

# Concurrent Webserver

Anitha Ganesha

February 10, 2014

## 1 Introduction

A websystem consists of a webserver, a number of clients and a network that connects the clients to the server. The protocol that is used to communicate between the client and the server is HyperTextTransfer ProtocolHTTP[2]. A webserver is a software that helps to deliver the webcontent that can be accessed through the internet[1].The most common applications of webserver includes hosting websites, data storage, gaming etc.[1]. The webserver application is a good example to study and understand concurrency. This is beacuse any webserver is required to handle the request from the various clients gracefully in real time.

## 2 Implementation

### 2.1 Optimal Goal

I will be implementing a java multithreaded webserver with a subset of functionalities of HTTP/1.1 protocol. The main functionalities contains[3]

- Implementing the GET and HEAD methods.
- For each GET and HEAD requests, appropriate responses will be returned. If the request is malformed, the program should handle it gracefully.
- Include Date, Server, Content-Type and Content-Length headers in all responses. Last-Modified should be included where appropriate.
- A shared data structure to store the information of the IP address of each client and the requested files.

### 2.2 Advanced Goal

Concurrency can also be achieved through several other architectures. One other architecture that could be evaluated is thread pool architecture.It differs from the mutithreaded architecture in the server loop, where in rather than starting a new thread per connection, the connection is wrapped in a Runnable and

handed off to a thread pool with a fixed number of threads. The Runnable's are kept in a queue in the thread pool. When a thread in the thread pool is idle it will take a Runnable from the queue and execute it. [4]. The main advantage of this type of connection is that the maximum number of threads that the server can handle is bounded by the queue depth and hence includes determinism in system execution. A comparison study of multithreaded architecture and thread pool architecture is an interesting topic which will be done if it is feasible.

### 3 Project Evaluation Methodology

In order to understand the capabilities of multithreaded webserver application, a simple sequential single threaded webserver application will also be developed.

**Speedup** The time taken by the concurrent application will be compared to time taken by sequential application by varying the size of the served pages, the number of pages.

**Scalability** The concurrent program will be tested for different number of the incoming requests and scalability of the application will be tested for increasing requests. This could be tested on machines with different number of cores as well.

**Performance impact due to logging** Since a shared data structure will be used to log the information of the request, the request response will be impacted by this. A detailed study of the performance of impact will be done. One of the tools that will be used for performance monitoring includes `httperf` which tests the performance of a server by simulating the HTTP workloads on the server.

### 4 References

1. <http://en.wikipedia.org/wiki/Webserver>
2. <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.95.679&rep=rep1&type=pdf>
3. <http://www.cse.chalmers.se/edu/year/2013/course/EDA343/>
4. <http://tutorials.jenkov.com/java-multithreaded-servers/thread-pooled-server.html>