cloudera®

Impala

The best analytic database for Hadoop

Alan Choi // Software Engineer

Notification

- The information in this document is proprietary to Cloudera. No part of this document may be reproduced, copied or transmitted in any form for any purpose without the express prior written permission of Cloudera.
- This document is a preliminary version and not subject to your license agreement or any other agreement with Cloudera. This document contains only intended strategies, developments and functionalities of Cloudera products and is not intended to be binding upon Cloudera to any particular course of business, product strategy and/or development. Please note that this document is subject to change and may be changed by Cloudera at any time without notice.
- Cloudera assumes no responsibility for errors or omissions in this document. Cloudera does not warrant the accuracy or completeness of the information, text, graphics, links or other items contained within this material. This document is provided without a warranty of any kind, either express or implied, including but not limited to the implied warranties of merchantability, fitness for a particular purpose or non-infringement.
- Cloudera shall have no liability for damages of any kind including without limitation direct, special, indirect or consequential damages that may result from the use of these materials. The limitation shall not apply in cases of gross negligence.

Agenda

- Impala overview
- Most common use cases
- SQL-on-Hadoop perf update
- Milestones

Agenda

- **Impala overview**
- Most common use cases
- SQL-on-Hadoop perf update
- Milestones

Analytic database for Hadoop requirements

Analytic Databases require…

Multi-user Perf & Usability	Meets user experience expectations at standard load (e.g. 100s or 1000s of users)
Compatibility	Familiar BI tools/SQL interfaces

Hadoop requires…

Flexibility	Use SQL to access any type of data, and access any type of data with more than just SQL
Native Integration	Unified resource management, metadata, security, and management across frameworks

Impala: analytic database for Hadoop

Impala delivers the best of both worlds.

Multi-user Perf & Usability	✓	• 10x performance vs. alternatives for BI workloads
Compatibility	✓	• Provides both ANSI SQL and vendor-specific extensions • Support for the leading BI tools

Flexibility	✓	• Supports the common native Hadoop file formats, e.g. Parquet, Avro, text • Works together with other Hadoop frameworks
Native Integration	✓	• Unified with Hadoop metadata, security, governance, and administration

Agenda

- Impala overview
- **Most common use cases**
- SQL-on-Hadoop perf update
- Milestones

cloudera

Most common use cases

Operational dashboards

Example: Healthcare Insurance Company

Goal:

- Visualizations of current hospital spending and comparison to peers and historical data
- Integrate 1000s of client hospital purchasing systems

Key benefits of Impala:

- Simplification via unification
- Saved license $ over traditional DBMS
- Enabled finer-grain details in source data vs. planned summarized extracts
- 3 nodes of Impala outperformed a rack of the traditional RDBMS on their workload

Data discovery

Example: Major Financial Institution

Goal:

- Fraud group looking at internal / external fraud
- Captured internal systems and external application/ website logs

Key benefits of Impala:

- Flexibility to have more data readily available without upfront modeling
- Ability to use existing BI visualization tools
- Better TCO

Agenda

- Impala overview
- Most common use cases
- **SQL-on-Hadoop perf update**
- Milestones

September SQL-on-Hadoop benchmark: Impala, Presto, Stinger, Spark SQL

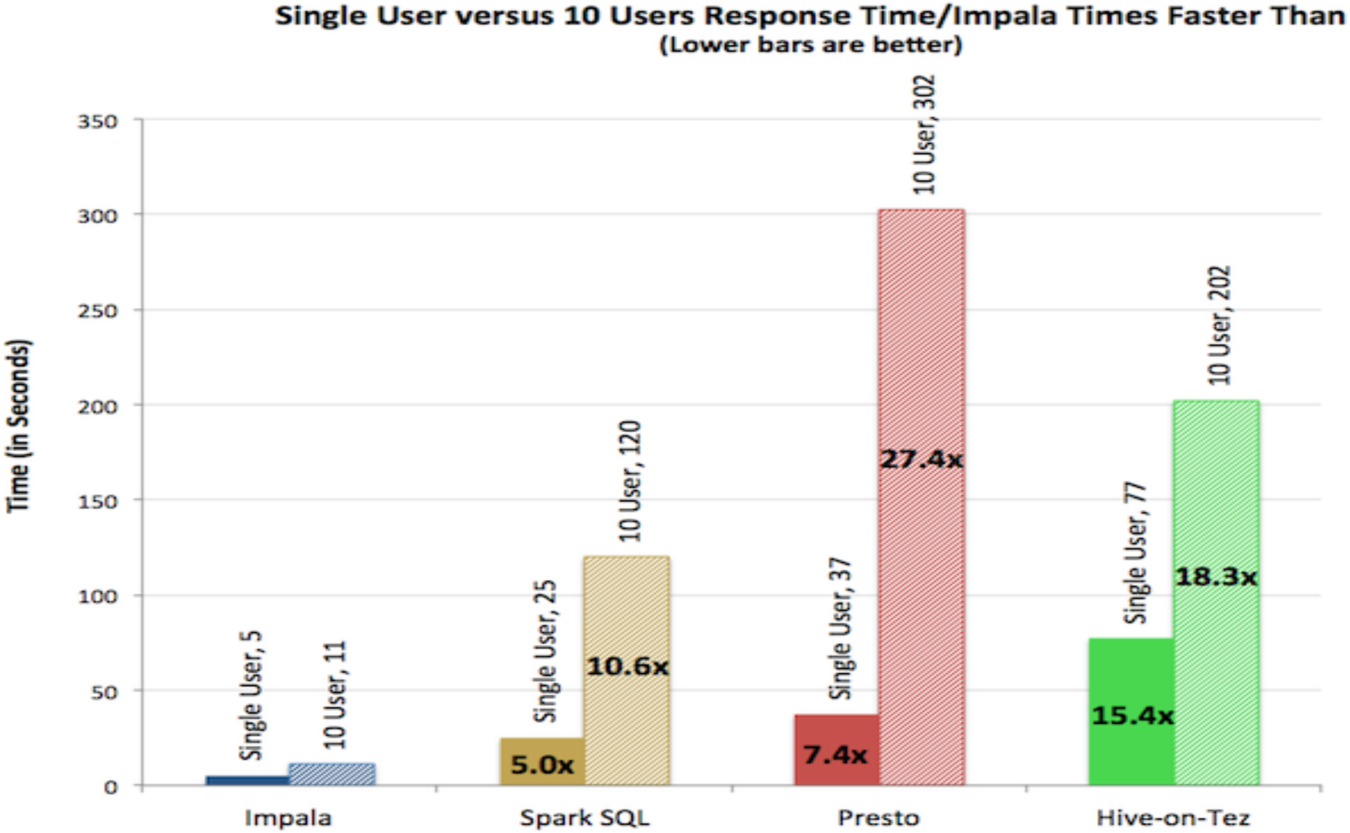
- Benchmarks on latest versions of:
  - Impala (1.4.0)
  - Presto (0.74)
  - Stinger (final) phase 3 => aka Hive 0.13.0
  - Spark SQL (1.1)

- As always, our public benchmarks are:
  - Based on industry standards (TPC)
  - Repeatable (https://github.com/cloudera/impala-tpcds-kit)
  - Methodical testing with multiple runs on same hardware
  - Help competing software put its best foot forward
    - SQL-92 join style for engines without CBO
    - JVM tuning for Presto
    - Run on optimal file formats for each

- Full blog: http://blog.cloudera.com/blog/2014/09/new-benchmarks-for-sql-on-hadoop-impala-1-4-widens-the-performance-gap/

Impala's Multi-User over 10x faster with just 10 users:

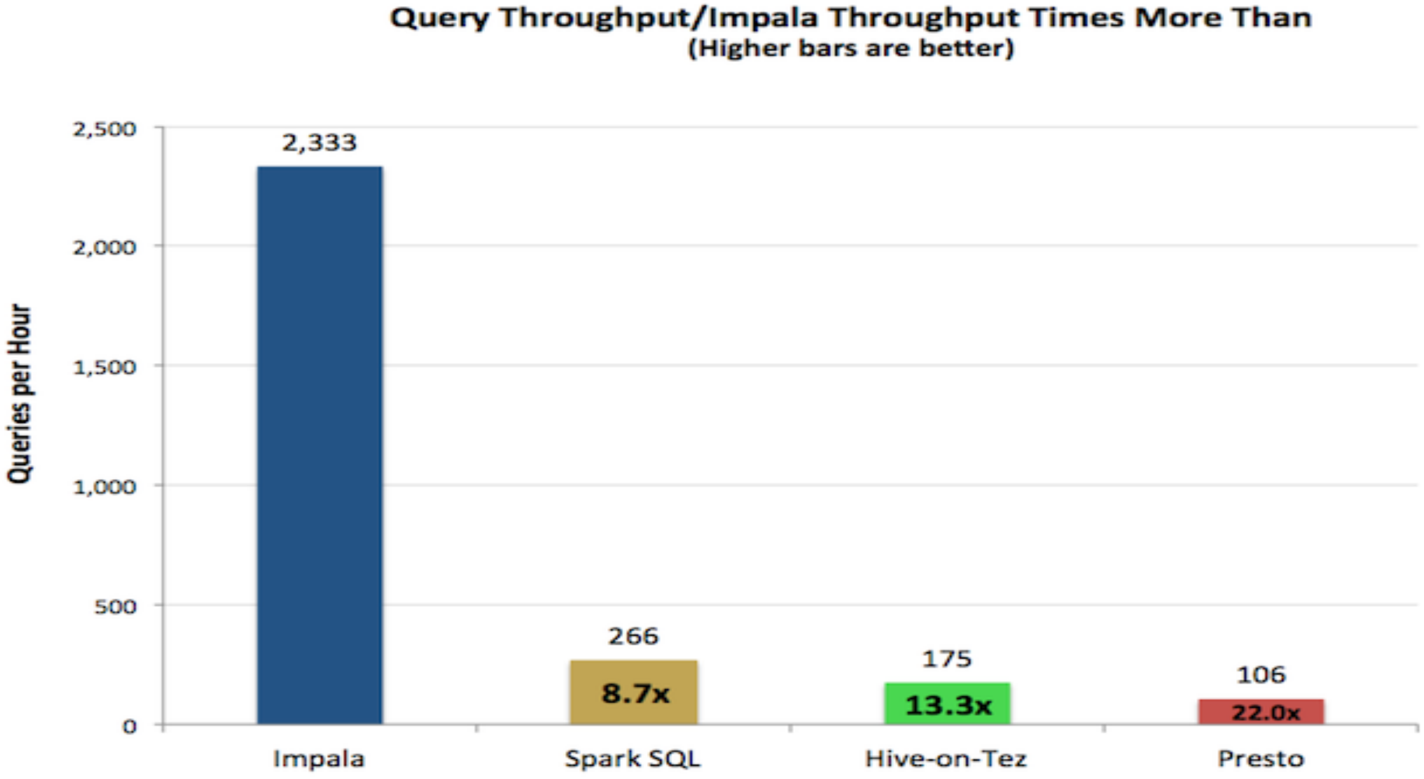
Gap widening compared to May's update



cloudera

Faster = more work in less time:

Impala enables over 8.7x throughput



cloudera

IBM Research validation

- New VLDB academic paper comparing Impala and Hive-based (both MR and Tez) for SQL-on-Hadoop
  - http://www.vldb.org/pvldb/vol7/p1295-floratou.pdf

- **Impala's significantly more efficient than Hive/Tez or Hive/MR**
  - "Impala's database-like architecture provides significant performance gains, compared to Hive's MapReduce or Tez based runtime"
  - Correctly attributes Impala's lead to it's CPU efficiency, IO manager, and overall architecture that resembles a shared-nothing parallel database

- **Parquet more efficient than ORC**
  - "The Parquet format skips data more efficiently than ORC which tends to prefetch unnecessary data especially when a table contains a large number of columns"

- Note: Paper is single-user only. Multi-user would make the gap even wider
  - Our published results show ~5x single-user Impala lead goes to ~10x with just 10 users in our blog: http://blog.cloudera.com/blog/2014/05/new-sql-choices-in-the-apache-hadoop-ecosystem-why-impala-continues-to-lead/
  - Same CPU efficiency, IO manager, and overall architectural reasons

- Additional Notes:
  - Impala 2.0 has disk-based joins and aggregations
  - Impala 1.4 is significantly faster on selective joins than Impala 1.2.2 used in the paper

Agenda

- Impala overview
- Most common use cases
- SQL-on-Hadoop perf update
- **Milestones**

Previous milestones

- **Impala 1.0 (April 2013)**
  - GA availability

- **Security: Impala 1.1 (summer 2013)**
  - Authentication (already available in 1.0)
  - Authorization via Apache Sentry
  - Auditing

- **Usability: Impala 1.2 (fall 2013)**
  - Custom language extensibility (UDFs, UDAFs)
  - Cost-based join-order optimization
  - On-par performance compared to traditional MPP query engines while maintaining native Hadoop data flexibility

- **Resource management: Impala 1.3 (spring 2014)**
  - Resource management

- **Compatibility: Impala 1.4 (July 2014)**
  - More standard SQL and vendor-specific extensions
  - DECIMAL data type

Impala 2.0 key updates

- Same great multi-user interactive performance

- Removed limits on SQL compatibility
  - SQL:2003 analytic/window functions
  - Subqueries in WHERE clause, EXISTS, and IN
  - Additional data types (CHAR and VARCHAR)
  - GRANT/REVOKE functions via Sentry
  - Additional vendor-specific SQL extensions

cloudera

Thank you.