The architecture and business model of the Fraunhofer FIRST/IGD knowledge management portal

Ronald Melster[1], Christian Storl[2]

[1]Fraunhofer FIRST,
Ronald.Melster@first.fraunhofer.de
[2]IT Service Omikron GmbH
storl@itso.de

Abstract. This paper describes a web-based knowledge management system for editing and visualising facts in a new innovative way. The foundation of the portal is a commercial framework for building information systems. Furthermore, the XTM topic map standard is used for representing the facts and their relationship in the system. In this paper, the overall architecture of the framework and the features we used are described. In the next chapter the representation and modelling of the XTM topic map standard is depicted in detail. In the last chapter the new three-dimensional user-interface is shortly illustrated.

We are on the edge from information society to knowledge society. Knowledge has been identified as an critical resource in the global competition between enterprises. Businesses increasingly sell information, knowledge, intelligent products and services. Therefore, capturing and sharing the limited resource knowledge becomes a vital task for any enterprise.

The knowledge management portal of Fraunhofer FIRST and IGD is a knowledge management system with a new and innovative interface for visualising knowledge elements and indicating the relationships between them. The approach goes far beyond the usual systems limited to HTML-based presentations.

The Knowledge Management portal by Fraunhofer FIRST and Fraunhofer IGD is used for collecting, editing and presenting knowledge in different views depending on the level of existing expertise of the user. Therefore, a system had to build

- with different dynamic views on the same set of information,
- with workflow support for the authoring process as well as for the processing of working operations,
- with easy publication in different file formats
- with knowledge map support – importing, exporting and editing knowledge maps, and
- which should be conformant to existing standards.

We decided to use the information management framework bluedot by IT Service Omikron as this framework provided most of the functionality needed and was flexible enough to integrate the collateral functions. In the next section we will briefly present the overall architecture of the bluedot framework and the process to instantiate the knowledge management portal. The chapter "1 Architectural overview" will present the structure of the framework on which the knowledge management portal is based.

1 Architectural overview

The content management system framework bluedot [BL00] we introduce, is a component-based information technology platform for running advanced information systems and corporate web sites across Inter- and Intranets.
Frameworks implement most of the functionality needed to run a system. Application-speficic functionality has to be added to augment the framework to a complete system.

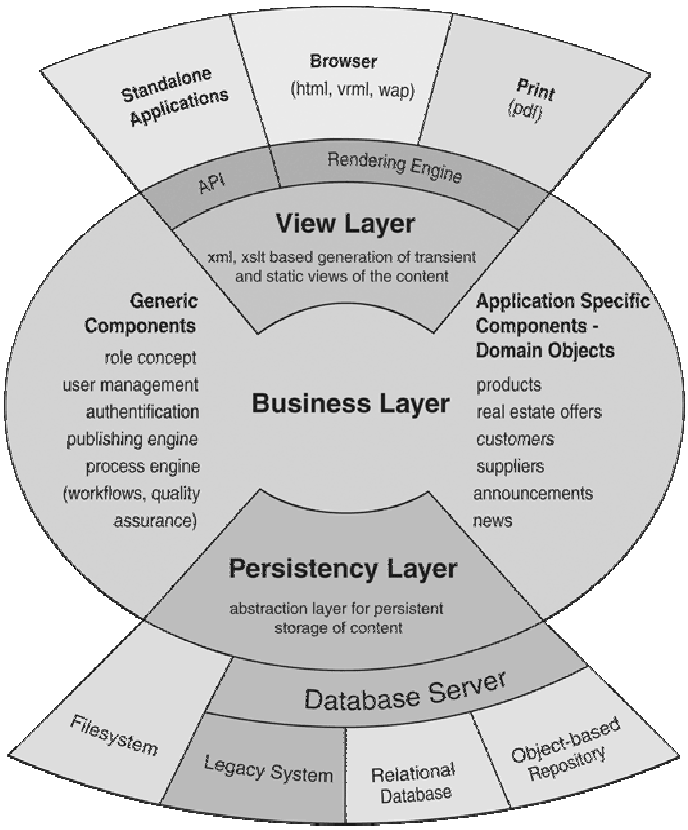

Figure 1: The architecture of the knowledge management portal

The basic system architecture of bluedot is based on a three-tier architecture separating a persistency layer for database (repository) access, a business layer for business objects and their functionality and a view layer for different media and ways to access the business objects. The figure on the right depicts the architecture of bluedot, which will now be described in more detail.

1.2 Persistency Layer

The persistency layer comprises a set of components with a common interface for access to information sources, independent of their specific implementation or technical realization. Therefore, access to file-system based information sources, XML-files, OODBMA, relational databases (ORACLE™, MySQL, MS-SQL™) and object-based repositories like Enabler™ are supported. The access is transparent for the components of the business layer.

1.3 Business Layer

The business logic of the information systems application domain is represented within the business layer. Application specific notions, concepts, and functionality are mapped within adaptable business components. As framework of constantly increasing, generic, off-the shelf business components can be incorporated to expedite the development process of new bluedot based systems, e.g.:

- process engine
- quality assurance
- role concept
- publication engine

Additional to the generic components, each bluedot-based instantiation needs application-specific business components, which incorporate the functionality specific for the information domain. In our case, this functionality was the management of knowledge and the appropriate presentation of this information in 3D-VRML as well text-based. The business components developed specifically for the knowledge management portal will be presented in the chapter….

1.3 View Layer

The view layer contains generic view classes which implement access to the business components, described above. For a specific system, these views can be configured to a specific set of components to retrieve information and present it to a specific target medium. Thus, the views themselves are independent of the target media format. Currently, the framework supports HTML, PDF, WML and VRML, but will be augmented to formats like Java3D and MPEG4, too.

In order to define the layout for presentation of the content, bluedot uses XSL stylesheets. These stylesheets format the information according the output media and the desired detail level.

2 The bluedot Enaction Process

The bluedot framework provides for short turn-around times for creating new instantiations. In order to develop a system that completely fulfills all needs of the customer, the following process is to be enacted:

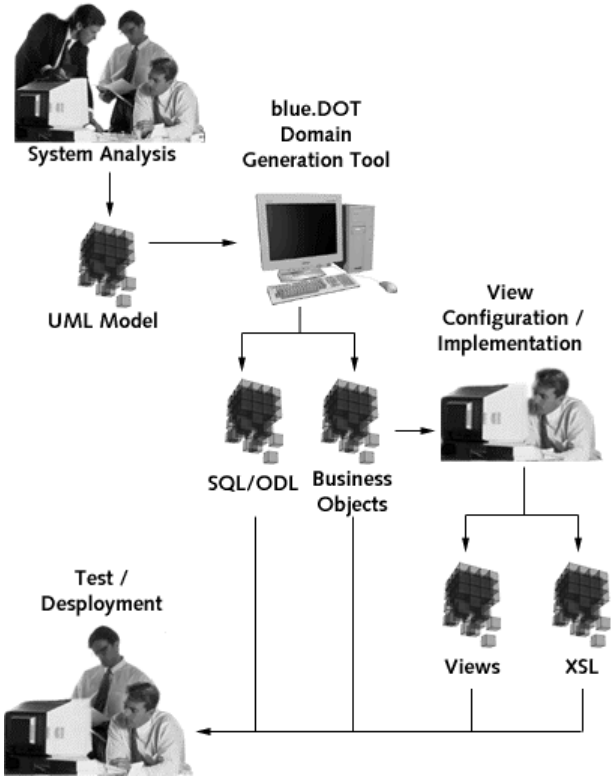

Figure 2: The bluedot Enaction Process

The first step is to do an analysis of the customers information domain. Result of this requirements analysis is a UML system model which is maintained by means of standard tools like Together™ or Rational Rose™. The static class diagrams of this model describe the classes of business objects of the domain to be created, including their attributes, methods and relations.

Using a standard tool these information are now exported to an XMI file, which afterwards is being processed by the bluedot domain generation tool. This tool is capable of producing (Java) class files for every class in the system model, which contains all attributes, constructors and access operations to the components of the persistency layer. The resulting classes conform to the business component interface, thus their instantiated objects are capable of presenting their content (attributes and related objects). The third step is the implementation/configuration of the views on the generated business object classes, which is accompanied by the creation of the appropriate XSLT-stylesheets that define the desired layout and format of the presentation. Finally the system is to be thoroughly tested (integration and performance tests) and deployed to the customers web-server.

3 The Knowledge Management Business Model

In the case of the knowledge management portal the business objects are knowledge elements (or topics) and the functionality which is not part of the framework is the management and visualisation of these topics.

The task was, therefore, to produce a Rational Rose™ file containing a business model for knowledge management. We started off with a widely-accepted standard – the XTM standard. This specification provides a model and grammar for representing the structure of information resources used to define the so-called topics, and the associations (relationships) between topics. It was developed by an independent consortium of parties developing the applicability of the topic map paradigm [Bi99] to the World Wide Web by leveraging the XML family of specifications.

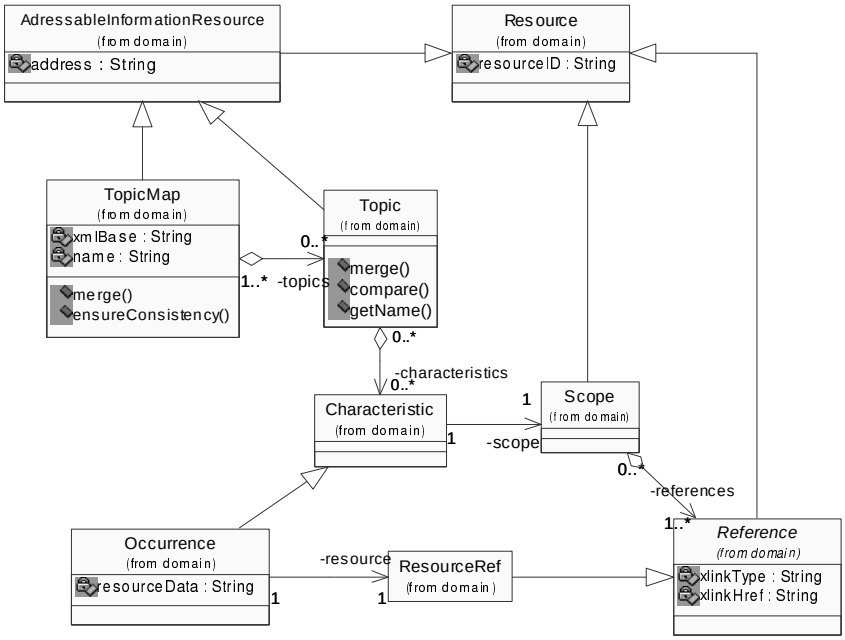

Figure 3: Class model **Topic/Topic Map**

The main concept of topic maps is the <topic> element. A topic is a representation of a subject in the system. A subject is anything about which information is collected. In the most generic sense, a subject is anything whatsoever, regardless of whether it exists or has any other specific characteristics, about which anything whatsoever may be asserted by any means whatsoever. The relationship between a topic and its subject is defined to be one of *reification*. The topic itself is a *addressable information resource* which means that is an information resource whose identity is computable (which means a computer system can retrieve the resource and make deterministic. Every addressable information resource therefore has an attribute ID for identification. A topic, in addition, has to functions which are described in the standard for merging topics.

Two topics are always deemed to have the same subject if:

1. they have one or more subject indicators in common,
2. they reify the same addressable subject, or
3. they have the same base name in the same scope.

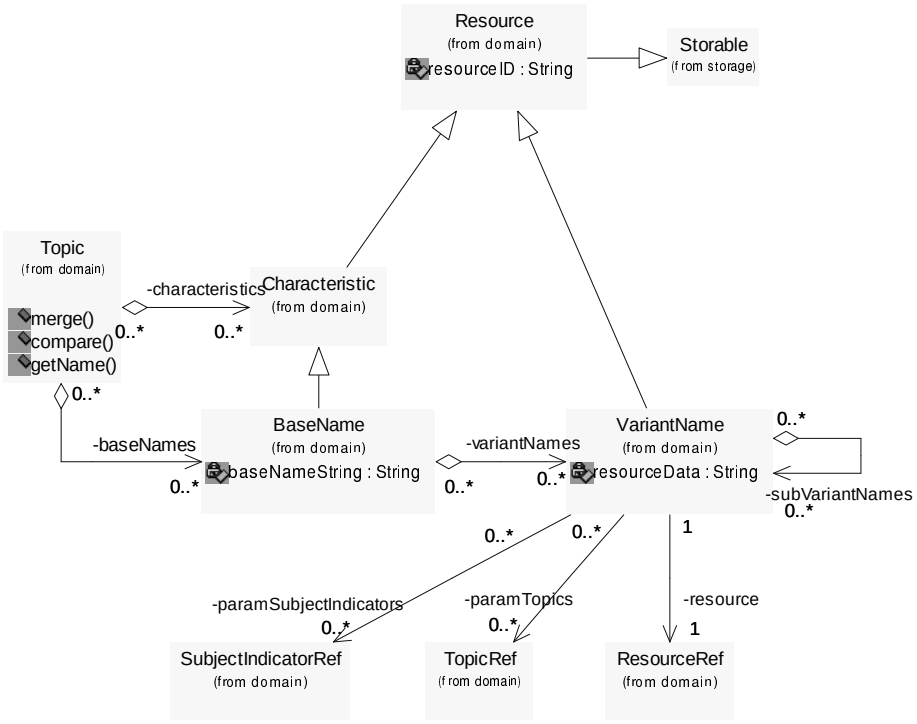

Figure 4: Class model Names

These conditions are determined by the function “compare”. If the result is true the two topics can be merged; if not, the topics have different subjects and have to be kept separately.
Topics aggregate to <topic maps>. A topic map, in turn, uses the “merge”-function of its single topics to realize its “merge” and “isConsistent” function. A consistent topic map is one in which there is one topic per subject and no further opportunities for merging or duplicate suppression.

Topics have *characteristics* which can be either of the following:

1. Topics have occurrences. An occurrence is any information relevant to the given topic. An occurrence can anything from a file, an URL or a reference to an offline document etc.
   Occurrences are modelled as the UML class <Occurrence> with references to the offline occurrence (<ResourceRef>) or containing the data itself in the attribute <resourceData>.
2. A topic may have zero or more names. Each name may exist in multiple forms. A name always has exactly one base form, known as the base name, and it

may, in addition, have one or more variants for use in specific processing contexts.
The class model for names are shown in Figure 4. The class <BaseName> comprises an attribute <baseNameString> containing the text for the base form of the name which is used for labelling the topic unless a variant exists that is deemed to be more apposite in the processing context. A variant name is an alternative form of a base name, that is optimized for a particular computational purpose, such as sorting or display. It may be any kind of a resource, including a string. Therefore, the class <VariantName> holds either an attribute for the name string or references to classes which may hold the variant name.

3. Finally, associations are relationships between one or more topics. The roles a topic plays in associations can be assigned to it. It expresses the nature of the topic's involvement as a member of an association. The class configuration can be seen in Figure 5.

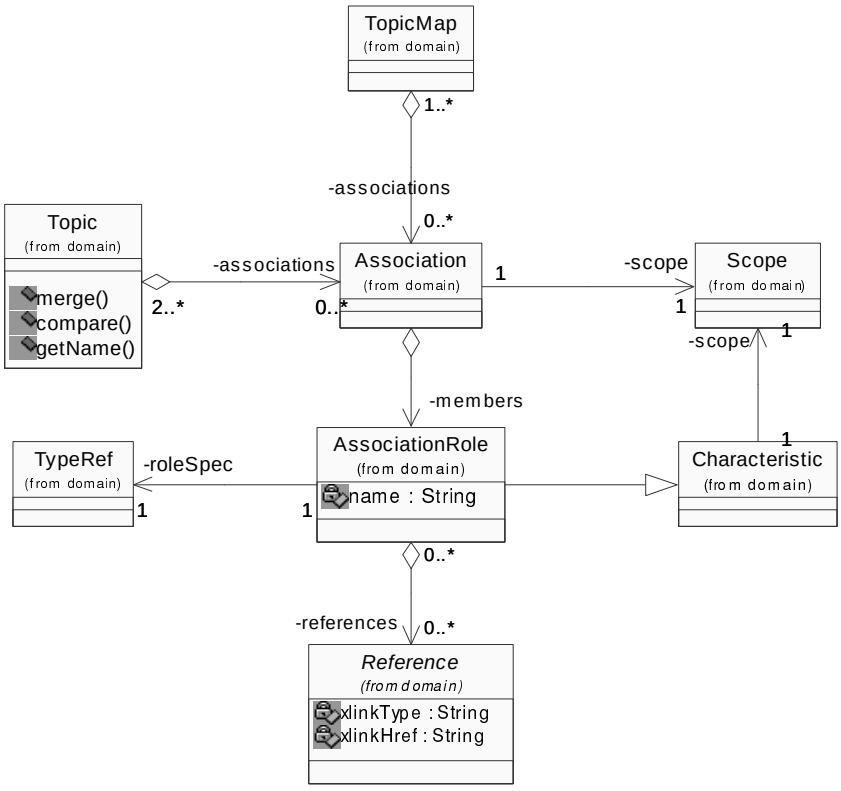

Figure 5: Class model **Association**

Another powerful concept of topic maps are scopes. Scopes specify the context in which the characteristics are valid. Every characteristic has a scope, which may be specified either explicitly, as a set of topics, or implicitly, in which case it is known as the unconstrained scope. Assignments made in the unconstrained scope are always valid. Scopes are expressed explicitly as a class which is a collection of references to topics. These topics form the context in which the characteristic is valid.

4 Conclusion

The Fraunhofer FIRST/IGD is a system which allows the dynamic publication of knowledge maps as three-dimensional landscapes. It is standard-conform (XML, XTM and UML) and therefore allows the interchange of content with other knowledge management systems. The web-based input and editing of topic maps yet has to be implemented.

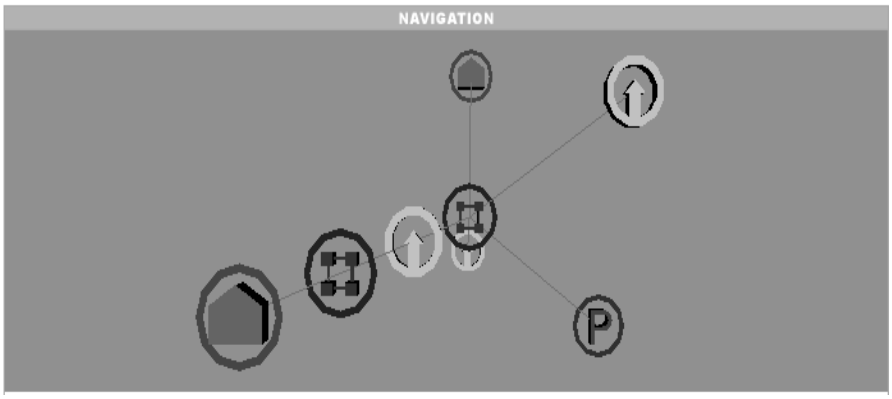

Figure 6: Screenshot of FIRST/IGD knowledge management portal

Moreover, the layouting algorithm and the three-dimensional navigation is still quite elementary and has to be refined for showing qualified and semantic information like the similarity of two or more topics in the topic map.

6 References

[To00] TopicMaps.Org Specification, XML Topic Maps (XTM) 1.0, http://www.topicmaps.org/xtm/index.html

[BL00] bluedot Technical White Paper, IT Service Omikron GmbH

[Bi99] ISO/IEC 13250:2000 Topic Maps: Information Technology – Document Description and Markup Languages, Michel Biezunski, Martin Bryan, Steven R. Newcomb, ed., 3 Dec 1999. See: http://www.y12.doe.gov/sgml/sc34/document/0129.pdf