

Lab2 文档

程序结构:

1:RPC 逻辑结构

更改了 lab1 中的实现方式，实现了 Client 和 Server 的分离，客户端不再直接调用服务器端的函数，而是通过中间接口 DataNodeStub 和 NameNodeStub 调用实现。

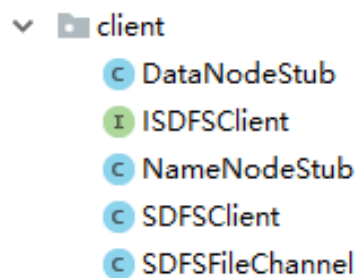
RPC 逻辑：每当客户端需要向服务端进行操作，例如创建文件、读取数据时，不再直接与 NameNode 或 DataNode 交互，通过对 Stub 的调用，Stub 与服务器端交互，发送 socket 到服务端，服务端接收 Socket，处理消息调用对应函数，进行操作获取数据，之后服务端通过 socket 返回消息到 Stub，Stub 获取返回消息返回给客户端。

以读取数据为例：

1. 服务端开启服务
2. 客户端执行打开文件的命令
3. 客户端通过 NameNodeStub 调用 open 函数
4. NameNodeStub 创建与服务端的连接，并通过 socket 向 NameNode 发送消息，表明要调用函数
5. NameNode 接收处理消息，执行服务端 open 函数，获取通道
6. NameNode 将通道封装，通过 socket 发送消息返回 NameNodeStub
7. NameNodeStub 接收服务端 NameNode 返回的消息，解析消息，将通道返回给客户端
8. 客户端通过获取的通道，执行 read 指令

9. 客户端调用 DataNodeStub 的 read 函数
10. DataNodeStub 创建与 DataNode 的连接，将消息封装，通过 socket 将消息发送给 DataNode
11. DataNode 接收消息，调用服务端 read 函数，获取 read 数据
12. DataNode 将读取的数据封装，通过 socket 将数据发送给 DataNodeStub
13. DataNodeStub 获取来着 DataNode 的数据，返回给客户端
14. 客户端获取数据，读取完成

代码结构：



Client 包主体不变，加入了两个新的类，也就是服务器端与客户端通信的接口，即 DataNodeStub 和 NameNodeStub

DataNodeStub: 继承 IDataNode 接口，函数与 DataNode 相同，每次客户端与服务器端通信，都通过 DataNodeStub 发送消息给 DataNode 并接收返回消息，实现数据读取写入

NameNodeStub: 继承 INameNode 接口，函数与 NameNode 相同，客户端需要与服务端 NameNode 交互式，通过该类发送 socket 消息给 NameNode，实现交互。

```

public void listenRequest() {
    // 每调用一次监听函数，端口就会加一，并且创立监听连接，
    // 端口加一是为了防止端口重用的问题
    ++port;
    SocketUtils.listen(port);
}

```

NmaeNode 与 DataNode 主体不变，增加了 listenRequest 函数，该函数将 NameNode 和 DataNode 实现为服务器，监听客户端发来的消息，并处理消息。

```

v socketUtil
  SocketClientMessage
  SocketServerMessage
  SocketUtils

```

新增 socketUtil 包：

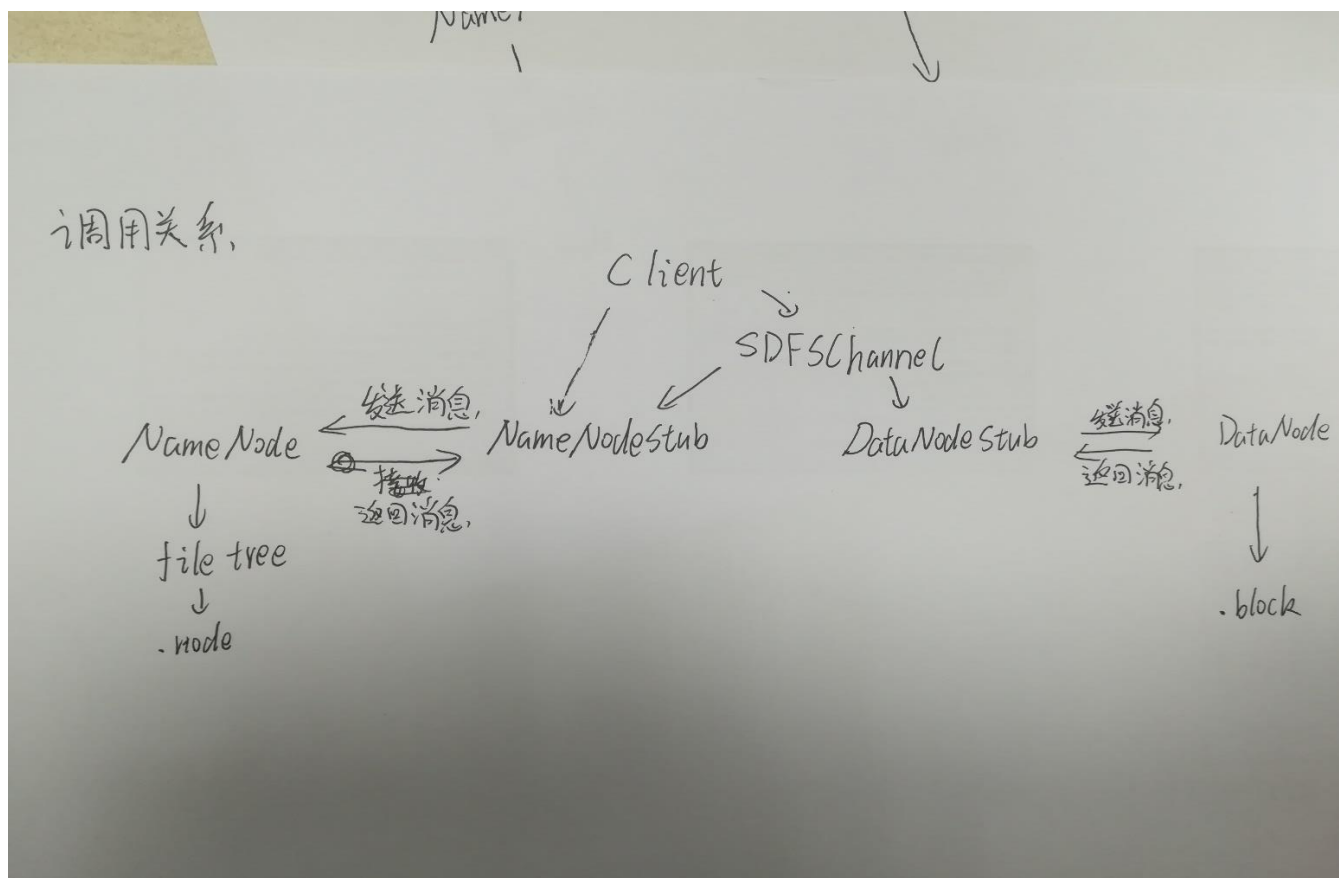
SocketClientMessage： 客户端发送的消息封装类，类中属性包含了本次客户端需要调用的函数所在类名称、函数名、函数参数类型、函数参数

SocketServerMessage： 服务端发送的消息封装类，类中属性包含了客户端本次调用的返回值、调用函数的抛出异常

SocketUtils： 客户端与服务端通过 Socket 发送消息的工具类，实现类客户端发送与接收消息的函数，服务端发送与接收消息的函数以及服务的监听函数。

消息发送接收实现过程是： 客户端发送消息，服务端接收消息，通过反射机制调用对应函数，服务端包装返回值发还给客户端，客户端接收处理消息

调用关系



如图示，NameNodeStub 和 DataNodeStub 作为服务端与客户端的通信接口，以 Stub 传送消息，分离了客户端与服务端，实现了 RPC。

结构更改：

原主体结构没有变化，更改了客户端调用的接口，在客户端与服务端的直接通信之间加了一层 RPC，两者通信通过中间接口 Stub 实现

遇到问题：

1：当客户端通过 Socket 发送给服务端的时候，Stub 没有办法直接调用服务端的函数，不知道如何调用

解决：学习了反射机制，将要调用的函数信息（函数所在类、函数名、参数等）封装进消息中，服务端获取消息，采用反射机制调用对应的函数。

2：客户端连接客户端发送消息，经常发生前两次消息可以发送，接下来的消息不能发送

解决：后来发现客户端 socket 只有在 stub 创建时连接一次服务器，有时候客户端 socket 不能发送成功。于是改了实现，每次需要发送消息时就创建新的连接，不使用旧连接

3：测试时发现多个测试一起运行，发送占用端口问题。

解决：为服务端分配不同的端口，每次启动一次服务端，端口就增加 1。客户端新建 Stub 对象时，会获取当前服务端最新端口，之后该 Stub 的所有连接都会发送到这个端口连接

关于修改测试用例：

测试用例主体没有修改

但是因为 NameNode 测试函数里面用到了 close 函数，close 函数需要使用服务端与客户端的连接。所以在每个 NameNode 测试函数里面开头部分都加上了两行代码：创建新的线程，并启动该函数中的 NameNode 的监听服务。