Title
Demo Abstract: Building IoT Applications with Accessors in CapeCode

ISBN
9781509017720

Authors
Lohstroh, M
Brooks, C
Lee, EA

Publication Date
2016-05-25

DOI
10.1109/ICCPS.2016.7479096

Peer reviewed

Demo Abstract: Building IoT Applications with Accessors in CapeCode*

Marten Lohstroh
University of California
Berkeley, USA
marten@eecs.berkeley.edu

Christopher Brooks
University of California
Berkeley, USA
cxh@eecs.berkeley.edu

Edward A. Lee
University of California
Berkeley, USA
eal@eecs.berkeley.edu

ABSTRACT

We demonstrate CapeCode, a tool for composing actor-oriented building blocks for applications in the Internet of Things design space.

1. INTRODUCTION

The *Internet of Things* (IoT) concept has received a lot of attention in the recent years, but in spite of intensive research efforts in both industry and academia, it is still quite far from its realization. The simplicity of the idea is deceptive, as the engineering challenges faced by the IoT are formidable. They can be compared to the ones that took decades to vanquish in order to turn "Arrays of Transistors" into the powerful microprocessors we have today.

The common denominator between *things* may be their network connection, but aside from that, their characteristics spread out along a wide spectrum of protocols, software stacks, and hardware platforms. Not only does interoperability of current-day technology suffer from "stovepipe" solutions, but the sheer diversity in requirements among different classes of IoT objects precludes the prospect of forging a universal standard to tie them together.

By networking sensors and actuators, IoT does not only enhance the reach of Cyber-Physical Systems (CPS), it changes their nature. Tools and techniques for modeling, analysis, and synthesis of CPS are effective when the system is assumed to be *closed*, yet IoT systems are inherently *open*. Moreover, the utility of a CPS is no longer exclusively determined at design time; it may evolve, and so may the system it is embedded in.

*This work was supported in part by TerraSwarm, one of six centers of STARnet, a Semiconductor Research Corporation program sponsored by MARCO and DARPA.

2. ACCESSORS

Accessors are actors that provide access to a thing, typically a (remote) service, sensor, or actuator [1]. The accessor abstracts away the details of the thing that it encapsulates. Accessors are reusable components that are composable with other accessors through an execution platform called a *host*. The host governs the interaction of components through the rigorous application of models of computation, offering clear abstractions for timing and synchrony, and guarantees with regard to determinism. The host uses the accessor as if it were a local object. An accessor host is to the Internet of Things what a browser is to the Internet. It renders a remote service by locally executing a proxy for a resource.

Accessors address the aforementioned problems by providing an abstraction that accommodates heterogeneous systems of systems through platform-based design. Evolution is supported by enabling seamless transitions between design and deployment, and vice versa, through unified access to simulated and deployed subsystems using an actor interface. Hence, accessors provide an ideal framework for the development of design methodologies for evolving heterogenous complex systems in the IoT.

3. DEMONSTRATION

We present our work in progress through CapeCode[1], a modeling and simulation environment based on Ptolemy II that implements an accessor host. We demonstrate how to create IoT applications through its graphical block-diagram editor and show how we combine simulated components, physical sensors and actuators, and Web services under a discrete-event semantics.

4. REFERENCES

[1] E. Latronico, E. A. Lee, M. Lohstroh, C. Shaver, A. Wasicek, and M. Weber. A Vision of Swarmlets. *IEEE Internet Computing, Special Issue on Building Internet of Things Software*, 19(2):20–29, March 2015.

[1] `http://chess.eecs.berkeley.edu/capecode/`