Big-data Management

Topics not Covered
What We Covered

• Storage (HDFS)
• Query processing (MapReduce, RDD, Hyracks)
• Higher-level data flow engines (Pig, SparkSQL, Spark Streaming)
• Storage formats (row, column, hybrid)
• Indexing (Global/local and LSM)
• Application-specific (Big Spatial Data)
Topics not Covered

• Key-value stores
• Big graph analytics
• Visualization
• Streaming
• Coordination
• Cloud platforms
Key-value Stores

- Provides a simple API to insert/delete/update/search key-value pairs
- Records are indexed by key (typically a string)
- Internal structure is typically a Log-structured-merge tree (LSM)
- Not generally suitable for large-scale analytics
Insert a record

Text rowID = new Text("row1");
Text colFam = new Text("myColFam");
Text colQual = new Text("myColQual");
ColumnVisibility colVis = new ColumnVisibility("public");
long timestamp = System.currentTimeMillis();
Value value = new Value("myValue").getBytes();

Mutation mutation = new Mutation(rowID);
mutation.put(colFam, colQual, colVis, timestamp, value);

Search for records

// specify which visibilities we are allowed to see
Authorizations auths = new Authorizations("public");

Scanner scan =
    conn.createScanner("table", auths);

scan.setRange(new Range("harry","john"));
scan.fetchColumnFamily(new Text("attributes"));

for(Entry<Key,Value> entry : scan) {
    Text row =
        entry.getKey().getRow();
    Value value = entry.getValue();
}
Big Graph Analytics

• Graphs are usually processed using a node-centric processing model
• Nodes and edges are both treated as first-class citizens
• Processing is normally iterative with a lot of iterations
Visualization

- Sometimes called Business Intelligence (BI)
- Focuses more on the end-user interface while producing nice graphs (e.g., bar charts and line graphs)
- Internally, the data is managed using the common big-data platforms but the systems are tuned to provide fast query response for ad-hoc queries
Streaming

• Some applications need to process data in real-time with a very small latency
• Examples: Twitter search, IoT applications, and social network trends
• Works primarily off main memory
• Keeps only the latest records to ensure real-time response

Flink

STORM
Coordination

• Most big-data systems are designed for shared-nothing large-scale analytics
• No coordination between machines is part of the design
• Coordination systems provide an easy way to coordinate the work in these distributed platforms, e.g., a catalog of information, work queue, and a global system status

Apache Zookeeper
Cloud Platforms

• Maintaining your own cluster is costly
• It could be underutilized most of the time
• Cloud platforms allow you to rent virtual machines to do your work and dispose them after
• They are well-integrated with big data platforms (such as Hadoop and Spark) to give the best user experience
• All you need is an internet connection and a credit card
Learning Materials

- Always start with the official project page
- Google for articles with code examples, questions on StackOverflow, and YouTube for presentations
- Read research papers
- Dig into the source code if you need to understand more