HORTONWORKS®

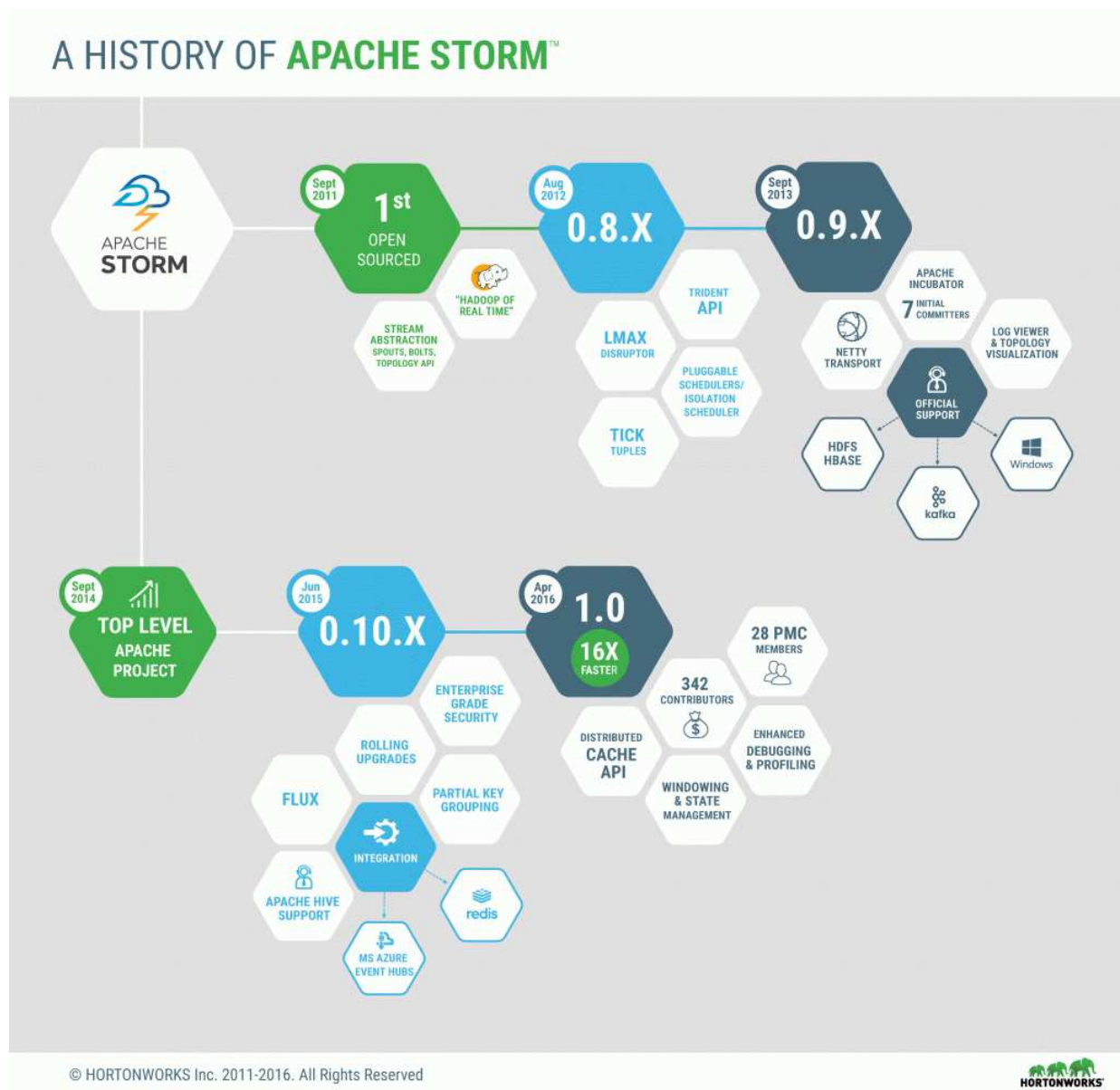Products   Solutions   Customers   Services & Support

May 18, 2016

A BRIEF HISTORY OF APACHE STORM

by Guest Author

PART 1: A LITTLE HISTORY

In this series of blog posts, we will provide an in-depth look select features introduced with the release of Apache Storm (Storm) 1.0. To kick off the series, we'll take a look how Storm has evolved over the years from its beginnings as an open source project, up to the 1.0 milestone release.

"THE HADOOP OF REAL-TIME"

Storm was originally created by Nathan Marz while he was at Backtype (later acquired by Twitter) working on analytics products based on historical and real-time analysis of the Twitter firehose. Nathan envisioned Storm as a

MENU   news-blogs   SIGN-UP ✉   Contact Sales? 💬

SHARE   in   f   🐦

concept of the "stream" as a distributed abstraction for data in motion, as well as a fault tolerance and reliability model that was difficult, if not impossible, to achieve with a traditional queues and workers architecture.

Nathan open sourced Storm to GitHub on September 19th, 2011 during his talk at Strange Loop, and it quickly became the most watched JVM project on GitHub. Production deployments soon followed, and the Storm development community rapidly expanded.

At the time Storm was introduced, Big Data analytics largely involved batch processing in map-reduce on Apache Hadoop or one of the higher level abstractions like Apache Pig and Cascading. The introduction of Storm helped drive a change in the way people thought about large scale analytics, spurring the rise of stream processing and real-time analytics.

Early versions of Storm introduced the familiar stream abstraction and the corresponding Spout/Bolt/Topology API that allowed developers to easily reason about streaming computations. Another feature Storm introduced, and one that to this day remains unique to Storm, is the concept of Distributed Remote Procedure Calls where the inherent parallelism and scalability of Storm can be leveraged in a synchronous, request-response paradigm.

The Storm 0.8.x line of releases introduced the Trident API which added support for exactly-once semantics, micro-batching for increased throughput, stateful processing, and a high-level API for joins, aggregations, grouping, functions, and filters. Other improvements at the time included pluggable schedulers, introduction of the LMAX Disruptor for higher throughput, tick tuples, and improvements to the Storm UI.

STORM MOVES TO APACHE

With encouragement from Andy Feng at Yahoo!, Nathan decided to propose moving Storm to Apache, and the project officially entered the Apache Incubator on September 18, 2013. This move marked the beginning of a fundamental shift in the Storm community away from a model where a single

MENU   **news-blogs**   SIGN-UP ✉   Contact Sales? 💬

SHARE   in   f   🐦

model. The move to Apache ensured that the project would be governed by a sustainable community, and that no single individual would represent a bottleneck in terms of decision making.

During Storm's time in the Apache Incubator, the 0.9.x line of releases was introduced. In 0.9.x Storm's underlying transport layer based on 0mq was replaced with an implementation based on Netty. Not only was the new Netty-based transport almost twice as fast as the previous implementation, but the fact that Netty is a pure Java framework freed users from the requirement of difficult to install, platform specific binaries. Finally, the Netty transport set the stage for authorization and authentication between worker processes.

The 0.9.x line of releases also introduced Apache Kafka integration as a first class component in the Apache Storm distribution. Prior to this point Kafka integration was a separate project that had been forked many times, and it was difficult for users to understand which versions were compatible with various versions of Storm and Kafka. Bringing Kafka integration into the Apache distribution ensured that compatibility was maintained with each Storm release. Similarly, the Storm community added support for HDFS and Apache HBase integration as well.

Other notable improvements from this time included performance improvements to the Netty transport, pluggable serialization for multi-lang components, topology visualization, a logviewer, and support for Microsoft Windows deployments.

Apache Storm became a top-level Apache project on September 17, 2014. In the time since first entering the Apache Incubator with a group 7 initial committers, the Apache Storm PMC has grown to 28 members strong with contributions from 342 individuals.

ENTERPRISE READINESS

Before graduating from the Apache Incubator, the Storm community was already working on the next major iteration of the platform: Version 0.10.

MENU   **news-blogs**   SIGN-UP ✉   Contact Sales? 💬

SHARE in f 🐦

early days of Apache Hadoop, Apache Storm originally evolved in an environment where security was not a high-priority concern. Rather, it was assumed that Storm would be deployed to environments suitably cordoned off from security threats. While a large number of users were comfortable setting up their own security measures for Storm, this proved a hindrance to broader adoption among larger enterprises where security policies prohibited deployment without specific safeguards.

The implementation of enterprise-grade security in Apache Storm was a momentous effort that involved active collaboration between Yahoo!, Hortonworks, Symantec, and the broader Apache Storm community. Some of the highlights of Storm's security features include:

> Kerberos Authentication with Automatic Credential Push and Renewal

> Pluggable Authorization and ACLs

> Multi-Tenant Scheduling with per-user isolation and configurable resource limits.

> User Impersonation

> SSL Support for Storm UI, Log Viewer, and DRPC (Distributed Remote Procedure Call)

> Secure integration with other Hadoop Projects (such as ZooKeeper, HDFS, HBase, etc.)

> User isolation (Storm topologies run as the user who submitted them)

Another important feature of Storm 0.10 was the introduction of Flux, a framework and set of utilities that make defining and deploying Storm topologies less developer-intensive. Prior to Flux, a common complaint from users was that the definition of a topology DAG was often tied up in java code, and any change required recompilation and packaging of the topology. Flux addressed that problem by providing a YAML DSL for defining topologies in a simple text file, decoupling the DAG definition from Java code.

Some of the key features of Flux include:

Easily configure and deploy Storm topologies (Both Storm core and Micro-batch API) without embedding configuration in your topology code

Support for existing topology code

Define Storm Core API (Spouts/Bolts) using a flexible YAML DSL

YAML DSL support for most Storm components (storm-kafka, storm-hdfs, storm-hbase, etc.)

Convenient support for multi-lang components

External property substitution/filtering for easily switching between configurations/environments (similar to Maven-style `${variable.name}` substitution)

Development of the 0.10 line of releases also saw a proliferation of additional integration components including JDBC/RDBMS integration, streaming ingest to Apache Hive, Microsoft Azure Event Hubs integration, and Redis support. Other important features included rolling upgrade support, logging improvements, and a partial key groupings implementation.

1.0 MILESTONE

On April 12, 2016 the Apache Storm community announced the release of Apache Storm 1.0, representing yet another major milestone in the evolution of the project. Version 1.0 includes a tremendous number of new features, and usability, management, and performance improvements.

In the coming weeks we will continue this blog series with more in-depth articles covering the important new features included in 1.0, including:

Performance improvements

Windowing and State Management

MENU   news-blogs   SIGN-UP ✉   Contact Sales? 💬

SHARE   in f 🐦 |

Management and Debugging improvements

Distributed Cache API

Automatic Backpressure Support

Resource Aware Scheduling

New Integration Components

For a sneak preview of these features, please see the Apache Storm 1.0 release announcement and stay tuned for more in-depth detail.

Tags:

HORTONWORKS   HADOOP   HDF   APACHE STORM

COMMENTS

- **streamanalytix** says:

  March 1, 2017 at 5:44 am
  I am extremely impressed with the view point. Thanks for sharing such useful information.

  Apache Spark Streaming

  Reply

- **sarika** says:

  April 14, 2017 at 3:35 am
  Its an awesome post and is quite meaningful.

  Reply

LEAVE A REPLY

Your email address will not be published. Required fields are marked *

Comment

MENU   **news-blogs**   SIGN-UP ✉   Contact Sales? 💬

       SHARE   in   f   🐦 |

Name *

Email *

Website

POST COMMENT

RELATED POSTS

9.11.17

Strata Data Conference New York – Powering...

The annual Strata Data Conference will make its next stop in New York the week of September 25-29. The core theme of the conference this year is around driving business transformations...

9.7.17

Top 5 Use Case

With the explosion businesses need to strategies and ado architecture. Build

VIEW ALL EVENTS   **VIEW ALL WEBCASTS**

Ecosystem   Press Room   Videos   Events   Webinars
            Be First Insights   Careers
                        1.408.675.0983 | 1.855.8.HORTON | 44 (0) 20 3826 1405

Apache, Hadoop, Falcon, Atlas, Tez, Sqoop, Flume, Kafka, Pig, Hive, HBase, Accumulo, Storm, Solr, Spark, Ranger, Knox, Ambari, ZooKeeper, Oozie, Phoenix, NiFi, HAWQ, Zeppelin, Atlas, Slider, Mahout, MapReduce, HDFS, YARN, Metron and the Hadoop elephant and Apache project logos are either registered trademarks or trademarks of the Apache Software Foundation in the United States or other countries.

Privacy Policy | Terms of Service | LCA Postings

MENU   **news-blogs**   SIGN-UP ✉   Contact Sales? 💬
            SHARE   in f ✆ |