Internet of Things: A Challenge for Software Engineering

Charles Consel, Milan Kabáč

an open-source mobile agent (Gloo Android application) with profile-management capabilities.

The main challenges that were faced in this context were: smart object discovery (NFC tags), universal smart object accessibility, mobile interface and sensor access, user profiling, and ad-hoc "on-the-fly" interaction between smart objects (heterogeneity). We developed several scenarios, such as the Location Sensor-Based Services outside Home (see Figure 2). The mobile agent running on the device continuously receives the location coordinates from its GNSS sensor and compares those locations with a user defined proximity area. Whenever the user enters this proximity area, the smartphone automatically triggers a series of requests (based on user-profile and preferences) to control smart objects (for example switching on the heating to a certain temperature).

The results to date are very promising: we integrated the previous smart objects with many partners in order to compose an Open Smart Neighbourhood ecosystem on top of the Web of Objects platform. This allowed us to deploy and test some recent research results in projects related with the Internet of Things, Machine to Machine communication (M2M) and Ambient Intelligence systems development. We have acquired insight into the use of smart WoO objects such as mobile devices, user profiles, location rules, requests, door-locks and smart home objects.

Figure 2: GNSS location-based automatic house entities requests

In our research, we collaborated with several SMEs, Universities and Research Centres, such as DEIMOS, VISUAL TOOLS, TELESPAZIO, ETIC, UPC, UPM, UPV (Spain), Thales, Odonata, Sogeti, UPEM, CEA, IMT (France), Concordia University (Canada), KAIST, KT, ETRI, Kwangwoon and Hankuk University, Innopia, Miksistem (Korea) and Smartec, Nma, University of Cairo (Egypt).

This research is also part of a horizontal task force with other ICT Future Internet projects - such as FIWARE, BUTLER and SOFIA - that deals with building new innovative applications and services for every-day working and living environments. Our work is partially supported by the Spanish MEC INNCORPORA-PTQ 2011, MiTYC TSI-020400-2011-29, and FEDER programs.

Links:
http://www.web-of-objects.com
http://www.youtube.com/user/webofobjectsproject

References:
[1] F. Belqasmi, R. Glitho, C. Fu: "RESTful web services for service provisioning in next-generation networks: a survey", Communications Magazine, IEEE , vol.49, no.12, pp.66-73, 2011
[2] D. Niyato, L. Xiao, L, P. Wang: "Machine-to-machine communications for home energy management system in smart grid", IEEE Communications Magazine, pp. 53-39, vol 49, 2011
[3] L. Ni, Y. Zhu, and M. Jian: "Semantic Sensor Net: An Extensible Framework", Springer, 2005.

Please contact:
Christophe Joubert
Prodevelop, Spain
E-mail: cjoubert@prodevelop.es

Internet of Things: A Challenge for Software Engineering

by Charles Consel and Milan Kabac

The Internet of Things (IoT) has become a reality with the emergence of Smart Cities, populated with large amounts of smart objects which are used to deliver a range of citizen services (e.g., security, well being, etc.) The IoT paradigm relies on the pervasive presence of smart objects or "things", which raises a number of new challenges in the software engineering domain.

The Object's World project

There are an abundance of research and industry initiatives that have been undertaken with the aim of promoting the emergence of IoT [1]. In line with this goal, the Object's World project brings together stakeholders from different domains to build and support the emergence of an IoT sector in France and beyond. The project is lead by SIGFOX, the world's first cellular network operator dedicated to low-bandwidth wireless objects. The cooperation between industry and research partners (e.g., sensor manufacturers, computer science and electrical engineering research labs) is of uttermost importance in overcoming technological barriers. This issue is currently hindering the development of an IoT sector. The main objectives of this project are the development of:

- expertise in the low-bandwidth network sector,
- low-cost transmitter/receiver chips,
- low-energy autonomous sensors, and
- software frameworks which cover the entire lifecycle of IoT applications.

Network infrastructures which support huge numbers of objects open up a range of opportunities for innovative services. Critically, these new opportunities rely on the ability to address the software engineering challenges of this new sector. We promote an approach that revolves around software frameworks. In areas such as mobile and web development, this approach has already been shown to facilitate software development by abstracting over implementation details and guiding the programmer.

A design-driven development approach

To guide and support the development of applications which orchestrate networked objects, our research group introduced a design-driven software development approach which draws on principles and techniques taken from the programming language domain.

In particular, we developed DiaSuite [2], a tool-based methodology which guides the developer through the entire lifecycle of an orchestrating application (Figure 1). DiaSuite offers a design language, providing high-level, declarative constructs that are dedicated to describing the application's architecture, along with the smart objects it orchestrates. The methodology relies on a compiler that generates support in the form of a Java programming framework, customized with respect to a given application design. By providing the developer with a programming framework, our approach ensures conformity between the design and implementation. The generated programming framework provides the developer with an abstract class per component declaration. The application logic is implemented by subclassing each abstract class and programming its abstract methods. To further ease the development of orchestrating applications, DiaSuite relies on Eclipse to guide developers during the implementation phase by introducing placeholders that need to be provided with code. Finally, our approach provides developers with a back-end to address the deployment and execution of orchestrating applications.

Orchestrating smart objects at a large scale

The development of orchestrating applications which are responsible for large numbers of smart objects raises a number of challenges. We have addressed these by introducing a new design language.

Service discovery

Standard service discovery at the individual object level does not address the needs of applications orchestrating large numbers of smart objects. Instead, a high-level approach which provides constructs to specifying sub-sets of interest is needed. Our approach allows developers to introduce application-specific concepts (e.g., regrouping parking spaces into lots or districts) at the design time and then these can be used to express discovery operations. Following our design-driven development approach, these concepts are used to generate code to support and guide the programming phase.

Data gathering

Applications need to acquire data from a large number of objects through a variety of delivery models. For instance, air pollution sensors across a city may only push data to the relevant applications when pollution levels exceed tolerated levels. Tracking sensors, however, might determine the location of vehicles and send the acquired measurements to applications periodically (e.g., 10 min. intervals). Data delivery models need to be introduced at design time since they have a direct impact on the application's program structure. In doing so, the delivery models used by an application can be checked against sensor features early in the development process.

Data processing

Data that is generated from hundreds of thousands of objects and accumulated over a period of time calls for efficient processing strategies to ensure the required performance is attained. Our approach allows for an efficient implementation of the data processing stage by providing the developer with a framework based on the MapReduce [3] programming model which is intended for the processing of large data sets.

Future work

We envisage to enrich our design-driven methodology with support for simulation of infrastructures of smart objects. To achieve this, we will leverage design-time declarations to generate application-specific simulation support, while keeping the application code unchanged.

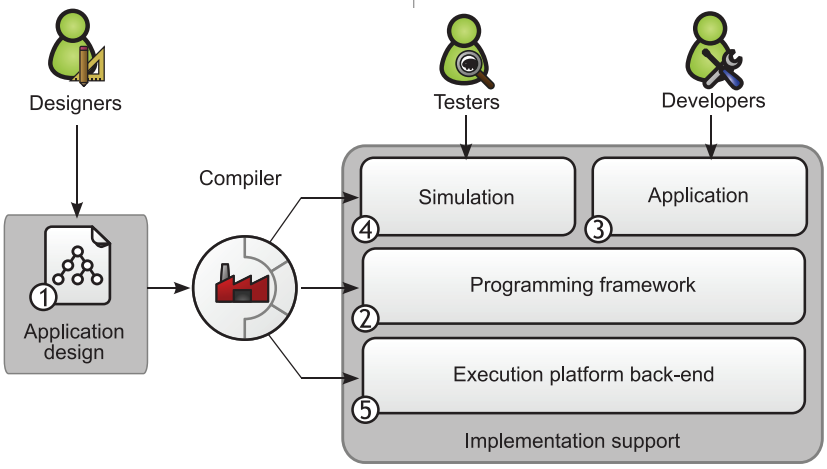

Figure 1: The DiaSuite tool-based methodology

Links:

http://www.sigfox.com/en/
http://www.telecom-design.com/en/
http://phoenix.inria.fr/

References:

[1] D. Miorandi et al.: "Internet of things: Vision, applications and research challenges," Ad Hoc Networks, vol. 10, no. 7, 2012
[2] D. Cassou, et al., "Towards A Tool-Based Development Methodology for Pervasive Computing Applications," IEEE Transactions on Software Engineering, 2011
[3] J. Dean and S. Ghemawat, "MapReduce: simplified data processing on large clusters," Communications of the ACM, 2008

Please contact:

Charles Consel
University of Bordeaux / Inria
Bordeaux - Sud-Ouest,
E-mail: charles.consel@inria.fr

Milan Kabac
Inria Bordeaux - Sud-Ouest,
E-mail: milan.kabac@inria.fr