## 1 The ALICE Installation

Cultural computing is a new human-computer interaction paradigm that aims to provide an interactive experience, closely related to the core aspects of a culture [5]. The ALICE project [4] implements an immersive mixed-reality installation that aims to address important cultural determinants, such as perception, rationality, logic, and self. Immersive mixed reality merges real and virtual worlds to produce new environments that confronts the user in an intense and a seemingly real experience [6]. The ALICE project creates an experience based on the novel "Alice's Adventures in Wonderland" by L. Carroll. The installation comprises six stages over two floors of 12 by 12 meters, employing a large palette of hardware and techniques, such as sensors, actuators, virtual reality, and embodied and virtual agents.

Each stage simulates a certain environment, e.g., nature mimicking, simulation of falling, or changes of the perception of the relative size of the space, corresponding to different parts of the narrative. The participant is engaged in an interactive experience, which includes movements, distortions of the concept of space, and counter-intuitive interactions [9]. The narrative scenario is implemented into a centralized control software, which relies on a unifying communication interface with the sensor and actuator components that we developed.

The production of immersive mixed-reality environments demands efforts from contributing experts from various fields, like artists, interaction designers, and software and electrical engineers. The collaboration within such a diverse

---

\* Supported by Dutch NWO project: ProThOS, no. 600.065.120.11N124.

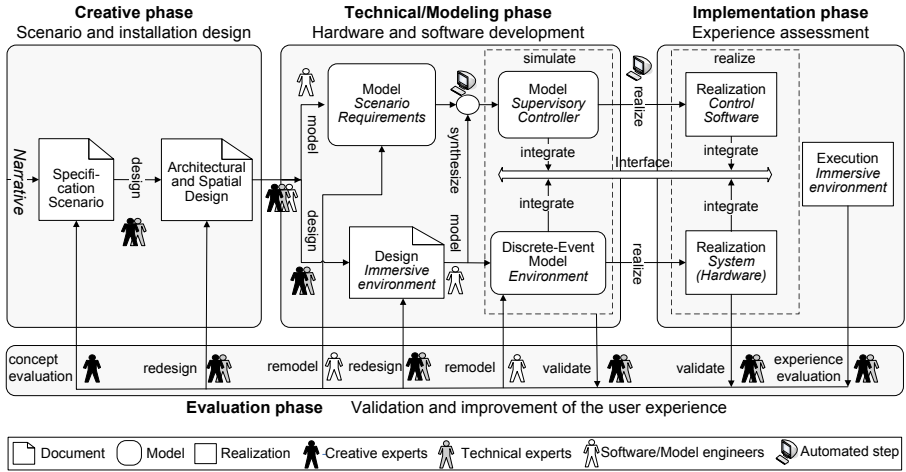


Fig. 1. The proposed model-driven engineering framework

environment requires additional clarity in the architectural, spatial, and control design. Moreover, the user experience is constantly evaluated and improved, demanding changes and additions to the existing features, e.g., additional audio or lighting effects. As a result, the installation is constantly changing and evolving.

Taking in consideration the number and diversity of the involved components and technologies, the maintenance and development of this installation becomes a daunting task. To mitigate system complexity and to ensure user and system safety, we search for engineering approaches that can guarantee safety and cater for flexibility and evolvability of the system. One such advocated approach is model-driven engineering, where models play a central role as means of communication between the involved parties [7].

## 2 A Synthesis-Based Model-Driven Approach

In this paper we propose a model-driven approach for design and development of immersive mixed-reality environments that relies on supervisory software synthesis [8,2]. The synthesis is based on the models of the underlying hardware and the coordination requirements, which implement the desired interactive scenario as interplay between the sensors and the actuators.

We distinguish four phases in the design and development of immersive mixed-reality environments: creative, technical and modeling, implementation, and evaluation phase, as abstractly depicted in Fig. 1. We also recognize three groups of contributing parties: (1) creative experts, e.g., media designers, (2) technical experts, e.g., system engineers, and (3) software and model engineers [1].

The modeling process begins with the narrative that inspires the immersive mixed-reality user experience. An informal specification of the scenario follows,

made together by the creative and technical experts. The scenario is most importantly influenced by the various design and implementation decisions, like choice of the medium or implementation of the hardware control. These decisions result in an architectural and spatial design of the environment.

The technical phase begins with modeling of the components, such as sensors and actuators. The environment behavior is specified as the parallel composition of the models of its components [2,3] that may synchronize to deliver the desired experience. Once the environment is modeled, the software engineers model the scenario requirements by specifying safety properties of the system.

The models of the environment and the scenario, as given by the coordination requirements, are input for the synthesis tool which automatically generates a model of a controller. The latter can be coupled with the model of the environment to analyse the scenario by interactive simulation or formal verification [3], before the physical environment and control software are actually built.

In the implementation phase, the control software is generated based on the synthesized models. The scenario and control structure are integrated with an interface to the physical realization of the environment, depicted below in Fig. 1. Once the realization is completed, the interactions of the users are studied, evaluating the implemented concepts and user experience [9].

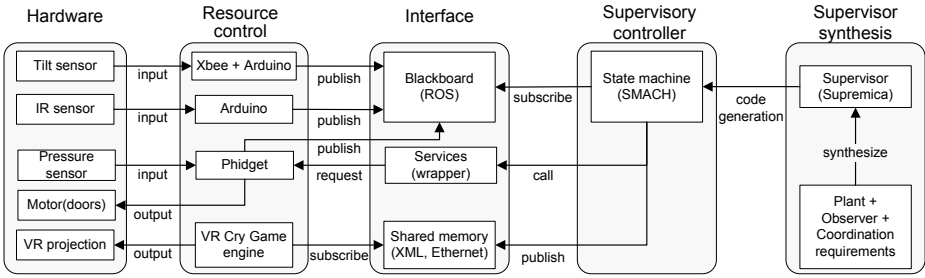
The scenario is adapted often to enrich the user experience, requiring changes in the requirements. We reuse the models to re-synthesize supervisory control software for the new scenario. The framework of Fig. 1 enables model reuse, whereas the automated synthesis significantly improves system evolvability.

### 3 Case Study: Eat Me, Drink Me

”Eat Me, Drink Me” is one of the stages in the ALICE project. Upon entering the stage, detected by a pressure floor sensor, the participant is ‘trapped’ by a sliding door that closes behind. The room has transparent walls with back-side projections, which alter the perception of the relative size of the room by a virtual reality projection dependent on the interactions.

The virtual reality is implemented as a five-sided CAVE, see [www.crytek.com](http://www.crytek.com) for synchronization of the audio and video streaming. As interaction props in this stage, we use a cookie box labeled ‘Eat Me’, and a bottle labeled ‘Drink Me’. The cookie box is equipped with an infrared motion sensor that detects whether a user takes a cookie, whereas the bottle is equipped with a tilt sensor that detects the action of drinking. Users in the CAVE are detected by pressure sensors on the floor and the actuation of the doors implemented with motors.

We depict the hardware and its interface to the control architecture in Fig. 2. We provide for a unified interface to the hardware resource control by employing a Blackboard interface implemented using the Robot Operating System (ROS), see [www.ros.org](http://www.ros.org). The supervisory controller is implemented as a state machine in SMACH, as a component of the system. The model of the controller is synthesized by employing Supremica, see [www.supremica.org](http://www.supremica.org).



**Fig. 2.** Control architecture and hardware interfaces of stage 3 of the ALICE project

## 4 Concluding Remarks

We proposed a model-driven approach for design and development of immersive mixed reality environments that relies on supervisory control synthesis to automatically generate the control software that implements the interactive scenario. We find that the use of formal models is a key element for successful application of a synthesis-based engineering process. The proposed framework most importantly affects the control software development process, switching the focus from interpreting requirements, coding, and testing to analyzing requirements, modeling, and validating the behavior of the system. The proposed control architecture and software generation approach greatly improves model reusability and the flexibility and evolvability of the mixed reality environment.

## References

1. Alcorn, S.: Theme Park Design: Behind the Scenes with an Engineer. Theme Perks Incorporated (2010)
2. Cassandras, C., Lafortune, S.: Introduction to discrete event systems. Kluwer Academic Publishers (2004)
3. Forschelen, S.T.J., Mortel-Fronczak, J.M., Su, R., Rooda, J.E.: Application of supervisory control theory to theme park vehicles. *Discrete Event Dynamic Systems* (4), 511–540 (2012)
4. Hu, J., Bartneck, C., Salem, B., Rauterberg, M.: ALICE’s adventures in cultural computing. *International Journal of Arts and Technology* 1(1), 102–118 (2008)
5. Kooijmans, T., Rauterberg, M.: Cultural computing and the self concept: Towards unconscious metamorphosis. In: Ma, L., Rauterberg, M., Nakatsu, R. (eds.) ICEC 2007. LNCS, vol. 4740, pp. 171–181. Springer, Heidelberg (2007)
6. Milgram, P., Kishino, F.: A taxonomy of mixed reality visual displays. *Transactions on Information and Systems* 77(12), 1321–1329 (1994)
7. Ogren, I.: On the principles for model-based systems engineering. *Systems Engineering* 3, 38–49 (2000)
8. Ramadge, P.J., Wonham, W.M.: Supervisory control of a class of discrete-event processes. *SIAM Journal on Control and Optimization* 25(1), 206–230 (1987)
9. Rauterberg, M.: From personal to cultural computing: how to assess a cultural experience. *uDayIV—Information nutzbar machen*, 13–21 (2006)