MapReduce概述

更新时间：2020-09-04 17:41:49

[**本页目录**](javascript:void(0))

* [应用场景](https://help.aliyun.com/document_detail/27875.html?spm=a2c4g.11186623.6.765.1dab6371hipm1x#title-qxb-g8s-k01)
* [MapReduce流程说明](https://help.aliyun.com/document_detail/27875.html?spm=a2c4g.11186623.6.765.1dab6371hipm1x#title-tmu-7sf-nro)
* [使用限制](https://help.aliyun.com/document_detail/27875.html?spm=a2c4g.11186623.6.765.1dab6371hipm1x#title-1km-9r2-lgz)

本文为您介绍MaxCompute支持的MapReduce编程接口及使用限制。

MaxCompute提供三个版本的MapReduce编程接口：

* MaxCompute MapReduce：MaxCompute的原生接口，执行速度快、开发快捷、不暴露文件系统。
* MaxCompute扩展MapReduce（[MR2](https://help.aliyun.com/document_detail/27876.html#t12014.html)）：对MaxCompute MapReduce的扩展，支持更复杂的作业调度逻辑。MapReduce的实现方式与MaxCompute原生接口一致。
* [Hadoop兼容版本MapReduce](https://help.aliyun.com/document_detail/44626.html#t12015.html)：高度兼容[Hadoop MapReduce](http://hadoop.apache.org/docs/r1.0.4/cn/mapred_tutorial.html)，与MaxCompute MapReduce [MR2](https://help.aliyun.com/document_detail/27876.html#t12014.html)不兼容。

以上三个版本在[基本概念](https://help.aliyun.com/document_detail/27879.html#t12018.html)、[作业提交](https://help.aliyun.com/document_detail/27878.html#t12017.html)、[输入与输出](https://help.aliyun.com/document_detail/27880.html#t12019.html)、[资源使用](https://help.aliyun.com/document_detail/27881.html#t12020.html)等方面基本一致，仅各版本的Java SDK有所不同。本文仅对MapReduce的基本原理做简单介绍，更多详情请参见[Hadoop Map/Reduce教程](http://hadoop.apache.org/docs/r1.0.4/cn/mapred_tutorial.html)。

**说明** 您无法通过MapReduce读写[外部表](https://help.aliyun.com/document_detail/45389.html#concept-tzd-tlg-vdb)中的数据。

**应用场景**

MapReduce支持下列场景：

* 搜索：网页爬取、倒排索引、PageRank。
* Web访问日志分析：
  + 分析和挖掘用户在Web上的访问、购物行为特征，实现个性化推荐。
  + 分析用户访问行为。
* 文本统计分析：
  + 热门小说的字数统计（WordCount）、词频TFIDF分析。
  + 学术论文、专利文献的引用分析和统计。
  + 维基百科数据分析。
* 海量数据挖掘：非结构化数据、时空数据和图像数据挖掘。
* 机器学习：监督学习、无监督学习和分类算法（例如决策树、SVM）。
* 自然语言处理：
  + 基于大数据的训练和预测。
  + 基于语料库构建单词同现矩阵，频繁项集数据挖掘、重复文档检测等。
* 广告推荐：用户单击（CTR）和购买行为（CVR）预测。

**MapReduce流程说明**

MapReduce处理数据过程主要分成Map和Reduce两个阶段。首先执行Map阶段，再执行Reduce阶段。Map和Reduce的处理逻辑由用户自定义实现，但要符合MapReduce框架的约定。MapReuce处理数据的完整流程如下：

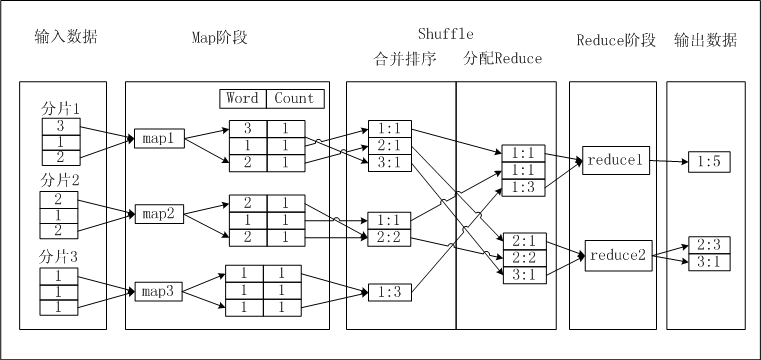
1. 输入数据：对文本进行分片，将每片内的数据作为单个Map Worker的输入。分片完毕后，多个Map Worker便可以同时工作。

在正式执行Map前，需要将输入数据进行分片。所谓分片，就是将输入数据切分为大小相等的数据块，每一块作为单个Map Worker的输入被处理，以便于多个Map Worker同时工作。

1. Map阶段：每个Map Worker在读入各自的数据后，进行计算处理，最终输出给Reduce。Map Worker在输出数据时，需要为每一条输出数据指定一个Key，这个Key值决定了这条数据将会被发送给哪一个Reduce Worker。Key值和Reduce Worker是多对一的关系，具有相同Key的数据会被发送给同一个Reduce Worker，单个Reduce Worker有可能会接收到多个Key值的数据。
2. 在进入Reduce阶段之前，MapReduce框架会对数据按照Key值排序，使得具有相同Key的数据彼此相邻。如果您指定了**合并操作（Combiner）**，框架会调用Combiner，将具有相同Key的数据进行聚合。Combiner的逻辑可以由您自定义实现。与经典的MapReduce框架协议不同，在MaxCompute中，Combiner的输入、输出的参数必须与Reduce保持一致，这部分的处理通常也叫做**洗牌（Shuffle）**。
3. Reduce阶段：进入Reduce阶段，相同Key的数据会传送至同一个Reduce Worker。同一个Reduce Worker会接收来自多个Map Worker的数据。每个Reduce Worker会对Key相同的多个数据进行Reduce操作。最后，一个Key的多条数据经过Reduce的作用后，将变成一个值。
4. 输出结果数据。

**说明** 上文仅是对MapReduce框架的简单介绍，更多详情请查阅[功能介绍](https://help.aliyun.com/document_detail/27879.html#concept-brv-lyf-vdb)。

下文将以WordCount为例，为您介绍MaxCompute MapReduce各个阶段的概念。

假设存在一个文本a.txt，文本内每行是一个数字，您要统计每个数字出现的次数。文本内的数字称为Word，数字出现的次数称为Count。如果MaxCompute MapReduce完成这一功能，需要经历以下流程，图示如下。[](http://static-aliyun-doc.oss-cn-hangzhou.aliyuncs.com/assets/img/zh-CN/9052129951/p1922.jpg)

**操作步骤**

1. 输入数据：对文本进行分片，将每片内的数据作为单个Map Worker的输入。
2. Map阶段：Map处理输入，每获取一个数字，将数字的Count设置为1，并将此<Word, Count>对输出，此时以Word作为输出数据的Key。
3. Shuffle>合并排序：在Shuffle阶段前期，首先对每个Map Worker的输出，按照Key值（即Word值）进行排序。排序后进行Combiner操作，即将Key值（Word值）相同的Count累加，构成一个新的<Word, Count>对。此过程被称为合并排序。
4. Shuffle>分配Reduce：在Shuffle阶段后期，数据被发送到Reduce端。Reduce Worker收到数据后依赖Key值再次对数据排序。
5. Reduce阶段：每个Reduce Worker对数据进行处理时，采用与Combiner相同的逻辑，将Key值（Word值）相同的Count累加，得到输出结果。
6. 输出结果数据。

**说明** 由于MaxCompute的所有数据都被存放在表中，因此MaxCompute MapReduce的输入、输出只能是表，不允许您自定义输出格式，不提供类似文件系统的接口。

**使用限制**

* MapReduce使用限制请参见[使用限制汇总](https://help.aliyun.com/document_detail/51847.html#concept-m4c-4lg-vdb)。
* 有关本地运行的MapReduce使用限制，请参见[本地运行和分布式环境运行差异](https://help.aliyun.com/document_detail/27882.html#section-k3l-t3g-vdb)。
* [按量计费开发者版](https://help.aliyun.com/document_detail/116741.html#concept-221119)资源仅支持MaxCompute SQL（支持使用UDF）、PyODPS作业任务，暂不支持MapReduce、Spark等其他任务。