Managing Things in an Ambient Space

Sujith Samuel Mathew[1]
Supervised by: Yacine Atif[2], Quan Z. Sheng[1], and Zakaria Maamar[3]

[1] School of Computer Science, University of Adelaide, Adelaide, Australia
[2] Faculty of IT, UAE University, Al Ain, UAE
[3] Zayed University, Dubai, UAE
{sujith,quanzheng.sheng}@adelaide.edu.au,
yacine.atif@uaeu.ac.ae, Zakaria.Maamar@zu.ac.ae

Abstract. We are surrounded by inanimate things that have inherent information. Unfortunately, the vast majority of applications built to use this information are built in ad-hoc manner, introducing issues with maintainability, share-ability and reusability. We discuss the architecture of Ambient Space Manager (ASM), a system to explore and control things within a context-aware ambient space. We define the context variables of such things to be the Capability, Location, Operations and QoS. Here we also elaborate the Capability context based on the atomic capabilities of things that enable them to offer contextual services.

Keywords: Web of Things, Context-aware Ambient Space, Future Internet.

1 Introduction

"*By 2015, wirelessly networked sensors in everything we own will form a new Web. But it will only be of value if the terabyte torrent of data it generates can be collected, analyzed and interpreted*" [1]. With billions of things around us, people find themselves in ambient environments (i.e., environments that provide seamless communication between people and things). Ambient applications make the management of large infrastructures like modern university campuses and multinational organizations easier to unleash the potential of their resources. Unfortunately, the vast majority of these applications are built in ad-hoc manners, introducing issues with maintainability, share-ability and reusability. Advances in Web services, Cloud computing, wireless networks and identification technologies, make processing power and communication capabilities available in increasingly smaller packages. Indeed, the Internet is evolving into the so-called "Web of Things" (WoT) or "Cloud of Things", an environment where everyday objects such as buildings, sidewalks, traffic lights, and commodities are identifiable, readable, recognizable, addressable, and even controllable via the Web [2,3,4].

With a plethora of things becoming ubiquitous on the Internet, there is a need to model scenarios for the WoT, and a plan to handle them in ambient environments. There are quite a few challenges in building an integrated system of heterogeneous

G. Pallis et al. (Eds.): ICSOC 2011, LNCS 7221, pp. 226–232, 2012.

physical things on the Web. Though the Semantic Web could meaningfully represent resources of physical things, there is a lack of standard interfaces to access these things by Web applications which use thing's capabilities in a given context. In addition, the heterogeneity of ubiquitous computing systems poses a major problem for system architects with respect to many protocols, component architecture, and data formats. There are no clear specifications of common characteristics of thing's attributes and processes, for controlling them or querying them over the Web. An architecture definition is required for managing things and we propose a system architecture, to manage the access and control of things in an ambient space.

2 Related Works

A contributing factor to the fast and widespread growth of the Internet is the increasing dependence on the Internet as an economical and efficient means of communication. The increasing availability of Internet access points and enhanced infrastructures of whole cities to support wired and wireless Internet connection are fueling this trend [5]. Bodies such as the IP for Smart Objects alliance (IPSO) [6] and the European Future Internet Initiative (EFII) [7] have also accelerated this trend to connect a variety of physical things into the Internet, with the intention of propagating and managing the wide use of Internet as the common medium for communication. Guinard and Trifa [8] successfully demonstrated Web mashups by exposing real world things as RESTful Web services. Their research compares two ways of interfacing real-world devices into the Web by having Web servers embedded in devices and secondly by connecting devices to an external proxy Web server, as a gateway. A recent product launched by OpenPicus, the FlyPort [9] hosts a compact Wi-Fi module which is 35X488 mm. The FlyPort supports Web server and socket applications. With digital inputs and output ports, this miniature Wi-Fi module can be augmented to any device to Web enable them. Michael Beigl et al. [10] define smart physical things as things augmented with computing and communication capabilities, which can be accessed by computer applications. Similarly, Friedemann [11] envisions smart things to be able to wirelessly communicate with people and other smart things, with the ability to understand the presence of surrounding objects. Today, these definitions do not formally encompass all things that could be on the Web, e.g., an RFID tagged chair or a Personal Digital Assistant (PDA), both can be accessed on the Internet.

3 System Architecture for Managing an Ambient Space

We propose the architecture of Ambient Space Manager (ASM), designed to manage the access and control of things in an ambient space. ASM provides a gateway to things on the Web for building ubiquitous applications. This architecture creates a support system for ambient spaces where ubiquitous things are seamlessly accessed.

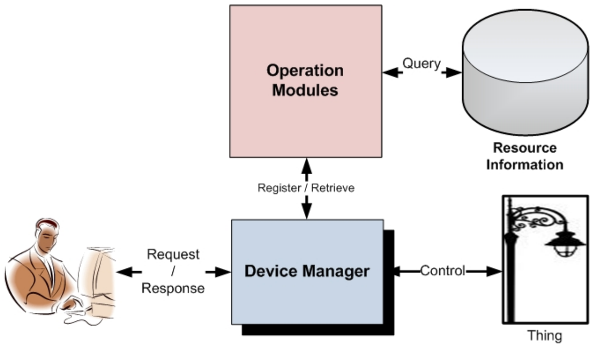

Fig. 1. Modules of the Ambient Space Manager (ASM)

To establish a separation of concerns in dealing with heterogeneous things in an ambient space, a layered architectural approach is required to provide modularity and integration patterns based on which ambient client applications are developed. As shown in Figure 1, the Device Manager interfaces users/agents with physical things in the ambient space whereas Operation Modules register and retrieve relevant information to facilitate the interaction with physical things. All relevant things within an ambient space are registered with ASM. Further, the Resource Information module encapsulates the knowledge base of things and related context information.

3.1 Web-Enabling Things

A thing becomes Internet-enabled if it is associated with networking capability (i.e. has an IP address), which uniquely identifies it within the Ambient Space (Figure 2a). A thing becomes Web-enabled when it is augmented with a Web server (Figure 2b) so that it can expose its functional and non-functional capabilities on the Web through HTTP. Though arguably, there is scope for WS* and REST in the area of Web services, advances in REST based Web service architectures is propagating the abstraction of physical things as services on the Web. This trend gives rise to the

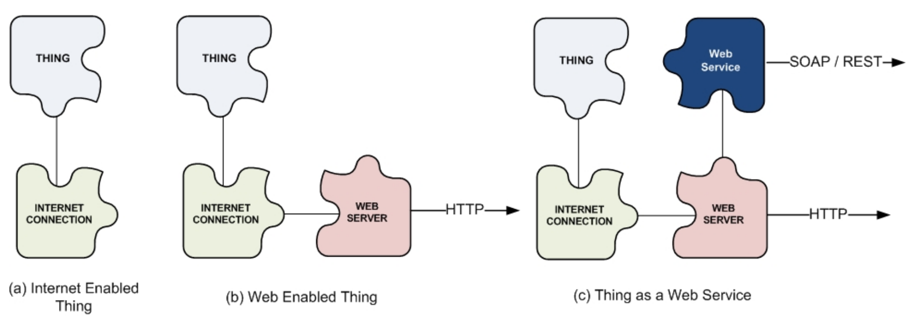

Fig. 2. Connecting things to the Web

possibilities of wrapping things in the physical world as Web services (Figure 2c). The FlyPort module [10] is a Wi-Fi module, hosting a tiny Web server and multiple power output points. This configuration enables it to be networked and controlled on the Web. The output points are connected to a thing, say a light as shown in Figure 3 and then controlled from the Web. The Web-enabled Smart light is registered with ASM, redefining its context within the ambient space.

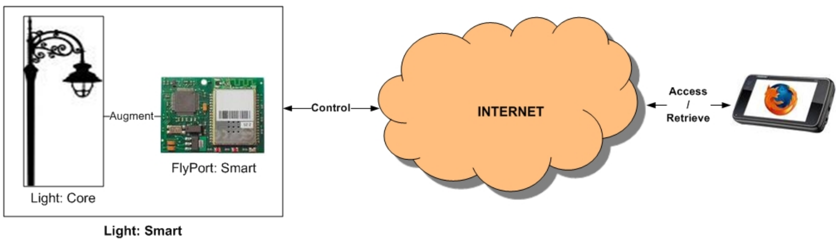

Fig. 3. Light augmented to be accessed on the Web

3.2 Context of Things in an Ambient Space

To address the challenge of managing things in an Ambient space, we define the context of things within an ambient space by its Capabilities, Location, Operations, and Quality of Service (QoS) labeled thereof as CLOQ. An initial configuration of ASM requires CLOQ context to be registered for all things within an ambient space. To represent the commonalities of things, a formal ontology is a necessity [3] and hence we also elaborate the ontological structure based on the capabilities of things. As presented in our earlier work [12], the Capability dimension is a formal ontology of thing's capabilities to support the design of systems in ambient spaces. This ontology proposal is specific to the application context of the proposed ASM. Further extension of this work aims at developing a bridging scheme to share common things specifications on WoT.

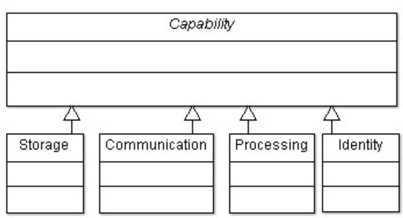

Fig. 4. UML: The abstract Capability class and IPCS sub-classes

Things on the Web are classified using four fundamental, atomic dimensions that characterize their intrinsic capabilities i.e. Identity, Processing, Communication and Storage referred to as the IPCS set, as shown in Figure 4. ***Identity (I)***: A thing must be uniquely identifiable with the use of an identification system. A thing could adopt multiple identification systems (e.g., Bluetooth address and IP address). Identity is the mandatory and minimal requirement for things to be managed by ASM. ***Processing (P)***: The processing capability of a thing is a system that has functions which allow a thing to be controlled or managed. ***Communication (C)***: The communication interface is a system enabling interaction with the thing describing the input, output, or both. ***Storage (S)***: Storage is a system that describes the information that a thing retains abstracting the states and values.

Based on the IPCS capabilities, we define the taxonomy of things on the Web. A ***Core*** thing has the bare capability of being uniquely identified within a given context. A ***Primitive*** thing has a unique identity and includes one additional capability of the other three dimensions. These are further categorized as the Fat, Plug and Fuzzy. A ***Complex*** thing has a unique identity and combines two of the other three dimensions. A ***Smart*** thing combines all the capabilities of the IPCS set. The combinations are shown in Table 1 where, *Identity* is not included as it is mandatory for all categories.

Table 1. Classification of Primitive Things and Complex Things

Primitive Things				
Names	P	C	S	Examples
Fuzzy	X			Washing machines, Microwave ovens, etc.
Plug		X		Headphones, Speakers, etc.
Fat			X	CD, DVD, etc.

Complex Things				
Names	P	C	S	Examples
Social	X	X		Remote controls, land-line phones, etc.
Sticky		X	X	USB stick, RFID tag, etc.
Gizmo	X		X	Calculator, Game Boy, etc.

3.3 Implementation Framework

Interaction with ASM is through the Device Manager (Figure 1). The user searches from things registered with ASM, based on CLOQ context. The Device Manager sends requests to the Operation modules which initiate the discovery of the requested thing. The search is based on the capabilities of things in the knowledge base within the Resource module. The knowledge base is an ontological classification of things based on IPCS capabilities. Querying the ontology is done using W3C recommended SPARQL language for RDF. The search potential of SPARQL is appropriate for querying [14] the knowledge base. The logic for dynamically querying the knowledge base is implemented using Jena APIs [13], an open source framework for building Semantic Web applications. Jena is Java based and provides an environment to programmatically access the ontology. The knowledge base is queried with Jena APIs, like `executeSPARQLQuery()`, that takes a SPARQL query as a parameter and runs it against the knowledge base (example shown below).

```
JenaOWLModel owlModel =
             ProtegeOWL.createJenaOWLModelFromReader(input);
String queryStr = "SELECT ?primary
             WHERE ?primary rdfs:subClassOf WOT:Primary";
QueryResults results =
             owlModel.executeSPARQLQuery(queryStr);
```

The query is parsed and the result is processed by the Operation Layer. From the list of things with the requested capabilities, their Location, Operations and QoS are filtered based on the request. The information is then made available, through the Device Manager over the Web to the external system.

4 Conclusions

The proposed architecture provides a modular approach to manage context-related information of things on the Web. The proposed contributions help to realize ambient spaces where real-world things seamlessly interact with each other to provide new services. We continue our work in further refining the context variables of Capability, Location, Operations and QoS. These variables form the building blocks of ASM to encapsulate the functions of everyday things as services on the Web. Hence a comparative study of existing Web service solutions is forthcoming. Considering the fact that things can be dynamic in space and time, managing the presence of things, composition of things and the significance of security and privacy is an interesting area of study for the future.

References

1. Raskino, M., Fenn, J., Linden, A.: Gartner, http://www.gartner.com/DisplayDocument?id=476440 (retrieved on May 14, 2011)
2. Mulligan, G.: The internet of things: Here now and coming soon. IEEE Internet Computing 14(1), 35–36 (2010)
3. Raggett, D.: The Web of Things: Extending the Web into the Real World. In: van Leeuwen, J., Muscholl, A., Peleg, D., Pokorný, J., Rumpe, B. (eds.) SOFSEM 2010. LNCS, vol. 5901, pp. 96–107. Springer, Heidelberg (2010)
4. Berners-Lee, T.: The web of things. special theme on the future web. ERCIM News – the European Research Consortium for Informatics and Mathematics (2008), http://ercim-news.ercim.eu/en72/keynote (retrieved on March 12, 2011)
5. Danigelis, A.: 10 cities with widespread wireless, http://dsc.discovery.com/technology/tech-10/wireless-cities-top.html (retrieved on March 12, 2011)
6. IPSO Alliance. Promoting the use of ip in networks of smart objects, http://www.ipso-alliance.org/Documents/IPSO_Briefing.pdf (retrieved on March 13, 2011)

7. The European Future Internet Initiative. White paper on the future internet PPP definition, http://www.futureinternet.eu/fileadmin/documents/reports/Future_Internet_2020-Visionary_Panel.pdf (retrieved on March 12, 2011)
8. Guinard, D., Trifa, V.: Towards the web of things:Web mashups for embedded devices. In: MEM 2009 in Proceedings of WWW 2009. ACM (2009)
9. Openpicus, flyport module, http://www.openpicus.com/cms/ (retrieved on March 12, 2011)
10. Beigl, M., Gellersen, H.W., Schmidt, A.: Mediacups: experience with design and use of computer-augmented everyday artifacts. Computer Networks 35(4), 401–409 (2001)
11. Mattern, F.: From smart devices to smart everyday objects. In: Proceedings of Smart Objects Conference (January 2003)
12. Mathew, S.S., Atif, Y., Sheng, Q.Z., Maamar, Z.: Ambient things on the web. Journal of Ubiquitous Systems and Pervasive Networks (JUSPN) 1(1), 1–8 (2010)
13. Jena - a semantic web framework for java, http://jena.sourceforge.net/ (retrieved on May 14, 2011)
14. Sirin, E., Parsia, B.: SPARQL-DL: SPARQL query for OWL-DL. In: 3rd OWL Experiences and Directions Workshop (2007)