SqoopProposal

Sqoop - A Data Transfer Tool for Hadoop

Abstract

Sqoop is a tool designed for efficiently transferring bulk data between Apache Hadoop and structured datastores such as relational databases. You can use Sqoop to import data from external structured datastores into Hadoop Distributed File System or related systems like Hive and HBase. Conversely, Sqoop can be used to extract data from Hadoop and export it to external structured datastores such as relational databases and enterprise data warehouses.

Proposal

Hadoop and related systems operate on large volumes of data. Typically this data originates from outside of Hadoop infrastructure and must be provisioned for consumption by Hadoop and related systems for analysis and processing. Sqoop allows fast provisioning of data into Hadoop and related systems by providing a bulk import and export mechanism that enables consumers to effectively use Hadoop for data analysis and processing.

Background

Sqoop was initially developed by Cloudera to enable the import and export of data between various databases and Hadoop Distributed File System (HDFS). It was provided as a patch to Hadoop project via the issue HADOOP-5815 and was maintained as a contrib module to Hadoop between May 2009 to April 2010. In April 2010, Sqoop was removed from Hadoop contrib via MAPREDUCE-1644 and was made available by Cloudera on GitHub.

Since then Sqoop has been maintained by Cloudera as an open source project on GitHub. All code available in Sqoop is open source and made publicaly available under the Apache 2 license. During this time Sqoop has been formally released three times as versions 1.0, 1.1 and 1.2.

Rationale

Hadoop is often used to process data that originated or is later served by structured data stores such as relational databases, spreadsheets or enterprise data warehouses. Unfortunately, current methods of transferring data are inefficient and ad hoc, often consisting of manual steps specific to the external system. These steps are necessary to help provision this data for consumption by Map-Reduce jobs, or by systems that build on top of Hadoop such as Hive and Pig. The transfer of this data can take substantial amount of time depending upon its size. An optimal transfer approach that works well with one particular datastore will typically not work as optimally with another datastore due to inherent architectural differences between different datastore implementations. Sqoop addresses this problem by providing connectivity of Hadoop with external systems via pluggable connectors. Specialized connectors are developed for optimal performance for data transfer between Hadoop and target systems.

Analyzed and processed data from Hadoop and related systems may also require to be provisioned outside of Hadoop for consumption by business applications. Sqoop allows the export of data from Hadoop to external systems to facilitate its use in other systems. This too, like the import scenario, is implemented via specialized connectors that are built for the purposes of optimal integration between Hadoop and external systems.

Connectors can be built for systems that Sqoop does not yet integrate with and thus can be easily incorporated into Sqoop. Connectors allow Sqoop to interface with external systems of different types, ensuring that newer systems can integrate with Hadoop with relative ease and in a consistent manner.

Besides allowing integration with other external systems, Sqoop provides tight integration with systems that build on to of Hadoop such as Hive, HBase etc - thus providing data integration between Hadoop based systems and external systems in a single step manner.

Initial Goals

Sqoop is currently in its first major release with a considerable number of enhancement requests, tasks, and issues logged towards its future development. The initial goal of this project will be to address the highly requested features and bug-fixes towards its next dot release. The key features of interest are the following:

- Support for bulk import into Apache HBase.
- Allow user to supply password in permission protected file.
- Support for pluggable query to help Sqoop identify the metadata associated with the source or target table definitions.
- Allow user to specify custom split semantics for efficient parallelization of import jobs.

Current Status

Meritocracy

Sqoop has been an open source project since its start. It was initially developed by Aaron Kimball in May 2009 along with development team at Cloudera and supplied as a patch to Hadoop project. Later it was moved to GitHub as a Cloudera open-source project where Cloudera engineering team has since maintained it with Arvind Prabhakar and Ahmed Radwan dedicated towards its improvement. Developers external to Cloudera provided feedback, suggested features and fixes and implemented extensions of Sqoop since its inception. Contributors to Sqoop include developers from different companies and different parts of the world.

Community

Sqoop is currently used by a number of organizations all over the world. Sqoop has an active and growing user community with active participation in user and developer mailing lists.

Core Developers

The core developers for Sqoop project are:

- Aaron Kimball: Aaron designed and implemented much of the original code.
- Arvind Prabhakar: Has been working on Sqoop features and bug fixes.
- Ahmed Radwan: Has been working on Sqoop features and bug fixes.
- Jonathan Hsieh: Has started working on Sqoop features and bug fixes.
- Other contributors to the project include: Angus He, Brian Muller, Eli Collins, Guy Le Mar, James Grant, Konstantin Boudnik, Lars Francke, Michael Hausler, Michael Katzenellenbogen, Pter Happ and Scott Foster.

All committers to Sqoop project have contributed towards Hadoop or related Apache projects and are very familiar with Apache principals and philosophy for community driven software development.

Alignment

Sqoop complements Hadoop Map-Reduce, Pig, Hive, HBase by providing a robust mechanism to allow data integration from external systems for effective data analysis. It integrates with Hive and HBase currently and work is being done to integrate it with Pig.

Known Risks

Orphaned Products

Sqoop is already deployed in production at multiple companies and they are actively participating in feature requests and user led discussions. Sqoop is getting traction with developers and thus the risks of it being orphaned are minimal.

Inexperience with Open Source

All code developed for Sqoop has been open source from the start. The initial part of Sqoop development was done within Hadoop project as a contrib module. Since then it has been maintained as an Apache 2.0 licensed open-source project on GitHub by Cloudera.

All committers of Sqoop project are intimately familiar with the Apache model for open-source development and are experienced with working with new contributors. Aaron Kimball, the creator of the project and one of the committers is also a committer on Apache MapReduce.

Homogeneous Developers

The initial set of committers is from a small set of organizations. However, we expect that once approved for incubation, the project will attract new contributors from diverse organizations and will thus grow organically. The participation of developers from several different organizations in the mailing list is a strong indication for this assertion.

Reliance on Salaried Developers

It is expected that Sqoop will be developed on salaried and volunteer time, although all of the initial developers will work on it mainly on salaried time.

Relationships with Other Apache Products

Sqoop depends upon other Apache Projects: Hadoop, Hive, HBase Log4J and multiple Apache commons components and build systems like Ant and Maven.

An Excessive Fascination with the Apache Brand

The reasons for joining Apache are to increase the synergy with other Apache Hadoop related projects and to foster a healthy community of contributors and consumers around the project. This is facilitated by ASF and that is the primary reason we would like Sqoop to become an Apache project.

Documentation

- All Sqoop documentation is maintained within Sqoop sources and can be built directly.
- Sqoop docs: http://archive.cloudera.com/cdh/3/sqoop/
- Sqoop wiki at GitHub: https://github.com/cloudera/sqoop/wiki
- Sqoop jira at Cloudera: https://issues.cloudera.org/browse/sqoop

Initial Source

- https://github.com/cloudera/sqoop/tree/

Source and Intellectual Property Submission Plan

- The initial source is already Apache 2.0 licensed.

External Dependencies

The required external dependencies are all Apache License or compatible licenses. Following components with non-Apache licenses are enumerated:

- HSQLDB: HSQLDB License - a BSD-based license.

Non-Apache build tools that are used by Sqoop are as follows:

- AsciiDoc: GNU GPLv2
- Checkstyle: GNU LGPLv3
- FindBugs: GNU LGPL
- Cobertura: GNU GPLv2

Cryptography

Sqoop does not depend upon any cryptography tools or libraries.

Required Resources

Mailing lists

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

Subversion Directory

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

Issue Tracing

JIRA Sqoop (SQOOP)

Other Resources

The existing code already has unit and integration 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

- Arvind Prabhakar (arvind at cloudera dot com)
- Ahmed Radwan (a dot aboelela at gmail dot com)
- Jonathan Hsieh (jon at cloudera dot com)
- Aaron Kimball (kimballa at apache dot org)
- Greg Cottman (greg dot cottman at quest dot com)
- Guy le Mar (guy dot lemar at quest dot com)
- Roman Shaposhnik (rvs at cloudera dot com)
- Andrew Bayer (andrew at cloudera dot com)
- Paul Zimdars (pzimdars at jpl dot nasa dot gov)

A CLA is already on file for Aaron Kimball.

Affiliations

- Arvind Prabhakar, Cloudera
- Ahmed Radwan, Cloudera
- Jonathan Hsieh, Cloudera
- Aaron Kimball, Odiago
- Greg Cottman, Quest
- Guy le Mar, Quest
- Roman Shaposhnik, Cloudera
- Andrew Bayer, Cloudera
- Paul Zimdars, JPL

Sponsors

Champion

- Tom White (tomwhite at apache dot org)

Nominated Mentors

- Patrick Hunt (phunt at apache dot org)
- Mohammad Nour El-Din (mnour at apache dot org)
- Tom White (tomwhite at apache dot org)
- Olivier Lamy (olamy at apache dot org)

Sponsoring Entity

- Apache Incubator PMC