Architecture of a Semantic Portal on Mobile Business

Ilja Krybus, Karl Kurbel
Chair of Business Informatics
European University Viadrina
POB 1786
D-15207 Frankfurt (Oder)
{krybus|kurbel}@uni-ffo.de

Abstract: Portals on the web are important public sources of information for expert knowledge. They function as powerful gateways that consolidate access and organize information. Existing web technologies provide the means which most current web portals apply. However, they leave some open issues that recent Semantic Web technologies promise to solve. Portals that employ semantic technologies are called semantic portals. In this paper, we present the synopsis of a semantic portal that is dedicated to distributing practical and scientific knowledge on the domain of Mobile Business. We explain the motivation, the architectural considerations, and the current portal prototype. Emphasis is placed on ontology use, request processing, and presentation. The advantages of our process-oriented and multilayered architecture approach are discussed.

1 Introduction

The internet has become one of the most frequently utilized sources for acquiring topical information. Especially information on technologies and other innovations is often available on the internet long before it is printed in journals or books. It may even be exclusively disseminated through the internet. As a consequence, strong dependencies from this source emerged for everybody who relies on such information.

The more one relies on this source, the stronger is the exposure to the problems that arose with the vast and growing amount of information available, such as the distribution of resources, the increasing amount of time needed to locate valuable information, and the limited relevance of results returned by web-searches.

A domain, for which these problems became recently very obvious, is Mobile Business, which can be understood as a subset and descendant of Electronic Business. With the maturing and spreading of corporate systems that are based on mobile technologies throughout the business landscape, a lot of information has been published online. Nevertheless finding relevant information on Mobile Business remains difficult for the said issues.

The concept of portals allows concentrating information for a selective area of interest. A prerequisite for distributing information using portals is to prepare and systematize it for effective use. When doing this, the relatively limited semantic breadth of the considered domain bears an additional challenge. This constraint implies that information managed by the portal shares the same or similar terminology, and that semantically distinct concepts may be similarly verbalized. As a consequence, information becomes hard to discern by syntactical means, despite of its even significantly different meaning.

Emerging semantic technologies promise remedy on such issues. They are designed to enable machines (e.g., Intelligent Agents) to locating and applying contents and services on the web by providing them with the semantics of the available items. An implication of these technologies, i.e. the semantics are applicable to provide added values to users, is of special interest to solving the existing problems of information supply. Semantic technologies address the meaning rather than plain syntax and allow a more precise treatment and selection of information.

Within the scope of an interdisciplinary joint project on Mobile Business[1] we are developing a semantic portal (the Mobile Internet Business Portal) that will offer consolidated access to practical and scientific knowledge in the regarding field.

In this paper, the state of our current work is described: in the remainder of this section we explain our viewpoint on portals and semantic portals in particular. In section 2 related research on semantic portals is explored. Section 3 covers the portal's architecture, as well as the design considerations that led us to it. It also contains descriptions of selected subsystems of the portal. Emphasis is placed on ontology use, request processing and adaptive presentation of contents (in general and tailored for mobile devices). Section 4 concludes this paper.

1.1 Portals

There are many divergent definitions for portals. Definitions vary between functional and technological foci, and range from describing structured websites to complex information systems. According to the definitions summarized in [De05] portals should be considered web-based application systems, or, "system[s] of integrated programs". In [Ka01] (community web) portals are defined as systems that "essentially provide the means to select, classify and access various information resources (e.g., sites, documents, data) for diverse target audiences (corporate, inter-enterprise, e-marketplace, etc.)." As summarized in [LW05], portals form "a gateway to the web that allows the plethora of information […] to be organized and customized through a single entry point", and are "used to consolidate information from a vast array of resources."

An evaluation [Kr06] has shown that the salient property of portals, i.e. to offer a single point of access, has two major implementations. Portals appear either as self-contained systems that encompass all provided services and contents themselves, or as hubs that collate external resources. With portals like Semantic-

---

[1] This research is supported by the German Federal Ministry of Education and Research under the grant No. 01AK060A.

Web.org[2] (in its current design), however, a third kind of portal has emerged: portals, which integrate sets of community managed RDF statements, i.e. a multitude of assertions about facts.

1.2 Semantic portals

Semantic portals, in a nutshell, are portals that make use of Semantic Web technologies. They "exploit semantics for providing and accessing information" [Ma03], and they "typically provide knowledge about a specific domain and rely on ontologies to structure and exchange this knowledge" [HS04]. Semantic Web technologies are applied to "constructing and maintaining the portal" [Ma03] as well. The degree and the focus of technology usage, however, varies. Examples are given in the next section.

We suppose that semantic portals could take the position of central building blocks in constructing the Semantic Web [Kr06]:

By applying semantic technologies, they demonstrate the value of these technologies to a potentially large audience. Because they are reaching many customers, portals could be employed for popularizing ontologies and establishing naming conventions (e.g., for named entities or domain specific taxonomies) across the internet. New ontologies could be collaboratively elaborated within semantic portals (cf., e.g., [Zh04]).

As was expressed in [Mc05], adding semantic descriptions to contents significantly increases the efforts spent in designing information bases. This observation is especially true for small collections of information, for which the ratio of ontology utilization versus its elaboration efforts is seemingly poor. When portals handle rather large collections this ratio becomes more attractive and the said obstacle less decisive.

Interconnecting portals seems to be more efficient than interconnecting diverse small internet resources, since the number of necessary ontology mediations is dramatically reduced. Not least, semantic portals may mediate between the Semantic Web and the current web by wrapping non-semantic contents with their ontologies thus raising the amount of information that can be located and processed exploiting semantics.

2 Related work

Several applications of semantics are already widely disseminated, e.g. RDF Site Summary[3] for distribution of news or blog-entries, Friend-of-a-friend or vCard profiles for communicating contact data, and Dublin Core Elements for bibliographic data. Research regarding semantics has been performed in a broad spec-

---

[2] http://www.semanticweb.org

[3] http://{web.resource.org/rss/1.0|www.w3.org/TR/vcard-rdf|www.foaf-project.org}

trum, including acquisition, reasoning, searching, mediation, visualization of ontologies, and other topics. Because of the integrative nature of portals, all this is of certain interest. In particular, research that already resulted in a prototype or a production stable semantic portal is a good reference. Important representatives of such portals are (among others) SEAL/OntoWeb, ODESeW/Esperonto, DERI/SW-Portal, and SWAD-E/SWED[4].

The OntoWeb portal [HS04] [Sp02] is an extension of the SEAL framework [Ma03] that serves as a communication and dissemination system for a thematic network. It defines several content types (e.g. person, organization) and content structures using concept definitions in an externally managed ontology. Portal navigation is derived from modeled subclass-relations, and search is provided on full text and instance properties. The portal is based on the ZOPE application server and its content management system.

Esperonto portal [Co03] emphasizes that it is a semantic intra- and extranet portal, which is used to disseminate the results of the Esperonto project. It combines multiple ontologies, which define the type and structure of information that in turn is stored directly within ontologies. Concept subclass-relations are used for organizing content. In this solution two front-ends are differentiated, a web-based ontology editor and the portal site. Information is stored in a database, which is accessed through a dedicated ontology API.

The SW-Portal originally served as community portal within the DERI research network [Zh04a]. After a re-launch[5], it is now billed as a "public entry point to access semantic web related information" [Zi04]. Compared to other portals, users are much more involved in extending the portal's ontologies [Zh04]. According to [Zh04a], in the SW-Portal servlets access the Jena framework[6] through intermediate services.

The Semantic Web Environment Directory (SWED), a meta-directory, was built as a demonstration to illustrate the nature of the Semantic Web [Re04]. It combines (partially) automated content acquisition with the ability to create and annotate resources locally. Instead of static subclass-relations, a dedicated ontology[7] is applied to the categorization and inter-linkage of content. The portal's user interface supports facetted browsing. Contents are presented using a template-engine that recursively locates templates which have been directly assigned to each concept.

Other semantic systems function as "portal generators" for annotated web content [Hy05], or consolidate access to traditional websites [AP05]. A detailed comparison of four semantic portals (OntoWeb, Esperonto, Mondeca ITM, Empolis K42) is documented in [La04]. An example for a different utilization of semantic technology in portals is the support of inter-portlet-communication [Di05] [PP03].

---

[4] http://{ontoweb.org|www.esperonto.net|sw-portal.deri.org|www.swed.org.uk}

[5] now SemanticWeb.org

[6] http://jena.sourceforge.net/

[7] http://www.w3.org/TR/2005/WD-swbp-skos-core-spec-20051102/

Taken together, existing portals have proven the feasibility, potential, and advantages of the semantic approach over traditional web technologies.

Seen from an architectural viewpoint, most semantic portals (like other portals) implement the popular 3-tier concept. On the data-tier, the structurally fixed database (or content management system) is usually replaced with an ontology store. On top of this, more valuable semantic search and/or navigation features and metadata enriched presentation are realized. Internal component interconnection frequently seems to be rather "hardwired". At least documentation reveals a certain shortage of more flexible component coordination and processing control. It is also remarkable that some portals are even almost excluding content produced for the traditional web. As a result, traditional content is not directly managed within the portals. Web front-ends often consist of generic templates (e.g. table based views). Adaptive presentation of content and support of access from mobile devices remain open issues.

3 The MIB approach to semantic portals

Summarizing the first two sections, the core requirements for semantic portals are:

- to store, organize and to manage semantic information (including content, meta-information/annotations, relations and structures),
- to integrate "traditional" web-content with semantic information,
- to provide means to semantically access content and to provide this access independently from criteria that are implied in the content (namely the terms used within the content and its specific syntax),
- to supplement semantic applications, e.g. automated content acquisition [Re04] or exchange between portals [HS04] (not dealt with in this paper),
- to present contents adopted to the context of information requests.

In conclusion of section 2 more attention should be paid to the portal architecture by optimizing it for maintenance, extendibility and reusability.

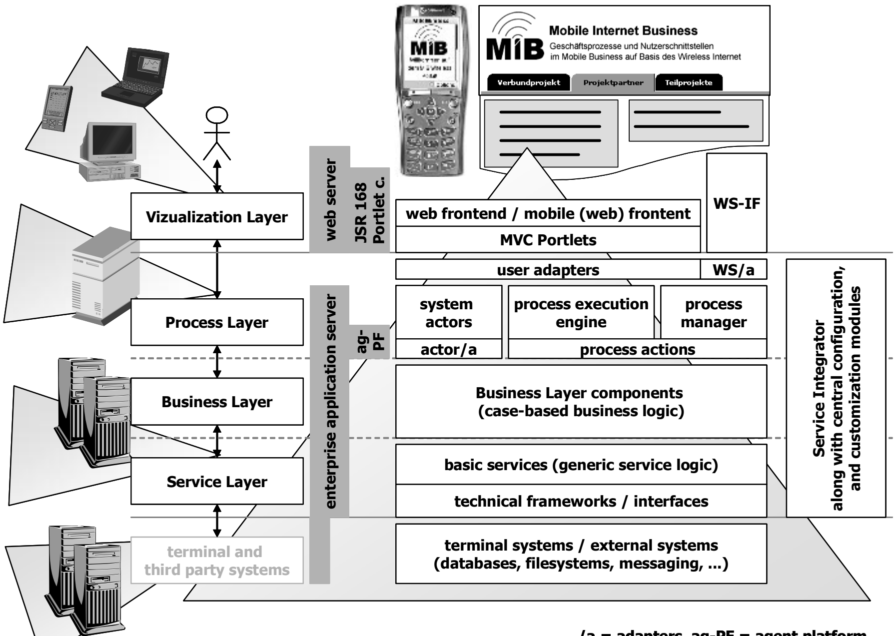
The MIB Portal is developed to meet these requirements. It exploits ontologies for structuring information and for facilitating internal operations. Lessons learnt from successful solutions are taken into consideration.

3.1 Architectural principles

Portals are integrated systems, open in nature, and may thus face rapidly changing demands. A portal that was originally deployed in a certain configuration will hardly remain unchanged over time. In order to provide the necessary degree of adaptability, the portal should be flexible in configuration and design, highly modular, and consist of rather self-contained components, which are loosely coupled to perform requested actions. For cost-effective maintenance and development high reusability of existing components is desired.

Standard-compliant programming, layering, utilization of Design Patterns [Ga97] or abstraction libraries/frameworks can increase the level of adaptability. Aside from these measures, a closer look at possible reasons for changes could reveal further opportunities to create sustainable systems: the need to change may occur in response to modification of (1) technical specifications, (2) use-cases/features, (3) processes and/or information flows, (4) visualization and interaction needs.

As a consequence, to minimize the extent of necessary changes, we decided to replace the common 3-tier-architecture in the portal with a different layering scheme. In this scheme, four dedicated layers are distinguished: Service (SL), Business (BL), Process (PL), and Visualization Layer (VL). (A coherent approach, Quasar, which considers two of the layers listed above, is explained in [Ha05].) The architecture is depicted in figure 1.



/a = adapters, ag-PF = agent platform,
WS-IF = web service interfaces, Portlet c. = Portlet container and portal

Figure 1: Mobile Internet Business Portal architecture overview: physical and logical component distribution by applied four layer approach

The SL contains all components whose tasks are mostly technology oriented, rather generic, and independent from concrete business logic. Some examples are database persistence, file management, generation of system-wide unique identifiers, and delivery of asynchronous messages. Interfacing third-party systems is another important task for SL components. The SL components increase the granularity of interfaces and data. Inquirers will not have to deal with elementary, technology specific items. They will work on compound documents, rather than on system-specific blocks; on frame-like objects, rather than on loose sets of RDF statements, etc. SL components can internally optimize system access and emulate missing functionality, if external systems do not provide it.

At the BL, components encapsulate all logic relevant to performing user tasks, abstracted from specific technologies. Custom treatment of documents or personalization may serve as example. In contrast to the SL, significant knowledge on both information and its meaning is exploited. Information is dealt with by case-based suppositions. Implemented logic is specialized to solve single facets of specific (business) use-cases. However, the services are not interconnected to processes but remain isolated and atomic.

The orchestration of services, interconnection and sequencing, is done at the PL. Declarative process models replace the “hardwired” programming. As a result, substitution or extension of actions performed within the process becomes very straightforward. E.g., an existing retrieval process (locate, load, extract, present information) may be enhanced by declaring personalization (e.g. by information filtering) an intermediate step in the process model without reprogramming existing modules. Changes can be applied even online at the serving portal. Variants of existing processes may be easily deployed. Another option is enabling dynamic definition of processes, in which successive actions are inferred within specialized process nodes (i.e. specialized “process actions” in figure 1) that evaluate context and semantic information that resulted from the current action.

Parameters, results, and initialization data make up the process context. Services within a process communicate via this context. Decisions to branch are realized by configurable nodes that consider the context too; services are thus not directly involved. In principle, the context of each process can be persisted on any event. If a user-controlled process is disrupted (e.g., in a mobile scenario: disconnection along transmission path, suspended work due to situational needs), work can be resumed.

Beyond coordination of services, the PL is responsible for integrating users into the portal. If user interaction is expected, the process switches into a wait-state. A view (e.g., state information, document output, web-form) is chosen based on the current process context. The user’s options are derived from the declared process model.

The VL implements the user interface. It adopts the MVC pattern: the model (e.g. a DOM or ontology instance) and state information are retrieved from the process context, and the control delegates processing to the portals backend. Only tasks that are specific to presentation (e.g. zoom-in, breadcrumb-navigation) are completely handled within this layer. In addition to state-based views other views are still allowed.

3.2 Use of ontologies

The portal supports complementary ontologies for annotating, classifying and defining content.

An ontology is employed as system of taxonomies (categorization bodies distinct from subclass-relationships) to externally describe contents, which are stored in or linked with the portal. This descriptive ontology adopts the Simple Knowledge Organisation System (SKOS)[8] which was designed to organize knowledge by using controlled vocabulary. Several technical taxonomies describing facets of the domain are transformed into SKOS. For example, terms and structure of the European Nomenclature of economic activities (NACE)[9] statistical taxonomy are translated into concepts and broader/narrower-relations of SKOS for expressing industrial sectors.

When publishing their contributions, authors can select categories to assign to their content. From these assignments, semantic navigation and search can be derived, and semantic proximity of contents is detected. Annotating content with category assignments, however, remains optional in the portal. Detailed annotation might enhance a contribution's placement/ranking, e.g., in searches. Nonetheless, all contents, including those which are not annotated, are accessible.

Ontology also defines properties and rules for structured information; i.e. concepts like "announcement of a conference", "description of a project" or "profile of a person". Some of them originate from public ontologies like vCard or FOAF, while others have been derived from typical documents (e.g. the conference-announcement concept from calls-for-papers).

Concepts may be instantiated and directly stored with (a specific sub-) model. As a result, instances are described through their contents, optional categorization, and the knowledge about their structure. This knowledge can be used to interpret or to adapt information towards the user's situational needs. While categorization allows selecting relevant contributions, structural knowledge enables picking information items particular to a given context. In the case of multi-access (e.g. supplementary support of mobile devices), the rather verbose output sent to a web client can be reduced to a summary of the most important facts.

Traditional web content (web-pages/sites, documents, images, etc.) may be "wrapped" by ontology instances. The latter will then reference either external resources or documents that are stored in the internal content management system. In both cases, the semantic wrapper confers annotation (mainly Dublin Core and relations to categories) to these resources.

The provision of such external meta-information allows jointly managing semantic and traditional web content within the same portal-engine. Therefore, all contents of the portal can be registered in the ontology. They are accessible through ontol-

---

[8] http://www.w3.org/2004/02/skos/

[9] http://europa.eu.int/comm/eurostat/ramon/

ogy queries, which may (but do not necessarily need to) facilitate inferred statements besides the asserted ones. Asserted and inferred statements are stored in separate models.

Beside organization and storage of content, the portal applies ontologies in order to perform several tasks concerning service-location, processing, and presentation of information. Among these tasks are ontology-based rendering and information streamlining for mobile access (which both are covered in consecutive subsections).

3.4 Request processing

At process-oriented portlets, incoming action requests are differentiated: if the request is constrained to the VL then it is handled within the portlet itself. Otherwise, a related process instance is acquired through a process-manager. Alternatively, new process instances may be created. Request parameters are transformed to a serialized RDF-representation and communicated to the process's context. Depending on actions associated with the request, the portlet may cause the process to transit to another state.

In a transition, declared BL components are obtained by name using the Service Integrator, which also considers the semantic role and location of a service inquirer. Methods of the obtained components are executed parameterized by the process' context. The processing results are then stored back to it.

From here to the next wait-state (e.g., a user-interaction) the process drives itself. The request is typically served in a synchronous operation, which is sufficient for most user-controlled functionality on a web portal. Optionally, processes can be controlled asynchronously by internal or external "system actors" like Intelligent Agents or message-driven systems.

Render requests do not cause transitions and do not necessitate parameters to be forwarded to the backend-system. A portlet's view method therefore retrieves the current (wait-) state from the associated process instance and selects a view module that is assigned to or inferred for the given state.

3.5 Ontology-based rendering

A particular subsystem is used for the presentation of contents that are stored as instances of the ontology. Within this system, we deployed templates that adaptively layout and arrange information for each considered concept (cf. [Re04]). Due to this specialization, it becomes easy to highlight important information and to group coherent facts. Generic views are yet supported within the development environment only. Reasons to limit use of generic views are enforced cutbacks on creative design as well as the focus on an audience, whose experience is formed by traditional web portals.

Templates are chosen according to an instance's concepts (classes) and with respect to their relations and optional context (e.g. the requesting client-system). If there are no templates provided for the known direct concepts, a meaningful substitute template is proposed as the result of an analysis of concept relations. Generalization/specialization (subclass) relations are the natural choice for such proposals. The mechanism, however, is not restricted to them.

Most concepts defined within our ontology declare literal and object properties. Object properties in turn can have multiple or very generic concepts as a range. As a consequence, template development could become difficult. In response to this issue, our template-mechanism uses recursion if object properties have to be rendered. Thus, e.g. rendering an address (that is modeled as an object property of an organization) will be automatically delegated to an appropriate template, which simplifies specific treatment of domestic vs. international (or other) addresses. The system circumvents inadvertently modeled cycles and unnecessarily deep nesting. Supplementary context parameters are used to control the output from the recursive templates.

Additionally, we distinguish two types of instances at the presentation stage that are indicated by object properties: autonomous and implicit instances.

All instances that are considered individual portal content, and which are comprehensive even without being referred to by the object property, are handled as autonomous instances. They are provided with a public identifier and are locatable using the portal's browsing or search facilities. Examples are conference announcements, organization profiles and thematic documents. For such instances, there is no need to incorporate full information. Rather, they are usually presented as links within the portal. (In figure 2, the instances linked by the hasOrganizer-property would be treated as autonomous instances.)

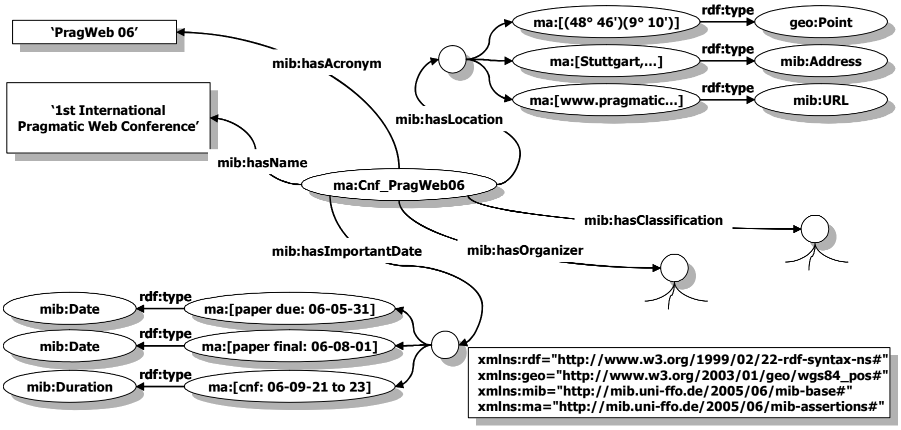


Figure 2: A ‘conference announcement’ instance references distinct concepts to ‘hasLocation’ and ‘hasImportantDate’ object properties, which are automatically considered, e.g., when sorting instances.

Implicit instances were defined to simplify information input, search and filtering as seen from the user’s viewpoint. Inspired by the object-oriented principle of composition, they are created to hold information which is meaningless in isolation, i.e. without being additionally described by the semantics of a referring object property. Dates, durations, and addresses are examples of this instance type: when they are presented without a context (e.g., a date not in relation to an instance of a conference announcement) then they are barely of any value to an agent. Such instances are never displayed without the surrounding context. Depending on the context, they will be presented within the referring autonomous instance in detail, summarized, as a list-entry, or as another layout alternative.

Because it is frequently required to output information as lists or tables, the ontology-based rendering subsystem is extended with a developed framework that allows instances to be meaningfully sorted and grouped. Supported instances can be ordered with respect to a chosen criterion, even if they belong to distinct concepts. For example, if a “conference announcement” instance with the object property “important dates” is assigned dates (06-08-01) and (06-12-31), and duration (06-09-21 to 06-09-23), the system will return (06-08-01), (06-09-21 to 06-09-23), (06-12-31) as a meaningful timeline. If instances are not comparable by the criterion, the ordering by classes or instance identifiers is used as default fallback rule. An instance that cannot be compared would be placed below the resulting schedule. Before this final option is chosen, a resolution mechanism similar to the one used for template-selection is attempted.

3.6 Mobile access

The portal supports access by mobile devices such as PDAs, and smart phones.

Mobile devices are discriminated by evaluation of client-signatures, which are transmitted with the headers of HTTP-requests. Capability profiles are then loaded with respect to the client from a profile database (in first priority: extended WURFL[10]) in order to choose presentation level protocols (WML, X-HTML MP, HTML) and to parameterize further rendering (support of images, width of display etc.).

Mobile users are offered a tailored subset of the entire portal’s functionality. The navigation is flattened and adapted to the special characteristics of mobile devices. The ontology-based rendering system is reused in order to streamline information presented for ontology instances.

The presentation of mobile content is special in the portal in that the presented information is pre-selected and reduced to the most important facts. This output-

---

[10] http://wurfl.sourceforge.net

streamlining is based on assumptions that have been made on the value of information items and which are considered for the related templates. As an alternative to this approach, we are experimenting with content priorities that are directly modeled within the ontology itself in order to choose the important items more adaptively, regarding the user's context.

Another option which is experimentally implemented and addressed for the use in mobile devices is the support of (simultaneous) multimodal access to information, i.e. in the case of the portal's prototype a combined visual and aural interaction on the basis of the XHTML+Voice [Ax04] standard proposal.

In a multimodal environment especially the aural interfaces bear new and yet unsolved challenges: aural interfaces are always fully serialized; users can listen attentively only for a relatively short amount of time; not all visualized information remains meaningful if it is spoken; and sometimes information must be extended to become comprehensible when it is spoken. While the user perceives visual information in an implicit context (e.g., information is grouped and laid out on the display), this implicit context is almost completely lost after serialization. The missing context must thus be added by transforming displayed facts into a set of valid and interconnected sentences.

Figure 3: Semantics applied for multi-device and multimodal output

Ontology-based rendering addresses some of these issues. In the prototype, the missed context is supplemented applying the modeled semantics for the content. E.g., in an instance of a conference announcement the conference location is bound to the event by inserting an appropriate transition (cf. figure 3), which is derived from the related object property. Content, which a screen reader application would usually misinterpret (e.g., a duration would be read as a formula), is spoken correctly.

4 Conclusion and future work

In an interdisciplinary project, we develop a semantic portal within the Mobile Business domain to meet the requirements summarized in the preface to section 3. As in other semantic portals, the MIB Portal uses ontologies for annotation, management, and for storage of contents. Additionally, ontologies ground internal decisions on portal operations like the presentation, which is implicitly controlled by the modeled knowledge. With our approach, semantic portals are extended by introducing a dedicated layer that coordinates automatic and user-driven processes. This allows to flexibly modifying and extending the portal's functionality. In many cases even seamless upgrades are possible by parallel operation of different process versions. The differentiation of the portal's modules with respect to the proposed four layer model reduces the impact of changed specifications, and thus, maintenance efforts. High reusability is achieved since the existing modules can be recombined by declaration rather than reprogramming to serve new tasks.

Aside from rich technological advantages for portal operation and development that the semantic approach provides jointly with the proposed four layer architecture as were summarized above, the key values are achieved for portal users. The combination of the portal concept with semantic technologies bears the potential to provide systematic access to knowledge in selective domains. At least within the portals, the application of semantics reduces information related issues that arise from the need to locate relevant information in huge collections.

The current work has concentrated on the technological foundations of the portal. Future work will include transformation and customization of the portal for the production phase. Supplementary means to semantically access content and further semantic applications will be investigated and implemented. Although rather flexible rendering mechanisms are already realized (cf. sections 3.5 and 3.6), a higher degree of adoption to context, mobility and multimodality is still appreciable. They will be addressed in future research.

References

[AP05] S. Auer, B. Pieterse. Vernetzte Kirche: Building a Semantic Web. In *Proceedings of ISWC Workshop Semantic Web Case Studies and Best Practices for eBusiness* (SWCASE05), 2005.

[Ax04] J. Axelsson, et al. XHTML+Voice Profile 1.2, 16 March 2004, available at http://www.voicexml.org/specs/multimodal/x+v/12/.

[Co03] O. Corcho et al. ODESeW. Automatic Generation of Knowledge Portals for Intranets and Extranets. In: Fensel, D. et al. (eds.): *Proceedings of The Semantic Web - ISWC 2003, Second International Semantic Web Conference*, LNCS, Vol. 2870, Springer (2003) 802-817.

[De05] V. Devedžić. Research community knowledge portals., In: *Int. J. Knowledge and Learning*, Vol. 1, Nos. 1/2 (2005), pp. 96–112.

[Di05] O. Díaz et al. Improving Portlet Interoperability Through Deep Annotation. In: *Proceedings of the 14th International World Wide Web Conference* (WWW 2005) 372-381

[Ga97] E. Gamma et al. *Design Patterns. Elements of Reusable Object-Oriented Software.* Addison-Wesley Professional (1997).

[Ha05] M. Haft et al. The Architect's Dilemma – Will Reference Architecture Help? In: Reussner et al. (eds.): *Quality of Software Architectures and Software Quality,* *Proceedings of QoSA-SOQUA 2005*, LNCS, Vol. 3712, Springer (2005) 106-122.

[HS04] J. Hartmann, Y. Sure. An Infrastructure for Scalable, Reliable Semantic Portals. In: *IEEE Intelligent Systems*, 19(3): 58-65, 2004.

[Hy05] E. Hyvönen et al. MuseumFinland -- Finnish Museums on the Semantic Web. In: *Journal of Web Semantics*, Vol. 3, No. 2 (2005), pp. 25.

[Ka01] G. Karvounarakis. et al.: Querying RDF Descriptions for Community Web Portals. In: *Proceedings of 17iemes Journees Bases de Donnees Avancees* (BDA'01), (2001) 133-144.

[Kr06] I. Krybus. Wissensportale im Web. In: *Arbeitsberichte Mobile Internet Business*, Nr. 4, April 2005, ISSN 1861-3926.

[La04] H. Lausen et al. *Semantic Web Portals – State of the Art Survey*. Technical Report DERI-TR-2004-04-03, Digital Enterprise Research Institute (DERI) (2004), available at http://www.deri.at/publications/techpapers/documents-/DERI-TR-2004-04-03.pdf.

[LW05] S. Li, W.A. Wood. Portals in the Academic World: Are they Meeting Expectations? In: *Journal of Computer Information Systems*, Summer 2005, pp. 32-41.

[Ma03] A. Maedche et al. SEmantic portAL - The SEAL approach. In: Fensel, D. et al. (eds.): *Spinning the Semantic Web*, MIT Press, Cambridge, MA (2003) pp. 317-359.

[Mc05] R. McCool. Rethinking the Semantic Web, Part I. In: *IEEE Internet Computing*, Nov./Dec. 2005, pp. 86-88.

[PP03] T. Priebe, G. Pernul. Towards Integrative Enterprise Knowledge Portals. In: *Proceedings of the 12th Intl. Conference on Information and Knowledge Management* (CIKM 2003).

[Re04] D. Reynolds et al. *Semantic Portals Demonstrator – Lessons Learnt*. SWAD-Europe deliverable 12.1.7, available at http://www.w3.org/2001/sw/Europe/-reports/demo_2_report (2004).

[Sp02] P. Spyns et al. OntoWeb - A Semantic Web Community Portal. In: Karagiannis, D. and Reimer, U. (eds.): *Proceedings of Practical Aspects of Knowledge Management, 4th International Conference*, PAKM 2002, LNCS, Vol. 2569, Springer (2002), pp. 189-200.

[Zh04] A.V. Zhdanova. The People's Portal: Ontology Management on Community Portals. In: *Proceedings of the 1st Workshop on Friend of a Friend, Social Networking and the Semantic Web* (FOAF'2004), pp. 66-74.

[Zh04a] A.V. Zhdanova et al. SW-Portal Prototype: Semantic DERI Use Case. Deliverable 15, 2004-08-31, *Digital Enterprise Research Institute* (DERI) (2004), available at http://sw-portal.deri.org/papers/deliverables/d15.pdf.

[Zi04] K. Zimmermann. *Usage Scenarios* (Relaunch SW). Deliverable 20 v0.3, 15 February 2005, Digital Enterprise Research Institute (DERI) (2004), available at http://www.deri.at/research/projects/sw-portal/2004/d20/v0.3/20050215/.