

软件工程综合实验数据 分析报告

Redis

作者：全体组员

2015/6/6

1. 分析依据

“用数据说话”是科学研究的基本素养之一。数据有助于我们寻找到规律、得出结论，同时它也是最能使人信服的。数据的真实、准确性能够很好地提高科学探究的效率，因此，尊重数据是科学探究活动必须遵循的准则。

随着软件工程实验课程中 8 个实验的完成，我们在 github 中积累了大量的文档和工作过程中的数据。这些文档和数据详实的记录了小组在本学期的实验过程，由于 github 对每一个变更的详细记录，保证了我们获得的这些数据的真实性、有效性和可追踪性。对这些数据进行充分地分析可以得出小组成员本学期的工作状况和规律，为以后的软件工程综合实验设计提供依据和建议。

2. 分析对象及目标

本文档中将基于已获得的各种数据从软件进度计划与控制、工作量和变更与管理这三个方面来进行整个实验的综合分析。

进行软件进度计划与控制过程的分析，可以得到计划的完成情况及其与实际执行的差异，并找到对应的原因，并针对这些原因及时做出相应的调整，保证整个项目在可控、定量的情况下有序的完成，同时也为以后制定项目计划提供经验。进行项目计划可以对整个项目起到一个计划、安排、监管的作用，也是进度的控制的手段，它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。

进行工作量分析可以得出小组成员在实验中完成任务的种类和质量，以便在后来的工作中改善项目组人员分配，提高工作效率，合理利用人力资源。工作量是小组每个成员的对小组的贡献的最客观反映。工作量在我们的实验中可以用工时来衡量，但并不是每个工作都能用工时进行精确的量化，所以在计算工作量时还需进行合理的修正。

从变更与管理分析中可以看出小组在每次实验中所提交文档的数量、所更新文档的频率及小组任务完成的进度。采用有效方法来记录变更和版本，根据版本的变化来和变更的进行来分析软件研发的进度和预期进行对比，以便调整软件开发的进度，实行变更控制和版本管理。

3. 分析要点

进行实验数据的分析，需要注意以下三点：

（1）数据整理

一般直接得到的原始数据并不满足数据分析的条件，需要从中提取出相关数据，剔除无关的数据，补充缺失的数据。

（2）数据统计

进行大量数据的统计分析时，需要重点关注数据的特征，如总量、方差、均值、最大/小值等，这些最基本的数据特征能反映事物的一般规律。若数据量足够大，还能得到数据的分布、置信度、相关性等特征，它们能从更深的角度反映事物的内涵。

（3）分析结论

对数据进行统计分析之后，需要对得出的数据特点和特征进行直观的解释，找到对实验影响显著的因素，这个过程并不容易。最后，还要根据这些结论和影响因素提出中肯的启示和建议。

4. 分析内容

4.1. 软件进度计划与控制

4.1.1. 输入数据

（1）MS Project 项目计划文件

	任务名称	实际开始时间	实际完成时间	完成百分比	实际完成百分比	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
1	上第一周课	年3月18日	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	年3月18日	年3月18日	100%	0%	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%	0%	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	年3月19日	年3月20日	100%	0%	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%	0%	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%	0%	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%	0%	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%	0%	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	年3月24日	年3月24日	100%	0%	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	年3月25日	年3月25日	100%	0%	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%	0%	2 工时	0 工时	¥0.00	2 工时	2015年3月25日	2015年3月25日
12	答辩	年3月25日	年3月25日	100%	0%	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%	0%	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%	0%	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%	0%	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持久化模块的进度表	2015年3月26日	2015年3月26日	100%	0%	4 工时	0 工时	¥0.00	4 工时	2015年3月26日	2015年3月26日
17	制作客户端模块的进度表	2015年3月26日	2015年3月26日	100%	0%	4 工时	0 工时	¥0.00	4 工时	2015年3月26日	2015年3月26日

(2) 工作日志

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。

4.1.2. 分析工具

MS Project 和 word 软件

4.1.3. 分析过程

MS Project 文件不仅可以进行项目的计划，还可以跟踪、记录项目的实际完成过程。在分析过程中，我们从项目的开始/结束时间、任务种类、工时等角度把项目的实际完成记录与项目计划做对比，可以发现一些非常明显的差异。然后，在工作日志中查找这些具有差异的任务具体工作内容，并结合备注信息，得到造成这些差异的原因，以对后面的项目计划做出调整。

在对 MS Project 文件和项目日志文件进行分析的过程中，经过反复思考也没有找到一种好的方式能对这些数据进行提取和处理，因为 MS Project 文件中包含的对其有影响的因素太多，也没有显著的规律，这也说明了进行软件进度计划 and 控制的复杂性。所以，我们最后选择了一种直接的对比分析，得到了一些好的结论。

4.1.4. 启示与结论

通过分析过程可以得到如下的结论：

- (1) 实验中除了部分任务没有完成，如需求分析网络的第二次互评审，其它的任务都是按照软件进度计划来严格完成的。但也由于一些任务在实际执行中被老师调整了，如撰写产品更新与展示需求分析报告、测试分析报告，使我们的最终完成时间与计划的最终时间有差异，但若除去这两个阶段所占用的时间，计划和实际时间将符合的极好，这也是我们在实验过程中不断对开发进度计划进行完善的结果。
- (2) 工作日志是对项目计划的良好补充，在工作日志中我们记录每个时间对应的工作内容，方便对以往的工作有一个统一和清晰的认识
- (3) 良好的软件进度计划离不开项目管理人员对整个项目的周期完整了解。管理人员对任务的分解一定要具体明确，方便相关人员的执行。项目任务的分解是基础，分解没有执行好，会对后面的进度计划以及人员分配等造成很大影响。
- (4) 项目计划与实际出现差异怎么办？一般可采取如下措施：
 - 1) 加强团队成员的相互监督力度；
 - 2) 加强团队沟通交流的次数，面对面对团队成员可能存在的疑问进行探讨；

- 3) 调整后续的任务计划, 注意计划的前后依赖关系, 特别对于任务时间的确定, 要注意节假日等一些特殊的日子;
- 4) 在每个大任务之间, 应该留下一定的空隙时间, 以应对突发事件的发生; 在计划后, 要反复斟酌、修改计划, 送给其他有经验的人把关。

4.2. 工作量

4.2.1. 输入数据

(1) MS Project 项目计划文件

同 2.1.1

(2) 小组成员工作统计表

2015-小组成员工作统计表

学号	姓名	实验准备阶段 (10%)	需求分析阶段 (20%)	实验 6 ~ 8 (20%)	测试需求规格 (20%)	测试阶段 (20%)	软件产品展示 (10%)
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%)	测试分析报告 (100%) 实验 7 方案 (100%)	Redis 安装(25%)
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%)	测试分析报告 (40%)、实验 6 方案 (0%)	AOF 模块 (25%)
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%)	测试分析报告 (100%) 实验 6~8 (100%)	RDB 模块 (25%)
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%)	测试分析报告 (40%)、实验 8 方案 (0%)	客户端模块 (25%)

4.2.2. 分析工具

MS Project、Matlab 和 word 软件

4.2.3. 分析过程

(1) 工时分析

基于 MS Project 中的项目的实际完成记录，可以得到在整个实验中每个小组成员的实际工时，如下表所示：

工时 \ 姓名 任务	陈志伟	林璐	王珊珊	王志鹏
A.上课	30h	30h	30h	30h
B.小组会议	12h	12h	12h	12h
C.知识学习	40h	40h	40h	40h
D.RUCM 需求建模	10h	14h	14h	14h
E.实验 6~8	16h	16h	14h	12h
F.软件测试	40h	40h	40h	40h
G.文档撰写	45h	43h	41h	40h
H.评审	4h	4h	4h	4h
I.演讲	0h	16h	0h	2h
J.记录	0h	0h	9h	0h
K.项目管理	9h	0h	0h	0h
合计	206h	215	204	194

(2) 工作量分析

在小组成员工作统计表中，我们把每个成员的工作分为两部分，分别为基准工作和额外工作。基准工作指实验中必须完成的任务，与小组成员工作统计表对应，由小组基准工作量统计表来记录。额外工作指为了维护小组任务的正常完成而做的工作，由额外工作量统计表来记录，每个小组成员做的额外工作都不同。

小组基准工作统计表如下所示：

2015 小组基准工作量统计表

学号	姓名	实验准备阶段 (10%)	需求分析阶段 (20%)	实验 6~8 (20%)	测试需求规格 (20%)	测试阶段 (20%)	软件产品展示 (10%)	总计
SY1406108	陈志伟	0.28	0.25	0.30	0.28	0.40	0.25	0.29
SY1406112	王珊珊	0.19	0.25	0.23	0.24	0.10	0.25	0.21
SY1406311	林璐	0.34	0.25	0.30	0.24	0.40	0.25	0.30
SY1406117	王志鹏	0.19	0.25	0.17	0.24	0.10	0.25	0.20

小组额外工作量统计表如下所示：

2015 额外工作量统计表

学号	姓名	演讲	记录	项目管理	合计	工作量
SY1406108	陈志伟	0h	0h	6h + 3h	9h	0.25
SY1406112	王珊珊	0h	6h + 3h	0h	9h	0.25
SY1406311	林璐	12h + 4h	0h	0h	16h	0.44
SY1406117	王志鹏	2h	0h	0h	2h	0.06

注：工时 = 前 7 周 + 后 6 周

取基准工作量占总贡献的 85%，额外工作量占 15%，加权平均后每个成员的工作量如下学期工作量总结表所示：

2015 年学期工作量总结表

学号	姓名	基准工作量(85%)	额外工作量(15%)	总贡献率
SY1406108	陈志伟	0.29	0.25	0.284
SY1406112	王珊珊	0.21	0.25	0.216
SY1406311	林璐	0.30	0.44	0.321
SY1406117	王志鹏	0.20	0.06	0.179

4.2.4. 启示与结论

由上述分析过程可得到如下结论：

- （1）在工时分析中，可以看到每个人的总工时都是不同的，这主要和每个人的额外工作量有关。同时由工时的大小可以间接得到小组成员的工作量的排序，即林璐 > 陈志伟 > 王珊珊 > 王志鹏，但工时只能衡量每个人花费多长时间，并没有考虑工作效率和产出物的质量，所以此方法不够科学。
- （2）工作量分析中针对每一阶段的任务，不仅考虑到成员完成任务的多少，还考虑到了成员完成的文档的质量等因素，所以在给出工作量的数据时避免了工时分析中考虑因素单一的缺点。最后由学期工作量总结表中可以得到小组成员的总工作量，即为总贡献率。这种方式也有不足之处，主要是对于每一阶段的工作量由人来给出有一定的主观性，无法做到绝对的公平，但结合工时分析可知二者的结论一样，故我们给出的工作量的估计值还是相对较为公平的。
- （3）对比工时分析每个人的合计工时和工作量分析中的每个人的贡献率可知，在工时分析中成员之间的工时差距不大，即极差为 $(215-194)/215 = 10\%$ ，而在工作量分析中却相差极大，即极差为 $(0.312-0.175)/0.312 = 44\%$ 。这主要和小组的奖惩措施有关，即若未完成任务或完成任务的质量太差，会从成员的

基准工作量 0.25 中减去一定的数值，而这个多出来的工作量会加到其他成员的工作量上，这样一减一加就很容易把成员之间的工作量拉开差距。

4.3. 变更与管理

4.3.1. 输入数据

(1) 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

(2) GitHub 提交历史

92 commits		
	14 7 days ago by destiny2015	
	10-更新工作日志 11 days ago by sswang1117	
	10-评审意见提交更新至05.20 13 days ago by sswang1117	
	11 小组成员工作统计表 15 days ago by destiny2015	
	9-工作日志更新至5.18 16 days ago by sswang1117	
	9-添加测试分析报告2.0 和 实验8方案 16 days ago by unknown	2 ▶
	11 测试分析报告（正式版） 16 days ago by destiny2015	
	9-实验六方案 16 days ago by sswang1117	
	9-工作日志更新 16 days ago by sswang1117	2 ▶

4.3.2. 分析工具

Github 和 word 软件

4.3.3. 分析过程

在进行版本变更分析时，我们需要查看每一个项目文档头部加入的版本变更历史表格。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。

Github 是优秀的版本控制与管理工具，在分析小组成员的提交历史时，可以以文件的前缀——周号来分组查看，这样可以直接得到每个小组成员每周的工作状况，十分方便和快捷。同时，由于多人协作难免不同小组成员对同一文件的修改而造成的冲突，这时需要手动更改冲突的部分，完成小组成员工作内容的合并。

4.3.4. 启示与结论

由变更与管理的分析我们可以得到如下结论：

- (1) 项目文档的版本变更历史主要是用来记录文档的变更过程，使用它可以告诉下一个编辑者和用户，谁修改了这个文档，修改了文档的哪些地方等，起到了一种间接沟通的作用。
- (2) 从 **Github** 的提交历史可以看出小组成员对工作任务的提交并不总是那么连续，在每周中总有同学没有按时提交而是在后面补交，这主要是因为小组内部使用的是 QQ 群进行的交流，大家完成自己负责的任务后会把文档等在第一时间提交到 QQ 群中，而疏忽了 **Github** 上文档的提交。这说明了大家对 **Github** 的重视不够，而非没有完成任务。
- (3) 变更与管理不仅仅是记录一些文档的变更过程，更重要的在于它起到了很好的监督的作用，只有成员自己编写、修改了文档，我们才会承认其完成了文档版本的一次更新。但需要特别注意只是填写了变更历史，而没有进行任何实质性的文档编写与修改的行为，这也很容易通过分析前后两个文档的版本差异来发现。