Kafka 学习笔记

Kafka Streams （一） 了解

概述

Kafka Streams是一个客户端程序库，用于处理和分析存储在Kafka中的数据，并将得到的数据写回Kafka或发送到外部系统。Kafka Stream基于一个重要的流处理概念。如正确的区分事件时间和处理时间，窗口支持，以及简单而有效的应用程序状态管理。Kafka Streams的入口门槛很低: 你可以快速的编写和在单台机器上运行一个小规模的概念证明（proof-of-concept）；而你只需要运行你的应用程序部署到多台机器上，以扩展高容量的生产负载。Kafka Stream利用kafka的并行模型来透明的处理相同的应用程序作负载平衡。

亮点

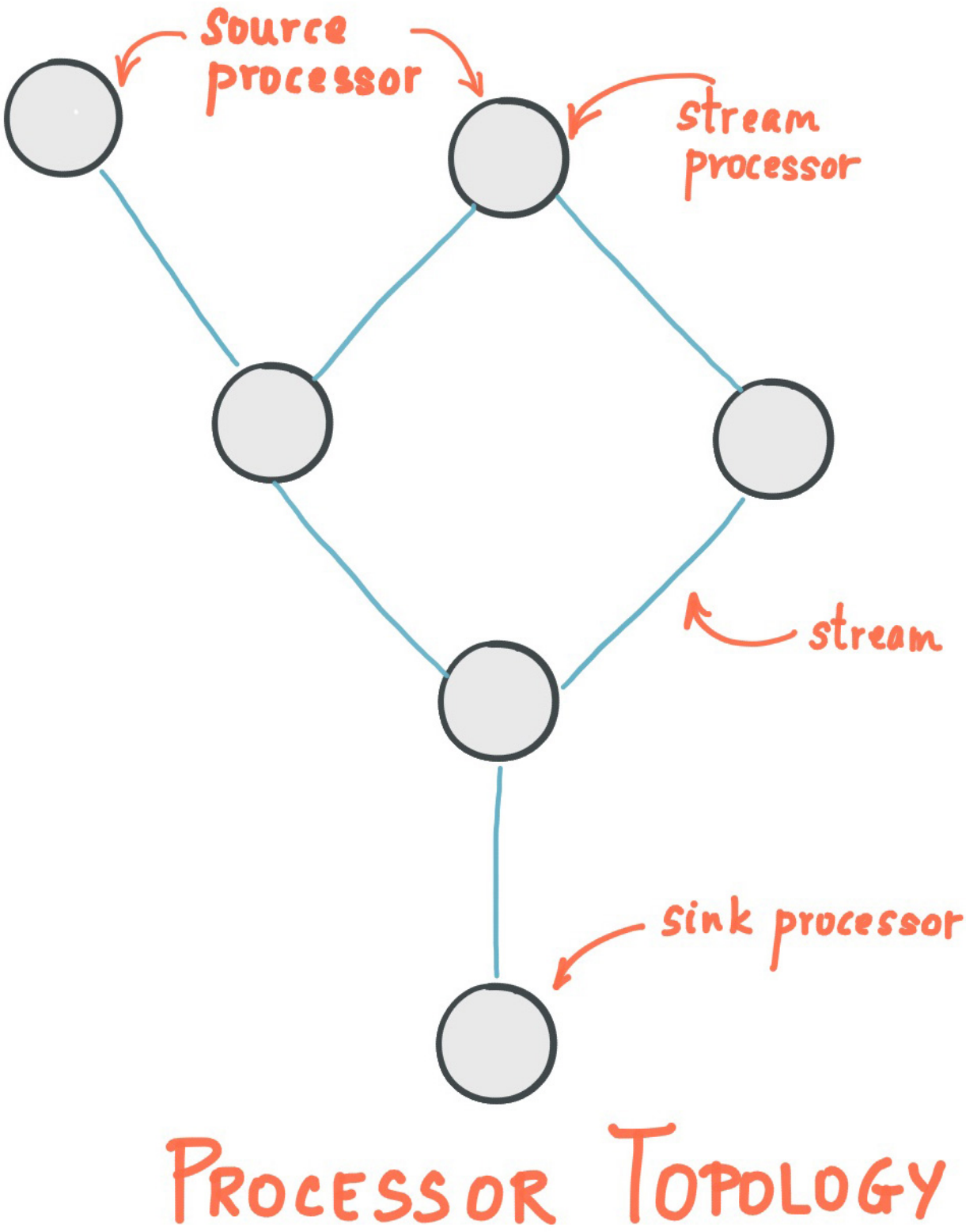
- 设计一个简单的、轻量级的客户端库，可以很容易地嵌入在任何java应用程序与任何现有应用程序封装集成。
- Apache Kafka本身作为内部消息层，没有外部系统的依赖，还有，它使用kafka的分区模型水平扩展处理，并同时保证有序。
- 支持本地状态容错，非常快速、高效的状态操作（如join和窗口的聚合）。
- 采用 one-recored-at-a-time（一次一个消息） 处理以实现低延迟，并支持基于事件时间(event-time)的窗口操作。
- 提供必要的流处理原语(primitive)，以及一个 高级别的Steram DSL 和 低级别的Processor API。

Stream处理拓扑

在消息消费的时候 Kafka 使用了 zero copy。

- 流是Kafka Stream提出的最重要的抽象概念：它表示一个无限的，不断更新的数据集。流是一个有序的，可重放（反复的使用），不可变的容错序列，数据记录的格式是键值对（key-value）。
- 通过Kafka Streams编写一个或多个的计算逻辑的处理器拓扑。其中处理器拓扑是一个由流（边缘）连接的流处理（节点）的图。
- 流处理器 是处理器拓扑中的一个节点；它表示一个处理的步骤，用来转换流中的数据（从拓扑中的上游处理器一次接受一个输入消息，并且随后产生一个或多个输出消息到其下游处理器中）。
- 源处理器（Source Processor）： 源处理器是一个没有任何上游处理器的特殊类型的流处理器。它从一个或多个kafka主题生成输入流。通过消费这些主题的消息并将它们转发到下游处理器。

- Sink处理器：sink处理器是一个没有下游流处理器的特殊类型的流处理器。它接收上游流处理器的消息发送到一个指定的Kafka主题。



- Kafka streams提供2种方式来定义流处理器拓扑：Kafka Streams DSL提供了更常用的数据转换操作，如map和filter；低级别Processor API允许开发者定义和连接自定义的处理器，以及和状态仓库交互。

时间

在 流处理 方面有一个重要的时间概念，以及它是如何建模和集成。例如：一些操作，如基于时间界限定义的窗口。

- 时间在流中的常见概念如下：
- 事件时间 - 当一个事件或数据记录发生的时间点，就是最初创建的“源头”。
- 处理时间 - 事件或数据消息发生在流处理应用程序处理的时间点。即，记录已被消费。处理时间可能是毫秒，小时，或天等。比原始事件时间要晚。
- 摄取时间 - 事件或数据记录是Kafka broker存储在topic分区的时间点。与事件时间的差异是，当记录由Kafka broker追加到目标topic时，生成的摄取时间戳，而不是消息创建时间（“源头”）。与处理时间的差异是处理时间是流处理应用处理记录时的时间。比如，如果一个记录从未被处理，那么久没有处理时间，但仍然有摄取时间。

`Kafka Streams` 通过 `TimestampExtractor` 接口为每个数据记录分配一个时间戳。该接口的具体实现了基于数据记录的实际内容检索或计算获得时间戳，例如嵌入时间戳字段提供的事件时间语义，或使用其他的方法，比如在处理时返回当前的 `wall-clock` （墙钟）时间，从而产生了流应用程序的处理时间语义。因此开发者可以根据自己的业务需要选择执行不同的时间。例如，每条记录时间戳描述了流的时间增长（尽管记录在stream中是无序的）并利用时间依赖性来操作，如join。

最后，当一个Kafka Streams应用程序写入记录到kafka时，它将分配时间戳到新的消息。时间戳分配的方式取决于上下文：

- 当通过处理一些输入记录（例如，在process（）函数调用中触发的context.forward（））生成新的输出记录时，输出记录时间戳直接从输入记录时间戳继承。
- 当通过周期性函数（如punctuate()）生成新的输出记录时。输出记录时间戳被定义为流任务的当前内部时间（通过context.timestamp()获取）。
- 对于聚合，生成的聚合更新的记录时间戳将被最新到达的输入记录触发更新。