Stream消息流和Stream Grouping消息流组

Stream消息流是Storm中最关键的抽象，是一个没有边界的Tuple序列。

Stream Grouping消息流组是用来定义一个流如何分配到Tuple到Bolt。

Storm核心的抽象概念是“流”。流是一个分布式并行创建和处理的无界的连续元组（Tuple）。流通过给流元组中字段命名来定义。在默认情况下，元组可以包含整型、长整型、短整型、字节、字符串、双精度浮点数、单精度浮点数、布尔型和字节数组。

Stream消息流

Stream消息流是Storm中最关键的抽象，是一个没有边界的Tuple序列，这些Tuple以分布式的方式并行地创建和处理。源源不断传递的元组Tuple就组成了流Stream，定义消息流主要是定义消息流中的Tuple。Tuple的定义在前面已经做了详细介绍，这里不再累述。消息流Tuple中的每个字段都有一个名字，并且不同Tuple对应字段的类型必须相同。两个Tuple的第一个字段的类型必须相同，第二个字段的类型必须相同，但是第一个字段和第二个字段可以有不同的类型。

每个消息流在定义时都会分配一个ID，因为单向消息流很普遍，OutputFieldsDeclarer定义了一些方法可以定义一个流而不用指定其ID。在这种情况下，该流有一个默认的ID。

Storm将每个待处理或者新产生的Tuple封装成“消息”，而一个消息流（Stream）则是一个没有边界的Tuple序列，而这些Tuple会以一种分布式的方式被并行地创建和处理。

Stream Grouping消息流组

定义Topology的其中一步是定义每个Bolt接受何种流作为输入。Stream Grouping（消息流组）就是用来定义一个流如何分配Tuple到Bolt。Storm包括6种流分组类型。

1. 随机分组（Shuffle Grouping）

随机分发元组到Bolt的任务，保证每个任务获得相等数量的元组。

1. 字段分组（Fields Grouping）

根据指定字段分割数据流并分组。例如，根据”user-id”字段，具有该字段的Tuple被分到相同的Bolt，不同的”user-id”值则会被分配到不同的Bolt。

1. 全部分组（All Grouping）

对于每一个Tuple来说，所有的Bolt都会收到，所有的Tuple被复制到Bolt的所有任务上，需小心使用该分组。

1. 全局分组（Global Grouping）

全部的流都分配到Bolt的同一个任务，就是分配给ID最小的Task。

1. 无分组（None Grouping）

不分组的含义是，流不关心到底谁会收到它的Tuple。目前无分组等效于随机分组，不同的是Storm将把无分组的Bolt放到订阅Bolt或Spout的同一线程中执行（在可能实现的前提下）。

1. 直接分组（Direct Grouping）

这是一个特别的分组类型。元组生产者决定元组由哪个元组消费者任务接收。该分组仅能被声明为direct stream的流使用。元组必须通过emitDirect方法直接发射。Bolt获取消费者任务ID，可以使用已提供的TopologyContext类或保持引用OutputCollector类的emit方法的输出，元组发送后返回任务ID列表。





