

A P2P Distributed Adaptive Directory

Gennaro Cordasco, Vittorio Scarano, and Cristiano Vitolo
ISISlab–Dipartimento di Informatica ed Applicazioni “R.M. Capocelli”
Università di Salerno, 84081 Baronissi (SA) – Italy

Abstract. We describe a P2P system that offers a distributed, cooperative and adaptive environment for bookmark sharing. DAD offers an adaptive environment since it provides suggestions about the navigation based on (a) the bookmarks, (b) the feedback implicitly provided by users and (c) the structure of the Web. Our system is fully scalable because of its peer-to-peer architecture and provides, also, an infrastructure to build easily P2P overlay networks.

1 Introduction

The navigation within the World Wide Web is usually performed in isolation and, in a certain way, it is surprising that the World Wide Web is successful despite the fact that it does not fully resemble everyday life and activity. In fact, our daily actions are continuously dictated and influenced by others as well as, of course, our actions influence others' choices. Our society relies on such background activities with continuous feedback that induce a sense of community that makes everybody conscious of everybody's behavior.

This dimension is hard to find on the World Wide Web. The popularity of Blogs as well as the decades-old success of newsgroups strengthen the feeling that the WWW is missing an important part of our standard interaction patterns, i.e., cooperation.

Given the copiousness of resources on the Web, information-finding would greatly benefit by the collaboration since others can draw our attention to unexplored (or partially explored) parts of the gigantic amount of resources available. As well as people usually rely on the experiences and behavior of others to help make choices and find desired information. Just watching customers moving in a store, or observing how the visitors of a museum gather together suggest which item/piece is more popular and deserves a visit.

Bookmarks are, nowadays, an important aid to navigation since they represent an easy way to reduce the cognitive load of managing and typing URLs. All the browsers have always provided, since the very beginning of the WWW, friendly ways of managing bookmarks. In this paper we deal with the problem of enriching this supportive framework for bookmarks (as provided by the browsers) by adding collaboration and (group) adaptation with a P2P system.

In this paper we present DAD, a distributed adaptive directory for cooperative bookmarks management. DAD provides a Peer to Peer environment that allows to share bookmarks with all the users connected. Based on the bookmarks, the system can evaluate suggestions to be recommended and inserted into the system. Users can provide feedback to the system so that interesting bookmarks rank higher as well as bookmarks

suggested by users whose bookmarks are well appraised by the group (i.e., that got a high score).

DAD offers users the same environment they are accustomed to, since our system can be used only to share bookmarks with others and the navigation occurs with user preferred browser. Nevertheless, our system offers a good degree of interoperability with Microsoft Internet Explorer bookmarks, in spite of being an external applications developed in 100% pure Java. DAD lets users organize the shared bookmarks in a limited-size version of an ontology that is distributed by Open Directory Project (<http://www.dmoz.org/>).

DAD is a Peer to Peer system (P2P), i.e., a distributed system where each node plays the same role and shares the same responsibility and all the communication is potentially symmetric. P2P characteristics, such as scalability and fault tolerance, make it a popular architecture for Internet applications (see, for example, Napster, Gnutella and Kazaa).

In the next section we refer to some related work; then, in Section 3 we describe the architecture of our system and briefly sketch the P2P infrastructure. Then we show the main characteristics of DAD. We describe some evaluation tests that were recently conducted in Section 4 and conclude the paper with Section 5 where we give some final remarks and the description of further work.

2 Related Work

Since our investigations lie at the crossing of two different areas, we describe here some of the related work in each of the following topics: Bookmarks sharing and Cooperative systems on the Web.

Bookmark sharing systems. Opencola (<http://www.opencola.com/>) is a commercial system that allows to share bookmarks between any web user. The architecture is based on a client component (on user machine) and on a server (that offers communication among clients and additional services). Each client offers to the user the possibility to search bookmarks by keyword, forwarding the search both to other clients and to a meta search engine (based on several other popular search engines).

Opencola main feature certainly consists in the way each user can provide feedback on the search results. In fact, each user can mark as interesting some of the bookmarks that are returned by the system. In this way, returned bookmarks are rearranged based on the similarity with the marked bookmarks. Sharing bookmarks can be configured by the user but Opencola does not use a single ontology for all the clients (which makes difficult for the categorization once the bookmark is found). Nevertheless, it allows to view the classification chosen by the “owners” of the bookmark so that a certain amount of suggestion on the categorization is provided.

WideSource (<http://widesource.com/>) is a freeware system. Unlike Opencola, Widesource uses a base ontology (i.e., common for all the clients) where to categorize bookmarks. Users can add folders to the lower levels of the tree (initially set to 2). WideSource places a certain overhead on the users since it forces the users to place all the bookmarks to be shared in a single folder (losing information on personal categorization) and, successively, into the system ontology. Then, the bookmarks are available for public searches. The same overhead on re-categorization is necessary when importing the bookmark.

Other systems are BISAgent [8], CoWing (Collaborative Web IndexiNG) [9], Baboo (<http://www.baboo.com/>), Sitemeer [17] and WebTagger [10].

Recommendation systems. Recommendation systems are a branch of social navigation systems [6, 14, 16]. A recommendation system suggests resources to users based on their preferences, similarities, and consumer and product attributes (e.g., user skills preferences, titles, authors, brands). The classical application of recommendation system is in on-line stores.

Social navigation in information space uses information about other people's behavior and suggestions to decide what to suggest to users in similar situations. The goal is to take advantage of the navigation of others who have similar tastes or interests.

Among the most popular systems, we name GroupLens [12], Fab [2] and Plastic (<http://www.plastic.com/>).

Group Adaptive System (GAS) [3] is a collaborative and adaptive environment for web users. It provides information that adapts to web users based on their navigation and their interests. GAS enables people in a group to perform asynchronous collaborative navigation on the web: in particular, people can interact and communicate with each other, be aware of group activity using footprints left on the pages, judge and comment viewed pages. The system puts together pages discovered by group members and suggestions by CLEVER-based model [4] in order to provide participants with recommendations (new resources) based on group interest.

WebMemex [13] provides recommended information based on the captured history of navigation from a list of people well-known to the users. It uses Latent Semantic Indexing to identify similarities between HTML pages in order to suggest links.

In [7] the authors use a RDF-based P2P infrastructure called Edutella [15] (based on JXTA(<http://www.jxta.org>)) to share educational material (data and metadata) with the goal of implementing personalized access to distributed learning materials.

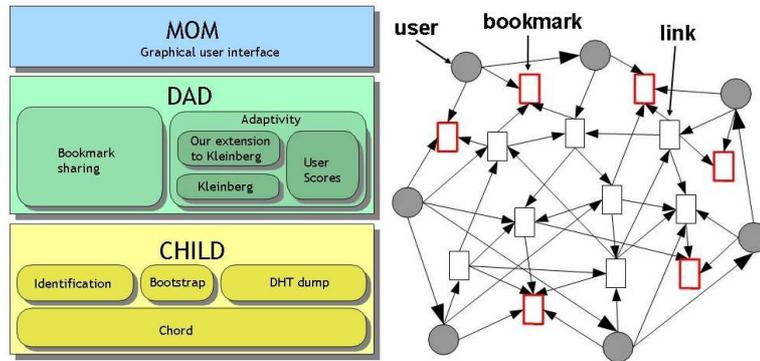


Fig. 1. (Left) The diagram of the architecture of our system. (Right) The graph used as starting set in our modified version of Kleinberg algorithm. Here pages are rectangles while users are circles.

3 Adaptivity and cooperation in DAD

Our system is structured on three separated layers (see Fig. 1 Left): on the bottom we have an overlay network named CHILD (CHord with Identification and Lonely Discovery). We would like to emphasize that the infrastructure (in Java) provided by

CHILD¹ can be used as a base to develop a distributed hash table(DHT) for any pure P2P system.

The middle layer, named DAD (Distributed Adaptive Directory), exploits the bottom level to obtain information about bookmarks and users, providing a collaborative and adaptive system to manage bookmarks. Finally the top layer, named MOM (Multipurpose Ontology Manager), realizes the graphical user interface that represents a typical DAD-peer and allow to use all the functionalities of our system. It also provides a smooth interaction with the top most used browser (MS Internet Explorer) but can be used with any other browser.

We describe the functionalities and the structure of the middle and top layer of our architecture. The architecture of the bottom layer is described in [5]. As pictorially described by the architecture of the system in Fig. 1 (Left), the functionalities of the middle layer are divided in two parts: the component that is in charge of sharing the bookmarks and the component that offers the adaptation by suggesting bookmarks and collecting feedback and scores for bookmarks and users.

3.1 Bookmarks sharing.

Bookmarks represent an important aid to navigation since they provide a way to place landmarks on the huge information space composed of the Web. As mentioned in the introduction, it is extremely helpful to share information about the navigation in order to support and guide the navigation of users. This component is in charge of providing the sharing and organization of bookmarks.

Several different kind of representation are used for sharing bookmarks. One of the main differences of the bookmark sharing systems described in Section 2 is whether the system archives the bookmarks in a fixed ontology (such as Widesource) or asks users to build their own classification (such as Opencola).

Our architecture uses an intermediate approach: while we recognize the importance of providing a fixed infrastructure for the shared bookmarks we also let the users organize their own bookmarks as they prefer.

Bookmarks organization is typically a crucial key to efficiently place landmarks on the information space. Since research shows that a high percentage of users usually does not organize at all the bookmarks (or organize them poorly) [1] we believe that leaving the users create their own ontology for sharing bookmarks can be confusing. The compromise proposed here allows users to create their own organization (even none, if that is the choice) for local (*private*) bookmarks but share them (or importing them) with a fixed ontology (*public* bookmarks).

As base ontology we choose a shortened version (i.e., only the first four levels) ontology of the Open Directory Project DMOZ (<http://www.dmoz.org/>) stored by using XML Bookmark Exchange Language (XBEL) [18]. The limitation in size of the original version of the ontology provided by DMOZ is due to the size: since it is distributed with the application, the complete version was too large (almost 300 Mbytes) to ensure easy downloading of our software.

Of course, specialized versions of the ontology can be used for specialized groups: one can think to distribute DAD for Java programmers, DAD for movie fans, etc. with

¹ The Java package is freely available under the GNU public license at the project home page <http://isis.dia.unisa.it/projects/DAD>.

the appropriate (and more detailed) ontology. In fact, some indications in this directions come from the results of preliminary tests: homogeneous communities feel disoriented by an ontology that is perceived as uselessly large. As a consequence, we report experiments with an additional specific community-oriented category and the comments are much better in terms of cooperation and versatility.

Bookmarks are stored in the DHT that is acting like a distributed filesystem. When a user places a bookmark in the ontology (i.e., makes it shared with the system) his/her peer is in charge of storing the bookmark for the whole system. When a user needs to look into a category in the ontology (an operation called *expansion*) his/her peer needs to know all the peers that are storing bookmarks for the category (called *interest group*).

When an expansion of a folder is asked, the peer has to contact all the peers that contain at least a bookmark in the folder. This set of peers is called the *interest group* for the category. The DHT (offered by the bottom layer CHILD) stores information about each interest group. Each category of the ontology is associated with an ID, the peer interested in the bookmarks of a category, first, accesses the DHT to get the members of the associated interest group and, successively, asks each member the bookmarks that it contributed to the category. Of course, the DHT holds information about all the members, whether present or not; the bookmarks that are effectively retrieved by the successive operation depend on the members of the interest group that are currently in the system. This means that, in general, one cannot be sure that an interesting bookmark that is found once will be there later on, since the member that holds it may not be logged anymore. As a consequence, when something interesting is found, a user should copy it among his/her own private bookmarks. This characteristic of the system is used to measure relative similarity among users (see next subsection).

Joining/leaving an interest group is totally transparent to the user and is managed by the operations of insertion/deletion of bookmarks into the ontology.

3.2 Adaptivity

Our system is adaptive since utilizes user bookmarks as well as user feedback and the underlying structure of the web to suggest and improve the navigation on the Web of all the users. DAD is also adaptable since the user has an explicit control on the rating of the bookmarks and the way they are presented/suggested. When a user places a bookmark in the ontology, the system extracts the links contained in the corresponding page and store the bookmark and the links together in the DHT.

In order to add adaptivity to our system we extended Kleinberg algorithm [11], used in systems like CLEVER [4]. The original algorithm allows to discover authoritative sources of information (authorities and hubs) and is based on the links structure of the Web. Next, we briefly sketch Kleinberg algorithm and motivations and, successively, we show our extensions.

Kleinberg algorithm. The goal of the algorithm is to assign a score to nodes of a graph (i.e., HTML pages) based on the edges among them (i.e., links among HTML pages). Of course, pages with high scores are signaled as significant. Two related scores are evaluated for each node: *authority* weight and *hub* weight. Initial authority and hub weights are set to 1 for each node. The algorithm requires a starting set S of pages, build a graph $G = (V, E)$ where $V = S \cup \{\text{pages linked from } S\}$ and E represent the set of link between the pages on V . Then an iterative algorithm is run on the graph, alternatively evaluating authority and hub weights depending on neighbors' weights.

The algorithm is shown to converge in identifying *good authorities* (nodes recognized as good sources of information) and good *hubs* (nodes recognized as containing good pointers to useful information). The motivating idea is that a link $X \Rightarrow Y$ is meant as “inferring authority from X on Y ” and is explicitly chosen (suggested) by X ’s author that knows both resources.

Our extension to Kleinberg model. In our setting, the starting set S consists of the bookmarks that are present (at a given time) in a category of our ontology. Our extension to Kleinberg algorithm consists in adding users to the set of nodes (i.e., bookmarks) therefore modelling the interactions between users and bookmarks. In this way, we add a new type of nodes (i.e. users) and new links (from a user to a page or from a user to another user) on the graph G (see Fig. 1 (Right)).

In fact, the original motivation in Kleinberg work was that a link $X \Rightarrow Y$ on the Web is meant as “inferring authority from X on Y ” and is explicitly chosen (suggested) by X ’s author that knows both resources. We noticed that, if user U_1 and user U_2 access page P_1 and page P_2 , (authored, respectively, by A_1 and A_2) and both pages point to a page P_3 (which was not actually visited by either user) the algorithm can “discover” P_3 and propose it as a possibly useful resource to U_1 and U_2 (or to any other similar user). Since this suggestion derives, in a certain way, both from U_1 and U_2 accesses and from A_1 and A_2 choices, it is widely supported by an authoritative chain that begins at users, passes through page authors and identifies a useful resource.

The new kinds of links in graph G have the following meaning:

- from user U to a page P : when U is the owner of page P , i.e., U placed P as shared bookmark into the category of the ontology.
- from user U_1 to user U_2 : when U_1 is showing interest in a bookmark that was inserted in the system by U_2 , i.e., when U_1 copies the bookmark inserted by U_2 locally among his/her own private bookmarks.

It should be noticed that there can be multiple owners of a bookmark if different users spontaneously inserted a bookmark in the system. In this case, however, the bookmark is presented only once to users of the system, keeping track, however, of the multiple ownership.

The reader should remember that the design of our system contemplates that a bookmark is available only if its owner is logged. Therefore, if user U finds interesting a bookmark and expects to use it afterward, then U must “copy” it into his/her own bookmarks, to be sure it will be accessible anytime. This action is referred to as “*marking a bookmark*” and makes a bookmark local on the peer. Marking is somewhat “forced” by our architecture so that we can use it to measure the approval and interest of other users’ suggestions.

User score. We define a user score for each category in the ontology, initially set to 0. When user U_1 shows interest in bookmark B (in category C) owned by U_2 , then the score of U_2 is increased. Which action is considered to “show interest” in a bookmark? We decided that when user U_1 marks B (i.e. copy it locally) he/she is showing explicit interest in B since he/she wants it to be available even if U_2 is not logged on.

User scores are stored in the DHT as fields of the interest group². For each category, the information about the interest group consist of the list of members, each with his/her score in the category.

For each interest group, we store information about all the members (and their score) as well for each user we store all the scores in the interest groups he/she is a member of. In this way, we efficiently support both the following queries: (a) given a category list all the members with their score, (b) given a user list all the interest groups he/she belongs to and the relative score. When a user looks into a category, the system can locally obtain the bookmarks (exploiting the DHT) and compute the bookmarks scores.

Adaptability. Each bookmark currently in our system has four scores:

- *authority* and *hub weight*: provided by Kleinberg algorithm;
- *owners' scores*: the scores of the user(s) that inserted the bookmark;
- *occurrences*: the number of users that keeps locally the bookmarks stored.

Notice that the occurrence of a bookmarks represents an additional evaluation of popularity among users (see Fig. 2). Notice also that the number of occurrences can be (and usually is) larger than the number of owners: each time a user marks a bookmark, the number of occurrence increases.

The problem is how to combine all these scores in such a way that the user is presented with a “global” ranking of bookmarks. We believe that the way scores are combined is not only dependent on the nature of topics treated (e.g., professional categories vs. entertainment categories) but is also strongly influenced by personal characteristics, attitude and tastes.

Given the subjectivity of the choice, our system provides an *adaptability* mechanism that allows each user to choose the relevance of each of the scores shown above. By using a control panel user can tune the relative weights of each score to compute a global score for each bookmark in a [1, 10] range.

3.3 The user interface

Bookmark visualization. We show in Fig. 2 the main window of our application. On the left-hand side, the button “System” offers the bookmark sharing. We can insert a bookmark in each folder of the unique ontology either by button “New bookmark” or by the (native) cut-and-paste facilities. By using the button “Internet Explorer” one can import all the bookmarks (Favorites) in IE by specifying the pathname of the directory that stores the Favorites. Folders in the ontology that contain at least a bookmark are presented in boldface.

A user can browse the hierarchy by asking to *expand* a folder which generates a request to the DHT to get all the bookmarks in the corresponding category. Of course, user can simply navigate by clicking on the bookmark name (user preferred browser can be configured).

When a folder of the hierarchy is shown, bookmarks are shown with different colors (see Fig. 2) in order to help the user in recognizing the status. Moreover, each bookmark is also shown with the number of occurrences (i.e. number of users that copied it

² There is a single DHT that, for each ID, stores information on a peer (user), information on a folder (interest group) or both.

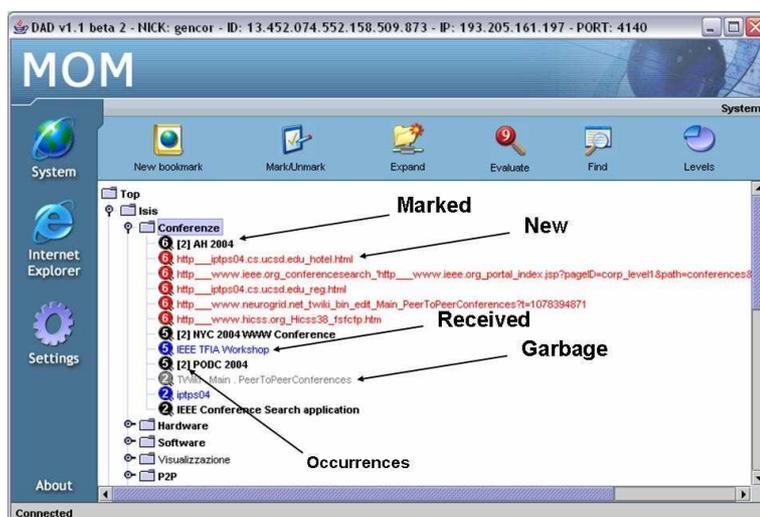


Fig. 2. The user interface (MOM)

locally). The bookmarks can be of four species:

Marked (Black): present locally to the user and that is shared with the users. Without an expansion of the folder only marked bookmarks are shown.

Received (Blue): found on the system (i.e. shared by another user).

New (Red): not inserted by any user but suggested by the system, applying the extended version of Kleinberg algorithm to all the bookmarks in the folder.

Garbage (Gray): ready to be disposed: it was originally marked and has now been unmarked (by using the “Mark/Unmark” button). The bookmark will not be visible anymore after the user logs out. Finally, it is possible (in the configuration) to set a limit to the number of bookmarks that are suggested by the system.

Some utilities. DAD allows each user to know the owner(s) of every bookmark, about their IDs, nicknames and his/her score. Moreover, our system is particularly well integrated with MS Internet Explorer: DAD keeps synchronized a category in the IE Favorites, with the non-empty folders of the ontology. Synchronization happens both ways: user can place bookmarks in DAD by placing in the category of the IE favorites and viceversa .

4 Preliminary evaluation

Two rounds of tests were conducted, the first with 12 participants and the second with 21, where 9 participants were in both tests. Both rounds took place for approximately 2 weeks (each), with a large variety of browsers (MS Internet Explorer, Mozilla, Opera, Safari) with many users on MS Windows 2000/XP, and some on Mac OS and Linux.

Before each round, each participant was given a pre-test questionnaire to inquiry on the way of using bookmarks and a brief personalized briefing on configuration and usage. All the users were left free to use the system at work as they preferred.

Pre-test questionnaire results show a wide variety of ways of organizing bookmarks: for example the number of bookmarks in their browser³ was I: Avg. 188, Std. 350,8;

³ We indicate the I/II round and the value average (Avg) and the Standard deviation (Std).

II: Avg 73.4, Std 56.7 and the depth of the hierarchy was I: Avg. 2.17, Std 0.98; II: Avg. 2.18, Std. 0.91). Anyway, the users were consistently using the bookmarks in their navigation habits: bookmarks inserted in the last week was I: Avg. 4.9 Std. 3.4; II: Avg. 5.8 Std. 3.9. Each day they visited bookmarks I: Avg. 6.5, Std. 5; II: Avg. 5.37, Std. 4.74.

The qualitative analysis was performed by a post-test questionnaire was given to ask about how much they found interesting the bookmarks provided by the other users, how much were interesting the bookmarks provided by the system and some questions on the user interface and the integration. The users were asked to indicate the agreement with statements from 1 (completely disagree) to 5 (completely agree). Finally, the subjects were asked for comments.

In the first round the users were given the base ontology (almost 65000 on 4 levels (DMOZ). After the first round, all the users (even the ones that used bookmarks less often) gained new interest in bookmarking resources. While, in general, users did not find particularly interesting the bookmarks of other users (62% 1-2, 30% 3, 8% 4-5), they appreciated much more the bookmarks suggested by the system (15% 1-2, 50% 3, 35% 4-5). Among the comments, the users found that the size of the ontology used was too large and that it took too much to browse the hierarchy and demanded for smaller size and more focused ontology.

Given the suggestions by the user, in the second round of tests we *added* an ad-hoc category that contained a small number of subcategories (10) that were directly related to the activity of the group. Participants were anyway left free to place bookmarks anyway on the base ontology. The result was that, while the activity decreased but not substantially (the total number of bookmarks was I: 238 (12 users); II: 241 (21 users)), the participants appreciated the bookmarks inserted by other participants (20% 1-2, 45% 3, 35% 4-5) as well as the bookmarks suggested by the system (15% 1-2, 50% 3, 35% 4-5). The maximum score of a user was 8 (compared to 3 in the first round). Some bookmarks were very popular: one of them was copied by 5 users, considering that the most active participants in II round was 10/11. Moreover, in the second round the participants were able to recognize (and liked) the utility of the adaptability mechanism (bookmarks ranking) given the ability to focus the activity on a subset of categories.

Almost 70% of the users that were browsing with MS Internet Explorer considered useful DAD's integration with the browser. Furthermore, 70% of users judged positively the opportunity of choosing the relative weights of bookmark scores by using the control panel.

5 Conclusions and Future Work

Our system is a pure P2P application (therefore inherently scalable) for managing communities of users that want to share the resources that they found, being supported by an adaptive mechanism and by a highly tunable ranking mechanism. We believe that efficiency (i.e. scalability and fault-tolerance) in providing adaptive services is an important factor and should be addressed, because of the growing interest in these services by a larger and larger audience. Our system is one of the first P2P systems to provide access to adaptive support to the navigation on the Web. In fact, as far as we know, in this field only the system in [7] exhibits a similar P2P architecture (taken by Edutella) but with a different purpose.

As future developments of this project, we plan to investigate the necessary modifications to our system in order to support bookmarks placed simultaneously in multiple folders. Several advantages can be obtained by this extensions as, for example, enrich the hierarchical structure of the ontology with cross-links among categories which several bookmarks contemporaneously belong to.

Another possible avenue of research is to use the Kleinberg model (as modified by us) in determining subgroups and similarities among users. From the architectural point of view, we are planning to add backup capabilities by adding a certain degree of redundancy therefore improving the reliability of the whole application.

References

1. D.Abrams, R.Baecker, M.Chignell, "*Information Archiving with Bookmarks: Personal Web space Construction and Organization*". In Proc. of Human Factors in Computing Systems (CHI 98) ACM Press, pp 41-48.
2. M. Balabanovic, Y. Shoham, "*Fab: Content-Based, Collaborative Recommendation*". In Communications of the ACM. Vol. 40, N. 3. March 1997.
3. M. Barra, P. Maglio, A. Negro and V.Scarano, "*GAS: Group Adaptive Systems*". In Proc. of 2th Inter. Conf. on Adaptive Hypermedia and Adaptive Web-based Systems (AH2002). May 29-31, 2002. Malaga, Spain.
4. S. Chakrabarti, B. Dom, D.Gibson, J. Keinberg, P. Raghavan and S. Rajagopalan, "*Automatic Resource list Compilation by Analyzing Hyperlink Structure and Associated Text*". Proc. of the 7th Inter. World Wide Web Conference, 1998.
5. G. Cordasco, V. Scarano, C. Vitolo, "*Architecture of a P2P Distributed Adaptive Directroy*". In Proc. of 13th Int. World Wide Web Conf. (WWW04) (Poster) New York May 17-22, 2004.
6. A. Dieberger , K. Hk, "Applying principles of social navigation to the design of shared virtual spaces' ", presented at WebNet'99, Honolulu, Hawaii, 1999.
7. P. Dolog, R. Gavrioloaie, W. NejdI, J. Brase, "Integrating Adaptive Hypermedia Techniques and Open RDF-based Environments". In Proc. of 12th Inter. World Wide Web Conf. (WWW03) Budapest (Hungary) May, 2003.
8. J.Jung, J. Yoon, G. Jo, "*BISAgent: collaborative Web browsing through sharing of bookmark information*", 16th World Computer Cong. 2000. Proc. of Conf. on Intelligent Inf. Processing.
9. R.Kanawati, M. Malek, "A multi-agent system for collaborative bookmarking", Proc. of the first Inter. joint Conf. on Autonomous Agents and multiagents Syst. (AAMAS 2002), Bologna, Italy, July 15-19, 2002.
10. R.M. Keller, S.R. Wolfe, J.R. Chen, J.L. Rabinowitz, N. Mathe, "A bookmarking service for organizing and sharing URLs". In Proc. for the Sixth Inter. World Wide Web Conf. (Santa Clara, Calif., Apr 1997).
11. J. Kleinberg, "Authoritative sources in a hyperlinked environment". In Proc. of the ACM-SIAM Symp. on Discr. Alg., 1998.
12. J. A. Konstan, B. N. Miller, D. Maltz, J.L. Herlocker , L. R. Gordon, J. Riedl, "*GroupLens: Applying Collaborative Filtering to Usenet News*". In Commun. of the ACM. Mar 1997.
13. A. A. Macedo, K. N. Truong, J. A. Camacho-Guerrero, M. Pimentel, "Automatically Sharing Web Experiences through a Hyperdocument Recommender System". In Proc. of ACM Hypertext 2003, Nottingham, UK.
14. A. J. Murno, K. Hook, D. Benyon, "*Computer Supported Cooperative Work. Social Navigation of Information Space*". Springer-Verlag. G. Britain 1999.
15. W. NejdI, B. Wolf, C. Qu, S. Decker, M.Sintek, A. Naeve, M. Nilsson, M. Palmer, T. Risch, "Edutella: A P2P Networking Infrastructure Based on RDF". 11th International World Wide Web Conference (WWW2002), Hawaii, USA, May 2002.
16. P. Resnick , H.R. Varian, "*Recommender Systems*". In Commun. of the ACM,40,3. Mar 1997.

17. J. Rucker, M.J. Polanco, "*Siteseer: Personalized navigation for the web*". Communication of ACM, vol: 40 no: 3, pp. 73-75, 1997.
18. <http://pyxml.sourceforge.net/topics/xbel/docs/html/xbel.html>