Why TCP/IP for Smart Object Networks

Internet of Things Architecture?

- *A plethora of proprietary or semi-closed protocol stacks advocating for collapsing layers with no clear demarcation between the various functions handled by the network protocols*
- *The main motivation for such an approach was to improve the efficiency of the networks, considering the high degree of constraints*

Rigid Approach

- By collapsing the layers, these architectures proved to be extremely rigid in the following ways:
  - **Link layer dependency:** Architectures were tied to a specific link layer. The emergence of a number of new low-power link layers. These architectures could not support the new links without performing protocol translation, which is a very costly and inefficient approach
  - **Dependency between the various networking functions of the networking stack:** With TCP/IP new applications are developed on a daily basis without having to change the transport or IP layers. If functions are collapsed into a single core component, this creates a dependency that dramatically slows down the support of additional functionalities.

Initial Goals of TCP/IP

- Internet communication must continue despite loss of networks or gateways (in the presence of link or node failures)
- The Internet must support multiple types of communication services
- Internet architecture must:
  - Accommodate a variety of networks ("networks" means link and physical layers)
  - Permit distributed management of its resources
  - Be cost-effective
  - Permit host attachment with a little effort
- Resources used in the Internet architecture must be accountable.
- The main goals for smart object networks are the same list as outlined above. The additional requirement is the support of large-scale networks made of billions of unattended and constrained devices for which new IP technologies have been developed.

Smart Object Networks Features

- *All Smart Object Networks do not have to connect to the public Internet. Depending upon the requirement, some Smart Object Networks need to connect to the Internet*
- *A brief recap of challenges, which can serve as a guideline for the design of Smart Object Networks Architecture*

Evolvability

- Although we have an idea of where the application space of smart objects is heading, we cannot know what direction it will take in the future. Therefore, smart object technology must inherently support the notion of evolvability.
- The mechanisms developed for smart objects should not be constrained by today's ideas, but must allow for the next generation of applications to take full advantage of the technology in pursuing its own application goals.

Scale

- *Smart object networks have a large number of nodes per system. Existing smart object systems have thousands of nodes, and they are likely to develop into systems composed of hundreds of thousands or even millions of nodes.*
- *Thus, smart object architecture must support an increasing number of nodes through its addressing, routing, and management mechanisms.*

Diversity of Applications

- The number of applications for smart objects is large, and so is the number of differences in each application
- A home automation application does not share all of the properties of an industrial automation application
- Smart object technology tailored to one specific application therefore may not work for other applications

Diversity of Communication Technologies

- Depending on the application and the environment in which the system is deployed, smart objects can use a wide range of communication technologies.
- Wireless communication is appropriate in many situations because of its deployment convenience, whereas wired communication is more suitable in other places.
- Many smart object systems use combinations of disparate technologies in the same deployment.

Interoperability

- Smart object networks need interoperability between the smart object devices and between the smart objects and existing network infrastructures
- With the large base of existing systems that smart objects enhance, a smart object architecture that makes interoperability and interconnection difficult or cumbersome will not prevail

Standardization

- Mechanisms and protocols that define the operation of smart objects must be standardized using open standards through well-established standardization practices
- Any patents covering the standardized technology must be disclosed and made available to be used by third parties. Open standards make the entry barrier low for manufacturers, and allow them to freely choose between different vendors
- Open standards was a key to the success of Internet

Potentially lossy communication technology

- Many of the communication technologies used for smart objects are inherently lossy (data sent are not guaranteed to reach their destinations)
- Smart object protocols and mechanisms need to take this into account when determining where and how to send data as well as determining when and how often

Lifetime

- Because of the large-scale installations and demanding applications for smart objects, smart object networks are meant to remain functional for many years
- This lifetime has implications both for the performance requirements of smart object mechanisms, which must be power-efficient, and for the mechanisms as such, which must remain operational over the lifetime of the system

Low Power Consumption

- Smart objects have severe power constraints
- Many smart objects are powered by batteries that cannot easily be replaced or recharged
- Other smart objects draw their energy from their surroundings, such as vibration or electromagnetic energy
- Power consumption must be low for the system to achieve its optimal lifetime
- Power requirement affects both the network protocols and the construction of nodes. The memory size and computational
- complexity of the nodes are limited by the power consumption constraints.

Low Cost

- Smart objects are deployed in large numbers; therefore a small reduction in per-device costs quickly translates into large savings in the cost of the entire system
- Just as the power consumption constraints affect the memory size and computational complexity of the nodes, so do cost constraints
- Because of constrained resources such as memory, power, and computation, any smart object architecture must be lightweight

TCP/IP Architecture for Smart Object Networks

- *Given these challenges, we now investigate the IP architecture to find out how well it meets them and their implications.*

Interoperability

- *Interoperability is a predominant characteristic of the IP architecture*
- *A single IP network operates across a variety of underlying media such as Ethernet or WiFi. Within the IP architecture, an IP network operates across both wired and wireless link layers without requiring any external mechanisms or add-ons*

- *Smart object networks extend from low-power wireless nodes to high-power data coordination servers*
- *A low-power wireless node typically runs a low-power, low-data-rate radio link layer, whereas the high-power data coordination server runs over a wired, high-speed Ethernet network. Still, these systems need to communicate with each other*
- *Because of its layered architecture, IP provides interoperability between these devices without any special servers, gateways, or custom software that connects the systems. IP naturally connects these two*

- *An IP-enabled device can interoperate with a large number of devices, computers, and servers. IP is not only the standard protocol that defines the Internet, it is also the de facto standard protocol used for networking computers outside the Internet*
- *IP-based smart objects are able to communicate with any given device without any additional hardware or software*

- *IP is available in most, if not all operating systems such as Microsoft Windows and Linux or microcontroller operating systems such as Contiki, TinyOS, and FreeRTOS*
- *Most software packages also provide the necessary device drivers for the underlying communication hardware*

- *IP-enabled smart objects interoperate with other systems and devices that run IP, but the IP architecture contains other protocols as well. The IP suite contains a set of protocols running on top of IP that include the transport protocols UDP and TCP; application layer protocols such as the Hypertext Transfer Protocol (HTTP), for web-style interaction and web service infrastructure; and the Simple Network Management Protocol (SNMP) for network configuration*
- *Thus a smart object that runs IP is able to interoperate with a large number of external systems*