

OmidProposal

Abstract

Omid is a flexible, reliable, high performance and scalable ACID transactional framework that allows client applications to execute transactions on top of MVCC key/value-based NoSQL datastores (currently Apache HBase) providing Snapshot Isolation guarantees on the accessed data.

Proposal

Omid is a flexible open-source transactional framework that provides ACID transactions with Snapshot Isolation guarantees on top of NoSQL datastores. In particular, the current codebase brings the concept of transactions to the popular Apache HBase datastore. Omid offers great performance, it is highly available, and scalable. Omid's current version is able to scale to thousands of clients triggering concurrent transactions on application data stored in HBase. Omid can scale beyond 100K transactions per second on mid-range hardware while incurring in a minimal impact on the speed of data access in the datastore. We're currently experimenting with a prototype version that can improve the performance up to ~380K TPS.

Omid has been publicly available as an open-source project in Github under Apache License Version 2.0 since 2011 [1]. During these years, it has generated certain interest in the open source community, especially since the public presentation of the first version in Hadoop Summit 2013 [2]. Currently the Github project has 241 Stars and 93 forks. Yahoo Inc. submits this proposal to the Apache Software Foundation with the aim to transfer the Omid project including its source code and documentation to Apache in order to start the build of a stable open source community around it.

1. <https://github.com/yahoo/omid>
2. [Omid presentation at Hadoop Summit 2013](#)

Background

An Omid prototype was first released as an open-source project back in 2011. Inspired by Google Percolator [1], it offered a lock-free approach to transactions in NoSQL datastores (See [2]). However, during these years, the design of Omid has evolved significantly. Whilst the current open-sourced version maintains many aspects of the original implementation, it is the result of a major redesign of the first prototype released in 2011.

Omid has now a more decentralized design that does not sacrifice the consistency and performance of the original version. The current design also enables Omid to scale to thousands of clients executing transactions concurrently on application data stored in HBase. Internally, Omid still utilizes a lock-free approach to support multiple concurrent clients. Its design also relies on a centralized conflict detection component, the TSO, which now resolves in an efficient manner writeset collisions among concurrent transactions without having to piggyback commit information to the clients. Another important benefit of Omid is that it doesn't require any modification of the underlying key-value datastore, HBase in this case. Moreover, the recently added high availability algorithm allows to eliminate the single point of failure represented by the TSO in those system deployments requiring a higher degree of dependability. Last but not least, the provided user API is very simple, mimicking transaction managers in the relational world: begin, commit, rollback.

Omid is used internally at Yahoo. Sieve, Yahoo's web-scale content management platform powering some of next-generation search and personalization products is using Omid as a transaction manager in its processing pipeline. Sieve essentially acts as a huge processing hub between content feeds and serving systems. It provides an environment for highly customizable, real-time, streamed information processing, with typical discovery-to-service latencies of just a few seconds. In terms of scale and availability, Omid's new design was largely driven by Sieve's requirements.

At Yahoo, we are also making an effort to disseminate the current status of the project through blog entries (See [3], [4] and [5]) and submissions to technical and academic conferences such as ATC 2016, Hadoop Summit 2016, HBaseConf 2016. Last but not least, Omid also appeared in a [TechCrunch] article in the last quarter of 2015 (See [6])

1. D. Peng and F. Dabek, Large-scale Incremental Processing Using Distributed Transactions and Notifications. USENIX Symposium on Operating Systems Design and Implementation, 2010
2. D. Gomez-Ferro, F. Junqueira, I. Kelly, B. Reed, and M. Yabandeh. Omid: Lock-free transactional support for distributed data stores. In Proc. of ICDE, 2013
3. <http://yahoohadoop.tumblr.com/post/129089878751/introducing-omid-transaction-processing-for>
4. <http://yahoohadoop.tumblr.com/post/132695603476/omid-architecture-and-protocol>
5. <http://yahoohadoop.tumblr.com/post/138682361161/high-availability-in-omid>
6. <http://techcrunch.com/2015/10/01/yahoos-open-source-omid-project-brings-scalable-transaction-processing-to-hbase/>

Rationale

Programming with ACID (Atomicity, Consistency, Isolation, Durability) transactions is very popular and it is featured in relational databases. However, in the Big Data ecosystem, applications typically use NoSQL datastores, which do not provide ACID transactions. Such NoSQL datastores used to give up transactional support for greater agility and scalability. However, while early NoSQL data store implementations did not include transaction support, the need for transactions soon emerged in Big Data applications when accessing shared data; for example, transactions are very important for modern, scalable systems that process content incrementally.

NoSQL datastores including HBase don't provide transactional frameworks to coordinate the access to the underlying data for preserving consistency. By using Omid, Big Data applications that need to bundle multiple read and write operations on HBase into logically indivisible units of work can execute transactions with ACID properties, just as they would use transactions in the relational database world. Omid extends the HBase key-value access API with transaction semantics. It can be exercised either directly, or via higher level data management API's. For example, Apache Phoenix (SQL-on-top-of-HBase) might use Omid as its transaction management component.

The following features make Omid an attractive choice for system designers and other projects in the Apache community:

- **Semantics.** Omid implements Snapshot Isolation (SI,) supported by major SQL and NoSQL technologies (e.g. Google Percolator).
- **Performance and Scalability.** Omid provides a highly scalable, lock-free implementation of SI. To the best of our knowledge, it is also one of the few open source NoSQL transactional platforms that can execute more than 100K transactions per second [1]. A new prototype still in development can go even further, up to ~380K TPS.
- **Reliability.** Omid has a high-availability (HA) mode, in which the core service performing writeset conflict resolution operates as primary-backup process pair with automatic failover. The HA support has zero overhead on the mainstream operation.
- **Adaptability.** Omid current version provides transactions on data stored in Apache HBase. However, Omid's components are generic enough to be adapted to any other key-value NoSQL datasource that supports MVCC.
- **Development.** Omid provides a very simple interface that mimics standard HBase APIs, making it developer friendly. Only minimal extensions to the standard interfaces have been introduced to enable transactions.
- **Simplicity.** Omid leverages the HBase infrastructure for managing its own metadata. It entails no additional services apart from those provided and used by HBase.
- **Track Record.** As we have mentioned, Omid is already in use by very-large-scale production systems at Yahoo. Also, Hortonworks is integrating Omid in a metastore implementation for Hive based on HBase

[1] See also [Haeinsa](#)

Current Status

Current Omid implementation is available in both, Yahoo's internal Github repository for internal use at Yahoo as well as in [Yahoo's Github public repository](#). Both repositories are managed by Omid's current developers at Yahoo.

As it is mentioned above, Yahoo is currently using Omid for providing transactions in Sieve, a web-scale content management platform that powers Yahoo's next-generation search and personalization products.

Meritocracy

The first version of Omid was originally created in 2011 by Maysam Yabandeh, Daniel Gomez-Ferro, Ivan B. Kelly, Benjamin Reed and Flavio Junqueira at the R&D Scalable Computing Group of Yahoo Labs in Spain.

During the years after its inception, Omid has matured to operate at Web scale and has been used internally by strategic projects at Yahoo such as Sieve. The current base of committers belong to the Yahoo team that took over the initial Omid prototype and rewrote it to meet the high availability and scalability requirements of the Sieve project. This base of committers has recently incorporated Hortonworks members that helped in the Omid adaptation to HBase 1.x versions.

With this initial committer base, we aim to form a larger community that can collaborate with new ideas over the current code base. This new community will run the project following the "[Apache Way](#)". Users and new contributors will be treated with respect and welcomed. To grow the community, we will encourage contributors to provide patches, review code, propose new features improvements, talk at conferences such as Hadoop Summit, HBaseCon, [ApacheCon](#), etc. Committership and PMC membership will be offered according to meritocracy.

Community

The public Yahoo Omid repository at Github currently has 241 Stars and 93 forks, which means that there is an important interest for the project in the open-source community, at least compared with other similar projects (See <https://github.com/yahoo/omid.git>).

Recently, Hortonworks contributors to the Apache Hive project which are working on storing Hive metadata in HBase (Apache Jira HIVE-9452) manifested interest in using Omid. We started with them a fruitful collaboration that resulted in Omid supporting HBase 1.x versions.

Salesforce is also interested in collaborating in doing a Proof of Concept for integrating Omid as a pluggable transaction manager in Apache Phoenix.

Yahoo, Hortonworks and Salesforce participants will constitute the initial set of committers and mentors for the proposal.

Core Developers

The core developers of Omid are all skilled software developers and research engineers at Yahoo Inc. and Hortonworks with years of experiences in their fields. At this moment, developers are distributed across U.S. and Israel. The aim is to incorporate more committers from different organizations and locations over time.

The current set of developers include experienced committers from Apache HBase, Hive and Hadoop projects that have been working with us in the current codebase found in Github.

Finally, some of the core developers are currently NOT affiliated with the ASF and would require new ICLAs to be filed.

Alignment

Omid enhances with transactions the already successful Apache HBase datastore project. We have collaborated with other developers inside and outside Yahoo which are involved in the Apache HBase community, so we have had reliable feedback from them.

Although Omid brings value into HBase, the design of the current version provides a general transaction scheme that can potentially be adapted to other MVCC key-value datastores such as Apache Cassandra.

Apache Phoenix is also a potential target. Phoenix is a SQL layer on top of HBase that can potentially integrate Omid in order to provide the well-known concept of transactions to Phoenix-based applications.

Known Risks

Orphaned products

Yahoo's Research and Search organizations have been taking care of Omid development since the first prototype creation in 2011. Yahoo has a long history participating in open-source projects, and has been also a long time contributor to the Apache community. For example, in Apache, Yahoo is an important contributor in many projects in the Hadoop ecosystem such as HBase, Pig, Storm or YARN, and has also open-sourced other well-known projects outside Hadoop, such as Zookeeper or Bookkeeper. So it is in the best interest of Yahoo to make Omid also a successful open-source Apache product. If this happens, we are sure that a larger community will be formed around the project in a relatively short period of time, contributing to the diversification and stabilization of the base of committers.

Inexperience with Open Source

This project has long standing experienced mentors and interested contributors from Apache HBase, Hive and Phoenix to help us moving through the open source process. We are actively working with experienced Apache community members to improve our project and further testing.

Homogeneous Developers

Omid has been supported by Yahoo since its inception in 2011. However, all current committers are employed by their respective companies shown in the Affiliations section.

Reliance on Salaried Developers

All the current developers are paid by their employers to contribute to this project. Yahoo developers will also continue maintaining the internal Omid repository at their company. Of course, other developers are welcomed to contribute to this project after it is open sourced in Apache.

Relationships with Other Apache Product

Current Omid incarnation serves transactional contexts to applications storing their data in HBase. However Omid design potentially allows to be adapted to serve transactions on top of other MVCC-based key-value datastores in Apache community such as Cassandra.

As a transactional framework, many other Apache projects such as Apache Spark, Apache Phoenix, Apache Storm, Apache Flink could potentially benefit from Omid to get transactional contexts. In particular, Apache Phoenix a SQL layer on top of HBase might use Omid as its transaction management component. Once we open source Omid as an Apache project, we expect to generate more interest in the surrounded communities.

Very recently, a new incubator proposal for a similar project called Tephra, has been submitted to the ASF. We think this is good for the Apache community, and we believe that there's room for both proposals as the design of each of them is based on different principles (e.g. Omid does not require to maintain the state of ongoing transactions on the server-side component) and due to the fact that both ~~Tephra and Omid~~ have also gained certain traction in the open-source community.

With regard to the Apache projects that Omid uses, apart from HBase, Omid relies on Apache Zookeeper and Curator projects in order to coordinate the (re)connection of transaction managers (acting as clients) to the conflict resolution component for transactions (server side.) They're also used in order to coordinate the master and backup replicas in high availability scenarios.

An Excessive Fascination with the Apache Brand

We are applying to the Incubator process because we think that it is the logical next step for the Omid project after we open-sourced the code in Github some years ago. Yahoo has a long-standing history of contributing to Apache projects. The developers and contributors understand the implications of making it an Apache project, and strongly believe that the growing community can benefit from the Apache environment, ecosystem, and infrastructure.

Documentation

Current documentation about the project is available in the wiki of Omid's [Github repository](#). It will be moved under <https://omid.incubator.apache.org/docs> if the project is accepted as an Apache Incubator.

Initial Source

Initial source code is currently hosted in Github for general viewing and contribution: <https://github.com/yahoo/omid.git>

Omid source code is written in Java code (99%) mixed with some shell script (1%) in order to configure and trigger the execution of main components.

The code will be moved to [Apache](#) if accepted as an Incubator project.

Source and Intellectual Property Submission Plan

The current Omid License for the code published in Github is Apache 2.0. If Omid fulfills and passes the conditions for being an Incubator project in the ASF, the source code will be transitioned via the Software Grant Agreement onto the ASF infrastructure and in turn made available under the Apache License, version 2.0.

External Dependencies

The required external dependencies that are not Apache projects are all Apache licenses or other compatible Licenses:

- Maven & Maven plugins (<http://maven.apache.org/>) [Apache 2.0]
- JDK7 or OpenJDK 7 (<http://java.com/>) [Oracle or Openjdk JDK License]
- Google Guava v11.0.2 (<https://github.com/google/guava/>) [Apache 2.0]
- Google Guice v3.0 (<https://github.com/google/guice/wiki/>) [Apache 2.0]
- Testng v6.8.8 (<http://testng.org/>) [Apache 2.0]
- SLF4J (<http://www.slf4j.org/>) v1.7.7 [MIT License]
- Netty (<http://netty.io/>) v3.2.6.Final [Apache 2.0]
- Google Protocol Buffers v2.5.0 (<https://developers.google.com/protocol-buffers/>) [BSD License]
- Mockito (<http://mockito.org/>) v1.9.5 [MIT License]
- LMAX Disruptor v3.2.0 (<https://lmax-exchange.github.io/disruptor/>) [Apache 2.0]
- Coda Hale/Yammer.com Dropwizard Metrics v3.0.1 (<http://metrics.dropwizard.io/3.1.0/>) [Apache 2.0]
- C.Beust, JCommander v1.35 (<http://jcommander.org/>) [Apache 2.0]
- Hamcrest v1.3 (<http://hamcrest.org/JavaHamcrest/>) [BSD License]

Cryptography

Omid project does not use cryptography itself. However, Apache HBase ~~the database on top of which Omid works in its current version~~ uses standard APIs and tools for SSH and SSL communication where necessary.

Required Resources

We request that following resources be created for the project to use:

Mailing lists

- `omid-private` (moderated subscriptions)
- `omid-commits` (commit notification)
- `omid-dev` (technical discussions)

Git repository

<https://github.com/apache/incubator-omid>

Documentation

<https://omid.incubator.apache.org/docs/>

JIRA instance

<https://issues.apache.org/jira/browse/omid>

Initial Committers

- Daniel Dai, Hortonworks (daijy<AT>hortonworks<DOT>com)
- Alan Gates, Hortonworks, (gates<AT>hortonworks<DOT>com)
- Lars Hofhansl, Salesforce (larsh<AT>apache<DOT>org)
- Flavio P. Junqueira, Confluent (fpj<AT>apache<DOT>org)
- Igor Katkov (katkovi<AT>yahoo-inc<DOT>com)
- Francis C. Liu (fcliu<AT>yahoo-inc<DOT>com)
- Thejas Nair, Hortonworks (thejas<AT>hortonworks<DOT>com)
- Francisco Perez-Sorrosal (fperez<AT>yahoo-inc<DOT>com)
- Sameer Paranjpye (sparanjpye<AT>yahoo<DOT>com)
- Ohad Shacham (ohads<AT>yahoo-inc<DOT>com)
- James Taylor, Salesforce (jamestaylor<AT>apache<DOT>org)

Additional Interested Contributors

- Ivan Kelly (ivank<AT>apache<DOT>org)
- Maysam Yabandeh (myabandeh<AT>dropbox<DOT>com)

Affiliations

- Edward Bortnikov, Yahoo Inc.
- Daniel Dai, Hortonworks
- Flavio P. Junqueira, Confluent
- Igor Katkov, Yahoo Inc.
- Ivan Kelly, Midokura
- Francis C. Liu, Yahoo Inc.
- Sameer Paranjpye, Arimo
- Francisco Perez-Sorrosal, Yahoo Inc.
- Ohad Shacham, Yahoo Inc.
- Maysam Yabandeh, Dropbox Inc.

Sponsors

Champion

- Daniel Dai, Hortonworks (daijy<AT>hortonworks<DOT>com)

Nominated Mentors

- Alan Gates, Hortonworks, (gates<AT>hortonworks<DOT>com)
- Lars Hofhansl, Salesforce (larsh<AT>apache<DOT>org)
- Flavio P. Junqueira, Confluent (fpj<AT>apache<DOT>org)
- Thejas Nair, Hortonworks (thejas<AT>hortonworks<DOT>com)
- James Taylor, Salesforce (jamestaylor<AT>apache<DOT>org)

Sponsoring Entity

Apache Incubator PMC