

Twitter推出新的流处理器系统Heron

作者 [张天雷](#) 发布于 2015年6月16日 | [讨论](#)

背景：

2011年，Twitter发布了开源的分布式流计算系统Storm。四年后，随着用户数量的急剧增加，Twitter每天要处理的事件已经增加到十亿以上。Storm系统应对如此庞大而复杂多样的流数据变得十分困难。为了解决该问题，Twitter公司近期开发了一套全新的流处理系统——Heron。近日，Twitter公司在SIGMOD 2015会议上对Heron进行了介绍。

问题及需求分析：

据Twitter公司的技术经理Karthik Ramasamy表示，Twitter公司之前对Storm所存在的问题以及新平台的功能需求进行了详细分析。首先，Storm调试能力较弱的问题曾让Twitter员工比较困扰。在Storm中，一个拓扑中的多个组件是捆绑在操作系统的一个进程中的。这就使得在拓扑出现问题时很难迅速定位问题的根源。用户不得不借助cleaner映射来帮助实现逻辑计算单元到物理进程的映射，从而辅助调试。此外，Storm还需要专门的集群资源来运行拓扑。这就使得它不能利用流行的集群调度软件进行计算资源的优化。而且，在使用Storm时，用户需要手动隔离机器才能部署一个新的产品拓扑。同样，在拓扑不再使用时还需要手动回收机器资源。所有Storm存在的这些问题严重限制了Twitter处理事件的能力，而且带来时间和计算资源的巨大浪费。因此，Twitter认为新的系统需要具备能够每分钟处理上亿的事件、大规模处理数据的延迟为亚秒级、行为可预测、高数据精度、遇到局部流量过高或流水线拥堵时能够自行调整输入速率、调试方便以及能够在共享式框架中部署等功能。

方案选择：

据Karthik透露，Twitter当初考虑了三种可能的方案——扩展现有的Storm系统、利用别的已经开源的系统和开发一套新的系统。然而，Storm系统的核心部件并不能满足现有的需求，而对其进行修改又需要比较长的开发周期。同时，其他的开源流处理框架不能满足Twitter在规模、吞吐量以及延迟等方面的需求。更关键的是，它们不能与Storm的API相兼容，迁移工作会十分复杂。因此，Twitter最终决定开发一套全新的实时流处理系统。

实现概述：

根据设计要求，Heron保持了与Storm相同的数据模型和API，运行的也是由spout和bolt构成的拓扑。其总体框架包含了一个调度器和若干个拓扑。该调度器只是一个抽象模型，可以具体化为YARN、Mesos或者ECS等，方便资源共享。用户利用API提交拓扑到调度器后，调度器把每个拓扑当作一个单独的任务，并根据集群中节点利用情况分派若干个容器来执行。在这些容器中，其中一个运行拓扑master，负责管理拓扑。剩余的每一个容器都需要运行一个流管理器来负责数据路由、一个metric管理器负责收集和报告各种各样的metric以及若干个Heron实例进程来运行用户自定义的spout/bolt代码。最后，拓扑的元数据，如物理规划和执行细节等都被保存在ZooKeeper中。

新系统所达成的目标：

因此，Heron能够轻松部署在运行如Mesos、YARN或者自定义调度框架的共享架构中。而且，Heron向后兼容Storm，使得原来基于Storm的其它系统可以继续使用。在Heron中运行已有的Storm拓扑完全不需要任何改变，移除了复杂的迁移过程。容器中的Heron实例执行的是单独的任务，保证了用户利用jstack或者heap dump等工具即可进行实例的调试。Metric收集器又使得拓扑中任何组件的失效变得十分明显，大大降低了调试的难度。此外，Heron有一个背压机制能够在运行过程中动态调整拓扑中数据流的速度，而不影响数据精度。同时，与2013年10月发布的Storm相比，Heron的吞吐量可以到达其10-14

倍，而延迟时间却只是它的1/15-1/5，资源消耗也更低。

目前，Heron已经作为Twitter的主要流处理器系统在运行，其中包括了数百个拓扑。由于Heron的低资源消耗特性，迁移后的新系统硬件减少了2/3，大大提高了物理资源的利用率。Karthik也表示，Twitter公司非常愿意与Storm社区或者其他开源的实时流处理系统社区分享Heron的经验和教训。