Managing enterprise applications as dynamic resources in corporate semantic webs: an application scenario for semantic web services

Fabien Gandon, Moussa Lo, Olivier Corby, Rose Dieng-Kuntz

Managing enterprise applications as dynamic resources in corporate semantic webs: an application scenario for semantic web services.

Position paper for the W3C Workshop on Frameworks for Semantics in Web Service, June 9-10, 2005

Fabien Gandon, Moussa Lo, Olivier Corby, Rose Dieng-Kuntz
INRIA, ACACIA Lab., 2004 rt des Lucioles, BP93, 06902 Sophia Antipolis
Fabien.Gandon@sophia.inria.fr

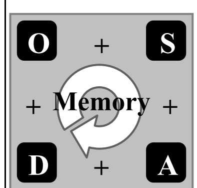
1. Corporate semantic webs

Semantically annotated information worlds are, in the actual state of the art, an effective way to make information systems smarter. If a corporate memory becomes an annotated world, corporate applications can use the semantics of the annotations and through inferences help the users use the corporate memory.

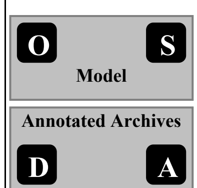
The ACACIA team at INRIA focuses on knowledge management solutions based on semantic Web technologies. As shown by the insert on the right [3], we use RDF Model, RDF Schema and OWL (essentially OWL Lite) to describe ontologies and implement knowledge models. Organizational entities and people are annotated in RDF and its XML syntax is used to store and exchange the annotations. This choice enables us to base our system on the W3C recommendations that benefit from all the web-based technologies for networking, display and navigation. This clearly is an asset for the integration to a corporate intranet environment that often relies on web technologies. Relying on W3C standards also enables us to integrate access to external sources in the corporate memory (e.g. digital libraries offering references in the application domain), interconnect parts of intranets to form extranets, generate focused portals for customized access (e.g. to address device independence, mobile access, etc.), etc. Clearly relying on open standards is important for effective knowledge representation and knowledge management solutions.

Our work resulted in the development of a semantic Web search engine (Corese [1]) enabling us to analyze, query and infer from descriptions in RDF(S)/OWL. CORESE implements a query language close to SPARQL [U] and a production rule language used to declare domain-dependent inference rules. Corese was tested with a variety of schemas such as the Gene ontology (13700 concept types). It also provides approximate search capabilities (vital to information retrieval systems) and comes with a semantic web server providing presentation capabilities to dynamically generate query interfaces and templates to render results.

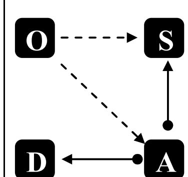
The Ontology, the Annotations and the State of Affairs form a virtual world capturing these aspects of the real world that are relevant for knowledge management.



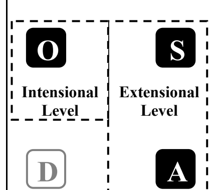
The memory is composed of the ***D***ocuments, their ***A***nnotations, the ***S***tate of affairs (user profiles and organization model) and the ***O***ntology. The whole follows a prototypical lifecycle, evolving and interacting with each other.



The ***O***ntology and the ***S***tate of affairs form the model on which is based the structuring of the memory. The archive structure relies on the ***A***nnotations of the ***D***ocumentary resources.



The ***A***nnotations and the ***S***tate of affairs are formalized using the conceptual vocabulary provided by the ***O***ntology.
The ***A***nnotations refer to the ***D***ocuments using their URI and the objects of the ***S***tate of affairs (e.g. document written by *Mr. Doe* for the *division Customer Service*)



The ***O***ntology defines modeling and annotation primitives at the intensional level. The ***S***tate of affairs and the ***A***nnotations instantiate these primitives describing models and annotations of the memory at the extensional level.

We can summarize our approach in three stages:

- To apply scenario-driven knowledge engineering techniques in order to capture the needed conceptual vocabulary. We then specify the corporate memory concepts and their relationships in an ontology and we formalize them in RDFS or OWL.
- To use the conceptual vocabulary of the ontology and the scenario analysis to develop corporate and user models. These models are implemented in RDF and instantiate the RDFS/OWL ontology description.
- To structure the corporate memory using RDF annotations on the documents: these annotations instantiate the RDFS/OWL ontology description and make reference to the corporate and user models.

Among the domain applications where we implemented corporate semantic webs and used Corese are:

- SAMOVAR: a system supporting a memory of vehicle projects for the car manufacturer Renault [5], and answering queries such as: “Find all fixing problems that occurred on the dashboard in a past project”.
- CoMMA: a multi-agent system for corporate memory management supporting the integration of a new employee and technological watch [3]. It answers distributed queries over distributed annotation bases such as “Find users who are interested in the technological news that was submitted about GSM v3”.
- KMP: a public knowledge management portal to cartography skills of firms in the Telecom Valley of Sophia Antipolis [8]. It answers queries such as: “Who are the possible industrial partners knowing how to design integrated circuits within the GSM field for cellular/mobile phone manufacturers?”.
- Life-line: a virtual staff for a health network [2] that guides physicians discussing the possible diagnoses and the alternative therapies for a given pathology, according to the patient’s features.
- MEAT: a memory of experiments of biologists on DNA microarray relying on automated annotation of scientific articles [6]. It answers queries such as ”Find all the articles asserting that the HGF gene plays a role in a lung disease”.

2. Corporate application management

Until the end of the 90's, enterprise modeling has been mainly used as a tool for enterprise engineering. But the new trends and the shift in the market rules led enterprises to become aware of the value of their memory and of the fact that enterprise model has a role to play in knowledge management too. Just like data-integration problem can benefit from corporate-level models, technology and application integration problem can benefit from these same models, and this was recognized by practitioners of Enterprise Application Integration. "*Organizations that are able to integrate their applications and data sources have a distinct competitive advantage: strategic utilization of company data and technology for greater efficiency and profit. But IT managers attempting integration face daunting challenges — disparate legacy systems; a hodgepodge of hardware, operating systems, and networking technology; proprietary packaged applications; and more. Enterprise Application Integration (EAI) offers a solution to this increasingly urgent business need. It encompasses technologies that enable business processes and data to speak to one another across applications, integrating many individual systems into a seamless whole.*" [9]

More and more often, the ACACIA team must face scenarios requiring not only knowledge access but also computation, decision, routing, transformation, etc. Until now, our corporate semantic webs focused on providing a unified and integrated access to a range of knowledge sources; but there is a growing demand to get the same facility to access corporate applications and services and to integrate both worlds.

Users expect IT managers to get very different computing systems (desktops, mobile phone, PDA, mainframes, etc.) to talk together and, even worth, to get the variety of applications that run on them to talk together. But what does it mean to talk together? Who talks to whom? What are the flows and processes? What are the purposes?

Users don't only want to get access to the needed pieces of information, they want it in a format they are used to, with some certification of quality or of provenance, with appropriate tools to analyze it, modify it, etc.

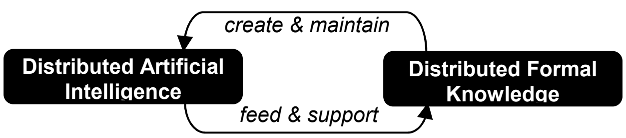
Usage scenarios are moving from a unified access to information to a unified access to information and applications. Corporate memories not only include information mediums but more generally:

- *information storage services* including: document sources (digital libraries, mailing-lists, forums, blogs, etc.) and dedicated systems (corporate or public databases, ERP, data warehouse, etc.);
- *information creation services* including: sensors (e.g. location tracking, presence & availability), computation and inference systems (e.g. data analysis tools);
- *information flows management services* including: secured transport channels, business rule engines and workflow systems, connectivity management, privacy enforcement and trust propagation;
- *information mediation services* including: matchmaking directories, translation and mapping services, contract and service quality enforcement;
- *information presentation services* including: multimedia transformation and translation, contextual adaptation, dynamic customization and manipulation interfaces;

All these services may be internal or external to the company yet users want them to interoperate smoothly and, even better, to automatically integrate their workflows at the business layer.

3. Corporate semantic web services

In the CoMMA [3] project we experienced with multi-agent architecture to provide distributed software architecture managing distributed memories. Societies of agents were dedicated to the management of the annotations and the ontology. We designed protocols sustaining the social structures of these groups of agents, in particular techniques for intelligently distributing annotations in the existing archives and for decomposing and routing queries to solve. This was our first experience with non-client-server distributed software architecture for a corporate semantic web and the association between distributed formal knowledge (semantic web) and distributed artificial intelligence (agents) proved to provide a very powerful paradigm for corporate memory management. [3]



Clearly, the evolution of web services towards semantic web services proposes an alternative to agent architectures and we are naturally investigating the extension of the Corese semantic Web Server to a distributed web platform relying on web services to deploy a flexible distributed software architecture that can match intranets structures.

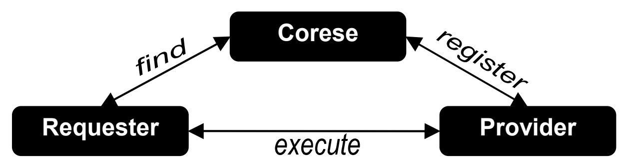
In the *my*Campus project [4] we experimented with a context-aware environment aimed at enhancing everyday campus life at Carnegie Mellon University (CMU). The environment revolves around a growing collection of task-specific online services (e.g. restaurant concierge, message filtering agent, etc.) capable of automatically accessing a variety of contextual information about their users. We introduced a Semantic Web architecture aimed at enabling the automated discovery and access of personal resources in support of a variety of context-aware applications. Within this architecture, each source of contextual information (e.g. a calendar, location tracking functionality, collections of relevant user preferences, organizational databases) is represented as a Web service. An e-Wallet acts as a directory of contextual resources for a given user, while enforcing her privacy preferences.

Web services are a standardized way of integrating Web-based applications using open standards over an Internet protocol backbone:

- *XML* technology is used to structure and tag data;
- *WSDL* is the Web Services Description Language for describing services and their programmatic interface;
- *SOAP* is the Simple Object Access Protocol, for remotely executing Web services;
- *UDDI*, Universal Description, Discovery, and Integration, is used to find required services.

Web services allow organizations to make public a programmatic access to one of their application without exposing the internal architecture of their IT systems. However, compared to agent-based platforms we used before, these technologies had the disadvantage to remain at the syntactic level while all the resources we manipulate are described in ontology-based models enabling us to leverage the semantics of descriptions in inferences.

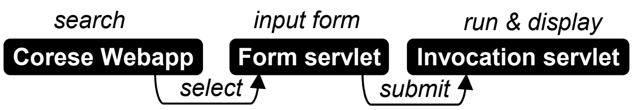
In the corporate memories we developed so far, the annotations generally describe documentary resources or corporate structures, but, when relying on schemata as the ones advocated in OWL-S [10], these annotations can describe web services available online (intranet, extranet, Internet). This means that Corese allows us to automate the identification of web services available to a user. Following a service-oriented architecture and a find-bind-execute schema [11] Corese nicely fits in the picture with semantic web services:



In this new architecture, we moved:

- from XML/WSDL structured descriptions to RDFS/OWL characterization of services: we use the profile and grounding of OWL-S plus the input and output description in the process description;
- from text-based UDDI search to the semantic search engine Corese to solve queries on the descriptions of the services, taking into account the ontologies used to characterize them.

Our current implementation is embedded in the semantic web server architecture and works in three steps: (1) we provide automatic discovery of web services using Corese queries upon their OWL-S annotations just as for other resources of the corporate memory; (2) when a service is selected by a user, instead of displaying the resource as it is the case for documents for instance, we dynamically generate a form from the grounding and the process providing an interface to call the service; (3) on submitting the form, the inputs are used to generate a dynamic client and the call to the web services. The output is then simply displayed as a web page.



This simple architecture already enables us to provide dynamic invocation of services without any prior knowledge of its description: Corese queries allow us to get the necessary information about a service from the knowledge base of service descriptions in order to dynamically invoke it. Here, in the corporate memory, Corese provides the equivalent of a corporate semantic UDDI registry.

The screenshot in Figure 1 shows two windows: (1) a window in the background corresponding to the generic search interface of Corese. It shows the result of a query where some services were found. One of them is the service "post" which provides an access to our ldap directory. The user selected this service and obtained (2) a window providing a form to specify the inputs of the service (here the name of an employee). Once submitted, this form triggers a call to the web service which is then dynamically executed and provides outputs displayed in the web interface (here the phone number of the employee).

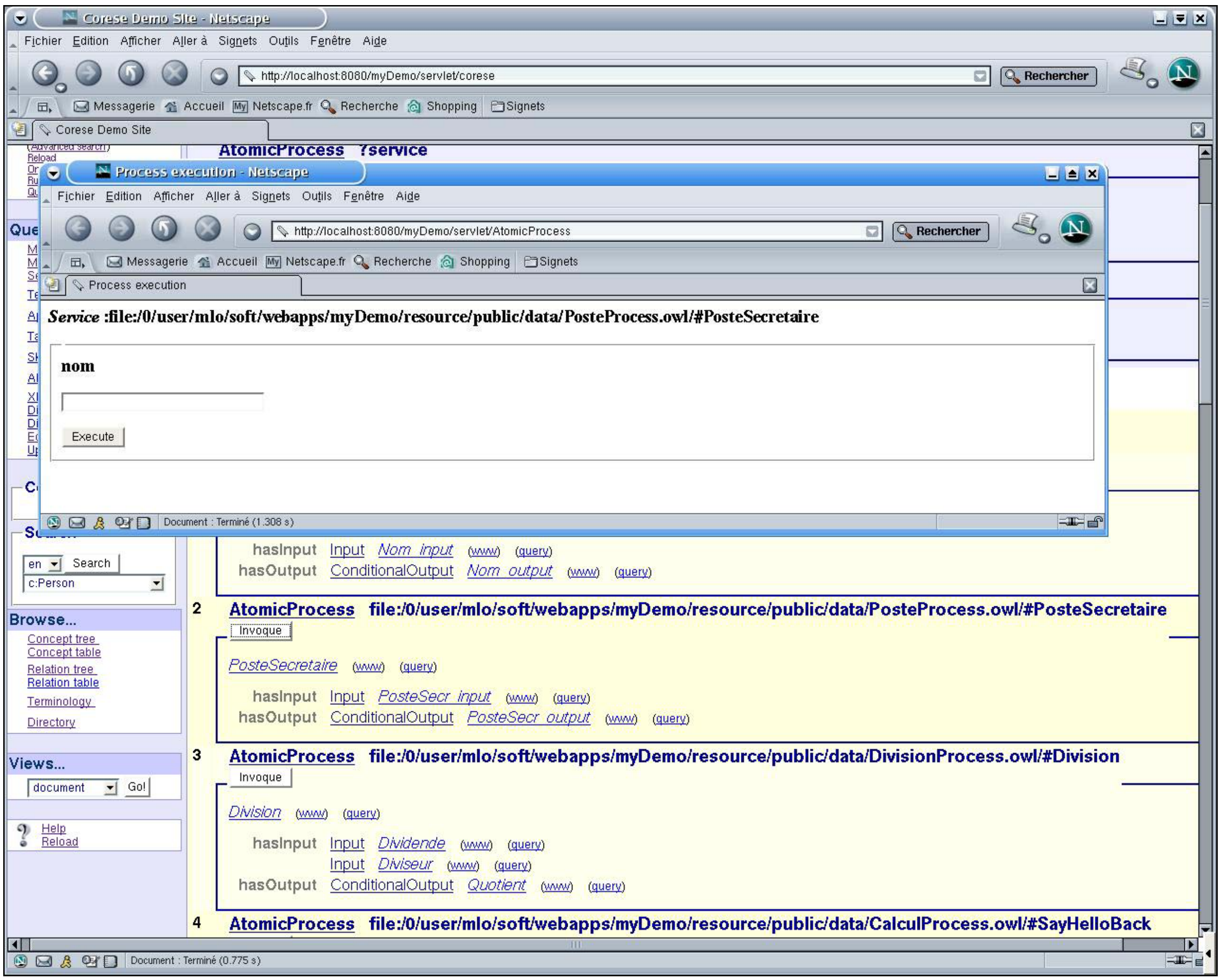


Figure 1: an example of discovering a corporate web service with Corese and dynamically invoking it.

4. Perspectives and future work

We are looking at several open issues and perspectives and especially two of them in the short term:

- *Composite service and choreography description*: to save the result of a composition allowing, for instance, IT managers to compose new services from existing ones and to publish them rapidly in their declarative form.
- *Manual vs. Semi-automatic vs. fully automatic composition and invocation* to provide high-level functionality through dynamic integration: how to describe and decompose service needs? In our exploration it seems to rely a lot on domain knowledge, and we think that, as claimed in [7], in many contexts users will want to control the composition process, influencing the service selection. Among the scenarios directly concerned here is the request from business managers to be able to implement business workflows in flexible (declarative) manners above the classical web services architectures.

These two first issues mean we are looking for a standardization of the works including: composite processes in OWL-S, WSFL (Web Services Flow Language), WSCI (Web Service Choreography Interface), WSCL (Web Services Conversation Language), XLANG (Microsoft BizTalk), WSMF (Web Services Management Framework), BPEL4WS (Business Process Execution Language for Web Services). Composition and choreography issues currently are the most symptomatic examples of the need for a standardization consortium to take the lead, and W3C definitively is the best candidate. We are witnessing a multiplication of contributions for each and every stage of the life-cycles of web-services and especially discovery and binding [12][13][14][15][19] and discovery and composition [7][16][17][18]. There is clearly a need to homogenize the different approaches before the differences hamper the foundations of Semantic Web Services such as interoperability.

In addition, our involvement in the semantic web deployment and our participation in the W3C working group on Semantic Web Best Practices mean we also are interested in:

- *Interaction and integration with emerging semantic web extensions* such as: SPARQL query language and protocol, SWRL rules description, etc.
- *User interfaces to semantic web services*: web services are primarily designed for B2B programmatic interactions but some services or compositions of services are finally called by users. Since their discovery, composition and invocation are dynamic this requires dynamically generated ergonomic user interfaces.

References

[1] Corby, O., Dieng-Kuntz, R., Faron-Zucker, C., Querying the Semantic Web with the CORESE search engine. In Proc. of the 16th European Conference on Artificial Intelligence (ECAI'2004), Valencia, 22-27 August 2004, IOS Press, p. 705-709

[2] Dieng-Kuntz, R., Minier, D., Corby, F., Ruzicka, M., Corby, O., Alamarguy, L., Luong, P.-H. Medical Ontology and Virtual Staff for a Health Network, EKAW2004, 2004

[3] Gandon, F., Distributed Artificial Intelligence and Knowledge Management: ontologies and multi-agent systems for a corporate semantic web, PhD Thesis in Informatics, 7th of November 2002, INRIA and University of Nice - Sophia Antipolis

[4] Gandon, F. and Sadeh, N., Semantic Web Technologies to Reconcile Privacy and Context Awareness, Web Semantics Journal. Vol. 1, No. 3, 2004.

[5] Golebiowska, J., Dieng, R., Corby, O., Mousseau, Building and Exploiting Ontologies for an Automobile Project Memory, K-CAP, ACM Press, 52-59, 2001

[6] Khelif, K., Dieng-Kuntz., R., Ontology-Based Semantic Annotations for Biochip Domain, KMOM Workshop ECAI2004 , 2004.

[7] Kim, J., Gil, Y., *Towards Interactive Composition of Semantic Web Services*, First International Semantic Web Services Symposium, AAAI, March 2004.

[8] KmP http://www-sop.inria.fr/acacia/soft/kmp.html

[9] Linthicum D. S., *Enterprise Application Integration*, Addison-Wesley Information Technology Series 1999, ISBN: 0201615835

[10] OWL-S 1.1 http://www.daml.org/services/owl-s/1.1/

[11] Qusay H. M., Service-Oriented Architecture (SOA) and Web Services: The Road to Enterprise Application Integration (EAI), April 2005 http://java.sun.com/developer/technicalArticles/WebServices/soa/

[12] Zein, O. K., Kermarrec, Y., An Approach for Describing/Discovering Services and for Adapting Them to the Needs of Users in Distributed Systems, , First International Semantic Web Services Symposium, AAAI, March 2004.

[13] Paolucci, M., Soudry, J., Srinivasan, N., Sycara, K., A Broker for OWL-S Web Services, First International Semantic Web Services Symposium, AAAI, March 2004.

[14] Decker, K, Sycara, K, and Williamson, M. Matchmaking and Brokering, In proc. of the Second International Conference on Multi-Agent Systems (ICMAS-96), The AAAI Press, 1996

[15] Paolucci, M,, Kawamura, T., Payne, T. R, Sycara, K.; Semantic Matching of Web Services Capabilities, In Proc. of the International Semantic Web Conference (ISWC'02), Springer Verlag, Sardegna, Italy, June 2002.

[16] Sirin, E., Parsia, B., Hendler, J., Composition-driven Filtering and Selection of Semantic Web Services, First International Semantic Web Services Symposium, AAAI, March 2004.

[17] Gomez-Perez, A., Gonzalez-Cabero, R., Lama, M., A Framework for Design and Composition of Semantic Web Services, First International Semantic Web Services Symposium, AAAI, March 2004.

[18] Sirin, E., Parsia, B., Planning for Semantic Web Services, workshop Semantic Web Services: Preparing to Meet the World of Business Applications, at The Third International Semantic Web Conference, Hiroshima, 2004

[19] Kifer, M., Lara, R., Polleres, A., Zhao, C., Keller, U., Lausen, H., Fensel, D., A Logical Framework for Web Service Discovery, workshop Semantic Web Services: Preparing to Meet the World of Business Applications, at The Third International Semantic Web Conference, Hiroshima, 2004