ORACLE

Safe Harbor

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

Statements in this presentation relating to Oracle's future plans, expectations, beliefs, intentions and prospects are "forward-looking statements" and are subject to material risks and uncertainties. A detailed discussion of these factors and other risks that affect our business is contained in Oracle's Securities and Exchange Commission (SEC) filings, including our most recent reports on Form 10-K and Form 10-Q under the heading "Risk Factors." These filings are available on the SEC's website or on Oracle's website at http://www.oracle.com/investor. All information in this presentation is current as of September 2019 and Oracle undertakes no duty to update any statement in light of new information or future events.

Copyright © 2019 Oracle and/or its affiliates.

Oracle OpenWorld 2019

SAN FRANCISCO

THT6128
How to Use an Oracle GoldenGate Initial Load with a Big Data Target

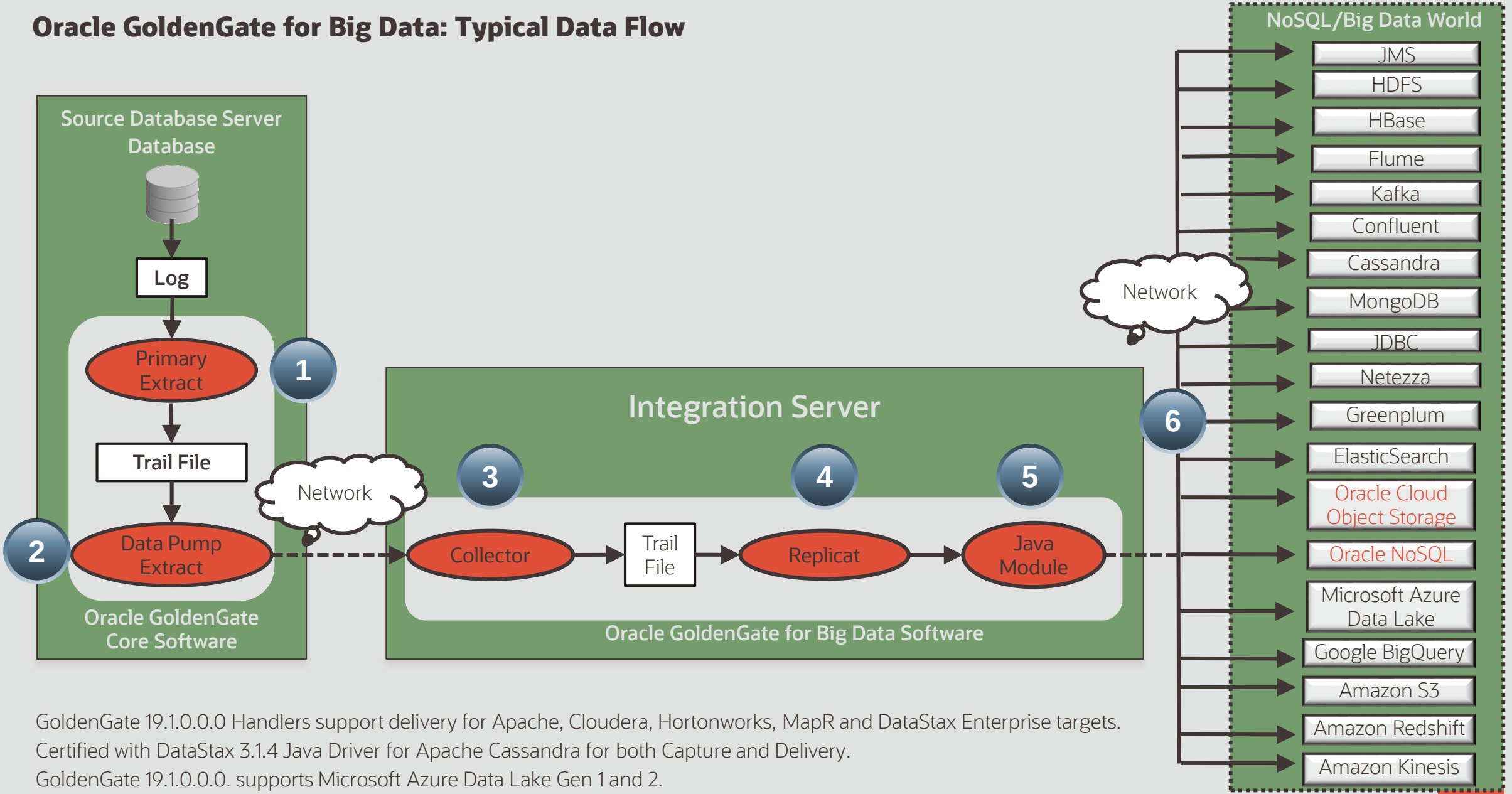
Randall Richeson

Senior Principal Instructor
Oracle University

Our Goal

Show how to use Oracle GoldenGate for an initial load with a Big Data target.

Oracle GoldenGate for Big Data: Typical Data Flow



GoldenGate 19.1.0.0.0 Handlers support delivery for Apache, Cloudera, Hortonworks, MapR and DataStax Enterprise targets.
Certified with DataStax 3.1.4 Java Driver for Apache Cassandra for both Capture and Delivery.
GoldenGate 19.1.0.0.0. supports Microsoft Azure Data Lake Gen 1 and 2.
GoldenGate 19.1.0.0.0 de-supports Flume.

Initial Load: Advantages of Oracle GoldenGate Methods for Big Data Targets

- Work across heterogeneous database types and platforms
- Do not require application down time
- Read directly from source tables without locking tables
- Fetch data in arrays to speed performance
- Use parallel processing with Where clauses or the @Range function
- Distribute data over multiple network controllers

Initial Load: Resource Limitations

How close are your systems?

What are your outage time constraints?

How large are your tables?

How much disk space do you have to store initial load records and changes?

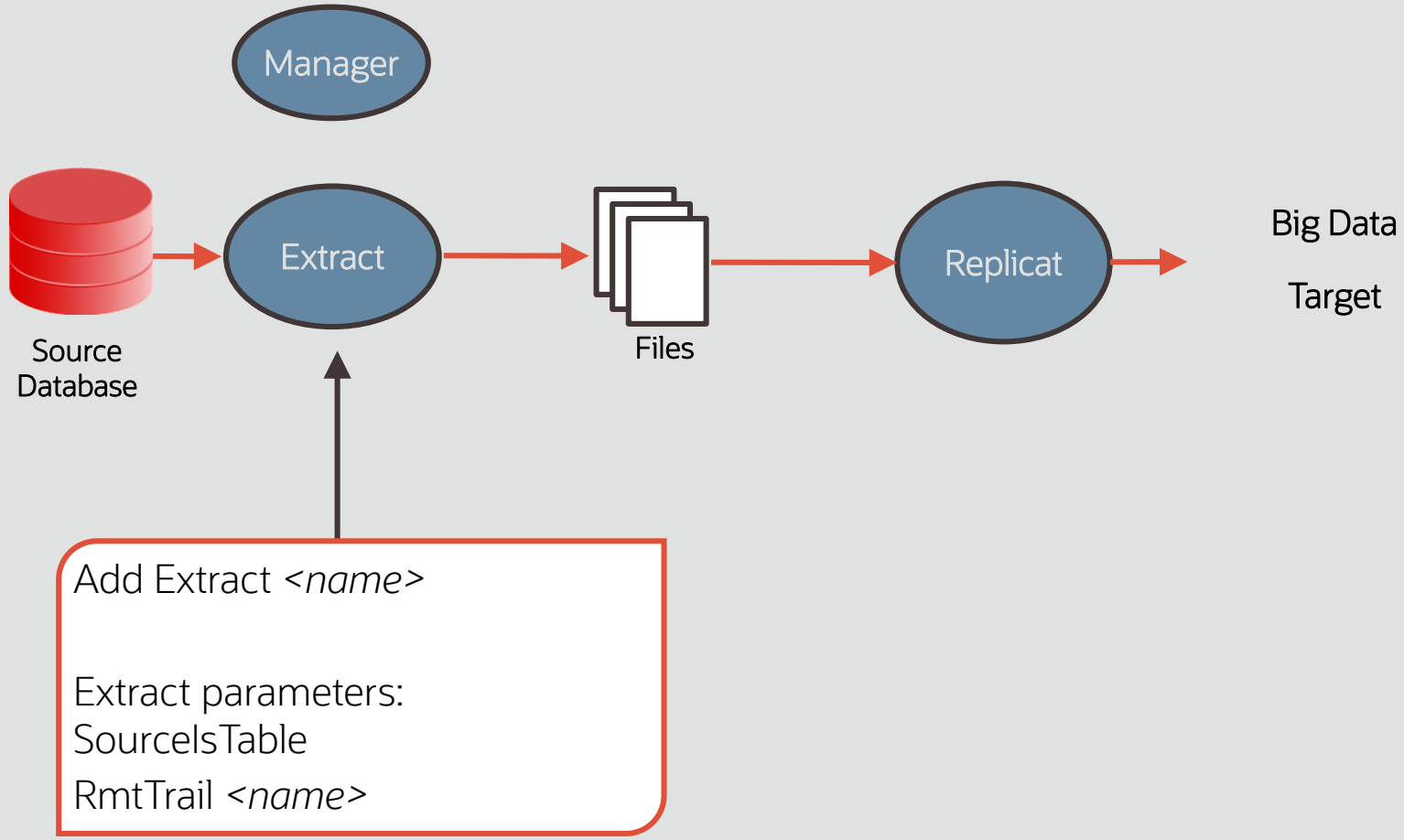
Prerequisites for Initial Load for Big Data Targets

- Disable DDL processing on the source.
- Configure the Manager process.
- Create change-synchronization groups for capture and replication of transactional changes during the initial load.
- Share parameters between process groups.

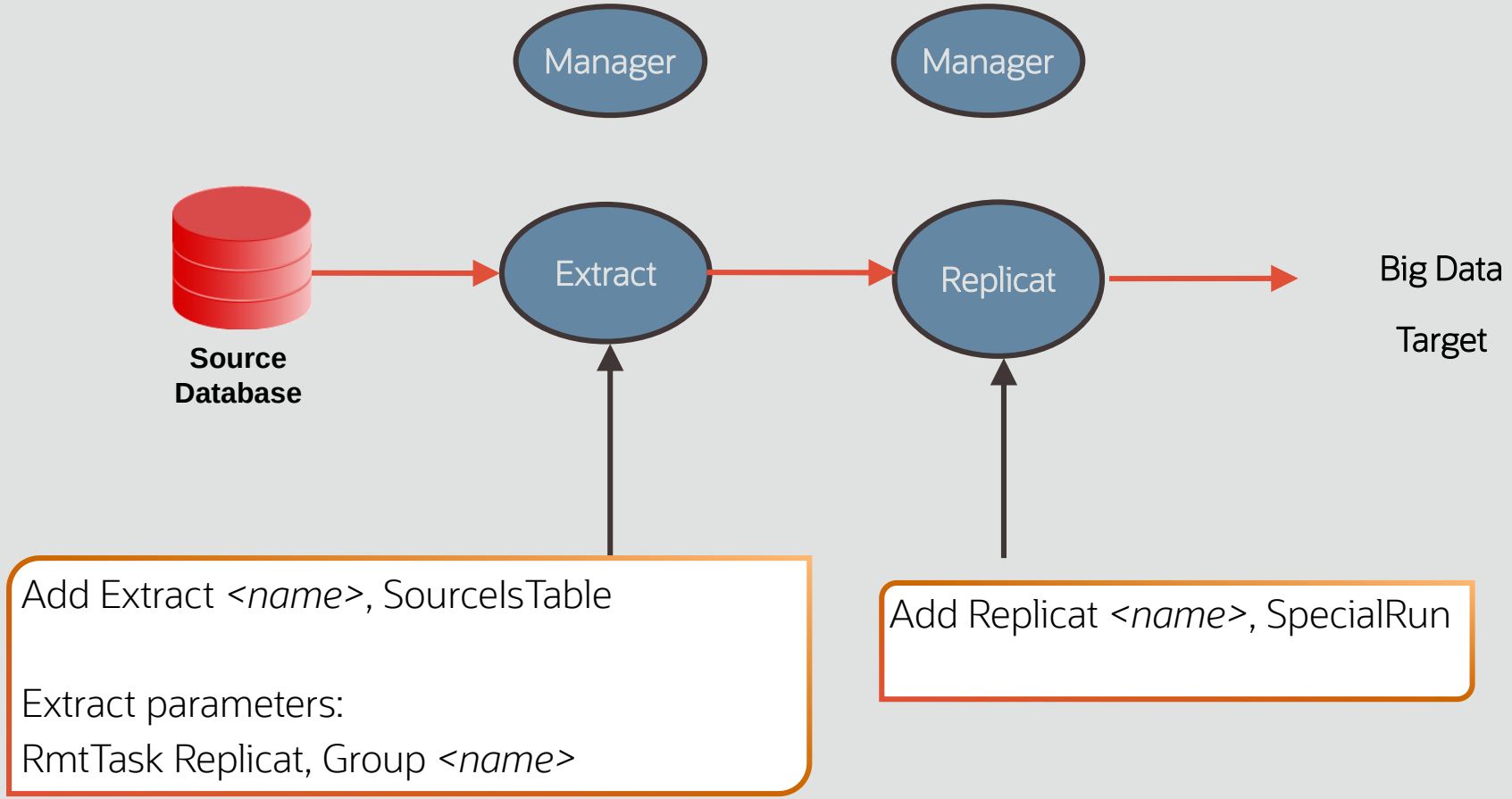
Initial Load: Oracle GoldenGate Methods

GoldenGate Method	Extract writes to	Load Method	Big Data Target
File to Replicat	Trail (canonical format)	Replicat	Yes
File to database utility	Formatted text file	Database utility	No
Direct load	Replicat (directly)	Replicat	Yes
Direct bulk load	Replicat (directly)	Replicat via SQL*Loader API	No

Initial Load: File to Replicat



Initial Load: Direct Load



Demo

Demo: Direct Load Method

Demo: Direct Load Method, Step 1

Create the Java Adapter hdfs.properties file.

```
[oracle@edvmr1p0 oggtrg]$ pwd
/u01/app/oggtrg
[oracle@edvmr1p0 oggtrg]$ cd ./dirprm
[oracle@edvmr1p0 dirprm]$ vi hdfs.properties
```


Demo: Direct Load Method. Step 2

Configure the parameters in the hdfs.properties file.

```
gg.handlerlist=hdfs
gg.handler.hdfs.type=hdfs
gg.handler.hdfs.includeTokens=false
gg.handler.hdfs.maxFileSize=1g
gg.handler.hdfs.pathMappingTemplate=/user/oracle/ogg/${toLowerCase[${fullyQualifiedTableName}]}
gg.handler.hdfs.fileRollInterval=0
gg.handler.hdfs.inactivityRollInterval=0
gg.handler.hdfs.fileNameMappingTemplate=${toLowerCase[${fullyQualifiedTableName}]}_${currentTimestamp}.json
gg.handler.hdfs.partitionByTable=true
gg.handler.hdfs.rollOnMetadataChange=true
gg.handler.hdfs.authType=none
gg.handler.hdfs.format=json_row
gg.handler.hdfs.format.prettyPrint=true
gg.handler.hdfs.format.jsonDelimiter=CDATA[]
gg.handler.hdfs.format.pkUpdateHandling=delete-insert
gg.handler.hdfs.format.generateSchema=false
gg.handler.hdfs.mode=tx
goldengate.userexit.timestamp=utc
goldengate.userexit.writers=javawriter
javawriter.stats.display=TRUE
javawriter.stats.full=TRUE
gg.log=log4j
gg.log.level=INFO
gg.report.time=30sec
gg.classpath=/opt/hadoop/share/hadoop/common/*:/opt/hadoop/share/hadoop/common/lib/*:/opt/hadoop/share/hadoop/hdfs/*:/opt/hadoop/share/hadoop/hdfs/lib/*:/opt/hadoop/etc/hadoop/:
jvm.bootoptions=-Xmx512m -Xms32m -Djava.class.path=ggjava/ggjava.jar
```

Demo: Direct Load Method, Step 3

Review the handler parameters in the hdfs.properties file.

```
gg.handlerlist=hdfs
gg.handler.hdfs.type=hdfs
gg.handler.hdfs.includeTokens=false
gg.handler.hdfs.maxFileSize=1g
gg.handler.hdfs.pathMappingTemplate=/user/oracle/ogg/${toLowerCase[${fullyQualifiedTableName}]}
gg.handler.hdfs.fileRollInterval=0
gg.handler.hdfs.inactivityRollInterval=0
gg.handler.hdfs.fileNameMappingTemplate=${toLowerCase[${fullyQualifiedTableName}]}_${currentTimestamp}.json
gg.handler.hdfs.partitionByTable=true
gg.handler.hdfs.rollOnMetadataChange=true
gg.handler.hdfs.authType=none
gg.handler.hdfs.format=json_row
gg.handler.hdfs.format.prettyPrint=true
gg.handler.hdfs.format.jsonDelimiter=CDATA[]
gg.handler.hdfs.format.pkUpdateHandling=delete-insert
gg.handler.hdfs.format.generateSchema=false
gg.handler.hdfs.mode=tx
```

The folder can be pre-configured or created dynamically by the handler !!!

Demo: Direct Load Method, Step 4

Review the handler parameters in the hdfs.properties file.

```
gg.handlerlist=hdfs
gg.handler.hdfs.type=hdfs
gg.handler.hdfs.includeTokens=false
gg.handler.hdfs.maxFileSize=1g
gg.handler.hdfs.pathMappingTemplate=/user/oracle/ogg/${toLowerCase[${fullyQualifiedTableName}]}
gg.handler.hdfs.fileRollInterval=0
gg.handler.hdfs.inactivityRollInterval=0
gg.handler.hdfs.fileNameMappingTemplate=${toLowerCase[${fullyQualifiedTableName}]}_${currentTimestamp}.json
gg.handler.hdfs.partitionByTable=true
gg.handler.hdfs.rollOnMetadataChange=true
gg.handler.hdfs.authType=none
gg.handler.hdfs.format=json_row
gg.handler.hdfs.format.prettyPrint=true
gg.handler.hdfs.format.jsonDelimiter=CDATA[]
gg.handler.hdfs.format.pkUpdateHandling=delete-insert
gg.handler.hdfs.format.generateSchema=false
gg.handler.hdfs.mode=tx
```

Demo: Direct Load Method, Step 5

Review the handler parameters in the hdfs.properties file.

```
gg.handlerlist=hdfs
gg.handler.hdfs.type=hdfs
gg.handler.hdfs.includeTokens=false
gg.handler.hdfs.maxFileSize=1g
gg.handler.hdfs.pathMappingTemplate=/user/oracle/ogg/${toLowerCase[${fullyQualifiedTableName}]}
gg.handler.hdfs.fileRollInterval=0
gg.handler.hdfs.inactivityRollInterval=0
gg.handler.hdfs.fileNameMappingTemplate=${toLowerCase[${fullyQualifiedTableName}]}_${currentTimestamp}.json
gg.handler.hdfs.partitionByTable=true
gg.handler.hdfs.rollOnMetadataChange=true
gg.handler.hdfs.authType=none
gg.handler.hdfs.format=json_row
gg.handler.hdfs.format.prettyPrint=true
gg.handler.hdfs.format.jsonDelimiter=CDATA[]
gg.handler.hdfs.format.pkUpdateHandling=delete-insert
gg.handler.hdfs.format.generateSchema=false
gg.handler.hdfs.mode=tx
```

Specifies that data from each table will go to its own folder and files.

Demo: Direct Load Method, Step 6

Review the handler parameters in the hdfs.properties file.

```
gg.handlerlist=hdfs
gg.handler.hdfs.type=hdfs
gg.handler.hdfs.includeTokens=false
gg.handler.hdfs.maxFileSize=1g
gg.handler.hdfs.pathMappingTemplate=/user/oracle/ogg/${toLowerCase[${fullyQualifiedTableName}]}
gg.handler.hdfs.fileRollInterval=0
gg.handler.hdfs.inactivityRollInterval=0
gg.handler.hdfs.fileNameMappingTemplate=${toLowerCase[${fullyQualifiedTableName}]}_${currentTimestamp}.json
gg.handler.hdfs.partitionByTable=true
gg.handler.hdfs.rollOnMetadataChange=true
gg.handler.hdfs.authType=none
gg.handler.hdfs.format=json_row
gg.handler.hdfs.format.prettyPrint=true
gg.handler.hdfs.format.jsonDelimiter=CDATA[]
gg.handler.hdfs.format.pkUpdateHandling=delete-insert
gg.handler.hdfs.format.generateSchema=false
gg.handler.hdfs.mode=tx
```

Demo: Direct Load Method, Step 7

Review the gg.classpath in the hdfs.properties file.

```
goldengate.userexit.timestamp=utc
goldengate.userexit.writers=javawriter
javawriter.stats.display=TRUE
javawriter.stats.full=TRUE
gg.log=log4j
gg.log.level=INFO
gg.report.time=30sec
gg.classpath=/opt/hadoop/share/hadoop/common/*:/opt/hadoop/share/hadoop/common/lib/*:/opt/hadoop/share/hadoop/hdfs/*:/opt/hadoop/share/hadoop/hdfs/lib/*:/opt/hadoop/etc/hadoop/:
jvm.bootoptions=-Xmx512m -Xms32m -Djava.class.path=ggjava/ggjava.jar
```

Demo: Direct Load Method, Step 8

Review the jvm.bootoptions in the hdfs.properties file.

```
goldengate.userexit.timestamp=utc
goldengate.userexit.writers=javawriter
javawriter.stats.display=TRUE
javawriter.stats.full=TRUE
gg.log=log4j
gg.log.level=INFO
gg.report.time=30sec
gg.classpath=/opt/hadoop/share/hadoop/common/*:/opt/hadoop/share/hadoop/common/lib/*:/opt/hadoop/share/hadoop/hdfs/*:/opt/hadoop/share/hadoop/hdfs/lib/*:/opt/hadoop/etc/hadoop/:
jvm.bootoptions=-Xmx512m -Xms32m -Djava.class.path=ggjava/ggjava.jar
```

Demo: Direct Load Method, Step 9

Launch ggsci to assess the target environment.

```
[oracle@edvmr1p0 oggtrg]$ pwd
/u01/app/oggtrg
[oracle@edvmr1p0 oggtrg]$ ggsci

Oracle GoldenGate for Big Data
Version 19.1.0.0.0 (Build 007)

Oracle GoldenGate Command Interpreter
Version 19.1.0.0.0 OGGCORE_OGGADP.19.1.0.0.0_PLATFORMS_190618.1416
Linux, x64, 64bit (optimized), Generic on Jun 18 2019 15:37:50
Operating system character set identified as UTF-8.

Copyright (C) 1995, 2019, Oracle and/or its affiliates. All rights reserved.



GGSCI (edvmr1p0) 1> info all

Program     Status      Group       Lag at Chkpt  Time Since Chkpt

MANAGER     RUNNING


GGSCI (edvmr1p0) 2>
```

Demo: Direct Load Method, Step 10

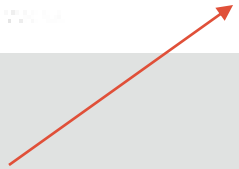
Create parameter file for the special task Replicat on the target.

```
GGSCI (edvmr1p0) 1> edit param rini



GGSCI (edvmr1p0) 2> view param rini

REPLICAT rini
getEnv (JAVA_HOME)
getEnv (LD_LIBRARY_PATH)
getEnv (PATH)
TARGETDB LIBFILE libggjava.so SET property=dirprm/hdfs.properties
MAP oggsrc.product, TARGET oggsrc.product;
```

Demo: Configuring Direct Load Method, Step 11

Create the special task Replicat on the target.

```
GGSCI (edvmr1p0) 3> add replicat rini, specialrun
REPLICAT added.
```



Creates this as a special task !!!

Demo: Configuring Direct Load Method, Step 12

Create the parameter file for the special task Extract on the source.

```
GGSCI (edvmrlp0) 2> view param eini

Extract eini
SETENV (ORACLE_SID='orcl')
SETENV (ORACLE_HOME='/u01/app/oracle/product/12.1.0.2/db_1')
UserIdAlias oggadmin
rmthost localhost, mgrport 7909
rmttask replicat, group rini
Table oggsrc.product;
```

Specifies the co-located or remote target machine.

Specifies the target manager port.

The target manager will start the specified replicat.

Demo: Configuring Direct Load Method, Step 13

Create the special task Extract on the source.

```
GGSCI (edvmr1p0) 4> add extract eini, sourceistable
EXTRACT added.
```

Creates this as a special task !!!

Demo: Configuring Direct Load Method, Step 14

Start the Extract on the source.

```
GGSCI (edvmr1p0) 6> start eini

Sending START request to MANAGER ...
EXTRACT EINI starting


GGSCI (edvmr1p0) 7> info *, tasks

EXTRACT    EINI        Initialized   2019-08-23 22:30   Status STARTING
Checkpoint Lag         Not Available
Process ID             2473
Log Read Checkpoint    Not Available
                       First Record         Record 0
Task                   SOURCEISTABLE
```

The target manager will start the specified replicat !!!

Demo: Configuring Direct Load Method, Step 15

Monitor the Extract on the source until it stops.

```
GGSCI (edvmr1p0) 9> info *, tasks

EXTRACT    EINI        Last Started 2019-08-23 22:36   Status STOPPED
Checkpoint Lag         Not Available
Log Read Checkpoint    Table OGGSRC.PRODUCT
                       2019-08-23 22:36:10  Record 100000
Task                   SOURCEISTABLE
```

It automatically stops when it is finished !!!

Note the record count.

Demo: Configuring Direct Load Method, Step 16

Monitor the Replicat on the target.

```
GGSCI (edvmr1p0) 3> info *, tasks

REPLICAT   RINI      Initialized   2019-08-23 22:21   Status RUNNING
Checkpoint Lag       00:00:00 (updated 00:14:40 ago)
Process ID           2481
Log Read Checkpoint  Not Available
Task                 SPECIALRUN


GGSCI (edvmr1p0) 4> info *, tasks

REPLICAT   RINI      Initialized   2019-08-23 22:21   Status STOPPED
Checkpoint Lag       00:00:00 (updated 00:14:45 ago)
Log Read Checkpoint  Not Available
Task                 SPECIALRUN
```

It automatically stops when it is finished !!!

Demo: Configuring Direct Load Method, Step 17

View the Extract report banner.

```
GGSCI (edvmr1p0) 10> view report eini


2019-08-23 22:35:54  INFO    OGG-01017  Wildcard resolution set to IMMEDIATE because SOURCEISTABLE is used.

***********************************************************************
                 Oracle GoldenGate Capture for Oracle
    Version 12.2.0.1.1 OGGCORE_12.2.0.1.0_PLATFORMS_151211.1401_FBO
   Linux, x64, 64bit (optimized), Oracle 12c on Dec 12 2015 03:12:27

Copyright (C) 1995, 2015, Oracle and/or its affiliates. All rights reserved.


                    Starting at 2019-08-23 22:35:54
***********************************************************************

Operating System Version:
Linux
Version #2 SMP Fri Feb 21 12:07:53 PST 2014, Release 3.8.13-26.2.1.el6uek.x86_64
Node: edvmr1p0
Machine: x86_64
                         soft limit   hard limit
Address Space Size   :    unlimited    unlimited
Heap Size            :    unlimited    unlimited
File Size            :    unlimited    unlimited
CPU Time             :    unlimited    unlimited

Process id: 2473

Description:
```

Demo: Configuring Direct Load Method, Step 18

View the Extract report statistics.

```
***********************************************************************
*                     ** Run Time Statistics **                       *
***********************************************************************


Report at 2019-08-23 22:36:10 (activity since 2019-08-23 22:35:56)

Output to rini:

From Table OGGSRC.PRODUCT:
       #                inserts:     100000
       #                updates:          0
       #                deletes:          0
       #               discards:          0


REDO Log Statistics
  Bytes parsed                     0
  Bytes output              14298929
```

Demo: Configuring Direct Load Method, Step 19

View the Replicat report banner.

```
GGSCI (edvmr1p0) 5> view report rini


***********************************************************************
                 Oracle GoldenGate for Big Data
                 Version 19.1.0.0.0 (Build 007)

                   Oracle GoldenGate Delivery
  Version 19.1.0.0.0 OGGCORE_OGGADP.19.1.0.0.0_PLATFORMS_190618.1416
    Linux, x64, 64bit (optimized), Generic on Jun 18 2019 15:43:03

Copyright (C) 1995, 2019, Oracle and/or its affiliates. All rights reserved.

                    Starting at 2019-08-23 22:35:56
***********************************************************************

Operating System Version:
Linux
Version #2 SMP Fri Feb 21 12:07:53 PST 2014, Release 3.8.13-26.2.1.el6uek.x86_64
Node: edvmr1p0
Machine: x86_64
                         soft limit   hard limit
Address Space Size   :    unlimited    unlimited
Heap Size            :    unlimited    unlimited
File Size            :    unlimited    unlimited
CPU Time             :    unlimited    unlimited

Process id: 2481

Description:
```

Demo: Configuring Direct Load Method, Step 20

View the Replicat statistics.

```
***********************************************************************
*                     ** Run Time Statistics **                       *
***********************************************************************


Report at 2019-08-23 22:36:15 (activity since 2019-08-23 22:36:04)

From Table OGGSRC.PRODUCT to oggsrc.product:
        #                       inserts:     100000
        #                       updates:          0
        #                       deletes:          0
        #                       upserts:          0
        #                     discards:           0
```

Demo: Configuring Direct Load Method, Step 21

Review the ggserr.log on the source.

```
GGSCI (edvmr1p0) 12> sh tail ggserr.log

    anon alloc: mmap(MAP_ANON)  anon free: munmap
    file alloc: mmap(MAP_SHARED)  file free: munmap
    target directories:
    /u01/app/oggsrc/dirtmp.
2019-08-23 22:35:56  INFO    OGG-00993  Oracle GoldenGate Capture for Oracle, eini.prm:  EXTRACT EINI started.
2019-08-23 22:35:58  INFO    OGG-00987  Oracle GoldenGate Command Interpreter for Oracle:  GGSCI command (oracle): info *  tasks.
2019-08-23 22:36:04  INFO    OGG-00987  Oracle GoldenGate Command Interpreter for Oracle:  GGSCI command (oracle): info *  tasks.
2019-08-23 22:36:04  INFO    OGG-02911  Oracle GoldenGate Capture for Oracle, eini.prm:  Processing table OGGSRC.PRODUCT.
2019-08-23 22:36:10  INFO    OGG-00991  Oracle GoldenGate Capture for Oracle, eini.prm:  EXTRACT EINI stopped normally.
2019-08-23 22:36:12  INFO    OGG-00987  Oracle GoldenGate Command Interpreter for Oracle:  GGSCI command (oracle): info *  tasks.
```

Demo: Configuring Direct Load Method, Step 22

Review the ggserr.log on the target.

```
GGSCI (edvmr1p0) 7> sh tail ggserr.log

2019-08-23T22:36:04.804+0000  INFO    OGG-00996  Oracle GoldenGate Delivery:  REPLICAT RINI started.
2019-08-23T22:36:04.840+0000  INFO    OGG-06505  Oracle GoldenGate Delivery:  MAP resolved (entry oggsrc.product): MAP "OGGSRC"."PRODUCT", TARGET oggsrc.prod
uct.
2019-08-23T22:36:04.840+0000  INFO    OGG-02756  Oracle GoldenGate Delivery:  The definition for table OGGSRC.PRODUCT is obtained from the trail file.
2019-08-23T22:36:04.840+0000  INFO    OGG-15056  Oracle GoldenGate Delivery:  The definition for target table oggsrc.product is derived from the source table
 oggsrc.product.
2019-08-23T22:36:04.840+0000  INFO    OGG-06511  Oracle GoldenGate Delivery:  Using following columns in default map by name: PRODUCT_ID, PRODUCT_CODE, PRODU
CT_NAME, SUPPLIER_ID.
2019-08-23T22:36:04.840+0000  INFO    OGG-06510  Oracle GoldenGate Delivery:  Using the following key columns for target table oggsrc.product: PRODUCT_ID.
2019-08-23T22:36:09.033+0000  INFO    OGG-00987  Oracle GoldenGate Command Interpreter:  GGSCI command (oracle): info *  tasks.
2019-08-23T22:36:14.887+0000  INFO    OGG-00987  Oracle GoldenGate Command Interpreter:  GGSCI command (oracle): info *  tasks.
2019-08-23T22:36:15.668+0000  INFO    OGG-00994  Oracle GoldenGate Delivery:  REPLICAT RINI stopped normally.
2019-08-23T22:36:19.803+0000  INFO    OGG-00987  Oracle GoldenGate Command Interpreter:  GGSCI command (oracle): info *  tasks.
```

Demo: Configuring Direct Load Method, Step 23

Verify the HDFS Handler created the target directory and file.

```
drwxr-xr-x   - oracle supergroup          0 2019-08-23 22:36 /user/oracle/ogg
[oracle@edvmr1p0 ~]$ hdfs dfs -ls /user/oracle/ogg
Found 1 items
drwxr-xr-x   - oracle supergroup          0 2019-08-23 22:36 /user/oracle/ogg/oggsrc.product
[oracle@edvmr1p0 ~]$ hdfs dfs -ls /user/oracle/ogg/oggsrc.product
Found 1 items
-rw-r--r--   1 oracle supergroup   32576147 2019-08-23 22:36 /user/oracle/ogg/oggsrc.product/oggsrc.product 2019-08-23 22-36-05.016.json
```

Demo: Configuring Direct Load Method, Step 24

Print the content of the JSON file created by the HDFS handler

```
[oracle@edvmr1p0 ~]$ hdfs dfs -cat /user/oracle/ogg/oggsrc.product/oggsrc.product_2019-08-23_22-36-05.016.json

{
    "table": "OGGSRC.PRODUCT",
    "op_type": "I",
    "op_ts": "2019-08-23 22:36:04.830828",
    "current_ts": "2019-08-23T22:36:05.006000",
    "pos": "-00000000000000000001",
    "PRODUCT_ID": 8482,
    "PRODUCT_CODE": "VJU43ZJJ8JO",
    "PRODUCT_NAME": "nulla. Cras eu tellus eu augue porttitor",
    "SUPPLIER_ID": 8482
}
{
    "table": "OGGSRC.PRODUCT",
    "op_type": "I",
    "op_ts": "2019-08-23 22:36:04.830828",
    "current_ts": "2019-08-23T22:36:06.084000",
    "pos": "-00000000000000000001",
    "PRODUCT_ID": 8483,
    "PRODUCT_CODE": "BDE25JLL5AT",
    "PRODUCT_NAME": "ac,",
    "SUPPLIER_ID": 8483
}
```

There are 100,000 records in the JSON format similar to these.

Where can I learn more from Oracle University ?

- Oracle GoldenGate 12c: Fundamentals for Oracle (4 days)
- Oracle GoldenGate 12c: Advanced Configuration for Oracle (4 days)
- Oracle GoldenGate 12c: Troubleshooting and Tuning (4 days)
- Oracle GoldenGate 12c: Management Pack Overview (2 days)
- Oracle GoldenGate 12c: Veridata Essentials (1 day)
- Oracle GoldenGate 12c: Integrate Big Data (3 days)

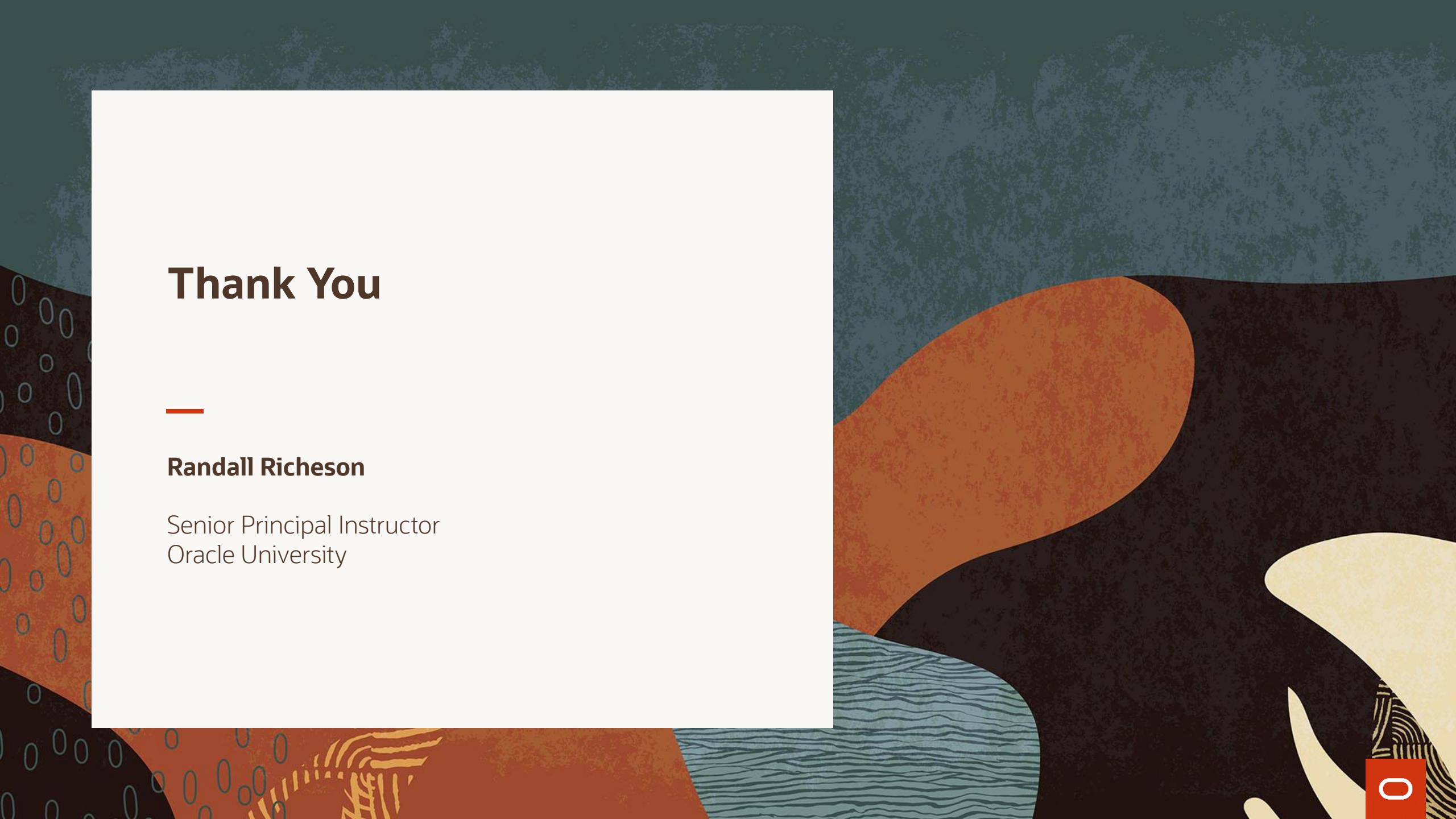
Session Survey

Help us make the content even better. Please complete the session survey in the Mobile App.

Thank You

Randall Richeson

Senior Principal Instructor
Oracle University



Safe Harbor

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

Statements in this presentation relating to Oracle's future plans, expectations, beliefs, intentions and prospects are "forward-looking statements" and are subject to material risks and uncertainties. A detailed discussion of these factors and other risks that affect our business is contained in Oracle's Securities and Exchange Commission (SEC) filings, including our most recent reports on Form 10-K and Form 10-Q under the heading "Risk Factors." These filings are available on the SEC's website or on Oracle's website at http://www.oracle.com/investor. All information in this presentation is current as of September 2019 and Oracle undertakes no duty to update any statement in light of new information or future events.

ORACLE