Openness in Shared Hypermedia Workspaces: The Case for Collaborative Open Hypermedia Systems

Jörg M. Haake, GMD-IPSI, Germany
joerg.haake@gmd.de
Uffe K. Wiil, Peter J. Nürnberg, *Aalborg University Esbjerg, Denmark*
{ukwiil, pnuern}@cs.aue.auc.dk

Abstract

Group work requires much flexibility regarding both the organization of shared workspaces and the organization of collaborative work. Shared hypermedia workspaces can provide this flexibility. However, this requires the provision of openness with respect to both the hypermedia workspace and the collaboration support offered in such a collaborative open hypermedia system. In this paper we address the issue of how to identify key requirements of open collaborative hypermedia systems. We start with a scenario in order to come up with a preliminary set of requirements. We then apply a regular schema to generate a more comprehensive set of requirements for open collaborative hypermedia systems.

1 Introduction

In this paper, we give a report of ongoing work between our research groups at Aalborg University Esbjerg and GMD-IPSI. From a historical perspective, research in hypermedia at GMD-IPSI has maintained a strong interest in hypermedia-based support for collaborative work while the researchers now at Aalborg University Esbjerg have a strong background in open hypermedia systems and the extension of hypermedia systems with cooperative functionality. This common interest in cooperative open hypermedia systems led us to carry out joint (and ongoing) research to establish what open collaborative hypermedia systems are and how they should be built.

1.1 Why collaborative open hypermedia systems?

There is an ongoing trend towards distributed and virtual organizations. As a consequence, distributed teams will perform many tasks in these organizations. Usually, co-workers in a team distribute work among themselves. Co-workers can then use their own work procedures and their favorite tools to do their work. Coordination and cooperation in the team requires some agreement on data exchange and document formats, and on policies to coordinate their work. However, co-workers usually enjoy much freedom in carrying out their work, which they can use to adapt their work practices to the task and team at hand. The downside of this is that the assembling of group results, the assessment of the current state of the whole work process, and the assessment of the joint results are each difficult. Integration of results, coordination among team members and adaptation to the task require much effort from the team members.

Two important issues in such a team are the organization of the overall shared workspace of the group, and the execution of cooperative work within such a shared workspace. When organizing a shared group workspace, one has to deal with the problems of emergent structures and of the integration of new material, which were unplanned but arise during the course of the group's work. For example, a changing understanding of the task as work progresses causes emergent structure. Different and changing demands of the task lead to the problem of dealing with a non-foreseeable multitude of information and document types in the shared workspace. Execution of cooperative work in a shared workspace requires support for communication, coordination, and cooperation of distributed team members. Different teams, organizations, and tasks etc. require different forms of support.

Here, we have to face a problem similar to the organization of the shared workspace: the problem of adapting support for cooperative work to changing needs while teamwork proceeds. Both problems require some form of "openness" from an underlying support system: that is, openness with respect to new forms of workspace structure and content, and openness with respect to cooperation support within such a shared workspace.

To address these problems we believe an open systems approach is required. Such an approach should combine open hypermedia services and open collaboration services. Open hypermedia services can be used as a means of structuring workspaces in a flexible way. Likewise, open collaboration services can be used to organize the collaboration processes in a flexible manner. In all, collaborative open hypermedia systems seem to be a promising approach with which to address the above problems.

1.2 A road towards collaborative open hypermedia systems

For the last 5 years, the Open Hypermedia Systems Working Group (OHSWG) has explored how openness can be realized for collaborative hypermedia systems in two respects. Firstly, with respect to new forms of structure and content of hypermedia information spaces, and secondly with respect to new tools for browsing and manipulating these information spaces [Wiil and Østerbye 1994, Wiil and Demeyer 1996, Wiil 1997, 1998, 1999]. Thus, it seems natural to build on the results of the open hypermedia systems community [Reich et al. 1999] and to investigate ways of extending open hypermedia systems with open collaboration services. Some results of this work are published elsewhere [Haake et al. 1999]. However, there are still many unanswered questions in this area. Among them, we will focus on the issue of requirements analysis, since it provides the basis for all subsequent design and implementation work.

1.3 Organization of this paper

In the next section we will present our requirements analysis for collaborative open hypermedia systems. Then, we will discuss some regularity within this set of requirements, which leads us to the proposal of a regular schema for deriving a more complete set of requirements. Finally, we discuss the implications of this approach for the design of collaborative open hypermedia systems.

2 Requirements on collaborative open hypermedia systems

In our work, we started with a scenario describing how a collaborative open hypermedia system might actually support an ongoing collaboration. We then took a closer look at the scenario in order to derive requirements for the underlying collaborative open hypermedia system. We looked at four different categories of requirements:

- Requirements concerning openness in cooperative applications
- Requirements concerning openness in hypermedia applications
- Requirements concerning the combination of open cooperative and open hypermedia applications
- Requirements concerning openness in general

2.1 A scenario

The following scenario is used to illustrate the use of an environment that provides both open collaboration and open hypermedia services. For simplicity, we use a team consisting of three team members: Ulrich, John and Paul. These three people form a team of co-authors who wish to write a joint journal paper. Each team member is located at a different site and works according to his own schedule. The team decides to use their open collaborative hypermedia authoring environment (which offers the open collaboration and open hypermedia services) as their primary means for writing the journal paper. This situation can be characterized as follows:

- co-authors are located at different sites;
- co-authors use their own favorite tools (e.g., different editors or hypermedia authoring tools);
- co-authors need to share common (but potentially distributed) hyperdocuments that represent their common workspace (including pre-products, drafts, and the final document); and,
- co-authors may collaborate both synchronously and asynchronously.

We will now consider a part of a hypothetical work process. The scenario is organized as a sequence of five phases, each demonstrating a different cooperation mode.

Individual work. After the team assembles and decides to write the paper, John starts with writing a first draft of the outline in his favorite hypermedia editor, DOLPHIN [Streitz et al., 1994]. In DOLPHIN, John creates a new hyperdocument, fills in nodes for the different top-level sections, and creates links among them. He uses the open hypermedia functionality to create some links to parts of existing documents (e.g., to indicate possible reuse or other relevant background material). So far, John has been working individually in his own workspace, which the hyperdocument he just created in DOLPHIN represents. After finishing the draft, John quits DOLPHIN and sends an e-mail containing a reference (i.e., a universal resource name, URN) of his workspace (represented by the hyperdocument he just created) to his co-authors. In this e-mail, he invites them to read and annotate the draft, and he proposes a date and time when they should discuss the draft in a telemeeting. Paul and Ulrich agree and send positive responses via e-mail.

Asynchronous work. Now, Ulrich accesses John's workspace. First, Ulrich uses his favorite editor, a service-enabled version of Emacs, to access the workspace. He asks his service-enabled Emacs to open the workspace. Simply by providing the URN, the service is able to request a copy of the homepage of the DOLPHIN document. Since the workspace is currently not in use, the service creates a new session automatically. Ulrich can now read and annotate the proposed outline by attaching new comment nodes via comment links. These new hypermedia objects are stored in a new workspace.

Loosely coupled heterogeneous work. Sometime later, Paul decides to work on the draft too. He is using a service-enabled version of MS Word. Like Ulrich, he opens the shared workspace using the URN. The service detects that a session is already operating over the same workspace. Paul is given the information about the running session (i.e., Ulrich's session) and decides to join this session. Now, there are two team members accessing the shared workspace. Since Ulrich decided to allow others to join his session, Paul was allowed to do so. The default setting for new sessions specifies that the so-called loosely coupled cooperation mode is selected. In this case, co-workers share the same workspace (data), but are permitted to navigate independently. Thus, Paul can navigate through the workspace freely. Whenever he visits a node currently in use by Ulrich, the service will display the presence and activities of Ulrich (e.g., by displaying Ulrich's name, by highlighting the objects currently edited, and by showing the changes). Of course, Ulrich is made aware of Paul's presence in the same way. Both can navigate through the workspace and follow links to the document parts that were originally created by John. When following such a link, the server decides whether the target document can be displayed in the current editor window or whether a new tool needs to be opened. However, even when opening a new tool, the service ensures that the loosely coupled cooperation mode is still maintained.

Tightly coupled heterogeneous work. Later on, Paul watches Ulrich changing

the structure of the outline of one section. Since Ulrich's changes would affect his own work, Paul decides to contact Ulrich directly by asking his editor to switch into tightly coupled cooperation mode. The editor informs the service about the requested change and the service switches the cooperation mode for the session accordingly. This leads to starting up a RealAudio connection between the two users (with the corresponding tools opening on their screens after Ulrich accepted the incoming conference call). Now, the two can discuss the proposed changes. Due to the fact that Paul and Ulrich are using different tools (Emacs and MS Word), they cannot enjoy shared views or telepointers. This would require the use of the same tool on both sides (or the sharing of one application via application sharing). After some discussion, Paul decides to stop working and quits his editor. Automatically, the service registers his decision to leave, thus leaving Ulrich as the only user in the session. Once Ulrich quits his editor, too, the service can end the session.

Tightly coupled homogeneous work. Sometime later, all three co-authors open their tools on the shared workspace (which at this time also contains the comments and changes of the co-authors). Since John opened the workspace with his DOLPHIN browser first, the other two co-authors simply join the workspace. Because John set the cooperation mode to tightly coupled and set the tool to be used to "DOLPHIN", Paul and Ulrich get DOLPHIN browsers opened on their screens, too. Now all three DOLPHIN browsers show the same page, provide telepointers and the WYSIWIS (What You See Is What I See) property [Stefik et al. 1987]. In addition, a RealAudio conference among the three sites is established by the server. The team can now discuss the outline, navigate as a group, point to positions in the shared browser window, and modify the workspace together. After they reach consensus, they document their new work plan as a structure consisting of linked task nodes in a separate top-level node in the workspace. Thus, the service is used for informal coordination purposes.

Work continues in the matter above.

2.2 Analysis

The above scenario shows an example of how a team can use a shared hypermedia workspace to support their collaborative editing task. This scenario can be used to extract requirements concerning systems, which aim to support this kind of work. We focus on four categories of requirements:

1. Requirements that arise from the definition of openness in general
2. Requirements that arise from the open nature of collaboration processes,
3. Requirements that arise from the open nature of the evolving shared information space,
4. Requirements that arise from the combination of open collaboration processes and open shared information spaces.

2.2.1 Requirements of openness in general

The quest for open services is not new. The provision of open services has been the goal of much previous work done, for instance in the hypermedia and CSCW fields as referred to in the introduction. The important characteristics of an open service in this paper, which need to be fulfilled as requirements, are:

- *(Req. 1) Availability of open services to an open set of applications*
  An open service should be available to an open set of applications in the computing environment. The service should be provided by computing entities or components in the computing environment that are potentially accessible by all applications (e.g., middleware components or operating system components).
- *(Req. 2) Orthogonality of open services*
  An open service should be orthogonal to other services used by participating applications (e.g., storage and display services). Applications should be able

to use the services of an open service without altering the existing services available in the application.

- *(Req. 3) Generality of open services*
  An open service should be general enough to be useful across applications. The service should be operational both internally in the application and across other applications of the same or different types — like for example "cut, copy and paste" services.
- *(Req. 4) Provision of different service levels*
  An open service should provide different levels of its services. Applications should be able to get exactly what they need in terms of services and levels of services.

2.2.2 Requirements of open collaboration processes

Collaboration in the scenario proceeds as a sequence of phases, each using a specific cooperation mode to support the specific style of collaboration employed by the team members. Openness applies here to:

- Documents and tools: the set of document types, formats and corresponding tools required by a team to perform its work.
- Organization of workspace: the set of collaboration styles adopted by a team.
- Support for coordination.

This leads us to the definition of the following requirements on open collaboration environments:

- *(Req. 5) Open set of documents and tools*
  A collaboration environment must be able to deal with every type of information the team needs. We see this need within our scenario when – during individual work – John links to and re-uses existing documents, and when all co-authors use their favorite editors.
- *(Req. 6) Open set of workspace organizations*
  Since teams use different working styles and procedures (dependent on the team members, their preferences, distribution, roles, the task at hand, etc.), a collaboration environment must be able to support a variety of different ways of collaborating. Therefore, an open set of workspace organizations (e.g. determining object types and relationships) must be supported. In the scenario, co-authors employ different working styles.
- *(Req. 7) Accommodate an open set of coordination policies*
  In order to enable productive group work, some coordination between the group members must be facilitated. This includes means for informal (e.g., shared plans) or formal (e.g., workflow) coordination. Openness refers here to the ability to accommodate different coordination policies. In our scenario, we see pre-planned draft passing and meetings as well as spontaneous meetings (e.g. when Paul visits a node in use by Ulrich).

Collaboration support in a cooperative environment usually requires management of shared data objects, management of shared user interfaces, support for group awareness, and support for communication:

- *(Req. 8) Support for sharing data between several (possibly concurrently working) users*
  Means for accessing shared data and for maintaining consistent states of shared data must be provided. Here, openness means that all types or sources of data can be accessed and shared, as repeatedly shown in the scenario.
- *(Req. 9) Support for coupling user interfaces for any tool*
  Support for maintaining shared display states are sometimes required (e.g., to provide WYSIWIS). Coupling of user interfaces must be supported, and means for propagating updates (caused by data changed by a remote collaborator) need to be provided. Here, openness means to be able to couple user interfaces for any tool team members might use, such as required by tightly coupled heterogeneous work in our scenario.

- *(Req. 10) Group awareness for any combination of tools*
  There needs to be some group awareness mechanism that allows the collaborators to recognize each other's presence and activities. In the scenario, this is used to allow Paul to recognize Ulrich's presence and to make it possible for him to join Ulrich's session. In an open system approach, this service must work for any combination of tools used by different clients.
- *(Req. 11) Support for communication channels and policies*
  If co-workers work on a shared workspace, some communication among group members is required. This communication can take place outside the shared environment (e.g., via e-mail or telephone) or inside the shared environment (e.g., by using either the shared workspace as a communication medium or other integrated communication channels such as audio/video conferencing). Thus, an open collaboration environment must support the communication channels and policies needed by the group. This is also exemplified in our scenario during tightly coupled work (synchronous case) and individual work (asynchronous case).

Usually, group work is organized in cooperative applications using the concept of a *session*. A session is defined by:

- a group of users (or the clients representing them);
- a common workspace on which these users work; and,
- a specific cooperation mode used by these users.

Session management includes operations for creating and destroying sessions, joining or leaving sessions, and selecting or negotiating cooperation modes or transitions between them. Usually, this functionality is implemented on top of some basic functionality for data sharing, concurrency control and update or notification management.

Each collaboration style to be used in a session requires a specific combination of support from the above services. We call such a combination a *cooperation mode* [Haake and Wilson, 1992]. Examples of aspects influenced by a cooperation mode are: the coupling of certain parameters of user interfaces (e.g., scrollbar positions); floor control policy (e.g., concurrent access vs. locking); concurrency control strategy; and, group awareness (e.g., showing the activities of others via telepointers). In addition, a session can determine the documents and tools used by the group of users (e.g., which tools are open, and which communication tools are configured in which way). Thus, a session reflects the context and the current state of the collaboration process.

Here, openness refers here to (1) the flexibility offered to define new cooperation modes and to switch between them; (2) the capability of integrating new tools (already addressed by Req. 5); and, (3) the possibility of integrating the needed information sources (already addressed by Req. 5). Thus, the following two requirements must be met in an open collaboration environment:

- *(Req. 12) Support for different cooperation modes*
  Our scenario shows examples for five different cooperation modes.
- *(Req. 13) Transitions between cooperation modes*
  As it was shown in our scenario, transitions between different cooperation modes must be supported.

2.2.3 Requirements of open shared information spaces

Collaboration in the scenario makes heavy use of a shared information space. Team members access this shared information space to change the structure and content of the workspace as required by their work. A number of requirements for services of an open hypermedia system (OHS) can be compiled from the scenario:

- *(Req. 14) Inter document linking capability*
  Users must be able create and

traverse links that go from one document type managed by one type of application to another document type managed by another type of application. In our scenario, this was the case in tightly coupled heterogeneous work, when co-authors used different tools (e.g. Emacs and Word). Thus, an open shared information space must be open to the integration of new applications that handle heterogeneous document types and formats.

- *(Req. 15) Intra document linking*
  Users must be able to create and traverse links that connect a part of a document with a part of another document. This is similar to the above requirement, except that a more fine-grained linking (and thus knowledge of the document format and constraints) is required.

- *(Req. 16) Open to new document types and formats*
  Requirements 14 and 15 both directly specify that an open shared information space must be open with respect to support of new document types and formats (including whatever structural constraints these document formats might have). In the scenario, this is visible when co-authors link any relevant material into the workspace. This is also in accordance with the corresponding requirement (Req. 5).

- *(Req. 17) Open to new computations over structure*
  Openness with respect to new applications and heterogeneous document types and formats leads to the need for support of computations on hypermedia structures. Since structures are likely to be presented in different manners by different applications requiring different types of computation, open shared information spaces must be open to new computation over structure. Examples of computations with structure are the computation of views, such as required by the presentation of the overall document structure in DOLPHIN's Navigation tool, or the presentation of embedded links in the OHS-enabled Emacs. In our scenario, this requirement would be illustrated if we were to add a spatial hypertext system, such as VIKI [Marshall et al. 1994]. VIKI includes a spatial parser, which computes aggregations based on spatial layout (which is another computation over structure). Imagine the use of this parsing algorithm by DOLPHIN to visualize spatial composites. This would require the spatial parser to offer an open hypermedia service that would be available to other tools.

The above requirements are derived from the associative structure domain. Similar requirements can be compiled from other structural domains. The latest trend in the open hypermedia systems community is to broaden their applicability by supporting hypermedia structures from structural domains other than the associative ([Nürnberg et al. 1997]). This leads to the final requirement:

- *(Req. 18) Open to new domain specific structures and computations*
  An open shared information space should be open to new structural domains. This includes openness to provide new domain specific structures and computations. As an example, consider support for argumentation phases during collaboration. Making hypermedia objects, which are specific to the argumentation domain (such as issues, positions, and specific links for argumentation) and the respective computations (such as computing argumentative cycles, contradicting arguments, or arguments without backing) available to other tools could provide better support for solving the task at hand. Since the need for these structures and computations arises during collaboration, means are required of defining new domain specific structures and computations (over hypermedia objects) in an open way. Likewise, the integration of existing open services offering computations over structure must be supported.
  In the future, this requirement is likely to be expanded into a set of require-

ments for each additional structural domain (such as associative hypermedia, spatial hypermedia) as these are explored in more detail in the context of open hypermedia systems.

2.2.4 Requirements of open collaborative information spaces

Collaboration in the scenario described in this paper makes heavy use of a shared information space, which also offers collaborative functionality. This combination of services leads to additional requirements.

In general, open collaborative information spaces need some way of dealing with the fact that content can be overlaid with structure, and that this does not change the content itself. Since structure can be much more than just annotations (e.g., links, composites, spatial structures and taxonomic structures), it is necessary to broaden the intermediate access pattern of early hypermedia authoring systems to deal with all kinds of structures. It is important to be able to distinguish between reading the document (reading both content and structure), overlaying the document with structure (writing structure) and writing the content of the document. Thus, access patterns in open collaborative hypermedia systems turn out to be more complex than the read-write pattern of (non-hypermedia) collaborative authoring systems and the read-annotate-write pattern of early hypermedia authoring systems.

This leads us to the definition of the following requirements on open collaborative information spaces:

- *(Req. 19) Access patterns for open collaborative information spaces*
  Different access patterns must be supported, including reading the document (reading both content and structure), overlaying the document with structure (writing structure) and writing the content of the document. In our scenario, this requirement would show if the co-authors decided to protect their private workspaces and to allow co-authors to comment, add material, or create new structure, but not to alter original content.
- *(Req. 20) Openness to new access patterns*
  Since different collaboration policies imply different roles of collaborators, new access patterns must be supported. Since collaboration policies usually develop and change during teamwork (e.g. to support different phases of group work), unforeseen collaboration policies as well as changes to existing policies must be supported. The scenario shows different cooperation modes, each of which could employ a different collaboration policy, requiring different or new access patterns.

In addition to access patterns, three different work settings must be supported by an open collaborative information space:

Work setting 1: individual work in a private workspace.

In this type of work setting, a single user works on his personal information located in a private workspace. The user will be able both to browse existing information and to author new information in her private workspace. New information can be created by adding new documents, by annotating documents, by overlaying documents with hypermedia structure, by grouping documents into collections, etc.

It is worth noticing that the individual components in this work setting (applications, middleware services and storage units) can run on different machines that are distributed across a (local or wide area) network.

Work setting 2: individual work in a shared workspace.

There are two fundamentally different ways to share information in this type of work setting. One can be part of a group of users that all have access and update privileges to information located in a shared workspace or one can share information with other users by creating information in a publicly accessible (shared) workspace. The latter cover four special cases distinguished by the number of readers and writers of the shared workspace. For example, a single writer can create information and make it

available to a limited group of users in a shared workspace. Another example is a group of writers who create information that is publicly available.

Many client-server hypermedia systems provide asynchronous types of collaborative work settings using underlying mechanisms such as user-defined locking and event notification [Wiil and Leggett 1993]. In this type of setting, several users can work on the same set of documents at the same time. Typically, only a single user can update (write) a document at a time (by acquiring a lock on the document). Other users might be able to follow the evolution of the document by having the document opened in read-only mode and subscribing to event notifications such as updates (saves) to the document.

Another situation resembles a typical WWW setting where a Web manager and some information providers place information in a shared (public) workspace. Only a few people can update this information, but it can be accessed (browsed) by the world.

Work setting 2 has the same services as were described for work setting 1 – i.e. browsing and authoring of information. In some situations all users are information providers and have authoring capabilities, while in other situations only the information providers have authoring capabilities.

Work setting 3: collaborative work in a shared workspace.

By combining shared workspaces with collaborative cooperation modes, some additional services and capabilities are gained. Shared workspaces allow the sharing of persistent information (e.g., documents and hypermedia structures). Different collaborative cooperation modes allow users different levels of session-oriented services. These additional session services enable that:

- multiple users jointly (synchronously) create hypermedia structures (in the case of a link this would mean that different users each add one or more endpoints to a shared link); and,
- multiple users jointly (synchronously) create new documents and edit existing documents by adding (updating) individual sections or paragraphs to the document.

Essentially, one can imagine that all the services and capabilities described under work setting 1 and work setting 2 can also occur in a synchronous manner involving multiple users operating at the same time potentially in different places (c.f., the time – space matrix [Ellis et al. 1991]).

In other situations, the set of users can be divided into the people who are creating and maintaining the information available in the shared (public) workspace and the outside users. The maintainers will have the session sharing capabilities as described above. The outside users will potentially be able to use the shared session capabilities to browse through the information in the workspace collaboratively. This could occur in an anarchistic manner or in a more controlled manner using turn-taking or floor control mechanisms.

Figure 1 shows the mapping between the different work processes (involving different cooperation modes) in the scenario and the work settings that were introduced above.

Work setting 1 covers individual work, setting 2 covers individual work and asynchronous work, and setting 3 covers

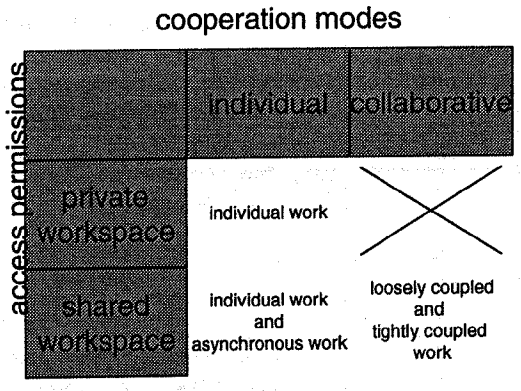

Figure 1. *How the scenario work processes map to the work settings.*

loosely coupled heterogeneous work, tightly coupled heterogeneous work, and tightly coupled homogeneous work.

The above discussion leads us to the following requirements on open collaborative information spaces:

- *(Req. 21) Support for different work settings*
  Different work settings (see the three identified above) needs to be supported. In our scenario, such work settings are used when co-authors work individually, or use asynchronous or synchronous cooperation modes.
- *(Req. 22) Support for switching between work settings*
  In the collaboration process, switching between work settings must be supported. In our scenario, transitions between the different cooperation modes illustrate this need. For example, during asynchronous collaboration one co-author joined another co-author. This caused a transition into the loosely coupled work situation.

3 Analysis of the requirements

The above requirements were derived by looking at openness from four different perspectives:

- openness in cooperative applications,
- openness in hypermedia applications,
- openness in combined cooperative hypermedia systems, and
- open systems in general.

When looking for regularity in these 22 requirements (see table 1), we actually noticed two common factors:

- an open set of some abstraction (e.g. documents, document types, tools, workspace organizations, coordination policies, cooperation modes, computations, access patterns, work settings, services) must be supported, and
- transitions between the individual members of one of the above open sets must be supported (e.g. within coordination policies, cooperation modes, access patterns, work settings).

Table 1: Summary of requirements

Req. No.	Caused by	Requirement Title
1	Open Systems in General	Availability of open services to an open set of applications
2		Orthogonality of open services
3		Generality of open services
4		Provision of different service levels
5	Open Cooperative Applications	Open set of documents and tools
6		Open set of workspace organizations
7		Accommodate open set of coordination policies
8		Support for sharing data between several (possibly concurrently working) users
9		Support for coupling user interfaces for any tool
10		Group awareness for any combination of tools
11		Support for communication channels and policies
12		Support for different cooperation modes
13		Transitions between cooperation modes
14	OHS	Inter document linking capability
15		Intra document linking
16		Open to new document types and formats
17		Open to new computations with structure
18		Open to new domain specific structures and computations
19	Combination of HM and CSCW	Access patterns for open collaborative hypermedia systems
20		Openness to new access patterns
21		Support for different work settings
22		Support for switching between work settings

Applying these two rules consistently to the original 22 requirements leads to a new set of requirements presented in table 2. In the following, we discuss some interesting observations made when looking at new or modified entries in table 2 (additions or modifications are highlighted in bold typeface).

From table 2 we can draw a number of observations:

1. Requiring both, the provision of an open set of some abstraction and transitions between them lead to the addition of previously missing requirements. For example, we now also include transitions between different

Table 2: Extended set of requirements

Req. No.	Caused by	Requirement Title
1	Open Systems in General	Availability of open services to an open set of applications
2		Orthogonality of open services
3		Generality of open services
4		Provision of different service levels
5	Open cooperative applications	Open set of documents and tools
5a		**Transitions between documents and tools**
6		Open set of workspace organizations
6a		**Transitions between workspace organizations**
7		Accommodate open set of coordination policies
8		Support for sharing data between several (possibly concurrently working) users
9		**Support for open set of couplings of user interfaces for open set of tools**
10		**Group awareness for open set of combinations of tools**
11		**Support for open set of communication channels and policies**
11a		**Transitions between communication channels and policies**
12		**Support for open set of (different) cooperation modes**
13		Transitions between cooperation modes **(can be considered equal to transitions between different couplings of user interfaces (R9) and between different combinations of tools (R10))**
14	OHS	Inter document linking capability **(can be considered equal to support linking between open set of documents)**
15		Intra document linking **(can be considered equal to support linking between open set of endpoints with one of an open set of documents)**
16		Open to new document types and formats **(open set of document types and formats)**
16a		**Transitions between document types and formats**
17		Open to new computations with structure **(open set of computations with structure)**
17a		**Transitions between computations with structure**
18		Open to new domain specific structures and computations **(open set of domain specific structures and computations)**
18a		**Transitions between domain specific structures and computations**
19	Combination of HM and CSCW	Access patterns for open collaborative hypermedia systems
20		Openness to new access patterns **(is equal to open set of access patterns)**
20a		**Transitions between access patterns**
21		Support for different work settings **(is equal to open set of work settings)**
22		Support for switching between work settings **(is equal to transitions between work settings)**

documents and tools (R5a). This requirement can also be seen as a justification for requiring open linking and document format or document type translations so that different tools can inter-operate on a shared set of documents.

2. Likewise, this also led to the inclusion of requirements to support switching between communication channels and policies (now reflected in R11a, which was previously missing).
3. Also, transitions between different computations with structure are now required (R17a). As an example, consider concatenations of computations over shared structure, which might require means for exchanging temporary results etc. As a consequence, this requirement must also hold for domain specific structures and computations (R18a).
4. Regarding access patterns for collaborative open hypermedia systems, we now also include transitions between access patterns (over time and situations) must be supported (R20a). Since access patterns are an important component of implementing different work settings and coordination policies, this requirement makes perfect sense.

Thus, it seems that applying the regular schema of asking for open sets and transitions between set members leads to a more comprehensive set of requirements. However, it is always the domain (in our case, open cooperation, open hypermedia and their combination) that determines the concrete abstractions and transitions.

4 Conclusions

In this paper, we presented the case for combining open hypermedia and open collaboration services in the form of open collaborative hypermedia systems. As pointed out in the introduction, group work requires much flexibility regarding both, the organization of shared workspaces and the organization of collaborative work. We identified important requirements of open collaborative hypermedia systems based on a scenario of collaborative work as well as taking into account more general requirements of open systems. We then focused on the idea of applying a regular schema to a preliminary set of requirements in order to come up with a more comprehensive set of requirements.

Our proposal for the design and implementation of open collaborative hypermedia systems has been reported in [Haake et al. 1999]. However, it remains to be seen as to whether the regularity reported in this paper can only be used to create more comprehensive requirements or whether it may also be used to discover more general design guidelines for open collaborative hypermedia systems. As a consequence, the corresponding architecture may very well be further simplified or turned into something with broader applicability. Since this is work in progress, we would appreciate any comments and suggestions. ❖

Acknowledgments

Many thanks to the members of the OHS working group for providing us with an excellent forum for open discussions. Many thanks also to SIGWEB for providing a forum for discussing work in progress. We also thank Shirley Holst for her valuable comments on previous drafts of this paper.

References

[Ellis et al. 1991] Ellis, C. A., Gibbs, S. J., and Rein, G. L. 1991. Groupware: some issues and experiences. *Communications of the ACM 34*(1):38-58.

[Haake and Wilson 1992] Haake, J. M., Wilson, B. 1992. Supporting Collaborative Writing of Hyperdocuments in SEPIA. *Proceedings of the ACM Conference on Computer-Supported Cooperative Work (CSCW '92), Toronto, Canada, October 31 - November 4, 1992*, pp. 138-146.

[Haake et al. 1999] Haake, J. M., Wiil, U. K., and Nürnberg, P. J. 1999. Having it both ways: Combining open collaboration and open hypermedia services. Submitted to *Computer*

Supported Cooperative Work: The Journal of Collaborative Computing. Kluwer.

[Marshall et al. 1994] Marshall, C., Shipman, F., and Coombs, J. 1994. VIKI: Spatial hypertext supporting emergent structure. *Proceedings of the European Conference on Hypermedia Technologies (ECHT 94)* (Edinburgh, Scotland, Sep), 13-23.

[Nürnberg et al. 1997] Nürnberg, P., Leggett, J., and Schneider, E. 1997. As we should have thought. *Proceedings of the Eighth ACM Conference on Hypertext (Hypertext 97)*, Southampton, UK, April 6-11, pp. 96-101.

[Reich et al. 1999] Reich, S., Wiil, U. K., Nürnberg, P. J., Davis, H. C., Grønbæk, K., Anderson, K. M., Millard, D. E., and Haake, J. M. 1999. Addressing Interoperability in Open Hypermedia: The Design of the Open Hypermedia Protocol. *New Review of Hypermedia and Multimedia*. London: Taylor Graham Publishing. In press.

[Stefik et al. 1987] Stefik, M., Bobrow, D. G., Lanning, S., and Tatar, D. 1987. WYSIWIS revised: Early experiences with multiuser interfaces. *ACM Transactions on Office Information Systems* 5(2), 147-167.

[Streitz et al. 1994] Streitz, N., Geissler, J., Haake, J., and Hol, J. 1994. DOLPHIN: Integrated meeting support across local and remote desktop environments and liveboards. *Proceedings of the 1994 ACM Conference on Computer-Supported Cooperative Work (CSCW'94), Chapel Hill, N.C., October 22-26, 1994*, pp. 345-358.

[Wiil and Leggett 1993] Wiil, Uffe K., and Leggett, John J. 1993. Concurrency Control in Collaborative Hypertext Systems. In Proceedings of the Fifth ACM Conference on Hypertext (Hypertext '93), (Seattle, WA, November), pp. 14-24. ACM Press.

[Wiil and Østerbye 1994] Wiil, U. K., and Østerbye, K. (eds.). Proceedings of the ECHT '94 workshop on open hypermedia systems. Technical report R-94-2038, Ålborg University, October 1994.

[Wiil and Demeyer 1996] Wiil, U. K., and Demeyer, S. (eds.). Proceedings of the 2nd workshop on open hypermedia systems, Hypertext '96. Technical report UCI-ICS 96-10, University of California, Irvine, April 1996.

[Wiil 1997] Wiil, U. K. (ed.). Proceedings of the 3rd workshop on open hypermedia systems, Hypertext '97. Technical report CIT-SR-97-01, Danish National Center for IT Research, August 1997.

[Wiil 1998] Wiil, U. K. (ed.). Proceedings of the 4th workshop on open hypermedia systems, Hypertext '98. Technical report CS-98-01, Dept. of Computer Science, Aalborg University Esbjerg, August 1998.

[Wiil 1999] Wiil, U. K. (ed.). Proceedings of the 5th workshop on open hypermedia systems, Hypertext '99. Technical report CS-99-01, Dept. of Computer Science, Aalborg University Esbjerg, May 1999.