

二 从 Hadoop 框架讨论大数据生态

2.1 Hadoop 是什么

- 1) Hadoop 是一个由 Apache 基金会所开发的分布式系统基础架构。
- 2) 主要解决，海量数据的存储和海量数据的分析计算问题。
- 3) 广义上来说，HADOOP 通常是指一个更广泛的概念——HADOOP 生态圈。



2.2 Hadoop 发展历史

- 1) Lucene 是 Doug Cutting 开创的开源软件，用 java 书写代码，实现与 Google 类似的全文搜索功能，它提供了全文检索引擎的架构，包括完整的查询引擎和索引引擎
- 2) 2001 年年底成为 Apache 基金会的一个子项目
- 3) 对于大数量的场景，Lucene 面对与 Google 同样的困难
- 4) 学习和模仿 Google 解决这些问题的办法：微型版 Nutch
- 5) 可以说 Google 是 hadoop 的思想之源(Google 在大数据方面的三篇论文)

GFS --->HDFS

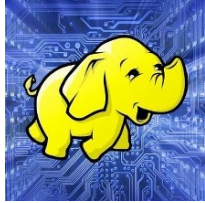
Map-Reduce --->MR

BigTable --->Hbase

- 6)2003-2004 年,Google 公开了部分 GFS 和 Mapreduce 思想的细节,以此为基础 Doug Cutting 等人用了 2 年业余时间实现了 DFS 和 Mapreduce 机制，使 Nutch 性能飙升
- 7)2005 年 Hadoop 作为 Lucene 的子项目 Nutch 的一部分正式引入 Apache 基金会。2006 年 3 月份,Map-Reduce 和 Nutch Distributed File System (NDFS) 分别被纳入称为 Hadoop 的项

目中

8) 名字来源于 Doug Cutting 儿子的玩具大象



9) Hadoop 就此诞生并迅速发展，标志这云计算时代来临

2.3 Hadoop 三大发行版本

Hadoop 三大发行版本：Apache、Cloudera、Hortonworks。

Apache 版本最原始（最基础）的版本，对于入门学习最好。

Cloudera 在大型互联网企业中用的较多。

Hortonworks 文档较好。

1) Apache Hadoop

官网地址：<http://hadoop.apache.org/releases.html>

下载地址：<https://archive.apache.org/dist/hadoop/common/>

2) Cloudera Hadoop

官网地址：<https://www.cloudera.com/downloads/cdh/5-10-0.html>

下载地址：<http://archive-primary.cloudera.com/cdh5/cdh/5/>

(1) 2008 年成立的 Cloudera 是最早将 Hadoop 商用的公司，为合作伙伴提供 Hadoop 的商用解决方案，主要是包括支持、咨询服务、培训。

(2) 2009 年 Hadoop 的创始人 Doug Cutting 也加盟 Cloudera 公司。Cloudera 产品主要为 CDH, Cloudera Manager, Cloudera Support

(3) CDH 是 Cloudera 的 Hadoop 发行版，完全开源，比 Apache Hadoop 在兼容性，安全性，稳定性上有所增强。

(4) Cloudera Manager 是集群的软件分发及管理监控平台，可以在几个小时内部署好一个 Hadoop 集群，并对集群的节点及服务进行实时监控。Cloudera Support 即是对 Hadoop 的技术支持。

(5) Cloudera 的标价为每年每个节点 4000 美元。Cloudera 开发并贡献了可实时处理大数据的 Impala 项目。

3) Hortonworks Hadoop

官网地址: <https://hortonworks.com/products/data-center/hdp/>

下载地址: <https://hortonworks.com/downloads/#data-platform>

(1) 2011 年成立的 Hortonworks 是雅虎与硅谷风投公司 Benchmark Capital 合资组建。

(2) 公司成立之初就吸纳了大约 25 名至 30 名专门研究 Hadoop 的雅虎工程师, 上述工程师均在 2005 年开始协助雅虎开发 Hadoop, 贡献了 Hadoop 80% 的代码。

(3) 雅虎工程副总裁、雅虎 Hadoop 开发团队负责人 Eric Baldeschwieler 出任 Hortonworks 的首席执行官。

(4) Hortonworks 的主打产品是 Hortonworks Data Platform (HDP), 也同样是 100% 开源的产品, HDP 除常见的项目外还包括了 Ambari, 一款开源的安装和管理系统。

(5) HCatalog, 一个元数据管理系统, HCatalog 现已集成到 Facebook 开源的 Hive 中。Hortonworks 的 Stinger 开创性的极大的优化了 Hive 项目。Hortonworks 为入门提供了一个非常好的, 易于使用的沙盒。

(6) Hortonworks 开发了很多增强特性并提交至核心主干, 这使得 Apache Hadoop 能够在包括 Window Server 和 Windows Azure 在内的 microsoft Windows 平台上本地运行。定价以集群为基础, 每 10 个节点每年为 12500 美元。

2.4 Hadoop 的优势

- 1) 高可靠性: Hadoop 底层维护多个数据副本, 所以即使 Hadoop 某个计算元素或存储出现故障, 也不会导致数据的丢失。
- 2) 高扩展性: 在集群间分配任务数据, 可方便的扩展数以千计的节点。
- 3) 高效性: 在 MapReduce 的思想下, Hadoop 是并行工作的, 以加快任务处理速度。
- 4) 高容错性: 能够自动将失败的任务重新分配。

2.5 Hadoop 组成

在 Hadoop 1.x 时代, Hadoop 中的 MapReduce 同时处理业务逻辑运算和资源的调度, 耦合性较大。

Hadoop1.x组成

Common（辅助工具）

MapReduce（资源调度+计算）

HDFS（数据存储）

1) Hadoop HDFS：一个高可靠、高吞吐量的分布式文件系统。

2) Hadoop MapReduce：一个分布式的资源调度和离线并行计算框架。

3) Hadoop Common：支持其他模块的工具模块（Configuration、RPC、序列化机制、日志操作）。

在 Hadoop2.x 时代，增加了 Yarn。Yarn 只负责资源的调度，MapReduce 只负责运算。

Hadoop2.x组成

MapReduce（计算）

Yarn（资源调度）

HDFS（数据存储）

C
o
m
m
o
n
（
辅
助
工
具
）

1) Hadoop HDFS：一个高可靠、高吞吐量的分布式文件系统。

2) Hadoop YARN：作业调度与集群资源管理的框架。

3) Hadoop MapReduce：一个分布式的离线并行计算框架。

4) Hadoop Common：支持其他模块的工具模块（Configuration、RPC、序列化机制、日志操作）。

2.5.1 HDFS 架构概述



HDFS架构概述

1) NameNode (nn): 存储文件的元数据, 如文件名, 文件目录结构, 文件属性 (生成时间、副本数、文件权限), 以及每个文件的块列表和块所在的DataNode等。




2) DataNode(dn): 在本地文件系统存储文件块数据, 以及块数据的校验和。



3) Secondary NameNode(2nn): 用来监控HDFS状态的辅助后台程序, 每隔一段时间获取HDFS元数据的快照。

2.5.2 YARN 架构概述



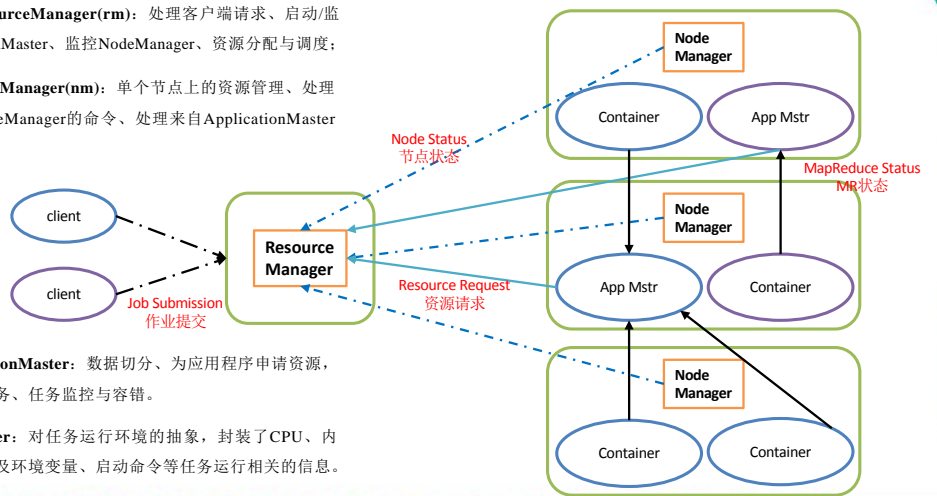
YARN架构

1) Resource Manager(rm): 处理客户端请求、启动/监控ApplicationMaster、监控NodeManager、资源分配与调度;

2) NodeManager(nm): 单个节点上的资源管理、处理来自Resource Manager的命令、处理来自ApplicationMaster的命令;

3) ApplicationMaster: 数据切分、为应用程序申请资源, 并分配给内部任务、任务监控与容错。

4) Container: 对任务运行环境的抽象, 封装了CPU、内存等多维资源以及环境变量、启动命令等任务运行相关的信息。

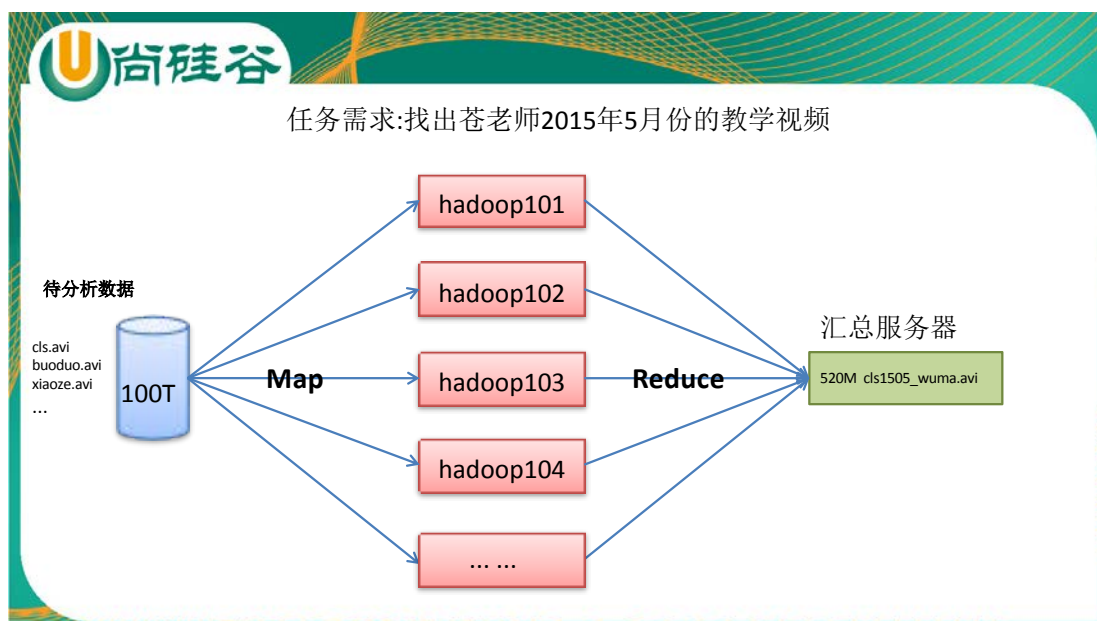


The diagram illustrates the YARN architecture. On the left, two 'client' nodes send 'Job Submission' (作业提交) to the 'Resource Manager'. The 'Resource Manager' is connected to three 'Node Manager' nodes. Each 'Node Manager' contains an 'App Mstr' and 'Container' components. Arrows show 'Node Status' (节点状态) being sent from 'Node Managers' to the 'Resource Manager', and 'Resource Request' (资源请求) being sent from 'App Mstr' to the 'Resource Manager'. Additionally, 'MapReduce Status' (MR状态) is sent from 'App Mstr' to the 'Node Manager'.

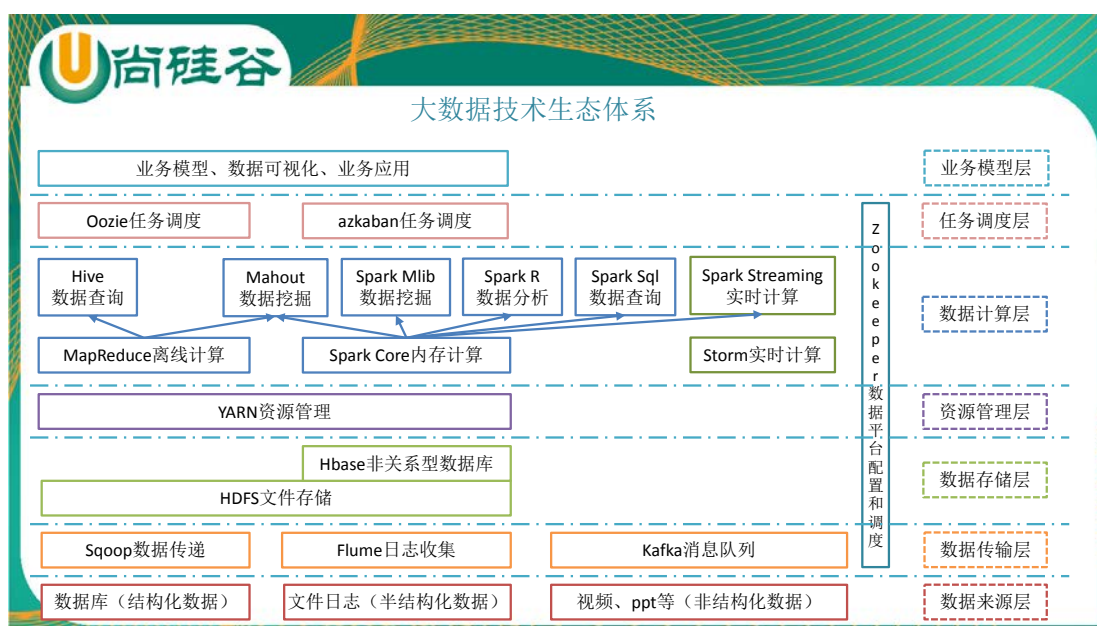
2.5.3 MapReduce 架构概述

MapReduce 将计算过程分为两个阶段: Map 和 Reduce

- 1) Map 阶段并行处理输入数据
- 2) Reduce 阶段对 Map 结果进行汇总



2.6 大数据技术生态体系



图中涉及的技术名词解释如下:

- 1) Sqoop: sqoop 是一款开源的工具, 主要用于在 Hadoop(Hive)与传统的数据库(mysql)间进行数据的传递, 可以将一个关系型数据库(例如: MySQL, Oracle 等)中的数据导进到 Hadoop 的 HDFS 中, 也可以将 HDFS 的数据导进到关系型数据库中。
- 2) Flume: Flume 是 Cloudera 提供的一个高可用的, 高可靠的, 分布式的海量日志采集、聚合和传输的系统, Flume 支持在日志系统中定制各类数据发送方, 用于收集数据; 同时, Flume 提供对数据进行简单处理, 并写到各种数据接受方(可定制)的能力。
- 3) Kafka: Kafka 是一种高吞吐量的分布式发布订阅消息系统, 有如下特性:

(1) 通过 $O(1)$ 的磁盘数据结构提供消息的持久化，这种结构对于即使数以 TB 的消息存储也能够保持长时间的稳定性能。

(2) 高吞吐量：即使是非常普通的硬件 **Kafka** 也可以支持每秒数百万的消息。

(3) 支持通过 **Kafka** 服务器和消费机集群来分区消息。

(4) 支持 **Hadoop** 并行数据加载。

4) **Storm**: **Storm** 为分布式实时计算提供了一组通用原语，可被用于“流处理”之中，实时处理消息并更新数据库。这是管理队列及工作者集群的另一种方式。**Storm** 也可被用于“连续计算” (continuous computation)，对数据流做连续查询，在计算时就将结果以流的形式输出给用户。

5) **Spark**: **Spark** 是当前最流行的开源大数据内存计算框架。可以基于 **Hadoop** 上存储的大数据进行计算。

6) **Oozie**: **Oozie** 是一个管理 **Hadoop** 作业 (job) 的工作流程调度管理系统。**Oozie** 协调作业就是通过时间 (频率) 和有效数据触发当前的 **Oozie** 工作流程。

7) **Hbase**: **HBase** 是一个分布式的、面向列的开源数据库。**HBase** 不同于一般的关系数据库，它是一个适合于非结构化数据存储的数据库。

8) **Hive**: **hive** 是基于 **Hadoop** 的一个数据仓库工具，可以将结构化的数据文件映射为一张数据库表，并提供简单的 **sql** 查询功能，可以将 **sql** 语句转换为 **MapReduce** 任务进行运行。其优点是学习成本低，可以通过类 **SQL** 语句快速实现简单的 **MapReduce** 统计，不必开发专门的 **MapReduce** 应用，十分适合数据仓库的统计分析。

10) **R 语言**: **R** 是用于统计分析、绘图的语言和操作环境。**R** 是属于 **GNU** 系统的一个自由、免费、源代码开放的软件，它是一个用于统计计算和统计制图的优秀工具。

11) **Mahout**:

Apache Mahout 是个可扩展的机器学习和数据挖掘库，当前 **Mahout** 支持主要的 4 个用例：

推荐挖掘：搜集用户动作并以此给用户推荐可能喜欢的事物。

聚集：收集文件并进行相关文件分组。

分类：从现有的分类文档中学习，寻找文档中的相似特征，并为无标签的文档进行正确的归类。

频繁项集挖掘：将一组项分组，并识别哪些个别项会经常一起出现。

12) **ZooKeeper**: **Zookeeper** 是 **Google** 的 **Chubby** 一个开源的实现。它是一个针对大型分布

式系统的可靠协调系统，提供的功能包括：配置维护、名字服务、 分布式同步、组服务等。
ZooKeeper 的目标就是封装好复杂易出错的关键服务，将简单易用的接口和性能高效、功能稳定的系统提供给用户。

2.7 推荐系统框架图

