MesosProposal

Abstract

Mesos is a cluster manager that provides resource sharing and isolation across cluster applications.

Proposal

Mesos is system for sharing resources between cluster applications such as Hadoop MapReduce, HBase, MPI, and web applications. It is motivated by three use cases. First, organizations that use several of these applications can use Mesos to share nodes between them, increasing utilization and simplifying management. Second, inspired by MapReduce, a wide array of new cluster programming frameworks are being proposed, such as Apache Hama, Microsoft Dryad, and Google's Pregel and Caffeine. Mesos provides a common interface for such frameworks to share resources, allowing organizations to use multiple frameworks in the same cluster. Third, Mesos allows users of a framework such as Hadoop to have multiple instances of the framework on the same cluster, facilitating workload isolation and incremental deployment of upgrades.

Background

Mesos was inspired by operational issues experienced in large Apache Hadoop deployments as well as a desire to provide a management system for a wider range of cluster applications. The Apache Hadoop community has long realized that the current model of having one instance of MapReduce control a whole cluster leads to problems with isolation (one job may cause the master to crash, killing all the other jobs), scalability, and software upgrades (an upgrade must be deployed on the whole cluster). Statically partitioning resources into multiple fixed-size MapReduce clusters is unattractive because it lowers both utilization and data locality. The community has discussed a two-level scheduling model where a simple, robust low-level layer enables multiple applications to launch tasks (https://issues.apache.org/jira/browse/MAPREDUCE-279). Mesos is such a layer, with the additional goal of supporting non-Hadoop applications as well.

Mesos started as a research project at UC Berkeley, but is now being tested at several companies (including Twitter and Facebook), and has attracted interest from other industry users and researchers as well. We are therefore proposing to place Mesos in the Apache incubator and build an open source community around it.

Rationale

Although a variety of cluster schedulers (e.g. Torque, Sun Grid Engine) already exist in the scientific computing community, they are not well suited for today's data center environment. These schedulers generally give jobs coarse-grained static allocations of the cluster (e.g. X nodes for the full duration of the job). This is problematic because many cluster applications are elastic (can scale up and down), so utilization is not optimal under static partitioning, and because data-intensive applications such as MapReduce need to run a few tasks on every node of the cluster to read data locally. To address these challenges, Mesos is designed around two principles:

- Fine-grained sharing: Mesos allocates resources at the level of "tasks" within a job, allowing applications to scale up and down over time and to take turns accessing data on cluster nodes.
- Application-controlled scheduling: Applications control which nodes their tasks run on, allowing them to achieve placement goals such as data locality.

In addition to these principles, Mesos is designed to be simple, scalable and robust, becuase a cluster manager must be highly available to support applications and should not become a bottleneck. Application-controlled scheduling already simplifies our design by pushing much of the complex logic of tracking job state to applications. In addition, Mesos employs an optimized C++ message-passing library to achieve scalability and supports master failover using Apache ZooKeeper.

Mesos already supports running Hadoop and MPI. We plan to add support for other systems as requested (and contributed) by the community.

Current Status

Meritocracy

Our intent with this incubator proposal is to start building a diverse developer community around Mesos following the Apache meritocracy model. We have wanted to make the project open source and encourage contributors from multiple organizations from the start. We plan to provide plenty of support to new developers and to quickly recruit those who make solid contributions to committer status.

Community

Mesos is currently being used by developers at Twitter and researchers in computer science and civil engineering at Berkeley. We hope to extend the user and developer base further in the future. The current developers and users are all interested in building a solid open source community around Mesos.

To work towards an open source community, we have been using the GitHub issue tracker and mailing lists at Berkeley for development discussions within our group for several months now.

Core Developers

Mesos was started by three graduate students at UC Berkeley (Benjamin Hindman, Andy Konwinski and Matei Zaharia), who were soon joined by a postdoc from the Swedish Institute of Computer Science (Ali Ghodsi). Although started as a research project, Mesos was always intended to solve operational issues with large clusters and to become an open-source project, building on our successful experience doing research that has been incorporated into Apache Hadoop (several scheduling algorithms).

Alignment

The ASF is a natural host for Mesos given that it is already the home of Hadoop, HBase, Cassandra, and other emerging cloud software projects. Mesos was designed to support Hadoop from the beginning in order to solve operational challenges in Hadoop clusters, and it aims to support a wide range of applications beyond Hadoop as well. Mesos complements the existing Apache cloud computing projects by providing a unified way to manage these systems and to share resources and data between them.

Known Risks

Orphaned Products

With the current core developers of Mesos being graduate students, there is a risk that these developers will eventually move on to other projects. However, because of the broad scope of Mesos, we all plan to continue working on projects related to it in the next several years. We are also actively working with developers at other organizations, such as Twitter, who are good candidates to become contributors.

Inexperience with Open Source

All of the core developers are active users and followers of open source. Matei Zaharia is a Hadoop committer and has experience with the Apache infrastructure and development process. Andy Konwinski has contributed patches to Hadoop through the Apache infrastructure as well. Ali Ghodsi has released open source software as part of his PhD work that was adopted by a Swedish company.

Homogeneous Developers

The current core developers are all researchers (graduate students and a young professor). However, we hope to establish a developer community that includes contributors from several corporations, and we are already working towards this with Twitter and Facebook.

Reliance on Salaried Developers

Given that the project started in an academic research environment, the core developers are all interested in it primarily for its own sake rather than for the sake of employment. We all intend to continue working on Mesos as volunteers.

Relationships with Other Apache Products

Mesos needs to work well with Hadoop, HBase, and other cloud software projects. Being hosted on the same infrastructure will facilitate this and ultimately help out both Mesos and the projects that can now be managed using it. There is, however, a risk that new projects will be built to run solely on Mesos, introducing a dependency.

An Excessive Fascination with the Apache Brand

While we respect the reputation of the Apache brand and have no doubts that it will attract contributors and users, our interest is primarily to give Mesos a solid home as an open source project following an established development model. Locating the project in Apache will also facilitate collaboration with Hadoop, HBase, and other Apache cluster computing projects, as discussed in the Alignment section.

Documentation

Information about Mesos can be found at http://mesos.berkeley.edu. The following sources may be useful to start with:

- Documentation for GitHub release: http://github.com/mesos/mesos/wiki
- Presentation at Hadoop User Group: http://www.cs.berkeley.edu/~matei/talks/2010/hug_mesos.pdf
- Tech report on system design and current features: http://mesos.berkeley.edu/mesos_tech_report.pdf (paper to appear at NSDI 2011 conference)

Initial Source

Mesos has been under development since spring 2009 by a team of graduate students and researchers. It is currently hosted on GitHub under a BSD license at http://github.com/mesos/mesos.

External Dependencies

The dependencies all have Apache compatible licenses, including BSD, MIT, Boost, and Apache 2.0.

Cryptography

Not applicable.

Required Resources

Mailing Lists

- mesos-private for private PMC discussions (with moderated subscriptions)
- mesos-dev
- mesos-commits
- mesos-user

Subversion Directory

https://svn.apache.org/repos/asf/incubator/mesos

Issue Tracking

JIRA Mesos (MESOS)

Other Resources

The existing code already has unit tests, so we would like a Hudson instance to run them whenever a new patch is submitted. This can be added after project creation.

Initial Committers

- Ali Ghodsi (ali at sics dot se)
- Benjamin Hindman (benh at eecs dot berkeley dot edu)
- Andy Konwinski (andyk at eecs dot berkeley dot edu)
- Matei Zaharia (matei at apache dot org)

A CLA is already on file for Matei Zaharia.

Affiliations

- Ali Ghodsi (UC Berkeley / Swedish Institute of Computer Science)
- Benjamin Hindman (UC Berkeley)
- Andy Konwinski (UC Berkeley)
- Matei Zaharia (UC Berkeley)

Sponsors

Champion

Tom White

Nominated Mentors

- Ian Holsman

- Brian McCallister
- Tom White

Sponsoring Entity

Incubator PMC