Introduction to the Turning Smart, Challenges and Experiences in Smart Application Development Minitrack

Peter Salhofer
FH JOANNEUM
peter.salhofer@fh-joanneum.at

Sandra Gesing
University of Notre Dame
sandra.gesing@nd.edu

Charlie Catlett
Argonne National Laboratory
catlett@anl.gov

1. Introduction

While Smart Applications - and more specifically Smart City Applications – have been a really hot topic for years now, the minitrack "Turning Smart: Challenges and Experiences in Smart Application Development" is offered for the first time at HICSS. Although there have already been minitracks on Smart Cities within the E-Government track dealing with concepts, potential fields of application and their impact on society and politics, we want to have a clear focus on the implementation of smart applications. Thus, this minitrack was positioned as part of the Software Technology track.

Our overall goal was to provide an overview of current efforts in actually developing such applications, seeking innovative software efforts that might serve as guidelines, recommendations or simply as the source of new ideas. We are happy that we could accept four excellent papers that demonstrate different approaches on how to plan or to implement smart applications in various fields.

2. Minitrack Content

The first paper focuses on an API that will help application programmers to interact with sensor- and controller-networks that are typically found in building automation systems (BAS). With the help of this API it should be possible to uniquely access devices that are exposed via custom APIs from different vendors (e.g. Apple, Amazon, Google…). The basic idea behind this approach is the concept of so called capabilities, that are provided by abstract devices. By supporting the basic concepts of 'devices', 'rooms', 'groups' and 'functions/capabilities' every application can interact with the underlying infrastructure. For example, the capabilities available in which room can be readily determined without worrying about the actual hardware or automation system used.

A second paper presents an approach on simulating smart applications prior to implementation in the field and therefore optimizing details such as the number of sensors and actuators by comparing different scenarios. This approach might turn out to be really helpful in reducing the total cost such systems, since the number of sensors not only influences the cost of the initial hardware installation but also the intensity and character of the dataflow and the number and complexity of events and interactions.

A very interesting approach in modeling the components of smart applications is presented in the third paper. The authors demonstrate how to use a graph-database to model and to query the topology of sensor-networks. This approach is additionally combined with a Phoenix framework web application, which in turn is backed by Erlang. Since Erlang is known for its reliability and performance, this gives a highly interesting tool stack.

Finally, there is a paper presenting a smart application framework that is currently widely discussed in Europe. FIWARE was developed to support and simplify the implementation of smart applications. This paper presents an evaluation of the FIWARE stack, comprising various tools and subsystems, by implementing a demonstration prototype.

3. Outlook

It was really encouraging to get a reasonable number of excellent papers, although this minitrack was offered for the first time. Since there is a lot of development going on in the field of Smart Applications we are already looking forward to re-organize this track for next year's addition of HICSS.

HℑCSS