Jo Vermeulen · Ruben Thys · Kris Luyten · Karin Coninx
{jo.vermeulen,kris.luyten,karin.coninx}@uhasselt.be
ruben.thys@student.uhasselt.be

Making Bits and Atoms Talk Today

A Practical Architecture for Smart Object Interaction

Received: date / Accepted: date

Abstract Bringing together the physical and digital worlds has been the subject of research for some time now. In particular, a number of successful prototypes that link physical objects with digital information (often called smart object systems) have already been presented. However, a generally accepted architecture to design such systems has not yet emerged. This paper presents a reusable and practical framework for developing smart object applications today. At the basis of our approach lies the use of Semantic Web technology to drive interaction between the physical and digital worlds. We used this framework to develop SemaNews, a novel application that combines the advantages of digital news feeds with those of physical newspapers. To verify the reusability of our architecture, we built a second prototype in a different application domain: STalkingObjects provides the basic components of a store of the future.

Keywords Bridging physical and digital worlds · Smart objects · Architecture · Semantic web

1 Introduction

Back in 1991, Mark Weiser envisioned computers to disappear into the background and integrate seamlessly in our daily environment [1]. Computing power would be integrated in ordinary objects such as light switches, books or even clothes, making them effectively *smart* objects. Ubiquitous computing and related research topics such as augmented reality [2] and tangible interaction [3] all share the same goal of bridging the gap between the physical and digital worlds. According to Gershenfeld, the most compelling opportunities for information technology lie where both worlds meet. He argues that "all of the bits in the world are of no use if they can't come out to where people live" [4].

Hasselt University – transnationale Universiteit Limburg
Expertise Centre for Digital Media – IBBT
Wetenschapspark 2, B3590 Diepenbeek (Belgium)

Although numerous prototypes that effectively couple physical and digital entities have already been demonstrated [5–7], a generally accepted software architecture for these applications has not yet emerged.

In this paper, we present an architecture for interacting with smart objects that we believe is generic enough to be reusable for a wide range of applications (Sect. 3). We show its value in practical, real-life applications by introducing *SemaNews*, a smart object application built upon this architecture that tries to combine the advantages of digital news feeds with those of physical newspapers (Sect. 6). To validate the reusability of our framework we have built a second prototype in a different application domain: *STalkingObjects* implements the basic components of a store of the future (Sect. 7). While we do not pretend to cover every aspect of smart object systems with this framework, we feel it is certainly of practical value and might be a stepping stone towards developing an all-embracing solution.

In summary, we describe two main contributions in this work:

– A generic software architecture for interacting with smart objects based on the scalable and uniform design of the World Wide Web and reusing key concepts of the Semantic Web (Sect. 3).
– SemaNews, a novel prototype based upon this architecture that allows people to read digital news articles with the convenience of a physical newspaper, while still preserving the interaction capabilities of the digital version (Sect. 6).

2 Background

There is currently no generally accepted software architecture for smart object systems. While some systems are bound to specific technologies or are only partially reusable [7,8], others have unrealistic hard- or software requirements [5], focus mostly on hardware technology [9] or only concentrate on interaction with one smart object at a time [10]. Nevertheless, a lot of these systems are inspiring and incorporate good ideas.

Rogers [11] states that the majority of *ubicomp* research pursues unrealistic goals, such as trying to make computers act on behalf of humans. The positioning of ubicomp as something yet to be achieved might explain the lack of practical architectures that can be used at this very moment. However, as Bell and Dourish [12] point out, ubiquitous computing is already here; it simply has not taken the form that we originally envisaged. With the availability of devices with wireless data communications and powerful computational properties, one could say computation is already ubiquitous. Rogers proposes a subtle deviation in ubicomp research: moving from smart and proactive environments to environments that *enable* people themselves to be smarter and more proactive in their everyday practices.

In this work we try to combine the advantages of existing systems in order to come to a *generic* and *practical* framework for realizing smart object systems today. We kept Rogers previously mentioned advice in mind while implementing the SemaNews and STalkingObjects prototypes. Both systems try to support people in their daily activities without making decisions on their behalf or forcing them to change their habits.

3 Architecture

Our approach is inspired by the architecture of the World Wide Web [13] and builds on three key principles:

1. smart objects have a unique identifier
2. the data associated with smart objects and the interaction that they provide or can participate in is described in a structured, machine-readable way
3. interaction with smart objects is driven by data format negotiation

First of all, we need a way to associate physical objects with digital information. A requirement for this is that each of these physical objects has a unique identifier (e.g. a barcode). This identifier will then be transformed into a *digital identifier*, more specifically into a Uniform Resource Locator (URL) [1], which represents the object in the digital world. We can now link arbitrary digital data to this URL and thus to its corresponding physical object. Any physical identification technology (barcodes, RFID, etc.) can be used, as long as each physical identifier can be transformed to a URL. Sect. 4 provides more details on how this mapping is performed.

Although any kind of data can be associated with a physical object, there is no way of interpreting the meaning of this data. Smart objects can be linked to different types of information (e.g. numbers, pictures, etc.) each with many possible meanings (e.g. the number 15.00 can represent an amount of money such as 15.00 EUR or a specific time of day such as 3 PM). Furthermore, we believe objects will not only be linked to data, but can also provide interaction by linking to actions or *services*. These services in turn can use data of other objects as their input. As an example, consider a smart display that can visualize a speaker according to his/her smart badge. Smart object applications cannot have prior knowledge of each object's associated data since it is impossible to predict with which objects they will interact. Our architecture thus needs a generic way to use digital information coming from the physical environment. Applications should be able to deduce the meaning of this data while still allowing for a fully dynamic system.

Smart objects in our framework publish the meaning of their data (its *semantics*) in a structured, machine-readable way. For this we rely on the basic building blocks of the Semantic Web [14]: knowledge representation languages such as the Resource Description Framework (RDF) [2] and the Web Ontology Language (OWL) [3]. Shared knowledge is defined by ontologies (specifications of conceptualizations). Since objects can also be associated with services, we describe their semantics (e.g. preconditions, effects, semantic data types for inputs and outputs, etc.) as well using OWL-S [4], an ontology for semantic web services. An OWL-S service can be mapped to a concrete realization of the service (such as a WSDL [5] description). This means existing services can be reused and extended with semantics.

Adding semantics to the data and services that smart objects are linked to allows for *data format negotiation*. This means that a service can automatically select suitable inputs from the objects in its surroundings. These inputs can be very fine-grained: while existing web services specify for example that they require a string as input, semantic web services describe that they require a person name, and thus can reject other inputs such as city names. Any ontology can be used to describe these concepts, unknown concepts will simply be ignored. The SemaNews prototype (Sect. 6) for example uses the Resource of a Resource (ROR) ontology while STalkingObjects (Sect. 7) describes its concepts with a custom-built store ontology. Data format negotiation allows services to *pull* possible inputs out of the environment, instead of requiring a user to *push* inputs to the correct services. Sect. 5 discusses how we implemented this mechanism.

Fig. 1 shows how these three principles are used in our architecture. Each smart object has a physical *identifier* which is mapped onto a URL in the digital world. In this digital world, object **(a)** is associated with *machine-readable data* while object **(b)** provides a *machine-readable service* (e.g. the smart display we mentioned earlier). When object **(b)** senses object **(a)** in the physical world,

---

[1] `http://www.ietf.org/rfc/rfc1738.txt`

[2] `http://www.w3.org/TR/rdf-concepts/`
[3] `http://www.w3.org/TR/owl-features/`
[4] `http://www.w3.org/Submission/OWL-S/`
[5] `http://www.w3.org/TR/wsdl`

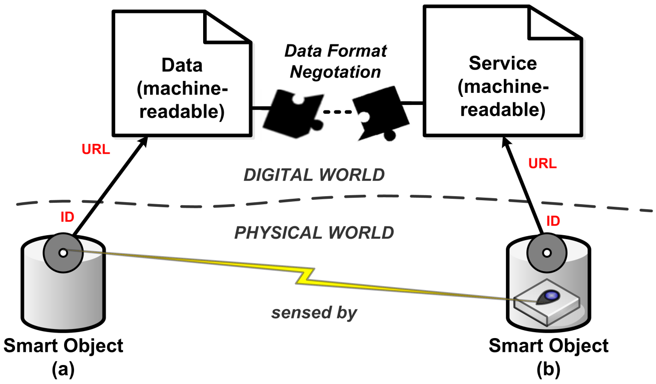


Fig. 1 A high-level overview of the system architecture.

it decides in the digital world if it can use object **(a)**'s data by means of *data format negotiation*.

4 Identification Technology

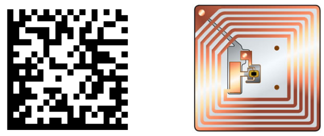
As we discussed in the previous section, every physical identifier should have a corresponding URL. There are a couple of methods to perform this mapping. One solution would be to create a unique URL based on the physical identifier: e.g. the RFID tag `00be3045fa` could be transformed into the URL `http://purl.org/tags/rfid/00be3045fa`. In the two prototypes we implemented however, URLs are embedded in the physical tags since these have sufficient storage capacity. Either approach will work, as long as a suitable URL can be obtained.

STalkingObjects uses Radio-Frequency Identification (RFID) tags [15] to identify physical objects. RFID has recently become an established low-cost identification technology and is already replacing the traditional barcodes in warehouse management solutions. It is not unlikely that in the future many objects in our everyday environment will be equipped with an RFID tag. Each tag contains a unique code that can be read by a wireless reader. Some types of tags have the ability to store additional data (typically up to 64 kilobytes). Fig. 2(b) shows an example of an RFID tag.

Semacode tags[6] (or Semacodes) are a special type of Data Matrix codes [16], two-dimensional barcodes that exist of a number of black and white regions. An example Semacode tag that encodes the URL `http://edm.uhasselt.be/` is shown in Fig. 2(a). The storage capacity of a Semacode tag depends on the dimension of its matrix (typically Semacode tags can store up to 2 kilobytes of data). Semacodes can be detected by cheap, low-resolution cameras such as those found in PDAs and mobile phones nowadays.

Because the architecture does not depend on a specific identification technology, it requires a small amount

[6] `http://semacode.org/`



(a) Semacode (b) RFID

Fig. 2 Two physical identification technologies.

of custom code to be able to use one. An object that provides interaction (e.g. object **(b)** in Fig. 1) needs to know when another physical object is near in order to pass the correct data to the data format negotiation process. Since the way to detect physical objects is specific to the employed identification technology, this component has to be provided. We feel this is no serious limitation since applications will often provide a custom user interface to ensure an optimal experience, which can include this component. STalkingObjects for instance provides this component within the "shoppingbuddy", while SemaNews integrates it in the user interface for the mobile phone. For more details, we refer to Sect. 6 and 7.

5 Machine-readable Semantics and Data Format Negotiation

As we previously mentioned, machine-readable semantics permit the use of digital information in a generic way. When both the inputs and outputs of services and data linked to smart objects are semantically described, these can be combined. This allows to automatically select appropriate inputs for a service or to select a suitable service according to the type of output it produces. Furthermore, by building upon semantic web technology we can reuse a large body of existing work done in areas such as ontology mapping or automatic service composition.

To realize data format negotiation, we developed *Sematcher*, a generic component that matches the semantics linked to a smart object with the semantic data types associated with the inputs of a number of OWL-S services. Given these OWL-S services and a recognized smart object, the procedure is as follows:

- create a list of semantic data types attached to the object
- for each service:
  - create a list of semantic data types attached to its inputs
  - compare this list with the object's list and determine matching `(data, input)`-tuples

As an example, suppose that an information kiosk in the city of Innsbruck is based on our architecture. It is linked to a service that accepts people's interests and names as defined by the FOAF ontology [7]. When

[7] `http://www.foaf-project.org`

Bob walks towards the kiosk and pulls his RFID-enabled identity card out of his wallet, Bob's FOAF profile is extracted and the data format negotiation component determines that his interests and name can serve as input to the kiosk service. After invoking the service, the kiosk displays information about events in Innsbruck according to Bob's interests and greets him with his name. Currently, Sematcher looks for exact matches or specializations of concepts (e.g. `Person` and `Scientist`), although this can easily be replaced with a more advanced algorithm.

An interesting application of data filtering is InfoPoint [17], a device to transfer data from one device to another. Unfortunately the authors do not clearly state how these data types are managed. While some applications use a central list of known data types, our architecture is more flexible since any ontology can be used.

6 SemaNews

The motivation behind SemaNews was the fact that although digital information is increasingly becoming part of our lives, it is often only easily accessible through computers. SemaNews makes digital content available in a physical form without losing the benefits of the digital version. More specifically, we tried to combine the advantages of physical newspapers with those of digital news feeds (e.g. news, blog posts, weather reports, etc.). The prototype allows people to read and interact with digital news articles in a comfortable way at any place and at any time, e.g. at the breakfast table or on the train.

SemaNews consists of two phases:

1. creating a physical newspaper from digital content
2. using this printed digital content in the physical world

The first step involves automatically creating and printing a PDF document from a collection of news feeds, customized to the user's liking (e.g. layout, image size, etc.). This process can be executed at a set time, for example every morning at 7 AM. A number of existing applications such as FeedJournal[8] already provide such a service. While this makes digital information available in a physical form, the link between the printed newspaper and its digital content is lost. Therefore SemaNews prints a Semacode tag (see Sect. 4) along with each article to link it back to its original, digital version.

Next, when reading the augmented newspaper, one can use the digital data associated with an article. Suppose for example that Alice reads an advertisement about a concert in the Innsbruck Arena and would like to know how to get there. With an ordinary newspaper, she would have to fire up her computer or mobile device and enter the location into a mapping service, thereby performing a manual conversion from physical to digital data. SemaNews, on the other hand, allows her to take a picture

Fig. 3 Alice chooses to show the location on Yahoo! Maps.

of the article's Semacode tag with her mobile phone, after which the system offers her a list of web services that can handle this kind of data. Alice then selects "Show location on Yahoo! Maps" which displays the concert location on a map, as depicted in Fig. 3.

Fig. 4 gives an overview of the SemaNews architecture. The SemaNews user interface runs on a mobile phone and lets users take pictures of the Semacode tags printed in the digital newspaper. Each Semacode tag contains a URL pointing to semantically-described content. For describing the semantics of news articles, we use the Resource of a Resource (ROR) [9] ontology. After taking a picture of a Semacode, the mobile phone application decodes the URL and calls Sematcher to perform data format negotiation. It passes a list of semantically-described, external web services to Sematcher which then decides which services can use the linked content as their input (e.g. Yahoo! maps is suitable for location data). Users can specify a list of external services they wish to use within SemaNews. If one or more services match, the mobile phone allows the user to select a service. After choosing one, the SemaNews application invokes this service with the correct inputs and shows the results to the user. The SemaNews user interface acts as a window on the digital world as shown in Fig. 4. This prototype illustrated the practical value of data format negotiation: users only have to choose between services that can process the data they wish to use.

7 STalkingObjects

We built a second prototype to validate the reusability of our architecture. STalkingObjects (Semantic Talking Objects) takes place in a shopping environment and uses the same basic components as SemaNews. The prototype improves the accessibility and usability of information in stores through a combination of portable computing, a specific ontology for stores and the framework described

[8] `http://www.feedjournal.com/`

[9] `http://www.rorweb.com/`

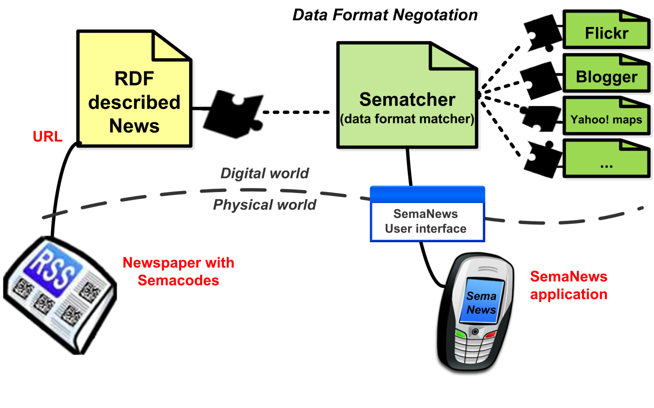


Fig. 4 An overview of the SemaNews prototype.

Fig. 5 The shopping buddy (implemented on a PDA) creates a mashup of nearby *smart products* and is coupled to the 3D simulation.

in this paper. More specifically, this prototype extends regular shopping carts with an embedded computer, also known as a "shopping buddy". This shopping buddy detects *smart products* in the store through an embedded RFID reader. It tries to visualize information coming from these smart products in a meaningful way.

A few existing proof-of-concept applications already incorporate digital technology in stores. Notable examples include the Metro Group's Future Store [10] and IBM's Customer Stop & Shop [11]. With STalkingObjects we do not aspire to contribute to this field, we discuss it mainly as another practical application of our architecture.

Suppose Bob is shopping in the new STalkingObjects store. His shopping buddy automatically creates a mashup[12] to visualize the smart products he encounters. This way Bob quickly notices new products or special offers, as Fig. 5 shows. Although Bob takes advantage of the features offered by the shopping buddy, he is in essence still shopping like before. When Bob decides to purchase an item, he just puts it in the shopping cart. Since smart objects have both a physical and digital representation, the shopping buddy can combine sensing a physical object being dropped in Bob's cart with selecting the corresponding price from its semantic description and updating Bob's total in the digital world. We argue that the powerful combination of physical objects and digital, machine-readable semantics that our architecture offers simplifies the development of a future store. Advanced concepts such as automatic check-outs or detailed in-store information screens can be realized more easily with our architecture.

Due to practical considerations, we developed this prototype in a 3D environment. This ensured that we could evaluate the STalkingObjects concept in a realistic setting: an environment containing a large amount of smart objects in a limited area. In addition, we developed

[10] `http://www.future-store.org/`
[11] `http://www-03.ibm.com/industries/retail/doc/content/bin/stop26shopfinal.pdf`
[12] `http://en.wikipedia.org/wiki/Mashup_(web_application_hybrid)`

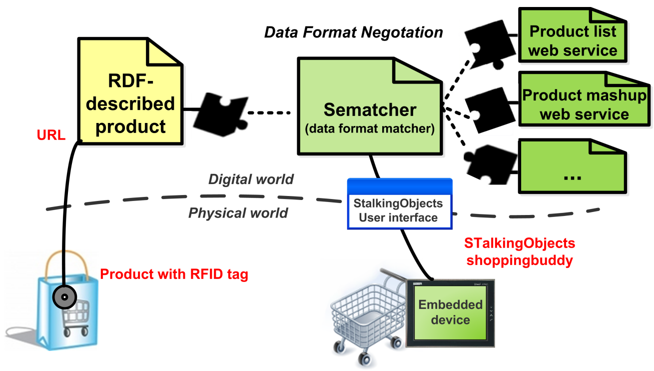


Fig. 6 An overview of the STalkingObjects prototype.

a PDA application that acts as a shopping buddy and is directly linked to the 3D simulation, as shown in Fig. 5.

Fig. 6 shows the basic components of the STalkingObjects prototype. STalkingObjects interacts with customers through a shopping buddy which is mounted on each shopping cart. This embedded device includes an RFID reader to detect smart objects in the store. Each product in the store has an RFID tag linking it to its semantic description. After recognizing a smart product, the shopping buddy calls Sematcher to decide which of the STalkingObjects services can use the product description. When a service matches, the shopping buddy does not disturb the customer but invokes the service in the background and aggregates the results on the screen. Similar to the SemaNews prototype (Sect. 6), the STalkingObjects shopping buddy acts as a window on the digital world.

8 Related Work

As we discussed in Sect. 2, there is a large body of existing work that tries to connect the virtual and the physical. However, surprisingly few approaches are usable today without much extra effort. The Cooltown project at HP Labs [18] comes closest to our architecture and pioneered many of the ideas found in this paper. An important difference however is our use of Semantic Web technologies. This allows for smarter interaction between the physical and virtual worlds without sacrificing generality, as demonstrated by both prototypes. Furthermore, the Semantic Web allows us to reuse existing work in topics such as ontology mapping.

Broll et al. [10] use a mobile phone to interact with services provided by physical objects such as posters. This work shows that semantic service descriptions can allow for more flexible interaction with services. Their system is more tailored towards user interface generation however.

9 Conclusions and Future Work

This paper presented a realistic and reusable software framework for developing smart object systems based on the design of the World Wide Web. It uses Semantic Web technology to drive the interaction between the physical and digital worlds. Upon this framework we built SemaNews, a smart object application that combines the advantages of digital news feeds with those of physical newspapers. To demonstrate the reusability of the architecture, we developed STalkingObjects, a prototype of a smart store. Video demonstrations of these prototypes can be found at `http://research.edm.uhasselt.be/smartobjects/`. We feel these are just two examples of applications that can be supported by our framework. For instance, much (if not all) of the infrastructure for some of the original scenarios described by Weiser [1] (e.g. Sal circling a quote in a newspaper and sending it to her office computer) can be provided by our architecture.

We see two main directions for future work. First of all, it would be useful to experiment with more advanced matching algorithms, for example to be able to decide between multiple possible inputs for a service. Secondly, we need to investigate how we can support sensor data. Sensor data becomes less relevant over time or when a certain treshold is reached, depending on the type of sensor and the application. This is currently not taken into account in our framework. RDF:Stores [19] is an interesting approach to support this kind of data that might be combined with our work.

Acknowledgements Part of the research at EDM is funded by ERDF (European Regional Development Fund), the Flemish Government and the Flemish Interdisciplinary institute for BroadBand Technology (IBBT).

References

1. Mark Weiser. The computer for the 21st century. *Scientific American*, 265(3):66–75, September 1991.
2. Ronald Azuma. A survey of augmented reality. *Presence*, 6(4):355–385, 1997.
3. Hiroshi Ishii and Brygg Ullmer. Tangible bits: towards seamless interfaces between people, bits and atoms. In *CHI '97: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 234–241, New York, NY, USA, 1997. ACM Press.
4. Richard N. Katz. Bits and atoms: An interview with neil gershenfeld. *Educause review*, 37(2), March 2002.
5. Pierre Wellner. The digitaldesk calculator: tangible manipulation on a desk top display. In *UIST '91: Proceedings of the 4th annual ACM symposium on User interface software and technology*, pages 27–33, New York, NY, USA, 1991. ACM Press.
6. Ron B. Yeh. Physical-digital ensembles for mobile interaction. In *CHI '06: CHI '06 extended abstracts on Human factors in computing systems*, pages 1791–1794, New York, NY, USA, 2006. ACM Press.
7. Roy Want, Kenneth P. Fishkin, Anuj Gujar, and Beverly L. Harrison. Bridging physical and virtual worlds with electronic tags. In *CHI '99: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 370–377, New York, NY, USA, 1999. ACM Press.
8. Aabhas V. Paliwal and Nabil Adam and Christof Bornhövd, Joachim Schaper. Semantic discovery and composition of web services for rfid applications in border control. In *Preparing to Meet the World of Business Applications a workshop at ISWC 2004*, November 2004.
9. Lars Erik Holmquist et al. Smart-its friends: A technique for users to easily establish connections between smart artefacts. In *UbiComp '01: Proceedings of the 3rd international conference on Ubiquitous Computing*, pages 116–122, London, UK, 2001. Springer-Verlag.
10. Gregor Broll et al. Supporting mobile service usage through physical mobile interaction. In *PERCOM '07: Proceedings of the Fifth IEEE International Conference on Pervasive Computing and Communications*, pages 262–271, Washington, DC, USA, 2007. IEEE Computer Society.
11. Yvonne Rogers. Moving on from weiser's vision of calm computing: Engaging ubicomp experiences. In *Proceedings of the 8th International Conference on Ubiquitous Computing*, pages 404–421, 2006.
12. Genevieve Bell and Paul Dourish. Yesterday's tomorrows: notes on ubiquitous computing's dominant vision. *Personal Ubiquitous Comput.*, 11(2):133–143, 2007.
13. Ian Jacobs and Norman Walsh. Architecture of the world wide web, volume one. `http://www.w3.org/TR/webarch/`, December 2004.
14. Tim Berners-Lee, James Hendler, and Ora Lassila. The Semantic Web. *Scientific American*, 284(5):34–43, 2001.
15. Roy Want. An introduction to rfid technology. *IEEE Pervasive Computing*, 5(1):25, 2006.
16. International Organization for Standardization. ISO/IEC 16022: Data Matrix, 2000.
17. Naohiko Kohtake, Jun Rekimoto, and Yuichiro Anzai. Infopoint: A device that provides a uniform user interface to allow appliances to work together over a network. *Personal Ubiquitous Comput.*, 5(4):264–274, 2001.
18. Tim Kindberg et al. People, places, things: web presence for the real world. *Mob. Netw. Appl.*, 7(5):365–376, 2002.
19. Michael Schneider. Rdf:stores - a lightweight approach on managing shared knowledge. In *UIC '06: Proceedings of the Third International Conference on Ubiquitous Intelligence and Computing*, volume 4159 of *Lecture Notes in Computer Science*, pages 229–239. Springer, 2006.