A journey from the Physical Web to the Physical Semantic Web[*]

Michele Ruta
michele.ruta@poliba.it
Polytechnic University of Bari
Bari, Italy

Floriano Scioscia
floriano.scioscia@poliba.it
Polytechnic University of Bari
Bari, Italy

Giuseppe Loseto
giuseppe.loseto@poliba.it
Polytechnic University of Bari
Bari, Italy

Filippo Gramegna
filippo.gramegna@poliba.it
Polytechnic University of Bari
Bari, Italy

Saverio Ieva
saverio.ieva@poliba.it
Polytechnic University of Bari
Bari, Italy

Agnese Pinto
agnese.pinto@poliba.it
Polytechnic University of Bari
Bari, Italy

Eugenio Di Sciascio
eugenio.disciascio@poliba.it
Polytechnic University of Bari
Bari, Italy

ABSTRACT

The *Physical Semantic Web* (PSW) is a novel paradigm built upon the Google Physical Web (PW) approach and devoted to improve the quality of interactions in the Web of Things. Beacons expose semantic annotations instead of basic identifiers, *i.e.*, machine-understandable descriptions of physical resources. This enables novel ontology-based object advertisement and discovery and – in turn– advanced user-to-thing and autonomous thing-to-thing interactions.

The demo shows the evolution from the PW to the PSW in a discovery scenario set in a winery, where bottles are equipped with Bluetooth Low Energy beacons and a customer can discover them using her smartphone. The final goal is to prove benefits of PSW over basic PW, including: rich semantic-based object annotation; dynamic annotations exploiting on-board sensors; enhanced discovery and ranking of nearby objects through semantic matchmaking; availability of interactions even without working Internet infrastructure, by means of point-to-point data exchanges.

CCS CONCEPTS

• **Information systems → Web Ontology Language (OWL)**; • **Human-centered computing → Ubiquitous and mobile computing**; • **Computing methodologies → Knowledge representation and reasoning**;

KEYWORDS

Semantic Web of Things, Physical Web, Resource discovery, Non-standard reasoning

ACM Reference Format:
Michele Ruta, Floriano Scioscia, Giuseppe Loseto, Filippo Gramegna, Saverio Ieva, Agnese Pinto, and Eugenio Di Sciascio. 2018. A journey from the Physical Web to the Physical Semantic Web. In *The 2018 Web Conference Companion, April 23–27, 2018, Lyon, France.* ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/3184558.3186981

1 INTRODUCTION

The *Physical Web*[1] (PW) [5] is a framework for the Internet of Things (IoT), proposed by Google Inc. and devoted to implement direct users-to-things interactions. This paradigm is based on resource discovery enabling mobile personal devices (*e.g.*, smartphones) to retrieve Uniform Resource Locators (URLs) of nearby objects through the *Eddystone* Bluetooth beacon protocol[2]. Finding URLs via Wi-Fi through *multicast DNS* (mDNS) and *Universal Plug and Play* (uPnP) is also supported. The approach does not require directory services, as smart objects are associated to unambiguous Web addresses.

Although such an approach promotes human-to-thing communication (providing indubitable benefits to end users from a direct interaction with pervasive computing contexts), there are some significant issues penalizing a full adoption of the Google PW framework in advanced applications. Particularly: i) apart from the early Bluetooth-based discovery, further interactions require a mandatory Internet connection to the Web; ii) direct thing-to-thing communication is not allowed yet; iii) discovery mechanisms are too simplistic to cope with typical interoperability and unpredictability problems in the IoT. The *Physical Semantic Web* (PSW) [2] proposal improves upon the PW by associating semantic annotations to object beacons instead of trivial identifiers, *i.e.*, machine-understandable descriptions in standard Semantic Web languages. Therefore object advertisement and discovery are enhanced and things can communicate among them through proper software agents without user mediation.

The PSW approach preserves the original Eddystone-URL beacons discovery, so legacy applications are still supported. Furthermore, in the PSW each URL could alternatively target: i) a basic Web

[*] Produces the permission block, and copyright information

WWW' 18 Companion, April 23–27, 2018, Lyon, France

ACM ISBN 978-1-4503-5640-4/18/04.
https://doi.org/10.1145/3184558.3186981

[1] https://google.github.io/physical-web/

[2] http://github.com/google/eddystone

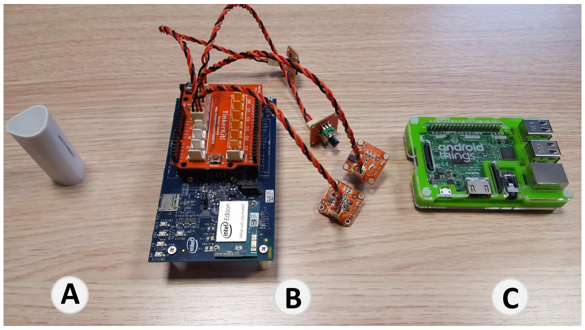


Figure 1: Demo beacons

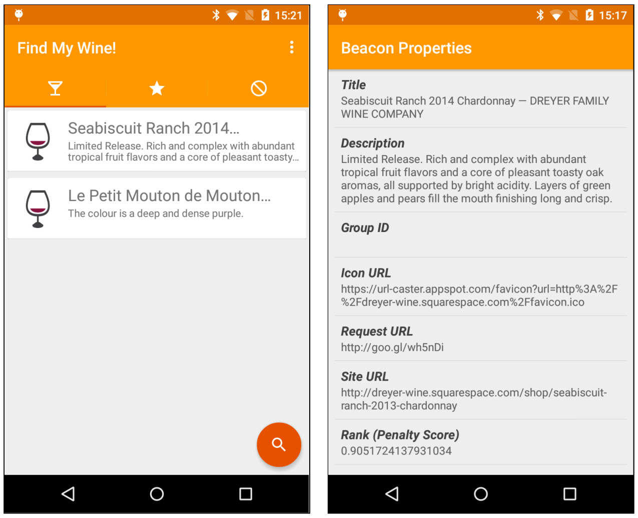
page; ii) an annotated Web page, where users may view contents and/or exploit metadata semantics; iii) a semantic annotation to be directly processed by software agents locally.

Object discovery ensues from a semantic matchmaking process [3], able to compare a request (*i.e.*, user or object profile) with several beacon annotations, referred to the same ontology. For each ⟨user/object profile, resource⟩ pair, a matchmaking score measures the relevance of the retrieved beacon w.r.t. request annotation. Concept Abduction and Contraction non-standard inferences [3] theoretically ground the PSW discovery and provide also an outcome explanation capability: it evidences compatible, incompatible and missing elements, so as to eventually refine the original query.

The proposed demo shows the evolution from the PW to the PSW in a four-step experience, as outlined in Section 2. The reference scenario is set in a winery, where bottles available for sale are equipped with Bluetooth Low Energy (BLE) beacons and a customer can discover them using a smartphone. Discussion in Section 3 highlights how relevant elements of the demo setup help understand both potentialities and benefits of the PSW proposal if compared with standard PW.

2 DEMO DESCRIPTION

The demonstration consists of a mobile application, showcasing the most significant PSW features. It can be conceptually divided in four stages, each described in the following subsections. Customers in a winery use their smartphones to discover bottles equipped with PW/PSW beacons. In each stage the user exploits progressively even more advanced functions, including the simulation of variations in environmental parameters (conditioning the discovery). For the purposes of demonstrating the app, a PSW-compatible Android smartphone[3] is used. Bottles are equipped with two standard PW beacons like the Bluvision BEEKs Lite[4] shown in Figure 1(A) and three PSW beacons (two PSW-URL and one PSW-UID), represented by two *Intel Edison*[5] boards like in Figure 1(B) and a *Raspberry Pi 3 Model B*[6] like in Figure 1(C). Generic Wi-Fi Internet connectivity and energy supply are the only requirements for the demo environment; a monitor can be used to replicate the smartphone screen for greater visibility and convenience.


15:21

Find My Wine!

Seabiscuit Ranch 2014...
Limited Release. Rich and complex with abundant tropical fruit flavors and a core of pleasant toasty...

Le Petit Mouton de Mouton...
The colour is a deep and dense purple.

(a) Beacons List

15:17

Beacon Properties

Title
Seabiscuit Ranch 2014 Chardonnay — DREYER FAMILY WINE COMPANY

Description
Limited Release. Rich and complex with abundant tropical fruit flavors and a core of pleasant toasty oak aromas, all supported by bright acidity. Layers of green apples and pears fill the mouth finishing long and crisp.

Group ID

Icon URL
https://url-caster.appspot.com/favicon?url=http%3A%2F%2Fdreyer-wine.squarespace.com%2Ffavicon.ico

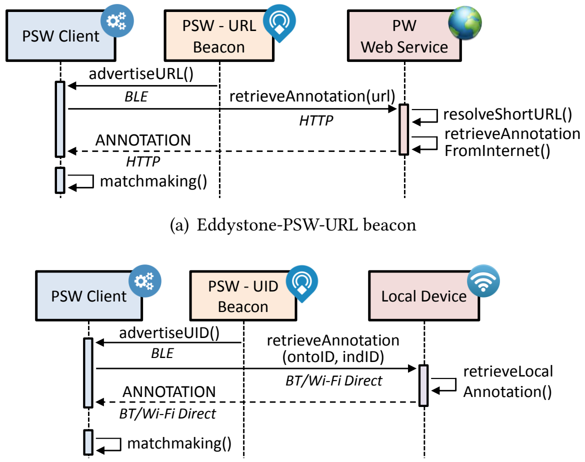
Request URL
http://goo.gl/wh5nDi

Site URL
http://dreyer-wine.squarespace.com/shop/seabiscuit-ranch-2013-chardonnay

Rank (Penalty Score)
0.9051724137931034

(b) Beacon Info

Figure 2: Stage 1 - screenshots of the Android client



(a) Eddystone-PSW-URL beacon

(b) Eddystone-PSW-UID beacon

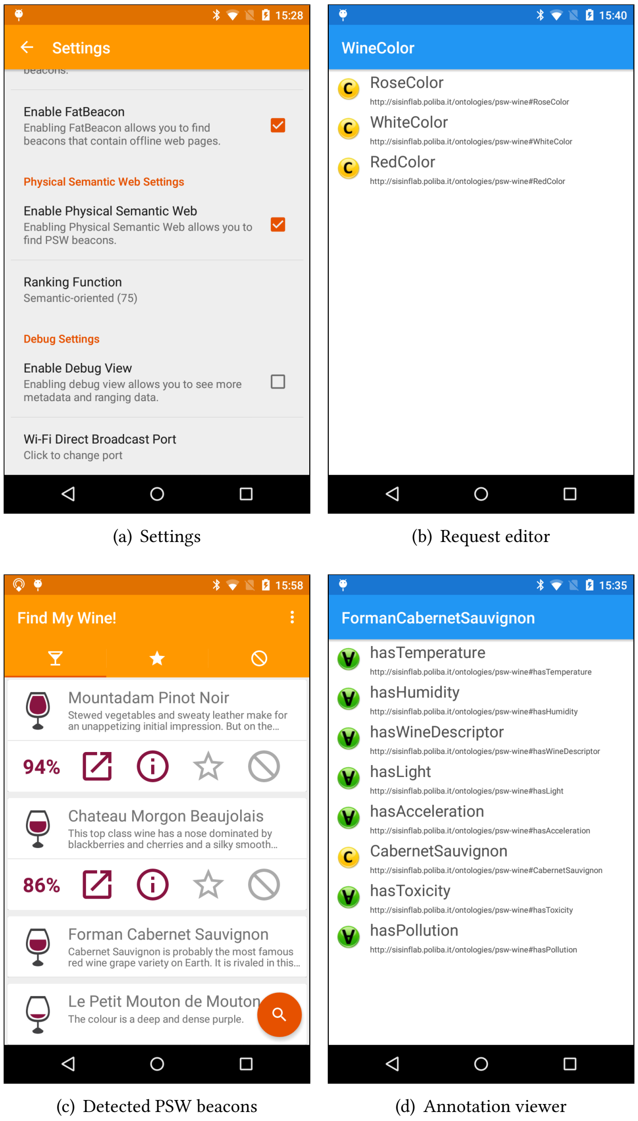
Figure 3: Discovery of PSW Eddystone beacons

2.1 Stage 1: Physical Web

Francesco is organizing an elegant dinner party. He enters a winery looking for a fine bottle. After activating Bluetooth on his smartphone, he starts the *Find My Wine!* Android application to discover the exposed beacon advertisements. Detected ads are listed in a result screen, as shown in Figure 2(a). At this stage, only two Physical Web beacons are visible in the cellar. They expose URLs to the specific wine Web pages and are ordered according to the physical distance from the user (estimated by comparing received signal strength and *transmitted power* field in Eddystone frames).

For each beacon, the app can retrieve some basic metadata (*e.g.*, title/description as in Figure 2(b)) or open the advertised Web page in the browser. All beacon information is retrieved by resolving the

[3] Required Android API level 21, corresponding to operating system version 5.0.
[4] https://shop.bluvision.com/products/beeks-lite
[5] http://software.intel.com/en-us/iot/hardware/edison/documentation
[6] https://www.raspberrypi.org/products/raspberry-pi-3-model-b/



15:28

Settings

beacons.

Enable FatBeacon ☑
Enabling FatBeacon allows you to find beacons that contain offline web pages.

Physical Semantic Web Settings

Enable Physical Semantic Web ☑
Enabling Physical Semantic Web allows you to find PSW beacons.

Ranking Function
Semantic-oriented (75)

Debug Settings

Enable Debug View □
Enabling debug view allows you to see more metadata and ranging data.

Wi-Fi Direct Broadcast Port
Click to change port

(a) Settings

15:40

WineColor

C RoseColor
http://sisinflab.poliba.it/ontologies/psw-wine#RoseColor

C WhiteColor
http://sisinflab.poliba.it/ontologies/psw-wine#WhiteColor

C RedColor
http://sisinflab.poliba.it/ontologies/psw-wine#RedColor

(b) Request editor

15:58

Find My Wine!

Mountadam Pinot Noir
Stewed vegetables and sweaty leather make for an unappetizing initial impression. But on the...

94%

Chateau Morgon Beaujolais
This top class wine has a nose dominated by blackberries and cherries and a silky smooth...

86%

Forman Cabernet Sauvignon
Cabernet Sauvignon is probably the most famous red wine grape variety on Earth. It is rivaled in this...

Le Petit Mouton de Mouton
The colour is a deep and dense purple.

(c) Detected PSW beacons

15:35

FormanCabernetSauvignon

∀ hasTemperature
http://sisinflab.poliba.it/ontologies/psw-wine#hasTemperature

∀ hasHumidity
http://sisinflab.poliba.it/ontologies/psw-wine#hasHumidity

∀ hasWineDescriptor
http://sisinflab.poliba.it/ontologies/psw-wine#hasWineDescriptor

∀ hasLight
http://sisinflab.poliba.it/ontologies/psw-wine#hasLight

∀ hasAcceleration
http://sisinflab.poliba.it/ontologies/psw-wine#hasAcceleration

C CabernetSauvignon
http://sisinflab.poliba.it/ontologies/psw-wine#CabernetSauvignon

∀ hasToxicity
http://sisinflab.poliba.it/ontologies/psw-wine#hasToxicity

∀ hasPollution
http://sisinflab.poliba.it/ontologies/psw-wine#hasPollution

(d) Annotation viewer

Figure 4: Stage 2 - screenshots of the Android client

advertised Eddystone URL with a remote *Physical Web Service*[7] via HTTP.

2.2 Stage 2: Physical Semantic Web

The user can enable the *Physical Semantic Web* option from the settings screen in Figure 4(a) to perform a more accurate and flexible resource discovery[8]. In addition to previously discovered wines, three PSW beacons (specifically, two PSW-URL and one PSW-UID) are advertised now through Intel Edison or Raspberry Pi 3 boards running the *PSW-toolkit* framework[9]. Each PSW-URL beacon exposes a URL used to download a Web Ontology Language [4] annotation from the Web (Figure 3(a)), whereas the PSW-UID beacon advertises an OWL annotation stored on the beacon itself. In the last case, as shown in Figure 3(b), the user device retrieves the resource description through a peer-to-peer connection (*e.g.*, Bluetooth) [2].

User's mobile device embeds a semantic matchmaking engine [3] allowing to rank retrieved annotations in relevance order w.r.t. a request. Requests can be either built implicitly –by means of context-aware profiler daemons running on the device– or explicitly, through *e.g.*, a graphical or speech-based user interface. The demo app provides a graphical editor of OWL requests to: i) navigate the reference taxonomy describing main wine features (*e.g.*, wine colors in Figure 4(b)); ii) show both class and object property hierarchy; iii) compose a request selecting basic OWL ontology elements. For the purposes of the demo, a simple reference ontology has been created, inspired by classical examples in [1]. OWL annotations of bottles contain both a description of the wine and conservation status information. Beacons are ranked w.r.t. the user request by means of a semantic-based utility function [2]. The adopted semantic matchmaking relies on non-standard inference services in Description Logics. Whenever the request is incompatible with an object's annotation (*i.e.*, their conjunction is unsatisfiable), *Concept Contraction* determines what part of the request cause the incompatibility and therefore should be retracted to pursue a match. If user's request and object's annotation are compatible, but the latter does not contain all requested features, *Concept Abduction* identifies the request elements not explicitly covered and therefore to be hypothesized about the object (under Open World Assumption) to reach a full match. As detailed in [3], *penalty functions* are associated to the outcome of Concept Contraction and Abduction –since minimal retraction and hypothesis are preferred– and provide a semantic distance metric of a set of objects w.r.t. a given request, enabling a logic-based relevance ranking. In the app settings, several weight configurations are available to combine the semantic distance with context-aware data-oriented attributes (*i.e.*, physical distance and user's preference) in the general penalty formula described in [2].

In case of trivial syntactic matching, all beacons would have the same score. A semantic-based matchmaking and a proper modelling of the reference knowledge base, make possible instead to highlight discrepancies between descriptions. By selecting a beacon, the user can: i) open in the browser the related Web page; ii) only for PSW devices, show the exposed OWL annotation (as in Figure 4(d)) and likewise the matchmaking outcome explanation. In the last case, the user can navigate missing and incompatible features between the bottle annotation and the user request. In Figure 4(c), *Mountadam Pinot Noir* obtains the highest rank because it covers all features specified by the user. *Chateau Morgon Beaujolais* and *Forman Cabernet Sauvignon* obtain a lower score because they cover the request partially and also contain inappropriate features.

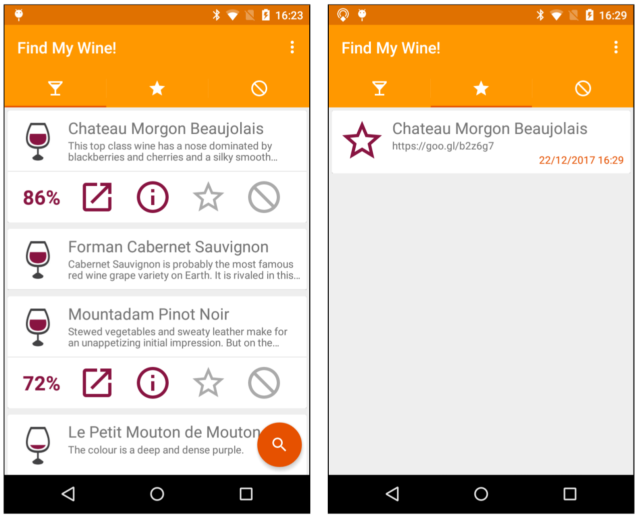
2.3 Stage 3: Advanced PSW features

PSW supports OWL annotations exposed by (both PSW-URL and PSW-UID) beacons changing dynamically due to events/phenomena

[7] https://github.com/google/physical-web/tree/master/web-service

[8] Typically, PSW support should be always on, given the backward compatibility of the proposed PW enhancements. Anyway, the user option has been implemented for demo purposes in order to make more evident discovery differences switching from PW-only mode to PW+PSW mode and back.

[9] http://github.com/sisinflab-swot/physical-semantic-web



Chateau Morgon Beaujolais
This top class wine has a nose dominated by blackberries and cherries and a silky smooth…

Forman Cabernet Sauvignon
Cabernet Sauvignon is probably the most famous red wine grape variety on Earth. It is rivaled in this…

Mountadam Pinot Noir
Stewed vegetables and sweaty leather make for an unappetizing initial impression. But on the…

Le Petit Mouton de Mouton
The colour is a deep and dense purple.

Chateau Morgon Beaujolais
https://goo.gl/b2z6g7
22/12/2017 16:29

(a) Updated list (b) Favorite beacons

Figure 5: Stage 3 - screenshots of the Android client

occurring to objects and/or context. In the proposed demo, the quality of each wine bottle can change according to storage conditions during its lifetime. An Arduino *TinkerKit Sensor Shield* is connected to one of the Intel Edison boards to emulate variations of environmental conditions in the winery, such as temperature, humidity and luminosity. Acting on the attached potentiometer, it is possible to simulate the situation of *e.g.*, a bottle left for some hours in a place with a very low temperature, very high humidity and high luminosity. Due to this reason, the wine quality decreases and the related annotation is updated. When a new discovery round starts, degradation of the wine is detected, leading to a final rank lower than before: see in Figure 5(a) the case of modifications on *Mountadam Pinot Noir*.

A *history* list keeps already explored objects when the user opens URLs or displays OWL annotations. Moreover, the user can mark one or more beacons as *favorites* (*i.e.*, useful, as in Figure 5(b)) or *spam* (*i.e.*, useless or unwanted): in the following discovery procedures the rank of such beacons will be respectively increased or decreased.

2.4 Stage 4: Infrastructureless PSW

No phone networks signal is available in the winery cellar; owners provide Wi-Fi Internet connection in the showroom area but Francesco would like visit the basement. In the Physical Web, a connection is always required to retrieve data from the Web. In the PSW, exploiting Eddystone-UID frames, knowledge exchange can also occur through point-to-point connectivity. The proposed app demonstrates that all PSW features are available even when Wi-Fi and cellular data networks are switched off on the smartphone. Furthermore, since in IoT scenarios connectivity is often intermittent, PSW allows caching OWL annotations retrieved from detected objects for a limited time.

3 DISCUSSION

The proposed demo provides interactive examples of the benefits of the Physical Semantic Web over the basic Physical Web framework. It is conceived as a four-stage journey, to engage users with various level of expertise and knowledge on the specific technologies involved. The reference scenario has been chosen to be easy to understand.

The first stage is dedicated to the current Physical Web paradigm. By trying out the basic PW discovery features, even users with no prior working knowledge of PW solutions can familiarize with the benefits of the general PW proposal as well as with its limits. The second stage introduces the PSW framework and allows an interactive first-hand experience of the enhancements to the overall capabilities. In particular, users should be able to appreciate the benefits of fine-grained matchmaking enabled by accurate semantic-based object annotation. The third stage demonstrates advanced features, such as favorite and spam caching, as well as the PSW ability to deal with dynamic annotations. In this case, users are able to set input data directly, by adjusting potentiometers attached to the demo PSW beacon.

Finally, the fourth and last demo stage shows how the PSW grants interaction even when a network infrastructure is unavailable, by exploiting point-to-point data exchanges only. This may appear as a corner case in the reference demo scenario, although most wineries are actually located in cellars without cellular or wireless local area network connectivity. Most importantly, however, that kind of situations is frequent in challenging IoT application scenarios (*e.g.*, environmental monitoring, surveillance, robot search and rescue teams, *etc.*), where stable Web connectivity cannot be granted. In those cases, basic interactions based on data retrieved from remote web services (*e.g.*, through simple URLs associated to beacons or barcodes) result unusable. The PSW still supports full capabilities, so enabling a whole new range of thing-to-thing cooperation scenarios through on-board autonomous software agents in addition to user-to-thing interactions.

A Web page[10] describing the proposed demo is made available. It contains further screenshots of the mobile application, links to the source code and the ontology and a sample demo screencast.

REFERENCES

[1] Natalya F Noy and Deborah L McGuinness. 2001. *Ontology development 101: A guide to creating your first ontology.* Technical Report. Stanford knowledge systems laboratory tech. rep. KSL-01-05 and Stanford medical informatics tech. rep. SMI-2001-0880.

[2] Michele Ruta, Floriano Scioscia, Saverio Ieva, Giuseppe Loseto, Filippo Gramegna, and Agnese Pinto. 2017. Knowledge Discovery and Sharing in the IoT: The Physical Semantic Web Vision. In *Proceedings of the Symposium on Applied Computing (SAC '17)*. ACM, New York, NY, USA, 492–498.

[3] Floriano Scioscia, Michele Ruta, Giuseppe Loseto, Filippo Gramegna, Saverio Ieva, Agnese Pinto, and Eugenio Di Sciascio. 2014. A mobile matchmaker for the Ubiquitous Semantic Web. *International Journal on Semantic Web and Information Systems* 10, 4 (dec 2014), 77–100.

[4] W3C OWL Working Group. 2012. *OWL 2 Web Ontology Language Document Overview (Second Edition).* W3C Recommendation. W3C. https://www.w3.org/TR/owl2-overview/.

[5] Roy Want, Bill N Schilit, and Scott Jenson. 2015. Enabling the Internet of Things. *Computer* 48, 1 (2015), 28–35.

[10] http://sisinflab.poliba.it/swottools/physicalweb/demo-wine.html