# 概述

MapReduce及其扩展解决了离线批处理问题，但是无法保证实时性。对于实时性要求高的场景，可以采用流式计算或者实时分析系统进行处理。

流式计算（Stream Processing）解决在线聚合（Online Aggregation）、在线过滤（Online Filter）等问题，流式计算同时具有存储系统和计算系统的特点，经常应用在一些类似反作弊、交易异常监控等场景。流式计算的操作算子和时间相关，处理最近一段时间窗口内的数据。

流处理和批处理：



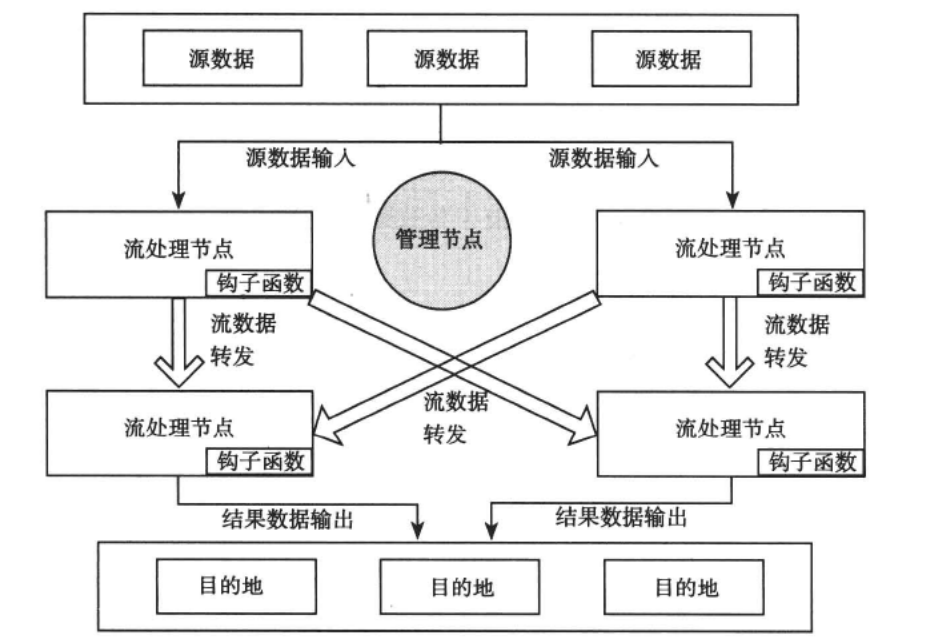
# 原理

流式计算强调的是数据流的实时性。MapReduce系统主要解决的是对静态数据的批量处理，当MapReduce作业启动时，已经准备好了输入数据，比如保存在分布式文件系统上。而流式计算系统在启动时，输入数据一般并没有完全到位，而是经由外部数据流源源不断地流入。另外，流式计算并不像批处理系统那样，重视数据处理的总吞吐量，而是更加重视对数据处理的延迟。

MapReduce及其扩展曹勇的是一种比较静态的模型，如果用它来做数据流的处理，首先需要将数据流缓存并分块，然后放入集群计算。如果MapReduce每次处理的数据量较小，缓存数据流的时间较短，但是，MapReduce架构造成的额外开销将会占很大比重；如果MapReduce每次处理的数据量较大，缓存数据流的时间会很长，无法满足实时性的要求。

## 流式计算框架

流式计算系统架构如下：



源数据写入到流处理节点，流处理节点内部运行用户自定义钩子函数对输入流进行处理，处理完后根据一定的规则转发给下游的流处理节点来处理。另外，系统中往往还有管理节点，用来管理流处理节点的状态以及节点之间的路由规则。

典型钩子函数包括：

聚合函数：计算最近一段时间窗口内数据的聚合值，如max、min、avg、sum、count等。

过滤函数：过滤最近一段时间窗口内满足某些特性的数据，如过滤1秒钟内重复的点击数。

# 流批处理

## 批处理框架



### Apache Hadoop MapReduce

### Apache Spark

### Apache Hive

### Presto/Trino

### Apache Tez

## 流处理框架



### Apache Flink

### Apache Storm

### Kafka Streams

### Apache Pulsar Functions

### Samza

## 流批处理一体框架



### Apache Spark Structured Streaming

### Apache Flink

### Apache Bean

### Google Cloud Dataflow

### KSQLDB（Confluent）

# 流式计算框架

## 开源流式计算框架



### Apache Flink

Flink是一款分布式的计算引擎，它可以用来做批处理，即处理静态的数据集、历史的数据集；也可以用来做流处理，即实时地处理一些实时数据流，实时地产生数据的结果；也可以用来做一些基于事件的应用，比如说滴滴通过Flink CEP实现实时监测用户及司机的行为流来判断用户或司机的行为是否正当。

总而言之，Flink是一个Stateful Computations Over Streams，即数据流上的有状态的计算。这里面有两个关键字，一个是Streams，Flink认为有界数据集是无界数据流的一种特例，所以说有界数据集也是一种数据流，事件流也是一种数据流。Everything is streams，即Flink可以用来处理任何的数据，可以支持批处理、流处理、AI、MachineLearning等等。另外一个关键词是Stateful，即有状态计算。有状态计算是最近几年来越来越被用户需求的一个功能。举例说明状态的含义，比如说一个网站一天内访问UV数，那么这个UV数便为状态。Flink提供了内置的对状态的一致性的处理，即如果任务发生了Failover，其状态不会丢失、不会被多算少算，同时提供了非常高的性能。

### Apache Spark

#### Spark Streaming

#### Spark Structured Streaming

参考：

[Structured Streaming概述](https://juejin.cn/post/7089033520462446629)

Spark Streaming是Apache Spark早期基于RDD开发的流式系统，用户使用DStream API来编写代码，支持高吞吐和良好的容错。其背后的主要模型是Micro Batch（微批处理），也就是将数据流切成等时间间隔（BatchInterval）的小批量任务来执行。

Structured Streaming则是在Spark 2.0加入的，经过重新设计的全新流式引擎。它的模型十分简洁，易于理解。一个流的数据源从逻辑上来说就是一个不断增长的动态表格，随着时间的推移，新数据被持续不断地添加到表格的末尾，用户可以使用Dataset/DataFrame 或者 SQL 来对这个动态数据源进行实时查询。

### Apache Strom

### Apache Kafka Streams

Kafka Streams被认为是开发实时应用程序的最简单方法。它是一个Kafka的客户端API库，编写简单的java和scala代码就可以实现流式处理。

### Apache Samza

### Google Dataflow（Apache Beam）

### Yahoo S4

## 主流商业产品 / 云服务平台



### Amazon Kinesis

### Google Cloud Dataflow

### Azure Stream Analytics

### Alibaba Cloud DataStream

### Tencent Cloud CLS + StreamCompute

### Cloudera DataFlow (CDF)

### Confluent ksqlDB