TephraProposal

Abstract

Tephra is a system for providing globally consistent transactions on top of Apache HBase and other storage engines.

Proposal

Tephra is a transaction engine for distributed data stores like Apache HBase. It provides ACID semantics for concurrent data operations that span over region boundaries in HBase using Optimistic Concurrency Control.

Background

HBase provides strong consistency with row- or region-level ACID operations. However, it sacrifices cross-region and cross-table consistency in favor of scalability. This trade-off requires application developers to handle the complexity of ensuring consistency when their modifications span region boundaries. By providing support for global transactions that span regions, tables, or multiple RPCs, Tephra simplifies application development on top of HBase, without a significant impact on performance or scalability for many workloads.

Tephra leverages HBase's native data versioning to provide multi-versioned concurrency control (MVCC) for transactional reads and writes. With MVCC capability, each transaction sees its own consistent "snapshot" of data, providing snapshot isolation of concurrent transactions. MVCC along with conflict detection and handling enables Optimistic Concurrency Control.

Tephra consists of three main components:

- Transaction Server – maintains global view of transaction state, assigns new transaction IDs and performs conflict detection;
- Transaction Client – coordinates start, commit, and rollback of transactions; and
- Transaction Processor Coprocessor – applies filtering to the data read (based on a given transaction's state) and cleans up any data from old (no longer visible) transactions.

Although Tephra only supports HBase now, it can be extended to support transactions on any store that has multi-versioning and rollback support. The transactions can span over multiple stores and storage paradigms.

Rationale

Tephra has simple abstractions which can be used by an application to add transaction support over HBase. By abstracting away transaction handling using Tephra, the application is freed of transaction logic, and the application developer can focus on the use case. Also, Tephra can be extended to support transactions on data sources other than HBase.

By making Tephra an Apache open source project, we believe that there will be wider adoption and more opportunities for Tephra to be integrated into other Apache projects.

Current Status

Tephra was built at Cask Data Inc. initially as part of open-source framework Cask Data Application Platform (CDAP) http://cdap.io/. It was later converted into an independent open source project with Apache 2.0 License https://github.com/caskdata/tephra.

Tephra is used in CDAP as the transaction engine. As part of CDAP, Tephra has been deployed at multiple companies.

Apache Phoenix is using Tephra as transaction engine in the next release.

Meritocracy

Our intent with this incubator proposal is to start building a diverse developer community around Tephra following the Apache meritocracy model. Since Tephra was initially developed in early 2013, we have had fast adoption and contributions within Cask Data. We are looking forward to new contributors. We wish to build a community based on Apache's meritocracy principles, working with those who contribute significantly to the project and welcoming them to be committers both during the incubation process and beyond.

Community

Core developers of Tephra are at Cask Data. Recently the developer community has expanded to include folks from Apache Phoenix. We hope to extend our contributor base significantly and we will invite all who are interested in working on distributed transaction engine.

Core Developers

A few engineers from Cask Data and outside have developed Tephra:
Andreas Neumann, Terence Yim, Gary Helmling, Andrew Purtell and Poorna Chandra.

Alignment

The ASF is the natural choice to host the Tephra project as its goal of encouraging community-driven open source projects fits with our vision for Tephra.

Additionally, many other projects with which we are familiar and expect Tephra to integrate with, such as Phoenix, Zookeeper, HDFS, log4j, and others mentioned in the External Dependencies section are Apache projects, and Tephra will benefit by close proximity to them.

Known Risks

Orphaned Products

There is very little risk of Tephra being orphaned, as it is a key part of Cask Data's products. The core Tephra developers plan to continue to work on Tephra, and Cask Data has funding in place to support their efforts going forward. Also with Phoenix using Tephra for transactions, Phoenix developers are keen on contributing to Tephra.

Inexperience with Open Source

Several of the core developers have experience with open source development. Andreas Neumann is an Apache committer for Oozie and Twill. Terence Yim is an Apache committer for Helix and Twill. Poorna Chandra is an Apache committer for Twill. Gary Helmling is a committer for Apache Twill and a committer and PMC member for Apache HBase. James Taylor is PMC chair for Apache Phoenix, PMC member of Apache Calcite, and an IPMC member.

Homogeneous Developers

The current core developers are all Cask Data employees. However, we intend to establish a developer community that includes independent and corporate contributors. We are encouraging new contributors via our mailing lists, public presentations, and personal contacts, and we will continue to do so.

Apache Phoenix developers have already contributed several patches to Tephra, and have expressed interest in becoming long term contributors.

Reliance on Salaried Developers

Currently, these developers are paid to work on Tephra. Once the project has built a community, we expect to attract committers, developers and community other than the current core developers. However, because Cask Data products use Tephra internally, the reliance on salaried developers is unlikely to change, at least in the near term.

Relationships with Other Apache Products

Tephra is deeply integrated with Apache projects. Tephra provides transactions over Apache HBase, and uses Apache Twill and Apache Zookeeper for coordination. A number of other Apache projects are Tephra dependencies, and are listed in the External Dependencies section.

In addition, Apache Phoenix is using Tephra as the transaction engine.

An Excessive Fascination with the Apache Brand

While we respect the reputation of the Apache brand and have no doubt that it will attract contributors and users, our interest is primarily to give Tephra a solid home as an open source project following an established development model. We have also given additional reasons in the Rationale and Alignment sections.

Documentation

The current documentation for Tephra is at https://github.com/caskdata/tephra.

Initial Source

Tephra codebase is currently hosted at https://github.com/caskdata/tephra.

Source and Intellectual Property Submission Plan

Tephra codebase is currently licensed under Apache 2.0 license. Cask Data owns the trademark for "Tephra". As part of the incubation process Cask Data will transfer the trademark to Apache Foundation.

External Dependencies

The dependencies all have Apache-compatible licenses:

- dropwizard metrics (Apache 2.0)
- fastutil (Apache 2.0)
- gson (Apache 2.0)
- guava-libraries (Apache 2.0)
- guice (Apache 2.0)
- hadoop (Apache 2.0)
- hbase (Apache 2.0)
- hdfs (Apache 2.0)
- junit (EPL v1.0)
- logback (EPL v1.0)
- slf4j (MIT)
- thrift (Apache 2.0)
- twill (Apache 2.0)
- zookeeper (Apache 2.0)

Cryptography

Tephra does not use cryptography itself, however it can run on secure Hadoop, which uses Kerberos.

Required Resources

Mailing Lists

- tephra-private for private PMC discussions (with moderated subscriptions)
- tephra-dev for technical discussions among contributors
- tephra-commits for notification about commits

Subversion Directory

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

Issue Tracking

JIRA Tephra (TEPHRA)

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

- Andreas Neumann <anew at apache dot org>
- Terence Yim <chtyim at apache dot org>
- Poorna Chandra <poorna at apache dot org>
- Gokul Gunasekaran <gokul at cask dot co>
- James Taylor <jamestaylor at apache dot org>
- Thomas D'Silva <tdsilva at apache dot org>
- Gary Helmling <garyh at apache dot org>

Affiliations

- Andreas Neumann (Cask Data)
- Terence Yim (Cask Data)
- Poorna Chandra (Cask Data)
- Gokul Gunasekaran (Cask Data)

- James Taylor (Salesforce.com)
- Thomas D'Silva (Salesforce.com)
- Gary Helmling (Facebook)

Sponsors

Champion

James Taylor <jamestaylor at apache dot org> (V.P., Apache Phoenix)

Nominated Mentors

- James Taylor <jamestaylor at apache dot org>
- Lars Hofhansl <larsh at apache dot org>
- Andrew Purtell <apurtell at apache dot org>
- Alan Gates <gates at apache dot org>
- Henry Saputra <hsaputra at apache dot org>

Sponsoring Entity

We are requesting that the Incubator sponsor this project.