

Streaming技术原理

www.huawei.com





目标

- 学完本课程后，您将能够：
 - 掌握实时流处理概念
 - 掌握Streaming系统架构
 - 掌握Streaming关键特性
 - 掌握CQL基本概念

泰克教育
TECH EDUCATION



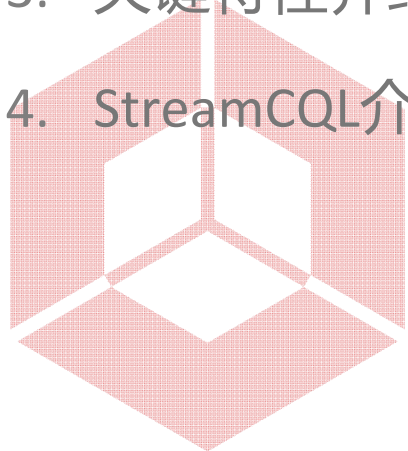
目录

1. Streaming简介

2. 系统架构

3. 关键特性介绍

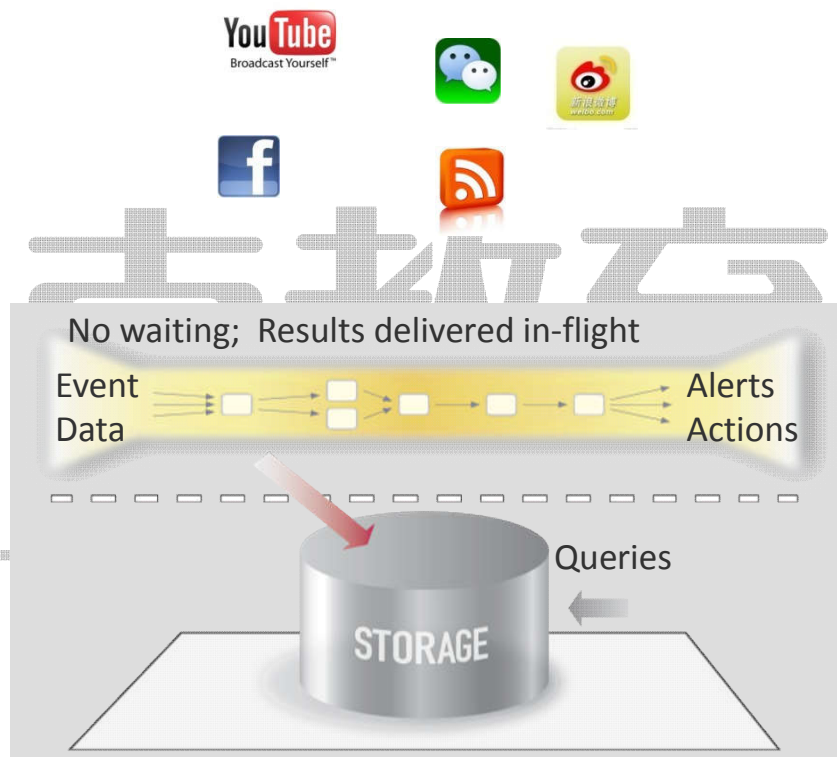
4. StreamCQL介绍



泰克教育
TECH EDUCATION

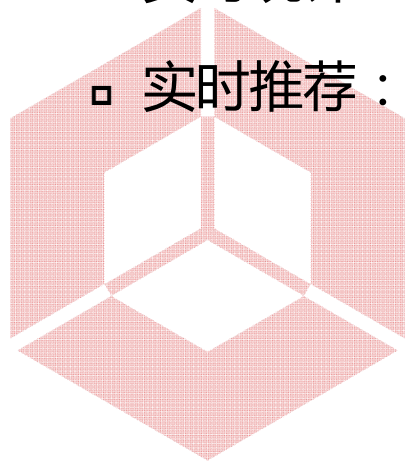
Streaming简介

- Streaming基于开源Storm，是一个分布式、实时计算框架。
- Streaming具有以下几种特点：
 - 实时响应，低延迟。
 - 数据不存储，先计算。
 - 连续查询。
 - 事件驱动。



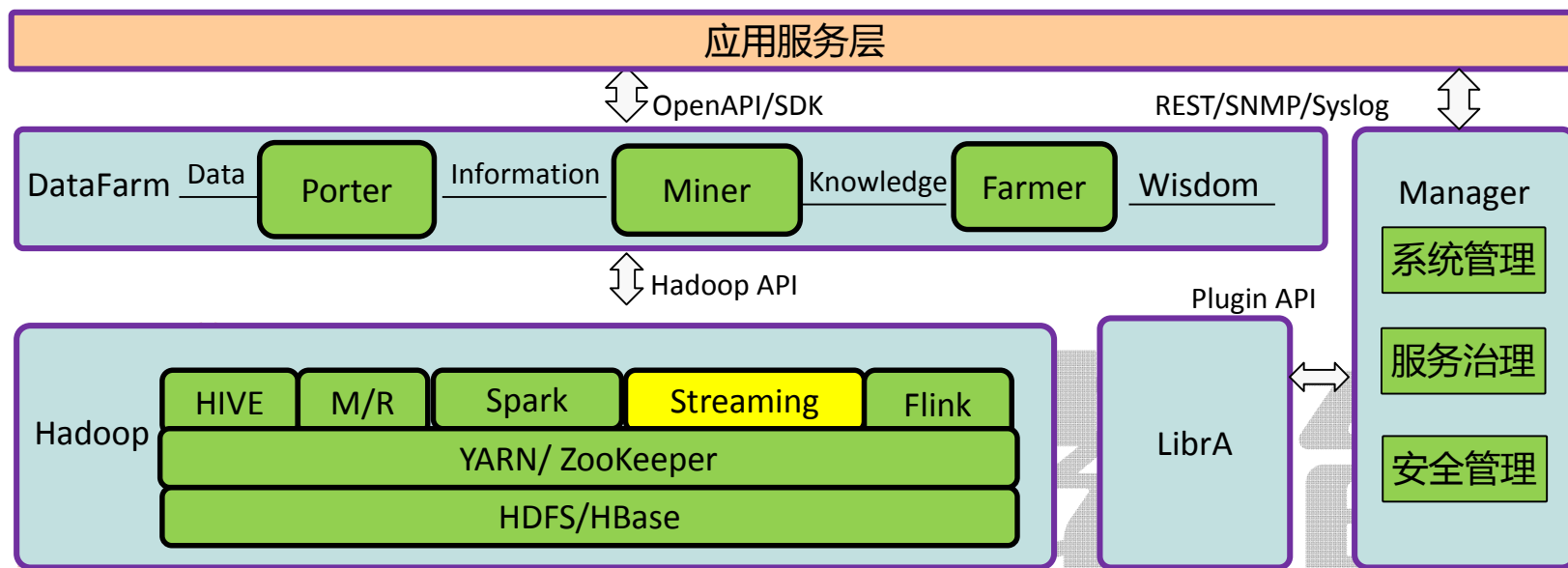
Streaming应用场景

- 主要应用于以下几种场景：
 - 实时分析：如实时日志处理、交通流量分析等。
 - 实时统计：如网站的实时访问统计、排序等。
 - 实时推荐：如实时广告定位、事件营销等。



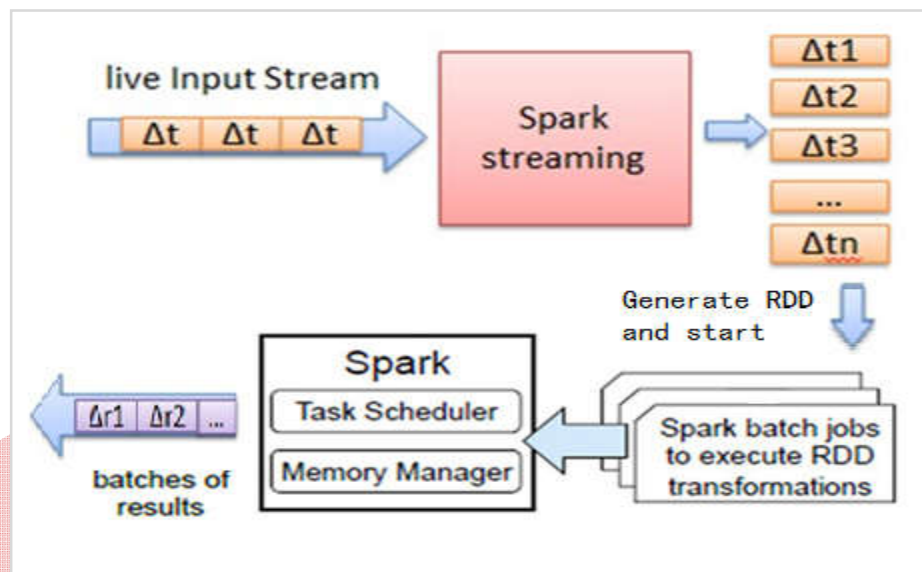
泰克教育
TECH EDUCATION

Streaming在FusionInsight中的位置

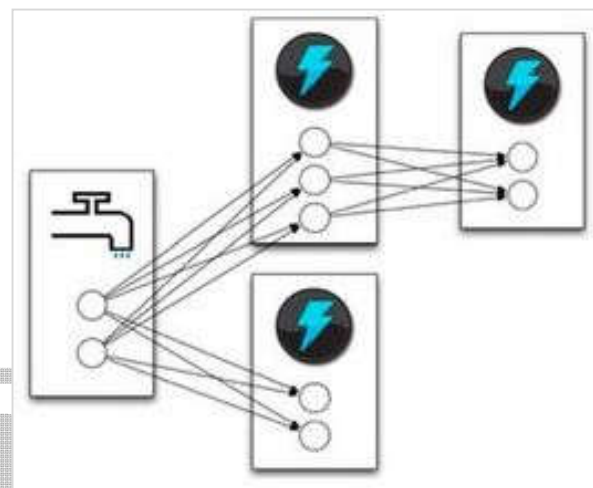


Streaming是一个实时的分布式的实时计算框架，在实时业务中有广泛的应用。

与Spark Streaming的比较



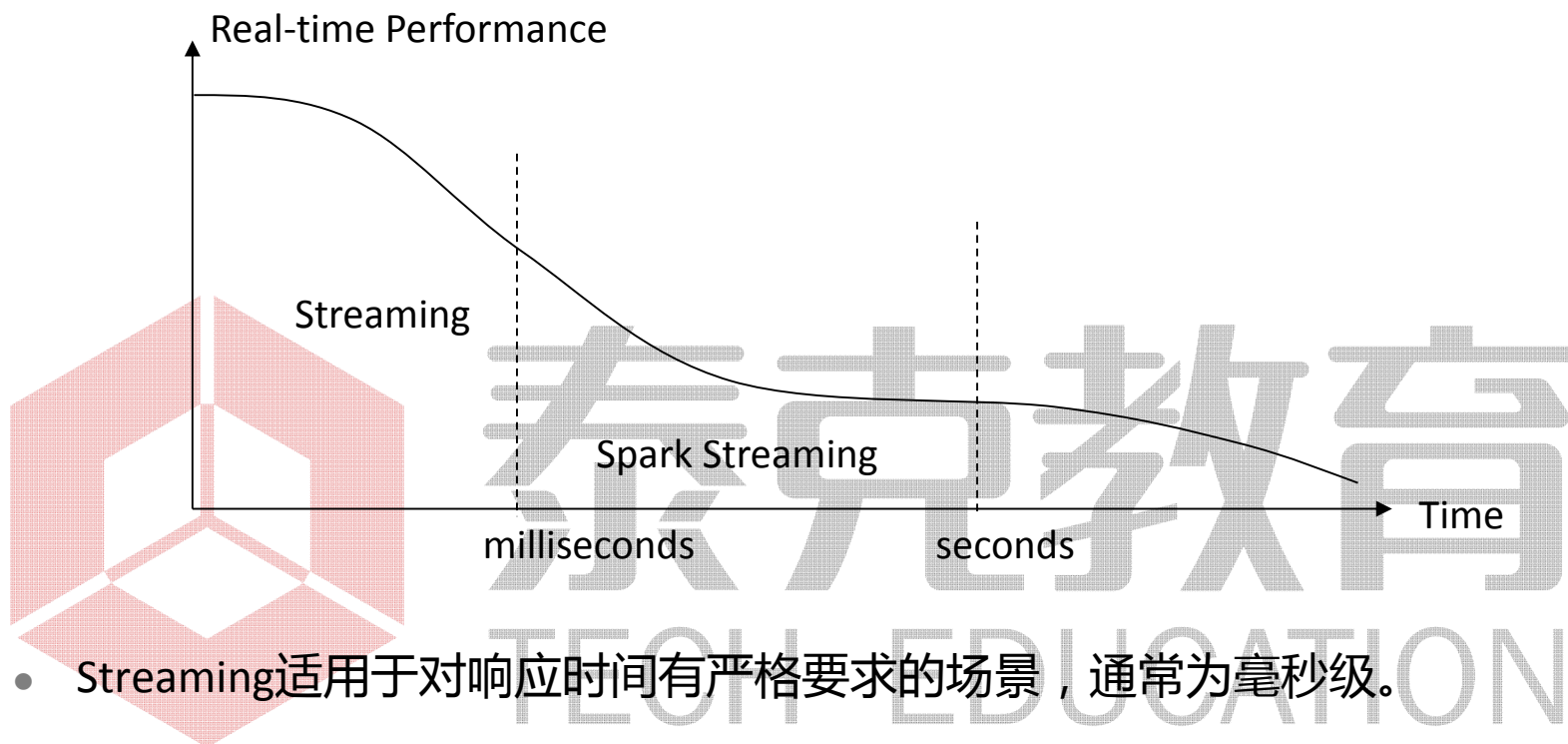
SparkStreaming微批处理流程



Streaming流处理流程

	SparkStreaming	Streaming
任务执行方式	执行逻辑即时启动，运行完回收	执行逻辑预先启动，持续存在
事件处理方式	事件需积累到一定量时才进行处理	事件实时处理
时延	秒级	毫秒级
吞吐量	高（约为Streaming的2~5倍）	较高

适用场景比较



- Streaming适用于对响应时间有严格要求的场景，通常为毫秒级。
- Spark Streaming适用于对响应时间要求不高的场景，通常为秒级。



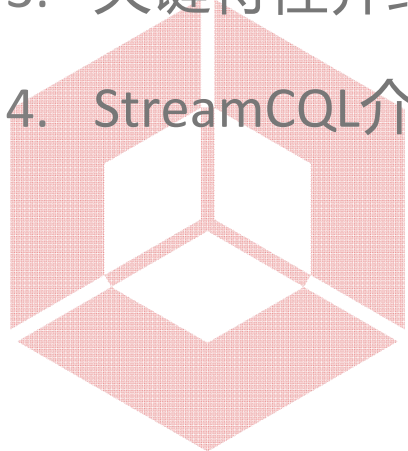
目录

1. Streaming简介

2. 系统架构

3. 关键特性介绍

4. StreamCQL介绍



泰克教育
TECH EDUCATION

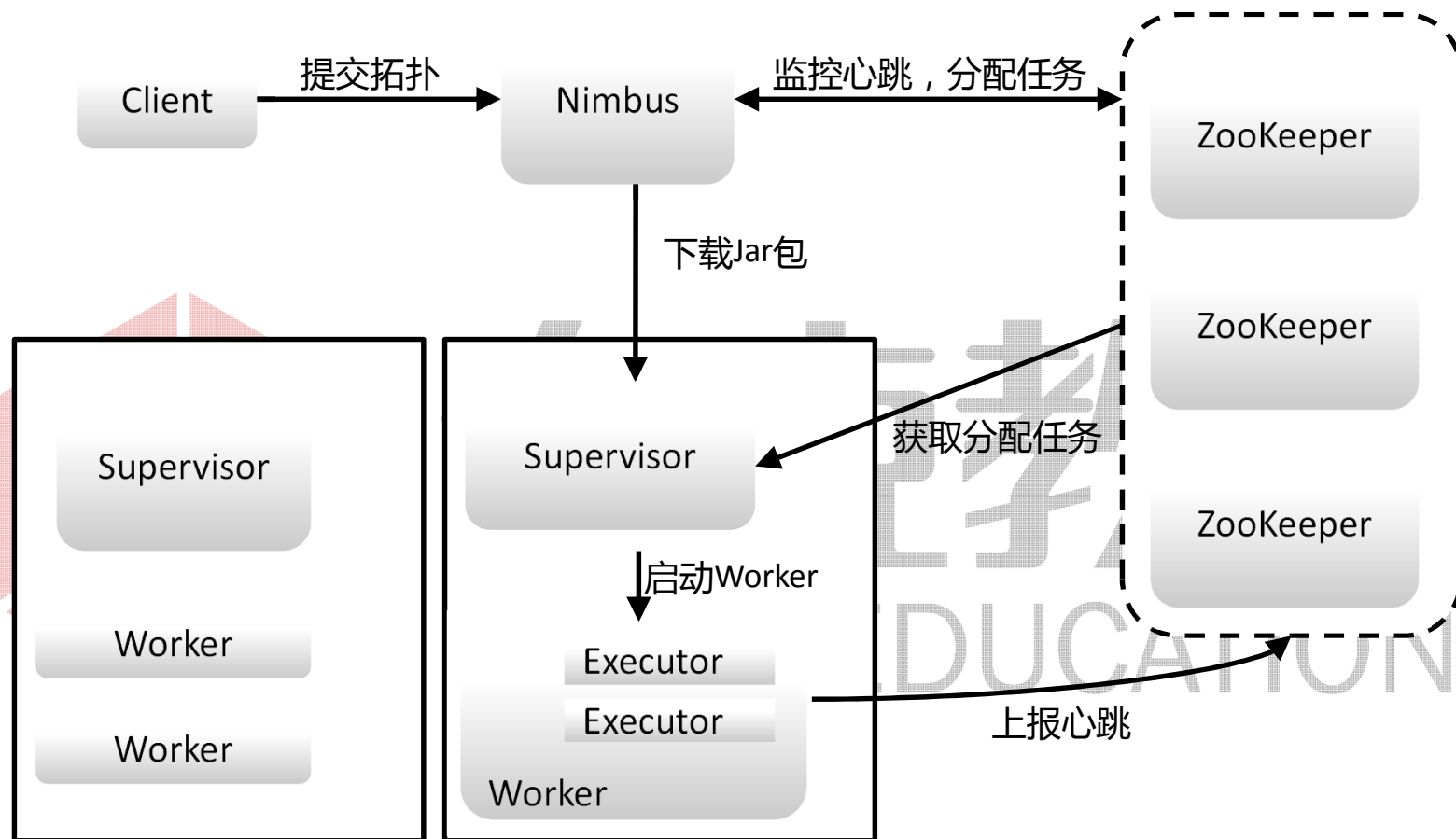
基本概念（1）

- Topology：Streaming中运行的一个实时应用程序。
- Nimbus：负责资源分配和任务调度。
- Supervisor：负责接受Nimbus分配的任务,启动和停止属于自己管理的worker进程。
- Worker：Topology运行时的物理进程。每个Worker是一个JVM进程。
- Spout：在一个Topology中产生源数据流的组件。
- Bolt：在一个Topology中接受数据然后执行处理的组件。

基本概念（2）

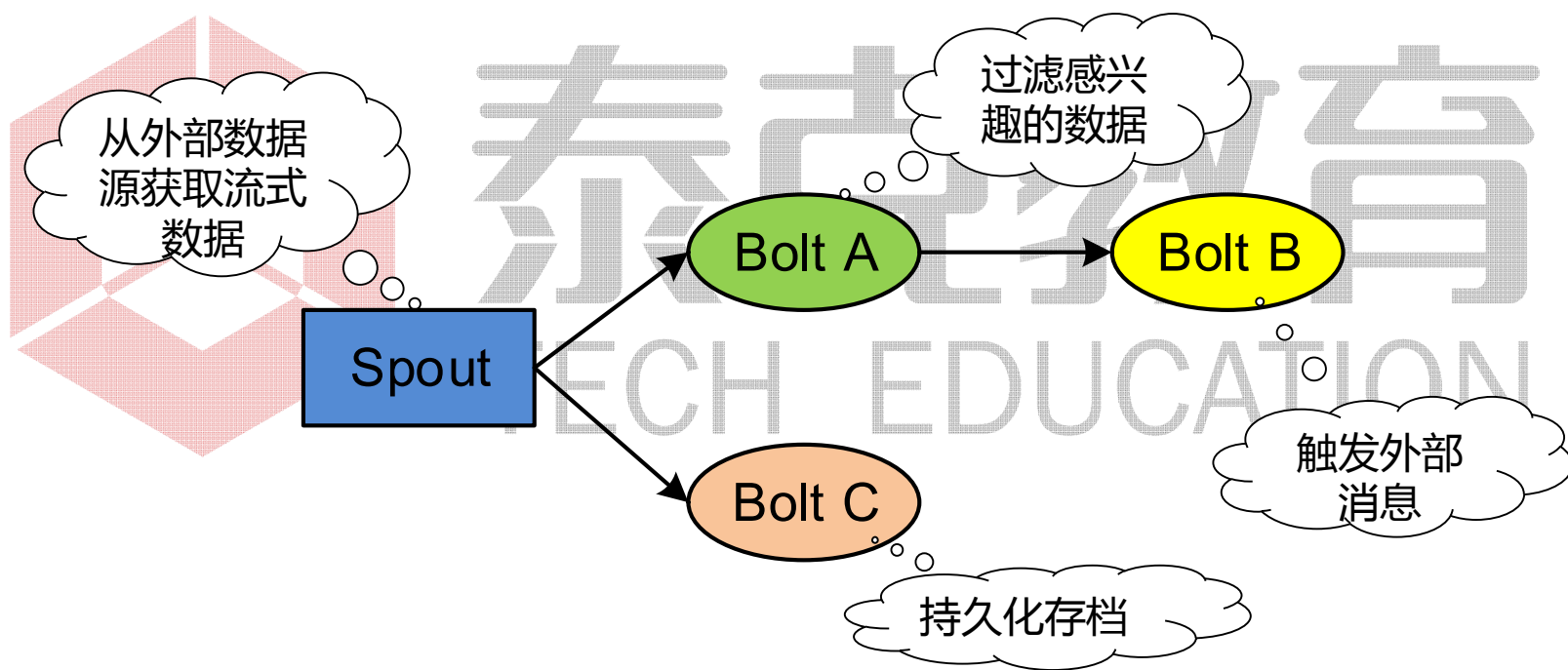
- Task：Worker中每一个Spout/Bolt的线程称为一个task。
- Tuple：Streaming的核心数据结构，是消息传递的基本单元，不可变Key-Value对，这些Tuple会以一种分布式的方式进行创建和处理。
- Stream：一个无边界的连续Tuple序列。
- ZooKeeper：为Streaming服务中各进程提供分布式协作服务。主备Nimbus、Supervisor、Worker将自己的信息注册到ZooKeeper中，Nimbus据此感知各个角色的健康状态。

系统架构



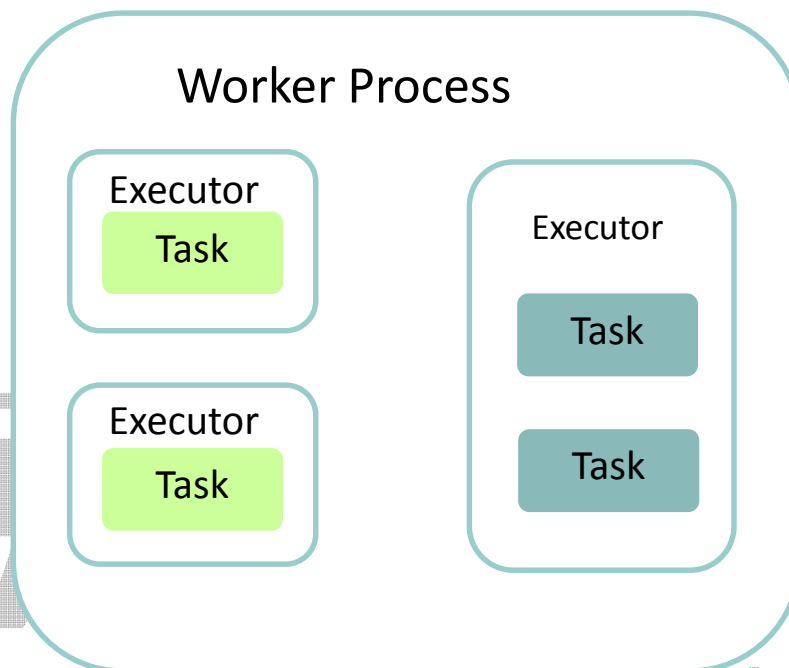
Topology介绍

- 一个Topology是由一组Spout组件（数据源）和Bolt组件（逻辑处理）通过Stream Groupings进行连接的有向无环图(DAG)。
- 业务处理逻辑被封装进Streaming中的Topology中。



Worker介绍

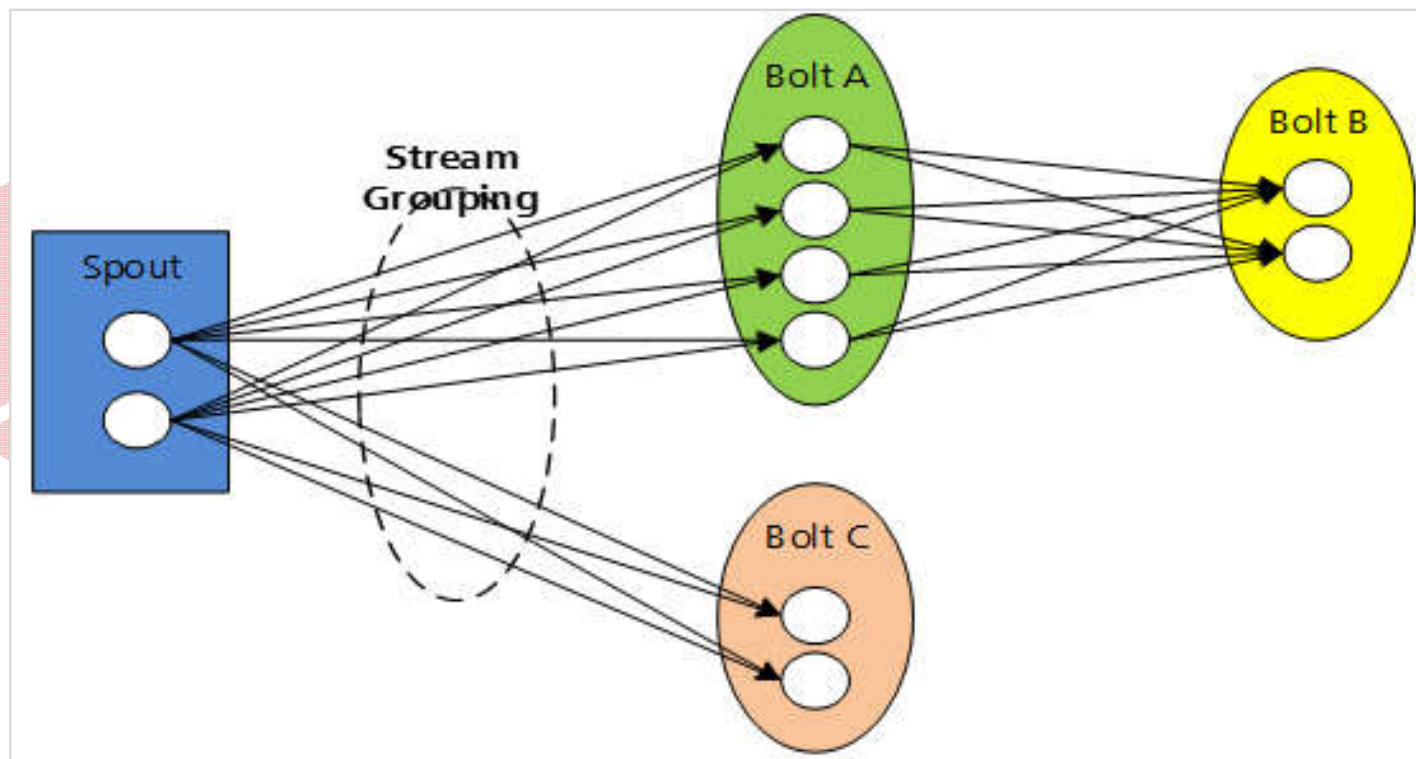
- Worker：一个Worker是一个JVM进程，所有的Topology都是在一个或者多个Worker中运行的。Worker启动后是长期运行的，除非人工停止。Worker进程的个数取决于Topology的设置，且无设置上限，具体可获得调度并启动的Worker个数则取决于Supervisor配置的slot个数。



- Executor：在一个单独的Worker进程中会运行一个或多个Executor**线程**。每个Executor只能运行Spout或者Bolt中的一个或多个task实例。
- Task：是最终完成数据处理的实体单元。

Task介绍

- Topology里面的每一个Component (Spout/Bolt) 节点都是并行运行的。在Topology里面，可以指定每个节点的并发度，Streaming则会在集群里面分配相应的Task来同时计算，以增强系统的处理能力。



消息分发策略

分组方式	功能介绍
fieldsGrouping (字段分组)	按照消息的哈希值分组发送给目标Bolt的Task。
globalGrouping (全局分组)	所有消息都发送给目标Bolt固定的一个Task。
shuffleGrouping (随机分组)	消息发送给目标Bolt的随机一个task。
localOrShuffleGrouping (本地或者随机分组)	如果目标Bolt在同一工作进程存在一个或多个Task，数据会随机分配给这些Task。否则，该分组方式与随机分组方式相同。
allGrouping (广播分组)	消息群发给目标Bolt的所有Task。
directGrouping (直接分组)	由数据生产者决定数据发送给目标Bolt的哪一个Task。需在发送时使用emitDirect(taskID, tuple)接口指定TaskID。
partialKeyGrouping (局部字段分组)	更均衡的字段分组。
noneGrouping (不分组)	当前和随机分组相同。



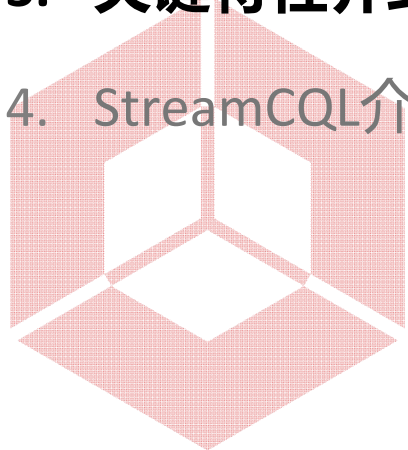
目录

1. Streaming简介

2. 系统架构

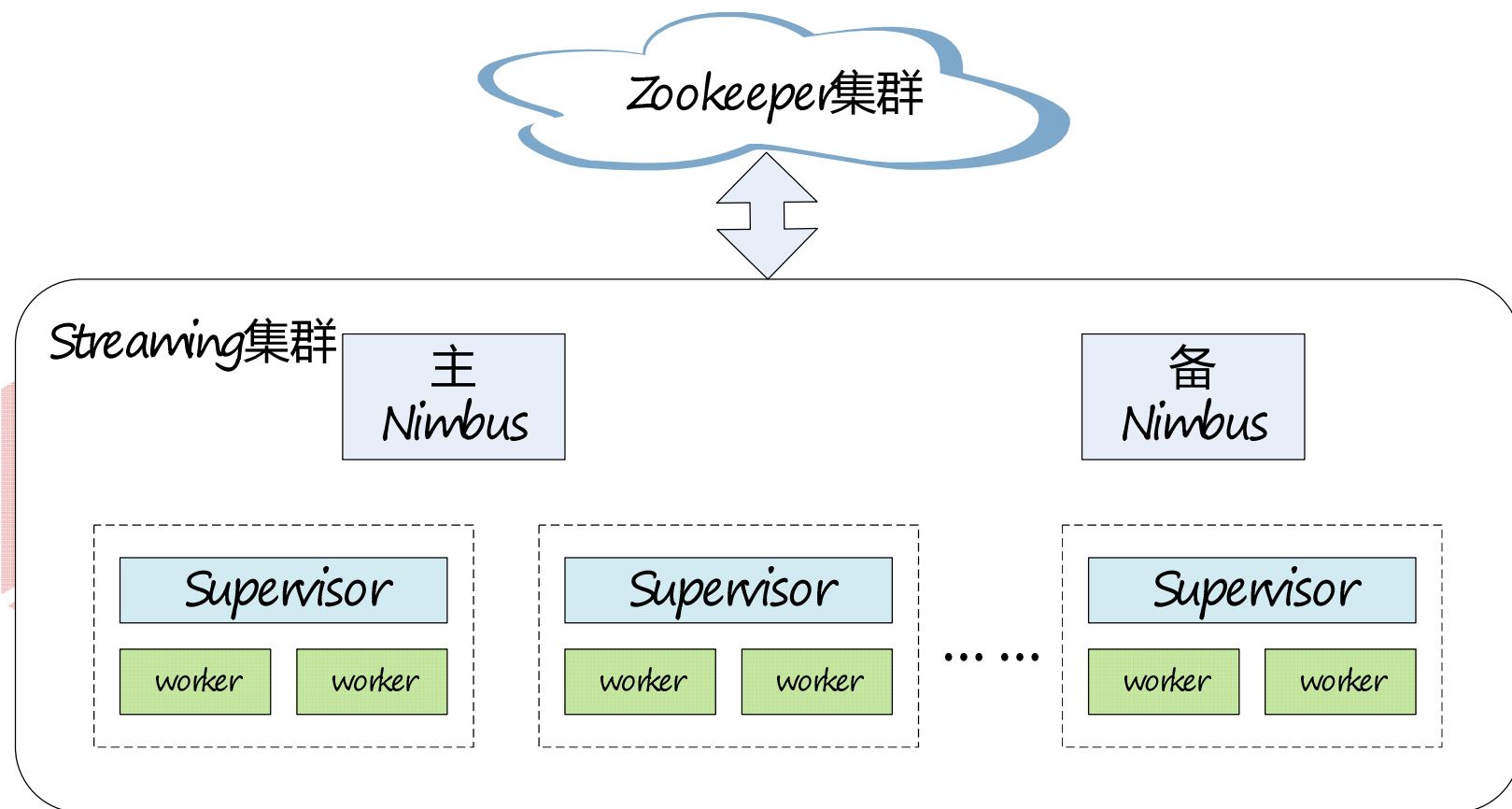
3. 关键特性介绍

4. StreamCQL介绍



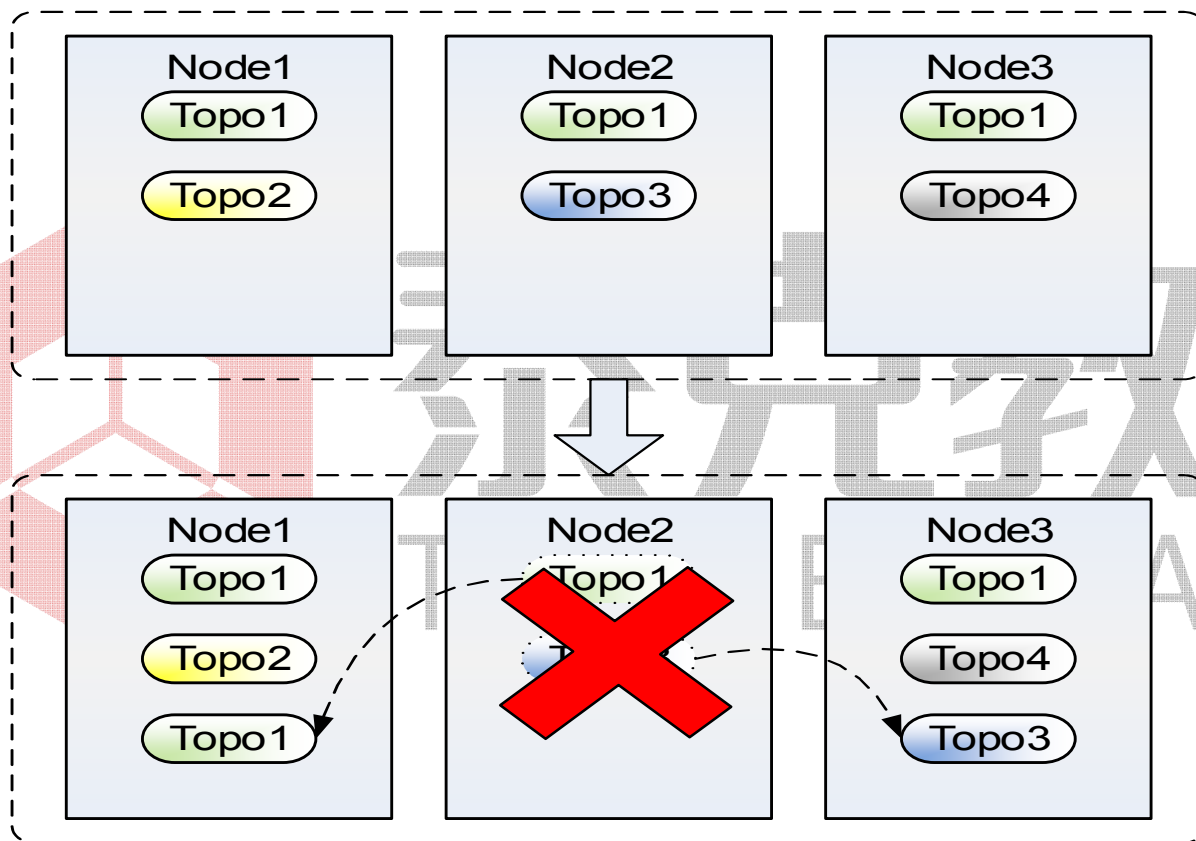
泰克教育
TECH EDUCATION

Nimbus HA



容灾能力

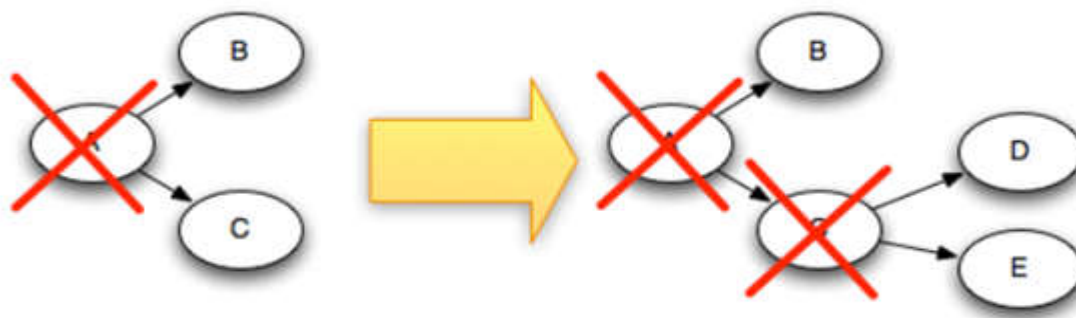
- 容灾能力：节点失效，自动迁移到正常节点，业务不中断。



消息可靠性

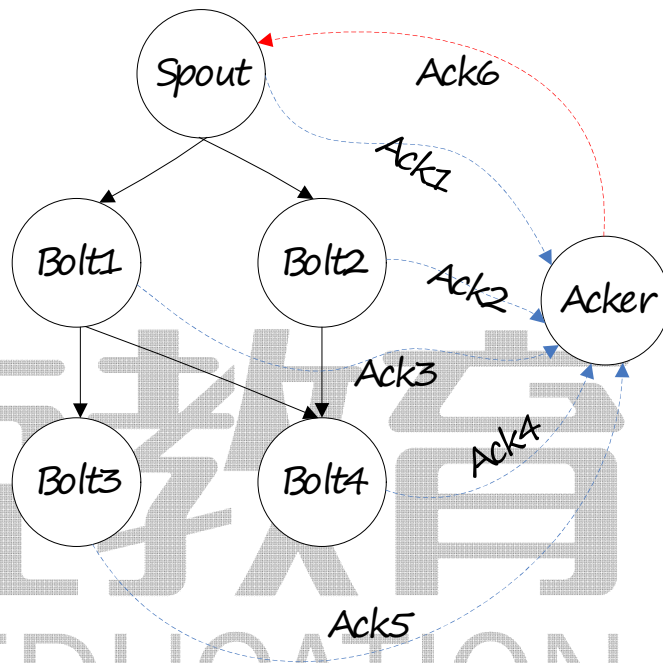
可靠级别	处理机制	说明
最多一次	无	吞吐量最大，适用于消息可靠性较低的场景。
最少一次	Ack机制	吞吐量较低，要求数据被完整处理，适用于消息可靠性要求高的场景。
精确一次	Trident	Trident是Streaming提供的特殊的事务性API，吞吐量最低。

- 在streaming里面一个tuple被完全处理的意思是：这个tuple以及由这个tuple所派生的所有的tuple都被成功处理。如果这个消息在Timeout所指定的时间内没有成功处理，这个tuple就被认为处理失败了。



ACK机制

- Spout发送一个Tuple时，会通知Acker一个新的根消息产生了，Acker会创建一个新的tuple tree，并初始化校验和为0。
- Bolt发送消息时向Acker发送anchor tuple，刷新tuple tree，并在发送成功后向Acker反馈结果。如果成功则重新刷新校验和，如果失败则Acker会立即通知Spout处理失败。
- 当tuple tree被完全处理（校验和为0），Acker会通知Spout处理成功。
- Spout提供ack()和fail()接口方法用于处理Acker的反馈结果，需要用户实现。一般在fail()方法中实现消息重发逻辑。

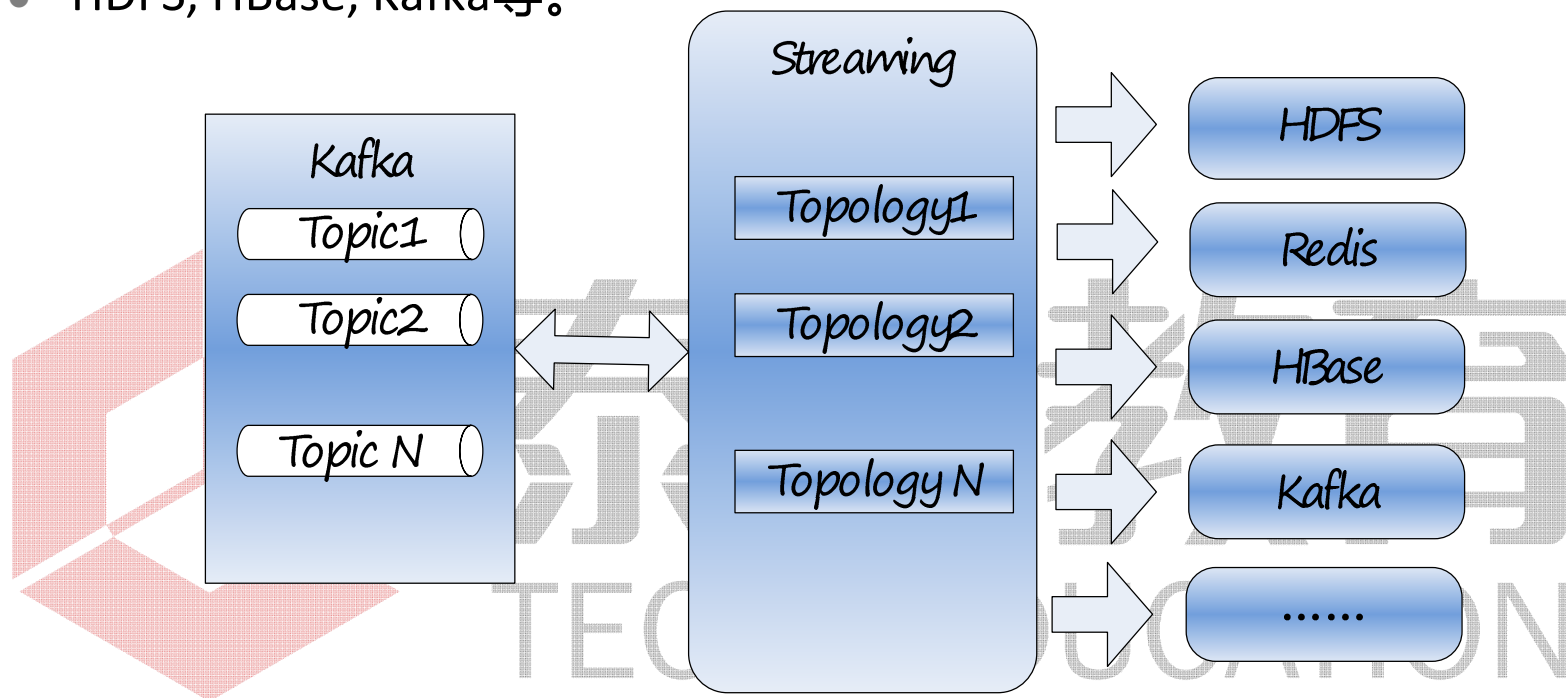


可靠性级别设置

- 如果并不要求每个消息必须被处理（允许在处理过程中丢失一些信息），那么可以关闭消息的可靠处理机制，从而可以获取较好的性能。
- 有三种方法可以关闭消息的可靠处理机制：
 - 将参数Config.Topology_ACKERS设置为0。
 - Spout发送一个消息时，使用不指定消息messageID的接口进行发送。
 - Bolt发送消息时使用Unanchor方式发送，使Tuple树不往下延伸，从而关闭派生消息的可靠性。

Streaming与其他组件

- HDFS, HBase, Kafka等。

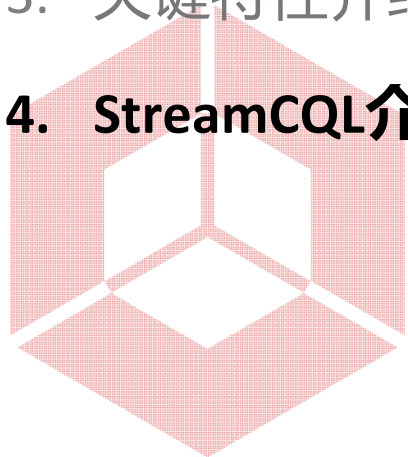


- 整合HDFS/Hbase等外部组件，将实时结果提供给其他组件，进行离线分析。



目录

1. Streaming简介
2. 系统架构
3. 关键特性介绍
- 4. StreamCQL介绍**



泰克教育
TECH EDUCATION

StreamCQL简介

- StreamCQL(Stream Continuous Query Language)是建立在分布式流处理平台基础上的查询语言(CQL)，架构支持构建在多种流处理引擎之上，目前主要适配Streaming。
- StreamCQL提供了较丰富的分布式流计算功能，除了具有过滤、转换等传统的SQL基本能力之外, StreamCQL引入基于流的时间窗口的计算，提供窗口数据的统计、关联等能力，以及流数据的拆分、合并等功能。

StreamCQL简介

- StreamCQL(Stream Continuous Query Language)是建立在分布式流处理平台基础上的查询语言(CQL)，架构支持构建在多种流处理引擎之上，目前主要适配Streaming。
- StreamCQL提供了较丰富的分布式流计算功能，除了具有过滤、转换等传统的SQL基本能力之外, StreamCQL引入基于流的时间窗口的计算，提供窗口数据的统计、关联等能力，以及流数据的拆分、合并等功能。

StreamCQL易开发

```
//Def Input:
public void open(Map conf,
TopologyContext context,
SpoutOutputCollector collector) {...}
public void nextTuple() {...}
public void ack(Object id) { ...}
public void
declareOutputFields(OutputFieldsDeclarer declarer) {...}
//Def logic:
public void execute(Tuple tuple,
BasicOutputCollector collector) {...}
    public void
declareOutputFields(OutputFieldsDeclarer ofd) {...}
//Def Output:
    public void execute(Tuple tuple,
BasicOutputCollector collector) {...}
        public void
declareOutputFields(OutputFieldsDeclarer ofd) {...}
//Def Topology:
public static void main(String[] args)
throws Exception {...}
```

Storm原生API

```
--Def Input:
CREATE INPUT STREAM S1 ...

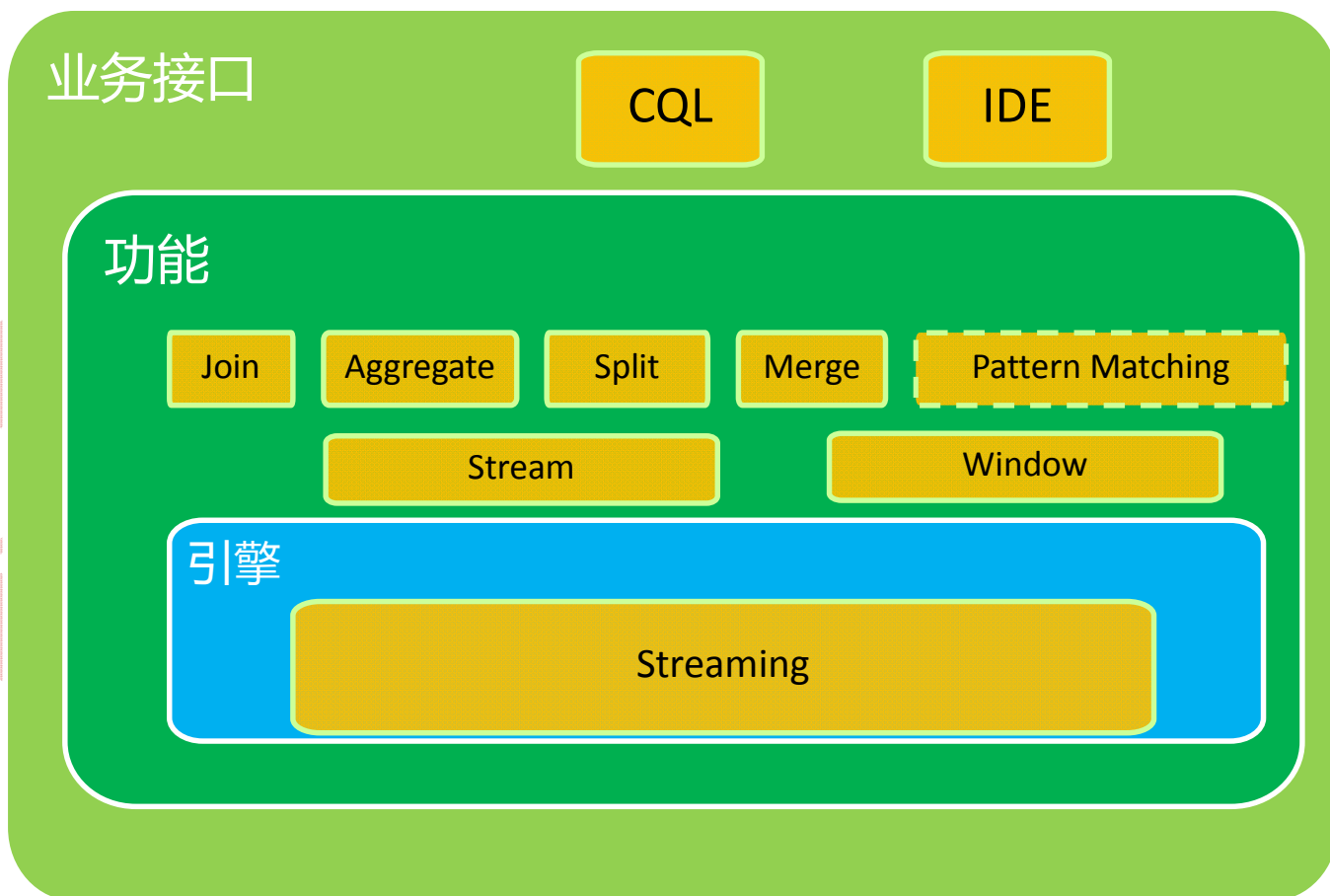
--Def logic:
INSERT INTO STREAM filterstr SELECT *
FROM S1 WHERE name="HUAWEI";

--Def Output:
CREATE OUTPUT STREAM S2...

--Def Topology:
SUBMIT APPLICATION test;
```

StreamCQL

StreamCQL与流处理平台





本章总结

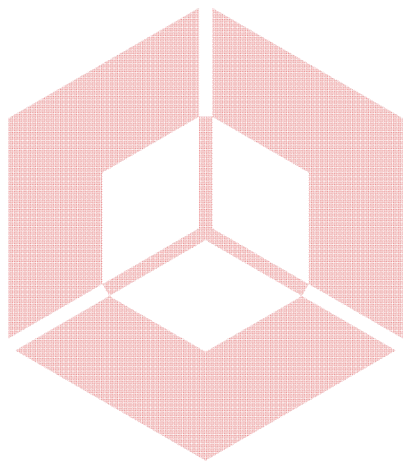
- 本章主要介绍了如下内容：

- Streaming定义。
- 描述了Streaming的应用场景。
- Streaming在FusionInsight产品的位置。
- Streaming的系统架构。
- Streaming的关键特性介绍。
- StreamCQL介绍。



思考题

1. Streaming是如何保障消息可靠性？

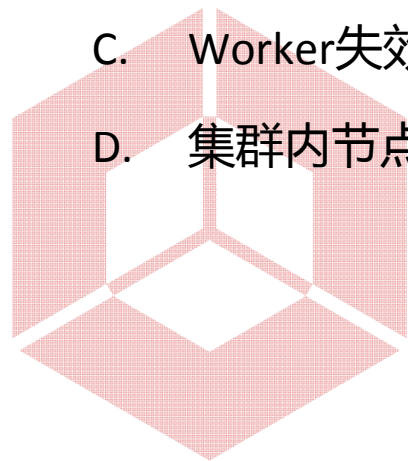


泰克教育
TECH EDUCATION



思考题

1. 以下对与Supervisor的描述正确的是()。
- A. Nimbus 主备部署，解决Nimbus单点问题，支持主从热切换。
 - B. Supervisor失效后能够自动恢复，并且不影响正在运行的业务。
 - C. Worker失效后能够自动恢复，继续运行。
 - D. 集群内节点失效后在该节点上的任务会被重分配到其他正常节点。

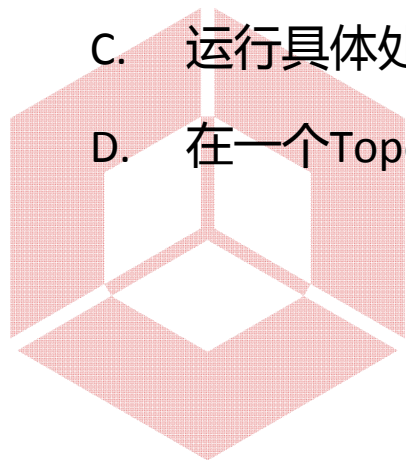


泰克教育
TECH EDUCATION



思考题

2. 以下对与Supervisor的描述正确的是()。
- A. Supervisor负责资源分配和任务调度。
 - B. 负责接受Nimbus分配的任务,启动和停止属于自己管理的worker进程。
 - C. 运行具体处理组件逻辑的进程。
 - D. 在一个Topology中接受数据然后执行处理的组件。

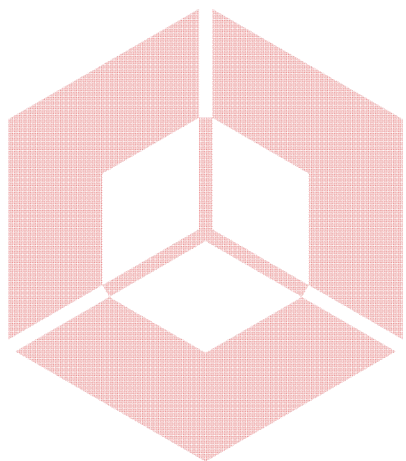


泰克教育
TECH EDUCATION



更多信息

- 下载培训资料：
 - <http://support.huawei.com/learning/trainFaceDetailAction?lang=zh&pbiPath=term1000025185&courseId=Node1000009072>
- eLearning课程：
 - <http://support.huawei.com/learning/nodeQueryAction!loadTrainProjectInfo?lang=zh&pbiPath=term1000025185&courseId=Node1000009421&navId=MW000001>
- 考试大纲：
 - <http://support.huawei.com/learning/Certificate!toExamOutlineDetail?lang=zh&nodeId=Node1000003516>
- 模拟考试：
 - <http://support.huawei.com/learning/Certificate!toSimExamDetail?lang=zh&nodeId=Node1000004285>
- 认证流程：
 - [http://support.huawei.com/learning/NavigationAction!createNavi#navi\[id\]=_40](http://support.huawei.com/learning/NavigationAction!createNavi#navi[id]=_40)



谢谢

www.huawei.com

泰克教育

TECH EDUCATION