Available online at www.sciencedirect.com
SCIENCE DIRECT®
Information Processing and Management 41 (2005) 5–21
INFORMATION PROCESSING & MANAGEMENT
www.elsevier.com/locate/infoproman

A personalized collaborative Digital Library environment: a model and an application ☆

M. Elena Renda [a,b,*], Umberto Straccia [a]

[a] *Istituto di Scienza e Tecnologie dell'Informazione—C.N.R. Via G. Moruzzi, 1 I-56124 Pisa, Italy*
[b] *Scuola Superiore Sant'Anna, Piazza Martiri della Libertà, 33 I-56100 Pisa, Italy*

Available online 25 May 2004

Abstract

The Web, and consequently the information contained in it, is growing rapidly. Every day a huge amount of newly created information is electronically published in Digital Libraries, whose aim is to satisfy users' information needs.

In this paper, we envisage a Digital Library not only as an information resource where users may submit queries to satisfy their daily information need, but also as a collaborative working and meeting space of people sharing common interests. Indeed, we will present a personalized collaborative Digital Library environment, where users may organize the information space according to their own subjective view, may build communities, may become aware of each other, may exchange information and knowledge with other users, and may get recommendations based on preference patterns of users.

Keywords: Digital Library; Collaboration; Personalization

1. Introduction

The amount of information published in electronic format and the number of users accessing it to satisfy their daily information need is growing at a tremendous rate. This is the building block of the digital information age. Remarkably, though more information is easily reachable and in smaller amount of time than a decade ago, it is getting increasingly difficult for individuals to control and effectively seek for information among the potentially infinite number of information sources available on the Internet. Ironically, just as more and more users are getting on-line, it is getting increasingly difficult to find relevant information in a reasonable amount of time, unless one knows exactly *what* to get, *from where* to get it and *how* to get it. New emerging services are urgently needed on the Internet to prevent computer users from being drowned by the flood of available information.

☆ An early, shorter version of this paper was presented at the 5th International Conference on Asian Digital Libraries (ICADL-02), Number 2555 in Lecture Notes in Computer Science, Springer-Verlag.

* Corresponding author. Address: Istituto di Scienza e Tecnologie dell'Informazione—C.N.R. Via G. Moruzzi, 1 I-56124 Pisa, Italy. Tel.: +39-50-3152890; fax: +39-50-3153464.
E-mail addresses: elena.renda@isti.cnr.it (M.E. Renda), umberto.straccia@isti.cnr.it (U. Straccia).

0306-4573/$ - see front matter © 2004 Elsevier Ltd. All rights reserved.
doi:10.1016/j.ipm.2004.04.007

Among the various digital information sources, *Digital Libraries* (DLs) (Fox & Marchionini, 2001) will play an important role not merely in terms of the information provided, but in terms of the *services* they provide to the information society. Informally, DLs can be defined as consisting of collections of information which have associated services delivered to user communities using a variety of technologies. The collections of information can be scientific, business or personal data, and can be represented as digital text, image, audio, video, or other media. This information can be digitized paper or born digital material and the services offered on such information can be varied, ranging from content operations to rights management, and can be offered to individuals or user communities. An essential technology component of DLs is that they are networked, meaning that access is increasingly becoming shared and collaborative.

Research on Digital Libraries began about a decade ago and a number of DLs were created as a result. Much of the research during the initial stages was on digitizing existing sources, creating large-scale collections, technological solutions, and providing simple forms of access. The first generation of DLs provided a small set of services to relatively well-prepared and knowledgeable user communities.

Even though DLs have evolved rapidly over the past decade, typically, DLs still are limited to provide a search facility to the digital society at large. In fact, a common characteristic of them is that they provide no or poor support to individual users, but are oriented towards a generic user. Indeed, DLs answer queries crudely rather than, for instance, learn the long-term or short-term requirements idiosyncratic to a specific user or, more general, specific to an information seeking task. In practice, what happens is that users use the same information resources over and over and would benefit from customization in a broad sense: the time consuming effort that the user put in searching documents and possibly downloading them from the DLs is often forgotten and lost. Later, the user may wish to perform a search about the same topic to find relevant documents that have, e.g., appeared since the last time a search was performed. This requires a repetition of the manual labour in searching and browsing to find the documents just like the first time. As DLs become more commonplace and the range of information they provide services upon increases, users' expectations will increase and users' are expecting more and more sophisticated services from their DLs. A 'quick and dirty search' facility is normally an integral part of any Digital Library, but users' frustrations with this increase as their demands become more complex and as the volume of information managed by Digital Libraries increases. There is a need for DLs to move from being passive with little adaptation to their users, to being more proactive in offering and tailoring information for individual users. If a DL is not personalized for individuals or communities of users then a Digital Library is defaulting on its obligation to offer the best service possible.

The emerging generation of DLs is more heterogeneous along several dimensions. The collections themselves are becoming more heterogeneous, in terms of their creators, content, media, and communities served. The range of library types is expanding to include long-term 'personal' DLs, and well as DLs that serve specific organizations, educational needs, and cultural heritage that vary in their reliability, authority, recency, and quality. The user communities are becoming heterogeneous in terms of their interests, backgrounds, and skill levels, ranging from novices to experts in a specific subject area. The growing diversity of Digital Libraries, the communities accessing them, and how the information is used requires the next generation of DLs to be more effective at providing information that is *tailored* to a person's background knowledge, skills, tasks, and intended use of the information.

Personalization helps people feel that the systems they are using, whether DLs or even the layout of their personal computer desktops, is 'theirs', and if it is 'theirs' then they will use the system in something approaching a partnership. Personalization can be defined as the way in which information and services can be tailored in a specific way to match the unique and specific needs of an individual user or a community of users. This is achieved by adapting the presentation and/or the services presented to the user by taking into account the user's task, background, history, device, information needs, location, etc., essentially the user's *context*. Personalization can be user-driven which involves a user directly invoking and supporting the

personalization process by providing explicit input, i.e., the user explicitly initiates actions and provides example information in order to control the personalization. On the other hand, personalization can be completely automatic (see Goncalves, Zafer, Ramakrishnan, & Fox, 2001), where the system observes some user activity and identifies the input used to tailor some aspect of the system in a personalized way. These two examples of user-driven and automatic personalization are at the extreme ends of the spectrum and many personalization tools will have elements of both approaches.

Among the various aspects, which may go under the label 'personalization', certainly the most immediate and, likely the most useful/used, relates to the information seeking task. In fact, the requirement of a personalized search 'assistant' in the context of DLs is already known and, to date, some DLs provide related, though simplified, search functionality (Bollacker, Lawrence, & Giles, 1999; Di Giacomo, Mahoney, Bollen, Monroy-Hernandez, & Meraz, 2001; Faensen, Faulstich, Schweppe, Hinze, & Steidinger, 2001; Fernandez, Sanchez, & Garcia, 2000; Foltz & Dumais, 1992; Information Filtering Resources, 2004; Moukas, 1996; Rocha, 1999). Informally, these DLs may fall in the so-called category of *alerting services*, i.e., services that notify a user (by sending an e-mail), with a list of references to new documents deemed as relevant. But, *searching* is just one aspect that should be addressed. Another orthogonal aspect of personalization concerns *information organization*, supporting the users in being able to organize the information space they are accessing to according to *their own subjective perspective* (Di Giacomo et al., 2001; Fernandez et al., 2000).

Early research on DL personalization involves simple models of user interests to make individual recommendations. That is, they rely on the so-called notion of *user profile*. Informally, a user profile is a (machine) representation of the preferences of a user, i.e., a user profile is a structured representation of the user's needs through which, for instance, an information seeking assistant, should act upon one or more goals based on that profile and autonomously, pursuing the goals posed by the user (even irrespective of whether the user is connected to the system). The user profile can be acquired either automatically or set-up manually. In the former case, machine learning techniques can be applied by observing user-system interactions and relying on implicit or explicit relevance assessments, while in the latter case the profile is defined by the user manually. In both cases, we have to describe *what* has to represented, that is which information pertaining to the user has to represented in the profile, and *how* this information is effectively represented (Amato & Straccia, 1999). In the Information Retrieval community, the acquisition of a user profile and the successive matching of documents against it, in order to *filter out* the irrelevant ones, is known as *Information Filtering* or *Content-based Filtering* (Belkin & Croft, 1992; Faloutsos & Oard, 1995).

Very seldomly, [1] DLs can also be considered as *collaborative meeting place* of people sharing common interests. Indeed, our vision is that DLs may be viewed as a *common working places* where users may become aware of each other, open communication channels, and exchange information and knowledge with each other or with experts. Indeed, usually users and/or communities access a DL in search of some information. This means that it is quite possible that users may have overlapping interests if the information available in a DL matches their expectations, backgrounds, or motivations. Such users might well profit from each other's knowledge by sharing opinions or experiences or offering advice. Some users might enter into long-term relationships and eventually evolve into a community if only they were to become aware of each other. Such a service might be important for a DL as it supplies very focused information. Hence, we are moving from services supporting an individual user towards services supporting *groups* of users: thus, we move from the study of individual human behaviour towards the discipline concerned with the study of human behaviour in groups and the technical support thereof. More fundamentally, we make a conceptual shift in our understanding of DLs: whereas the classical view of DLs was manipulation of data

[1] Di Giacomo et al. (2001) is an exception.

by isolated individuals, our view of DLs is manipulation and exchange of data and information as well as cooperation by individuals aware of their environment as well as other users.

Concerning the information seeking task, the *recommendation* of items based on preference patterns of other users is probably the most important one. The use of opinions and knowledge of other users to predict the relevance value of items to be recommended to each user in a community is known as *Collaborative* or *Social Filtering* (see Billsus & Pazzani, 1998; Breese, Heckerman, & Kadie, 1998; Goldberg, Nichols, Oki, & Terry, 1992; Herlocker, Konstan, Borchers, & Riedl, 1999; Resnick, Iacovou, Suchak, Bergstorm, & Riedl, 1994). These methods are built on the assumption that a good way to find interesting content is to find other users who have similar interests, and then recommend items that those similar users like. In contrast to information filtering methods, collaborative filtering methods do not require any content analysis as they are based on aggregated user ratings of these items.

Both approaches share the common goal of assisting in the users' search for items of interest, and thus attempt to address one of the key research problems of the information age: locating relevant information in a haystack that is growing rapidly. Providing personalized information organization and search in the context of a collaborative DL environment as additional services to the uniform and generic information search offered today, is likely to be an important step to make relevant information available to people with minimal user effort (Amato & Straccia, 1999).

The contribution of our paper towards this step is as follows: (i) we will formalize an abstract collaborative DL environment, where users and communities may search, share and organize their information space according to their own personal view; (ii) we will present an instance of the environment as the system developed within the EU funded project CYCLADES; [2] and (iii) for completeness, we will sketch out the recommendation algorithms that rely both on personalized information organization and on the users' opinions. The underlying techniques used for recommendation fall in the afore mentioned categories of content-based and collaborative filtering methods and their combination.

The outline of the paper is as follows. In the next section we will formalize the main concepts of our personalized collaborative DL environment. In Section 3 we will present CYCLADES, while Section 4 concludes.

2. A model of a personalized and collaborative DL

The main principle underlying our collaborative and personalized DL environment is based on the *folder paradigm*. That is, users and communities of users may organize the information space into their own folder hierarchy, as e.g., may be done with directories in operating systems, bookmark folders in Web browser and folders in e-mail programs. The idea of organizing the information space into folders is not new within DLs. For instance, in Di Giacomo et al. (2001) users are allowed to define folders of bookmarks (i.e., URLs). Another example can be found in Rucker and Polanco (1997).

A folder becomes a holder of information items, which are usually semantically related and, thus, implicitly determines what the folder's topic is about. Therefore, rather than speaking about a user profile, we will deal with a *folder profile*, a representation of what a folder is *about*. As a consequence, the user's set of folder profiles represents the set of topics the user is interested in and, thus, the profile of a user consists of the set of profiles related to his folders.

Around this principle, there are three main concepts in our model of a personalized and collaborative DL, namely *actors*, *objects* and *functionality*. Informally, the *actors* will be able to act on *objects* by means of the DL's *functionality*. These concepts will be explained below.

[2] http://www.ercim.org/cyclades

2.1. Actors

In our model we will distinguish two types of *actors*: the set $\mathscr{U}$ of *users* u and the set $\mathscr{C}$ of *communities* C. We adopt a usual view of communities: a community may be seen as a set of users sharing a common (scientific, professional) background or view of the world. Under this assumption, it is quite reasonable to assume that there is a chance that users searching for information within a DL may have overlapping interests. In particular, communities are characterized by a shared interest in the information made available.

We postulate that a community $C \in \mathscr{C}$ has a membership function $\mu_C : \mathscr{U} \to \{0, 1\}$, where $\mu_C(u) = 1$ (for ease $u \in C$) indicates that the user u belongs to the community C. We do not require that a user has to belong necessarily to a community: it is a user's choice to join a community or to leave it. A user may also belong to different communities as well. It is not our purpose to address the issue of how a community may be created and which are the policies concerning to join and to leave it. We simply assume that there is a *community administrator* (a user $u^C \in \mathscr{U}$) for each community $C \in \mathscr{C}$, who is in charge of defining these policies (similarly, we will not address the issue of becoming a community administrator within the environment). It could even be the case that anyone is authorized to build a community.

2.2. Objects

Now, let us define the objects on which the actors (users and communities) may act. Basically, our model considers three types of *objects*: data items, collections and folders.

Data items. At first, we have the set $\mathscr{D}$ of *data items* d. The set of data items is the information space and the data items are the information resources that a user is usually interested in discovering or searching for within the DL. Examples of data items usually managed by DLs are papers, reports, journals, proceedings, notes, annotations, discussions, URIs, or just a metadata record, which consists of a set of attributes and related values specifying features of a document, according to a specific schema, like, for example, Dublin Core (see Dublin Core,WWW). We do not postulate much more about them. So, for instance, the set of data items $\mathscr{D}$ might well be distributed over several resources, heterogeneous in content, in format and media (like text, images, audio, video, audio-video).

Collections. A natural way to give an organization to the set of data items is to organize them into *collections*. A collection may be seen as a set of data items, which are grouped together according to some relatedness criteria, as, for instance, the set of data items created within the same year, or those created by the same author, or those about the same topic, say "recommender systems", or, more obvious, the set of data items belonging to the same digital archive. We assume that there is a set $\mathscr{L}$ of collections L and a membership function $\mu_L : \mathscr{D} \to \{0, 1\}$, where $\mu_L(d) = 1$ (for ease $d \in L$) indicates that the data item d belongs to the collection L. We also assume that there is at least one collection in $\mathscr{L}$, called *universal collection* and denoted $L_\top$, which includes all the data items $d \in \mathscr{D}$. Note that a data item may well belong to several collections. Furthermore, we do not specify whether the collections are materialized or are just "views" over $\mathscr{D}$. This does not play a significant role in our context. Finally, like for communities, we will assume that for each collection $L \in \mathscr{L}$ there is a *collection administrator* (a user $u^L \in \mathscr{U}$), who is in charge of defining both the collection L and the access policies to it.

Folders. Finally, we have *folders*. Essentially, a folder is a container for data items. A folder should be seen as the main environment in which users will carry out their work. As is common in many applications, folders may be organized by users according to their own folder hierarchy. Each user has a set of hierarchically organized folders, each of which may be seen as a thematic container of the user's selected data items. In fact, each folder typically corresponds to one subject (or discipline, or field) the user is interested in.

In order to accomplish a truly personalized interaction between user and system, this correspondence is implemented in a way, which is fully idiosyncratic to the user; this means that e.g., a folder named *Knowledge Representation and Reasoning* and owned by user `Tim` will not correspond to any "objective" definition or characterization of what "knowledge representation and reasoning" is, but will correspond *to what Tim means by* "knowledge representation and reasoning", i.e., to his personal view of (or interest in) "knowledge representation and reasoning". As we will see later on, this user-oriented view of folders is realized by learning the "semantics of folders" from the current contents of the folders themselves.

An important aspect in our model is that we will allow two types of folders:

- *private folders*, owned by a user only. This kind of folder can only be accessed and manipulated by its owner. Like in e-mail programs where the incoming e-mails is placed into folders, in a private folder a user puts all the data items he gathers from a resource and are worth saving. All the data items belonging to a private folder are invisible to other users; and
- *community folders*, which can be accessed and manipulated by all members of the community that owns the folder. Community folders are used to share data items with other users and to build up a common folder hierarchy. Community folders may also contain *discussion forums* (a kind of data item) where notes may be exchanged in threaded discussions (similar to news groups).

Formally, we assume that there is a set $\mathscr{F}$ of (either private or community) folders F. For each user u, with $\langle \mathscr{F}^u, \preceq^u \rangle$, we indicate the user's folder hierarchy, where $\mathscr{F}^u \subseteq \mathscr{F}$, $\preceq^u$ is a tree-like order on $\mathscr{F}^u$ and with $F^u_\top$ we indicate its *home folder* or *top folder*, i.e., the root folder of the hierarchy $\langle \mathscr{F}^u, \preceq^u \rangle$. Furthermore, given a folder $F \in \mathscr{F}$, we assume that (i) there is a membership function $\mu_F : \mathscr{U} \to \{0,1\}$, where $\mu_F(u) = 1$ (for ease $F \in u$) indicates that the folder F belongs to the user's u folder hierarchy, i.e., $F \in \mathscr{F}^u$; (ii) there is a membership function $\mu_F : \mathscr{C} \to \{0,1\}$, where $\mu_C(d) = 1$ (for ease $F \in C$) indicates that the folder F is a community folder and belongs to the community C; and (iii) there is a membership function $\mu_F : \mathscr{D} \to \{0,1\}$, where $\mu_F(d) = 1$ (for ease $d \in F$) indicates that the data item d belongs to folder F.

Fig. 1 shows an example of community, users, data items and their relations. In it, users u_1 and u_2 belong to the same community C_1. User u_2 has no private folders, while F_4 and F_5 belong to the same community C_1 and are accessible to both users.

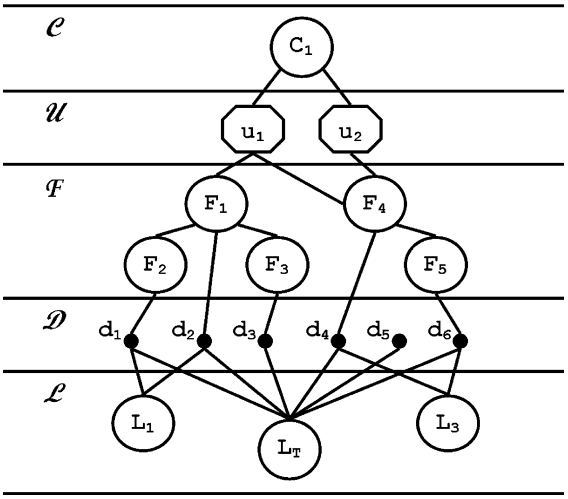

Fig. 1. Personalized information space organization.

2.3. Functionality

At any time, the user performs her actions with respect to the *current folder* (at the beginning it is the user's home folder—aka top folder). The current folder determines the (*semantical*) *context* of the user. This allows the system to provide to the user personalized functionality, especially during a search and/or recommendation task.

A user may perform a set of actions (described below), depending whether she is a member of a community or not, and whether she is a collection administrator or a community administrator. The functionality can be grouped into five categories: those pertaining to the management of folders, to the collections, to the communities, to the search and to the recommendation activities.

Folder management. A user can perform basic folder management actions on the folders she has access to: (i) with respect to "folder hierarchy", folder management operations include creating a new folder as a child of an existing folder, deleting a folder, moving a subfolder from an existing parent folder to a new parent folder (community administrators are allowed to manage the folder hierarchy of a community); and (ii) with respect to "folder content", folder management actions include saving data items from a search session in folders (see below), deleting, undeleting and destroying data items, moving and copying data items from one folder to another, rating and annotating data items, downloading and uploading data items.

Collection management. A collection administrator can create, edit, delete and define the access policies of collections. New collections may be defined in terms of others, e.g., using meet, join and refinement operators, and, thus a whole hierarchy of collections can be created, tailored to individual needs and/or to the needs of a community. In this way we move on to a more sophisticated information space model, where the 'physical archives' are hidden from the end-user. The users and communities may organize their information resources space according to their personal needs, by defining appropriate collections.

Collaborative support. Collaboration between users is supported through the possibility of sharing community folders along with their contents and folder structure. Discussion forums may be created within folders to allow informal exchange of notes and arguments. Rating and annotation of data items also may take the form of discussions among the members of a community. In order not to loose shared activity in the collaborative DL environment, mutual awareness may be supported through event icons (a kind of data item) displayed in the environment. Activity reports that are daily received by e-mail may also be possible. Also, users may view the list of all existing communities so that they become aware of ongoing community activity. This does not mean that they can look inside communities, since only, e.g., the title, the description and the identity of the community administrator are available. To become a member, users may directly join the community if this is allowed by the community's policy, or may contact the administrator to be invited to the community. In summary, collaboration support concerns with inviting or removing members to or from a community, leaving a community, viewing communities, joining a community (only for communities open to subscription), contacting community managers or other users (e.g., via e-mail), creating discussion forums, adding notes to a discussion forum, editing event notification preferences (icons, daily report) and rating data items.

Search data items. The user can issue a query q, whose result is an ordered subset (the result list) of data items $d \in \mathscr{D}$. The user is allowed to store selected data items of the result list within her folder hierarchy. There are different types of search.

In *ad-hoc search* a user u specifies a query q (we do not specify the syntax of queries, which depends on the indexing capabilities of the underlying DL) and the action of the system will be, according to the user specified options:

- to look for relevant data items within a set of user specified folders $F_i \in \mathscr{F}^u$ that she has access to, i.e., to search within $\{d \in \mathscr{D} : d \in F_i\}$; and
- to search within a specified set of collections $C_1, \ldots, C_n$, i.e., to search within $\bigcup_i \{d \in \mathscr{D} : d \in C_i\}$.

We further allow two types of *filtered search*:

- *personalized search* is like the usual ad-hoc search, except that the user u specifies a query q, optionally a list of collections C_i, and a folder $F \in u$, and the action of the system will be to look for data items $d \in \mathscr{D}$ (restricted to the collections C_i, if specified), such that d is relevant both to the query and to the folder F; and
- *what's new search*, which is as the *personalized search*, except the user specifies no query, but a folder $F \in u$, and the action of the system will be to look for data items $d \in \mathscr{D}$ such that d is relevant to the folder F and *has not yet been retrieved by the user with respect to the current folder*. That is, the user is interest in looking for data items relevant to a user specified folder, which have yet to be included in the information space of the system since the last time the user looked.

Technically, for all types of searches there exist widely known methods. Ad-hoc search is the usual task of information retrieval (Salton & McGill, 1983), while filtered search may be accomplished in at least two ways: (i) through techniques of query expansions (see Carpineto, De Mori, Romano, & Bigi, 2001), i.e., we expand the query q with significant terms of the folder profile f of F and then submit the expanded query; or (ii) we first issue the query q as an ad-hoc query, and then filter the result list with respect to the folder profile (see Belkin & Croft, 1992; Callan, 1998; Faloutsos & Oard, 1995; Mostafa, Mukhopadhyay, Lam, & Palakal, 1997). A discussion about advantages and disadvantages of both methods for implementing filtered search can be found in Amato and Straccia (1999).

Recommendation. A user may get recommendations of *data items*, *collections*, *users*, and *communities*, which are issued to users based on other users' (implicit or explicit) ratings, and on the perceived similarity between the interests of the user, as represented by a given folder, and the interests of these other users, as represented by their folders. All recommendations are specific to a given user folder. That means that they have always to be understood in the context not of the general interests of the user, but of the specific interests (topic) of the user represented by a folder.

Without doubt, the above set of actions provides us an enhanced personalized collaborative DL environment. Several of the items above are eligible to be the subject of deeper investigations but, in this paper we will put more emphasis on the recommendation issue.

3. An application: CYCLADES

An instantiation of the model of a personalized collaborative DL environment we have presented, is implemented in the CYCLADES system, which is accessible from the CYCLADES home page [3]

`http://www.ercim.org/cyclades`

The objective of CYCLADES is to provide an integrated environment for users and groups of users (communities) that want to use, in a highly personalized and flexible way, 'open archives', i.e., electronic archives of documents compliant with the Open Archives Initiative [4] (OAI) standard. Informally, the OAI is an agreement between Digital Archives in order to provide interoperability between them. In particular, the OAI defines an easy-to-implement harvesting protocol over HTTP, which give *data providers* (the individual archives) the possibility to make the documents' metadata in their archives externally available. This external availability of the metadata records then makes it possible for *service providers* to build higher

[3] Currently, the login page is http://calliope.ics.forth.gr:7007/Cyclades/LoginForm.html
[4] http://www.openarchives.org

levels of functionality. To date, there is a wide range of archives available in terms of content, i.e., the family of OAI compliant archives is multidisciplinary in content.

Under the above definition, *CYCLADES is an OAI service provider*. As a consequence, the set $\mathscr{D}$ of data items includes the set of metadata records harvested from OAI compliant archives. CYCLADES provides an open collaborative virtual archive environment, which (among others) supports users and communities (and their members) with functionality for

- advanced search in *large*, *heterogeneous*, *multidisciplinary digital archives*;
- collaboration;
- filtering; and
- recommendation.

From a logical point of view we may depict the functionality of the CYCLADES system as in Fig. 2.

Compliant with the model specified in Section 2, the functionality of the CYCLADES system can be grouped into four categories related to *collaboration*, *search*, *filtering*, *recommendation*, and of data items grouped into *collections*. The CYCLADES system supports indeed all the functionality described in the model.

Fig. 3 shows the main user interface.

On it, you may recognize the home (top level) folder of a user. It contains several folders. Among them, there are some (shared) folders belonging to communities (created by someone) to which the user joined, like the 'Physics-Gravity' folder, while others are private folders and have been created directly by the user, for instance, the 'Logic Programming' folder. These folders contain community or user collected OAI records relevant to some topics (e.g., gravity and logic programming, respectively). Fig. 4, for example, shows the content of the folder 'Physics-Gravity', the folder of the community of physicists. In it there are several other folder and metadata records. Some records have been rated (e.g., the 'Astronaut Protection ...' record) and some records have notes attached (e.g., 'The Lunar Scout ...' record). There is also a discussion forum. These functions are only some of those pertaining to the collaborative support package. The CYCLADES system already provided some record, community, collection and user recommendations deemed by the system as relevant to this folders.

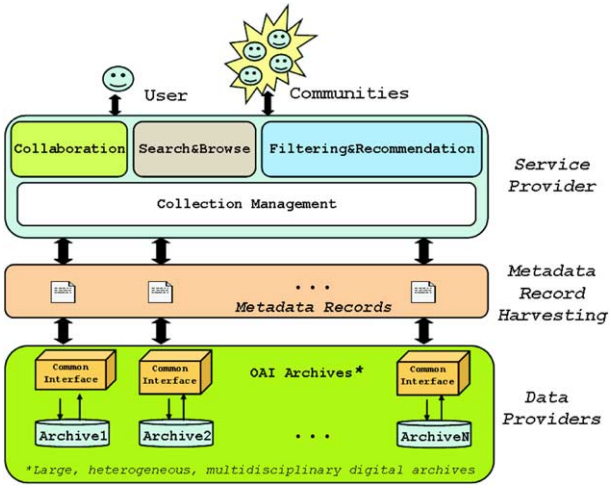

Fig. 2. Logical view of CYCLADES functionality.

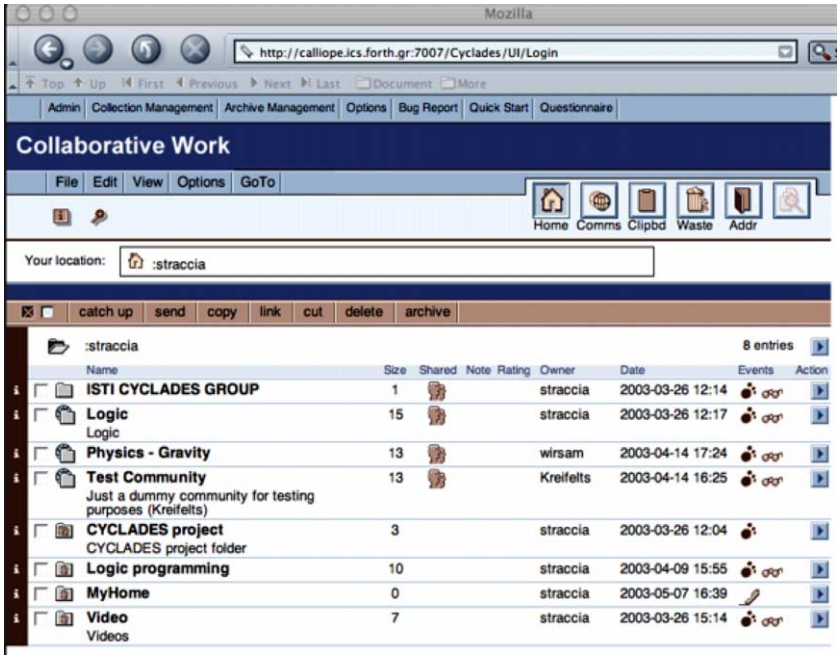

Fig. 3. User interface: home folder.

Fig. 5 shows the formulation of a simple query and the result of its execution. It should be noted that at this point a user is allowed to save some user selected records to a folder. This is the main way to populate folders with records gathered from a CYCLADES information space (i.e., the OAI archives).

The architecture of the CYCLADES system is depicted in Fig. 6. It should be noted that each box is a Web service distributed over the Internet. The CYCLADES system, accessible through Web browsers, provides the user with different environments, according to the actions the user wants to perform. The functionality CYCLADES provides have been developed by different services described below.

The *Collaborative Work Service* provides the folder-based environment for managing metadata records, queries, collections, external documents, received recommendations, ratings and annotations. Furthermore, it supports collaboration between CYCLADES users by way of folder sharing in communities, discussion forums and mutual awareness. One component of this service is the *Rating Management Service*, which stores and manages user's ratings.

The *Search and Browse Service* supports the activity of searching records from the various collections, formulating and reusing queries, associated with the folder by the user, and browsing schema, attribute values, and metadata records.

The *Access Service* is in charge of interfacing with the underlying metadata archives. In this project, only archives adhering to the OAI specification will be accounted for; however, the system is extensible to other kinds of archives by modifying the Access Service only. A user may ask CYCLADES to include newly OAI compliant archives as well.

The *Collection Service* manages collections (their definition, creation, and update) and stores them, thus allowing a dynamic partitioning of the information space according to the users' interests, and making the individual archives transparent to the user.

The *Filtering and Recommendation Service* provides filtered search, recommendations of records, collections, users, and communities deemed relevant to the user's interests.

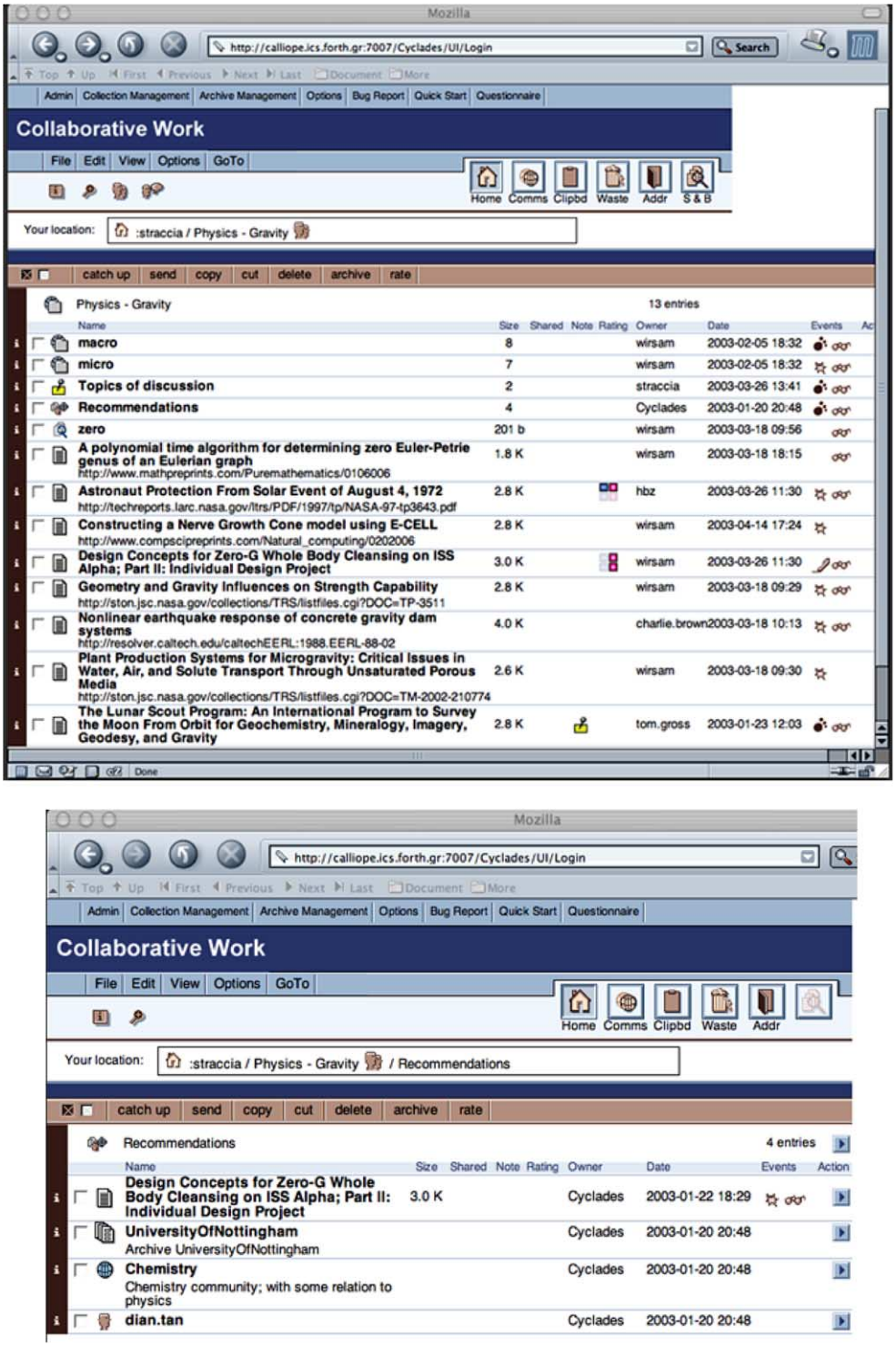

Fig. 4. User interface: folder content and recommendations.

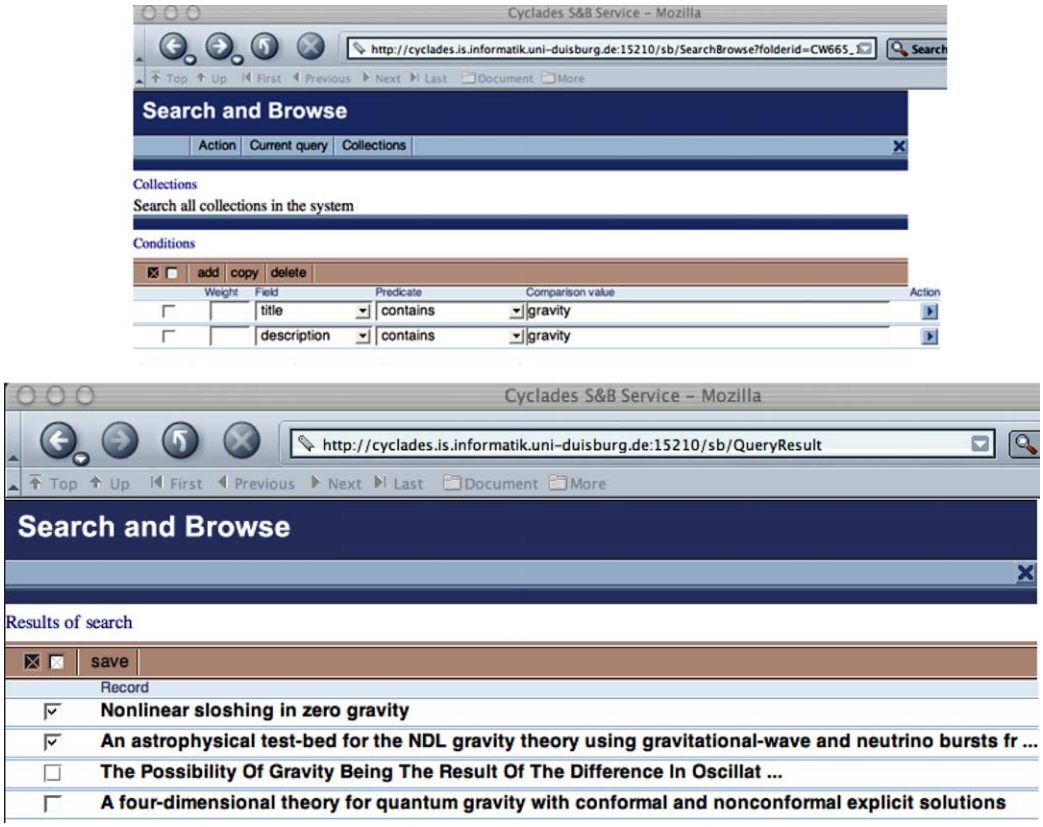

Fig. 5. User interface: simple query and query result.

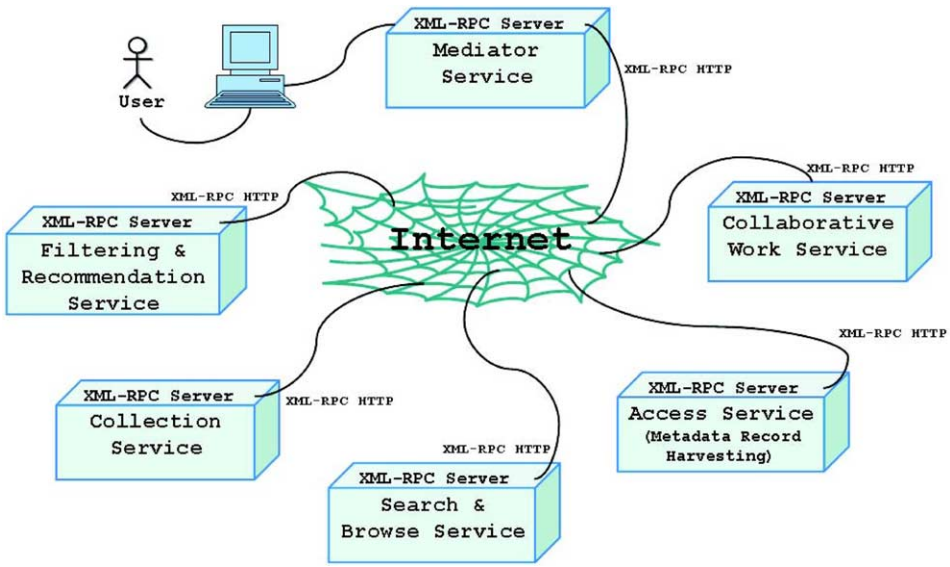

Fig. 6. Architecture.

The *Mediator Service*, the main entry point to the CYCLADES system, acts as a registry for the other services, checks if a user is entitled to use the system, and ensures that the other services are only called after proper authentication.

All of these services interoperate in a distributed environment. Security and system administration are provided centrally (by the Mediator Service). The CYCLADES services can run on different machines, and needs only an HTTP connection to communicate and collaborate. Indeed, the services communicate using XML-RPC, a simple protocol for implementing cross-platform, distributed applications. This protocol is based on Internet standards: method calls and responses are transmitted using HTTP, and the bodies of the calls and responses are encoded in XML.

The *Collaborative Work Service*, the *Search and Browse Service*, the *Access Service* (for archive management), and the *Collection Service* (for collection management) provide their own user interfaces. The *Mediator Service* itself provides the registration and login interface, and a system administration interface (for assigning access rights, etc.). Additionally, the *Mediator Service* integrates the user interfaces of the other services, and makes sure that those services and their interfaces are called only for authorized users, and only via the *Mediator Service*.

3.1. Recommendation algorithms in CYCLADES

A consequence of our environment is that, (i) by allowing users to organize the information space according to their own subjective view; and (ii) by supporting a collaborative environment, it is possible to provide a set of recommendation functions that, to the best of our knowledge, have not yet been investigated. Indeed, the recommendations regard not only the data items and the collections made available by the DL, but also the users and communities. We will just sketch out the algorithms (see Renda & Straccia, 2002 for more details). The algorithms below are those implemented in the CYCLADES system.

Preliminaries. For ease of presentation, we will assume that data items are pieces of text (e.g., text documents). It is worth noting that our algorithms can be extended to manage data items of different media types, like audio and video. By t_k, d_j, and F_i we will denote a text term, a data item, and a folder, respectively. Terms are usually identified either with the words, or with the stems of words, occurring in data items. For ease, following the well-known vector space model (Salton & McGill, 1983), a data item d_j is represented as a vector of *weights* $d_j = \langle w_{j1}, \ldots, w_{jm} \rangle$, where $0 \leqslant w_{jk} \leqslant 1$ corresponds to the "importance value" that term t_k has in the data item d_j, and m is the total number of unique terms in the indexed universal collection $L_\top$. The *folder profile* (denoted f_i) for folder F_i is computed as the *centroid* of the data items belonging to F_i. This means that the profile of F_i may be seen as a data item itself (Belkin & Croft, 1992) (i.e., the mean, or prototypical, data item of F_i) and, thus, is represented as vector of weighted terms as well, $f_i = \langle w_{i1}, \ldots, w_{im} \rangle$. Of course, more complicated approaches for determining the folder profile may be considered as well, e.g., taking into account the hierarchical structure of the folders (Dumais & Chen, 2000). Conceptually, they do not change much in our algorithm. Given a folder F_i, data item $d_j \in F_i$ and user $u_k \in \mathscr{U}$ such that $F_i \in u_k$, by $0 \leqslant r_{ijk} \leqslant 1$ we denote the *rating* given by user u_k to data item d_j relative to folder F_i (a data item within a community folder may be accessed, e.g., read, annotated and rated, by many different users). We further assume that whenever a data item d_j belongs to a folder F_i of a user u_k, an *implicit* default rating $\breve{r}$ is assigned. Indeed, the fact that $d_j \in F_i \in \mathscr{F}^{u_k}$ is an implicit indicator of being d_j relevant to folder F_i for user u_k. Finally, we average out the ratings r_{ijk} given by users u_k relative to the same data item–folder pair (i, j) and indicate it as r_{ij}.

In summary, we may represent:

- the data items as a 2-dimensional matrix, where a row represents a data item d_j and a column represents a term t_k. The value of the cell is the weight w_{jk} of term t_k in the data item d_j;

- the folder profiles as a 2-dimensional matrix, where a row represents a folder profile f_i and a column represents a term t_k. The value of the cell is the weight w_{ik} of term t_k in the folder profile f_i;
- the ratings as a 2-dimensional matrix, where a row represents a folder F_i and a column represents a data item d_j. The value of the cell is the rating r_{ij}.

The three matrixes are shown in Table 1, where $v = |\mathscr{F}|$ is the number of folders and $n = |L_\top|$ in the number of data items.

The *content similarity* (denoted $CSim(\cdot,\cdot)$) between two data items, or between a data item and a folder profile, or between two folder profiles, is the computation of a correlation (e.g., *cosine*) among two rows within the matrixes (a) and (b) of Table 1. Similarly, the *rating similarity* of two folders F_1 and F_2 (denoted $RSim(F_1, F_2)$) can be determined as a correlation (e.g., *Pearson correlation coefficient*) (Breese et al., 1998; Herlocker et al., 1999) between two rows of the matrix (c) in Table 1. Finally, the *similarity* (denoted $Sim(F_1, F_2)$) between two folders F_1 and F_2, which takes into account both the content and collaborative aspects, can be determined as a linear combination between their content similarity and their rating similarity.

Our recommendation algorithms follow a similar four-step schema described roughly below. In what follows, let u be a user and let $F \in u$ be a folder (the *target folder*) for which the recommended items should be found. The sketch of the algorithm is as follows:

- select a set of most similar folders F_i to F, according to the similarity measure *Sim*;
- from the set of selected folders, determine a pool of possible recommendable items;
- for each of the items in the pool, compute a recommendation score;
- select and recommend a subset of items with highest score, and not yet recommended to F.

Table 1
(a) The data item matrix, (b) the folder profile matrix, (c) the folder-data item rating matrix

	t_1	...	t_k	...	t_m
(a)					
d_1	w_{11}	...	w_{1k}	...	w_{1m}
d_2	w_{21}	...	w_{2k}	...	w_{2m}
...	...	...	...	...	...
d_j	w_{j1}	...	w_{jk}	...	w_{jm}
...	...	...	...	...	...
d_n	r_{n1}	...	w_{nk}	...	w_{nm}
(b)					
f_1	w_{11}	...	w_{1k}	...	w_{1m}
f_2	w_{21}	...	w_{2k}	...	w_{2m}
...	...	...	...	...	...
f_i	w_{i1}	...	w_{ik}	...	w_{im}
...	...	...	...	...	...
f_v	w_{v1}	...	w_{vk}	...	w_{vm}
	d_1		d_j		d_n
(c)					
F_1	r_{11}	...	r_{1j}	...	r_{1n}
F_2	r_{21}	...	r_{2j}	...	r_{2n}
...	...	...	...	...	...
F_i	r_{i1}	...	r_{ij}	...	r_{in}
...	...	...	...	...	...
F_v	r_{v1}	...	r_{vj}	...	r_{vn}

We proceed with a more detailed description of the above algorithm, specialized for the two cases: recommendation of users [5] and recommendation of data items.

Recommendation of users. The algorithm has the following schema:

- select a set $MS(F)$ of most similar folders to the target folder $F \in u$;
- for each folder $F_i \in MS(F)$, consider the users for which the folder F_i belongs to their folder hierarchy, i.e., compute the *pool of possible recommendable users* $P_U = \{u' \in \mathscr{U} : \exists F_i, F_i \in MS(F), F_i \in u'\} \setminus \{u\}$; [6]
- compute the recommendation score for each possible recommendable user, i.e., for each user $u' \in P_U$ determine the *user hits factor* $h(u') = |\{F_i : F_i \in MS(F), F_i \in u'\}|$ (the number of folders F_i judged as similar to the target folder F belonging to user u'). For each user $u' \in P_U$ the *recommendation score* $s(F, u')$ is computed as: $s(F, u') = h(u') \cdot \sum_{F_i \in MS(F), F_i \in u'} Sim(F, F_i)$;
- according to the recommendation score, select a set of most recommendable users, not yet recommended to the target folder F.

Note that the more a folder $F_i \in u'$ is similar to the target folder $F \in u$, the more related, in terms of interests, are the users u' and u. Additionally, the more similar folders belong to the same user u', the more this u''s interests overlap those of user u, which explains the recommendation score computation.

Recommendation of data items. This algorithm has much in common with the one we have seen above. The only difference concerns the computation of the recommendable data items and their recommendation score. Indeed, we will exploit the fact that data items are pieces of text and that there might be ratings associated:

- the *pool of possible recommendable data items* is determined by the set of data items belonging to the folders $F_i \in MS(F)$, i.e., $P_D = \{d \in \mathscr{D} : \exists F_i, F_i \in MS(F), d \in F_i\} \setminus \{d \in \mathscr{D} : \exists F' \in u, d \in F'\}$ (we do not want to recommend data items already known to the user);
- the recommendation score for $d_j \in P_D$ with respect to F is computed as a linear combination of a *content-based recommendation score* and a *rating-based recommendation score*. The content-based recommendation score of $d_j \in P_D$ with respect to the target folder F is the content similarity between d_j and the folder profile of F. The *ratings-based recommendation score* of d_j with respect to F is the weighted sum $s^{\mathrm{R}}(F, d_j) = \overline{r} + \frac{\sum_{F_i \in MS(F)} (r_{ij} - \overline{r}_i) \cdot RSim(f, f_i)}{\sum_{F_i \in MS(F)} \cdot RSim(f, f_i)}$, where $\overline{r}$ and $\overline{r}_i$ are the mean of the ratings in the target folder F and the mean of the ratings in the folder $F_i \in MS(F)$, respectively.

4. Conclusions

Since the Web, and consequently the information contained in it, is growing rapidly, every day a huge amount of "new" information is electronically published and new Digital Libraries are available to satisfy the user information needs. We described here a Digital Library that is not only an information resource where users may submit queries to get what they are searching for, but also a collaborative working and meeting space. Indeed, users looking within an information resource for relevant data might have overlapping interests, which may turn out to be of reciprocal interest for the users: users might well profit from each other's knowledge by sharing opinions and experiences. As such, we have formalized a personalized collaborative Digital Library environment in which the user functionality may be organized into four

[5] The recommendation of communities and collections are quite similar.

[6] Given two sets A and B, $A \setminus B$ means the elements in $A - B$.

categories: users may (i) search for information; (ii) organize the information space (according to the ''folder paradigm''); (iii) collaborate with other users sharing similar interests; and (iv) get recommendations. We also described the CYCLADES system, which is indeed an implementation of the environment. We are aware that many concepts and techniques presented in this paper are eligible to be the subject of further investigations, which we will address in the future.

Acknowledgements

This work is funded by the European Community in the context of the CYCLADES project IST-2000-25456, under the Information Societies Technology programme.

References

Amato, G., & Straccia, U. (1999). User profile modeling and applications to digital libraries. In *Lecture Notes in Computer Science: vol. 1696. Proceedings of the 3rd European conference on research and advanced technology for digital libraries (ECDL-99), Paris, France* (pp. 184–197). Springer-Verlag.

Belkin, N. J., & Croft, B. W. (1992). Information filtering and information retrieval: two sides of the same coin? *Communications of the ACM, 35*(12), 29–38.

Billsus, D., & Pazzani, M. J. (1998). Learning collaborative information filters. In *Proceedings of the 15th international conference on machine learning, San Francisco, CA, USA* (pp. 46–54).

Bollacker, K., Lawrence, S., & Giles, C. L. (1999). A system for automatic personalized tracking of scientific literature on the Web. In *Proceedings of the 4th ACM/IEEE joint conference on digital libraries (DL-99), Berkeley, CA, USA* (pp. 105–113). ACM Press.

Breese, J. S., Heckerman, D., & Kadie, C. (1998). Empirical analysis of predictive algorithms for collaborative filtering. In *Proceedings of the 14th annual conference on uncertainty in artificial intelligence, Madison, Wisconsin, USA* (pp. 43–52).

Callan, J. (1998). Learning while filtering documents. In *Proceedings of the 21st annual international ACM SIGIR conference on research and development in information retrieval (SIGIR-98), Melbourne, Australia* (pp. 224–231).

Carpineto, C., De Mori, R., Romano, G., & Bigi, B. (2001). An information-theoretic approach to automatic query expansion. *ACM Transactions on Information Systems, 19*(1), 1–27.

Di Giacomo, M., Mahoney, D., Bollen, J., Monroy-Hernandez, A., & Meraz, C. M. R. (2001). Mylibrary, a personalization service for digital library environments.

Dublin Core (WWW). Dublin Core Resource Page: http://dublincore.org/.

Dumais, S., & Chen, H. (2000). Hierarchical classification of Web content. In *Proceedings of the 23rd annual international ACM SIGIR conference on research and development in information retrieval, Athens, Greece* (pp. 256–263).

Faensen, D., Faulstich, L., Schweppe, H., Hinze, A., & Steidinger, A. (2001). Hermes: a notification service for digital libraries. In *ACM/IEEE joint conference on digital libraries (DL-01), Roanoke, Virginia, USA* (pp. 373–380).

Faloutsos, C., & Oard, D. W. (1995). A survey of information retrieval and filtering methods. Technical Report CS-TR-3514, University of Maryland.

Fernandez, L., Sanchez, J. A., & Garcia, A. (2000). Mibiblio: personal spaces in a digital library universe. In *Proceedings of the 5th international ACM/IEEE joint conference on digital libraries (DL'00), San Antonio, Texas, USA* (pp. 232–233).

Foltz, P. W., & Dumais, S. T. (1992). Personalized information delivery: an analysis of information filtering methods. *Communications of the ACM, 35*(12), 51–60.

Fox, E. A., & Marchionini, G. (2001). Digital libraries: introduction. *Communications of the ACM, 44*(5), 30–32.

Goldberg, D. J., Nichols, D., Oki, B. M., & Terry, D. (1992). Using collaborative filtering to weave information tapestry. *Communications of the ACM, 35*(12), 61–70.

Goncalves, M., Zafer, A. A., Ramakrishnan, N., & Fox, E. A. (2001). Modeling and building personalized digital libraries with PIPE and 5SL. In *Proceedings of the 43rd Joint DELOS-NSF workshop on personalization and recommender systems in digital libraries, Dublin, Ireland* (pp. 67–72).

Herlocker, J. L., Konstan, J., Borchers, A., & Riedl, J. (1999). An algorithmic framework for performing collaborative filtering. In *Proceedings of the 22nd annual international ACM SIGIR conference on research and development in information retrieval, Berkeley, CA, USA* (pp. 230–237).

Information Filtering Resources (accessed on 01/02/2004). Information Filtering Resources: http://www.enee.umd.edu/medlab/filter.

Mostafa, J., Mukhopadhyay, S., Lam, W., & Palakal, M. (1997). A multilevel approach to intelligent information filtering: model, system, and evaluation. *ACM Transactions on Information Systems, 15*(4), 368–399.

Moukas, A. (1996). Amalthaea: information discovery and filtering using a multiagent evolving ecosystem. In *Proceedings of the 1st international conference and exhibition on the practical application of intelligent agents and multiagent technology, London, GB*.

Renda, M. E., & Straccia, U. (2002). A recommendation system in a collaborative digital library environment. Technical Report 2002-TR-06, Istituto di Elaborazione dell'Informazione—CNR, Pisa, Italy.

Resnick, P., Iacovou, N., Suchak, M., Bergstorm, P., & Riedl, J. (1994). GroupLens: an open architecture for collaborative filtering of netnews. In *Proceedings of the ACM conference on computer supported cooperative work, Chapel Hill, NC, USA* (pp. 175–186). ACM.

Rocha, L. M. (1999). Talkmine and the adaptive recommendation project. In *Proceedings of the 4th international ACM/IEEE joint conference on digital libraries (DL'99), Berkeley, CA, USA* (pp. 242–243).

Rucker, J., & Polanco, M. J. (1997). Siteseer: personalized navigation for the Web. *Communications of the ACM, 40*, 73–76.

Salton, G., & McGill, J. M. (1983). *Introduction to modern information retrieval*. Reading, Massachussetts: Addison Wesley Publ., Co.