

18-749 – Team 5

Fault-Tolerance Evaluation Design Proposal

Chief Experimenter

Kevin Smith

Implementation Changes Required

- The invocations will need to be modified so as to return different sized replies
- Clients will need to be modified to wait for an inter-request time period before sending the next request
- Client will need to be modified to allow command line options for running in evaluation mode and specifying size of reply messages, number of clients, and inter-request time period
- Clients will send their hostname to the server when making requests so that the server can record this information in its probe files
- Clients and servers will have to be able to record the time when requests are sent, received, etc.
- Clients and servers will have to keep track of all data required in the data files while running
- When test completes, clients and servers flush all the data they have been recording to the necessary data files
- Scripts must be created which automate the entire process of launching clients and servers for all the 48 different configurations to be tested, specifying the parameters of evaluation, etc.

Required Scripts

- One script will be able to remotely launch servers and clients and specify different parameters of evaluation. This script will run through the 48 different configurations required.
- MATLAB scripts for data analysis

Gotchas

- On the server, arrays will be used for the reply objects of varying size
- In the client invocations, the request size will be specified along with the client hostname as parameters
- For keeping time on the client and server, Java's `System.currentTimeMillis()` will be used unless a finer-grained method can be implemented such as OS-specific C code using JNI for interfacing with Java. Another option is Java's `System.nanoTime()` but that is only available in Java 5.

Client Invocations (original)

<i>METHOD</i>	<i>ONE_WAY</i>	<i>DB_ACCESS</i>	<i>SZ_REQUEST</i>	<i>SZ_REPLY</i>
login()	No	Yes	? bytes ¥	4 bytes
logout()	No	Yes	? bytes §	4 bytes

¥ 2 bytes * (length of username + length of password)

§ 4 bytes + (2 bytes * length of username)

public int login(String username, String password)

In the baseline version, the login() method receives two strings as parameters, the username and the password (In Java, strings use unicode characters which are 2 bytes). It queries the database to see if the password is correct, and if so, updates the user table to specify that the user is logged in. The login() method returns a 4 byte acknowledgement in the form of an int. This acknowledgement tells the client whether or not the login action was successful. The acknowledgement is false if the method could not connect to the database or the password is incorrect.

public int logout(String username, int sessionNumber)

In the baseline version, the logout() method receives a username and a session number. The session number specifies the chat room in which the user was chatting. The logout() method sets the logged_in value to false for the user's entry in the users table in the database. It then looks at the messages table in the database, and if there are no longer any users in the room specified by sessionNumber, it clears the messages table of the messages from that chat room.

Client Invocations (proposed)

<i>METHOD</i>	<i>ONE_WAY</i>	<i>DB_ACCESS</i>	<i>SZ_REQUEST</i>	<i>SZ_REPLY</i>
login()	No	Yes	? bytes ¥	4, 256, 512, or 1024
logout()	No	Yes	? bytes §	4, 256, 512, or 1024

¥ 24 bytes + (2 bytes * length of clientname)

§ 18 bytes + (2 bytes * length of clientname)

public int [] login(String username, String password, String clientname, int replySize)

In the evaluation version, the login() method will receive two additional parameters, the connecting client machine name and a reply size. The login() method will perform the same functionality as the baseline version but will also add the time it started, the time it ended, the connecting client machine name, and its method name to the four corresponding probe lists. In addition to these changes, the method will return an int array where the first value is the acknowledgement and whose size is specified by the replySize parameter.

public int [] logout(String username, int sessionNumber, String clientName, int replySize)

In the evaluation version, the logout() method will see the same changes and additions as specified above in the description of the evaluation version of login().