C2ware: A Middleware Supporting Collaborative Workspaces over CMIS Repository

Hong-Chang Lee[1], Jong-Eun Park[1] and Myung-Joon Lee[2,*]

[1,2] *School of Electrical Engineering, University of Ulsan, Ulsan, 689-749, Korea*
{myhyunii, cjswowhddms}@mail.ulsan.ac.kr[1], mjlee@ulsan.ac.kr[2]

Abstract

CMIS is a standard to support interoperability among content management systems, proposing a data model to represent process and manage resources. However, since CMIS does not provide the operations for group works and the related access control functionality, there are essential difficulties in supporting various group works other than simple sharing or management of resources. In this paper, we present a middleware named C2ware that provides collaborative services in a form of web service, supporting collaborative workspaces over CMIS repositories. For this, we provide a formal specification of general services required for group works. Based on the specification, the middleware creates workspaces over CMIS repository to support various types of asynchronous resource sharing, and provides collaborative services on the workspaces. In addition, to evaluate the effectiveness of C2ware, we develop a smartphone application which works together with the middleware, conducting an experiment with the application in various group works.

Keywords: Group work, Collaborative service, Workspace, CMIS, C2ware

1. Introduction

As collaborative systems such as BSCW[1], Alfresco [2] and CoPortlet [3] appeared to support group works efficiently, users and groups who perform group works with such technologies have been increased rapidly. In general, such collaborative systems provide asynchronous resource sharing or communication facility for group works. Such systems, however, lack the interoperability among them due to their own protocols and data models, which makes them difficult to respond promptly to new platforms such as Apple iOS or Google Android.

CMIS(Content Management Interoperability Service) [4] is a standard to support interoperability and compatibility among CMSs(Content Management Systems) [5], suggesting a standard data model for representing, processing and managing resources. CMIS abstracts the indigenous resource management functions and services operated by each CMS into a common model. In addition, it supports interoperability, flexible maintenance environment and scalability. As of now, Apache Chemistry [6] that supports CMIS standard specifications is widely used, and a number of CMSs such as Alfresco and IBM FileNet [7] are broadening their boundary as they become to support CMIS. In spite of many advantages, CMIS also has limitations in controlling access to users and groups, and provide only basic operations for resource handling, not defining high-level operations for group works.

---

* Corresponding author

In this paper, we present a middleware named C2ware that provides collaborative services in a form of web service, supporting collaborative workspaces over CMIS repository. Since any repository that supports CMIS can be used as a back-end storage of the middleware, C2ware can support not only group works but also effective content management. For this, we analyze various collaborative situations and define the general services required for group works, providing a formal specification of those services. Based on the formal specification, C2ware provides workspaces using CMIS repository and collaborative services over web service. Also, it provides the operations in a form of web service based on JAX-WS to make those services to be available on clients in various platforms. In addition, to evaluate the effectiveness of C2ware, we conduct an experiment in various group works with a smartphone application which is developed to work together with the middleware.

The rest of this paper is organized as follows. In the next section, we propose a formal specification of collaborative services. In section 3, we present C2ware which supports collaborative services. Finally, we give the result of an experiment to show the effectiveness of C2ware in section 4 and conclude our paper in section 5.

2. Formal Specification of Collaborative Services

To define general collaborative services required for group works, we present a formal specification of collaborative services. The collaborative services based on this specification can help efficient system development and interoperable operations in various environments through common service interfaces.

2.1 Formal Specification

Collaborative services are sets of operations to process group. Table 1 shows the functional description of collaborative services.

Table 1. The Description of the Collaborative Services

Service Group	service
UserService (basic operations for users in relation to group works)	login, logout, createUser, deleteUser, getUserInfo, getUserList, …
GroupService (operations to create group and constitute membership)	createGroup, getGroupInfo, addMember getGroupList, requestToJoin, , …
WorkspaceService (workspaces and operations for sharing resources)	createWorkspace, createFolder, copyRes , uploadRes, getResList, …

The formal specification defines various elements necessary for realizing and providing the collaboration services, and it defines explicitly the requirements to execute the services and their processed results. Collaborative services are specified formally through the definition of the domain model which consists of the Entity model and the Service model. The Entity model represents the basic individual entities necessary for group works. Table 2 shows descriptions on the Entity model.

Table 2. Representative Properties of each Entity on the Entity Model

Entity	Property
User Entity (Properties for each user participating in the group)	uid, uname, password, createDate, …
Group Entity (Properties for group where user is participating in)	gid, gname, owner, createDate, modifyDate, workspaceId, accessible, …
Resource Entity (Properties for resources)	rid, rname, owner, createDate, modifyDate, type, parentId, …
Workspace Entity (Properties for virtual space for sharing resources)	wid, wname, owner, member, createDate, modifyDate, lock, ownerAccessId, …

The Service model formally specifies the collaborative services that have been functionally defined and classified. The specification of the model determines the direction for the implementation of each operation and defines explicitly their interface to be called. Table 3 shows some formal specifications defined on the Service model.

Table 3. The Formal Specification of some Operations in the Service Model

Operation	Requirement	Result
createGroup (user,uSet,group)	user $\in$ Users, uSet $\in$ Users group $\in$ Univ_Group - Group	group $\in$ Groups group.owner $\rightarrow$ user uSet $\in$ group.member
addGroupMember (user1,user2,group)	group $\in$ Groups user1, user2 $\in$ Users user1=group.owner $\wedge$ user2$\notin$group.member	user2 $\in$ group.member
createFolder (user,ws,res1,res2)	ws $\in$ Workspaces user $\in$ Users res1,res2 $\in$ Resources res1.type='folder' $\wedge$ res1.workspaceId=ws.id res2.workspaceId $\neq$ ws.id	res2.workspaceId $\rightarrow$ ws.id res2.parentId $\rightarrow$ res1.id res2.type $\rightarrow$ 'folder'
more services …		

2.2 Access Control Technique for Supporting Group Works

A privilege is a unit to designate access permissions for the executable operations in a specific workspace. There are several privileges to supporting access control such as Read, Write, Bind and etc. Each operation of collaborative services is mapped with proper privilege according to its content. The users who are executing certain operation can have the relevant privilege through an acl. The acl is a set of privileges and defines all usable privileges in a specific workspace. The middleware checks whether a user has the access permission for the operation in a specific workspace using ACL information. Table 4 shows the some defined access control lists.

Table 4. Access Control List

ACL / Property	Owner	Member	ReadOnly	ReadWrite	WriteOnly	Guest
read-content	o	o	o	o	x	x
read-property	o	o	o	o	x	x
read-acl	o	o	o	o	o	x
read-list	o	o	o	o	o	o
write-content	o	o	x	o	o	x
write-property	o	o	x	o	o	x
write-acl	o	x	x	x	x	x
bind-folder	o	x	x	x	x	x
bind-resource	o	o	x	o	o	x

3. Development of C2ware Supporting Collaborative Services

On the basis of the specification of collaborative services, we present a middleware named C2ware which is composed of a CMIS repository and a database storing metadata to realize collaborative services. The CMIS repository provides virtual spaces for storing and managing actual resources on the basis of a CMIS standard. Also, the database stores and manages the metadata and ACL information that are necessary when executing a collaborative service.
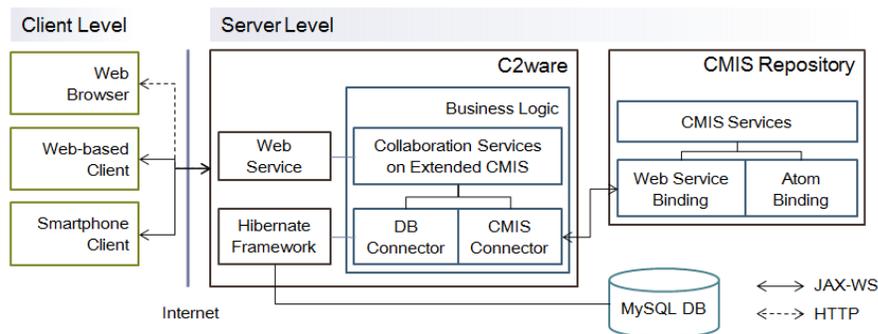

Figure 1. The Overall Structure of C2ware which Composed of a CMIS Repository and a Database

3.1 CMIS Repository and Database for C2ware

A CMIS repository provides virtual spaces where user and group can share resources asynchronously and performs a role as actual back-end storage of the middleware. Since C2ware and the CMIS repository communicate using standard CMIS operations, any CMIS-based repository can be used as a repository for the middleware. Unfortunately, since the CMIS repository does not present proper operations for group works, C2ware needs a database to manage the metadata necessary for realizing the defined collaborative services. For this, we build up a database that manages only metadata for group works. The information stored in the database represents the instances of the relevant entity model necessary for defining group works.

3.2 C2ware Supporting Collaborative Services

C2ware provides the collaborative services to clients in a form of web service. The middleware mainly processes the requests of clients and returns the results of the processing. To perform such roles effectively, it largely consists of four components to run over the

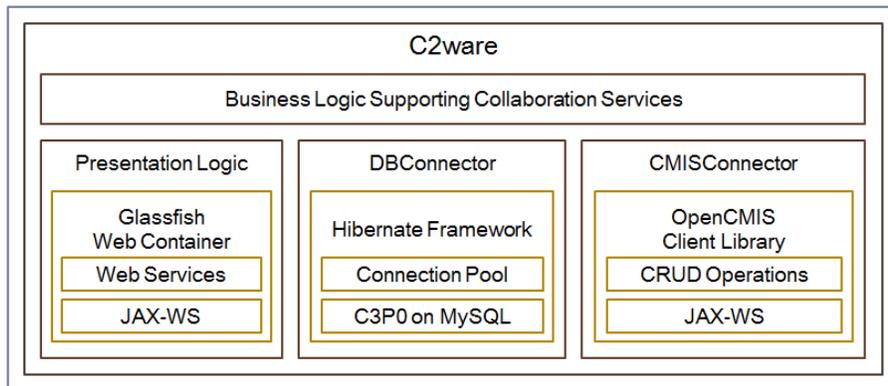CMIS repository and the database. They are Presentation logic, Business logic, CMISConnector and DBConnector.

Figure 2. The Program Stack of the Developed C2ware

The Presentation logic performs a role of an interface that provides the defined collaborative services to clients. It provides the service in a form of web service based on JAX-WS and SOAP messages to communicate with clients in various platforms and maintain high compatibility. The Business logic mainly implements all functions to support group works and provides the interfaces of them. The functions are implemented based on the defined collaborative services. The Business logic also handles all resources necessary for performing group works.

The DBConnector module provides the operations to process the metadata, maintaining the connections with the database using Hibernate framework. The CMISConnector module provides operations to support management of resources using a CMIS repository, maintaining the connection with the repository using the OpenCMIS[8] client library.

4. Experiment

To evaluate the usefulness of C2ware, we present a prototype of smartphone application that works in Android platform. This application provides the functions to perform group works based on resource sharing in a smartphone, presenting several workspaces to support various forms of group works. The Smartphone application has features for managing users, groups, resources and workspaces. Also, it has user membership function and resource control functions like upload, download, copy, cut, paste and etc.

An experiment to show the effectiveness of developed C2ware was conducted by 40 students for 4 weeks. During the 4 weeks of the experiment, each student was requested to check the task assigned to groups that he belonged to and to submit the results through the smartphone application every week

After completing the experiment, we did a survey targeting those students on the system and application. 60% of the users who used this application replied that it was helpful to group works. As a useful function, 37% of them pointed out easy group management and 35% pointed out easy resource sharing in a group. In the level of satisfaction for the application, 52% of the users answered as satisfied. We think it is remarkable that more than half of the users answered they were satisfied and it was helpful to their group works even though the application has no fancy user interfaces.

5. Conclusions

We described on the development of a middleware named C2ware that provides workspace-based collaborative services over the resource management functions of CMIS. At first, we defined a formal specification on collaborative services and access control techniques for group works. Based on the defined collaborative services and the access control techniques, we presented C2ware for providing collaborative services over CMIS. The developed C2ware provides workspaces for asynchronous resource sharing and supports effective group works in various environments through web service. To show the usefulness of C2ware, we developed a prototype of a smartphone application running on Android platform. Working together with the middleware, the application supports group works based on resource sharing in mobile environment.

In addition, we showed the effectiveness of C2ware through a practical experiment. Although the application was developed as a prototype only to evaluate, more than half of the users answered that they were satisfied. We believe the result is quite encouraging and the developed middleware could be useful for developing many practical collaborative applications.

References

[1] R. Bentley, T. Horstmann, K. Sikkel and J. Trevor, "Supporting collaborative information sharing with the World Wide Web: The BSCW Shared Workspace system", in Proc. of 4th Int. World Wide Web Conference, Boston, Massachusetts, **(1995)**, pp. 63-74.

[2] D. Caruana, J. Newton, M. Farman, M. Uzquiano and K. Roast, "Professional Alfresco: Practical Solutions for Enterprise Content Management", Wrox Press, UK **(2010)**.

[3] H.-C. Lee, T.-H. Lee, Y.-S. Park and M.-J. Lee, "CoPortlet: A WebDAV-based Collaborative Portlet supporting Individualized Services", in 3rd International Forum on Strategic Technologies(IFOST), **(2008)**, pp. 289-292.

[4] D. Choy, A. Brown, R. Mcveigh and F. Muller, "Content Management Interoperability Services TC", OASIS Standard **(2009)**.

[5] A. Mauthe and P. Thomas, "Professional Content Management Systems: Handling Digital Media Assets", John Wiley & Sons **(2004)**.

[6] Chemistry Project, The Apache Software Foundation, http://chemistry.apache.org.

[7] T. R. Groff and T. P. Jones, "FileNet: A Consultant's Guide to Enterprise Content Management", 1st American pbk. ed., Elsevier Butterworth-Heinemann, **(2004)**.

[8] OpenCMIS, Apache Software Foundation, http://chemistry.apache.org/java/opencmis.html.