Tedu.cn
达内教育

欢迎大家来到第五阶段课程

《分布式流媒体》实训项目

预习课

TNV DAY06

预习课

tedu.cn
达内教育

预习内容

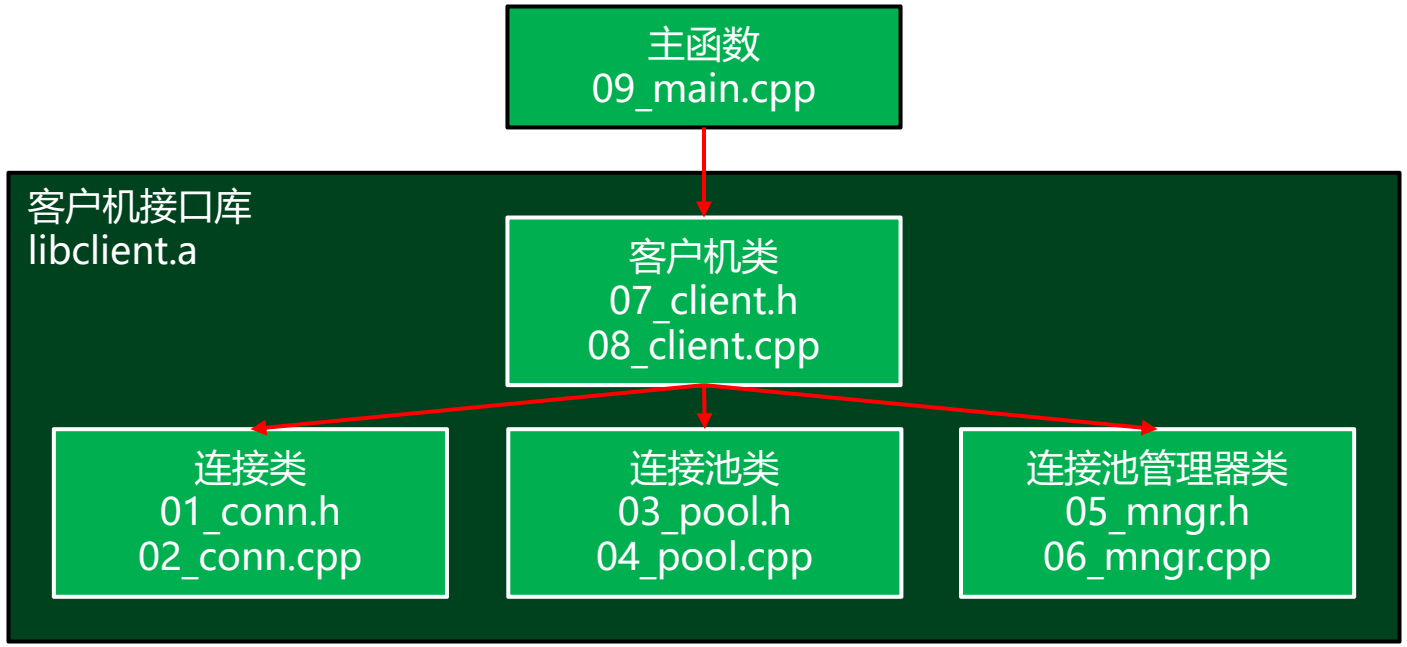
客户机（1）

Tedu.cn
达内教育

客户机（1）

组织结构

- 05_client



开发计划

知识讲解

序号	内容	文档/代码	时间
58	声明连接类	05_client/01_conn.h	1小时
59	实现连接类	05_client/02_conn.cpp	
60	声明连接池类	05_client/03_pool.h	1小时
61	实现连接池类	05_client/04_pool.cpp	
62	声明连接池管理器类	05_client/05_mngr.h	

开发计划

知识讲解

序号	内容	文档/代码	时间
63	实现连接池管理器类	05_client/06_mngr.cpp	
64	声明客户机类	05_client/07_client.h	
65	实现客户机类	05_client/08_client.cpp	1小时
66	定义主函数	05_client/09_main.cpp	
67	构建脚本	05_client/Makefile	

连接类(conn_c)的属性、构造和析构

知识讲解

- 成员变量
  - 目的地址：m_destaddr
  - 连接超时：m_ctimeout
  - 读写超时：m_rtimeout
  - 连接对象：m_conn
  - 错误号：m_errnumb
  - 错误描述：m_errdesc

连接类(conn_c)的属性、构造和析构

知识讲解

- 构造函数
  - 检查目的地址
  - 复制目的地址
- 析构函数
  - 关闭连接
  - 释放目的地址

Tedu.cn
达内教育

附录：程序清单

TNV/src/05_client/01_conn.h

知识讲解

```
//客户机
//声明连接类
//
#pragma once

#include <string>
#include <lib_acl.hpp>
//
//连接类
//
class conn_c: public acl::connect_client {
public:
        //构造函数
        conn_c(char const* destaddr, int ctimeout = 30,
                int rtimeout = 60);
```

TNV/src/05_client/01_conn.h

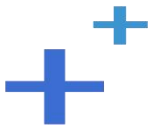
```
//析构函数
~conn_c(void);

//从跟踪服务器获取存储服务器地址列表
int saddrs(char const* appid, char const* userid,
        char const* fileid, std::string& saddrs);
//从跟踪服务器获取组列表
int groups(std::string& groups);

//向存储服务器上传文件
int upload(char const* appid, char const* userid,
        char const* fileid, char const* filepath);
//向存储服务器上传文件
int upload(char const* appid, char const* userid,
        char const* fileid, char const* filedata, long long filesize);
```

知识讲解

TNV/src/05_client/01_conn.h

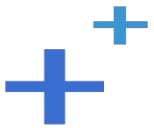
知识讲解

```
//向存储服务器询问文件大小
int filesize(char const* appid, char const* userid,
        char const* fileid, long long* filesize);
//从存储服务器下载文件
int download(char const* appid, char const* userid,
        char const* fileid, long long offset, long long size,
        char** filedata, long long* filesize);
//删除存储服务器上的文件
int del(char const* appid, char const* userid, char const* fileid);

//获取错误号
short errnumb(void) const;
//获取错误描述
char const* errdesc(void) const;
```



TNV/src/05_client/01_conn.h

```
protected:
        //打开连接
        bool open(void);
        //关闭连接
        void close(void);

private:
        //构造请求
        int makerequ(char command, char const* appid,
                char const* userid, char const* fileid, char* requ);
        //接收包体
```

TNV/src/05_client/01_conn.h

```
        int recvbody(char** body, long long* bodylen);
        //接收包头
        int recvhead(long long* bodylen);

        char*                m_destaddr; //目的地址
        int                  m_ctimeout; //连接超时
        int                  m_rtimeout; //读写超时
        acl::socket_stream* m_conn;      //连接对象
        short                m_errnumb;  //错误号
        acl::string          m_errdesc;  //错误描述
};
```

知识讲解

TNV/src/05_client/02_conn.cpp


```
//客户机
//实现连接类
//
#include <sys/sendfile.h>
#include <lib_acl.h>
#include "02_proto.h"
#include "03_util.h"
#include "01_conn.h"

//构造函数
conn_c::conn_c(char const* destaddr, int ctimeout /* = 30 */,
        int rtimeout /* = 60 */): m_ctimeout(ctimeout),
        m_rtimeout(rtimeout), m_conn(NULL) {
        //检查目的地址
        acl_assert(destaddr && *destaddr);
```

TNV/src/05_client/02_conn.cpp

```
        //复制目的地址
        m_destaddr = acl_mystrdup(destaddr);
}

//析构函数
conn_c::~conn_c(void) {
        //关闭连接
        close();
        //释放目的地址
        acl_myfree(m_destaddr);
}
```

知识讲解

直播课见