Network Programming--
Suggested Term Project Specifications

Instructor: Prof. Ying-Dar Lin
TA: Yi-Neng Lin

2/23/2006
Term Projects

- Basically open for all possibilities
  - Apply techniques learned in the course
- Suggested topics
  - Chatroom
  - Proxy server
  - Multithreaded web server
Choice 1: Chatroom

- **Purpose**
  - Chatroom supporting multiple themes/sessions
  - Use client-server techniques
  - Can be dynamically established or terminated
Programming Background

- Four client-server models
  - Iterative connectionless
  - Iterative connection-oriented
  - Concurrent connectionless
  - Concurrent connection-oriented

Serve one request at a time

<table>
<thead>
<tr>
<th>Concurrent connectionless</th>
<th>Iterative connectionless</th>
</tr>
</thead>
<tbody>
<tr>
<td>Concurrent Connection-oriented</td>
<td>Iterative Connection-oriented</td>
</tr>
</tbody>
</table>

TCP

UDP
Concurrent Connection-Oriented Methods

(1) Single-process concurrent servers

(2) Master-slave concurrent servers

(3) Pre-allocated Master-slave

(4) Multi-threading

- Different from fork (the dashed lines in (1)~(3)) in that:
  - Share memory (variables)
  - No inter-process comm.
  - Lock/un-lock manipulations
Functionalities

- **Basic**
  - Lookup all chatrooms for their topics and participants
  - Chatroom manager decides whether to admit a new participant
  - Ex: /room, /command, /create, etc

- **Advanced**
  - 浸水桶 (ignore msg from a certain user)
  - Conversation recording

- **More functions please refer to the detailed spec.**
Choice 2: Web Proxy Server

- **Purpose**
  - Develop a multi-threaded web proxy supporting a set of functionalities

- **Background**
Functionalities

- Caching is not needed

- Support
  - GET, HEAD commands
  - If-Modified-Since conditional request

- Implement following HTTP codes
  - 304: Not Modified
  - 501: Not Implemented

- Other functions you want
Choice 3: Multithreaded Web Server

- **Purpose**
  - Develop a multithreaded web server capable of processing a set of HTTP requests

- **Background**

![Diagram of client-server interaction with ASCII messages](image)

Messages are in ASCII format
Functionalities

- **Concurrency**
  - Multithreaded

- **Based on HTTP/1.1**
  - MIME (Multipurpose Internet Mail Extensions)
  - CGI program support
    - Perl, C, etc
  - Status codes
    - Basic: 200 (OK), 403 (Forbidden), 404 (Not Found),
    - Implement as many as you can
Other Notes

- Detailed specification shall be released within two weeks
  - Testing
  - Problem discussions
  - Report format