Design Studio for Rigorous System Design using Architecture Styles and BIP

Submitted by Katie Dey on Wed, 04/12/2017 - 12:43pm. Contributor: Anastasia Mavridou

ABSTRACT
When building large concurrent systems, one of the key difficulties lies in coordinating component behavior and, in particular, management of the access to shared resources of the execution platform. Components may interact through buses, shared memories, and message buffers, leading to resource contention and potential deadlocks compromising safety-critical operations. The concurrent nature of such interactions is the root cause of the complexity of the resulting software. In order to analyze the behavior of such a system, one has to consider all possible interleavings of the operations executed by its components. Thus, the complexity of software systems is exponential in the number of their components, making a-posteriori verification of their correctness practically infeasible. An alternative approach consists in ensuring correctness-by-construction by applying automatic transformations to obtain executable code from formally defined models [1].

Following this latter approach, we use a notion of architectures defined in [2] to formalize design patterns for the coordination of concurrent components. Architectures provide means for ensuring correctness-by-construction by enforcing global properties characterizing the coordination between components. An architecture can be defined as an operator A that, when applied to a set of components B, builds a composite component A(B) meeting a characteristic property F. Composability is based on an associative, commutative, and idempotent architecture composition operator : architecture composition preserves the safety properties enforced by the individual architectures.

To enhance usability and support scale of architectures, we specify architecture styles, which are families of architectures sharing common characteristics such as the type of the involved components and the properties they enforce. Architecture styles define all architectures for an arbitrary set of components, which satisfy minimal assumptions on their interfaces. Informally, architecture styles can be understood as solutions to coordination problems, e.g., mutual exclusion, data access control, authentication, resource management, etc. To enforce, for instance, mutual exclusion on a set of components, an engineer would need to instantiate an architecture from the mutual exclusion architecture style, compose it with other architectures if needed, and apply it on the components.

We present a design studio developed using the WebGME tool [4] for the specification of architecture diagrams [3], which is a graphical language that describes architecture styles with rigorous semantics. WebGME is a novel, web- and cloud-based, collaborative, scalable (meta)modeling tool that supports the design of Domain Specific Modeling Languages (DSML) and the creation of corresponding domain models. The operational semantics of architecture styles are based on the Behaviour-Interaction-Priority (BIP) framework [1] for the component-based design of concurrent systems. BIP is supported by a toolset including compilers for generating code executable by dedicated engines, as well as tools for deadlock detection, state reachability analysis, and an interface with the nuXmv model checker.

The demonstrated design studio provides support for graphically defining and checking the consistency of architecture diagrams, instantiating and applying architectures, integrating component behavior specifications, and translating architecture specifications into the BIP framework. The design studio is accessible via the CPS-VO Portal.

Anastasia Mavridou is a Postdoc at the Institute for Software Integrated Systems, Vanderbilt University, where she works with Prof. Janos Sztipanovits. Her research interests lie in the area of component-based design, modeling and analysis of concurrent systems with a focus on correct-by-construction techniques. She received her PhD in Computer Science from Ecole Polytechnique Federale de Lausanne (EPFL), Switzerland in 2016 under the supervision of Prof. Joseph Sifakis and Dr. Simon
Bliudze and her MSc in Computer Science from University of Tulsa, USA in 2012.

References:

Anastasia Mavridou  
License: Creative Commons 2.5

Other available formats:

Design Studio for Rigorous System Design using Architecture Styles and BIP
Switch to normal viewer
Switch to experimental viewer

1521617 Academia Presentation HCSS 2017