Implementing an IRC Server Using an Object-Oriented Programming Model for Concurrency

Bachelor Thesis

PROEJCT PLAN

Project Period	11.04.11 – 03.07.11
Student Name	Fabian Gremper
Status	6th semester
Email address	fgremper@student.ethz.ch
Supervisor name	Scott West

1 Project Description

1.1 Overview

The Internet Relay Chat (IRC) is a form of real-time Internet text messaging. It was one of the most popular chat protocols in the early stages of the Internet. Most of the popular IRC server daemons (from now on also called IRC servers) were therefore not written with object-oriented principles in mind.
This bachelor thesis focuses on writing a new IRC server daemon from scratch. It will be programmed using an object-oriented language and will make use of an object-oriented programming model for the concurrency aspects of the application.

1.2 Scope of the work

The application will function as an IRC server and provide a working subset of the functionality documented in the IRC protocol specification in RFC1459.
The application will be implemented in a way that follows the principles of object-oriented programming and use SCOOP as a programming model for concurrency.

1.3 Intended results

The intended result is a fully-functioning IRC server application.

2 Background Material

2.1 Reading list

- EIFFEL website, http://www.eiffel.com/, 2011.
- SCOOP website, http://scoop.origo.ethz.ch/, 2011.

3 Project Management

3.1 Objectives and priorities

The expected functionality the IRC server application should provide follows:

- The server follows the RFC1459 protocol and works with common IRC clients.
- The server allows many users to connect simultaneously and interact with each other directly or using channels.
- Channels support channel modes +ismnt and allow setting topics.
- Users can gain voice and operator privileges in channels.
- Users with operator privileges can alter the privileges of users in a channel and kick users from a channel.
- Users can retrieve information about the available channels and other users on the server.
- The servers will ping its clients periodically and removes timed out clients.

The application will be written in Eiffel and make use of new concepts in concurrency modeling provided by SCOOP.

3.2 Criteria for success

The minimal objectives described above will be implemented. The IRC server will be interoperable with the mIRC client and will be able to handle a minimum of 50 users simultaneously.

3.3 Method of work

Biweekly meetings will be held to report on the progress of the project and to make decisions about next steps.

3.4 Quality management

3.4.1 Documentation

The final project report will document design choices, implementation and evaluation of the project.

3.42 Validation steps

The IRC server will be tested for interoperability with the mIRC client after new features have been implemented. Continuous feedback from the supervisor will guide the development process.

4 Plan with milestones

4.1 Project steps

The project involves the following milestones:

1) Familiarizing with Eiffel and SCOOP; studying the IRC protocol, RFC1459.
2) Implementing server-side sockets.
3) Allowing several users to be connected simultaneously and handle concurrency aspects.
4) Implementing the IRC server functionality that is expected.
5) Analyzing SCOOP concurrency.
6) Evaluation and writing the report.

4.2 Deadline

The deadline for the project is July 3, 2011.

4.3 Tentative schedule

	Week											
	1	2	3	4	5	6	7	8	9	10	11	12
Step 1	■	■										
Step 2			■									
Step 3				■								
Step 4					■	■	■	■	■	■		
Step 5			▨	▨							■	
Step 6						▨	▨	▨	▨	▨	■	■

References

[1] Chair of Software Engineering: Semester-/Diplomarbeiten; Online at: http://se.inf.ethz.ch/projects/index.html, consulted in October 2002.

[2] Bertrand Meyer: Object-Oriented Software Construction, 2nd edition, Prentice Hall, 1997.