Hadoop-MapReduce

实验方法总结报告

2015 年 6 月

1. 说明

本学期实验课的目标是学习和综合运用多种软件工程方法从事相应的软件工程活动，其中需要对软件工程活动中出现的实际问题进行合理有效的处置，通过实验对方法的实施效果做出客观的分析和评判。通过对真实的开源软件（本小组选择的开源软件为 Hadoop-MapReduce）进行需求分析和测试分析，体验并熟练掌握软件需求分析和依据软件需求进行软件测试设计的有效方法。

实验的主要内容包括：分析和理解软件；获取并严谨定义软件需求（包括功能、安全性、并行能力等需求）；细化需求，采用预定的规范和方法，严谨地定义需求；对需求进行评审和检验；依据软件需求确定测试需求，据此进行软件测试设计；把测试设计实现为可操作的测试等。在实验中，我们使用 RUCM 和 RTCM 对软件需求和测试需求进行建模。

通过本学期的实验课程的学习，我们小组完成了任务书、评审单、需求规格说明书、复评审问题处理报告、测试需求规格说明书、测试报告，工作日志、进度控制分析报告、变更与管理分析报告、工作量分析报告。

在整个实验过程中，本小组采用了一些方法辅助完成实验。现介绍如下。

2. 软件需求分析阶段

为了确定 Hadoop-MapReduce 的需求，我们首先想到 MapReduce 的 3 个主要的使用场景：

1.用户提交作业；

2.系统处理作业；

3.系统出现异常，做异常处理。

为了写需求用例，我们小组分工分别阅读了 Hadoop-MapReduce 的相关源代码和网络资料，最终根据以上 3 个场景画出了用例图。为了规格化的描述用例，我们使用了 RUCM。对每个用例，分析其前置条件、后置条件、以及它对系统或外部环境的状态进行什么样的修改。对每个用例中的主流程和备选流程，使用规范严谨的句法进行描述，为测试提供准确的设计和判定依据。

在用例图中，关于 Actor 的选择是个重点也是难点。本组在做软件需求分析时，对 Actor 进行过几次修改，修改依据是评审课堂上，老师和同学们的建议。

3. 测试需求分析阶段

为了对 Hadoop-MapReduce 的需求进行测试，我们将测试需求与软件需求进行对照，更加直观地表达我们想要测试的内容。

对每一个需求用例，参照其 Precondition 编写测试需求的前置条件，参照其 Basic Flow 编写测试需求的主要流程，参照其 Post Condition 编写测试需求的后置条件，判定需求和测试是否对应。对于 Alternative Flow 中的异常处理内容，编写另一个测试需求对其进行覆盖。在 RTCM 中，测试过程要对程序运行过程中状态的变化进行检测，对于程序运行生成的文件等结果，需要测试者进行检查。

在经过评审之后，按照老师和同学们的意见，为体现 MapReduce 的特性，我们新增了并行能力处理测试用例，并与单机处理程序进行对比。

4. 测试阶段

在测试阶段，我们按照测试需求规格说明书中的测试用例来做测试，防止遗漏。

Hadoop-MapReduce 使用 Java 语言实现，因此我们主要使用 Java 来实现测试规格。为顺利进行测试，我们搭建了 Hadoop 集群来进行大数据处理，包括 1 个主节

点，2 个从节点，来增强集群的并行处理能力。对于一些整体功能上的测试，比如配置测试、处理 Mapper 任务测试等，我们编写脚本程序进行实现。对于测试策略和测试实现的具体描述在测试规格描述中有详细介绍。

在测试 MapReduce 的并行计算能力时，我们使用 KNN 算法在 MapReduce 和单机上运行，得到的结果出乎意料，显示单机处理作业时，时间要更短一些。为了找到原因，我们做了几点猜测，比如网络带宽不够，限制了 MapReduce 的处理速度；数据集不够大，难以体现 MapReduce 的处理优势；从节点只有 2 台，太少，并不是真正的 Hadoop 集群。

为科学地给出一个可能原因，我们对猜测中的数据集大小进行了测试。我们在 Hadoop 集群和单机上，使用 KNN 算法，分别对 78M、242M、484M 的数据集进行处理，发现当数据集增大时，MapReduce 的处理时间会减少。

由于实验室条件限制，诸如增加从节点的个数，提高网速等办法，暂时无法检验。

5. 管理工具

本实验课中，我们使用的管理工具是 Github 和 Microsoft Project。前者用于版本管理，小组产生的所有的提交都在 Github 里面有据可查。每次提交或更新都有记录，便于版本管理和工作认定。

Microsoft Project 主要用于项目进度计划和标注实际项目进度。需求分析前期，使用 Project 做好项目进度计划，将每个操作步骤细化，将工作细化到个人，将耗时细化到小时，增强实验中每一步的可操作性。后期实验过程中，按照项目计划来完成实验，保证项目的进度。

Github 和 Microsoft Project 这两个工具非常实用，能给项目实施和管理带来很大的便利。