事务

Storm中的事务是专门针对Topology提出来的，是为了解决元组在处理失败重新发送后的一系列问题的。简而言之，事务拓扑就是指Storm以并行和顺序处理混合的方式处理元组，一方面是处理消息，将消息分为一批批，并行处理同一批内的消息以及批与批之间的消息；另一方面是提交消息，按严格的顺序提交，总之实现对消息精确的处理。

1. 事务拓扑的特点

事务拓扑这一特性使消息在语义上确保能够以安全的方式重发消息，并保证它们只被处理一次。在不支持事务性拓扑的情况下，无法在保证准确性、可扩展性、容错性的前提下完成计算。

1. 事务拓扑的目的

事务拓扑的目的是满足对消息处理由着及其严格要求的场景，如实时计算某个用户的页面点击次数，要求结果完全精确。

1. 事务拓扑的原理

事务拓扑是Storm 0.7引入的特性，0.8之后版本已经被封装为Trident，提供了更加方便和直观的接口。

Storm的事务拓扑是完全基于底层的Spout/Bolt/Acker原语实现的，通过一层巧妙的封装得出一个优雅的实现，这也是Storm最大的魅力之一。

Storm使用ZooKeeper储存事务元数据，默认就是拓扑使用的ZooKeeper。可以通过修改transactional.zookeeper.servers和transactional.zookeeper.port配置参数键指定其他的ZooKeeper。

在使用事务性拓扑时，数据源要能够重发，有时候甚至要重复多次，因此需要确认数据源Spout是否具备这项能力。