The following is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, and timing of any features or functionality described for Oracle’s products remains at the sole discretion of Oracle.
Program Agenda

- Oracle Big Data Connectors Overview
- Oracle Loader for Hadoop
- Oracle SQL Connector for HDFS
- Performance Tuning
- Summary
Oracle Big Data Solution

Stream
- Oracle Event Processing
- Apache Flume
- Oracle GoldenGate

Acquire – Organize – Analyze
- Cloudera Hadoop
- Oracle NoSQL Database
- Oracle R Distribution
- Oracle Big Data Connectors
- Oracle Data Integrator

Decide
- Oracle Real-Time Decisions
- Endeca Information Discovery
- Oracle BI Foundation Suite

Oracle Database
Oracle Advanced Analytics
Oracle Spatial & Graph
Oracle Big Data Connectors

Connecting Hadoop to Oracle Database

Acquire – Organize – Analyze
Oracle Big Data Connectors
Connecting Hadoop to Oracle Database

Batch oriented
- Transform input data
- Schema on read
- Unstructured data, less useful after relevant data is extracted
- Write once, read many times

Real-Time
- Fast access to a specific record
- Schema on write
- High availability, reliability, security
- Read, write, delete update

Hadoop
Database

Oracle Big Data Connectors
Oracle Big Data Connectors
Licensed Together

- Oracle SQL Connector for HDFS
- Oracle Loader for Hadoop
- Oracle R Connector for Hadoop
- Oracle Data Integrator Application Adapters for Hadoop
- **Announcing at OOW 2013**: Oracle XQuery for Hadoop
Oracle Loader for Hadoop and Oracle Direct Connector for HDFS

- High speed load from Hadoop to Oracle Database
- Access data on HDFS from Oracle Database
- Aggregate data from both Hadoop and Oracle Database

Load speed: 15 TB/hour
Oracle R Connector for Hadoop
R Analytics leveraging Hadoop and HDFS

5x faster in BDC 3.0

Linearly Scale a Robust Set of R Algorithms
Leverage MapReduce for R Calculations
Compute Intensive Parallelism for Simulations
Oracle R Connector for Hadoop

R Analytics leveraging Hadoop and HDFS

Oracle R Client

Hadoop

Load

Oracle Loader for Hadoop

5x faster in BDC 3.0
Oracle Data Integrator Application Adapters for Hadoop

Benefits
- Consistent tooling across BI/DW, SOA, Integration and Big Data
- Reduce complexities of processing Hadoop through graphical tooling
- Improves productivity when processing Big Data (Structured + Unstructured)

Improving Productivity and Efficiency for Big Data
Announcing: Oracle XQuery for Hadoop (OXH)

- OXH is a transformation engine for Big Data
- XQuery language executed on the Map/Reduce framework
Oracle Loader for Hadoop
Oracle SQL Connector for HDFS
High speed load from Hadoop to Oracle Database
Load Data into the Database

Two Options

- **Oracle Loader for Hadoop**
  - Map Reduce job transforms data on Hadoop into Oracle-ready data types
  - Use more Hadoop compute resources

- **Oracle SQL Connector for HDFS**
  - Oracle SQL access to data on Hadoop via external tables
  - Use more database compute resources
  - Includes option to query in-place
Performance

• 15 TB / HOUR

• 25 TIMES FASTER THAN THIRD PARTY PRODUCTS

• REDUCED DATABASE CPU USAGE IN COMPARISON
<table>
<thead>
<tr>
<th><strong>Use Case</strong></th>
<th>Oracle Loader for Hadoop</th>
<th>Oracle SQL Connector for HDFS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Continuous or frequent load into production database, requiring reduced use of database CPU resources</td>
<td>Bulk load of large volumes of data</td>
<td>Uses more database CPU resources</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>Input Data Formats</strong></th>
<th>Oracle Loader for Hadoop</th>
<th>Oracle SQL Connector for HDFS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Load various types of input data: HBase, JSON files, Weblogs, sequence files, custom formats, etc.</td>
<td>Load text (HDFS files, and Hive table files)</td>
<td>Load Oracle Data Pump files: Generated by Oracle Loader for Hadoop from HBase, JSON files, Weblogs, sequence files, custom formats, etc.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>Functionality</strong></th>
<th>Oracle Loader for Hadoop</th>
<th>Oracle SQL Connector for HDFS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Load</td>
<td>Load and also query in place (Note: Query requires full table scans since data files are external to the database)</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>Performance</strong></th>
<th>Oracle Loader for Hadoop</th>
<th>Oracle SQL Connector for HDFS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Uses more time on Hadoop for pre-processing data.</td>
<td>End-to-end time is faster because no time is spent processing on Hadoop. Trade-off is more database CPU resources are used.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>Usability</strong></th>
<th>Oracle Loader for Hadoop</th>
<th>Oracle SQL Connector for HDFS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Likely to be preferred by Hadoop developers</td>
<td>Likely to be preferred by Oracle developers</td>
<td></td>
</tr>
</tbody>
</table>
Oracle Loader for Hadoop

Partition, sort, and convert into Oracle data types on Hadoop

SHUFFLE/SORT

Connect to the database from reducer nodes, load into database partitions in parallel

MAP

REDUCE

MAP

REDUCE

MAP

REDUCE

MAP

REDUCE

Features

Offloads data pre-processing from the database server to Hadoop

Works with a range of input data formats

Automatic balancing in case of skew in input data

Online and offline modes
<table>
<thead>
<tr>
<th>Cust ID</th>
<th>Movie ID</th>
<th>Genre ID</th>
<th>Time</th>
<th>Recommended</th>
<th>Activity</th>
<th>Price</th>
</tr>
</thead>
<tbody>
<tr>
<td>1046915</td>
<td>null</td>
<td>null</td>
<td>2012-07-01:00:33:18</td>
<td>null</td>
<td>9</td>
<td>null</td>
</tr>
<tr>
<td>1144051</td>
<td>768</td>
<td>9</td>
<td>2012-07-01:00:33:39</td>
<td>N</td>
<td>6</td>
<td>null</td>
</tr>
<tr>
<td>1264225</td>
<td>null</td>
<td>null</td>
<td>2012-07-01:00:34:01</td>
<td>null</td>
<td>8</td>
<td>null</td>
</tr>
<tr>
<td>1088645</td>
<td>null</td>
<td>null</td>
<td>2012-07-01:00:34:18</td>
<td>null</td>
<td>8</td>
<td>null</td>
</tr>
<tr>
<td>1088645</td>
<td>null</td>
<td>null</td>
<td>2012-07-01:00:34:28</td>
<td>null</td>
<td>8</td>
<td>null</td>
</tr>
<tr>
<td>1363545</td>
<td>27205</td>
<td>9</td>
<td>2012-07-01:00:35:09</td>
<td>Y</td>
<td>11</td>
<td>3.99</td>
</tr>
<tr>
<td>1156900</td>
<td>20352</td>
<td>14</td>
<td>2012-07-01:00:35:12</td>
<td>N</td>
<td>7</td>
<td>null</td>
</tr>
<tr>
<td>1336404</td>
<td>null</td>
<td>null</td>
<td>2012-07-01:00:35:27</td>
<td>null</td>
<td>9</td>
<td>null</td>
</tr>
<tr>
<td>1022288</td>
<td>null</td>
<td>null</td>
<td>2012-07-01:00:35:38</td>
<td>null</td>
<td>8</td>
<td>null</td>
</tr>
<tr>
<td>1129727</td>
<td>1105903</td>
<td>11</td>
<td>2012-07-01:00:36:08</td>
<td>N</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>1305981</td>
<td>null</td>
<td>null</td>
<td>2012-07-01:00:36:27</td>
<td>null</td>
<td>8</td>
<td>null</td>
</tr>
</tbody>
</table>

---

Apache Weblogs

JSON files

Sensor data

Machine logs

Twitter feeds
1. Load by Reading Data through Input Format Interface

- Parse weblogs with regular expressions
- Hive tables
- Oracle NoSQL Database
- User written input format
- ORACLE LOADER FOR HADOOP
- REDUCE
- REDUCE
- REDUCE
- REDUCE
- ORACLE Data Warehouse
Load Weblog Data
Using Regular Expression Input Format

Raw Weblog Data

User id | Session id | Session start time | Session End time
---|---|---|---
2471626 | 76.185.60.162:2471647:ts1 | 6:00:06 | 6:13:29
...
...
...

Filtered, Structured Data

Oracle Loader for Hadoop with Regular Expression input format
Weblogs transformed on Hadoop
Submitting an Oracle Loader for Hadoop Job

MyConf.xml

InputFormat:

  <property>
  mapreduce.inputformat.class
  </property>
  <value>RegExInputFormat</value>

Database connection information

Target table name/schema

...
User-written Input Format Implementation

- Oracle Loader for Hadoop reads Avro IndexedRecords
- Input format implementation should read input data records and put into an Avro IndexedRecord
- Sample input format implementation shipped with Oracle Loader for Hadoop kit
- Users can implement input formats for HBase, nosql data stores, custom formats, etc.
2. Use OLH’s Connectivity to Hadoop Technologies

- JSON files
- Hive’s HBase Storage Handler
- JSON SerDe
- ORACLE LOADER FOR HADOOP
- Hive external tables

Oracle Data Warehouse
Process JSON Files and Load into Database

| INTERACTION_ID | DEMOGRAPHIC_GENDER | KLOUT_SCORE | ... | LANGUAGE_TAG | ...
|----------------|--------------------|-------------|-----|--------------|-----|

Filtered, Structured Data

Use JSON Serde to read into a Hive table

Load Hive table using Oracle Loader for Hadoop

JSON Twitter data

Hadoop
Use JSON SerDe to Access Through Hive

CREATE EXTERNAL TABLE tweets
    (......
    )
ROW FORMAT SERDE 'com.cloudera.serde.JSONSerDe'
STORED AS TEXTFILE
LOCATION '/user/oracle/tweets'

- Load data from Hive tables using Oracle Loader for Hadoop
Load Data from HBase

Use Hive HBase Storage Handler or Input Format Implementation

Hive's HBase Storage Handler

Input format to read HBase records into Avro IndexedRecords

Hive external table, will read data using Hive HBase Storage Handler

Load directly with HBase to Avro Input Format implementation
Install of Oracle Loader for Hadoop
Key Benefits

- Load from a wide range of input sources
- Performance
  - 10x faster than comparable third party products
- Offload database server processing on to Hadoop
  - Reduced impact on database during load
- Easy to use
- Developed and supported by Oracle
Oracle SQL Connector for HDFS

Use Oracle SQL to Load or Access Data on HDFS

- Load into the database using SQL
- Option to access and analyze data in place on HDFS
- Access Hive (internal and external) tables and HDFS files
- Automatic load balancing to maximize performance

Access or load into the database in parallel using external table mechanism
Hadoop Cluster
Install of Oracle SQL Connector for HDFS
Oracle SQL Connector for HDFS

- Load data from external table with Oracle SQL
  - `INSERT INTO <tablename> AS SELECT * FROM <external tablename>`

- Access data in-place on HDFS with Oracle SQL
  - Note: No indexes, no partitioning, so queries are a full table scan

- Data files are read in parallel
  - Ex: If there are 96 data files and the database can support 96 PQ slaves, all 96 files can be read in parallel
  - OSCH automatically balances the load across the PQ slaves
Oracle SQL Connector for HDFS

- Generates definition and creates external table pointing to data files on HDFS
- When external table is accessed with SQL, data is streamed from HDFS

```sql
CREATE TABLE "TWEET"."HIVE_ORA_EXT_TAB"
(
    "INTERACTION_ID" VARCHAR2(4000),
    "DEMOGRAPHIC_GENDER" VARCHAR2(4000),
    "KLOUT_SCORE" INTEGER,
    "KLOUT_AMPLIFICATION" INTEGER,
    "KLOUT_NETWORK" VARCHAR2(4000),
    "KLOUT_TRUE_REACH" VARCHAR2(4000),
    "LANGUAGE_TAG" VARCHAR2(4000),
    "LANGUAGE_CONFIDENCE" INTEGER,
    "SALIENCE_CONTENT_SENTIMENT" INTEGER,
    "DT" VARCHAR2(4000)
)
```

**Location Files contain URIs:**
hdfs://.../user/hive/warehouse/dw_augmentation/000000_0
... ... ...
Oracle SQL Connector for HDFS

Input Data Formats

- Text files

- Hive tables
  - Internal and external tables
  - Text data
  - Oracle external table data types map to Hive table data types

- Oracle Data Pump files generated by Oracle Loader for Hadoop
Oracle SQL Connector for HDFS

Data Pump Files

- Oracle Data Pump: Binary format data file

- Oracle Loader for Hadoop generates Oracle Data Pump files for use by Oracle SQL Connector for HDFS

- Load of Oracle Data Pump files is more efficient – uses about 50% less database CPU
  - Hadoop does more of the work, transforming text data into binary data optimized for Oracle
Key Benefits

- Extremely fast load performance
  - 15 TB/hour from Oracle Big Data Appliance to Oracle Exadata

- Load data pump files for reduced database CPU usage

- Unique option to query HDFS data in-place

- Easy to use for Oracle DBAs and Hadoop developers
- Developed and supported by Oracle
Certification with Hadoop Versions

- Certified by Oracle
  - CDH 4.3, CDH 3
  - Apache Hadoop 1.0, 1.1

- Intel announces certification of their distribution at OOW

- Process for third party vendors to certify their distributions
Media Company
Real-time Analytics with 360 Customer View

Objectives
- Find the hidden value in large volumes of online and social media behavior, merged with data in transactional systems

Solution
- Starter rack BDA with connectors for integration of all data for full customer view
- Partner NGData’s Lily platform
- Cost-effective storage on the BDA
- Real-time analytics of all data

Benefits
- Analysis in real-time instead with a two week lag
- Lower TCO and fast time to value
- BDA, connectors, database: integrated single system for all data for a simplified IT environment
Financial Services
Risk and Finance

Objectives
 Aggregate structured and unstructured data from multiple sources
 Scale to increasing volumes of data
 Consolidate existing silos of information for unified access to enterprise data

Solution
 Oracle Big Data Appliance, Oracle Exadata, Oracle Big Data Connectors, Oracle Enterprise Management for comprehensive technology solution
 Oracle Data Integrator for end-to-end data lineage
 Supports data and analytics for risk and finance

Benefits
 Fast and nimble way to get new data into ODS
 Deliver better SLAs to users
 Simplified architecture
 Additional storage and compute resources available for new development projects

Aggregate all data
Some New Features in BDC 2.3

- Performance moves from 12 TB to 15 TB

- Ease of use and flexibility
  - Easier way to map columns to be loaded to target table columns
  - Per column override while mapping Hive column types to external table column types

- Works out-of-the-box with Kerberos authentication protocol
Performance Tuning
Performance Tuning

- Parallelism
- Network bandwidth
- Hadoop property values
- Database target table definition, tablespace parameters, session settings
- Using the sampler (for Oracle Loader for Hadoop)
Key: Degree of Parallelism
Key: Degree of Parallelism

Number of reducer slots

OSCH: Number of location files

OLH: Number of partitions in target table

Number of CPU cores
Parallelism

- Oracle Loader for Hadoop
  - Reduce tasks load data in parallel to the database
  - Goal: Number of partitions in the database should be a multiple of number of reduce tasks

- Oracle SQL Connector for HDFS
  - SQL> alter session enable parallel query (or DML);
  - Number of location files in external table should be a multiple of DOP (which is determined by number of database cores)
Network Bandwidth

- Configure InfiniBand
  - Read and follow Oracle BDA Documentation

- Multi-homing for Hadoop
  - Hadoop needed to support multiple network interfaces to maximize use of InfiniBand bandwidth
  - Enabled by collaboration between Oracle and Cloudera, committed to Apache Hadoop by Cloudera

- For Oracle Loader for Hadoop, configure SDP
Hadoop Property Values

- Batch size: Number of records to be inserted in batch into the database
- Buffer size for direct path load
- Specifying when reduce tasks begin
- Reusing JVM
- …
Database Parameters

- Session parameters
  - Enable parallel query and DML
    - `SQL> alter session enable parallel query (or DML);`
- Table definition
  - For maximum throughput: NOLOGGING, PARALLEL, NOCOMPRESS
- Tablespace
  - Use ASM
Sampler for Oracle Loader for Hadoop

- Distributes load evenly among reducer tasks
  - Reduces slow down due to data skew

- Enable Sampling (by config parameter) for this automatic load balancing
Summary

- Oracle Loader for Hadoop and Oracle SQL Connector for HDFS are products for high speed loading from Hadoop to Oracle Database
  - Cover a range of use cases
  - Several input sources
  - Flexible, easy-to-use, developed and supported by Oracle

- The fastest load option loads at 15 TB/hour
Hardware and Software
Engineered to Work Together