CrunchProposal

Crunch - Easy, Efficient MapReduce Pipelines in Java and Scala

Abstract

Crunch is a Java library for writing, testing, and running pipelines of MapReduce jobs on Apache Hadoop.

Proposal

Crunch is a Java library for writing, testing, and running pipelines of MapReduce jobs on Apache Hadoop. Its main goal is to provide a high-level API for writing and testing complex MapReduce jobs that require multiple processing stages. It has a simple, flexible, and extensible data model that makes it ideal for processing data that does not naturally fit into a relational structure, such as time series and serialized object formats like JSON and Avro. It supports running pipelines either as a series of MapReduce jobs on an Apache Hadoop cluster or in memory on a single machine for fast testing and debugging.

Background

Crunch was initially developed by Cloudera to simplify the process of creating sequences of dependent MapReduce jobs, especially jobs that processed non-relational data like time series. Its design was based on a paper Google published about a Java library they developed called FlumeJava that was created in order to solve a similar class of problems. Crunch was open-sourced by Cloudera on GitHub as an Apache 2.0 licensed project in October 2011. During this time Crunch has been formally released twice, as versions 0.1.0 (October 2011) and 0.2.0 (February 2012), with an incremental update to version 0.2.1 (March 2012) . These releases are also distributed by Cloudera as source and binaries from Cloudera's Maven repository.

Rationale

Most of the interesting analytical and data processing tasks that are run on an Apache Hadoop cluster require a series of MapReduce jobs to be executed in sequence. Developers who are creating these pipelines today need to manually assign the sequence of tasks to perform in a dependent chain of MapReduce jobs, even though there are a number of well-known patterns for fusing dependent computations together into a single MapReduce stage and for performing common types of joins and aggregations. This results in MapReduce pipelines that are more difficult to test, maintain, and extend to support new functionality.

Furthermore, the type of data that is being stored and processed using Apache Hadoop is evolving. Although Hadoop was originally used for storing large volumes of structured text in the form of webpages and log files, it is now common for Hadoop to store complex, structured data formats such as JSON, Apache Avro, and Apache Thrift. These formats allow developers to work with serialized objects in programming languages like Java, C++, and Python, and allow for new types of analysis to be performed on complex data types. Hadoop has also been adopted by the scientific research community, who are using Hadoop to process time series data, structured binary files in the HDF5 format, and large medical and satellite images.

Crunch addresses these challenges by providing a lightweight and extensible Java API for defining the stages of a data processing pipeline, which can then be run on an Apache Hadoop cluster as a sequence of dependent MapReduce jobs, or in-memory on a single machine to facilitate fast testing and debugging. Crunch relies on a small set of primitive abstractions that represent immutable, distributed collections of objects. Developers define functions that are applied to those objects in order to generate new immutable, distributed collections of objects. Crunch also provides a library of common MapReduce patterns for performing efficient joins and aggregation operations over these distributed collections that developers may integrate into their own pipelines. Crunch also provides native support for processing structured binary data formats like JSON, Apache Avro, and Apache Thrift, and is designed to be extensible to support working with any kind of data format that Java supports in its native form.

Initial Goals

Crunch is currently in its first major release with a considerable number of enhancement requests, tasks, and issues recorded towards its future development. The initial goal of this project will be to continue to build community in the spirit of the "Apache Way", and to address the highly requested features and bug-fixes towards the next dot release.

Some goals include:

- To stand up a sustaining Apache-based community around the Crunch codebase.
- Improved documentation of Java libraries and best practices.
- Support the ability to "fuse" logically independent pipeline stages that aggregate the same data in different ways into a single MapReduce job.
- Performance, usability, and robustness improvements.
- Improving diagnostic reporting and debugging for individual MapReduce jobs.
- Providing a centralized place for contributed extensions and domain-specific applications.

Current Status

Meritocracy

Crunch was initially developed by Josh Wills in September 2011 at Cloudera. Developers external to Cloudera provided feedback, suggested features and fixes and implemented extensions of Crunch. Cloudera's engineering team has since maintained the project with Josh Wills, Tom White, and Brock Noland dedicated towards its improvement. Contributors to Crunch include developers from multiple organizations, including businesses and universities.

Community

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

Since open sourcing the project, there have been eight individuals from five organizations who have contributed code.

Core Developers

The core developers for Crunch are:

- Brock Noland: Wrote many of the test cases, user documentation, and contributed several bug fixes.
- Josh Wills: Josh wrote much of the original Crunch code.
- Gabriel Reid: Gabriel significantly improved Crunch's handling of Avro data and has contributed several bug fixes for the core planner.
- Tom White: Tom added several libraries for common MapReduce pipeline operations, including the sort library and a library of set operations.
- Christian Tzolov: Christian has contributed several bug fixes for the Avro serialization module and the unit testing framework.
- Robert Chu: Robert did the left/right/outer join implementations for Crunch and fixed several bugs in the runtime configuration logic.

Several of the core developers of Crunch have contributed towards Hadoop or related Apache projects and are familiar with Apache principles and philosophy for community driven software development.

Alignment

Crunch complements several current Apache projects. It complements Hadoop MapReduce by providing a higher-level API for developing complex data processing pipelines that require a sequence of MapReduce jobs to perform. Crunch also supports Apache HBase in order to simplify the process of writing MapReduce jobs that execute over HBase tables. Crunch makes extensive use of the Apache Avro data format as an internal data representation process that makes MapReduce jobs execute quickly and efficiently.

Known Risks

Orphaned Products

Crunch is already deployed in production at multiple companies and they are actively participating in creating new features. Crunch is getting traction with developers and thus the risks of it being orphaned are minimal.

Inexperience with Open Source

All code developed for Crunch has been open sourced by Cloudera under Apache 2.0 license. All committers to Crunch are intimately familiar with the Apache model for open-source development and are experienced with working with new contributors.

Homogeneous Developers

The initial set of committers is from a reduced 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 submission of patches from developers from several different organizations is a strong indication that Crunch will be widely adopted.

Reliance on Salaried Developers

It is expected that Crunch 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

Crunch depends upon other Apache Projects: Apache Hadoop, Apache HBase, Apache Log4J, Apache Thrift, Apache Avro, and multiple Apache Commons components. Its build depends upon Apache Maven.

Crunch's functionality has some indirect or direct overlap with the functionality of Apache Pig and Apache Hive but has several significant differences in terms of their user community and the types of data they are designed to work with. Both Hive and Pig are high-level languages that are designed to allow non-programmers to quickly create and run MapReduce jobs. Crunch is a Java library whose primary community is Java developers who are creating scalable data pipelines and MapReduce-based applications. Additionally, Hive and Pig both employ a relational, tuple-oriented data model on top of HDFS, which introduces overhead and limits expressive power for developers who are working with serialized objects and non-relational data types. Crunch uses a lower-level data model that gives developers the freedom to work with data in a format that is optimized for the problem they are trying to solve.

An Excessive Fascination with the Apache Brand

We would like Crunch to become an Apache project to further foster a healthy community of contributors and consumers around the project. Since Crunch directly interacts with many Apache Hadoop-related projects and solves an important problem of many Hadoop users, residing in the Apache Software Foundation will increase interaction with the larger community.

Documentation

- Crunch wiki at GitHub: https://github.com/cloudera/crunch/wiki
- Crunch jira at Cloudera: https://issues.cloudera.org/browse/crunch
- Crunch javadoc at GitHub: http://cloudera.github.com/crunch/apidocs/

Initial Source

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

Source and Intellectual Property Submission Plan

- The initial source is already licensed under the Apache License, Version 2.0. https://github.com/cloudera/crunch/blob/master/LICENSE.txt

External Dependencies

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

- com.google.protobuf : New BSD
- org.hamcrest: New BSD
- org.slf4j: MIT-like License

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

- Cobertura: GNU GPLv2

Note that Cobertura is optional and is only used for calculating unit test coverage.

Cryptography

Crunch uses standard APIs and tools for SSH and SSL communication where necessary.

Required Resources

Mailing lists

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

Github Repositories

http://github.com/apache/crunch git://git.apache.org/crunch.git

Issue Tracking

JIRA Crunch (CRUNCH)

Other Resources

The existing code already has unit and integration tests so we would like a Jenkins instance to run them whenever a new patch is submitted. This can be added after project creation.

Initial Committers

- Brock Noland (brock at cloudera dot com)
- Josh Wills (jwills at cloudera dot com)

- Gabriel Reid (gabriel dot reid at gmail dot com)
- Tom White (tom at cloudera dot com)
- Christian Tzolov (christian dot tzolov at gmail dot com)
- Robert Chu (robert at wibidata dot com)
- Vinod Kumar Vavilapalli (vinodkv at hortonworks dot com)

Affiliations

- Brock Noland, Cloudera
- Josh Wills, Cloudera
- Gabriel Reid, TomTom
- Tom White, Cloudera
- Christian Tzolov, TomTom
- Robert Chu, WibiData
- Vinod Kumar Vavilapalli, Hortonworks

Sponsors

Champion

- Patrick Hunt

Nominated Mentors

- Tom White
- Patrick Hunt
- Arun Murthy

Sponsoring Entity

- Apache Incubator PMC