

Hadoop 生态核心组件介绍

在了解 Hadoop 生态系统及其组件之前，我们首先了解一下 Hadoop 的三大组件，即 HDFS、MapReduce、YARN，它们共同构成了 Hadoop 分布式计算框架的 **核心**。

HDFS (Hadoop Distributed File System)：HDFS 是 Hadoop 的 **分布式文件系统**，它是将大规模数据分散存储在多个节点上的基础。HDFS 主要负责数据的存储和管理，可以将大数据集分成多个数据块，并将这些数据块分配到不同的计算节点上存储，提高数据的可靠性和处理效率。

MapReduce：MapReduce 是 Hadoop 的 **分布式计算框架**，它提供了一种简单的编程模型，通过将大规模数据分解成多个小任务并行处理，可以大大提高数据处理的效率。MapReduce 模型包括 Map 和 Reduce 两个阶段，其中 Map 阶段将数据分解成多个小块进行处理，Reduce 阶段将处理结果合并。

YARN (Yet Another Resource Negotiator)：YARN 是 Hadoop 的 **资源管理器**，它负责为多个应用程序分配和管理计算资源，可以有效地提高计算资源的利用率。YARN 可以将**集群**中的计算资源划分为多个容器，为不同的应用程序提供适当的资源，并监控和管理各个应用程序的运行状态。



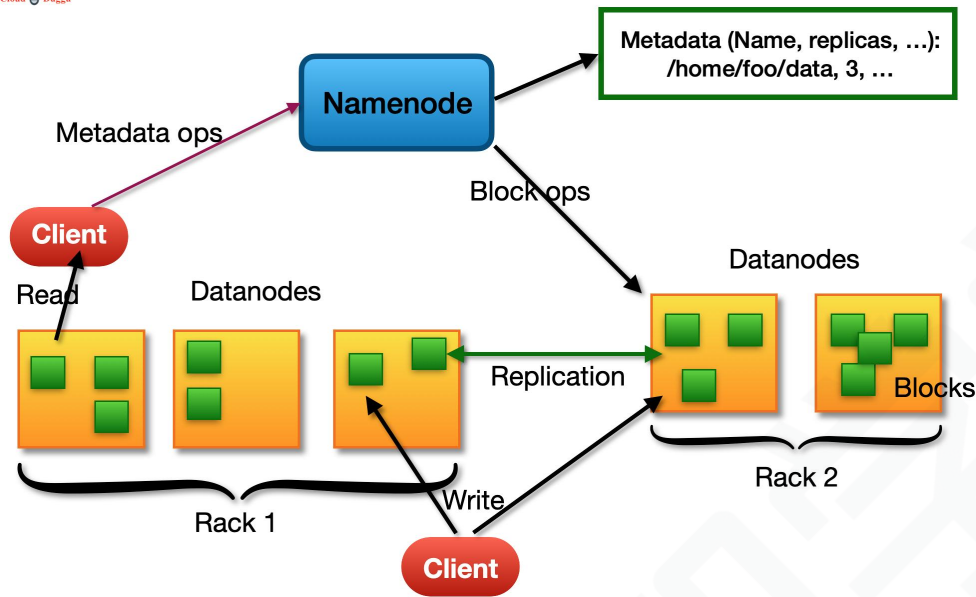
1.HDFS

HDFS 是 Hadoop 的分布式文件系统,旨在在廉价硬件上存储大型文件。它具有高度容错能力,并为应用程序提供高吞吐量。HDFS 最适合那些拥有非常大数据集的应用程序。

Hadoop HDFS 文件系统提供 Master 和 Slave 架构。主节点运行 Namenode 守护进程,从节点运行 Datanode 守护进程。

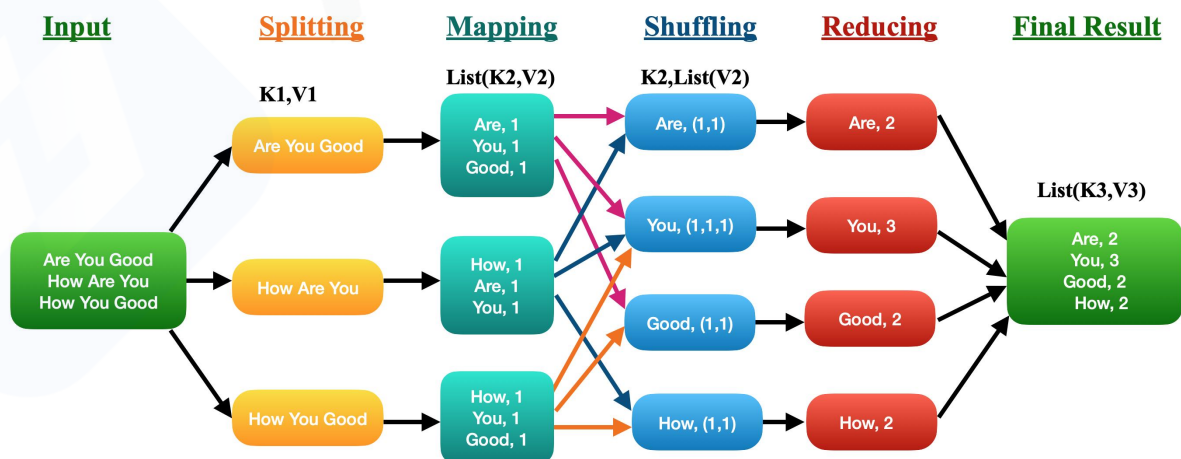


HDFS Architecture



2.MapReduce

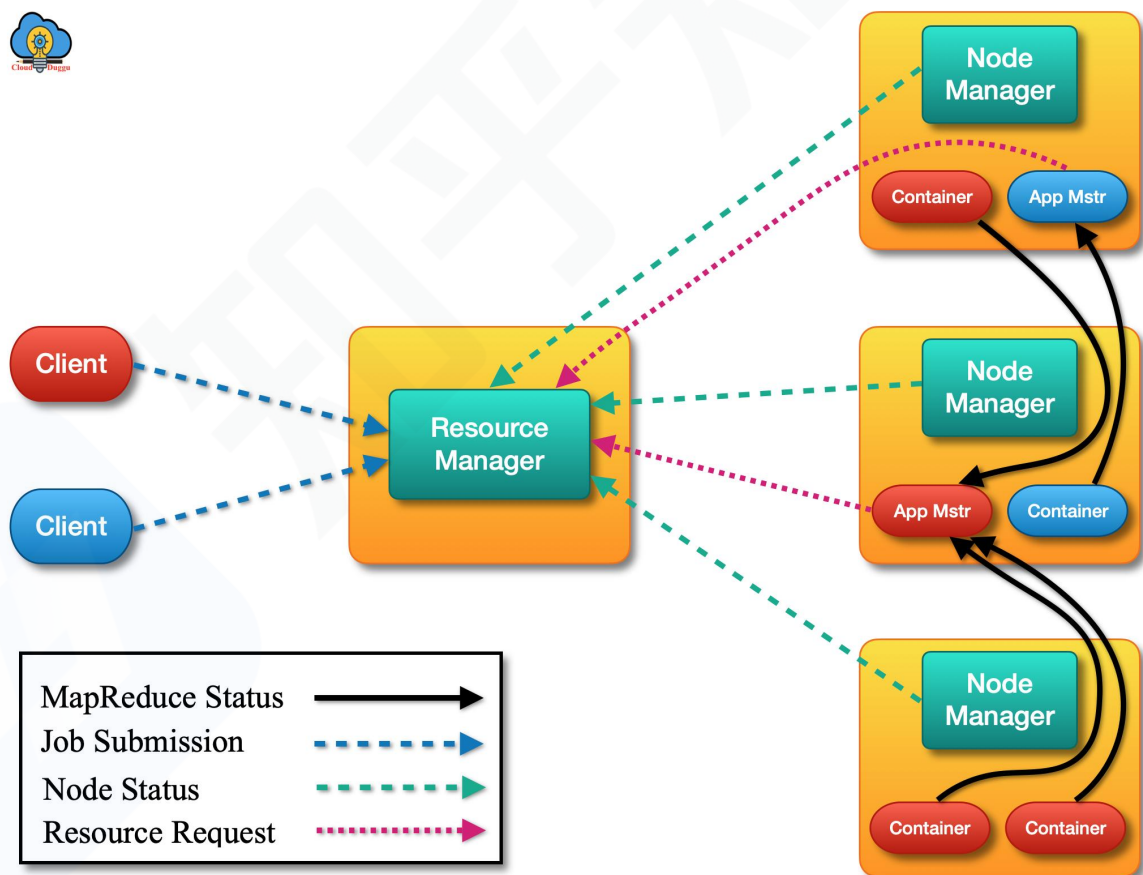
MapReduce 是 Hadoop 的数据处理层，它将任务分成小块，并将这些小块分配给通过网络连接的许多机器，并将所有事件组装成最后的事件数据集。MapReduce 所需的基本细节是键值对。所有数据，无论是否结构化，在通过 MapReduce 模型传递之前都需要转换为键值对。在 MapReduce 框架中，处理单元被移至数据，而不是将数据移至处理单元。



3.YARN

YARN 代表 **Yet Another Resource Negotiator**，它是 Hadoop 集群的资源管理器。YARN 用于实现 Hadoop 集群中的资源管理和作业调度。YARN 的主要思想是将作业调度和资源管理拆分到各个进程中进行操作。

YARN 提供了两个守护进程；第一个称为资源管理器（**Resource Manager**），第二个称为节点管理器（**Node Manager**）。这两个组件都用于处理 YARN 中的数据计算。资源管理器运行在 Hadoop 集群的主节点上，并协商所有应用程序中的资源，而节点管理器托管在所有从节点上。节点管理器的职责是监视容器、资源使用情况（例如 CPU、内存、磁盘和网络）并向资源管理器提供详细信息。



4.Hive

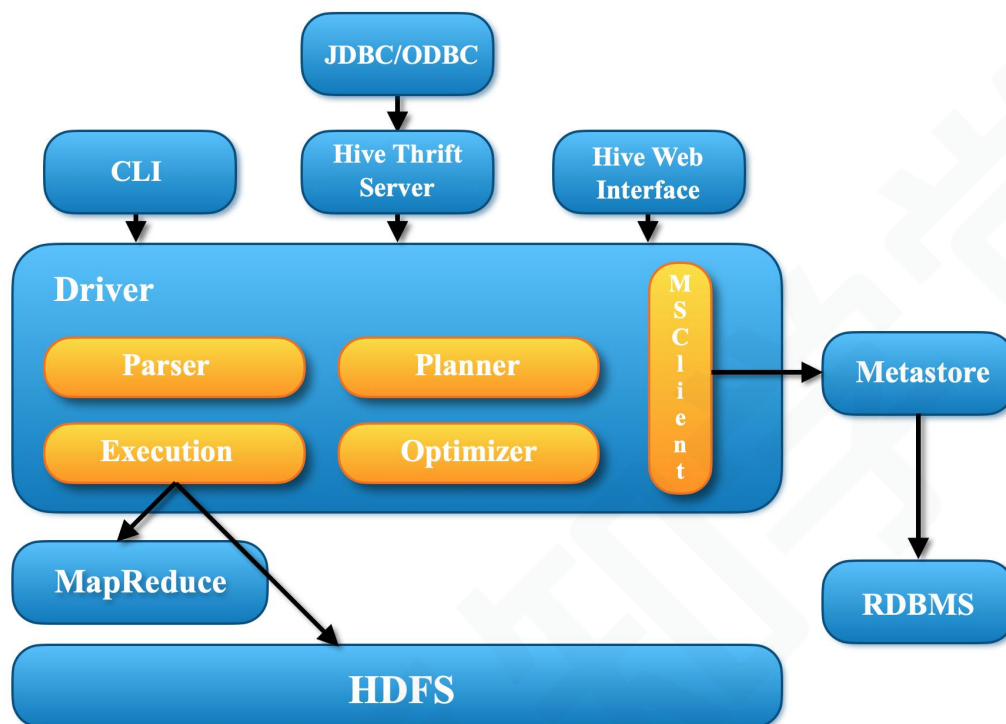
Hive 是 Hadoop 的 **数据仓库** 项目。Hive 旨在促进非正式数据汇总、即席查询和大量数据的解释。借助 HiveQL，用户可以对 HDFS 中的数据集中存储执行即席查询，并使用该数据进行进一步分析。Hive 还支持自定义的用户定义函数，用户可以使用这些函数来执行自定义分析。

让我们了解 Apache Hive 如何处理 SQL 查询：

- 用户将使用命令行或 Web UI 向驱动程序（例如 ODBC / JDBC）提交查询。
- 驱动程序将借助查询编译器来解析查询以检查语法 / 查询计划。
- 编译器将向元数据数据库发送元数据请求。
- 作为响应，Metastore 将向编译器提供元数据。
- 现在编译器的任务是验证规范并将计划重新发送给驱动程序。
- 现在驱动程序将向执行引擎发送执行计划。
- 该程序将作为映射缩减作业执行。执行引擎将作业发送到名称节点作业跟踪器，并为该作业分配一个存在于数据节点中的任务跟踪器，并在此处执行查询。
- 查询执行后，执行引擎将从数据节点接收结果。
- 执行引擎将结果值发送给驱动程序。
- 驱动程序会将结果发送到 Hive 接口（用户）。



Apache Hive Architecture

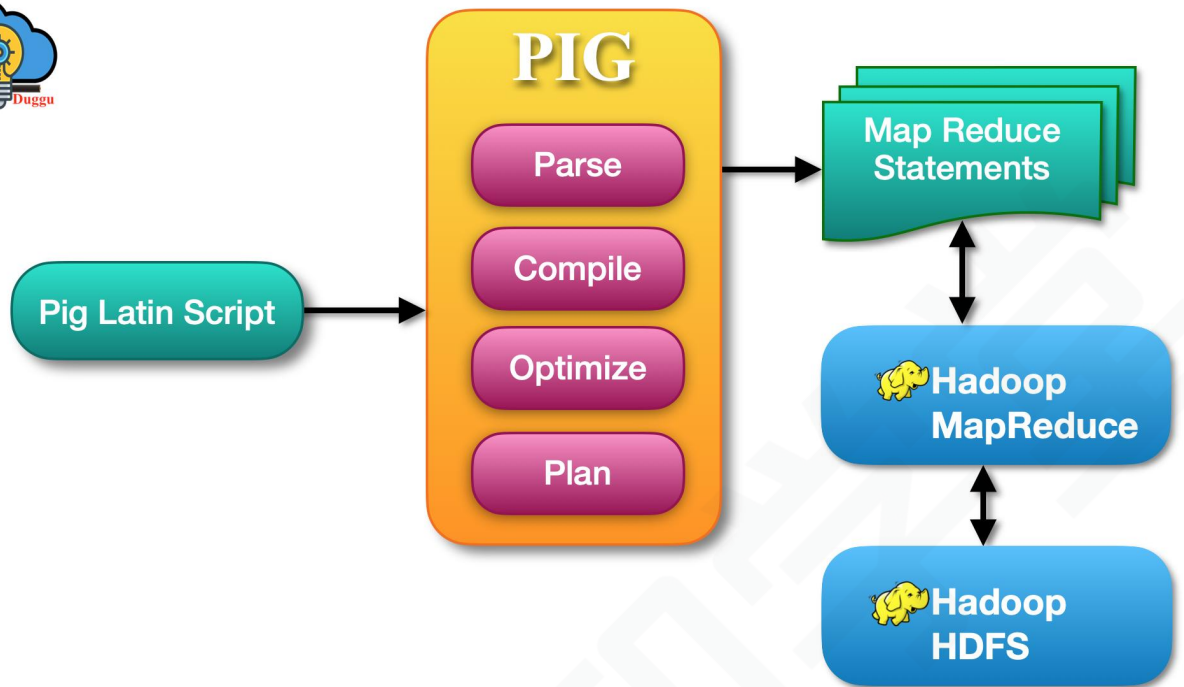


5.Pig

Pig 由 Yahoo 开发，用于分析存储在 Hadoop HDFS 中的大数据。Pig 提供了一个分析海量数据集的平台，该平台由用于通信数据分析应用程序的高级语言组成，并与用于评估这些程序的基础设施相链接。

Pig 具有以下关键属性：

- **优化机会**：Pig 提供了查询优化，帮助用户专注于意义而不是效率。
- **可扩展性**：Pig 提供了创建用户定义函数以进行特殊用途处理的功能。

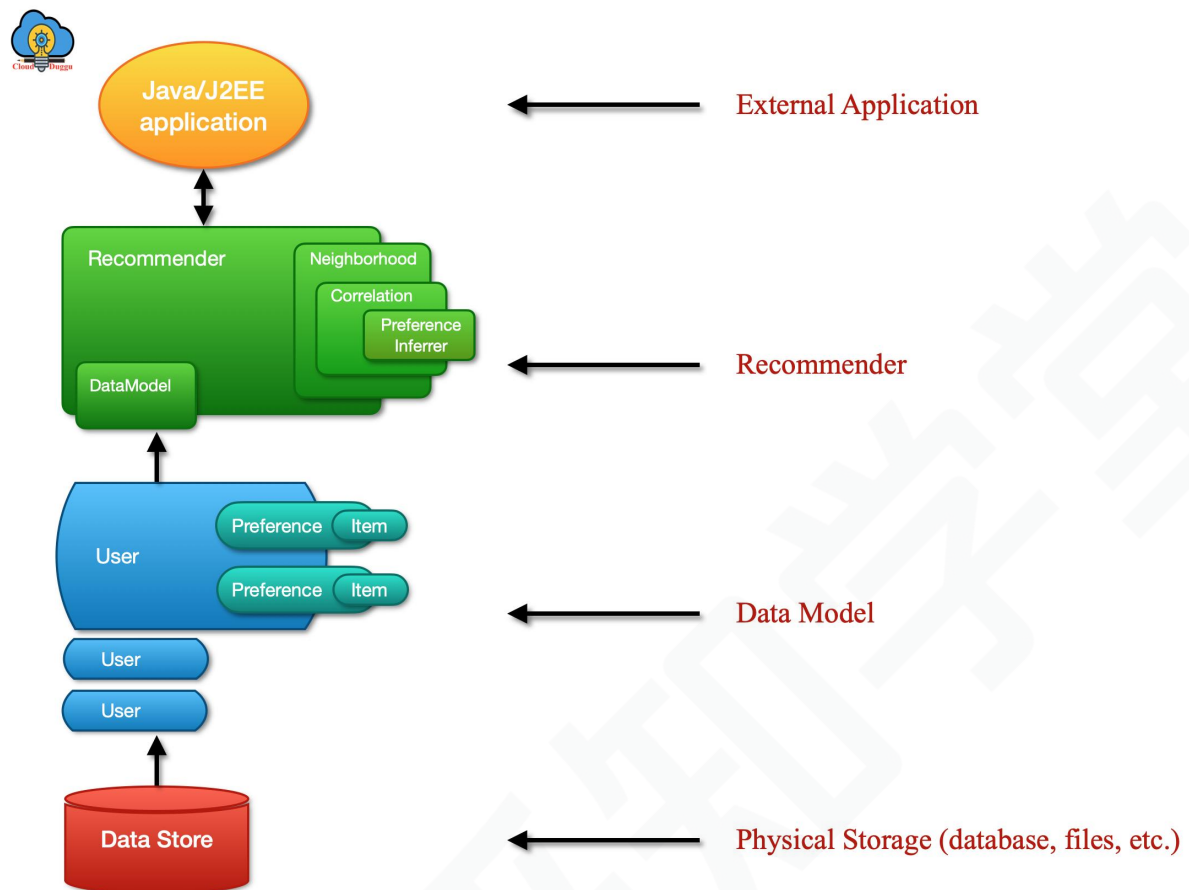


6.Mahout

Mahout 是一个用于 **创建机器学习应用程序的框架**。它提供了一组丰富的组件，您可以通过选择的算法构建定制的推荐系统。Mahout 的开发目的是提供执行、可扩展性和合规性。

以下是定义这些关键抽象的 Mahout 接口的重要包：

- `DataModel`
- `UserSimilarity`
- `ItemSimilarity`
- `UserNeighborhood`



7.HBase

HBase 是继 Google Bigtable 之后创建的分布式、开源、版本化、非关系型数据库。它是 Hadoop 生态系统的重要组成部分, 利用 HDFS 的容错功能, 提供对数据的实时读写访问。HBase 尽管是数据库, 但也可以称为数据存储系统, 因为它不提供触发器、查询语言和二级索引等 RDBMS 功能。

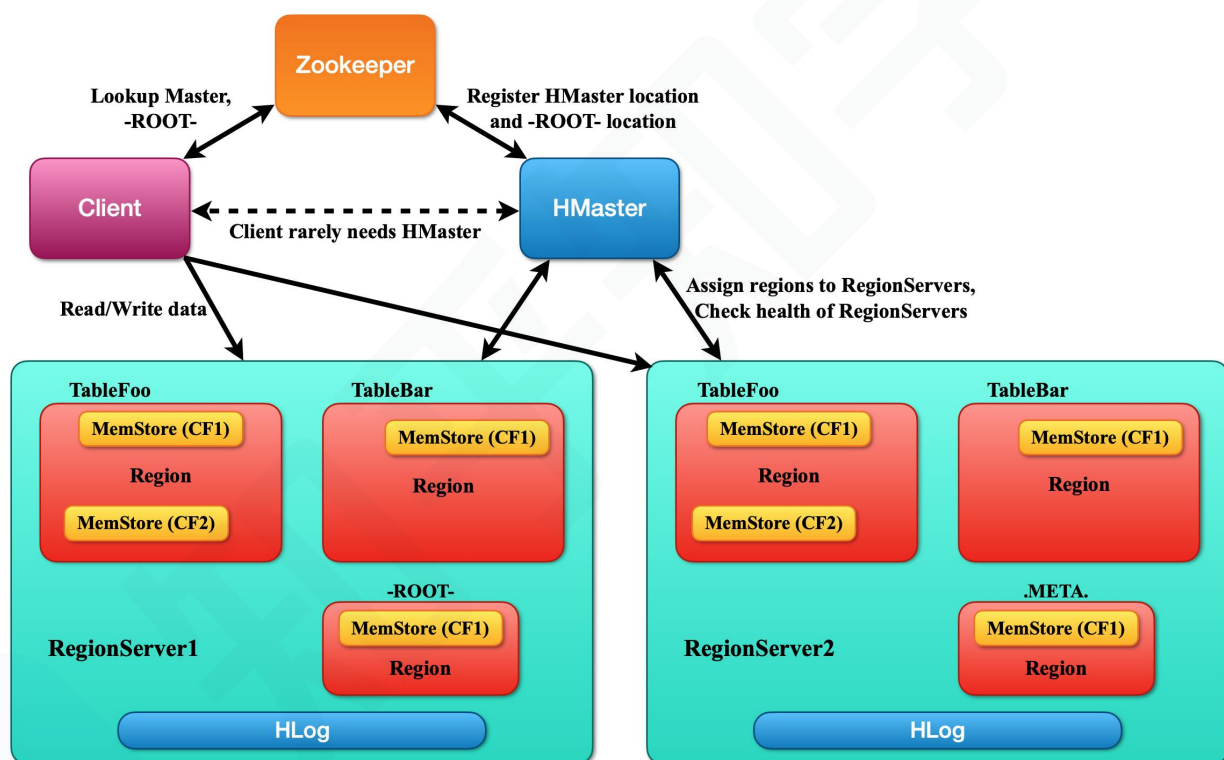
HBase 具有以下功能:

- 它提供持续的模块化可扩展性。
- 它提供定期的读取和写入。
- 直观且可配置的表分片。
- RegionServer 之间的自动故障转移支持。

- 它提供中央基类，用于支持带有 Apache HBase 表的 Hadoop MapReduce 作业。
- 使用 Java API 进行客户端访问很简单。
- 查询谓词通过服务器端过滤器下推。
- 它提供了 Thrift 网关和 REST-ful Web 服务，支持 XML、Protobuf 和二进制数据编码选择。

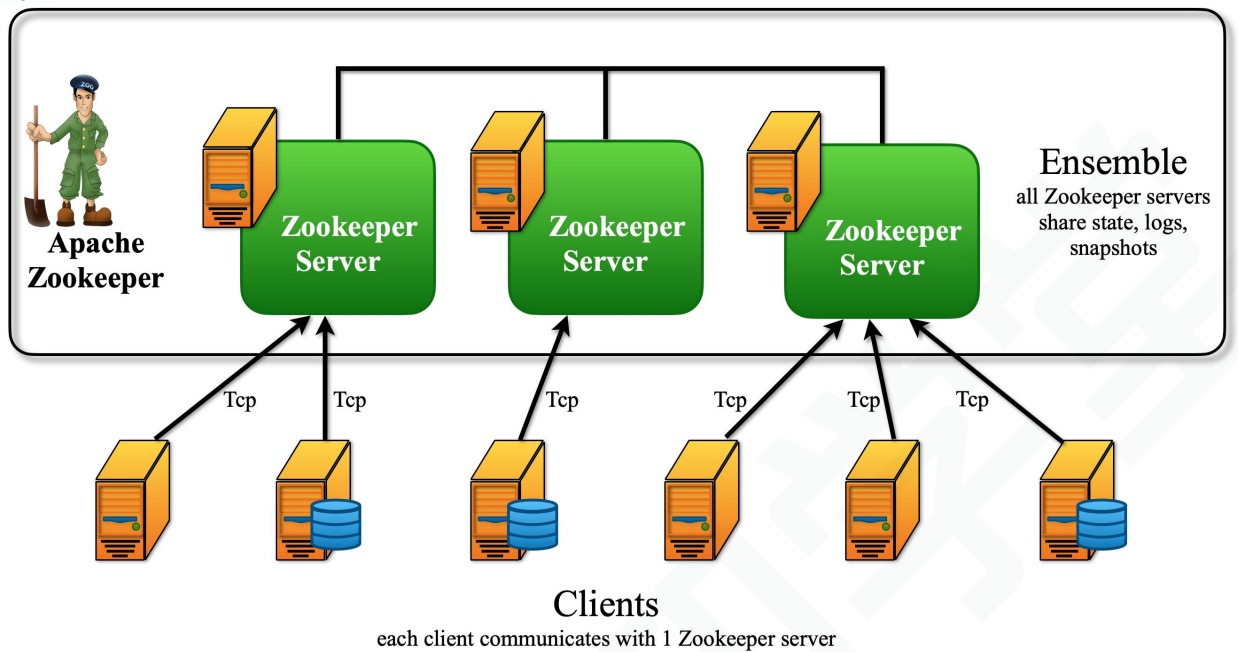


HBase Architecture



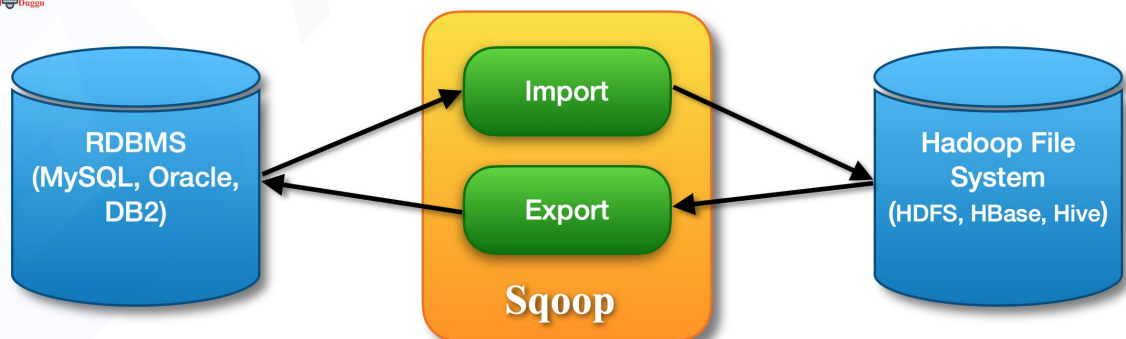
8.Zookeeper

Zookeeper 充当 Hadoop 不同服务之间的协调者，用于维护配置信息、命名、提供分布式同步、提供群组服务。Zookeeper 用于修复这些新部署在分布式环境中的应用程序的错误和竞争条件。



9.Sqoop

Sqoop 是一个数据传输工具，用于在 Hadoop 和关系数据库之间传输数据。它用于将数据从关系数据库管理系统 (MySQL 或 Oracle) 或大型机导入到 Hadoop (HDFS)，并在 Hadoop MapReduce 中转换数据。它还用于将数据导出回 RDBMS。Sqoop 使用 **map-reduce** 来导入和导出数据，因此它具有并行处理和容错特性。

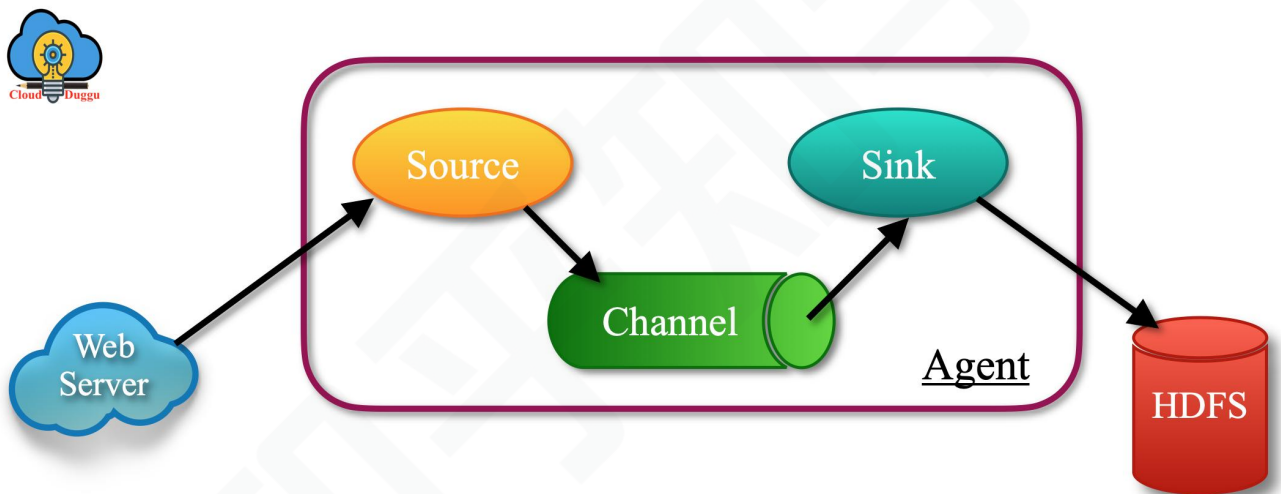


10.Flume

Flume 是一种类似于 Sqoop 的日志传输工具，但它适用于非结构化数据（日志），而 Sqoop 用于结构化和非结构化数据。Flume 是一个可靠、分布式且可用的系统，用于高效地收集、聚合大量日志数据并将其从许多不同的源移动到 HDFS。它不仅限于日志数据聚合，还可以用于传输大量事件数据。

Flume 具有以下三个组件：

- Source
- Channel
- Sink



11.Oozie

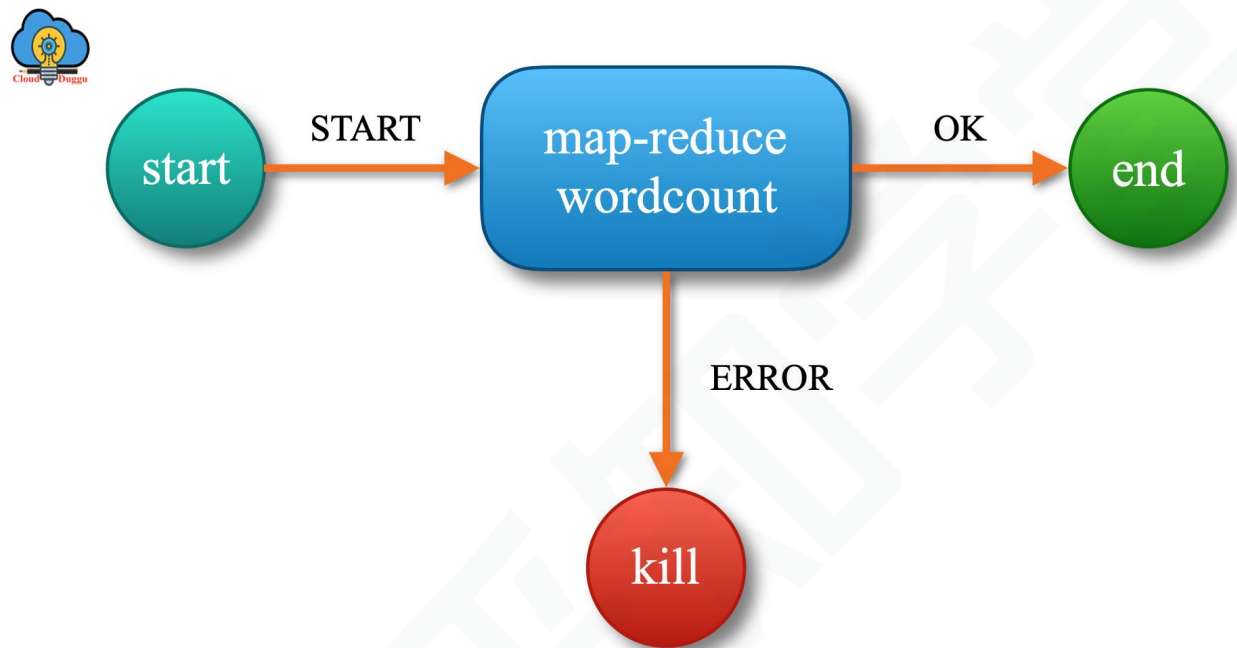
Oozie 是一个 **工作流调度框架**，用于调度 Hadoop Map / Reduce 和 Pig 作业。Apache Oozie 工作流程是 Hadoop Map / Reduce 作业、Pig 作业等操作的集合，排列在控制依赖 DAG（有向无环图）中。从一个动作到另一个动作的“**控制依赖性**”表明，除非第一个动作完成，否则另一个动作不会开始。

Oozie 工作流有以下两个节点，即 **控制流节点** 和 **操作节点**。

控制流节点（Control Flow Nodes）：这些节点用于提供控制工作流执行路径的机制。

操作节点 (Action Node)：操作节点提供了一种机制，工作流通过该机制触发计算 / 处理任务

的执行，例如 Hadoop MapReduce、HDFS、Pig、SSH、HTTP 作业。



12.Ambari

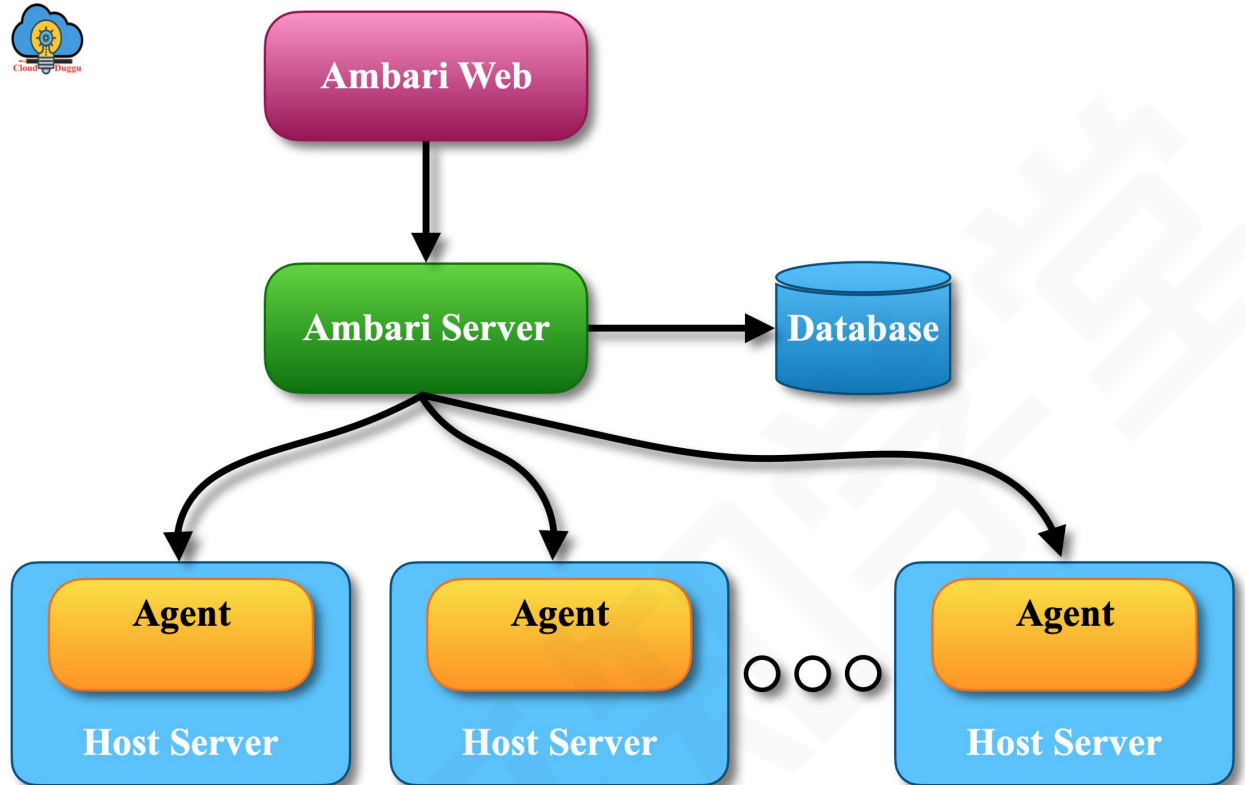
Ambari 用于配置、管理和监控 Apache Hadoop 集群。

它向系统管理员提供以下任务：

Hadoop 集群的配置：它提供了一种在任意数量的节点上安装 Hadoop 服务的媒介。它还处理集群的 Hadoop 服务配置。

Hadoop 集群的管理：它提供了一个中央控制来管理 Hadoop 服务，例如整个集群的启动、停止和重新配置。

Hadoop 集群监控：它提供了一个用于监控 Hadoop 集群的仪表盘（例如节点关闭、剩余磁盘空间不足等）。



13. Spark

Spark 是一个通用且快速的集群计算系统。它是一个非常强大的大数据工具。Spark 提供了 Python、Scala、Java、R 等多种语言的丰富 API。Spark 支持 Spark SQL、GraphX、MLlib、Spark Streaming、R 等高级工具。这些工具用于执行不同类型的操作，我们将在 Spark 部分中看到。

