Storm

——分布式实时计算系统

计科1605 宋文宇

1. Storm是什么?
2. Storm产生背景?
3. Storm与Hadoop的比较?
4. Storm集群架构及概念阐释

1. Storm是什么?

APACHE STORM™

Apache Storm is a **free** and **open source distributed realtime computation** system. Storm makes it easy to reliably process unbounded streams of data, doing for realtime processing what Hadoop did for batch processing. **Storm is simple**, can be used with **any programming language**, **and is a lot of fun to use!**

2. Storm产生背景

随着互联网的更进一步发展，从Portal信息浏览型到Search信息搜索型到SNS关系交互传递型，以及电子商务、互联网旅游生活产品等将生活中的流通环节在线化。2011年twitter对Storm开源。使得现在的开发人员可以快速的搭建一套健壮、易用的实时流处理框架，配合SQL产品或者NoSQL产品或者MapReduce计算平台，就可以低成本的做出很多以前很难想象的实时产品。极大的满足了用户对于实时性、高效率的需求。

3. Storm与Hadoop的比较

1. 应用场景不同：

——Hadoop是分布式**批处理**计算，较多用其进行数据挖掘和分析

——Storm是分布式**实时**计算，较多用于对实时性要求较高的场景

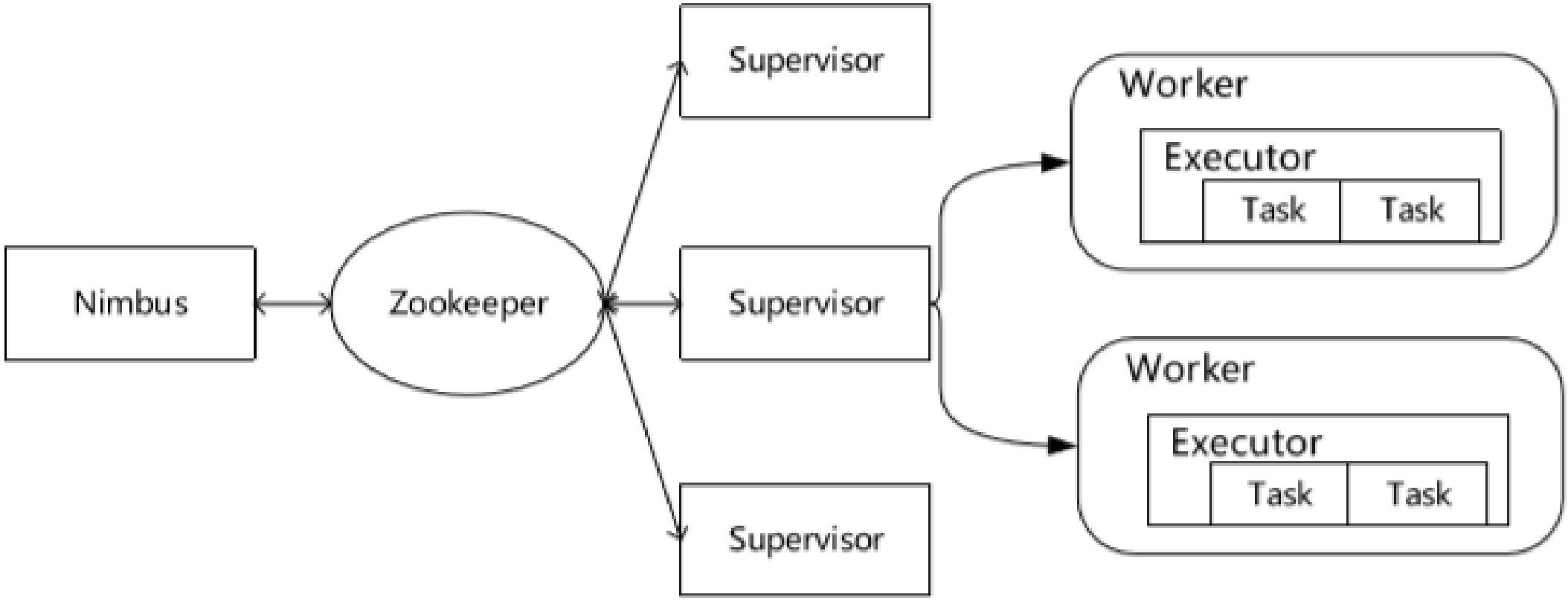
3. Storm与Hadoop的比较

2. 数据处理的方式不同：

——Hadoop是**磁盘级**计算，计算时需要在磁盘中读取数据；其采用的是MapReduce的逻辑，把数据进行切片计算用这种方式来处理大量的**离线数据**，并且只能处理**已经存在HDFS或者HBase中的数据**。Hadoop效率的提高是通过**移动计算到这些存放数据的机器上**来实现的。并且在计算完毕后一定要**结束**。

——Storm是**内存级**计算，需要进行计算的数据直接**通过网络导入内存**。Storm处理的是实时消息队列中的数据，需要写好一个Topology逻辑，然后对接受的数据进行处理。所以Storm效率的提高是通过**移动数据平均分配到机器资源**实现的。并且Storm**没有结束状态**，当前数据计算完后暂停在当前的状态，**当有新数据输入时继续计算**。

4.1 Storm的集群架构



4.2 相关概念阐释

1. Tuple

Storm集群中消息传递的基本数据单元　类型没有规定，Storm支持所有的基本类型和自定义类型作为Tuple的值类型。

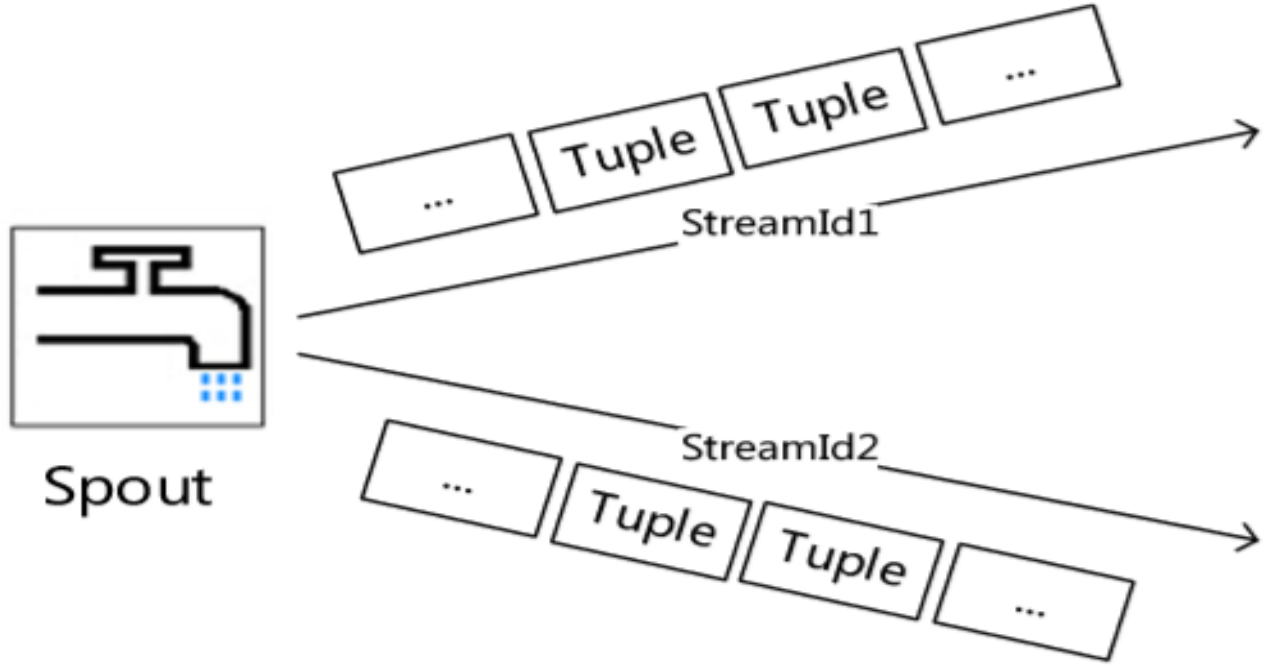
2. Stream

很多个tuple组成的序列就组成了一个Stream。

4.2 相关概念阐释

3. Spout

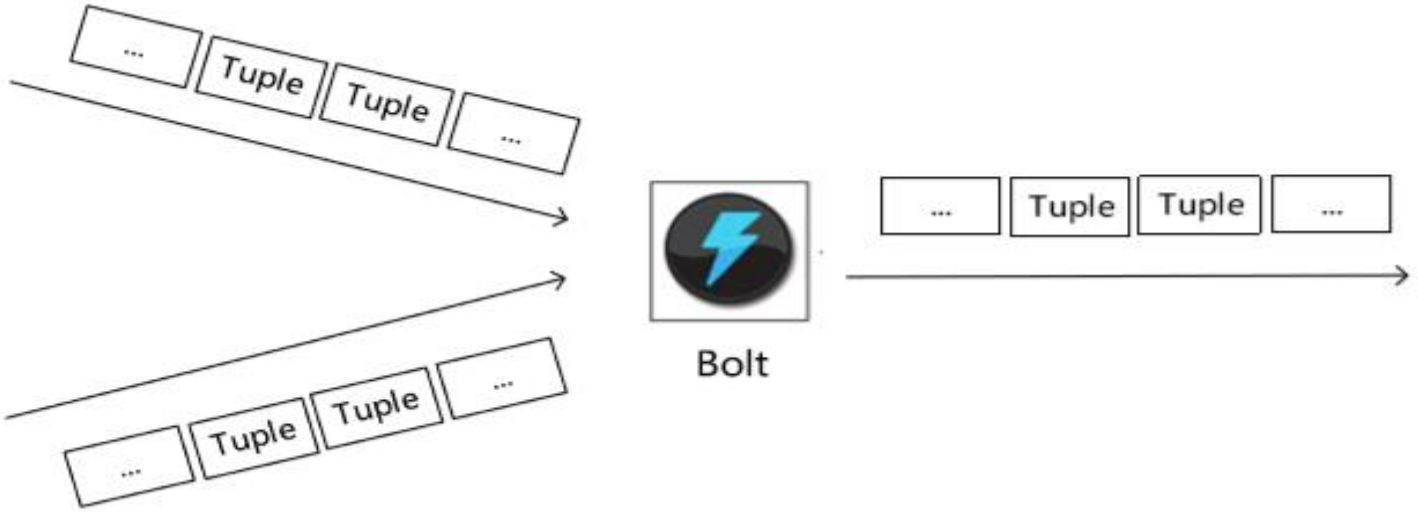
在Topology中产生源数据流的组件。获取外部数据后，通过调用nextTuple函数，将数据发送至Bolt



4.2 相关概念阐释

4. Bolt

在Topology中接受Spout的数据然后执行处理工作的组件。Bolt在接受到消息后会调用execute方法，用户可以在其中执行自己想要的任何操作。如果遇到复杂流的处理可能会将tuple发送给另一个Bolt进行处理，也就是说需要经过多个Blots的处理



4.2 相关概念阐释

Topology	一个实时计算应用程序逻辑被封装在Topology对象中，类似Hadoop中的job，Topology会一直运行直到你显式杀死它。
Nimbus	负责资源分配和任务调度，类似Hadoop中的JobTracker。
Supervisor	负责接受Nimbus分配的任务，启动和停止自己管理的Worker进程
Worker	运行具体处理组件逻辑的进程。
Executor	Executor为Worker进程中的具体的物理线程，同一个Spout/Bolt的Task可能会共享一个物理线程
Task	每一个Spout/Bolt具体要干的活，各个节点之间进行Grouping的单位。
Tuple	消息传递的基本单元。
Stream	源源不断传递的Tuple就组成了Stream。