软件复用课程设计

Client-Server 应用程序

程序文档2.0

小组： Team10

成员： 谢志杰 1352975

计鹏玥 1352914

王思尧 1352896

谢明玥 1352937

徐锦程 1353012

文档变更记录

|  |  |  |
| --- | --- | --- |
| 更改人 | 日期 | 更改内容 |
| 计鹏玥 | 2016年5月2日 | 创建文档 |
| 计鹏玥，谢明玥 | 2016年5月3日 | 接口说明 |
|  |  |  |
|  |  |  |

[文档变更记录 1](#_Toc3881)

[1．开发规划 1](#_Toc20117)

[1.1开发人员 1](#_Toc6361)

[1.2开发计划 1](#_Toc9126)

[1.3开发环境和工具 1](#_Toc31966)

[2．功能扩展 2](#_Toc8456)

[2.1 SaveToFile&Packer 2](#_Toc3845)

[2.1.1 功能扩展说明 2](#_Toc14286)

[2.1.2 扩展方案 2](#_Toc7846)

[2.1.3 问题&解决 2](#_Toc23098)

[2.2 Room&Unpacker 3](#_Toc23575)

[2.2.1 功能扩展说明 3](#_Toc27104)

[2.1.2 组内广播 3](#_Toc13013)

[2.1.3 按周归档 3](#_Toc15289)

[3．接口规范 3](#_Toc1461)

[3.1 File 3](#_Toc2558)

[3.1.1 GetPackageByTimer.java 3](#_Toc26074)

[3.1.2 SaveToFile.java 4](#_Toc10700)

[3.2 utils 4](#_Toc25171)

[3.2.1 Packer.java 4](#_Toc19157)

[3.2.2 PackPerDay.java 5](#_Toc15740)

[3.2.3 PackPerWeek.java 5](#_Toc31148)

[3.2.4 Pair.java 6](#_Toc32414)

[3.2.5 Room.java 6](#_Toc8563)

[3.2.6 Unpacker.java 7](#_Toc18788)

1．开发规划

1.1开发人员

|  |  |
| --- | --- |
| 学号 | 姓名 |
| 1352975 | 谢志杰 |
| 1352914 | 计鹏玥 |
| 1352896 | 王思尧 |
| 1352937 | 谢明玥 |
| 1353012 | 徐锦程 |

1.2开发计划

|  |  |  |
| --- | --- | --- |
| 起始时间 | 结束时间 | 计划 |
| 2016年4月20日 | 2016年4月27日 | 基于选择的组件进行开发，保存所有收到的消息到文件，并将每天所有的输出文件归档成一个压缩包。 |
| 2016年4月27日 | 2016年5月4日 | 对文件进行输出和压缩，配置同组成员，并只向组内成员广播消息；将每周的七个归档文件重新生成一个压缩包。 |
|  |  |  |
|  |  |  |

1.3开发环境和工具

GitHub - 版本管理工具

Java - 开发语言

JUnit - 单元测试框架

RabbitMQ - 中间件

2．功能扩展

2.1 SaveToFile&Packer

2.1.1 功能扩展说明

(1) 客户端/服务端保存所有收到消息到文件

A.文件格式不限

B.文件路径可配置

(2) 将每天所有输出文件（客户端/服务端/性能管理）归档成一个压缩包

A.压缩包格式不限

B.压缩包路径可配置

2.1.2 扩展方案

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **时间** | **客户端** | **服务端** |
| 1 | 4.20 - 4.22 | 通过本小组新增构件SaveToFile构件来实现将消息写入文件功能，主要方法包括通过带参数的构造函数实现指定文件路径以及写入文件功能。  通过使用与Server端相同的压缩打包构件Packer来实现文件打包功能。 | 通过对我方的构件MessageUtils添加消息日志功能，以及添加新的压缩打包构件Packer，完成相应功能扩展。 |
| 2 | 4.23 - 4.25 | 本小组使用的PM构件提供方在不久后更新了PM构件并且实现了本次功能扩展要求的功能，因此本小组决定使用PM构件提供方提供的添加了扩展功能的PM构件来替换本小组自己实现的扩展构件，以此完成上述扩展功能的要求。 | 我们发现PM构件提供方（Team1）在PMv2.0上实现了相应功能，通过在PM构件上增加对消息的日志功能以及对所有归档文件的压缩打包功能来完成此次的功能扩展。于是我们决定直接使用最新的PM构件来重构代码。 |

2.1.3 问题&解决

|  |  |  |
| --- | --- | --- |
| **No.** | **问题** | **解决** |
| 1 | PM构件提供方的新版本构件中添加的消息记录与文件打包功能与PM性能管理构件的主要目的似有偏差。 | 由于这些功能的内部实现与性能管理的内部实现基本类似，我们决定继续使用其新增扩展功能，而放弃我们小组自己实现的新增功能。 |
| 2 | 构件提供方的最新PM构件在适配过程中出现了updateIndex阻塞，原因为一个readLock出现了死锁。且内部的定时器没有启动，导致没有文件记录，这个问题出现在IntervalLogger和ArchiveLogger上。 | 我们与构件提供方协调，将所有我们遇到的漏洞及相关细节提供给提供方，协助修复漏洞。 |

2.2 Room&Unpacker

2.2.1 功能扩展说明

(1) 配置同组成员，并只向同组成员广播消息。

(2) 将每周的七个归档文件重新生成一个压缩包。

2.1.2 组内广播

本项目的服务器端使用的是rabbitMQ作为消息队列，而rabbitMQ本身提供了exchange的概念，适用于我们的聊天室的概念。rabbitMQ提供了四种exchange：

•Direct

•Fanout

•Topic

•Headers

具体一点，我们使用的是：

Fanout Exchange – 不处理route key。我们只需要简单的将queue绑定到exchange上。一个发送到exchange的消息都会被转发到与该exchange绑定的所有queue上。很像子网广播，每台子网内的主机都获得了一份复制的消息。Fanout exchange转发消息是最快的。

通过制定不同的exchange name，我们就达到了区分不同组的目的。

2.1.3 按周归档

每一周，对本周产生的所有压缩包进行一次解压，并重新压缩成为一个新的压缩包。

3．接口规范

3.1 File

3.1.1 GetPackageByTimer.java

1. GetPackageByTimer(String path)

|  |  |  |
| --- | --- | --- |
| 参数 | 类型 | 含义 |
| path | String | 文件路径 |

3.1.2 SaveToFile.java

(1) SaveToFile()

获取配置信息所指定的路径。

1. SaveToFile(String s)

|  |  |  |
| --- | --- | --- |
| 参数 | 类型 | 含义 |
| s | String | 文件路径 |

1. getBufferedWriter()

获取输出流，并创建文件。

1. write(String str)

|  |  |  |
| --- | --- | --- |
| 参数 | 类型 | 含义 |
| str | String | 准备写入文件的内容 |

1. getDirectoryPath()

获取文件夹路径。

1. finalized()

关闭文件输入流。

3.2 utils

3.2.1 Packer.java

(1) Packer(String ipath, String opath)

|  |  |  |
| --- | --- | --- |
| 参数 | 类型 | 含义 |
| ipath | String | 文件读取路径 |
| opath | String | 文件保存路径 |

1. packupSuffix(String suffix)

|  |  |  |
| --- | --- | --- |
| 参数 | 类型 | 含义 |
| suffix | String | 文件后缀名称 |

(3) addToZipFile(File file, ZipOutputStream zos)

|  |  |  |
| --- | --- | --- |
| 参数 | 类型 | 含义 |
| file | File | 文件 |
| zos | ZipOutputStream | 文件输出为压缩的zip格式 |

1. PackerTimer(String logFilename)

|  |  |  |
| --- | --- | --- |
| 参数 | 类型 | 含义 |
| logFilename | String | 日志文件的名字 |

3.2.2 PackPerDay.java

1. PackPerDay(String iPath, String oPath)

|  |  |  |
| --- | --- | --- |
| 参数 | 类型 | 含义 |
| iPath | string | 文件读取路径 |
| oPath | string | 文件保存路径 |

3.2.3 PackPerWeek.java

(1) PackPerWeek(String iPath, String oPath)

|  |  |  |
| --- | --- | --- |
| 参数 | 类型 | 含义 |
| iPath | string | 文件读取路径 |
| oPath | string | 文件保存路径 |

3.2.4 Pair.java

(1) Pair(First l, Second r)

|  |  |  |
| --- | --- | --- |
| 参数 | 类型 | 含义 |
| l | First | first值 |
| r | Second | Second值 |

(2) getL()

|  |  |  |
| --- | --- | --- |
| 返回值 | 类型 | 含义 |
| first | First | 获取first值 |

(3) getR()

|  |  |  |
| --- | --- | --- |
| 返回值 | 类型 | 含义 |
| second | Second | 获取second值 |

(4) setL()

|  |  |  |
| --- | --- | --- |
| 参数 | 类型 | 含义 |
| l | First | 设置first值 |

(5) setR()

|  |  |  |
| --- | --- | --- |
| 参数 | 类型 | 含义 |
| r | Second | 设置second值 |

3.2.5 Room.java

(1) Room(int capacity, long roomID)

|  |  |  |
| --- | --- | --- |
| 参数 | 类型 | 含义 |
| capacity | int | 小组内最多可容纳成员人数 |
| roomID | long | 小组ID |

(2) getID()

|  |  |  |
| --- | --- | --- |
| 返回值 | 类型 | 含义 |
| roomID | long | 获取到的小组ID |

(3) isEmpty()

|  |  |  |
| --- | --- | --- |
| 返回值 | 类型 | 含义 |
|  | boolean | 小组内成员人数是否为0 |

(4) size()

|  |  |  |
| --- | --- | --- |
| 返回值 | 类型 | 含义 |
| people.size | int | 小组内当前成员数 |

(5) addOne(long id)

|  |  |  |
| --- | --- | --- |
| 参数 | 类型 | 含义 |
| id | long | 小组增加的成员的ID |

|  |  |  |
| --- | --- | --- |
| 返回值 | 类型 | 含义 |
|  | boolean | 小组是否承购添加成员 |

(6) minOne(long id)

|  |  |  |
| --- | --- | --- |
| 参数 | 类型 | 含义 |
| id | long | 小组删除成员的ID |

3.2.6 Unpacker.java

(1) unZip(String zipFile, String outputFolder)

|  |  |  |
| --- | --- | --- |
| 参数 | 类型 | 含义 |
| zipFile | String | 压缩文件的文件名 |
| outputFolder | String | 文件的解压路径 |