A Semantic Web Approach for Adaptive Hypermedia

Serge Garlatti and Sébastien Iksal
Department of Artificial Intelligence and Cognitive Sciences
ENST Bretagne - Technopôle de Brest Iroise
B.P. 832, 29285 Brest Cedex, France
{serge.garlatti, sebastien.iksal}@enst-bretagne.fr

Abstract

Adaptation/personalization is one of the main issues for web services. Adaptive web applications have the ability to deal with different users' needs for enhancing usability and comprehension and for dealing with large repositories. Indeed, adaptive web services - also often called Adaptive Hypermedia Systems - can provide different kinds of information, different layouts and different navigation tools according to users' needs. We propose an open-ended adaptive hypermedia environment which is based on the virtual document and semantic web approaches and which is able to manage adaptive techniques at knowledge level. The aim is to simplify the creation and the management of adaptive web services by using ontologies and semantic properties for adaptation. Indeed, they are declarative parameters for computing on the fly services. Indeed, the specification of the adaptive mechanisms is defined by semantic properties associated to a hypermedia document by an author. These properties have the following roles: define how to evaluate the links/content for grouping them together in different classes according to a user model, determine how to manage these classes for each adaptive technique and assign user stereotypes to adaptive techniques. Then, an author can determine the relevant adaptive techniques for a given user group.

Keywords

Adaptive navigation and presentation, Virtual Document, Composition Engine, Semantic Web

1. Introduction

Nowadays, numerous services are available on the Web, for instance, portals, e-learning, digital libraries, on-line information systems, virtual museums, e-business and digital newspapers are current services. Adaptation/personalization is one of the main issues for web services. Adaptive web services have the ability to deal with different users' needs for enhancing usability and comprehension and for dealing with large repositories. Indeed, adaptive web services - also often called Adaptive Hypermedia Systems - can provide different kinds of resources, different navigation tools and different layouts according to users' needs [14]. The creation and/or maintenance of adaptive web services from these repositories require the following features: i) methods to facilitate web service creation and management and ii) reuse, sharing and exchange of resources through the internet/intranet iii) selection of the relevant resources and their organization according to user's needs.

The new age of Internet is the Internet of meanings and provides some of these features. The Semantic Web is a vision for making the contents of the web understandable to the machines. It would be a basis for creating intelligent web services in the future. Then, this third web generation has the ability to enhance information retrieval, the reuse, sharing and exchange of resources through the internet/intranet and to deal with automatic or semiautomatic services. Indeed, it is well known that keyword-based information access presents severe limitations concerning precision and recall. On the contrary, intelligent search engines, relying on semantic web initiative [14] and semantic metadata, overcome these limitations [14]. Nevertheless, information space is so huge that it is not sufficient to have a precise search engine. It is necessary to take into account user interests and to have an accurate metadata schema to be sure to focus on relevant pieces of information. Adaptive Hypermedia systems can be viewed as automatic or semiautomatic services dealing with different users' needs and distributed resources. This kind of services can generate dynamically a service adapted to the users.

It is not sufficient to rely on the semantic web initiative. Indeed, flexible hypermedia and more particularly that of virtual documents can lead to methods facilitating web service design and maintenance. According to Watters, "A virtual document is a document for which no persistent state exists and for which some or all each instance is generated automatically at run time" [14]. They have the ability to select the relevant resources and their organization according to user's needs. A virtual document, as a service, is composed of an information space – resources - and mechanisms – at least a composition engine - to compute on the fly numerous different real documents from a specification. The service maintenance may be done by changing the specification. Moreover, a composition engine may also be used for creating new services as soon as these services are compatible with the principles underlying the composition engine. These principles are implemented in the composition rules for selecting resources according to a metadata schema and organizing them. For instance, if you have a composition engine able to compute a concert summary of ten minutes, you could reuse this composition engine for a TV show summary. It is possible whether the video segmentation, selection and organization principles are suitable for these shows.

We have designed an open-ended adaptive hypermedia environment which is based on the virtual document and semantic web approaches and which is able to manage selection, organization and adaptation at knowledge level. Virtual document and adaptive hypermedia are closely related – they can be viewed as the two faces of the same coin. At present, we have focused our study on five adaptive navigation techniques (direct guidance, annotation, hiding, sorting and partial hiding) from which adaptive content can be deduced. The specification of the adaptation mechanisms is defined by semantic properties associated to an adaptive document by an author. These properties have the following roles: define how to evaluate the links/content for grouping them together in different classes according to a user model, determine how to manage these classes for each adaptive technique and assign user stereotypes to adaptive techniques. Then, an author can determine the relevant adaptive techniques for a given user group. Indeed, some experiments have shown that it is necessary to provide the relevant adaptive techniques to the current user [14]. For instance, annotation technique is advised for expert and hiding technique for novice.

Adaptive documents rely on some principles which are firstly presented. Secondly, we define the different views of digital document and the corresponding architecture for our adaptive hypermedia environment. Thirdly, the adaptation will be analyzed via our adaptive semantic composition engine. Fourthly, the composition of the delivered document is presented. Finally, we conclude by some perspectives.

2. Design Principles

In our framework, we consider an adaptive hypermedia as an adaptive virtual document. We define it as follows: an adaptive virtual document consists of a set of information fragments, their corresponding metadata, different ontologies and a composition engine. The latter selects and/or filters the relevant information fragments[1] – resources -, organizes and assembles them according to an author/designer specification. Composition and specification are the two stages of an adaptive virtual document. The **composition** consists of three functions which define the composition rules: the **selection** which retrieves the relevant set of fragments, the **organization** which provides an overall document structure and the **assembly** which defines the layout of the document. The organization determines how to access the relevant set of fragments and can be computed on the fly or specified by an author/designer. We are interested in author-oriented adaptive virtual documents. An author-oriented document has the following characteristics: authors have know-how which enables them to give coherence to a document. This coherence depends on the content and its organization according to user's needs. In such a framework, an author has to specify the content by means of the metadata and one or more organizations for this potential content. He also has to specify the different "rules" for adaptation. We call the outcome of this specification a **generic document** from which several adapted documents will be generated on the fly.

In a digital document, three different views may coexist: semantic, logical and layout [14]. These views are closely related to the semantic web architecture: i) semantic: logic, ontologies, RDFS/RDF, ii) logical: XML, iii) layout: XSL/XSLT. First of all the three views are presented. Secondly, our adaptive composition engine architecture based on these three views is analyzed.

2.1 Different views of a document

The three views have a specific structure organizing them. The semantic structure of a document conveys the organization of the meaning of the document content. As resources are distributed through Internet, there is no content at all in the semantic structure, but only a specification of the potential content. The semantic structure - as an overall document structure - plays the role of a site map in a static hypermedia document or the role of the "knowledge tree" in the approach of P. Brusilovsky [14].

The logical structure reflects the syntactic organization of a document. A document (for example books and magazines) can be broken down into components (chapters and articles). The logical view fits the syntactic level of the semantic web architecture. A logical structure is encoded in XML and represents a web page [14]. The layout view describes how the documents appear on a device and a layout structure describes it, (e.g. size and color of headings, texts, etc). An adaptive composition engine relying on these three views has been designed. In this paper, we focus on the semantic view and the corresponding adaptation mechanisms.

2.2 Adaptive Composition Engine

The adaptive composition engine architecture is based on two different studies: ICCARS[2] Project and CANDLE[3] Project which is a European project. Our composition engine is divided into three engines: semantic composition, logical composition and layout composition (cf. figure 1). They are sequential processes according to the three views described previously. In a virtual document framework, the three functions are distributed in these processes as follows: selection and organization are achieved in the semantic composition; assembly is divided into logical and layout compositions. The aim of the semantic composition engine is

1 Fragments are reusable units. Then, they have their own metadata. Fragments can be atomic or abstract. Atomic fragments are information units and cannot be decomposed. Abstract fragments are composed of atomic fragments and/or abstract fragments and one or more semantic structures.

2 ICCARS : **I**ntegrated and **C**ollaborative**C**omputer **A**ssisted **R**eporting**S**ystem (http://iccars.enst-bretagne.fr)

3 CANDLE : **C**ollaborative**A**nd**N**etwork **D**istributed **L**earning **E**nvironment (http://www.candle.eu.org)

to compute an adapted document from an information space, a user model and a generic document. The latter is defined by an author and is a narrative structure for presenting a particular viewpoint on a set of articles or course elements. It is a directed graph in which nodes have a content specification according to a metadata schema and edges are semantic relationships. These relationships belong to those analyzed by Rhetorical Structure Theory (RST)[4] [14]. The adapted document is an instance of the generic document in which content specification is replaced by fragments matching it. In our framework, the information space is reduced to a small subset of fragments to ensure the document coherence. This subset is defined by an author and is associated to the generic document. Nevertheless, it is possible to release this constraint and to provide access to internet/intranet on demand.

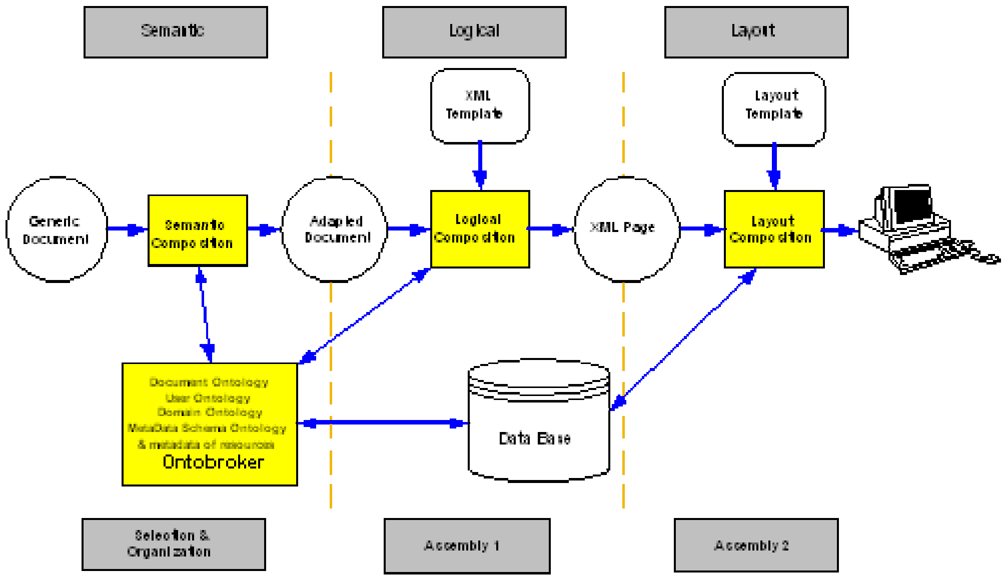

Fig. 1. The Composition Engine Architecture

The logical composition engine browses the adapted document according to user interactions and computes for each node an XML web page with content and navigation tools. An XML template and a user model are used to get this XML web page. In navigation tools, links have properties for managing adaptation [14]. The layout composition engine generates an HTML page from the XML web page in applying the layout rules given by the layout template and using an XSLT processor [14]. Adaptation processes can take place in the three engines. But we focus on the first one: the semantic composition engine.

For generating documents, the composition engine requires different types of models: the document model, the user model, the domain model and the metadata model. All these models are represented at a knowledge level by means of ontologies which are always organized – related – in the same way. They are separated to simplify their maintenance. As these ontologies are loosely coupled, they can be modified without effort. Consequently, it is easier to maintain adaptive documents and to design new adaptive documents.

The document model formalizes the specification of the generic document and its properties. It represents the author's competences and know-how for creating adaptive document. It defines all the different categories of fragments which may be specified in a generic document (for instance, news in-brief, chronicle, press review, editorial, record, research, poll, news items, etc.), their relationships and the RST relationships. The generic document can be considered as an instance of the document model. It also defines the different adaptive techniques and their properties. The domain model defines all the concepts and their relationships in a particular domain (for instance, for special reports about fishing, the

4 The main goal of the Rhetorical Structure Theory is to give coherence to a document;

different categories of fish, vessels and their equipments and tools, people, etc. are defined). The user model is composed of **personal data** (identity, age, town, professional activity, role (reader, author, etc.), **history**, **session**,**preferences** information about interests and preferred adaptive techniques, **knowledge level** about the domain concepts – it is an overlay model. The metadata schema is composed of six parts : **General**: identification of the resource : title, authors, etc. ; **Lifecycle**: information about the management of versioning, number, date, authors, etc.; **Meta Metadata**: Information about the metadata, creator, validator, language, date, etc.; **Rights**: Information about use, cost, license, access restrictions, etc.; **Technical**: Information about format, size, software version, etc.; **Classification**: Description of the content : domain concepts, necessary knowledge level, etc.. The metadata schema is related to the document model for specifying the type of fragments, the domain model for defining the main concepts of a fragment. For designing a generic document an author as to use a subset of the metadata entries for specifying the potential content. In the ICCARS project, an author chooses the relevant fragments for each node of the generic document and the authoring tool will deduce the corresponding specification in a limited information space to ensure semantic coherence. The metadata used to specify the content are as follows: MD.6.1.1and MD.6.2.1 in table 1.

MD.6	Classification		Domain value
MD.6.1	Domain	List	
MD.6.1.1	Concept	Unique value	Domain ontology
MD.6.1.2	Level	Unique value	User Ontology : knowledge level
MD.6.2	Application	Unique value	
MD.6.2.1	Type of fragment	Unique value	Document Ontology: types of fragments

Table 1. Some metadata specifying the content

Potentially, a subset of fragments – a small one – can match the specification. These fragments are called variants of fragments. Some of their other metadata entries differ. They will be used by the adaptation process for evaluating them. Then, another subset of the metadata entries will be reserved for variants of fragments. These two subsets of metadata entries are mutually exclusive and depend on the application.

The semantic and logical composition engines rely on OntoBroker. It is used to browse the different models – ontologies – and to retrieve fragments which match the content specifications by means of the fragment metadata included in [14]. OntoBroker contains four ontologies [14] – one per model - and metadata closely related to them. These ontologies are: a domain ontology for representing contents; a metadata ontology at the information level which describes the indexing structure of fragments; a user ontology which may define different stereotypes and individual features; a document ontology which represents the author’s competences and know-how for creating adaptive document [14]. Ontobroker also contains resources’ metadata for information retrieval. These ontologies are parameters for the composition engine and their organization and relationships have to be stable across services for ensuring the composition engine reusability.

3. Adaptation Specification

The main goal of adaptation in a hypermedia document is to provide the relevant navigation tools, information units and organization. The specification of the adaptation mechanisms is defined by semantic properties associated with a generic document by an author. Then, it can be used in various contexts which require different adaptation mechanisms.

3.1 Principles

Five adaptive navigation techniques are managed by our system: annotation, direct guidance, hiding, partial hiding and sorting [14]. Adaptive content is deduced from them. The properties specifying the adaptation mechanism have the following roles: define how to evaluate the links/content for grouping them together in different classes, determine how to use these classes in the different adaptive techniques and assign user stereotypes to adaptive techniques. Then, an author can determine the relevant techniques for a given user group. First of all the evaluation method is presented. Secondly, the use of the evaluation method for adaptive navigation techniques is analyzed. Thirdly, the association of stereotypes to adaptive techniques is presented.

3.2 Evaluation method

For adaptive navigation techniques, one can evaluate the relevance of the links or the relevance of the destination of links – fragments. In our framework, we propose to evaluate the fragments by means of a uniform evaluation method for all adaptive techniques. Indeed, we propose to define up to five disjoint and totally ordered classes of fragments – at least two classes. They are classified by an evaluation of the destination fragments of the links, that is to say the variants of fragments matching the content specification of each generic document node. These classes are called: very good, good, average, bad, very bad. An author defines the necessary and sufficient conditions for class membership. We have chosen to have up to five classes for user comprehension. Indeed, it could be difficult for a user to deal with too many classes. For instance, several studies have proposed up to five different classes for managing annotation [14], but not more.

The rules for class membership are defined by Boolean expressions composed of unary and binary operators (not, and, or) and comparators (= | < | > | <= | >= | <> | in) between objects. These objects can take into account a user stereotype – user model features coming from the user ontology -, a subset of metadata dedicated to variants of fragments – metadata ontology – and a knowledge level about domain concepts – computed from the metadata and the user model – percentage of known concepts or percentage of concepts having a sufficient level of knowledge. A rule is a condition which has to be satisfied by the fragment and the user model to be a member of the corresponding class. Four examples and an explanation:

- Very Good = "(JOB = Fish wholesaler) AND (PCTAGEKNOWN >= 100)"
- Good = "((JOB = Fisherman) AND (PCTAGEKNOWN >= 75)"
- The fragment is a member of the class “Good” if the user is a fisherman and he knows at least 75% of the domain concepts which defined the semantic content of the fragment.
- Average = "(JOB = Fisherman) AND (PCTAGEKNOWN < 50)"
- Bad = "(NOT (JOB = Sea Job))"

In this example, four different classes have been defined which are mutually exclusive and there are always rank in the same way (Very Good > Good > Average > Bad). The first type of conditions - (JOB = Fish wholesaler) - are compared to the current user model and the second type of conditions – (PCTAGEKNOWN < 50) - are used to compare the knowledge property – overlay model - of the current user model and the domain concepts representing the fragments in metadata. Each fragment matches the content specification of the current node of the generic document is classified in one of these classes according to the current user model.

For the knowledge level, two cases are distinguished: the known concept and the concept having a required knowledge level. Indeed, an author may specify a minimal knowledge level to each relevant concept describing the content. The computed elements are:

- PCTAGEKNOWN: percentage of known concepts.
- PCTAGEGOODLEVEL: percentage of concepts having a sufficient level of knowledge.

3.3 Evaluation management for adaptive navigation techniques

For managing the different adaptive navigation techniques, it is necessary to define how to manage the different classes of fragments. It is sufficient to decide which classes are kept or suppressed for a given technique. For instance, direct guidance, annotation and sorting are managed as follows: i) direct guidance: the best class is kept and the others are suppressed, ii) annotation: all classes are kept and links will be annotated according to the class relevance, iii) sorting: all classes are kept and links will be ordered according to the class order. For hiding and partial hiding, an author has to specify which classes are kept.

3.4 Stereotypes associated to adaptive navigation techniques

A user may have an adaptive navigation technique whether its stereotype matches the user model. All user model features can be used to define a stereotype. A stereotype is defined by Boolean expressions composed of unary and binary operators (not, and, or) and comparators (= | < | > | <= | >= | <> | in) between user model features. An example and its meaning:

- ANNOTATION = "((Age > 18) OR (JOB = Student)) AND (Location In Brittany)"
- Annotation is for user being 18 years old or being students in Brittany

The adaptive presentation is deduced from the applied adaptive navigation technique. Indeed, they are applied on links which are evaluated by the destination content, that is to say fragments. According to the current adaptive navigation technique, each evaluated fragment is presented to the user or not. When there is link filtering – hiding, partial hiding and direct guidance – some fragments are not presented because the corresponding links are suppressed. Then, there is a content adaptation – content adaptation - and a modification of the semantic structure of the document. For each generic document, the following semantic features are defined for adaptation management: the number of classes, membership rules for each class and the management of classes for hiding and partial hiding techniques and for each technique, a user stereotype.

4. Adaptive Composition

The main goal of the semantic composition engine is to compute on the fly an adapted document which is an instanciation of the generic document by selecting content and modifying the semantic structure. The semantic composition engine consists of three processes: the first one selects the relevant fragments for the current node, the second one evaluates the fragments and classifies them in the different classes specified by an author and the third one determines the allowed adaptive navigation techniques and applies them – that is to say kept or not some or all fragment variants.

1. First of all, the content specification is used to query the search engine – Ontobroker – and selects the relevant fragments from the information space associated to the generic document. The outcome of this process is a set of fragment variants. These fragments differ in a subset of the metadata schema, for instance, the knowledge level for some domain concepts, the technical properties (format, size, etc.), etc.

2. Secondly, all fragment variants are evaluated, that is to say each fragment metadata subset dedicated to variants is compared to the different fragment classes. Then, each fragment variant belongs to one class and it has a semantic property called “class member” whose value is “very good” or “good” or “average” or “bad” or “very bad”. This property will be used to manage adaptive techniques.
3. Thirdly, the user model is compared to all stereotypes associated with the different adaptive techniques (annotation, hiding, direct guidance, etc.). Those which fit the user model are kept. For instance, whether the current user model only matches the stereotype of the hiding and direct guidance techniques. they are available to the user. According to the enabled adaptive navigation technique, a fragment variant is kept or not in the adapted document. For instance, if a user is allowed to have annotation, hiding and direct guidance techniques, all fragment variants are kept whatever their class. Indeed, it is necessary to keep all fragments to be able to annotate them. On the contrary, if a user is only allowed to have hiding and direct guidance, the fragment variants belonging to the best class are only kept - class “Very Good”. The others are deleted because the user is not allowed to use them. Consequently, this kind of deletion leads to content adaptation. Some adaptive navigation techniques, like hiding, partial hiding or direct guidance, allow the removal of the irrelevant fragment variants whether they are the only techniques available. They have direct consequences on the document content and structure and then on content adaptation. Whether some fragment variants are deleted, the document content and structure may be modified.

The logical composition aims at computing the current web page structure – XML - with a content and navigation tools for accessing the rest of the adapted document. The navigation tools are the local and global guides for navigation [14] resulting from the available adaptive navigation techniques. Among the enabled adaptive navigation techniques, the user has to choose by means of his preferences one of them otherwise the default one is chosen. For the adaptive navigation techniques, author constraints have priority over user preferences. An XML web page is generated from an XML template. A template describes the logical structure of a web page but without any content or navigation tools. It contains queries for computing navigation tools and for loading the content. The content is given by the selected fragment in the current node of the adapted document. The navigation tools depend on the selected adaptive navigation technique. For defining the links in navigation tools, the logical composition engine has to browse the adapted document and then “translates” semantic relationships into hyperlinks. Let A, B be nodes and R1 be a relationship from A to B. As soon as B has several variants, the relationship R1 is considered as several relationships of the same type, one per fragment from the source A to each destination fragment in B. It has also to use the “class member” property of fragment variants to manage the current adaptive navigation technique, that is to say it has to decide which links are kept or not.

5. Conclusion and Perspectives

In this paper, we have presented an open-ended adaptive hypermedia environment which is able to manage selection, organization and adaptation at knowledge level. It is based on the virtual document and semantic web approaches. The composition engine uses four ontologies: document ontology, user ontology, domain ontology and metadata ontology. The first one defines the author know-how – how to select and to organize a document at knowledge level – and the available adaptation techniques and their properties. They are always organized – related – in the same way and are separated to simplify their maintenance. As these ontologies are loosely coupled, they can be modified without effort. Consequently, it is easier to maintain adaptive documents and to design new adaptive documents. These ontologies are parameters for the composition engine and their organization and relationships have to be stable across services for ensuring the composition engine reusability. They can also be reused, shared and exchanged through internet. The composition engine has been implemented with the adaptive mechanisms. At present, the authoring tool is only able to deal with metadata tagging, searching of fragments according to the metadata schema and non adaptive generic document. It is possible to define adaptive generic document but without the authoring tool.

Moreover, the specification of the adaptive techniques is defined by semantic properties associated to a generic document by an author. The semantic properties specifying the adaptation mechanism have the following roles: define how to evaluate the fragments for grouping them together in different classes, determine how to use these classes in the different adaptive techniques and assign user stereotypes to adaptive techniques. Then, an author can determine the relevant techniques for a given user group. The management of adaptive techniques is made at an abstract level for the entire document. Consequently, it is easier to generate new adapted documents because it is sufficient to modify these semantic properties. Then, an author can deal with new adaptive documents and new user behaviors without changing the generic document a lot. Nevertheless, the management of adaptation at document level does not enable an author to change the adaptation mechanism at a very fine level of granularity. For instance, it is not possible to specify for each link or content how to manage adaptation like in AHA.

Our evaluation can be used for adaptive information retrieval. Indeed, a user can specify the adaptation criteria for adaptive information retrieval on the basis of combination of the following features: his interests and knowledge according to two different adaptive techniques. For example, a user can ask the search engine to sort the outcome by interests - up to five subsets - and to annotate each subset by knowledge. This possibility has been applied in the ICCARS project.

6. References

1. Brusilovsky, P., Methods and techniques of adaptive hypermedia. User Modeling and User-Adapted Interaction, 1996. 6(2-3): p. 87-129.
2. Watters, C. and M. Shepherd. Research issues for virtual documents. in Workshop on Virtual Document, Hypertext Functionality and the Web. 1999. Toronto.
3. Specht, M. and A. Kobsa. Interaction of domain expertise and interface design in adaptive educational hypermedia in Second Workshop on Adaptive Systems and User Modeling on the World Wide Web. 1999. Toronto & Banff: Eindhoven University of Technology.
4. Christophides, V., Electronic Document Management Systems. 1998, UCH/FORTH: http://www.ics.forth.gr/~christop/.
5. Brusilovsky, P. and H. Nijhavan. A Framework for Adaptive E-Learning Based on Distributed Re-usable Learning Activities. in Proceedings of World Conference on E-Learning, E-Learn 2002. 2002. Montreal, Canada.
6. Mann, W.C. and S.A. Thompson, Rhetorical Structure Theory: Toward a functional theory of text organization. Text, 1988. 8(3): p. 243-281.
7. Bray, T., et al., Extensible Markup language (XML) 1.0, (Second Edition). 1998, W3C: http://www.w3.org/TR/2000/REC-xml-20001006.
8. Adler, S., et al., Extensible Stylesheet Language (XSL) Version 1.0. 2000, W3C: http://www.w3.org/TR/xsl/.
9. Brusilovsky, P., E. Schwarz, and G. Weber. ELM-ART: An intelligent tutoring system on World Wide Web. in Third International Conference on Intelligent Tutoring Systems, ITS-96. 1996. Montreal.
10. Fensel, D., et al. On2broker in a Nutshell. in the 8th World Wide Web Conference. 1999. Toronto.

11. Iksal, S. and S. Garlatti, Documents Virtuels et Composition Sémantique: Une Architecture Fondée sur des Ontologies. NimeTic'01. 2001, Ecole des Mines d'Alès: Nimes. 91-96.
12. Iksal, S., et al., Semantic composition of special reports on the Web: A cognitive approach, in Hypertextes et Hypermédia H2PTM'01, P. Balpe, et al., Editors. 2001, Hermès. p. 363-378.
13. De Bra, P. and L. Calvi, AHA! An open Adaptive Hypermedia Architecture. The New Review of Hypermedia and Multimedia, 1998. 4: p. 115-139.
14. Garlatti, S., S. Iksal, and P. Kervella. Adaptive On-line Information System by means of a Task Model and Spatial Views. in Second Workshop on Adaptive Systems and user Modeling on the World Wide Web. 1999. Toronto and Banff: Eindhoven University of Technology.