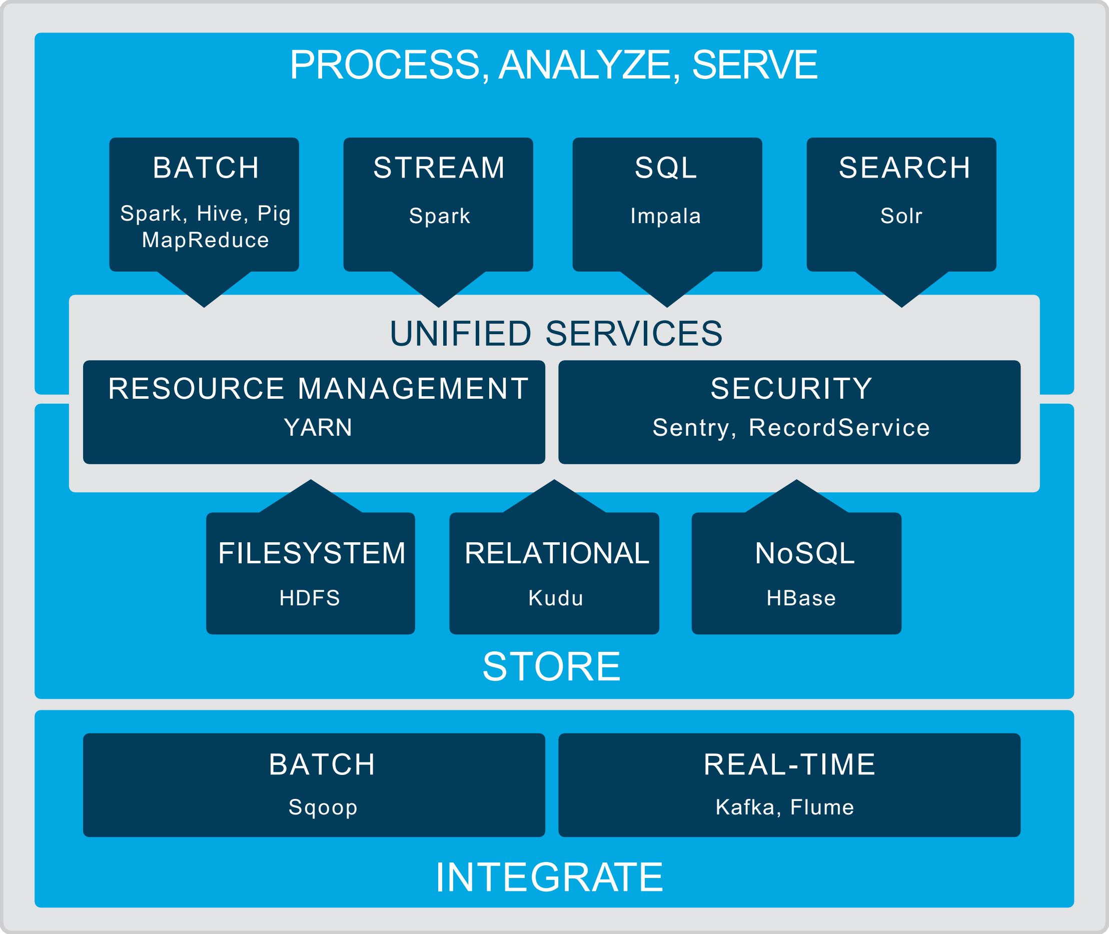
# Introduction to Apache Hive

# Review: Hadoop Data Processing and Analysis

- **Hadoop includes many tools for data processing and analysis**



.

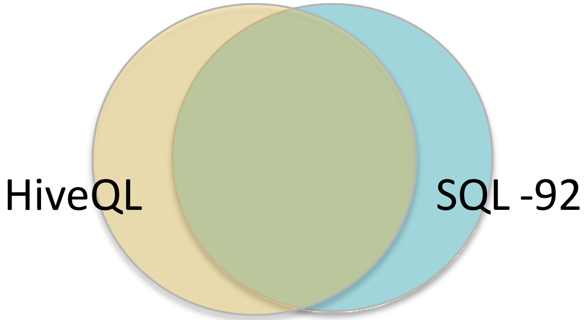
# What Is Apache Hive?

- **Hive is data warehouse infrastructure for Hadoop**
  - Alternative to writing low-level MapReduce code
  - Uses a SQL-like language called HiveQL
  - Generates jobs that run on the Hadoop cluster
  - Originally developed by Facebook
    - Now an open source Apache project

# HiveQL

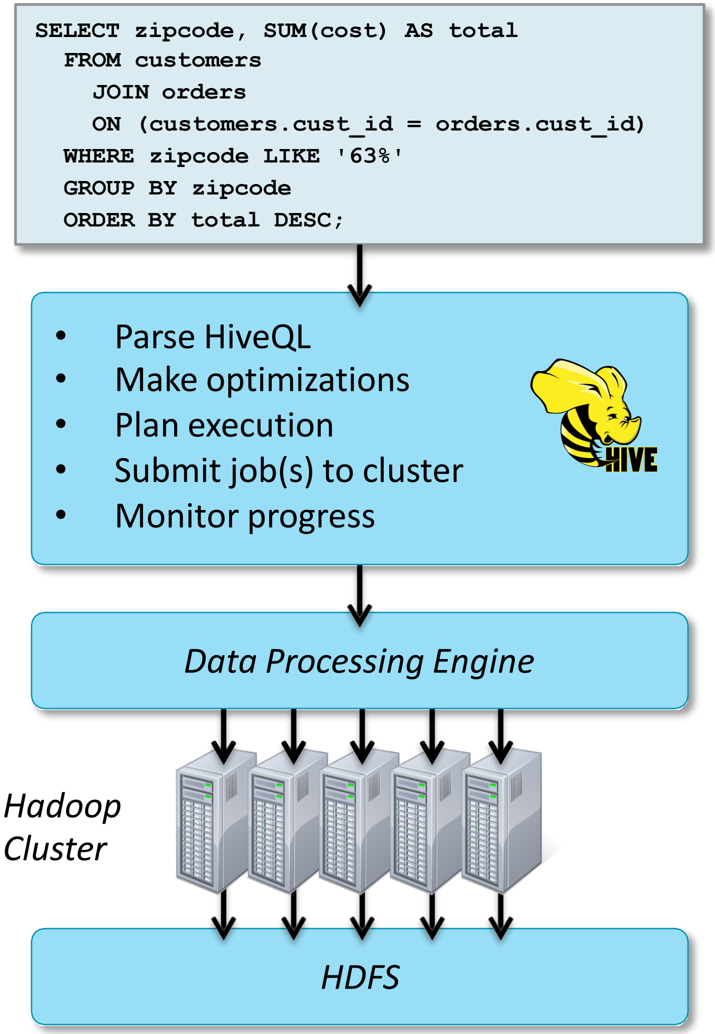
- **HiveQL implements a subset of SQL-92**
  - Plus a few extensions found in MySQL and Oracle SQL dialects

```
SELECT zipcode, SUM(cost) AS total
  FROM customers
    JOIN orders
    ON (customers.cust_id = orders.cust_id)
  WHERE zipcode LIKE '63%'
  GROUP BY zipcode
  ORDER BY total DESC;
```



# Hive High-Level Overview

- Hive turns HiveQL queries into data processing jobs
- Then it submits those jobs to the data processing engine (MapReduce or Spark) to execute on the cluster



```
SELECT zipcode, SUM(cost) AS total
  FROM customers
    JOIN orders
    ON (customers.cust_id = orders.cust_id)
  WHERE zipcode LIKE '63%'
  GROUP BY zipcode
  ORDER BY total DESC;
```

- Parse HiveQL
- Make optimizations
- Plan execution
- Submit job(s) to cluster
- Monitor progress

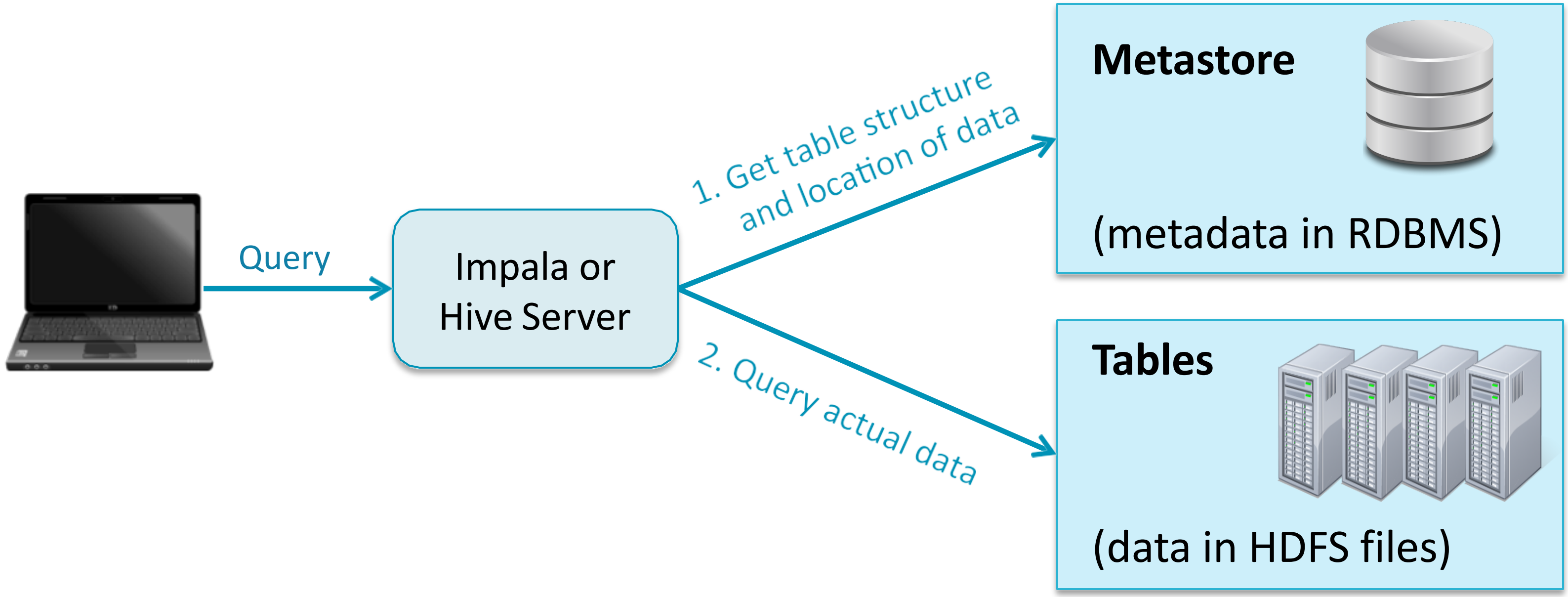
# How Hive and Impala Load and Store Data (1)

- **Queries operate on tables, just like in an RDBMS**
  - A table is typically an HDFS directory containing one or more files
  - Default path: **`/user/hive/warehouse/`*`tablename`***
  - Supports many formats for data storage and retrieval
- **You specify the structure and location of tables when creating them**
  - This metadata is stored in the *metastore*
    - Contained in an RDBMS such as MySQL
- **Hive and Impala work with the same data**
  - Tables in HDFS, metadata in the metastore

.

# How Hive and Impala Load and Store Data (2)

- **Hive and Impala use the metastore to determine data format and location**
  - The query itself operates on data stored in a filesystem (typically HDFS)



Query

Impala or Hive Server

1. Get table structure and location of data

2. Query actual data

**Metastore**

(metadata in RDBMS)

**Tables**

(data in HDFS files)

# Your Cluster Is Not a Database Server

- **Client-server database management systems have many strengths**
  - Have very fast response time
  - Include support for transactions
  - Allow modification of existing records
  - Can serve thousands of simultaneous clients
- **Hive and Impala do not turn your cluster into an RDBMS**
  - No support for updating and deleting records
  - No transaction support
  - No referential integrity

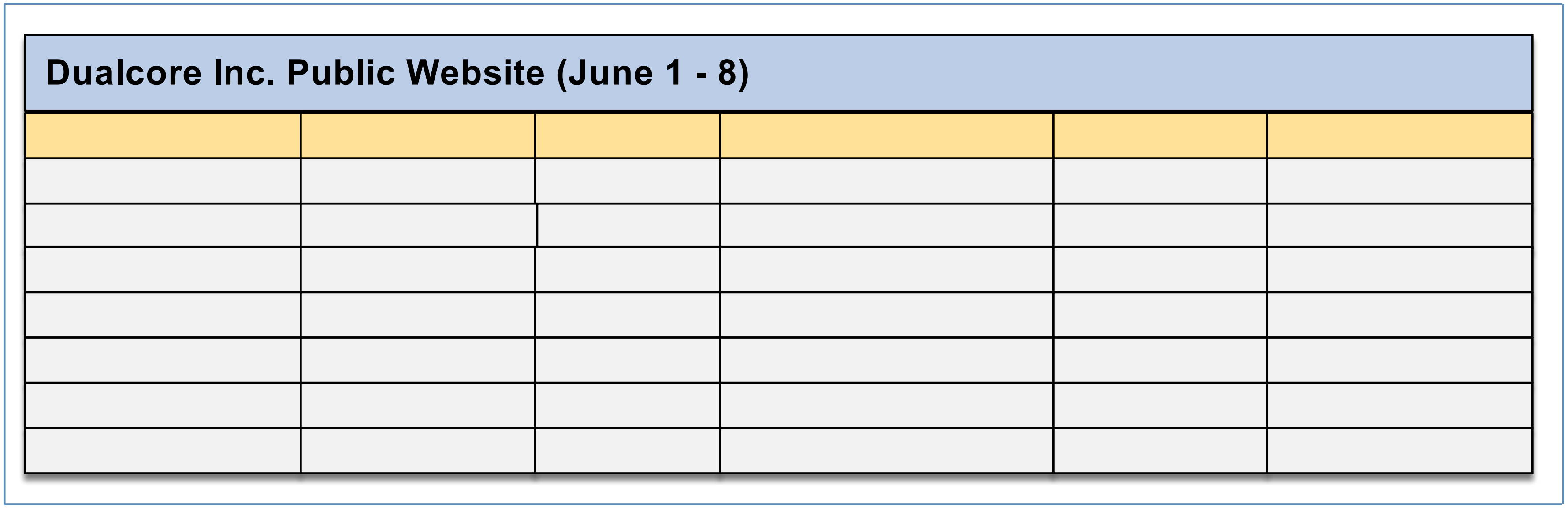
# Comparing Hive and Impala to a Relational Database

| Feature | RDBMS | Hive | Impala |
|---|---|---|---|
| Query language | SQL (full) | SQL (subset) | SQL (subset) |
| Update individual records | Yes | No* | No |
| Delete individual records | Yes | No* | No |
| Transactions | Yes | No* | No |
| Index support | Extensive | Limited | No |
| Latency | Very low | High | Low |
| Data size | Terabytes | Petabytes | Petabytes |
| Storage cost | Very high | Very low | Very low |

* Hive now has limited, experimental support for **`UPDATE`**, **`DELETE`**, and transactions. Cloudera neither recommends nor supports using these features in Hive.

# Use Case: Log File Analytics

- **Server log files are an important source of data**
- **Hive and Impala allow you to treat a directory of log files like a table**
  - Allows SQL-like queries against raw data

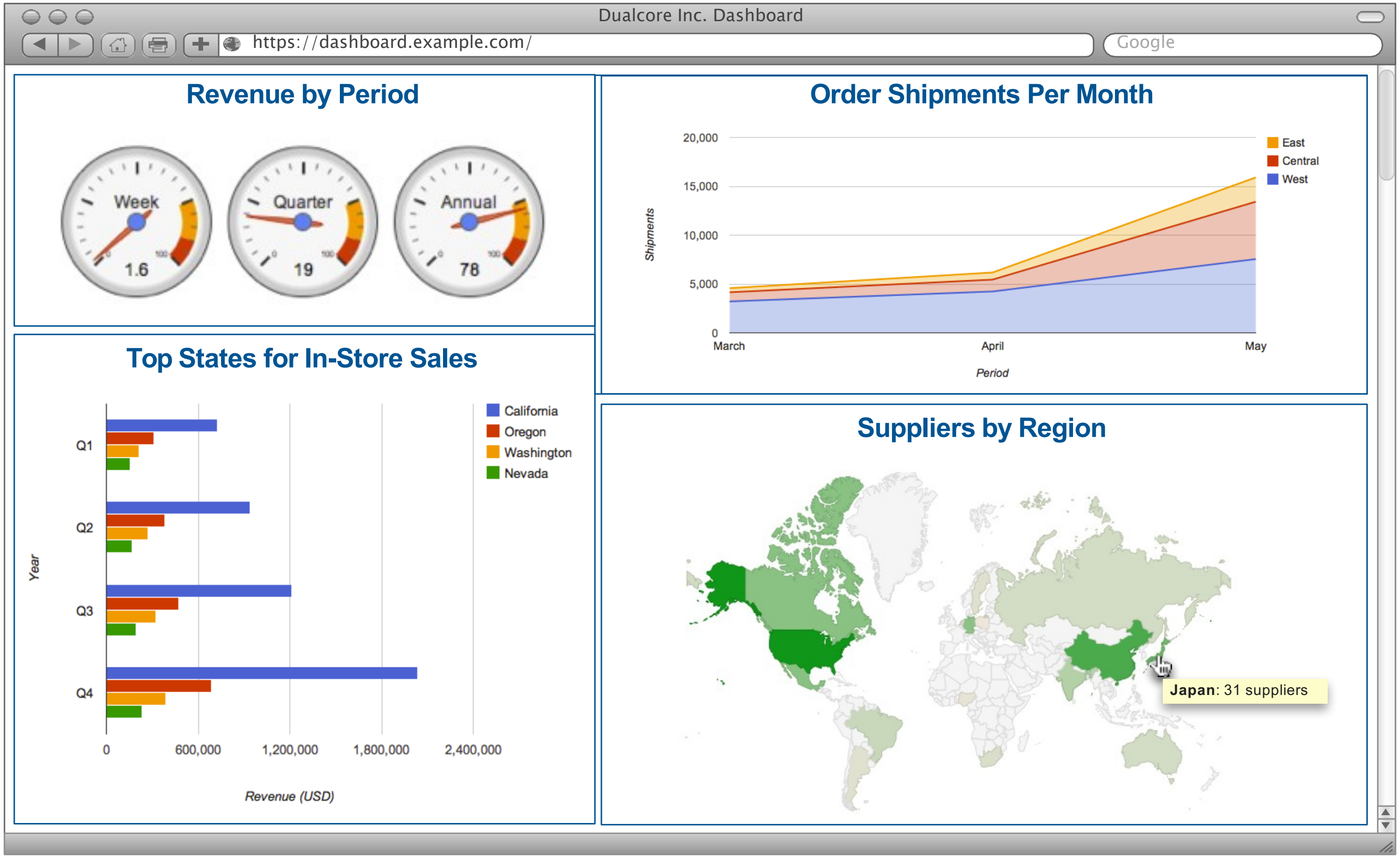


**Dualcore Inc. Public Website (June 1 - 8)**

| | | | | | |
|---|---|---|---|---|---|
| | | | | | |
| | | | | | |
| | | | | | |
| | | | | | |
| | | | | | |
| | | | | | |
| | | | | | |

# Use Case: Business Intelligence

- **Many leading business intelligence tools support Hive and Impala**



# Lab -  Hive Installation – 90 Minutes