Specification of the interactions among
the Scholnet services (D2.2.2)

Deliverable Type: I
Number: D2.2.2;
Nature: Report
Contractual Date of Delivery:
Actual Date of Delivery: 07 January 2002
Task: WP2

Name of responsible: Donatella Castelli
Pasquale Pagano

CNR-IEI
Area di Ricerca di Pisa
56124 Pisa
Italy

castelli@iei.pi.cnr.it
pagano@iei.pi.cnr.it

Abstract: This Internal Report contains a list of sequence diagrams that specify in detail the interactions among the Scholnet services. It has been edited by CNR assembling the contributions delivered by each service leader.

Keyword List:
Interaction specification, sequence diagrams, digital library services.
Table of Content

1. INTRODUCTION .................................................................................................................. 3
2. LIBRARY MANAGEMENT .................................................................................................... 4
3. QUERY MEDIATOR ............................................................................................................ 13
4. INDEX ................................................................................................................................ 16
5. BROWSE ............................................................................................................................. 19
6. COLLECTION ...................................................................................................................... 22
7. ANNOTATION ..................................................................................................................... 28
8. PERSONALIZED INFORMATION DISSEMINATION ............................................................ 39
9. USER INTERFACE ............................................................................................................... 40
REFERENCES ........................................................................................................................ 52
1. Introduction

This report collects a set of UML–like sequence diagrams [UML] that specify the interaction among the Scholnet services. The focus is on message sequences, that is how messages are sent and received among a number of services. These diagrams were produced in order to check the completeness of the Scholnet protocol as much as possible before starting the implementation phase. As an effect of this specification, the interactions among the services were clarified, missing and wrong service requests were identified. All the corrective actions taken have been integrated in a revised version of the Architectural Specification Report [D2.2.1].

A basic sequence diagram, for each verb that sends service requests to the other services during its processing, has been produced. To simplify the graphical presentation, a basic sequence diagram reports only one step of the interaction chain required to process a verb, i.e. it indicates only the service requests invoked by the service implementing the verb and it does not show those invoked by the other services in order to serve the requests received. The complete chain of interactions for processing a verb can be obtained by composing the given basic sequence diagrams.

In the case of services, like for example, the Collection or the Library Management, that can access a functionality both through a verb of the GUI or through the direct invocation of an engine verb only the interaction diagram of the second has been specified.

The presentation of the basic sequence diagrams is organized in sections, each section reports the sequence diagrams related to the verbs of a specific service. Note that no basic interaction diagram has been produced for services, like Registry or Repository, that do not interact with any other service, in order to carry out their processing. A set of sequence diagrams that describe the interactions required at the system start-up time has also been included in order to specify how each service gets the needed configuration parameters.

The completeness of the User Interface verbs has been verified by checking them against the use scenarios envisaged in the Functional Specification Reports [D2.1.1].

The sequence diagrams have been produced by using the Microsoft Visio Professional 2002 tool [VISIO]. These diagrams have two axes: the vertical axis shows time and the horizontal axis shows a set of services involved in the sequence. Each service is represented by a rectangle with a name inside. A vertical dashed line, called the service’s lifeline, indicates the messages sent and received and the activation of the service. Communication among the services is represented as horizontal message lines between the services’ lifelines. A continuous line indicates a service request, a dotted line a service response. Each service request is specified by giving its name, the list of its parameters and, in a different line, the returned result. The fixed parameters are located up to the continuous line, the keywords parameter are located under this line. The full arrowhead shows whether the message is synchronous, and the half arrowhead shows whether the message is asynchronous. The sequence diagram has to be read starting at the top of the diagram and reading down in order to view the exchange of messages taking place as time passes.
2. Library Management

Service Start-Up

Library management

Meta Service

ListRepositories

auths=1,verbs=1

XML list

Analyze and store the data

ListAuthorities

XML list

Analyze and store the data
Reject

Library Management

Registry

MDS

Reject/
type/user-id

reason, title, date, login, password

Not authorised

[The user is authorised to reject documents for the authority specified]

AuthenticateUser/
login/password

UserRights

Check authorisation

Analyze the document structure file

Xml response

[If not yet authenticated]

[then proceed with login/password]

StartSession/
login/password

DiscardUpload/
sessionId/userId

Xml response

Send an e-mail to author

[Remove the incoming request manifestations if type=delete]

Xml response

[if doc-id document have manifestations of external type and type=delete]

Xml response

Xml response

[if not yet authenticated]

Xml response

Xml response

Xml response

EndSession/
sessionId

Xml response
UserWithdraw

UserWithdraw/ 
/type/handle
new-addr, reason

[Not authorised]
[The user is authorised to require the withdrawal of the document]

[The user is not the owner]
[The user is the owner of the document]

Xml response

AuthenticateUser/ 
/login/password
libmgt

UserRights

AuthenticateUserHandles/ 
/login/handle

Text response

Store the incoming request
A Digital Library Testbed to Support Networked Scholarly Communities

UserSubmit

UserSubmit/ %/multipart-mixed MIME file

Library Management

[if not yet authenticated]
AuthenticateUser/ login/password

Registry

UserRights

Not authorised

[The user is authorised to submit a document]

StartSession/ login/password

MDS

Xml response

RequestUpload/ sessionId/namelist/ sizelist

Xml response

Xml response

Xml response

Store as external manifestation uri in the document structure file, the doc-id received for each file

Store the incoming request

EndSession/ sessionId

Xml response

Xml response

Xml response
A Digital Library Testbed to Support Networked Scholarly Communities

UserSubmitNewVersion

UserSubmitNewVersion/ %multipart-mixed MIME file

Library Management

[if not yet authenticated]
AuthenticateUser/ login/password

Registry

libmg

UserRights

Check authorisation

AuthenticateUserHandles/ login/handle

MDS

Not authorised

[The user is authorised to upload a new version of his/her document]

The user is not the owner

[The user is the owner of the document]

Xml response

UserSubmitNewVersion

Store the incoming request

AuthenticateUser

StartSession/ login/password

Text response

The user is not the owner

[The user is the owner of the document]

Xml response

StartSession/ sessionId/namelist/sizelist

Xml response

Xml response

Xml response

Xml response

Xml response

Xml response

Xml response

Xml response

Xml response

Xml response

Xml response

RequestUpload/ sessionId/namelist/sizelist

EndSession/ sessionId

Store as manifestation attribute of the document structure file, the doc-id received for each file

Store the incoming request
UserSubmitForm

Library Management

Registry

[if not yet authenticated]
AuthenticateUser/login/password

UserRights

Check authorisation

Analyze the document structure file

Call the UserSubmit LibMgt verb

Not authorised

[The user is authorised to submit documents for the authority specified]

Show the result of the request. If the request contains some composite or video manifestation, the user-id is returned to the user. The user has to use this user-id in order to upload the document to the MDS ftp server.
UserSubmitForm

Library Management

[if not yet authenticated]
AuthenticateUser/login/password

UserRights

Check authorisation

ListUserHandles/login

Xml response

Call the UserWithdraw LibMgt verb

UserWithdrawForm

Registry

Not authorised

[The user is authorised to submit documents for the authority specified]

Show the result of the request
3. Query Mediator

Service Start-Up

- **QM**: RegionDescription region
  - XML region description
  - Analyze and store the data
- **Meta Service**: ListAuthorities
  - XML response
  - Analyze and store the data
  - ServiceConfigDescription
  - XML response
  - ListAuthority
  - XML response
  - ListIndex
  - XML response
  - ListIndexFields
  - [For each index server]
- **Index**: ListCollections
  - XML response
  - Analyze and store the data
- **Collection**: GetCollectionMetadata CollId
  - XML response
  - [For each collection]
  - Analyze and store the data
SearchFielded

SearchFielded/ search-specification

result-set

[Incorrect parameter/s]

XML error response

QM

Verify parameters

Transform the query on a collection on a query to authorities

[result-set not null]

Using the rs-rel+ relation identifies the result-set for each authority to be used

[Identifies the Index servers for the authorities and sends the request]

SearchFielded/ search-specification

result-set

Index

XML result set

Merge the results

XML result set

Arrange the result list taking into account the specified result-format
A Digital Library Testbed to Support Networked Scholarly Communities

**SearchFeedback**

- **SearchFeedback/result-set/search-specification**
- **QM**
- **Index**

- [Incorrect parameter(s)]
- XML error response

- **Verify parameters**
  - Extract from each relevant document terms that can be added to the original query linked to the specified result-set
  - Transform the relevance feedback query in a fielded query

- **SearchFielded/result-set/search-specification**
  - [Identifies the Index servers for the authorities and sends the request]

- XML result set
  - **Merge the results**
  - **Arrange the result list taking into account the specified result-format**

- XML result set
4. Index

Service Start-Up

Index

Meta Service

ServiceConfigDescription/indexer

XML service configuration

Analyze and store the data

ListAuthorities

XML response

Analyze and store the data

ListRepositories

XML response

Analyze and store the data
SearchFielded

Index

SearchFielded/ search-specification

result-set

[Incorrect parameter/s]

XML error response

Verify parameters

Perform the search on the specified field

Filter the result taking into account the specified result-set

Arrange the result list taking into account the specified result-format

XML result set
IndexUpdate

IndexUpdate's metadata-format

[Incorrect parameter(s)]

XML error response

Verify parameters

Lock the Index Server database

Identify the repositories that have to be updated

ListContents

XML response

Update the Index Server database

Unlock the Index Server database

Text result

Index

Repository
5. Browse
A Digital Library Testbed to Support Networked Scholarly Communities

Browse

- Browse/field
  - Range, authority/collection, format
  - [Incorrect parameter/s]
  - XML error response

- XML result set

- Verify parameters
  - Perform the browse on the specified field
  - Filter the result taking into account the specified range
  - Filter the result taking into account the specified authority/collection
  - Arrange the result list taking into account the specified format
BrowseUpdate

BrowseUpdate/metadata-format

BrowseUpdate/metadata-format

[incorrect parameter/s]

XML error response

Verify parameters

Lock the browse database

Identify the repositories that have to be updated

ListContents

authority, file-after, meta-format

XML response

Update the browse database

Unlock the browse database

Text result
6. Collection

Service Start-Up

- ListRepositories
  - auths=1, verbs=1
  - XML list
  - Analyze and store the data
  - ListAuthorities

- XML list
  - Analyze and store the data
  - ListIndices

- XML list
  - Analyze and store the data
  - RegionDescription

- XML list
  - Analyze and store the data
CreateDefaultCollection

Collection

Repository

QM

CreateDefaultCollection

Analyze the repository list

ListSets

[For each Repository server]

Xml sets list

Compose a unique sets list

ListSearchMethods

Xml response

Create for each set a collection and register its collection metadata

Response
CreateCollection

CreateCollection/collection-specification

[if not yet authenticated]
AuthenticateUser/login/password
collection

UserRights

Check authorisation

Create the collection membership condition

Create and register the collection metadata

Not authorised

[The user is authorised to administer the CS]

Xml response
DeleteCollection

DeleteCollection/CollectionId

login, password

[The user is the collection owner]

Not authorised

Xml response

[if not yet authenticated]
AuthenticateUser/login/password

Update the father-child relation for each collection related with the deleted collection

Delete the collection metadata

UserRights

Check authorisation
Synchronise

Collection Master

Collection Slave

Synchronise/
host/port

Lock any collection
service request
SynchroniseCollection

Update the set of
collection metadata
& unlock the list
collection request
SynchroniseCollection

Unlock the service

Xml response
7. Annotation

Annotation Service Start-Up

Annotation Service

MetaService

RegionDescription

XML response

analyze and store data

ListRepositories

XML response

analyze and store data
Begin Session

Annotation Service

BeginSession (login)

ListUserGroups (login)

group list

create structures about user groups

ListUserProjects (login)

project list

create structures about user projects

assign sessionid

Success (return sessionid)

Failure
End Session

Annotation Service

EndSession (sessionid)

Response (success/failure, sessionid)

End the session with sessionid
Annotation Service

FetchAnnotation (sessionid, annotation_handle)

check user authority

Failure (user not authorized)

fetch annotation

Response (return xml document)
Create Annotation

CreateAnnotation (sessionid, document_handles, xml document)

Annotation Service

check user authority

Failure (user not authorized)

annotation (document_handle, set)

success

create annotation

(OR)

failure

Repository

Response (return xml document)
Update Annotation

Annotation Service

UpdateAnnotation (sessionid, annotation_handle, xml document)

check user authority

Failure (user not authorized)

update annotation

Response (return xml document)
Search Annotation

Annotation Service

SearchAnnotation (sessionid, search_specification)

check user authority

Failure (user not authorized)

search annotation

Response (return xml document)
GetRefDocuments

Annotation Service

GetRefDocuments (sessionid, annotation_handle)

check user authority

get referenced documents

Response (return xml document)

Failure (user not authorized)
GetAnnotationsOfDocument

Annotation Service

GetAnnotationsOfDocument (sessionId, document_handle)

check user authority

Failure (user not authorized)

get annotations of document

Response (return xml document)
DeleteAnnotation

DeleteAnnotation (sessionid, annotation_handle)

check user authority

find document handles linked to the annotation to be deleted

Failure (user not authorized)

annotation (handle, remove)

success

delete annotation

failure

Response (return xml document)
Display Annotations

Annotation Service

DisplayAnnotations (sessionid)

check user authority

Failure (user not authorized)

display annotation

Response (return xml document)
8. Personalized Information Dissemination

Get Metadata Records (PULL)
9. User Interface
**login**

UserInterface

AuthenticateUser/login/password

User rights or error code 400

[if not yet authorized]

Not authorized message

[if authorized]

Authorized message

**submitRegistrationForm**

UserInterface

AddUser/login/password/email

User Settings or Error Code

invoked from registration form

submitRegistrationForm/login/password/email

[if not successful]

Not registered message

[if successful]

Registered message
**generateTopicForm**

```
UserInterface

<table>
<thead>
<tr>
<th>getTopics/userID</th>
</tr>
</thead>
<tbody>
<tr>
<td>list of topics</td>
</tr>
</tbody>
</table>

[topics of user]

<table>
<thead>
<tr>
<th>submitTopicForm/ &lt;post-input-fields&gt;</th>
</tr>
</thead>
</table>
```

**submitTopicForm**

```
UserInterface

| submitTopicForm/ <post-input-fields> |

[topic form successfully created]

<table>
<thead>
<tr>
<th>createTopic/user-id/ topic-id/ topic-label/ notification-flag</th>
</tr>
</thead>
</table>
```

<table>
<thead>
<tr>
<th>return codes</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>some message</th>
</tr>
</thead>
</table>
**generateGroupMaintenanceForm**

- UserInterface
- Registry

generateGroupMaintenanceForm

- listUserGroups/login
- XML response

[if user wants detailed info about a group and edit the group]

displayGroupInfo/groupName

**displayGroupInfo**

- UserInterface
- Registry

displayGroupInfo/groupname

- XML response

[group info or message]

[activated from group maintenance]

displayGroupInfo/groupname

[user edits group info (if owner), group members etc. and submits]

submitGroupChanges
submitGroupChanges

UserInterface

submitGroupChanges/<list-of-post-input-fields>

Registry

[if group has been marked for deletion]
deleteGroup/groupname/owner/password
text/plain response

[for each user which has been added to user list of group]
addUserToGroup/groupname/owner/password/login
text/plain response

[for each user which has been removed from user list of group]
deleteUserFromGroup/groupname/owner/password/login
text/plain response

modifyGroupInfo/groupname/owner/password/group-settings
text/plain response

some status message
submitSearchForm

Note: displaySearchResult has to provide a button that submitSearchResult can be invoked over and over again for cross-language search.
submitThesaurusSearch

submitThesaurusSearch/sessionid/searchstring

UserInterface

Thesaurus

searchTerm/sessionid/string

XML response

display search results (in a new window)

displayTermInfo

displayTermInfo/sessionid/term

UserInterface

Thesaurus

listTerm/sessionid/targetterm

XML response

display term info (in a new window)

[user wants to edit term and is allowed to]
genenerateTermEditForm?xml-data-of-term
displayBrowseResults

If no keyword args are given, it is assumed that the browsing process is starting. The browseable fields etc. are displayed.
A Digital Library Testbed to Support Networked Scholarly Communities
References