Next Generation Semantic Web Applications

Enrico Motta and Marta Sabou

Knowledge Media Institute
The Open University, UK
{e.motta, r.m.sabou}@open.ac.uk

Abstract. In this short paper, we examine current Semantic Web application and we highlight what we see as a shift away from first generation Semantic Web applications, towards a new generation of applications, designed to exploit the large amounts of heterogeneous semantic markup, which are increasingly becoming available. Our analysis aims both to highlight the main features that can be used to compare and contrast current Semantic Web applications, as well as providing an initial blueprint for characterizing the nature of Semantic Web applications. Indeed, our ultimate goal is to specify a number of criteria, which Semantic Web applications ought to satisfy, if we want to move away from conventional semantic systems and develop a *new generation of Semantic Web applications*, which can succeed in applying semantic technology to the challenging context provided by the World-Wide-Web.

Introduction

The past few years have witnessed a growing interest in the Semantic Web [1], as shown by the rapid increase of the amount of semantic markup available on the Web, by the growing number of organizations starting research and development activities in this research area, and by the number of Semantic Web applications, which now exist. Indeed, current data appear to show that the growth of the Semantic Web is mirroring that of the Web in the early nineties [2], a strong indicator that a large scale Semantic Web is likely to become a reality sooner rather than later.

The availability of semantic markup opens up novel possibilities to develop smart, web-based functionalities. For instance, in the brief history of the Semantic Web we have already seen Semantic Web applications that support intelligent data aggregation and presentation [3, 4], semantic search [5], automatic annotation [6, 7], question answering [8], and Semantic Web browsing [9]. However, if we look closely at the way these applications make use of web-based semantic and non-semantic resources, we can highlight a clear distinction between those applications which truly embrace the Semantic Web paradigm, and those which are more akin to conventional knowledge-based systems. At a coarse-grained level of abstraction, this distinction can be expressed by the difference between 'closed' semantic systems, which typically use a single ontology to perform data aggregation in a domain-specific fashion, and 'open' systems, which are heterogeneous with respect to both the ontological characterization and the provenance of the semantic data they handle.

In this paper we will explore this issue in some detail and we will propose a set of features that, in our view, will increasingly characterize the Semantic Web applications. Our analysis aims to be both *descriptive* and *prescriptive*. Descriptively, the objective here is to characterize the space of current Semantic Web applications, provide dimensions to compare and contrast them, and identify key trends. Prescriptively, our goal is to specify a number of criteria, which Semantic Web applications ought to satisfy, if we want to move away from conventional semantic systems and develop a *new generation of Semantic Web applications*, which can succeed in applying semantic technology to the challenging context provided by the World-Wide-Web.

Features of open Semantic Web applications.

In what follows we introduce seven dimensions for analyzing Semantic Web applications and we use them to characterize a representative sample of Semantic Web systems. In particular we will compare and contrast systems such as CS Aktive Space [3], which can be characterized as *first generation* Semantic Web applications, from more recent systems, such as PiggyBank [9], a Semantic Web browser, or PowerAqua [8], a question answering system for the Semantic Web, which in our view provide early examples of the *next generation* of Semantic Web applications.

- **Semantic data generation vs reuse**. Early Semantic Web applications, such as CS Aktive Space [3], were developed in a context in which little semantic information was available on the Web. Hence, these applications produced and consumed their own data, much like traditional knowledge-based applications. In contrast with CS Aktive Space, more recent applications, such as PiggyBank or PowerAqua, are designed to operate with the semantic data that already exist. In other words, they worry less about bootstrapping a Semantic Web, than about providing mechanisms to exploit available semantic markup[1].
- **Single-ontology vs multi-ontology systems.** A first generation system, such as CS Aktive Space, makes use of a specific ontology, in this case the AKT Reference Ontology [10], to support data aggregation and provide a unified semantic model to the data acquired from several different sources. In contrast with CS Aktive Space, neither PiggyBank nor PowerAqua rely on any specific ontology. On the contrary, these systems can consume any number of ontologies at the same time. The rationale for this choice is that these systems simply assume that they operate on a large scale Semantic Web, characterized by huge amounts of heterogeneous data, which could be defined in terms of many different ontologies. In this context, it clearly does not make much sense to make a ‘closed domain’ assumption. It is interesting here to compare PowerAqua with an earlier question answering system, AquaLog [11]. While AquaLog is also ontology-independent, it cannot use multiple ontologies concurrently to answer a particular query. In other words,

[1] Actually PiggyBank also provides bootstrapping mechanisms to extract semantic data from HTML, however these are meant to provide extra flexibility to the system, rather than being an essential aspect of its modus operandi.

while AquaLog also assumes that a Semantic Web query system must be able to operate with different ontologies, it still assumes that it is feasible to make use of one ontology at the time. This feature makes AquaLog especially suitable for semantic organizational intranets, where organizational data are usually annotated in terms of a single organizational ontology. However it clearly makes it unsuitable for the Semantic Web at scale, where heterogeneous semantic data may need to be combined to answer specific queries.

- **Openness with respect to semantic resources.** This criterion distinguishes between those systems which are closed with respect to the semantic data they use and those which are able to make use of additional, heterogeneous semantic data, at the request of their user. For instance, a system such as CS Aktive Space cannot take into account RDF data available from a particular Web site, in response to a request from a user who wish to use them. CS Aktive Space can only use the data that the system developers have scraped from the various relevant sites and re-described in terms of the AKT Reference Ontology. In contrast with CS Aktive Space, a system such as PowerAqua has no such limitation: if new data become available, PowerAqua can use them with no configuration effort, to try and answer queries.
- **Scale as important as data quality.** The key feature of the Web is its size. Because publishing on the Web is so inexpensive, it has grown "like a virus", acquired gigantic proportions in an incredibly short time, and revolutionized the way we access and publish information, shop, operate our businesses, socialize, and interact with our peers and with organizations. As argued earlier, it is likely that the Semantic Web will follow a similar growth pattern and as a result we will soon be able to build applications, which will explore, integrate, and exploit large amounts of heterogeneous semantic data, generated from a variety of distributed sources. Given this context it is interesting to distinguish between those applications that are designed to operate at scale and those which are more similar to the small-medium sized knowledge-based applications of the past. Indeed all the Semantic Web applications mentioned in our introduction take scale seriously. The difference is primarily between those applications like PowerAqua, which do not require any extra effort to bring in new sources, and those like CS Aktive Space, which require additional programming to bring in new information. In other words, scale per se is much less a useful discriminator between the first and second generation of Semantic Web applications, than a system's ability to link its performance to the amount of semantic data existing on the Web. Two important implications arise from this emphasis on scale as one of the key features of Semantic Web applications. Firstly, the moment a system has to reason with very large amounts of heterogeneous semantic data, drawn from different sources, then necessarily these systems have to be prepared to accept variable data quality. Secondly, intelligence in these large-scale semantic systems becomes a side-effect of a system's ability to operate with large amounts of data, rather than being primarily defined by their reasoning ability. This aspect strongly distinguishes Semantic Web applications from traditional knowledge-based systems, where the intelligence of a system was typically defined in terms of its ability to carry out complex tasks, such as, for instance, diagnosis, planning, or scheduling [12].

- **Openness with respect to Web (non-semantic) resources.** In our view a system that operates on a large-scale, rapidly evolving Semantic Web, should also take into account the high degree of change of the conventional Web. In particular, platforms such as TAP [5] and PiggyBank [9] provide facilities for integrating data acquisition mechanisms in their architecture, to facilitate the extraction of data from arbitrary sources on the Web. Analogously, automatic annotation systems such as KIM [7] and Magpie [6] can work on any Web page, although of course the quality of the annotation may degrade if the page in question does not reflect the current ontology used by these systems to drive automatic annotation.
- **Compliance with the Web 2.0 paradigm.** As pointed out by Tim O'Reilly, a key principle behind the success of the Web 2.0 paradigm is that of *Harnessing Collective Intelligence.* In other words many of today's most successful Web applications are based on massively distributed information publishing and annotation initiatives, such as Wikipedia[2], Flickr[3], etc. While it ought to be emphasized that a large scale Semantic Web will primarily be constructed by exploiting automatic data generation and integration mechanisms, it is also important to note that Semantic Web applications cannot ignore the lessons from the success of Web 2.0 applications and therefore they ought to embed Web 2.0 features. If this premise is correct, then there are at least two very important implications. The first one is that, like typical Web 2.0 applications, Semantic Web systems also need to provide mechanisms for users to add and annotate data. Indeed, at conferences such as the European Semantic Web Conference we have already seen the value of allowing distributed semantic annotation. However tools to support user annotation are still rather primitive, and better tools are badly needed. Another important aspect of integrating Web 2.0 principles into Semantic Web activities concerns the need to integrate artefacts such as folksonomies into Semantic Web applications. Although systems such as PiggyBank already provide tagging mechanisms, this is only a preliminary step. The next step will be to perform a deeper integration of folksonomies and ontologies. In our group we are examining the use of relation extraction mechanisms to achieve this goal.
- **Open to services.** Web services have revolutionized the Web, transforming it from a static information space to a dynamic data sharing infrastructure. In our view it is also essential that Semantic Web applications integrate web service technology in their architecture, and indeed we can already highlight a number of applications that do so. For instance, both TAP [5] and PiggyBank [9] seamlessly integrate scraping services into their data acquisition architectures. Another good example of the use of services is Magpie [6], which integrates services into its annotation mechanisms, by dynamically associating a highlighted item with all the services which are relevant to the type of the item in question. For example, when highlighting an instance of class 'researcher', Magpie could automatically retrieve all services it knows about, which make sense for a researcher. For instance, one such service could list all the projects that a researcher is involved in.

[2] http://en.wikipedia.org/

[3] http://www.flickr.com/

Conclusions

The main conclusion of the above analysis is that the growth of the Semantic Web has been promptly followed by changes in the way Semantic Web applications are developed. By analyzing and contrasting some older and newer systems, we have identified a set of features, which in our view will characterize the next generation of Semantic Web applications. In particular we observe that the latest Semantic Web systems are geared to take advantage of the vast amount of heterogeneous semantic data available online. Freed from the burden of creating their own semantic data, they concentrate on finding and meaningfully combining the available semantic markup. In our view this is not an accidental feature but an important indicator of a shift taking place from the first generation of Semantic Web systems to the next one. In a nutshell, next generation Semantic Web systems will necessarily have to deal with the increased heterogeneity of semantic sources.

Finally, Semantic Web applications will also tend to reflect the major developments in conventional Web systems, and as a result in the future we will see an increasing degree of integration of key Web technologies, such as social tagging and web services, in Semantic Web applications.

References

1. Berners-Lee, T., Hendler, J., Lassila, O.: The Semantic Web. *Scientific American*, 284(5):34 – 43. 2001.
2. Lee, J., Goodwin, R.: The Semantic Webscape: a View of the Semantic Web. In WWW '05: Special interest tracks and posters of the 14th international conference on World Wide Web, pages 1154--1155, New York, NY, USA, 2005. ACM Press.
3. Schraefel, M.C., Shadbolt, N.R., Gibbins, N., Glaser, H., Harris, S.: CS AKTive Space: Representing Computer Science in the Semantic Web. In Proceedings of the 13th International World Wide Web Conference.
4. Hyvönen E., Mäkelä E., Salminen M., Valo A., Viljanen K., Saarela S., Junnila M. and Kettula S.: MuseumFinland - Finnish Museums on the Semantic Web. Journal of Web Semantics, vol. 3, no. 2, pp. 25, 2005.
5. Guha, R. and McCool, R.: Tap: a semantic web platform. Computer Networks, 42(5):557--577, August 2003.
6. Dzbor, M., Motta, E., and Domingue, J.B.: Opening Up Magpie via Semantic Services. In McIlraith et al. (eds), The SemanticWeb - ISWC 2004, Third International Semantic Web Conference. Hiroshima, Japan, November 2004. Lecture Notes in Computer Science, 3298, Springer-Verlag 2004.
7. Popov, B., Kiryakov, A., Kirilov, A., Manov, D., Ognyanoff, D., and Goranov. M.: KIM – A Semantic Annotation Platform. In D. Fensel, K. Sycara, and J. Mylopoulos (eds.), The Semantic Web - ISWC 2003, Second International Semantic Web Conference. Lecture Notes in Computer Science, 2870, Springer-Verlag, 2003.
8. Lopez, V., Motta, E., Uren, V.: PowerAqua: Fishing thSemantic Web. In York Sure and John Domingue (eds.), The Semantic Web: Research and Applications, 3rd European Semantic Web Conference, ESWC 2006, Budva, Montenegro, June 11-14, 2006. Lecture Notes in Computer Science 4011, Springer 2006, ISBN 3-540-34544-2.
9. Huynh, D., Mazzocchi, S., Karger, D.: Piggy Bank: Experience the Semantic Web Inside Your Web Browser. In Gil et al. (eds), The Semantic Web - ISWC 2005, 4th International

Semantic Web Conference, ISWC 2005. Galway, Ireland, November 6-10, 2005. Lecture Notes in Computer Science, 3729 Springer-Verlag, 2005.

10. AKT Reference Ontology. http://www.aktors.org/publications/ontology/.
11. Lopez, V., Motta, E.: Ontology Driven Question Answering in AquaLog. 9th International Conference on Applications of Natural Language to Information Systems (NLDB 2004).
12. Schreiber, G., Akkermans, H., Anjewierden, A., de Hoog, R., Shadbolt, N., Van de Velde, W., and Wielinga, B. Knowledge Engineering and Management - The CommonKADS Methodology. MIT Press, December 1999.
13. Mika, P.: Flink: SemanticWeb Technology for the Extraction and Analysis of Social Networks. *Journal of Web Semantics*, 3(2), 2005.