Big Data

Kevin Kalmbach
Principal Sales Consultant, Public Sector Engineered Systems
Program Agenda

• What is Big Data and why it is important?
• What is your Big Data Strategy?
• Oracle Big Data Offering
Big Data Defined

**Volume**
- Very large quantities of data

**Velocity**
- Extremely fast streams of data

**Variety**
- Wide range of datatype characteristics

**Value**
- High potential business value if harnessed
Big Data. Big Architecture.

Challenge:
Exploiting Synergies
Discovering New Value in Unconventional Data

**Exhibit 25: Structured Data Example**

<table>
<thead>
<tr>
<th>Name</th>
<th>Thread pitch (mm)</th>
<th>Minor diameter tolerance</th>
<th>Nominal diameter (mm)</th>
<th>Head shape</th>
<th>Price for 50 screws</th>
<th>Available at factory outlet?</th>
<th>Number in stock</th>
<th>Flat or Phillips head?</th>
</tr>
</thead>
<tbody>
<tr>
<td>M4</td>
<td>0.7</td>
<td>4g</td>
<td>4</td>
<td>Pan</td>
<td>$10.08</td>
<td>Yes</td>
<td>276</td>
<td>Flat</td>
</tr>
<tr>
<td>M5</td>
<td>0.8</td>
<td>4g</td>
<td>5</td>
<td>Round</td>
<td>$13.69</td>
<td>Yes</td>
<td>183</td>
<td>Both</td>
</tr>
<tr>
<td>M6</td>
<td>1</td>
<td>6g</td>
<td>6</td>
<td>Button</td>
<td>$10.42</td>
<td>Yes</td>
<td>1040</td>
<td>Flat</td>
</tr>
<tr>
<td>M8</td>
<td>1.25</td>
<td>8g</td>
<td>8</td>
<td>Button</td>
<td>$11.98</td>
<td>No</td>
<td>298</td>
<td>Phillips</td>
</tr>
<tr>
<td>M10</td>
<td>1.5</td>
<td>6g</td>
<td>10</td>
<td>Round</td>
<td>$16.74</td>
<td>Yes</td>
<td>488</td>
<td>Phillips</td>
</tr>
<tr>
<td>M12</td>
<td>1.75</td>
<td>1g</td>
<td>12</td>
<td>Pan</td>
<td>$18.26</td>
<td>No</td>
<td>998</td>
<td>Flat</td>
</tr>
<tr>
<td>M14</td>
<td>2</td>
<td>8g</td>
<td>14</td>
<td>Round</td>
<td>$21.18</td>
<td>No</td>
<td>235</td>
<td>Phillips</td>
</tr>
<tr>
<td>M16</td>
<td>2</td>
<td>8g</td>
<td>16</td>
<td>Button</td>
<td>$25.57</td>
<td>Yes</td>
<td>292</td>
<td>Both</td>
</tr>
<tr>
<td>M18</td>
<td>2.1</td>
<td>1g</td>
<td>18</td>
<td>Button</td>
<td>$20.87</td>
<td>No</td>
<td>564</td>
<td>Both</td>
</tr>
<tr>
<td>M20</td>
<td>2.4</td>
<td>4g</td>
<td>20</td>
<td>Pan</td>
<td>$20.00</td>
<td>Yes</td>
<td>486</td>
<td>Both</td>
</tr>
<tr>
<td>M24</td>
<td>2.55</td>
<td>9g</td>
<td>24</td>
<td>Round</td>
<td>$33.01</td>
<td>Yes</td>
<td>992</td>
<td>Phillips</td>
</tr>
<tr>
<td>M28</td>
<td>2.7</td>
<td>1g</td>
<td>28</td>
<td>Button</td>
<td>$35.66</td>
<td>No</td>
<td>1067</td>
<td>Phillips</td>
</tr>
<tr>
<td>M30</td>
<td>3.2</td>
<td>1g</td>
<td>36</td>
<td>Pan</td>
<td>$41.52</td>
<td>No</td>
<td>434</td>
<td>Both</td>
</tr>
<tr>
<td>M50</td>
<td>4.5</td>
<td>1g</td>
<td>50</td>
<td>Pan</td>
<td>$44.72</td>
<td>No</td>
<td>740</td>
<td>Flat</td>
</tr>
</tbody>
</table>

**Exhibit 26: Quasi-Structured Data Example**

Manage the structured data from your applications.

Manage and extract value from the unstructured data you generate (logs, sensors, feeds, call center,...)
Impact on Your Information Architecture

New
higher volume, variety processing

New
analytical methods for highly detailed data

New
correlation with structured data

- Extreme scalability
- Hardware economics
- Dynamic data model

- Statistics
- Visualization
- Iterative

- Share metadata
- Leverage skills
- Unify dashboards
What is your Big Data Strategy?

- How are you planning to make use of quasi-structured data coming from social media sites, web logs, or sensors and automated data gathering devices to make better decisions?

- How are you planning to leverage your existing business intelligence infrastructure so that you can widely disseminate this data?

- Where do you believe the potential return on investment could come from in analyzing all of your data?
Hadoop Ecosystem
Some Facts about Apache Hadoop

• It’s Open Source (using the Apache license)
• Based on work done by Google in the early 2000s
  – Specifically, on papers describing the Google File System (GFS) published in 2003,
    and MapReduce published in 2004
• Currently around 40 core Hadoop committers from ~10 companies
  – Cloudera, Yahoo!, Facebook, Apple and more
• Hundreds of contributors writing features, fixing bugs
• Users include: IBM, eBay, Twitter, Amazon, NetFlix, and Visa
• Vendors Integrations: Teradata (AsterData, HortonWorks), IBM (Netezza), Quest,
  HP (Vertica), MicroStrategy, plus many smaller companies
• Many related projects, application, tools
Maximize the Value of Enterprise Big Data

Oracle Integrated Solution Stack for Big Data

**ACQUIRE**
- HDFS
- Oracle NoSQL Database
- Enterprise Applications

**ORGANIZE**
- Hadoop (MapReduce)
- Oracle Loader for Hadoop
- Oracle Data Integrator

**ANALYZE**
- Data Warehouse
- In-Database Analytics

**DECIDE**
- Business Intelligence Applications
Hadoop Components

• Hadoop consists of two core components
  – The Hadoop Distributed File System (HDFS)
  – The MapReduce Application Framework

• Other projects based around core Hadoop
  – Referred to as the ‘Hadoop Ecosystem’
  – Include things like Pig, Hive, HBase, Flume, Oozie, Sqoop, etc.
  • Tools for ingest, retrieval, query and representation

• A set of machines running HDFS and MapReduce is a Hadoop Cluster
  – Individual machines are known as nodes
  – A cluster can have as few as one node or several thousand
  – Facebook has 4,500 nodes, while the median number is 150
  – More nodes = better performance!
Hadoop Core Concepts

• Data Distribution
• Parallel Code Execution
• Extreme Fault Tolerance
Hadoop Core Concepts – Data Distribution

• Based on clusters of servers with local storage
• Shared Nothing Architecture
  – minimal communication between nodes
• MapReduce is the Functional Programming Paradigm used to process data in the Hadoop cluster
  – MapReduce is a batch process – not real-time
  – Consists of two primary “programmable” phases:
    1. Map: reads the file and extracts intermediate data
    2. Reduce: aggregates/combines output of the mappers
• Data Movement/Network usage is minimized
  – Data is initially loaded into large ‘blocks’ of 64 or 128Mb and distributed among the nodes of the system
  – Computation work is done on data stored locally on a node whenever possible
  – No data transfer over the network is required for initial processing
Hadoop Core Concepts – Fault Tolerance

• If a cluster node fails, the master will detect that failure and re-assign the work to a different node on the system
• Restarting a task does not require communication with nodes working on other portions of the data
• If a failed node restarts, it is automatically added back to the system and assigned new tasks
• If a node appears to be running slowly, the master can redundantly execute another instance of the same task and use the results from the first task to finish
• Replicating all data at least 3 times allows work to be reassigned to another node and still work on the data locally; minimizing or eliminating network impact for failures.
Hadoop Ecosystem

The Hadoop Distributed File System (HDFS)
HDFS General Concepts

• **HDFS** is the file system responsible for storing data on the cluster
  – Written in Java (based on Google’s GFS)
  – Sits on top of a native file system (ext3, ext4, xfs, etc)
  – POSIX like file permissions model

• **Provides redundant storage for massive amounts of data**
  – Stores data in in very large files (hundreds of MB/GB/TB each)
  – Designed for extreme Fault Tolerance

• **HDFS is optimized for large, streaming reads of files**
  – Files in HDFS are generally ‘write once’, ‘read many’
  – Does not generally support random read/writes

• **HDFS performs best with a modest number of large files**
  – Millions, rather than billions, of files
  – Each file 100Mb or more
How Clients Write to HDFS

Writing files to HDFS

- Client consults Name Node
- Client writes block directly to one Data Node
- Data Node replicates block
- Cycle repeats for next block
How Files Are Read by Clients in HDFS

Client reading files from HDFS

- Client receives Data Node list for each block
- Client picks first Data Node for each block
- Client reads blocks sequentially

Tell me the block locations of Results.txt

Blk A = 1, 5, 6
Blk B = 8, 1, 2
Blk C = 5, 8, 9

metadata

Results.txt: Blk A: DN1, DN5, DN6
Blk B: DN7, DN1, DN2
Blk C: DN5, DN8, DN9
How Files Are Stored in HDFS

Individual files are split into blocks

- Each block is usually 64MB or 128MB
- If a file is smaller than a block the full 64MB/128MB will not be used
- Blocks are stored as standard files (xfs, etc)
- Blocks from a single file are distributed across the cluster
- Files can be compressed (multiple Codex available with varying speed/compression ratio tradeoffs)
- Each block is replicated to multiple nodes (3 nodes by default)

A cluster has 2 types of nodes that operate in a master-worker pattern

- The NameNode – manages the data in the cluster
- The DataNodes – stores and retrieves the data
The HDFS Name Node

• The NameNode runs the NameNode daemon that manages HDFS Namespace Metadata
  – What directories and files are on the cluster
  – What blocks comprise a given file
  – On which DataNodes the blocks for a given file are located
• Metadata held in RAM for fast access (need lots of RAM on this server)
• When a client application writes a file, the NameNode supplies a list of DataNodes to write the blocks to
• When a client reads a file, it communicates with the NameNode to determine which DataNodes the blocks reside on
• Listens for heartbeats from the DataNodes – if a node is dead it will be removed from the active node list
  – Issue commands to the DataNodes to re-replicate data so as to maintain 3 copies.
The HDFS Name Node - Persistence

• Only one NameNode in an HDFS cluster - single point of failure
  – If the NameNode is down, the entire HDFS file system in unavailable
  – If the name node metadata is lost then the entire cluster is unrecoverable
  – Usually run on more reliable hardware than the rest of the cluster

• HDFS Metadata is persisted to 2 files for crash recovery
  – Namespace image: stores file and directory information (fslist)
  – Edit log (tracks metadata updates made in RAM)
  – Metadata persistence files written both locally (RAID 1) and to a separate NFS mounted file system for recoverability.
  – Block locations are reconstructed from data nodes when the system starts
The HDFS Secondary Name Node

• Daemon that runs on a separate server than the NameNode
• Primary task is to periodically (1 hour by default) merge namespace image file with the edit file.
• Stores result of merge and copies it back to the NameNode.
• In the event of NameNode Server failure, the metadata files can be copied from NFS mount and the NameNode Daemon started here. This then becomes the new NameNode
• If the NFS file is lost, the merged file on Secondary Name Node can be used for recovery as well.
• The Secondary NameNode is not a backup NameNode! There is no automatic failover!
The HDFS Data Nodes

- Machines in the cluster that store and retrieve the file blocks
- Clients connect directly to the DataNodes to read/write data
- Blocks are stored in a set of directories specified in a set of Hadoop configuration files
- Blocks are distributed across Data Nodes when data is initially loaded into HDFS
- Data Processing (called MapReduce) is done on the data node using locally stored blocks
- Periodically sends heartbeat (3 seconds by default) to the NameNode to confirm availability
- Every 10th message is a block report that tells the NameNode which blocks it is storing
The HDFS Data Nodes – Data Integrity

• DataNodes implement replication on file creation by forwarding blocks to the other DataNodes after writing them locally
• Default Replication level is 3 (configurable)
• Block checksums are created by the DataNodes when files are written
• Clients reading/writing the data will verify success by checking the checksum
• The DataNode also runs a DataBlockScanner daemon that will detect corrupted blocks and “heal” them by using one of the replicas
• Data generally not replicated/backed-up outside of the cluster – although redundant clusters can be used with data loaded into both.
How Files Are Accessed in HDFS

• Applications read and write HDFS files directly via the Java API
• Access to HDFS from the command line is achieved with the hadoop fs command
• Typically, files are created on a local file system and then must be moved into HDFS
• Likewise, files stored in HDFS may need to be moved to a machine’s local file system
Oracle’s recommended Big Data approach

1. Common platform to manage all your structured and unstructured data
   - Easily load your map-reduce output into your DWH
   - Infiniband connect your Big Data with your structured data systems
   - Leverage the same Oracle skills to manage all your data

2. Engineered system to save time and money
   - Preinstalled and preconfigured for large scale Big Data management
   - Pre loaded Cloudera Distribution of Apache Hadoop, Oracle NoSQL and statistical environment R

3. Get one enterprise-level support for all your platform
   - Oracle provides support for all your HW
   - Oracle provides support for all your Oracle SW and the Open Source SW (CDH, R),…
18 Sun X4270 M2 servers
- 864 GB main memory
- 216 cores
- 648 TB Storage

Software
- Oracle Linux
- Java Hotspot VM
- Cloudera CDH
- Cloudera Manager
- Open Source R Distribution
- Oracle NoSQL Database CE (or EE*)

Big Data Connectors
- Oracle Loader for Hadoop
- Oracle Data Integrator Application Adapter for Hadoop
- Oracle R for Hadoop Connector
- Oracle Direct Connector for HDFS

* Stand-alone software, can be pre-installed and pre-configured with BDA
Cloudera Management Console

- Reduces the risks of running Hadoop in production
- Improves consistency, compliance and administrative overhead
- Production support for CDH and certified integrations (MySql, Postgres, Oracle)

Cloudera Manager

- Authorization Manager
- Activity Monitor
- Service Monitor
- Resource Manager
- Service & Configuration Manager
Cloudera Management Console

- Identify system issues and performance shortfalls by analyzing task distribution
- Trace issues down to individual tasks and logs
- Centralized administration of all system permissions
- Automated generation of configuration settings with built-in validations to prevent errors or downtime
- Drill down capability to the detailed status of individual nodes
- Automatically tracks services on hosts and their resource requirements
- Add new service instances with a few clicks
Cloudera Management Console

Central Point of Management For All The Resources In The Hadoop Stack
Cloudera Management Console

Service Dashboards Give Consolidated Status
Visualization at the Speed of Thought
Oracle Exalytics

40 Intel processor cores
1 TB main memory
40 Gb InfiniBand connection to Oracle Exadata

Oracle TimesTen In-Memory Database
- Adaptive in-memory caching of analytics
- In-memory columnar compression

Oracle Business Intelligence Foundation Suite
Oracle Endeca Information Discovery
Understand the complete picture with context from any source

BI Star Schemas
Metrics: Margin = Sales - Cost, …
Dimensions: Customer, Product, …

Transactions
<table>
<thead>
<tr>
<th>TxnID</th>
<th>ProductID</th>
<th>Category</th>
<th>Amount</th>
</tr>
</thead>
<tbody>
<tr>
<td>12324</td>
<td>506</td>
<td>Mountain Bike</td>
<td>499</td>
</tr>
<tr>
<td>12325</td>
<td>507</td>
<td>Road Bike</td>
<td>1399</td>
</tr>
</tbody>
</table>

“Why did sales fall?
Was it bad reviews? An unsuccessful marketing campaign? Quality issues?”

Enterprise Applications
Unstructured Text Fields
“Customer reported receiving wheel with bent spokes…”

Enterprise Content System
MS Office Documents, Internal Notes
Marketing Strategy Document
Chapter 1. …

Websites
Product Reviews
“The new 506 from Acme is a solid road bike. 3.5 stars.”

Social Media
Customer Comments
“Thinking about buying a new road bike. Anyone know anything about the Acme 506?”

Oracle Endeca Information Discovery
Understand the complete picture with context from any source

BI Star Schemas
Metrics: Margin = Sales - Cost, …
Dimensions: Customer, Product, …

Transactions
<table>
<thead>
<tr>
<th>TxnID</th>
<th>ProductID</th>
<th>Category</th>
<th>Amount</th>
</tr>
</thead>
<tbody>
<tr>
<td>12324</td>
<td>506</td>
<td>Mountain Bike</td>
<td>499</td>
</tr>
<tr>
<td>12325</td>
<td>507</td>
<td>Road Bike</td>
<td>1399</td>
</tr>
</tbody>
</table>

“Why did sales fall?
Was it bad reviews? An unsuccessful marketing campaign? Quality issues?”

Enterprise Applications
Unstructured Text Fields
“Customer reported receiving wheel with bent spokes…”

Enterprise Content System
MS Office Documents, Internal Notes
Marketing Strategy Document
Chapter 1. …

Websites
Product Reviews
“The new 506 from Acme is a solid road bike. 3.5 stars.”

Social Media
Customer Comments
“Thinking about buying a new road bike. Anyone know anything about the Acme 506?”
Unstructured Analytics

Benefits
- Unprecedented Information Visibility
- Leverage Existing BI Investments
- Self-Service Data Discovery
- Reduced IT Costs, Better Business Decisions

Unique Features
- Contextual Search, Navigation, Analytics
- Dynamic Data and Metadata
- Content Acquisition and Text Enrichment
- In-Memory Performance
Benefits of Oracle Big Data

• Integrate new data types into your existing business intelligence infrastructure
  – Start managing all your unstructured data: sensor, social network, feeds, logs, call center,…

• Take action using all available data sources
  – Leverage insight from your structured and unstructured data

• Save time and money with engineered systems
  – Avoid putting together the HW and SW pieces yourself

• Get one point of support for all your business intelligence components, including Big Data
  – One support for all your Big Data HW and SW components
Hardware and Software

Engineered to Work Together