Distributed System Design

COMP 6231: Winter 2019

Instructor: R. Jayakumar

Distributed Library Management System (DLMS) using Web Service

Submission Date: 12/04/2019

By: Md. Hasibul Huq

ID – 40087646

Overview: The distributed library management system is aims to connect a group of libraries. This system is used by two type of users. Those who manage library are managers and those who uses the library facilities like borrowing books are user. In this system managers can perform few actions like:

1. Add Item
2. Remove Item / Decrease the number of items
3. List of the items

Additional feature in this system for managers is the list of people and their required item which is currently is not available.

On the other hand, Users have some functionality like:

1. Borrow Item
2. Return Item
3. Find Item
4. Exchange Item

Additional features for users are they can see their borrowed item list.

In this project 3 servers are used, and the name of the servers are

1. Concordia Server
2. McGill Server
3. Montreal Server

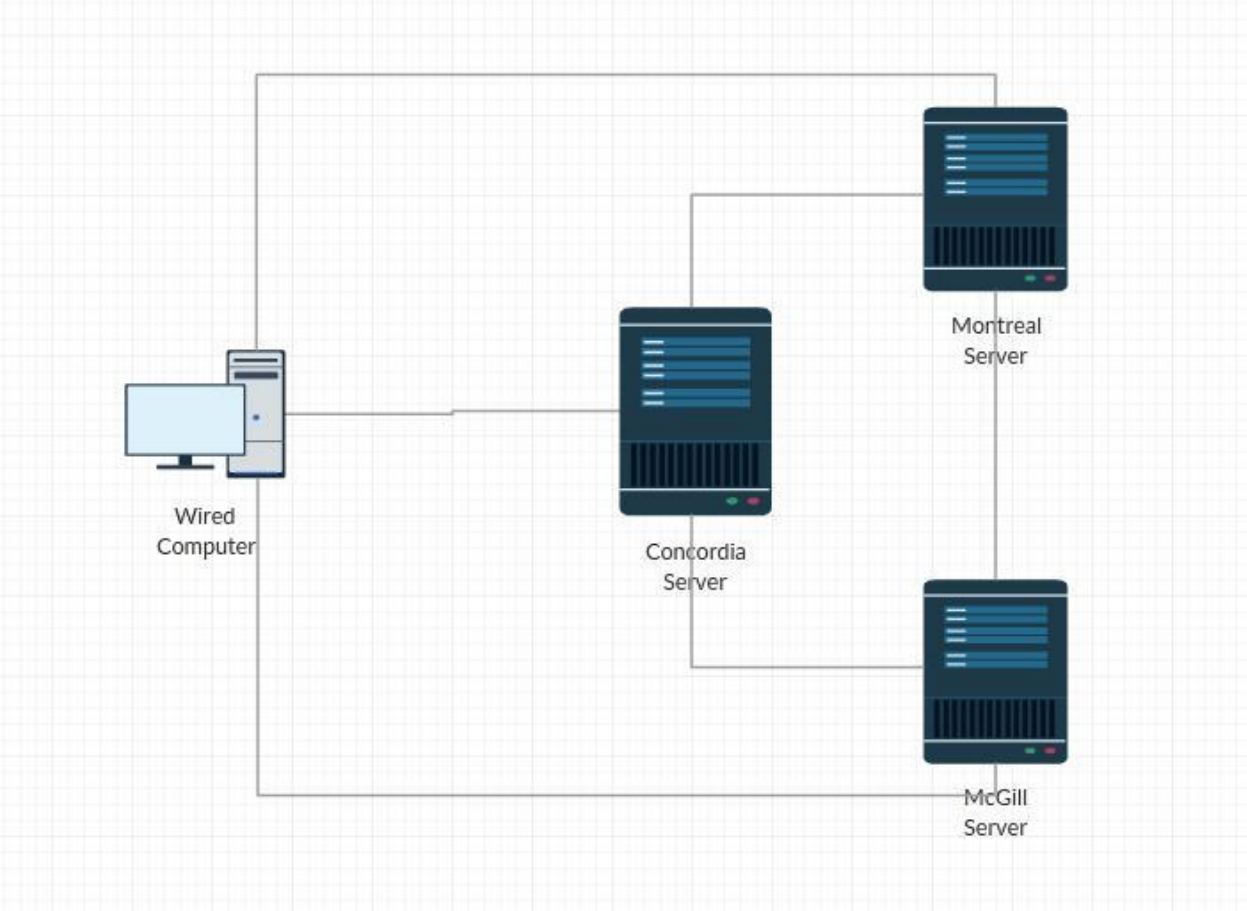
The users of the system are library managers and library users identified by a unique managerID and userID respectively, which is constructed from the acronym of their library and a 4-digit number. Whenever the user performs an operation, the system identifies the server that user belongs to by looking at the ID prefix and perform the operation on that server. The user also maintains a log (text file) of the actions they performed on the system and the response from the system when available. There are different managers for the three libraries/servers. They create availability of items in their library along with the quantity of the items. A user can borrow an item offered by any library, if it is still available.

This System is designed to explain the entire functionality of the WEB Service. Services that can be accessed over network are called web services. Web services are meant for applications to access data in the format of XML. A single web service can be used by different kinds of applications.
Java provides its own API to create both SOAP as well as REST web services.

1. JAX-WS: JAX-WS stands for Java API for XML Web Services. JAX-WS is XML based Java API to build web services server and client application.

2. JAX-RS: Java API for RESTful Web Services (JAX-RS) is the Java API for creating REST web services. JAX-RS uses annotations to simplify the development and deployment of web services. Both of these APIs are part of standard JDK installation, so we don't need to add any jars to work with them. Both of these APIs use annotations very heavily.

In this assignment, web service and UDP are the main techniques. This application contains 3 servers and 1 client, client where in users and managers are being handled simultaneously.

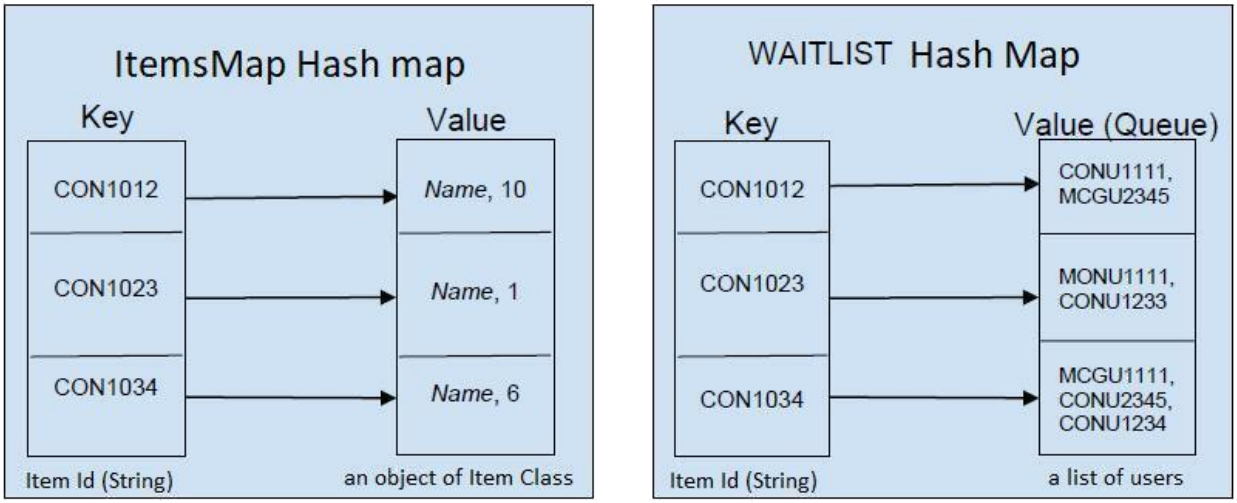


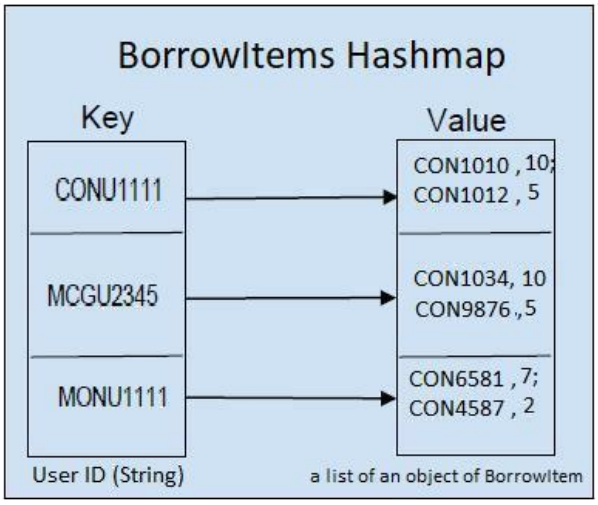
Pic: System Architecture

Client relates to 3 servers through WSDL and 3 servers are communicating with each other by UDP connection. Client can decide its server from the user name. And user and manager operation all are handled from the specific server based on user id or manager id. All the operation in the server side are logged into a log file for each server. And the client requests and response also create a log file for each user.

To store data three different hash maps are used in each server. One hash map is containing item name and quantity with a unique item key which is named itemId here. To store borrowed item

list, borrowedItems hash map is used where user id is unique and the list of items with number of days are stored. If the item is not available during borrowing user can wait. To keep this record here waitingList hash map is used.





Pic: Hashmaps

To facilitate the user with the borrow and find item from different server UDP connection is necessary. UDP send and receive method was implemented in each server with multi threading facility. So that concurrent user can borrow and find their item.

Instead of CORBA IDL here web Service works with the WSDL interface and the WSDL interface is executed by WSGEN command.

Here most difficult part is replacing CORBA information's and replace it with the web service codes. For that, Refactoring needed and need to change the binding methodology of CORBA with WEB Service.

Test Cases:

1. Login as CONM1111 and add Item Item Id CON9876 Name: DS and quantity 2 (Item Added)
2. Login as MONM1111 and add item Id MON9876 Name DS and quantity 1 (Item added)
3. Login as MONM1111 and add item Id MON9877 Name APP and quantity 1 (Item added)
4. Login as CONU1234 and Borrow item CON9876. (item Borrowed) and exchange CON9876 with MON9876. (Item Exchanged).
5. Login as MCGU1234 and Try to Borrow MON9876 (asked for waiting list. Yes. Added in waiting List)
6. Login as CONU1234 return item MON9876. (Item returned and allocated to MCGU1234)
7. Login as MCGU1234 and try to exchange MON9876 with MON9877 (Exchanged)
8. Login as CONM1234 and show the list item. It will return 2 correct result and one wrong result. Try 3 times. In the 4th time all the servers will give the correct result
9. Login as CONM0000 and show the list item. It will crash the Concordia server in RM one. and the server will be restarted.

Test Scenario:

Sce. Id	Requirement Name	Test Scenario	Test Cases
1	Login	User name	1. Validate with valid user name 2. Validate with invalid username 3. Validate access with username 4. Validate server access with Username
2	Manager Facility	Menu Selection	1. Validate menu selection
3	Manager Facility	Add Item	1. Validate Item name 2. Validate Item id with valid id 3. Validate Item id with an invalid id 4. Validate item Quantity 5. Either it will add a new item or will increase the existing id
4	Manager Facility	Remove Item	1. Validate Item Id with valid id 2. Validate item id with invalid id 3. Validate Item Quantity 4. It will decrease item quantity or remove it completely
5	Manager Facility	List Item	1. It will show the items available in the server
6	Manager Felicity	Logout	1. It will logout the user and will ask for username to login
7	User Facility	Borrow Item	1. Validate Item Id with valid item id 2. Validate Item Id with invalid item id 3. If valid the item will be borrowed 4. Validate with an Item id which is not available 5. Validate with item id which is available in different server 6. Validate one user can not borrow multiple item from Different server
8	User Facility	Return Item	1. Validate item id with valid item id. 2. Validate with item id which is available in different server
9	User Facility	Find Item	1. Validate with Item name which is available in one of the servers or in any of them
10	User Facility	Exchange Item	Enter a new Item Id and old item id 1. to exchange the book.
11	User Facility	Logout	1. It will logout the user and will ask for username to login