HeronProposal

Heron Proposal

Abstract

Heron is a real-time, distributed, fault-tolerant stream processing engine initially developed by Twitter.

Proposal

Heron is a real-time stream processing engine built for high performance, ease of manageability, performance predictability and developer productivity[1]. We wish to develop a community around Heron to increase contributions and see Heron thrive in an open forum.

Background

Heron provides the ability for developers to compose directed acyclic graphs (DAGs) of real-time query execution logic (i.e. a topology) and submit the topology to execute on a pluggable job scheduling system (e.g., Apache Aurora, YARN, Marathon, etc). Users can employ either the native Heron API or the Apache Storm API to develop the topology. Heron supports the Storm API for ease of migration, but beyond that Heron's architecture differs considerably from Storm's.

Users submit a topology to the scheduler using the Heron client, which uses the Heron binary libraries to deploy all daemons required to run and manage the topology. The topology therefore has no reliance on centrally managed Heron services, only on a generic job scheduling system, which lends itself well to be run on top of Apache Aurora/Mesos or Apache Hadoop/YARN (among others).

The scheduler runs each topology as a job consisting of multiple containers. One of the containers runs the topology master, responsible for managing the topology. The remaining containers each runs a stream manager responsible for data routing, a metrics manager that collects and reports various metrics and a number of processes called Heron instances which run the user-defined logic on the stream of tuples. Parallelism is achieved via process-based isolation of Heron instances, which provides predictable performance while simplifying debugging. The containers are allocated and managed by the scheduler framework based on resource availability of nodes in the cluster. The metadata for the topology, such as the physical plan and execution details, are stored in the pluggable Heron State Manager (e.g. Apache ZooKeeper).

Rationale

Heron is a general-purpose, modular and extensible platform that can be leveraged to support common, real-time analytics use cases. There is an increasing demand for open-source, scalable real-time analytics systems. We believe that Heron can be leveraged by other organizations to build streaming applications that can benefit from its robustness, high performance, adaptability to cloud environments and ease of use. Moreover, we hope that open-sourcing Heron will help to further evolve the technology as the project attracts contributors with diverse backgrounds and areas of expertise.

We believe the Apache foundation is a great fit as the long-term home for Heron, as it provides an established process for community-driven development and decision making by consensus. This is exactly the model we want for future Heron development.

Initial Goals

- Move the existing codebase, website, documentation, and mailing lists to Apache-hosted infrastructure.
- Integrate with the Apache development process.
- Ensure all dependencies are compliant with Apache License version 2.0.
- Incrementally develop and release per Apache guidelines.

Current Status

Heron is a stable project used in production at Twitter since 2014 and open sourced under the ASL v2 license in 2016. The Heron source code is currently hosted at github.com (https://github.com/twitter/heron), which will seed the Apache git repository.

Meritocracy

By submitting this incubator proposal, we're expressing our intent to build a diverse developer community around Heron that will conduct itself according to The Apache Way and use a meritocratic means of building it's committer base. Several companies and universities have already expressed interest in and contributed to Heron. Our goal is to grow the Heron community by encouraging open communication, contribution and participation of all types, and ensuring that contributors are recognized appropriately.

Community

Heron is currently being used by Twitter, Google, Machine Zone and ndustrial.io and has received significant contributions by Microsoft and Streamlio. By bringing Heron into the Apache ecosystem, we believe we can attract even more developers who are interested in creating real-time systems to build the project's contributor base.

Core Developers

Current core developers are engineers from Twitter, Google, Microsoft and Streamlio.

Alignment

Heron utilizes a number of Apache technologies. Heron leverages Apache ZooKeeper for coordination and has scheduler implementations to integrate with Apache Mesos, Apache Aurora and Apache Hadoop's YARN (via Apache REEF) as well as spout implementations to integrate with Apache Kafka and metrics implementations to integrate with Scribe. Heron also implements the Apache Storm user-level API, which allows topologies written against Storm to run in Heron. We believe that having Heron at Apache will help further the growth of the streaming compute community, as well as encourage cooperation and developer cross pollination with other Apache projects.

Known Risks

Orphaned Products

The risk of the Heron project being abandoned is minimal. It is used in production at Twitter and Google and other companies are evaluating or adopting it for production use.

Inexperience with Open Source

All of the core contributors to the project have considerable experience with open source software development. Bill Graham[2], Ashvin Agrawal[3] and Supun Kamburugamuve[4], committers on the project, are PMCs on other Apache projects and Bill and Ashvin have gone through the Apache incubator process. Twitter has already donated numerous projects to the ASF (e.g., Apache Mesos, Apache Aurora, Apache Parquet). We also plan to be mentored by experienced ASF members that can help with any roadblocks.

Homogenous Developers

Initial committers come from 5 separate organizations. Our intention is increase the diversity of contributing developers and their affiliations. To date github contributions have come from approximately 50 contributors from outside the Twitter team.

Reliance on Salaried Developers

It is expected that Heron development will occur on both salaried time and on volunteer time. The majority of initial committers are paid by their employers to contribute to this project. We are committed to recruiting additional committers from other organizations as well as non-salaried committers to join project.

Relationships with Other Apache Products

As mentioned in the Alignment section, Heron implements the Apache Storm API and integrates with multiple Apache schedulers (Apache Mesos, Apache Aurora and Apache Hadoop's YARN) as well as Apache ZooKeeper and Apache Thrift.

An Excessive Fascination with the Apache Brand

Heron's popularity is growing in the streaming compute space and we are long time supporters of the Apache brand. This proposal is not for the purpose of generating publicity through. Rather, the primary benefits to joining Apache are those of community building and open decision making outlined in the Rationale section.

Documentation

This proposal exists online as http://wiki.apache.org/incubator/HeronProposal. Extensive documentation can be found on github at https://twitter.github.io/heron and the source code is well documented.

Source and Intellectual Property Submission Plan

The Heron codebase is currently hosted on Github: https://github.com/twitter/heron. During incubation, the codebase will be migrated to Apache infrastructure. The source code is already ASF 2.0 licensed.

External Dependencies

All external libraries have ASF 2.0 compatible licenses except for pylint. The pylint library is GPL licensed, but is only used for pre-build Python style checks and is neither bundled with, nor relied upon by, the Heron source or binary release artifacts.

Cryptography

Heron does not use any cryptography libraries.

Required Resources

Mailing lists

- private@heron.incubator.apache.org (with moderated subscriptions)
- dev@heron.incubator.apache.org
- commits@heron.incubator.apache.org
- user@heron.incubator.apache.org

Subversion Directory

Git is the preferred source control system: git://git.apache.org/heron

Issue Tracking

JIRA: Heron (HERON)

Initial Committers

- Andrew Jorgensen (andrew at andrewjorgensen dot com)
- Ashvin Agrawal (ashvin at apache dot org)*
- Avrilia Floratou (avrilia dot floratou at gmail dot com)
- Bill Graham (billgraham at apache dot org)*
- Brian Hatfield (bmhatfield at gmail dot com)
- Chris Kellogg (cckellogg at gmail dot com)
- Huijun Wu (huijun dot wu dot 2010 at gmail dot com)
- Karthik Ramasamy (karthik at gmail dot com)
- Maosong Fu (maosongfu at gmail dot com)
- Neng Lu(freeneng at gmail dot com)
- Runhang Li (obj dot runhang at gmail dot com)
- Sanjeev Kulkarni (sanjeevrk at gmail dot com)
- Supun Kamburugamuve (supun at apache dot org)*
- Thomas Sun (tom dot ssf at gmail dot com)
- Yaliang Wang (yaliang dot w dot wang at ieee dot org)

Affiliations

- Andrew Jorgensen (Google)
- Ashvin Agrawal (Microsoft)
- Avrilia Floratou (Microsoft)
- Bill Graham (Twitter)
- Brian Hatfield (Google)
- Chris Kellogg (Twitter)
- Huijun Wu (Twitter)
- Karthik Ramasamy (Streamlio)
- Maosong Fu (Twitter)
- Neng Lu (Twitter)
- Runhang Li (Twitter)
- Sanjeev Kulkarni (Streamlio)
- Supun Kamburugamuve (Indiana University)
- Thomas Sun (Twitter)
- Yaliang Wang (Twitter)

Sponsors

Champion

- Julien Le Dem (julien at apache dot org)

Nominated Mentors

- Jake Farrell (jfarrell at apache dot org)
- Jacques Nadeau (jacques at apache dot org)
- Julien Le Dem (julien at apache dot org)
- P. Taylor Goetz (ptgoetz at apache dot org)

Sponsoring Entity

The Apache Incubator

Footnotes

- 1 - Papers detailing Heron are available at http://dl.acm.org/citation.cfm?id=2742788 and http://sites.computer.org/debull/A15dec/p15.pdf.
- 2 - http://home.apache.org/phonebook.html?uid=billgraham
- 3 - http://home.apache.org/phonebook.html?uid=ashvin
- 4 - http://home.apache.org/phonebook.html?uid=supun