Towards Linking the Digital and Real World with OpenThingMap

Damian Lasnia [a,1], Theodor Foerster [a], Arne Bröring [a,b,c]
[a] *Institute for Geoinformatics, University of Muenster, Germany*
[b] *52°North, Münster, Germany*
[c] *ITC Faculty, University of Twente, Netherlands*

Abstract Currently digital inventories of outdoor features evolve such as OpenStreetMap. A digital inventory of real-world things is still missing, but is required to enable full search and interaction in the digital world. This article presents an approach to link any real world object to a georeferenced digital representation, thus suggests a concept for location within the areas of Web of Things, Ubiquitous Computing and Ambient Technology. The approach is twofold based on a) OpenFloorMap as an inventory for buildings and b) OpenThingMap as an inventory for things. The inventories are based on a lightweight data model and are populated through the knowledge of the crowd using advanced mobile devices.

Keywords: Crowdsourcing, Spatial Inventories, Mobile Applications.

Introduction

Building inventories of real-world features in databases can be realized by Crowdsourcing. Projects such as OpenStreetMap [1] use mobile devices which are able to measure the location context (i.e. GPS), the knowledge of the crowd and a lightweight data model to build an inventory of outdoor features. This supports outdoor wayfinding and search for locations, represented as POIs. So, every real-world object outside is represented in the digital world, which is described through tags. However, as most of our daily life is spent inside, an approach to map the inside of buildings as well as things residing within the building, has not been proposed yet.

Therefore, this article a) describes the OpenFloorMap for modeling buildings and b) proposes OpenThingMap as a way to link any entity in the real-world to the digital world. Thereby, the interaction between the things in real-world and the digital world will be seamlessly.

The presented approach is based on smartphone technology, which provides a set of sensors to measure the real-world in such a way, that digital representations of things can be created easily. The approach is based on light-weight data models and is based on the Web of Things.

Section 2 presents OpenStreetMap and Web of Things as the main building blocks of the presented approach. Section 3 describes the OpenFloorMap and OpenThingMap, respectively. Section 4 presents a conclusion and provides an outlook for future work.

---

[1] Corresponding Author.

1. Related work

1.1. OpenStreetMap

The OpenStreetMap project [1] was found in 2004 and has a rapidly growing data inventory of streets and POIs. Its access and use is free of charge (Creative Commons license). The data is reported by so-called mappers, who use digitizing tools or mobile devices with GPS capabilities to capture the data. OpenStreetMap is used for several applications such as mapping or routing. The data model is based on three types of objects, nodes, ways and closed ways (to represent polygons). The thematic data is captured through tags (key value pairs). The set of available tags is thereby unlimited. Specific renderers use a set of designated tags to create cartographic representations.

1.2. Web of Things

The Web of Things [2] evolved from the Internet of Things ([3]) and integrates real-world "things" with the Web. Examples for such things are household appliances, embedded and mobile devices, but also smart sensing devices. Often, the user interaction takes place through a cell phone acting as the mediator within the triangle of human, thing, and Web. Applications of the Web of Things are influenced by the idea of ubiquitous computing [4] and range from smart shoes posting your running performance online, over management of logistics (e.g., localization of goods in the production chain), to insurance (e.g., car insurance costs based on the actually driven kilometres).

The Web of Things leverages existing Web protocols as a common language for real objects to interact with each other. HTTP is used as an application protocol rather than a transport protocol as it is generally the case in web service infrastructures such as OGC's SWE framework [5]. Things are addressed by URLs and their functionality is accessed through well-defined HTTP operations (GET, POST, PUT, etc.). Hence, Web of Things applications follow the REST paradigm [6]. Specific frameworks (e.g. [7, 8]) offer REST APIs to enable access to things and their properties as resources. These REST APIs may not only be used to interact with a thing via the Web, also website representations of things may be provided to display dynamically generated visualizations of data gathered by the thing. Then, the mash-up paradigm and tools from the Web 2.0 realm can be applied to easily build new applications. An example application may use Twitter to announce the status of a washing machine or may let a fridge post to an Atom feed to declare which groceries are about to run out.

2. Approach

The proposed approach establishes an ubiquitous, tight coupling between our real life and web content that is connected with things in our environment. This is based on two aspects – OpenFloorMap and OpenThingMap.

2.1. OpenFloorMap

The OpenFloorMap is based on the need for a location referencing model for all human accessible spaces, and because current flagship smartphones are capable to measure room extent up to an applicable degree of accuracy. To enable easy data capturing and data management, the data model of OpenFloorMap (Figure 1) applies the simplicity of the OpenStreetMap data model (Section 1.1). In particular, the data model of the OpenFloorMap consists of two-dimensional levels and three-dimensional rooms inside those. A set of levels, a set of geographic coordinates and a unique identifier are the representation of a building in the real world. Each building is associated with a set of POIs in OpenStreetMap, which represents the entries to the building.

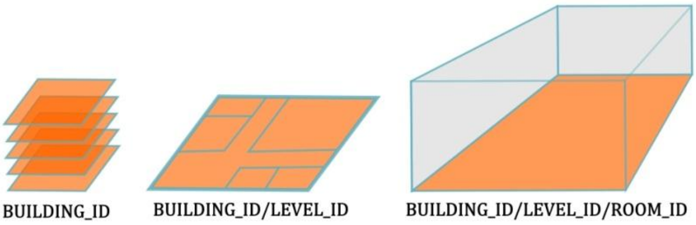

Figure 1: Data model of OpenFloorMap.

The OpenFloorMap is based on an Android application (Figure 2). The user can capture the layout of the rooms and report the data to the server. Modern smartphones provide APIs to access their build-in sensors that are capable to provide parameters for a system of equations based on trigonometric functions to determine room extend: Proximity sensors measure the distance between the camera and room corners, gyroscopes and orientation sensors determine the device’s attitude. An assisted user interface adds building and level information. In a browser-based application, reported rooms can be arranged via drag-and-drop to represent the floor’s actual layout.

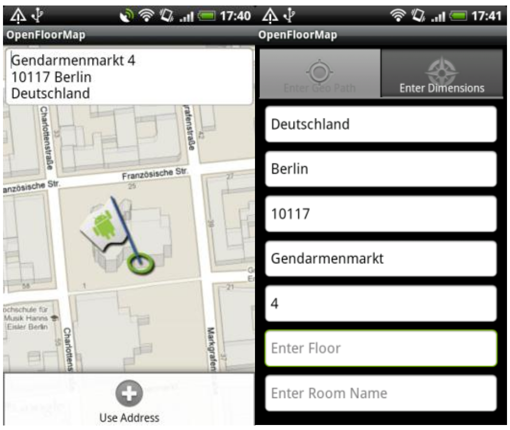

Figure 2: Android application user interfaces

2.2. OpenThingMap

The OpenThingMap integrates web content into our every day`s environment. It is based on the OpenFloorMap (Section 3.1). Persons are able to access the OpenThingMap via gateways and explore the digital world in terms of an in-world knowledge browser. This means that the user is able to set the gateway's position to any human accessible space inside buildings (http://ofm.url/buildings/<building_id>/levels/<level_id>/rooms/<room_id>) or the outside-world (Latitude, Longitude) to gather a catalogue of surrounding things and their linked content in the web. Photos may link to social network profiles, the microwave to its manual and the air conditioner to its remote control web service. Things that expose their capabilities in a standardized format become localizable through spatial queries as physical entity, no more just as URIs. This enablement of spatial queries within Mashups of things [2] is a great practical achievement of the OpenFloorMap integration towards personalization ambient spaces within universal environments.

We propose that things act as gateway between the real and the digital world and thereby link them (Figure 3). A gateway is the most specified class within OpenThingMap. It does not only represent itself, but the user as well. Smart things are generalized gateways with no user log in, but differ from things because they host embedded or attached computer components to provide direct connectivity with the web. The generalization of a smart thing class is the thing class. The thing class can hold everything in the real world that can be referenced either with geographic coordinates or a room in OpenFloorMap with local room coordinates. Things can be linked with static web resources e.g. for descriptive purposes and with web feeds to announce a thing's status.

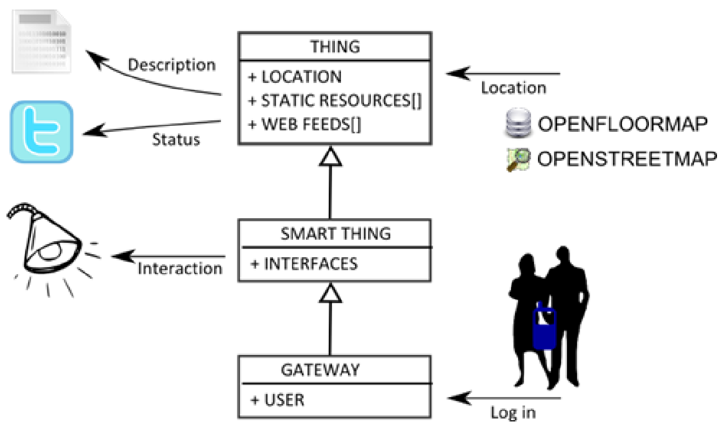

Figure 3: Concept of OpenThingMap.

3. Conclusion

This article presents an approach how to incorporate real-world things into the digital world for search and interaction. The approach is twofold using a) the OpenFloorMap and b) the OpenThingMap. Both concepts are based on a light-weight data model, the knowledge of the crowd and the capabilities of current smartphones.

The presented approach on light-weight data models and protocols is a complementary to the existing semantic approaches for annotating data [9]. Future research questions should address machine readable search and bind capabilities in OpenThingMap and the inclusion of other digital sources which are not modeled as things, such as noise and air quality (phenomena).

References

[1] M.M. Haklay and P. Weber. OpenStreetMap: user-generated street maps. *IEEE Pervasive Computing*, pages 12–18, 2008.

[2] D. Guinard and V. Trifa. Towards the Web of Things: Web Mashups for Embedded Devices. In *International World Wide Web Conference*, Madrid, Spain, 2009.

[3] N. Gershenfeld, R. Krikorian, and D. Cohen. The Internet of Things. *Scientific American*, 291(4):76–81, 2004.

[4] M. Weiser. The Computer for the 21st Century. *Scientific American*, 265(9):94–104, 1991.

[5] M. Botts, G. Percivall, C. Reed, and J. Davidson. OGC sensor web enablement: Overview and high level architecture. *GeoSensor Networks*, pages 175–190, 2008.

[6] R. T. Fielding and R. N. Taylor. Principled Design of the Modern Web Architecture. *ACM Transactions on Internet Technology*, 2(2):115–150, 2002.

[7] J. Pinto, R. Martins, and J.B. Sousa. Towards a REST-style Architecture for Networked Vehicles and Sensors. In *Proceedings of 8th IEEE International Conference on Pervasive Computing and Communications: WoT 2010: First International Workshop on theWeb of Things*, pages 745 –750, 2010.

[8] B. Ostermaier, F. Schlup, and K. Romer. WebPlug: A Framework for the Web of Things. In *Proceedings of 8th IEEE International Conference on Pervasive Computing and Communications: WoT 2010: First International Workshop on the Web of Things*, pages 690–695, 2010.

[9] V. Stirbu, P. Selonen, and A. Palin. The location graph: towards a symbolic location architecture for the web. In *Proceedings of the 3rd International Workshop on Location and the Web*, page 10. ACM, 2010.