Chapter 1
Introduction to Smart Environments

ABSTRACT

Now the Internet of Things (IoT) is growing fast into a large industry with huge potential economic impact expected in near future. The IoT technology evolves to a substrate for resource interconnection and convergence. The users' needs go beyond the existing web-like services, which do not provide satisfactory coupling and automatic composition when the user tries to solve tasks from her/his everyday life. New generation of services (named "smart services") emerges. In this chapter, we introduce the problem of effective use of the multitude of IoT-enabled devices and other digital resources that now surround our lives. The devices support and assist human by provision of digital services. This is the key objective of a smart environment. Our focus is on such a particular class of smart environments as smart spaces. This class targets IoT-enabled computing environments, where a smart space is created and then provides an infrastructure for applications to construct and deliver value-added services based on cooperative activity of environment participants, either human or machines.

INTRODUCTION

We are already surrounded by huge number of various devices. These devices serve different purposes; they have some measuring capabilities (sensor devices), data processing capabilities (processing devices), and possibility

DOI: 10.4018/978-1-5225-2653-7.ch001

to influence the environment (environment control devices). Some of the devices have only one of these capabilities, e.g., sensors, some combine all three functions, i.e., such devices are called smart systems. Most important is that our living environment is changing very fast by the process of total digitalization.

Our lives are changing, thanks to the services delivered by surrounding devices. For example, a smart TV can record our preferred shows. A refrigerator can control what food is missing and remind to buy it by printing a shopping list or sending a text message to the preconfigured phone or email. Smart home sensors and actuators can keep most comfortable temperature in our house, while also optimizing the expenses, e.g., by using most of energy when it has the lowest price. Nevertheless, what is still missing is a scalable ecosystem that enables efficient co-working of all surrounding devices, everywhere at any time.

There have been many attempts to build such ecosystems. A number of studies on various specific topics has been presented recently, see examples in (Korzun, Borodin, Timofeev, Paramonov, & Balandin, 2015; Balandina, Balandin, Koucheryavy, & Mouromtsev, 2015; Korzun & Balandin, 2016). As a simplified ecosystem development scenario, we can consider the following example. A number of solutions exists that enables joint use of various sets of devices by controlling them via centralized management system in a cloud. The problem, however, is that such a solution implies high scalability restrictions. The user becomes strongly dependent on a managing service in a cloud, which is physically located somewhere, possible on another side of the planet. This problem creates a lot of issues and challenges, starting from additional delays due to distant data transmission and queuing before getting service in cloud, to the questions of privacy. Just by cutting the outgoing Internet connections the user will completely lost service. The strangest of all is that the devices that are to deliver required services are still around the user. They have capabilities and resources to provide the services that the user needs.

Another important observation is that most of modern devices have significant embedded memory, processing power and communication capabilities, including support of short range communications, e.g., via Bluetooth. Some devices have advanced embedded control and management system, which could perform various tasks beyond basic functional of the device. Moreover, for the most time we see that the device resources are significantly underused. As a result, we for most of devices there is significant resource redundancy.