

北京航空航天大学

软件工程综合实验方法 总结报告

Redis

SY1406108 陈志伟 SY1406112 王珊珊

SY1406311 林 璐 SY1406117 王志鹏

2015/06/11

目录

1 范围	1
1.2 编写目的	1
2 实验过程所遇问题	1
2.1 选题问题	1
2.1.1 问题描述	1
2.1.2 处理方法	1
2.1.3 有效性评估	2
2.2 进度计划问题	2
2.2.1 问题描述	2
2.2.2 处理方法	3
2.2.3 有效性评估	3
2.3 工作量分析问题	4
2.3.1 问题描述	4
2.3.2 处理方法	5
2.3.3 有效性评估	5
2.4 配置管理分析问题	6
2.4.1 问题描述	6
2.4.2 处理方法	7
2.4.3 有效性评估	7
2.5 需求分析问题	9
2.5.1 问题描述	9
2.5.2 处理方法	10
2.5.3 有效性评估	10
2.6 测试与评审问题	11
2.6.1 问题描述	11
2.6.2 处理方法	11
2.6.3 有效性评估	12
3 参考文献	12

1 范围

1.1 编写目的

本文档主要说明本组在 8 个实验中遇到的难点问题，并描述处理方法（RUCM 标准），同时对有效性进行评估，一共包括 6 个问题。

2 实验过程所遇问题

2.1 选题问题

2.1.1 问题描述

一个短时间内组建的团队，四个成员彼此不熟悉，需要共同对一个开源软件进行为期 17 周的分析学习。首先面对的问题是：应该如何选择开源软件，使得：

- 1) 具有合适规模，能在 17 周内进行比较充分的分析；
- 2) 在四个成员的能力范围内，比如编程语言、计算机基础等；
- 3) 具有一定价值，成员能从中有所收获。

2.1.2 处理方法

Method Specification		
Name	选题	
Brief Description	从众多开源项目中选择规模合适，成员能力范围内，有学习价值的工程	
Precondition	开源网站中的项目	
Primary Actor	所有组员，共同调研讨论	
Dependency	None	
Generalization	None	
Basic Flow	Steps	
	1	各成员分头进行调研；
	2	提交各自调研软件的基本介绍，包括 Spark, python 语言和 Redis；
	3	开会讨论，结合各自能力水平，软件规模和价值等进行，并初步评价是否易于分工；
	4	排除 Spark，因为规模过大，且编程语言为 scala，入门代价较大，超出成员能力范围；
	5	排除 python，因为大家比较偏向于应用类的软件，对

		于实验室工作等比较有价值；
	6	确定 Redis，规模合适，C 语言，且应用广泛；
	7	基于 Redis 资料，初步进行分工。
	Postcondition	确定选题为 Redis

解决方式主要是调研和讨论。调研可以充分调动各个组员收集信息和资料，局限在于开发经验和对其他成员的了解有限，不能完全符合现实情况；讨论则可以综合、权衡各方面，形成比较合理的选择。从大家合作过程和最后的实验结果来看，分工明确且合理，能满足实验要求，因此这种方式是有效的。

2.1.3 有效性评估

本问题处理方法的有效性体现在其规模是否合适，是否在能力范围内，且是否有实用价值。

首先，Redis 代码规模在 2-3 万行，且代码风格清晰，具有比较完整的文档资料，因此规模是合适的；

其次，Redis 代码由 C 语言编写，是大家学习过的语言；且 Redis 是一种键值数据库，大家对数据库也很熟悉。因此对其分析在能力范围内；

最后，Redis 应用广泛，随着应用对高性能需求的增加，NoSQL 逐渐在各大名企的系统架构中生根发芽。国内像新浪、淘宝，国外像 Flickr、Github 等均在使用 Redis 的缓存服务。

2.2 进度计划问题

2.2.1 问题描述

进度计划即为每个成员制定合适的项目任务，以及能够合理控制每个成员对项目的执行情况并进行追踪，通过这种管理监督能够按照计划进行项目执行并且保质保量地完成。同时是也是进度的控制，可以对项目的进度进行详细的划分，这样目标更明确也更好的分配人员高效并行的完成。

1) 项目实施前：

计划设计，针对老师分配的任务，根据任务的形式、难易程度、相关程度先对项目的需要完成什么样的人物有一个具体的轮廓，再结合根据之前的项目计划.mpp 进行结合每个人的工作范围和能力对每个人员具体分配本周详细计划任务也即本周的具体的项目的计划，其中考虑到每个子任务的需求、依赖性关系、各人能力进行拆分重组之后，每个人就专注完成自己的任务。此阶段设计项目计划，一般组长制定计划。

2) 项目实施后：

完成任务后，同时任务记录员根据每周具体的每个人的完成量记录成日志，与之前立的计划进行对比，了解出入在哪里，以便制定下周的工作详细计划可以更贴近实际完成工作。

2.2.2 处理方法

Method Specification		
Name	进度计划	
Brief Description	根据项目需求、人员专业指定既定项目计划，按计划实施项目工作，最终由记录员做出项目实施工作记录，并指导下一阶段的项目计划的确立。	
Precondition	开源网站中的 Redis 项目	
Primary Actor	组长和记录员	
Dependency	None	
Generalization	None	
Basic Flow	Steps	
	1	了解项目工作的具体实施需求要求
	2	了解每个成员的工作特点和特长、能力基础
	3	使用 MSProject 工具根据每个人的特点和工作要求做出项目计划表细致到小时
	4	按项目计划表实施项目
	5	管理员根据项目具体实施过程用 Word 书写工具记录项目具体实施情况
	6	将.mpp 计划表与工作日志做出对比，如果不契合的地方可以做出及时的调整
	Postcondition	项目计划表即项目计划.mpp 和工作日志

2.2.3 有效性评估

本问题处理方法的有效性体现在其计划制定是否合理，人员是否基本按照计划实施，项目是否能够控制，并且计划的完成是否高质量和高效，最重要的地方是与工作日志做出对比

首先：项目计划表的完成特别细致和精确，考虑到每一个人的特长和优势，兼顾项目的实施需求。

如图是.mpp 表截图：

	任务名称	实际开始时间	实际完成时间	完成百分比	计划完成百分比	实际工期	剩余工期	实际成本	实际工时	实际开始时间	实际完成时间	备注
1	上第一周课	2015年3月18日	2015年3月18日	100%	0%	8 工时	0 个工作日	¥0.00	12 工时	2015年3月18日	2015年3月18日	
2	组队	2015年3月18日	2015年3月18日	100%	OK	1 工时	0 工时	¥0.00	4 工时	2015年3月18日	2015年3月18日	
3	分配开源软件调研任务	2015年3月18日	2015年3月18日	100%	OK	2 工时	0 工时	¥0.00	8 工时	2015年3月18日	2015年3月18日	
4	查找资料	2015年3月19日	2015年3月20日	100%	OK	16 工时	0 个工作日	¥0.00	48 工时	2015年3月19日	2015年3月20日	
5	调研Python	2015年3月19日	2015年3月20日	100%	OK	16 工时	0 工时	¥0.00	16 工时	2015年3月19日	2015年3月20日	
6	调研Spark	2015年3月19日	2015年3月20日	100%	OK	16 工时	0 工时	¥0.00	16 工时	2015年3月19日	2015年3月20日	
7	调研Redis	2015年3月19日	2015年3月20日	100%	OK	16 工时	0 工时	¥0.00	16 工时	2015年3月19日	2015年3月20日	
8	小组第一次会议	2015年3月23日	2015年3月23日	100%	OK	4 工时	0 工时	¥0.00	16 工时	2015年3月23日	2015年3月23日	
9	制作ppt	2015年3月24日	2015年3月24日	100%	OK	2 工时	0 工时	¥0.00	2 工时	2015年3月24日	2015年3月24日	
10	上第二周课	2015年3月25日	2015年3月25日	100%	OK	8 工时	0 个工作日	¥0.00	6 工时	2015年3月25日	2015年3月25日	
11	课堂ppt演讲	2015年3月25日	2015年3月25日	100%	OK	2 工时	0 工时	¥0.00	2 工时	2015年3月25日	2015年3月25日	
12	答疑	2015年3月25日	2015年3月25日	100%	OK	0.5 工时	0 工时	¥0.00	2 工时	2015年3月25日	2015年3月25日	
13	整理评审意见	2015年3月25日	2015年3月25日	100%	OK	2 工时	0 工时	¥0.00	2 工时	2015年3月25日	2015年3月25日	
14	项目计划书	2015年3月26日	2015年3月26日	100%	OK	8 工时	0 个工作日	¥0.00	20 工时	2015年3月26日	2015年3月26日	
15	制作服务器模块的进度表	2015年3月26日	2015年3月26日	100%	OK	4 工时	0 工时	¥0.00	4 工时	2015年3月26日	2015年3月26日	
16	制作RDB持久化模块的讲	2015年3月26日	2015年3月26日	100%	OK	4 工时	0 工时	¥0.00	4 工时	2015年3月26日	2015年3月26日	

再次：工作的日志的完成，及时记录项目实施阶段每个人真实的工作情况。如图工作日志的截图：

工作日志

日期	工作记录
2015. 03. 20	小组讨论： 1. 各组员汇报项目调研情况：陈志伟 Python，林璐 Spark，王志鹏 Redis； 2. 通过讨论难度和规模，最后确定 Redis 为本次试验项目。
2015. 03. 23	1. 每个组员下载 Redis 3.0 源码，本地安装调试； 2. 搜集相关的技术文档与书籍共享到聊天群里。
2015. 03. 24	1. 每个组员初步学习 Redis； 2. 陈志伟总结 Redis 相关资料，初步形成 ppt；其他组员修改； 2. 林璐汇总资料，形成最终项目介绍 ppt。
2015. 03. 26	1. 陈志伟编写项目计划书； 2. 每个组员制作各自负责模块的进度表； 3. 陈志伟汇总并制作计划书。
2015. 03. 28	1. 陈志伟上传 RUCM、UML 和 RSA 相关资料与安装包； 2. 每个组员学习 RUCM 和 UML 知识； 3. 每个组员学习 RUCM 和 RSA 软件的使用。
2015. 03. 29	小组讨论： 1. 初步确定 Redis 的项目范围； 2. 确定模块分工。

最后：通过每一阶段的工作计划和工作日志的做出对比，发现计划的内容基本完成，则可以标志着进度计划的重要性和有效性。

2.3 工作量分析问题

2.3.1 问题描述

为期 17 周的实验中，有很多工作，不仅包括 8 个实验的制品，还包括附加的组织协调工作、会议、知识学习、记录、演讲等工作。此处的问题在于，如何量化各项工作，从而能公平、有依据地衡量组员工作，改进小组分工结构。

2.3.2 处理方法

Method Specification		
Name	工作量分析	
Brief Description	将实验过程涉及的工作进行分解量化	
Precondition	各项工作内容，工作过程的制品和管理记录	
Primary Actor	组长	
Secondary Actor	其他组员	
Dependency	None	
Generalization	None	
Basic Flow	Steps	
	1	在实验最初阶段，将预期工作进行大致估计和分解，包括：上课、小组会议、知识学习、文档撰写、需求分析、测试分析、网络评审、演讲、记录、项目管理等；
	2	各成员根据各项任务，结合自身情况，确定预计工时；
	3	在实验不同阶段，组长统计组员的实际工时和贡献，并利用 project 工具进行管理；具体衡量原则如 4-6 ：
	4	针对软件 进行基本分工，每个成员负责一个模块，此部分工作量基本比较均衡固定，且贯穿整个需求、测试实验；
	5	针对管理、记录、演讲等 日常事务，自愿原则分配，基本符合各成员特点，此部分工作量略有差异；
	6	针对实验 6-8 这类文档撰写等 杂项，组长事前针对具体工作进行分工，会因为实际情况略有调整和差异；
	7	每个实验阶段结束，组长发布工作统计表，注明每个成员的工作量，使各个成员及时了解自己的工作情况，以作督促；
	Postcondition	抽象工作进行细化和分解，用工时衡量工作量。

初始阶段对预期工作进行估计和分解的好处在于，实验的工作量基本可控，局限在于由于缺少经验和对成员的了解，可能出现偏差，但由于组长考虑比较周到，总的效果是好的。

每个阶段结束，工作统计表真实透明，组员可据此了解工作量，并调整下一阶段的工作量，有助于提高工作效率与积极性。

2.3.3 有效性评估

本问题处理方法的有效性体现在过程是否可控，成员是否积极承担工作。

实验过程的工作量基本控制在合理范围内。可以从中期的分析报告中看出对实验的控制过程：

	计划总工时	实际总工时	差值
陈志伟	123h	104h	19h
林璐	135h	115h	20h
王志鹏	135h	101h	34h
王珊珊	135h	107h	28h

同时，每个阶段根据工作量得到的各个成员工作统计，让成员及时了解自己在组内的水平，从而激发工作积极性。部分统计如图所示：

这种方法对于提高成员积极性很有作用，事实表明，一旦发现和别人差距较大，或者某项工作完成度不够，成员往往会在下次工作表现更为主动负责。

学号	姓名	实验准备阶段	需求分析阶段	实验 6~8	测试需求规格
SY1406108	陈志伟	调研 python (100%)、初步制作 ppt(25%)	软件项目计划书 (45%)、RUCM (25%)、规格说明书 (25%)	ppt 制作 (50%)、工作量分析报告 (100%)	撰写初稿 (25%)、修改测试需求规格说明书 (100%)
SY1406112	王珊珊	修改 ppt(25%)	软件项目计划书 (15%)、RUCM (25%)、规格说明书 (25%)	变更与管理分析报告 (90%)	撰写初稿 (25%)、修改测试需求规格说明书 (60%)
SY1406311	林璐	调研 Spark (100%)、ppt 修改及演讲 (50%)	软件项目计划书 (25%)、RUCM (25%)、规格说明书 (25%)	ppt 制作及演讲 (50%)、工作日志 (100%)	撰写初稿 (25%) 修改测试需求规格说明书 (60%)
SY1406117	王志鹏	调研 redis (80%)	软件项目计划书 (15%)、RUCM (25%)、规格说明书 (25%)	进度控制分析报告 (70%)	撰写初稿 (25%)、修改测试需求规格说明书 (60%)

2.4 配置管理分析问题

2.4.1 问题描述

在开发软件时变更是不可避免的，而变更加剧了项目软件人员之间的混乱，如果变更之前没有经过分析，变更实现之前没有进行记录，没有想需要知道变更的人员报告变更，或者没有提高质量和减少错误的方式控制变更，就会产生混乱。而配置管理是一种标识，组织和控制修改的技术。目的是使错误达到最小最有效的提高生产率。

在实验中，配置管理包括两部分：

- 1) 版本控制
- 2) 变更控制

2.4.2 处理方法

Method Specification		
Name	配置管理分析	
Brief Description	包括版本控制和变更控制分析	
Precondition	无	
Primary Actor	所有小组成员	
Dependency	Github	
Generalization	None	
Basic Flow	Steps	
	1	确定使用 github 作为变更管理工具
	2	确定使用 word 表格作为版本管理工具
	3	所有小组成员学习使用 github
	4	小组成员各自在文档头部表格内填入版本号及其他信息
	5	小组成员各自实验产出及时提交到 github
	6	负责人及时在 github 上将各成员的产出合并
	7	负责人将文档头部表格内的版本按顺序合并
	Postcondition	得到 github 的记录文档, 每个产出都带有纪录版本号的表格

2.4.3 有效性评估

下图是其中一个版本变更历史表格：

版本变更历史

版本	提交日期	编制人	说明
Version 1.0	2015-03-31	全体组员	初步完成需求规格说明书的第一版
Version 1.1	2015-04-01	陈志伟	完成 1.1 版本的修改
Version 1.2	2015-04-01	林璐	完成 1.2 版本的修改
Version 1.3	2015-04-01	王志鹏	完成 1.3 版本的修改
Version 1.4	2015-04-02	王珊珊	完成 1.4 版本的修改
Version 2.0	2015-04-04	全体组员	完成 2.0 正式本版的修改
Version 3.0	2015-04-10	陈志伟	完成 3.0 版本的修改
Version 3.1	2015-04-11	王志鹏	完成 3.1 版本的修改
Version 3.2	2015-04-11	林璐	完成 3.2 版本的修改
Version 3.3	2015-04-11	王珊珊	完成 3.3 版本的修改
Version4.0	2015-04-17	陈志伟	完成 4.0 版本的修改
Version4.1	2015-04-17	王志鹏	完成 4.1 版本的修改
Version4.2	2015-04-18	林璐	完成 4.2 版本的修改
Version4.3	2015-04-18	王珊珊	完成 4.3 版本的修改

表格中记录了每一个成员在完成自己负责的部分后的版本变更信息，包括版本号、修改人和时间等，方便了对历史版本的查看和追踪。

下图是 github 提交历史纪录：

94 commits		
	软工实验最终版提交 3 days ago by destiny2015	49 ▶
	11-更新工作日志到6.6 4 days ago by sswang1117	
	14 13 days ago by destiny2015	
	10-更新工作日志 16 days ago by sswang1117	
	10-评审意见提交更新至05.20 18 days ago by sswang1117	
	11 小组成员工作统计表 20 days ago by destiny2015	
	9-工作日志更新至5.18 21 days ago by sswang1117	
	9-添加测试分析报告2.0 和 实验8方案 22 days ago by unknown	2 ▶
	11 测试分析报告（正式版） 22 days ago by destiny2015	
	9-实验六方案 22 days ago by sswang1117	
	9-工作日志更新 22 days ago by sswang1117	2 ▶

每一次提交都有周号，也有提交人姓名。可以很容易地知道每个人做的工作，以及版本变更的历史，方便后续的分析。

总之，我们组的配置管理，包括版本管理和变更管理的结果和数据表明，该实验方法简单有效地完成了配置管理的目标。

2.5 需求分析问题

2.5.1 问题描述

在我们选一个好的开源项目以后，紧接着就要对该项目进行需求分析。通常来说，对一个软件进行需求分析，主要有四大任务：

- 1) 确定系统的总需求：功能需求、性能需求、可靠性和可用性需求、出错处理需求、接口需求、约束需求、逆向需求、将来可能提出来的需求（可扩展性）等。

2) 分析系统的数据要求：由于我们的软件系统一般都是对一系列数据或者信息进行处理，因此在软件开发的过程中，对系统运行过程中涉及到的数据进行分析也是很重要的。通常我们通过对数据进行建模来分析，即 E-R 图。

3) 导出系统的逻辑模型：在需求分析中，我们可以通过一系列的模型来导出系统的逻辑模型，以方便我们对系统有一个更加直观的了解。通常涉及到的模型有：功能模型、数据模型、行为模型、算法逻辑模型等。。

4) 修正系统的开发计划：在可行性分析阶段的最后我们制定了一个开发计划，在进行需求分析后，我们可以根据我们分析的结果对开发计划中不合理的部分进行修正。

考虑到我们是对已有软件进行需求分析，并且按照老师的要求，我们着重进行系统的功能需求和非功能性需求。

由于小组成员对该软件均不熟悉，并且对需求建模语言也是第一次接触，因此必须采取一定方法促进进展。

2.5.2 处理方法

Method Specification		
Name	需求分析	
Brief Description	对 reids 系统进行功能性和非功能性需求分析，并进行需求建模	
Precondition	选定研究课题为 redis 系统	
Primary Actor	所有组员，共同调研讨论	
Dependency	None	
Generalization	None	
Basic Flow	Steps	
	1	每个组员初步学习 redis
	2	小组讨论确定每人负责一个模块和一个非功能需求
	3	每个组员学习 RUCM 和 UML 知识
	4	每个组员学习使用 RUCM 和 RSA 软件
	5	每个小组成员深度学习各自负责的模块
	6	各自完成模块的 RUCM 和 UML 建模
	Postcondition	产出需求分析报告

2.5.3 有效性评估

需求分析的结果是需求分析报告，从需求分析报告中可以直观地看出实验方法的有效性。

首先，分模块这个方法，大大提升了小组学习 redis 系统的效率，每个小组成员并行工作，对各自的模块进行深度学习这个方法，缩短了需求分析的时间。

其次，小组成员共同学习 rucm 十分有效，从需求分析报告的 rucm 建模结果中可以看出，每个成员都对 rucm 有了一定的理解和认识，产出也令人满意。

在非功能性需求方面，小组成员各自选择自己感兴趣的方向进行分析，大大提升了分析效率。

总之，将需求拆分给每个成员的这种方法，不仅可以提高需求分析效率，同时也能保证需求分析的正确性。

2.6 测试与评审问题

2.6.1 问题描述

测试分析是软件工程中比较重要的一个部分，标志从需求规格的说明以及后续工作是否真正满足要求。在本项目我们选好需求规格分析说明以后，紧接着就要对项目做测试需求规格说明，对一个软件进行软件测试与评审问题，主要有四大任务：

1) 编写软件测试规格说明书：

A.编写测试计划：

测试计划的制定是要建立在充分理解需求的基础上的，当需求分析文档终稿的完成后，根据已有的需求分析文档终稿，分析指定日期和具体计划任务分配完成软件规格测试说明书。

B.设计测试用例：

根据以下要求来设计测试用例，每个人完成其对应的任务：

响应一个测试用例描述了针对某个目标对程序进行测试所采用的一组实际输入、程序执行条件、测试步骤和预期的输出，以核实某个程序或其中的特定路径是否满足特定需求。在需求分析文档确立基线以后，需编写项目的测试用例，具体的任务和责任人如下：

过程要点	详细说明
输入条件	测试需求明确，测试计划明确
工作内容	根据每一步测试计划编写全部的测试用例
退出标准	测试用例需要覆盖大部分测试需求
责任人	全体组员

2) 编写软件测试分析报告：

A.编写测试代码

根据测试用例来编写测试代码来作为具体的测试过程的实施依据。

B.测试的结果展示与分析

测试代码执行后，将测试的结果展示出来，并对测试的结果进行分析，此结果是否满足测试目标。

软件的测试分析阶段是项目实施中一个非常重要的阶段，很大程度检验的项目的前期工作是否完美的落实。

2.6.2 处理方法

Method Specification	
Name	测试与评审

Brief Description	对 Redis 进行设计软件测试用例，并且完成测试代码对软件进行功能性和非功能性测试分析	
Precondition	需求分析文档终稿的完成后	
Primary Actor	组内所有人	
Dependency	None	
Generalization	None	
Basic Flow	Steps	
	1	学习 RTCM 知识
	2	明确测试目标
	3	编写测试计划
	4	设计测试用例
	5	编写测试代码
	6	测试结果的展示与分析
	Postcondition	软件测试规格说明书和软件测试分析报告

2.6.3 有效性评估

测试分析的结果是软件测试分析报告，从软件测试分析报告中可以直观地看出软件测试用例和软件测试代码的有效性。

首先，小组成员学习 RTCM 建模方法十分有效，设计出满足测试目标的软件测试用例，并且能够对应上软件需求用例。

其次，在测试代码编写执行后，软件测试的结果基本符合测试目标的要求。根据结果分析也能够理解其中的原因。

在非功能性需求方面，小组成员完成了非功能性测试的要求。

总之，将测试分析在项目中起到了关键性的收尾检验性的功能，因此本组实验验证了软件测试用例的设计合理性。

3 参考文献

- [1] <http://redis.io/>
- [2] 黄健宏. Redis 设计与实现. 机械工业出版社[M]. 2014-06.
- [3] 肖丁、吴建林等. 软件工程模型与方法[M]. 北京邮电大学出版社. 2008-03.
- [4] <http://blog.csdn.net/renfufei/article/details/38474435>
- [5] 蔡建平. 软件测试实验指导教程[M]. 清华大学出版社. 2009-11.