很多人都在谈大数据技术，但客户在真正启动大数据项目的时候有哪些技术思考？ 如何来选型大数据平台？从以下POC考题及案例就可以了解客户对大数据的细粒度需求。看了考题，你能全部做下来吗？

[1       概述..........................................................................................................................4](http://blog.csdn.net/tonygds/article/details/51993334#_Toc426643251)

[1.1         测试时间、地点、人员.....................................................................................4](http://blog.csdn.net/tonygds/article/details/51993334#_Toc426643252)

[1.1.1          测试时间...............................................................................................4](http://blog.csdn.net/tonygds/article/details/51993334#_Toc426643253)

[1.1.2          测试地点...............................................................................................4](http://blog.csdn.net/tonygds/article/details/51993334#_Toc426643254)

[1.1.3          测试人员...............................................................................................4](http://blog.csdn.net/tonygds/article/details/51993334#_Toc426643255)

[1.2         测试环境.........................................................................................................4](http://blog.csdn.net/tonygds/article/details/51993334#_Toc426643256)

[1.2.1          测试组网图...........................................................................................4](http://blog.csdn.net/tonygds/article/details/51993334#_Toc426643257)

[1.2.2          测试软硬件列表.....................................................................................5](http://blog.csdn.net/tonygds/article/details/51993334#_Toc426643258)

[2       服务层、应用开发层测试............................................................................................5](http://blog.csdn.net/tonygds/article/details/51993334#_Toc426643259)

[2.1         服务框架部分..................................................................................................5](http://blog.csdn.net/tonygds/article/details/51993334#_Toc426643260)

[3       基础组件测试——模拟场景部分.................................................................................10](http://blog.csdn.net/tonygds/article/details/51993334#_Toc426643261)

[3.1         结构化数据加载场景.......................................................................................10](http://blog.csdn.net/tonygds/article/details/51993334#_Toc426643262)

[3.2         结构化数据批量场景.......................................................................................11](http://blog.csdn.net/tonygds/article/details/51993334#_Toc426643263)

[3.3         结构化数据查询场景.......................................................................................11](http://blog.csdn.net/tonygds/article/details/51993334#_Toc426643264)

[3.3.1          结构化数据简单查询............................................................................11](http://blog.csdn.net/tonygds/article/details/51993334#_Toc426643265)

[3.3.2          结构化数据复杂查询............................................................................11](http://blog.csdn.net/tonygds/article/details/51993334#_Toc426643266)

[3.4         非结构化数据加载场景——小文件...................................................................12](http://blog.csdn.net/tonygds/article/details/51993334#_Toc426643267)

[3.5         非结构化数据查询场景——小文件...................................................................12](http://blog.csdn.net/tonygds/article/details/51993334#_Toc426643268)

[3.6         非结构化数据加载场景——大文件...................................................................13](http://blog.csdn.net/tonygds/article/details/51993334#_Toc426643269)

[3.7         非结构化数据查询场景——大文件...................................................................13](http://blog.csdn.net/tonygds/article/details/51993334#_Toc426643270)

[3.8         半结构化数据加载场景...................................................................................14](http://blog.csdn.net/tonygds/article/details/51993334#_Toc426643271)

[3.9         半结构化数据查询场景...................................................................................14](http://blog.csdn.net/tonygds/article/details/51993334#_Toc426643272)

[4       基础组件测试——组件功能性部分.............................................................................15](http://blog.csdn.net/tonygds/article/details/51993334#_Toc426643273)

[4.1         大数据应用开发平台各组件功能测试...............................................................15](http://blog.csdn.net/tonygds/article/details/51993334#_Toc426643274)

[4.1.1          HDFS...................................................................................................15](http://blog.csdn.net/tonygds/article/details/51993334#_Toc426643275)

[4.1.2          HBase..................................................................................................20](http://blog.csdn.net/tonygds/article/details/51993334#_Toc426643276)

[4.1.3          MapReduce&YARN.................................................................................25](http://blog.csdn.net/tonygds/article/details/51993334#_Toc426643277)

[4.1.4          Hive....................................................................................................28](http://blog.csdn.net/tonygds/article/details/51993334#_Toc426643278)

[4.1.5          Impala.................................................................................................31](http://blog.csdn.net/tonygds/article/details/51993334#_Toc426643279)

[4.1.6          Storm..................................................................................................35](http://blog.csdn.net/tonygds/article/details/51993334#_Toc426643280)

[4.1.7          Spark...................................................................................................35](http://blog.csdn.net/tonygds/article/details/51993334#_Toc426643281)

[4.1.8          Sqoop..................................................................................................39](http://blog.csdn.net/tonygds/article/details/51993334#_Toc426643282)

[4.1.9          Kerberos...............................................................................................40](http://blog.csdn.net/tonygds/article/details/51993334#_Toc426643283)

[4.1.10       Flume..................................................................................................40](http://blog.csdn.net/tonygds/article/details/51993334#_Toc426643284)

[4.1.11       Solr.....................................................................................................41](http://blog.csdn.net/tonygds/article/details/51993334#_Toc426643285)

[5       管理能力测试...........................................................................................................42](http://blog.csdn.net/tonygds/article/details/51993334#_Toc426643286)

[5.1         安装部署.......................................................................................................42](http://blog.csdn.net/tonygds/article/details/51993334#_Toc426643287)

[5.2         集群统一管理工具..........................................................................................43](http://blog.csdn.net/tonygds/article/details/51993334#_Toc426643288)

[5.3         日志管理.......................................................................................................43](http://blog.csdn.net/tonygds/article/details/51993334#_Toc426643289)

[5.4         性能监测.......................................................................................................44](http://blog.csdn.net/tonygds/article/details/51993334#_Toc426643290)

[5.5         故障管理.......................................................................................................45](http://blog.csdn.net/tonygds/article/details/51993334#_Toc426643291)

[5.6         数据管理能力.................................................................................................46](http://blog.csdn.net/tonygds/article/details/51993334#_Toc426643292)

[5.7         角色管理能力.................................................................................................47](http://blog.csdn.net/tonygds/article/details/51993334#_Toc426643293)

[5.8         多集群部署能力.............................................................................................47](http://blog.csdn.net/tonygds/article/details/51993334#_Toc426643294)

1             概述

1.1        测试时间、地点、人员

**1.1.1  测试时间**

初步拟定POC测试时间为xxxx, 将分批次安排意向厂商入场进行测试工作。

**1.1.2  测试地点**

某银行总行信息科技部开发中心

**1.1.3  测试人员**

|  |  |
| --- | --- |
| 单位 | 人员 |
| 银行信息科技部 | 大数据应用开发平台项目成员 |
| 意向厂商 | POC测试人员 |

1.2        测试环境

1.2.1     测试组网图

各节点安装服务参见<表1.1.1>

表1.1.1

|  |  |  |
| --- | --- | --- |
| 主机名 | ip | 服务 |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

1.2.2     测试软硬件列表

测试软件及支撑软件参见<表1.1.2.1>

表1.1.2.1

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 编号 | 软件名称 | 版本 | 数量 | 备注 |
| 1 |  |  |  |  |
| 2 |  |  |  |  |

测试硬件清单参见<表1.1.2.2>表1.1.2.2

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 编号 | 硬件名称 | 规格型号 | 数量 | 备注 |
| 1 |  |  |  |  |
| 2 |  |  |  |  |
| 3 |  |  |  |  |
|  |  |  |  |  |

2             服务层、应用开发层测试

2.1        服务框架部分

服务框架部分包括两部分验证，其一是面向开发者，验证平台所提供的开发工具及相关的服务是否充分满足应用开发的需要；其二是在平台处于运行时状态下，服务层能否实现架构设计的预期效果。

**2.1.1组件功能说明**

|  |  |  |  |
| --- | --- | --- | --- |
| 架构层次 | 编号 | 技术组件 | 是否必选 |
| 应用开发层 | 1 | CLT命令行工具 | **必选** |
| 2 | 管理控制台 | 可选 |
| 3 | SDK | **必选** |
| 4 | IDE优化 | 可选 |
| 5 | 数据上传下载工具 | 可选 |
| 服务层 | 6 | REST API | **必选** |
| 7 | 数据路由 | **必选** |
| 8 | 同步异步转换 | 可选 |
| 9 | API接口封装 | **必选** |
| 10 | 业务逻辑容器（UDF/MR） | 可选 |
| 安全管理 | 11 | 身份认证 | **必选** |
| 12 | 权限管理 | **必选** |
| 元数据管理 | 13 | 元数据管理 | **必选** |

**2.1.2设计能力考察**

提交物要求：1、以word形式提供具体的设计方案；

**2.1.2.1应用开发层**

**SDK**

SDK是对RESTfulAPI的封装，提供更高层次的抽象，便于用户理解和使用平台的服务。

n  设计要求

1、    SDK的基本框架，包、类的整体结构设计、对于重点类的设计，请举例一二。

2、    SDK与业界同类产品、系统（CDAP、ODPS SDK）上的差异和相应考虑？

3、    SDK是平台与业务应用的接口，SDK设计时如何考虑对业务应用的支持？

4、    SDK与REST API如何权衡两者的关系，设计上考虑的侧重点分别是什么？

**CLT命令行工具**

CLT是面向开发者的客户端工具，开发者通过命令行的形式提交各种指令，与平台进行交互。其支持的指令集包括权限管理、资源（脚本\Jar包等）上传、结构化数据操作等DML、DDL。

命令行工具是开发者最主要的工具，其功能应覆盖开发过程中的各主要环节，对开发全过程提供完整支持。

n 设计要求

命令行支持范围、分类、设计思路、实现方案。

**管理控制台**

管理控制台是对CLT的补充，其功能是后者功能集的子集，提供更直观的图形化界面，便于非专业开发人员操作。此处的管理控制台以应用开发者和管理人员为对象，主要服务于开发过程中的管理。提供权限控制、用户管理、日志管理等功能，交互友好，便于非专业开发人员使用。

n  要求

管理控制台支持的操作是命令行的子集，选择的策略。管理控制台本身的作为子系统的架构设计方案。

**数据上传下载工具**

在开发过程中使用的命令行工具，简化测试数据的上传下载操作。

n  设计要求

介绍相应的设计方案。

**IDE优化**

基于自主研发平台集成相应的插件，支持java代码的开发、调试等。

n  设计要求

与我行Java自主研发平台充分结合，如何设计以优化开发过程，提升效率。

**2.1.2.2服务层**

**REST API**

服务层以统一的RESTful API方式对外提供服务，通过SDK方式进行封装，进而形成不同的客户端组件，便于开发人员使用。

REST API作为对外暴露的接口供外部应用调用，在平台的全生命周期内要保证稳定性和向后兼容性，需要保证接口的良好设计。API不只是对原生API一对一的封装，还要兼顾异构组件标准化和API增强的目标（详见高阶架构相应章节描述）。

n  设计要求

1、  整体API层次结构的大致结构和设计思想

2、  对异构组件兼容性考虑、API增强考虑

**数据路由**

基于元数据的管理，根据访问数据对象的不同，将相应服务路由到不同的计算框架或不同的集群。

n  设计要求

1、如何保证服务层元数据信息与各集群元数据信息的一致性？

2、如何保证数据路由的高效和高可靠？

3、如何实现数据对象的解析，保证解析高效？

**同步异步转换**

支持同步及异步调度，异步调度主要考虑对于延时较长的任务，避免服务阻塞。

n  设计要求

平台的服务如何针对客户端调用和其他应用调用进行封装？同步、异步服务的设计原则，异步调度的实现方式、异步转同步封装的实现方式。

**API接口封装**

对内部技术组件API的适配，保证向上提供相对统一的服务API。

n  设计要求

评估API适配设计好坏要关注有哪些方面？是否会带来效率的损失？

**业务逻辑容器（UDF/MR）**

用户可以编写UDF（自定义）函数及MapReduce代码，托管到平台的相应层次（服务层或计算层），实现一定的业务逻辑。

n  设计要求

1、 在多组件支持UDF的情况下，对UDF的定义方式能否做到与计算组件无关？

2、 在计算层组件不支持的情况下（如Impala），是否有其他方式支持有限的UDF定义？在设计上的关注点是什么？其局限性有哪些？

3、  平台的服务如何针对客户端调用和其他应用调用进行封装？同步、异步服务的设计原则，异步调度的实现方式、异步转同步封装的实现方式。

**2.1.3实施能力考察**

提交物要求：具备落地实施能力的厂商，提交相应的源代码、目标码以供验证；

**2.1.3.1应用开发层**

应用开发层是对开发过程的支持，因此对该层用例并不以考察最终运行态的各项指标为重点（但将同步结合技术能力的要求进行考察），而是考察在开发环境下相应的工具是否满足开发者高效、便捷的工作需要，是否能够支撑完整的工作流程。

**创建项目空间、用户及赋权**

“项目空间”对应平台上的每一个具体应用，同时也是“多租户”中的“租户”概念。通过平台层面的管理功能，创建“项目空间”及相应的“项目管理员”角色，切换至“项目管理员”完成为其他用户的创建、赋权等操作。

n  考察用例

基于CLT工具创建开发人员账户，并赋予相应权限。

n  相关技术组件

CLT（必选）或管理控制台（可选）

**创建表、加载基础数据**

通过CLT完成数据表创建，该表创建语句应符合通用SQL或HiveQL语法规范。表创建语法应考虑后续使用方式：一、是否要求复杂关联查询或单表低延迟查询等等；二、考虑相应的多集群、多存储技术（HDFS/Hbase）的因素。

n  考察用例

1、  使用开发人员账户登录CLT，创建数据表Table\_A。

2、  使用“数据传输工具”或“管理控制台”将本地测试数据文件（文件大小不超过50M）上传并加载至数据表Table\_A中。

3、  使用CLT，通过查询语句，检查数据加载正确。

n  相关技术组件：

CLT（必选）、数据上传下载工具（可选）、管理控制台（可选）

**代码开发、编译、部署及调试**

n  考察用例

1、  基于IDE完成工程的创建、基于SDK完成代码开发并在本地完成编译打包。IDE环境优先考虑与我行JAVA自主研发平台整合。代码部分按照技术能力部分相应场景要求编写。

2、  通过CLT完成Jar包的上传和部署。Jar包上传后作为“项目空间”下的资源，相应权限在该空间下统一管理。

3、  通过CLT命令启动相关任务或调用方法，完成程序运行调试。

4、  通过CLT或管理控制台（可选）查询相应数据库表、日志信息确认程序运行结果。

n  相关技术组件：

IDE（可选）、SDK（必选）、CLT（必选）、管理控制台（可选）

**数据共享**

项目间的数据共享进行统一权限控制，在未授权的情况下，“项目空间”内的数据文件及其他资源对其他“项目空间”均不可见，在授权情况下可不同程度（读、写等权限分离）的共享使用。

n  考察用例

1、  通过CLT或管理控制台（可选）对数据表、数据文件（非结构化）的访问权限进行管理。

2、  其他项目空间下的授权用户可使用该共享数据，而未授权用户无法查看相应数据。

n  相关技术组件：

CLT（必选）、管理控制台（可选）

**2.1.3.2服务层**

**REST API**

n  考察用例

通过浏览器或模拟器验证相关REST API接口。

**数据路由**

n  用例

1、执行不同的查询指令，通过日志跟踪数据路由过程，通过变更服务层元数据内容检验对数据路由的影响。

2、变更集群的数据文件或其他数据内容，检验服务层元数据是否做到一致性维护。

**同步异步转换**

n  考察用例

模拟外部应用调用平台服务，通过日志跟踪及相关管理界面等技术方式验证同步、异步执行及转换情况。

**API接口封装**

n  考察用例

无

**业务逻辑容器**

n  考察用例

将指定Java类部署至平台，通过模拟器或CLT等方式调用相应方法，确认执行代码执行正确性。

3             基础组件测试——模拟场景部分

**测试种子数据：**本测试章节中的种子数据均取自数据仓库中的集市，代码类表为截止到T日的全量数据，事实类表将根据情况选择部分数据。出于信息安全考虑，POC所用测试数据将以固定分隔符（Tab）分隔的每行一条记录的文本文件形式提供，文件内容为脱密的生产数据。数据文件名称同源表名称，详细文件清单参见附录一。

3.1        结构化数据加载场景

|  |  |
| --- | --- |
| **测试编号** |  |
| **测试项目** | 结构化数据加载场景 |
| **测试目的** | 测试大数据应用开发平台对存量和增量文件加载处理能力。监控该平台加载过程中的资源利用情况。 |
| **测试要求** | 1)         加载的文件为统一提供的测试数据文件，源文件格式不能改变。  2)         厂商根据银行提供的目标表结构，提供相应的目标表建表语句，创建目标表，目标表结构不能改变。  3)  从装载启动到装载结束，不能进行系统参数调整等影响装载执行过程的行为。 |
| **预置条件** | 硬软件环境搭建完毕，种子数据的数据文件已生成 |
| **测试步骤** | 1、银行人员将种子数据文件放到指定文件目录，执行导入脚本，将数据文件导入到数据库中。  2、记录数据导入时间，同时收集CPU和I/O资源消耗等信息。  3、统计数据库可用空间大小、每个表的占用空间大小。  4、检查每个表的记录数目，对金额等字段的正确性进行检查（各家厂商结果比对）。 |
| **预期结果** | 1．  数据文件全部推送到hfds对应目录。数量、内容正确。  2．  数据文件同时备份到接口机规定目录下。数量、内容正确。 |
| **评分依据** | 衡量大数据应用开发平台对结构化数据处理的正确性、系统处理效率。 |
| **测试结果** | |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | | **文件名** | **加载类型（全/增）** | **文件大小**  **（字节）** | **文件行数** | **数据库目标表名** | **目标表大小**  **（字节）** | **目标表行数** | **加载耗时** | |  |  |  |  |  |  |  |  | |  |  |  |  |  |  |  |  | |  |  |  |  |  |  |  |  | |

3.2        结构化数据查询场景

|  |  |
| --- | --- |
| **测试编号** |  |
| **测试项目** | 结构化数据查询场景 |
| **测试目的** | 测试简单查询案例（单表）在不同并发级别下，在各厂商平台上的运行效率。为确保数据加载的一致性及SQL语句的正确性，需要对查询结果作正确性检查。验证大数据平台对结构化数据的处理能力和处理效率。 |
| **测试依据** |  |
| **预置条件** | 数据加工测试完成 |
| **测试步骤** | 1、厂商提供执行测试案例TC\_01脚本，规定执行测试时间，递增并发级别 1、80、160、320、640、1280、1800、3600可逐级往上增加并发，记录并发sql执行次数；  2、记录测试案例的时间，CPU和I/O资源消耗情况等等。  3、对正确性进行验证：数据加工的正确性。 |
| **预期结果** | 能够正常查看表中数据，得到正确的数据结果 |
| **评分依据** | 1、衡量大数据应用开发平台对结构化数据的需求功能实现的设计和实施方面的能力；  2、衡量大数据应用开发平台对结构化数据的数据需求查询结果的正确性和查询效率； |
| **测试结果** | 对于查询性能测试，测试指标主要是各个SQL案例在不同并发级别下的执行时间和返回结果的正确性。同时还需要收集Cpu消耗比率、I/O读写情况（kb/s）等指标作参考。执行时间即查询请求提交到结果全部返回的全部时间。 |

3.3        非结构化数据加载场景——小文件

|  |  |
| --- | --- |
| **测试编号** |  |
| **测试项目** | 非结构化数据加载场景——小文件 |
| **测试目的** | 测试大数据应用开发平台对非结构化数据的加载处理场景。监控加载是否成功以及资源利用情况。 |
| **测试依据** |  |
| **预置条件** | 硬软件环境搭建完毕，非结构化数据的数据文件已生成。非结构化数据为小文件，几k到1M之间，包括图片、音频、视频文件等，数据规模：TB级别。 |
| **测试步骤** | 1、将非结构化数据文件放到不同的指定文件目录，测试在不同的并发级别下执行导入脚本，将对应的非结构化数据文件逐条导入。  2、记录不同并发级别下数据导入时间，同时收集资源消耗等信息。  3、对表的元数据信息及存储信息进行检查，对记录数目进行比对。 |
| **预期结果** | 1、不同并发级别下数据文件均可全部写入，元数据和存储信息正确。  2、不同并发级别下数据写入效率对比。  3、不同并发级别下资源使用情况对比。 |
| **评分依据** | 衡量大数据应用开发平台对非结构化数据处理的正确性、系统处理效率。 |
| **测试结果** |  |
|  |  |

3.4        非结构化数据查询场景——小文件

|  |  |
| --- | --- |
| **测试编号** |  |
| **测试项目** | 非结构化数据查询场景——小文件 |
| **测试目的** | 对非结构化数据进行单笔高并发查询进行验证。 |
| **测试依据** |  |
| **预置条件** | 硬软件环境搭建完毕，非结构化数据已写入。非结构化数据为小文件，几k到1M之间，包括图片、音频、视频文件等。数据规模：TB级别。 |
| **测试步骤** | 1、厂商提供执行脚本，测试不同的并发级别下单笔记录的查询，保证功能完整性和性能要求。  2、记录不同并发级别下的查询时间，资源消耗情况等。  3、对查询数据正确性进行验证。 |
| **预期结果** | 1、不同并发级别下数据的访问均正确。  2、不同并发级别下数据查询效率对比。  3、不同并发级别下资源使用情况对比。 |
| **评分依据** | 1、衡量大数据应用开发平台对非结构化数据的需求功能实现的设计和实施方面的能力；  2、衡量大数据应用开发平台对非结构化数据的数据需求查询结果的正确性和查询效率； |
| **测试结果** |  |

3.5        非结构化数据加载场景——大文件

|  |  |
| --- | --- |
| **测试编号** |  |
| **测试项目** | 非结构化数据加载场景——大文件 |
| **测试目的** | 测试大数据应用开发平台对非结构化数据的加载的处理场景。监控加载是否成功以及资源利用情况。 |
| **测试依据** |  |
| **预置条件** | 硬软件环境搭建完毕，非结构化数据的数据文件已生成。非结构化数据为大文件，1M以上到几百M之间，包括图片、音频、视频文件等。数据规模：TB级别 |
| **测试步骤** | 1、将非结构化数据文件放到指定文件目录下，执行批量导入脚本，将对应的非结构化数据文件导入（选择合适的数据存储格式进行：a、无压缩无加密方式；b、压缩方式；c、加密方式三种场景的数据导入测试）。  2、记录各种组合情况下的数据导入时间，压缩比、占用空间，同时收集资源消耗等信息。  3、对元数据信息及存储信息进行扫描检查，对记录数目进行比对，对数据正确性做抽查。 |
| **预期结果** | 1、在各种组合情况下数据文件均可全部导入，元数据和存储信息正确。  2、在各种组合情况下对数据写入效率，压缩比、占用空间进行比对。  3、在各种组合情况下对资源使用情况进行比对。 |
| **评分依据** | 衡量大数据应用开发平台对非结构化数据处理的正确性、系统处理效率。 |
| **测试结果** |  |

3.6        非结构化数据查询场景——大文件

|  |  |
| --- | --- |
| **测试编号** |  |
| **测试项目** | 非结构化数据查询场景——大文件 |
| **测试目的** | 根据关键字对非结构化数据进行单笔查询。 |
| **测试依据** |  |
| **预置条件** | 硬软件环境搭建完毕，非结构化数据已写入。非结构化数据为大文件，1M以上到几百M之间，包括图片、音频、视频文件等。 |
| **测试步骤** | 1、厂商提供执行脚本，测试各种组合情况下（a、无压缩无加密方式；b、压缩方式；c、加密方式）的单笔查询，保证功能完整性和性能要求。  2、记录各种组合情况下的查询时间，资源消耗情况等。  3、对查询数据正确性进行验证。 |
| **预期结果** | 1、在各种组合情况下数据的访问均正确。  2、在各种组合情况下对数据查询效率进行比对。  3、在各种组合情况下对资源使用情况进行比对。 |
| **评分依据** | 1、衡量大数据应用开发平台对非结构化数据的需求功能实现的设计和实施方面的能力；  2、衡量大数据应用开发平台对非结构化数据的数据需求查询结果的正确性和查询效率； |
| **测试结果** |  |

3.7        半结构化数据加载场景

|  |  |
| --- | --- |
| **测试编号** |  |
| **测试项目** | 半结构化数据加载场景 |
| **测试目的** | 测试大数据应用开发平台对半结构化数据的加载的处理场景。监控加载是否成功以及资源利用情况。 |
| **测试依据** |  |
| **预置条件** | 硬软件环境搭建完毕，半结构化数据的数据文件已生成。 |
| **测试步骤** | 1、将半结构化数据文件放到指定文件目录下，执行批量导入脚本，将对应的半结构化数据文件导入到HDFS或HBase中。  2、记录数据导入时间，同时收集资源消耗等信息。  3、对元数据信息及存储信息进行检查。 |
| **预期结果** | 1、数据文件均可全部正常导入到HDFS或HBase中，元数据和存储信息正确。  2、对在加载过程中的资源使用情况进行统计。 |
| **评分依据** | 衡量大数据应用开发平台对半结构化数据处理的正确性、系统处理效率。 |
| **测试结果** |  |

3.8        半结构化数据查询场景

|  |  |
| --- | --- |
| **测试编号** |  |
| **测试项目** | 半结构化数据查询场景 |
| **测试目的** | 对半结构化数据进行计算和查询 |
| **测试依据** |  |
| **预置条件** | 硬软件环境搭建完毕，半结构化数据已写入HDFS或HBase中。 |
| **测试步骤** | 1、厂商提供执行脚本，测试半结构化数据的求和模式（统计某单次出现次数等）、过滤模式（筛选出满足一定条件的数据等）、数据组织模式（对数据的分组归类等），保证功能完整性和性能要求。  2、记录各种计算及查询模式下的查询时间，资源消耗情况等。  3、对查询数据正确性进行验证。 |
| **预期结果** | 1、在各种计算及查询模式下数据的查询结果均正确。  2、对各种计算及查询模式下数据查询效率进行分析。  3、对各种计算及查询模式下资源使用情况进行分析。 |
| **评分依据** | 1、衡量大数据应用开发平台对半结构化数据的需求功能实现的设计和实施方面的能力；  2、衡量大数据应用开发平台对半结构化数据的数据需求查询结果的正确性和查询效率； |
| **测试结果** |  |

3.9        现有分析工具集成场景测试

3.9.1     工具集成——Freequery（可选题）

|  |  |
| --- | --- |
| **测试编号** |  |
| **测试项目** | 结构化数据查询接口实现场景——工具集成——Freequery |
| **测试目的** | 测试结构化数据存储于大数据平台后，厂商可否支持已有查询工具使用SQL或类SQL语句进行灵活查询。 |
| **测试依据** |  |
| **预置条件** | 硬软件环境搭建完毕，结构化数据已导入。 |
| **测试步骤** | 1、将厂商提供接口程序（jdbc）添加入freequery产品包  2、在freequery中创建相关数据源。  3、在freequery中创建查询。  4、查看是否返回查询结果。 |
| **预期结果** | 通过sql或类sql查询可返回结果。 |
| **评分依据** | 衡量大数据应用开发平台对查询工具的集成能力，验证查询数据的正确性。 |
| **测试结果** |  |
|  |  |

3.9.2     工具集成——Tableau9（必选题）

|  |  |
| --- | --- |
| **测试编号** |  |
| **测试项目** | 工具集成——Tableau9 |
| **测试目的** | 测试能否通过tableau9创建相关数据源，进行基于hadoop结构化数据存储的数据可视化分析。通过Tableau9工具直接验证hadoop的hive、sparkSQL的连通性，不需要厂商工具集成的开发。 |
| **测试依据** | Tableau9提供的接口（按技术方案选取）：hive（cloudera\hortonworks\MapR）、spark SQL |
| **预置条件** | 硬软件环境搭建完毕，结构化数据已写入。 |
| **测试步骤** | 1、厂商提供配置信息  2、数据服务人员尝试访问结构化数据表  3、完成数据加载并进行简单分析制图 |
| **预期结果** | 能够基于tableau9产品内置接口访问hadoop平台内结构化数据表 |
| **评分依据** | 衡量大数据应用开发平台对查询工具的集成能力，验证查询数据的正确性。 |
| **测试结果** |  |

4             基础组件测试——组件功能性部分

4.1                大数据应用开发平台各组件功能测试

4.1.1     HDFS

4.1.1.1 loader推送文件至HDFS

|  |  |
| --- | --- |
| **测试编号** |  |
| **测试项目** | HDFS功能测试 |
| **测试子项目** | Loader将本地文件上传到HDFS |
| **测试目的** | 检测Hadoop Loader是否将大数据上传到HDFS |
| **测试依据** |  |
| **预置条件** | 1.     已部署好大数据平台系统，配置好Hadoop Loader服务。  2.     大数据平台和接口机网络正常。  3.     将上游系统的数据下载到loader服务节点的本地目录下，如/data1/loaddata/ |
| **测试步骤** | 1.     Loader配置文件loadConfig.xml配置上传的源目录和目的路径。  2.     配置文件Config.properties中的ScanInterval=10  #每隔10秒loader检查源路径下数据  3.     等待数据文件全部推送结束，查看HDFS对应目录。  4.     查看文件备份目录。 |
| **预期结果** | 1、  数据文件全部推送到hfds对应目录。数量、内容正确。  2、  数据文件同时备份到接口机规定目录下。数量、内容正确。 |
| **测试结果** |  |

4.1.1.2 HDFS导入导出测试

|  |  |
| --- | --- |
| **测试编号** |  |
| **测试项目** | HDFS功能测试 |
| **测试子项目** | Hdfs导入导出测试 |
| **测试目的** | 测试使用客户端导入与导出hdfs数据。 |
| **测试依据** |  |
| **预置条件** | 1.搭建测试环境，系统运行正常 |
| **测试步骤** | 1． 使用客户端连接集群  2． 使用put把数据拷贝至hdfs中  3． 使用get从hdfs中下载数据 |
| **预期结果** | 支持客户端导入与导出hdfs数据。 |
| **测试结果** |  |

4.1.1.3 loader回传HDFS文件到本地系统

|  |  |
| --- | --- |
| **测试编号** |  |
| **测试项目** | HDFS功能测试 |
| **测试子项目** | Hadoop-loader服务将HDFS文件导出到本地 |
| **测试目的** | 导出HDFS文件到本地文件系统 |
| **测试依据** |  |
| **预置条件** | 1．按照测试组网图搭建测试环境；  2．在测试环境下完成产品初始配置副本数为3；  3．各服务正常。  4．准备HDFS文件/bank/TINT/111.txt |
| **测试步骤** | 1.     1. Loader配置文件downloadConfig.xml配置下载的的源目录和目的路径。  2.     2. 配置文件Config.properties中的ScanInterval=10  #每隔10秒  3.     3. loader检查源路径下数据  4.     4. 等待数据文件全部推送结束，查看本地文件系统的路径。 |
| **预期结果** | HDFS文件复制至本地文件系统正常运行。 |
| **测试结果** |  |

4.1.1.4 HDFS存储能力

|  |  |
| --- | --- |
| **测试编号** |  |
| **测试项目** | HDFS功能测试 |
| **测试子项目** | HDFS查看存储文件以及存储数据的大小 |
| **测试目的** | 用于从本地文件系统上传文件到HDFS中 |
| **测试依据** |  |
| **预置条件** | 1．按照测试组网图搭建测试环境； |
| **测试步骤** | 1．根据已上传至HDFS文件系统内的文件，查看HDFS的存储 |
| **预期结果** | HDFS各节点存储分布平衡，存储正常 |
| **测试结果** |  |

4.1.1.5 HDFS文件读取

|  |  |
| --- | --- |
| **测试编号** |  |
| **测试项目** | HDFS功能测试 |
| **测试子项目** | HDFS文件读取 |
| **测试目的** | 该命令用于在终端显示（标准输出stdout）文件中的内容，类似Linux系统中的cat |
| **测试依据** |  |
| **预置条件** | 1．按照测试组网图搭建测试环境；  2．在测试环境下完成产品初始配置副本数为3；  3．启动HDFS，mr系统。 |
| **测试步骤** | 1．测试显示文件内容命令  格式参考如下  hadoop fs –cat 文件名 |
| **预期结果** | 显示文件内容正确 |
| **测试说明** |  |

4.1.1.6 Namenode主节点故障测试

|  |  |
| --- | --- |
| **测试编号** |  |
| **测试项目** | HDFS功能测试 |
| **测试子项目** | Namenode主节点故障测试 |
| **测试目的** | 测试namenode主节点故障下的可用情况 |
| **测试依据** |  |
| **预置条件** | 1.集群正常运行； |
| **测试步骤** | 1、       向HDFS中持续进行文件的读写操作，模拟背景业务；  2、       背景业务运行过程中，将主NameNode节点的进程通过管理系统停止来模拟故障；  3、观察背景业务是否中断，业务是否有损失；  4、观察是否有主备倒换告警上报 |
| **预期结果** | Namenode进程能够自动主备切换。 |
| **测试结果** |  |

4.1.1.7 DataNode节点故障及恢复测试

|  |  |
| --- | --- |
| **测试编号** |  |
| **测试项目** | HDFS功能测试 |
| **测试子项目** | Datanode主节点故障测试 |
| **测试目的** | 测试datanode节点故障下的可用情况 |
| **测试依据** |  |
| **预置条件** | 1.集群正常运行； |
| **测试步骤** | 1、       向HDFS中持续进行文件的读写操作，模拟背景业务；  2、       背景业务运行过程中，将集群中的一个DataNode节点的进程通过管理系统停止来模拟故障；  3、观察背景业务是否中断，业务是否有损失。 |
| **预期结果** | 单个Datanode节点的故障不会影响整个系统提供的服务。 |
| **测试结果** |  |

4.1.1.8 HDFS备份恢复

|  |  |
| --- | --- |
| **测试编号** |  |
| **测试项目** | HDFS功能测试 |
| **测试子项目** | HDFS备份恢复能力 |
| **测试目的** | 验证HDFS的备份恢复能力 |
| **测试依据** |  |
| **预置条件** | 1.集群正常运行； |
| **测试步骤** | 1.         在HDFS中创建目录/backup/a、/backup/b，分别向两个目录中上传10G数据。同时启动一个能够长时间运行HFDS读写脚本，来模拟背景业务；  2.         对目录/backup/a进行全量备份，观察备份是否成功，备份成功后将/backup目录下的a、b目录同时删除；  3.         使用备份的数据进行恢复，观察/backup/a是否恢复成功，内容是否与删除前的一致，观察/backup/b是否不会被恢复；  4.         再向/backup/a中追加1G的数据文件，并执行增量备份；  5.         删除/backup/a目录，并使用增量备份内容进行恢复，观察恢复的内容是否与删除前的一致；  观察/backup/a中追加的1G数据文件是否会被恢复。 |
| **预期结果** | HDFS具备指定目录进行备份的能力；  HDFS具备在线备份的能力，不影响业务 |
| **测试结果** |  |

4.1.1.9 HDFS白名单功能

|  |  |
| --- | --- |
| **测试编号** |  |
| **测试项目** | HDFS功能测试 |
| **测试子项目** | 大数据平台HDFS白名单功能测试 |
| **测试目的** | 验证大数据平台HDFS白名单功能测试 |
| **测试依据** |  |
| **预置条件** | 1.系统正常运行； |
| **测试步骤** | 1、开启HDFS白名单开关；  2、不加入客户端A在HDFS的白名单中，使用客户端A访问HDFS。  将客户端A加入HDFS的白名单中，使用客户端A访问HDFS |
| **预期结果** | 加入白名单的客户端有权限访问集群；  未加入白名单的客户端无权限访问集群； |
| **测试结果** |  |

4.1.1.10                HDFS用户权限

|  |  |
| --- | --- |
| **测试编号** |  |
| **测试项目** | HDFS功能测试 |
| **测试子项目** | HDFS用户权限控制 |
| **测试目的** | 验证HDFS户权限控制功能测试 |
| **测试依据** |  |
| **预置条件** | 1.系统正常运行； |
| **测试步骤** | 1、HDFS权限测试  验证合法用户对自己拥有权限的数据有操作权限：  验证合法用户对自己没有权限的数据没有操作权限： |
| **预期结果** | 用户只能够访问、操作自己权限范围内的数据 |
| **测试结果** |  |

4.1.1.11                集群节点在线扩容

|  |  |
| --- | --- |
| **测试编号** |  |
| **测试项目** | HDFS功能测试 |
| **测试子项目** | 对集群的节点在线扩容 |
| **测试目的** | 验证Hadoop集群是否具备在线扩容能力 |
| **测试依据** |  |
| **预置条件** | 各集群正常运行，待扩容的服务器正常运行 |
| **测试步骤** | 1.在当前节点规模时，提交一个排序的terasort任务（第一次）。  2.选择待扩容服务器进行在线扩容，同时向集群提交一个排序的terasort任务（第二次）。  3.观察扩容过程对在线应用的影响及耗时  4.观察扩容完成后集群的服务状态  5. 提交一个排序的terasort任务（第三次） |
| **预期结果** | 1、整个扩容过程在线完成，系统已有任务不受影响；  2、操作简单，具有一键扩展功能。 |
| **测试结果** |  |

4.1.1.12                集群节点在线收缩

|  |  |
| --- | --- |
| **测试编号** |  |
| **测试项目** | HDFS功能测试 |
| **测试子项目** | 对集群的节点在线收缩 |
| **测试目的** | 验证Hadoop集群是否具备在线收缩能力 |
| **测试依据** |  |
| **预置条件** | 各集群正常运行，待扩容的服务器正常运行 |
| **测试步骤** | 1.当前规模时，提交一个排序的terasort任务（第一次），记录完成时间。  2.选择1台服务器进行在线收缩，同时向集群提交一个排序的terasort任务（第二次）。  3.观察收缩过程对在线应用的影响及耗时  4.观察收缩完成后集群的服务状态  5.提交一个排序的terasort任务（第三次），记录完成时间。 |
| **预期结果** | 1、整个扩容过程在线完成，系统已有任务不受影响；  2、操作简单，具有一键收缩功能。 |
| **测试结果** |  |

4.1.2     HBase

4.1.2.1 Hbase存储能力

|  |  |
| --- | --- |
| **测试编号** |  |
| **测试项目** | Hbase功能测试 |
| **测试子项目** | Hbase查看存储表 |
| **测试目的** | 用于存储大数据表的能力 |
| **测试依据** |  |
| **预置条件** | 1．按照测试组网图搭建测试环境；  2．导入数据表。 |
| **测试步骤** | 1．根据已上传的Hbase表，查看habse的存储表能力 |
| **预期结果** | Hbase页面中Table Details中存储了上传的Hbase表，存储正常 |
| **测试结果** |  |

4.1.2.2 Hbase查看数据

|  |  |
| --- | --- |
| **测试编号** |  |
| **测试项目** | Hbase功能测试 |
| **测试子项目** | Hbase查看数据 |
| **测试目的** | 验证Hbase功能 |
| **测试依据** |  |
| **预置条件** | 1.     按照测试组网图搭建测试环境；  2.     启动HDFS，mr系统。  3.     启动Hbase |
| **测试步骤** | 1.     登陆Hbase主节点所在服务器  2.     切换到mr用户下，执行Hbase shell命令  3.     在shell中输入：scan '表名称'  Hbase(main):010:0> scan 'table' |
| **预期结果** | 能够正常查看表中数据 |
| **测试说明** |  |

4.1.2.3 Hbase导入测试

|  |  |
| --- | --- |
| **测试编号** |  |
| **测试项目** | Hbase功能测试 |
| **测试子项目** | Hbase导入测试 |
| **测试目的** | 测试Hbase能够根据extimporttsv.input配置项正确导入数据 |
| **测试依据** |  |
| **预置条件** | 1．  按照测试组网图搭建测试环境；  2．  在测试环境下完成产品初始配置副本数为3；  3．  启动HDFS，mr系统。  4．  启动Hbase |
| **测试步骤** | 1．  准备数据并将数据保存到HDFS文件系统中/test；  2．  Hbase中建表  3．  新建模板bank.ini，按照建表的实际情况填写，上传至HDFS的/test目录下  模板原型：  4．  使用命令行方式执行导入：Hbase com.citicbank.Hbase.importer.ExtImportTsv -Dextimporttsv.conf=/test/bank.ini |
| **预期结果** | Hbase可以正确将路径extimporttsv.input中的文件导入表中。 |
| **测试结果** |  |

4.1.2.4 Hbase导出测试

|  |  |
| --- | --- |
| **测试编号** |  |
| **测试项目** | Hbase功能测试 |
| **测试子项目** | Hbase导出测试 |
| **测试目的** | 测试Hbase能够根据模板文件中的Hbase\_table正确导出数据 |
| **测试依据** |  |
| **预置条件** | 1．  按照测试组网图搭建测试环境；  2．  在测试环境下完成产品初始配置副本数为3；  3．  启动HDFS，mr系统。  4．  启动Hbase  5．  准备好Hbase表文件。 |
| **测试步骤** | 1.     通过配置文件方式设定导出规则  2.     执行导出操作  3.     检查导出的表数据和生成的报告 |
| **预期结果** | Hbase能够正确导出表数据。 |
| **测试结果** |  |

4.1.2.5 Hbase实时查询大表数据

|  |  |
| --- | --- |
| **测试编号** |  |
| **测试项目** | Hbase功能测试 |
| **测试子项目** | Hbase查看数据 |
| **测试目的** | 验证Hbase功能 |
| **测试依据** |  |
| **预置条件** | 1.     按照测试组网图搭建测试环境；  2.     启动HDFS，mr系统。  3.     启动Hbase |
| **测试步骤** | 1.     登陆Hbase主节点所在服务器  2.     切换到mr用户下，执行Hbase shell命令  3.     在shell中输入：get '表名称', '行名称' |
| **预期结果** | 能够正常查看表中某行数据 |
| **测试结果** |  |

4.1.2.6 大数据量hbase表入库测试

|  |  |
| --- | --- |
| **测试编号** |  |
| **测试项目** | HBASE性能测试 |
| **测试子项目** | 大数据量hbase表入库测试 |
| **测试目的** | 验证大数据量hbase表入库性能 |
| **测试依据** |  |
| **预置条件** | 1.     按照测试组网图搭建测试环境；  2.     在测试环境下完成产品初始配置；  3.     启动HDFS，mr。  4.     启动hbase  5.     源数据库文件大小，如100G |
| **测试步骤** | 1.     创建表testtable  2.     上传原始数据文件到HDFS  3.     准备模板  创建文件testtale.ini  4.     运行导入命令：  Hbasecom.citicbank.hbase.importer.ExtImportTsv -Dextimporttsv.conf=/test/testable.ini |
| **预期结果** | 能够正常完成hbase大数据量表的导入，导入时间符合预期 |
| **测试结果** |  |

4.1.2.7 Hbase用户权限测试

|  |  |
| --- | --- |
| **测试编号** |  |
| **测试项目** | Hbase功能测试 |
| **测试子项目** | hbase用户权限控制 |
| **测试目的** | 验证hbase户权限控制功能测试 |
| **测试依据** |  |
| **预置条件** | 1.系统正常运行； |
| **测试步骤** | 1. 创建用户组与角色  2. 准备普通：a用户、b用户、c 用户；  3. 准备管理员用户：d用户；  4. 分别创建a1表属于a用户、b1表属于b用户、c1表数据c用户；赋予b用户访问a1表某列族权限，赋予c 用户访问a1表某一列的权限以及某一单元格对其不可见；  5. 分别a、b、c用户访问a1表，b1表，c1表；  6. 记录用户访问表现象 |
| **预期结果** | 1. 用户能够访问自己的表数据；  2. 不同用户相互间不能访问表数据；  3. 管理员用户可以访问所有用户表数据；  4. 用户访问表数据权限可修改控制 |
| **测试结果** |  |

4.1.2.8  HBase白名单功能

|  |  |
| --- | --- |
| **测试编号** |  |
| **测试项目** | Hbase功能测试 |
| **测试子项目** | HBase白名单功能测试 |
| **测试目的** | 验证大数据平台白名单功能测试 |
| **测试依据** |  |
| **预置条件** | 1.系统正常运行； |
| **测试步骤** | 1、开启HBase白名单开关；  2、不加入客户端A在HBase的白名单中，使用客户端A访问HBase。  将客户端A加入HBase的白名单中，使用客户端A访问HBase |
| **预期结果** | 加入白名单的客户端有权限访问集群；  未加入白名单的客户端无权限访问集群； |
| **测试结果** |  |

4.1.2.9  PE测试Hbase性能

|  |  |
| --- | --- |
| **测试编号** |  |
| **测试项目** | HBASE性能测试 |
| **测试子项目** | PE基准测试 |
| **测试目的** | 验证大数据量hbase表入库性能 |
| **测试依据** |  |
| **预置条件** | 1.     按照测试组网图搭建测试环境；  2.     在测试环境下完成产品初始配置；  3.     启动HDFS，mr。  4.     启动hbase |
| **测试步骤** | 测试如下用例：  1.     通过PE工具，测试hbase顺序写性能，数据量1000万文件  2.     通过PE工具，测试hbase顺序读1000万性能，数据量1000万文件  3.     通过PE工具，测试hbase随机读100万性能，数据量100万文件  4.     通过PE工具，测试hbase顺序扫描1000万性能，数据量1000万文件  5.     通过PE工具，测试hbase随机扫描1000万性能，数据量1000万文件  6.     通过PE工具，测试hbase随机写1000万性能，数据量1000万文件 |
| **预期结果** | PE测试性能符合预期 |
| **测试结果** |  |

4.1.2.10                HMaster节点失效及恢复测试

|  |  |
| --- | --- |
| **测试编号** |  |
| **测试项目** | Hbase可靠性测试 |
| **测试子项目** | HMaster节点失效及恢复测试 |
| **测试目的** | HMaster是否具备HA功能 |
| **测试依据** |  |
| **预置条件** | 1.集群正常运行； |
| **测试步骤** | 1.用PE对hbase进行读写操作  2.对主节点通过管理系统停止主HMaster进程；  3.在Manager页面观察备HMaster状态，其由备切换为主  4.主备切换过程中，观察HBase客户端日志，读写操作执行状态和HMaster倒换时间  5.倒换完成，检查倒换告警信息是否上报正确 |
| **预期结果** | HMaster故障情况下能够自动进行切换 |
| **测试结果** |  |

4.1.2.11                HRegionServer故障情况下能够自动进行切换

|  |  |
| --- | --- |
| **测试编号** |  |
| **测试项目** | HBase可靠性测试 |
| **测试子项目** | RegionServer节点失效及恢复测试 |
| **测试目的** | RegionServer节点故障对系统的影响 |
| **测试依据** |  |
| **预置条件** | 1.集群正常运行； |
| **测试步骤** | 1.用PE对hbase进行读写操作  2.对一个RegionServer节点通过管理系统停止；  3.观察HBase客户端日志，观察读写操作是否受到影响 |
| **预期结果** | Regionserver节点故障并不影响HBase集群提供的服务 |
| **测试结果** |  |

4.1.3     MapReduce&YARN

4.1.3.1 运行wordcount

|  |  |
| --- | --- |
| **测试编号** |  |
| **测试项目** | MapReduce&YARN测试 |
| **测试子项目** | Wordcount测试 |
| **测试目的** | 测试MapReduce&YARN运行wordcount功能 |
| **测试依据** | DAP需求文档 |
| **预置条件** | 1. REDHAT系统正常运行  2. ZDH集群系统正常运行  3. ZDHManager及其他依赖服务正常运行 |
| **测试步骤** | 1.      ssh登录到主机，在后台执行命令su – mr  2.      上传文件到hdfs，命令：hadoop fs -mkdir /rr  hadoop fs -put /file1.txt /rr/  3.      执行命令yarn jar /home/mr/yarn/share/hadoop/mapreduce/hadoop-mapreduce-examples-2.5.0-cdh5.3.2.jar wordcount /rr /output提交一个wordcount作业。 |
| **预期结果** | Wordcount作业运行成功 |
| **测试说明** |  |

4.1.3.2 YARN用户权限

|  |  |
| --- | --- |
| **测试编号** |  |
| **测试项目** | yarn安全性测试 |
| **测试子项目** | yarn用户权限控制 |
| **测试目的** | 验证yarn户权限控制功能测试 |
| **测试依据** |  |
| **预置条件** | 1.系统正常运行； |
| **测试步骤** | 1、Yarn权限测试  没有权限的用户不能向queue提交任务  有权限的用户可以向queue提交任务 |
| **预期结果** | 用户只能够访问、操作自己权限范围内的数据 |
| **测试结果** |  |

4.1.3.3 YARN白名单功能

|  |  |
| --- | --- |
| **测试编号** |  |
| **测试项目** | yarn安全性测试 |
| **测试子项目** | yarn白名单功能测试 |
| **测试目的** | 验证大数据平台yarn白名单功能测试 |
| **测试依据** |  |
| **预置条件** | 1.系统正常运行； |
| **测试步骤** | 1、开启YARN白名单开关；  2、不加入客户端A在Yarn的白名单中，使用客户端A访问Yarn。  将客户端A加入Yarn的白名单中，使用客户端A访问Yarn |
| **预期结果** | 加入白名单的客户端有权限访问集群；  未加入白名单的客户端无权限访问集群； |
| **测试结果** |  |

4.1.3.4 YARNResourceManager主节点故障测试

|  |  |
| --- | --- |
| **测试编号** |  |
| **测试项目** | 大数据平台yarn可靠性测试 |
| **测试子项目** | yarn ResourceManager主节点故障测试 |
| **测试目的** | 测试ResourceManager主节点故障下的可用情况 |
| **测试依据** |  |
| **预置条件** | 1.集群正常运行； |
| **测试步骤** | 1、       执行Wordcount测试，模拟背景业务；  2、       背景业务运行过程中，将集群中的主ResourceManager进程通过管理系统停止来模拟故障；  3、观察背景业务是否中断，业务是否有损失查看是否有相关主备倒换告警信息。 |
| **预期结果** | ResourceManager进程能够自动主备倒换 |
| **测试结果** |  |

4.1.3.5 YARN NodeManager节点故障测试

|  |  |
| --- | --- |
| **测试编号** |  |
| **测试项目** | 大数据平台yarn可靠性测试 |
| **测试子项目** | yarn NodeManager主节点故障测试 |
| **测试目的** | 验证NodeManager进程故障对系统的影响 |
| **测试依据** |  |
| **预置条件** | 1.集群正常运行； |
| **测试步骤** | 1、       执行Wordcount测试，模拟背景业务；  2、       背景业务运行过程中，将集群中的一个NodeManager进程通过管理系统停止来模拟故障；  3、观察背景业务是否中断。。 |
| **预期结果** | 整体服务可以正常执行完成 |
| **测试结果** |  |

4.1.4     Hive

4.1.4.1 HiveSQL基本操作

|  |  |
| --- | --- |
| **测试编号** |  |
| **测试项目** | Hive测试 |
| **测试子项目** | HiveSQL基本操作测试 |
| **测试目的** | 测试hive sql是否支持基本sql命令 |
| **测试依据** |  |
| **预置条件** | 1．按照测试组网图搭建测试环境；  2．hdfs、Mapreduce、hive正常运行；  3.进入hive命令行操作。 |
| **测试步骤** | 1. 用tpch工具生成8张表数据  2.将生成的数据上传到hdfs上  3.在hive里建立跟数据一致结构的8张表  4.在hive执行如下sql测试hive sql  DROP TABLE customer;  DROP TABLE orders;  DROP TABLE lineitem;  DROP TABLE supplier;  DROP TABLE nation;  DROP TABLE region;  DROP TABLE part;  DROP TABLE q8\_national\_market\_share;  -- create the tables and load the data  create external table part (P\_PARTKEY INT, P\_NAME STRING, P\_MFGR STRING, P\_BRAND STRING, P\_TYPE STRING, P\_SIZE INT, P\_CONTAINER STRING, P\_RETAILPRICE DOUBLE, P\_COMMENT STRING) ROW FORMAT DELIMITED FIELDS TERMINATED BY '|' STORED AS TEXTFILE LOCATION '/part';  create external table customer (C\_CUSTKEY INT, C\_NAME STRING, C\_ADDRESS STRING, C\_NATIONKEY INT, C\_PHONE STRING, C\_ACCTBAL DOUBLE, C\_MKTSEGMENT STRING, C\_COMMENT STRING) ROW FORMAT DELIMITED FIELDS TERMINATED BY '|' STORED AS TEXTFILE LOCATION '/customer';  Create external table lineitem (L\_ORDERKEY INT, L\_PARTKEY INT, L\_SUPPKEY INT, L\_LINENUMBER INT, L\_QUANTITY DOUBLE, L\_EXTENDEDPRICE DOUBLE, L\_DISCOUNT DOUBLE, L\_TAX DOUBLE, L\_RETURNFLAG STRING, L\_LINESTATUS STRING, L\_SHIPDATE STRING, L\_COMMITDATE STRING, L\_RECEIPTDATE STRING, L\_SHIPINSTRUCT STRING, L\_SHIPMODE STRING, L\_COMMENT STRING) ROW FORMAT DELIMITED FIELDS TERMINATED BY '|' STORED AS TEXTFILE LOCATION '/lineitem';  create external table orders (O\_ORDERKEY INT, O\_CUSTKEY INT, O\_ORDERSTATUS STRING, O\_TOTALPRICE DOUBLE, O\_ORDERDATE STRING, O\_ORDERPRIORITY STRING, O\_CLERK STRING, O\_SHIPPRIORITY INT, O\_COMMENT STRING) ROW FORMAT DELIMITED FIELDS TERMINATED BY '|' STORED AS TEXTFILE LOCATION '/orders';  create external table supplier (S\_SUPPKEY INT, S\_NAME STRING, S\_ADDRESS STRING, S\_NATIONKEY INT, S\_PHONE STRING, S\_ACCTBAL DOUBLE, S\_COMMENT STRING) ROW FORMAT DELIMITED FIELDS TERMINATED BY '|' STORED AS TEXTFILE LOCATION '/supplier';  create external table nation (N\_NATIONKEY INT, N\_NAME STRING, N\_REGIONKEY INT, N\_COMMENT STRING) ROW FORMAT DELIMITED FIELDS TERMINATED BY '|' STORED AS TEXTFILE LOCATION '/nation';  create external table region (R\_REGIONKEY INT, R\_NAME STRING, R\_COMMENT STRING) ROW FORMAT DELIMITED FIELDS TERMINATED BY '|' STORED AS TEXTFILE LOCATION '/region';  -- create the result table  create table q8\_national\_market\_share(o\_year string, mkt\_share double);  -- the query  insert overwrite table q8\_national\_market\_share  select  o\_year, sum(case when nation = 'ROMANIA' then volume else 0.0 end) / sum(volume) as mkt\_share  from  (  select  year(o\_orderdate) as o\_year, l\_extendedprice \* (1-l\_discount) as volume,  n2.n\_name as nation  from  nation n2 join  (select o\_orderdate, l\_discount, l\_extendedprice, s\_nationkey  from supplier s join  (select o\_orderdate, l\_discount, l\_extendedprice, l\_suppkey  from part p join  (select o\_orderdate, l\_partkey, l\_discount, l\_extendedprice, l\_suppkey from lineitem l join  (select o\_orderdate, o\_orderkey  from orders o join  (select c.c\_custkey  from customer c join  (select n1.n\_nationkey  from nation n1 join region r  on n1.n\_regionkey = r.r\_regionkey and r.r\_name = 'EUROPE'  ) n11 on c.c\_nationkey = n11.n\_nationkey  ) c1 on c1.c\_custkey = o.o\_custkey  ) o1 on l.l\_orderkey = o1.o\_orderkey and o1.o\_orderdate >= '1995-01-01'  and o1.o\_orderdate < '1996-12-31'  ) l1 on p.p\_partkey = l1.l\_partkey and p.p\_type = 'LARGE BRUSHED BRASS'  ) p1 on s.s\_suppkey = p1.l\_suppkey  ) s1 on s1.s\_nationkey = n2.n\_nationkey  ) all\_nation  group by o\_year  order by o\_year; |
| **预期结果** | Hive sql执行成功 |
| **测试结果** |  |

4.1.4.2 HiveAggregation处理性能

|  |  |
| --- | --- |
| **测试编号** |  |
| **测试项目** | Hive测试 |
| **测试子项目** | HiveAggregation处理性能测试 |
| **测试目的** | 测试结构化数据存储引擎引擎的HiveAggregation处理性能 |
| **测试依据** |  |
| **预置条件** | 1. 按照测试组网图搭建测试环境,包含m个数据节点/计算节点、2个元数据节点/作业调度节点，HDFS配置副本数为3  2. 非结构化存储引擎已经运行  3.  HiBench测试工具已经完成配置  HiBench/ hivebench/conf/configure.sh配置如下：  DATASIZE=100GB  4. 使用hadoop用户登陆 |
| **测试步骤** | 进入HiBench/hivebench目录  执行conf/configure.sh  执行bin/prepare.sh  执行bin/run-aggregation.sh |
| **预期结果** | HiveAggregation成功执行结果符合预期。 |
| **测试结果** |  |

4.1.4.3 HiveJoin处理性能

|  |  |
| --- | --- |
| **测试编号** | 5.6 |
| **测试项目** | Hive测试 |
| **测试子项目** | HiveJoin处理性能测试 |
| **测试目的** | 测试结构化数据存储引擎引擎的HiveJoin处理性能 |
| **测试依据** |  |
| **预置条件** | 1. 按照测试组网图搭建测试环境,包含m个数据节点/计算节点、2个元数据节点/作业调度节点，HDFS配置副本数为3  2. 非结构化存储引擎已经运行  3.  HiBench测试工具已经完成配置  HiBench/terasort/conf/configure.sh配置如下：  DATASIZE=100GB  4. 使用hadoop用户登陆 |
| **测试步骤** | 进入HiBench/hivebench目录  执行conf/configure.sh  执行bin/prepare.sh  执行bin/run-join.sh |
| **预期结果** | HiveJoin成功执行结果符合预期。 |
| **测试结果** |  |

4.1.5     Impala

4.1.5.1  ImpalaSQL基本操作

|  |  |
| --- | --- |
| **测试编号** |  |
| **测试项目** | Impala测试 |
| **测试子项目** | ImpalaSQL基本操作测试 |
| **测试目的** | 测试hive sql是否支持基本sql命令 |
| **测试依据** |  |
| **预置条件** | 1．按照测试组网图搭建测试环境；  2．hdfs、Mapreduce、impala正常运行； |
| **测试步骤** | 1. 用tpch工具生成8张表数据  2.将生成的数据上传到hdfs上  3.在impala里建立跟数据一致结构的8张表  4.在impala执行如下sql测试impala sql  -----建表------------------------------  create external table part  (p\_partkey int,   p\_name string,   p\_mfgr string,   p\_brand string,   p\_type string,   p\_size int,   p\_container string,   p\_retailprice double,   p\_comment string  )row format delimited fields terminated by '|'  location '/impala/part';  create external table region  (r\_regionkey int,   r\_name string,   r\_comment string  )row format delimited fields terminated by '|'  location '/impala/region';  create external table nation  (n\_nationkey int,   n\_name string,   n\_regionkey int,   n\_comment string  ) row format delimited fields terminated by '|'  location '/impala/nation';  create external table supplier  (s\_suppkey int,   s\_name string,   s\_address string,   s\_nationkey int, --Foreign key to N\_NATIONKEY   s\_phone string,   s\_acctbal double,   s\_comment string  ) row format delimited fields terminated by '|'  location '/impala/supplier';  create external table partsupp  (ps\_partkey int,  --Foreign key to P\_PARTKEY   ps\_suppkey int,  --Foreign key to S\_SUPPKEY   ps\_availqty int,   ps\_supplycost double,   ps\_comment string  ) row format delimited fields terminated by '|'  location '/impala/partsupp';  create external table customer  (c\_custkey int,   c\_name string,   c\_address string,   c\_nationkey int,    --Foreign key to N\_NATIONKEY   c\_phone string,   c\_acctbal double,   c\_mktsegment string,   c\_comment string  ) row format delimited fields terminated by '|'  location '/impala/customer';  create external table orders  (o\_orderkey int,   o\_custkey int,    --Foreign key to C\_CUSTKEY   o\_orderstatus string,   o\_totalprice double,   o\_orderdate timestamp,   o\_orderpriority string,   o\_clerk string,   o\_shippriority int,   o\_comment string  ) row format delimited fields terminated by '|'  location '/impala/orders';  create external table lineitem  (l\_orderkey int,   l\_partkey int,   l\_suppkey int,   l\_linenumber int,   l\_quantity double,   l\_extendedprice double,   l\_discount double,   l\_tax double,   l\_returnflag string,   l\_linestatus string,   l\_shipdate timestamp,   l\_commitdate timestamp,   l\_receiptdate timestamp,   l\_shipinstruct string,   l\_shipmode string,   l\_comment string  ) row format delimited fields terminated by '|'  location '/impala/lineitem';  -----------建立临时表-------------------------------------------  create table q8\_national\_market\_share(o\_year string, mkt\_share double);  -------------------------查询------------------------------------  insert overwrite table q8\_national\_market\_share   select     cast(o\_year as string), sum(case when nation = 'BRAZIL' then volume else 0.0 end) / sum(volume) as mkt\_share   from     (   select     year(o\_orderdate) as o\_year, l\_extendedprice \* (1-l\_discount) as volume,     n2.n\_name as nation       from         nation n2 join           (select o\_orderdate, l\_discount, l\_extendedprice, s\_nationkey            from supplier s join             (select o\_orderdate, l\_discount, l\_extendedprice, l\_suppkey              from part p join                (select o\_orderdate, l\_partkey, l\_discount, l\_extendedprice, l\_suppkey                 from lineitem l join                   (select o\_orderdate, o\_orderkey                    from orders o join                      (select c.c\_custkey                       from customer c join                         (select n1.n\_nationkey                          from nation n1 join region r                          on n1.n\_regionkey = r.r\_regionkey and r.r\_name = 'AMERICA'                          ) n11 on c.c\_nationkey = n11.n\_nationkey                       ) c1 on c1.c\_custkey = o.o\_custkey                    ) o1 on l.l\_orderkey = o1.o\_orderkey and o1.o\_orderdate >= '1995-01-01'                            and o1.o\_orderdate < '1996-12-31'                 ) l1 on p.p\_partkey = l1.l\_partkey and p.p\_type = 'ECONOMY ANODIZED STEEL'              ) p1 on s.s\_suppkey = p1.l\_suppkey           ) s1 on s1.s\_nationkey = n2.n\_nationkey     ) all\_nation   group by o\_year   order by o\_year   LIMIT 2147483647; |
| **预期结果** | Impala sql执行成功 |
| **测试结果** |  |

4.1.6     Storm

4.1.6.1  流数据筛选

|  |  |
| --- | --- |
| **测试编号** |  |
| **测试项目** | Storm测试 |
| **测试子项目** | storm流数据筛选 |
| **测试目的** | 验证storm支持流数据筛选功能 |
| **测试依据** |  |
| **预置条件** | 验证支持流数据筛选功能 |
| **测试步骤** | I1.使用模拟器，将数据文件中的信息实时发送给流处理软件  I2.将过滤后的数据流进行输出 |
| **预期结果** | 能够将符合业务要求的数据筛选出来，并形成数据资产 |
| **测试说明** |  |

4.1.7     Spark

4.1.7.1  数据库基本操作

|  |  |
| --- | --- |
| **测试编号** |  |
| **测试项目** | Spark SQL功能测试 |
| **测试子项目** | 数据库基本操作 |
| **测试目的** | 验证数据库基本操作是否正常。 |
| **测试依据** |  |
| **预置条件** | 1．按照测试组网图搭建测试环境；  2．Spark正常运行；  3．进入spark-sql命令行操作。 (bin/spark-sql) |
| **测试步骤** | 1. 在spark-sql命令行中输入如下命令：  CREATE (DATABASE|SCHEMA) [IF NOT EXISTS] database\_name  [COMMENT 'database\_comment']  [LOCATION hdfs\_path];  此命令表示创建名为database\_name 的数据库。COMMENT是为数据库添加注释，LOCATION表示数据库在hdfs中的存储路径。  2. 在spark-sql命令行中输入如下命令查看新建数据库的属性：     describe database database\_name;  3. 在spark-sql命令行中输入如下命令删除新建的数据库：  DROP (DATABASE|SCHEMA) [IF EXISTS] database\_name; |
| **预期结果** | 1. 执行步骤1，成功创建数据库database\_name；通过show databases命令可以查看新增的数据库。  2. 执行步骤2，成功显示数据库database\_name的注释和在hdfs的存储路径；  3. 执行步骤3，成功删除数据库database\_name。通过show database命令可以查看数据库已不存在。 |
| **测试说明** |  |

4.1.7.2  表操作

|  |  |
| --- | --- |
| **测试编号** |  |
| **测试项目** | Spark SQL功能测试 |
| **测试子项目** | 表操作 |
| **测试目的** | 验证对表的基本操作 |
| **测试依据** |  |
| **预置条件** | 1．按照测试组网图搭建测试环境；  2．Spark正常运行；  3．进入spark-sql命令行操作。 |
| **测试步骤** | 1. 创建表：Create table testable (a int, b int, c int);  增加列：Alter table testable add columns (d int);  修改列名：Alter table testable change a a1 int;  修改列类型：Alter table testable change a1 a1 string;  修改表名：ALTER table table\_name rename to new\_table\_name; |
| **预期结果** | 操作正常，返回OK。同时通过show tables/describe table\_name命令查看操作均成功。 |
| **测试说明** |  |

4.1.7.3  本地文件系统导入数据

|  |  |
| --- | --- |
| **测试编号** |  |
| **测试项目** | Spark SQL功能测试 |
| **测试子项目** | 本地文件系统导入数据 |
| **测试目的** | 验证是否支持由本地文件系统导入数据 |
| **测试依据** |  |
| **预置条件** | 1．按照测试组网图搭建测试环境；  2．Spark正常运行； |
| **测试步骤** | 1. 创建本地文件/home/mr/test，内容如下：  a 1  c 3  e 5  2. 进入spark-sql命令行，创建表，以空格作为分隔符：  create table if not exists table\_name(name string,value int) row format delimited fields terminated by ' ';  3.导入数据：  load data local inpath '/home/mr/test' overwrite into table table\_name;  4.查询表：  select \* from table\_name;  5.删除表：  drop table table\_name; |
| **预期结果** | 1. 执行步骤1，成功创建文件/home/mr/test；  2. 执行步骤2，成功创建表table\_name；  3. 执行步骤3，返回值为OK。  4. 执行步骤4，显示内容为如下：  a 1  c 3  e 5  5.执行步骤5，成功删除表table\_name，通过show tables无法查看到相关表。 |
| **测试说明** |  |

4.1.7.4  数据查询操作

|  |  |
| --- | --- |
| **测试编号** |  |
| **测试项目** | Spark SQL功能测试 |
| **测试子项目** | 数据查询操作 |
| **测试目的** | 验证数据查询操作是否正常。 |
| **测试依据** |  |
| **预置条件** | 1．按照测试组网图搭建测试环境；  2．Spark正常运行；  3．SparkSQL中存在表，且有数据；  4．进入spark-sql命令行操作。 |
| **测试步骤** | 1. 查询table\_name表内容：  select \* from default.table\_name;  1.       基于分区查询表内容  SELECT \*  FROM table\_name WHERE [PARTITION ><=value]  2.       执行聚合函数查询语句  SELECT  COUNT(\*) FROM … WHERE … = ( SELECT  AVG(… ) FROM …WHERE … );  3.       执行join/group by/子查询等查询语句  CREATE TABLE ... AS SELECT  … FROM …  LEFT OUTER JOIN(SELECT  … FROM … GROUP BY …) AS … ON …; |
| **预期结果** | 1 2、数据查询操作正常，正确显示table\_name表内容。  3 4、能正确查询出数据。 |
| **测试说明** |  |

4.1.7.5  数据order排序

|  |  |
| --- | --- |
| **测试编号** |  |
| **测试项目** | Spark SQL功能测试 |
| **测试子项目** | 数据order排序 |
| **测试目的** | 验证数据order排序操作是否正常。 |
| **测试依据** |  |
| **预置条件** | 1．按照测试组网图搭建测试环境；  2．Spark正常运行；  3．SparkSQL中存在表，且有数据；  4．进入spark-sql命令行操作。 |
| **测试步骤** | 1. 将输出结果按某列进行升序、降序的排列  select \* from table\_name order by value asc;  2. 将输出结果按某列进行升序、降序的排列  select \* from table\_name order by value desc; |
| **预期结果** | 1. 执行步骤1，输出结果按照value列进行升序排序；  2. 执行步骤2，输出结果按照value列进行降序排序。 |
| **测试说明** |  |

4.1.7.6 WordCount性能

|  |  |
| --- | --- |
| **测试编号** |  |
| **测试项目** | WordCount性能 |
| **测试子项目** | WordCount性能 |
| **测试目的** | 测试Spark运行WordCount性能 |
| **测试依据** |  |
| **预置条件** | 1．  HDFS已经运行  2．  HDFS上准备500GB数据  3．  Spark Standalone集群已经启动 |
| **测试步骤** | 1．进入Spark安装的用户，切换到Spark主目录  2．执行bin/spark-shell  --master spark://[master\_hostname]:port  3．sc.textFile(“hdfs\_input\_dir\_path”).flatMap(\_.split(“”)).map(w => (w,1)).reduceByKey(\_ + \_).saveAsTextFile(“hdfs\_output\_dir\_path”)  4．再重复执行步骤3两次，记录平均值 |
| **预期结果** | 1．  记录下平均每节点的写入吞吐量  2．  记录下集群的写入吞吐量 |
| **测试说明** |  |

4.1.7.7 Terasort性能

|  |  |
| --- | --- |
| **测试编号** |  |
| **测试项目** | Terasort性能 |
| **测试子项目** | Terasort性能 |
| **测试目的** | 测试Spark的Terasort性能 |
| **测试依据** |  |
| **预置条件** | 1．  HDFS已经运行  2．  HDFS上准备好1TB数据  3．  Spark Standalone集群已经启动  4．  Terasort的jar包已准备好，比如放在本地路径：/home/mr/terasort\_jar/terasort.jar |
| **测试步骤** | 1．  进入Spark安装的用户，切换到Spark主目录  2．  执行bin/spark-submit  --master spark://[master\_hostname]:port  --class Terasort /home/mr/terasort\_jar/terasort.jar hdfs\_input\_data\_path hdfs\_output\_data\_path  3．  再重复执行步骤4两次，记录平均值 |
| **预期结果** | 1．  记录下平均每节点的写入吞吐量  2．  记录下集群的写入吞吐量 |
| **测试说明** |  |

4.1.8     Sqoop

4.1.8.1 Sqoop功能测试

|  |  |
| --- | --- |
| **测试编号** |  |
| **测试项目** | Sqoop功能测试 |
| **测试子项目** | sqoop支持从传统数据库向hadoop数据迁移 |
| **测试目的** | 验证sqoop支持从传统数据库向hadoop数据迁移 |
| **测试依据** |  |
| **预置条件** | 1.系统正常运行； |
| **测试步骤** | 1、传统的mysql数据迁移测试  验证从传统的mysql数据库导出数据到hadoop的hdfs  验证从hadoop的hdfs目录导入数据到mysql    2、传统的db2数据迁移测试  验证从传统的db2数据库导出数据到hadoop的hdfs  验证从hadoop的hdfs目录导入数据到db2    3、传统的oracle数据迁移测试  验证从传统的oracle数据库导出数据到hadoop的hdfs  验证从hadoop的hdfs目录导入数据到oracle |
| **预期结果** | Sqoop支持从传统的mysql、db2、oracle数据库导出数据到hdfs；  Sqoop支持从hdfs导入数据到传统的mysql、db2、oracle数据库。 |
| **测试结果** |  |

4.1.9     Kerberos

4.1.9.1  Kerberos安全测试

|  |  |
| --- | --- |
| **测试编号** |  |
| **测试项目** | Kerberos安全性测试 |
| **测试子项目** | 提供kerberos管理的能力 |
| **测试目的** | 提供kerberos管理的能力 |
| **测试依据** |  |
| **预置条件** | 1．按照测试组网图搭建测试环境；  2．各系统运行正常； |
| **测试步骤** | 1.配置各服务的kerberos  2.启用kereros功能  3.禁用kereros功能 |
| **预期结果** | 对zk，hdfs，yarn，hbase和hive启用kerberos功能后，未进行kerberos认证的机器和用户不能访问集群的zk节点，hdfs目录，不能向集群提交mr作业，不能访问hbase，不能使用hive。进行kerberos认证的机器和用户可以访问集群的zk节点，hdfs目录，向集群提交mr作业，访问hbase和hive。  禁用kerberos功能后，对机器和用户不做限制。 |
| **测试说明** |  |

4.1.10  Flume

4.1.10.1                Flume功能测试

|  |  |
| --- | --- |
| **测试编号** |  |
| **测试项目** | Flume功能测试 |
| **测试子项目** | Flume功能测试 |
| **测试目的** | 测试flume基本功能 |
| **测试依据** |  |
| **预置条件** | 1．  HDFS已经运行  2．  Flume已经运行 |
| **测试步骤** | 1．配置conf  配置flume-env.sh文件，配置如下：  export JAVA\_HOME=/usr/java/jdk  export JAVA\_OPTS="-Xms100m -Xmx2000m -Dcom.sun.management.jmxremote"  #FLUME\_CLASSPATH=""  2.配置flume.server如下  # Name the components on this agent  a1.sources = r1  a1.sinks = k1  a1.channels = c1  # Describe/configure the source  a1.sources.r1.type = avro  a1.sources.r1.bind = localhost  a1.sources.r1.port = 44444  # Describe the sink  a1.sinks.k1.type = logger  # Use a channel which buffers events in memory  a1.channels.c1.type = memory  a1.channels.c1.capacity = 1000  a1.channels.c1.transactionCapacity = 100  # Bind the source and sink to the channel  a1.sources.r1.channels = c1  a1.sinks.k1.channel = c1  3.启动flume  在flume的bin目录下，执行命令如下：  ./flume-ng agent -c ../conf -f ../conf/flume.server -n a1  4.数据导入验证  在flume的bin目录下，执行命令如下：  ./flume-ng avro-client -H localhost -p 44444 -F /home/mr/flume/CHANGELOG  5.验证结果 |
| **预期结果** | Flume功能正常 |
| **测试说明** |  |

4.1.11  Solr

4.1.11.1                Solr单条索引上传查询

|  |  |
| --- | --- |
| **测试编号** |  |
| **测试项目** | Solr功能测试 |
| **测试子项目** | 单条索引上传查询 |
| **测试目的** | 测试solr单条索引上传查询功能 |
| **测试依据** |  |
| **预置条件** | Solr进程启动正常 |
| **测试步骤** | I1.命令行模式创建实例  I2.通过管理界面提交单条索引数据记录  3 通过管理界面能够查询上传的单条索引记录 |
| **预期结果** |  |
| **测试结果** |  |

4.1.11.2                Solr批量索引上传查询

|  |  |
| --- | --- |
| **测试编号** |  |
| **测试项目** | Solr功能测试 |
| **测试子项目** | 批量索引上传查询 |
| **测试目的** | 测试批量索引上传查询功能 |
| **测试依据** |  |
| **预置条件** | Solr进程启动正常 |
| **测试步骤** | I1.命令行模式创建实例  I2.命令行提交多条索引数据记录  3 通过管理界面能够查询上传的多条索引记录 |
| **预期结果** |  |
| **测试结果** |  |

5             管理能力测试

5.1        安装部署

|  |  |
| --- | --- |
| **测试编号** |  |
| **测试项目** | 自动安装部署 |
| **测试子项目** | 自动安装部署 |
| **测试目的** | 检测系统是否支持集群的自动安装和部署 |
| **测试依据** |  |
| **预置条件** | 1．系统运行正常。  2．操作维护系统与集群所在资源之间通信正常。 |
| **测试步骤** | 1．提供一系列服务器IP地址，进行批量自动化安装。  2．能够集中查看安装日志，方便的查看关键结果，成功或是失败  3．验证可通过设定不同引擎和指定节点方式，以接口、脚本提供自动化安装能力 |
| **预期结果** | 自动安装成功，进入待运行状态。等待运维人员准予运行。 |
| **测试结果** |  |

5.2           集群统一管理工具

|  |  |
| --- | --- |
| **测试编号** |  |
| **测试项目** | 集群统一管理工具 |
| **测试子项目** | 集群统一管理工具 |
| **测试目的** | 验证Hadoop是否提供统一的管理功能 |
| **测试依据** |  |
| **预置条件** | 操作维护系统已安装并运行正常； |
| **测试步骤** | 1、   使用超级管理员帐号登录到集群管理系统；  2、   打开并浏览Hadoop服务页面  3、 打开并浏览集群主机页面 |
| **预期结果** | 系统提供统一的WEB管理操作入口；  能通过界面对主机、服务进行日常的运维、管理等操作。  1，Hadoop服务管理可提供以下功能：  l  可查看各个自有和通用服务的运行状态，健康状态及配置状态，状态刷新周期可配置；  l  可对自有和通用组建服务进行增加，启动，停止，重启及删除操作；  l  可对集群进行启动，停止，升级，卸载，同步配置，导出配置及导出安装模板操作；  2，集群主机管理可提供以下功能：  l  可显示集群中主机的名称，IP地址，所在机架，网络读写速率，健康状态，CPU，内存及磁盘使用率，并可对上述字段进行排序；  l  可对集群中的主机进行启动所有角色服务，停止所有角色服务，设置机架信息，删除主机及强制删除主机，同时可增加主机；  可提供根据主机名称，机架名称，CPU使用率范围，主机IP，主机健康状态，内存使用率范围，磁盘使用率范围进行主机查询； |
| **测试结果** |  |

5.3           日志管理

|  |  |
| --- | --- |
| **测试编号** |  |
| **测试项目** | 日志管理 |
| **测试子项目** | 日志管理 |
| **测试目的** | 检测系统是否支持对用户操作日志和系统日志的管理 |
| **测试依据** |  |
| **预置条件** | 1．系统运行正常。  2．浏览器所在客户端与操作维护系统、操作维护系统与集群所在资源之间通信正常。  3．至少有一个有管理权限的用户。  4．模拟一些用户操作、系统操作、系统异常。 |
| **测试步骤** | 1．图形用户界面须管理员经密码确认方可登录，进入日志查询管理界面；  2．按照操作结果、操作类型、操作时间、操作用户等条件进行查询；  3．查看查询结果，并查看日志详情；  4．对指定时间范围的用户日志、系统日志进行删除操作。 |
| **预期结果** | 1．支持查看不同用户和不同组建的日志；  2．可查看用户操作日志，用户操作日志包括对用户登入、登出、权限变更、IP、操作类型、操作内容、操作时间等内容的记录与展示。  3．支持系统日志查看，系统日志包括对系统启动、关闭、用户增减、权限变更、性能异常、功能异常、硬件异常、网络异常、危险操作等内容的记录与展示； |
| **测试结果** | 查看系统的安全日志，显示系统登录信息，包括登陆成功、失败以及注销等信息  查看系统的操作日志，显示用户对Hive、HBase、YARN、HDFS等组件执行增删改的操作记录  查看系统的审计日志，可通过系统的审计日志，对指定时间段、对指定组件、对指定主机上保存的系统运行日志进行收集 |

5.4           性能监测

|  |  |
| --- | --- |
| **测试编号** |  |
| **测试项目** | 性能监测 |
| **测试子项目** | 性能监测 |
| **测试目的** | 检测系统是否支持对系统级性能指标进行管理和展示 |
| **测试依据** |  |
| **预置条件** | 1．系统运行正常。  2．浏览器所在客户端与操作维护系统、操作维护系统与集群所在资源之间通信正常。  3．至少有一个有管理权限的用户。 |
| **测试步骤** | 1．图形用户界面须管理员经密码确认方可登录，进入性能监测管理界面；  2．查看系统整体、通用组建、自有组建的实时性能变化情况  3．查询一段时间范围内的历史监控数据信息，并对历史监控数据信息进行删除操作。  4. 通过图形界面，查看目前压力对应的应用或用户。 |
| **预期结果** | 1．能查看服务器性能图表和图形；  2．能查看系统历史性能数据；  3．能删除历史性能数据。 |
| **测试说明** | 登录集群主页，查看整个集群信息    查看集群中所有主机健康状态，包括主机名、IP、机架、健康状态、安装服务情况  1、支持集群中服务的状态及性能指标的监控  l  HDFS  HDFS容量统计；/HDFS文件（所有文件及文件夹数）/HDFS块信息统计/HDFS连接数/HDFS DataNode节点数/NameNode内存使用情况    l  Hbase  RegionServer统计信息/所有RegionServer每秒请求数/RegionServer队列（压缩/Flush）统计信息/ Hbase Region数统计      可新增统计指标图    l  Hive  Hive Server任务使用逻辑CPU数统计/HiveServer任务内存使用统计/HiveServer请求数统计/HiveServer会话统计/Hive数据空间使用统计/执行HQL语句总数量统计/执行HQL语句成功的百分比    2、集群中主机的监控条目，主机CPU利用率，内存使用率，主机网络信息，主机磁盘分区及其使用率监控    主机节点具体指标统计图 |

5.5           故障管理

|  |  |
| --- | --- |
| **测试编号** |  |
| **测试项目** | 故障管理 |
| **测试子项目** | 故障管理 |
| **测试目的** | 验证大数据平台故障告警功能，在故障发生时能产生相应告警 |
| **测试依据** |  |
| **预置条件** | 1．按照测试组网图搭建测试环境；  2．系统运行正常；  3．使用管理员账号登录存储管理操作维护中心。 |
| **测试步骤** | 1、 在系统运行过程中，根据规范中的告警指标，对网络配置进行手工修改；对数据的读写权限进行修改。  2、 查看存储操作维护系统“故障告警系统”  3、 手工修复产生的故障  4、 再次查看故障告警信息。  5、 进入告警查询页面，分别按照告警级别/告警码对当前告警和历史告警分别进行查询操作。  6、模拟节点故障，对节点剔除和加入  7、模拟性能故障，对应用进程挂起和清理。 |
| **预期结果** | 1．在当前告警中出现刚才模拟的告警提示，告警级别、告警内容、发生时间、告警处理建议等信息展示正确。  2．告警消除后，前面产生的告警在当前告警中删除，在历史查询中能够查询到此条告警，恢复时间正确  3．当前告警、历史告警按级别和按告警码查询结果展示正确 |
| **测试说明** | 查看系统告警信息    对服务角色进行过滤，查看告警条目给出相应解决方案    对需要清除的告警条目进行手工清除，系统给出告警的处理建议    按照告警级别、发生的时间范围、告警类型、告警所属的服务、角色等条件进行过滤查询    可根据主机，起止时间等查询事件信息  查看历史告警信息    对告警阈值进行设置，超过阈值会自动上报告警信息 |

5.6           数据管理能力

|  |  |
| --- | --- |
| **测试编号** | 8.7 |
| **测试项目** | 数据管理能力 |
| **测试子项目** | 对数据文件进行备份和恢复  Manager页面管理数据的配置 |
| **测试目的** | 数据管理能力 |
| **测试依据** |  |
| **预置条件** | 1．按照测试组网图搭建测试环境；  2．各系统运行正常； |
| **测试步骤** | 1.     HDFS的备份与恢复  2.     对HDFS的数据配置进行管理 |
| **预期结果** | 可以对HDFS的数据进行配置操作，对HDFS文件能够进备份和恢复 |
| **测试说明** | 1.对HDFS的文件能够进行备份和恢复  在HDFS中创建目录/backup/a、/backup/b，对目录/backup/a进行全量备份，观察备份是否成功    将/backup目录下的a、b目录同时删除，使用备份的数据进行恢复，观察/backup/a是否恢复成功，内容是否与删除前的一致，观察/backup/b是不会被恢复    2.Manager页面可配置HDFS的副本参数，数据的存储目录等，对数据管理 |

5.7           角色管理能力

|  |  |
| --- | --- |
| **测试编号** | 8.8 |
| **测试项目** | Manager管理角色的功能 |
| **测试子项目** | Manager管理角色的功能 |
| **测试目的** | Manager管理角色的功能 |
| **测试依据** |  |
| **预置条件** | 1．按照测试组网图搭建测试环境；  2．各系统运行正常； |
| **测试步骤** | 登陆到manager管理平台，进入系统  1.在线修改当前用户的权限  2.在线创建用户  3.在线删除，正在运行应用的用户。 |
| **预期结果** | Manager提供角色管理的功能 |
| **测试说明** | manager页面提供角色管理等功能，方便集群管理角色功能。用户登录方式可与行内单点登陆系统对接。    提供多种的权限策略  对权限管理可自定义多种角色  用户可分为多级别管理，并可对用户自有权限赋予其他用户。 |

5.8           多集群部署能力

|  |  |
| --- | --- |
| **测试编号** | 8.9 |
| **测试项目** | 多集群管理能力 |
| **测试子项目** | 多集群管理能力 |
| **测试目的** | 测试多集群管理能力 |
| **测试依据** |  |
| **预置条件** | 1．按照测试组网图搭建测试环境；  2．各系统运行正常； |
| **测试步骤** | 1.     登陆manager管理平台，点击“添加”新增多个集群  2.     在主页上查看多个集群的信息，可对多个集群进行管理 |
| **预期结果** | Manager平台能够进行多个集群的管理 |
| **测试说明** | 新增多个集群，可以对多个集群进行同时管理和监控  点击Manager界面安装-安装集群，然后点击添加，增加新的集群名称  在“主页”可查看多个集群的信息，可同时管理多个集群的能力 |