A. Juarez, J. Hu, and L. Feijs, "RoboDB: an application of Semantic Web Technologies to robotics," in Semantic Web Challenge, 2011.

RoboDB: an application of Semantic Web Technologies to robotics

Alex Juarez, Jun Hu, and Loe Feijs

Department of Industrial Design
Eindhoven University of Technology
Den Dolech 2, 5600AZ Eindhoven, Netherlands
acordova@tue.nl, j.hu@tue.nl, l.m.g.feijs@tue.nl

Abstract. RoboDB is a knowledge acquisition system that gathers information about robots. RoboDB uses Semantic Web technologies and tools to help the user in creating semantic descriptions of robot embodiments and their capabilities, as well as in building an ontology of robotics projects, research institutions, people, and other aspects of robotics. RoboDB allows its contributors to also become moderators of much needed discussions, and potentially reach agreements on the different terminology used in this heterogeneous and dynamic field. The semantic data collected in RoboDB has been used in two application scenarios: interaction with virtual agents using ROILAbot, and the creation of a robotics directory for the Dutch Robotics Network (RoboNed).

1 Introduction

RoboDB is a knowledge acquisition system that gathers information about robots. Robots are heterogeneous, complex devices with numerous sensors, actuators, and capabilities associated to them. Similarly, robotics is a rapidly evolving field where constant innovation and change is the order of the day. As a result, it is extremely difficult to have an overview of the field itself, let alone an overview of the different types of robots available. Even simple queries like "find robots with two or more legs" or "find robotics research projects working in human-robot interaction" are difficult (and sometimes impossible) to answer with common search engine searches. RoboDB uses Semantic Web technologies (SWT) and tools to help the user in creating semantic descriptions of robot embodiments and their capabilities, as well as in building an ontology of robotics projects, research institutions, people, and other aspects of robotics.

1.1 The time is right

A robot that wants to use the Web to obtain information about other robots and their capabilities, will need to *make sense* of this information automatically. SWT have reached a level of maturity that makes them suitable for this task.

SWT have been previously applied in the field of robotics. A survey of the literature reveals a body of robotics research to illustrate this. For example, the study of Chella et al. [2] used ontologies to describe the environment in which a mobile robot moves and to make decisions about the proper way to navigate through space. Johnston et al. [6] used semantic technologies to represent and manage the relationship between entities that were recognized by robotic vision software. Yanco and Drury [14] created a taxonomy for human-robot interaction (HRI) to allow the comparison of different HRI systems.

From the perspective of community-based generation of semantic content, the idea of collaborative generation and curation of semantic data is now new. Systems like Shortipedia [13] and APOSDLE [4] have demonstrated the feasibility and scalability of the approach. Furthermore, Semantic Mediawiki is a system in constant evolution, with many third party extensions like SMW+ [10] periodically expanding the functionality and performance of the original application. The novelty of RoboDB lies in the combination of the collaborative data generation approach, the representation power and flexibility of SWT to describe robot embodiments, and the user-centered design used in its development. To the best of our knowledge, there is no publicly-available system with the same characteristics as RoboDB.

Due to limitations in space we will not extend into a detailed description of how RoboDB is used, or how each of the features mentioned in this paper was implemented. For this we refer to the RoboDB site (`http://www.robodb.org`).

2 System architecture

RoboDB is a software extension (plug-in) to be deployed on top of the Semantic Mediawiki (SMW) system [5]. SMW builds upon the Mediawiki technical infrastructure. It extends the logic layer by adding PhP scripts to enable semantic annotations, visualization of semantic data, and semantic content management.

RoboDB uses other available SMW extension (plug-in) packages to manage different aspects of semantic content creation:

Halo extension [8] provides enhanced user interface features that help in annotating web content and visualizing existing semantic information. RoboDB uses especially the Ontology Browser, a visualization tool that presents semantic annotations in a tree-like structure.
Triple store connector [11] is an extension that integrates a reasoning engine with RoboDB. The TSC is based on the Jena Semantic Web Framework (`http://jena.sourceforge.net`), an open-source programatic environment for RDF(-S)/OWL that includes a rule-based inference engine.

RoboDB integrates these components into a cohesive solution, while at the same time it extends the functionality mentioned above as follows (See Figure 1 for RoboDB's system architecture):

Guided creation of robots descriptions RoboDB replaces the traditional edit mechanism of SMW by a guided, interactive procedure to create a description of the robot physical structure and its capabilities. This "wizard" application provides a visual interface where the user can add components (sensors, actuators, etc.) and connect them to produce an abstract representation of the robot. Semantic annotations are created automatically behind the scenes and added to the user content. Web pages with content not related to robots can be created and edited using the original edit features from SMW.

Export the semantic data to other formats like OWL/MPEG-V. Semantic annotations and robot structural descriptions are encoded in RDF(-S). RoboDB also exports semantic data to other formats like OWL and the new standard for data exchange between virtual and real worlds MPEG-V.

A refreshed user interface. RoboDB keeps some functionality and layout from SMW, and adds a renewed user interface, designed and revised during the iterative development process.

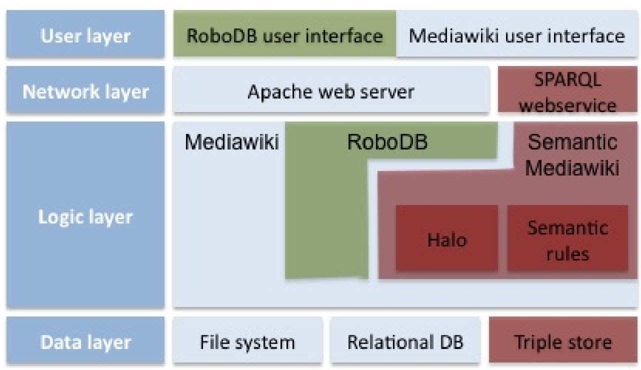

Fig. 1: RoboDB system architechture architecture

3 Design process

The design methodology adopted during the developement of RoboDB was the spiral model [1], which proposes an iterative design process with four phases (*requirements*,*design*,*implementation*, and *evaluation*), in which the outcome of each iteration is fed to the next one in the form of requirements and constraints. There were three very distinctive phases of design: the initial prototype based on Semantic Mediawiki, a second prototype that included guided input methods ("wizards") based on web forms, and a third prototype included an interactive graphic interface to describe robots. The following sections describe some of the design decision and lessons learned during this process.

3.1 Design decisions

Not every entity is a page. Semantic Mediawiki works under the principle that every entity that can be annotated in the knowledge base is a wiki page. This proved to be an unfeasible approach to model the robot structure, since creating a wiki page for each of the robot sensors and actuators and then connect them via properties actually made the process quite complex and cumbersome. Instead we decided to use XML as an intermediate internal representation for the robot structure. This had two benefits: a) it allowed us to reduce the clutter of wiki pages in RoboDB, and b) it was also the initial step to develop an interactive application to create visual representations of the robot structure in the form of connected graphs. When the semantic information of a robot is exported to OWL format, the XML description of the robot is converted into OWL entities.

How it looks is as important as what it does. During the different iterations of the development it was clear that the users reacted poorly to the original Mediawiki interface. This motivated us to change the look and feel of the site, and to streamline the way some applications worked (e.g. the static and derived facts boxes, the ontology browser display). Even subtle changes on the interface proved to have a positive impact on the perception of the system.

Complex processes are better served interactively. Modelling a robot's structure by manually creating the required semantic annotations proved to be a daunting and complicated task, especially as the complexity of robots increased. This was also true when trying to annotate pages about robotics research projects for the Dutch robot directory (See Section 4.1). This motivated us to replace the traditional annotation mechanism from SMW by a form-based page creation wizard. This worked well for simple robots and robotics research project pages, however, complex robots still proved to be difficult to handle even with dynamic forms and feedback to the user. Finally we implemented an interactive graph applet that allows the users to visually create the robot structure by adding nodes robot components and connections between them. Figure 2 shows an example of this: the guided process to generate a robot description.

3.2 Lessons learned

The lessons learned were either influenced or a direct result of the design decisions made for RoboDB (and viceversa).

Real-world users are not interested in becoming Semantic Web experts. One of our first explorations consisting in modelling robots and their capabilities using tools like Protégé[3] and TopBraid Composer [12] received a negative reaction from our users. Although they acknowledged the importance of adding meaning to the robot descriptions, they wanted a tool that would guide them in the process of creating semantic data, and not require them to know much about Semantic Web.

There is a fine line between freedom and usability. The balance between building a simplified, semi-automatic mechanism to create semantic information and the freedom needed to generate knowledge about constantly changing concepts is hard to find. This is especially true when modelling knowledge about evolving devices such as robots. The choice of system features that need to be automated and restricted versus those that can be left open to the user is crucial for the success of this initiative.

Do not give me “yet-another-wikipedia”!. Users had a clear perception that creating yet-another-wiki was not useful. They saw little value in including

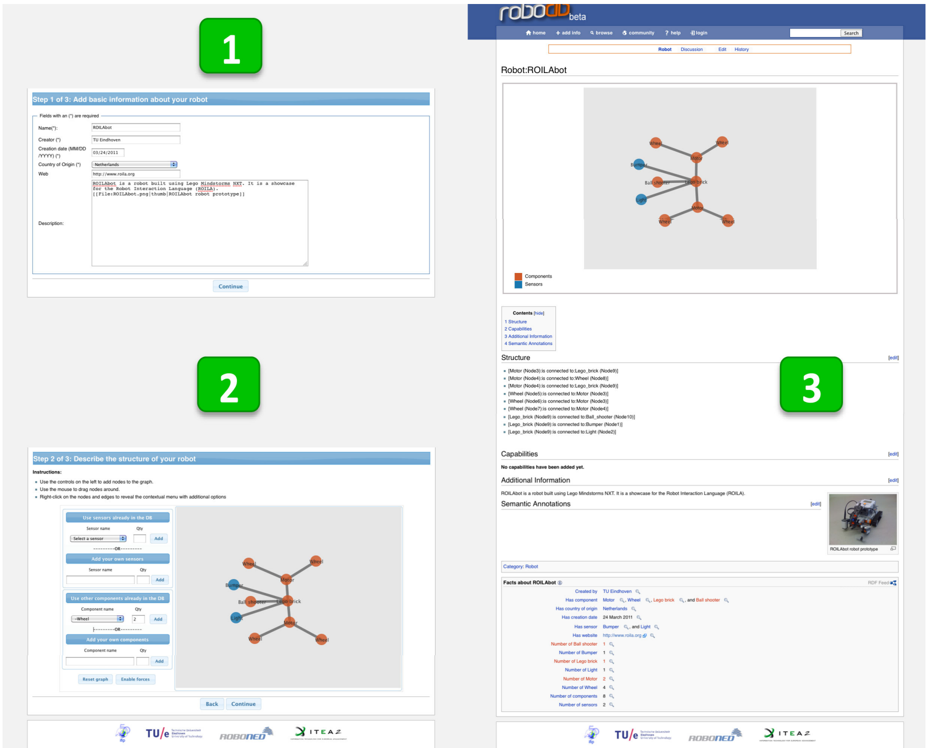

Fig. 2: Example of an interactive process to create a robot’s structure description. 1) Basic information about the robot (form-based data collection) 2) Interactive robot structure creation (Java Applet) 3) Final result. (This image is best viewed with the magnification tool from your PDF viewer)

information that was already available either in Wikipedia or other websites. Instead they’d rather have semantic or even web-links to those external pages and leave in RoboDB only that information that is “new”.

4 Use cases

4.1 Dutch robotics directory

The Dutch Robotics Network (RoboNed) is an organization that groups robotics researchers from academia and industry in the Netherlands. RoboNed collaborated with RoboDB to develop a prototype for the directory of Dutch Robotics, a web-based application where researchers, potential customers, and the general public could easily find information about other past and present research initiatives, and developments in the Dutch robotics community. A key requirement was one of the classic scenarios for SWT: to generate some structure in the information in order to link the robot semantic data to other related data. For example, a user could “navigate” through the data by searching for a robotics project page, and then following the semantic links to its associated robot pages. Once there they could find out other research projects were using the same robot, fields of application, people working on them, etc.

The RoboDB platform was well suited for their needs as it provides a collaborative framework for content creation and the SWT needed for this purpose are already in place. The prototype for the Dutch robotics directory is already available with data from more than 80 robotics research projects. The interested reader can visit `http://www.robodb.org` to explore it.

4.2 ROILAbot

ROILAbot is a robot built using Lego Mindstorms. It serves as a platform to demonstrate the use of the Robot Interaction Language (ROILA), a language designed especially to improve speech recognition in robots [7]. ROILAbot has its virtual counterpart in the virtual world Second Life, and virtual agents can interact with it and command it to move. However, users of the virtual world do not always know what type of language the robot can “speak”. In the demonstration prototype, the robot structures created in RoboDB were used to model the capabilities of ROILAbot as *logic rules* (using Jena rule format) and concepts from the OWL-S upper ontology. This enabled virtual agents to automatically determine what kind of movements the robot can do, the type of components it has, commands it can accept, and languages it can speak (i.e. ROILA). Using an external ontology that maps English to ROILA we have also enabled the automatic translation of words for virtual world users to be able to give commands to ROILAbot. Figure 3 shows a snapshot of a virtual agent interacting with the virtual version of ROILAbot in SecondLife.

At the moment, the modeling of robot capabilities in RoboDB is done manually. Therefore, it is not yet open to the public on the RoboDB website, as we are still evaluating the appropriate mechanisms and tools that would better help the user, e.g. the Semantic Rules extension for SMW [9].

5 Conclusion

We introduced RoboDB, a knowledge acquisition system that gathers information about robots. Although the system is still in its infancy and clearly needs to be developed more, it has shown that the semantic web approach is valid when modeling complex knowledge about robots and robotics. We believe in its potential to generate quality semantic data about robots and robotics that can be reused not only by the robotics community by also by the general public.

References

1. B.W. Boehm. A spiral model of software development and enhancement. *Computer*, 21(5):61–72, 1988.

Fig. 3: A virtual agent interacting with the virtual ROILAbot in Second Life

2. A. Chella, M. Cossentino, R. Pirrone, and A. Ruisi. Modeling ontologies for robotic environments. In *Proceedings of the 14th international conference on Software engineering and knowledge engineering*, page 80. ACM, 2002.
3. John H. Gennari, Mark A. Musen, Ray W. Fergerson, William E. Grosso, Monica CrubÈzy, Henrik Eriksson, Natalya F. Noy, and Samson W. Tu. The evolution of protègè: an environment for knowledge-based systems development. *International Journal of Human-Computer Studies*, 58(1):89 – 123, 2003.
4. C. Ghidini, M. Rospocher, L. Serafini, B. Kump, V. Pammer, A. Faatz, A. Zinnen, J. Guss, and S. Lindstaedt. Collaborative knowledge engineering via semantic mediawiki. *Proceedings of the I-Semantics*, pages 3–5, 2008.
5. M. Krötzsch, D. Vrandečić, and M. Völkel. Semantic mediawiki. In *The Semantic Web-ISWC 2006*, volume 4273/2006 of *Lecture Notes in Computer Science*, pages 935–942. Springer, Nov. 2006.
6. R. Mendoza, B. Johnston, F. Yang, Z. Huang, X. Chen, and M.A. Williams. OBOC: Ontology Based Object Categorisation for Robots. In *Proceedings of the 4th International Conference on Computational Intelligence, Robotics and Automation (CIRAS 2007), Palmerston North, New Zealand*. Citeseer, 2007.
7. Omar Mubin, Christoph Bartneck, and Loe Feijs. Towards the design and evaluation of roila: A speech recognition friendly artificial language. In H. Loftsson, E. Rögnvaldsson, and S. Helgadóttir, editors, *Proceedings of the 7th International Conference on Natural Language Processing*, volume 6233/2010 of *Advances in Natural Language Processing*, pages 250–256, 2010.
8. Ontoprise. HALO Extension for semantic mediawiki. `http://smwforum.ontoprise.com/smwforum/index.php/Halo_extension`.
9. Ontoprise. Rule Knowledge Extension for semantic mediawiki. `http://smwforum.ontoprise.com/smwforum/index.php/Help:Rule_Knowledge_extension_1.2.1`.
10. Ontoprise. Semantic mediawiki + (smw+). `http://smwforum.ontoprise.com/smwforum/index.php/Semantic_MediaWiki_Plus`.
11. Ontoprise. Triple Store Connector for semantic mediawiki. `http://smwforum.ontoprise.com/smwforum/index.php/TripleStore_Basic`.
12. TopQuadrant. Top Braid Composer. `http://www.topquadrant.com/products/TB_Composer.html`.

13. D. Vrandecic, V. Ratnakar, M. Krötzsch, and Y. Gil. Shortipedia aggregating and curating semantic web data. *Web Semantics: Science, Services and Agents on the World Wide Web*, 2011.
14. H.A. Yanco and J.L. Drury. A taxonomy for human-robot interaction. In *Proceedings of the AAAI Fall Symposium on Human-Robot Interaction*, pages 02–03, 2002.

A How RoboDB meets the Challenge criteria

Application has to be an end-user application. RoboDB is an end user application accessible and editable by everyone on the web. At the same time, it is a domain specific application that can serve as reference for the robotics community and is actively used by the RoboNed organization.
Information sources should be heterogeneous. RoboDB contains information of diverse nature (e.g. robots structural descriptions, information about robotics projects and institutions, etc.) that is administered by the robotics community itself. Although at the moment it does not include information from sources other than the RoboNed database, it has the potential to incorporate data from other existing ontologies related to robotics, or other information sources like DBpedia.
Meaning of data has to play a central role. As shown in the use cases presented previously, the meaning of information is central when describing robots (e.g. what does it mean for a robot to be able to walk, or talk a specific language?). Data is manipulated in interesting ways, as the robot's embodiment descriptions are created interactively and automatically converted to semantic annotations ready to be queried by external applications.

A.1 Additional criteria

Application provides an attractive user-interface. Creating an attractive and interactive user interface was crucial for the user's acceptance of this system. This was expecially true when creating descriptions of the robot structure.
Novelty in applying semantic web technologies to a domain or task. Although the application of semantic web technologies to robotics is not new, it has been usually related to the planning domain, and the robot-as-service research. Only recently are initiatives like RoboDB trying to go beyond and completely represent a robot and its capabilities using the semantic web.
The application has clear commercial potential and/or a large user base. Although at the moment the user base is the robotics community of the Netherlands (RoboNed), it clearly has the potential to appeal to the international robotics community.
Functionality is different or goes beyond pure information retrieval. Although information retrieval is one of the current use cases of RoboDB, our second use case shows that the information can be effectively used in different, heterogeneous applications.