Self-management in Internet of Things

Timo Töyry

Aalto University

Abstract—**Along with rise of the Internet of things the number of devices requiring configuration and management will explode. Users would not be able to configure and manage all the devices. The majority of devices in the Internet of things must be able to manage themselves.**

In this paper are represented some methods on wireless sensor networks that can be used to enable self-management on the nodes of the wireless sensor networks and in devices of the Internet of things.

I. INTRODUCTION

In Internet of things (IoT) [1], [2] huge number of small devices will be connected to the Internet in some way. Most likely the devices would be some kind of environmental sensors, but large numbers of all kind of devices are going to exist. Because of the huge number of the devices it will be impossible for people to actually configure and maintain all required devices. This will be an limiting issue for IoT. One solution to the problem is to make the devices autonomous enough so they can manage themselves automatically.

In some visions of IoT looks somewhat similar to current wireless sensor networks (WSN) [3], [4] with one major exception the IoT will be everywhere unlike the WSNs. The IoT will be most likely to be used to collect all kinds of information about the physical world. The information can be then processed to be used on environmental studies, emergency detection, product tracking through manufacturing process and so on [5].

WSNs consist of small, relatively inexpensive and usually battery powered devices, called motes, that have a radio-interface for inter node communication. Motes have often quite modest computational, communication and storage capabilities.

Since WSNs are the closest thing to IoT that currently exist rest of the paper is going to focus on self-management on WSNs [6], [7], [8]. In WSNs self-management can take place on either on network- or mote-level.

The rest of the paper is structured as follows section II introduces some objectives that an autonomous system must met, next section III describes some techniques used on WSN motes that can be used to implement self-management and last in section IV short conclusion of this paper.

II. PROPERTIES OF AUTONOMOUS SYSTEMS

Autonomous systems have some self-management objectives in addition to their mission objectives [9]. To successfully reach the self-management objectives the system must be aware of its and its environments current state, to be able to detect changing conditions and to be able to adjust itself to adapt to changes in the environment. The monitoring and adaptation to changes will require both hardware and software support to be built in the system [10], [11]. The most important self-management objectives are introduced briefly in subsections II-A to II-D.

A. Self-configuration

The system is capable to readjust itself. Readjustment of the system is required if its environment changes or to reach an objective set for the system.

B. Self-optimization

The system can measure its current performance and it able to compare it against to the known optimum level of performance. The system will adjust its operation to reach closer the optimal performance. The system is also able to change its operation to cope with new user set policies.

C. Self-healing

The system tries to recover from faults or to avoid them. Self-healing can be implemented in two different styles. They are reactive and proactive modes. In reactive mode the system detects and recovers from faults as they occur. The system also tries to repair the faulted functions if possible. In proactive mode the system monitors its state to detect and adjust its behaviour before reaching an undesired state.

D. Self-protection

The system defends itself against internal and external threats, which can be accidental, such as cascading failures, or malicious attacks against the system. To manage the threats the system must be aware of its environment and have means to react to detected threats.

E. Self-management in wireless sensor networks

The self-management intelligence is located in the motes of the WSN. Mostly the self-management concerns the motes, however there are also some network level self-management objectives such as self-discovery and self-organization.

Self-discovery means that the motes of the WSN search for services available for them automatically. An example of a service is routing in multihop networks, where the messages for and from motes out of reach of the network controller are routed through other motes that are in reach of the controller.

In self-organization the motes of the WSN adjust the logical topology of the network to improve the reliability and scalability of the network.

III. METHODS TO IMPLEMENT SELF-MANAGEMENT

There are several methods developed to make development, deployment and maintenance of WSNs easier. Some of these methods can be used to implement some self-management features into motes. In next subsections couple of most promising methods are described and explained how they can exploited to implement self-management.

A. Virtual machines

There are some virtual machines designed to run on WSN motes, for example Maté [12] and SwissQM [13]. Virtual machines in WSN enable more flexible development and deployment of WSNs by allowing easy updates of software of the motes.

The virtual machine executing the code application software have complete knowledge of the state of the application. Also the virtual machines can modify the code of the application on execution. So if the virtual machine is smart enough it can adjust behaviour of the applications when needed.

With virtual machines it is also possible to do dynamically reconfigure the running application. Use of virtual machines are usually more costly on resources than use of native code, due the overheads in computation caused by the virtual machine.

B. Dynamic configuration

Dynamic system reconfigurability can be achieved with replacing the application software or alter its configuration on-the-fly. One method to replace an application from live mote is to utilize dynamic linking and loading [6] to change required application modules.

Traditionally most of the configuration of the application is hardcoded to the application binary image, which must be replaced if a configuration change is desired. Replacement of the binary image is costly operation in terms of resource use.

If application logic is separated from the configuration it is easier to replace only needed parts of the configuration to change behaviour of the application, if the desired change in behaviour is small enough so the old application logic can handle the new behaviour too [6], [7]. Replacing configuration is also cheaper operation in terms of resource use than replacement of whole binary image.

Support for dynamic changes on the system configuration is required so the system can be adjusted for new conditions. Reconfigurable system is also the starting point for development of self-managing system that can adjust its operation automatically in response to changes in its environment.

C. System monitoring

System can be monitored in both hardware and software. Main idea is to detect undesired events on the system [6]. With software monitoring it is possible to detect various software bugs that cause undesired behaviour, such as failure to put the CPU into sleep mode to save energy. Also health of the hardware can be monitored. Usually it is monitored with self-tests, which are performed are system start up and possibly also periodically when the system is up and running. The self-tests are performed by executing some specific test patterns and measuring their effect on the system with special hardware probes.

Detected problems are reported to the controller of the network. In a autonomous system the system will also try to fix, or isolate if fixing is not possible, the detected problems. The system will also require some sensors to monitor its environment to be able to fully self-manage itself.

IV. CONCLUSIONS

Self-managing autonomous systems will be a key factor of success of IoT. Without self-management the IoT would not likely grow to be significant thing. Many things studied with WSNs to make their usage easier can be used to increase self-management of the motes. Also many results obtained on WSNs can be transferred quite directly into IoT.

REFERENCES

[1] N. Mitton and D. Simplot-Ryl, "From the Internet of things to the Internet of the physical world," *Comptes Rendus Physique*, vol. 12, no. 7, pp. 669–674, Aug. 2011.

[2] L. Atzori, A. Iera, and G. Morabito, "The Internet of Things: A survey," *Computer Networks*, vol. 54, no. 15, pp. 2787–2805, Oct. 2010.

[3] J. Yick, B. Mukherjee, and D. Ghosal, "Wireless sensor network survey," *Computer Networks*, vol. 52, no. 12, pp. 2292 – 2330, 2008.

[4] I. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "A survey on sensor networks," *Communications Magazine, IEEE*, vol. 40, no. 8, pp. 102 – 114, aug 2002.

[5] M. Chui, M. Löffler, and R. Roberts, "The Internet of Things," *McKinsey Quarterly*, Mar. 2010.

[6] N. Finne, "Towards Adaptive Sensor Networks," Dissertation for the degree of Licentiate of Philosophy in Computer Science, Uppsala University, 2011.

[7] T. Bourdenas and M. Sloman, "Starfish: policy driven self-management in wireless sensor networks," *Proceedings of the 2010 ICSE Workshop*, 2010.

[8] L. Ruiz, T. Braga, F. Silva, H. Assuncao, J. Nogueira, and a.a.F. Loureiro, "On the design of a self-managed wireless sensor network," *IEEE Communications Magazine*, vol. 43, no. 8, pp. 95–102, Aug. 2005.

[9] J. Kephart and D. Chess, "The Vision of Autonomic Computing," *Computer*, no. January, pp. 41–50, 2003.

[10] P. Vanroy, "Self Management and the Future of Software Design," *Electronic Notes in Theoretical Computer Science*, vol. 182, no. contract 004265, pp. 201–217, Jun. 2007.

[11] M. Hinchey and R. Sterritt, "Self-Managing Software," *Computer*, vol. 39, no. 2, pp. 107–109, Feb. 2006.

[12] P. Levis and D. Culler, "Mat: a tiny virtual machine for sensor networks," in *Proceedings of the 10th international conference on Architectural support for programming languages and operating systems*, ser. ASPLOS-X. New York, NY, USA: ACM, 2002, pp. 85–95.

[13] R. Müller, G. Alonso, and D. Kossmann, "A virtual machine for sensor networks," in *Proceedings of the 2nd ACM SIGOPS/EuroSys European Conference on Computer Systems 2007*, ser. EuroSys '07. New York, NY, USA: ACM, 2007, pp. 145–158.