GRUTER
Enjoy Connecting

Introduction to Apache Tajo

Bay Area Hadoop User Group

November, 5th, 2013

About Me

- Hyunsik Choi (pronounced: “Hyeon-shick Cheh”)
- PhD (Computer Science & Engineering, 2013)
- Director of Research, Gruter Corp, Seoul, South Korea

- **Open-source Involvements**
  - Full-time contributor to Apache Tajo (2013.6 ~)
  - Apache Tajo (incubating) PPMC member and committer (2013.3 ~)
  - Apache Giraph PMC member and committer (2011. 8 ~)

- **Contacts**
  - Email: hyunsik@apache.org
  - Linkedin: http://linkedin.com/in/hyunsikchoi/

Gruter

- Big Data infrastructure startup
- Hadoop platforms; Hadoop ecosystem consu lting; big data analytics layers
- Teheran Rd. tech district, Seoul, South Korea

GRUTER

Enjoy Connecting

Presentation

Apache Tajo

- Project overview
- System architecture
- Distributed processing model
- Query optimization approach
- Project status
- Project roadmap
- Q & A

Introduction to Tajo

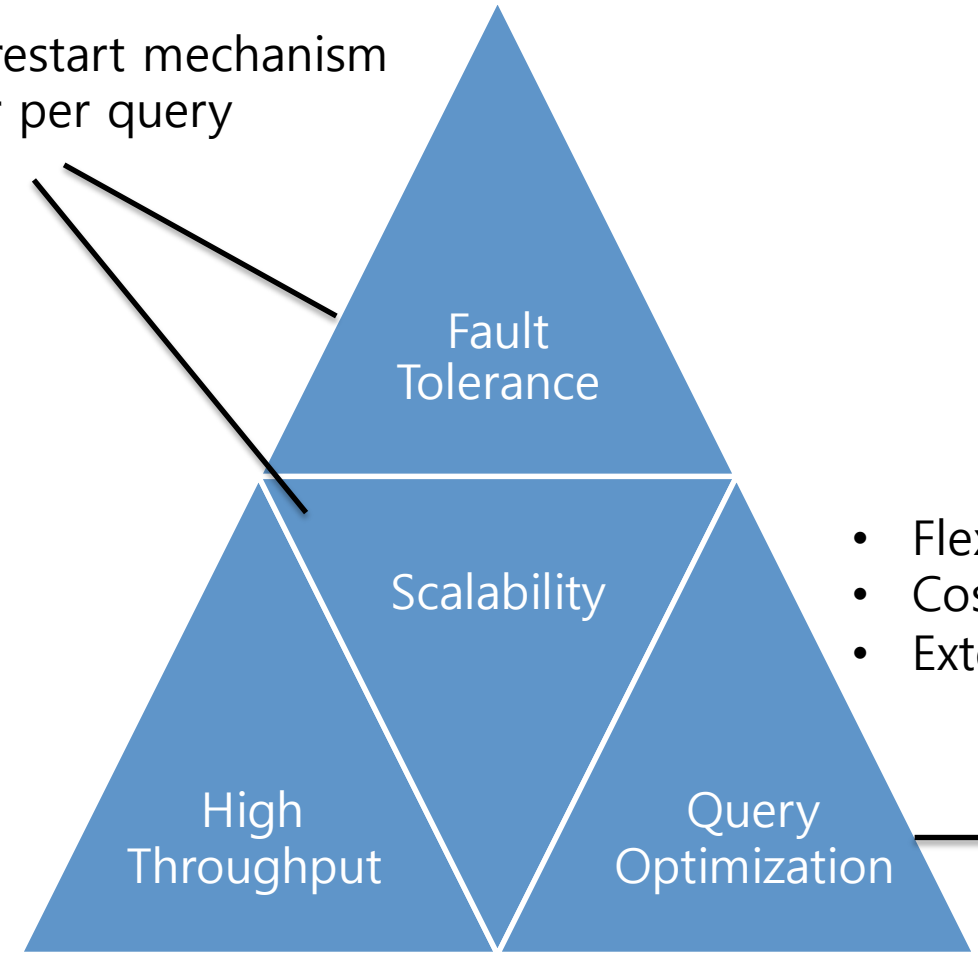
- **Tajo**
  - Big Data Warehouse System on Hadoop
  - Developed since 2010
  - Apache incubation project (entered the ASF in March 2013)

- **Features**
  - SQL standard compliance
  - Fully distributed SQL query processing
  - HDFS as a primary storage
  - Relational model (will be extended to nested model in the future)
  - ETL as well as low-latency relational query processing (100 ms ~)
  - UDF support

- **News**
  - 0.2-incubating: released November 2013
  - 0.8-incubating: to be released December 2013

Design Principles

- Failed tasks restart mechanism
- QueryMaster per query

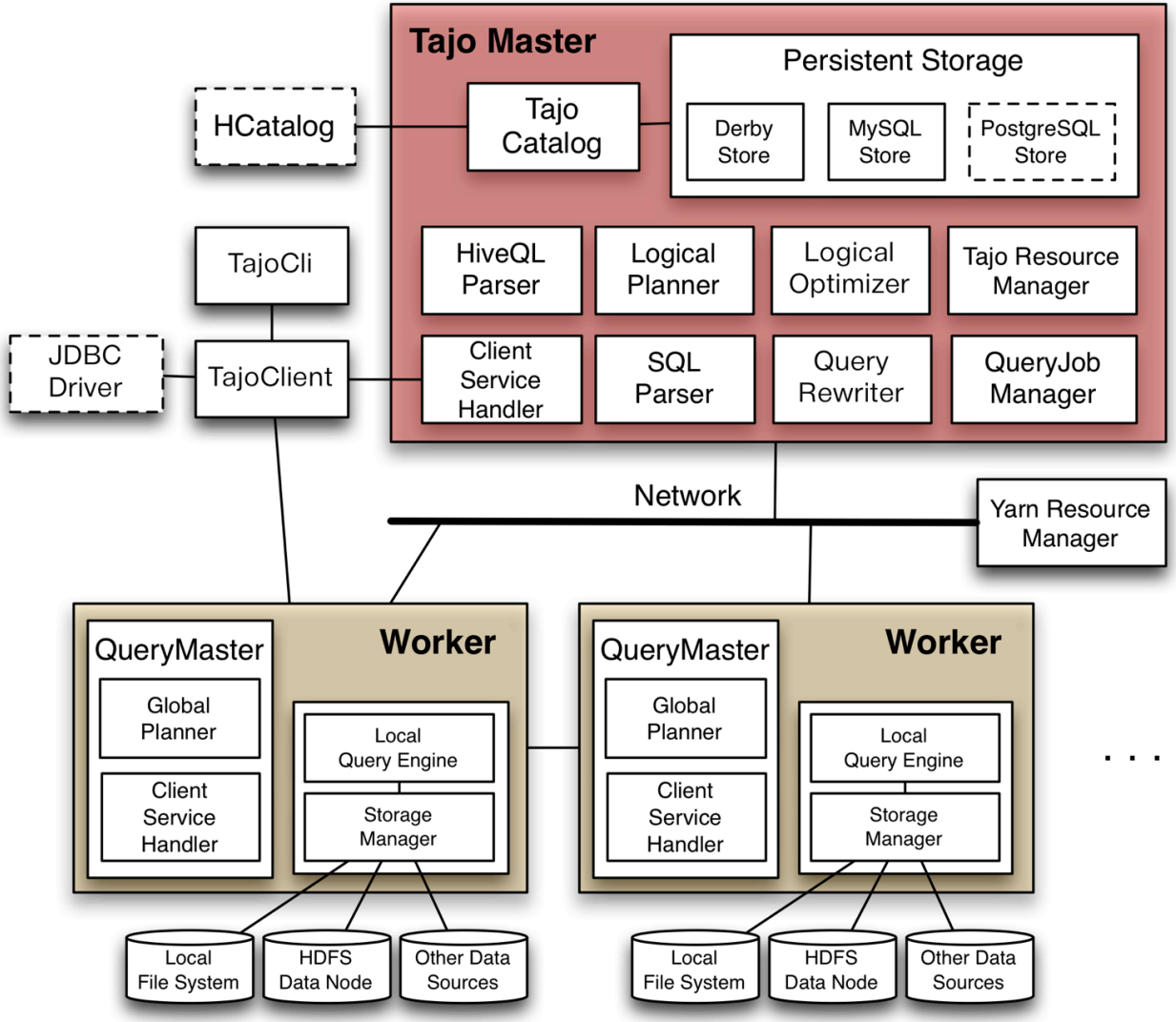


- Flexible DAG framework
- Cost-based optimization
- Extensible rewrite engine

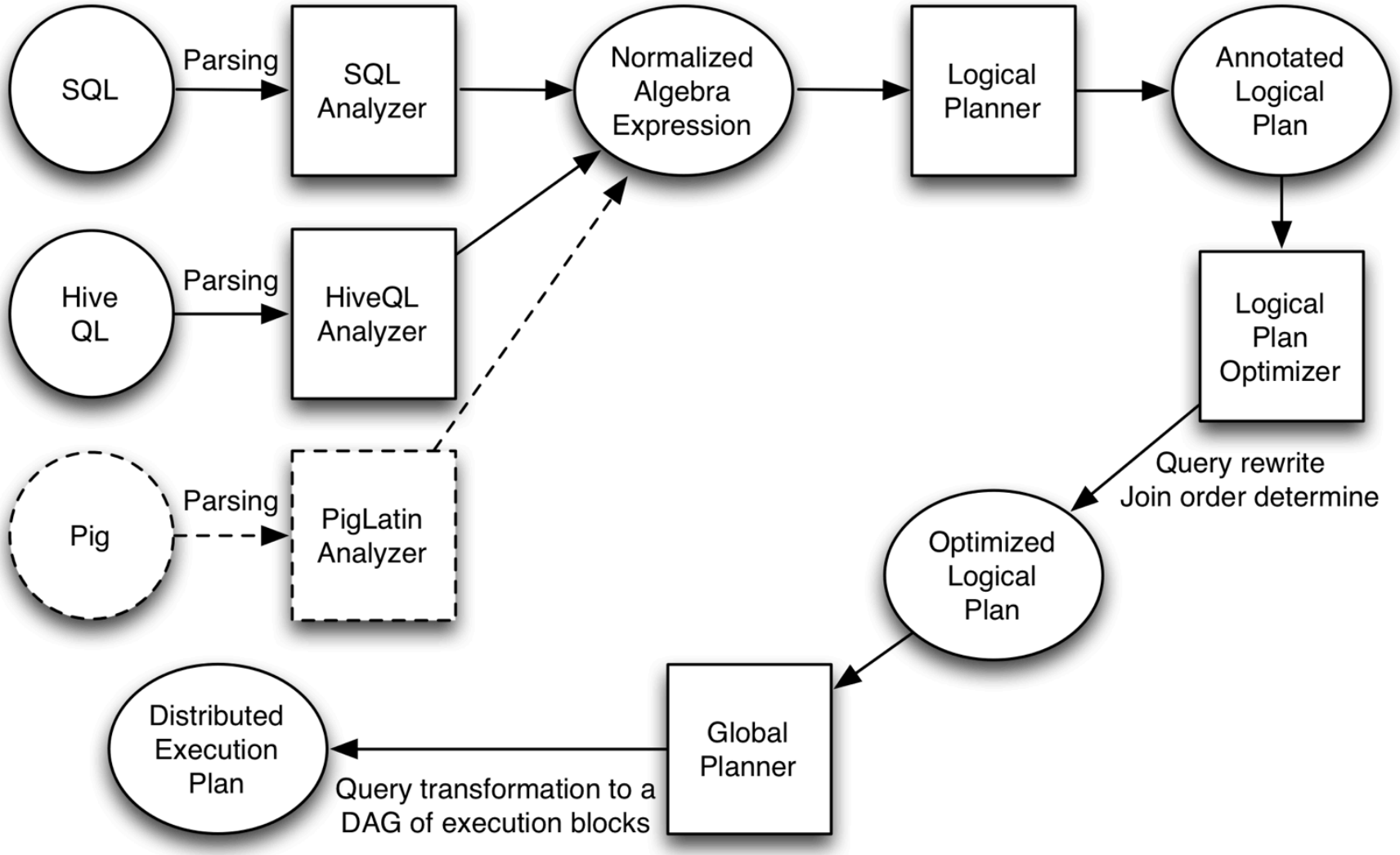
Architecture

- HDFS (Primary Storage)
- Master-Worker Model + QueryMaster per Query
  - RPC Implementation in Java (Protocol Buffer + Netty)
- Tajo Master
  - Always on standby and instant execution of some kinds of queries (DDLs)
  - Responsible for serving remote APIs to Clients
  - Query Parser and Coordination of QueryMasters
  - Embedded CatalogServer (or run independently)
- Query Master (per Query)
  - Logical plan transform to a distributed execution plan.
  - Control execution blocks (steps in a job)
  - Task scheduling
- Tajo Worker
  - Storage Manager
  - Local Query Engine

Architecture



Query Planning Process



Tajo Distributed Processing Model

- A query = a directed acyclic graph
- A vertex is a processing unit and contains:
  - A logical plan (a DAG of logical operators)
  - An enforcer (properties to force physical planning)
- Each edge represents a data flow between vertices and contains:
  - Transmission type (Pull and Push)
  - Shuffle type (range, hash, and ..)
  - The number of partitions

Data Shuffle (Edge)

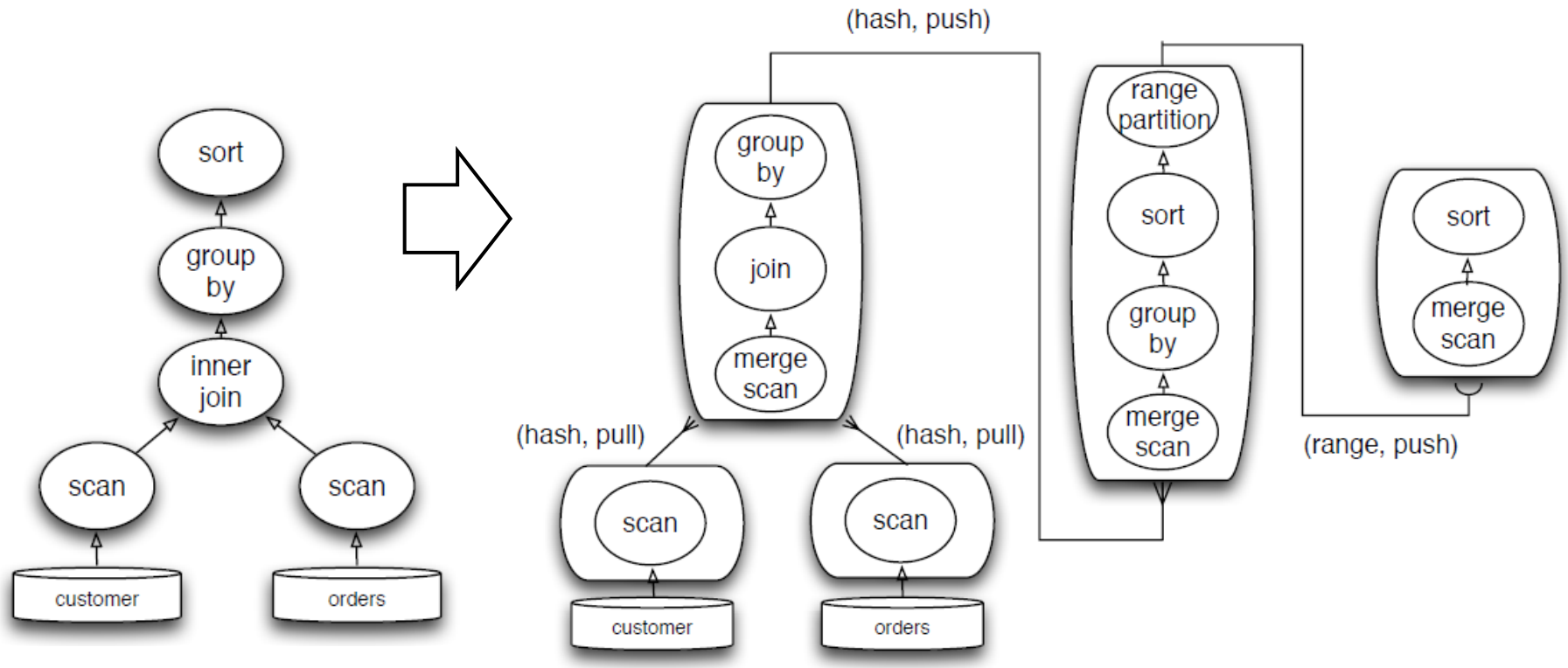
- **Shuffle Methods**
  - Hash
    - Hash repartitioning (intermediate data repartitioning via hash keys)
  - Range
    - Range repartitioning (intermediate data repartitioning to corresponding d isjoint-range-assigned nodes)

- **Transmission Methods**
  - Pull
    - Step 1: Intermediate data local disk materialization
    - Step 2: Materialized intermediate data pull
  - Push (will be supported in 0.8)
    - Intermediate data transmission (no materialization)
    - >> inter-operator pipelining enabled

An Example of Distributed Execution Plan

Join-groupby-sort query plan

Distributed query execution plan




```
select col1, sum(col2) as total, avg(col3) as average from r1, r2
where r1.col1 = r2.col2 group by col1 order by average;
```

Query Optimization

- **Cost-based Join Optimization (Greedy Heuristic)**
  - Best join order guessing eliminated!

- **Extensible Rewrite Rule Engine**
  - Enhanced rewrite rule interface with
    - Query block graph for relationships of query blocks in a query
    - Join graph for representing join relations
    - Other utilities for plan and expressions

Query Optimization

- **Progressive Query Optimization**
  - Runtime statistics collection
  - Ad hoc range and partition determination according to operator type (join, aggregation, and sort)
  - Query Reoptimization (planned)
    - Runtime join order determination and distributed join strategy
    - Pull-based or push-based transmission determination

Current Status

- SQL Support
  - Standard: ANSI SQL 2003 compliance
  - Non-standard: Extensive PostgreSQL support
    - Functions (regexp_replace, split_part,…)
  - Scheduled:
    - OuterJoin (0.8), Exists (1.0), In Subquery (1.0)

- **Distributed Join, groupby, sort operators available**
- **Blocking/Asynchronous Java Client API**
- **Tajo Catalog**
  - Derby and MySQL persistent store
  - tajo_dump, an utility for backup and restore

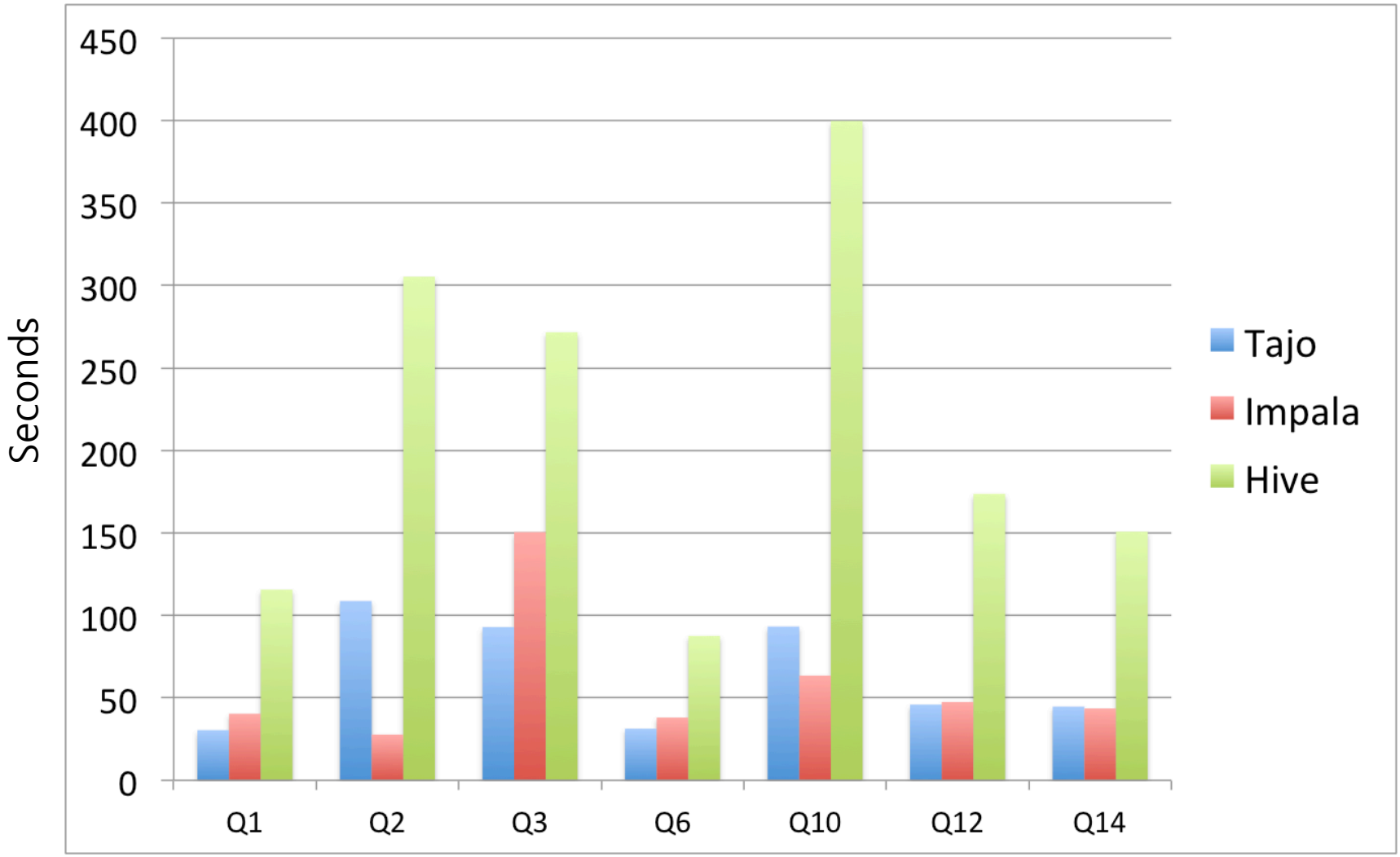
Current Status

- **Various file format supports**
  - CSV format
  - RowFile (Tajo's own row store file format)
  - RawFile (for local disk/network materialization)
  - RCFile (Text/Binary (de)serializer and Hive compatible)
  - Parquet (the next release)
- **Scanner/Appender Interface for custom file formats**
- **Very fast scan performance**
  - QueryMaster schedules tasks with balancing disk volume loads for each node
  - As a result, disk-bound queries show average scan 60-110 MB /s per disk (SATA2 and SAS)

Experiments

- Tajo (master branch) vs. Impalad_version 1.1.1 vs. Hive 0.10-cdh4
- TPC-H data set 100GB
- **Cache dropped for each experiment**
- 10G networks
- 6 cluster nodes
- Each machine is equipped with :
  - Intel Xeon CPU E5 2640 2.50GHz x 4
  - 64 GB memory
  - 6 SATA2 disks

Experiments



Some of TPC-H Queries on 100GB

Roadmap

- 0.2 release is being voted on incubator-general@a.o
- December 2013: Apache Tajo 0.8 Release
  - More SQL standard features; more stability
  - Outer join support
  - Hadoop 2.2.0-beta porting
  - Table Partitioning: Hash, Range, List, Column (Hive style)
  - Catalog access to HCatalog
- Q1 2014: Apache Tajo 1.0-alpha release
  - More powerful rewrite rules
  - More fault tolerance
  - Window functions
  - A prototype of JIT query compilation & vectorized engine

Get Involved!

- Getting Started
  - http://wiki.apache.org/tajo/GettingStarted
- Checkout the development branch
  - http://wiki.apache.org/tajo/BuildInstruction
- Jira - Issue Tracker
  - https://issues.apache.org/jira/browse/TAJO
- Join the mailing list
  - tajo-dev-subscribe@incubator.apache.org