

TDTS04 - Assignment 5

1. Explain what is the purpose of the name server that is being used by the chat system. How would a call by the client have looked like if the name server would not have existed?

The purpose of the name server is to store information about the server and the clients connected to the server. This is done by connecting names of the connections to their IP-addresses. Basically, the name server is a link between the clients and server to easier communicate without having to manually store the information (information such as destination IP-address, port numbers etc.).

Without the name server each client would have had to store all information of the other applications instead of fetching the information from the name server.

2. The server in the lab is using the callback function. This is one way of implementing asynchronous method calls in CORBA and does not affect how objects are implemented in the server. Please explain why this is the case. Are there other alternative options for how asynchronous calls can be implemented in CORBA?

The "Callback function": The client automatically receives messages from the server without considering if the client is ready or not. This is because the message is sent as a request.

An alternative option for how asynchronous calls can be implemented in CORBA is the "Polling model": The client don't receive calls from the server automatically. Instead the calls are being stored in a "Poller" value type and client check the status of the message and decide when to poll the message. In other words, it depends on how the programming code is implemented when (and if) the calls are being received.

Your report should explain and illustrate how you tested the functionality of your implementation:

The chat was with multiple terminal windows: The name server (orbd), the server and multiple clients. First, the chat was implemented. To test the chat we joined multiple clients, checked if all clients received messages where needed (for example when someone left the chat) and tested some bugs:

- Multiple clients with same name is not allowed.
- You can only use command "list" and "join" without joining. All other commands require you to join.
- Do all clients received information about when a client is joining and leaving?
- Does only registered clients receive posts?
- What happens if input of client is not a valid command?

Secondly, we implemented the Gomuko (Five in A Row) game and tested possible bugs:

- Is the game board printed out to all registered clients?
- You should not be able to add color on occupied space or outside of board.
- Is the chosen color a valid color?

In addition we also printed out to the server interesting information such as if a player has joined, left, if a player has joined a game session and the Gomuko game board. Also, while implementing the game we printed other debugging information to see what was wrong in the code.

Any limitations should also be clearly stated:

As it is now, there's a limitation if this chat would be implemented outside of localhost. Since the callback function is implemented the server just calls messages to all clients and if a client is about to input something, the message will be added to the input. This greatly confuses and annoys the client.

ALEJO720 Alexander Jonsson (alejo720@student.liu.se)

JONMA993 Jonas Malmlöf (jonma993@student.liu.se)

Sources:

"Programming Asynchronous Method Invocations with CORBA Messaging (Column 16)" 1999 written by "Douglas C. Schmidt" and "Steve Vinoski",
<http://www.cs.wustl.edu/~schmidt/PDF/C++-report-col16.pdf>

https://en.wikipedia.org/wiki/Name_server

"What is a name server?" 2016 written by "Jacob Nicholson",
<http://www.inmotionhosting.com/support/domain-names/dns-nameserver-changes/what-is-a-name-server>