**计算机网络课程项目报告**

1. 项目概述

本项目是一个实现Web缓存的代理服务器，支持基本的Web缓存功能，支持多线程。代理服务器能够对HTTP Get请求及远端服务器响应内容进行缓存，在再次收到客户端相同请求时，能够检查缓存的存在以及有效性，根据具体情况为客户端提供缓存服务，以提高上网效率。当缓存不存在时，代理服务器会向远端服务器发送请求，并缓存和转发给客户端。当缓存存在时，则提取缓存响应内容中上一次修改时间，并向远端服务器询问是否被修改，如果被修改则向远端服务器发送请求，更新缓存并转发给客户端。对于HTTP其它请求，则不予缓存。由于现网络大多数都使用https协议。代理服务器通常采用网络隧道的方式传输https协议。所以，本代理服务器支持对http connect请求进行处理，并在建立连接后，在客户端和远端服务器之间进行盲转发。因此，本代理服务器在运行时，能够保证用户正常上网。

1. 项目开发环境和工具

操作系统：Windows 11

开发语言：java

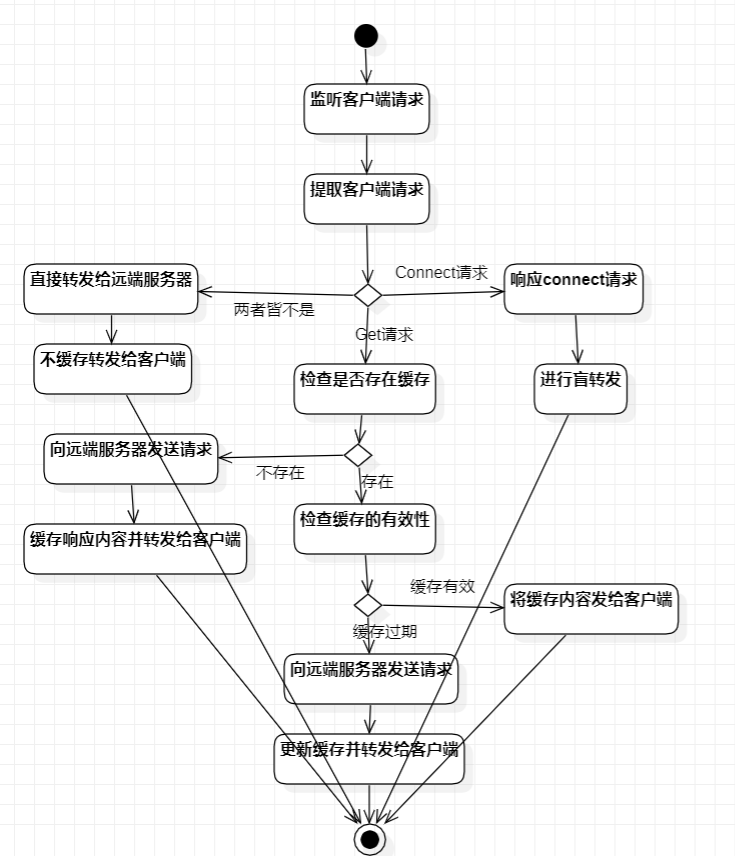
集成开发环境：intellij IDEA 2021.1

JDK版本：15

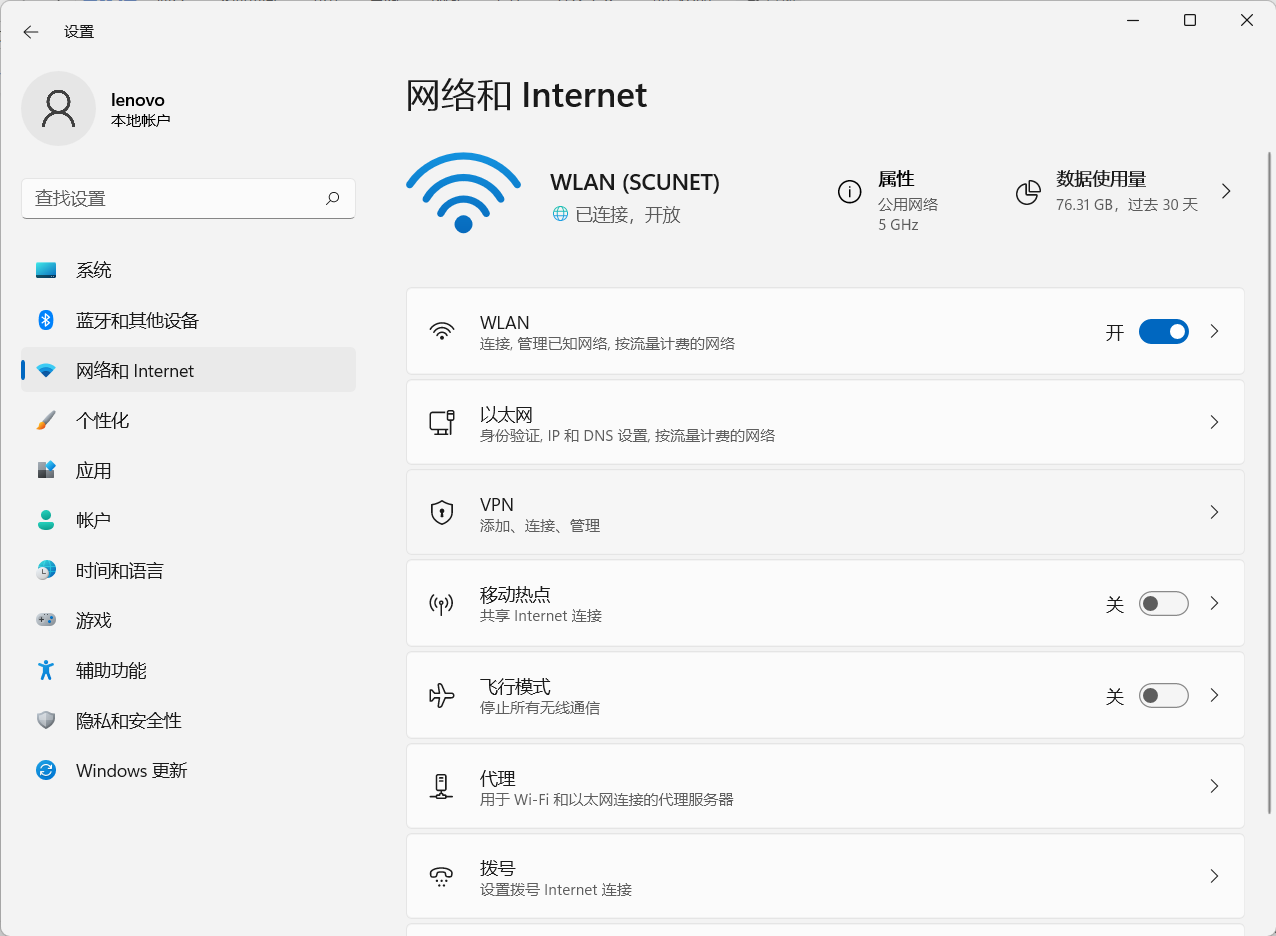
1. 项目设计

本项目主要包括两个文件。HTTPProxy.java文件包含了本项目中全部源码，cache.properties由程序运行后自动生成，以键值对的方式保存缓存内容（键为http请求的url,值为远端服务器响应内容）。

程序运行主要流程：



1. 使用方法
2. 打开电脑的网络设置页面，选择代理选项

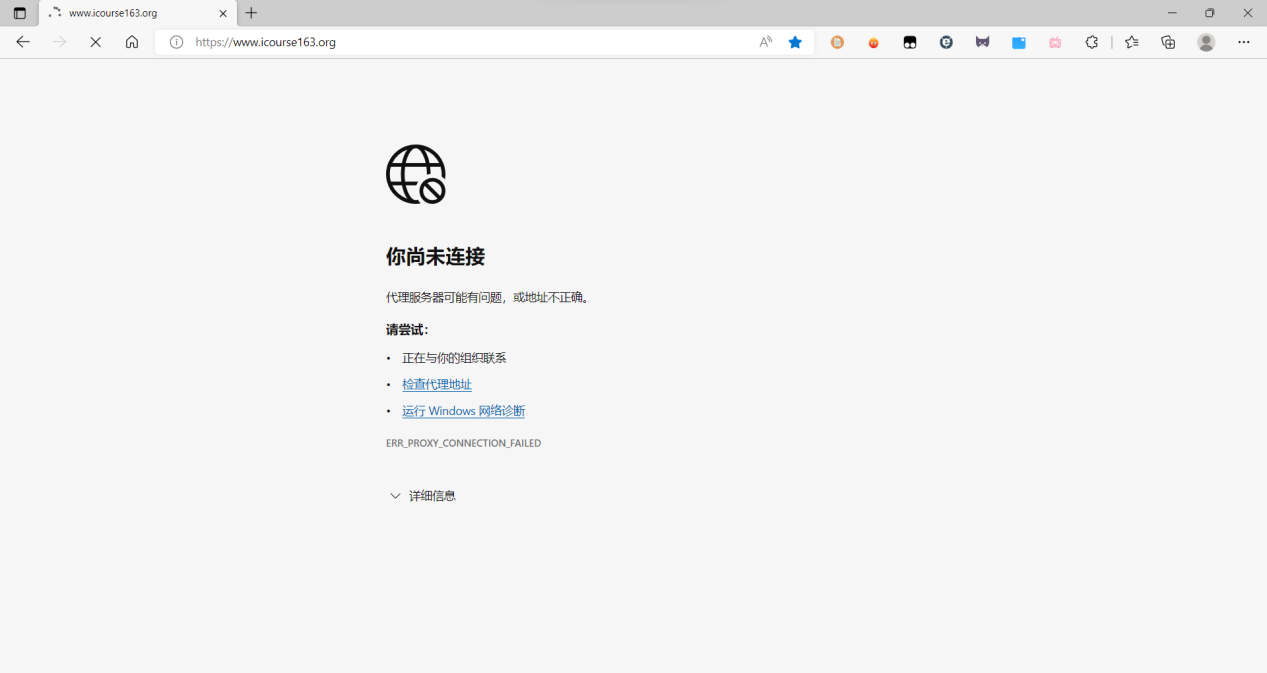




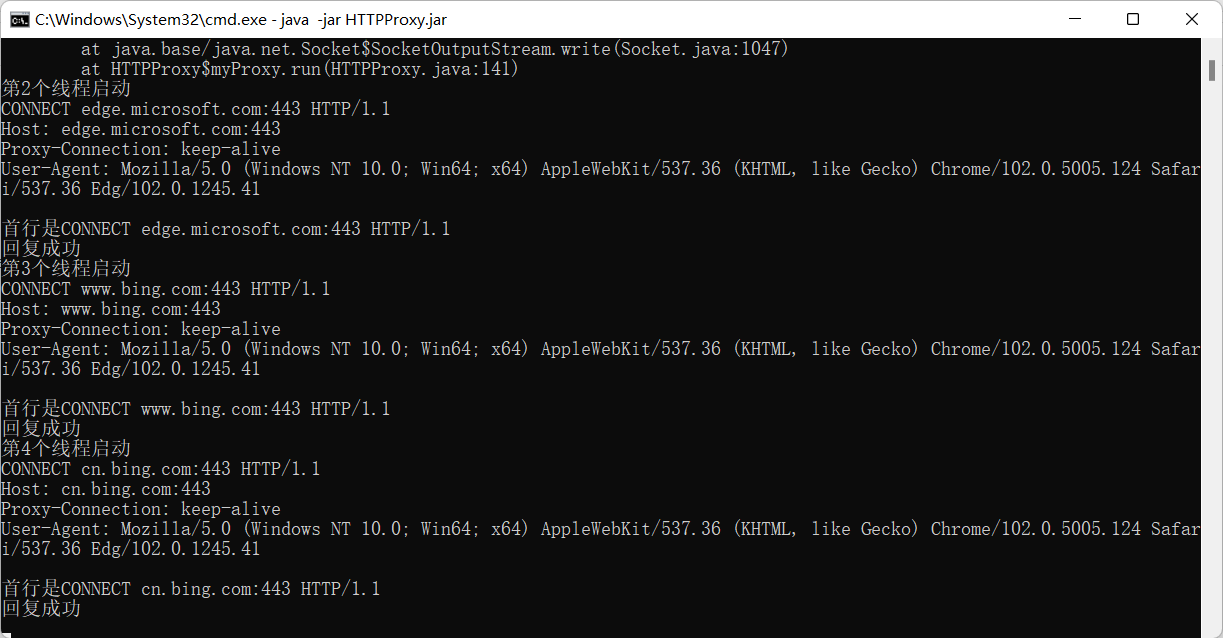
1. 编辑代理服务器信息，地址设为本机地址127.0.0.1，端口号为8341,点击保存。



1. 此时浏览器将无法正常使用，如果仍能正常使用，则存在其他代理设置，比如谷歌助手等插件，可以关闭插件或者换一个浏览器。



1. 运行HTTPProxy.jar，需要电脑有java环境。打开文件目录下的cmd,输入java -jar HTTPProxy.jar。程序正常启动。

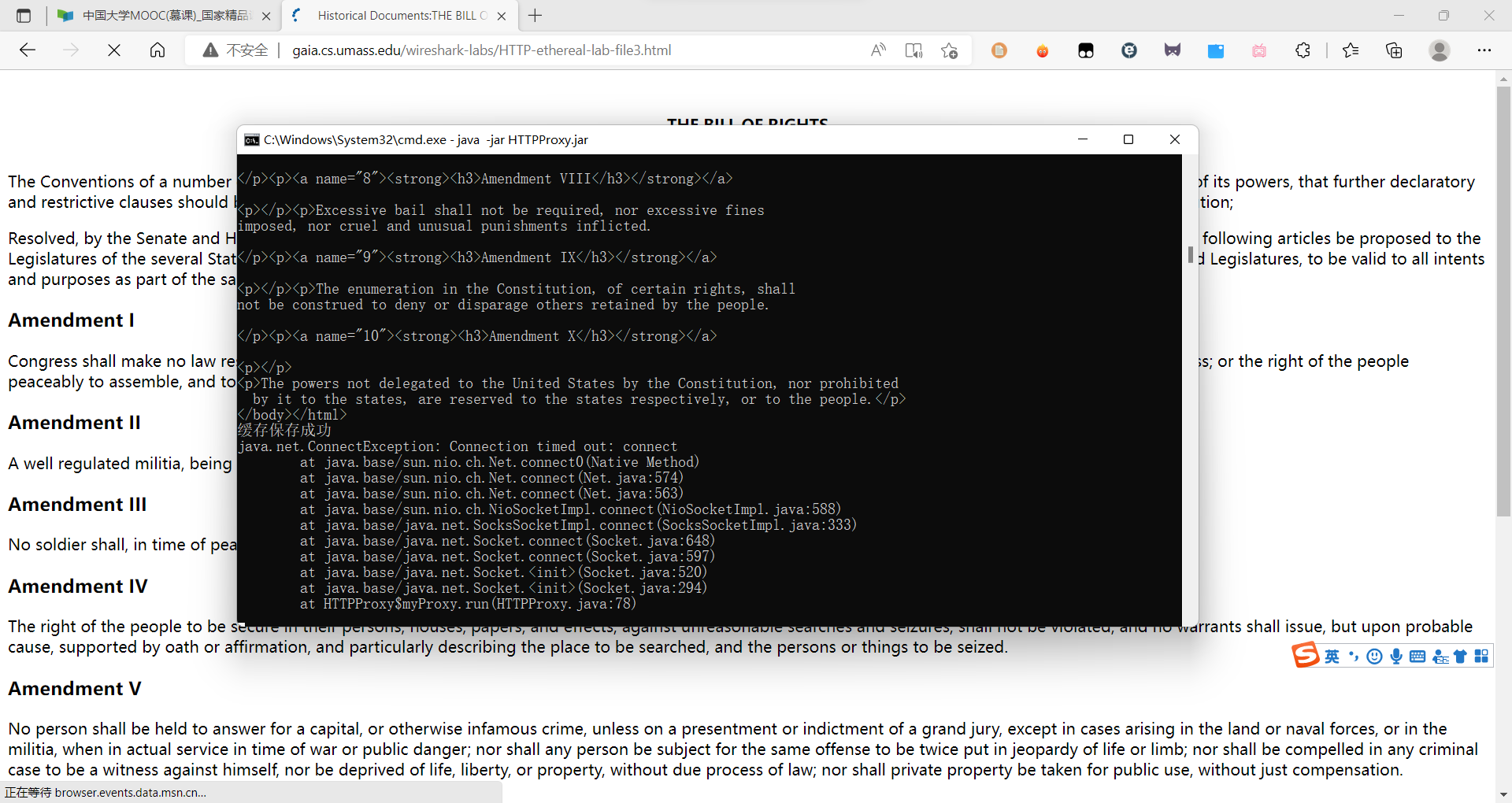


此时可以正常上网。Jar文件目录下自动生成cache.properties文件。

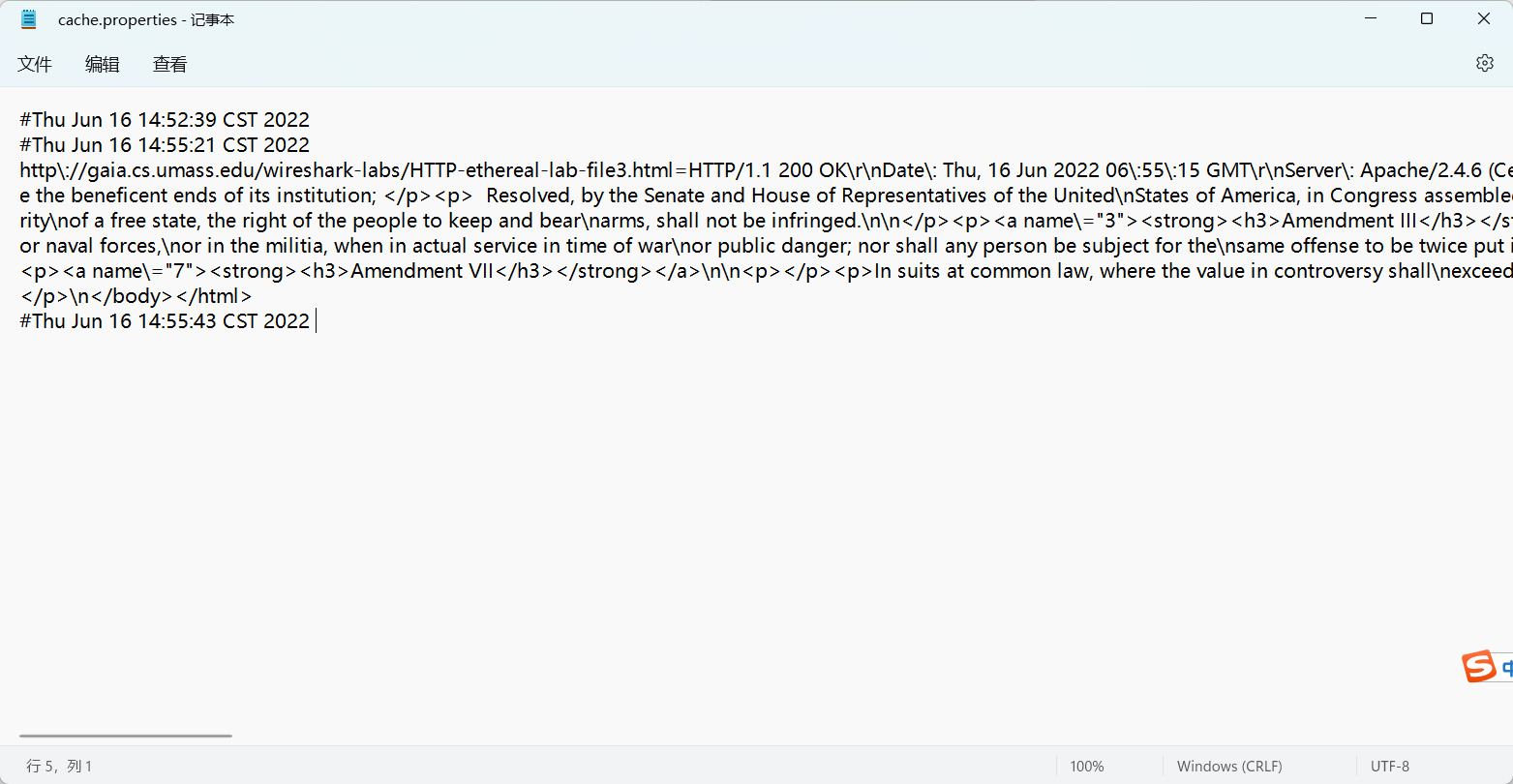


我们拿实验课Lab11中的HTTP网站来试一试。

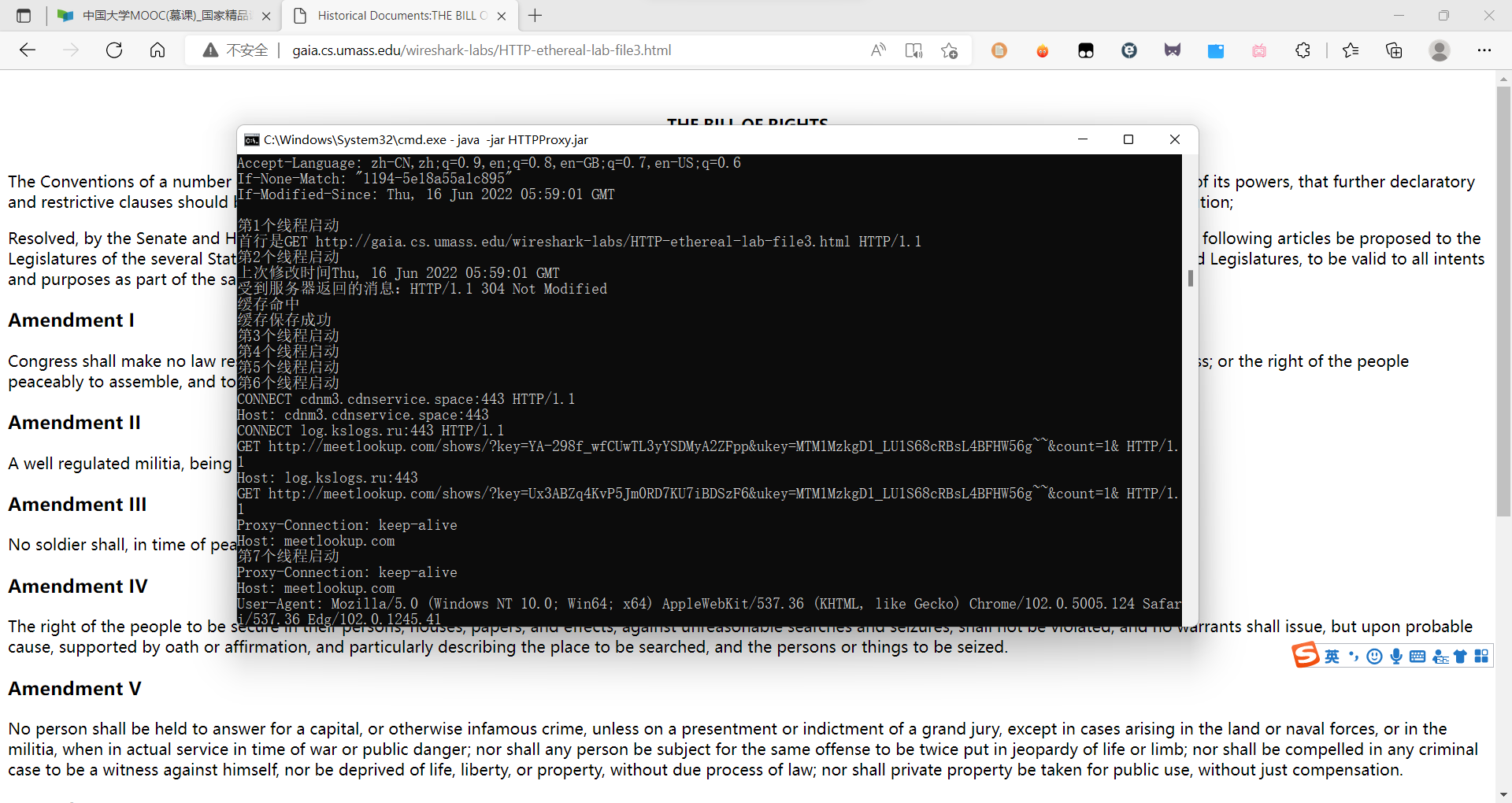
http://gaia.cs.umass.edu/wireshark-labs/HTTP-ethereal-lab-file3.html



查看一下cache.properties文件。



刷新网页，第二次访问该网址，可以看到显示缓存命中。



值得注意的是，浏览器一般也有缓存功能。所以如果浏览器在此之前存在缓存，则启动代理服务器之后，第一次访问该网址发送的请求便不是单纯的请求资源，而是询问远端服务器是否修改了资源，得到的响应是304之类。而代理服务器不会缓存这种请求和响应，因为程序设定只缓存响应内容中有Last-modified字样的。所以要测试其功能，可能需要清理浏览器的缓存。

1. 项目总结

完成本项目费了一番功夫，也很有收获。做项目时遇到的主要难点是如何处理https协议和如何设计缓存结构。我不甘心我做的这个代理服务器面对绝大多数网站不但无能为力，而且阻碍用户的正常使用。为此，我查阅了大量资料，解决了很多问题和bug。因此，收获也很大，首先是更深入地理解了http协议和应用层工作的流程。其次，有效地锻炼了我的动手能力和解决问题能力。计算机网络这门课不但要掌握理论知识，更要不断实践，实践出真知。唯有如此，我们才能把这么课学好学扎实。