Lesson 2 : Hadoop & NoSQL Data Loading using Hadoop Tools and ODI12c

Mark Rittman, CTO, Rittman Mead

Oracle Openworld 2014, San Francisco

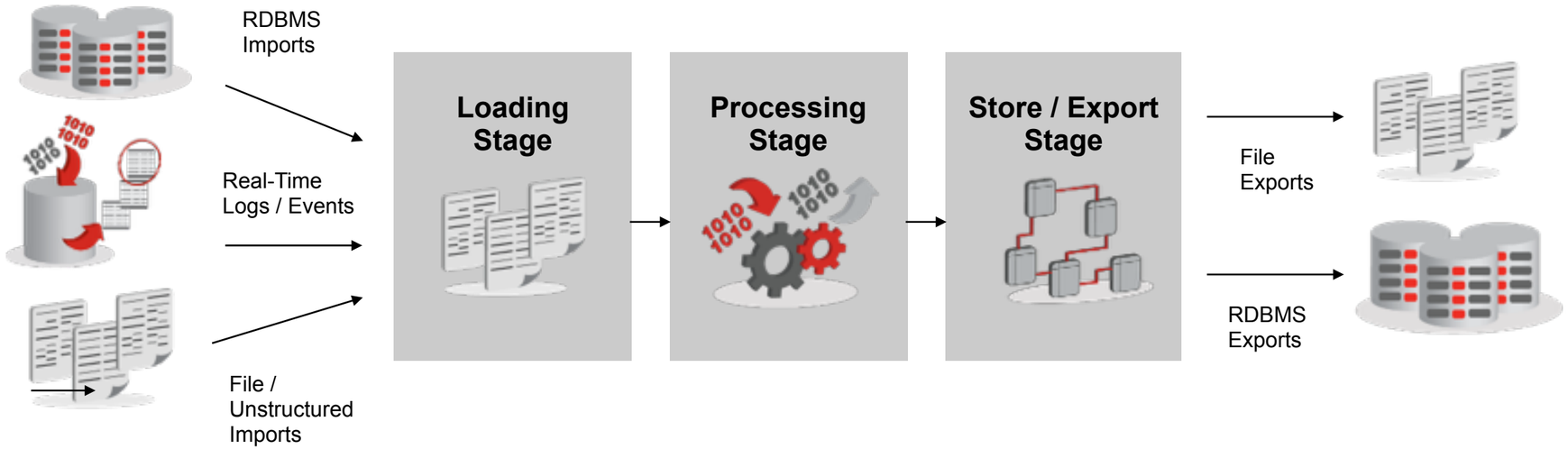
T : +44 (0) 1273 911 268 (UK) **or** (888) 631-1410 (USA) **or**
+61 3 9596 7186 (Australia & New Zealand) **or** +91 997 256 7970 (India)

E : info@rittmanmead.com
W : www.rittmanmead.com

rittmanmead
INTEGRATED ANALYTICS

Moving Data In, Around and Out of Hadoop

- Three stages to Hadoop data movement, with dedicated Apache / other tools
  - **Load** : receive files in batch, or in real-time (logs, events)
  - **Transform** : process & transform data to answer questions
  - **Store / Export** : store in structured form, or export to RDBMS using Sqoop



T : +44 (0) 1273 911 268 (UK) **or** (888) 631-1410 (USA) **or**
+61 3 9596 7186 (Australia & New Zealand) **or** +91 997 256 7970 (India)

E : info@rittmanmead.com
W : www.rittmanmead.com

rittmanmead
INTEGRATED ANALYTICS

Lesson 2 : Hadoop Data Loading

Hadoop Data Loading / Ingestion Fundamentals

T : +44 (0) 1273 911 268 (UK) **or** (888) 631-1410 (USA) **or**
+61 3 9596 7186 (Australia & New Zealand) **or** +91 997 256 7970 (India)

E : info@rittmanmead.com
W : www.rittmanmead.com

rittmanmead
INTEGRATED ANALYTICS

Core Apache Hadoop Tools

- **Apache Hadoop, including MapReduce and HDFS**
  - Scaleable, fault-tolerant file storage for Hadoop
  - Parallel programming framework for Hadoop
- Apache Hive
  - SQL abstraction layer over HDFS
  - Perform set-based ETL within Hadoop
- Apache Pig, Spark
  - Dataflow-type languages over HDFS, Hive etc
  - Extensible through UDFs, streaming etc
- **Apache Flume, Apache Sqoop, Apache Kafka**
  - Real-time and batch loading into HDFS
  - Modular, fault-tolerant, wide source/target coverage

T : +44 (0) 1273 911 268 (UK) **or** (888) 631-1410 (USA) **or**
+61 3 9596 7186 (Australia & New Zealand) **or** +91 997 256 7970 (India)

E : info@rittmanmead.com
W : www.rittmanmead.com

Demo

Hue with CDH5 on the Big Data Lite VM

T : +44 (0) 1273 911 268 (UK) **or** (888) 631-1410 (USA) **or**
+61 3 9596 7186 (Australia & New Zealand) **or** +91 997 256 7970 (India)

E : info@rittmanmead.com
W : www.rittmanmead.com

rittmanmead
INTEGRATED ANALYTICS

Other Tools Typically Used…

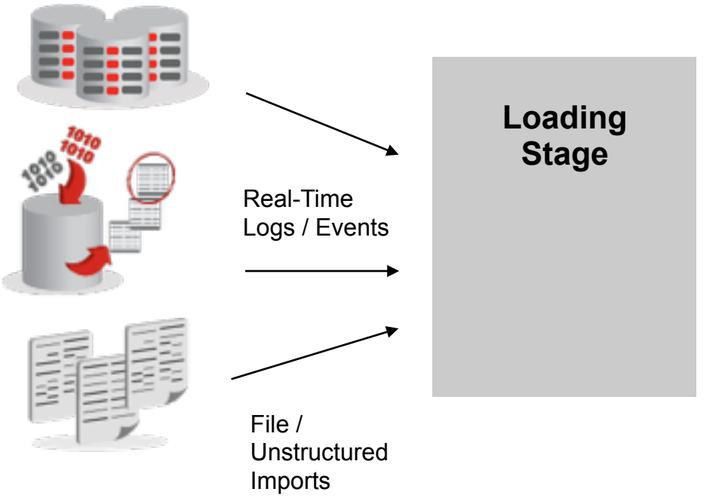
- Python, Scala, Java and other programming languages
  - For more complex and procedural transformations
- Shell scripts, sed, awk, regexes etc
- R and R-on-Hadoop
  - Typically at the “discovery” phase
- And down the line - ETL tools to automate the process
  - ODI, Pentaho Data Integrator etc

T : +44 (0) 1273 911 268 (UK) **or** (888) 631-1410 (USA) **or**
+61 3 9596 7186 (Australia & New Zealand) **or** +91 997 256 7970 (India)

E : info@rittmanmead.com
W : www.rittmanmead.com

Data Loading into Hadoop

- Default load type is real-time, streaming loads
  - Batch / bulk loads only typically used to seed system
- Variety of sources including web log activity, event streams
- Target is typically HDFS (Hive) or HBase
- Data typically lands in “raw state”
  - Lots of files and events, need to be filtered/aggregated
  - Typically semi-structured (JSON, logs etc)
  - High volume, high velocity
    - Which is why we use Hadoop rather than RBDMS (speed vs. ACID trade-off)
  - Economics of Hadoop means its often possible to archive all incoming data at detail level

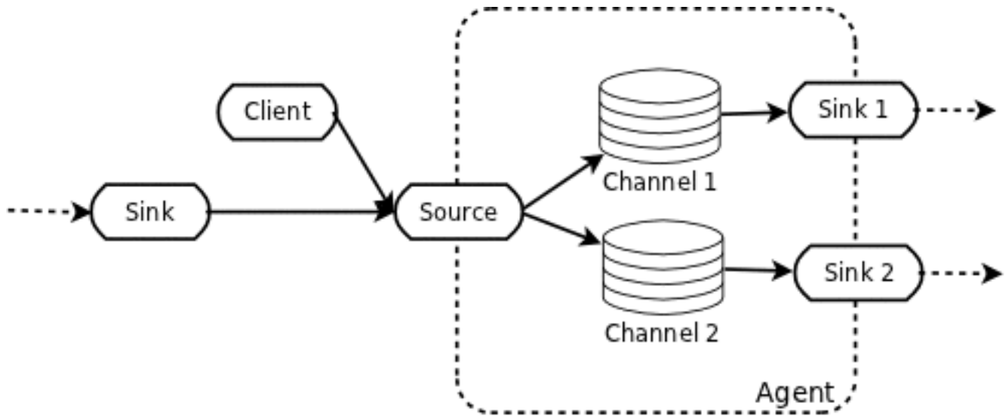


T : +44 (0) 1273 911 268 (UK) **or** (888) 631-1410 (USA) **or**
+61 3 9596 7186 (Australia & New Zealand) **or** +91 997 256 7970 (India)

E : info@rittmanmead.com
W : www.rittmanmead.com

Apache Flume : Distributed Transport for Log Activity

- Apache Flume is the standard way to transport log files from source through to target
  - Initial use-case was webserver log files, but can transport any file from A>B
  - Does not do data transformation, but can send to multiple targets / target types
  - Mechanisms and checks to ensure successful transport of entries
- Has a concept of “agents”, “sinks” and “channels”
  - **Agents** collect and forward log data
  - **Sinks** store it in final destination
  - **Channels** store log data en-route
- Simple configuration through INI files
  - Handled outside of ODI12c

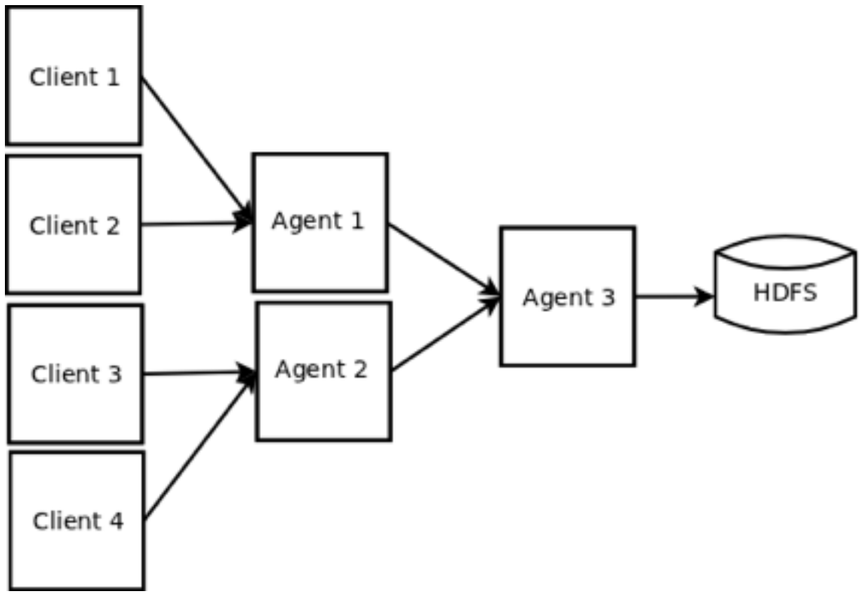


T : +44 (0) 1273 911 268 (UK) **or** (888) 631-1410 (USA) **or**
+61 3 9596 7186 (Australia & New Zealand) **or** +91 997 256 7970 (India)

E : info@rittmanmead.com
W : www.rittmanmead.com

Apache Flume : Agents, Channels and Sinks

- Multiple agents can be used to capture logs from many sources, combine into one output
  - Needs at least one source agent, and a target agent
- Agents can be multi-step, handing-off data across the topology
- Channels store data in files, or in RAM, as a buffer between steps
- Log files being continuously written to have contents trickle-fed across to source
- Sink types for Hive, HBase and many others
- Free software, part of Hadoop platform

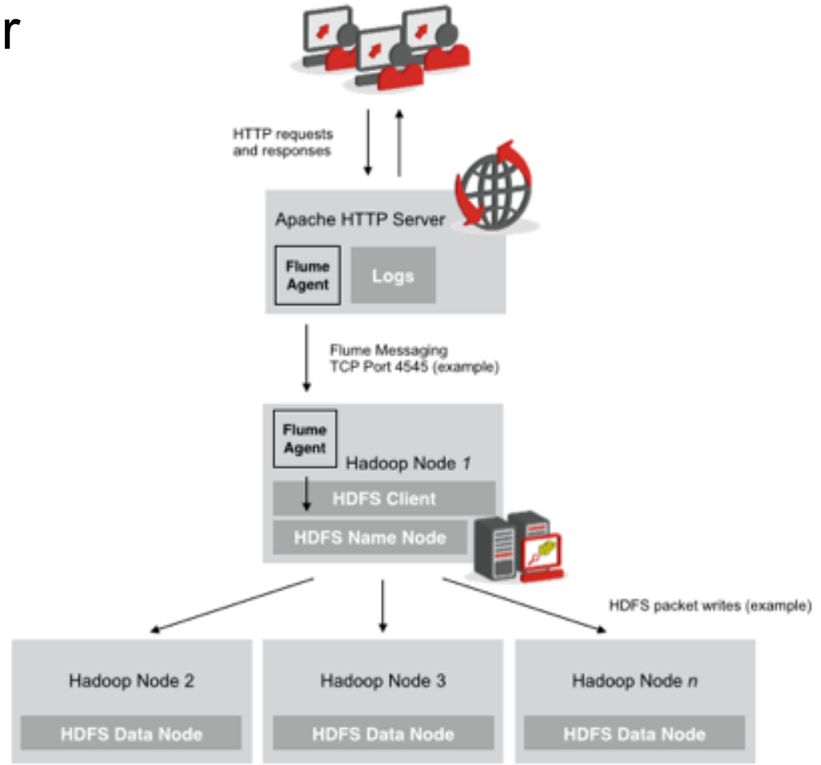


T : +44 (0) 1273 911 268 (UK) **or** (888) 631-1410 (USA) **or**
+61 3 9596 7186 (Australia & New Zealand) **or** +91 997 256 7970 (India)

E : info@rittmanmead.com
W : www.rittmanmead.com

Typical Flume Use Case : Copy Log Files to HDFS / Hive

- Typical use of Flume is to copy log entries from servers onto Hadoop / HDFS
- Tightly integrated with Hadoop framework
- Mirror server log files into HDFS, aggregate logs from >1 server
- Can aggregate, filter and transform incoming data before writing to HDFS
- Alternatives to log file “tailing” - HTTP GET / PUSH etc



T : +44 (0) 1273 911 268 (UK) **or** (888) 631-1410 (USA) **or**
+61 3 9596 7186 (Australia & New Zealand) **or** +91 997 256 7970 (India)

E : info@rittmanmead.com
W : www.rittmanmead.com

rittmanmead
INTEGRATED ANALYTICS

Flume Source / Target Configuration

- Conf file for source system agent
  - TCP port, channel size+type, source type

```
## SOURCE AGENT ##
## Local instalation: /home/ec2-user/apache-flume
## configuration file location:  /home/ec2-user/apache-flume/conf
## bin file location: /home/ec2-user/apache-flume/bin
## START Agent: bin/flume-ng agent -c conf -f conf/flume-src-agent.conf -n source_

# http://flume.apache.org/FlumeUserGuide.html#exec-source
source_agent.sources = apache_server
source_agent.sources.apache_server.type = exec
source_agent.sources.apache_server.command = tail -f /etc/httpd/logs/access_log
source_agent.sources.apache_server.batchSize = 1
source_agent.sources.apache_server.channels = memoryChannel
source_agent.sources.apache_server.interceptors = itime ihost itype

# http://flume.apache.org/FlumeUserGuide.html#timestamp-interceptor
source_agent.sources.apache_server.interceptors.itime.type = timestamp

# http://flume.apache.org/FlumeUserGuide.html#host-interceptor
source_agent.sources.apache_server.interceptors.ihost.type = host
source_agent.sources.apache_server.interceptors.ihost.useIP = false
source_agent.sources.apache_server.interceptors.ihost.hostHeader = host

# http://flume.apache.org/FlumeUserGuide.html#static-interceptor
source_agent.sources.apache_server.interceptors.itype.type = static
source_agent.sources.apache_server.interceptors.itype.key = log_type
source_agent.sources.apache_server.interceptors.itype.value = apache_access_combine

# http://flume.apache.org/FlumeUserGuide.html#memory-channel
source_agent.channels = memoryChannel
source_agent.channels.memoryChannel.type = memory
source_agent.channels.memoryChannel.capacity = 100

## Send to Flume Collector on Hadoop Node
# http://flume.apache.org/FlumeUserGuide.html#avro-sink
source_agent.sinks = avro_sink
source_agent.sinks.avro_sink.type = avro
source_agent.sinks.avro_sink.channel = memoryChannel
source_agent.sinks.avro_sink.hostname = 81.155.163.172
source_agent.sinks.avro_sink.port = 4545
```

- Conf file for target system agent
  - TCP port, channel size+type, sink type

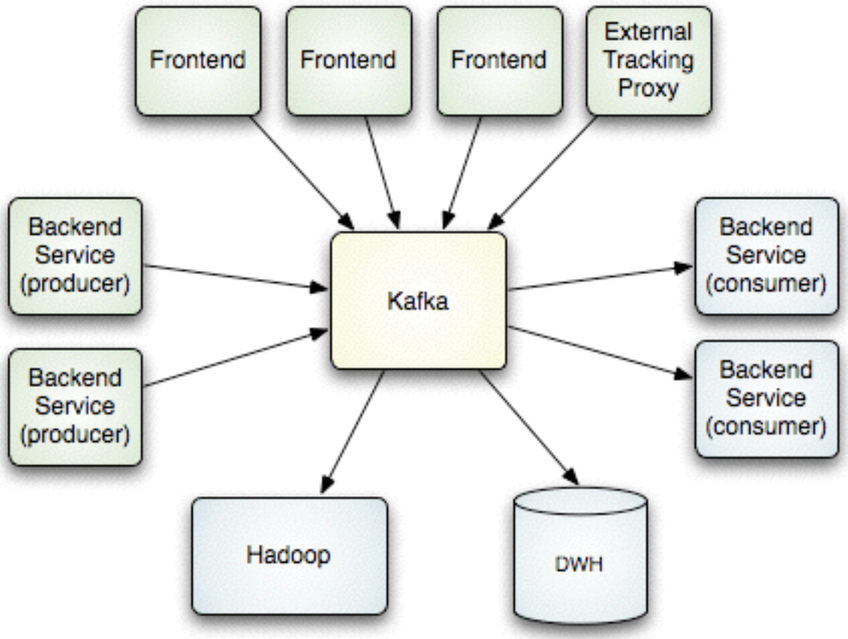
```
## TARGET AGENT ##
## configuration file location:  /etc/flume-ng/conf
## START Agent: flume-ng agent -c conf -f /etc/flume-ng/conf/flume-trg-agent.conf

#http://flume.apache.org/FlumeUserGuide.html#avro-source
collector.sources = AvroIn
collector.sources.AvroIn.type = avro
collector.sources.AvroIn.bind = 0.0.0.0
collector.sources.AvroIn.port = 4545
collector.sources.AvroIn.channels = mc1 mc2

## Channels ##
## Source writes to 2 channels, one for each sink
collector.channels = mc1 mc2

#http://flume.apache.org/FlumeUserGuide.html#memory-channel

collector.channels.mc1.type = memory
collector.channels.mc1.capacity = 100

collector.channels.mc2.type = memory
collector.channels.mc2.capacity = 100

## Sinks ##
collector.sinks = LocalOut HadoopOut

## Write copy to Local Filesystem
#http://flume.apache.org/FlumeUserGuide.html#file-roll-sink
collector.sinks.LocalOut.type = file_roll
collector.sinks.LocalOut.sink.directory = /var/log/flume-ng
collector.sinks.LocalOut.sink.rollInterval = 0
collector.sinks.LocalOut.channel = mc1

## Write to HDFS
#http://flume.apache.org/FlumeUserGuide.html#hdfs-sink
collector.sinks.HadoopOut.type = hdfs
collector.sinks.HadoopOut.channel = mc2
collector.sinks.HadoopOut.hdfs.path = /user/root/flume-channel/%{log_type}/%y%m%d
collector.sinks.HadoopOut.hdfs.fileType = DataStream
collector.sinks.HadoopOut.hdfs.writeFormat = Text
collector.sinks.HadoopOut.hdfs.rollSize = 0
collector.sinks.HadoopOut.hdfs.rollCount = 10000
collector.sinks.HadoopOut.hdfs.rollInterval = 600
```

T : +44 (0) 1273 911 268 (UK) **or** (888) 631-1410 (USA) **or**
+61 3 9596 7186 (Australia & New Zealand) **or** +91 997 256 7970 (India)

E : info@rittmanmead.com
W : www.rittmanmead.com

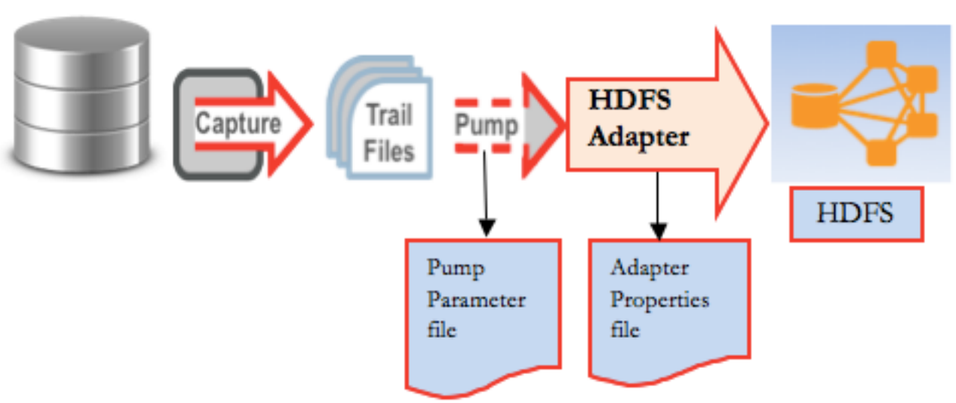
Apache Kafka : Reliable, Message-Based

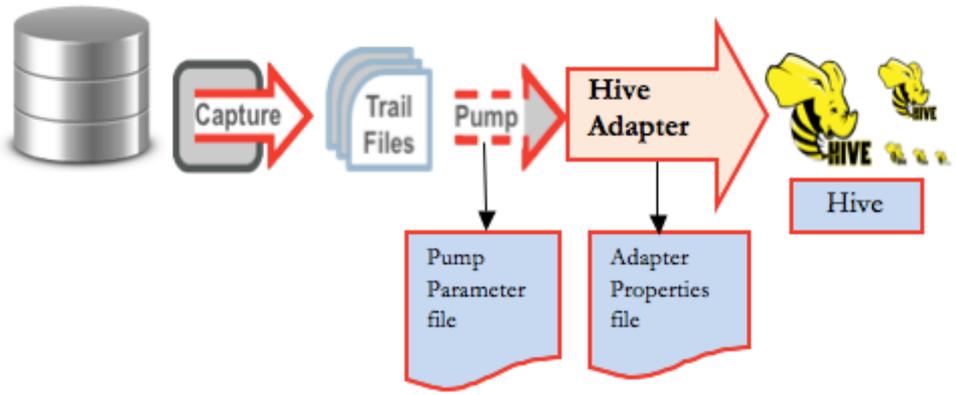
- Developed by LinkedIn, designed to address Flume issues around reliability, throughput
  - (though many of those issues have been addressed since)
- Designed for persistent messages as the common use case
  - Website messages, events etc vs. log file entries
- Consumer (pull) rather than Producer (push) model
- Supports multiple consumers per message queue
- More complex to set up than Flume, and can use Flume as a consumer of messages
  - But gaining popularity, especially alongside Spark Streaming



GoldenGate for Continuous Streaming to Hadoop

- Oracle GoldenGate is also an option, for streaming RDBMS transactions to Hadoop
- Leverages GoldenGate & HDFS / Hive Java APIs
- Sample Implementations on MOS Doc.ID 1586210.1 (HDFS) and 1586188.1 (Hive)
- Likely to be formal part of GoldenGate in future release - but usable now
- Can also integrate with Flume for delivery to HDFS - see MOS Doc.ID 1926867.1



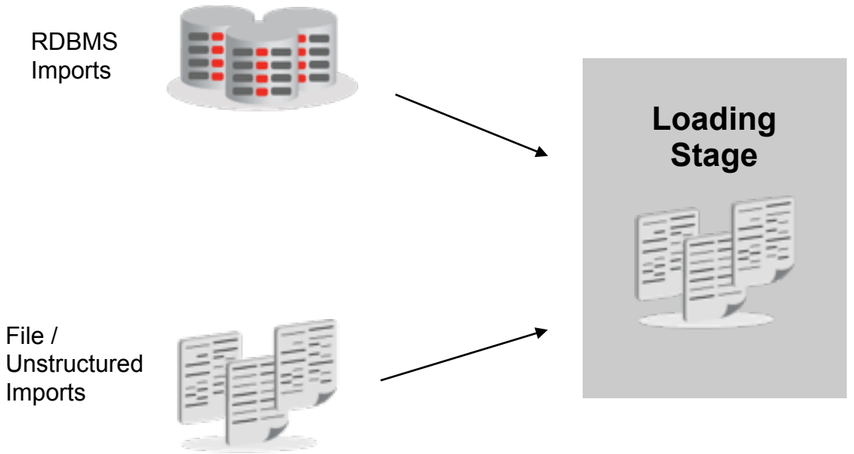


T : +44 (0) 1273 911 268 (UK) **or** (888) 631-1410 (USA) **or**
+61 3 9596 7186 (Australia & New Zealand) **or** +91 997 256 7970 (India)

E : info@rittmanmead.com
W : www.rittmanmead.com

Bulk-Loading into Hadoop

- Typically used for initial data load, or for one-off analysis
- Aim for bulk-loading is to copy external dataset into HDFS
  - From files (delimited, semi-structured, XML, JSON etc)
  - From databases or other structured data stores
- Main tools used for bulk-data loading include
  - Hadoop FS Shell
  - Sqoop



Hadoop FS Shell Commands

- Follows typical Unix/Linux command naming

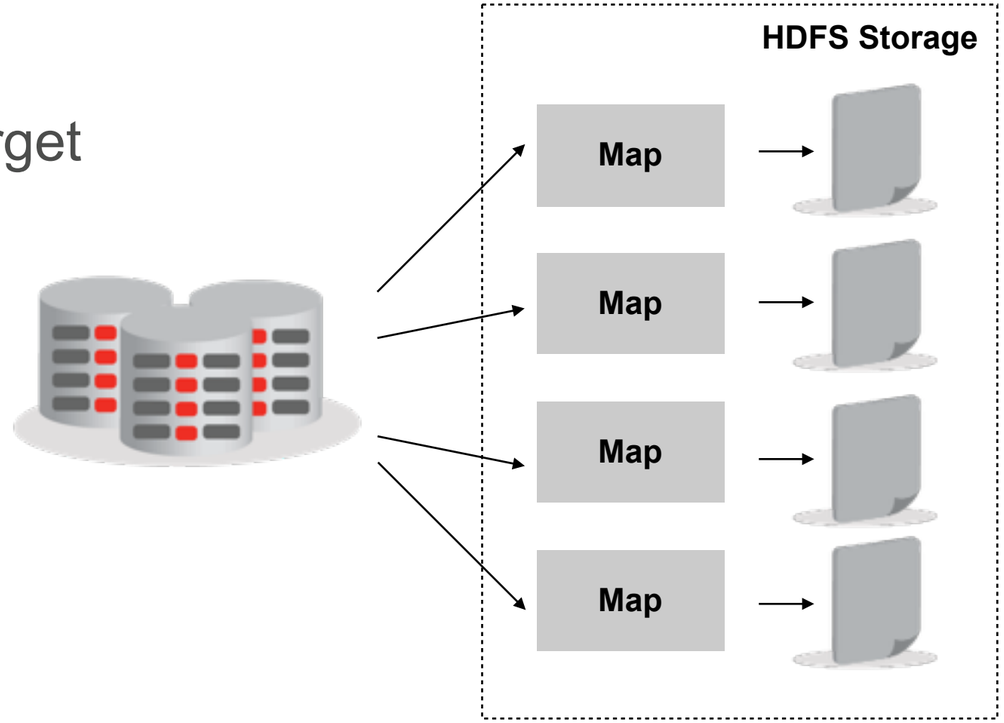
```
[oracle@bigdatalite mapreduce]$ hadoop fs -mkdir /user/oracle/my_stuff
[oracle@bigdatalite mapreduce]$ hadoop fs -ls /user/oracle
Found 5 items
drwx------   - oracle hadoop          0 2013-04-27 16:48 /user/oracle/.staging
drwxrwxrwx   - oracle hadoop          0 2012-09-18 17:02 /user/oracle/moviedemo
drwxrwxrwx   - oracle hadoop          0 2012-10-17 15:58 /user/oracle/moviework
drwxrwxrwx   - oracle hadoop          0 2013-05-03 17:49 /user/oracle/my_stuff
drwxrwxrwx   - oracle hadoop          0 2012-08-10 16:08 /user/oracle/stage
```

- Additional commands for bulk data movement

```
$ hadoop fs -copyFromLocal <local_dir> <hdfs_dir>
$ hadoop fs -copyToLocal <hdfs_dir> <local_dir>
```

Apache Sqoop : SQL to Hadoop

- Apache top-level project, typically ships with most Hadoop distributions
- Tool to transfer data from relational database systems
  - Oracle, mySQL, PostgreSQL, Teradata etc
- Loads into, and exports out of, Hadoop ecosystem
  - Uses JDBC drivers to connect to RBDMS source/target
  - Data transferred in/out of Hadoop using parallel Map-only Hadoop jobs
    - Sqoop introspects source / target RBDMS to determine structure, table metadata
    - Job tracker splits import / export into separate jobs, based on split column(s)



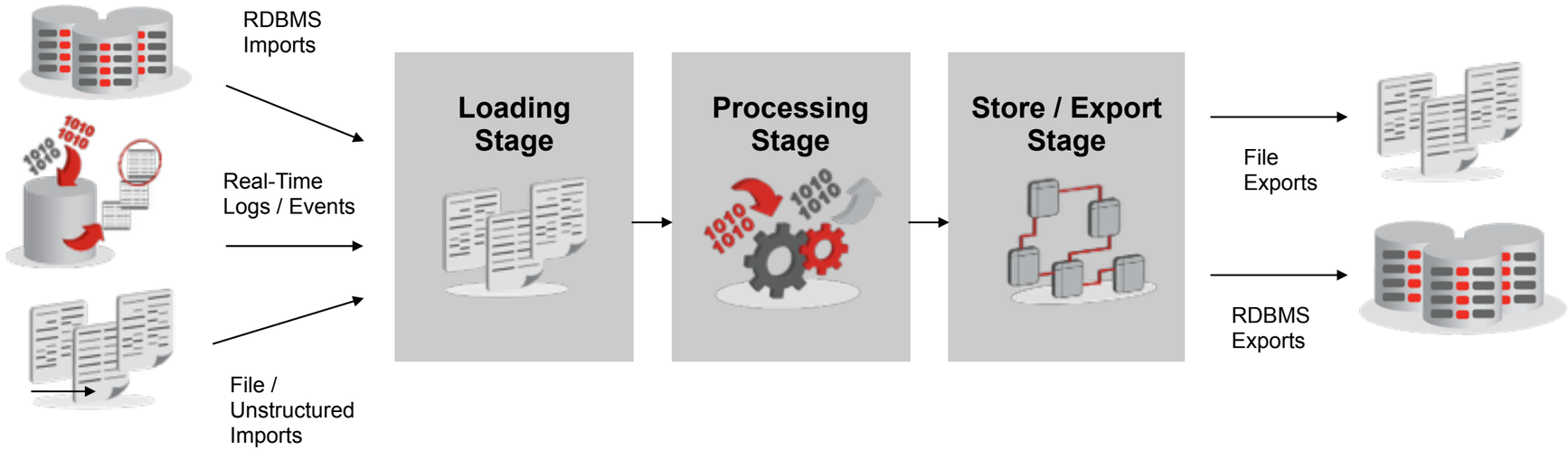
Sqoop Command-Line Parameters

```
sqoop import –connect jdbc:oracle:thin:@centraldb11gr2.rittmandev.com:1521/
ctrl11g.rittmandev.com –username blog_refdata –password password –query 'SELECT
p.post_id, c.cat_name from post_one_cat p, categories c where p.cat_id = c.cat_id
and $CONDITIONS' –target_dir /user/oracle/post_categories –hive-import –hive-
overwrite –hive-table post_categories –split-by p.post_id
```

- `–username`, `–password` : database account username and password
- `–query` : SELECT statement to retrieve data (can use `–table` instead, for single table)
- `$CONDITIONS`, `–split-by` : column by which MapReduce jobs can be run in parallel
- `–hive-import`, `–hive-overwrite`, `–hive-table` : name and load mode for Hive table
- `– target_dir` : target HDFS directory to land data in initially (required for SELECT)

Data Storage and Formats within Hadoop Clusters

- Data landing in Hadoop clusters typically is in raw, unprocessed form
- May arrive as log files, XML files, JSON documents, machine / sensor data
- Typically needs to go through a processing, filtering and aggregation phase to be useful
- Final output of processing stage is usually structured files, or Hive tables



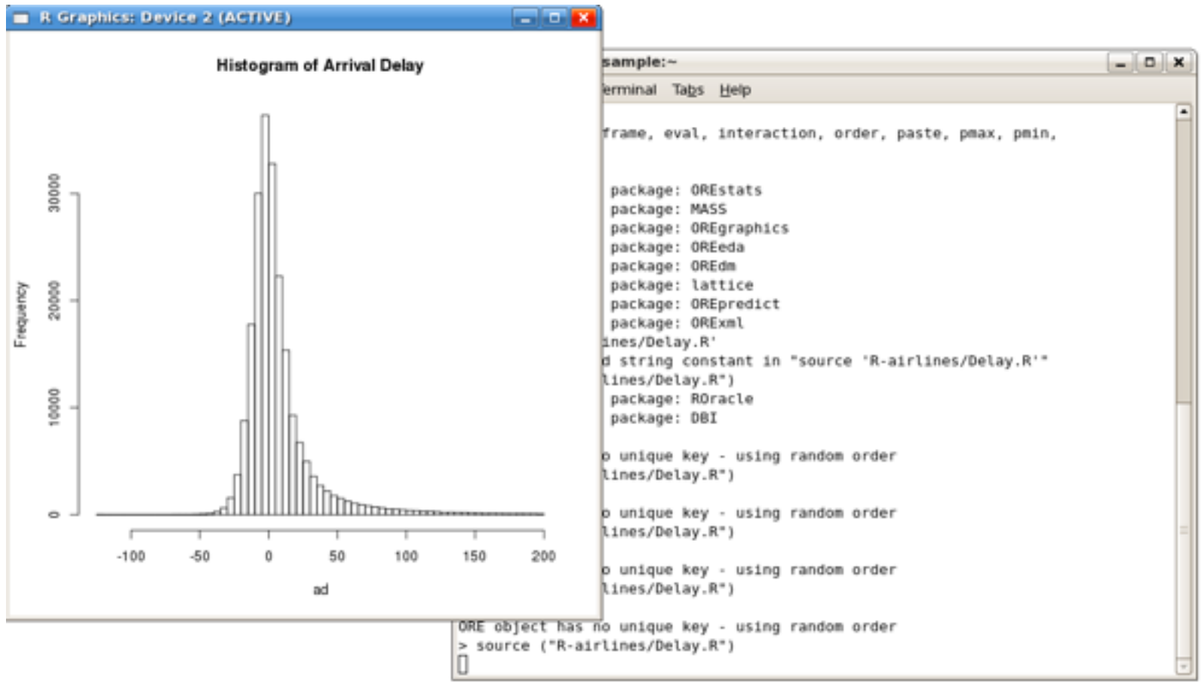
T : +44 (0) 1273 911 268 (UK) **or** (888) 631-1410 (USA) **or**
+61 3 9596 7186 (Australia & New Zealand) **or** +91 997 256 7970 (India)

E : info@rittmanmead.com
W : www.rittmanmead.com

rittmanmead
INTEGRATED ANALYTICS

Initial Data Scoping & Discovery using R

- R is typically used at start of a big data project to get a high-level understanding of the data
- Can be run as R standalone, or using Oracle R Advanced Analytics for Hadoop
- Do basic scan of incoming dataset, get counts, determine delimiters etc
- Distribution of values for columns
- Basic graphs and data discovery
- Use findings to drive design of parsing logic, Hive data structures, need for data scrubbing / correcting etc



T : +44 (0) 1273 911 268 (UK) **or** (888) 631-1410 (USA) **or**
+61 3 9596 7186 (Australia & New Zealand) **or** +91 997 256 7970 (India)

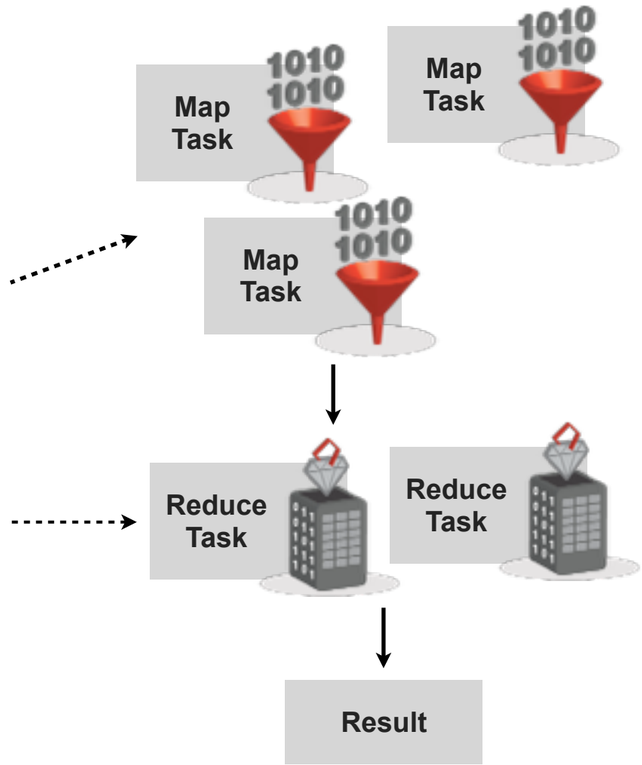
E : info@rittmanmead.com
W : www.rittmanmead.com

rittmanmead
INTEGRATED ANALYTICS

Apache Hive : SQL Access + Table Metadata Over HDFS

- Apache Hive provides a SQL layer over Hadoop, once we understand the structure (schema) of the data we're working with
- Exposes HDFS and other Hadoop data as tables and columns
- Provides a simple SQL dialect for queries called HiveQL
- SQL queries are turned into MapReduce jobs under-the-covers
- JDBC and ODBC drivers provide access to BI and ETL tools
- Hive metastore (data dictionary) leveraged by many other Hadoop tools
  - Apache Pig
  - Cloudera Impala
  - etc

```
SELECT a, sum(b)
FROM myTable
WHERE a<100

GROUP BY a
```



T : +44 (0) 1273 911 268 (UK) **or** (888) 631-1410 (USA) **or**
+61 3 9596 7186 (Australia & New Zealand) **or** +91 997 256 7970 (India)

E : info@rittmanmead.com
W : www.rittmanmead.com

Demo

Hive within Hue on the Big Data Lite VM

T : +44 (0) 1273 911 268 (UK) **or** (888) 631-1410 (USA) **or**
+61 3 9596 7186 (Australia & New Zealand) **or** +91 997 256 7970 (India)

E : info@rittmanmead.com
W : www.rittmanmead.com

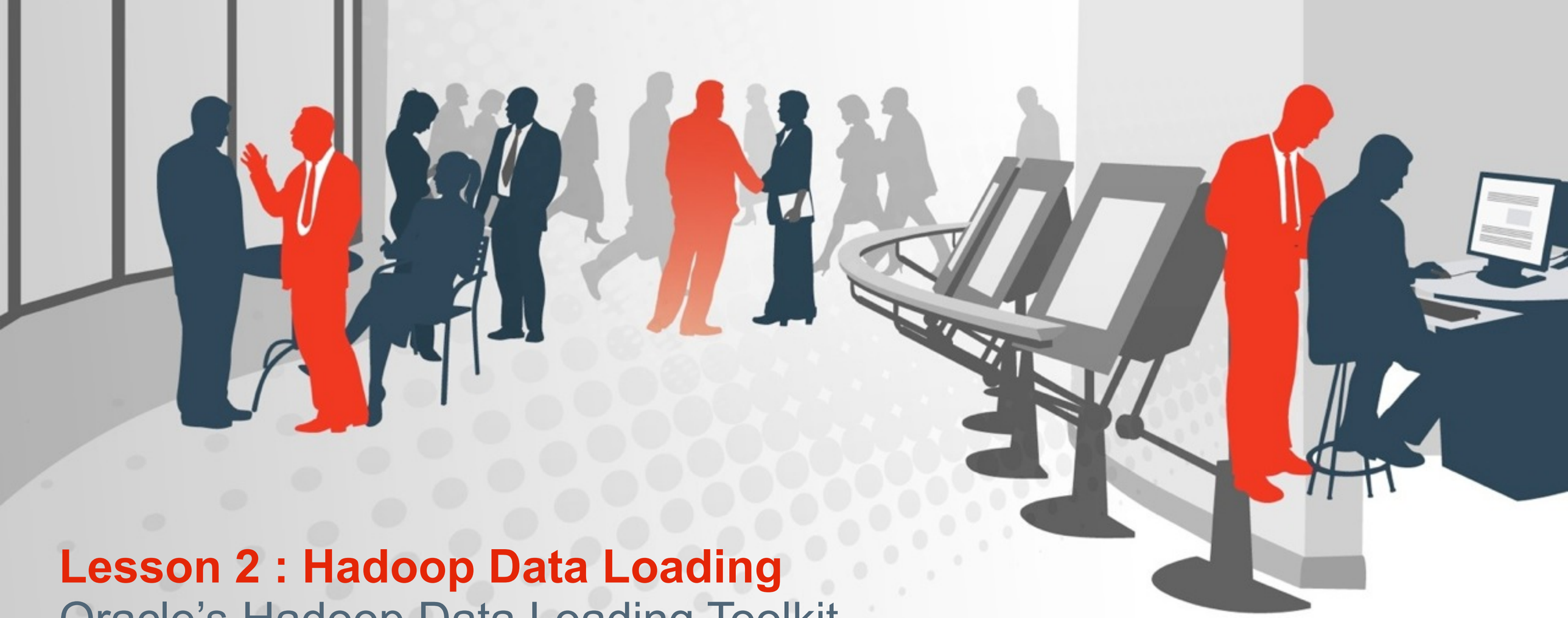
rittmanmead
INTEGRATED ANALYTICS

Hive SerDes & Storage Handlers

- Plug-in technologies that extend Hive to handle new data formats and semi-structured sources
- Typically distributed as JAR files, hosted on sites such as GitHub
- Can be used to parse log files, access data in NoSQL databases, Amazon S3 etc

```
CREATE EXTERNAL TABLE apachelog (
  host STRING,
  identity STRING,
  user STRING,
  time STRING,
  request STRING,
  status STRING,
  size STRING,
  referer STRING,
  agent STRING)
ROW FORMAT SERDE 'org.apache.hadoop.hive.contrib.serde2.RegexSerDe'
WITH SERDEPROPERTIES (
  "input.regex" = "([^ ]*) ([^ ]*) ([^ ]*) (-|\\[[^\\]]*\\])
([^ \"]*|\"[^\"]*\") (-|[0-9]*) (-|[0-9]*)(?: ([^ \"]*|\"[^\"]*\")
([^ \"]*|\"[^\"]*\"))?",
  "output.format.string" = "%1$s %2$s %3$s %4$s %5$s %6$s %7$s %8$s %9$s"
)
STORED AS TEXTFILE
LOCATION '/user/root/logs';
```

```
CREATE TABLE tweet_data(
  interactionId string,
  username string,
  content string,
  author_followers int)
ROW FORMAT SERDE
  'com.mongodb.hadoop.hive.BSONSerDe'
STORED BY
  'com.mongodb.hadoop.hive.MongoStorageHandler'
WITH SERDEPROPERTIES (
  'mongo.columns.mapping'='{"interactionId":"interactionId",
  "username":"interaction.interaction.author.username",
  "content":\"interaction.interaction.content",
  "author_followers_count":"interaction.twitter.user.followers_count"}'
  )
TBLPROPERTIES (
  'mongo.uri'='mongodb://cdh51-node1:27017/datasiftmongodb.rm_tweets'
  )
```



Lesson 2 : Hadoop Data Loading

Oracle's Hadoop Data Loading Toolkit

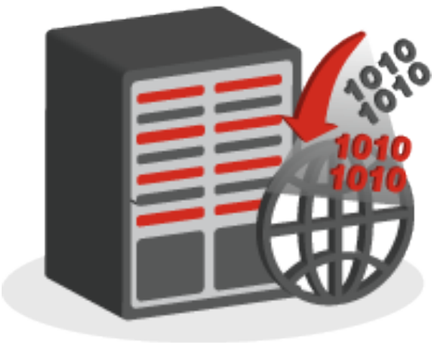
T : +44 (0) 1273 911 268 (UK) **or** (888) 631-1410 (USA) **or**
+61 3 9596 7186 (Australia & New Zealand) **or** +91 997 256 7970 (India)

E : info@rittmanmead.com
W : www.rittmanmead.com

rittmanmead
INTEGRATED ANALYTICS

Oracle's Big Data Products

- Oracle Big Data Appliance - Engineered System for Big Data Acquisition and Processing
  - Cloudera Distribution of Hadoop
  - Cloudera Manager
  - Open-source R
  - Oracle NoSQL Database
  - Oracle Enterprise Linux + Oracle JVM
  - New - Oracle Big Data SQL
- Oracle Big Data Connectors
  - Oracle Loader for Hadoop (Hadoop > Oracle RBDMS)
  - Oracle Direct Connector for HDFS (HFDS > Oracle RBDMS)
  - Oracle R Advanced Analytics for Hadoop
- Oracle Data Integrator 12c



Oracle Big Data Connectors

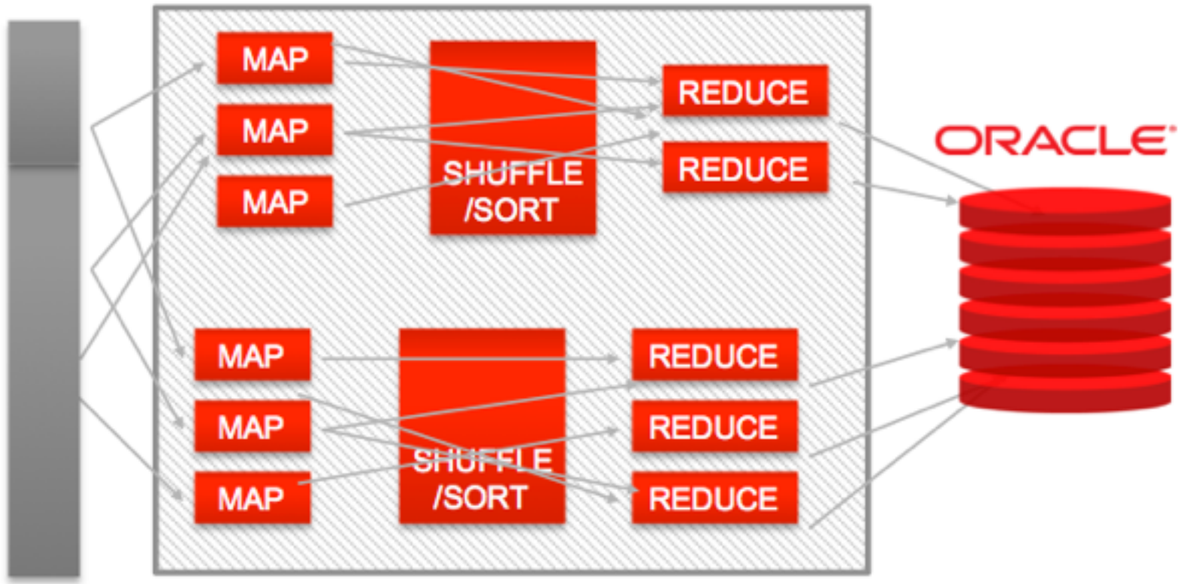
- Oracle-licensed utilities to connect Hadoop to Oracle RBDMS
  - Bulk-extract data from Hadoop to Oracle, or expose HDFS / Hive data as external tables
  - Run R analysis and processing on Hadoop
- Leverage Hadoop compute resources to offload ETL and other work from Oracle RBDMS
- Enable Oracle SQL to access and load Hadoop data

T : +44 (0) 1273 911 268 (UK) **or** (888) 631-1410 (USA) **or**
+61 3 9596 7186 (Australia & New Zealand) **or** +91 997 256 7970 (India)

E : info@rittmanmead.com
W : www.rittmanmead.com

Oracle Loader for Hadoop

- Oracle technology for accessing Hadoop data, and loading it into an Oracle database
- Pushes data transformation, “heavy lifting” to the Hadoop cluster, using MapReduce
- Direct-path loads into Oracle Database, partitioned and non-partitioned
- Online and offline loads
- Key technology for fast load of Hadoop results into Oracle DB



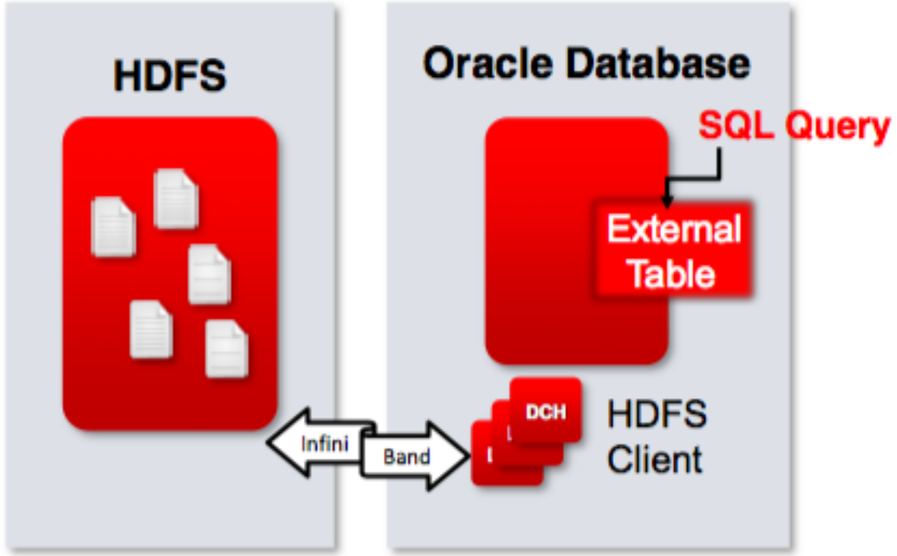
T : +44 (0) 1273 911 268 (UK) **or** (888) 631-1410 (USA) **or**
+61 3 9596 7186 (Australia & New Zealand) **or** +91 997 256 7970 (India)

E : info@rittmanmead.com
W : www.rittmanmead.com

rittmanmead
INTEGRATED ANALYTICS

Oracle Direct Connector for HDFS

- Enables HDFS as a data-source for Oracle Database external tables
- Effectively provides Oracle SQL access over HDFS
- Supports data query, or import into Oracle DB
- Treat HDFS-stored files in the same way as regular files
  - But with HDFS's low-cost
  - … and fault-tolerance

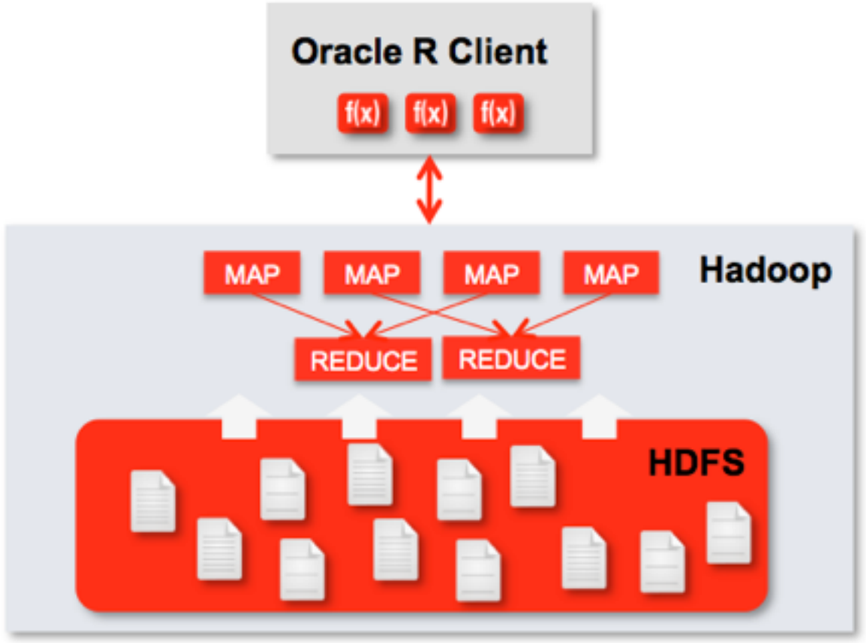


T : +44 (0) 1273 911 268 (UK) **or** (888) 631-1410 (USA) **or**
+61 3 9596 7186 (Australia & New Zealand) **or** +91 997 256 7970 (India)

E : info@rittmanmead.com
W : www.rittmanmead.com

Oracle R Advanced Analytics for Hadoop

- Add-in to R that extends capability to Hadoop
- Gives R the ability to create Map and Reduce functions
- Extends R data frames to include Hive tables
  - Automatically run R functions on Hadoop by using Hive tables as source



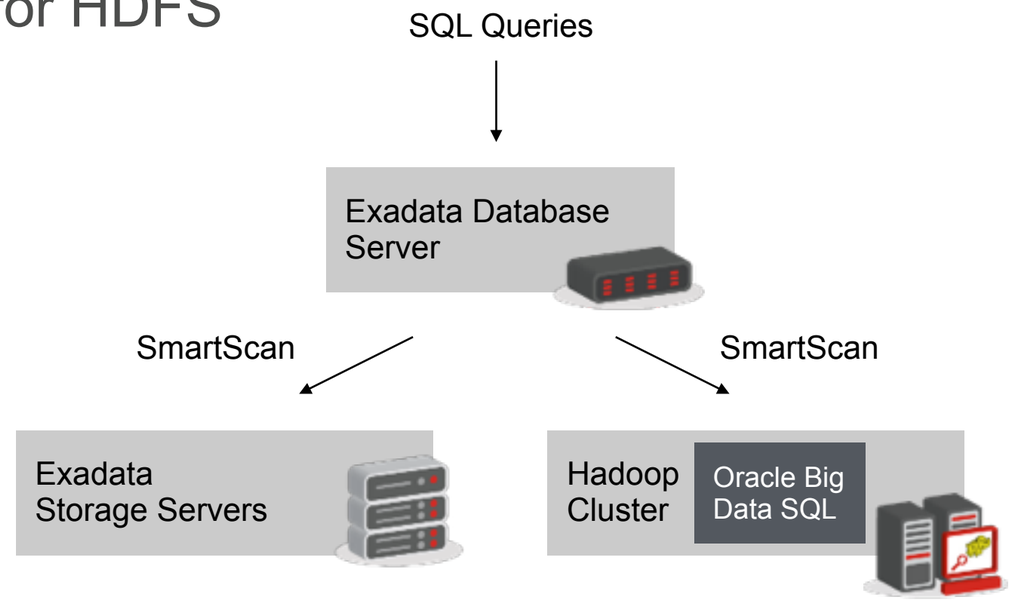
T : +44 (0) 1273 911 268 (UK) **or** (888) 631-1410 (USA) **or**
+61 3 9596 7186 (Australia & New Zealand) **or** +91 997 256 7970 (India)

E : info@rittmanmead.com
W : www.rittmanmead.com

rittmanmead
INTEGRATED ANALYTICS

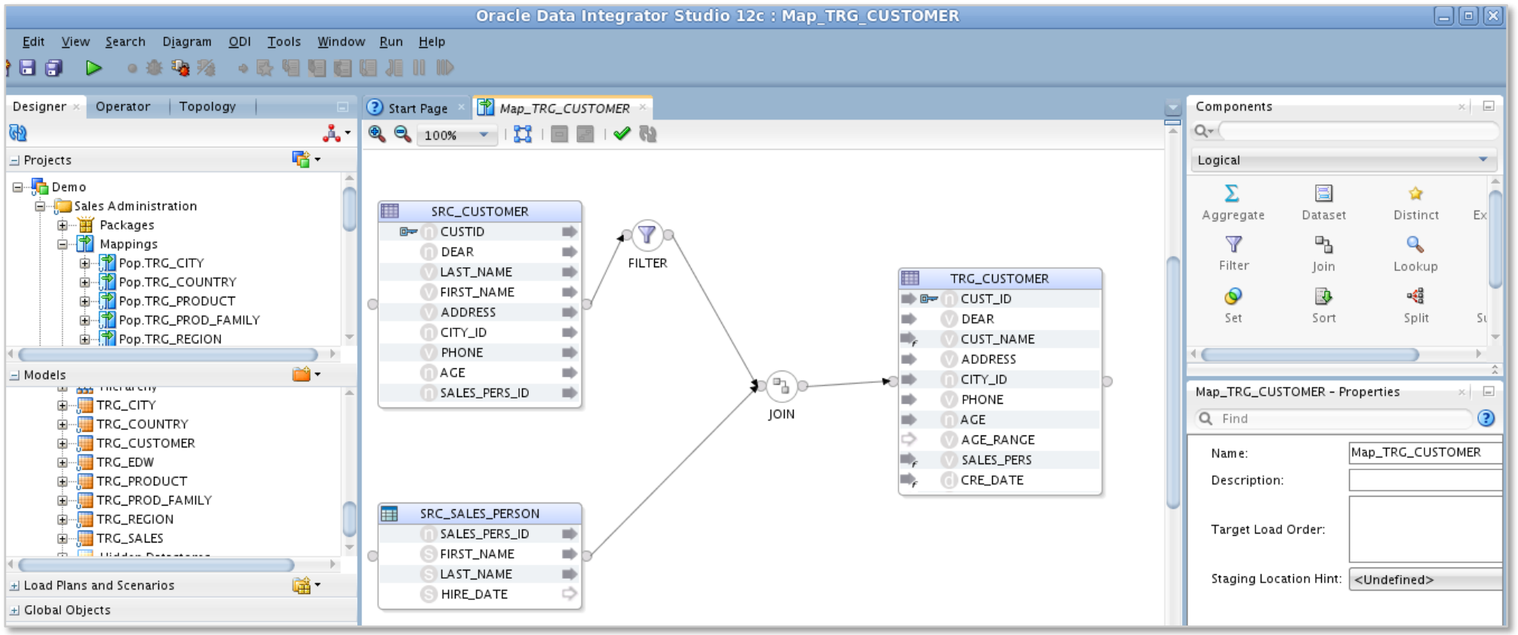
Oracle Big Data SQL

- Part of Oracle Big Data 4.0 (BDA-only)
  - Also requires Oracle Database 12c, Oracle Exadata Database Machine
- Extends Oracle Data Dictionary to cover Hive
- Extends Oracle SQL and SmartScan to Hadoop
  - More efficient access than Oracle Direct Connector for HDFS
- Extends Oracle Security Model over Hadoop
  - Fine-grained access control
  - Data redaction, data masking



Oracle Data Integrator 12c

- Oracle's data integration tool for loading, transforming and integrating enterprise data
- Successor to Oracle Warehouse Builder, part of wider Oracle DI platform
- Connectivity to most RBDMS, file and application sources



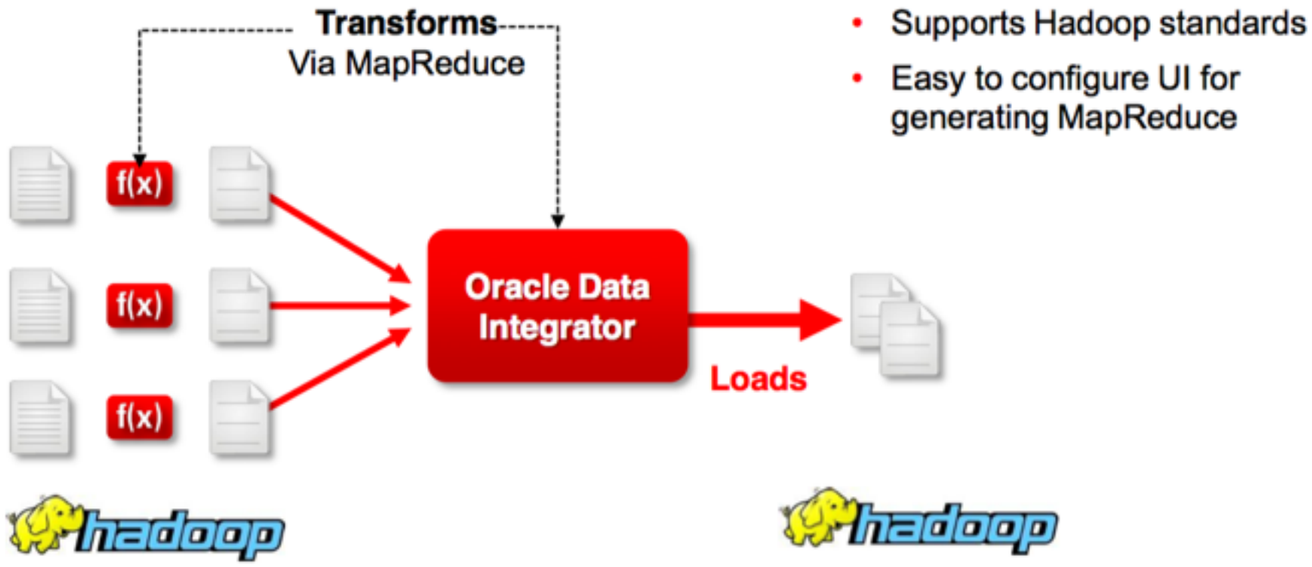
T : +44 (0) 1273 911 268 (UK) **or** (888) 631-1410 (USA) **or**
+61 3 9596 7186 (Australia & New Zealand) **or** +91 997 256 7970 (India)

E : info@rittmanmead.com
W : www.rittmanmead.com

rittmanmead
INTEGRATED ANALYTICS

Oracle Data Integrator on Hadoop

- ODI provides an excellent framework for running Hadoop ETL jobs
  - ELT approach pushes transformations down to Hadoop - leveraging power of cluster
- Hive, HBase, Sqoop and OLH/ODCH KMs provide native Hadoop loading / transformation
  - Whilst still preserving RDBMS push-down
  - Extensible to cover Pig, Spark etc
- Process orchestration
- Data quality / error handling
- Metadata and model-driven



T : +44 (0) 1273 911 268 (UK) **or** (888) 631-1410 (USA) **or**
+61 3 9596 7186 (Australia & New Zealand) **or** +91 997 256 7970 (India)

E : info@rittmanmead.com
W : www.rittmanmead.com

rittmanmead INTEGRATED ANALYTICS

Demo

ODI12c within the Big Data Lite VM

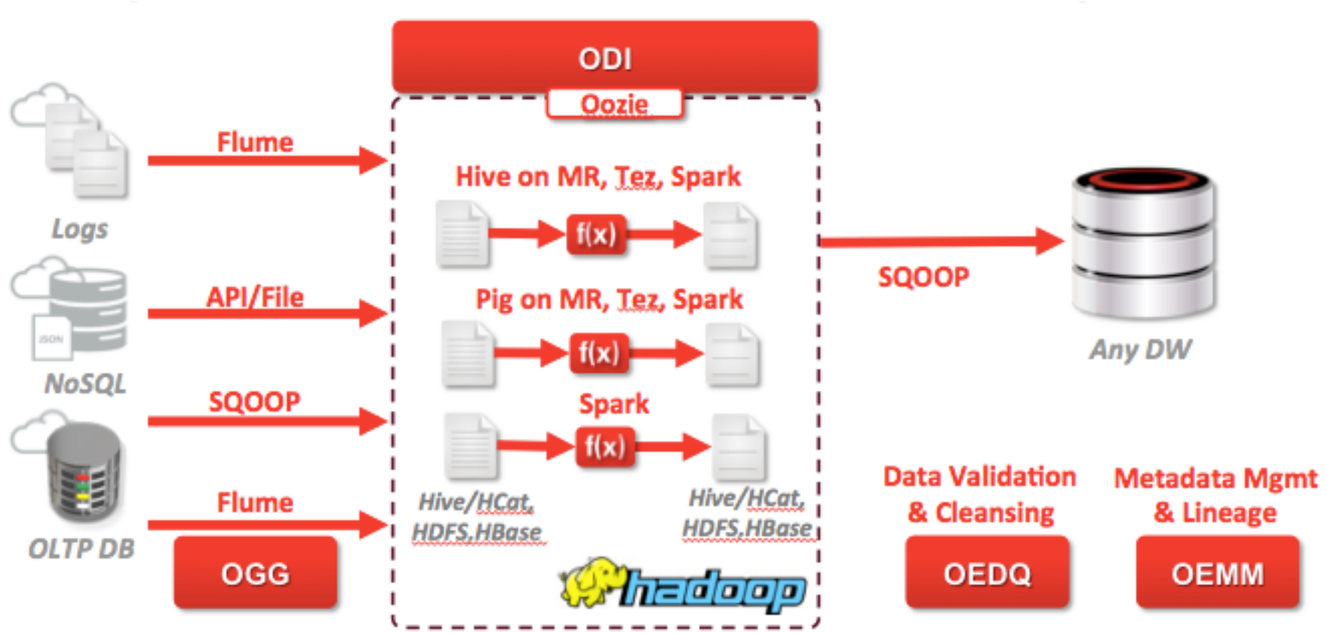
T : +44 (0) 1273 911 268 (UK) **or** (888) 631-1410 (USA) **or**
+61 3 9596 7186 (Australia & New Zealand) **or** +91 997 256 7970 (India)

E : info@rittmanmead.com
W : www.rittmanmead.com

rittmanmead
INTEGRATED ANALYTICS

Native Processing using MapReduce, Pig, Hive etc

- Native processing using Hadoop framework, using Knowledge Module code templates
- ODI generates native code for each platform, taking a template for each step + adding table names, column names, join conditions etc
  - Easy to extend
  - Easy to read the code
  - Makes it possible for ODI to support Spark, Pig etc in future
  - Uses the power of the target platform for integration tasks
    - Hadoop-native ETL



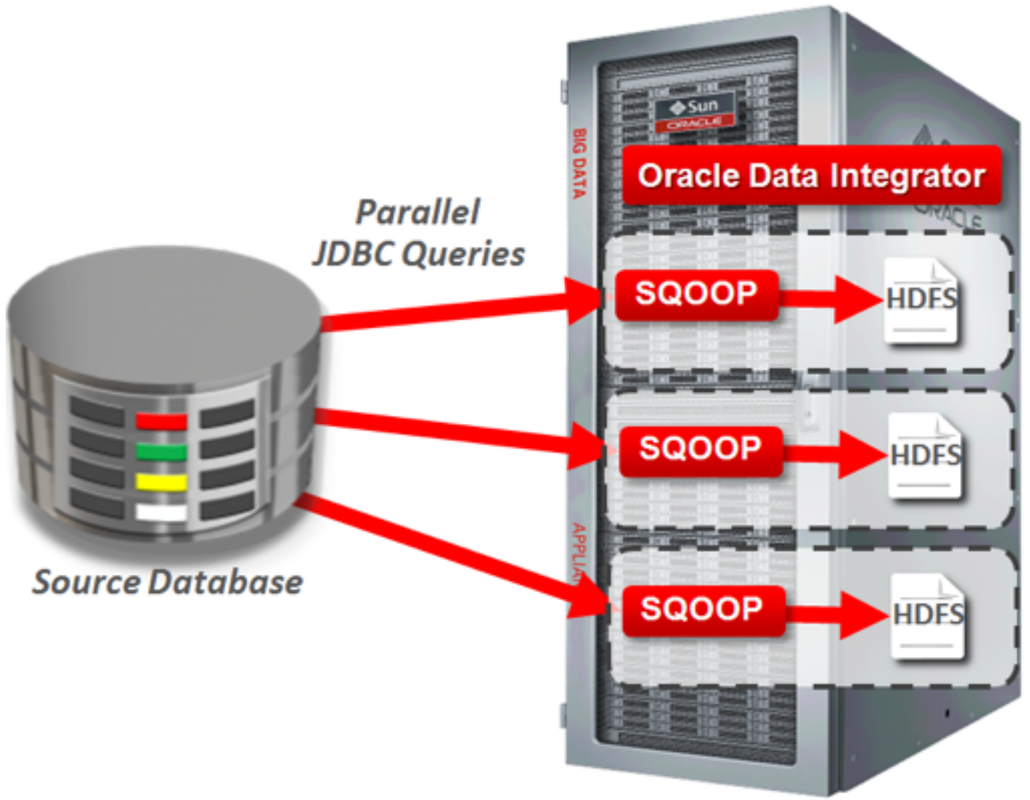
T : +44 (0) 1273 911 268 (UK) **or** (888) 631-1410 (USA) **or**
+61 3 9596 7186 (Australia & New Zealand) **or** +91 997 256 7970 (India)

E : info@rittmanmead.com
W : www.rittmanmead.com

rittmanmead
INTEGRATED ANALYTICS

Bulk Data Loading into Hadoop through Sqoop, Files

- ODI12c 12.1.3 comes with Sqoop support, for bulk-import and export out of RDBMS
  - Preferred method for bulk-loading database sourced data
- File loading can be done through IKM File to Hive, or through Hadoop FS shell

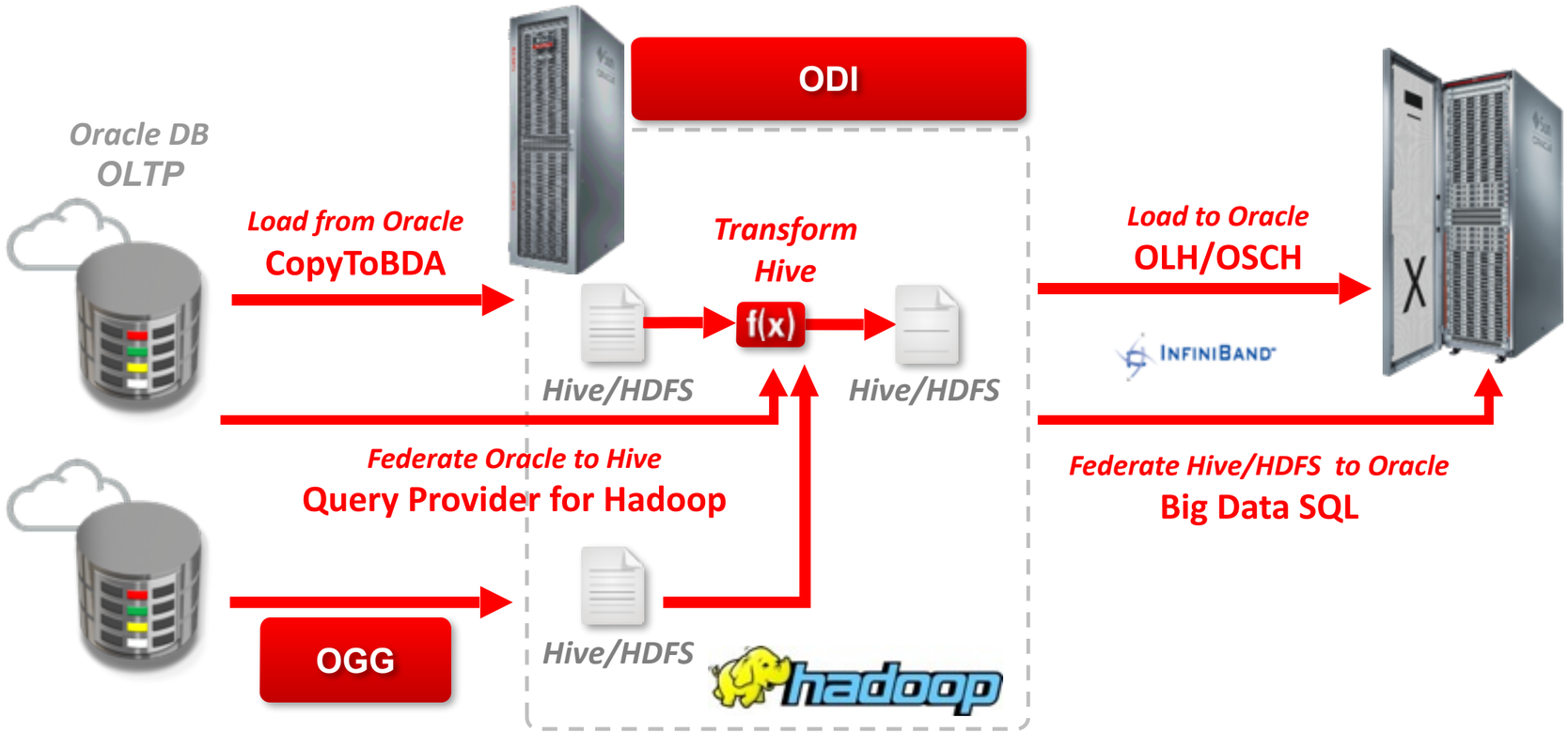


T : +44 (0) 1273 911 268 (UK) **or** (888) 631-1410 (USA) **or**
+61 3 9596 7186 (Australia & New Zealand) **or** +91 997 256 7970 (India)

E : info@rittmanmead.com
W : www.rittmanmead.com

ODI Integration with Oracle Big Data Adapters & GoldenGate

- GoldenGate (and Flume) for data loading
- OLH and ODCH for data exporting
- ORAAH for analysis



T : +44 (0) 1273 911 268 (UK) **or** (888) 631-1410 (USA) **or**
+61 3 9596 7186 (Australia & New Zealand) **or** +91 997 256 7970 (India)

E : info@rittmanmead.com
W : www.rittmanmead.com

rittmanmead
INTEGRATED ANALYTICS

Loading from NoSQL Sources

- NoSQL databases are often used in conjunction with Hadoop
- Typically provide a flexible schema vs No schema (HDFS) or tabular schema (Hive)
- Usually provide CRUD capabilities vs. HDFS's write-only storage
- Typical use-cases include
  - High-velocity event loading (Oracle NoSQL Database)
  - Providing a means to support CRUD over HDFS (HBase)
  - Loading JSON documents (MongoDB)
- NoSQL data access usually through APIs
  - Primarily aimed add app developers
- Hive storage handlers and other solutions can be used in a BI / DW / ETL context
  - HBase support in ODI12c 12.1.3

Lesson 2 : Hadoop Data Loading

Example Scenario : Webserver Log Analysis

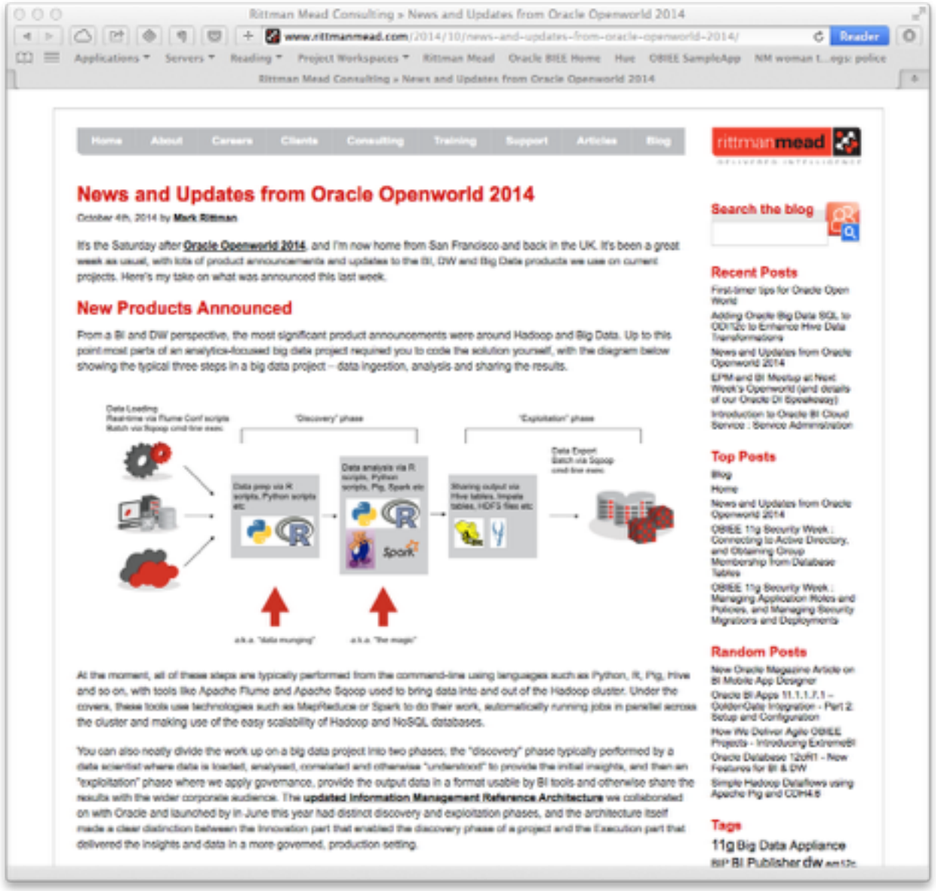
T : +44 (0) 1273 911 268 (UK) **or** (888) 631-1410 (USA) **or**
+61 3 9596 7186 (Australia & New Zealand) **or** +91 997 256 7970 (India)

E : info@rittmanmead.com
W : www.rittmanmead.com

rittmanmead
INTEGRATED ANALYTICS

Scenario Overview

- Rittman Mead website hosts the Rittman Mead blog, plus service offerings, news, article downloads etc
- Typical traffic is around 4-6k pageviews / day
- Hosted on Amazon AWS, runs on Wordpress
- We would like to better understand site activity
  - Which pages are most popular?
  - Where do our visitors come from?
  - Which blog articles and authors are most popular?
  - What other activity around the blog (social media etc) influences traffic on the site?



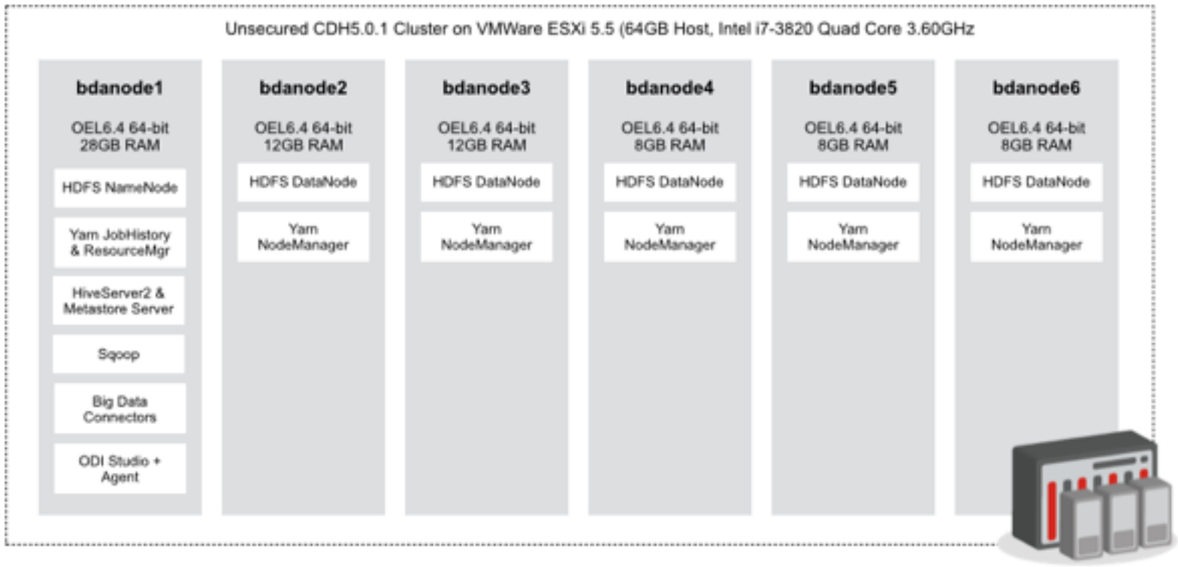
T : +44 (0) 1273 911 268 (UK) **or** (888) 631-1410 (USA) **or**
+61 3 9596 7186 (Australia & New Zealand) **or** +91 997 256 7970 (India)

E : info@rittmanmead.com
W : www.rittmanmead.com

rittmanmead
INTEGRATED ANALYTICS

ODI and Big Data Integration Example

- In this seminar, we'll show an end-to-end ETL process on Hadoop using ODI12c & BDA
- Load webserver log data into Hadoop, process enhance and aggregate, then load final summary table into Oracle Database 12c
- Originally developed on full Hadoop cluster, but ported to BigDataLite 4.0 VM for seminar
  - Process using Hadoop framework
  - Leverage Big Data Connectors
  - Metadata-based ETL development using ODI12c
  - Real-world example



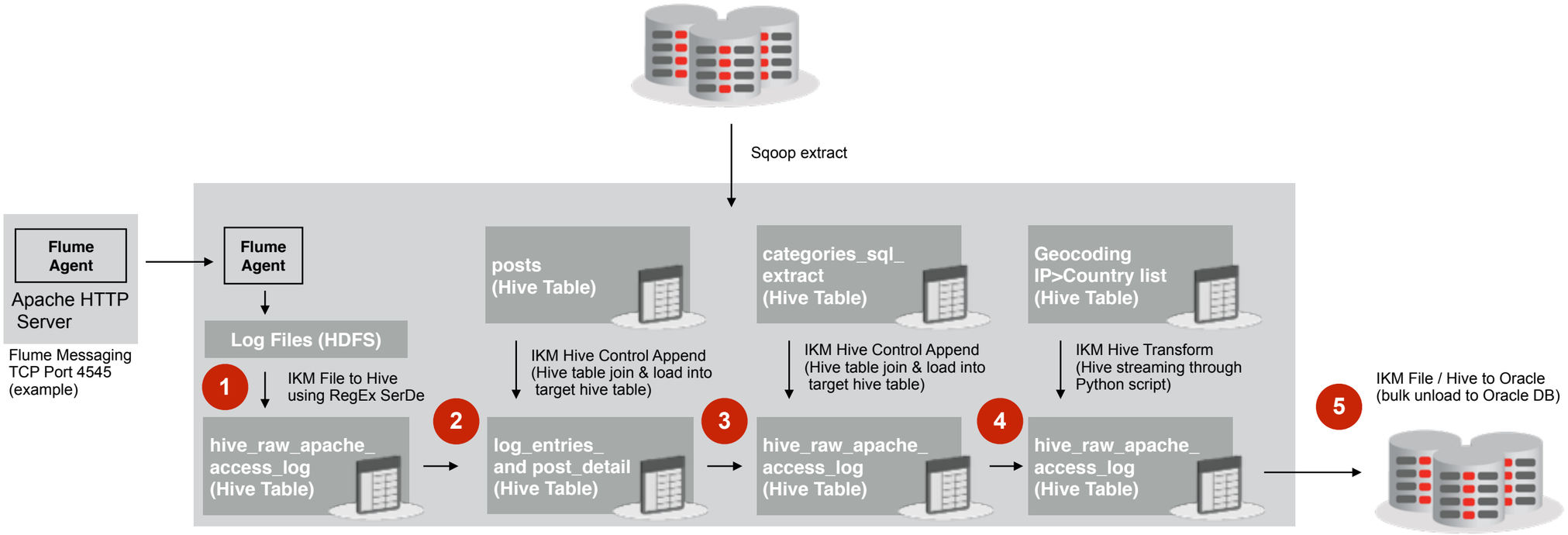
T : +44 (0) 1273 911 268 (UK) **or** (888) 631-1410 (USA) **or**
+61 3 9596 7186 (Australia & New Zealand) **or** +91 997 256 7970 (India)

E : info@rittmanmead.com
W : www.rittmanmead.com

rittmanmead
INTEGRATED ANALYTICS

Initial ETL & Data Flow through BDA System

- Five-step process to load, transform, aggregate and filter incoming log data
- Leverage ODI's capabilities where possible
- Make use of Hadoop power + scalability



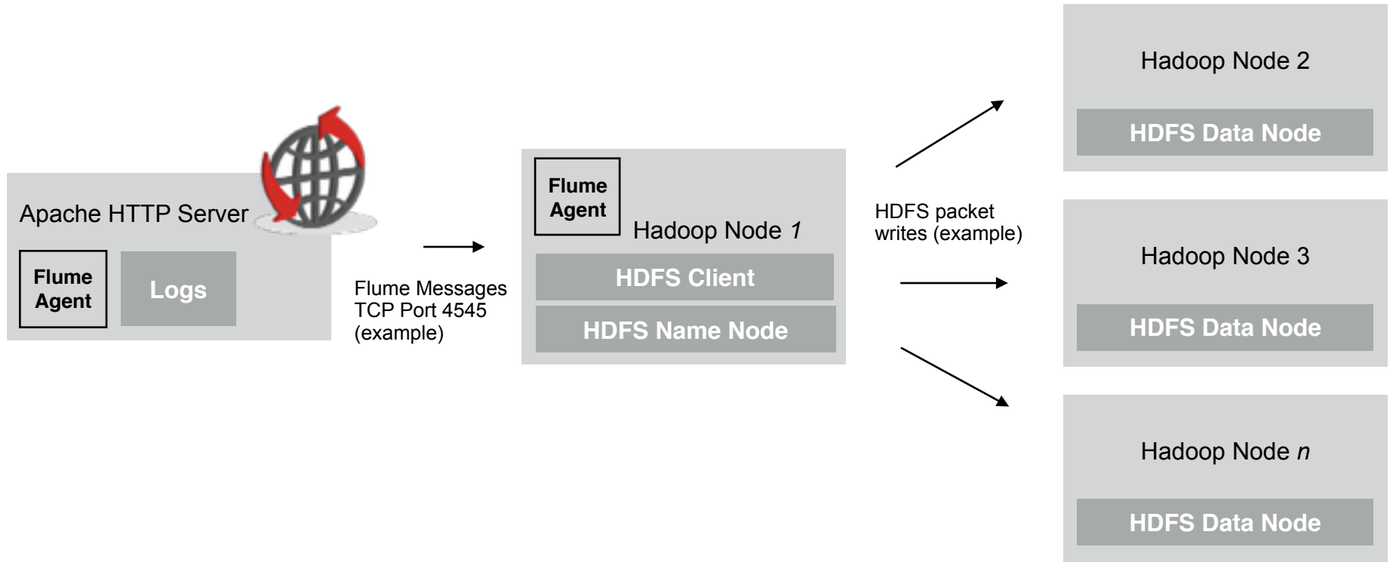
T : +44 (0) 1273 911 268 (UK) **or** (888) 631-1410 (USA) **or**
+61 3 9596 7186 (Australia & New Zealand) **or** +91 997 256 7970 (India)

E : info@rittmanmead.com
W : www.rittmanmead.com

rittmanmead INTEGRATED ANALYTICS

1 Load Incoming Log Files into Hadoop using Flume

- Web server log entries will be ingested into Hadoop using Flume
- Flume collector configured on webserver, sink on Hadoop node(s)
- Log activity buffered using Flume channels
- Effectively replicates in real-time the log activity on RM webserver



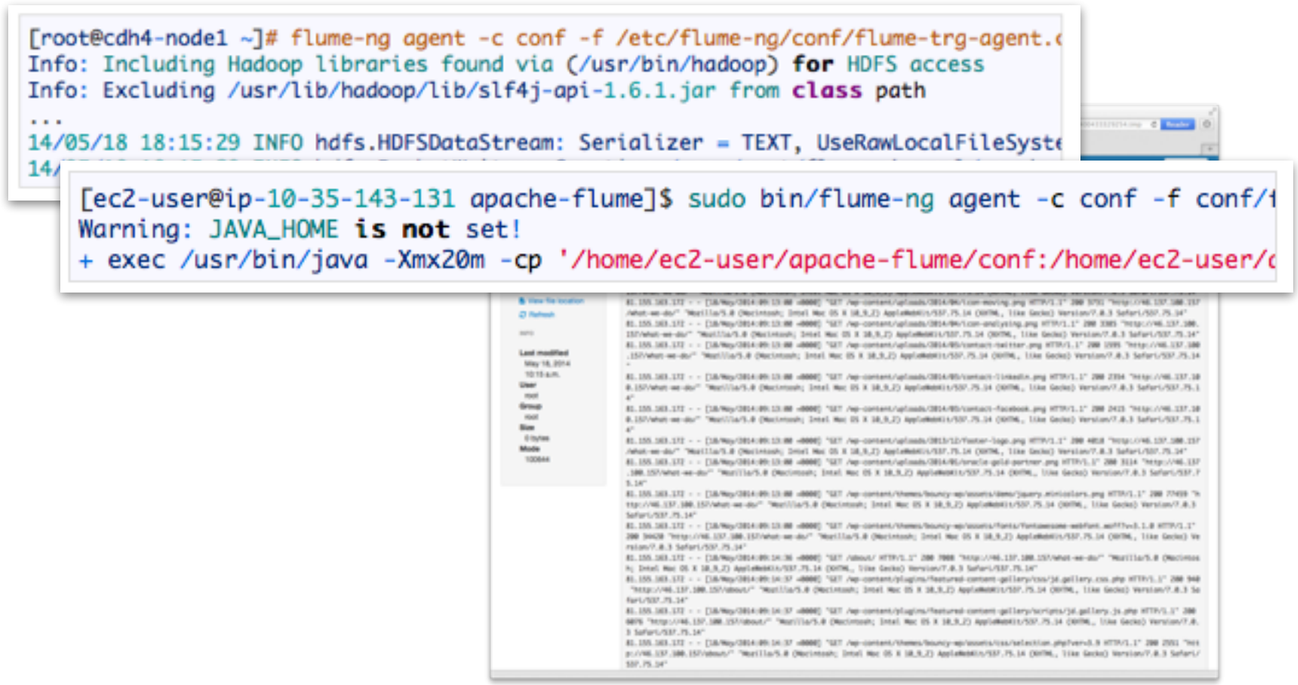
T : +44 (0) 1273 911 268 (UK) **or** (888) 631-1410 (USA) **or**
+61 3 9596 7186 (Australia & New Zealand) **or** +91 997 256 7970 (India)

E : info@rittmanmead.com
W : www.rittmanmead.com

rittmanmead INTEGRATED ANALYTICS

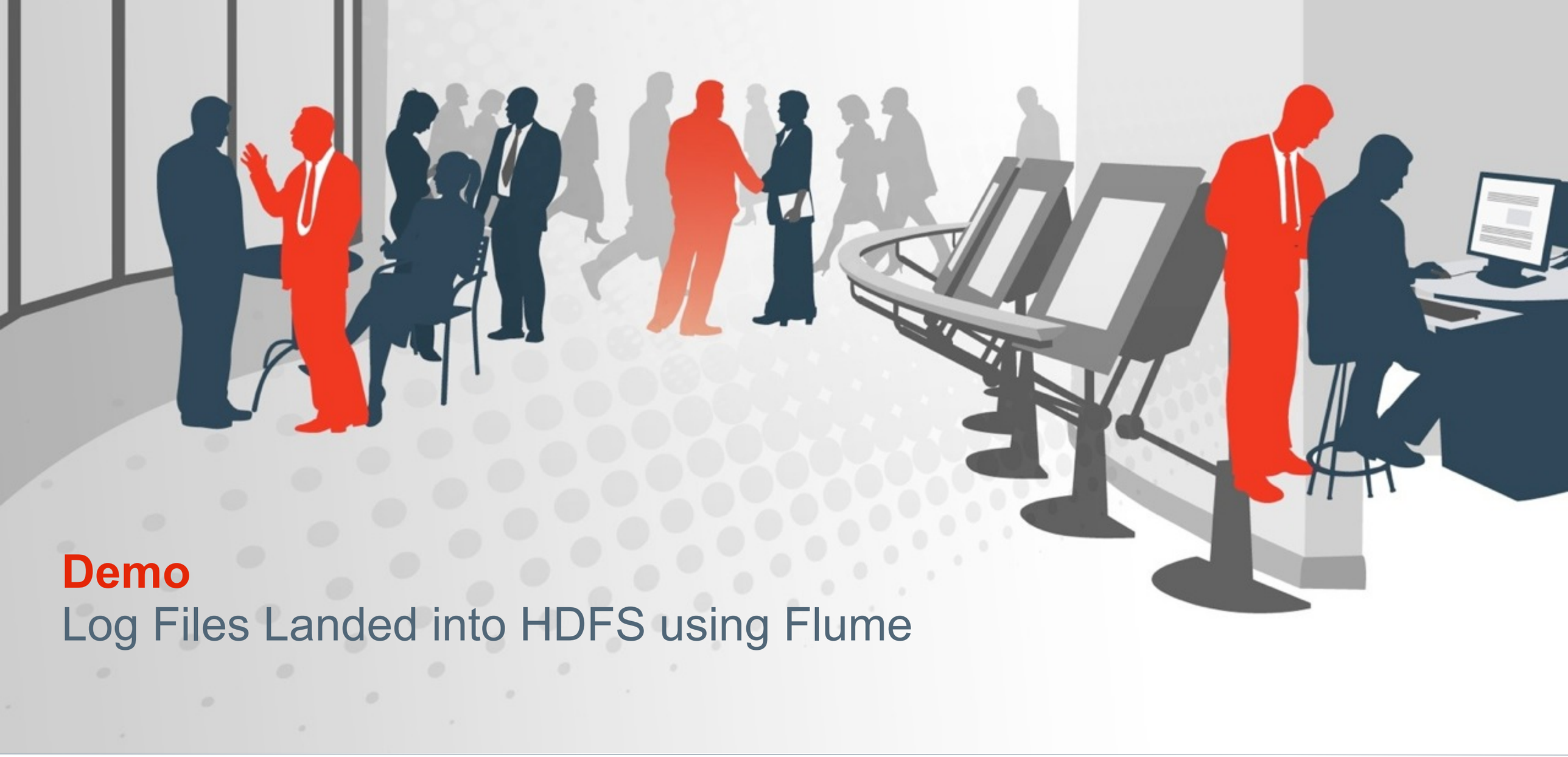
Starting Flume Agents, Check Files Landing in HDFS Directory

- Start the Flume agents on source and target (BDA) servers
- Check that incoming file data starts appearing in HDFS
  - Note - files will be continuously written-to as entries added to source log files
  - Channel size for source, target agents determines max no. of events buffered
  - If buffer exceeded, new events dropped until buffer < channel size



T : +44 (0) 1273 911 268 (UK) **or** (888) 631-1410 (USA) **or**
+61 3 9596 7186 (Australia & New Zealand) **or** +91 997 256 7970 (India)

E : info@rittmanmead.com
W : www.rittmanmead.com



Demo

Log Files Landed into HDFS using Flume

T : +44 (0) 1273 911 268 (UK) **or** (888) 631-1410 (USA) **or**
+61 3 9596 7186 (Australia & New Zealand) **or** +91 997 256 7970 (India)

E : info@rittmanmead.com
W : www.rittmanmead.com

rittmanmead
INTEGRATED ANALYTICS

Initial Data Discovery / Exploration using R

- Run basic analysis and output high-level metrics on the incoming data
- Copy sample of incoming log files (via Flume) to local filesystem for analysis
  - Or use ORAAH to access them in HDFS directly

```
[oracle@bigdatalite ~]$ hadoop fs -ls /user/oracle/rm_logs
Found 5 items
-rwxr-xr-x   1 oracle oracle   41364846 2014-09-27 23:49 /user/oracle/rm_logs/access_log
-rwxr-xr-x   1 oracle oracle  359299529 2014-09-27 23:53 /user/oracle/rm_logs/access_log-20140323
-rwxr-xr-x   1 oracle oracle  364337863 2014-09-27 23:53 /user/oracle/rm_logs/access_log-20140330
-rwxr-xr-x   1 oracle oracle  350690205 2014-09-27 23:53 /user/oracle/rm_logs/access_log-20140406
-rwxr-xr-x   1 oracle oracle  328643100 2014-09-27 23:53 /user/oracle/rm_logs/access_log-20140413
[oracle@bigdatalite ~]$ hadoop fs -copyToLocal /user/oracle/rm_logs/access_log-20140323 $HOME/sample_logs
```

- Do initial count to check how many rows in file

```
[oracle@bigdatalite ~]$ R
> logfilename <- "/home/oracle/sample_logs/access_log-20140323"
> length(readLines(logfilename))
[1] 1435524
```

Initial Data Discovery / Exploration using R

- Display range of values for log file elements, split on the standard Apache log delimiter char
- Use output to determine potential column names

```
> df <- read.table(logfilename, colClasses="character", header=FALSE, sep="", quote="\"'")
> str(df)
'data.frame':              1435524 obs. of  10 variables:
 $ V1 : chr  "103.255.250.7" "54.228.204.102" "170.148.198.157" "184.171.240.84" ...
 $ V2 : chr  "-" "-" "-" "-" ...
 $ V3 : chr  "-" "-" "-" "-" ...
 $ V4 : chr  "[16/Mar/2014:03:19:24" "[16/Mar/2014:03:19:30" "[16/Mar/2014:03:19:30" "[16/Mar/2014:03:19:40" ...
 $ V5 : chr  "+0000]" "+0000]" "+0000]" "+0000]" ...
 $ V6 : chr  "GET / HTTP/1.0" "POST /wp-cron.php?doing_wp_cron=1394939970.6438250541687011718750 HTTP/1.0" "GET /feed/
HTTP/1.1" "POST /wp-login.php HTTP/1.0" ...
 $ V7 : chr  "301" "200" "200" "200" ...
 $ V8 : chr  "235" "-" "36256" "3529" ...
 $ V9 : chr  "-" "-" "-" "-" ...
 $ V10: chr  "Mozilla/5.0 (compatible; monitis.com - free monitoring service; http://monitis.com)" "WordPress/3.7.1;
http://www.rittmanmead.com" "Mozilla/5.0 (Windows NT 6.1; WOW64; rv:10.0.9) Gecko/20100101 Firefox/10.0.9" "-" ...
```

Initial Data Discovery / Exploration using R

- Apply column names to data frame based on Apache Combined Log Format
- Convert date/time column to date datatype

```
> colnames(df) = c('host','ident','authuser','datetime','timezone','request','status','bytes','referer','browser')
> df$datetime <- as.Date(df$datetime, "[%d/%b/%Y:%H:%M:%S")
> str(df)
'data.frame':             1435524 obs. of  10 variables:
 $ host    : chr  "103.255.250.7" "54.228.204.102" "170.148.198.157" "184.171.240.84" ...
 $ ident   : chr  "-" "-" "-" "-" ...
 $ authuser: chr  "-" "-" "-" "-" ...
 $ datetime: Date, format: "2014-03-16" "2014-03-16" ...
 $ timezone: chr  "+0000]" "+0000]" "+0000]" "+0000]" ...
 $ request : chr  "GET / HTTP/1.0" "POST /wp-cron.php?doing_wp_cron=1394939970.6438250541687011718750 HTTP/1.0"
"GET /feed/ HTTP/1.1" "POST /wp-login.php HTTP/1.0" ...
 $ status  : chr  "301" "200" "200" "200" ...
 $ bytes   : chr  "235" "-" "36256" "3529" ...
 $ referer : chr  "-" "-" "-" "-" ...
 $ browser : chr  "Mozilla/5.0 (compatible; monitis.com - free monitoring service; http://monitis.com)"
"WordPress/3.7.1; http://www.rittmanmead.com" "Mozilla/5.0 (Windows NT 6.1; WOW64; rv:10.0.9) Gecko/20100101 Firefox/10.0.9"
"-" ...
```

Initial Data Discovery / Exploration using R

- Display first *n* rows from the data frame

```
> head(df,3)
             host ident authuser   datetime timezone
1   103.255.250.7     -        - 2014-03-16   +0000]
2  54.228.204.102     -        - 2014-03-16   +0000]
3 170.148.198.157     -        - 2014-03-16   +0000]
                                                                  request
1                                                                GET / HTTP/1.0
2 POST /wp-cron.php?doing_wp_cron=1394939970.6438250541687011718750 HTTP/1.0
3                                                           GET /feed/ HTTP/1.1
  status bytes referer
1    301   235       -
2    200     -       -
3    200 36256       -
                                                                              browser
1 Mozilla/5.0 (compatible; monitis.com - free monitoring service; http://monitis.com)
2                                          WordPress/3.7.1; http://www.rittmanmead.com
3        Mozilla/5.0 (Windows NT 6.1; WOW64; rv:10.0.9) Gecko/20100101 Firefox/10.0.9
```

Initial Data Discovery / Exploration using R

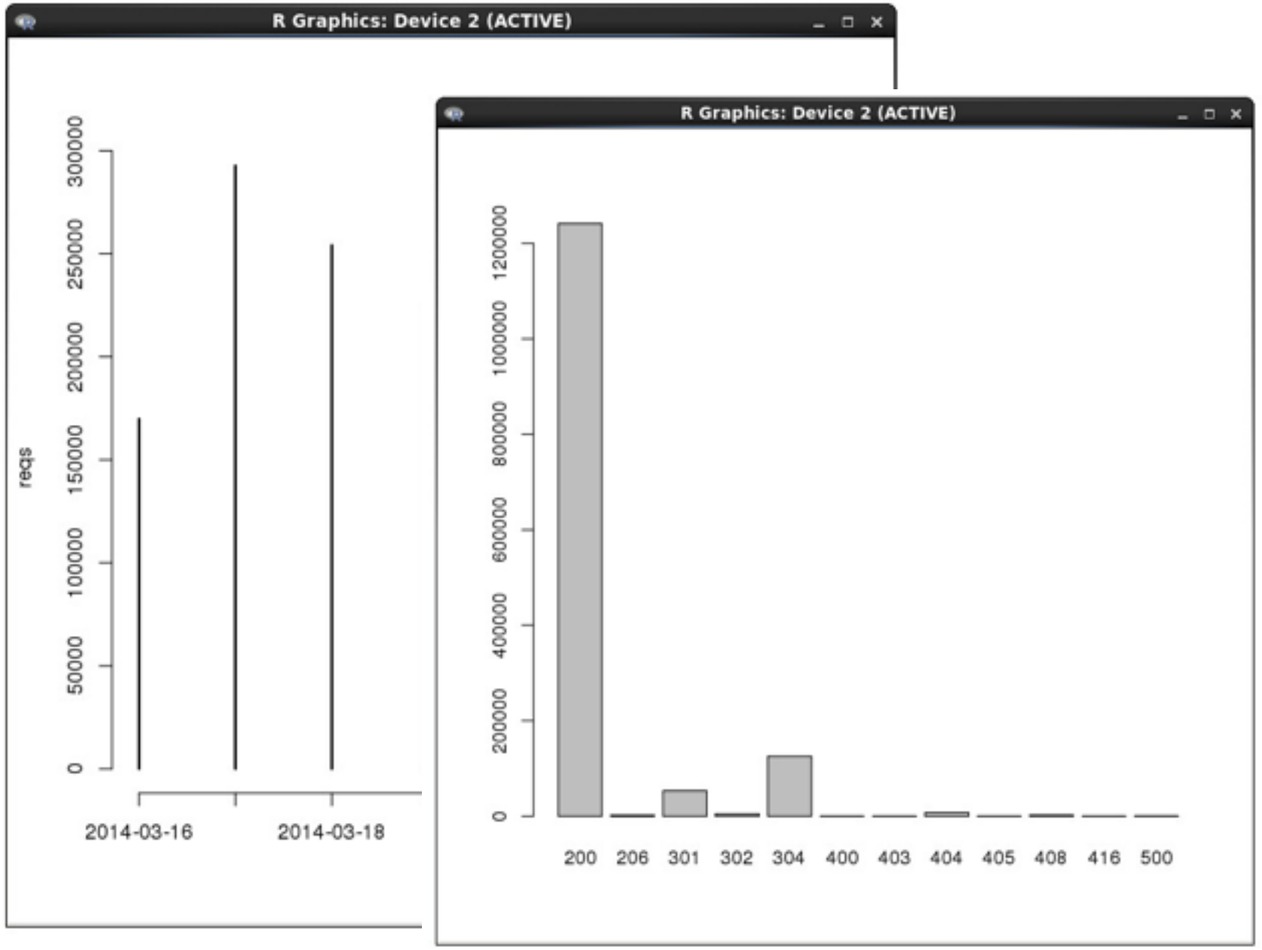
- Display quick charts of column values
  - Determine high and low values
  - Understand distribution etc

```
> reqs <- table(df$datetime)
> plot(reqs)
> status <- table(df$status)
> barplot(status)
```

- Display count of unique site visitors

```
> length(unique(df$host))
[1] 33147
```

- Run similar queries until point where basic structure + content of data is understood



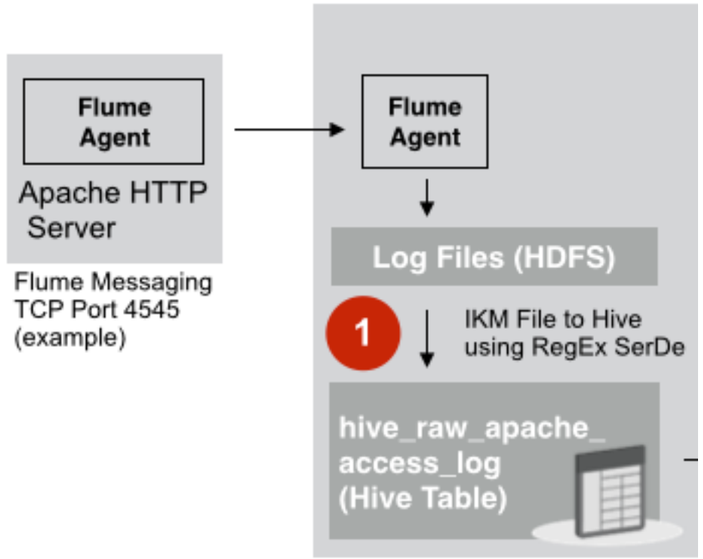
T : +44 (0) 1273 911 268 (UK) **or** (888) 631-1410 (USA) **or**
+61 3 9596 7186 (Australia & New Zealand) **or** +91 997 256 7970 (India)

E : info@rittmanmead.com
W : www.rittmanmead.com

rittmanmead
INTEGRATED ANALYTICS

Parse and Process Log Files into Structured Hive Tables

- Next step in process is to load the incoming log files into a Hive table
  - Provides structure to data, makes it easier to access individual log elements
  - Also need to parse the log entries to extract request, date, IP address etc columns
  - Hive table can then easily be used in downstream transformations
- Option #1 : Use ODI12c **IKM File to Hive (LOAD DATA)** KM
  - Source can be local files or HDFS
  - Either load file into Hive HDFS area, or leave as external Hive table
  - Ability to use SerDe to parse file data
- Option #2 : Define Hive table manually using SerDe

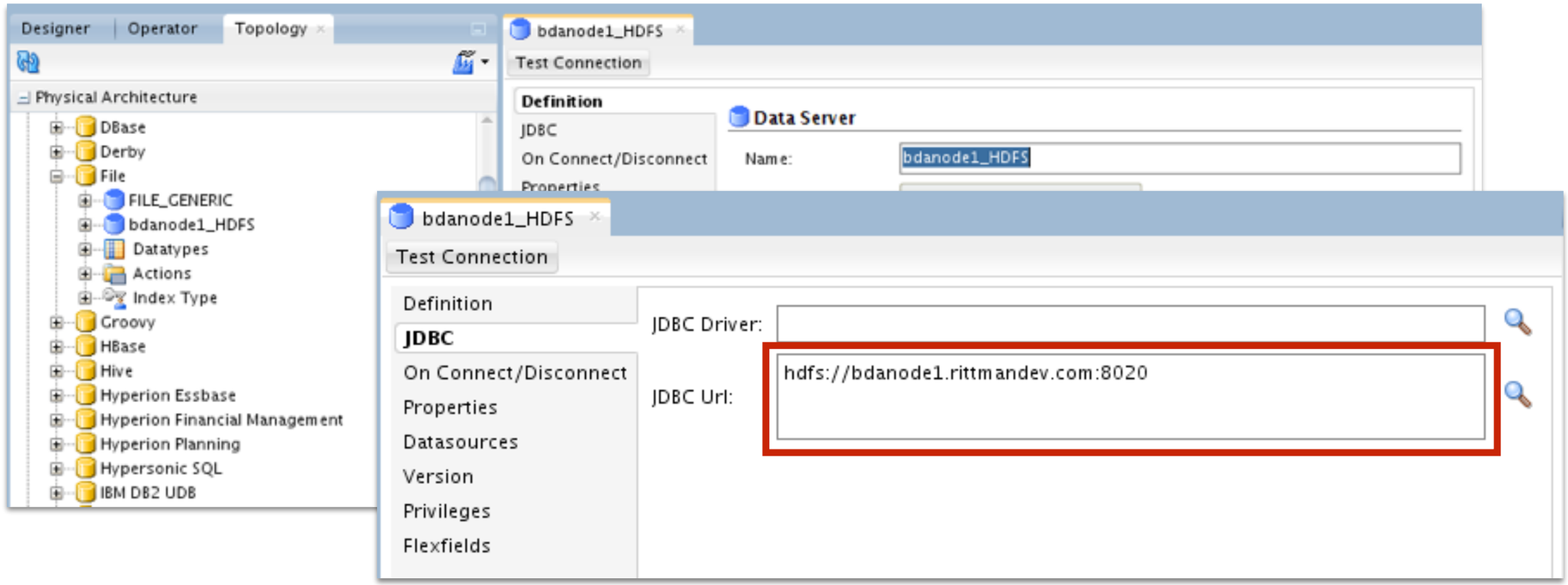


T : +44 (0) 1273 911 268 (UK) **or** (888) 631-1410 (USA) **or**
+61 3 9596 7186 (Australia & New Zealand) **or** +91 997 256 7970 (India)

E : info@rittmanmead.com
W : www.rittmanmead.com

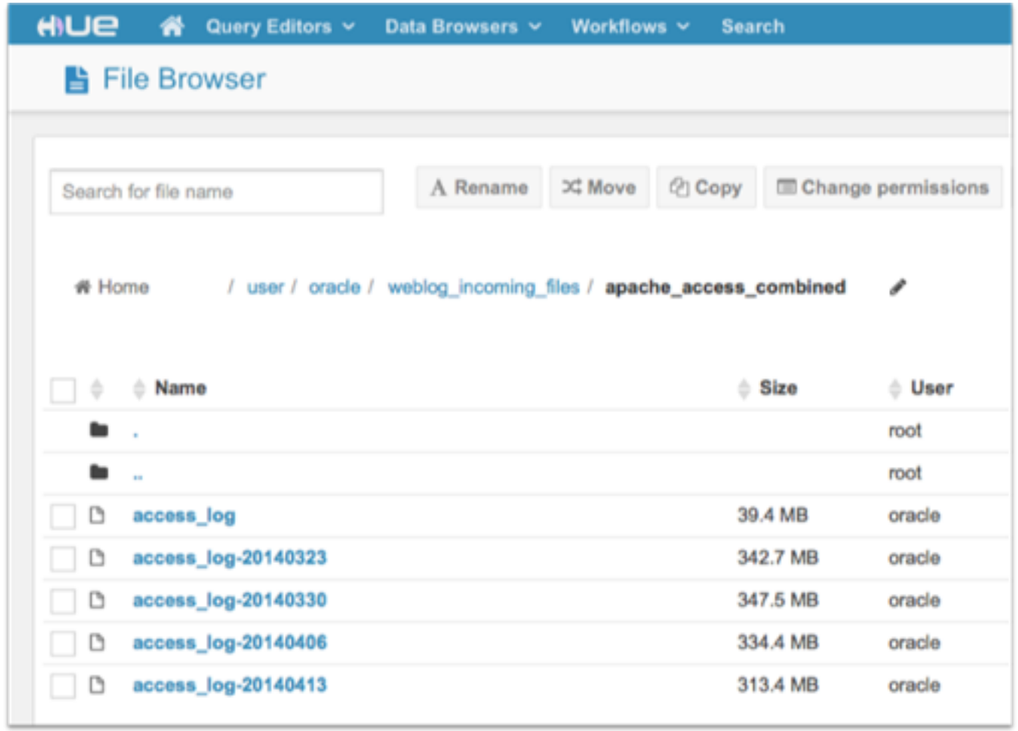
Configuring ODI12c Topology and Models

- HDFS data servers (source) defined using generic File technology
  - Workaround to support IKM Hive Control Append
  - Leave JDBC driver blank, put HDFS URL in JDBC URL field



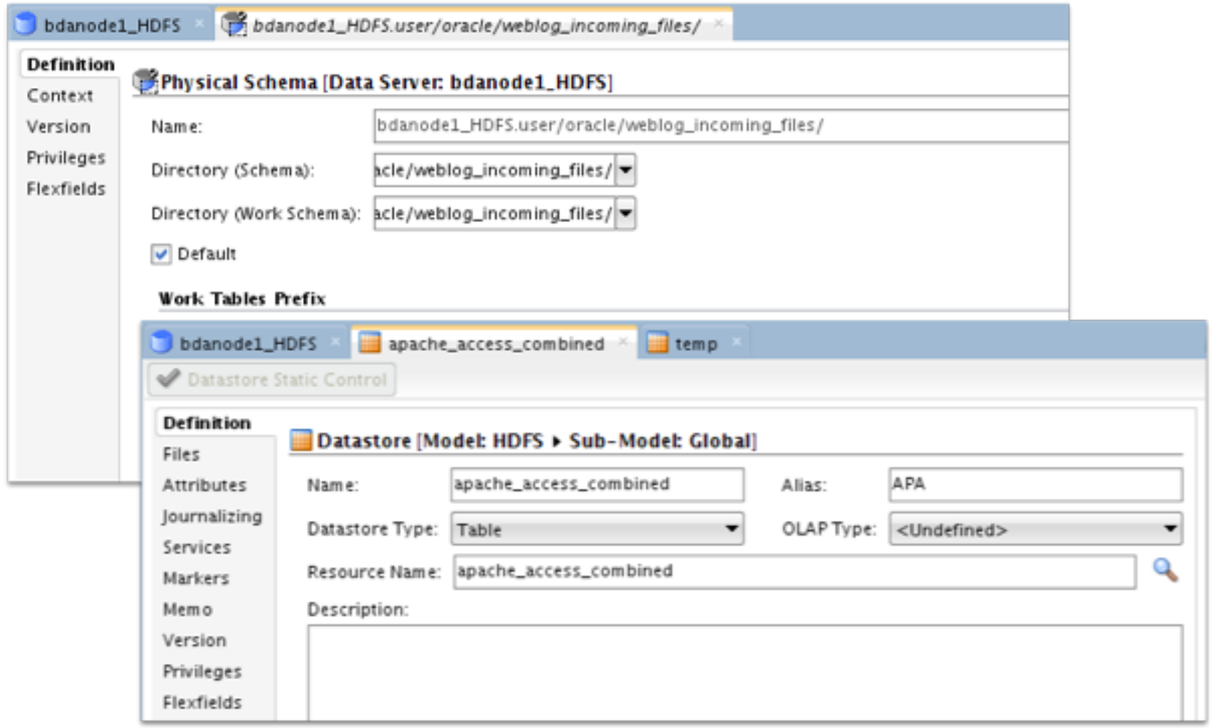
Defining Physical Schema and Model for HDFS Directory

- Hadoop processes typically access a whole directory of files in HDFS, rather than single one
  - Hive, Pig etc aggregate all files in that directory and treat as single file
- ODI Models usually point to a single file though - how do you set up access correctly?



Defining Physical Schema and Model for HDFS Directory

- ODI appends file name to Physical Schema name for Hive access
  - To access a directory, set physical schema to parent directory
  - Set model Resource Name to directory you want to use as source
  - **Note** - *need to manually enter file/ resource names, and "Test" button does not work for HDFS sources*



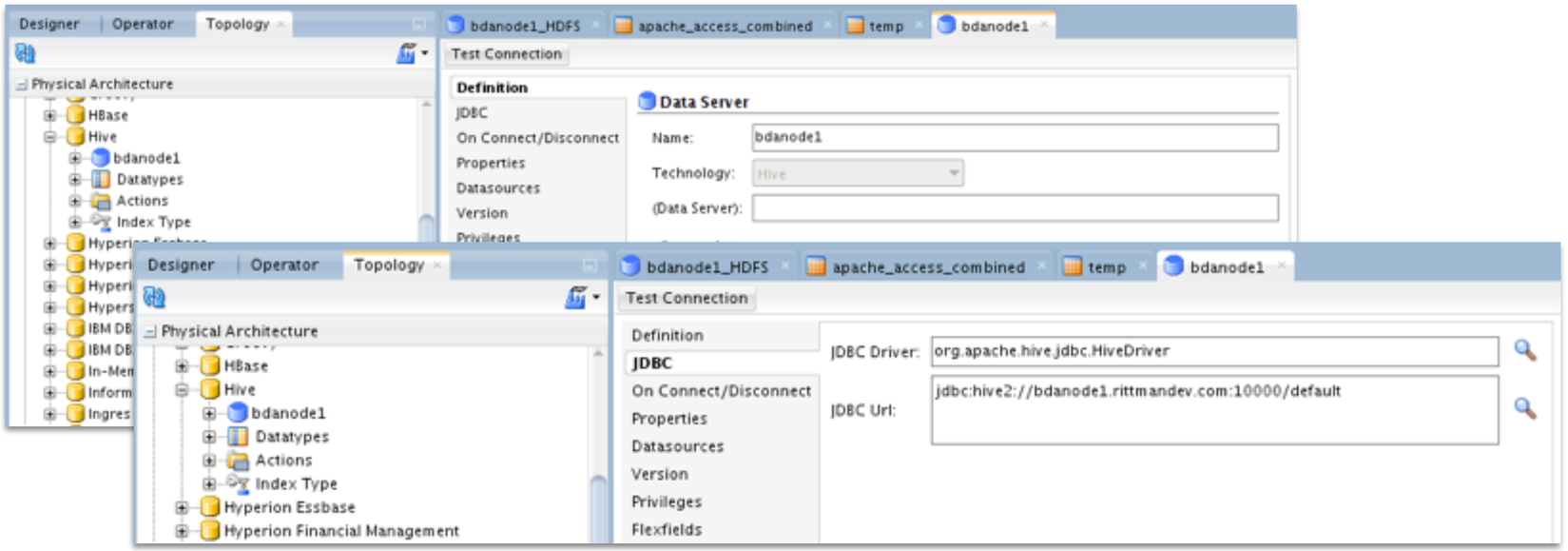
T : +44 (0) 1273 911 268 (UK) **or** (888) 631-1410 (USA) **or**
+61 3 9596 7186 (Australia & New Zealand) **or** +91 997 256 7970 (India)

E : info@rittmanmead.com
W : www.rittmanmead.com

rittmanmead
INTEGRATED ANALYTICS

Defining Topology and Model for Hive Sources

- Hive supported “out-of-the-box” with ODI12c (but requires ODIAAH license for KMs)
- Most recent Hadoop distributions use HiveServer2 rather than HiveServer
  - Need to ensure JDBC drivers support Hive version
  - Use correct JDBC URL format (jdbc:hive2//…)



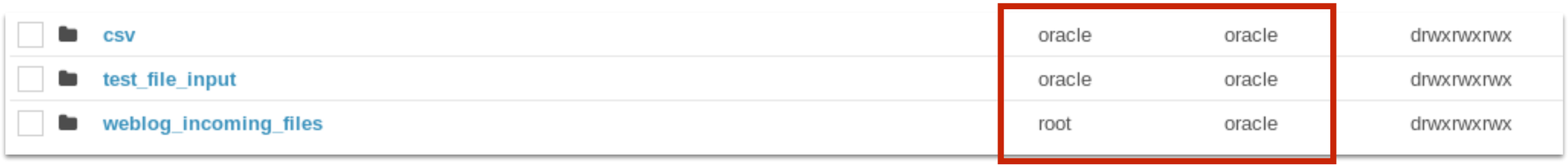
T : +44 (0) 1273 911 268 (UK) **or** (888) 631-1410 (USA) **or**
+61 3 9596 7186 (Australia & New Zealand) **or** +91 997 256 7970 (India)

E : info@rittmanmead.com
W : www.rittmanmead.com

rittmanmead
INTEGRATED ANALYTICS

Hive Tables and Underlying HDFS Storage Permissions

- Hadoop by default has quite loose security
- Files in HDFS organized into directories, using Unix-like permissions
- Hive tables can be created by any user, over directories they have read-access to
  - But that user might not have write permissions on the underlying directory
  - Causes mapping execution failures in ODI if directory read-only
- Therefore ensure you have read/write access to directories used by Hive, and create tables under the HDFS user you'll access files through JDBC
  - Simplest approach - create Hue user for "oracle", create Hive tables under that user

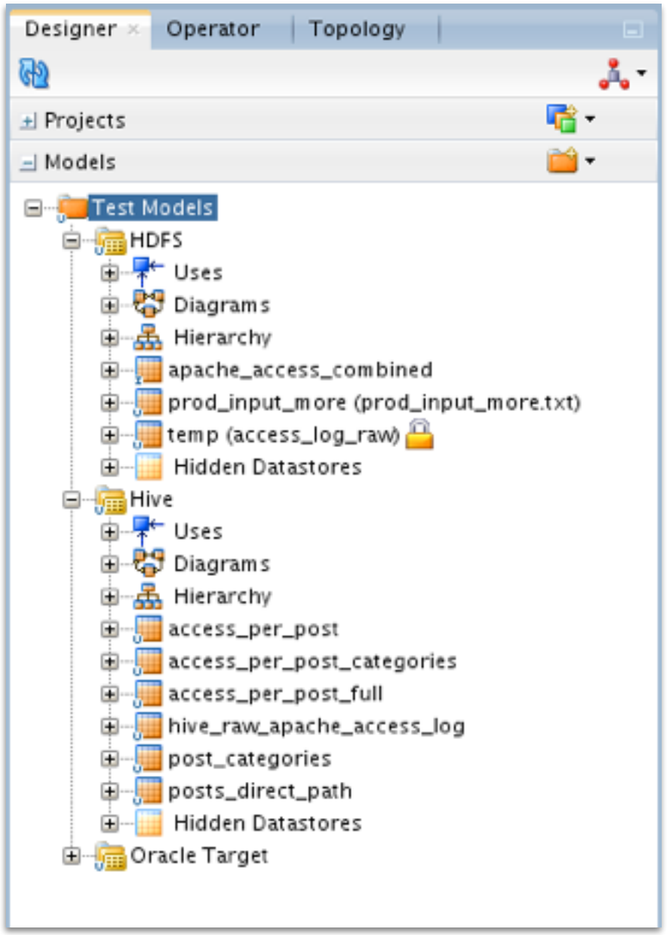


T : +44 (0) 1273 911 268 (UK) **or** (888) 631-1410 (USA) **or**
+61 3 9596 7186 (Australia & New Zealand) **or** +91 997 256 7970 (India)

E : info@rittmanmead.com
W : www.rittmanmead.com

Final Model and Datastore Definitions

- HDFS files for incoming log data, and any other input data
- Hive tables for ETL targets and downstream processing
- Use RKM Hive to reverse-engineer column definition from Hive



T : +44 (0) 1273 911 268 (UK) **or** (888) 631-1410 (USA) **or**
+61 3 9596 7186 (Australia & New Zealand) **or** +91 997 256 7970 (India)

E : info@rittmanmead.com
W : www.rittmanmead.com

rittmanmead
INTEGRATED ANALYTICS

Demo

Viewing the Hive Loading Structures in ODI12c

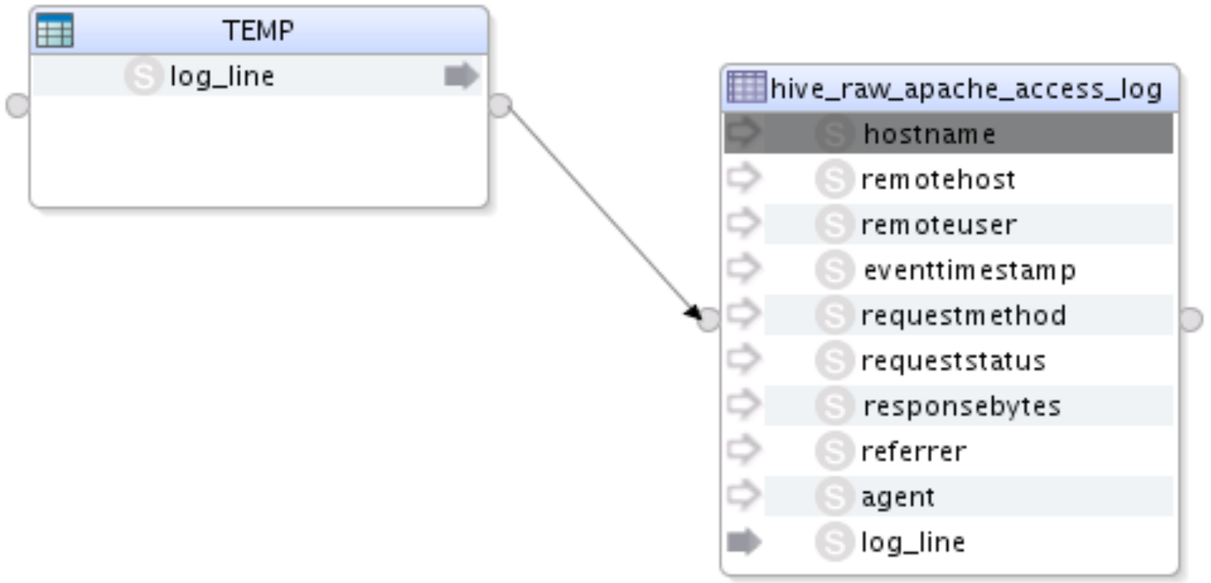
T : +44 (0) 1273 911 268 (UK) **or** (888) 631-1410 (USA) **or**
+61 3 9596 7186 (Australia & New Zealand) **or** +91 997 256 7970 (India)

E : info@rittmanmead.com
W : www.rittmanmead.com

rittmanmead
INTEGRATED ANALYTICS

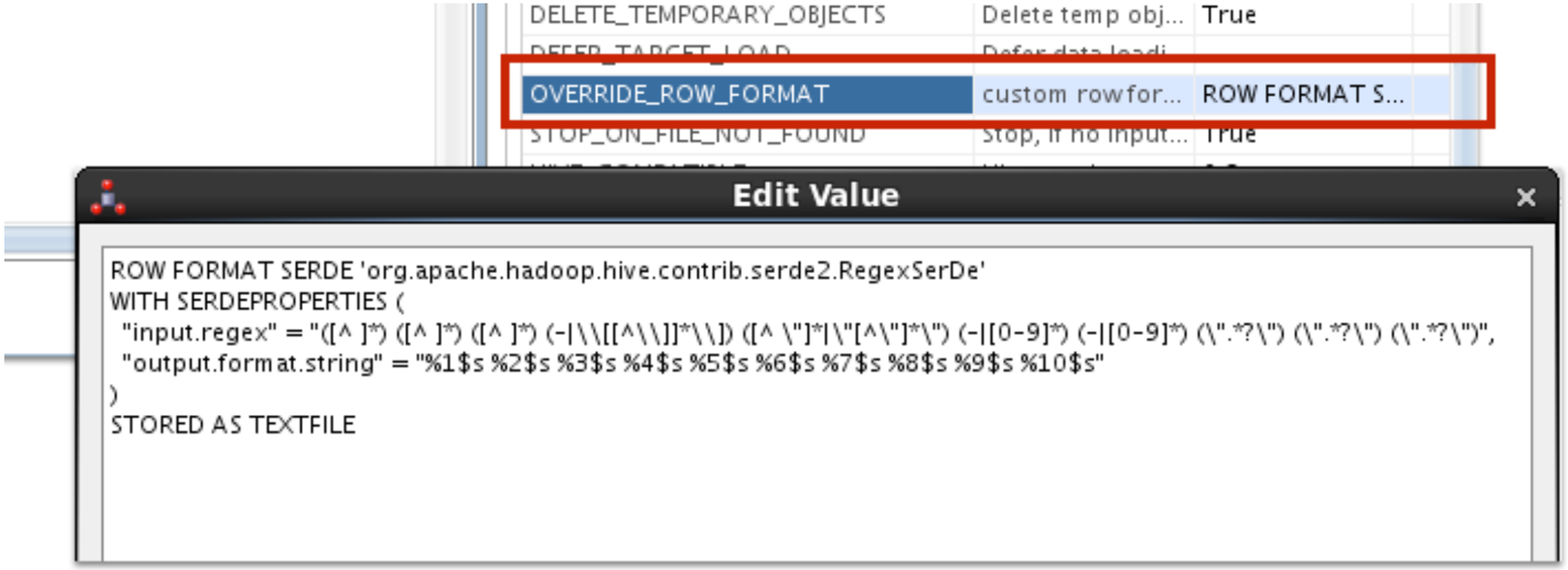
Using IKM File to Hive to Load Web Log File Data into Hive

- Create mapping to load file source (single column for weblog entries) into Hive table
- Target Hive table should have column for incoming log row, and parsed columns



Specifying a SerDe to Parse Incoming Hive Data

- SerDe (Serializer-Deserializer) interfaces give Hive the ability to process new file formats
- Distributed as JAR file, gives Hive ability to parse semi-structured formats
- We can use the RegEx SerDe to parse the Apache CombinedLogFormat file into columns
- Enabled through OVERRIDE_ROW_FORMAT IKM File to Hive (LOAD DATA) KM option



T : +44 (0) 1273 911 268 (UK) **or** (888) 631-1410 (USA) **or**
+61 3 9596 7186 (Australia & New Zealand) **or** +91 997 256 7970 (India)

E : info@rittmanmead.com
W : www.rittmanmead.com

Distributing SerDe JAR Files for Hive across Cluster

- Hive SerDe functionality typically requires additional JARs to be made available to Hive
- Following steps must be performed across ALL BDA nodes:
  - Add JAR reference to HIVE_AUX_JARS_PATH in /usr/lib/hive/conf/hive.env.sh

```
export HIVE_AUX_JARS_PATH=/usr/lib/hive/lib/hive-contrib-0.12.0-cdh5.0.1.jar:$
(echo $HIVE_AUX_JARS_PATH…
```

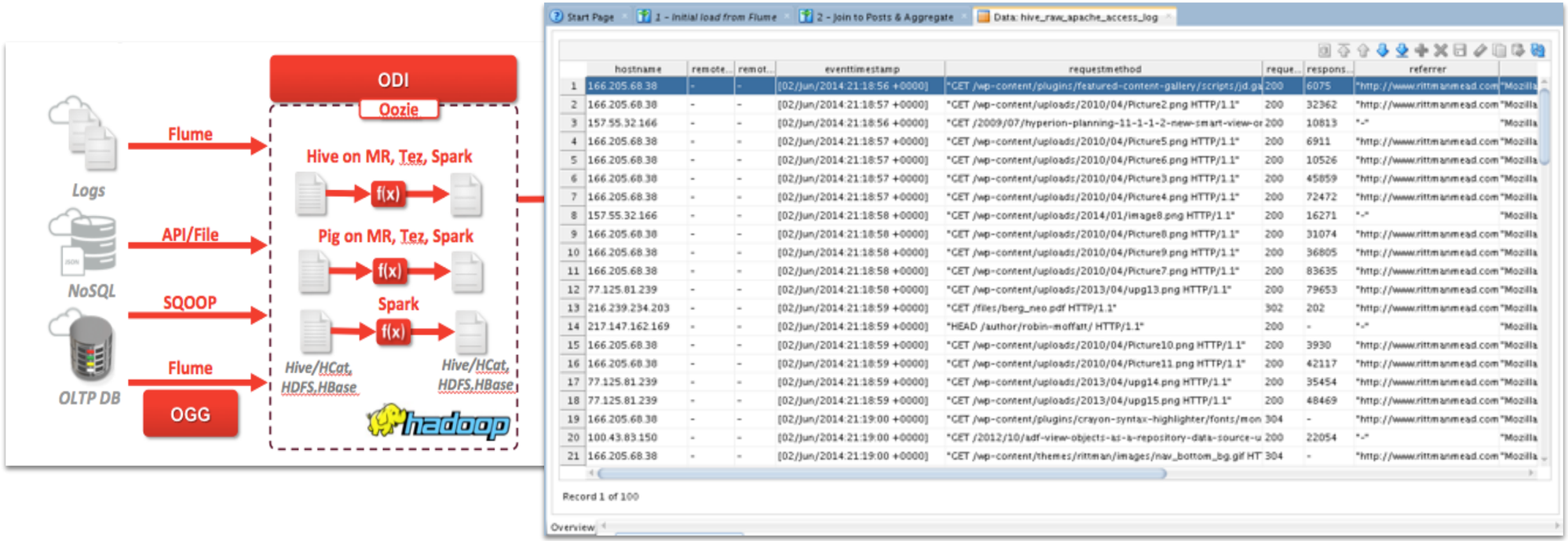
  - Add JAR file to /usr/lib/hadoop

```
[root@bdanode1 hadoop]# ls /usr/lib/hadoop/hive-*

/usr/lib/hadoop/hive-contrib-0.12.0-cdh5.0.1.jar
```

  - Restart YARN / MR1 TaskTrackers across cluster

Executing First ODI12c Mapping

- EXTERNAL_TABLE option chosen in IKM File to Hive (LOAD DATA) as Flume will continue writing to it until source log rotate
- View results of data load in ODI Studio



T : +44 (0) 1273 911 268 (UK) **or** (888) 631-1410 (USA) **or**
+61 3 9596 7186 (Australia & New Zealand) **or** +91 997 256 7970 (India)

E : info@rittmanmead.com
W : www.rittmanmead.com

rittmanmead
INTEGRATED ANALYTICS

Alternative to ODI IKM File to Hive Loading

- You could just define a Hive table as EXTERNAL, pointing to the incoming files
- Add SerDe clause into the table definition, then just read from that table into rest of process

```
CREATE EXTERNAL TABLE apachelog (
  host STRING,
  identity STRING,
  user STRING,
  time STRING,
  request STRING,
  status STRING,
  size STRING,
  referer STRING,
  agent STRING)
ROW FORMAT SERDE 'org.apache.hadoop.hive.contrib.serde2.RegexSerDe'
WITH SERDEPROPERTIES (
  "input.regex" = "([^ ]*) ([^ ]*) ([^ ]*) (-|\\[[^\\]]*\\])([^ \"]*|\"[^\"]*\") (-|[0-9]*) (-|[0-9]*)(?: ([^ \"]*|\"[^\"]*\")
([^ \"]*|\"[^\"]*\"))?",
  "output.format.string" = "%1$s %2$s %3$s %4$s %5$s %6$s %7$s %8$s %9$s"
)
STORED AS TEXTFILE
LOCATION '/user/root/logs';
```

Demo

Viewing the Parsed Log Data in Hive

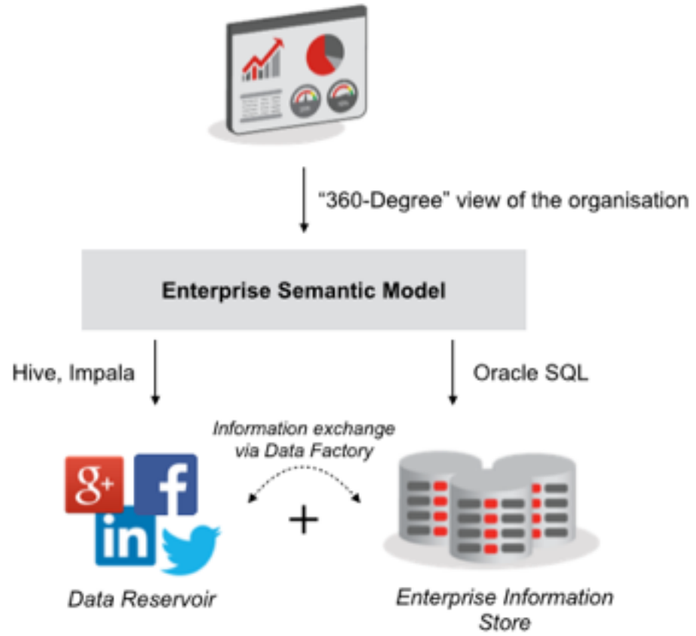
T : +44 (0) 1273 911 268 (UK) **or** (888) 631-1410 (USA) **or**
+61 3 9596 7186 (Australia & New Zealand) **or** +91 997 256 7970 (India)

E : info@rittmanmead.com
W : www.rittmanmead.com

rittmanmead
INTEGRATED ANALYTICS

Adding Social Media Datasources to the Hadoop Dataset

- The log activity from the Rittman Mead website tells us what happened, but not “why”
- Common customer requirement now is to get a “360 degree view” of their activity
  - Understand what’s being said about them
  - External drivers for interest, activity
  - Understand more about customer intent, opinions
- One example is to add details of social media mentions, likes, tweets and retweets etc to the transactional dataset
  - Correlate twitter activity with sales increases, drops
  - Measure impact of social media strategy
  - Gather and include textual, sentiment, contextual data from surveys, media etc



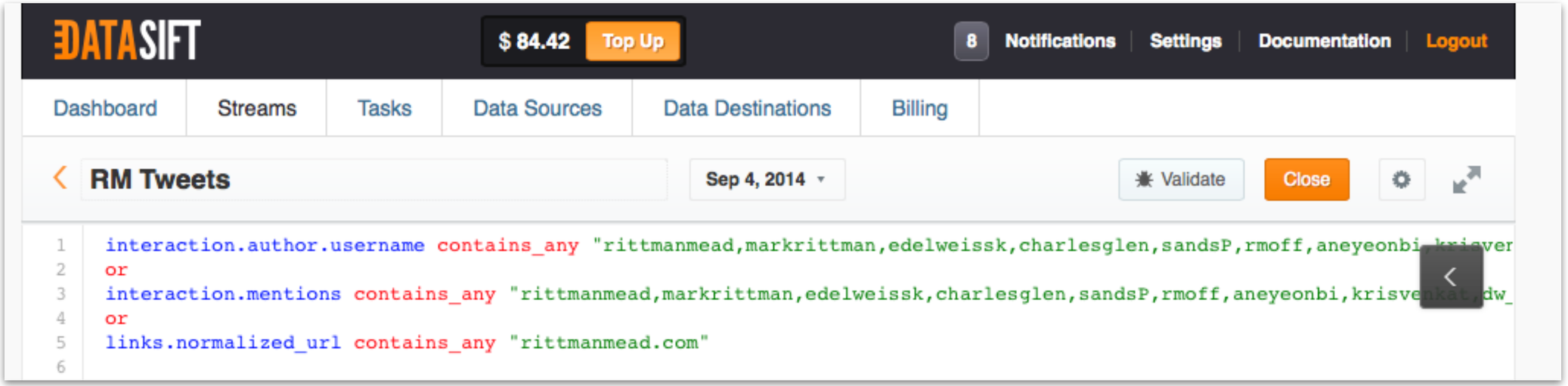
T : +44 (0) 1273 911 268 (UK) **or** (888) 631-1410 (USA) **or**
+61 3 9596 7186 (Australia & New Zealand) **or** +91 997 256 7970 (India)

E : info@rittmanmead.com
W : www.rittmanmead.com

rittmanmead
INTEGRATED ANALYTICS

Example : Supplement Webserver Log Activity with Twitter Data

- Datasift provide access to the Twitter “firehose” along with Facebook data, Tumblr etc
- Developer-friendly APIs and ability to define search terms, keywords etc
- Pull (historical data) or Push (real-time) delivery using many formats / end-points
  - Most commonly-used consumption format is JSON, loaded into Redis, MongoDB etc



T : +44 (0) 1273 911 268 (UK) **or** (888) 631-1410 (USA) **or**
+61 3 9596 7186 (Australia & New Zealand) **or** +91 997 256 7970 (India)

E : info@rittmanmead.com
W : www.rittmanmead.com

rittmanmead
INTEGRATED ANALYTICS

What is MongoDB?

- Open-source document-store NoSQL database
- Flexible data model, each document (record) can have its own JSON schema
- Highly-scalable across multiple nodes (shards)
- MongoDB databases made up of collections of documents
  - Add new attributes to a document just by using it
  - Single table (collection) design, no joins etc
  - Very useful for holding JSON output from web apps
    - for example, twitter data from Datasift

```
{
    "interactionId": "f35ace79b903eedfe5198f386d6fda0c",
    "subscriptionId": "ABC123456a891e3406789123216789",
    "hash": null,
    "hashType": null,
    "interaction": {
        "schema": {
            "version": 3
        },
        "source": "Twitter for Android",
        "type": "twitter",
        "created_at": "Thu, 09 May 2013 08:59:46 +0000",
        "content": "RT @Real_Liam_Payne: Thank you denmarkkkk :) love youuuu :)",
        "id": "f35ace79b903eedf75198f386d6fd40b",
        "author": {
            "hash_id": "c6add0a675830a785598a513d586203c"
        },
        "tags": [
            "type.share",
            "demo.tag",
            "another.one",
            "foobar"
        ]
    }
}
```

basic-interaction-meta.json hosted with ❤ by GitHub view raw

T : +44 (0) 1273 911 268 (UK) **or** (888) 631-1410 (USA) **or**
+61 3 9596 7186 (Australia & New Zealand) **or** +91 997 256 7970 (India)

E : info@rittmanmead.com
W : www.rittmanmead.com

Hive and MongoDB

- MongoDB Hadoop connector provides a storage handler for Hive tables
  - Rather than store its data in HDFS, the Hive table uses MongoDB for storage instead
  - Define in SerDe properties the Collection elements you want to access, using dot notation
  - https://github.com/mongodb/mongo-hadoop

```
{
    "interactionId": "f35ace79b903eedfe5198f386d6fda0c",
    "subscriptionId": "ABC123456a891e3406789123216789",
    "hash": null,
    "hashType": null,
    "interaction": {
        "schema": {
            "version": 3
        },
        "source": "Twitter for Android",
        "type": "twitter",
        "created_at": "Thu, 09 May 2013 08:59:46 +0000",
        "content": "RT @Real_Liam_Payne: Thank you denmarkkkk :) love youuuu :)",
        "id": "f35ace79b903eedf75198f386d6fd40b",
        "author": {
            "hash_id": "c6add0a675830a785598a513d586203c"
        },
        "tags": [
            "type.share",
            "demo.tag",
            "another.one",
            "foobar"
        ]
    }
}
```

basic-interaction-meta.json hosted with ❤ by GitHub view raw

```
CREATE TABLE tweet_data(
  interactionId string,
  username string,
  content string,
  author_followers int)
ROW FORMAT SERDE
  'com.mongodb.hadoop.hive.BSONSerDe'
STORED BY
  'com.mongodb.hadoop.hive.MongoStorageHandler'
WITH SERDEPROPERTIES (
  'mongo.columns.mapping'='{"interactionId":"interactionId",
  "username":"interaction.interaction.author.username",
  "content":\"interaction.interaction.content",
  "author_followers_count":"interaction.twitter.user.followers_count"}'
  )
TBLPROPERTIES (
  'mongo.uri'='mongodb://cdh51-node1:27017/datasiftmongodb.rm_tweets'
  )
```

Demo

MongoDB and the Incoming Twitter Dataset

T : +44 (0) 1273 911 268 (UK) **or** (888) 631-1410 (USA) **or**
+61 3 9596 7186 (Australia & New Zealand) **or** +91 997 256 7970 (India)

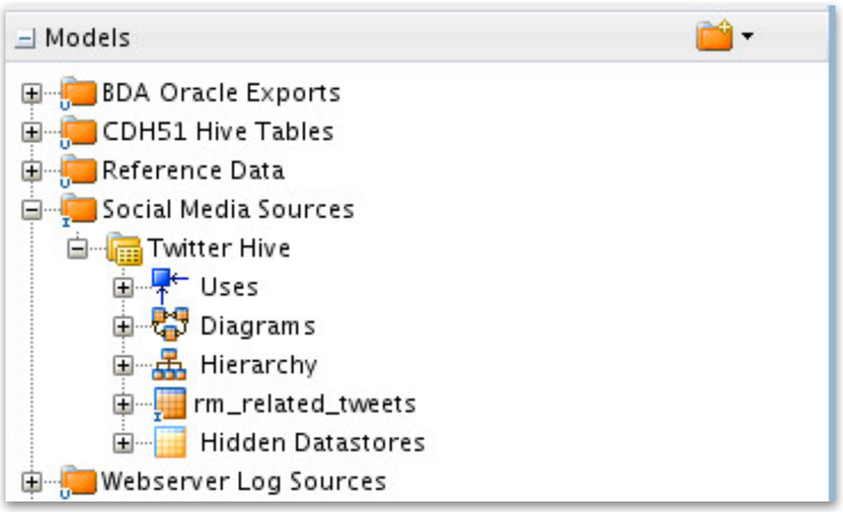
E : info@rittmanmead.com
W : www.rittmanmead.com

rittmanmead
INTEGRATED ANALYTICS

Adding MongoDB Datasets into the ODI Repository

- Define Hive table outside of ODI, using MongoDB storage handler
- Select the document elements of interest, project into Hive columns
- Add Hive source to Topology if needed, then use Hive RKM to bring in column metadata

```
CREATE TABLE tweet_data(
  interactionId string,
  username string,
  content string,
  author_followers int)
ROW FORMAT SERDE
  'com.mongodb.hadoop.hive.BSONSerDe'
STORED BY
  'com.mongodb.hadoop.hive.MongoStorageHandler'
WITH SERDEPROPERTIES (
  'mongo.columns.mapping'='{"interactionId":"interactionId",
  "username":"interaction.interaction.author.username",
  "content":\"interaction.interaction.content",
  "author_followers_count":"interaction.twitter.user.followers_count"}'
  )
TBLPROPERTIES (
  'mongo.uri'='mongodb://cdh51-node1:27017/datasiftmongodb.rm_tweets'
  )
```



T : +44 (0) 1273 911 268 (UK) **or** (888) 631-1410 (USA) **or**
+61 3 9596 7186 (Australia & New Zealand) **or** +91 997 256 7970 (India)

E : info@rittmanmead.com
W : www.rittmanmead.com

rittmanmead
INTEGRATED ANALYTICS

Demo

MongoDB Accessed through Hive

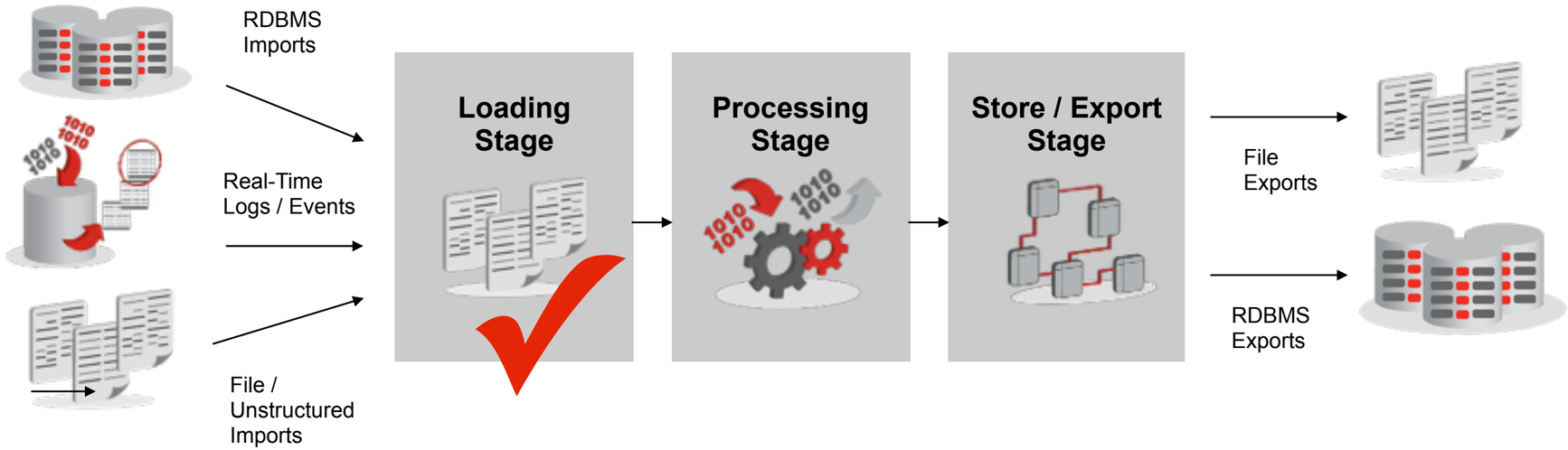
T : +44 (0) 1273 911 268 (UK) **or** (888) 631-1410 (USA) **or**
+61 3 9596 7186 (Australia & New Zealand) **or** +91 997 256 7970 (India)

E : info@rittmanmead.com
W : www.rittmanmead.com

rittmanmead
INTEGRATED ANALYTICS

Summary : Data Loading Phase

- We've now landed log activity from the Rittman Mead website into Hadoop, using Flume
- Data arrives as Apache Webserver log files, is then loaded into a Hive table and parsed
- Supplemented by social media activity (Twitter) accessed through a MongoDB database
- Now we can start processing, analysing, supplementing and working with the dataset…



T : +44 (0) 1273 911 268 (UK) **or** (888) 631-1410 (USA) **or**
+61 3 9596 7186 (Australia & New Zealand) **or** +91 997 256 7970 (India)

E : info@rittmanmead.com
W : www.rittmanmead.com

rittmanmead
INTEGRATED ANALYTICS

Lesson 2 : Hadoop & NoSQL Data Loading using Hadoop Tools and ODI12c

Mark Rittman, CTO, Rittman Mead

Oracle Openworld 2014, San Francisco

T : +44 (0) 1273 911 268 (UK) **or** (888) 631-1410 (USA) **or**
+61 3 9596 7186 (Australia & New Zealand) **or** +91 997 256 7970 (India)

E : info@rittmanmead.com
W : www.rittmanmead.com

rittmanmead
INTEGRATED ANALYTICS