TashiProposal

Tashi Proposal

A proposal to the Apache Software Foundation Incubator PMC by

David O'Hallaron^*, Michael Kozuch[*], Michael Ryan[*], Steven Schlosser[*], Jim Cipar, Greg Ganger, Garth Gibson, Julio Lopez, Michael Strouken, Wittawat Tantisiriroj[+], Doug Cutting[#], Jay Kistler[#], Thomas Kwan[#]^

^*^Intel Research Pittsburgh, ^+^Carnegie Mellon University, ^#^Yahoo!

July 10, 2008

1. Abstract

Tashi is a cluster management system for cloud computing on Big Data.

2. Proposal

The Tashi project aims to build a software infrastructure for cloud computing on massive internet-scale datasets (what we call *Big Data*). The idea is to build a cluster management system that enables the Big Data that are stored in a cluster/data center to be accessed, shared, manipulated, and computed on by remote users in a convenient, efficient, and safe manner. The system aims to provide the following basic capabilities:

(a) *On-demand provisioning of storage and compute resources.* Users request a number of compute nodes, which can be either virtual or physical machines, and a set of disk images to boot up on the nodes. In response they receive their own persistent logical cluster of compute and storage nodes, which they can then manage and use.

(b) *Extensible end-to-end system management.* Tashi will define open non-proprietary interfaces for management tasks such as observation, inference, planning, and actuation. This will keep the system vendor-neutral and allow different research and development groups to plug in different implementations of different management modules.

(c) *Cooperative storage and compute management.* The system will define new non-proprietary interfaces and methods that will allow compute and storage management to work together in concert.

(d) *Flexible storage models.* The system will support a range of different storage models, such as network-attached storage, per-node storage, and hybrids, to allow developers, researchers, and large scale cluster/data center operators to experiment with different kinds of file systems.

(e) *Flexible machine models.* The system will support different machine models. In particular, it will be VMM-agnostic, able to run different virtual machine monitors such as KVM and Xen. Also, in order to address the cluster squatting problem (when clusters are balkanized by users who reserve and hold nodes for their exclusive use) the system will support a novel bi-model booting capability, in which virtual machine and physical machine instances can boot from the same disk image.

3. Rationale and Approach

Digital media, pervasive sensing, web authoring, mobile computing, scientific and medical instruments, physical simulations, and virtual worlds are all delivering vast new datasets relating to every aspect of our lives. A growing fraction of this Big Data is going unused or being underexploited due to the overwhelming scale of the data involved. Effective sharing, understanding, and use of this new wealth of raw information poses one of the great challenges for the new century.

In order to compute on this emerging Big Data, many research and development groups are purchasing their own racks of compute and storage servers. The goal of the Tashi project is to develop a layer of utility software that turns these raw racks of servers into easily managed cloud computers that will allow remote users to share and explore their Big Data.

To our knowledge there are no open source projects addressing cluster management for Big Data applications. We need a project such as Tashi for a number of reasons: (1) No cloud computing cluster management systems have tackled the problem of having both compute and storage management working together in concert, which we believe will be necessary to support Big Data. (2) We need non-proprietary interfaces for cloud computing, and open source is the way to develop these. For example, Google's new App Engine and Amazon's web services require people to build to proprietary API's, so that their applications are no longer vendor neutral, but are tied to a particular service provider. (3) We need an extensible system that can serve as a platform to stimulate research in cluster management for cloud computing.

The Tashi system is targeted at two (not always distinct) communities:

(1) As a production system for organizations who want to offer medium to large scale clusters to their users. For example, many companies and university departments are purchasing such clusters, and a system like Tashi would help them provide their users with access to the cycles and storage in the clusters.

(2) As an extensible research platform for distributed systems researchers.

The approach for the project is to build on existing cluster management work pioneered by projects such as Usher (UCSD), Cluster on Demand (Duke), and EC2/S3 (Amazon), and then develop the new capabilities that will be required to support Big Data cloud computing.

4. Need for a Community Effort

A number of events at Yahoo, Carnegie Mellon, and Intel Research Pittsburgh motivated the development of Tashi and convinced us to work together in the context of an open-source community:

(a) In 2006 the Parallel Data Lab (PDL) at Carnegie Mellon built a cluster of 400 nodes from industry donations, with a goal of creating a "Data Center Observatory" that would allow systems researchers to study and monitor applications running on the cluster. This dream has been slow to materialize because of the cost and complexity of supporting and managing multiple applications and systems groups.

(b) In Fall 2007, Yahoo began offering access to their M45 research cluster to researchers at Carnegie Mellon, and in order to support M45 as well as their own internal production clusters, began to develop some cloud computing infrastructure on their own.

(c) In Fall 2007, Intel Research Pittsburgh purchased a moderate-sized 100-node cluster and made it available to applications groups at Carnegie Mellon working on various Big Data applications such as computational photography, machine translation, automatic speech recognition, and event detection in spatio-temporal video streams. Provisioning and scheduling the cluster in the face of so many different application demands has proven to be difficult.

The difficulties of managing and provisioning these different clusters convinced us that the problem was too big for any one of us to solve completely on our own, and that we needed to band together create a open-source community effort focused on developing a single software system.

Another important reason to develop an open-source community around Tashi is that we need non-proprietary vendor-neutral APIs for the emerging area of cloud computing, and open source is the best way to achieve that.

5. Known Risks

Commitment to future development. The risk of the developers abandoning the project is small, mainly because they all own and manage moderate to large scale clusters, and desperately need something like Tashi to provision and manage those clusters. We also need a system like Tashi to serve as an extensible platform for our research.

Experience with open source. Yahoo has had a significant and positive experience with the Apache Software Foundation (ASF) and Hadoop. While Intel and Carnegie Mellon have developed some non-ASF style open source projects in the past (e.g., Internet Suspend/Resume, OpenDHT, and OpenDiamond), they have no experience with ASF-style open source communities. However, they hope to benefit from Yahoo's considerable experience in this area.

Diversity of developer community. The initial code base for Tashi was developed by a single research programmer, Michael Ryan, at Intel Research Pittsburgh. An important reason for putting Tashi in the incubator is to expand the set of developers to include programmers from Carnegie Mellon and Yahoo, initially, and later, hopefully, from other groups such as Usher at UCSD, Eucalyptus from UCSB, Cluster-on-Demand from Duke University, and the RAD Lab at University of California, Berkeley.

Relationship to other Apache projects. There are no Apache projects such as Tashi that focus on systems support for cloud computing. However, the Tashi project is closely related to Hadoop/HDFS. The VM-based provisioning of Tashi will subsume the now deprecated sub-clustering functionality of Hadoop-on-demand. The Tashi prototype uses HDFS to host the cluster boot images. Also, we expect that many Tashi logical clusters will run Hadoop jobs.

Reasons that Tashi is an ASF project. There are three main reasons for developing Tashi through Apache rather than, say, Source{{`Forge. (1) Our Yahoo partner has had a very positive experience with the Hadoop project. (2) We recognize the need to build a strong developer community, and Apache is centered around building such communities. (3) The ASF also offers substantial legal oversight that makes it attractive for cross-organizational collaborative efforts such as Tashi. With Source}}Forge, for example, you have few guarantee about the title of the code. Thus, people can easily post code they don't own, and/or change the license terms of other open source code that they include in their projects. So users of code from Source`Forge must be wary. On the other hand, Apache vets all contributions, keeping signed documents from every committer on file, etc.

6. Related Work

A small sampling of some closely related work:

[1] M. McNett, D. Gupta, A. Bahdat, G. Voelker, "Usher: An Extensible Framework for Managing Clusters of Virtual Machines", Proceedings of the 21st Large Installation System Administration Conference (LISA 07), 2007.

[2] D. Irwin, J. Chase, L. Grit, A. Yumerefendi, D. Becker, "Sharing Networked Resources with Brokered Leases", Usenix, 2006.

[3] J. Chase, D. Irwin, L. Grit, J. Moore, S. Sprenkle, "Dynamic Virtual Clusters in a Grid Site Manager", HPDC, 2003.

[4] S. Garfinkel, "An Evaluation of Amazon's Grid Computing Services: EC2, S3, and SQS", Tech Report TR-08-07, School for Engineering and Applied Sciences, Harvard University, 2007.

[5] RedHat oVirt System, http://ovirt.org, 2008

[6] Eucalyptus, Rich Wolski, http://eucalyptus.cs.ucsb.edu

7. Source

We have working code, a pre-alpha proof-of-concept prototype that was developed by Michael Ryan at Intel Research Pittsburgh. The prototype is currently running on the 100-node cluster there. We will enter the incubator with clean code, developed entirely by Michael Ryan, that is unencumbered by any licensing issues.

8. Required Resources

(a) Proposed Mailing lists:

- tashi-private (with moderated subscriptions)
- tashi-dev
- tashi-commits
- tashi-user

(b) Subversion directory

- http://svn.apache.org/repos/asf/incubator/tashi

(c) Issue tracking:

- Tashi will use JIRA for bug tracking.

9. Initial Committers

Initially, there will be one committer each from Carnegie Mellon and Intel Research:

- Michael Stroucken (mxs@cmu.edu)
- Michael Ryan (michael.p.ryan@intel.com)

10. Sponsors

- *Champion:* Doug Cutting (cutting@apache.org)
- *Nominated mentors:*
  - Matthieu Riou (matthieu@offthelip.org),
  - Craig L Russell (clr@apache.org),
  - Paul Freemantle (pzfreo@gmail.com)
- *Sponsoring entity:* Apache Incubator PMC