DorisProposal

Apache Doris

Abstract

Doris is a MPP-based interactive SQL data warehousing for reporting and analysis.

Proposal

We propose to contribute the Doris codebase and associated artifacts (e.g. documentation, web-site content etc.) to the Apache Software Foundation, and aim to build an open community around Doris's continued development in the 'Apache Way'.

Overview of Doris

Doris's implementation consists of two daemons: Frontend (FE) and Backend (BE).

Frontend daemon consists of query coordinator and catalog manager. Query coordinator is responsible for receiving users' sql queries, compiling queries and managing queries execution. Catalog manager is responsible for managing metadata such as databases, tables, partitions, replicas and etc. Several frontend daemons could be deployed to guarantee fault-tolerance, and load balancing.

Backend daemon stores the data and executes the query fragments. Many backend daemons could also be deployed to provide scalability and fault-tolerance.

A typical Doris cluster generally composes of several frontend daemons and dozens to hundreds of backend daemons.

Users can use MySQL client tools to connect any frontend daemon to submit SQL query. Frontend receives the query and compiles it into query plans executable by the Backend. Then Frontend sends the query plan fragments to Backend. Backend will build a query execution DAG. Data is fetched and pipelined into the DAG. The final result response is sent to client via Frontend. The distribution of query fragment execution takes minimizing data movement and maximizing scan locality as the main goal.

Background

At Baidu, Prior to Doris, different tools were deployed to solve diverse requirements in many ways. And when a use case requires the simultaneous availability of capabilities that cannot all be provided by a single tool, users were forced to build hybrid architectures that stitch multiple tools together, but we believe that they shouldn't need to accept such inherent complexity. A storage system built to provide great performance across a broad range of workloads provides a more elegant solution to the problems that hybrid architectures aim to solve. Doris is the solution.

Doris is designed to be a simple and single tightly coupled system, not depending on other systems. Doris provides high concurrent low latency point query performance, but also provides high throughput queries of ad-hoc analysis. Doris provides bulk-batch data loading, but also provides near real-time mini-batch data loading. Doris also provides high availability, reliability, fault tolerance, and scalability.

Rationale

Doris mainly integrates the technology of Google Mesa and Apache Impala.

Mesa is a highly scalable analytic data storage system that stores critical measurement data related to Google's Internet advertising business. Mesa is designed to satisfy complex and challenging set of users' and systems' requirements, including near real-time data ingestion and query ability, as well as high availability, reliability, fault tolerance, and scalability for large data and query volumes.

Impala is a modern, open-source MPP SQL engine architected from the ground up for the Hadoop data processing environment. At present, by virtue of its superior performance and rich functionality Impala has been comparable to many commercial MPP database query engine. Mesa can satisfy the needs of many of our storage requirements, however Mesa itself does not provide a SQL query engine; Impala is a very good MPP SQL query engine, but the lack of a perfect distributed storage engine. So in the end we chose the combination of these two technologies.

Learning from Mesa's data model, we developed a distributed storage engine. Unlike Mesa, this storage engine does not rely on any distributed file system. Then we deeply integrate this storage engine with Impala query engine. Query compiling, query execution coordination and catalog management of storage engine are integrated to be frontend daemon; query execution and data storage are integrated to be backend daemon. With this integration, we implemented a single, full-featured, high performance state the art of MPP database, as well as maintaining the simplicity.

Current Status

Doris has been an open source project on GitHub (https://github.com/baidu/palo).

Meritocracy

Doris has been deployed in production at Baidu and is applying more than 200 lines of business. It has demonstrated great performance benefits and has proved to be a better way for reporting and analysis based big data. Still We look forward to growing a rich user and developer community.

Community

Doris seeks to develop developer and user communities during incubation.

Doris makes use of Apache Impala. It was identified during early review of the proposal that the Doris community will need to work with Impala to define a suitable API.

Core Developers

- Ruyue Ma (https://github.com/maruyue, maruyue@baidu dot com)
- Chun Zhao (https://github.com/imay, buaa.zhaoc@gmail dot com)
- Mingyu Chen (https://github.com/morningman,chenmingyu@baidu dot com)
- De Lihttps://github.com/lide-reed, mailtolide@sina dot com
- Hao Chen (https://github.com/chenhao7253886, chenhao16@baidu dot com)
- Chaoyong Li (https://github.com/cyongli, lichaoyong@baidu dot com)
- Bin Lin (https://github.com/lingbin, lingbinlb@gmail dot com)

Alignment

Doris is related to several other Apache projects:

- Doris can also read data stored in Apache Hadoop clusters powered by the HDFS filesystem.
- Doris is closely integrated with Impala, which has graduated from Apache Incubator.
- Doris uses Apache Thrift as its RPC and serialization framework of choice.

Known Risks

Orphaned Products

The core developers of Doris team plan to work full time on this project. There is very little risk of Doris getting orphaned since at least one large company (Baidu) is extensively using it in their production. For example, currently there are more than 200 use cases using Doris in production. Furthermore, since Doris was open sourced at the beginning of October 2017, it has received more than 660 stars and been forked nearly 170 times. We plan to extend and diversify this community further through Apache.

Inexperience with Open Source

The core developers are all active users and followers of open source. They are already committers and contributors to the Doris Github project. All have been involved with the source code that has been released under an open source license, and several of them also have experience developing code in an open source environment. Though the core set of Developers do not have Apache Open Source experience, there are plans to onboard individuals with Apache open source experience on to the project.

Homogenous Developers

The most of core developers are from Baidu, but after Doris was open sourced, Doris received a lot of bug fixes and enhancements from other developers not working at Baidu.

Reliance on Salaried Developers

Baidu invested in Doris as the OLAP solution and some of its key engineers are working full time on the project. In addition, since there is a growing Big Data need for scalable OLAP solutions, we look forward to other Apache developers and researchers to contribute to the project. Also key to addressing the risk associated with relying on Salaried developers from a single entity is to increase the diversity of the contributors and actively lobby for Domain experts in the BI space to contribute. Apache Doris intends to do this.

An Excessive Fascination with the Apache Brand

Doris is proposing to enter incubation at Apache in order to help efforts to diversify the committer-base, not so much to capitalize on the Apache brand. The Doris project is in production use already inside Baidu, but is not expected to be an Baidu product for external customers. As such, the Doris project is not seeking to use the Apache brand as a marketing tool.

Documentation

Information about Doris can be found at https://github.com/baidu/palo. The following links provide more information about Doris in open source:

- Doris wiki site: https://github.com/baidu/palo/wiki
- Codebase at Github: https://github.com/baidu/palo
- Issue Tracking: https://github.com/baidu/palo/issues
- Overview: https://github.com/baidu/Doris/wiki/palo-Overview
- FAQ: https://github.com/baidu/palo/wiki/palo-FAQ

Initial Source

Doris has been under development since 2017 by a team of engineers at Baidu Inc. It is currently hosted on Github.com under an Apache license at https://github.com/baidu/palo.

External Dependencies

Doris has the following external dependencies.

- Google gflags (BSD)
- Google glog (BSD)
- Apache Thrift (Apache Software License v2.0)
- Apache Commons (Apache Software License v2.0)
- Boost (Boost Software License)
- rapidjson (Tencent)
- Google RE2 (BSD-style)
- lz4 (BSD)
- snappy (BSD)
- Twitter Bootstrap (Apache Software License v2.0)
- d3 (BSD)
- LLVM (BSD-like)

Build and test dependencies:

- Apache Ant (Apache Software License v2.0)
- Apache Maven (Apache Software License v2.0)
- cmake (BSD)
- clang (BSD)
- Google gtest (Apache Software License v2.0)

Required Resources

Mailing List

There are currently no mailing lists. The usual mailing lists are expected to be set up when entering incubation:

- private@doris.incubator.apache.org
- dev@doris.incubator.apache.org
- commits@doris.incubator.apache.org

Subversion Directory

Upon entering incubation, we want to move (or copy) the existing repo from https://github.com/baidu/palo to Apache infrastructure at https://github.com/apache/incubator-doris.

Issue Tracking

Doris currently uses GitHub to track issues. Would like to continue to do so while we discuss migration possibilities with the ASF Infra committee.

Other Resources

The existing code already has unit tests so we will make use of existing Apache continuous testing infrastructure. The resulting load should not be very large.

Initial Committers

- Ruyue Ma (https://github.com/maruyue, maruyue@baidu dot com)
- Chun Zhao (https://github.com/imay, buaa.zhaoc@gmail dot com)
- Mingyu Chen (https://github.com/morningman,chenmingyu@baidu dot com)
- De Lihttps://github.com/lide-reed, mailtolide@sina dot com
- Hao Chen (https://github.com/chenhao7253886, chenhao16@baidu dot com)
- Chaoyong Li (https://github.com/cyongli, lichaoyong@baidu dot com)
- Bin Lin (https://github.com/lingbin, lingbinlb@gmail dot com)
- Sijie Guo (guosijie@gmail dot com)
- Zheng Shao (zshao@apache.org)

Affiliations

Seven of the initial committers are employees of Baidu Inc..

Sponsors

Champion

- Dave Fisher, wave@apache.org

Nominated Mentors

- Luke Han, lukehan@apache.org
- Dave Fisher, wave@apache.org
- Willem Jiang, ningjiang@apache.org

Sponsoring Entity

We are requesting the Incubator to sponsor this project.