INTERNET OF THINGS — Finland

1 / 2013

From verticals to platforms to applications

CONTENTS

INTERNET OF THINGS *Finland*

Editor:
Samu Varjonen,
University of Helsinki

Graphic Design:
Hanna Sario, Unigrafia

www.iot.fi

TiVit | INTERNET OF THINGS Tekes

Wilhelm Rauss

FINLAND'S NATIONAL INTERNET OF THINGS PROGRAM AS TREND-SETTER TO CONNECT ANYTHING, ANYTIME, ANYWHERE

A new ubiquitous computing and communication era has started silently, and slowly but steadily we experience changes and adaptations in our everyday environments. The widespread use of mobile phones and connected devices has already become indispensable for most of us and expectations for faster, smarter and safer communication networks for a multitude of connected devices are higher than ever before.

Finland's national Internet of Things (IoT) Program helps the Finnish industry to pioneer the development of new products, services and standards for IoT and has a global competitive advantage due to its existing know-how and active cross-industrial cooperation in the Information and Communications Technology (ICT) sector.

In order to prosper on a global level, IoT needs to support a multitude of diverse "smart" objects, which are extended with sensors, actuators, RFIDs or processors. Those objects must be uniquely identifiable and can be monitored or manipulated via various networks; they can autonomously transmit data and communicate with other objects or machines.

Some of the key challenges of our research and development activities are the elaboration of strong security and privacy foundations, development of common IoT platforms, international standardization efforts and efforts to reduce the energy consumption of devices that are attached to objects. Besides that, the IoT Program researches the potential for new lucrative business models, products and services.

Due to its strong background in ICT, the Finnish industry is already a key contributor to IoT standards at IETF, IEEE, 3GPP, ETSI, NFC Forum, W3C, ZigBee Alliance and other relevant standardization forums.

The research and development conducted in the IoT Program is funded by Tekes and steered by Tivit. Our consortium consists of more than 35 national and international partners from big companies, SMEs, research organizations and international cooperation partners. All-in-all more than 250 scientists and international experts take part in our activities.

Tekes is the Finnish Funding Agency for Technology and Innovation. It is the most important expert organization for financing research, development and innovation in Finland. Research, development and innovation funding is targeted to projects that create in the long-term the greatest benefits for the economy and society. Tekes does not derive any financial profit from its activities, nor claim any intellectual proprietary rights.

> *"Billions of connected devices will change our ways of living."*

Tivit is one of Finland's Strategic Centres for Science, Technology and Innovation (In Finnish: "SHOK" or "Strategisen huippuosaamisen keskittymät") and brings together strategically important research programs or projects, thereby giving those involved a framework in which they not only benefit from the wide range of partners involved.

Feel free to visit our website (www.iot.fi) where you can download our "Internet of Things Strategic Research Agenda", find a list of our partners, our publications and additional program information.

This magazine will give an insight into some of the R&D activities performed by our consortium partners within the IoT Program. Enjoy reading!

Wilhelm Rauss
Ericsson R&D

Focus Area Director of Finland's National Internet of Things Program

wilhelm.rauss@ericsson.com

IOT CALLS FOR NATIONAL AND INTERNATIONAL COLLABORATION

In the next decades we will live in a world surrounded by tens of billions of devices that will interoperate and integrate smoothly with the conventional Internet, provide secure and reliable services, enhance the life of people in healthcare, smart homes, industry automation, environmental monitoring and more.

The technical solution for realizing such an interconnected "smart" world is more complex than the setup of the traditional Internet and naturally calls for strong international collaboration. In order to ensure that Finland will become a recognized leader in the IoT domain, Tekes granted financial support for the national IoT Program, which started in the beginning of 2012. The budget for this industry-driven four-year Program is around 60 Million Euros.

The Ericsson R&D Center Finland, who is the biggest investor and driver of the IoT Program, defined together with other consortium partners work package teams with team sizes of up to 60 persons, which perform various research and development tasks that benefit from (inter-) national cooperation. Teams are typically led jointly by one representative from the industry and one from a Finnish university.

We see it as an advantage that experts of otherwise competing companies find common ground by researching and resolving common problems together and by sharing research results within the consortium.

The main objective of WP 1 (Networking and Communications) is the development of new technological solutions, network designs, and architectures that can cope with billions of IoT entities, and connection enablers for the suppliers of the data with the respective consumers.

> *We see it as an advantage that experts of otherwise competing companies find common ground by researching and resolving common problems together and by sharing research results within the consortium.*

The purpose of WP2 (IoT Management) is to propose and solve those technical issues which are important for the management of the IoT devices, gateways and networks in a scalable manner that can support billions of IoT devices used in a large number of IoT applications of different nature. A special focus is put on scalability, security, energy efficiency and autonomous operation of the solutions.

The rationale of WP3 (Services and Applications Development Support) is to provide a layer linking IoT infrastructure with IoT applications and services. The goal is to facilitate service and application development in decentralized, complex, heterogeneous and dynamic environments.

The goal of WP4 (Human Interaction) is to gain a better understanding on the best ways to provide interactive solutions in an IoT environment. The WP seeks to research user experience aspects of interaction with IoT, study the best ways to empower people to configure and access their IoT environment, and apply innovative visualization methods to convey the IoT state, content, and capabilities.

The target of WP5 (Ecosystem) is to support Finnish firms in forming a successful IoT ecosystem by identifying their role in the ecosystem and developing suitable business models.

In WP6 (Trials and Demos) we plan, implement, analyze and showcase ambitious IoT solutions to demonstrate their benefits to our stakeholders and the general public.

Identified XWP (Cross-Workpackage) activities, namely "Security, Privacy and Trust", "Energy Efficiency" and "Standardization and Architectural Issues" are issues, which need to be considered in all work packages

The challenge is the way from silos to platforms

Over time, various vertical industry segments have been solving challenges of the urban population (such as water supply, energy resources, transportation, pollution, public safety, health, corruption, housing, waste etc.) by developing engineering and software solutions, which were not necessarily interoperable with each other. Nowadays, the new trend is to develop "smart" environments, where smart objects can collect, store, exchange or broadcast needed information (such as location, temperature, pollution, meta-information etc.) by utilizing numerous fixed and wireless communication methods.

To improve the interoperability of applications and devices we will need to build software applications on industry-independent platforms, which are using standardized communication protocols. The practical advantage of such platforms is full interoperability and IoT optimized connectivity for applications and smart objects of various verticals. In the IoT Program the way towards such platforms is being researched.

Consortium partners of the IoT Program come from various industry sectors, which gives us an excellent product and research portfolio.

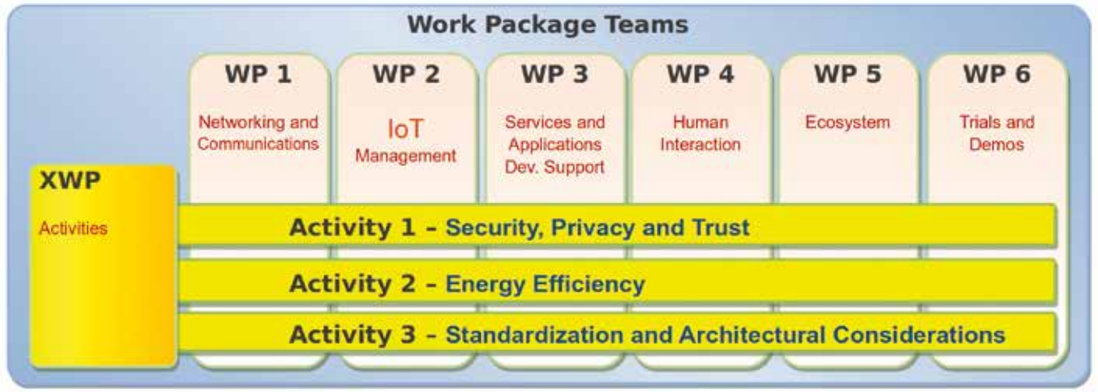


Figure 1. *IoT Program Work Packages*

As an example, big companies such as Intel, Nokia, F-Secure, Elektrobit, Renesas Mobile and Ericsson have a strong background in soft- and hardware development, ICT, security, the automotive and wireless industries and consumer electronics.

Participating SMEs like Mikkelin Puhelin, There Corporation, Mobisoft, Finwe and the Finnet Group bring benefits to our joint research with their experience in IT services, ICT, energy management, home automation, digital services, vehicle communication etc.

On an international level, we are happy to cooperate with other organizations, such as the Wuhan University China, the French Agency for International Business Development, the Finnish-Russian University Cooperation in Telecommunications, Intel USA and other organizations in Europe and Asia.

Eight consortium partners come from Finnish academic research institutions; however, most contributions come from VTT Technical Research Centre of Finland, the University of Oulu, Tampere University of Technology, Aalto University and the University of Helsinki.

You can find an updated list of our partnerships on the Internet: www.iot.fi/partners

Prof. Sasu Tarkoma from the University of Helsinki takes care of the academic coordination of the Program to ensure high-quality IoT research, which is disseminated in world-class conferences, workshops and journals.

Prof. Sasu Tarkoma
University of Helsinki
Academic Coordinator of Finland's national Internet of Things Program

sasu.tarkoma@helsinki.fi

HERE AN EXCERPT OF OUR ACHIEVEMENTS SO FAR:

- **Submission of more than 50 scientific articles for IEEE SECON, ACM SIGCOMM workshop on Mobile Cloud Computing, IEEE Globecom workshop on IoT, ACM ExtremeCom, IEEE Communications, IEEE Network magazine, IEEE Transactions on Mobile Computing, and various conferences**
- **Significant contributions to IETF CoAP and HOMENET, IEEE 802.11ah, 3GPP LTE**
- **Evaluation of cryptographic libraries and algorithms**
- **Feedback to the CoAP resource directory and mirror proxy drafts at the IETF**
- **Research and prototypes for low-power, low-cost sensor networking design for snow environments**
- **State-of-the-art review of M2M communications in the LTE-context from traffic point of view**
- **Literature review related to security and energy efficiency of various resources-constrained networks**
- **The World's first implementation of IETF HOMENET technology; a routed network that configures the routing protocols, network prefixes, router advertisements, DNS, and even NAT64 automatically**
- **General 3D visualization prototypes of IoT**
- **Device management application scenarios, use cases and requirements**
- **Proposal for an IoT identification scheme that works regardless of access method (3GPP vs. non-3GPP)**
- **Participation in the international evaluation contest OAEI 2011.5**
- **IoT market, value networks, and business models: the state of the art evaluation (SOTA report)**
- **New national and international IoT partnerships**

STANDARDS FOR EMBEDDED DEVICES IN THE NETWORKED SOCIETY

Ericsson is known for having a vision of 50 billion connected devices by 2020. This builds on the proposition that anything that can benefit from being connected will be connected. This is the foundation for the Networked Society.

The Networked Society embraces all stakeholders: people, businesses and society in general. Different stakeholders have different interests and drivers for adopting ICT solutions. For people, it is more about lifestyle, fun and "wants" rather than "needs." Enterprises are exposed to an ever-increasing competitive business environment requiring cost reduction, branding and differentiation. From a society perspective, saving energy, sustainability, efficiency and safety are important drivers.

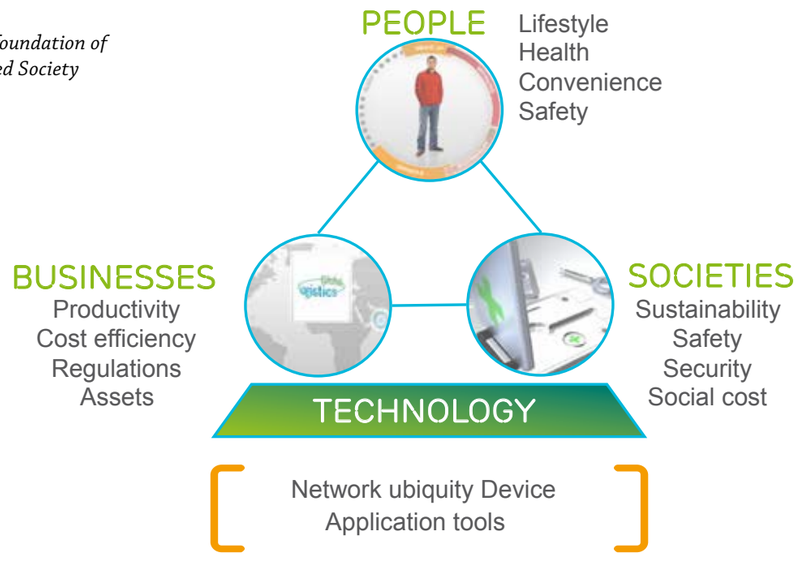


Figure 1. *Drivers and foundation of the Networked Society*

The underlying fundamental enabler that makes this happen is technology evolution. The key enabling technologies are ubiquitous connectivity, smart devices, and the ability to integrate smart objects in different applications. We are now at the meeting point in time where viable technologies are available at the same time as concrete needs from the different stakeholders have emerged.

Pushing the limits of M2M and the intranet of things

The Networked Society builds on personal communications as well as communication embedded in real-world objects or things, i.e. both M2M and the Internet of Things (IoT), the latter representing the bulk of future deployed devices. The things we are interested in are very diverse and range from industrial machines to vehicles, appliances, lights, and buildings. The things are not limited to tangible objects; smart places and environmental observations are very important for many applications.

The application space is very wide; improvements in traffic safety and traffic management, for example. Transforming the electricity grid to a smart grid, driven by new requirements like energy efficiency, microgeneration, electrical vehicles, and consumer energy awareness is another. Agriculture, water management and environmental monitoring are other less technology-intensive usage areas.

These applications are already being deployed today, but the focus is on single applications and most of the time are characterized by "one device - one application." In some cases, even special networks are being built for single applications. We do not believe this will lead to sustainable business in the end.

How can we benefit from the ongoing development, yet allow a richer, more open architecture to emerge? Can we reuse what we are deploying? In order to do this, we have to open up or even break the current application silos.

The Internet of Things

Instead of deploying devices with a single purpose or application in mind, we should allow devices to serve multiple applications, and applications to employ multiple devices. We should also open up and reduce application development costs and time to market by moving away from proprietary and legacy technologies and solutions.

The proposition is to move to a horizontal system with a focus on reuse of common enablers, and a true transformation to using the benefits of IP and Web technologies all the way, even in the tiniest device. Connectivity, access to data, data representation, and processing and storage elements are important common capabilities in such a system.

This will allow a truly open market to develop and deploy the different solution components, allow commodity components to be used, and enable easier interconnection with existing applications and Internet services.

Solutions for the Internet of Things

Needless to say, devices are instrumental for the Internet of Things. We are already witnessing the deployment of a range of different devices. However, this development is only in its infancy, and to get to a true mass market, several technical and commercial challenges have to be solved. Costs for developing and manufacturing the devices need to be further reduced. The availability and compatibility of the devices to different environments need to improve. The ease at which the devices can be deployed also has to improve.

These challenges relate, in part, to ongoing technology development (such as advances in microelectronics and sensors), agreements on standards, reaching economies of scale, and business ecosystems to produce the right equipment at the right price. But one key issue is that the market is currently quite fragmented. Each industry vertical has developed its own technical solutions without much regard to reuse and commonality. In particular, for many industries (such as building automation), the current

> *Current solutions inherit much from past legacy networks whereas off-the-shelf Internet technologies would be a more flexible and inexpensive platform*

solutions inherit much from past legacy networks whereas off-the-shelf Internet technologies would, in many cases, have provided a much more flexible and inexpensive platform. In addition, even in a single industry sector, the number of alternative solutions is large. For instance, in building control and automation, there is KNX, LonWorks, X.10, BACnet and ZigBee to name a few.

What is needed is an architecture that is based on IP, a common set of application tools, and a reasonable set of link layer solutions. We believe that we should start by putting IP into even the smallest devices. Today, IP can run in very constrained devices as well as in very constrained environments [1]. The industry is already on this track as demonstrated by momentum in product releases, standards, and industry alliances such as the IPSO Alliance.

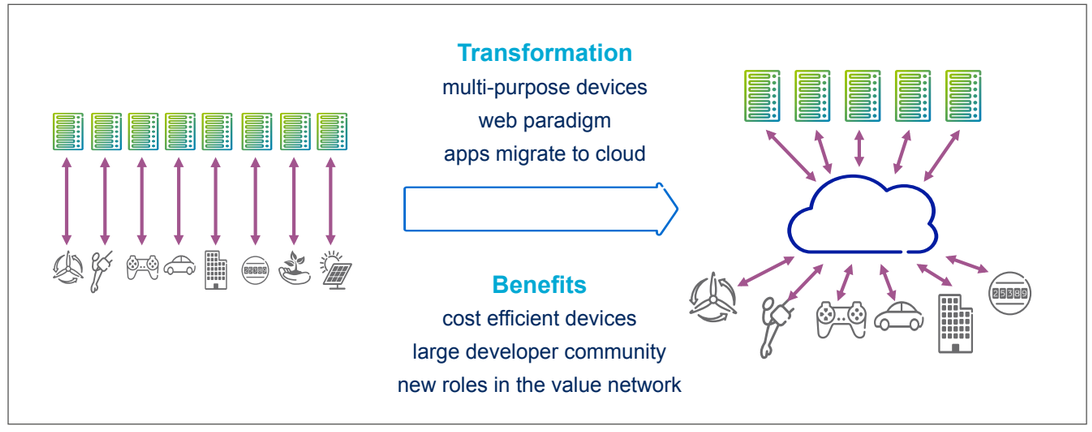


Figure 2. *Moving from silos to an Internet of Things*

From a commercial standpoint, it is also important to build on link layer communications that support multiple applications. Deployment of new IoT devices on existing networking infrastructures is a natural requirement.

Furthermore, we should turn to widely accepted development tools. Today, development is often done with proprietary tools. Going mainstream means that we can make use of the thousands of developers out there. To this end, open APIs are also important, and the prospect of AppStores for IoT devices is attractive.

> *Embedded web services are the means to get the valuable data in and out of the devices, using the well-established technology that is widely used by many developers*

A key concept is that of embedded web services. Embedded web services are the means to get the valuable data in and out of the devices, using the well-established technology that is widely used by many developers. It will also ease the integration to existing Internet services and Enterprise systems. Variants of the Web Services model suitable for the tiniest devices have already been defined. For instance, Constrained Application Part (CoAP) [2,5] employs the REST paradigm but employs a more lightweight solution than HTTP.

It is also necessary to make simple profiles of the sensor data and there are efforts in this direction from both the research community and in standardization. CoRE link formats [3] combined with SenML [4] is one example. Examples of more heavy profiles that are dedicated include ZigBee Smart Energy Profile 2.0 [6], which basically is a vertical application profile that does not differentiate between the data and the application in which it is intended to be used.

Appropriate cloud-based application enablement services are required to ease integration of IoT resources in applications. These include managed connectivity services, IoT device management and IoT resource management. IoT resource management includes discovery and directory services, data capture and integration as well as IoT data and event processing like storage and stream processing. It is important that applications can expose their information to others, discover what other resources exist, and control how their own information is distributed further and federated.

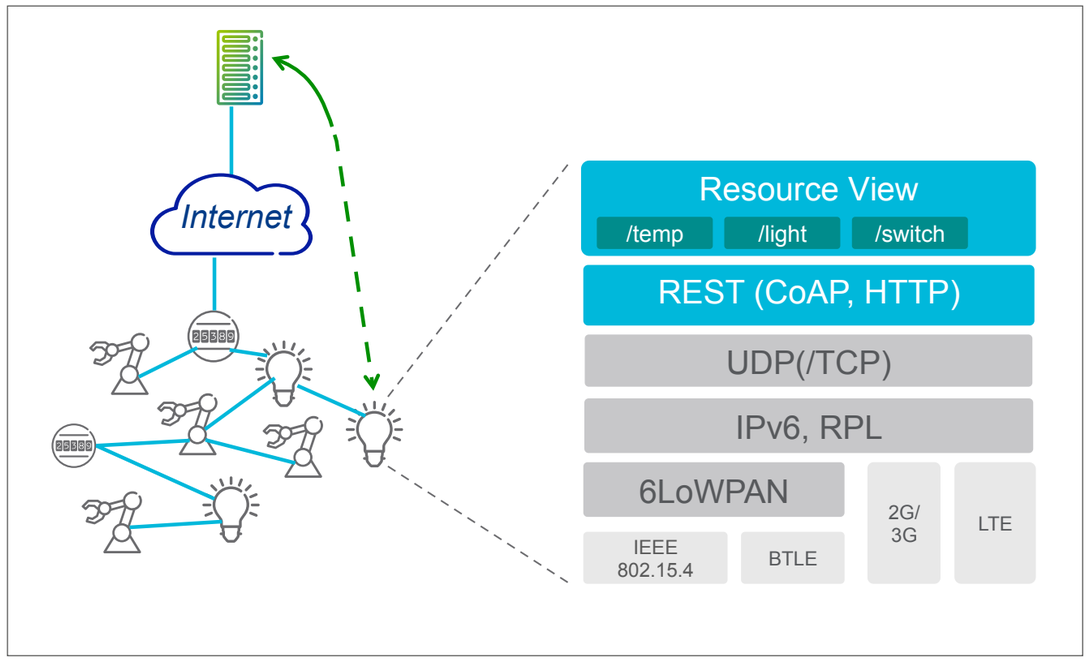


Figure 3. *The Embedded IP Toolbox*

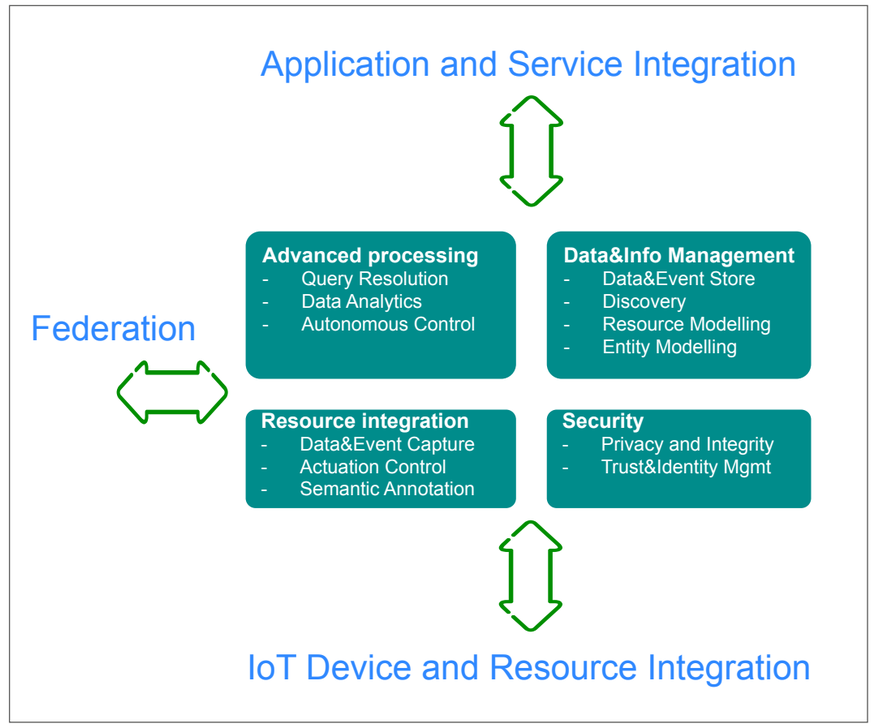


Figure 4. *Capabilities of application integration of IoT resources*

Moving towards an ecosystem for the Internet of Things

Our vision of the Networked Society is not just about technology. It is equally important to create an ecosystem of device vendors, application innovators, network operators, infrastructure vendors, cloud service providers, and others to create a feasible business model that does not require application builders to excel in every area.

Ericsson takes a holistic view on the Internet of Things by driving the vision, the mentioned technology evolution as well as engaging and driving the necessary ecosystem formation. We also provide key enabling solutions to make the Internet of Things happen, like managed connectivity services for IoT devices via our Device Connection Platform and turn-key systems integration activities towards different industry sector applications. The Ericsson approach is to ensure that all the necessary parts exist for the stakeholders and user to benefit from the Internet of Things.

Jari Arkko & Jan Höller
Ericsson Research

References

[1] "Interconnecting Smart Objects with IP," A. Dunkels and JP Vasseur, Morgan Kaufmann/ Elsevier 2010, ISBN 978-0-12-375165-2

[2] "Constrained Application Protocol (CoAP)," Z. Shelby, K. Hartke, C. Bormann, B. Frank. Internet Draft draft-ietf-core-coap (Work In Progress), IETF, March 2012.

[3] "CoRE Link Format," Z. Shelby. Internet Draft draft-ietf-core-link-format (Work In Progress), IETF, January 2012.

[4] "Media Types for Sensor Markup Language (SENML)," C. Jennings, Z. Shelby, and J. Arkko. Internet Draft draft-jennings-senml (Work In Progress), IETF; January 2012.

[5] "Constrained RESTful Environments (CoRE) WG," http://tools.ietf.org/wg/core.

[6] "ZigBee Smart Energy Profile Specification," Version 2.0, ZigBee Alliance, to be published

IoT ECOSYSTEM: CURRENT STRUCTURE AND EVOLUTION PHASE

A business ecosystem represents a network of interacting companies and individuals along with their socio-economic environment. Like the organisms in the biological ecosystems, the firms in the business ecosystem co-evolve their capabilities around specific innovations - a common set of core assets - by both competing and cooperating with each other. In the case of the IoT business ecosystem, these core assets may be in a form of hardware and software products, platforms or standards that focus on the connected devices, on their connectivity, on the application services, or on the services supporting the provisioning, assurance, and billing of the application services, as exemplified in the table below.

Core	Hardware platform	Software platform	Standards
Connected device	Arduino, T-Mote Sky	TinyOS, Contiki OS	HGI
Connectivity	Wi-Fi or ZigBee SoC	Californium, Erbium	IPSO Alliance, ZigBee Alliance
Application services	Cloud infrastructure	Pachube	SOA, JSON, EPC
Supporting services	M2M optimized GGSN	NSN M2M suite, EDCP	ETSI M2M TC

Table 1. *IoT Business ecosystem core assets*

Structurally, many mature business ecosystems can be described using a keystone model which assumes that the ecosystem is dominated by a major hub firm interacting with a large number of small suppliers. The presence of the hubs makes the network robust to the removal of individual nodes, provided that the hubs are intact. By limiting and removing the number of players that would negatively affect the ecosystem, and by providing the remaining players with a foundation (software platforms, development tools, etc.) to survive and succeed, the keystone player increases the stability, diversity, and productivity of the ecosystem as a whole.

The next figure shows the network of 43 major M2M alliances, as reported by MindCommerce for 2011-2012. Focusing on M2M alliances, the figure excludes many IoT platform vendors, such as ThingWorks, Sensinode, Cosm, etc.; nevertheless, it can be seen as a first approximation of the contemporary IoT ecosystem. As can be seen in the figure, no clear hubs – keystone firms – can be identified in the network yet. Still, a number of firms have notably more interactions, making them hub candidates. These are mainly communications service providers (CSP), e.g., Telefonica, Vodafone, NTT Docomo. With the exception of Jasper Wireless, the specialized platform vendors, such as Axeda, SensorLogic, Transatel, NextM2M and Digi, have a relatively few connections at the moment. Furthermore, some of the CSP, such as Verizon, also build M2M connectivity platforms. We consider this to be a reflection of the IoT domain being in an early phase of its evolution, which shall be discussed next.

When seen from the industry evolution perspective, the structure of an ecosystem often evolves over time, from a vertically integrated to a vertically disintegrated or specialized structure. In the course of such vertical disintegration/specialization, different stages of the development, production, and marketing become the responsibility of different firms, rather than being vertically integrated within the boundaries of a single firm.

In case of software ecosystems, this vertical disintegration process iterates through five phases. In the first, the Innovation phase, the software development takes place in-house within the firms seeking competitive advantage by automating core business processes. In the second phase of Productization and Standardization, firms improve their in-house software by adopting the best practices of their competitors towards industry-wide standardized offering. Also, the first software products emerge in the market. The third, the Adoption and Transition phase, is characterized by the growing user base and market share of the emerging standard offerings; outsourcing the software development is increasingly common in this phase. In the fourth, the Service and Variation phase, one of the competing offerings becomes the dominant design attracting the majority of the subsequent software development activities. Finally, in the Renewal phase, new software-related business opportunities are sought as bringing competitive advantage, which then initiates a new evolution cycle.

Two characteristics of the contemporary IoT field are essential in describing the current state of its evolution.

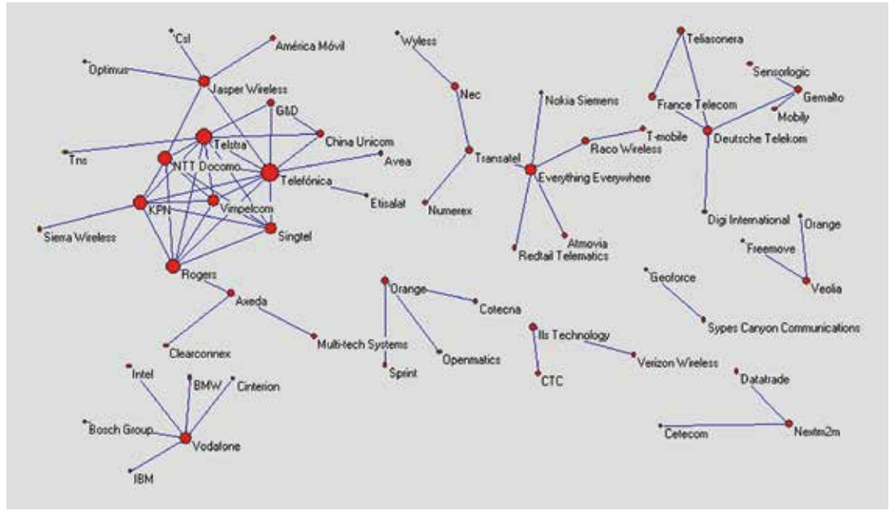


Figure 1. *The network of 43 major M2M alliances*

First, products targeting specific vertical application domains (automotive, machinery) or the horizontal consumer market (home automation, consumer electronics) have started to appear, with wellbeing devices (e.g., Withings) and smart home solutions (e.g., GreenWave Reality) being among the most prominent examples. This can be contrasted with the situation a few years ago when IoT technologies were mainly implemented as a part of industrial in-house solutions based on machine-to-machine communications and/or embedded systems.

Second, the solutions available today rely on various co-existing platforms, protocols, and interfaces, either proprietary or standard. This indicates the lack of a de-facto standard, which makes inter-vendor interoperability challenging, and slows down the entry of new firms and new products in the IoT market. For instance, Z-Wave – a short-range wireless technology for home automation – represents a vertically integrated protocol stack that only works on top of Z-Wave proprietary radio; it does not specify the interoperability with the Internet protocols, and thus a dedicated gateway is needed to convert Z-Wave application protocols into a convenient presentation format. Likewise, the KNX protocols for building automation specify the layers from the link up to application layer, with a dedicated gateway device needed to perform the conversion to TCP/IP.

A notably different approach is taken by the ZigBee protocol stack running on top of IEEE 802.15.4 radio. The stack complements the network (originally non-IP) and application level protocols by defining the so-called public application profiles enabling cross-vendor interoperability within specific application domains, such as home automation, smart energy, healthcare etc. The universality and flexibility of ZigBee comes at the cost of greater complexity, thus making it less attractive for constrained smart objects.

Given the appearance of products but lack of a dominant design and abundance of proprietary protocols and platforms, the IoT ecosystem could be seen as belonging to no later than the Productization and Standardization phase. Although the upcoming IETF protocols, such as CoAP, RPL and 6LowPAN, represent a promising alternative to proprietary or prohibitively complex web protocols, they are just leaving the research labs and making their way into the industrial products and solutions, while the protocol standardization has just been completed or is still being finalized. Therefore, the competition is still upcoming between the traditional and proprietary solutions, on the one hand, and the new IETF-based solutions, on the other hand, for the position of the new dominant design in future IoT applications.

Certain factors may inhibit the evolution process. Among these are a small market size, a high degree of market regulation, a high degree of required customer-specific tailoring, the need to coordinate innovation efforts spanning several vertical layers, the internal complexity of the business processes being automated by the software and the need to maintain compatibility with older systems. Furthermore, according to the technology acceptance models, the widespread adoption depends on the expected performance and the perceived ease of use. For example, if the new protocols provide only minor benefits as compared with the proprietary or HTTP-based solutions, if they require significant investments that are unlikely to pay off, or if they are complex to implement, their adoption and consequently, the emergence of a new dominant design is likely to be hindered, similarly to the failure of the WAP protocol in the past.

Oleksiy Mazhelis
University of Jyväskylä

Sergey Balandin, Ekaterina Dashkova, Yevgeni Koucheryavy

SERVICES AND APPLICATIONS DEVELOPMENT SUPPORT: IoT APPLICABILITY FOR mHEALTH AND e-TOURISM

This project studies IoT applicability for real business solutions in mHealth and e-Tourism use scenarios. The general applicability of Internet of Things (IoT) depends on availability of an efficient and scalable programming platform for applications and service development. The mHealth and e-Tourism application domains require supporting a large set of intercommunicated elements. The data flows come from a variety of smart devices and sensors attached to the user and embedded in surrounding things. In the preliminary studies the architecture for mHealth and e-Tourism IoT solutions was created, where mobile devices are used as hubs for initial processing and storing of the collected information. The key target of this study is to develop methods for efficient data collection, refinement, interpretation and service adaptation to personal needs.

In this study the IoT applications have been abstracted by a dataflow network model. This model is well suited for obtaining meaningful information by efficient sensor data manipulation and refinement. For simplicity we consider only unidirectional dataflows, i.e., where raw sensor data is refined into intermediate data, which is passed over to the following levels and so on. As a result of orchestration of the sensor data, the model is well suited for sensor data processing and creation of various adaptable services [1].

Also the solution shall take into account that many units of ubiquitous environment in IoT have limited power supply and use unreliable wireless channels. Thus a processing unit may become unavailable for a period of time and temper the functionality of dependent services. The architecture of IoT solutions and middleware should be created in a way to address the problem of reliability and availability of end-user services [2].

The ultimate goal of this project is to make preliminary R&D for real-life demos of IoT use in mHealth and e-Tourism. We selected the latest version of Smart-M3 platform (with Redsib) [3] as a basis for further development. Smart-M3 is the open source middleware for creation of smart space applications [4]. A key part of this project was targeted in designing and implementing an agent substitution mechanism for the Smart-M3 platform. This task is based on ideas and pre-studies published in [5]. In this study we refined description of the manned agents' behavior and clearly specified cases and procedure for agent substitution.

The substitution mechanism is implemented as a module of the platform's core element that provides services to substitute the lost or compromised agent by another. The substitute agent gets the same data processing program and operational context allowing all other dependent agents and services to run without downtime. We summarized a detailed description of the substitution mechanism and main scientific results in paper [6] accepted for publication at ICC'13 WS–SCPA. It is important to mention that the designed mechanism can be ported to other IoT platforms or even directly incorporated into the selected services.

We prepared a demo case that illustrates the agent substitution mechanism. The first software demo system controls the amount of light in a room. The system consists of:

- **Sensors that measure the amount of light inside and outside the room;**
- **Actuators that allow controlling window blinds and lamp-light intensity;**
- **A remote control unit that allows the user to set the desired amount of light;**
- **An agent that controls actuators using information from sensors and remote control.**

The system controlling window blinds and the intensity of lamp light to keep amount of light in the room at the desired level. Even when the control agent loses connection with the network it is immediately substituted by another control agent. As a result we achieve that the service is provided without interruption of the system operation and users are not disturbed and even do not notice when the main agent goes down. The demo was presented on April 25 2013 at the 13th FRUCT conference (www.fruct.org/conference13).

Another direction of the project work is development of services based on automatic detection of user's presence in the target IoT space. This is needed for broad deployment and acceptance by users of IoT solutions, as all proactive services require information about user's presence in the room, i.e., to recognize events when the user appears and leaves the room. For example, this can be used for guiding and handling patients in a hospital or tourist center, when just by entering to the corresponding building they start receiving personalized recommendation and guidance.

> *The project studies IoT applicability for mHealth and e-Tourism domains that require support of a large set of intercommunicated elements and data flows coming from a variety of smart devices and sensors on the user and embedded in surrounding things.*

The presence detection is implemented using Innorange Footfall Technology (http://www.innorange.fi/). The technology is based on the dedicated sensor (TP-Link WDR3600 with the USB Bluetooth dongle), which tracks MAC addresses of participants' mobile devices. Every device produces mobile network traffic (within Wi-Fi or Bluetooth connection). Each traffic unit has received signal strength indication (RSSI) value. The closer the device is located to the sensor the higher the RSSI value is. The traffic is continuously monitored. If the RSSI value is greater than the threshold then the participant is treated as present in the room. The last presence time is periodically recorded in the user profile (a part of the Smart Room space) and forms the user's presence history which can be further analyzed for services personalization purposes.

With the help of Innorange and PetrSU we created the first demo of such a service that shows use of the technology integrated into the Smart-M3 based development of Smart Room. The demo service allows users to participate in the event held in the room (showing presentations and checking room sensors measurements) and offers personalized options (recommending based on user's interests). Also this demo was presented on April 25 2013 at the 13th FRUCT conference (www.fruct.org/conference13). In addition we are currently preparing a paper that summarizes our experience on the topic and describes the service architecture with automatic detection of user presence in the target IoT space.

The next step is to combine the obtained results and create a real reliable service demo for m-Health and e-Tourism use cases.

References

[1] L. Xiao and Z. Wang, Internet of things: a new application for intelligent traffic monitoring system, Journal of Networks, vol. 6, no. 6, pp. 887–894, 2011. [Online]. Available: https://www.academypublisher.com/~academz3/ojs/index.php/jnw/article/view/0606887894 [07 April 2013]

[2] Y.-K. Chen, Challenges and opportunities of internet of things, in Design Automation Conference (ASP-DAC), 2012 17th Asia and South Pacific, February 2012, pp. 383–388.

[3] F. Morandi, L. Roffia, A. DElia, F. Vergari, and T. S. Cinotti, Redsib: a Smart-M3 semantic information broker implementation, in Proceedings of the 12th Conference of Open Innovations Association FRUCT and Seminar on e-Travel. Oulu, Finland. St.-Petersburg: SUAI, November 2012, pp. 86–98.

[4] J. Honkola, H. Laine, R. Brown, and O. Tyrkko, Smart-M3 information sharing platform, in 2010 IEEE Symposium on Computers and Communications (ISCC). IEEE, 2010, pp. 1041–1046.

[5] A. Vasilev, I. Paramonov, and S. Balandin, Mechanism for robust dataflow operation on smart spaces, in Proceedings of the 12th Conference of Open Innovations Association FRUCT and Seminar on e-Travel. Oulu, Finland, November 5-9, 2012. St.-Petersburg: SUAI, 2012, pp. 154–164.

[6] Vasilev, I. Paramonov, S. Balandin, E. Dashkova, Y. Koucheryavy, Mechanism for Context-Aware Substitution of Smart-M3 Agents Based on Dataflow Network Model, IEEE International Conference on Communications (ICC WS - SCPA), Budapest, Hungary, June 2013. (Accepted for publication)

Sergey Balandin
FRUCT Oy, Tampere University of Technology

Ekaterina Dashkova
FRUCT Oy, Tampere University of Technology

Yevgeni Koucheryavy
Tampere University of Technology

EXECUTIVE SUMMARY OF THE STATE OF THE ART REPORT, EXTRACTS

The IoT technical and business perspectives are expected to merge at several levels. At the most profound level, the trends affecting IoT businesses include, from the business perspective, the digitalization of services and from the technical perspective, the cloudification of services. Broadly speaking, the analysis and description of the business and physical domains, as well as the discussion of ecosystems and solutions, provides a background for discussing business.

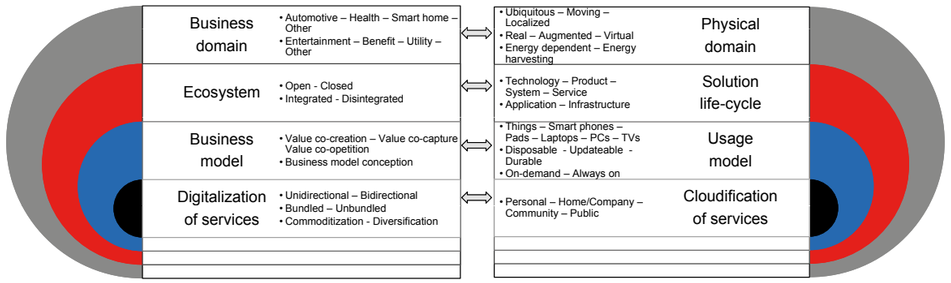
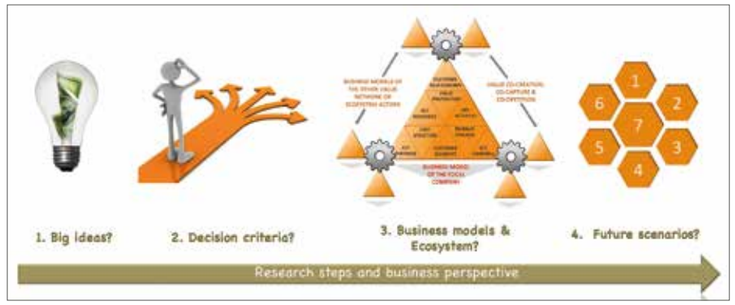


Figure 1. *Business perspective and technical perspective*



New business models are anticipated to be the main driving force of **an IoT ecosystem creation**. The objective of the business model analysis is to gain insight into the processes that have a role in the transformation of existing IoT businesses and ecosystems as well as in the emergence and creation of new IoT businesses models and ecosystems. The key to these processes is in understanding the processes of **value creation and capture**.

In order to analyze the business potential of the IoT phenomenon within this project, business opportunities were approached via so-called Delphi study (see next article) and interactive workshops in selected areas.

For instance, **the workshops** addressed three business domains: connected home, healthcare/wellbeing and automotive. **28 future business scenarios** with the sets of **drivers, limitations, challenges and opportunities** were created during 2012. The scenarios indicated that the existing structures and mechanisms for providing IoT services for home, health, and automotive environments remain as they are in near future.

However, new services, business trends and opportunities are emerging and the participants of the IoT program should be prepared to play more active role as the **IoT "market makers"** and exercise **full cooperation in generating new business models and ecosystems**.

Alex Shveykovskiy, University of Oulu

BUILDING NETWORKED IoT BUSINESS MODEL SCENARIOS WITH A DELPHI STUDY

The recent discourse on IoT has emphasized technology and different technology layers. Currently, there is a pressing need for research of emerging IoT ecosystems from a business perspective. Theoretical understanding and empirical research are needed on what IoT business models are and how they are connected to the underlying ecosystem. We focus on this critical research gap by studying business models in the IoT ecosystem context. We have constructed a framework for analyzing different types of IoT business models. The research draws on service and business model literatures and an empirical research based on a Delphi study in the IoT community. The Delphi method is a systematic, interactive method which relies on a panel of experts. The experts answer questionnaires in two or more rounds. Delphi is based on an idea that forecasts or decisions from a structured group of individuals are more accurate than those from unstructured groups.

Based on the literature review and the Delphi study we will look at our study results through theoretical frameworks classifying IoT business models developed by us [1, 2], and the managerial cognition perspective towards business models developed by Tikkanen et al. [3]. The IoT business model frameworks help to visualize IoT business models and their evolution in relation to the type of customers or products and services and openness of the ecosystem. According to the managerial cognition perspective a business model can be conceptualized as a combination of firm-related material structures and processes and intangible cognitive meaning structures in the minds of people. The intangible structures of business models consist of belief systems – reputational rankings, industry recipes, boundary beliefs and product ontologies. Industry recipes express the persuasions of the management related to economic, competitive, and institutional logic of the firm. Boundary beliefs define the identity of the company with a certain inter-organizational community. Product ontologies link product or service attributes, usage conditions, and buyer characteristics into a hypothetically superior offering on the target market. Reputational ranking denotes the own performance of the firm related to its socially evaluated competition [3].

In the 1st and 2nd rounds of the Delphi study we collected case examples of the current and possible future IoT business models, as well as views of challenges and success factors of these case examples. A summary of the 1st and 2nd Delphi rounds as cases is presented in the Table on next page.

Industry / Application area	Case description	Products / services / benefits offered	Technology needed
Manufacturing	IoT-adapted manufacturing processes	Customization of products during the production process.	Situation-aware smart machines and robots
Health	Health related products and services	Medical expertise	Sensors, IoT communication infrastructure
	Health guidance service	Monitoring of key parameters; analyzes by medical experts.	Sensors
Home	Home owner's digital service	Monitor and manage facilities.	Plug-and-play devices, installation package, open and user-friendly applications
	Saving energy	Measuring temperature, and thus decreasing energy consumption	Sensors
Traffic	Traffic data marketplace	Real-time traffic, environment, weather, road condition, incident, etc. related data	Databank, sensors
Shopping	Electronic shopping assistant	Key information about a product which a customer points to in a shop, for example price per unit, production/expiration date, ingredients, calories, country of origin, etc.	Electronic shopping assistant device, RFID
Food	Food security tracking system	Tracing of food products from original material providers to consumers	Sensors, RFID
Real estate	Real-time waste monitoring	Reducing the costs of waste collection.	Sensors

Table 1. Cases drawn from the 1st and 2nd Delphi rounds.

References

[1] Leminen, S., Westerlund, M., Rajahonka, M. & Siuruainen, R. Towards IOT ecosystems and business models. S. Andreev et al. (Eds.): NEW2AN/ruSMART 2012, LNCS 7469, pp. 15--26. Springer-Verlag, Heidelberg (2012) The 5th conference on Internet of Things and Smart Spaces ruSMART 2012. August 27-28, 2012. St.-Petersburg, Russia http://rusmart.e-werest.org/2012.html (Conference proceedings) ISBN 978-3-642-32685 http://www.springerlink.com/content/23005812265560x7/

[2] Leminen, S., Westerlund, M., Rajahonka, M & Siuruainen, R. Internet of Things – Opportunities for Innovative Service Business Models. Abstract and presentation on 19th – 20th September 2012, Cambridge, UK, The Future of Services in a Connected World, Service Operations Management Forum, Fifth International Workshop

[3] Tikkanen, H., Lamberg, J-A., Parvinen, P., Kallunki, J. (2005), Managerial cognition, action, and the business model of the firm. Management Decision. 43, 76, pp. 789-809.

Seppo Leminen, Mervi Rajahonka and Riikka Siuruainen
Laurea University of Applied Sciences, Espoo, Finland

Mika Westerlund
Carleton University, Sprott School of Business, Canada

IoT FOR INTELLIGENT TRAFFIC SYSTEM

The purpose of the ITS pilot is to expand the work carried out in 2012 and further develop the "pre-pilot", which was the main result in the first year of the IoT Program. The main idea is to utilize latest available common, generic standard architectures, interfaces and data formats to ensure that pilot results can also be used in the future elsewhere. The 2012 pre-pilot was able to offer internet-based traffic information services for fixed users. The information consisted of weather, friction, temperature, speed, vehicle behavior and environmental data, that was gathered from in-vehicle road weather units, sensors and the infrastructure. The data sources were buses operating in the pilot corridor, all the taxis in Tampere, one road side unit and a couple of cars.

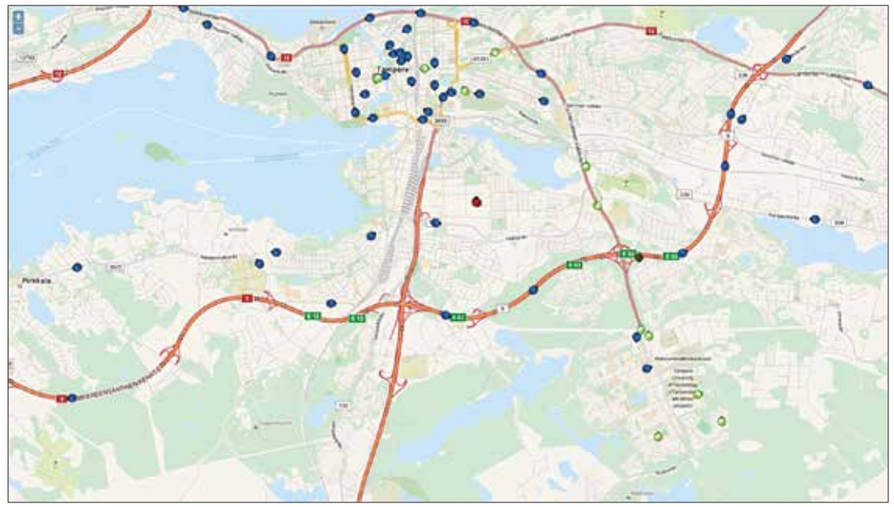


Figure 1. *The Pre-pilot User Interface*

The ITS pilot corridor was originally planned to be the "Hervannan valtaväylä"-road between the Tampere city center and the Hervanta suburb. However, the pre-pilot had even wider coverage, since the data collected from taxis covered the whole city of Tampere. The fleet used collected data and forwarded it via different communication technologies. In 2013 the ITS pilot will be developed further and an HTML5-based version will be prepared. Figure 2 describes the scope of the ITS pilot. Especially in 2013 methods to utilize the collected and processed information will be studied. The focus will be on recognizing the real end user needs and patterns to actually change driver or passenger behavior with the intellectual combination of history, real time and forecast data.

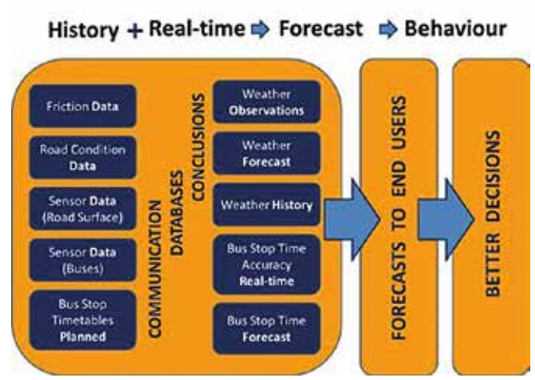


Figure 2. *The scope and Aim of the ITS pilot*

Pekka Eloranta, Mobisoft

Jarmo Salmela

MORE FUN WITH INTERNET OF THINGS STUFF

Fun sells. Internet of Things applications have usually a sales argument based on cost savings and concept of security. How to combine the both selling points and create user experience that motivates to use the service in long run. Gamification concepts may help to motivate users to reach their serious goals why they subscribed the service in the first place.

For business management the main message is that gamification is trendy, it is on hype curve and major technology trend watchers notice it as a major trend [4, 2, 1]. So, it is not too far fetched to look at what it is and what it is not.

Gamification is the concept of applying game-design thinking to non-game applications to make them more fun and engaging [3]. Gamification is the use of game thinking and game mechanics in a non-game context in order to engage users and solve problems. Gamification is used in applications and processes to improve user engagement, Return On Investment, data quality, timeliness, and learning [11]. What it is not is creating another new gameplay such as "the angry managers".

Demographics are an important selling point. People under 30 have lived with their Playstations, those under 50 may still remember their first encounter with Commodore64 games. Games are consumed in daily life. By one estimate creating Wikipedia took eight years and 100 million hours of work, but that's only half the number of hours spent in a single week by people playing World of Warcraft [7]. Game thinking is not just for young or marginal users. [1, 9].

An easy first pragmatic level is to communicate to users a sense of context, meaning and overall progression. There are simple tools such as levels, points, missions, badges, rankings and trophies on achieving milestones in the usage of service or advances overall [9]. The user environment can be easy for novices and provide more direct shortcuts to advanced users.

A word of warning comes from Gartner. Poor design makes applications fail. The challenges are in the creation of player-centric applications. Game design talent is needed in the actual design elements such as in balancing competition and collaboration or defining a meaningful game economy. Badges and leader boards are tools to implement the underlying engagement model [5].

What is good, Internet of Things creates a lot of data. It opens possibilities to measure achievement and behavior and to provide motivated feedback. The first application area is in learning to use the possibilities of the new service environment. The second application area is to motivate users in intended usage, such as saving water or energy.

Examples on motivation, learning and engagement

Zynga Inc is a provider of social games in Facebook, their best known game is FarmVille launched in Facebook 2009 and its sequel, FarmVille 2, in 2012. In FarmVille you earn in-game coins and experience points that can be used to raise the player's level.

Foursquare was released in 2009 and it is a location-based social web-site. Each check-in on a location awards the user points and sometimes badges.

H2 Wellbeing Oy has released HeiaHeia, which is a social web service that motives to exercise more and allows shared activities on Facebook and/or Twitter.

Nike released its Nike+ sensor and iPod kit originally in 2006. Today's version can be used to track running, but the user can choose a goal for workout as well. Audio feedback is provided on milestones and congratulations are provided whenever a user achieves a personal best.

> *"Where is the fun?" is a relevant question also in the context of Internet of Things.*

Green Goose uses wireless sensors that can be attached to objects, such as a toothbrush, water bottle or bike, to detect when you perform a task you have set yourself and rewards you with lifestyle points. The company sees Interactive Toys and "appsessories" as a hot trend. Their sensors were originally pitched as a money-saving tool in 2010.

Privacy and data security

The first big NO for gamification is privacy. Users are doubtful if the approach reveals too much about their life or, in the extreme, gives a hint to unwanted visitors, when their flat is empty and they can be robbed. Privacy-related issues also include the ownership of data and legal issues on the handling of the database.

Privacy problems can be solved in the design phase. Participation is voluntary and users know what they should share. Presence information is not needed in game-like context and a player may see other players only as a group or as a team average, instead of individuals. Or the player may see a more abstract goal. The solution to privacy issues is based on the overall design of the motivation.

Security is a similar issue and can be solved in the design phase, if the requirements include it both on the system and gamification design level. Although, there are rare cases when the designer has the idea to make art installations using the windows of a high rise as a screen – or even play Tetris switching lights on and off [6]. But most cases in the gamification design level involve simply a one-directional link and some real-life data that is read-only being used in the feedback system.

On the system service level, security and privacy require a maintenance policy and process. Servers are updated, security patches run and data is protected in disturbances. The users should be able to trust that the process works at least as well as the overall facilities maintenance and does not cause worries.

From the users's perspective, the service provider must have a clear message on how issues of privacy and security are solved and handled. The users will ask for it.

If the privacy and security seem to be an obstacle, remember that Foursquare and other location-based social services have solved or lived with the issue.

Two mind sets, interface needed

The life-cycle of Internet of Things devices is long. In the case of consumer and housing, it is expected to lasts longer than fridge and freezer, which means a life-cycle over five years, preferably longer. In the case of games the life-cycle is usually short. A viral game may be popular for some months, but usually games include several releases of versions or themes during their life-cycle. Users expect new features on a regular basis. The need for new features and versions may easily face extreme opposition in design and co-operation with Internet of Things hardware designers or game designers may not be rosy.

From the motivational UI design – or game design for short – point of view the first problem is how to get real-life data, any of it. The Internet of Things point of view is at the same time what the minimal viable product is and how to make it as low cost as possible. All interfaces are expensive and they may create some unwanted complexity. Use of the legacy metering devices for water or electricity requires case-by-case solutions which raise the costs. What can be gotten from a metering device is a number or interval, i.e., very low level data in any case.

The key is to provide some other interfaces somewhere. The cloud applications may be the solution for the advanced interfaces. Sometimes the home gateway device is useful.

The interface, or technical API, is also a solution for the different requirements on design.

Conclusions – where is the fun?

Use of game mechanics to motivate users or to support a sustainable wanted lifestyle is trendy. In real life not so many success stories are yet reported, but the huge success of sports / lifestyle services with social game-like features is at least promising.

"Where is the fun?" is a relevant question also in the context of Internet of Things. Without motivated users the benefits of new services may not be reached. Users must be engaged in the use. Positive motivation works better than discussion on possible savings and security.

Game companies and home automation developers do not meet and mix easily. Skillful development of ecosystems as well as project management are needed to get out the best of both competencies. Service development is teamwork.

Gamification is on a hype curve. It works in getting users involved and motivated. Creating a variety of short-term and long-term goals is important as well as rewarding efforts continuously with some occasional unexpected rewards. Similar kind of thinking should be utilized in Internet of Things projects.

NOTE ON REFERENCES / FURTHER ON GAMIFICATION

- **The Gamification Wiki www.gamification.org/ is the best source on the net. With Wikipedia Gamification article there are plenty of up-to-date references.**
- **Zichwerman and Cunningham: Gamification by design is the best starting point in the area in book format and almost all presentations use it as a source. [8]. Big picture in business can found in [1].**
- **www.slideshare.net has gamification as a search tag. Select what services fit your needs.**
- **Presentation by Margare Wallace is introduction to subject [9].**
- **The important thing to do is to have some hands-on experience on the inspiring games with useful game mechanics. Try FarmVille, Foursquare, HeijaHeija, Nike+ or some other motivational and casual games.**

Sources:

[1] Chatfield, Tom: Fun Inc, Why games are the 21st century's most serious business (Virgin Books 2010)

[2] Deloitte Tech Trends 2013: Gamification Goes to Work, Moving beyond points, badges, and leaderboards. (2013) http://www.deloitte.com/view/en_US/us/Services/consulting/technology-consulting/e53b217fb162c310VgnVCM1000003256f70aRCRD.htm

[3] Gamification Wiki: http://www.gamification.org/wiki/Gamification

[4] Gartner Inc: Top Predictions for IT Organizations and Users, 2013 and Beyond: Balancing Economics, Risk, Opportunity and Innovation, Oct 2012

[5] Gartner Inc: Gartner Says by 2014, 80 Percent of Current Gamified Applications Will Fail to Meet Business Objectives Primarily Due to Poor Design, Nov 2012 http://www.gartner.com/newsroom/id/2251015

[6] Mikontalo tetris, 2007 http://www.youtube.com/watch?v=dS7TZ1RmJzE (video)

[7] Tierney, John: On a Hunt for What Makes Gamers Keep Gaming, NY Times, 6.12.2010. http://www.nytimes.com/2010/12/07/science/07tierney.html?_r=0

[8] Zichwerman and Cunningham: Gamification by design (O'Reilly 2011)

[9] Wallace, Margaret, Rules of Engagement: 10 Ways in Which Game Mechanics Are Changing the World (Mindtrek conference, Tampere 2011), presentation http://www.margaretwallace.comhttp://www.margaretwallace.com , http://www.slideshare.net/MargaretWallace/rules-of-engagement-how-gamification-is-changing-the-world

[11] Wikipedia, Gamification: http://en.wikipedia.org/wiki/Gamification

Game examples:

FarmVille: http://en.wikipedia.org/wiki/Farmville , http://company.zynga.com/games/farmville

Green Goose: http://www.greengoose.com/

HeiaHeia: http://www.heiaheia.com/corporate/

Nike+: http://en.wikipedia.org/wiki/Nike_plus , http://nikeplus.nike.com/plus/

Jarmo Salmela
TeliaSonera, Broadband Technology Services unit

APPLICATIONS OF COLLABORATIVE ANALYSIS

Sensor data gathering and analysis are important ingredients for IoT services and applications. Sensor data gathering, or sensing the environment, should be efficient, in particular, in terms of data communication and energy consumption. Data processing, on the other hand, should be able to scale with the increasing amounts of data.

Our work has focused on analyzing the health of sensing platforms, which is crucial for realizing robust and efficient sensor data gathering and processing.

F-Secure and the University of Helsinki have investigated application of collaborative analysis techniques to device health and security monitoring on smartphones. Such methods of analysis can provide evidence that the platform and its processes are performing correctly or can help detect problems and threats.

The key idea is to transmit appropriate application and context data from devices to a cloud platform for statistical analysis and data mining. The cloud backend can leverage existing knowledge bases and information gathered from a large number of devices in order to identify suspicious or harmful applications and activities.

The work has initially focused on smartphones as hubs and portals for the Internet of Things, but the models can be applied to other sensing systems such as smart watches and augmented reality devices. The techniques are especially useful for sensing platforms that are capable of running multiple applications.

The approach is inspired by the Carat project (http://carat.cs.berkeley.edu), which debugs energy problems within a smartphone community. The Carat work showed that it is feasible to detect anomalies in the energy consumption of individual applications in individual devices through statistical analysis of data from multiple devices.

Since the data about the energy consumption of individual applications are not available on popular smartphone platforms, one has to resort to treating the corresponding software processes as black boxes and modeling their energy and other parameters. The models are "collaborative": they are based on correlating energy level, active applications, and context data collected in many similar devices over certain periods of time.

The same technique can be used to model software processes in various devices, cars, or other multi-process systems. While the approach can also be used to model sensors with a single process, its power is particularly visible when one has to deal with multiple blackbox processes or subsystems.

While detecting anomalies in energy consumption of applications is certainly important in health monitoring of sensing platforms and, in particular, can help identify malicious or infected applications on mobile devices, there are many types of security issues and threats that are not connected with energy-related information. A natural extension of the approach is to analyze other types of features and run-time data gathered in devices. For instance, one can look for uncommon applications, abnormally large numbers of installations, anomalous data connection usage, unusual capabilities for an application of a given type, etc.

More generally, we are exploring two cases of collaborative analysis application:

- In a multi-process system, when we know what run-time or other information indicates suspicious or malicious activities but do not know what applications are responsible for generating such pieces of information, analysis of data collected from multiple devices can be used to find that out. A similar case is when relevant data values can not be obtained directly but can be efficiently computed or estimated by correlating indirect observations. This is especially useful on closed platforms, where information of many types can not be accessed without jailbreaking/unlocking of the devices.

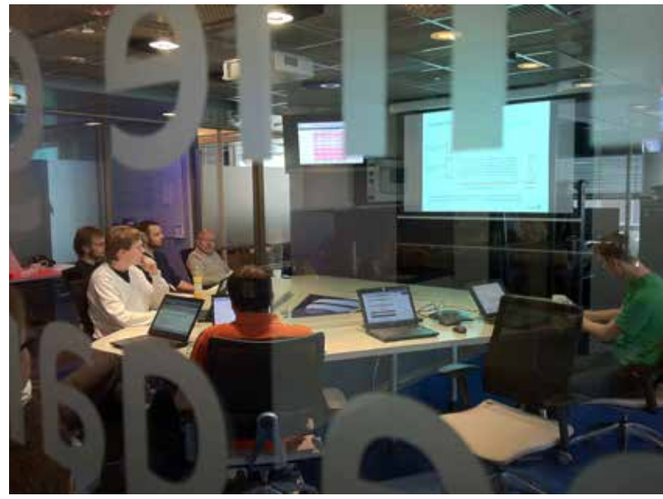
- When we do not have specific pointers to attacks and threats, we can try to detect those as anomalies. Since it is hard to know what normal is if your view is restricted to a single device, collaborative analysis of data collected in a large number of devices comes to help.

Our first prototype, implemented by F-Secure Security Labs in collaboration with the NODES group of the Department of Computer Science, University of Helsinki, demonstrates the first case above for the Android platform. The high-level system architecture is simple, with the major components being:

- A light client that gathers and pre-processes appropriate data in Android devices. The gathering and pre-processing logic has been designed to minimize the amount of data to be transmitted from the device while properly taking into account essential information.

- A backend system for storing, correlating, and analyzing the client data. As we aim at having large numbers of clients and volumes of data, we need to prepare for running heavy computations in the backend. The analysis code is written in Scala and runs in a Spark cluster.

- A protected channel implementation for sending the client data to the backend to ensure backend authentication and confidentiality of the transmitted data. In particular, this is important to address privacy concerns for the client.



We are currently testing individual components and features of the prototype, and we expect to see it ready for full-scale testing in the near future.

To conclude, we will mention a number of challenges and directions for future work.

Among the challenges, assessing the reliability of the analysis results and the amount of data required for high-confidence decisions poses interesting mathematical problems. On the data collecting side, ways of accessing and the quality of the data to be gathered may vary significantly between releases and versions of the platform. When the gathered data are sparse or inaccurate, more advanced analysis techniques will be required.

Thinking of possible future work, we can consider extending the technology to cover other popular mobile platforms and to analyze various statistics of applications for advising the user on their quality and reliability. More generally, as collaborative analysis can be used for finding interesting correlations between events and activities, one could try to apply it to other types of sensory data, such as device and user movement, spatiotemporal density, radiation and pollution readings, etc., in a search for ways of optimizing devices and also mobile networks and processes in those.

Carat project, http://carat.cs.berkeley.edu

Alexey Kirichenko, F-Secure
Sasu Tarkoma and Eemil Lagerspetz, University of Helsinki

A RISK-DRIVEN SECURITY ANALYSIS AND METRICS DEVELOPMENT FOR WSN-MCN ROUTER

This paper discusses security requirements and metrics development for a Wireless Sensor Networks (WSN) - Mobile Cellular Networks (MCN) based router used in IoT scenarios. A risk-driven security analysis is considered as a part of the requirements analysis and security metrics development processes.

Here we consider an IoT system consisting of a Wireless Sensor Networks (WSN) - Mobile Cellular Networks (MCN) based router for a secure, energy efficient, and scalable wireless content distribution and retrieval type communication. The network is composed of the MCN routers and a large amount of Central Processing Units (CPU), memory, and energy-restricted sensor/actuator (S/A) devices, see Figure 1. Mobile routers communicate with application servers in the Internet and surrounding sensor network(s). Access to the sensor network opens up also an opportunity for fraudulent misuse unless it is properly secured. Therefore, a Risk Analysis (RA) was considered as a part of the requirements analysis and security metrics development processes and it was carried out to identify security threats, along with their severity and impact, and security objectives, and controls.

For the definition of requirements, different application scenarios were formed. The scenarios considered house automation, smart grid, environmental, automotive, smart traffic, and eHealth applications. Each scenario was literally described as narrative descriptions that were chopped to use cases and different actors, preconditions and assumptions were identified. Use cases were divided into sequential steps in detail with the parallel and iterative definition of signal flow diagrams. Each step identified the required generic and specific functionalities that were grouped into main classes. All the scenarios targeted massive-scale sensor networks.

Network architecture

It was noticed from our scenarios that very large-scale sensor networks are characterized by correct content distribution and delivery to external servers instead of end-to-end connections between sensor and server hosts. In many cases information is location oriented but the required location information is usually given by geographical or applications oriented coordinates instead of sensors addresses. Therefore, network architecture should treat location dependent contents as a primitive coupling of location and contents. However, it can, and it should, decouple location and identity, and retrieve contents, e.g., by location dependent name and apply new approaches to routing named contents to improve scalability, security, and performance. In other words, location dependent names enable us to use named data abstraction instead of named host abstraction.

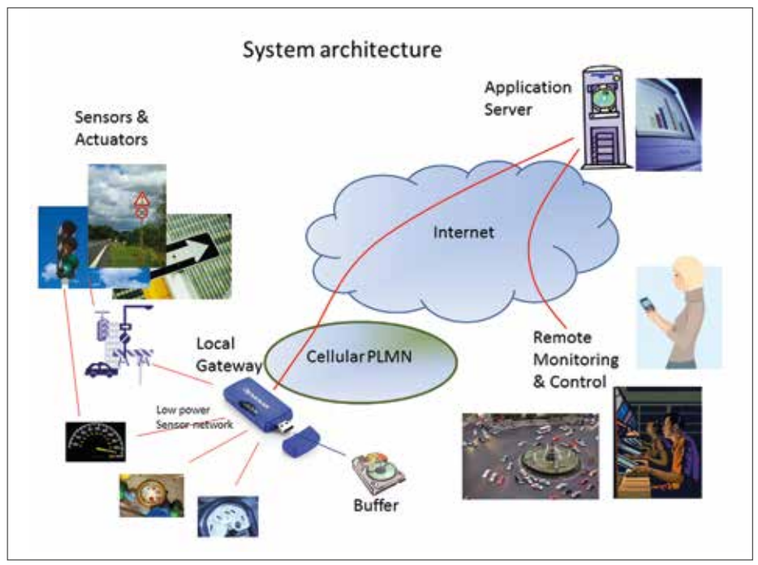


Figure 1. *High level diagram of the system architecture.*

Security risks, objectives, and controls

Table 1 lists some prioritized risks of the target system. Risks with more probability and somewhat low severity were prioritized over risks with somewhat higher severity but low probability. The risk survey and analysis is described in [1] in more detail. Figure 2 presents an example of the deduction of Security Objectives (SOs) and Controls (SCs) from unauthorized access to system and/or data risk. The most important SOs and SCs in a large IoT network are listed in Table 2 and 3, respectively.

Prioritized risk 1: 1.1 1.2 1.3 1.4 1.5	Unauthorized access to system and/or data Destruction of information or resources Corruption/ modification of information or resources Theft, loss or removal of information or resources Disclosure of information Interruption of services
Prioritized risk 2: 2.1 2.2 2.3 2.4	DoS attacks Congestions, crashes, radio jamming Traffic analysis attacks Protocol deceive or violation attacks Sybil attacks
Prioritized risk 3: 3.1 3.2 3.3 3.4 3.5	Exposure to physical attacks Node capturing Node injection Node tampering Location and/or topology changes Generate a physical event monitored by the sensors
Prioritized risk 4:	Malicious resource consumption
Prioritized risk 5:	System delays
Prioritized risk 6:	Bogus denial of a transaction
Prioritized risk 7:	Bogus transaction claims
Prioritized risk 8: 8.1 8.2	High level of distribution Remote management unable to see physical tampering Remote management cause fragile network organization

Table 1. *Prioritized risks*

Security objective 1:	Data protection; integrity, confidentiality, privacy
Security objective 2: 2.1	Protection of network connections Securing routing protocols
Security objective 3:	Authorized and fair access
Security objective 4:	Defend against malicious resource consumption
Security objective 5:	Conceal the physical location of nodes
Security objective 6: 6.1	Key management Key hierarchy for secure multicasting
Security objective 7: 7.1 7.2	Availability Service availablity Infrastructure availability
Security objective 8	Protection against wrong kind of inputs
Security objective 9:	Node capturing prevention
Security objective 10:	Node injection prevention
Security objective 11:	Node tampering prevention
Security objective 12:	Authorization of administrators and users
Security objective 13:	Node movement prevention
Security objective 14:	Protection of trust and reputation

Table 2. *Security objectives*

Security control 1: 1.1: 1.2 1.3	Confidentiality management Encryption: Periodic dissemination of fresh keys Tamper resistant nodes Node concealment
Security control 2: 2.1 2.2 2.3	Integrity management Anomaly detection systems Timeliness detection of data Originality of data
Security control 3: 3.1 3.1.1 3.2 3.3 3.4 3.5	Access control Identification and authentication Pseudonym Authorization Accountability QoS classification Host and network based intrusion detection
Security control 4: 4.1 4.2	Congestion management Congestion prevention Congestion control
Security control 5: 5.1 5.2 5.3	Conceal the physical location of nodes Unvisibility/undetectability Secure location information Automatic and accurate location detection
Security control 6: 6.1 6.2 6.3 6.4 6.5 6.6 6.7 6.8	Secure routing Trust levels based on undirectional evidences Data classification according to privacy levels Routing redundancy Centralized update and delivery of routing tables Location aware routing Estimation of distances between nodes Random walk forwarding Fake packets
Security control 7: 7.1 7.2 7.3 7.4 7.5 7.6	Tamper resistant nodes Software version certification Digital rights management (DRM) Randomized clock signal for critical operations Intrinsic self-test into the detector Destruction of test circuitry Restricted program counter
Security control 8:	Secure computing
Security control 9:	Chargeable connections
Security control 10:	Network control message restrictions
Security control 11: 11.1 11.2	Reputation information Request reputation information Provide reputation information

Table 3. *Security controls*

Security metrics development

The building blocks of the security metrics development and management process are based on the findings in [2]. However, experiences from applying this process in risk-driven development of the mobile edge router product led to modification needs. Hence, it is suggested that this process for IoT devices comprises the derivation of usage scenarios and functional system, networks, and device architecture requirements. The suggested metrics development and management process for IoT devices can be summarized as follows:

- Activity A: $RA_{phase\ 1}$ - $RA_{phase\ 4}$ of RA and SO/SC Analysis,
- Activity B: Iterative decomposition of SOs/SCs,
- Activity C: Iterative planning, design, and use of the measurement architecture,
- Activity D: Iterative feasibility survey,
- Activity E: Integration of QoS and performance metrics.

Anticipation of the effects of scalability requires special attention in the risk survey and analysis of IoT device metrics development and management process, see Figure 3.

Activity A: Risk survey and risk analysis

The RA outcome is used to develop SOs, SCs, and to choose between implementation alternatives. The choice of which issues are selected as main SOs depends on the priority of risks and the criticality of other needs. Here, the risk survey and analysis process consisted of iterative phases. The first phase, $RA_{phase\ 1}$, is conducted during and after the product usage scenarios and use cases definition. The second phase, $RA_{phase\ 2}$, is performed during and after the product functional requirements and device and network interfaces definitions whereas the third phase, $RA_{phase\ 3}$, is performed during the product design and specification. The fourth phase, $RA_{phase\ 4}$, is done when the product is being verified.

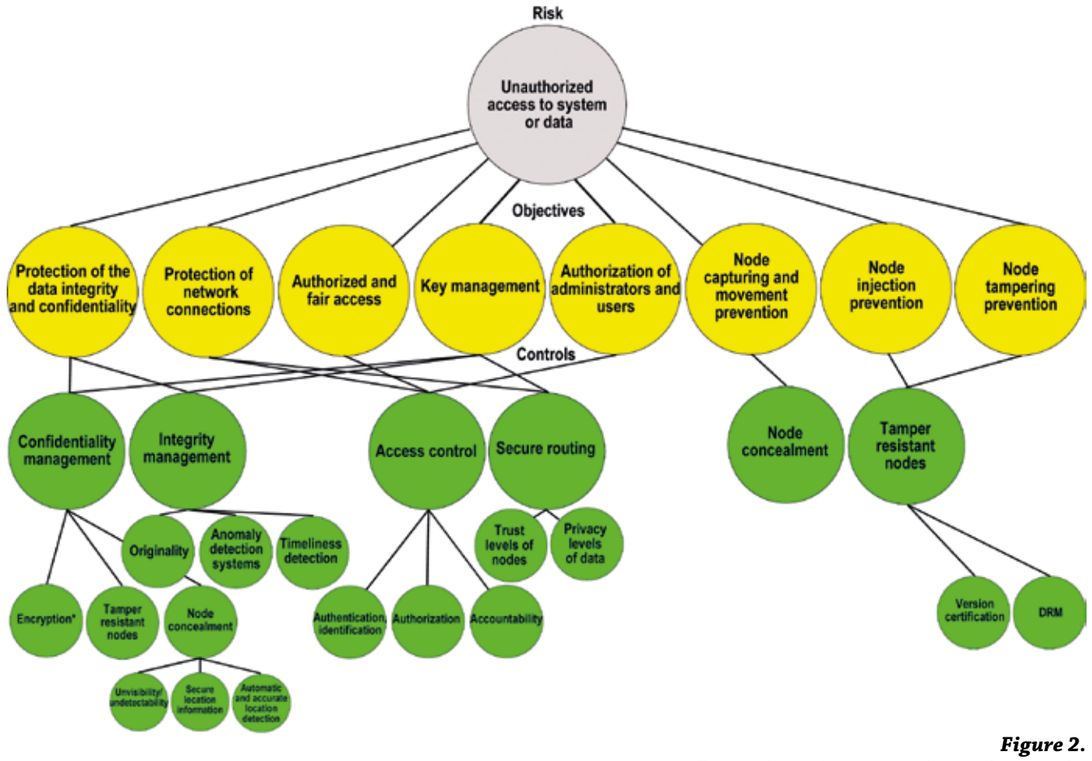
Activity B: SOs decomposition

Activity B is divided into sub-activities:

- Actual SO decomposition,
- Association of the decomposition results with BMs, DMs, infrastructure objects, and timing,
- Consideration of the effects of the scalabity on the frequency of the associated measurements,
- Compensation of evidence gaps and biases.

Base Measures (BMs) are abstract measurable properties of the System under Investigation (SuI) whereas Derived Measures (DMs) are a hierarchy of more detailed measures representing interpretation of the BM.

Even though factors enabling security effectiveness (assurance that stated SOs are met) such as configuration correctness and efficiency (assurance that adequate security effectiveness has been achieved) can be measured, it is not possible to achieve complete evidence of the robustness of the solutions that are taken. In practice, there are various gaps and biases between security effectiveness measurement objectives and the evidence offered by practical security correctness metrics which need to be compensated for



*Periodic dissemination of fresh keys.

Figure 2.
From risks to objectives and control –example.

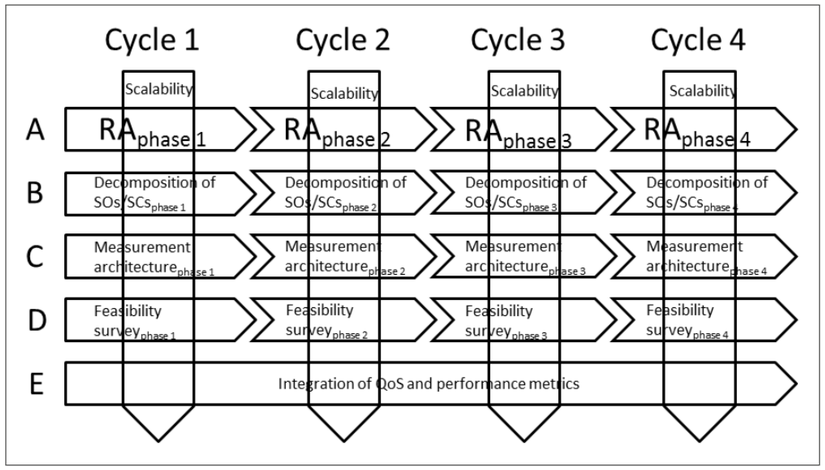


Figure 3.
IoT device metrics development process.

Activity C: Measurement Architecture

Savola and Abie [2] define Measurement Architecture (MA) as the collection of the technical and non-technical means to gather the data needed for security metrics use. MA planning should be started as early as possible during the metrics development. It can support various types of measurement methods either automated or manual.

Activity D: Feasibility survey

Feasibility analysis is needed to answer especially the questions 'Can I trust these security metrics?' and 'Does the use of these security metrics bring benefits?'. In Savola, Frühwirth, Pietikäinen 2012, is introduced a feasibility analysis method for security metrics, which is based on the Feasibility Level (FL) requirements.

Activity E: Integration of QoS and other metrics

Some non-security metrics with security relevance are often available and attainable in the SuI. These metrics can be reused to offer partial security-relevant evidence for the security metrics model. Examples include Quality-of-Service (QoS) metrics [2], other performance indicators, load metrics, and delay, delay variation, and packet loss rate [3].

Conclusions

In this paper we discussed risk analysis and security metrics development processes of the WSN-MCN-based edge router. The analysis led to security objectives and controls that are used to define the security metrics of the WSN-MCN router and to guide the requirements analysis and the network and system architecture design.

Acknowledgements

The research was conducted in the Internet of Things program of Tivit (Finnish Strategic Centre for Science, Technology and Innovation in the field of ICT), funded by Tekes.

References

[1] T.Frantti,H. Hietalahti and R.Savola, "Requirements of Secure WSN-MCN Edge Router", Proceedings of the IEEE International Conference on Information Networking (ICOIN 2013), pp. 210-215, 28-30 January, Bangkok, 2013.

[2] Savola and Abie 2010, "Development of measurable security for a distributed messaging system," Int. Journal on Advances in Security, 2010, 2(4), 358–380.

[3] Savola and Frantti 2009, "Core security parameters for VoIP in Ad Hoc Networks," WPMC '09, 5 p.

[4] R. Savola, C. Frühwirth and A. Pietikäinen, "Risk-driven security metrics in agile software development – an industrial pilot study," Journal of Universal Computer Science, vol. 18, no. 12 (2012), 1679-1702.

Tapio Frantti, Hannu Hietalahti
Renesas Mobile Europe Ltd.

Reijo Savola
VTT Technical Research Centre of Finland

OPERATOR OPPORTUNITIES IN THE INTERNET OF THINGS

Getting closer to Ericsson's vision of more than 50 billion connected devices by 2020 means knowing how to address the diverse connectivity needs for the massive number and variety of devices, while simultaneously facilitating smooth and efficient network provisioning.

Drivers for the Ericsson Device Connection Platform

Estimates from different market analysts vary in terms of predicted figures - but they all agree that data usage will at least double every year until 2015, when data will outweigh voice 30 times over. These predictions are based on the concept that anything that benefits from being connected will be connected. Consumers are increasingly getting used to constantly connected devices, behavior patterns are changing and the value of connectivity for people, business and society is becoming more and more evident.

More than 50 billion connected devices is a vision where the convenience brought to people's lives through the use of mobile networks will be considered normal and expected; a vast number of M2M interactions will constantly take place; and a myriad of new services will raise dependency on mobile networks and secure a massive number of connections. Devices will access mobile networks directly or through gateways. They will communicate with each other, be part of an end-to-end M2M system, as well as communicating with individuals and central control systems. People will make use of numerous everyday devices that benefit from M2M connectivity at home, at work, on the move, remote locations and elsewhere. The most obvious examples include: washing machines, coffee makers, car keys, ticket machines, fridges, window sensors, and utility meters. In addition, mobile devices will be adapted to serve as many other things; such as acting as a connected wallet, connecting to medical services, and working as an interactive location guide. In the world of connected devices, we all benefit from these applications.

In the world of more than 50 billion connected devices there are fewer accidents due to improved safety, our way of life is more sustainable due to more efficient use of resources, we are energy smart, and healthcare and education is available for everyone.

Operators have started to realize that their networks can provide value beyond the existing flat-rate plans. This will come about by applying differentiated connectivity plans tailored to meet the needs of different devices and different types of users.

Today's networks are designed to deliver and enforce different connectivity plans and types. However, to fully cater for the demands created by new types of devices and applications, innovative support systems will be required.

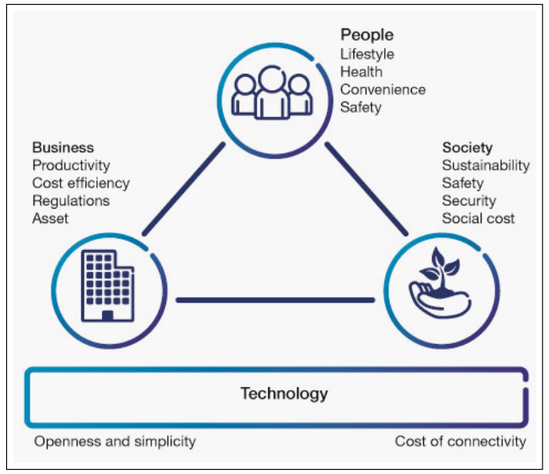


Figure 1.
Device connectivity benefits people, business and society

The fundamental features of such systems include:

- support for IP connectivity over private networks, as well as over the internet;
- efficient provisioning of a large number of subscriptions;
- capabilities to create and enforce tailored subscriptions with respect to QoS and charging models; and
- mobile network operator and enterprise-management portals.

To meet the market need for M2M support systems, Ericsson provides a Software as a Service (SaaS), solution - Ericsson Device Connection Platform, EDCP, offering operators and M2M enterprises an initial low-cost solution for connecting devices and supporting applications, with the potential to expand and adapt to the growing needs of the market.

Ericsson Device Connection Platform Architecture

Functional architecture

Figure 2 shows how the EDCP solution interfaces with enterprises as well as mobile operators, providing functionality in three main areas:

- device connectivity;
- policy control and charging; and
- management and provisioning of subscriptions and devices.

Devices are connected to enterprise applications through the EDCP and via the operator's mobile network. For transparent IP connectivity, the GGSN supports private IP networks, while the device access enabler grants access to devices on the internet. The platform includes a service execution environment, which provides support functionality to enterprise applications, such as subscribe/notify communication scheme and location services.

The policy and charging control block handles the various settings for tailored subscriptions, such as data capping and charging levels. Enforcement of the parameters takes place in the GGSN and online charging systems (OCS). The latter components also pre-rate and sort charging information – Call Detail Records (CDRs)- for each enterprise and operator. CDRs then are transferred to the operator's billing system according to a desired control cycle.

For operators and enterprise users, dedicated portals provide access to the platform for service level agreement (SLA), order and account management components. The operator can, for example, create enterprise-specific subscriptions, set up portals and monitor SLA reports. Through the self-service portal the enterprise can purchase services, order SIM cards, and monitor real-time/statistical data on the devices. The self-service portal also includes provisioning of subscriptions into the EDCP components as well as auto configuration of connectivity parameters into the devices. All devices supported by the EDCP are provisioned in the subscription database.

The OSS/diagnostics component provides operational and maintenance functions, such as alarm handling, as well as statistics for SLA reporting. A subset of status information and alarms is provided to the operator's network operation center.

Deployment architecture

Software as a service offered in a cloud style is a convenient and a cost-effective way to connect devices and applications. The cloud model uses pay-as-you-grow characteristics, rapid elasticity of system resources and ease of use. In the M2M arena there will be many different devices. Some will send and receive small amounts of data infrequently, some will send small amounts often and others will send and receive large amounts of data often or rarely. What M2M devices have in common, however, is that they could all benefit from the convenience of re-using infrastructure nodes for M2M services such as provisioning, connectivity, charging and policy.

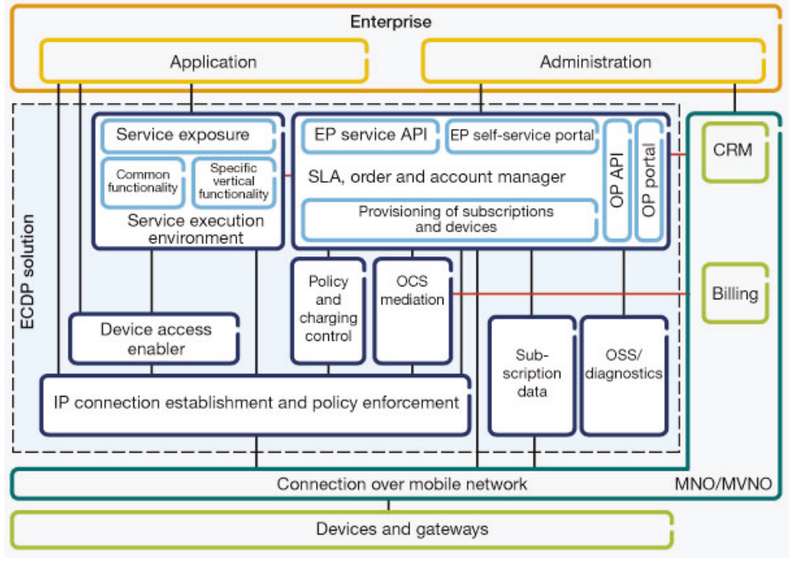


Figure 2.
Ericsson Device Connection Platform architecture

Berndt Wallin, Lars-Örjan Kling, Tomas Holm, Robert Skog, Miguel Blockstrand
Ericsson

Xiang Su, Jukka Riekki, Janne Haverinen, Johanna Nieminen , Jukka K. Nurminen

ENABLING SEMANTICS FOR THE INTERNET OF THINGS – DATA REPRESENTATIONS AND ENERGY CONSUMPTIONS

The development of Internet of Things (IoT) applications can be facilitated by encoding the meaning of the data in the messages sent by IoT nodes, but the constrained resources of these nodes challenge the common Semantic Web solutions for doing this.

Internet of Things (IoT) is expected to bring the Internet truly into our everyday lives by connecting a vast amount of devices and objects (the so-called things) to the Internet. All these things will communicate with other peers and servers in the Internet. The resulting uniform access to things will introduce significant possibilities for IoT applications.

Even more can be achieved if semantics is included in the information produced by the IoT nodes. Semantics enables machine-interpretable and self-descriptive data and facilitates information integration and share, and inference for new knowledge. However, since IoT nodes are often small devices with modest computing, communication, memory and energy resources, they introduce challenges not present in the common scenarios of Semantic Web. Hence, the main challenge is to add semantics without breaking the constraints on resource usage. In this article, we study how to enable richer semantics for IoT data, and evaluate different approaches with energy efficiency with a simple sensor system. Our sensor node measures acceleration and magnetic field, both in three dimensions, and temperature as well. This kind of sensors could be widely deployed in the IoT smart environments. We focus on different data formats enabling semantics, rather than protocols, architectures or ontologies in this paper.

Data formats

One of the main challenges of IoT data formats is mapping between data formats and models used for constrained IoT nodes and data formats and models used in the Web and Semantic Web. A data format should set minimal requirements for both IoT nodes and the consumers of data. That is, the solution should increase the nodes' resource consumption as little as possible, the solution should be general and any consumer should be able to interpret the data with minimal effort and apriori knowledge. Moreover, the data format should be compatible with Semantic Web, as only then the existing Semantic Web tools can be used.

Semantic Web communities, like W3C, have established specifications for formal knowledge representations, like RDF, OWL, N3 and Turtle. These knowledge representations can also be utilized for representing IoT data. The simplest way of semantically representing a measurement made by an IoT device with RDF, is denoting the IoT device as the subject, the measured quantity as the property, and the measured value as the object. For example, "Sensor 1" is the subject, "Temperature" is the property, and "25" is the value. The unit of measurement can be defined separately.

However, these formats are designed to be used by Web applications; hence resource usage was not the main issue in their development. SenML and Entity Notation (EN) [1] are targeted for resource-constrained devices. A SenML description carries a single base object consisting of attributes and an array of entries. Each entry, in turn, consists of attributes such as a unique identifier for the sensor, the time the measurement was made, and the current value. SenML can be represented in JSON, XML and Efficient XML Interchange (EXI) formats. The SenML format can be extended with further semantic custom attributes. For example, the Resource Type attribute can be used to define the meaning of a resource. EN is another lightweight data format that supports Semantic Web technologies. EN has been designed to be compatible with RDF and OWL and it has almost equal expressivity as RDF and N3 on the data exchange level. Its compact format can only include a UUID and some variables (for example, sensor measurements, etc. are variables in EN).

We compare the semantic expressivity of RDF, N3, SenML and EN in Table 1. RDF, N3 and EN can be mapped to conceptual graphs straightforwardly, as they all have a (subject, property, object) triplet structure as the base representation. Hence, they support ontologies. SenML has a more arbitrary data structure, which cannot be mapped to a conceptual graph in a similar fashion. Hence, SenML data cannot be utilized by knowledge-based systems as easily as the other alternatives. On the other hand, SenML may be easy to produce by IoT nodes, because it resembles the basic data structures of programming languages. The compact EN format has the same benefit. The type of the data can be defined with all these formats, which facilitates associating measured data values to concepts. RDF and N3 support rich XML Schema data types, while SenML allows only four basic data types. EN packets do not include data type information, but such information can be accessed from related knowledge representations. All these data formats support external semantic information, but in different fashions.

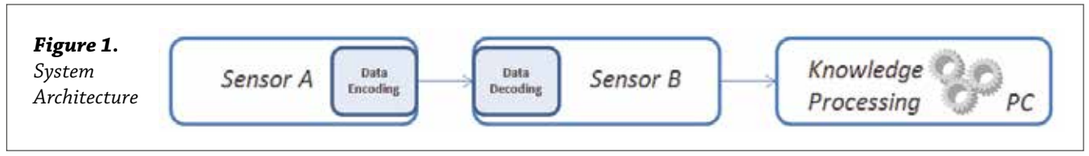


Figure 1. *System Architecture*

	RDF	N3	SenML	EN
Conceptual Graphs	Y	Y	N	Y
Triplet Relations	Y	Y	N	Y
Device Type	Y	Y	Y	Y
Data Types	XSD	XSD	4 types	N
External Semantics	Y	Y	Y	Y

Table 1. *Data format comparison*

Energy efficiency

Energy consumption is a key issue for IoT nodes. Hence, when semantics is added into IoT, energy-efficiency is a key criterion for comparing alternative solutions. Energy consumption together with other limited resources is one of the key drivers in wireless sensor network research. For example, it is reported in [2] that communication is over 1,000 times more expensive in terms of energy than performing a trivial aggregation operation. However, widely cited surveys [3, 4] do not have any explicit discussion on adding semantics to the data. It seems that integrating sensors into Semantic Web has not yet attracted the attention of researchers.

We measured the energy consumptions of encoding and decoding for different semantic data formats of the same data in a sensor system. As shown in Figure 1, this system consists of two sensors (based on Atmel's 8-bit ATmega32 microcontroller) communicating with Bluetooth and a knowledge processing component on a PC. Sensor A encodes the different formats and sends them to Sensor B. Sensor B decodes these data formats to formats compatible with a knowledge processing component. As a result, the knowledge system can reason additional knowledge and actions based on the data generated by IoT nodes.

Figure 2 presents energy consumption comparison on sensor A. Generating SenML/EXI messages requires more computing energy than other alternatives, but transmission energy consumption for SenML/EXI is among the lowest ones. When comparing overall energy consumption, SenML/EXI requires more energy than the two times longer SenML/JSON and SenML/XML messages. The short EN format requires the least energy and other alternatives consume at least double that amount. Generating short EN messages only consumes about 35% of generating RDF/XML messages, which consume the largest amount of energy. But on the other hand, the receiver of the short EN messages needs one more step (on sensor B or PC) to extend the short EN packet into a complete EN packet that is directly comparable with RDF and N3.

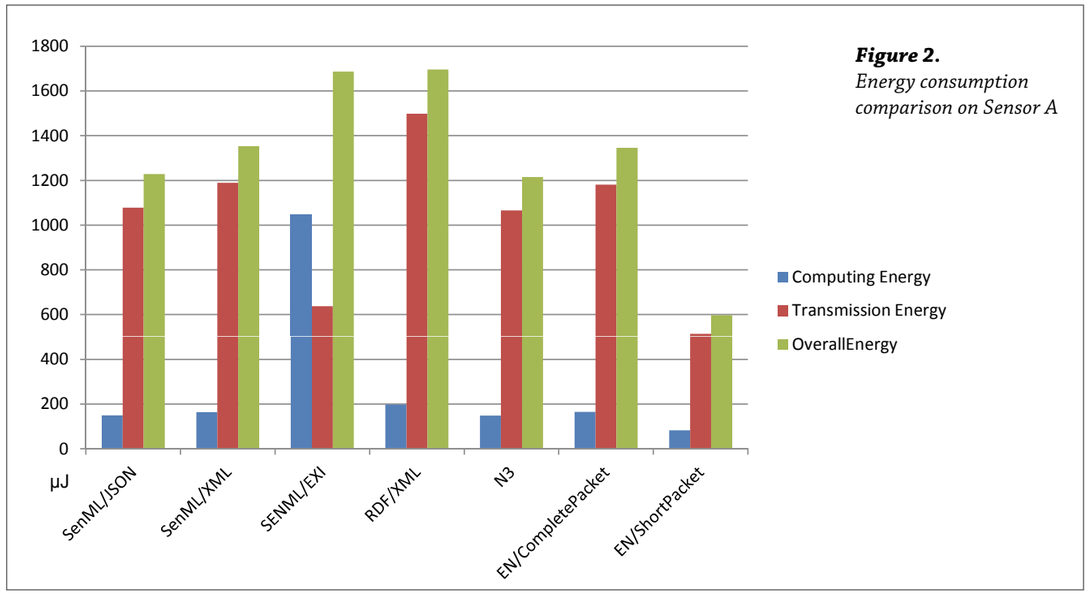


Figure 2. *Energy consumption comparison on Sensor A*

Discussion

We are studying the best ways to add semantics to IoT data. Even this simple experiment illustrates how big a difference a data format can make in energy consumption. One interesting potential scenario for our future work is a gateway receiving data from several similar sensors, aggregating the data values, and sending the resulting data forward.

Many other factors have an effect on energy consumption, but we will mainly focus on data formats supporting semantics; on their expressivity and resource consumption. The other factors include the header lengths of the protocols, messaging patterns and architecture. In addition, the meaning encoded in the messages needs to be shared by all entities producing and consuming the data. That is, ontologies are needed. Moreover, as IoT systems will produce large amounts of data, reasoning techniques that scale and infer useful information in a reasonable amount of time are called for. These reasoning techniques need to be deployable into devices with varying computing resources.

Acknowledgment

This work was funded by the Internet of Things (IoT) program funded by TIVIT and Tekes. We also would like to thank HPY Research Foundation and Tauno Tönning Säätiö for funding. The majority of Janne Haverinen's work was done when he was working at University of Oulu. All of Johanna Nieminen's work was done when she was working at Nokia Research Center, Helsinki.

References

[1] X. Su, J. Riekki and J. Haverinen, Entity Notation: enabling knowledge representations for resource-constrained sensors, Personal and Ubiquitous Computing, volume 16 issue 7, Oct. 2012 pp 819-834.

[2] V. Cantoni , L. Lombardi, P. Lombardi , Challenges for Data Mining in Distributed Sensor Networks, 18th International Conference on Pattern Recognition (ICPR'06), p. 1000-1007 (2006).

[3] J. Yick, B. Mukherjee,D. , Wireless sensor network survey, Computer Networks, Volume 52, Issue 12, 22 August 2008, pp 2292-2330.

[4] K. Sohrabi, J. Gao,V. Ailawadhi,G.J. Pottie, Protocols for self-organization of a wireless sensor network, Personal Communications, IEEE , vol.7, no.5, pp.16-27.

Xiang Su, Jukka Riekki, Janne Haverinen
Department of Computer Science and Engineering
University of Oulu, Finland

Johanna Nieminen
TeliaSonera, Helsinki, Finland

Jukka K. Nurminen
Department of Computer Science and Engineering
Aalto University, Finland

ONTOLOGY ALIGNMENT FOR INTEROPERABILITY ON THE IoT

The Internet of Things is coming, but it needs a semantic backbone to flourish. Some 50 billion devices are expected to be connected to the Internet by 2020, making interoperability a major concern. Most of these devices will be deployed for industrial and public infrastructure domains, where a need for the emergence of standardized domain models, i.e. ontologies, is well recognised. We believe, however, that in the customer segment of IoT that comprises smart homes, smart offices, connected vehicles, and similar, creation of standard ontologies is much more challenging but also less beneficial. Therefore we investigate how IoT environments can function with the help of ontology alignment solutions that discover the mappings between the concepts from two alternative domain models in an automated fashion.

Our work is motivated by a vision of the Internet of Things where 3rd-party software application development for IoT environments like smart homes is as easy and as popular as the development of applications for smartphones nowadays. One barrier is a big number of various and non-interoperable IoT platforms, and too small a market penetration of each. We aim at a solution, therefore, which enables developing applications that are generic in the sense of being able to communicate with sensors and to control actuators connected to the Internet through different platforms. This is in contrast to the present restriction of always developing an application for a very particular IoT platform.

Figure 1, depicting our prototype system setup, exemplifies this concept. Assume one user has a ThereGate gateway and a Z-wave contact sensor, while another user has a Texas Instruments USB dongle and a ZigBee contact sensor. Each platform defines its own format for queries and its own way of describing door open/close events, including different data structures and names for properties ('DoorOpen': 'true' vs. 'action': 'open'). Yet, both users are able to deploy exactly the same application code from an online IoT App Store and successfully run it. Note that interoperability is not burdened on the application or its execution platform, as is in many other approaches. We assume no particular execution platform and the application can define yet another, its own, data representation format. It is a smart proxy in-between the application and the sensor that manages the interoperation. An additional task of the smart proxy is the discovery of appropriate sensors/actuators within an IoT environment to match the requirements of the application.

The central element of our Semantic Smart Gateway Framework (SSGF) is a smart environment registry that contains semantic descriptions of 'things' (a door), connector devices (a contact sensor) and their associations to 'things' (the contact sensor attached to the door), as well as the deployed applications. These descriptions are based on an IoT ontology we developed, which is the extension of W3C Semantic Sensor Network (SSN) ontology, which is, in turn, based on DUL (DOLCE Ultra Light) upper ontology. The IoT ontology only provides vocabulary related to generic sensing/actuating, while for any domain-specific concepts some custom classes are to be used and can be freely defined. For match-making of sensors/actuators and applications, an application's requirements are expressed as SPARQL query patterns. Similarly to semantic descriptions of things and devices, these patterns are defined using our IoT ontology plus some custom classes for domain-specific concepts. An ontology alignment solution is then applied to find the mappings between the custom classes used in device descriptions and application patterns.

The same ontology alignment solution is also utilized for the second, a more complex, alignment task that is the automated transformation of data formats used by an application and a sensor/actuator. Figure 2 depicts the related workflow.

The semantic descriptions of both the device and the application have to include examples for all relevant query and response messages, which can be e.g. XML, JSON, or complex URIs. From an analysis of these example messages, OWL ontology models for the device and the application are generated and heuristically refined. These ontology models are then fed into the ontology alignment solution to discover the concept mappings. Finally, these mappings are used in the run-time by the message translator component of the smart proxy to provide two-way communication message transformation between the application and the devices.

SSGF facilitates automated deployment of generic and legacy IoT software in environments where heterogeneous devices also have been deployed. SSGF functionality can be implemented by an IoT platform provider to enable their platform to run applications not originally designed for it, i.e. to extend the range of applications available to their customers. Alternatively, SSGF can be delivered by an independent party as a service, resulting in a novel "interoperability-as-a-service" paradigm. Practically, this means operating a scalable web portal where the end-users can register their things and devices, as well as deploy application descriptions from app stores. The data traffic between applications and end-users' devices will also go through this web portal.

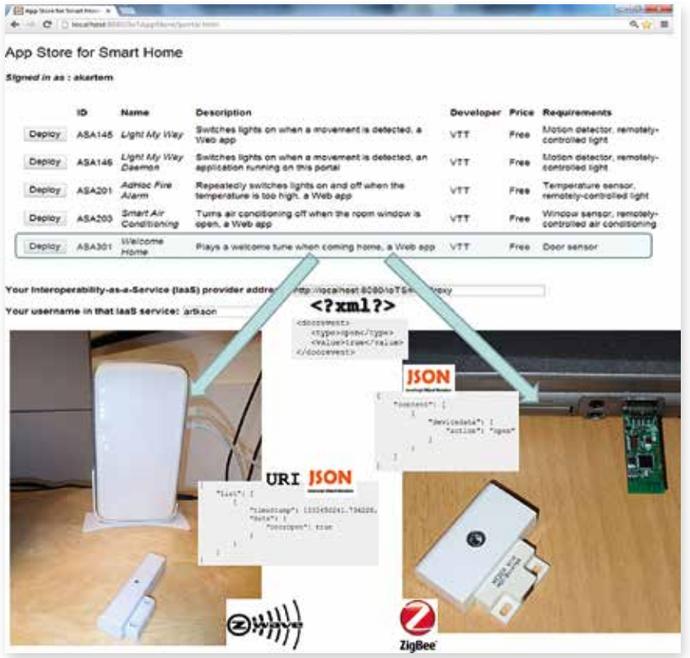


Figure 1. *Deploying an application to two IoT platforms*

More technical details about SSGF can be found in [1].
[1] Kotis K. and Katasonov A. (2013) Semantic Interoperability on the Internet of Things: The Semantic Smart Gateway Framework, Int. J. Distributed Systems and Technologies, IGI Global, in press

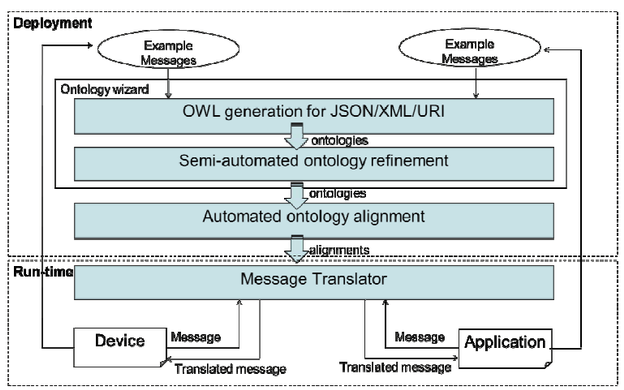


Figure 2. *Workflow for automated translation between data formats*

Artem Katasonov and Konstantinos Kotis,
VTT Technical Research Centre of Finland

Pin Nie, Patrik Nisen and Jukka K. Nurminen

COMBINING SENSOR NETWORKS WITH SOCIAL NETWORKS BY XMPP

On the development of sensor networks, a recent trend towards a web of things leverages substantial web technologies and services for the integration of physical world and virtual cyberspace. In order to further simplify sensor application development, we created an XMPP sensor bot to combine sensor networks with social networks via instant messaging and presence service. In this demo, we show a complete end-to-end solution to enable two-way communication between wireless sensor nodes and any Jabber clients on the Internet. The prototype is implemented on SunSPOT and demonstrated in four use cases.

Sensor networks are able to perform persistent environmental, structural and object monitoring which greatly enhances our situational awareness in real time. By combining sensor networks with social networks, we can build a strong link between the environment and public. As a result, a tight integration of physical world and virtual cyberspace will improve our daily activities and reduce the negative impact on the environment.

There are a number of methods to bridge the gap between sensor networks and social networks, such as the Internet of Things (IoT) [1] and the Web of Things (WoT) [2]. An extensive study on the integration of sensors and social networks is provided in [3]. In our research paper [4], we identified two critical elements to boost the integration of sensor networks into the Internet, namely a uniform communication language and a ubiquitous application connectivity. Therefore, in favor of its common XML data representation and pervasive instant messaging and presence service, we selected XMPP as the basis to develop an end-to-end (E2E) solution enabling twoway communication between wireless sensor nodes and XMPP instant messaging clients. Driven by large-scale application scenarios, existing works e.g., [5] on applying XMPP on sensor networks are rather complicated. Moreover, most sensor applications require re-programming of the sensor nodes for different use cases. These two drawbacks limit the flexibility of sensor application development, especially for stand-alone use cases which do not require large scalability. For daily use, people may switch multiple applications through versatile functionality of a generic sensor platform, similar to the way we use many apps on our smartphones.

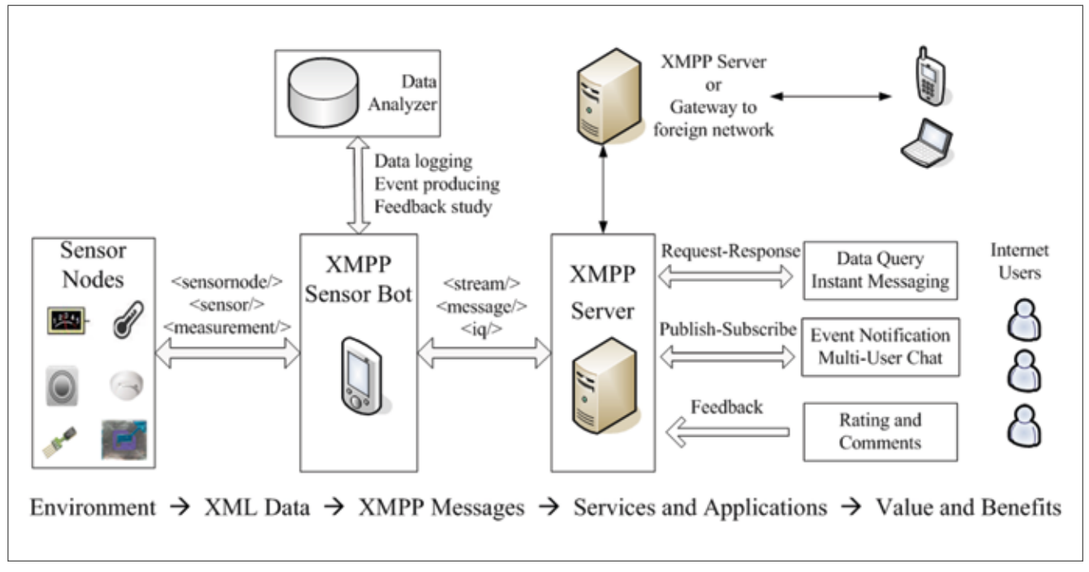


Figure 1. *XMPP Architecture integrated with sensor networks*

Ideally, a flexible sensor application architecture should allow changes of different processing logics of multiple types XMPP Architecture integrated with sensor networks of measurements produced by a generic hardware platform without re-programming the sensor device itself. Different measurements can be used for different purposes depending on the application context. In many cases, we just need to run new data processing logic at the gateway and to update measurement parameters on the sensor devices accordingly. Driven by this notion, we created an XMPP sensor bot to communicate with wireless sensor nodes and to execute different measurement processing logics for diverse application scenarios. All processing logics are defined in a single text file of XML format called *rulebook*, which can be updated by the user. After the initial launch, the user can remotely switch the application from one to another by sending a specific message *"app = app id"*. There is no hassle to reprogram the sensor node or the gateway. To create a new application, the user just needs to add new processing logics with associated measurements in the rulebook and re-launch the XMPP sensor bot at the gateway. Our solution simplifies sensor application development and exhibits flexibility in application deployment. Our contribution makes it easier to integrate application knowledge on a generic sensor device without demanding programming skills.

Flexible XMPP Sensor Bot

Figure 1 illustrates our XMPP-based architecture integrated with sensor networks. On the right side of the figure, our solution extensively leverages existing XMPP networks and services. On the left side, we implemented two Java programs for data acquisition and measurement processing logics. One program is a sensor client on the wireless sensor node (SunSPOT) and the other program is a host (XMPP sensor bot) running on a gateway which is connected to other XMPP servers on the Internet. All messages are encoded in a uniform XML format. To save energy and memory space on the wireless sensor node, we developed abbreviated XML tags and attributes to encapsulate data at the sensor side. This encoding policy produces shorter packets and makes transmission more efficient. The following example reports battery measurement (encoded in type 2) of the *"node1"* a value of 80%. A more efficient and scalable solution is to apply a binary XML standard, e.g., Efficient XML Interchange (EXI). However, we could not find such highly optimized library on SunSPOT.

< sn i =' node1 ' >< m y =' 2 ' v =' 80 ' / >< /sn >

The hardware capability on SunSPOT includes a light sensor, an accelerometer, battery measurement, a temperature sensor, two buttons and eight tri-color LEDs. The XMPP sensor bot host program supports three types of comparators *(less-than, greater-than and equals)* when testing measurement values from the sensor node. In addition, the XMPP sensor bot uses regular expression patterns and the operator *contains* to validate incoming messages from external XMPP entities and then activates the associated action when the condition matches. On the other side, the sensor client receives task configurations from the XMPP sensor bot and reports the required measurements at a configurable sampling interval, which balances application sensitivity and energy preservation. By combining three types of physical measurement, a button user interface, eight tricolor LEDs and three types of comparator, we can define several event triggers for a few useful applications.

> *Ideally, a flexible sensor application architecture should allow changes of different processing logics of multiple types.*

Demos

Figure 2 presents four use cases to demonstrate the functionality of our prototype and flexibility of the design. These demos are door bell, coffee maker monitor, toilet status monitor, and senior fall detector.

The first demo *door bell* implements two-way wireless communication between a visitor and the host. When the visitor presses the left button, the SunSPOT sensor node transmits an event signal to the XMPP sensor bot program which sends a notification message to the host Jabber client via instant messaging service. The host could reply his status in three options: *busy, free, or wait*. A *busy* reply blinks all LEDs on the SunSPOT in red. A *free* reply sets LEDs to green and a *wait* reply sets LEDs to orange. This interaction improves the host responsiveness and also shows the number of visitors within a certain time.

The second demo uses light and temperature sensors on SunSPOT to infer if there is a fresh pot of coffee on the coffee maker. The measurement processing logic defined in the rulebook generates an event when the temperature measurement stays above a threshold and light measurement is below a threshold of darkness, meaning a pot of fresh coffee is ready. A user can ask the XMPP sensor bot if there is any fresh coffee by sending a message "got coffee?" and book the coffee by sending a message "book coffee".When the coffee is reserved, all LEDs are in red and can be cleared by pressing the left button. This demo is the most complicated scenario in our use cases, because it correlates two types of measurements for event detection and also supports two-way communication.

The third demo monitors toilet usage by using the light sensor. The SunSPOT detects a light transition from dark to bright and vice versa. The transition indicates if the toilet is occupied or unoccupied. The toilet's status can be retrieved upon request by the user. This demo prevents

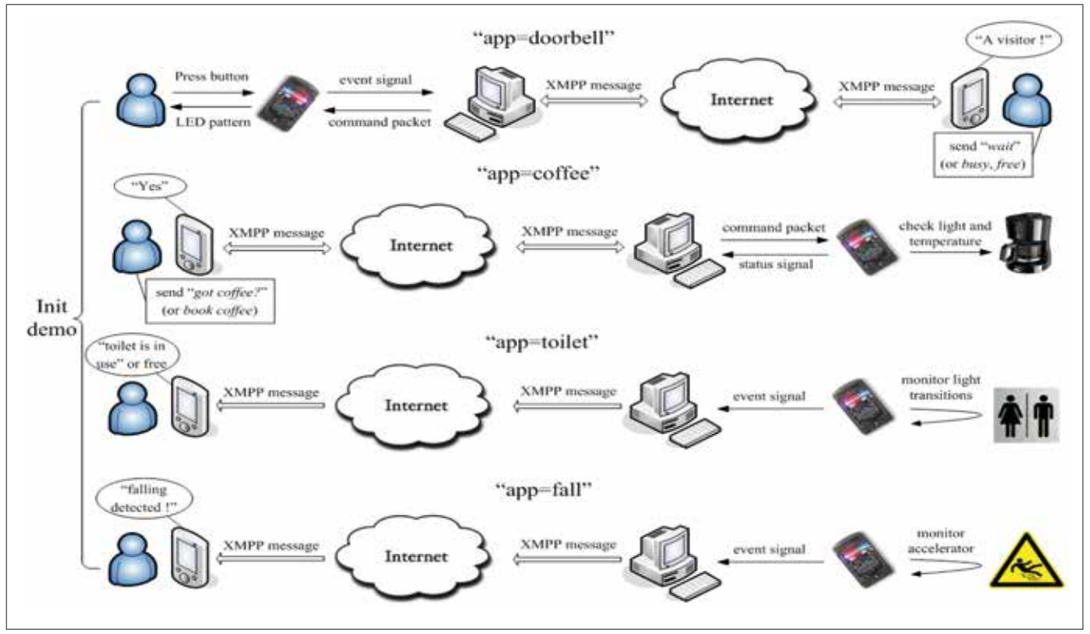


Figure 2. *Four demos on the application of XMPP Sensor Bot*

conflict use of the toilet. However, it cannot tell if the user forgot to turn off the light after the use of toilet.

The fourth demo monitors vibration amplitude and detects falling by using the accelerometer. If the acceleration measurement exceeds a predefined threshold over a certain time, a fall event is generated and an alert notification is sent to an associated supervisor (e.g., a relative) via instant messaging service. This demo can be used to set of an alarm when seniors have accidents.

All four demos implement different measurement processing logics in the rulebook. The hardware platform and network setup remain the same. The sensor client and the gateway host program are programmed only once. After the initial launch, the user can remotely change the application by requesting the XMPP sensor bot with the application identifier.

Conclusions

Our prototype with four demos exhibits great flexibility to combine sensor networks with social networks by using XMPP. We believe the sensor technology would benefit our daily life in a variety of applications. For future work, we plan to measure energy consumption and information throughput of each demo case for performance evaluation.

References

[1] L. Atzori, A. Iera, and G. Morabito, "The internet of things: A survey," Computer Networks, vol. 54, October 2010.

[2] D. Guinard, V. Trifa, F. Mattern, and E. Wilde, From the Internet of Things to the Web of Things: Resource Oriented Architecture and Best Practices. Springer, 2011.

[3] C. C. Aggarwal and T. Abdelzaher, "Integrating sensors and social networks," in Social Network Data Analytics. Springer, 2011.

[4] P. Nie and J. K. Nurminen, "Integrate wsn to the web of things by using xmpp," in S-cube'12: 3rd International Conference on Sensor Systems and Software. Springer, 2012.

[5] A. Rowe, M. E. Berges, G. Bhatia, E. Goldman, R. Rajkumar, J. H. Garrett, J. M. F. Moura, and L. Soibelman, "Sensor andrew: Largescale campus-wide sensing and actuation," IBM Journal of Research and Development, vol. 55, 2011.

Pin Nie, Patrik Nisen and Jukka K. Nurminen
Aalto University, School of Science Department of Computer Science and Engineering

NETWORKING SMALL DEVICES

One of major challenges for IoT and machine-to-machine communication is the connectivity of constrained devices to IP networks. The networking research in the TiViT Internet of Things SHOK program has concentrated especially on arranging IP connectivity and data delivery with resource constrained sensor devices, which do not always support the required network protocols by definition. During the first year the project concentrated on building and optimizing the communication network solutions to optimize the power efficiency of devices and to support so-called "sleeping nodes". The work focused on the Representational State Transfer (REST) based Constrained Application Protocol (CoAP) web transfer protocol to arrange the end-to-end connectivity of constrained sensor nodes and services. Since the majority of current sensor devices does not support networking protocols such as IP (Internet Protocol) and UDP (User Datagram Protocol), which are required for CoAP, the project also defined and developed solutions for the CoAP gateway to be used with energy-optimized protocols and sleepy devices as well as improved the communication between the CoAP gateway and end-user applications. In this article we provide an overview of developed communication architecture and demonstrated solutions for networking small IoT devices.

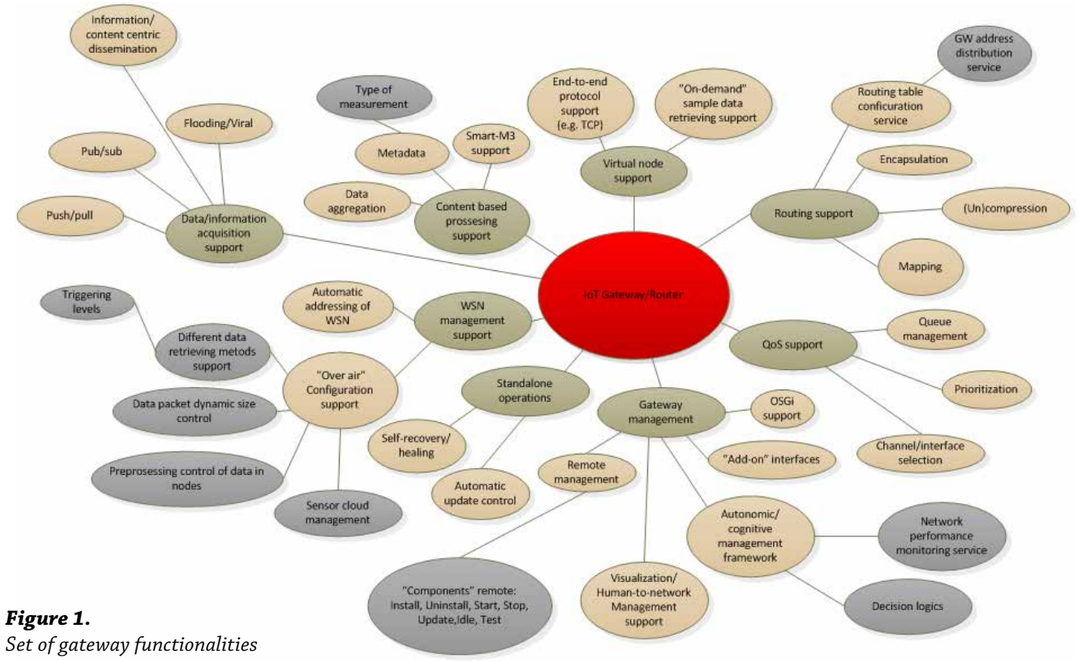


Figure 1.
Set of gateway functionalities

The connectivity to Internet architecture is one of the key elements for the Internet of Things. The Things are usually small and very restricted devices, having constraints in communication, energy and processing. From the connectivity point of view the devices may also form different kinds of network topologies from ad hoc communication to centralized star-like network topologies depending on for what purpose the devices are used and on possible environmental restrictions. For example, in wireless sensor network (WSN) the radio technology used can restrict the coverage area and in order to provide the connectivity for all the devices, some of them need to route and forward information from others, thus forming a mesh network topology. The devices may also be in "sleep" mode even long periods to save battery, which makes especially the real-time communication and system management challenging.

These different restrictions also dictate what kind of functionalities can be implemented for devices and system. One of the key components of the network system is a gateway, which enables the interconnection between different networks using different communication protocols. In order to provide end-to-end connectivity in IoT, i.e., the information delivery from sensors up to the users, gateways are often needed. One of the challenges is to find the basic/minimum set of functionalities for the gateway device based on the application scenario where it is used, and to be able to further enhance the system with

>>>

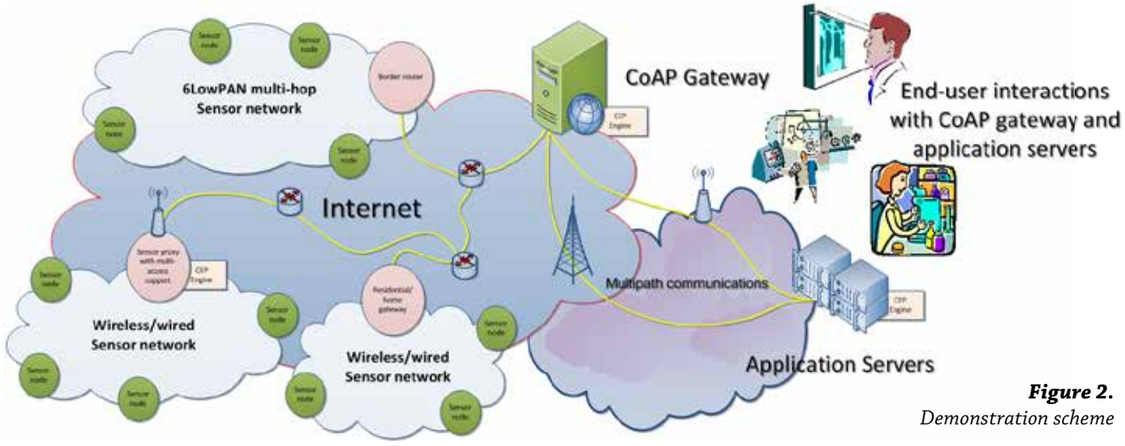


Figure 2.
Demonstration scheme

new functionalities in a modular way if the application and the use of the device changes. In Figure 1 we illustrate some of the basic functionalities for the gateway.

Due to the fact that nowadays almost all the end-user services and network communications are IP-based, the raw sensor data needs to be encapsulated into IPv4/IPv6 and either UDP or TCP protocol format to be effectively used in Internet and e.g. with web services. In addition, since the sensor devices are usually resource restricted and cannot include the same functionalities as more powerful devices, the CoAP protocol was introduced to enable web integration for sensors. CoAP is based on the REST architecture model similar to for example the current World Wide Web. It operates on top of a UDP/IP protocol stack, utilizes the request/respond model and HTTP-like messaging methods, and follows a server-client architecture including definitions for e.g. different types of proxies.

In the CoAP system especially for machine-to-machine and IoT communications, the so-called resource directories and proxies have a central role. The resource directory is an entity that is able to store the location of resources, such as temperature data provided by a sensor, and further provide it e.g. for end-user or web services. Proxy, on the other hand, is an intermediary node capable of forwarding and caching data, and doing protocol conversions. In our demonstration scheme during the TiViT IoT project, we defined gateway as an entity that is able to make the protocol conversions and in addition act as a resource directory and mirror proxy. The main target in our research was to improve especially the energy efficiency of the networked sensor system and to provide capabilities to further improve the autonomic features in both device and data management.

Gateways and Clients

During the first year the TiViT Internet of Things project concentrated on improving the CoAP-based communications in resource restricted small devices. Several demonstrations were built mainly concentrating on sensor gateway and device side protocol and functionality development. The main components which were developed and demonstrated were the gateway and proxy solutions enabling the seamless integration of sensor devices with web, 6LowPAN multi-hop capabilities, IPv6 to IPv4 interoperability in CoAP communications, Complex Event Processing and event management support, and multipath connectivity. The Figure 2 illustrates our demonstration scheme, including the gateways, decision making and end-user connectivity enhancements.

For providing sensor information from the constrained devices to the web, an implementation of the CoAP protocol and necessary functionality to read and send values on embedded devices was developed. The implementation can be run with just a few kilobytes of memory so it is suitable for even many of the most constrained devices. A less constrained implementation of the CoAP protocol was used on the gateway side (see Figure 2 top middle). The gateway works as a resource directory and a mirror proxy that is capable of storing both pointers to the resources and also the values when needed. The gateway implementation can provide the information both with CoAP and HTTP for simple integration with web services.

As an example of a multi-hop sensor network (see upper left corner of Figure 2), an energy metering sensor network with CoAP-based communications to the Internet via a CoAP gateway was built and demonstrated. The 6LowPAN border router connects the multi-hop 802.15.4-based sensor network to the Internet and CoAP gateway.

The 6to4 home gateway (see Figure 2 bottom middle) supporting dual-stack IPv4 and IPv6 CoAP Resource Directory functionality was also implemented for home networks. It was optimised for CoAP implementations on embedded linux devices and it supports integration and interoperability for low power CoAP clients.

In order to support autonomic behavior and event processing in IoT system, the Distributed Decision Engine (DDE) complex event processing system (see Figure 3) was introduced. The DDE was designed for information collection and processing from different information sources with the goal to provide an easy way to integrate different standard and non-standard information sources and to enable system-wide decision making. The DDE consists of 3 functional entities communicating in a publish-subscribe manner: Producers that produce the data, Consumers that do something with the data, and EventCaches that manage the data and the communication between Producers and Consumers. The EventCaches can be further cascaded and communicate with each other through the network, acting as a decision making agent in the network. In the demonstration DDE was used as a proxy/gateway, transforming the analog sensor data to IP/UDP/CoAP packets. The temperature and illumination sensors were used as data Producers and DDE EventCache on the sensor proxy node (see Figure 2 lower left corner) collected the data from the sensors. Finally the Consumer forwarded the data to the CoAP gateway/server using the CoAP protocol.

Using a multipath protocol for connecting the gateway to the Internet service was also studied and demonstrated (see Figure 2 right-hand side). As seen in other implementations, the sensor devices are assumed to connect with a low energy protocol to the gateway and then the connection from the gateway is made with multipath TCP (MPTCP). The benefits of multipath are increased availability, bandwidth and mobility. Many IoT devices are deployed in a mobile fashion, e.g. with users, trucks, ships, containers and automobiles. Most of these devices will connect to the Internet through a cellular network. The multipath using several cellular connections to improve the available bandwidth and the coverage was studied. Early analysis shows that it is possible to raise coverage and thus availability in a cellular network with multipath. However, the bandwidth seems not to increase as well as one could assume. Two radio interfaces that generally give 10 Mbps each rarely gave more than 15 Mbps aggregated, with the averages only little bit over 10 Mbps. However, the multipath increased the time when a several megabits connection was achieved.

Conclusions

One of the major challenges for IoT and machine-to-machine communication is the connectivity of constrained devices to IP networks. In this article we gave an overview of the networking research done in the TiViT Internet of Things SHOK program during the first year for tackling this challenge. The work was focused on Representational State Transfer (REST) based Constrained Application Protocol (CoAP) web transfer protocol to arrange the end-to-end connectivity of constrained sensor nodes and services. Since the majority of current sensor devices does not support networking protocols such as IP (Internet Protocol) and UDP (User Datagram Protocol), the project also defined, developed and demonstrated solutions for a CoAP gateway to be used with energy-optimized protocols and sleepy devices as well as improved the communication between the CoAP gateway and end-user applications.

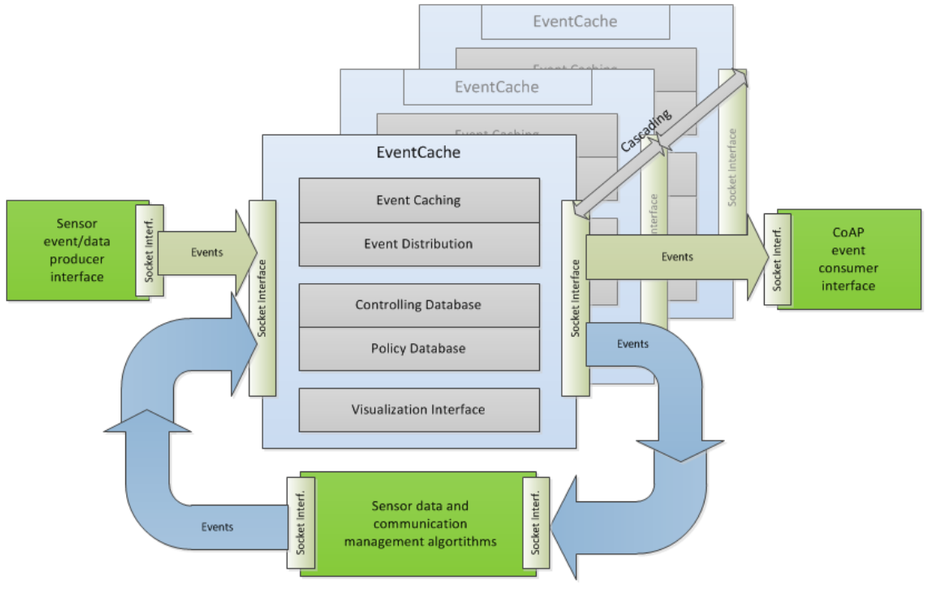


Figure 3.
DDE architecture

Jyrki Huusko, Mikko Majanen, VTT Technical Research Centre of Finland
Jan Melen, Ari Keränen, Ericsson Research
Sebastian Sonntag, Fida Khattak, Aalto University
Bilhanan Silverajan, Tampere University of Technology

Laila Daniel and Markku Kojo

TRANSPORT PROTOCOLS FOR INTERNET OF THINGS

While traditional networks like the Internet and cellular networks are engineered networks, Internet of Things (IoT) are self-engineered networks in the sense that nodes that are randomly scattered in a given geographical region function as a computer network in cooperating to solve a particular task. IoT typically have limited resources in terms of computation, communication, radio and battery life. As off-the-shelf items they are inexpensive, small in size and prone to failures in their operating environment. As IoT are deployed in a range of applications from real-time tracking to ubiquitous computing, they need to perform increasingly complex tasks in a reliable and efficient manner. A key to the operation of the IoT is the protocols designed for their use which should be simple, scalable, robust and efficient in making near-optimal use of resources, energy efficient, easy to maintain and deploy and also customizable to the need of the applications.

In this article, we focus on the analysis of transport protocols for IoT. A transport protocol is needed in IoT/ wireless sensor networks (WSN) for reliable data delivery that applications may require and to provide congestion control to regulate the data flow that applications may send to the network and also to achieve some sort of fairness in sharing the scarce network resources.

Requirements for the Transport Protocol for IoT

Broadly speaking, IoT can be regarded as WSN that is connected to the Internet. The transport protocol design for WSN is an active area of research and there is a vast literature on the topic. Based on our study on transport protocols for the WSN, we came up with the following requirements for the IoT transport.

Easy connectivity to the Internet: IoT transport should provide easy connectivity to the Internet. A typical topology for IoT is shown in Figure 1. The things are usually connected to a gateway that has a connectivity to the Internet.

Simple reliability and congestion control mechanisms: Based on the type of data transfer between the IoT and the sink, we can decide on the congestion control mechanisms needed. The sink collects the data from different IoT and it can be a special node or a server in the Internet.

In a push data type of transfer, fast transfer of a small amount of data from IoT to the sink is needed. Especially in scenarios involving actuators, the communication paradigm is usually request-reply. The sensor nodes act on the requests from the sink. Only simple reliability and congestion control mechanisms are needed in a push data and request-reply kind of data transfer.

The other data transfer scenarios include continuous data flow from the sensors to the sink, large data files are being transferred in a bursty manner, and reprogramming the sensor nodes or software updates. In the above scenarios TCP-like reliability and congestion control mechanisms are needed. It is preferred to have modifications only on the sender side (in IoT) than at the sink node.

Cross-layer assisted transport: The IoT transport would be able to use the crosslayer information regarding the link layer / physical layer status to enhance the congestion control mechanisms.

Energy efficient: The IoT transport should be energy efficient as the IoT devices usually run on batteries

Scalability: The IoT transport protocol should scale as the number of IoT connected to the Internet can be very large,

Low memory footprint: As the IoT have limited processing and memory capabilities, the IoT transport should have a small memory footprint.

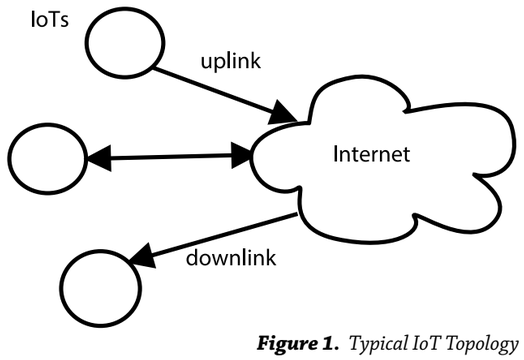


Figure 1. *Typical IoT Topology*

Proposed TCP Features for IoT transport

We propose a simplified TCP as a viable option for the IoT transport due to the following reasons.

As TCP [4, 1] is still the dominant transport protocol in the Internet, an IoT device with TCP/IP stack can easily be connected to the Internet. Most of the transport protocols for WSN are specifically designed for a particular kind of data or reliability mechanism needed, cannot be directly connected to the Internet without a protocol converter or a proxy on an intermediary gateway. The deployment of proxies always brings scaling problems.

Most of the transport protocols for WSN in some way or other implement many features of TCP protocol, for example, many WSN transport protocols use TCP mechanisms such as initial handshake, ACKs, congestion detection by dupacks or timeout, AIMD rate adaptation etc. So instead of reinventing the wheel, adapting TCP to the IoT environment may be a better design choice for the IoT transport.

To have a low memory footprint, it is possible to simplify/remove parts of the TCP implementation that are not essential based on the requirements of the transport and the data type used in a specific IoT environment.

TCP implementations with a low memory foot print such as uIP, lwIP [2] are already available as open source and are implemented and tested in operating systems like Contiki [3] which is an operating system for many microcontrollers and low-power embedded devices.

The proposed TCP for IoT may have the following features. As the IoT packets are quite small, including TCP header compression could reduce the overhead due to the TCP header.

In IoT scenarios where data transfer is either push mode or request-reply, we configure TCP similar to the stop and wait protocol. Packet loss can be detected by retransmission timeout and we go for the simplest congestion control mechanism of retransmission timeout and back-off. In bulk transfer of data between IoT and sink, standard TCP sliding window mechanisms and congestion control can be used. Other minor changes such as TCP SACK-related adjustments, RTO adjustments, etc may be included.

TCP fast open [5] may be implemented in scenarios where fast data transfer avoiding the three-way handshake delay is needed. This feature allows to send data with the TCP SYN packet. We have to study the feasibility of the implementation of cookie authentication mechanisms in IoT devices. The simplified reliability and congestion control mechanisms allow a small footprint for the simplified TCP.

> *TCP based solutions are viable for transport in IoT.*

References

[1] Allman, M., Paxson, V., and Blanton, E. (2009). TCP Congestion Control. Internet RFCs, ISSN 2070-1721, RFC 5681.

[2] Dunkels, A. (2003). Full tcp/ip for 8-bit architectures. In Proceedings of the 1st international conference on Mobile systems, applications and services, MobiSys '03, pages 85–98.

[3] Dunkels, A., Grönvall, B., and Voigt, T. (2004). Contiki - a Lightweight and Flexible Operating System for Tiny Networked Sensors. In Proceedings of the First IEEE Workshop on Embedded Networked Sensors (Emnets-I).

[4] Postel, J. (1981). Transmission Control Protocol. Internet RFCs, ISSN 2070-1721, RFC 793.

[5] Radhakrishnan, S., Cheng, Y., Jerry Chu, H-K., Jain, A., and Raghavan, B. (2011). TCP Fast Open. In Proceedings of the Seventh COnference on emerging Networking EXperiments and Technologies, CoNEXT '11, pages 20:1–20:12.

Laila Daniel and Markku Kojo
Department of Computer Science,
University of Helsinki

Otto Waltari

CONTENT-CENTRIC NETWORKING IN IoT

In the modern Internet, content is addressed by hostnames. A hostname refers to a device in the network, thus it is eventually pointing to a location. This is the paradigm in the dominating Internet Protocol (IP), which was originally designed for sharing physical resources rather than disseminating data. Today the vast majority of network traffic is about moving data which is not related to its location.

Content-Centric Networking (CCN) is a new networking architecture that addresses content by name instead of location. It has no notion of hostnames and data is published with names describing the data. The essence of named content is that data is requested based on the actual content instead of where the content is stored. CCN also provides in-network short-term storage for content while it is being disseminated in the network. For long-term storage, separate repositories can be deployed in the network.

We see that the Internet of things field can benefit from the Content-Centric architecture because of various reasons. First of all, it provides a higher abstraction level between the sensors and the user application. It is irrelevant to know where the requested sensor data comes from as long as it can be addressed with a unique name and we can verify its authenticity and freshness. Secondly, sensor devices can benefit from the in-network storage. Short-term storage is similar to caching, which reduces workload on sensor devices that are used by several user applications. On the other hand the repositories designed for long-term storage can be used to store the historical data of a sensor.

Finally, the content naming scheme can be extended with suffixes. By default CCN is pull-oriented and content only flows in one direction. With name suffixes we are, however, able to send actuator commands to sensors with remotely controllable features, such as light switches. This can be seen as requesting an action instead of data. The sensor side only has to treat the request accordingly.

We have implemented a testbed in a greenhouse with some sensors providing their readings out as CCN content objects. The system runs on a ThereGate home automation router with some Z-Wave sensors attached. We are looking forward to demonstrating this in near future.

Otto Waltari
University of Helsinki

Tuomas Tirronen

LTE ENHANCEMENTS AND M2M

3GPP Long Term Evolution (LTE) is an evolution of both radio and core network aspects of the previous 3G technologies, achieving, for example, increased data rates, better spectral efficiency and lower latencies in the whole network. The work in 3GPP continues all the time by enhancing different aspects of LTE to provide even more flexibility in use scenarios for network equipment vendors, network operators and the end-users of the technology.

Visions of a future with billions of connected devices have been made and this connectivity will be provided by a variety of current and future communication technologies. The different traffic patterns and different requirements of various M2M applications will pose some challenges for these technologies, including LTE. In the Finnish IoT SRA we have been studying some of the issues and solutions for supporting wide-scale adoption of LTE for M2M applications.

As a cellular technology, LTE has desirable characteristics for M2M communications. The deployment of new devices using LTE radio is easy from a connectivity point-of-view and the coverage and deployment possibilities are superior compared to those of short-range technologies. Also, 3GPP has an established security architecture which can be used to secure the communication between the M2M entities.

Some future challenges include making the LTE radio technology and procedures energy efficient from the user equipment (UE) point-of-view and how to enable efficient usage of resources in scenarios with a large number of M2M devices. Based on the above, our work in the IoT SRA so far has focused on LTE radio energy consumption, signaling reduction and on 3GPP security mechanisms.

Power consumption

In the IoT project, we have studied LTE energy and power consumption aspects and investigated the possibilities to enable more energy-efficient M2M communications. We have developed and used a power consumption model with which we can address the energy consumption of M2M devices using LTE radio access. We model the transceiver of the device to consist of receiver chain, transmitter chain, fine and coarse clocks and a baseband processing part. The key LTE feature which can be used to lower the energy consumption is discontinuous reception (DRX). DRX enables the device to turn most of its radio circuitry off during non-active communication periods providing sleeping opportunities. Our results clearly indicate that if the current maximum allowed DRX cycle length of 2.56 seconds is extended, it is possible to trade-off the device responsiveness for substantially longer battery lifetime.

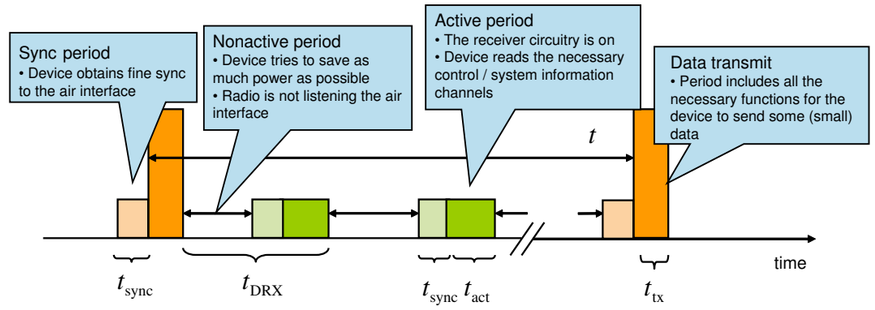


Figure 1. *Overview of DRX operation in the M2M context. The assumption is that the device sends infrequent data, such as sensor readings in uplink, without requiring explicit application-level ACKs. The operation consists of data transmit periods, which have multiple cycles of DRX sleep and active periods between them. The longer the DRX cycles are, the less time the M2M device needs to spend having its radio receiver on, thus saving energy.*

Longer cycles mean fewer cycles in total and less time when the receiver chain needs to be on, ultimately resulting in lower energy consumption.

Thus, by giving up on the responsiveness of M2M devices, the energy consumption can be reduced significantly. Our results indicate that extending the maximum DRX cycle to around two minutes would yield notable gains, after which the relative gain is not as large when compared to the current maximum. Further, just extending the data sending periodicity does not give notable benefits after a certain point. Instead, the energy consumption during sleeping times, in our model consisting of base power consumption and the coarse granularity clock, is a key parameter whose optimization would yield even greater savings.

From a radio resource point-of-view, the device can be either in idle or connected state. DRX can be used in both of these states, thus the cycle length in both could also be increased for longer energy saving opportunities. Keeping the M2M device in the connected state would have the benefit of a reduced number of signaling messages needed for transmitting the data. However, in the current networks and implementations, the UEs are dropped to the idle state when they are not actively participating in data transmission.

3GPP is studying the possible implications of extended DRX cycles on LTE in different scenarios, in both idle and connected mode, and at this point we do not yet know what will be the allowed DRX cycle lengths in the future.

Signaling reduction

Another possibility for optimizing the LTE radio access and core network for M2M is to perform signaling optimizations. For devices sending small data infrequently, the number of signaling messages over the air interface and between the core network nodes can become a problem especially when many devices are communicating at the same time. One optimization possibility is to reduce the required total number of radio resource control (RRC) protocol messages over the air and signaling messages in the core network when the transmitted data sizes are small and infrequent. There are several different options for how the signaling reduction could be made and the 3GPP working groups are discussing the possible future alternatives at the moment.

Security

Generic Bootstrapping Architecture (GBA) is a security solution standardized by 3GPP. It extends the security infrastructure of cellular networks to the Internet. GBA provides a secure and flexible user authentication mechanism for application services. The cellular operator is responsible for establishing a shared secret between the user Subscriber Identity Module (SIM) and the network service (such as Google, Facebook etc.) being accessed by the user. It essentially provides a login service to network services with SIM cards. This authentication mechanism has the additional advantage of not requiring any user enrollment phase.

In the past, we had implemented a GBA prototype for browsers in iPhones and iPads. The prototype allowed a cloud administrator to log into the cloud framework for management and administrative tasks. This login was secured with strong SIM credentials stored in the iPhone/iPad SIM card.

In the IoT SRA project, our goal was to investigate if the same technologies and standards can be reused "as is" on some of the most resource-constrained devices by efficient programming. GBA could then be used as one option for secure authentication and communication in the IoT ecosystem. We have developed a GBA prototype on a 8-bit micro-controller Arduino Mega board. This prototype implements the entire standard over HTTP interfaces as defined by 3GPP, including cryptographic (AES-128) and hash (MD5, SHA256) algorithms within a few kB of RAM. After authentication, this prototype is able to securely communicate its sensor data (temperature values) to a Mirror Proxy running in the cloud. As a part of the IoT SRA we continue to investigate how 3GPP standards can be applied to the IoT space and how we can gain from the existing cellular infrastructure.

Tuomas Tirronen
NomadicLab / Ericsson Research

Ali Hazmi, Mikko Valkama and Juho Pirskanen

IEEE 802.11AH: PROMISING TECHNOLOGY FOR IoT AND M2M APPLICATIONS

The rapid developments of the Internet-of-Things (IoT) and Machine-to-Machine (M2M) applications stimulate for the design of a new radio interface that can satisfy the conflicting requirements of these applications, including small-size, infrequent traffic, energy efficiency, large device populations and long transmission ranges. Our aim here is to show the feasibility of IEEE 802.11AH technology for M2M and IoT applications and identify the main challenges.

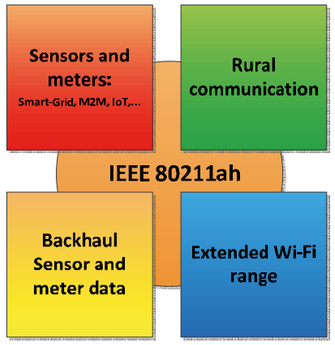


Figure 1. *IEEE 802.11ah main use cases.*

Introduction

An emerging technology based on the IEEE 802.11 standard family is currently being developed and standardized, under the IEEE 802.11ah group, and aims to define improved PHY and MAC layers that are properly adapted to M2M and IoT application scenarios.

One of the main targets of this standard is to ensure transmission ranges up to 1 km and data rates per user larger than100 kbps. The standard is currently being drafted, but some essential details are already available. The MAC layer, for example, will include a power saving mechanism and an enhanced approach to perform channel access, which will allow the access point to support thousands of stations. The technology is very promising and can be deployed in many use cases like sensor networks and smart grid applications. Additional use cases considered by IEEE 802.11ah are shown in Fig.1.

In the following we will discuss the main challenges and requirements of M2M and IoT applications. Then, we study the IEEE 802.11ah features and show how this technology can be used efficiently to satisfy the above challenges and requirements.

M2M and IoT requirements

Wireless M2M technologies enable a wide range of important applications [1], including smart metering, healthcare monitoring, fleet management and many others. In these applications, a large number of M2M devices are expected to be connected to the Internet or to each other. The primary challenge in order to make it happen promptly is clearly to adjust the existing technology and architecture to handle the massive numbers of communicating entities. Additionally, it is deemed essential for the network operators to be able to offer M2M services and devices at lower cost levels while serving relatively larger areas. Furthermore, an M2M device (sensor, actuator, or smart meter) has to operate, in many scenarios, without battery replacement/recharge up to many years. Energy efficiency is thus becoming a paramount concern when designing an M2M network.

To address these challenges, the emerging IEEE 802.11ah specifications are proposing a number of improvements and new features.

Why IEEE 802.11ah technology?

Recently, a new amendment, IEEE 802.11ah [2] has been developed with the aim to fulfill the stringent M2M and IoT requirements, while at the same time not significantly degrading user experience when coexisting with older IEEE 802.11 releases at sub-1-GHz. The development of this emerging technology is still at its early stages and the respective standardization committee is currently in the process of collecting system design proposals. The complete standard is expected to be finalized by the year 2014. Meanwhile, the motivating goal is to enhance the design of the PHY and MAC layers of the state-of-the-art IEEE 802.11ac [3] technology such that it could efficiently operate at the unlicensed sub-1-GHz bands.

Due to lower center frequencies, the lower path loss at sub-1-GHz provides longer distances when compared to typical WLAN frequencies around 2.4 GHz and 5 GHz. Also, the power consumption of the devices can be pushed down at these frequencies, because of the propagation properties and simpler needed device components. Therefore, the expected low-cost and large coverage make 802.11ah radio technology highly attractive for deployment in rural areas compared to WiMAX technology. The main targets to be achieved by an IEEE 802.11ah amendment can be summarized on the following points:

- Transmission range up to 1 km.
- Data rates > 100 kbps.
- Maintaining the 802.11 WLAN user experience for fixed, outdoor, point-to-multi-point applications.
- Maximum number of stations (STAs)s to be served around 6000.
- Better energy efficiency than existing proprietary solutions like ZigBee

The PHY layer of IEEE 802.11ah is using an OFDM-based waveform consisting of a total of 64 tones/sub-carriers spaced by 31.25 kHz. The modulations supported include BPSK, QPSK and 16/64/256 QAM. It will support multi-user MIMO and single-user beam forming. The STAs will support the reception of 1 MHz and 2 MHz PHY transmissions modes. The channelization (i.e. operating frequency) depends on the region. In Europe, for example, it will be within 863-868 MHz, allowing either five 1 MHz channels or two 2 MHz channels [2].

The baseline of the MAC layer in IEEE 802.11ah is using the conventional contention scheme based on carrier sense multiple access with collision avoidance mechanism (CSMA/CA) which is the basic access mechanism in WLAN systems. Some improvements of the MAC layer are also expected. Improvement of the IEEE 802.11ah MAC will allow longer sleeping time. In IEEE 802.11ah, as we target higher energy efficient applications, the time that STA can take in sleeping mode is more flexible and can be relatively long, up to many days. Enhancements to the power save poll (PS-Poll) scheme are investigated. Additionally, new Traffic Indication Map (TIM) coding to support a large number of devices is considered. Furthermore, a grouping scheme will be used, where a limited set of STAs will be allowed to contend at the same time, hence reducing the collision probability. Additional MAC features are also being considered as the specification is evolving.

IEEE 802.11ah feasibility study

In the following we show the performance of the IEEE 802.11ah technology and investigate how it will impact the M2M deployments by studying the achievable data rate, energy efficiency and device population. Further details can be found in [4].

In Fig. 2 we show the maximum achievable data rate for variable range in outdoor channels and 4096 bytes packet size case for variable link reliability when BPSK and 2 MHz mode are used. As can be seen the target data rate of 100 kbps can be served at a range of 1 km for a link reliability of 60%.

In Fig. 3 we show the Energy consumption and time distribution of STA power states for 256 bits and 1000 bits packet size cases with a traffic model of a mean message inter-arrival time of 30s with exponential distribution (the PHY data rate of 0.7 Mbps and collision probability pc = 0.15). The PER is assumed to be 10%. The power values for different states are selected as follows: Pidle=1.35mW, Ptx=2.55mW and Prx=1.5mW. Interestingly, the energy efficiency of IEEE 802.11ah can reach the values of around 700 and 200 Kbit/J for 1000- and 256-bit messages, respectively. Additional simulations have been conducted to compare the energy consumption and achievable throughput with ZigBee application for sensor use cases confirming that IEEE 802.11ah has better performance than ZigBee from both perspectives.

Conclusions

IEEE 802.11ah technology is clearly a step forward in supporting ubiquitous M2M and IoT applications, however, there is still some work to be done before the targeted performance requirements can be effectively satisfied.

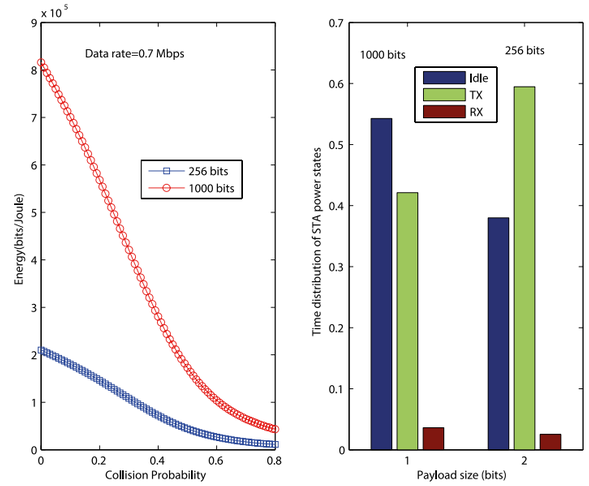


Figure 2. *IEEE 802.11ah maximum achievable data rate for variable range in outdoor channels and 4096 bytes packet size case for a different link reliability.*

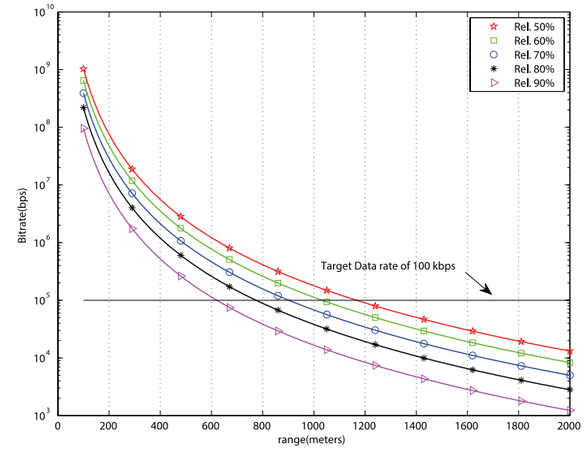


Figure 3. *Energy consumption and time distribution of STA power states for 256 bits and 1000 bits packet size cases with a traffic model of a mean message inter-arrival time of 30s.*

References

[1] H. Cho, "Machine to Machine (M2M) Communications Technical Report", IEEE 802.16p-10/0005, 2010.

[2] IEEE P802.11ah, "Specification framework for TGah", doc.: IEEE 802.11-11/1137r13, January 2013.

[3] IEEE P802.11ac, "Specification framework for TGac", IEEE 802.11-09/0992r21, 2011.

[4] A. Hazmi, J. Rinne and M. Valkama, "Feasibility Study of IEEE 802.11ah Radio Technology for IoT and M2M use Cases", IEEE Globecom Workshop, pp.1687-1692, 3-7 Dec. 2

Ali Hazmi , Mikko Valkama
Department of Electronics and Communications Engineering Tampere University of Technology

Juho Pirskanen
Renesas Mobile Europe Ltd

Consortium:

MOBISOFT

Lahden 4G-Service Oy

FRUCT

MPY

UNIVERSITY OF HELSINKI

CYBERCUBE

Laturi

FINNET

COMPONENTALITY

refecor

RENESAS
Renesas Mobile Corporation

TeliaSonera

F-Secure

NOKIA

MATTERSOFT

POLAR

ERICSSON

FINWE

FINNPARK

metso

Arch·Red

JYVÄSKYLÄN YLIOPISTO
UNIVERSITY OF JYVÄSKYLÄ

there.

Aalto University

TAMPERE UNIVERSITY OF TECHNOLOGY

LAUREA

UNIVERSITY
OF TAMPERE

intel

VTT

UNIVERSITY of OULU
OULUN YLIOPISTO

www.iot.fi