

# 毕业设计中期检查答辩

黄丛宇

06161032

指导老师：马瑞芳

西安交通大学 软件学院 软件62班

May 8, 2010

# 目录

- 1 课题名称
- 2 背景和意义
- 3 课题主要任务
- 4 已完成任务
- 5 存在的问题
- 6 后期工作计划

## 基于动态链接技术的web服务器动态扩展功能 接口的设计与实现

- Web服务器：

- 互联网的核心组成部分，支撑整个互联网应用服务。
- 适应互联网应用的不断更新变化。
- 必须保证7\*24小时的运行。

- Web服务器现状：

- 大部分都不支持功能的动态增加。
- 必须重启或重新编译。

- Web服务器：

- 互联网的核心组成部分，支撑整个互联网应用服务。
- 适应互联网应用的不断更新变化。
- 必须保证7\*24小时的运行。

- Web服务器现状：

- 大部分都不支持功能的动态增加。
- 必须重启或重新编译。

- 针对以上问题，本课题将基于动态链接库技术，使服务器在运行期间，可以动态的获知模块的增加并加载模块。
- 本系统实现了服务器的基本功能，重点实现模块动态加载特性。

- 针对以上问题，本课题将基于动态链接库技术，使服务器在运行期间，可以动态的获知模块的增加并加载模块。
- 本系统实现了服务器的基本功能，重点实现模块动态加载特性。

# 课题主要任务

- ① 学习动态链接库和web服务器设计实现的相关知识和技术。
  - Nonblocking IO, IO multiplexing, 线程池, 动态链接技术等。
- ② 技术可行性论证, 需求分析, 构造系统结构。
  - 分析HTTP协议, 设计接口, 构造服务器的整体框架等。
- ③ 搭建开发环境, 设计和编码实现。
  - 完成服务器的编码。实现预期的功能。
- ④ 撰写毕业设计论文
- ⑤ 查找外文文献并翻译。



# 课题主要任务

- ① 学习动态链接库和web服务器设计实现的相关知识和技术。
  - Nonblocking IO, IO multiplexing, 线程池, 动态链接技术等。
- ② 技术可行性论证, 需求分析, 构造系统结构。
  - 分析HTTP协议, 设计接口, 构造服务器的整体框架等。
- ③ 搭建开发环境, 设计和编码实现。
  - 完成服务器的编码。实现预期的功能。
- ④ 撰写毕业设计论文
- ⑤ 查找外文文献并翻译。

# 课题主要任务

- ① 学习动态链接库和web服务器设计实现的相关知识和技术。
  - Nonblocking IO, IO multiplexing, 线程池, 动态链接技术等。
- ② 技术可行性论证, 需求分析, 构造系统结构。
  - 分析HTTP协议, 设计接口, 构造服务器的整体框架等。
- ③ 搭建开发环境, 设计和编码实现。
  - 完成服务器的编码。实现预期的功能。
- ④ 撰写毕业设计论文
- ⑤ 查找外文文献并翻译。

# 课题主要任务

- ① 学习动态链接库和web服务器设计实现的相关知识和技术。
  - Nonblocking IO, IO multiplexing, 线程池, 动态链接技术等。
- ② 技术可行性论证, 需求分析, 构造系统结构。
  - 分析HTTP协议, 设计接口, 构造服务器的整体框架等。
- ③ 搭建开发环境, 设计和编码实现。
  - 完成服务器的编码。实现预期的功能。
- ④ 撰写毕业设计论文
- ⑤ 查找外文文献并翻译。

# 课题主要任务

- ① 学习动态链接库和web服务器设计实现的相关知识和技术。
  - Nonblocking IO, IO multiplexing, 线程池, 动态链接技术等。
- ② 技术可行性论证, 需求分析, 构造系统结构。
  - 分析HTTP协议, 设计接口, 构造服务器的整体框架等。
- ③ 搭建开发环境, 设计和编码实现。
  - 完成服务器的编码。实现预期的功能。
- ④ 撰写毕业设计论文
- ⑤ 查找外文文献并翻译。

- 完成动态链接库技术和web服务器设计相关资料的搜集和学习。
  - 掌握了动态链接库的基本使用。
  - 掌握了Nonblocking IO和IO Multiplexing的使用和注意事项。
  - 学习有关线程池的资料并编写了一个简单的线程池，以备后期使用。

- 完成动态链接库技术和Web服务器设计相关资料的搜集和学习。
  - 掌握了动态链接库的基本使用。
  - 掌握了Nonblocking IO和IO Multiplexing的使用和注意事项。
  - 学习有关线程池的资料并编写了一个简单的线程池，以备后期使用。

- 完成技术可行性论证，需求分析，构造系统结构。
  - 研究学习了HTTP/1.1协议（RFC2616）。理解掌握了HTTP协议的基本内容和处理过程。
  - 根据HTTP协议的内容设计插件的接口。
  - 完成服务器整体架构的设计。
    - 对于连接的处理采用状态机。
    - 每个IO事件调用一个线程进行处理。
    - 采用Inotify检测插件的增加和删除。

- 完成技术可行性论证，需求分析，构造系统结构。
  - 研究学习了HTTP/1.1协议（RFC2616）。理解掌握了HTTP协议的基本内容和处理过程。
  - 根据HTTP协议的内容设计插件的接口。
  - 完成服务器整体架构的设计。
    - 对于连接的处理采用状态机。
    - 每个IO事件调用一个线程进行处理。
    - 采用Inotify检测插件的增加和删除。



- 完成技术可行性论证，需求分析，构造系统结构。
  - 研究学习了HTTP/1.1协议（RFC2616）。理解掌握了HTTP协议的基本内容和处理过程。
  - 根据HTTP协议的内容设计插件的接口。
  - 完成服务器整体架构的设计。
    - 对于连接的处理采用状态机。
    - 每个IO事件调用一个线程进行处理。
    - 采用Inotify检测插件的增加和删除。

- 完成开发环境的搭建。完成服务器设计并编码实现。
  - 开发环境：Linux(Debian5.0)+gcc+vim(gedit)
  - 完成服务器的编码工作。
    - 实现了HTTP/1.1的GET, POST和HEAD方法。
    - 实现了HTTP/1.1中的Persistent Connections和Pipelining。
    - 实现了HTTP/1.0中的Nonpersistent Connections。
    - 可以动态的加载和删除插件。
    - 实现了一个验证性质的插件：dir\_index。

- 完成开发环境的搭建。完成服务器设计并编码实现。
  - 开发环境：Linux(Debian5.0)+gcc+vim(gedit)
  - 完成服务器的编码工作。
    - 实现了HTTP/1.1的GET, POST和HEAD方法。
    - 实现了HTTP/1.1中的Persistent Connections和Pipelining。
    - 实现了HTTP/1.0中的Nonpersistent Connections。
    - 可以动态的加载和删除插件。
    - 实现了一个验证性质的插件：dir\_index。

目前已经完成课题任务的  
80%

目前已经完成课题任务的  
80%

# 存在的问题

## ❶ 对HTTP协议的理解不够透彻。

- 缓存
- 条件获取

## ❷ 代码存在BUG。

- 内存泄漏
- 线程并发出现数据混乱

# 存在的问题

- ① 对HTTP协议的理解不够透彻。
  - 缓存
  - 条件获取
- ② 代码存在BUG。
  - 内存泄漏
  - 线程并发出现数据混乱

# 后期工作计划

- ① 继续深入研究学习HTTP协议。
- ② 调成程序，修改BUG。
- ③ 编写毕业设计论文。
- ④ 查找翻译外文资料。
- ⑤ 准备毕业设计答辩文档。



*That 's all!*