Welcome to Apache Gora#

Table of contents

1 What is Apache Gora?

The Apache Gora open source framework provides an in-memory data model and persistence for big data. Gora supports persisting to column stores, key value stores, document stores and RDBMSs, and analyzing the data with extensive Apache Hadoop MapReduce support.

2 Why Apache Gora?

Although there are various excellent ORM frameworks for relational databases, data modeling in NoSQL data stores differ profoundly from their relational cousins. Moreover, data-model agnostic frameworks such as JDO are not sufficient for use cases, where one needs to use the full power of the data models in column stores. Gora fills this gap by giving the user an easy-to-use in-memory data model and persistence for big data framework with data store specific mappings and built in Apache Hadoop support.

The overall goal for Gora is to become the standard data representation and persistence framework for big data. The roadmap of Gora can be grouped as follows.

- Data Persistence : Persisting objects to Column stores such as HBase, Cassandra, Hypertable; key-value stores such as Voldermort, Redis, etc; SQL databases, such as MySQL, HSQLDB, flat files in local file system of Hadoop HDFS.
- Data Access : An easy to use Java-friendly common API for accessing the data regardless of its location.
- Indexing : Persisting objects to Lucene and Solr indexes, accessing/querying the data with Gora API.
- Analysis : Accesing the data and making analysis through adapters for Apache Pig, Apache Hive and Cascading
- MapReduce support : Out-of-the-box and extensive MapReduce (Apache Hadoop) support for data in the data store.

3 Background

ORM stands for Object Relation Mapping. It is a technology which abstacts the persistency layer (mostly Relational Databases) so that plain domain level objects can be used, without the cumbersome effort to save/load the data to and from the database. Gora differs from current solutions in that:

- Gora is specially focussed at NoSQL data stores, but also has limited support for SQL databases.
- The main use case for Gora is to access/analyze big data using Hadoop.
- Gora uses Avro for bean definition, not byte code enhancement or annotations.

- Object-to-data store mappings are backend specific, so that full data model can be utilized.
- Gora is simple since it ignores complex SQL mappings.
- Gora will support persistence, indexing and anaysis of data, using Pig, Lucene, Hive, etc.

4 Apache Gora Trademark Attribution

Apache Gora, Gora, Apache, the Apache feather logo, and the Apache Gora project logo are trademarks of The Apache Software Foundation.