ThriftProposal

Thrift Proposal

Abstract

Thrift is a framework for efficient cross-language data serialization, RPC, and server programming.

Proposal

Thrift is a software library and set of code-generation tools designed to expedite development and implementation of efficient and scalable backend services. Its primary goal is to enable efficient and reliable communication across programming languages by abstracting the portions of each language that tend to require the most customization into a common library that is implemented in each language. Specifically, Thrift allows developers to define datatypes and service interfaces in a single language-neutral file and generate all the necessary code to build RPC clients and servers.

Background

Thrift was initially developed at Facebook starting in 2006 to power RPC and data logging for a number of backend services for the site, such as Search and News Feed. The package was designed for open source, and was released in early 2007. Since then, a number of other developers have submitted patches to the project and become de facto owners of major parts. Support for many languages has been developed entirely outside of Facebook.

Rationale

The need for high performance, reliable communication across different programming languages seems to be growing more and more common in modern programming, particularly when writing software for the web. Historically, this problem has forced developers to standardize on one language/framework or adopt heavier-weight systems, such as CORBA or SOAP. These systems tend to make tradeoffs that aren't always ideal for the use case. SOAP, for example, may be ideal for calling across disparate web services, but is unnecessarily verbose for service calls on an intranet.

Most of these systems also require developers to learn the particulars of their type systems, especially when dealing with containers or objects. One of the primary goals of Thrift is to allow developers to program across languages while still using the standard idioms and style in each language. Custom type systems also makes code reuse more difficult. Thrift allows developers to avoid creating unnecessary wrapper interfaces by operating directly on native types.

Initial Goals

Though the bulk of Thrift's initial development is complete, there are still some large areas for future development. Some areas we hope to focus on in Apache:

- Better log storage/replay
- Meta-data serialization
- Higher-performance serialization, standard C extension model across Python/PHP/etc.
- Extending the abstraction to a multi-client that can fan-out across multiple servers

Current Status

Meritocracy

Though initial development was done at Facebook, Thrift was intended to be released as an open source project from its inception. Since release, many developers have adopted the framework and submitted significant patches. Large portions of the codebase are now managed by those most familiar with and responsible for them. Any potentially controversial change is discussed on the public mailing list (http://lists.pub.facebook.com/mailman/listinfo/thrift/) and good suggestions are frequently implemented.

Community

Thrift is currently in use across a number of organizations, and we expect this to grow as Thrift becomes a relevant and useful tool for building more open source projects.

Core Developers

Thrift currently has developers across many organizations (e.g. Facebook, Powerset, ReCaptcha, AmieStreet), many of whom are contributors to other open source projects.

Alignment

Thrift is designed to integrate cleanly with other projects. We think this is a particularly good fit for Apache due to integration potential with other projects, specifically Hadoop/Hbase.

Known Risks

Orphaned Products

Thrift is already deployed into production at multiple large websites that are frequently iterating on the featureset. There's no realistic chance of it becoming orphaned.

Inexperience with Open Source

The project has already been open source for nearly a year and has attracted many developers already. Part of the reason to join Apache is to make the project work even better as open source by removing some obstacles, such as Facebook hosting the SVN, and putting the resources all in a truly open space, being able to have more committers, etc. Most of the core developers have a history of working with open source tools.

Homogenous Developers

The current set of developers work across a variety of organizations. Naturally, most are websites with significant backend structure (and hence a need for Thrift), but the problems they are solving are diverse, and many don't work in the same programming languages.

Reliance on Salaried Developers

Thrift is a "means to an end." None of the developers (to my knowledge) working on Thrift are salaried specifically to work on Thrift. Rather, Thrift is useful in building other projects, which may or may not be for salary. Realistically, it is likely that a decent portion of work on Thrift will be done by someone at a company, but not specifically tasked with working on Thrift. So long as the tool is relevant and useful, this should result in developers contributing time both at work and personally.

Relationships with Other Apache Projects

Thrift has already been introduced into the Hbase project. (See https://issues.apache.org/jira/browse/HADOOP-2389) Since Thrift is a development tool, it is designed and well-suited for use in other projects. As a start, we definitely plan to continue integration work with Hbase.

An Excessive Fascination with the Apache Brand

Thrift has already attracted a stable base of developers. The reasons for joining Apache are not to advertise the project, but rather to demonstrate the commitment to open source by divorcing the trunk from any one corporation and pursuing further integration with other Apache projects.

Documentation

Existing page: http://developers.facebook.com/thrift/

Mailing list (with archives): http://lists.pub.facebook.com/mailman/listinfo/thrift/

Initial Source

Currently hosted by Facebook: http://svn.facebook.com/svnroot/thrift/

Source and Intellectual Property Submission Plan

All code currently hosted in the Facebook public SVN folder will be contributed.

External Dependencies

All dependencies (libevent, Boost) have Apache compatible licenses.

Cryptography

None

Required Resources

Mailing lists

- thrift-dev
- thrift-private

Subversion Directory

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

We'd also be interested in using git to store the repo. Does apache have infrastucture set up to support that? It'd make it easier for non-committer developers to work on patches, checkpoint commits, etc.

Issue Tracking

JIRA Thrift

Initial Committers

- Mark Slee (mcslee at facebook dot com)
- Aditya Agarwal (aditya at facebook dot com)
- Marc Kwiatkowski (marc at facebook dot com)
- David Reiss (david at facebook dot com)
- James Wang (jwang at facebook dot com)
- Chris Piro (cpiro at facebook dot com)
- Ben Maurer (bmaurer at andrew dot cmu dot edu)
- Kevin Clark (kevin at powerset dot com)
- Jake Luciani (jakers at gmail dot com)

Affiliations

- People with Facebook email addresses - Facebook
- Ben Maurer - ReCaptcha
- Kevin Clark - Powerset

Sponsors

Champion

- Doug Cutting

Mentors

- Paul Querna
- Upayavira
- Jason van Zyl

Sponsoring Entity

- Incubator