

# BRIDGING NAVIGATION, SEARCH AND ADAPTATION.

## *Adaptive Hypermedia Models Evolution*

Evgeny Knutov, Paul De Bra, David Smits, Mykola Pechenizkiy

*Department of Computer Science, Eindhoven University of Technology, P.O. Box 513, 5600 MB, Eindhoven, the Netherlands  
e.knutov@tue.nl, debra@win.tue.nl, d.smits@tue.nl, m.pechenizkiy@tue.nl*

**Keywords:** Search, Navigation, Browsing, Adaptation, Personalization, Open Corpus, Dexter Model, AHAM, GAF.

**Abstract:** Adaptive Hypermedia Systems (AHS) have long been concentrating on adaptive guidance of links between domain concepts. In this paper we first study parallels between navigation and linking in hypertext on the one hand and information searching or querying on the other hand. We show that to a large extent linking and searching can be modeled in the same way. Secondly we present a transition towards search in AHS by aligning the web search process with the layered structure of AHS and link adaptation process. In the end we sketch the on-going implementation of an open corpus adaptation carried out in the context of the 'Grapple' adaptive e-learning environment.

## 1 Introduction

The most referenced (but certainly not only) Adaptive Hypermedia (AH) model dates back to 1999. Since AHAM (De Bra et al., 1999) new terms, definitions and models have been introduced and realized in prototypes (Knutov et al., 2009). Most AH models focus on a layered architecture and concentrate on adaptation to the linking and navigation between concepts of an application domain. This architecture dates back to the Dexter Hypertext Model (Halasz and Schwartz, 1994). With the exploding popularity of the Web searching rather than linking is becoming the prevailing form of information access. Hence, apart from adaptive linking there is now also a need to provide personalized search in order to meet the requirements of every particular user. This paper deals with bringing linking and search together, both in an adaptive hypermedia context. We consider the issues of adaptive searching, searching in an adaptive environment, or for instance more familiar in AH field - Open Corpus Adaptation (Brusilovsky and Henze, 2007). In the end we discuss the on-going implementation of an open-corpus adaptation in the context of Grapple (De Bra et al., 2010) project.

In this paper we bring navigation and search in AH context together by means of the following steps:

- We revisit Dexter Hypertext Model and draw parallels between navigation links and queries (section 2);
- We look at the problem of adaptive search and have a brief look at search in Open Corpus environments as a representative use-case (section 3);
- We show 2 use-cases for the interchangeability of navigation and search (section 4);
- In section 5 we show the evolution of Hypertext/Hypermedia modelling from Dexter Model through AHAM to the proposed GAF model, outline advantages of each framework in adaptive environment and as a result we align the conventional search process with the generic adaptation process model (derived from GAF);
- In section 6 we discuss the ongoing implementation of an open corpus adaptation in the Grapple adaptive e-learning environment as a real use-case of bridging navigation and adaptation in one go;
- In section 7 we conclude with the issues and advantages by comparing search methods to conventional hypermedia navigation; we outline challenges and future work directions.

## 2 Browsing, Navigation and Queries: the Dexter Model Revisited

In this section we revisit the Dexter Model to show that hyperlinks are essentially already represented by queries, which makes it possible to replace the navigation structure of the hypertext model with queries (and particularly considering search queries) instead of resolving navigation links.

In Figure 4 (left model) the layered structure of the Dexter Hypertext Reference model is presented. Here the storage layer emphasizes ‘glueing’ components and links together to form hypertext networks. These components are generic containers of data (where there is no difference between content types, graphical and textual components). On the other hand, the within-component layer of Dexter Model is concerned with the contents and structure within the components of the hypertext network.

The Hypertext system requires functions to refer to locations(items) within the content of an individual component. It is done by anchoring (e.g. to support span-to-span links). These anchors provide aforementioned functionality while at the same time maintain a clear separation of storage and within-component layers.

The basic addressability in the storage layer of the Dexter Model concerns the component. This component could be an atom, a link, or a composite entity which may be comprised of other components. Atomic components are primitives which are determined by the within-components layer. Atomic instances can be called ‘nodes’ of the hypertext system. Links here are entities which represent relations between other components. They are usually a sequence of 2 or more ‘endpoint specifications’ each referring to a component in the hypertext. A more detailed structure of the overall organization of the storage layer is shown in Figure 1, it includes specifiers, links and anchors.

Simplifying the model and considering only the Web model of linking, where only the ‘TO’ resolver exists (in terms of Dexter Model) we can see the complementarity of a linking and searching notions (Figure 2).

## 3 Web Search and Adaptive Search

In this section we summarize adaptation methodologies that are (or can be) applied in the search environment, such as querying information using a web search engine. Figure 3 represents conventional Web

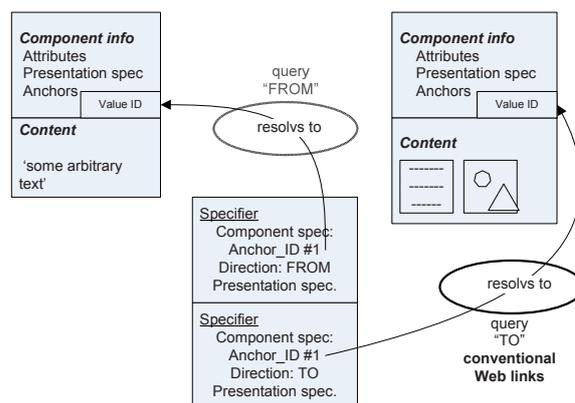


Figure 1: Dexter Model Storage layer (incl. specifiers, links, anchors)

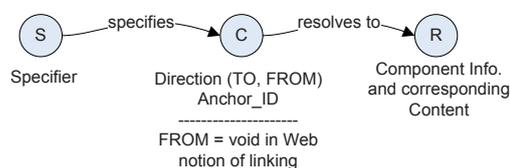


Figure 2: Linking - Query model

search functionality enhanced with adaptation features (presented with oval-shaped blocks).

To supplement the schematic view of ‘adaptive search’ we generalize how these approaches can be applied within the search process and present them in the in Table 1. As three major components we distinguish adaptation of the input parameters, then the search query itself and output results. In addition we consider user modelling aspects involved in the search process.

### Open Corpus Search and Navigation.

As defined in (Brusilovsky and Henze, 2007) search falls under one of the types of information access in AHS. In particular it is represented by Adaptive Information Retrieval (Adaptive IR), typically implemented using keyword-based search mechanisms. Examples of keyword-based search in Open-Corpus AH can be found in the ‘SisKill’ and ‘Webert’ systems where link ordering and annotation is performed or in ‘ML-Tutor’, and ‘YourNews, TaskSieve’ where link ordering and generation is done and many other custom systems (Olston and Chi, 2003). As a result a ‘search’ mechanism or Adaptive IR contributes to the AH field and brings Open Corpus adaptation closer. Therefore we consider that employing the complementarity of search and navigation may have a high impact in Open Corpus Adaptation.

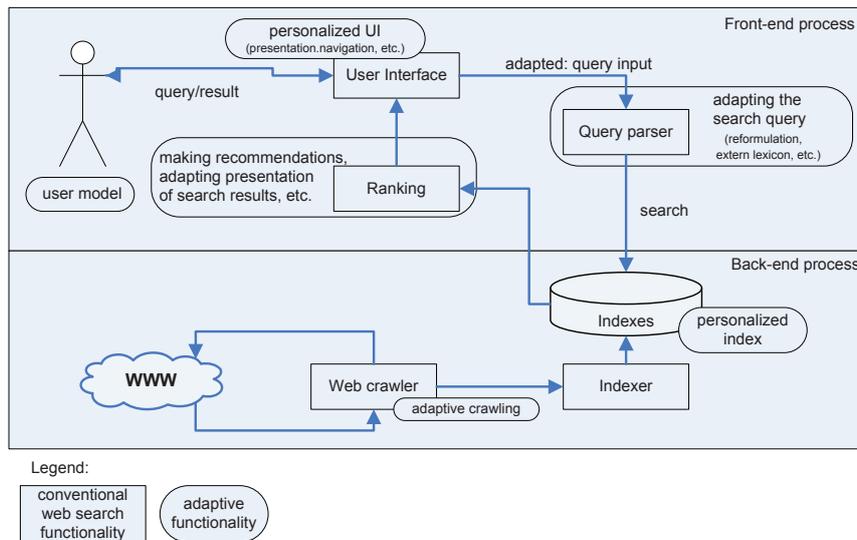


Figure 3: Adaptive Search

## 4 Use-Cases

We show two representative use-cases which fully comply with the idea of *'query-link' interchangeability*.

**Use-Case 1: I'm Feeling Lucky** Using the 'I'm Feeling Lucky' button on Google takes you directly to the first result page. This is an example of what we mean by *link-query* interchangeability: the search query is interpreted as a link specification and in fact the 'click' works just like a web link bringing us directly to the top ranked search result of the search engine. As defined in (Rose and Levinson, 2004) this type of online behaviour is defined by the navigational goal, which is stated by the user when he or she has a specific web site in mind and the reason why the search query is done is that he or she may not know or remember the URL or it is just easier to do a search rather than type in an URL (e.g. 'Royal Dutch airlines', 'Eindhoven University of Technology', etc.).

**Use-Case 2: Collaborative Searching and Linking** Collaborative linking and recommendations were investigated in the field of data mining involving collaborative filtering and recommendations (Smyth et al., 2009). Here we consider it from the perspective of navigation and search. When the system recommends a link to a user, this link is represented by the query which filters the results of others, aggregates, then ranks and presents the top rated link to the concerned user based on the collaborative results.

Often these links are represented by the immediate query which is executed on the click. It retrieves and presents the result to the user, rather than showing some pre-calculated link. In fact the navigation here is completely replaced by the query execution and data retrieval process, but the user still sees the hyperlink on the web page and perceives this as a navigation rather than as querying.

Thus 'collaborative link' (and navigation in general) is essentially a query which takes into account the premises of all the users involved in inferring (filtering) a particular link and resolves the destination by presenting the result of the query. The 'HeyStaks' (Smyth et al., 2009) — social networking and recommendation engine is a good example of collaborative linking, querying and presenting links as a result of a query over preferences of multiple users involved in 'stak' creation which serves a basis for recommendation and ranking using mechanisms of collaborative promotions.

**Use-Case 3: Open-corpus** According to definition Open corpus AH system is an AHS which operates on an open corpus of documents, e.g., a set of documents that is not known at design time and, moreover, can constantly change and expand" (Brusilovsky and Henze, 2007). Open-corpus systems in general provide the flexibility of search and navigation in one go. Essentially they replace the conventional navigation with the indexed terms and search mechanism on the open corpus. Open-corpus domain has the direct application of navigation and search which is in-

Table 1: Generic approaches to search adaptation & adaptive search.

Parameters	Adaptation	User modelling aspects
Input	<ul style="list-style-type: none"> <li>- adapting user input (translating keywords (e.g. into English), aligning with an ontology, using a definition of keyword, rephrasing search requests, etc.);</li> <li>- letting the user choose the field of search or use the domain/user pre-defined information (predefined set of topics, types, etc.) (e.g. pictures, blogs, internet, intranet, videos, etc.);</li> <li>- narrowing down search input according to the user profile;</li> <li>- (auto)suggesting predefined search parameters (suggested search);</li> <li>- personalizing the search input (e.g. using particular search engine for a particular user/interest/data type: use mapquest looking for address or google image search looking for pictures)</li> </ul>	<ul style="list-style-type: none"> <li>- (auto)suggesting predefined search parameters for the user (suggested search using UM)</li> </ul>
Search query	<ul style="list-style-type: none"> <li>- query reformulation (e.g. query expansion);</li> <li>- adapting query parameters (matching/extracting/mapping keywords of a query);</li> <li>- adding external lexicon (e.g. using predefined queries comprising a certain lexicon);</li> <li>- using predefined labelled queries (according to user/group historical search data, user parameters [preferences, interests]);</li> <li>- using semantically related queries (if there is an available ontology)</li> </ul>	<ul style="list-style-type: none"> <li>- querying multiple UM from different systems (e.g. distributed UM);</li> <li>- querying for user undefined parameters (e.g. user interests are not defined in the profile but can be retrieved (queried) from the access log);</li> <li>- using personalized indexes for search</li> </ul>
Output	<ul style="list-style-type: none"> <li>- ranking search results, relating to the results of other users, etc. (e.g. collaborative recommendations);</li> <li>- adapting the output content and presentation (annotating/highlighting/presentation) (e.g. AH methods);</li> <li>- adapting the output navigation (creating lists, tables of contents, local/global maps, etc.)</li> </ul>	<ul style="list-style-type: none"> <li>- user profile can be updated with the results (or search keywords) of the searched information to be used for suggesting/recommending relevant 'next searches';</li> <li>- used search keywords can comprise and accumulate and update the user personalized search indexes</li> </ul>

creasingly considered in adaptive applications is scrutinized. This is where resources come from search results in dynamic learning object repositories or from a Web search engine (see section 3). The ongoing implementation of an open-corpus adaptation within the adaptive e-learning environment "Grapple" (De Bra et al., 2010) is carried out. We discuss the details of this implementation in section 6.

## 5 From Dexter Model, through AHAM, to GAF

In Figure 4 we show the evolution of the Hypertext reference models, from Hypertext to Adaptive Hypermedia to the new Generic Adaptation Framework (GAF) which encapsulates most recent developments in AH and adjacent fields.

A brief discussion of the Dexter model can be found in (section 2), so here we would like to concentrate on the adaptation features evolution and outline major differences of these systems.

The AHAM (De Bra et al., 1999) reference model

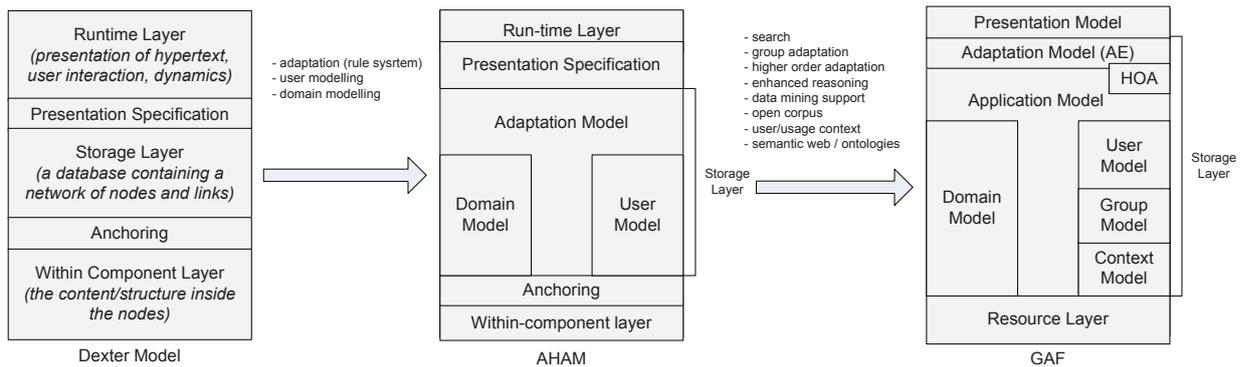


Figure 4: From Dexter, through AHAM, to GAF

could be considered as an adaptive extension to the Dexter model. Major points of AHAM are:

- Any AHAM application must be based on a *Domain Model (DM)*, describing how the information content of the application or ‘hyperdocument’ is structured (using a conceptual representation of knowledge).
- A *User Model (UM)* must be devised and its sustainability should be maintained representing preferences, knowledge, goals, navigation and other relevant user aspects.
- The presentation of content and link structure must be adapted to the user’s behaviour as well as to the user’s knowledge and interest. Thus an *Adaptation Model (AM)* should be defined consisting of adaptation rules. The rules define both the process of generating the adaptive presentation and that of updating UM.

In AHAM the Storage layer split to support Domain and User modelling in order to facilitate adaptation to user attributes based on the conceptual structure of the domain, represented by the concept-link structure. And the Adaptation Model (AM) encapsulates the Adaptive Engine (AE) functionality, the rule system performing adaptation based on the value of UM attributes.

Moving towards a more elaborate framework, GAF will enhance adaptation capabilities and include new methodologies and techniques, facilitating more elaborate adaptation. In figure 5 we present an extended draft architecture of GAF and briefly outline the enhancements (comparing to AHAM).

- *Ontologies* are used in order to provide interoperability in adaptive applications. These ontologies must be agreed upon, considering concept structures and meanings, therefore ontologies as a base concept structures are accepted in more and more

research fields. A Domain Model based on an ontology makes interoperability feasible.

- *Open corpus adaptation* which is increasingly considered in adaptive applications is scrutinized. This is where resources come from search results in dynamic learning object repositories or from a Web search engine (see section 3).
- *Data Mining* is a valuable tool with respect to clustering users into groups based on their navigational patterns or capturing long term effects of adaptation rules.
- *Group-based adaptation* will extend the adaptation by taking group models into account. It determines partitioning of the users into groups and adapting to the group model.
- *Higher order adaptation* will monitor the user’s behaviour also to adapt the adaptation behaviour.
- *Multimedia adaptation* provides a content type independence at any application level, providing a generalization of adaptation techniques and methods to work with.
- *Context Awareness* allows system and application to be decoupled from the existing environment, and makes them more sensitive to adapt in many other ways rather than through a set of predefined rules. We consider usage and user context for GAF: both capturing the context of user behaviour and Domain Model usage, allowing to adapt to user and concept contexts (e.g. environment settings).

**Search illustrates compliance with the generic AHS.** Figure 5 presents compliance of a search process and an overlaying Generic Adaptation Process (GAP) sequence chart. Where GAP represents the process chart constructed by coupling the layers of GAF — a general purpose AHS proposed in section 5 and described in (Knutov, 2008; Knutov, 2009).

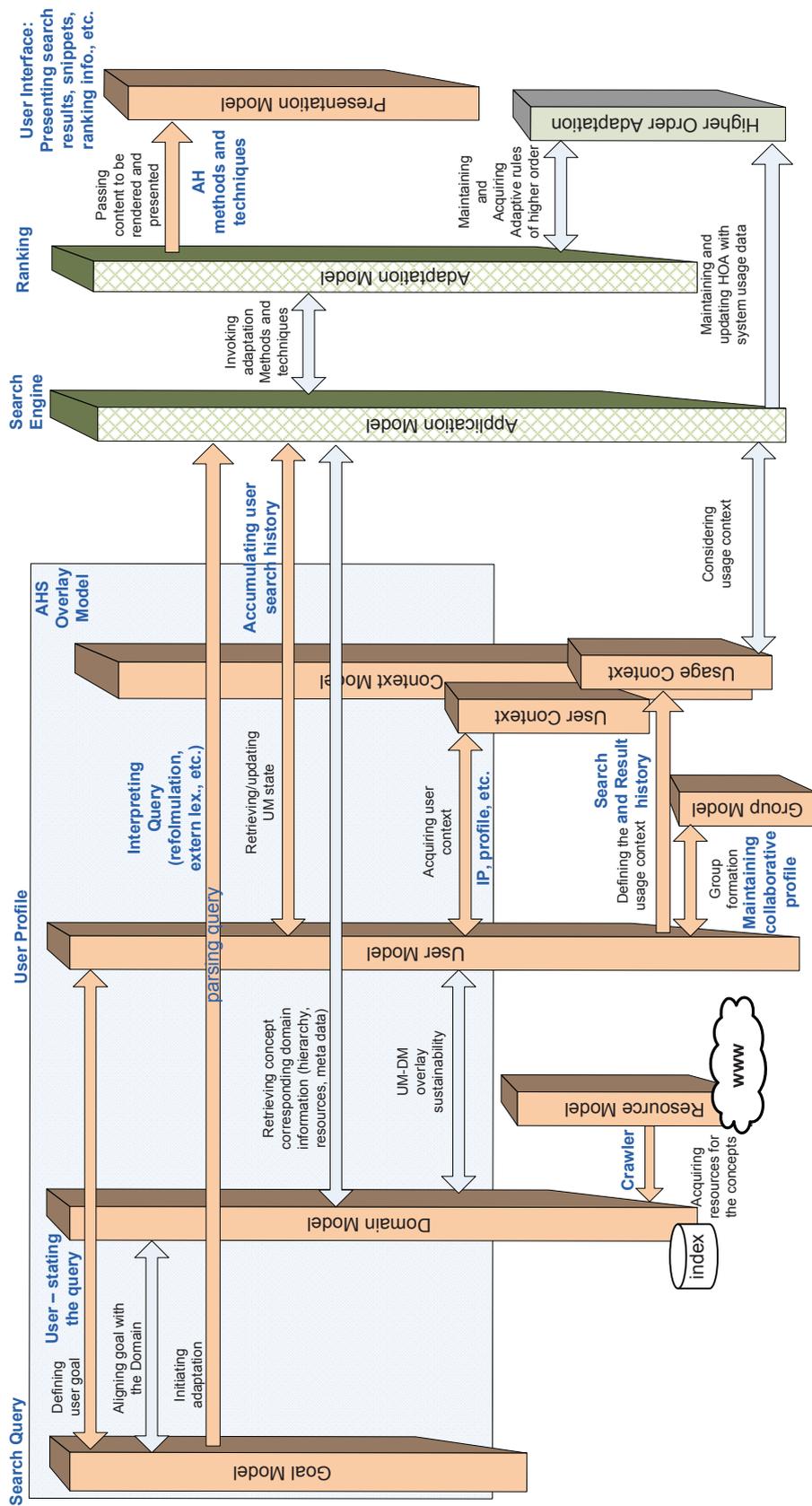


Figure 5: Search compliance with Generic Adaptation Process

Here we assign search process steps from Figure 3 to a single layer or a transition in the system (mentioned on Figure 4). Though we're facing certain issues discriminating Recommendation Engine functionality, in particular Search Engine and Ranking mechanisms (in this respect Application Model (AM) and Adaptation Model/Engine (AE) can be treated accordingly) we could align the search process and describe its functionality (in terms of aforementioned models) with GAF. On the one hand this proves a generic property of GAF, and on the other hand it opens new horizons to facilitate search aspects in the AH field.

The search process complies with the reference structure of AHS as follows:

- *The User states the goal* thus formulating a new search query, which can be considered as stating or choosing a particular concept (set of concepts) to follow in AHS. It can be interpreted and aligned with DM (availability of concepts, concept structures and sequences, etc.) and UM (considering user competencies, preferences, experience, etc.) thus re-formulating and refining the search query (matching it with the common lexicon or using semantically related terms).
- The *Domain Model* is defined by the search index, representing keywords used to facilitate fast and reliable information retrieval, which is acquired from the *Resource Model* (and essentially WWW). The index information is obtained from WWW by means of crawling which is similar to the process of resolving content information of a concept in AHS.
- The *Context Model* defines user and usage context properties such as IP address, user profile/stereotype, or search and result histories accordingly.
- The *Group Model* refers to maintaining a collaborative profile of the user or stereotyping search results by location or user age group and gender, which later can be used to rank and recommend results.
- Retrieving and updating UM refers to *storing and accumulating UM search history* which can be used to reformulate queries or retrieve personalized results.
- *Application and Adaptation Models* may refer to the Search Engine and Ranking mechanisms, however it may not be entirely clear how to distinguish some particular parts of those. Here we would refer to the Adaptation Model for Ranking, since they both to some extent perform adaptation of the results. The Application Model then serves

as the core of the system: coupling other layers and dispatching information in AHS or performing a search as the Search Engine.

- *The Presentation Model* renders search results and presents a ranked result list, snippets, additional rank information, groups result, etc.

## 6 Ongoing Implementation of Open Corpus Adaptation

Hereafter we discuss the ongoing implementation of the aforementioned use-case 3 an Open Corpus adaptation, which is carried out in the context of adaptive e-learning environment 'Grapple' <sup>1</sup>.

The adaptation engine part in Grapple and the UM service part need domain model information and adaptation model to do the reasoning part (here we consider the adaptation model to be part of the domain model). The Open Corpus service of Grapple allows domain and adaptation model to be stored externally and later retrieved. The information can be stored in separate files using the GAM format (which represents concept definitions and their relations), or inside resources using the meta element description.

In the Open Corpus setting the URI that identifies the concept is used as a URL to locate domain model information for the concept. When the concept URI is used as a URL the resulting document is scanned for a meta element with *name* attribute *gale.dm*. The contents of this meta element is expected to be GAM code which describes the concept. In nothing is found, the Open Corpus service searches further. It performs the search from the current path in the URL up to the root of the server specified. The first description found on the current concept is used. Figure 6 presents the idea of an Open Corpus Adaptation in 'Grapple'.

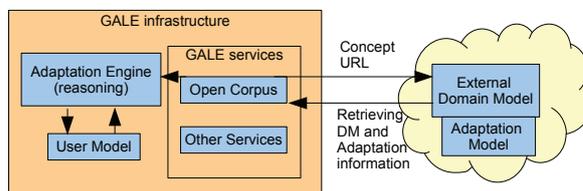


Figure 6: Open Corpus Adaptation

<sup>1</sup><http://www.grapple-project.org/>

## 7 Conclusions and Future Work

We conclude by summarizing the benefits of linking and querying interoperability in an AH context.

*Links and Queries in Adaptation* — flexibility and interchangeability of links and queries prompt the area of AH research to go more towards the direction of Open Corpus adaptation, AE rule systems and Recommender systems. The flexibility to choose between conventional adaptive navigation techniques or adapt search queries using corresponding techniques (Table 1) will facilitate AHS development and face them towards the area of more traditional web search and web information systems (WIS) in general.

*Interoperability and Re-usability* — usage of linking and queries makes systems more flexible in terms of interoperability, compatibility and re-usability. Queries have more flexibility as an intermediate interchange format (e.g. for RuleSystems used in AH systems). The lack of the properties actually stops the spread of AHS into the area of *Open Corpus*, that's why this paper can become a pushing point and we particularly focus on the implementation of the open-corpus adaptation.

*Dynamic Nature of Queries* — Queries are more of a dynamic nature rather than links. Queries represent a parametric structure, which makes them more dynamic and in practice they can generalize different link types.

*Data Provenance* — Queries contain provenance information that can be easily analyzed and interpreted (e.g. using data base query provenance information). This is also possible with the linked structures, however comparing to conventional hyperlinks requires additional metadata.

*Search and Recommender Engines are more flexible for introducing or discovering new rules* — Rule systems which are conventionally used in AHS are facilitated by using queries, providing the compatibility properties with the existing AH rule systems (e.g. ECA type of rules).

In the future we plan to extend the search adaptation process sequence, elaborate the description, in particular inter-layer transactions, emphasizing the interoperability of a new AH developments (Ontologies, Open Corpus, Higher-Order Adaptation etc.) in the context of the search process. This may require unifying search and linking methods for AH field. We also plan to present new use-cases and show how exactly user experience, data provenance and open corpus adaptation are facilitated by the linking and search interchangeability and compliance in the AH field. We intend to map search goals classification on navigational behaviour to show that differ-

ent search queries may be complimentary to navigation and browsing. The ongoing implementation of an Open-Corpus adaptation shows the real use-case linking and domain model and adaptation model extraction which facilitates AHS to use external (open corpus) information to perform adaptation to a particular user's needs.

## 8 ACKNOWLEDGEMENTS

This work has been supported by the **NWO GAF: Generic Adaptation Framework** project and **Grapple** project.

## REFERENCES

- Brusilovsky, P. and Henze, N. (2007). Open Corpus Adaptive Educational Hypermedia. In Brusilovsky, P., Kobsa, A., and Nejdl, W., editors, *The Adaptive Web*, pages 671–696. Springer.
- De Bra, P., Houben, G.-J., and Wu, H. (1999). AHAM: a Dexter-Based Reference Model for Adaptive Hypermedia. In *Hypertext*, pages 147–156. ACM.
- De Bra, P., Smits, D., van der Sluijs, K., Cristea, A., and Hendrix, M. (2010). Grapple personalization and adaptation in learning management systems. In *ED-MEDIA'10: Proc. of World Conference on Educational Multimedia, Hypermedia and Telecommunications 2010*, pages 3029–3038. Chesapeake, VA: AACE.
- Halasz, F. and Schwartz, M. (1994). The dexter hypertext reference model. *Commun. ACM*, 37(2):30–39.
- Knutov, E. (2008). GAF: Generic Adaptation Framework. In Nejdl, W., Kay, J., Pu, P., and Herder, E., editors, *Adaptive Hypermedia and Adaptive Web-Based Systems, 5th International Conference, AH 2008, Hannover, Germany, July 29 - August 1, 2008. Proceedings*, pages 400–404. Springer.
- Knutov, E. (2009). Gaf: Generic adaptation framework. Hypertext 2009 SRC poster.
- Knutov, E., De Bra, P., and Pechenizkiy, M. (2009). AH 12 years later: a comprehensive survey of adaptive hypermedia methods and techniques. *New Rev. Hypermedia Multimedia*, 15(1):5–38.
- Olston, C. and Chi, E. H. (2003). ScentTrails: Integrating browsing and searching on the Web. *ACM Trans. Comput.-Hum. Interact.*, 10(3):177–197.
- Rose, D. E. and Levinson, D. (2004). Understanding user goals in web search. In *WWW '04: Proceedings of the 13th international conference on World Wide Web*, pages 13–19, New York, NY, USA. ACM.
- Smyth, B., Briggs, P., Coyle, M., and O'Mahony, M. (2009). Google shared. a case-study in social search. In Houben, G.-J., McCalla, G. I., Pianesi, F., and Zancanaro, M., editors, *UMAP*, pages 283–294. Springer.