ORACLE
BIG DATA

Oracle Big Data SQL

Day with Development

Alexey Filanovskiy - DWGL
Marty Gubar - Big Data PM

ORACLE

Agenda

- Introduction + Quick Demo
- Big Data SQL: Technical Overview
- Get Started! Fingers on Keyboards
  - Creating Big Data SQL-enabled Tables
  - Performance Features
  - Security
  - NoSQL Access
- Questions any time

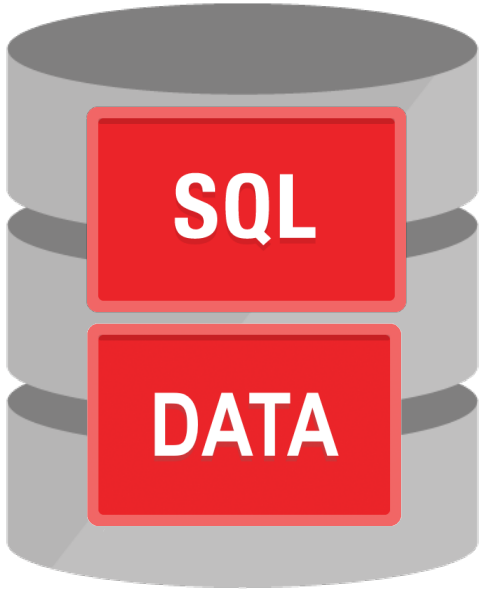
The New Normal at Oracle Customers

Expansion of Data Management Components

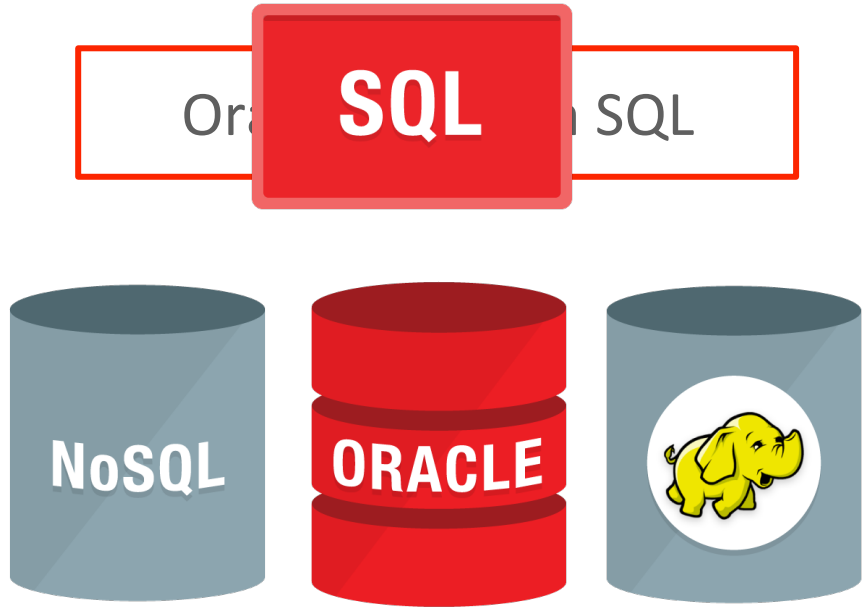
Expansion of Programming Environments

Database Strategy for Big Data

Conventional view of Data Management

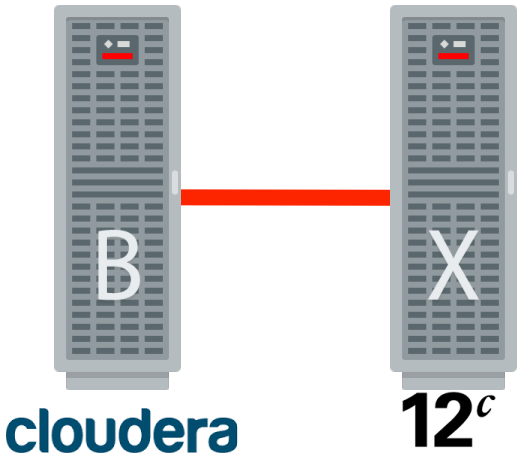


Emerging view of Data Management

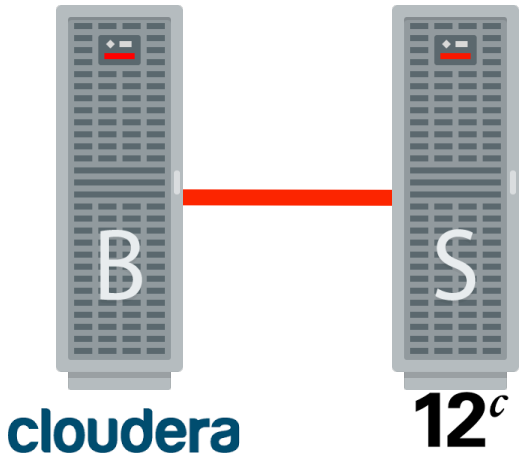


Big Data SQL Configurations

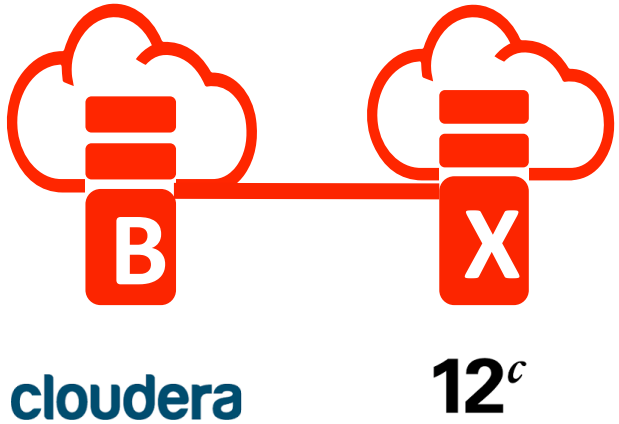
Engineered Systems



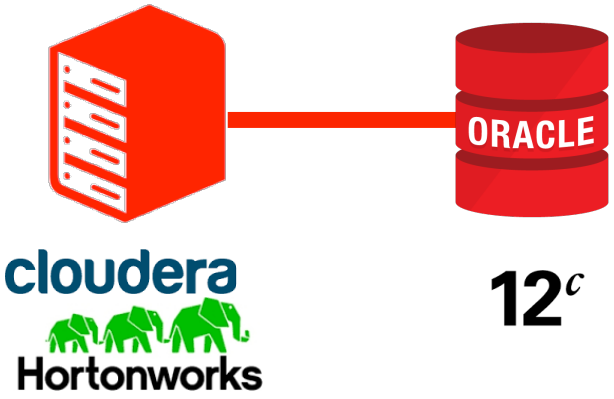
Engineered Systems



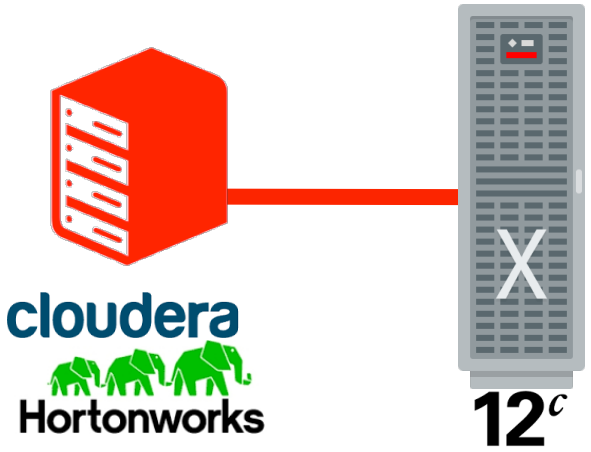
Oracle Cloud*



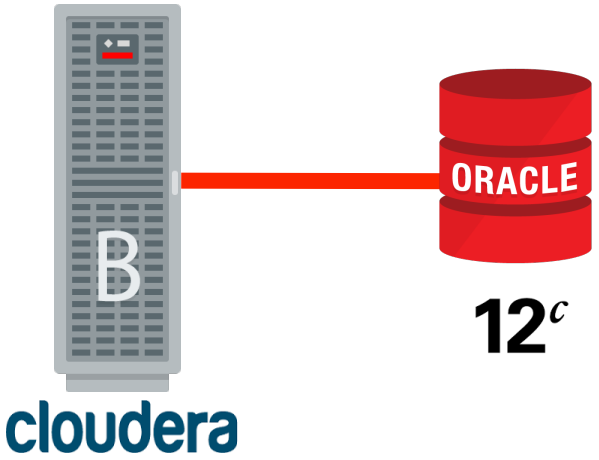
Commodity Servers



Mixed Deployment



Mixed Deployment



*** Coming Soon!**

Oracle Big Data Management System

anonymous Logout

- Access
- Secure
- Analyze

Access Any Data

Personalized recommendations (NoSQL)

Clicks, ratings & comments (JSON on Hadoop)

Revenue from purchases (Oracle Relational Data Warehouse)

Example: Click data in its raw format.

Create Big Data SQL Enabled Tables

Query Complex Sources

release 1.0 Set Screen Reader Mode On

File Browser

ACTIONS

View as binary

Download

View file location

Refresh

INFO

Last modified
Dec. 30, 2014 8:25 a.m.
User
oracle
Group
oracle
Size
31.0 MB
Mode
100644

Home

Page 1 to 1 of 7949

/ user / oracle / moviework / applog_json / **movieapp_log_json.log**

{"custid":1185972,"movieid":null,"genreid":null,"time":"2012-07-01:00:00:07","recommended":null,"activity":8}
{"custid":1354924,"movieid":1948,"genreid":9,"time":"2012-07-01:00:00:22","recommended":"N","activity":7}
{"custid":1083711,"movieid":null,"genreid":null,"time":"2012-07-01:00:00:26","recommended":null,"activity":9}
{"custid":1234182,"movieid":11547,"genreid":6,"time":"2012-07-01:00:00:32","recommended":"Y","activity":7}
{"custid":1010220,"movieid":11547,"genreid":6,"time":"2012-07-01:00:00:42","recommended":"Y","activity":6}
{"custid":1143971,"movieid":null,"genreid":null,"time":"2012-07-01:00:00:43","recommended":null,"activity":8}
{"custid":1253676,"movieid":null,"genreid":null,"time":"2012-07-01:00:00:50","recommended":null,"activity":9}
{"custid":1351777,"movieid":608,"genreid":6,"time":"2012-07-01:00:01:03","recommended":"N","activity":7}
{"custid":1143971,"movieid":null,"genreid":null,"time":"2012-07-01:00:01:07","recommended":null,"activity":9}
{"custid":1363545,"movieid":27205,"genreid":9,"time":"2012-07-01:00:01:18","recommended":"Y","activity":7}
{"custid":1067283,"movieid":1124,"genreid":9,"time":"2012-07-01:00:01:26","recommended":"Y","activity":7}
{"custid":1126174,"movieid":16309,"genreid":46,"time":"2012-07-01:00:01:35","recommended":"N","activity":7}
{"custid":1234182,"movieid":11547,"genreid":6,"time":"2012-07-01:00:01:39","recommended":"Y","activity":7}
{"custid":1067283,"movieid":null,"genreid":null,"time":"2012-07-01:00:01:55","recommended":null,"activity":9}
{"custid":1377537,"movieid":null,"genreid":null,"time":"2012-07-01:00:01:58","recommended":null,"activity":9}
{"custid":1347836,"movieid":null,"genreid":null,"time":"2012-07-01:00:02:03","recommended":null,"activity":8}
{"custid":1137285,"movieid":null,"genreid":null,"time":"2012-07-01:00:03:39","recommended":null,"activity":8}
{"custid":1354924,"movieid":null,"genreid":null,"time":"2012-07-01:00:03:51","recommended":null,"activity":9}
{"custid":1036191,"movieid":null,"genreid":null,"time":"2012-07-01:00:03:55","recommended":null,"activity":8}
{"custid":1363545,"movieid":27205,"genreid":9,"time":"2012-07-01:00:04:03","recommended":"Y","activity":5}
{"custid":1273464,"movieid":null,"genreid":null,"time":"2012-07-01:00:04:39","recommended":null,"activity":9}
{"custid":1346299,"movieid":424,"genreid":18,"time":"2012-07-01:00:05:02","recommended":"Y","activity":4}
{"custid":1399170,"movieid":null,"genreid":null,"time":"2012-07-01:00:05:34","recommended":null,"activity":8}

File Browser

ACTIONS

- View as binary
- Download
- View file location
- Refresh

INFO

Last modified
Dec. 30, 2014 8:25 a.m.
User
oracle
Group
oracle
Size
31.0 MB
Mode
100644

Home

Page 1 to 1 of 7949

/ user / oracle / moviework / applog_json / **movieapp_log_json.log**


```
{"custid":1185972,"movieid":null,"genreid":null,"time":"2012-07-01:00:00:07","recommended":null,"activity":8}
{"custid":1354924,"movieid":1948,"genreid":9,"time":"2012-07-01:00:00:22","recommended":"N","activity":7}
{"custid":1083711,"movieid":null,"genreid":null,"time":"2012-07-01:00:00:26","recommended":null,"activity":9}
{"custid":1234182,"movieid":11547,"genreid":6,"time":"2012-07-01:00:00:32","recommended":"Y","activity":7}
{"custid":1010220,"movieid":11547,"genreid":6,"time":"2012-07-01:00:00:42","recommended":"Y","activity":6}
{"custid":1143971,"movieid":null,"genreid":null,"time":"2012-07-01:00:00:43","recommended":null,"activity":8}
{"custid":1253676,"movieid":null,"genreid":null,"time":"2012-07-01:00:00:50","recommended":null,"activity":9}
{"custid":1351777,"movieid":608,"genreid":6,"time":"2012-07-01:00:01:03","recommended":"N","activity":7}
{"custid":1143971,"movieid":null,"genreid":null,"time":"2012-07-01:00:01:07","recommended":null,"activity":9}
{"custid":1363545,"movieid":27205,"genreid":9,"time":"2012-07-01:00:01:18","recommended":"Y","activity":7}
{"custid":1067283,"movieid":1124,"genreid":9,"time":"2012-07-01:00:01:26","recommended":"Y","activity":7}
{"custid":1126174,"movieid":16309,"genreid":46,"time":"2012-07-01:00:01:35","recommended":"N","activity":7}
{"custid":1234182,"movieid":11547,"genreid":6,"time":"2012-07-01:00:01:39","recommended":"Y","activity":7}
{"custid":1067283,"movieid":null,"genreid":null,"time":"2012-07-01:00:01:55","recommended":null,"activity":9}
{"custid":1377537,"movieid":null,"genreid":null,"time":"2012-07-01:00:01:58","recommended":null,"activity":9}
{"custid":1347836,"movieid":null,"genreid":null,"time":"2012-07-01:00:02:03","recommended":null,"activity":8}
{"custid":1137285,"movieid":null,"genreid":null,"time":"2012-07-01:00:03:39","recommended":null,"activity":8}
{"custid":1354924,"movieid":null,"genreid":null,"time":"2012-07-01:00:03:51","recommended":null,"activity":9}
{"custid":1036191,"movieid":null,"genreid":null,"time":"2012-07-01:00:03:55","recommended":null,"activity":8}
{"custid":1363545,"movieid":27205,"genreid":9,"time":"2012-07-01:00:04:03","recommended":"Y","activity":5}
{"custid":1273464,"movieid":null,"genreid":null,"time":"2012-07-01:00:04:39","recommended":null,"activity":9}
{"custid":1346299,"movieid":424,"genreid":18,"time":"2012-07-01:00:05:02","recommended":"Y","activity":4}
{"custid":1399170,"movieid":null,"genreid":null,"time":"2012-07-01:00:05:34","recommended":null,"activity":8}
```

ACTIONS

View as binary

Download

View file location

Refresh

INFO

Last modified
Dec. 30, 2014 8:25 a.m.
User
oracle
Group
oracle
Size
31.0 MB
Mode
100644

Home

Page 1 to 1 of 7949

/ user / oracle / moviework / applog_json / **movieapp_log_json.log**


```
{"custid":1185972,"movieid":null,"genreid":null,"time":"2012-07-01:00:00:07","recommended":null,"activity":8}
{"custid":1354924,"movieid":1948,"genreid":9,"time":"2012-07-01:00:00:22","recommended":"N","activity":7}
{"custid":1083711,"movieid":null,"genreid":null,"time":"2012-07-01:00:00:26","recommended":null,"activity":9}
{"custid":1234182,"movieid":11547,"genreid":6,"time":"2012-07-01:00:00:32","recommended":"Y","activity":7}
{"custid":1010220,"movieid":11547,"genreid":6,"time":"2012-07-01:00:00:42","recommended":"Y","activity":6}
{"custid":1143971,"movieid":null,"genreid":null,"time":"2012-07-01:00:00:43","recommended":null,"activity":8}
{"custid":1253676,"movieid":null,"genreid":null,"time":"2012-07-01:00:00:50","recommended":null,"activity":9}
{"custid":1351777,"movieid":608,"genreid":6,"time":"2012-07-01:00:01:03","recommended":"N","activity":7}
{"custid":1143971,"movieid":null,"genreid":null,"time":"2012-07-01:00:01:07","recommended":null,"activity":9}
{"custid":1363545,"movieid":27205,"genreid":9,"time":"2012-07-01:00:01:18","recommended":"Y","activity":7}
{"custid":1067283,"movieid":1124,"genreid":9,"time":"2012-07-01:00:01:26","recommended":"Y","activity":7}
{"custid":1126174,"movieid":16309,"genreid":46,"time":"2012-07-01:00:01:35","recommended":"N","activity":7}
{"custid":1234182,"movieid":11547,"genreid":6,"time":"2012-07-01:00:01:39","recommended":"Y","activity":7}
{"custid":1067283,"movieid":null,"genreid":null,"time":"2012-07-01:00:01:55","recommended":null,"activity":9}
{"custid":1377537,"movieid":null,"genreid":null,"time":"2012-07-01:00:01:58","recommended":null,"activity":9}
{"custid":1347836,"movieid":null,"genreid":null,"time":"2012-07-01:00:02:03","recommended":null,"activity":8}
{"custid":1137285,"movieid":null,"genreid":null,"time":"2012-07-01:00:03:39","recommended":null,"activity":8}
{"custid":1354924,"movieid":null,"genreid":null,"time":"2012-07-01:00:03:51","recommended":null,"activity":9}
{"custid":1036191,"movieid":null,"genreid":null,"time":"2012-07-01:00:03:55","recommended":null,"activity":8}
{"custid":1363545,"movieid":27205,"genreid":9,"time":"2012-07-01:00:04:03","recommended":"Y","activity":5}
{"custid":1273464,"movieid":null,"genreid":null,"time":"2012-07-01:00:04:39","recommended":null,"activity":9}
{"custid":1346299,"movieid":424,"genreid":18,"time":"2012-07-01:00:05:02","recommended":"Y","activity":4}
{"custid":1399170,"movieid":null,"genreid":null,"time":"2012-07-01:00:05:34","recommended":null,"activity":8}
```

Access | Hue – File Browser – movie | http://denaa246:7070/orc

Marty

orabda:7777/apex/f?p=106:8::::::

Apps | BIEE | OOW | CS50.tv | Big Data | SecureCluster | Big_Data_Use_Cases | Big Data Science Boo | Configuration Tasks | EM | Save to Instapaper | scaj51 | News | Technology Launchp | Other Bookmarks

- Secure
- Analyze

Personalized recommendations (NoSQL)

Clicks, ratings & comments (JSON on Hadoop)

Revenue from purchases (Oracle Relational Data Warehouse)

Example: Click data in its raw format.

- Create Big Data SQL Enabled Tables
- Query Complex Sources

release 1.0 Set Screen Reader Mode On

Oracle Big Data Management System

anonymous Logout

- Access
- Secure
- Analyze

Access Any Data

Personalized recommendations (NoSQL)

Clicks, ratings & comments (JSON on Hadoop)

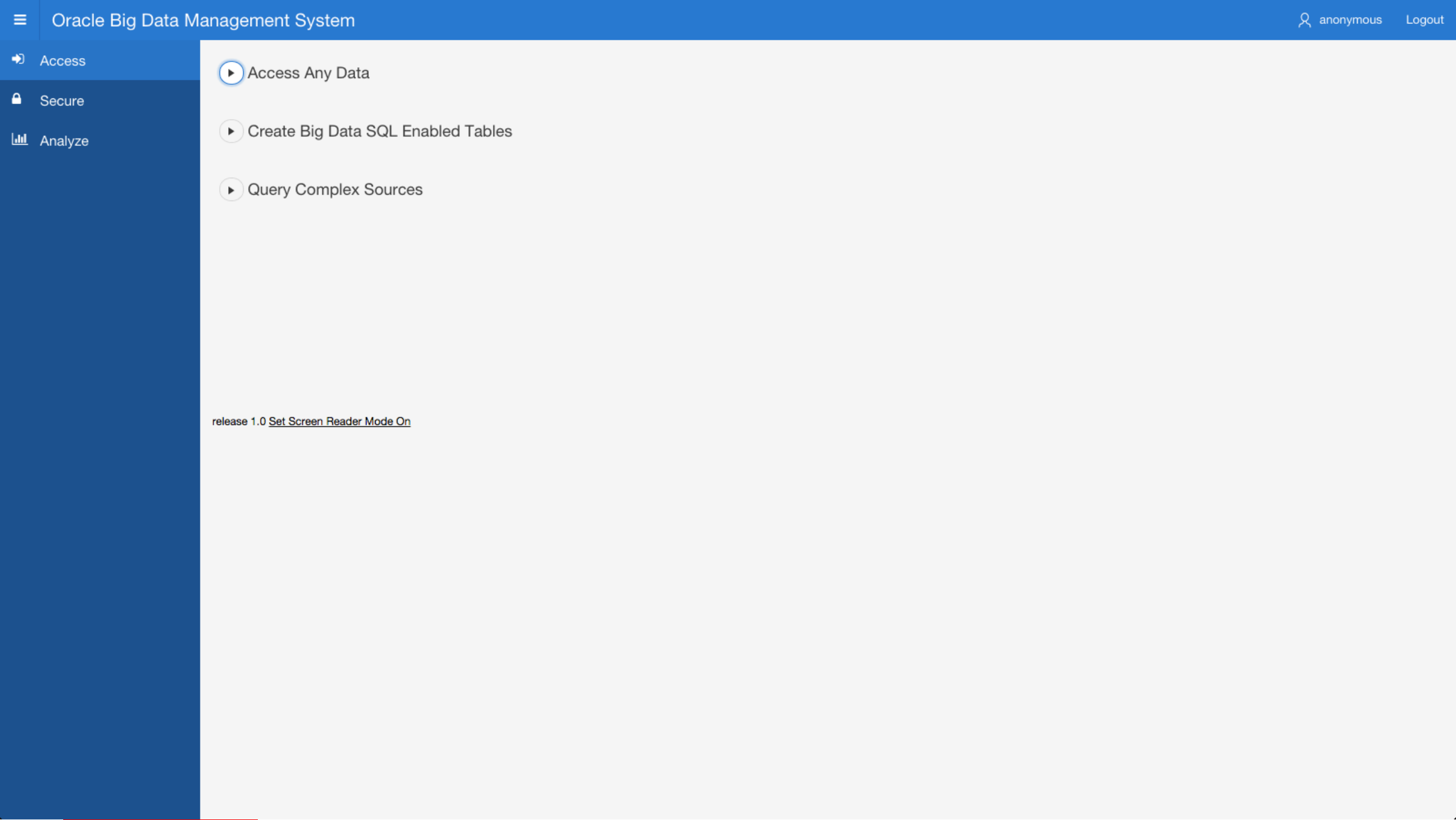
Revenue from purchases (Oracle Relational Data Warehouse)

Example: Click data in its raw format.

Create Big Data SQL Enabled Tables

Query Complex Sources

release 1.0 Set Screen Reader Mode On


Oracle Big Data Management System
anonymous
Logout
Access
Secure
Analyze
Access Any Data
Create Big Data SQL Enabled Tables
Query Complex Sources
release 1.0 Set Screen Reader Mode On

Access
Secure
Analyze

Access Any Data

Create Big Data SQL Enabled Tables

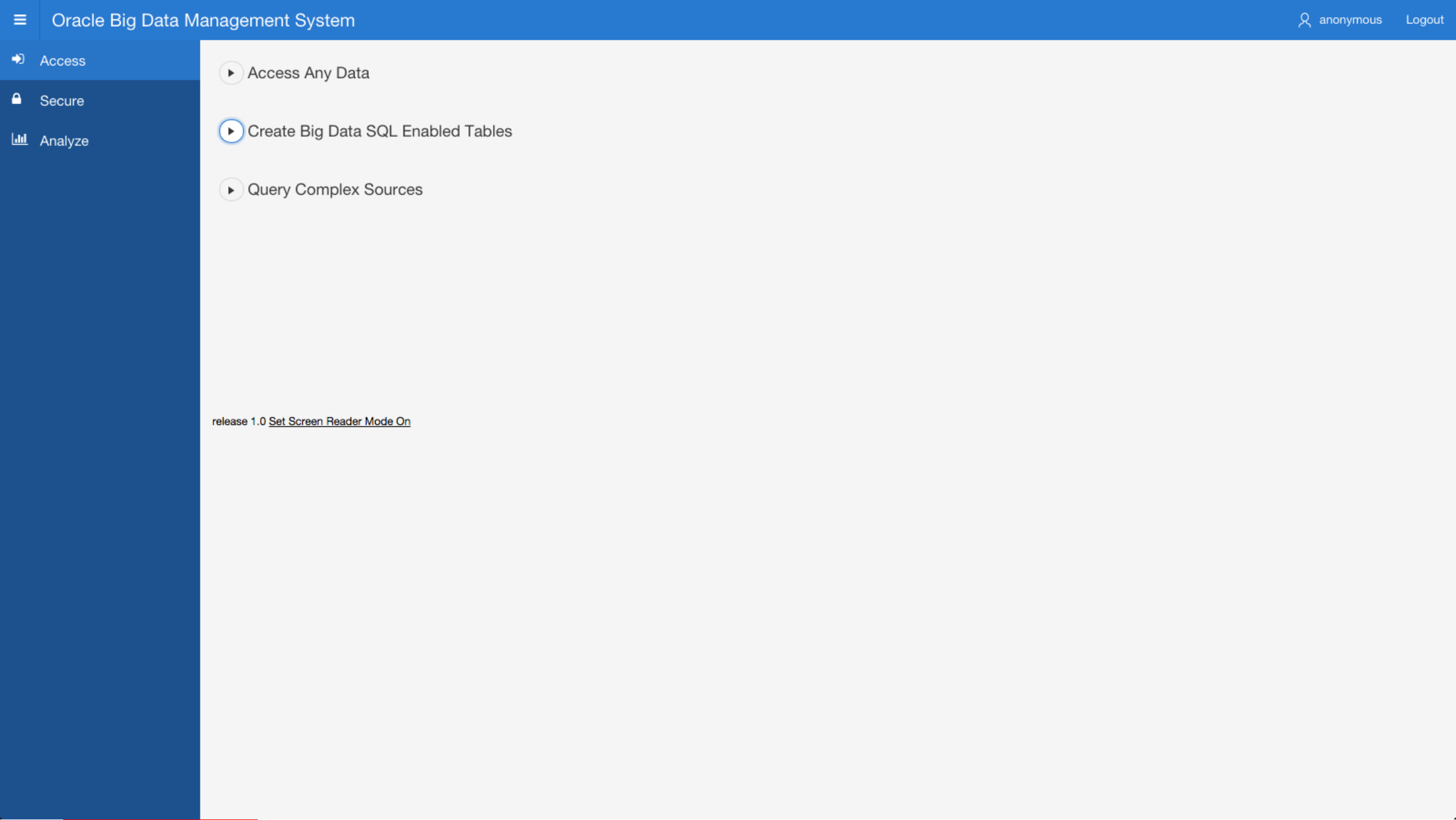
Clicks (HDFS)

```
CREATE TABLE movielog_t
   (click VARCHAR2(4000))
   ORGANIZATION EXTERNAL
   (TYPE ORACLE_HDFS
    DEFAULT DIRECTORY default_dir
    LOCATION ('/user/oracle/moviework/applog_json/')
   )
 REJECT LIMIT UNLIMITED;
```

Recommendations (NoSQL DB)

```
CREATE TABLE recommendation_t
      custid NUMBER,
      sno NUMBER,
      genreid NUMBER,
      movieid NUMBER )
      ORGANIZATION EXTERNAL
      (
          TYPE ORACLE_HIVE
          DEFAULT DIRECTORY default_dir
          ACCESS PARAMETERS
          ( com.oracle.bigdata.tablename:moviework.recommendation )
          )
      REJECT LIMIT UNLIMITED;
```

Query Complex Sources


Oracle Big Data Management System
anonymous
Logout
Access
Secure
Analyze
Access Any Data
Create Big Data SQL Enabled Tables
Query Complex Sources
release 1.0 Set Screen Reader Mode On

Access
Secure
Analyze

Access Any Data

Create Big Data SQL Enabled Tables

Query Complex Sources

Query Clicks (JSON)

```
SELECT
    m.click.custid,
    m.click.movieid,
    m.click.genreid,
    m.click.time
FROM movielog_t m
WHERE rownum < 20;
```

Custid	Movieid	Genreid	Time
1185972	-	-	2012-07-01:00:00:07
1354924	1948	9	2012-07-01:00:00:22
1083711	-	-	2012-07-01:00:00:26
1234182	11547	6	2012-07-01:00:00:32
1010220	11547	6	2012-07-01:00:00:42
1143971	-	-	2012-07-01:00:00:43
1253676	-	-	2012-07-01:00:00:50
1351777	608	6	2012-07-01:00:01:03
1143971	-	-	2012-07-01:00:01:07
1363545	27205	9	2012-07-01:00:01:18
1067283	1124	9	2012-07-01:00:01:26
1126174	16309	46	2012-07-01:00:01:35

Query Recommendations (NoSQL DB)

```
SELECT *
FROM recommendation_t
WHERE rownum <=20;
```

Custid	Movieid	Genreid	Sno
1024438	121	11	1
1024438	196	11	1
1024438	435	11	1
1024438	488	11	1
1024438	857	11	1
1024438	956	11	1
1024438	1273	11	1
1024438	8367	11	1
1024438	9346	11	1
1024438	9836	11	1
1024438	10539	11	1
1024438	10911	11	1

Access
Secure
Analyze

Oracle Data Redaction Over Customers Table

Define Redaction Policy Over Data in HDFS and NoSQL DB

Query Redacted Data

Access
Secure
Analyze

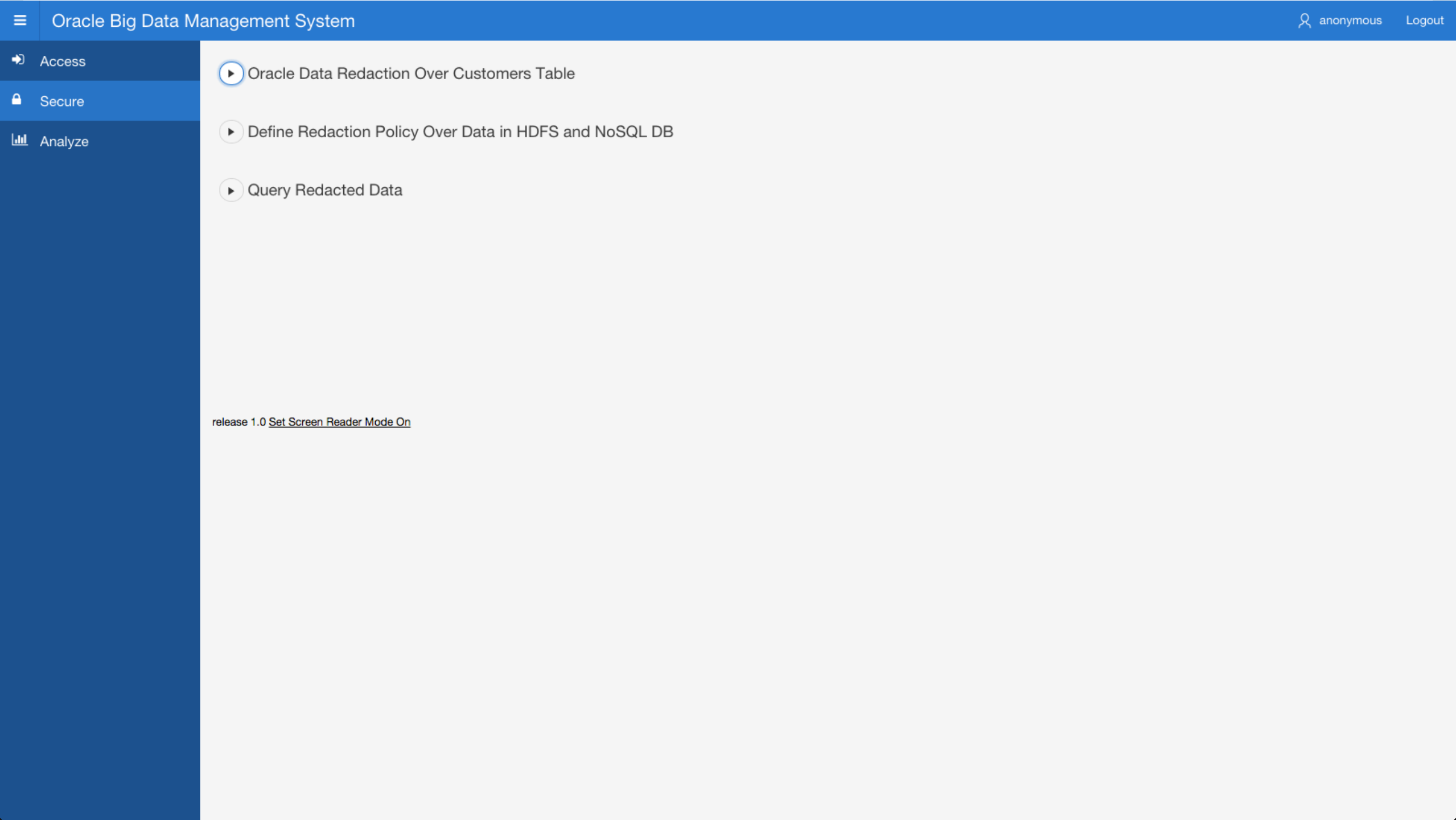
Oracle Data Redaction Over Customers Table

```
SELECT cust_id,
       last_name,
       first_name
FROM customer
```

CUST_ID	LAST_NAME	FIRST_NAME
9999999	Su****	Otto
9999999	Ro****	Bhadrabuja
9999999	Sr******	Motilal
9999999	Ka*****	Qing
9999999	Az******	Lyuba
9999999	Br*****	Cittabhoga
9999999	Er******	Gracie
9999999	Mc*****	Wiley
9999999	Tr******	Alisha
9999999	Ti*****	Jason
9999999	Ho****	Claudia
9999999	Pa***	Gregorio
9999999	Ri****	Adolfo
9999999	Al****	Leo
9999999	Ca*****	Marci

1 - 15 Next ▶

Define Redaction Policy Over Data in HDFS and NoSQL DB


Oracle Big Data Management System
anonymous
Logout
Access
Secure
Analyze
Oracle Data Redaction Over Customers Table
Define Redaction Policy Over Data in HDFS and NoSQL DB
Query Redacted Data
release 1.0 Set Screen Reader Mode On

Oracle Data Redaction Over Customers Table

Define Redaction Policy Over Data in HDFS and NoSQL DB

```
BEGIN
-- JSON file in HDFS
DBMS_REDACT.ADD_POLICY (
  object_schema => 'MOVIEDEMO',
  object_name => 'MOVIELOG_V',
  column_name => 'CUSTID',
  policy_name => 'movielog_v_redaction',
  function_type => DBMS_REDACT.PARTIAL,
  function_parameters => '9,1,7',
  expression => '1=1' );

-- Recommendations data from Oracle NoSQL Database
DBMS_REDACT.ADD_POLICY(
  object_schema => 'MOVIEDEMO',
  object_name => 'RECOMMENDATION',
  column_name => 'CUSTID',
  policy_name => 'recommendation_redaction',
  function_type => DBMS_REDACT.PARTIAL,
  function_parameters => '9,1,7',
  expression => '1=1' );
END;
 /
```

Query Redacted Data

Access
Secure
Analyze

Oracle Data Redaction Over Customers Table

Define Redaction Policy Over Data in HDFS and NoSQL DB

Query Redacted Data

Access
Secure
Analyze

Oracle Data Redaction Over Customers Table

Define Redaction Policy Over Data in HDFS and NoSQL DB

Query Redacted Data

Clicks

Custid	Movieid	Genreid	Time
9999999	-	-	2012-07-01:00:00:07
9999999	1948	9	2012-07-01:00:00:22
9999999	-	-	2012-07-01:00:00:26
9999999	11547	6	2012-07-01:00:00:32
9999999	11547	6	2012-07-01:00:00:42
9999999	-	-	2012-07-01:00:00:43
9999999	-	-	2012-07-01:00:00:50
9999999	608	6	2012-07-01:00:01:03
9999999	-	-	2012-07-01:00:01:07
9999999	27205	9	2012-07-01:00:01:18
9999999	1124	9	2012-07-01:00:01:26
9999999	16309	46	2012-07-01:00:01:35
9999999	11547	6	2012-07-01:00:01:39
9999999	-	-	2012-07-01:00:01:55
9999999	-	-	2012-07-01:00:01:58

1 - 15 Next ▸

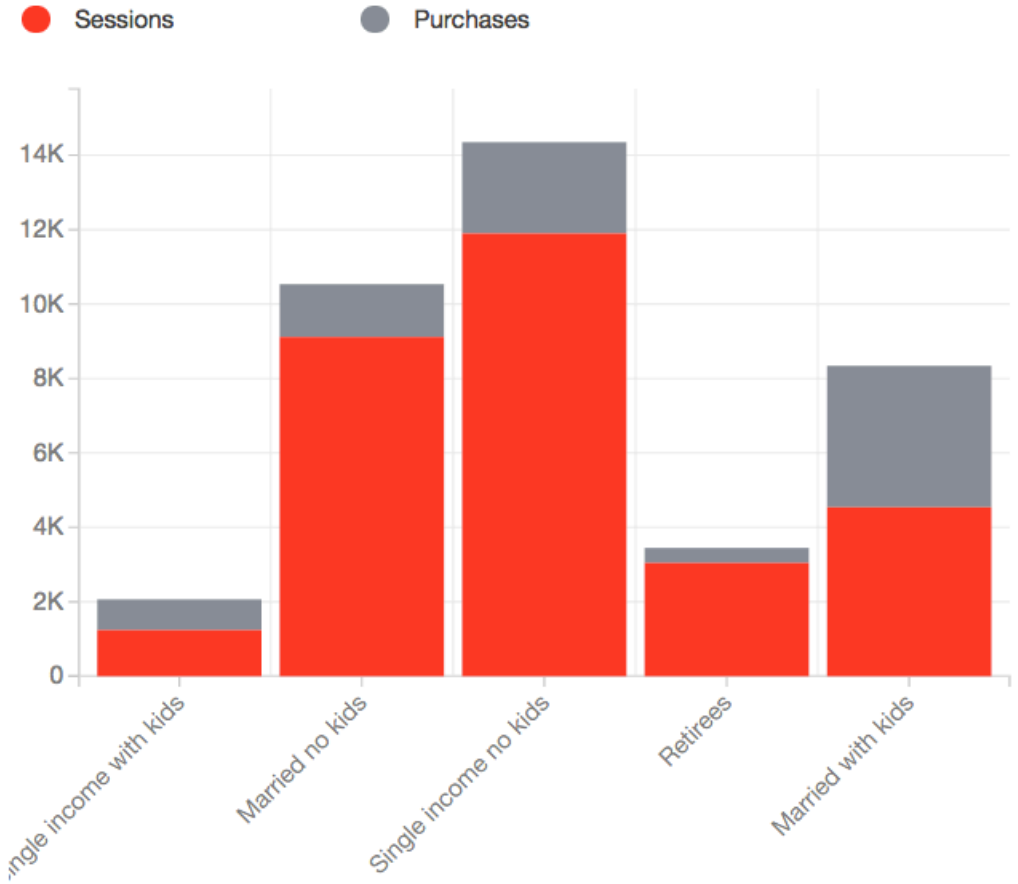
Recommendations

Custid	Movieid	Genreid	Sno
9999999	121	11	1
9999999	196	11	1
9999999	435	11	1
9999999	488	11	1
9999999	857	11	1
9999999	956	11	1
9999999	1273	11	1
9999999	8367	11	1
9999999	9346	11	1
9999999	9836	11	1
9999999	10539	11	1
9999999	10911	11	1
9999999	11046	11	1
9999999	11866	11	1
9999999	12444	11	1

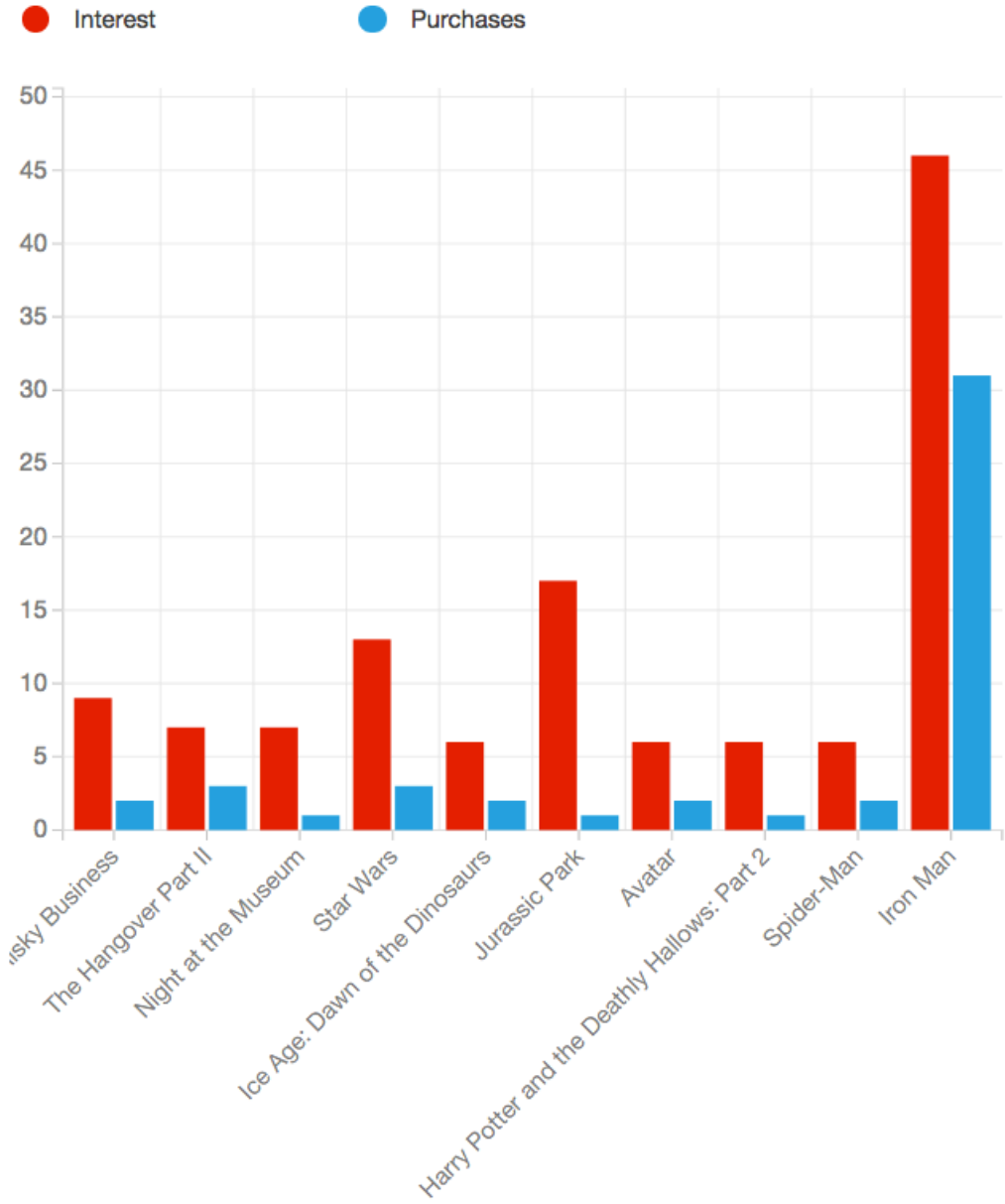
1 - 15 Next ▸

Access
Secure
Analyze

Session Conversions



Recommendation Effectiveness



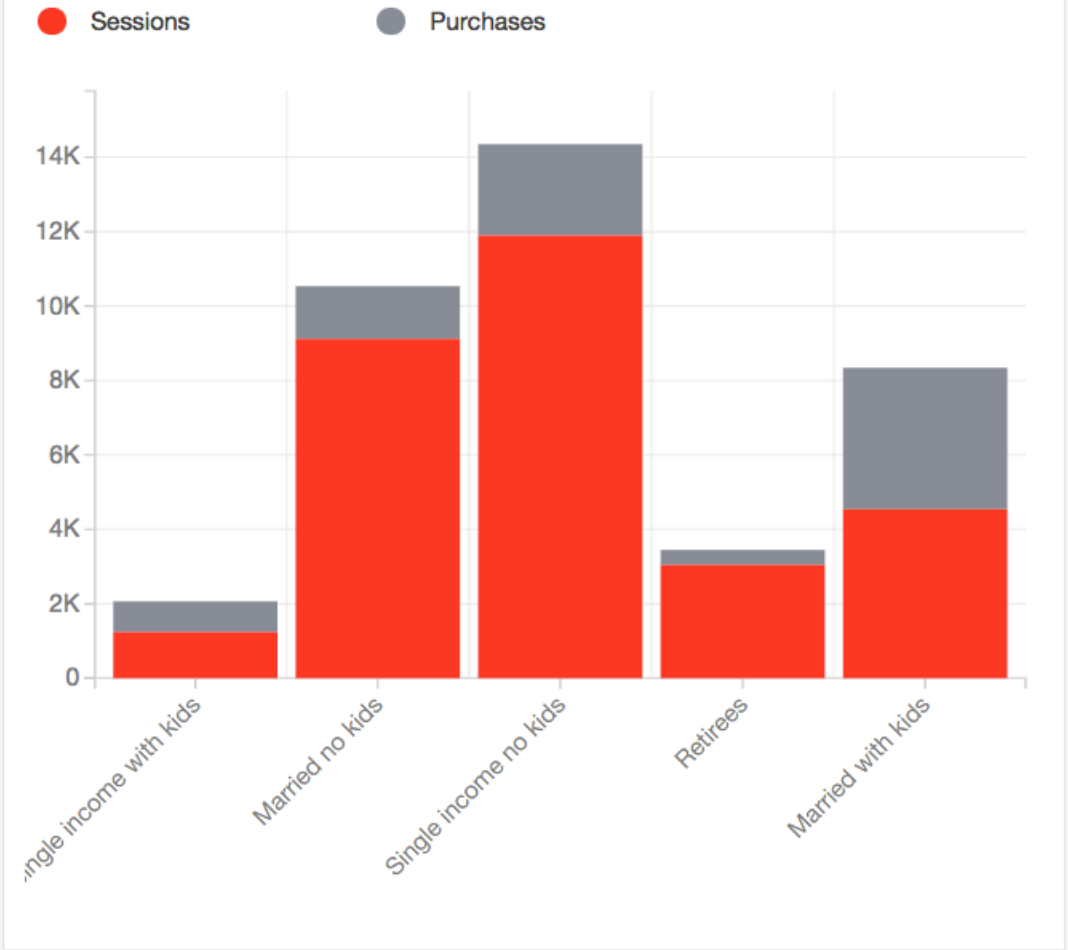
Sessionize | Hue - File Browser - movie | http://denaa246:7070/orc | Marty

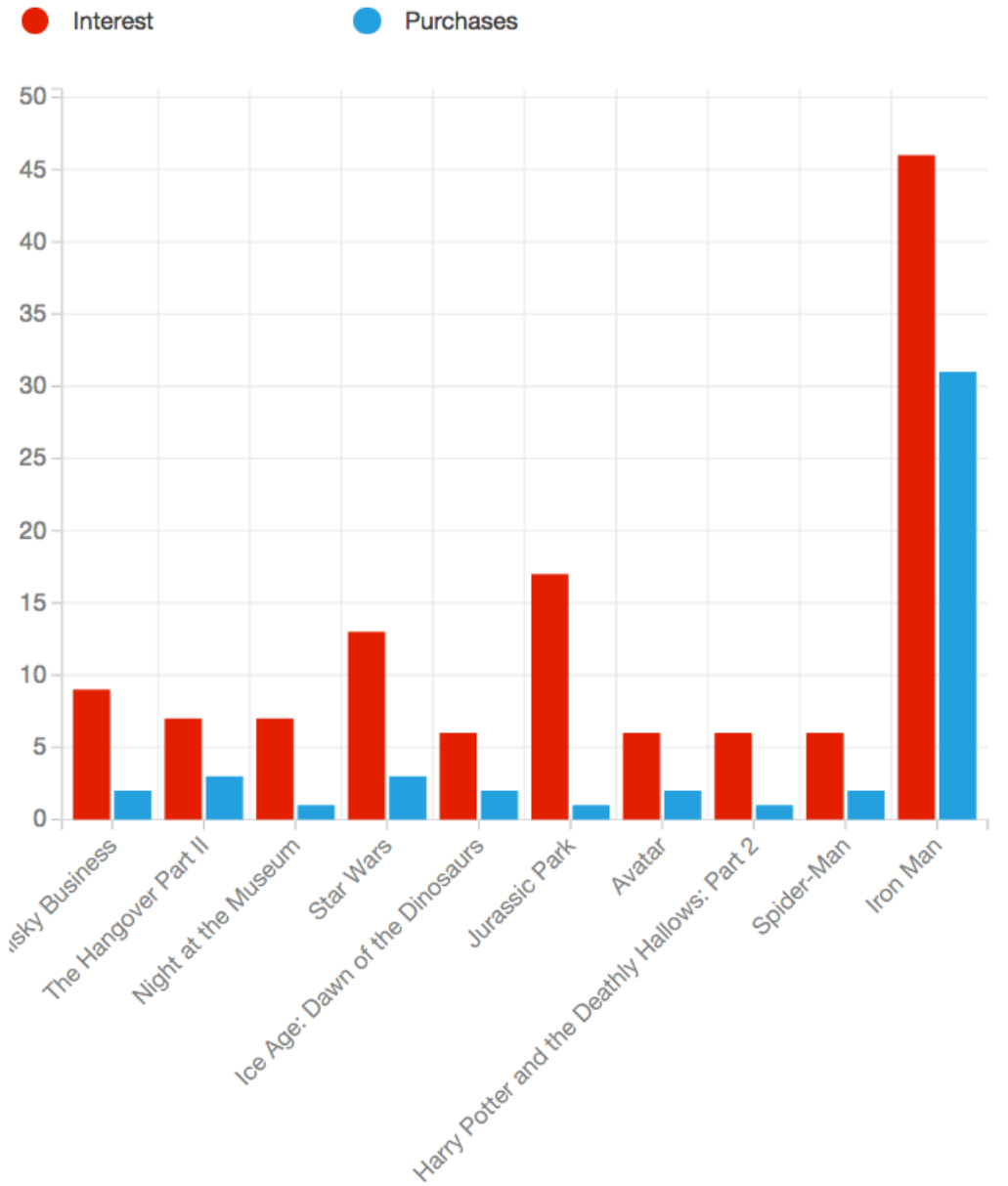
orabda:7777/apex/f?p=106:4:::NO:::

Apps | BIEE | OOW | CS50.tv | Big Data | SecureCluster | Big_Data_Use_Cases | Big Data Science Boo | Configuration Tasks | EM | Save to Instapaper | scaj51 | News | Technology Launchp | Other Bookmarks

Secure

Analyze





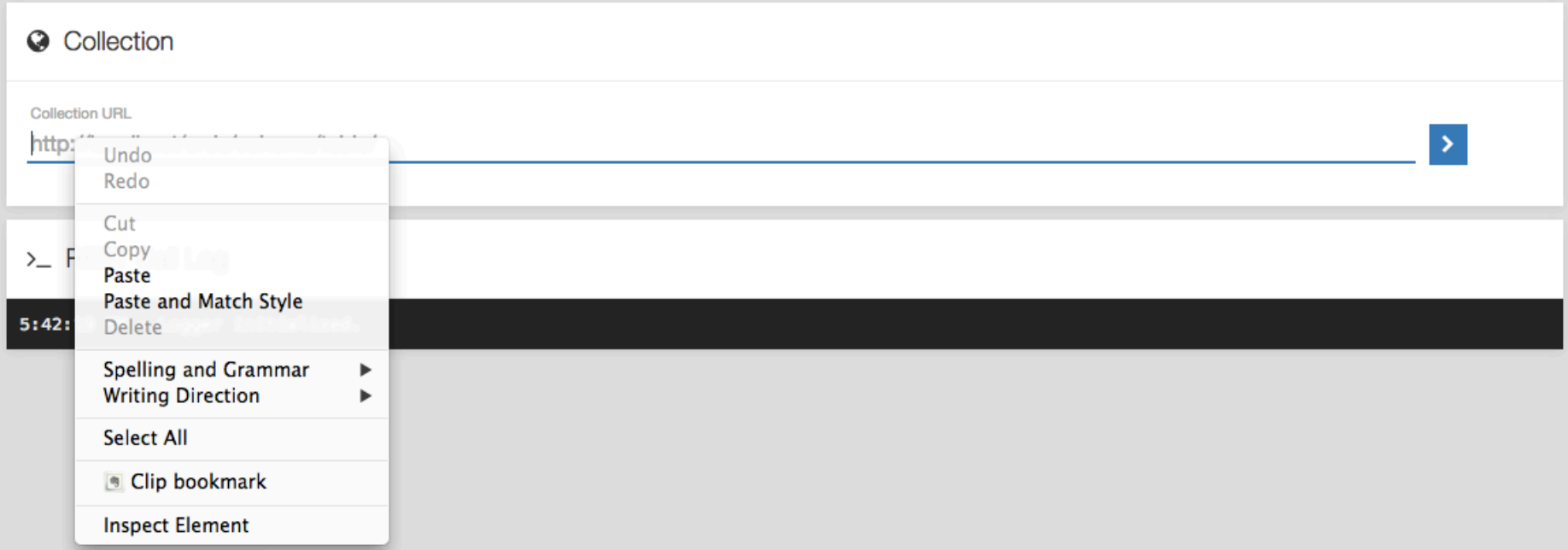
Collection

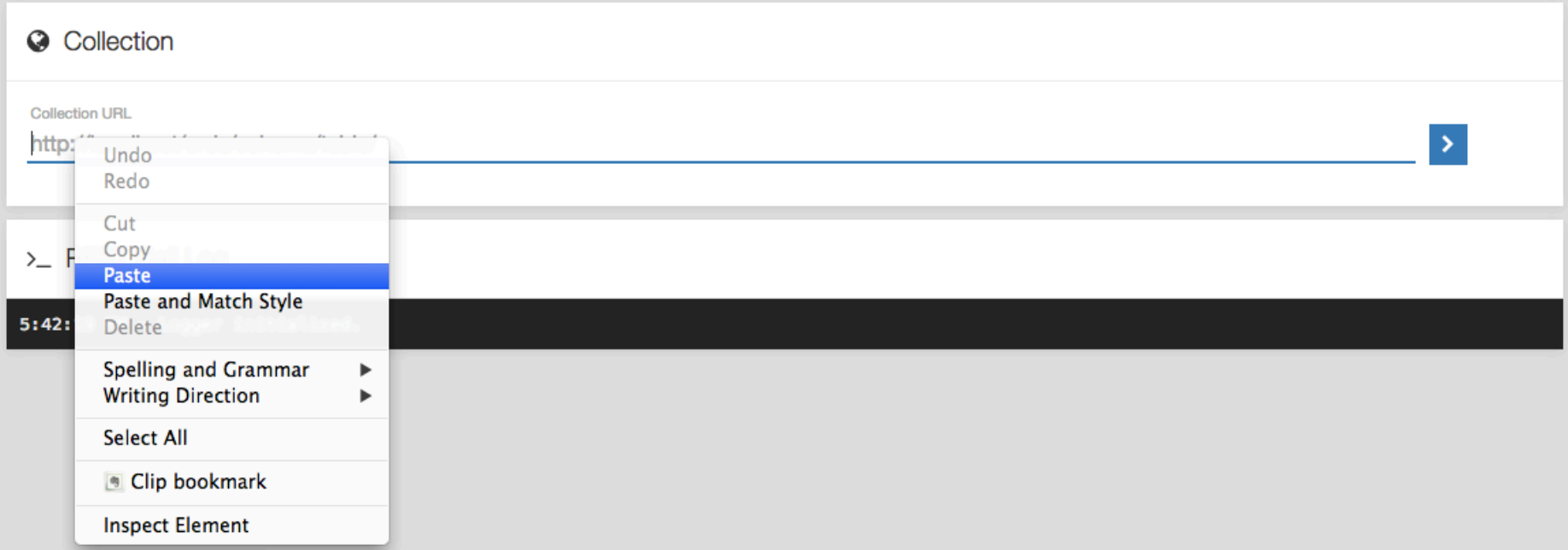
Collection URL

http://localhost/ords/schema/table/

>_ REST Call Log

```
5:42:19 PM - Logger initialized.
```


Collection
Collection URL
http:
Undo
Redo
Cut
Copy
Paste
Paste and Match Style
Delete
Spelling and Grammar
Writing Direction
Select All
Clip bookmark
Inspect Element
5:42:


Collection
Collection URL
http:
Undo
Redo
Cut
Copy
Paste
Paste and Match Style
Delete
Spelling and Grammar
Writing Direction
Select All
Clip bookmark
Inspect Element
5:42:

Collection

Collection URL

http://denaa246:7070/ords/moviedemo/customer_rfm/?q={"rfm_monetary":5,"rfm_recency":1}

>_ REST Call Log

```
5:42:19 PM - Logger initialized.
```

Collection

Collection URL

http://denaa246:7070/ords/moviedemo/customer_rfm/?q={"rfm_monetary":5,"rfm_recency":1}

>_ REST Call Log

```
5:42:19 PM - Logger initialized.
```

Collection

Collection URL

http://denaa246:7070/ords/moviedemo/customer_rfm/?q={"rfm_monetary":5,"rfm_recency":1}

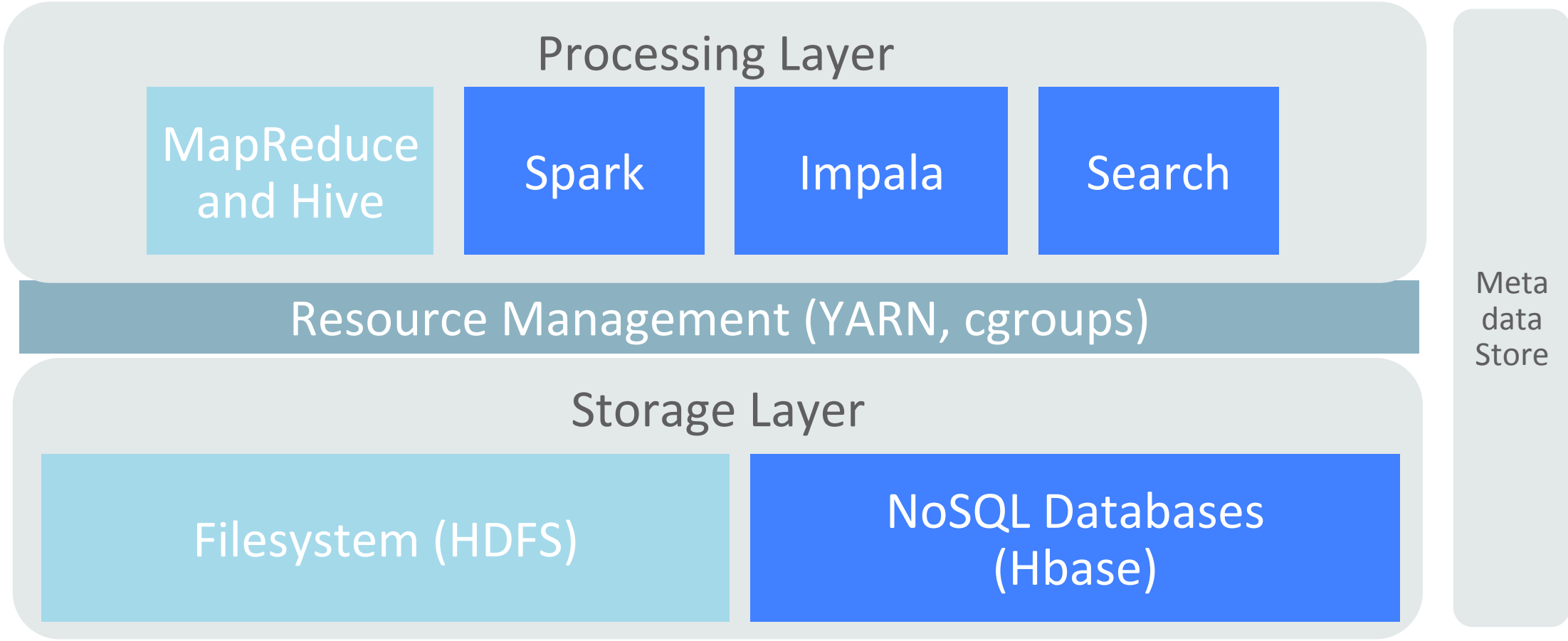
Data

Column: cust_id | Operation: equals | Filter: Value

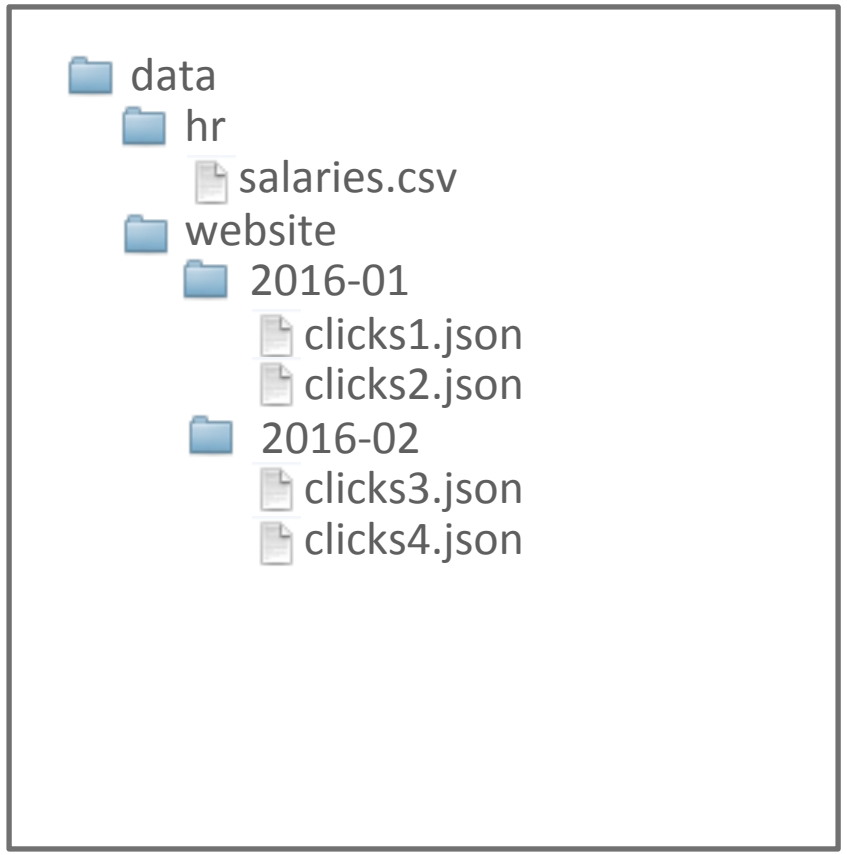
cust_id	last_name	first_name	rfm_recency	rfm_frequency	rfm_monetary	rfm_combined
9999999	Ma***	Souta	1	3	5	135
9999999	De******	Jerry	1	3	5	135
9999999	Wi****	Jayashankar	1	4	5	145
9999999	Pi****	Otis	1	4	5	145
9999999	Va*****	Vladik	1	4	5	145
9999999	En**	Haruna	1	3	5	135
9999999	Ju	Huan	1	3	5	135
9999999	No****	Collin	1	3	5	135
9999999	Ko***	Wei-Kang	1	4	5	145
9999999	Ta***	Khandut	1	3	5	135
9999999	Sa**	Abdelwahab	1	3	5	135

Technical Details:
A Primer on Hadoop and Hive

Hadoop Architecture

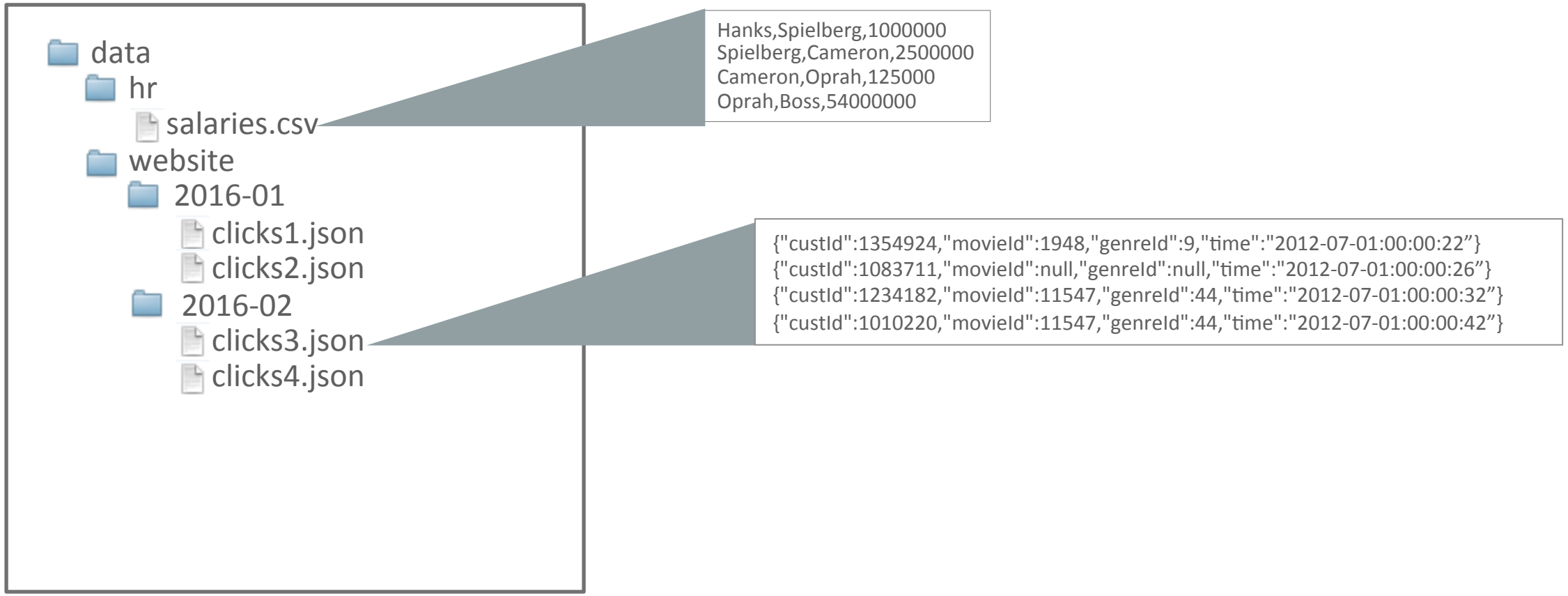


How Data is Stored in HDFS

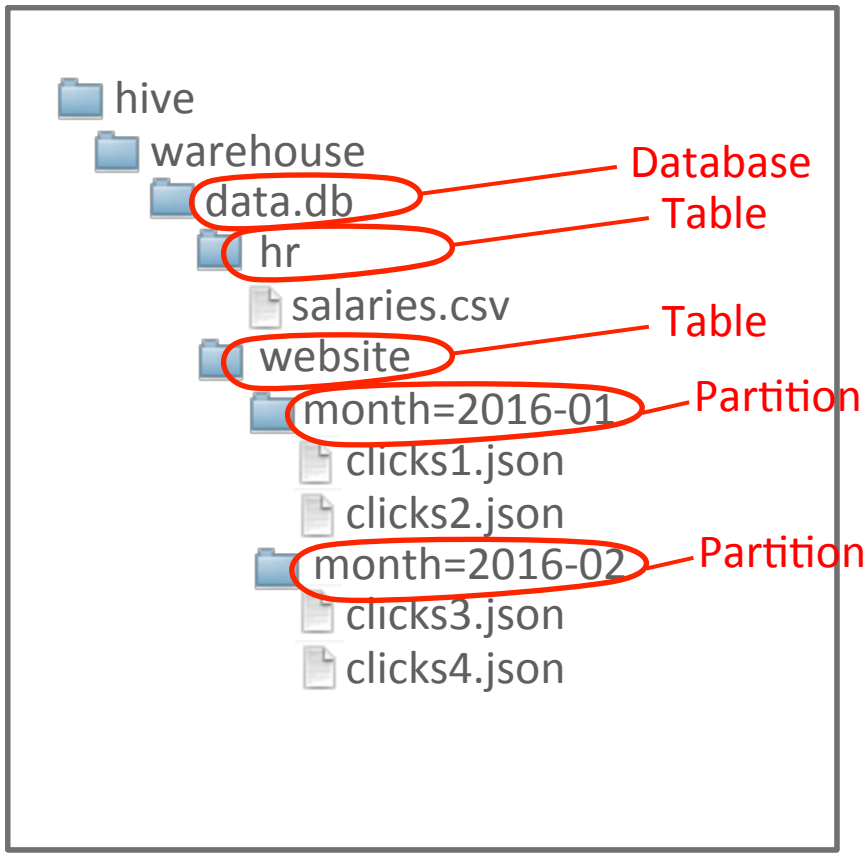


- Data stored in files and organized into folders
  - Can be any file type
  - Replicated 3x across cluster
- Schema on Read
  - The tool reading the data interprets the data as it sees fit

How Data is Stored in HDFS

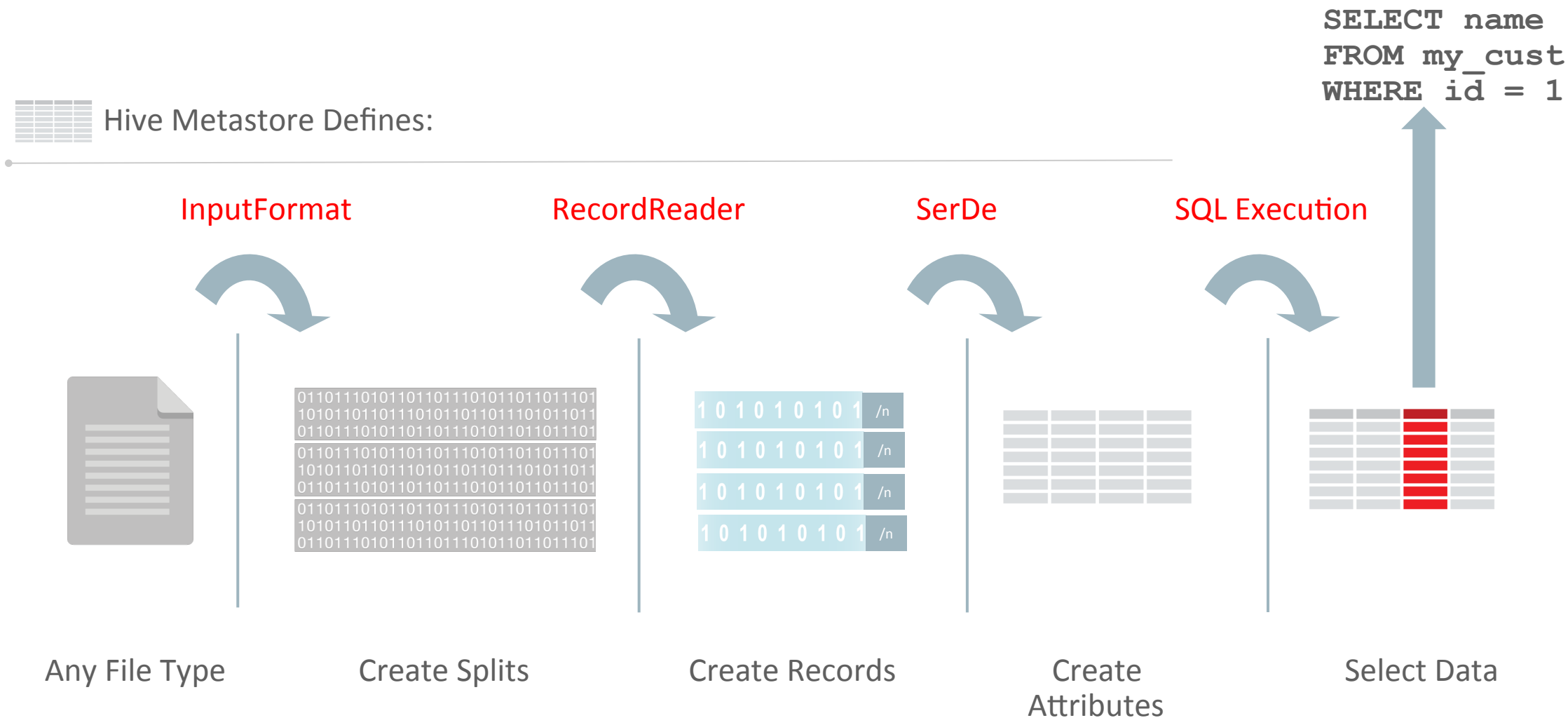


Organize and Describe Data with Hive

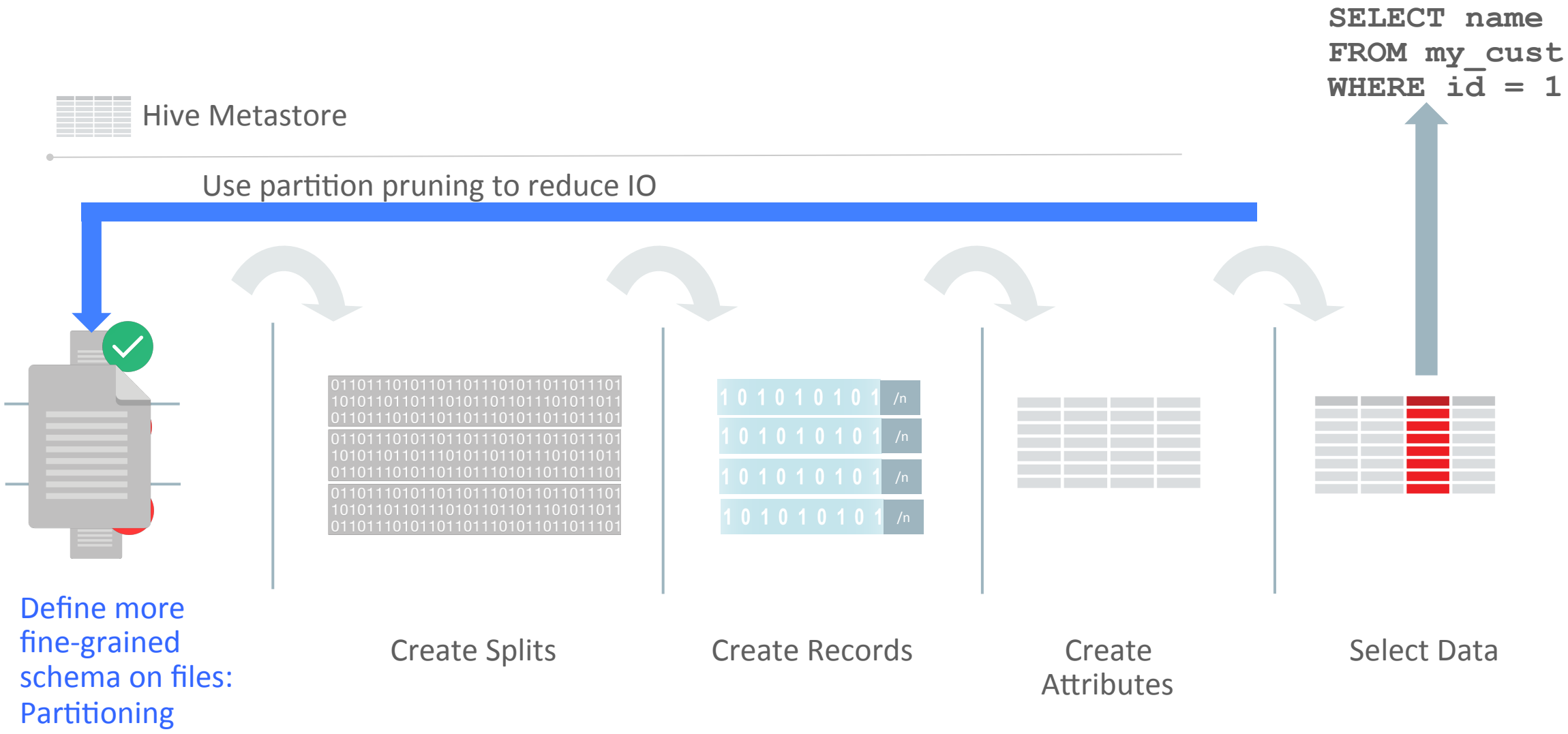


- Information is captured in **Hive Metastore**
- HDFS Folders become:
  - Databases
  - Tables
  - Partitions
- **Table** includes metadata for parsing files using Java classes
  - InputFormat defines chunks called splits based on file type
  - RecordReader creates rows out of splits
  - SerDe creates columns

How does Hive read ANY data?



How does Hive read data FASTER?

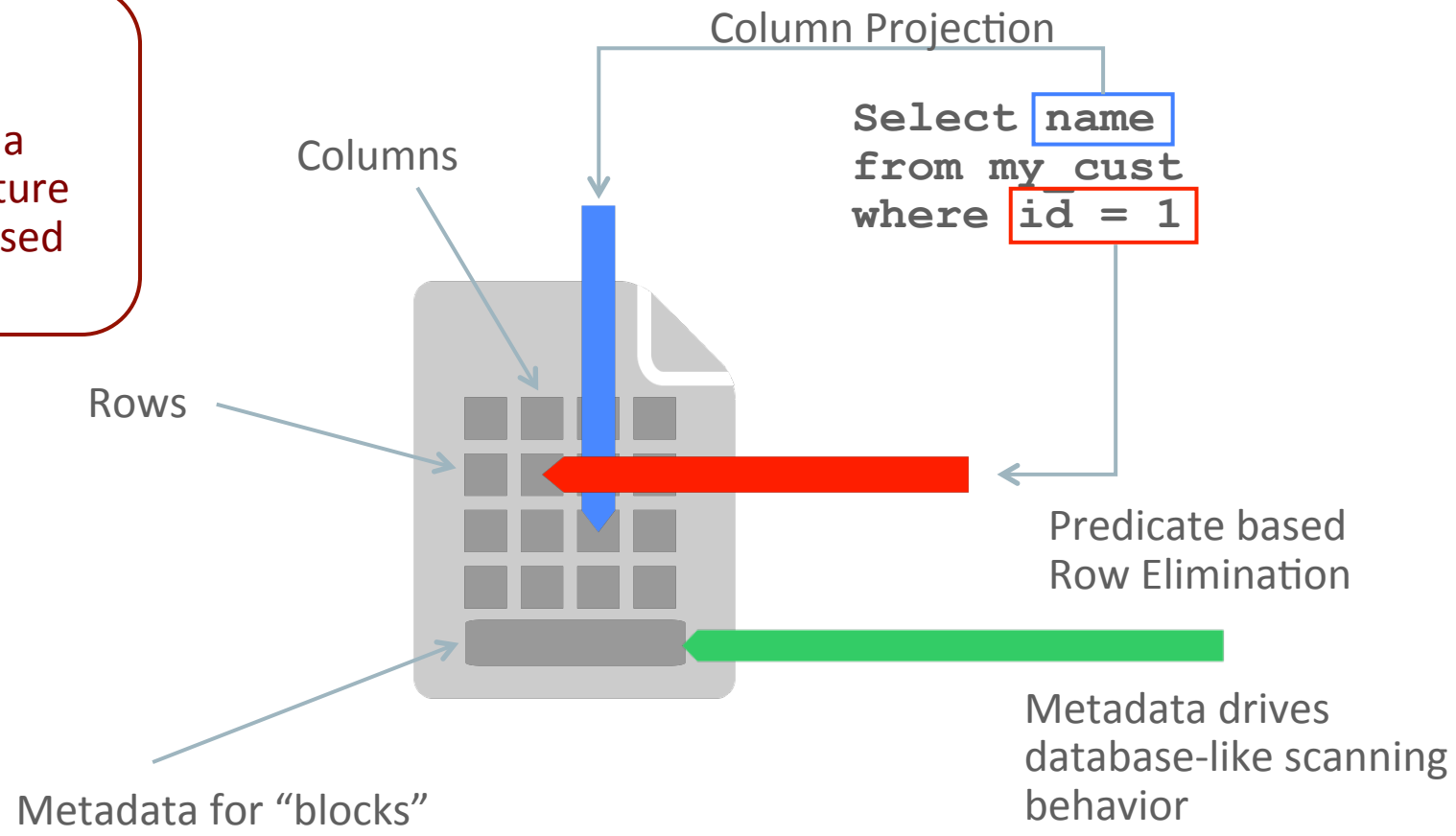


How does Parquet Work?

Create and Query Parquet Files

Schema on Write

Parquet implements a database storage structure with metadata and parsed data elements

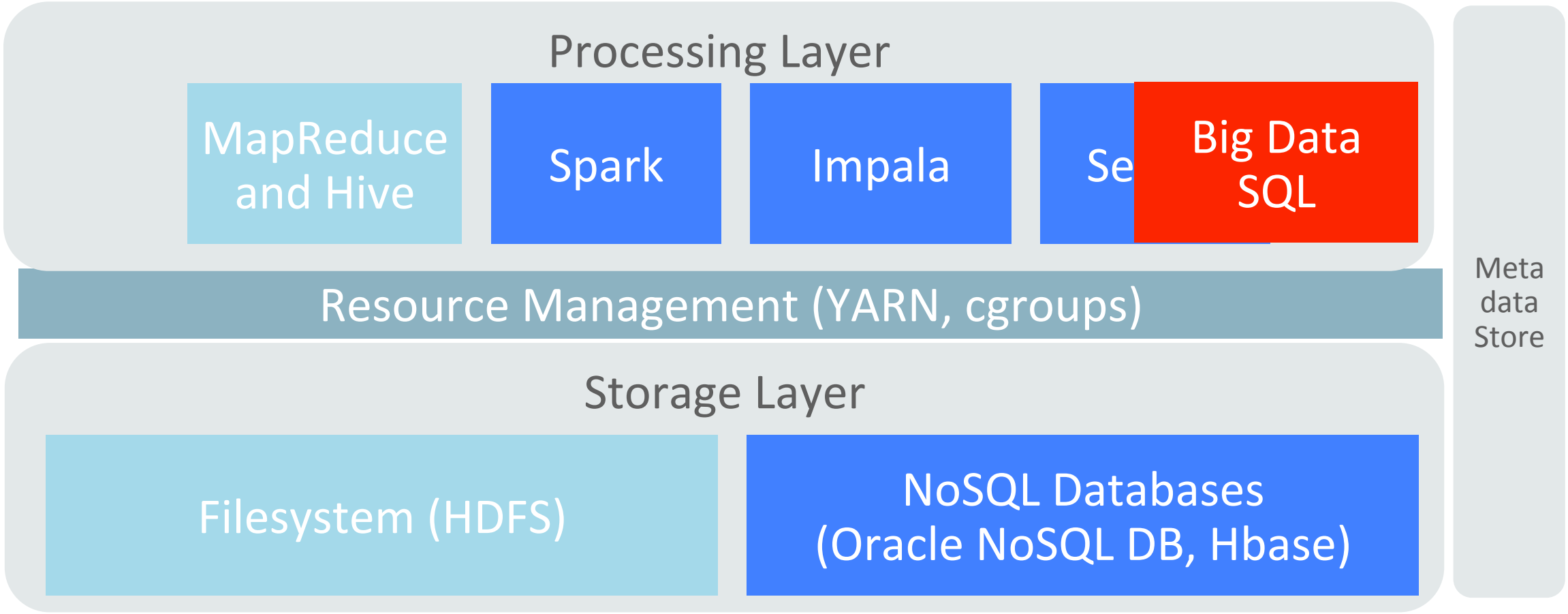


Metadata: Extend Oracle External Tables

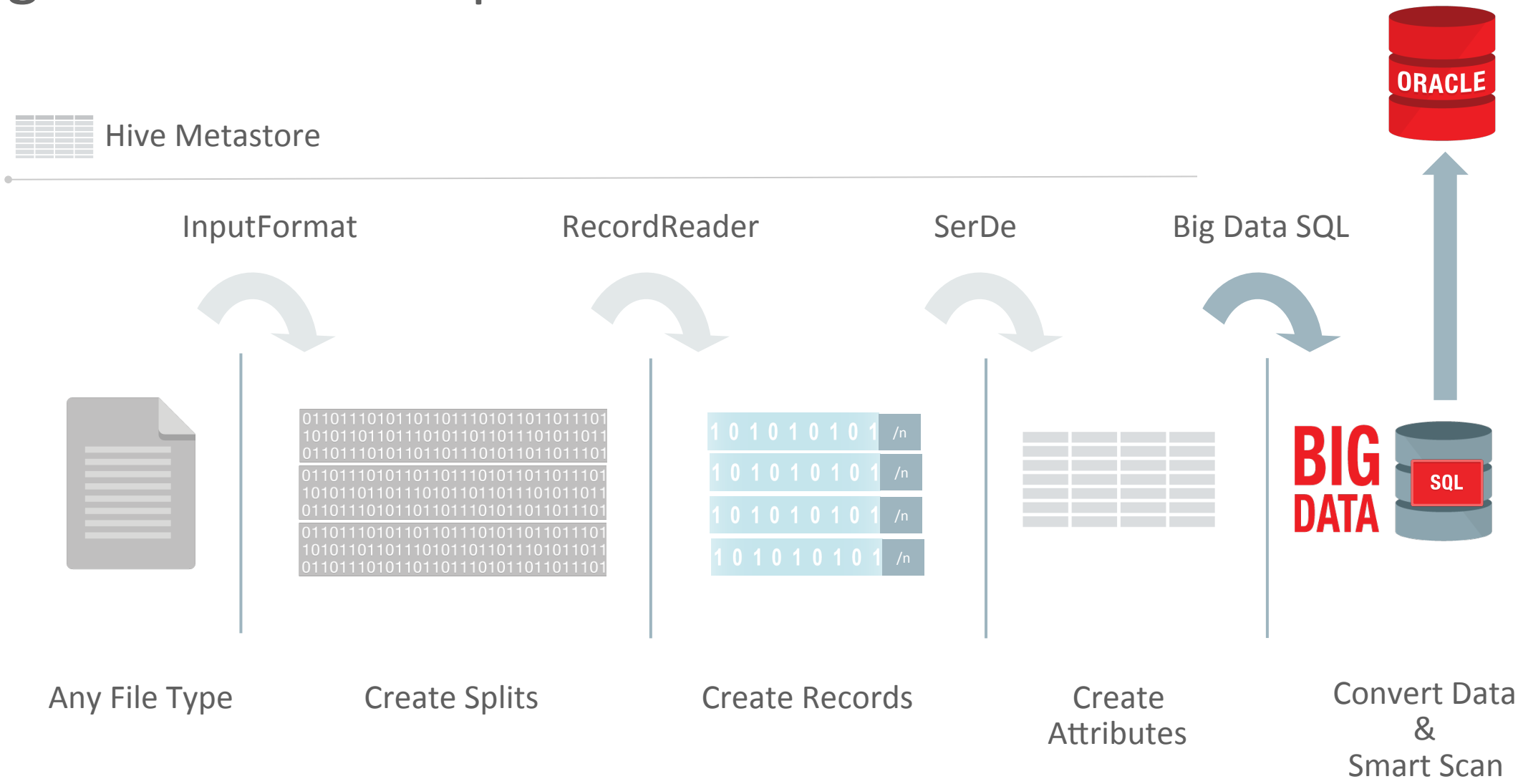
```
CREATE TABLE movielog (
  click VARCHAR2(4000))
ORGANIZATION EXTERNAL (
  TYPE ORACLE_HIVE
  DEFAULT DIRECTORY DEFAULT_DIR
  ACCESS PARAMETERS
  (
com.oracle.bigdata.tablename logs
com.oracle.bigdata.cluster mycluster
  ))
REJECT LIMIT UNLIMITED;
```

- New types of external tables
  - ORACLE_HIVE (leverage hive metadata)
  - ORACLE_HDFS (specify metadata)
- Access parameters for Big Data
  - Hadoop cluster, Hive database/table, more
- Tooling available to automatically generate tables
  - SQL Developer & DBMS_HADOOP package

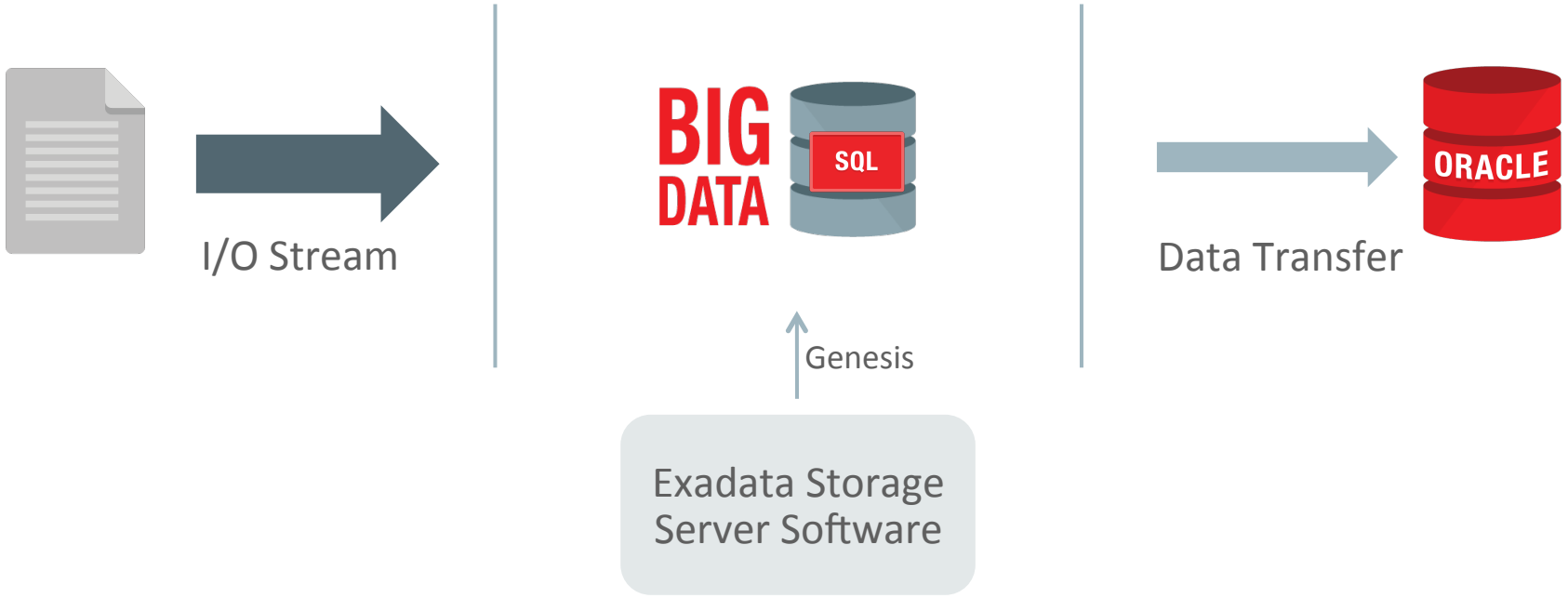
Big Data SQL: Another Hadoop Processing Engine



Big Data SQL on top of “Hive” Data

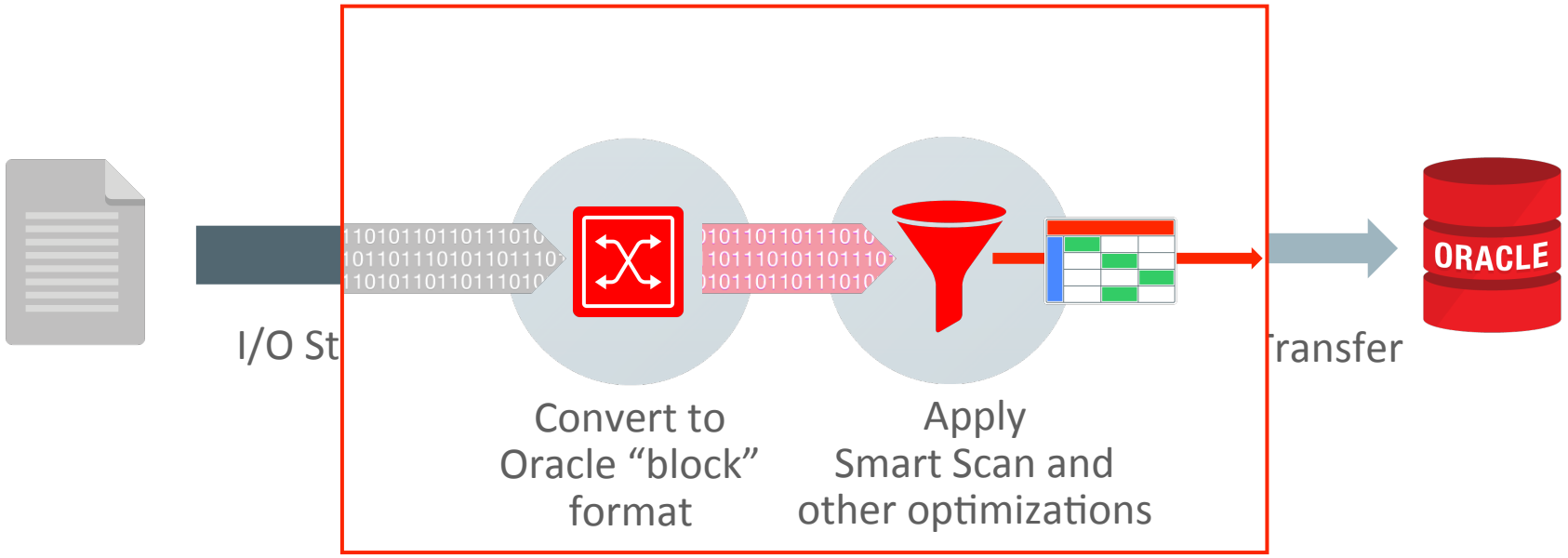


Anatomy of a Big Data SQL Cell



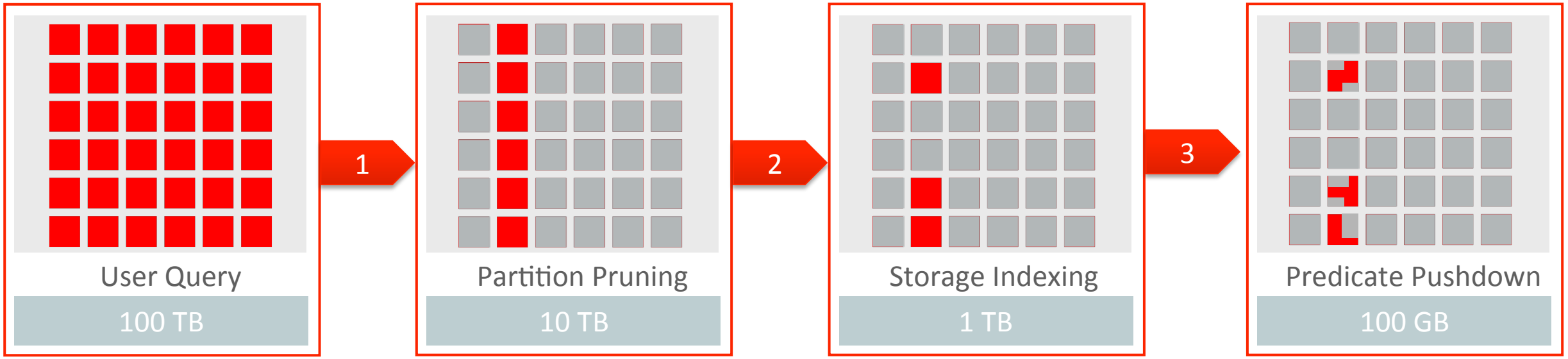
Anatomy of a Big Data SQL Cell

Smart Scan



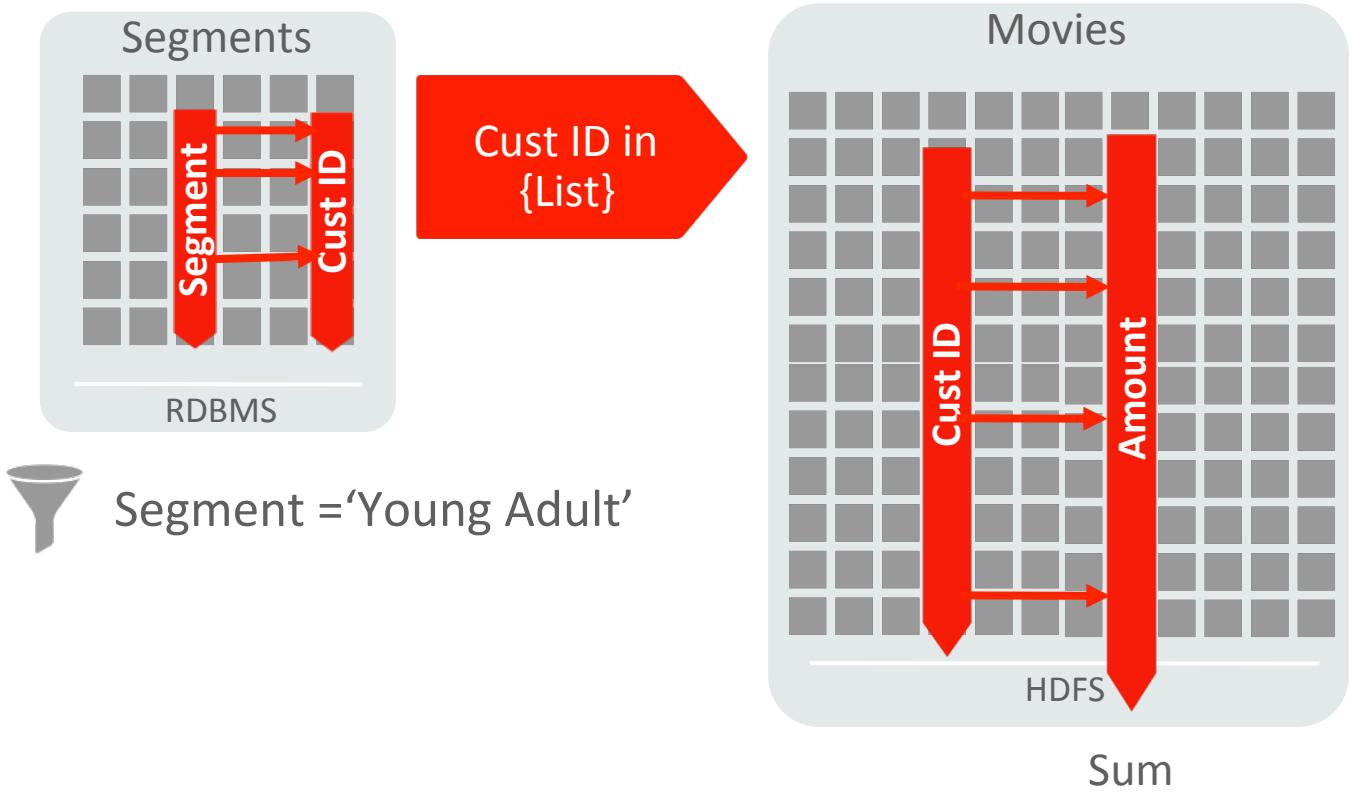
Big Data SQL Performance Features

IO Reduction Features Deliver Compound Results



Join Optimization: Bloom Filter on Hadoop Nodes

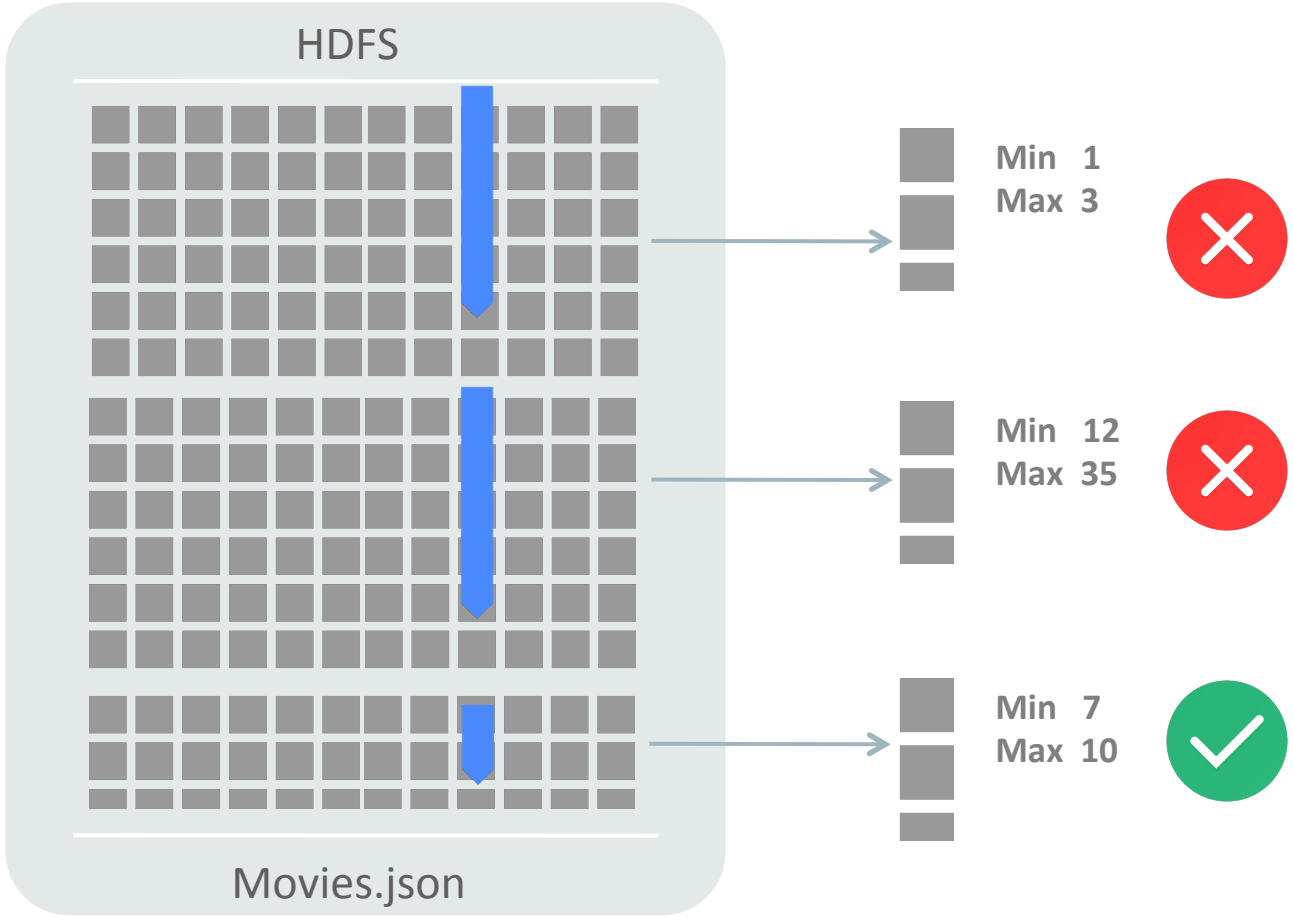
Example: Total movie sales for customer segment



- Converts joins of data in multiple tables into scans
- Result:
  - Scans are pushed down to Hadoop nodes and executed locally
  - No data moved to Database to process joins
  - Massive speed up of query
- Works with data spanning DB and Hadoop as well as data in two Hadoop data sets

Big Data SQL Storage Index

Example: Find revenue for movies in a category **9** (Comedy)



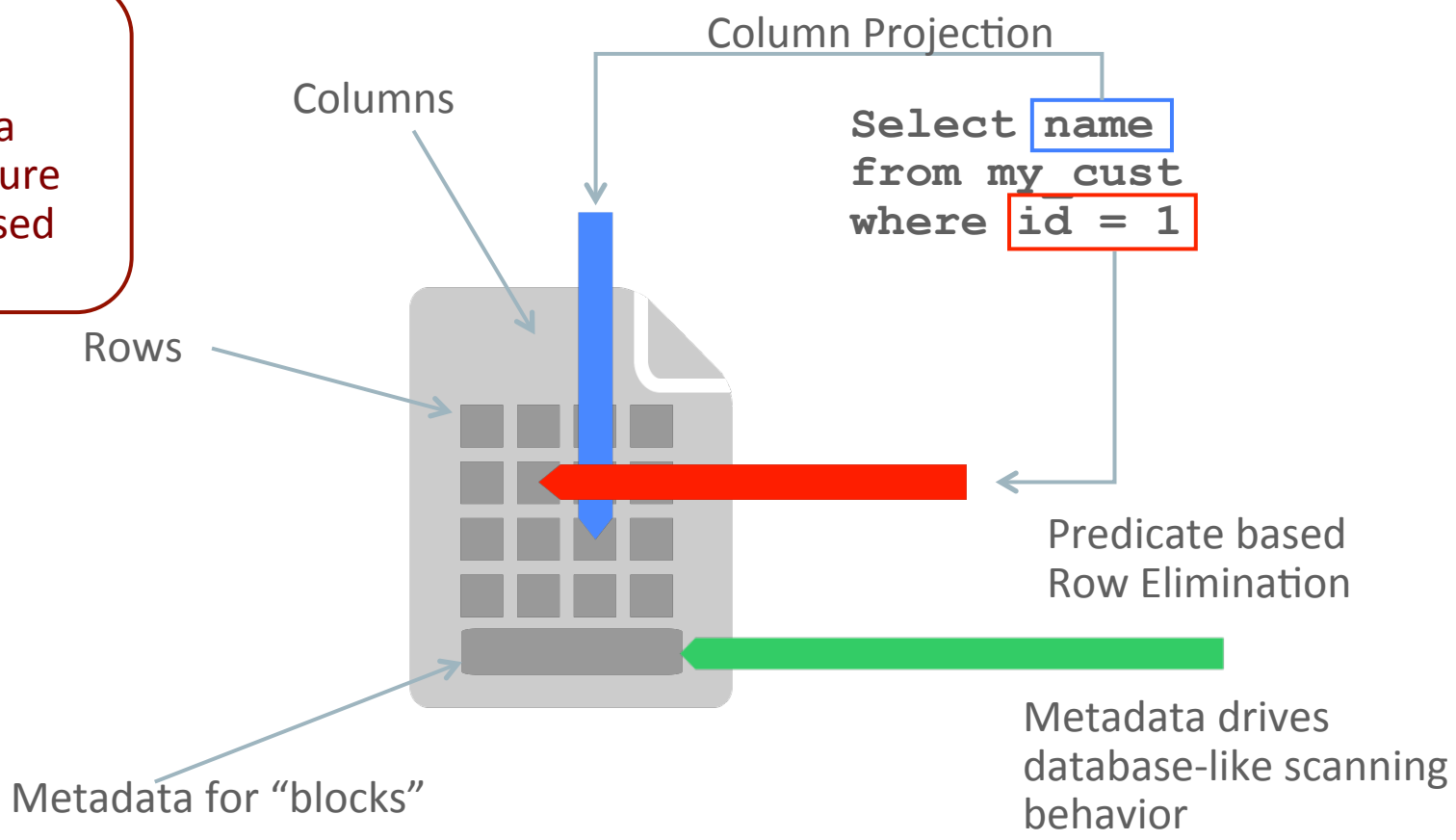
- Big Data SQL Storage Index works on HDFS Chunks and creates an SI for each chunk
- Min / max value is recorded for columns included in a storage index (max # of colums = 32)
- Storage index provides partition pruning like performance for un-modeled data sets

Predicate Pushdown to Intelligent Sources

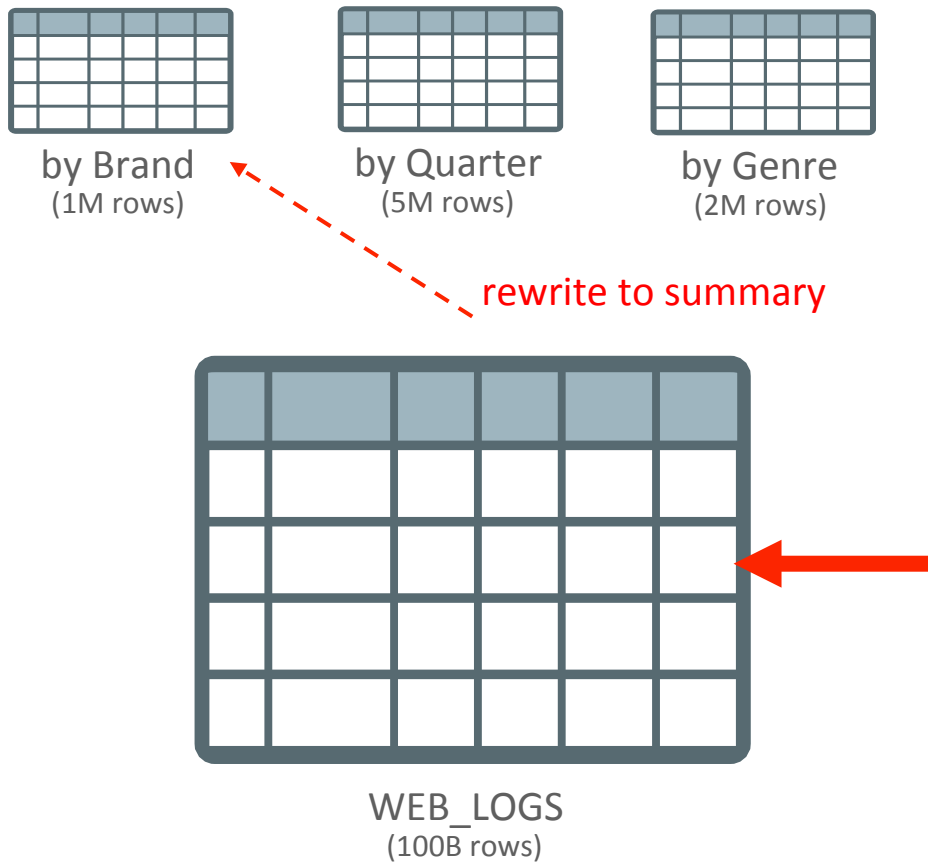
Example: Querying Parquet

Schema on Write

Parquet implements a database storage structure with metadata and parsed data elements

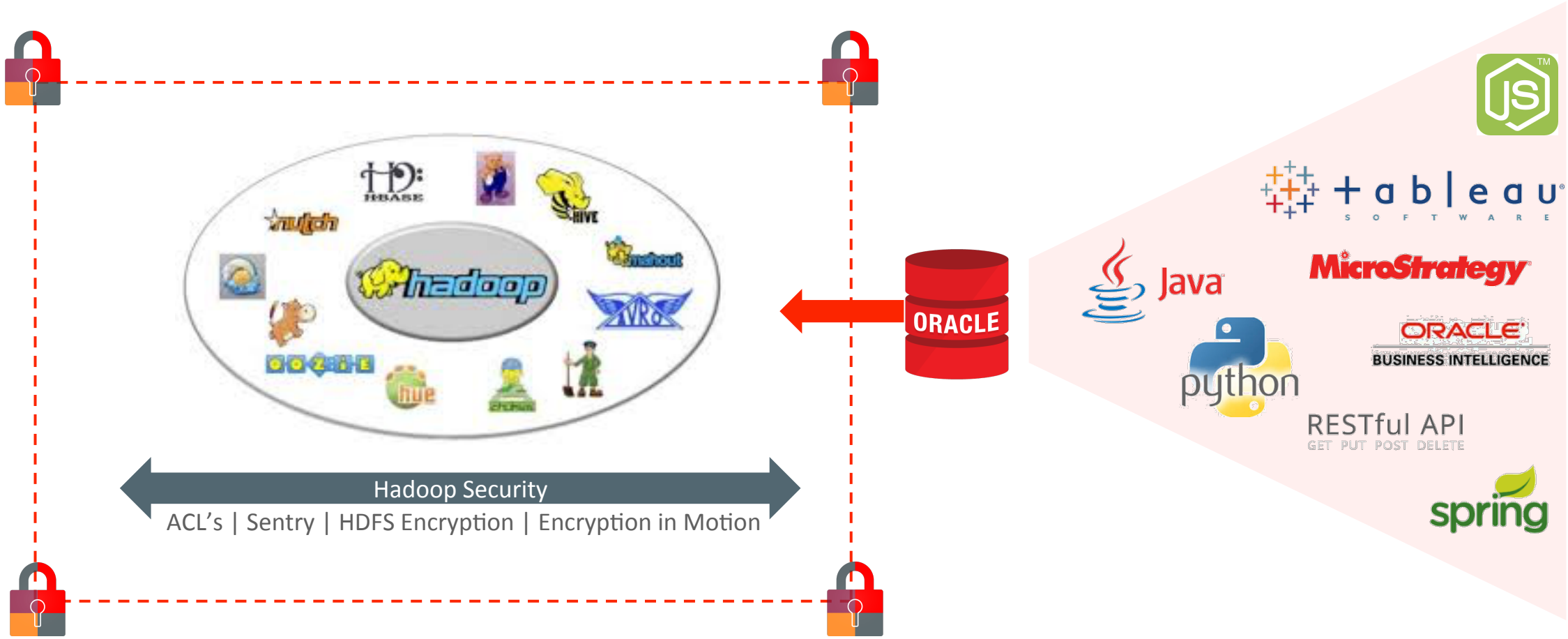


Automatic Query Rewrite

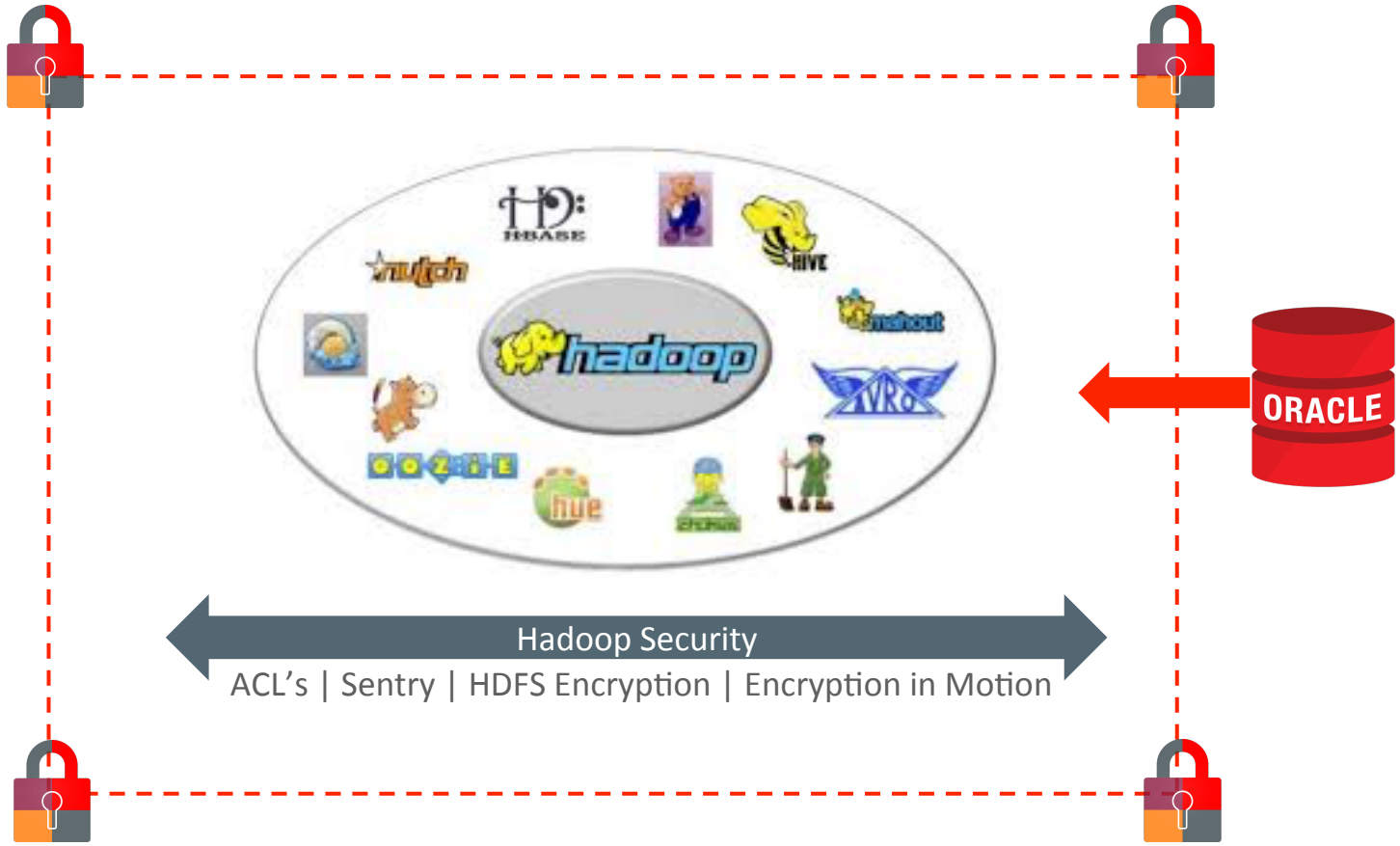


- **Materialized view** query rewrite automatically redirects detail query to appropriate summary data
  - Store summaries in Oracle Database
  - Use existing summaries in HDFS
- No changes to query required
- **Orders of magnitude** performance improvement

Big Data SQL Security Features

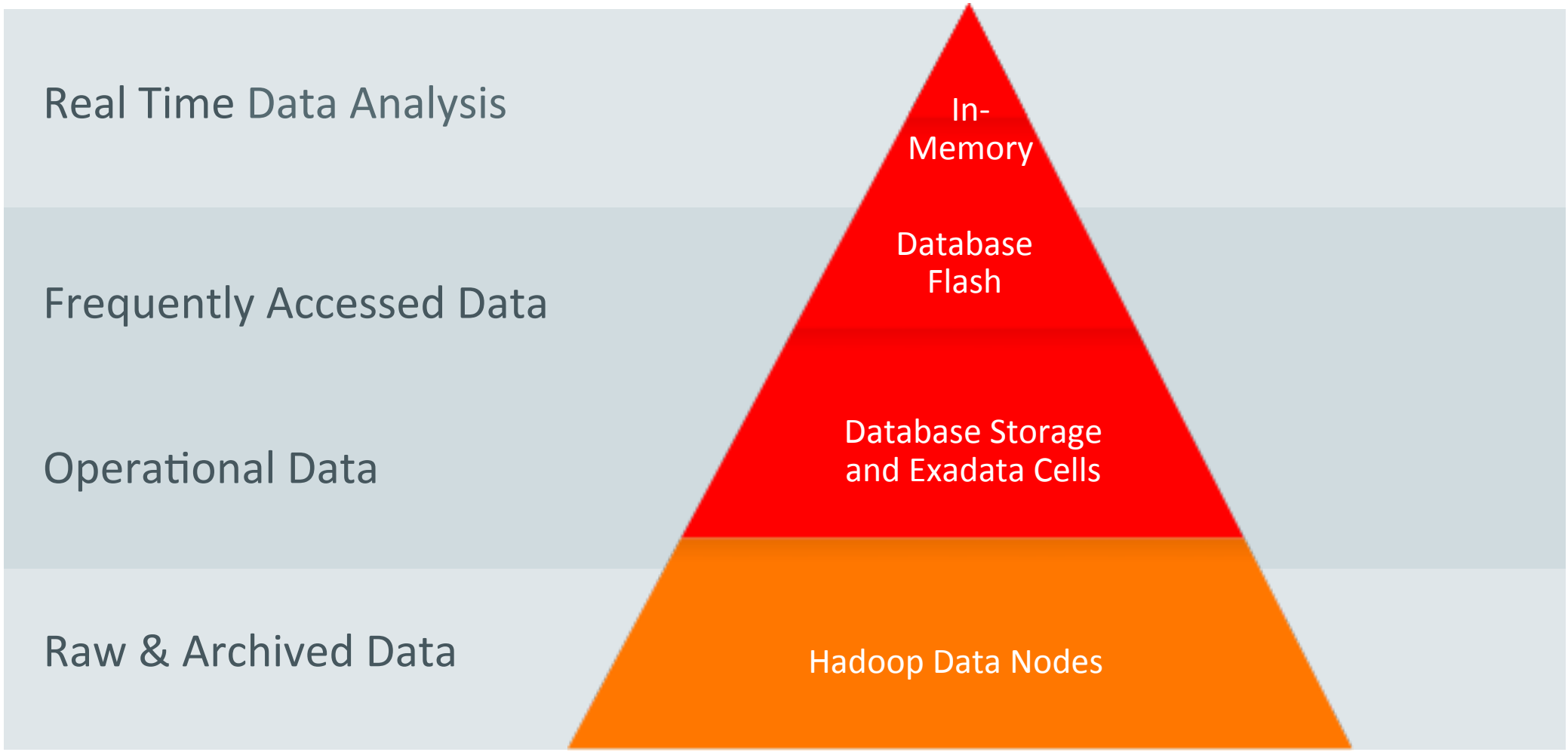


Big Data SQL Security Features

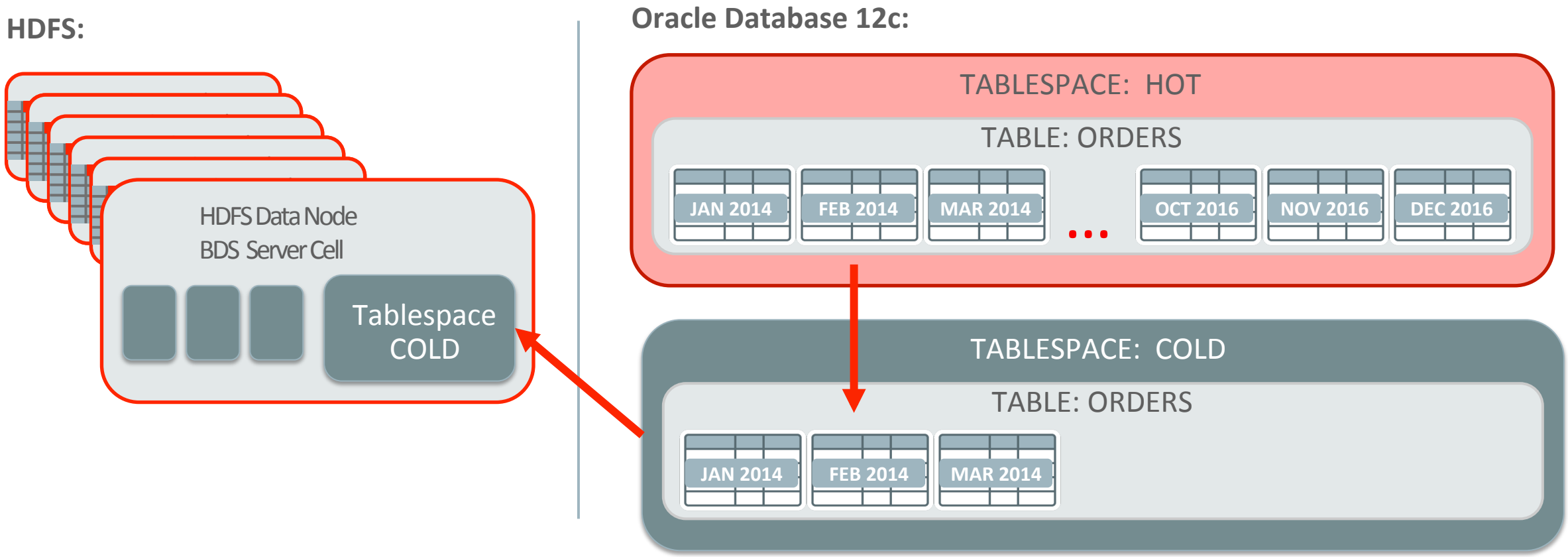


- Same security models apply to a wider range of data stores
- Advanced features such as data redaction can now be applied enabling joins between disparate sources
- Oracle security layers on top of existing Hadoop functionality

Big Data use case: Data Tiering



Archive Data to “Cold” Partitions

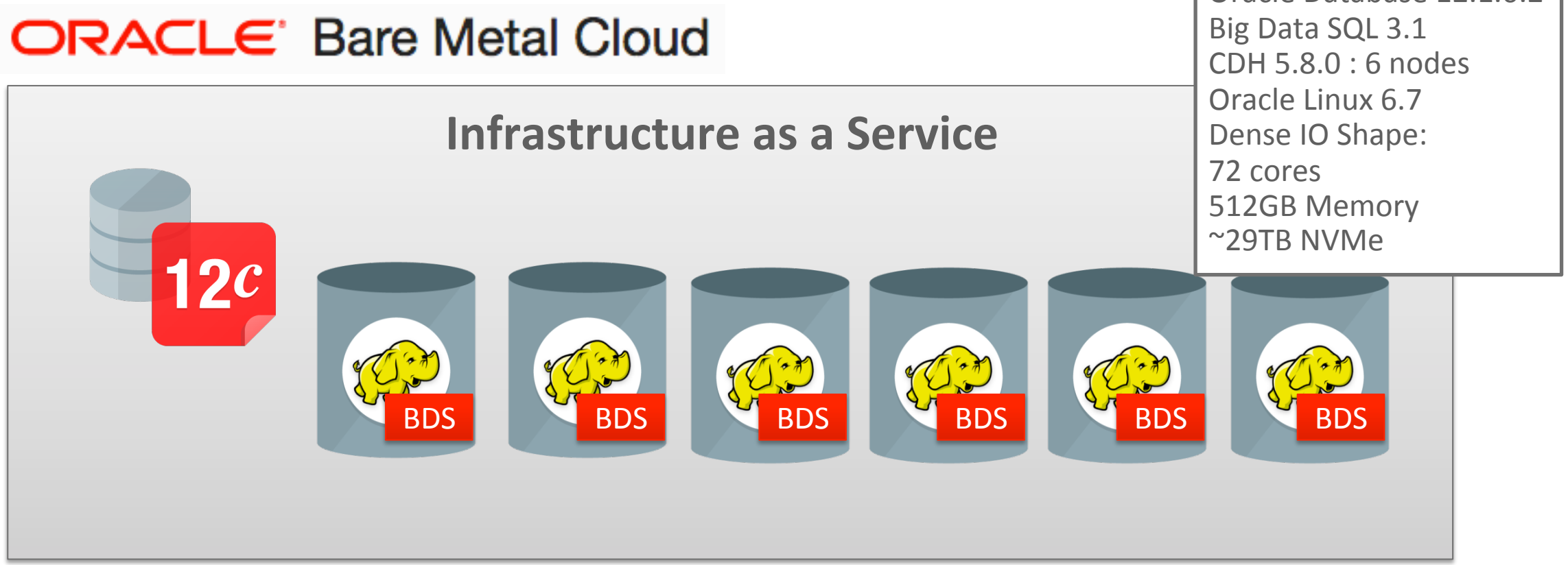


Big Data SQL Key Features

- **(Hive) Partition Pruning**
  - Read Hive catalog and prune partitions before query is run
  - 12.2 – Optimize partition pruning by leveraging Oracle Database metadata
  - Goal: IO Elimination
- **Storage Index**
  - Maintain metadata on elements to mark blocks for IO skipping
  - Goal: IO Elimination
- **Smart Scan**
  - Final filtering pass to ensure only requested elements are sent to Oracle Database
  - Goal: Data Movement Elimination
- **Bloom Filtering**
  - Convert joins into Bloom filter and push down to Hadoop nodes
  - Goal: Join Optimization and Localization
- **Predicate and Column Projection Pushdown**
  - Push down query elements into files like Parquet and ORC
  - Goal: IO Elimination on optimized file formats
- **Security**
  - Apply Oracle Database security policies on non-Oracle data stores
  - Goal: Enable Advanced Security features

Workshop Environment

Environment

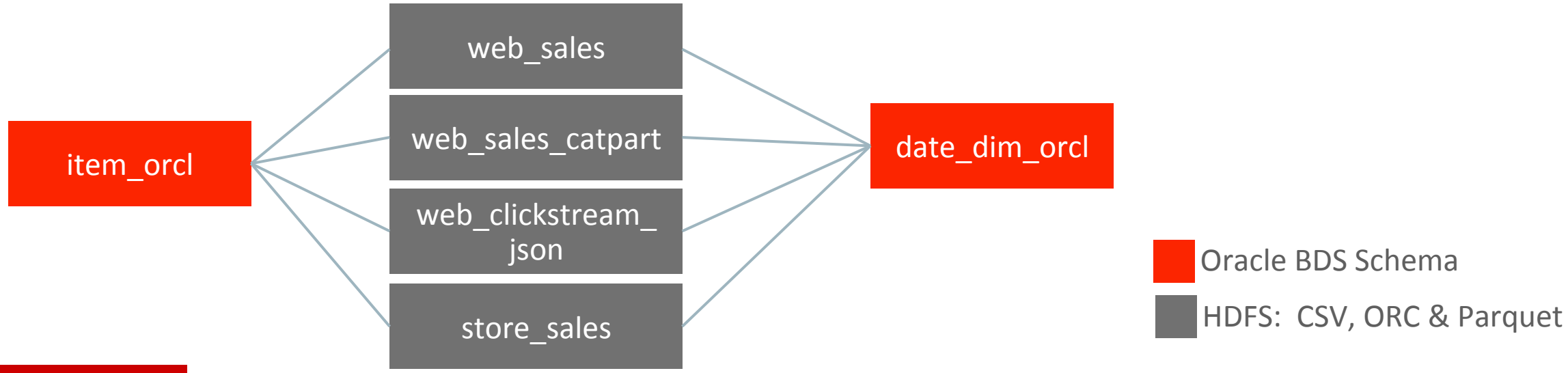


Fun Facts:

Oracle Database 12.1.0.2
Big Data SQL 3.1
CDH 5.8.0 : 6 nodes
Oracle Linux 6.7
Dense IO Shape:
72 cores
512GB Memory
~29TB NVMe

Schema

- TPC-DS Data
  - Over 20 tables in Hive - we will focus on a few:
    store_sales, web_sales, date_dim, item, web_clickstream_json, web_sales_catpart
  - Lookup tables have been created in bds schema:
    - date_dim_orcl, item_orcl, customer_orcl



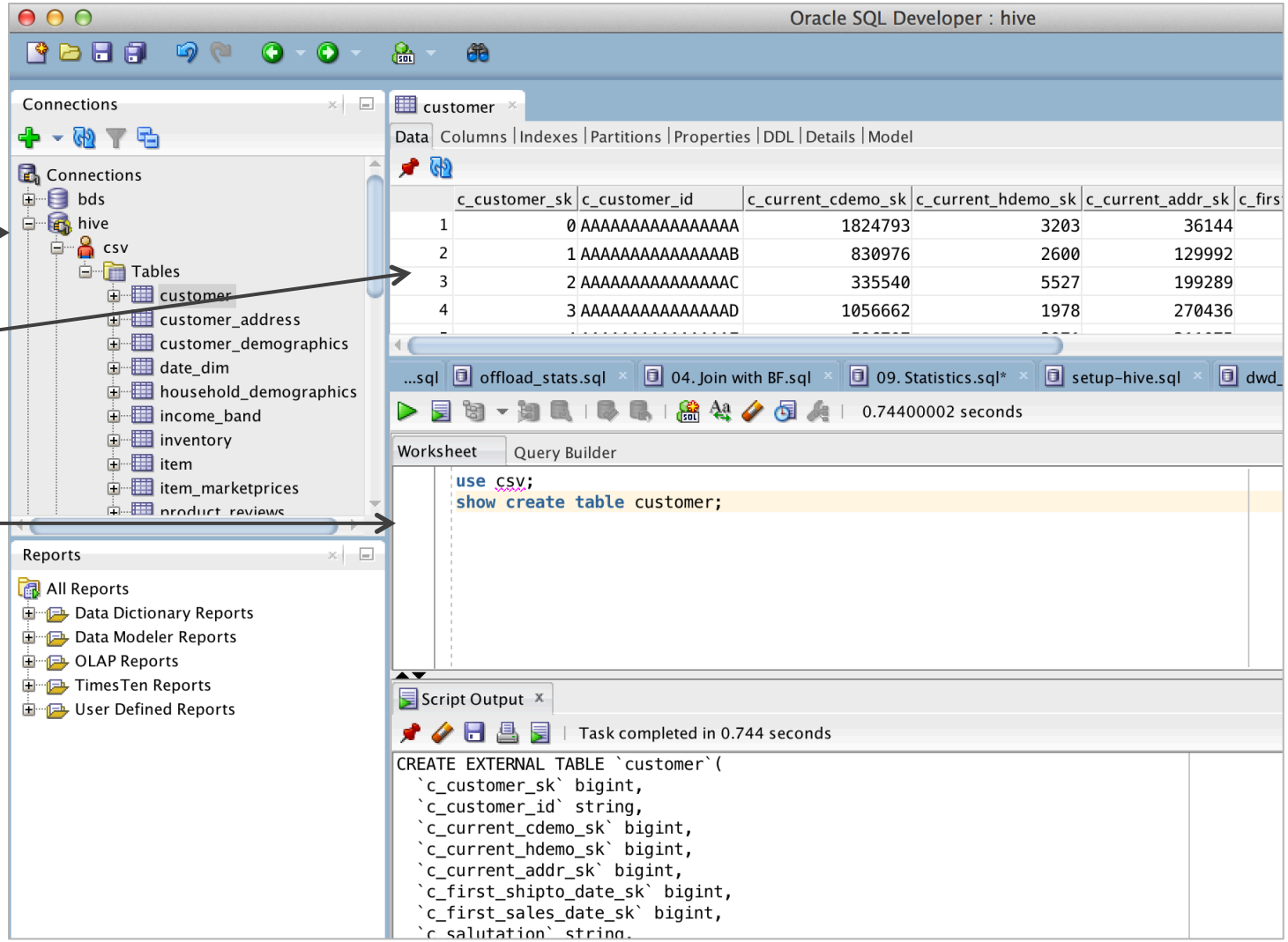
SQL Developer

Hive JDBC Support

Hive Connection

Hive object browser

Hive SQL Worksheet

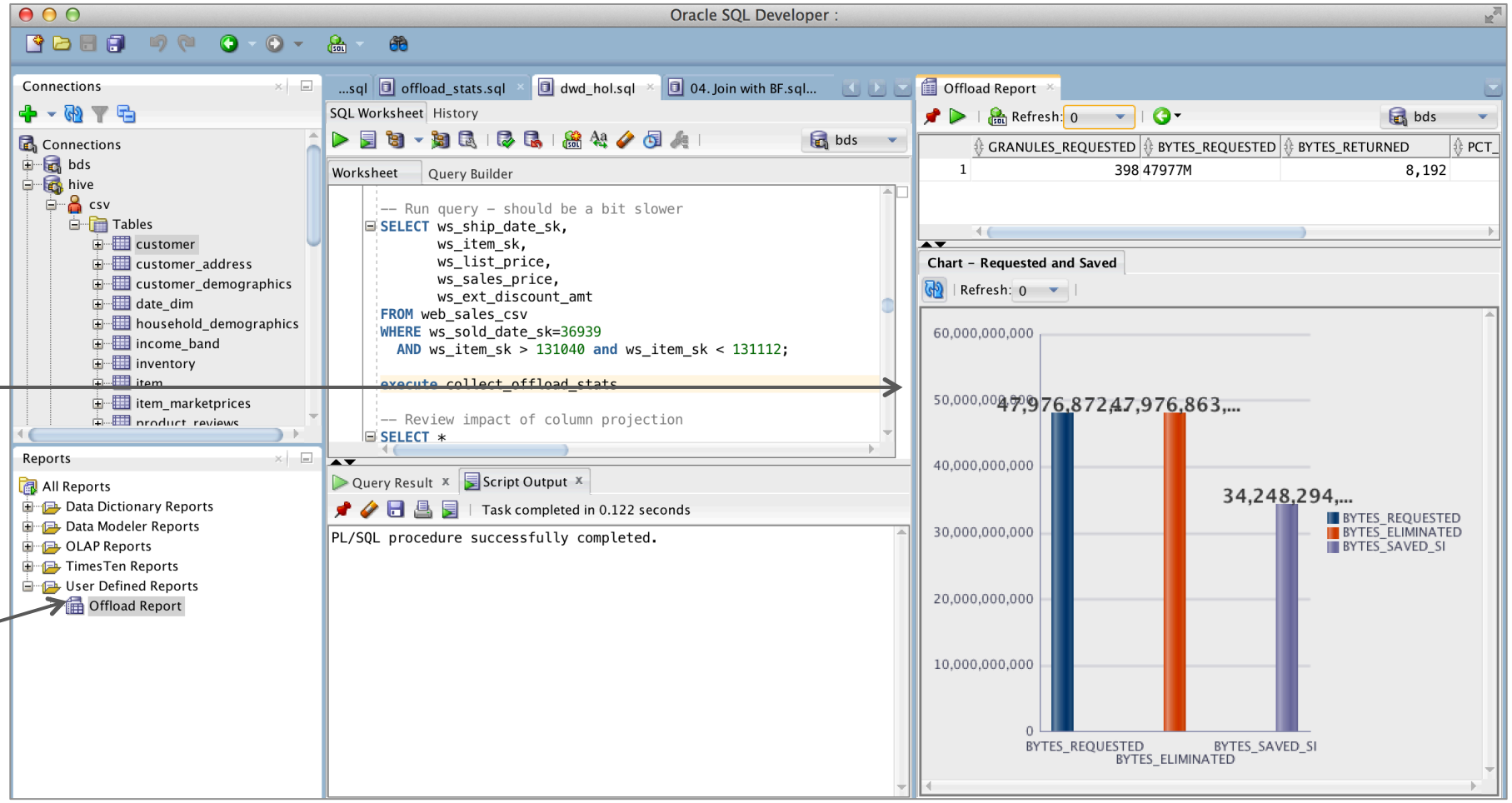


SQL Developer

Offload Report

2 Add report to SQL Developer

1 Import offload-report.xml



Part 1: Creating Tables

Metadata: Extend Oracle External Tables

```
CREATE TABLE movielog (
  click VARCHAR2(4000))
ORGANIZATION EXTERNAL (
  TYPE ORACLE_HIVE
  DEFAULT DIRECTORY DEFAULT_DIR
  ACCESS PARAMETERS
  (
com.oracle.bigdata.tablename logs
com.oracle.bigdata.cluster mycluster
  ))
REJECT LIMIT UNLIMITED;
```


- New types of external tables
  - ORACLE_HIVE (leverage hive metadata)
  - ORACLE_HDFS (specify metadata)
- Access parameters used to describe how to identify sources and process data on the hadoop cluster

Access Parameters: HDFS Example

```
CREATE TABLE WEB_SALES_CSV
(
  WS_SOLD_DATE_SK NUMBER
, WS_SOLD_TIME_SK NUMBER
, WS_ITEM_SK NUMBER
)
ORGANIZATION EXTERNAL
(
  TYPE ORACLE_HDFS
  DEFAULT DIRECTORY DEFAULT_DIR
  ACCESS PARAMETERS
  (
    com.oracle.bigdata.cluster=orabig
    com.oracle.bigdata.fileformat=TEXTFILE
    com.oracle.bigdata.rowformat: DELIMITED FIELDS TERMINATED BY '|'
    com.oracle.bigdata.erroropt: {"action": "replace", "value": "-1"}
  )
  LOCATION ('/data/tpcds/benchmarks/bigbench/data/web_sales')
)
REJECT LIMIT UNLIMITED;
```

- Access Parameters describe source data and processing rules
- Schema-on-Read

Access Parameters: ORACLE_HIVE

```
CREATE TABLE WEB_SALES_CSV
(
  WS_SOLD_DATE_SK NUMBER
, WS_SOLD_TIME_SK NUMBER
, WS_ITEM_SK NUMBER
)
ORGANIZATION EXTERNAL
(
  TYPE ORACLE_HIVE
  DEFAULT DIRECTORY DEFAULT_DIR
  ACCESS PARAMETERS
  (
    com.oracle.bigdata.cluster=orabig
    com.oracle.bigdata.tablename: csv.web_sales
   com.oracle.bigdata.erroropt: {"action": "replace", "value": "-1"}
    com.oracle.bigdata.datamode=automatic
  )
REJECT LIMIT UNLIMITED;
```

- Access Parameters refer to metadata in Hive
- Add processing rules

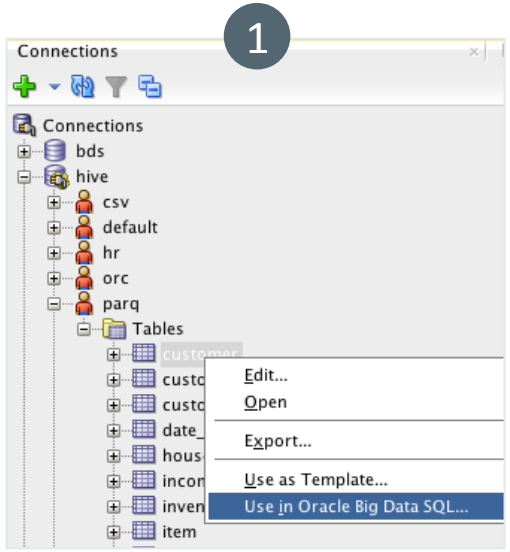
Recommended Approach

Use ORACLE_HIVE When Possible

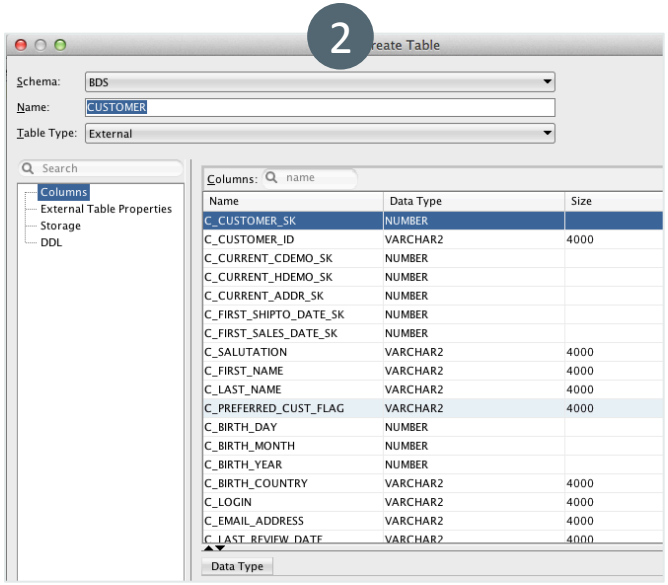
- Oracle Database query execution accesses Hive metadata at describe time
  - Changes to underlying Hive access parameters will not impact Oracle table (one exception… column list)
- Metadata an enabler for performance optimizations
  - Partition pruning and predicate pushdown into intelligent sources
- Utilize tooling for simplified table definitions
  - SQL Developer and DBMS_HADOOP packages

Creating Tables

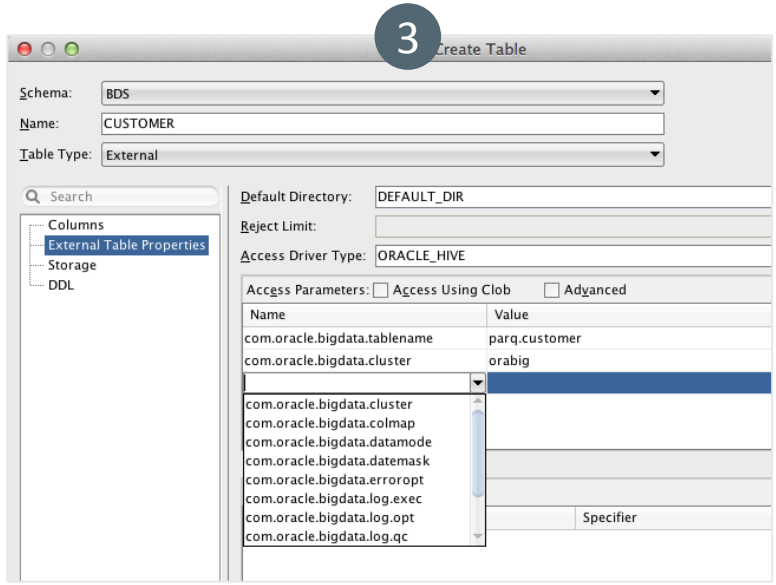
SQL Developer with Hive JDBC



Right-click on Hive Table. **Use in Oracle Big Data SQL**



Review generated columns. Update as needed - focusing on data types and precision



Add optional access parameters. Automatically generate table or save DDL.

See: https://blogs.oracle.com/datawarehousing/entry/oracle_sql_developer_data_modeler

Viewing Hive Metadata from Oracle Database

- ALL_HIVE_DATABASES, ALL_HIVE_TABLES, ALL_HIVE_COLUMNS

ALL_HIVE_COLUMNS

CLUSTER_ID	DATABASE_NAME	TABLE_NAME	COLUMN_NAME	HIVE_COLUMN_TYPE	ORACLE_COLUMN_TYPE	LOCATI
orabig	csv	customer	c_customer_sk	bigint	NUMBER	hdfs://h
orabig	csv	customer	c_customer_id	string	VARCHAR2(4000)	hdfs://h
orabig	csv	customer	c_current_cdemo_sk	bigint	NUMBER	hdfs://h
orabig	csv	customer	c_current_hdemo_sk	bigint	NUMBER	hdfs://h
orabig	csv	customer	c_current_addr_sk	bigint	NUMBER	hdfs://h
orabig	csv	customer	c_first_shipto_date_sk	bigint	NUMBER	hdfs://h
orabig	csv	customer	c_first_sales_date_sk	bigint	NUMBER	hdfs://h
orabig	csv	customer	c_salutation	string	VARCHAR2(4000)	hdfs://h
orabig	csv	customer	c_first_name	string	VARCHAR2(4000)	hdfs://h
orabig	csv	customer	c_last_name	string	VARCHAR2(4000)	hdfs://h
orabig	csv	customer	c_preferred_cust_flag	string	VARCHAR2(4000)	hdfs://h
orabig	csv	customer	c_birth_day	int	NUMBER	hdfs://h
orabig	csv	customer	c_birth_month	int	NUMBER	hdfs://h
orabig	csv	customer	c_birth_year	int	NUMBER	hdfs://h
orabig	csv	customer	c_birth_country	string	VARCHAR2(4000)	hdfs://h
orabig	csv	customer	c_login	string	VARCHAR2(4000)	hdfs://h
orabig	csv	customer	c_email_address	string	VARCHAR2(4000)	hdfs://h
orabig	csv	customer	c_last_review_date	string	VARCHAR2(4000)	hdfs://h
orabig	csv	customer_address	ca_address_sk	bigint	NUMBER	hdfs://h
orabig	csv	customer_address	ca_address_id	string	VARCHAR2(4000)	hdfs://h

Creating Tables

DBMS_HADOOP Package

```
declare
  DDLout VARCHAR2(4000);
begin
  DDLout := null;

  dbms_hadoop.create_extddl_for_hive (
      CLUSTER_ID=> 'orabig',
      DB_NAME=> 'parq',
      HIVE_TABLE_NAME=> 'store_sales',
      HIVE_PARTITION=>FALSE,
      TABLE_NAME=> 'store_sales_orcl',
      PERFORM_DDL=>FALSE,
      TEXT_OF_DDL=>DDLout
     );

  dbms_output.put_line(DDLout);
end;
/
```

- PL/SQL Package used to create table or generate DDL
- Combine with ALL_HIVE* dictionary views to automate creation of many tables
- Consider optimizing data type conversions - especially precision
  - string -> varchar2(?)

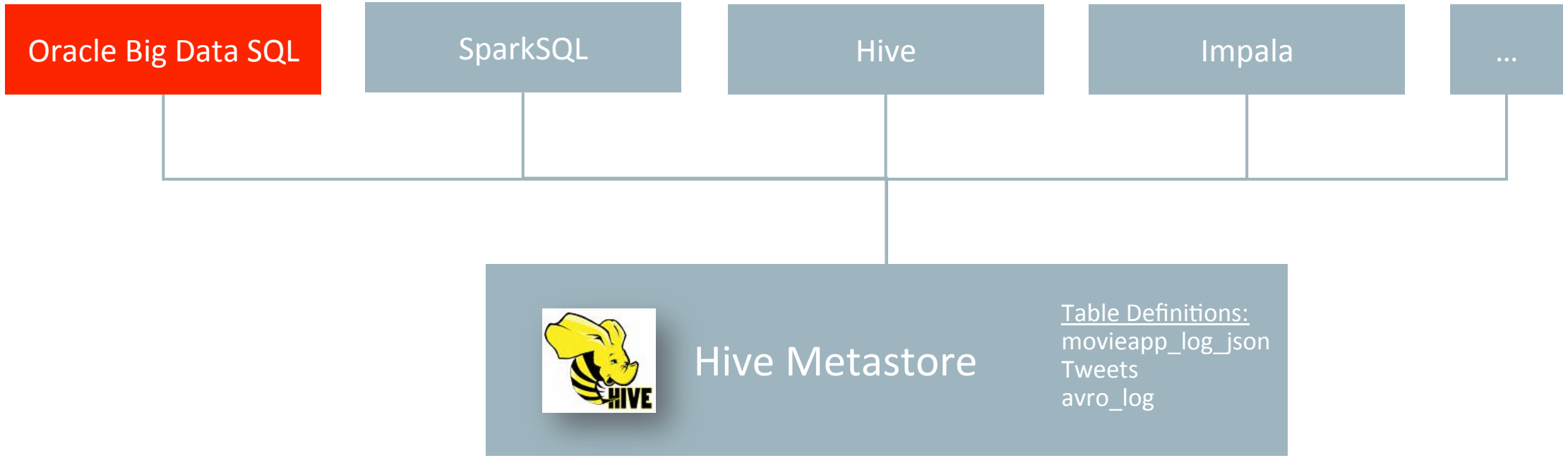
Part 1: Hands-on

Creating Tables

Part 2: Performance Features

SQL-on-Hadoop Engines Share Metadata, not MapReduce

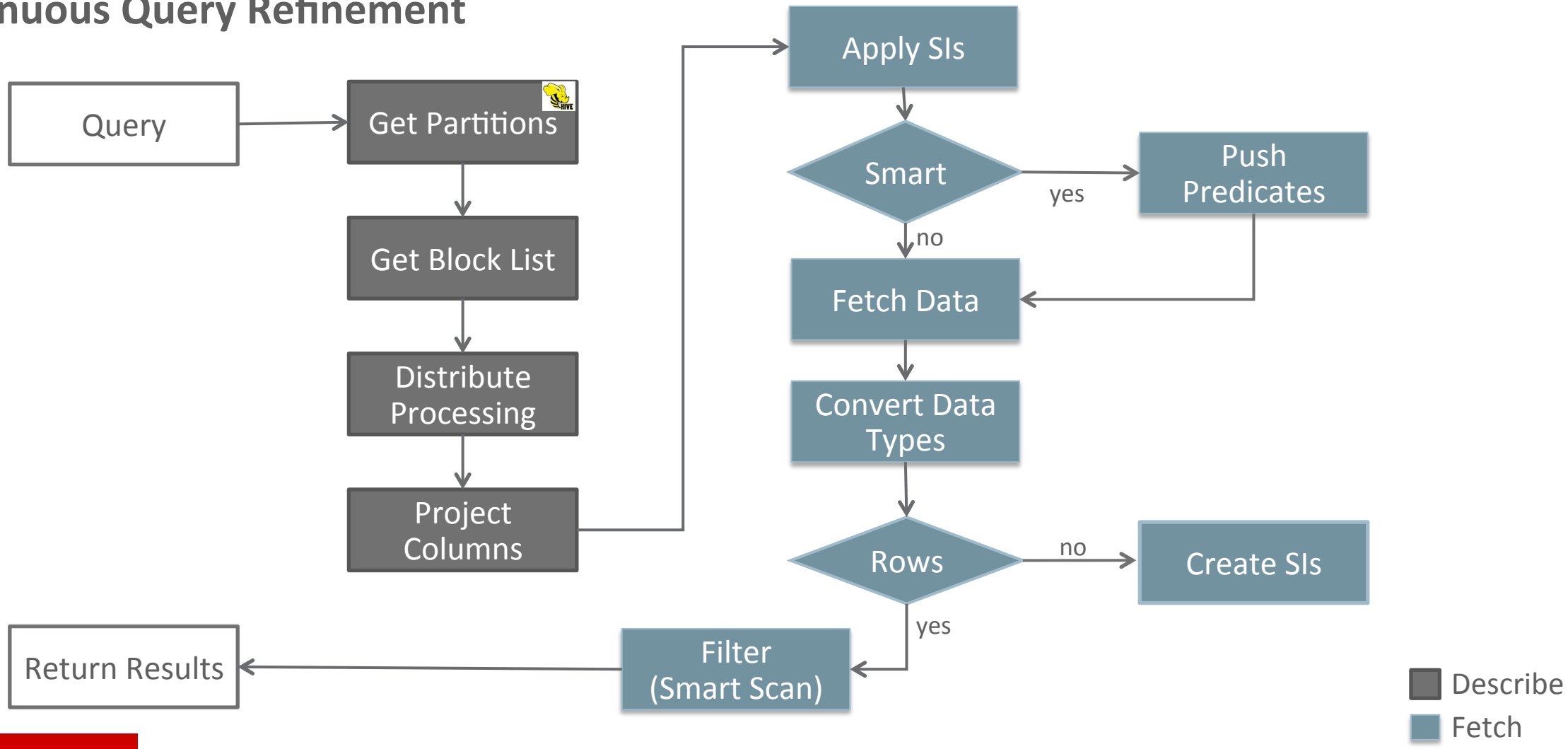
Hive Metastore



Metastore maps DDL to Java access classes

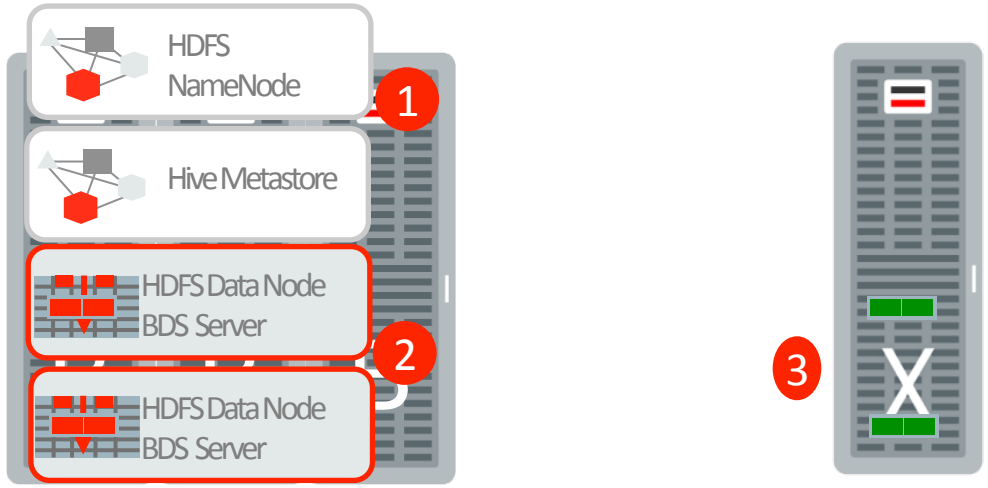
Big Data SQL Query Processing

Continuous Query Refinement



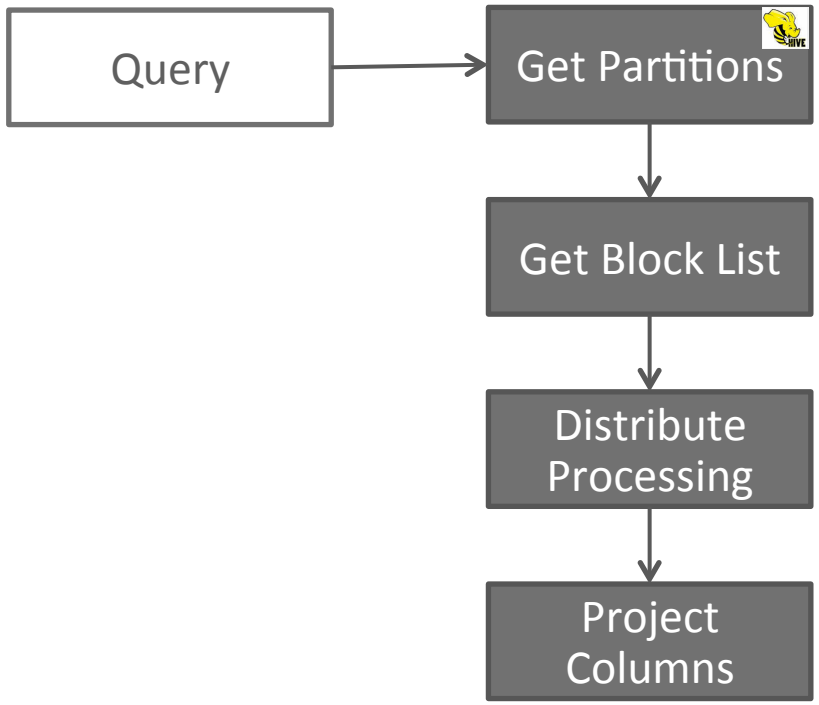
Big Data SQL Query Execution

How do we query Hadoop?



1. Describe time determines:
   - Data locations (partition pruned)
   - Data structure
   - Parallelism
2. Fast reads using Big Data SQL Server
   - Schema-for-read using Hadoop classes
   - Predicate pushdown for intelligent sources
   - Smart Scan selects only relevant data
3. Process filtered result
   - Move relevant data to database
   - Join with database tables
   - Apply database security policies

Partition Pruning and a Big Data SQL Query



Partition pruning reduces the number of partitions that are going to be scanned

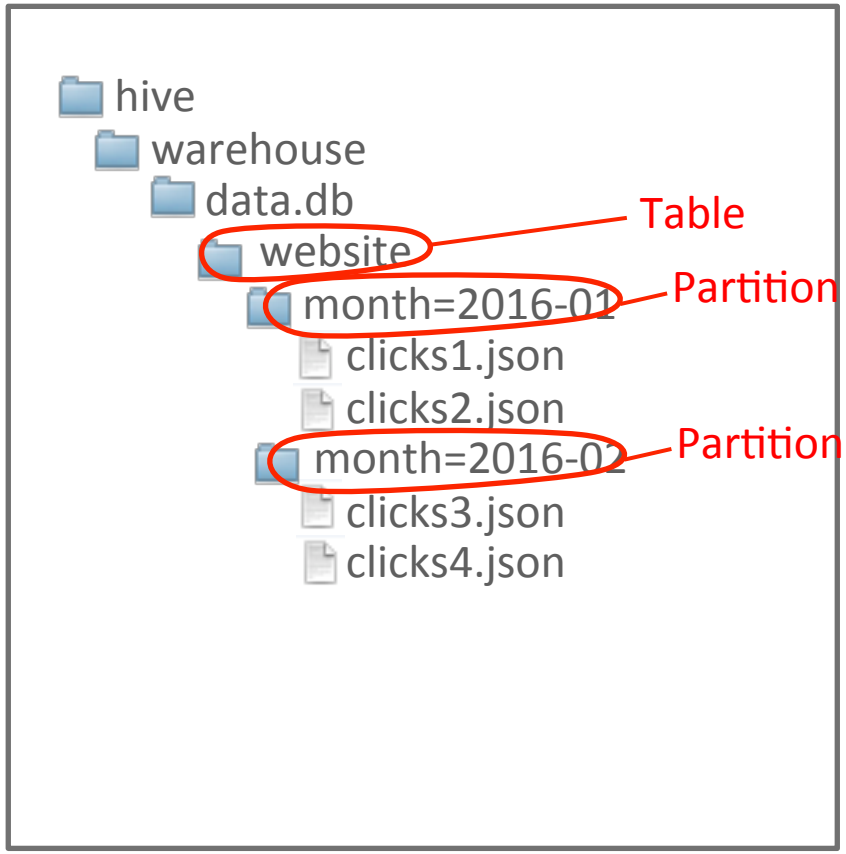
Only retrieve block list for relevant partitions

IO Reduction

Describe

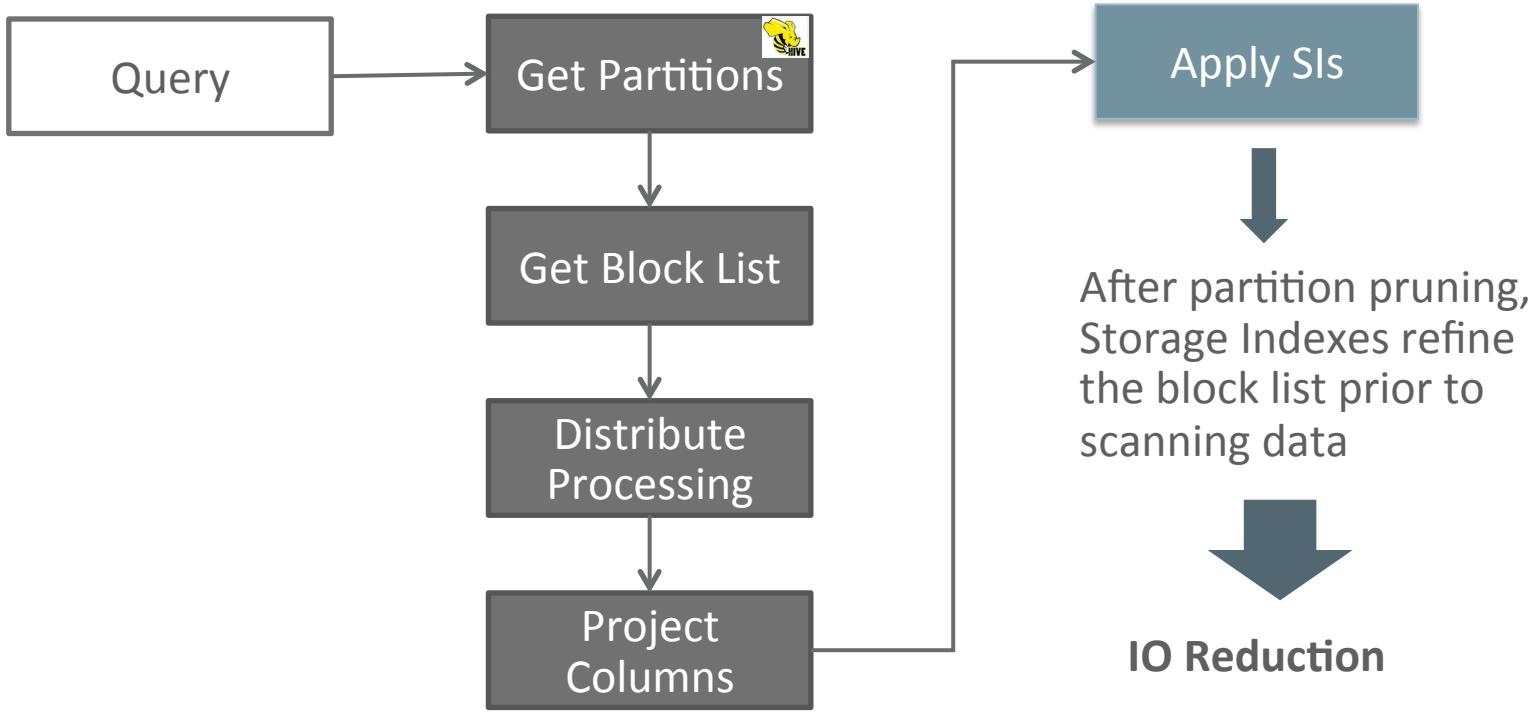
Fetch

Partition Pruning



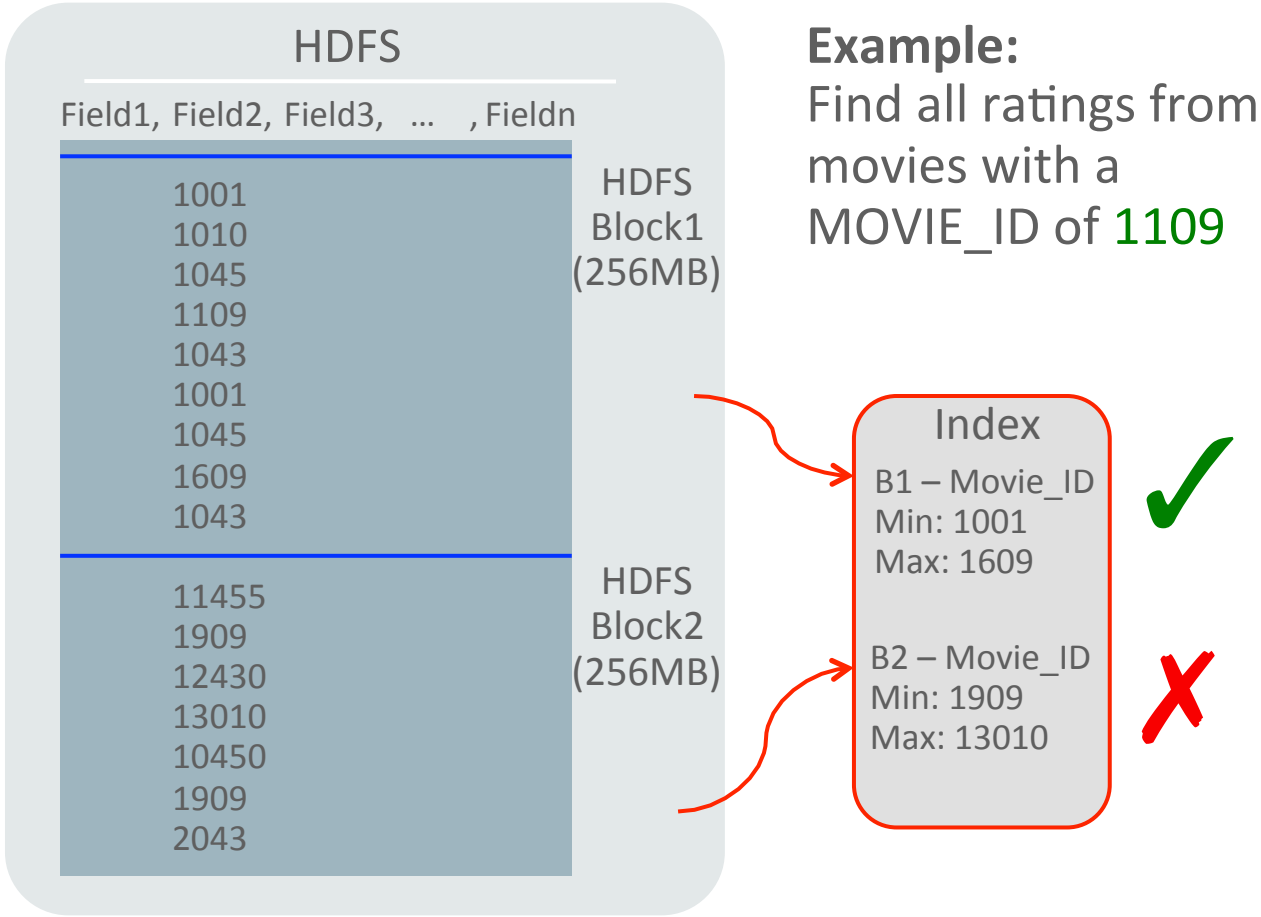
- Optimization only available for Hive
  - HDFS does not have partition concept. Files identified by extab LOCATION clause will be scanned
- Only those partitions (i.e. folders) that meet condition will be scanned
- The following predicates are supported:
  =; !=; <; <=; >; >=, LIKE, AND and OR

Storage Indexes and a Big Data SQL Query



Describe
Fetch

Oracle Big Data SQL Storage Index



Example:
Find all ratings from movies with a MOVIE_ID of 1109

- Storage index provides query speed-up through transparent IO elimination of HDFS Blocks. It's a *negative index*
- Columns in SQL are mapped to fields in the HDFS file via External Table Definitions
- Min / max value is recorded for each HDFS Block in a storage index

Big Data SQL Storage Index

Things to note and know

- Indexes up to 32 columns per external table
- SI on external tables defined with Storage Handlers (HBase, NoSQL Database) is not supported
- If multiple external tables address the same HDFS file, multiple SIs are maintained on a per external table basis
- SI is local to the Hadoop node, if a query runs on a different node, no SI may be available resulting in slower performance until the SI is built on that node
- Invalidation of an SI is automatic, it is based on a check on the underlying files at runtime
- SI works on number, character and date data types
- **If possible, sort data to maximize SI query performance benefit**

Influencing Storage Index Creation

- SI is generated after a block is queried and no result data is found for that block
- Columns in "where" clause used to prioritize SI target fields
- Warm up SI by running queries
  - Run multiple times to hit all replicas
  - Or, simply let user query workload generate SIs on demand

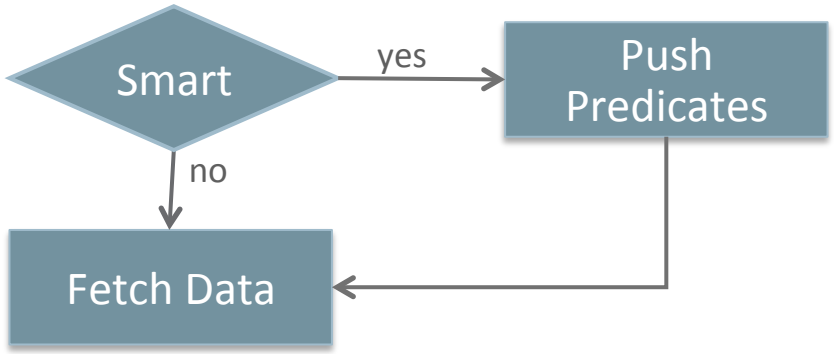
Example "warm-up" query:

```
SELECT count(*)
FROM WEB_SALES_CSV
WHERE
  WS_SOLD_DATE_SK = -1
  AND WS_SOLD_TIME_SK = -1
  AND WS_SHIP_DATE_SK = -1
  AND WS_ITEM_SK = -1
  AND WS_BILL_CUSTOMER_SK = -1
  AND WS_SHIP_CUSTOMER_SK = -1
  AND WS_WEB_PAGE_SK = -1
  AND WS_WEB_SITE_SK = -1
  AND WS_SHIP_MODE_SK = -1
  AND WS_WAREHOUSE_SK = -1
  AND WS_PROMO_SK = -1
  AND WS_ORDER_NUMBER = -1
  AND WS_QUANTITY = -1;
```

Blocks:
No column has a value with -1; all blocks will be have SIs

Columns:
"Where" cols will have SIs

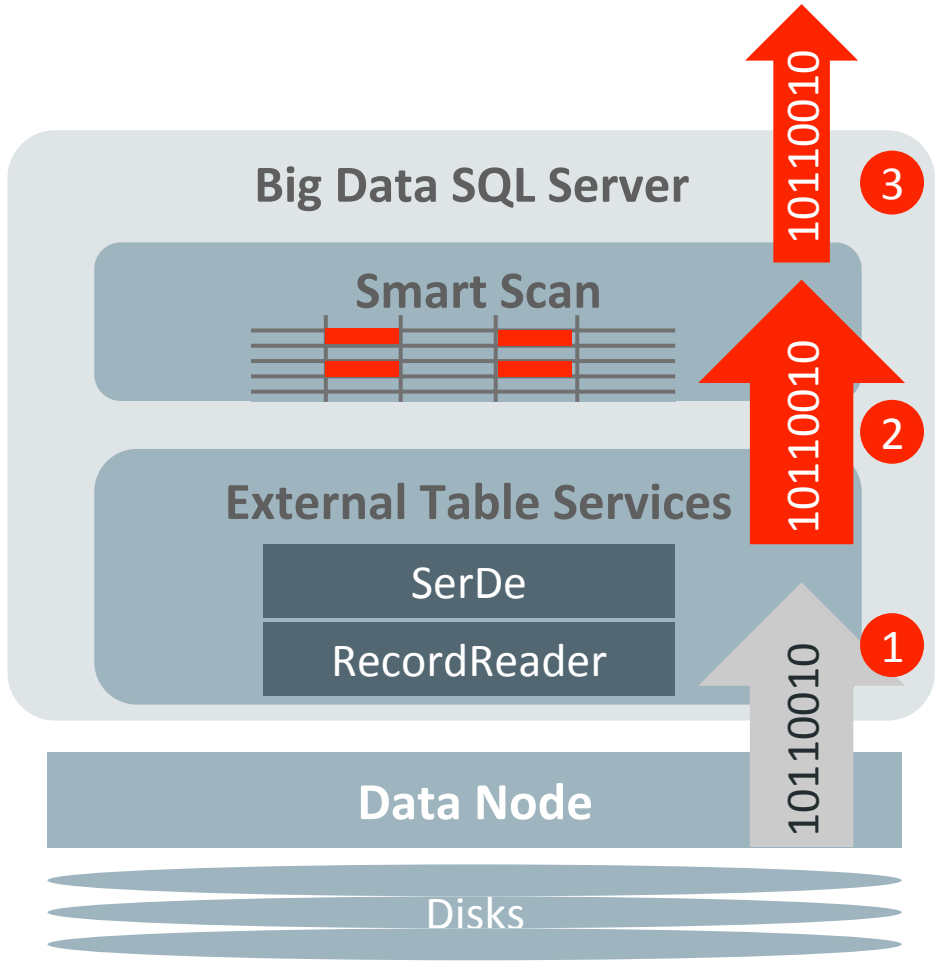
Predicate Pushdown and a Big Data SQL Query



Query predicates are pushed to intelligent data sources (like Parquet and ORC) which use columnar formats and metadata to eliminate IO
only reading the subset of data requested

IO Reduction

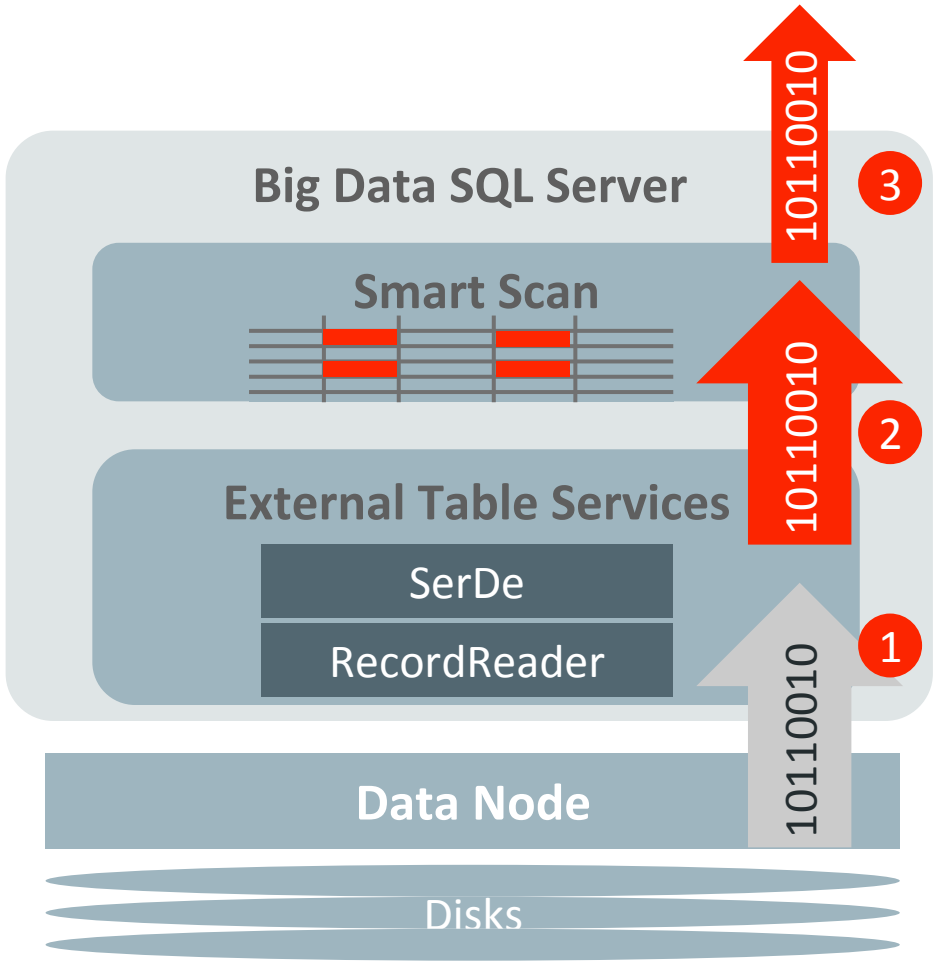
Big Data SQL Server Dataflow



1. Read data from HDFS Data Node
   - Direct-path reads
   - C-based readers when possible
   - Use native Hadoop classes otherwise
   - Push predicates to intelligent sources
2. Translate bytes to Oracle format
3. Apply Smart Scan to Oracle bytes
   - Apply filters
   - Project Columns
   - Parse JSON/XML
   - Score models

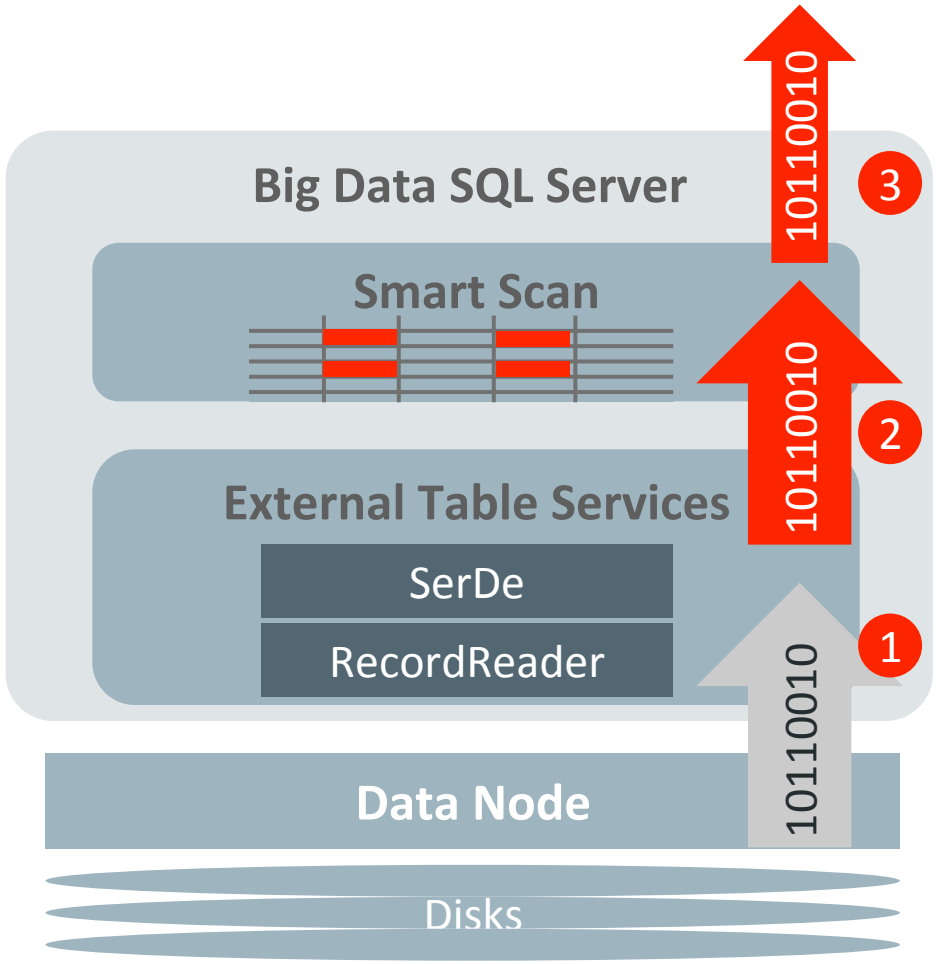
ORACLE

Data Types: Parquet, ORC and AVRO files



- Parquet, ORC and AVRO store data in native types; Oracle Database has its own data types
- Minimize "conversion" by matching data types
  - Matching types enables simple hand-off
  - Example:
    Map Hive FLOAT to Oracle BINARY_FLOAT
    Map Hive DOUBLE to Oracle BINARY_DOUBLE
- Pay attention to precision - esp. with Strings
- Compatibility matrix between **hive** data types and **Oracle** data types are found: http://docs.oracle.com/bigdata/bda45/BDSUG/concepts.htm#BIGUG21117

Data Types: Text file



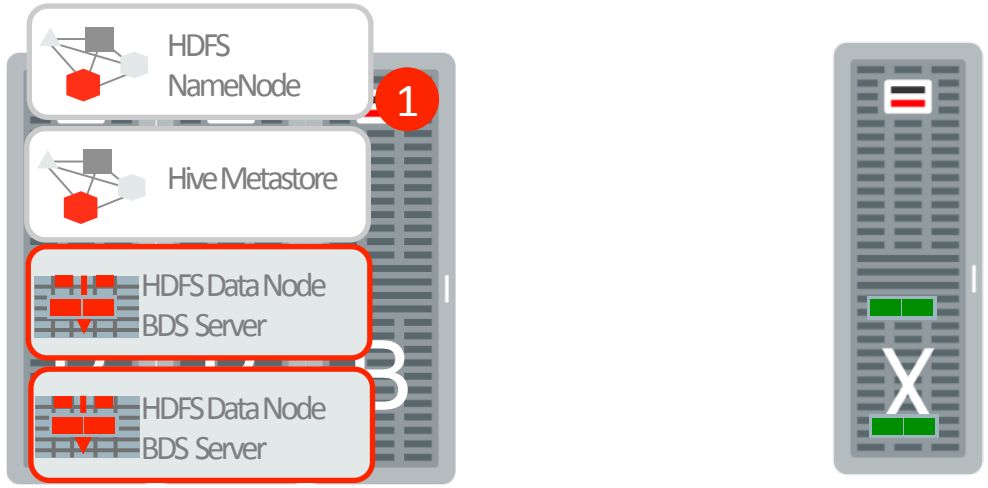
- Every record is a string - conversion must take place for each field
- Whenever possible, use Oracle Database lightweight column types, like NUMBER, instead BINARY_DOUBLE
- Pay attention to precision - esp. with Strings

Pay Attention to Data Types

- Data is converted from Java types to Oracle Data types. Want this conversion to be as efficient as possible when converting 100s of columns and billions of rows!
- CSV files in C-Mode
  - Avoid mapping to BINARY_FLOAT or BINARY_DOUBLE. It's extremely expensive. Map to Oracle NUMBER if possible.
- Java-Mode - it's the reverse recommendation!
  - File's InputFormat has already converted from the file storage format to Hive data type
  - For Parquet, ORC, AVRO/binary and RCFile (and all the NoSQL DB's), Hive FLOATs and DOUBLEs are stored in binary format, so need to convert from file storage format (binary IEEE FLOATs and DOUBLEs) to Oracle BINARY_FLOAT and BINARY_DOUBLE
  - Map Hive FLOAT to Oracle BINARY_FLOAT and Hive DOUBLE to Oracle BINARY_DOUBLE

Big Data SQL Query Execution

ORACLE_HDFS



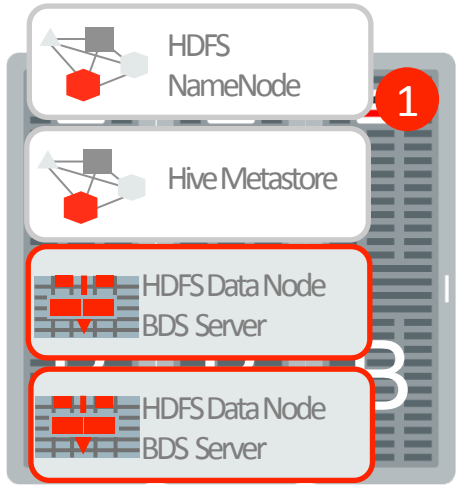
1 Describe time determines:

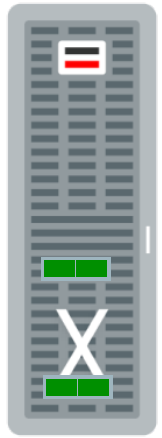
- Data locations

- External Table LOCATION field specifies where data resides
- Get splits/blocks/hosts that contain the data blocks for the associated files
- Database query coordinator distribute work to BDS Cells across the cluster - prioritizing data local scans
- Storage Indexes filter blocks
- Data converted to Oracle format for selected cols
- Smart Scan filters rows and returns selected cols

Big Data SQL Query Execution

ORACLE_HIVE





1 Describe time determines:

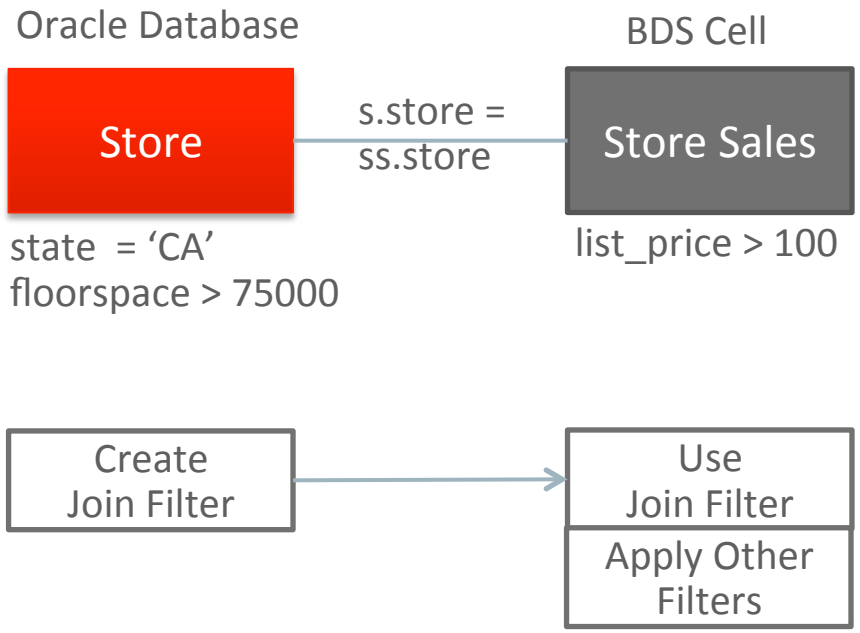
- Data locations

- Access parameter identifies hive source
- Retrieve from Hive Metastore:
  - List of partitions that meet the query condition
  - Access methods for underlying data files
- Get splits/blocks/hosts that contain the data blocks for the associated files
- Database query coordinator distributes work to BDS Cells across the cluster - prioritizing data local scans
- Storage Indexes filter blocks
- Predicates pushed to “intelligent” sources
- Data converted to Oracle format for selected cols
- Smart Scan filters rows and returns selected cols

Join Optimization with Bloom Filters

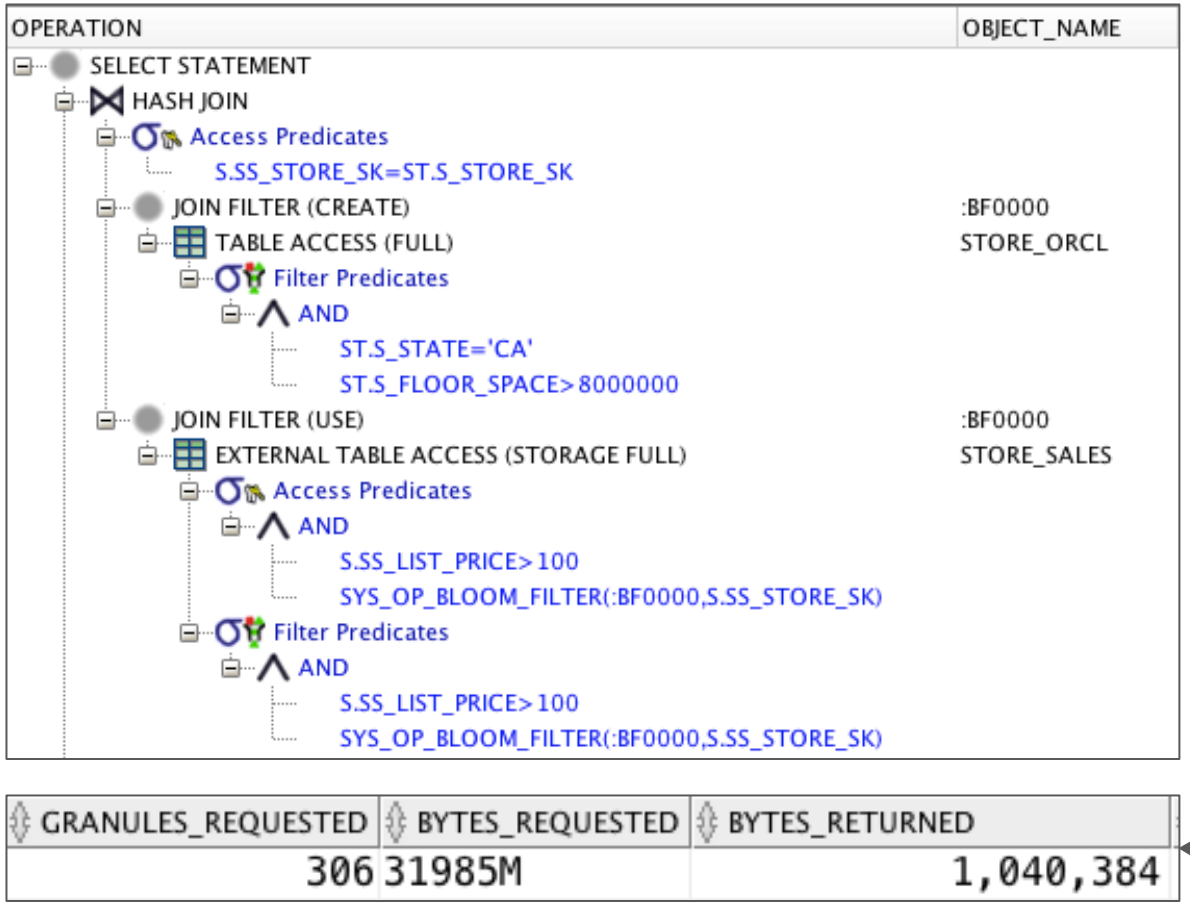
- A Bloom filter…
  - is a low-memory data structure that tests membership in a set
  - correctly indicates when an element is *not* in a set. There could be false positives
  - is used to filter data in BDS cells - especially when joining large facts to small dimension tables
- Optimizer will automatically utilize Bloom filters to improve performance
  - Can be influenced by PX_JOIN_FILTER/NO_PX_JOIN_FILTER hints

Join Optimization with Bloom Filters



- Query joins store and store sales table
- Bloom filter (bit vector) is created based on the join column
- Bloom filter & other filters are applied on the BDS Cell
  - Is this store in the bit vector?
- Potential massive reduction in data returned to database for join

Join Optimization Example:



OPERATION	OBJECT_NAME
SELECT STATEMENT	
HASH JOIN	
Access Predicates	
S.SS_STORE_SK=ST.S_STORE_SK	
JOIN FILTER (CREATE)	:BF0000
TABLE ACCESS (FULL)	STORE_ORCL
Filter Predicates	
AND	
ST.S_STATE='CA'	
ST.S_FLOOR_SPACE>8000000	
JOIN FILTER (USE)	:BF0000
EXTERNAL TABLE ACCESS (STORAGE FULL)	STORE_SALES
Access Predicates	
AND	
S.SS_LIST_PRICE>100	
SYS_OP_BLOOM_FILTER(:BF0000,S.SS_STORE_SK)	
Filter Predicates	
AND	
S.SS_LIST_PRICE>100	
SYS_OP_BLOOM_FILTER(:BF0000,S.SS_STORE_SK)	

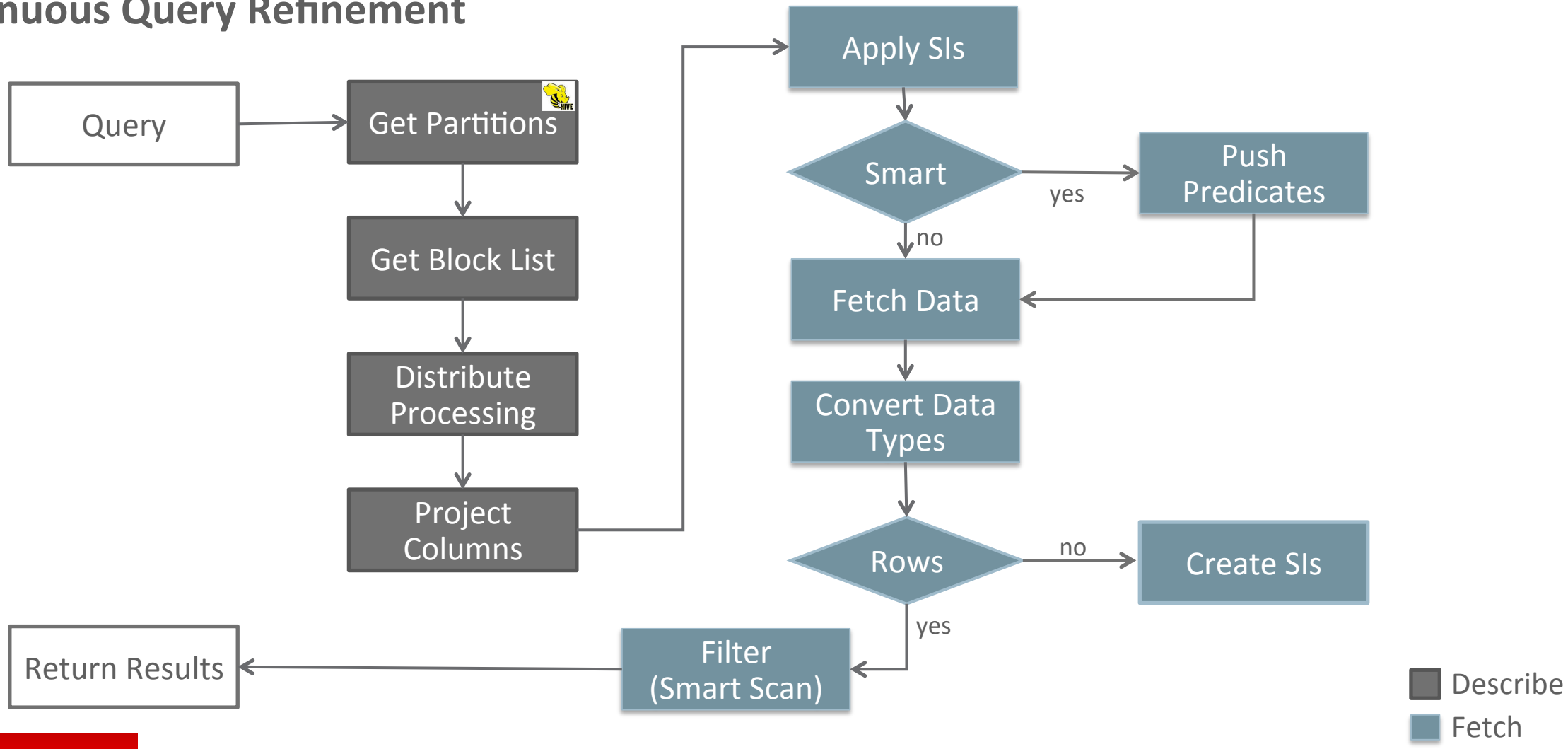
GRANULES_REQUESTED	BYTES_REQUESTED	BYTES_RETURNED
306	31985M	1,040,384

← Smart Scan Filter: 99%

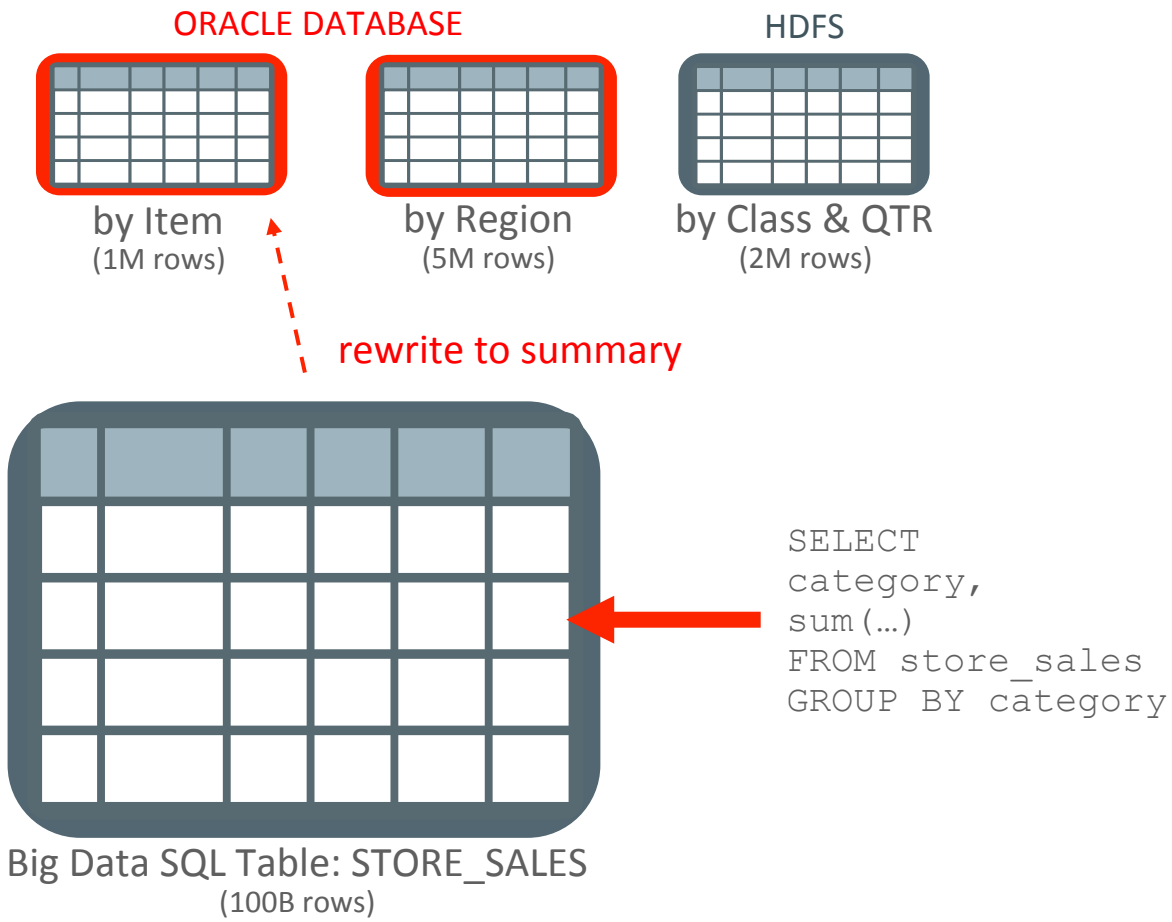
```
SELECT
      st.s_manager,
      st.s_hours,
      s.ss_list_price,
      s.ss_sales_price,
      s.ss_ext_discount_amt
FROM store_sales s, store_orcl st
WHERE s.ss_store_sk = st.s_store_sk
  AND st.s_state='CA'
  AND st.s_floor_space > 8000000
  AND s.ss_list_price > 100;
```

Big Data SQL Query Processing Summary

Continuous Query Refinement

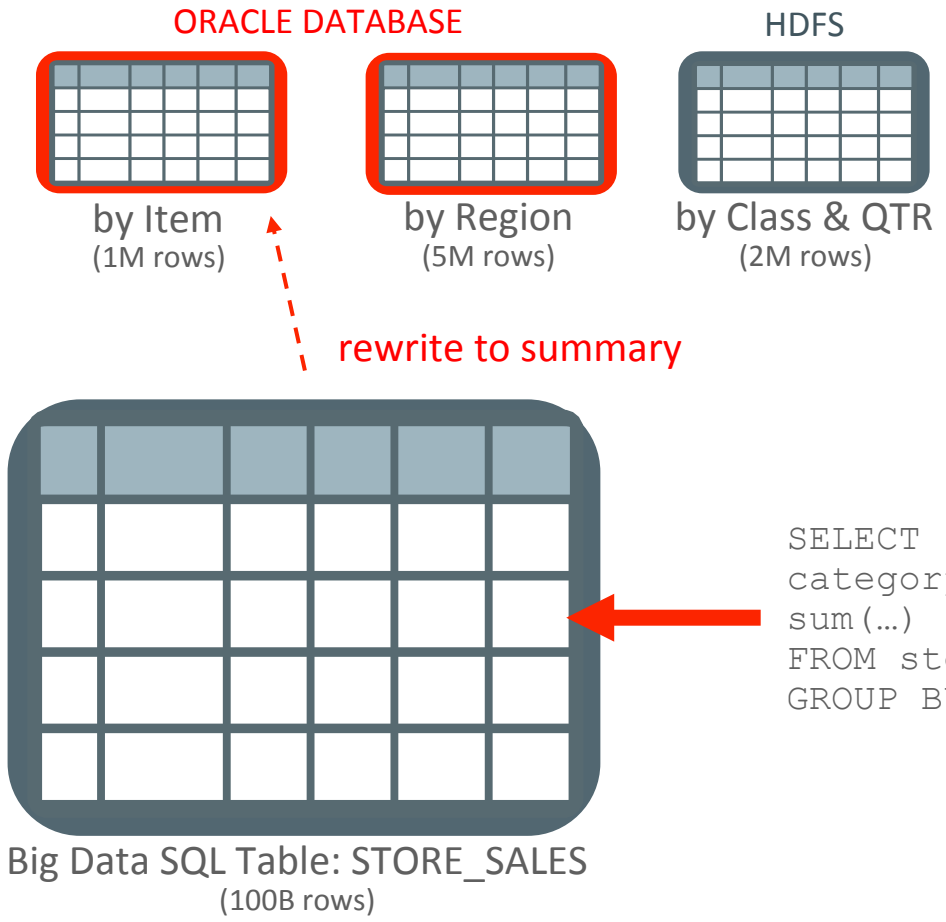


Enhance Performance with Automatic Query Rewrite



- **Orders of magnitude** performance improvement
- **Materialized view** query rewrite automatically redirects detail query to appropriate summary data
  - Store summaries in Oracle Database
  - If available, use existing summaries in HDFS
- No changes to query required

Create Materialized View Over BDS External Tables

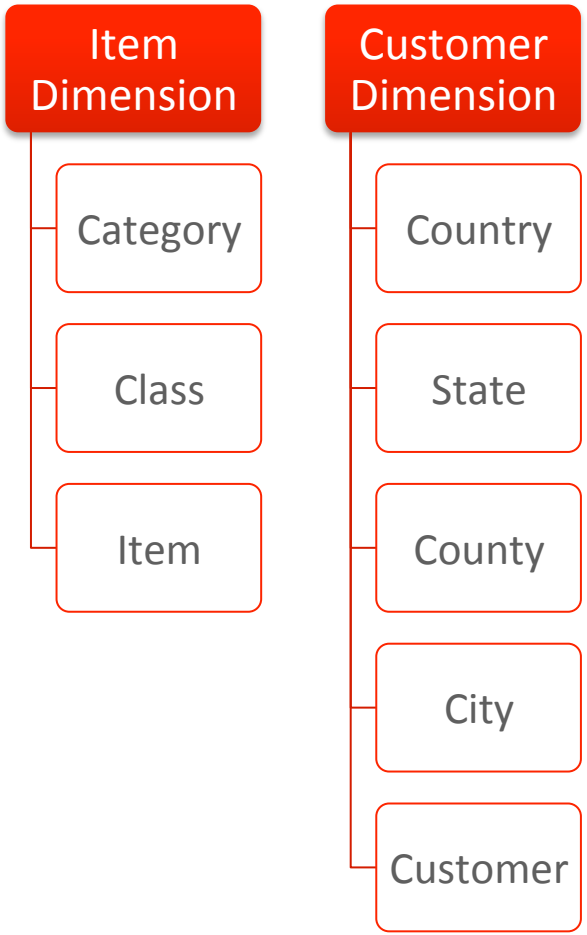


- Optimizer picks best summary to match query
- MV will need to be manually maintained
- Set query_rewrite_integrity to stale_tolerated
- Remember to collect statistics!

Example: MV by Item

```
CREATE MATERIALIZED VIEW mv_store_sales_item
ON PREBUILT TABLE
ENABLE QUERY REWRITE AS (
  select ss_item_sk,
         sum(ss_quantity) as ss_quantity,
         sum(ss_ext_wholesale_cost) as
             ss_ext_wholesale_cost,
         sum(ss_net_paid) as ss_net_paid,
         sum(ss_net_profit) as ss_net_profit
  from bds.store_sales
  group by ss_item_sk
);
```

Add Metadata Describing Hierarchies



- Use dimension objects to enhance rewrite effectiveness
- Hierarchies define column relationships

```
CREATE DIMENSION BDS.ITEM_DIM
  LEVEL ITEM IS (ITEM_ORCL.I_ITEM_SK)
  LEVEL CLASS IS (ITEM_ORCL.I_CLASS)
  LEVEL CATEGORY IS (ITEM_ORCL.I_CATEGORY)
HIERARCHY PROD_ROLLUP (
  ITEM CHILD OF
  CLASS CHILD OF
  CATEGORY  )
ATTRIBUTE ITEM DETERMINES (
  ITEM_ORCL.I_SIZE, ITEM_ORCL.I_COLOR,
ITEM_ORCL.I_UNITS,
  ITEM_ORCL.I_CURRENT_PRICE,I_WHOLESALE_COST
  );
```

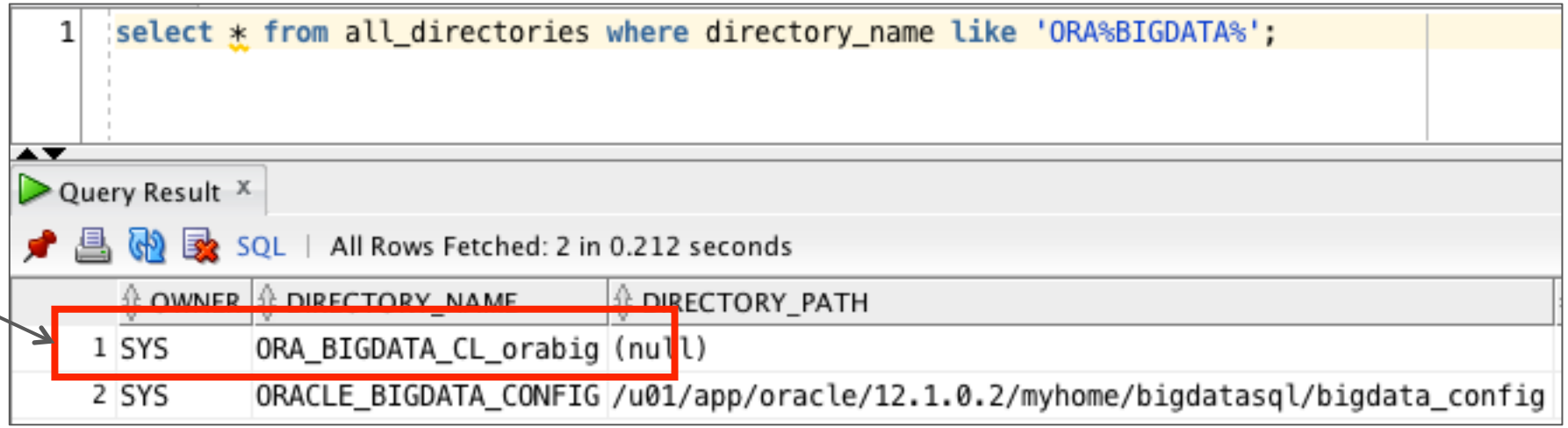
Security: Overview

- Use Oracle security policies to access data
  - GRANT BDSQL_USER role to all Big Data SQL users
  - GRANT access to cluster
  - GRANT access to tables
  - Apply fine-grained policies to table data

Cluster Access

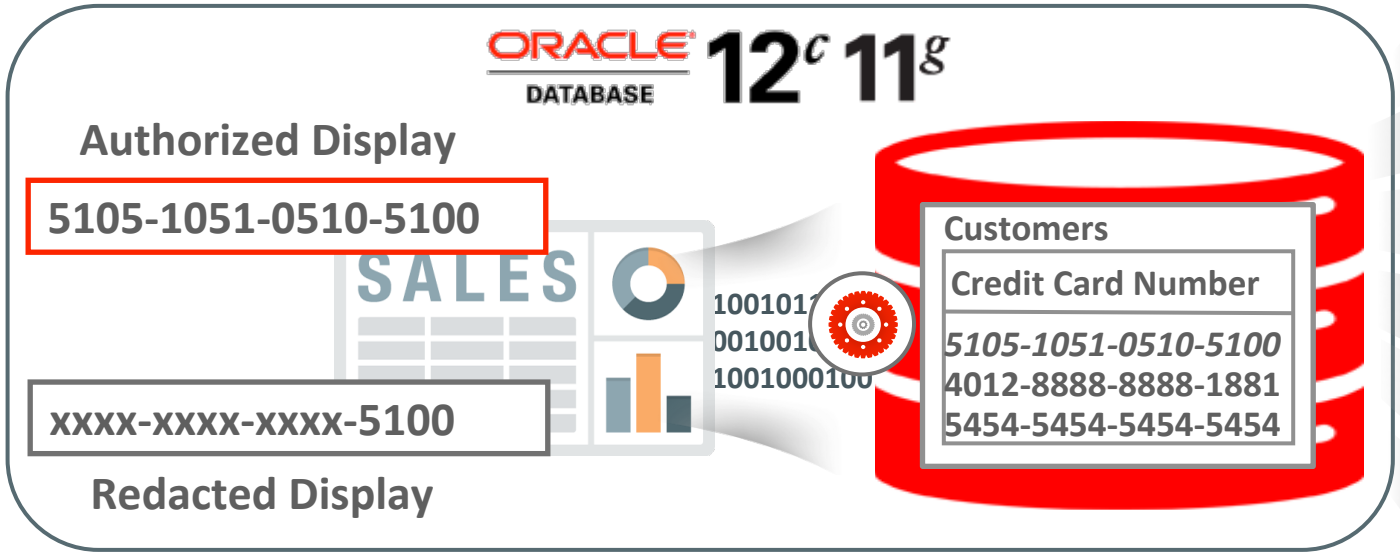
- Oracle directory objects represent cluster
  - ORA_BIGDATA_CL_<cluster>
- GRANT SELECT on cluster directory to users/groups to allow coarse grained access to data in cluster

“orabig” cluster



Oracle Data Redaction *(Dynamic Data Masking)*

Redacting Sensitive Data for Applications



- Real-time and accurate redaction
- Comprehensive Transformations
- Near zero impact on production *
- No application changes required *
- Flexible GUI and scripting options
- Pre-installed in Oracle Database

* In most cases

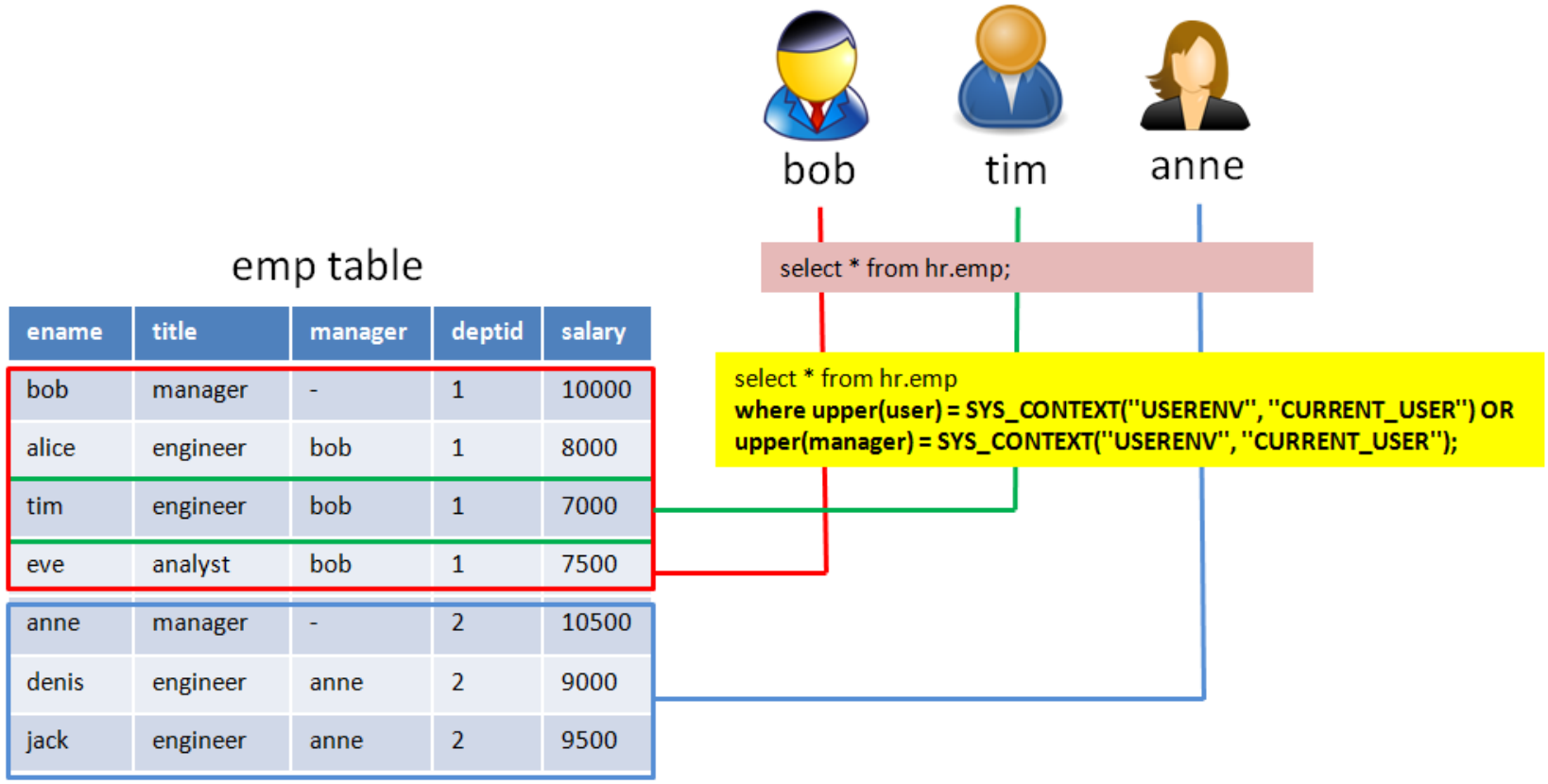
Oracle Data Redaction. Use case

```
BEGIN
SQL>  set wrap off
SQL>  set LINESIZE 150
SQL>   SELECT c_first_name,
         c_last_name,
         c_birth_country,
         c_email_address
    FROM bds.customer_csv t
   WHERE ROWNUM <4;  2    3    4    5    6

C_FIRST_NAME         C_LAST_NAME                    C_BIRTH_COUNTRY      C_EMAIL_ADDRESS
-------------------- ------------------------------ -------------------- ----------------------
Michael              Leal                           CAMEROON             xxxx@SoftHome.net
Christina            Vaughn                         MOZAMBIQUE           xxxx@netdove.net   .com
Marie                Sandoval                       ANGOLA               xxxx@gmx.fr        on.com

SQL>
END;
/
PL/SQL procedure successfully completed.
```

Use case:
Hide pre-domain part of the email

- Query data over the table without data masking policy
- Create policy for this table which hides part of the email
- Run query over the table which has Data Redaction policy

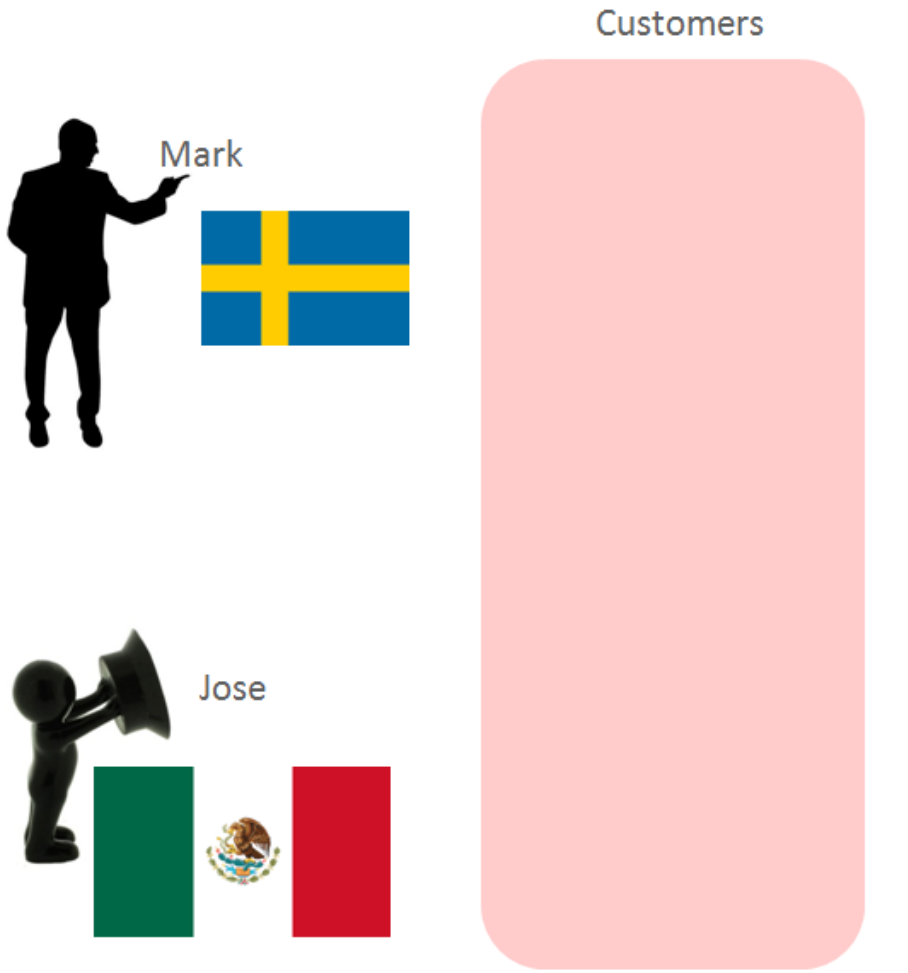
Virtual Private Databases



emp table

ename	title	manager	deptid	salary
bob	manager	-	1	10000
alice	engineer	bob	1	8000
tim	engineer	bob	1	7000
eve	analyst	bob	1	7500
anne	manager	-	2	10500
denis	engineer	anne	2	9000
jack	engineer	anne	2	9500

Virtual Private Databases



Use case:

- There is World Wide company which has customers all around the world
- Mark, who works in Sweden, have to see only Swedish customers
- Jose, who works in Mexico, have to see only Mexican customers

Virtual Private Databases. Creation of the packages

```
 CREATE OR REPLACE PACKAGE pck_vpd AS
   p_country_name employee.country_name%TYPE;
   PROCEDURE set_country(v_country_name
employee.country_name%TYPE);
   FUNCTION predicate(obj_schema VARCHAR2,
                      obj_name   VARCHAR2) RETURN VARCHAR2;
END pck_vpd;
....
  BEGIN
    IF p_country_name IS NOT NULL THEN
    return 'c_birth_country = ' ||''''|| p_country_name||'''';
    ELSE
      RETURN '1=1';
      END IF;
  end predicate;
end pck_vpd;
```

```
BEGIN
dbms_rls.add_policy(object_schema => 'bds',
  object_name     => 'CUSTOMER_CSV',
  policy_name     => 'CHOOSE_CONTRY',
  function_schema => 'BDS',
  policy_function => 'pck_vpd.predicate',
  statement_types => 'select,update,delete');
END;
```

Virtual Private Databases. Query the data

```
SQL> connect mark/welcome1;
Connected.
SQL> set wrap off
SQL> SELECT c_first_name,
         c_last_name,
         c_birth_country,
         c_email_address
    FROM bds.customer_csv t
   WHERE ROWNUM <4;  2    3    4    5    6

C_FIRST_NAME         C_LAST_NAME                    C_BIRTH_COUNTRY      C_EMAIL
-------------------- ------------------------------ -------------------- -------
Randy                Peacock                        SWEDEN               xxxx@vf
Susie                Sawyer                         SWEDEN               xxxx@st
William              Bohn                           SWEDEN               xxxx@gm

SQL>    SELECT COUNT(1) FROM bds.customer_csv t;

  COUNT(1)
----------
     33168

SQL> connect jose/welcome1
Connected.
SQL> SELECT c_first_name,
         c_last_name,
         c_birth_country,
         c_email_address
    FROM bds.customer_csv t
   WHERE ROWNUM <4;   2    3    4    5    6

C_FIRST_NAME         C_LAST_NAME                    C_BIRTH_COUNTRY      C_EMAIL
-------------------- ------------------------------ -------------------- -------
Deborah              Davis                          MEXICO               xxxx@ve
Katie                Daugherty                      MEXICO               xxxx@ne
Anne                 Crist                          MEXICO               xxxx@bu

SQL>  SELECT COUNT(1) FROM bds.customer_csv t;

  COUNT(1)
----------
     32939
```

- Whenever Mark connect to the database he is automatically see only Sweden customers

- Jose sees only Mexican customers

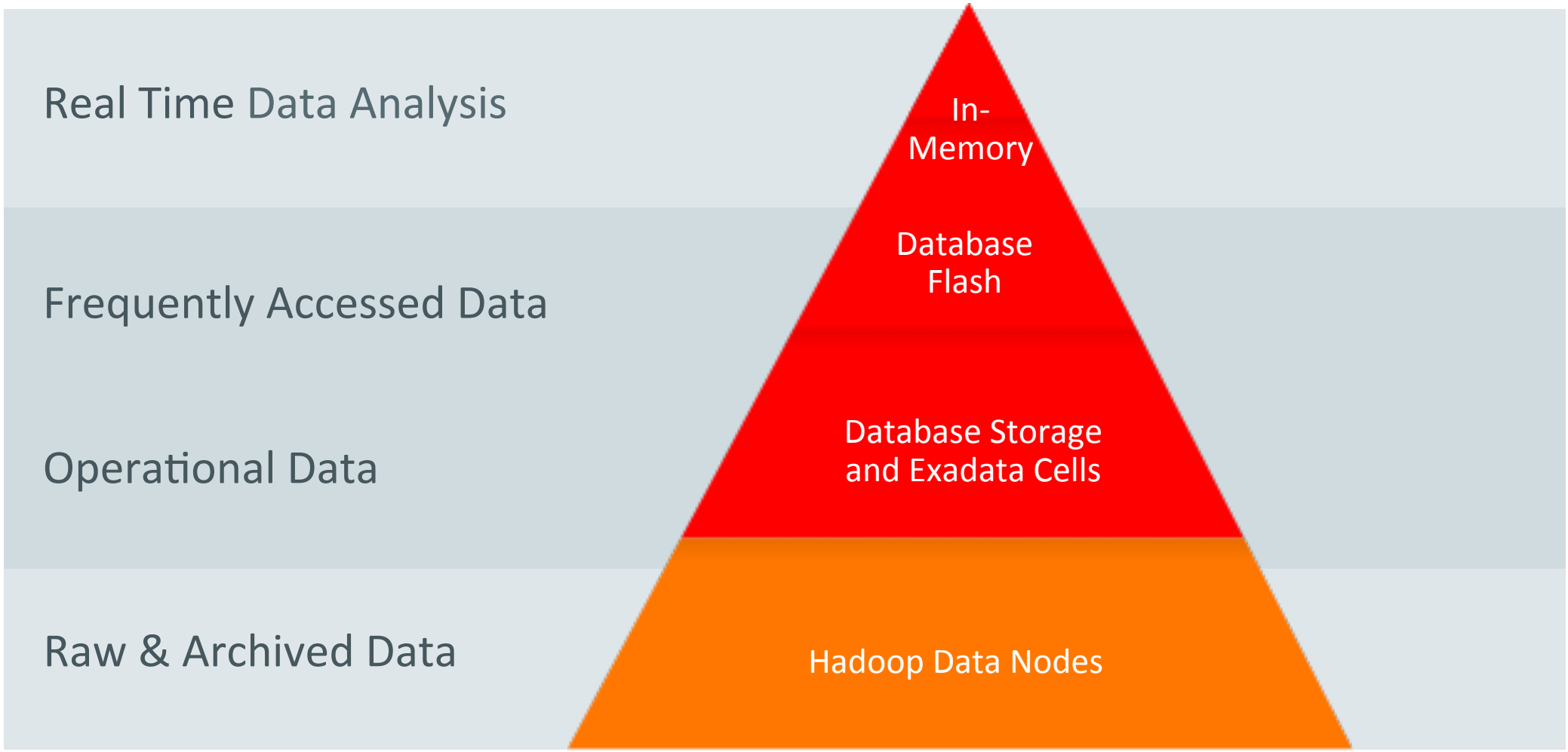
Virtual Private Databases. Creation of the packages

```
SQL> connect bds/bds
Connected.
SQL> SELECT c_first_name,
         c_last_name,
         c_birth_country,
         c_email_address
    FROM bds.customer_csv t
   WHERE ROWNUM <4;  2    3    4    5    6

C_FIRST_NAME         C_LAST_NAME                    C_BIRTH_COUNTRY      C_EMAIL
-------------------- ------------------------------ -------------------- -------
Jenifer              Jackson                        GREECE               xxxx@hu
Susan                Gavin                          COSTA RICA           xxxx@ma
Helen                Sutton                         SLOVAKIA             xxxx@eu

SQL>  SELECT COUNT(1) FROM bds.customer_csv t;

  COUNT(1)
----------
   7000358

     _
```

Other users observe all the data

Part 4: ILM

Big Data use case: Data Tiering

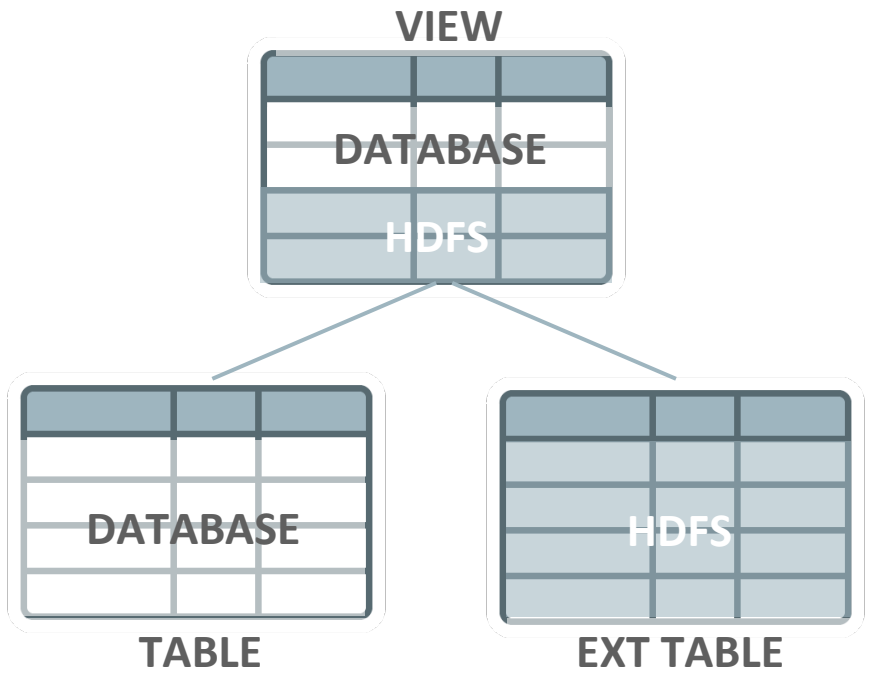


Review Two Approaches for Supporting ILM

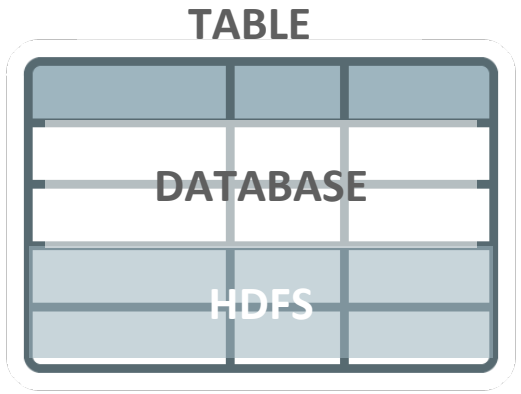
- Move Oracle Tablespace(s) to HDFS
  - Single table access both data sources
- Use Copy To Hadoop to copy Oracle data to HDFS
  - Define external table over data in HDFS
  - Database view used to combine sources

Archive Data: Big Data SQL Implementation Options

1. View Combines Data Sources

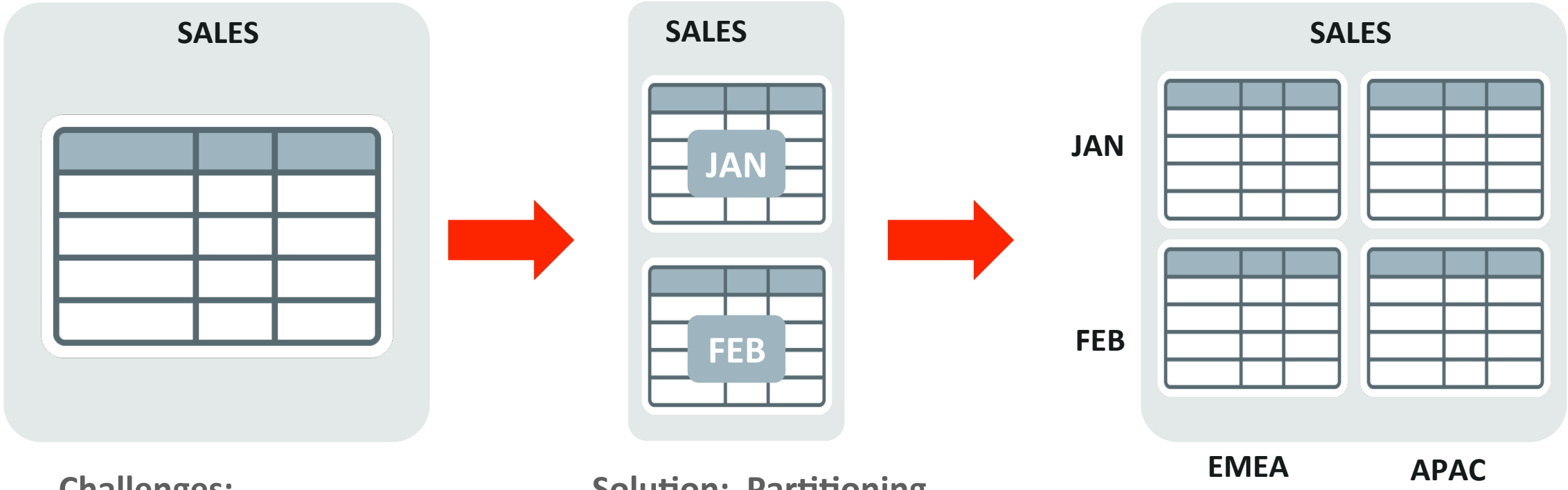


2. Table Storage Split Across Tiers



Oracle Partitioning

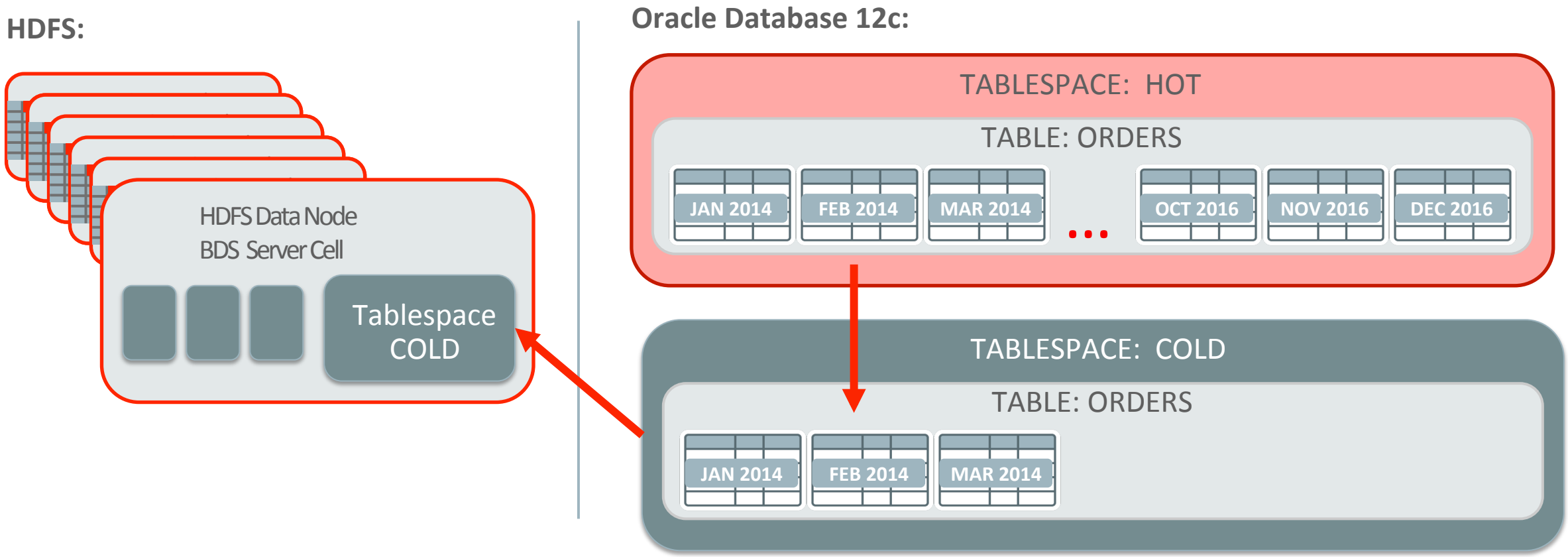
Enables large databases and indexes to be split into smaller, more manageable pieces



Challenges:
Large tables are difficult to manage

Solution: Partitioning
- Divide and conquer
- Easier data management
- Improve performance

Archive Data to “Cold” Partitions



Optimized Data Storage and Access

- Data in HDFS stored in Oracle Database format
- Queries utilize Oracle Database access structures and performance features
  - Indexes, Hybrid Columnar Compression, Partition Pruning, etc.
  - Use In-Memory option for speed of thought queries
- Big Data SQL Smart Scan offloads scans/filters to Hadoop Cluster

Archiving Data: Comparison

	External Tables	External Tablespaces
Access Data in Any Format	Yes	No
Use Advanced Access Methods (indexes, In-Memory, etc.)	No	Yes
Usage of Data in HDFS	Oracle DB + Any Hadoop Process	Oracle DB
Data Management	View	Table Partition
Big Data SQL Smart Scan	Yes	Yes
Storage Indexes	Yes	Yes
Bloom Filters	Yes	Yes
Partition Pruning	Hive	Oracle
Predicate Pushdown	Yes	Yes

Steps: Detailed

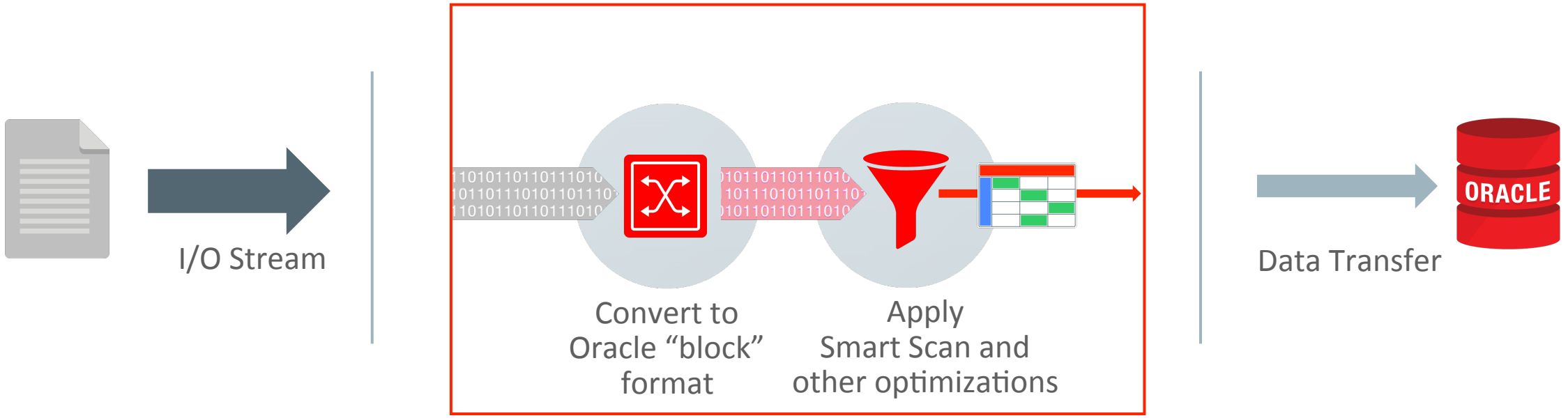
- Set up NFS Gateway or fuse-dfs
- Create a “cold tablespace”
- Move partitions (or tables, or whatever) to that tablespace
- Take the tablespace offline
- Copy the tablespace’s data files to hdfs
  - There is a specific directory where they should go
- Rename the tablespace’s data file to point to the hdfs location (NFS or fuse)
- Bring the tablespace online

Steps: Simplified

- Use **bds-copy-tbs-to-hdfs.sh** script to automate the process
  - Script will optionally install fuse-dfs
- Create a “cold tablespace”
- Move partitions (or tables, or whatever) to that tablespace
- Run single **bds-copy-tbs-to-hdfs.sh** command to do the rest. It will:
  - Take the tablespace offline
  - Copy the tablespace’s data files to hdfs
  - Rename the tablespace’s data file to point to the hdfs location (NFS or fuse)
  - Bring the tablespace online

Anatomy of a Big Data SQL Cell

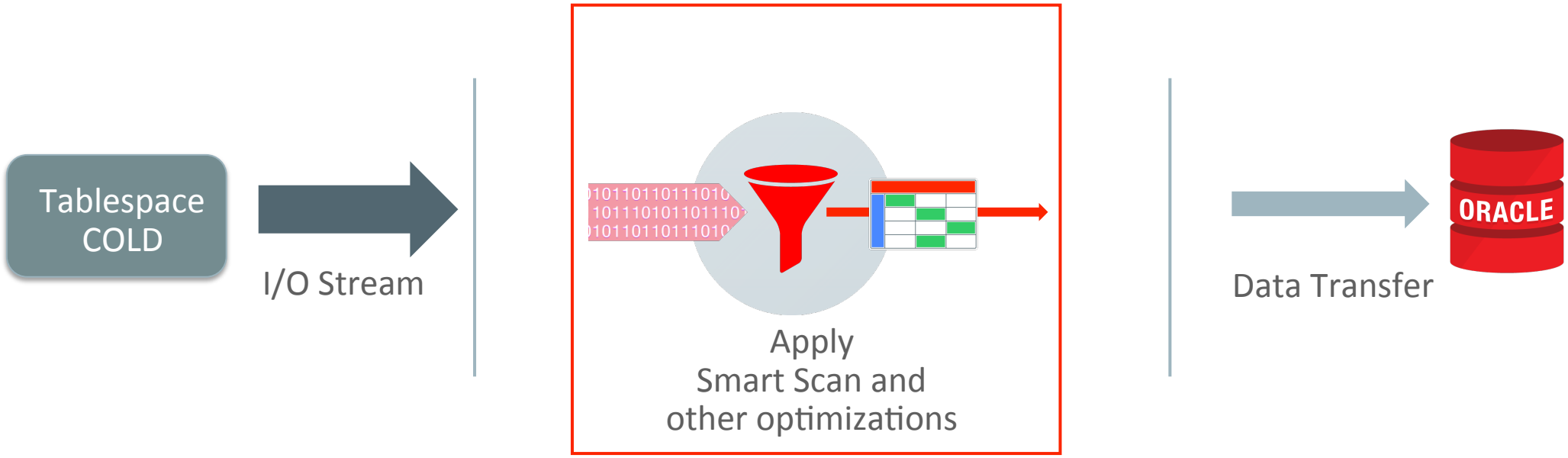
Smart Scan



Anatomy of a Big Data SQL Cell

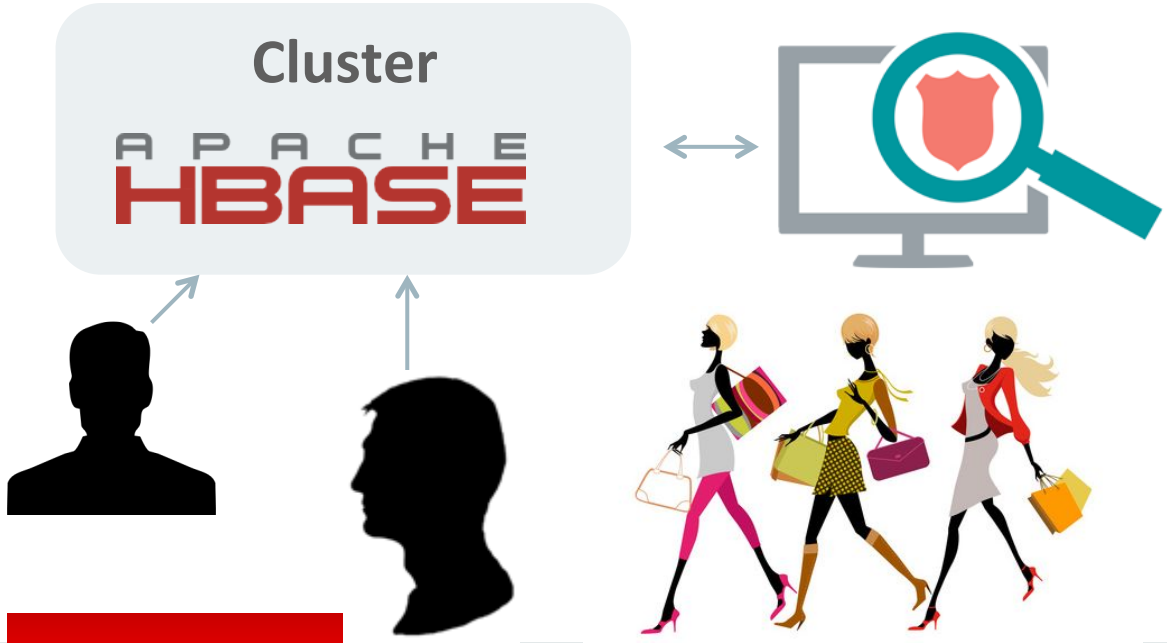
Smart Scan on Oracle Tablespace in HDFS

No conversion to Oracle block format required



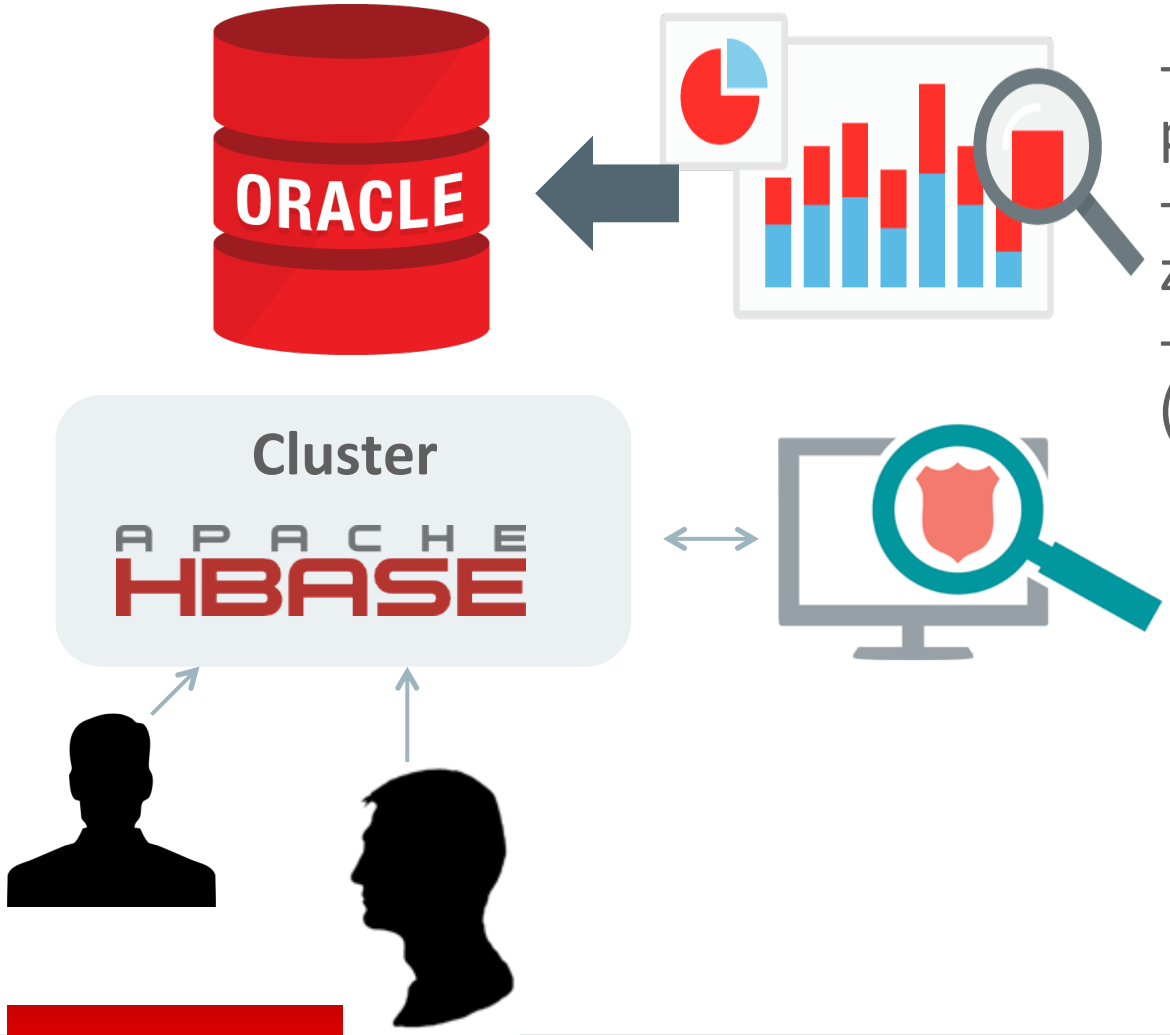
Part 5: NoSQL Databases

Big Data SQL and NoSQL databases. Use Case Example



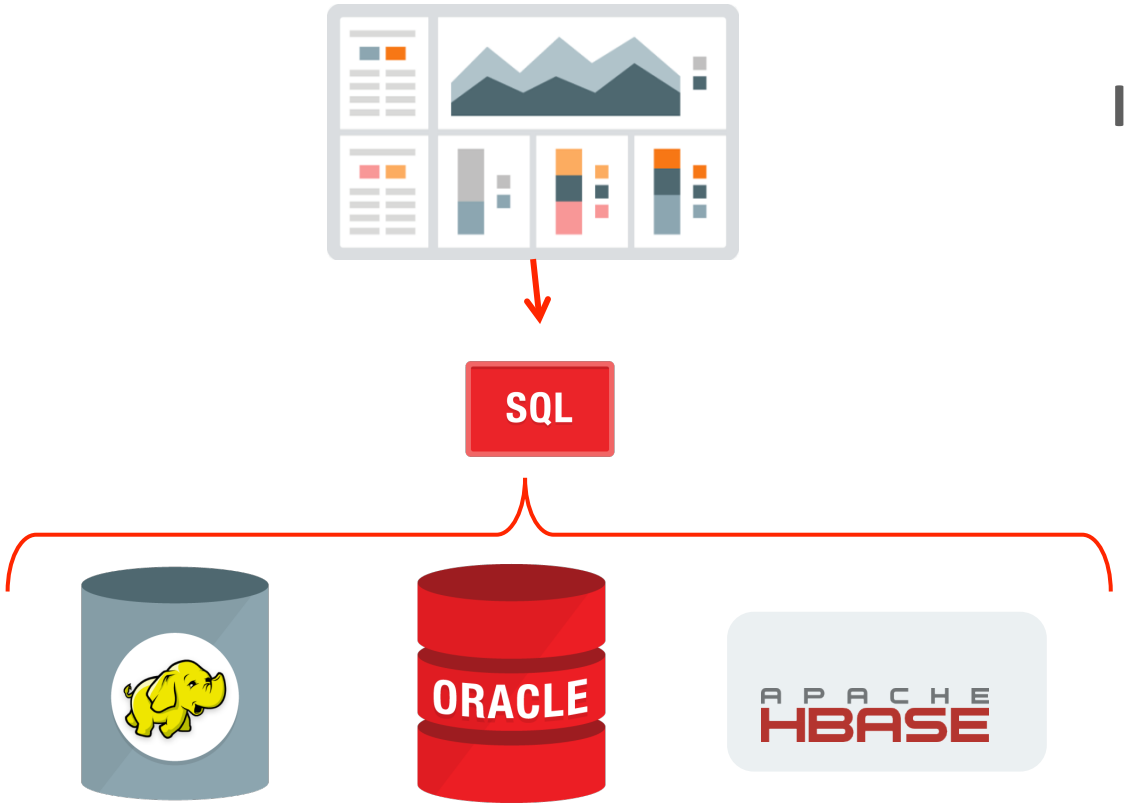
- I have huge company with tens of millions customers
- I do store customers profiles in HBase database, because:
    - Amount of customers (10s of millions)
    - All need low latency read response:
        - People (when use online services)
        - Applications (like anti-fraud apps)
    - Profile tags (metrics) could vary (data is pretty sparse)
    - Application developers want to have more flexibility and want easily add new columns
    - It's just fashion ☺

Big Data SQL and NoSQL databases. Use Case Example



-I also do have Oracle Database like analytical platform
- I do have existing BI application, which has zero idea what HBase is
- I just need to use data from those profiles (stored in HBase) in my reporting

Big Data SQL and NoSQL databases. Use Case Example



I want to have SQL interface over all those sources

Example of implementation

Hbase DDL

```
hbase(main):001:0>  create 'emp', 'personal data', 'professional data‘;
hbase(main):001:0> put 'emp','6179995','personal data:name','raj'
hbase(main):001:0> put 'emp','6179995','personal data:city','palo alto'
hbase(main):001:0> put 'emp','6179995','professional data:position','developer'
hbase(main):001:0> put 'emp','6179995','professional data:salary','50000‘
hbase(main):001:0> put 'emp','6274234','personal data:name','alex'
hbase(main):001:0> put 'emp','6274234','personal data:city','belmont'
hbase(main):001:0> put 'emp','6274234','professional data:position','pm'
hbase(main):001:0> put 'emp','6274234','professional data:salary','60000'
```

Create table in Hbase and put data there

Hive DDL

hive> CREATE EXTERNAL TABLE IF NOT EXISTS emp_hbase
(rowkey STRING,
 ename STRING,
city STRING,
position STRING,
salary STRING)
STORED BY
'org.apache.hadoop.hive.hbase.HBaseStorageHandler'
WITH SERDEPROPERTIES ('hbase.columns.mapping' =
':key#binary,personal data:name, personal data:city,
professional data:position, professional data:salary')
TBLPROPERTIES (**'hbase.table.name' = 'emp'**);

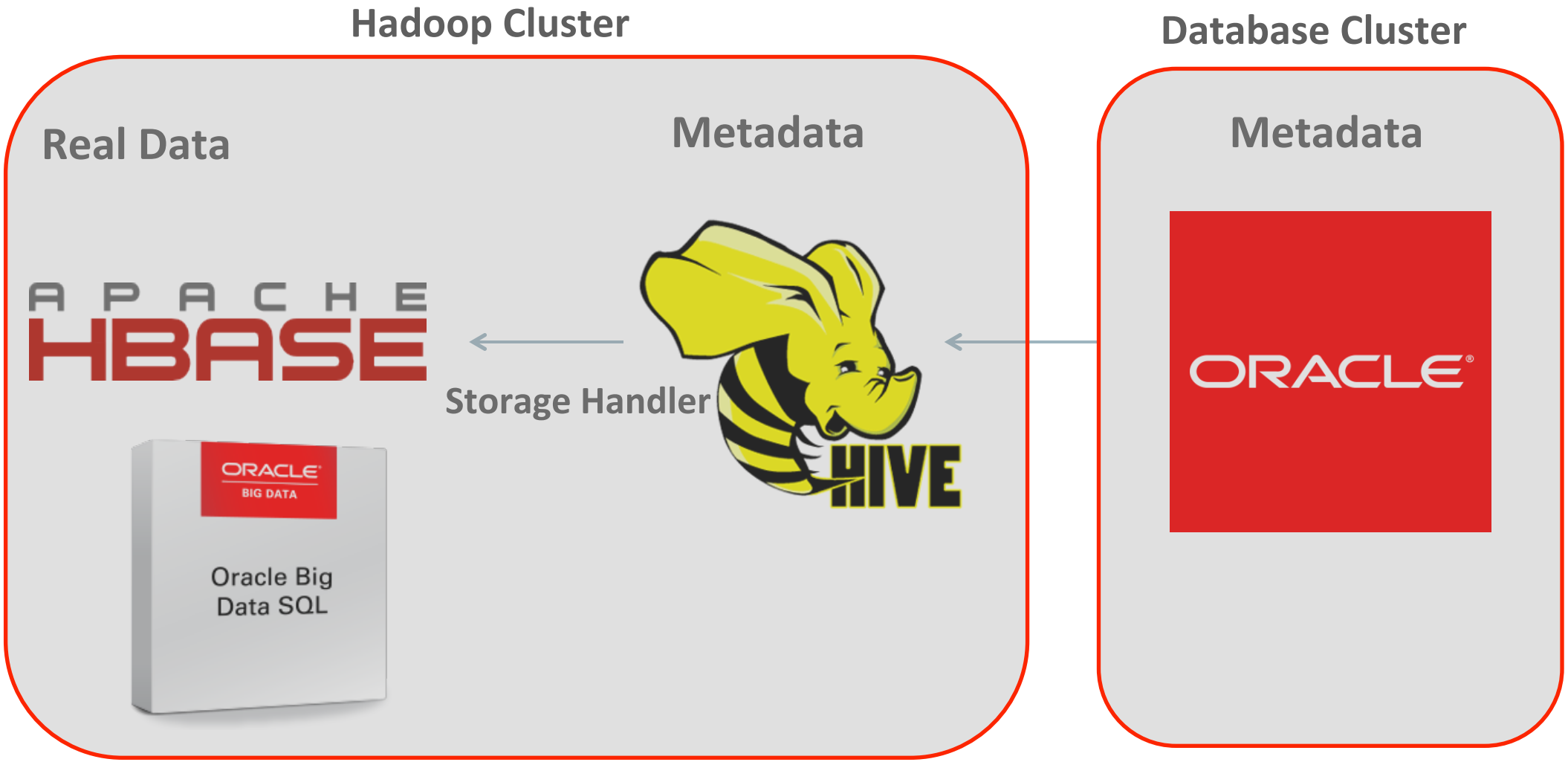
In hive define external table, that linked with Hbase table

Oracle DDL

```
SQL> CREATE TABLE emp_hbase
   (
  rowkey number, ename VARCHAR2(4000), city
VARCHAR2(4000),  position VARCHAR2(4000),salary number
   )
   ORGANIZATION EXTERNAL
    ( TYPE ORACLE_HIVE
      DEFAULT DIRECTORY "DEFAULT_DIR"
      ACCESS PARAMETERS
      ( com.oracle.bigdata.cluster=bds30
        com.oracle.bigdata.tablename=emp_hbase)
    )
   REJECT LIMIT UNLIMITED
  PARALLEL ;
```

In Oracle RDBMS define external table, that linked with hive table

Chain of DDLs



Report example

Business question: I need a report of all sales by years for each position (job role) For answer on it I need information from **STORE_SALES** table (sales), **EMP_HBASE** (position) and **DATE_DIM** (year)

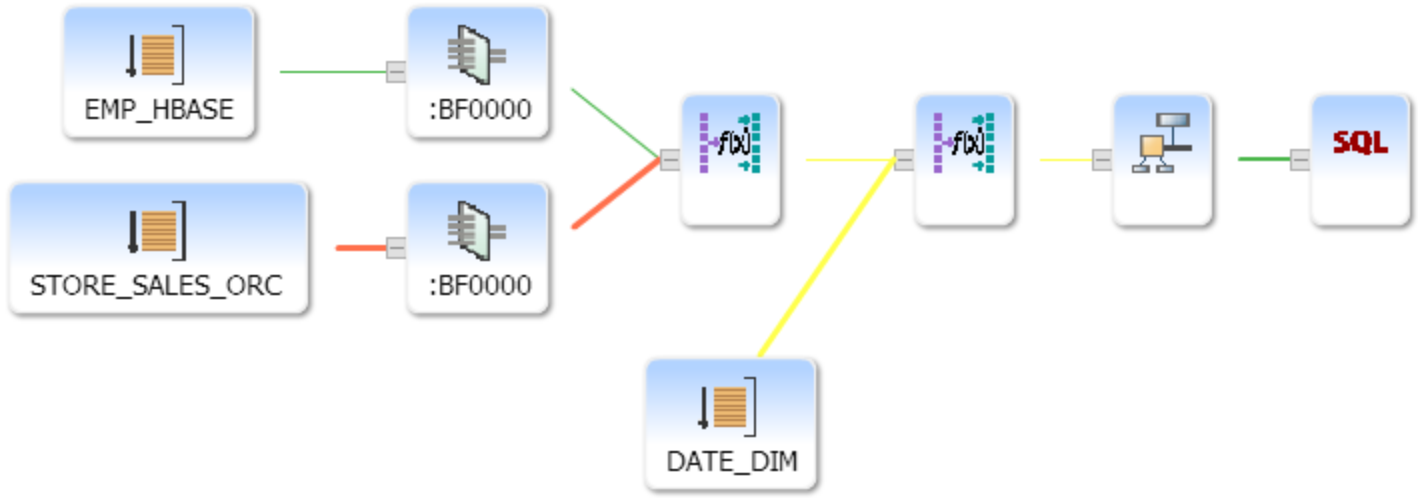
Report example

Business question: I need a report of all sales by years for each position (job role)
For answer on it I need information from **STORE_SALES** table (sales), **EMP_HBASE** (position) and **DATE_DIM** (year)

Report example

Business question: I need a report of all sales by years for each position (job role)
For answer on it I need information from **STORE_SALES** table (sales), **EMP_HBASE** (position) and **DATE_DIM** (year)

```
SQL> SELECT e.position,
       d.d_year,
       SUM(s.ss_ext_wholesale_cost),
FROM store_sales_orc s,
 emp_hbase e, date_dim d
 WHERE e.rowkey = s.ss_customer_sk
   AND s.ss_sold_date_sk =d. d_date_sk
   AND e.rowkey > 0
 GROUP BY e.position, d.d_year
```



Report example

Business question: I need a report of all sales by years for each position (job role)
For answer on it I need information (sales) from **STORE_SALES** table (fact table), **EMP_HBASE** (position) and **DATE_DIM(year**)

Operation	Name	Line ID	Estimated Rows	Cost	Timeline(1...	Actual Rows
SELECT STATEMENT		0				10
HASH GROUP BY		1	42	32M		10
HASH JOIN		2	1,867	32M		478K
HASH JOIN		3	1,867	32M		478K
JOIN FILTER CREATE	:BF0000	4	2	2		2
EXTERNAL TABLE ACCESS STORAGE FULL	EMP_HBASE	5	2	2		2
JOIN FILTER USE	:BF0000	6	6,385M	32M		944K
EXTERNAL TABLE ACCESS STORAGE FULL	STORE_SALES_ORC	7	6,385M	32M		944K
TABLE ACCESS FULL	DATE_DIM	8	110K	298		110K

98% of the Job was done on the cell side

2% of the Job was done on the db side

EMP_HBASE

rowkey (PK)

APACHE HBASE

STORE_SALES

ss_sold_date_sk (FK)
ss_customer_sk (FK)

Apache orc

DATE_DIM

d_date_sk (PK)

Hadoop Cluster

Database Cluster

Useful links

Blogpost:
https://blogs.oracle.com/datawarehousing/tags/bds_start
Youtube video:
https://www.youtube.com/playlist?list=PLkGlRYRzedXbjekBpGsMf2p5thMYajrOt