# Oracle NoSQL Database

## Accessing NoSQL Data by Using External Tables

Prathima Trivedi

ORACLE

ORACLE



[1]Hello, and welcome to this online, self-paced course about Oracle NoSQL Database.

My name is Prathima, and I'll be guiding you through this course, which focuses on accessing NoSQL data from Oracle Database by using external tables.


Using the Player
Foundations of SOA Adoption and Architecture
ORACLE
Server Technologies Curriculum
Ron Pinkerton
Senior Curriculum Developer
Outline
Search
Using the Player
About This Course
Roadmap
Facets of a SOA Service
Concepts, Principles and Guidelines
Service Identification
Course Summary
Oracle Service-Oriented Architecture
Introduction
Explore the features of the course player.
ORACLE
PROPERTIES
Allow user to leave interaction: Anytime
Show 'Next Slide' Button: Don't show
Completion Button Label: Next Slide
Properties...
Edit in Engage

### Introduction

Before we begin, take a look at some of the features of this Flash-based course player. If you've viewed a similar self-paced course in the past, feel free to skip this slide.

### Outline

This is the Outline tab. It's set up to automatically progress through the course in a linear fashion, but you can also review the material in any order. Just click a slide title in the outline to display its contents.

### Transcript

Click the Transcript tab to view the audio transcript for each slide.

### Search

Click the Search tab to find specific information in the course.

### Player Controls

Use these controls to pause, play, or move to the previous or next slide. Use the interactive progress bar to fast forward or rewind the current slide. Some interactive slides in this course may contain additional navigation and controls.

### Change View

Click this button to change the view of this player. Some parts of the course are programmed to open automatically in full-screen mode, but you can control the view at any time by clicking this button.

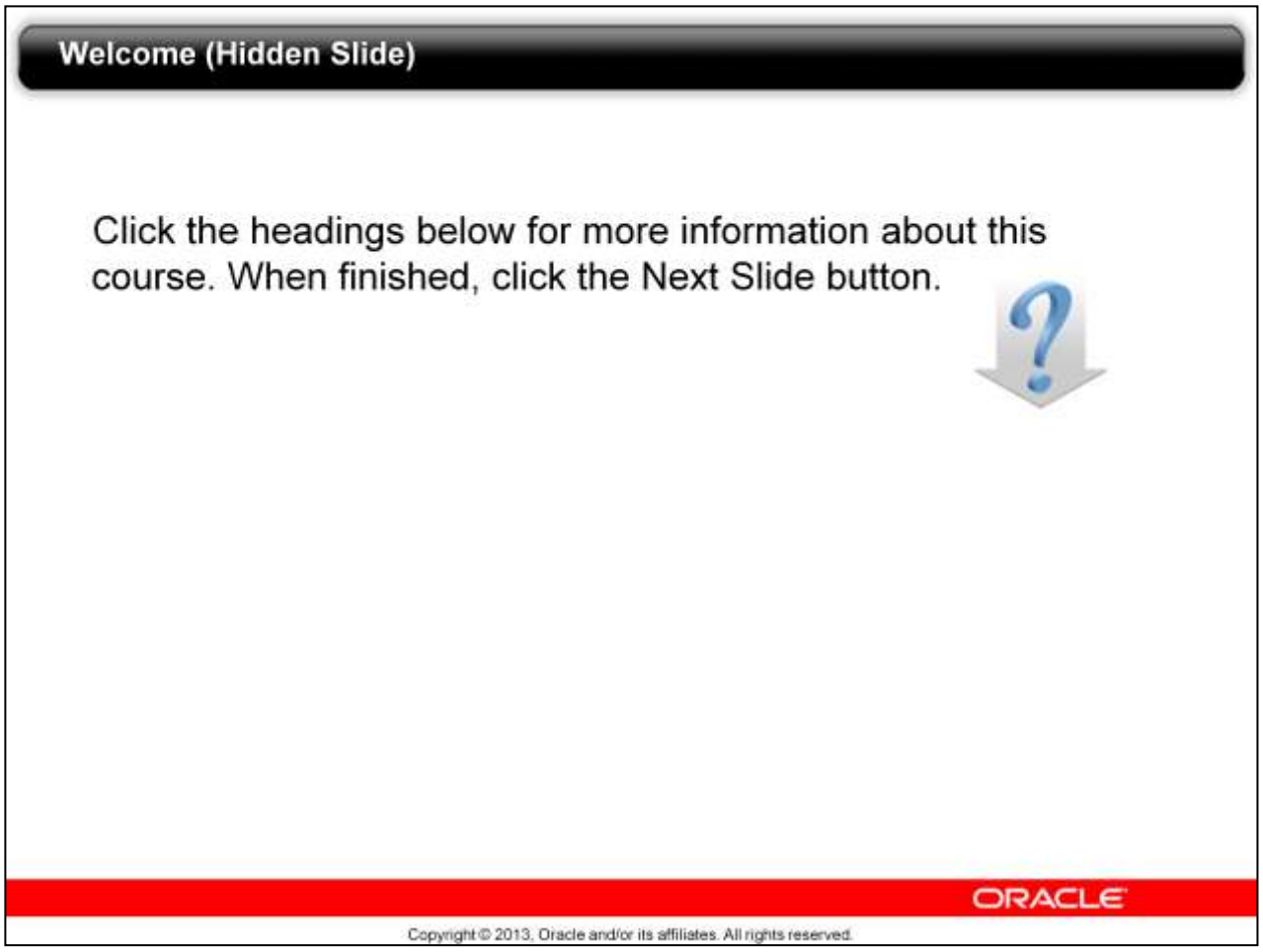
### Bookmark

Click the Bookmark button to save or bookmark your current place in the course.

### Attachments

Click the Attachments button to access any attachments associated with this course.


So, you know the title of the course, but you may be asking yourself, “Is this the right course for me?” Click the bars to learn about the course objectives, target audience, and prerequisites.




So you know the title of the course, but you may be asking yourself, “Am I in the right place?” To help you answer this question, you can access information here regarding the course objectives, the target audience, and the prerequisites. When finished, click the Next Slide button.

## What skills will I learn? (Hidden Slide)

At the end of this course, you should be able to:

- Start Oracle NoSQL Database
- Access data from the formatter classes
- Configure external tables
- Create and publish the configuration file
- Use SQL commands to retrieve NoSQL data

ORACLE



What can you expect to get out of this course?

Here are the core learning objectives.

## Who is the target audience? (Hidden Slide)

This course is meant for:

- Oracle NoSQL Database Developers
- Oracle Database Developers

ORACLE



Who is this course for? Here is the intended audience.

## What are the prerequisites? (Hidden Slide)

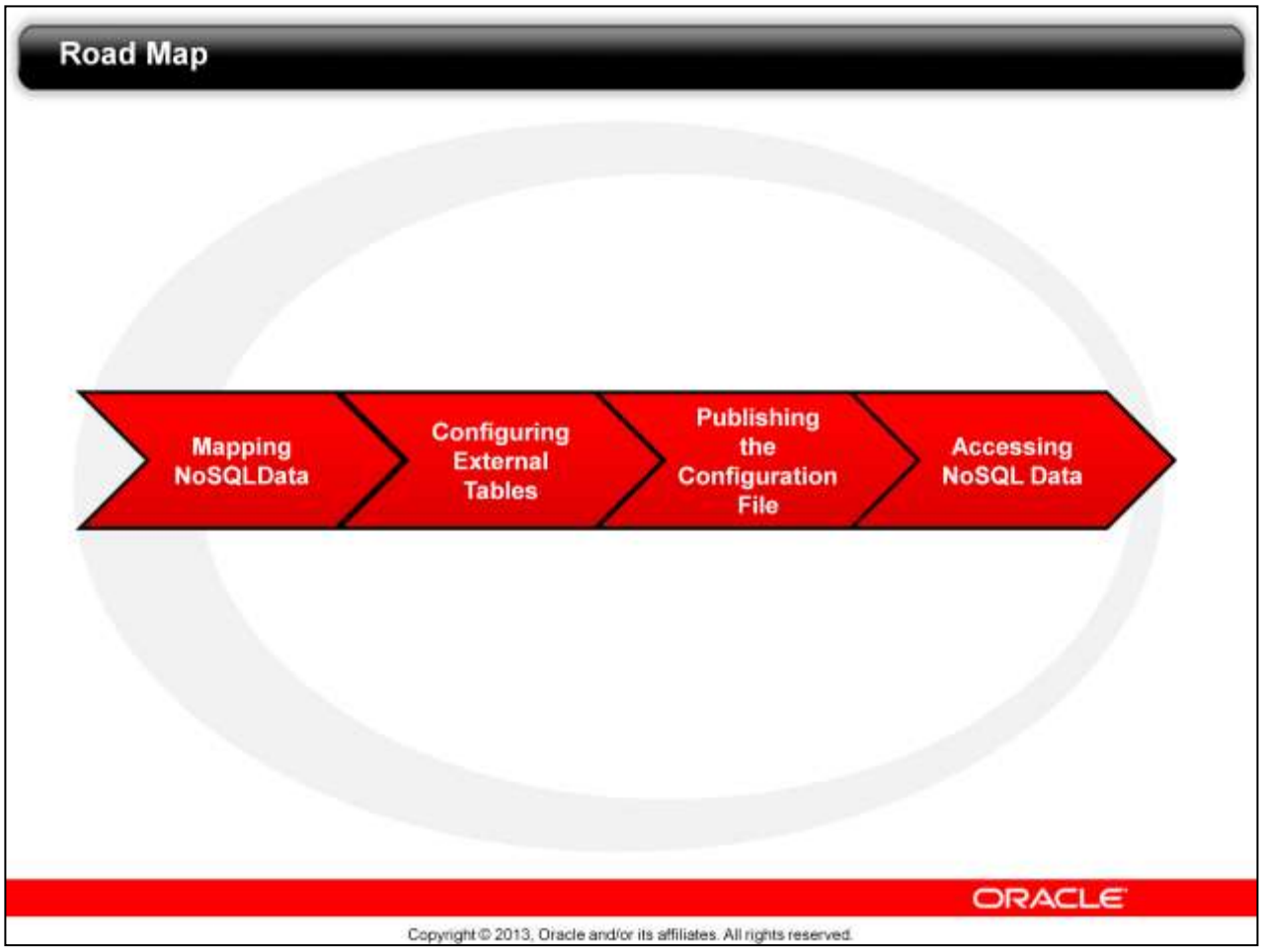
Prerequisite skills include:

- Basic understanding of Oracle NoSQL Database
- Familiarity with Oracle Database external table concepts
- Ability to compile and execute a Java program

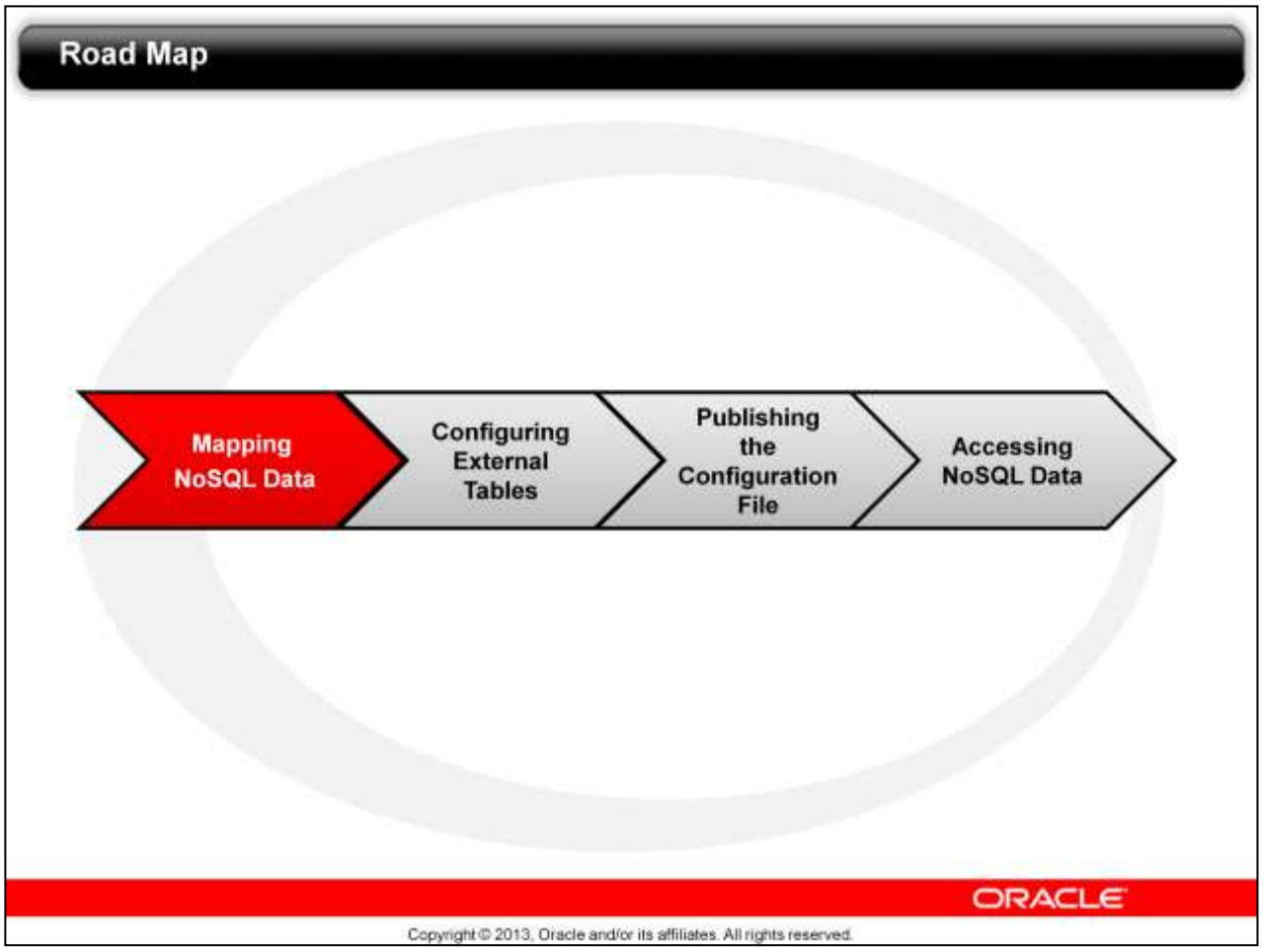
ORACLE



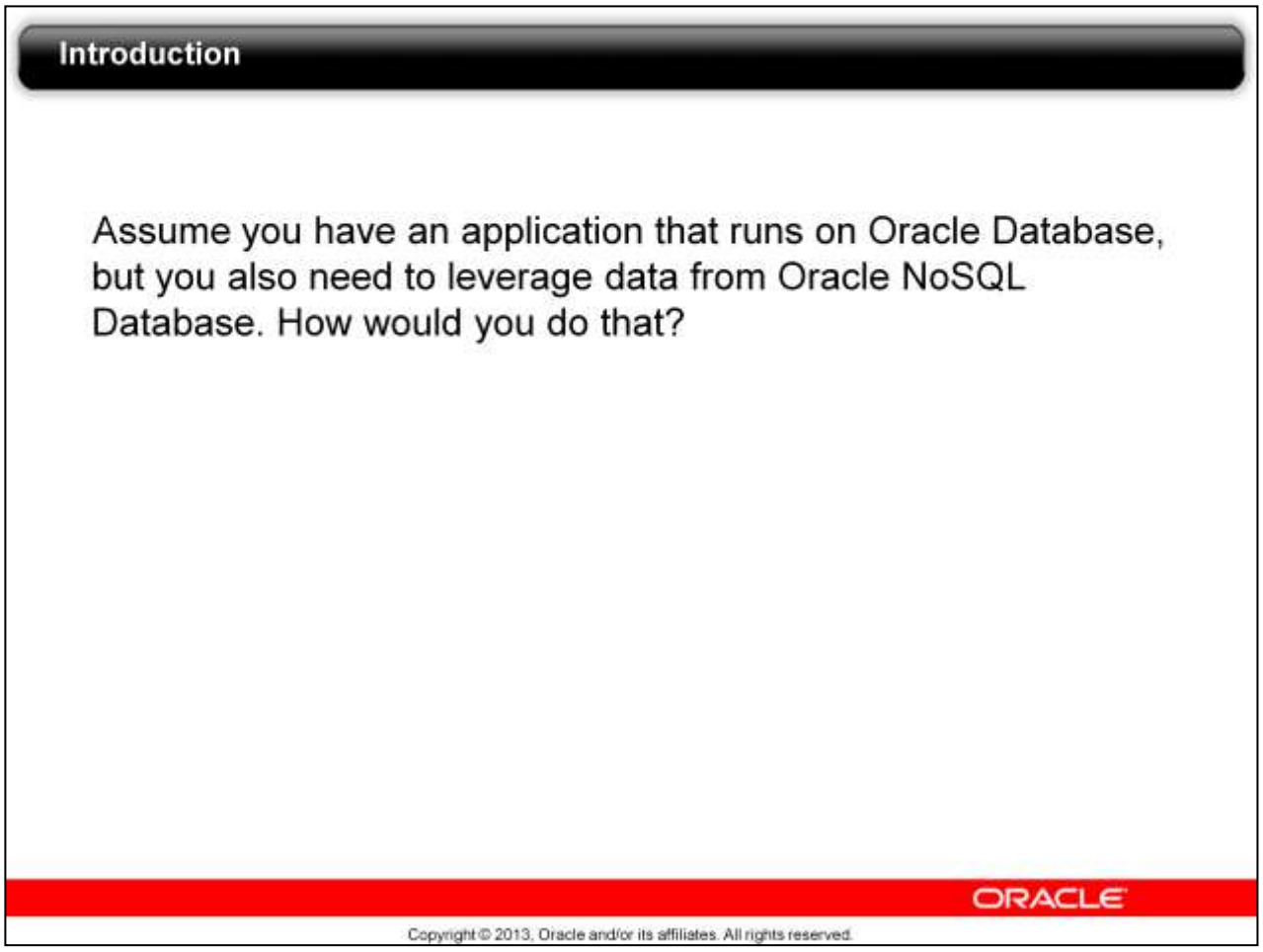
Before taking this course, you should have a basic understanding of Oracle NoSQL Database and Oracle Database. You should also know how to compile and execute a Java program.




In this course, I will talk about [1] Accessing data from formatter classes, and configuring the preprocessor, [2]exposing NoSQL data as external tables, [3]publishing the configuration file, and finally [4] fetching data by using SQL commands.




Let's begin with[1] mapping NoSQL data to Oracle Database data. In this topic, you'll learn about Oracle NoSQL Database support for external tables, including formatter classes and the preprocessor.



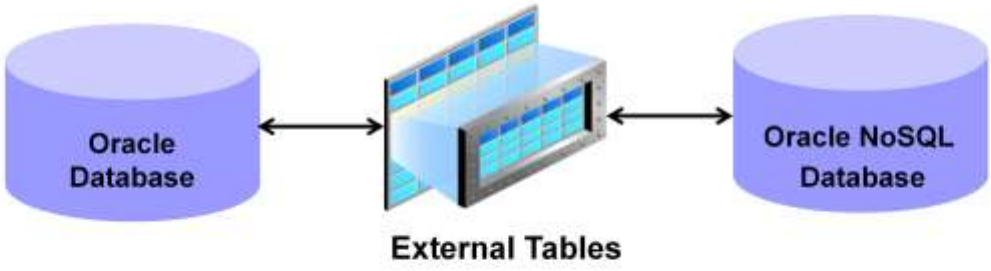

In this scenario, you must integrate Oracle NoSQL Database with Oracle Database. Oracle is the only SQL vendor that has an enterprise class NoSQL database that supports functionalities such as:

- Querying NoSQL data from Oracle Database
- Accessing NoSQL data from Apache Hadoop
- Sharing data with Oracle Coherence
- Processing persist history and event streams with Oracle Event Processing
- Storing and querying Resource Description Framework (RDF) data

In the next slides, I will explain and demonstrate the steps involved to solve the problem in the scenario.

## External Table Feature

You use external tables to connect Oracle Database and Oracle NoSQL Database.

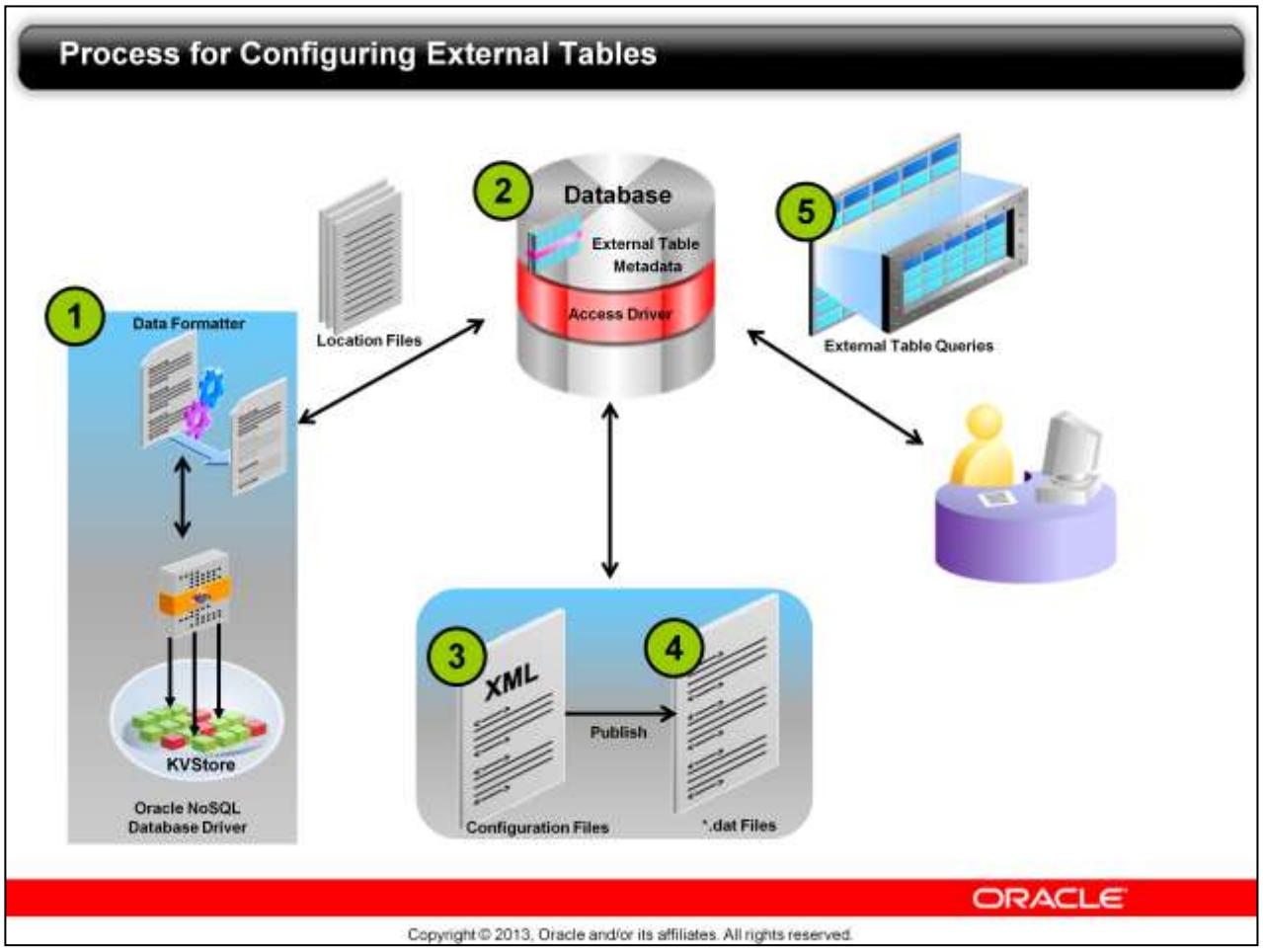



ORACLE



With the Oracle Database External Table feature, you can create a table that reads data from an external data source. Using this capability, you can read NoSQL data into Oracle Database by using SQL queries. These external sources are usually one or more flat text files. However, in this case, you will use Oracle NoSQL Database as the external data source.

Oracle NoSQL Database comes with the `oracle.kv.exttab` package. This package contains the public API and utilities for accessing NoSQL data through Oracle Database external tables.

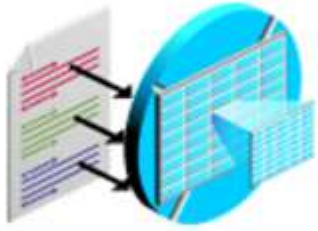



The figure shows the steps involved in configuring external tables. You configure connection details about the NoSQL database and Oracle database in the location files. These location files also map a formatter class to a specified external table on the relational database side. You also set a NoSQL key prefix in the configuration file that is used to fetch records with keys matching the prefix. These records are then processed in the formatter class, which transforms the data and later sends the information to the relational database side for filtering, ordering, and grouping based on the query.

I will talk more about each step in the following slides.

## Data Mapping

- Formatter is an interface for organizing the user-supplied class to transform the NoSQL record into the SQL Loader format.
- Location files contain connectivity and key range parameters for Oracle NoSQL Database.

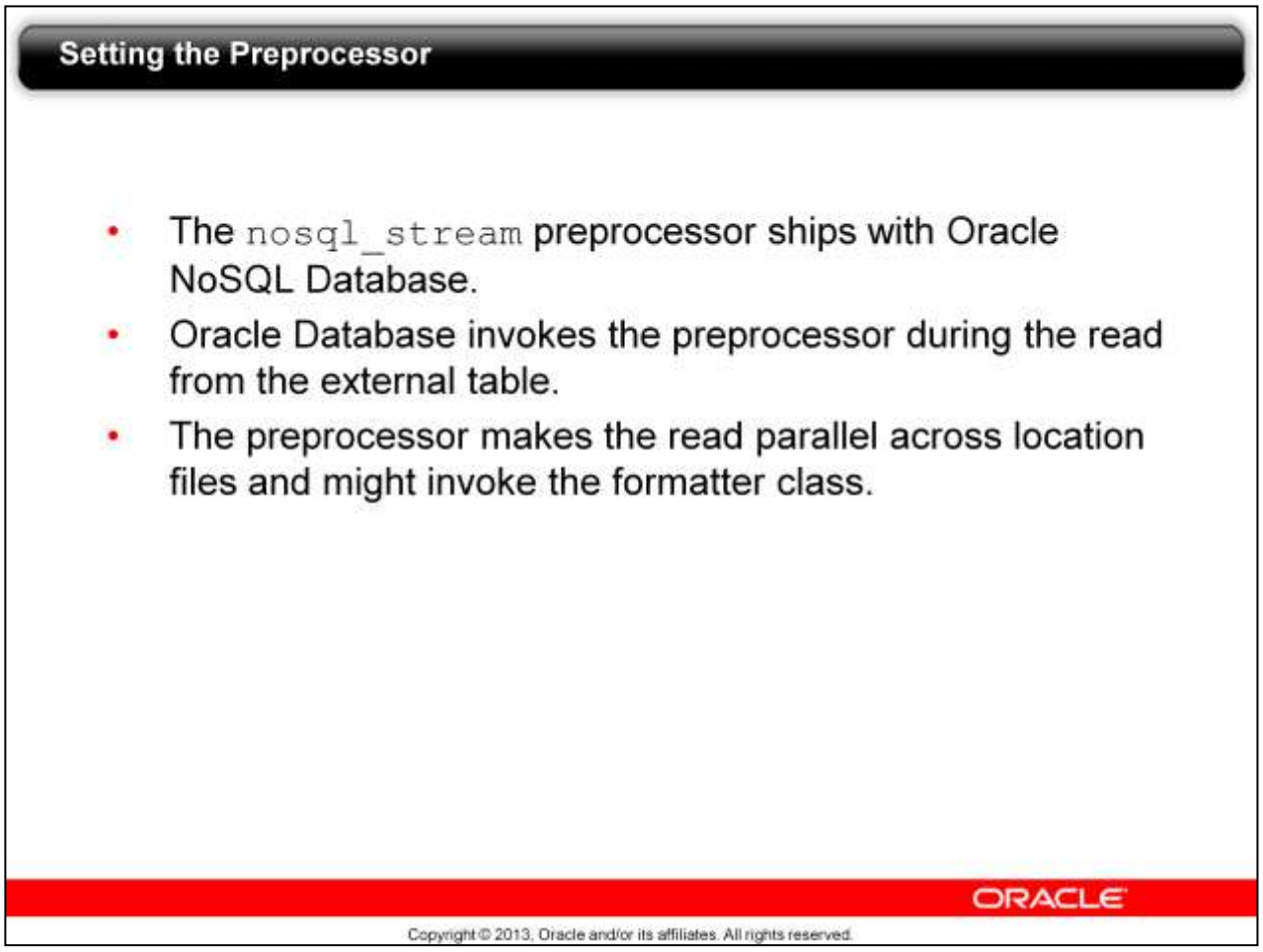


ORACLE



First, you map the NoSQL data to the Oracle Database data. The `oracle.kv.exttab` package contains [1]the Formatter interface, which allows your class to accept a record and return a string. You can map the string to the external table fields. This method is a simple select-and-filter mechanism.

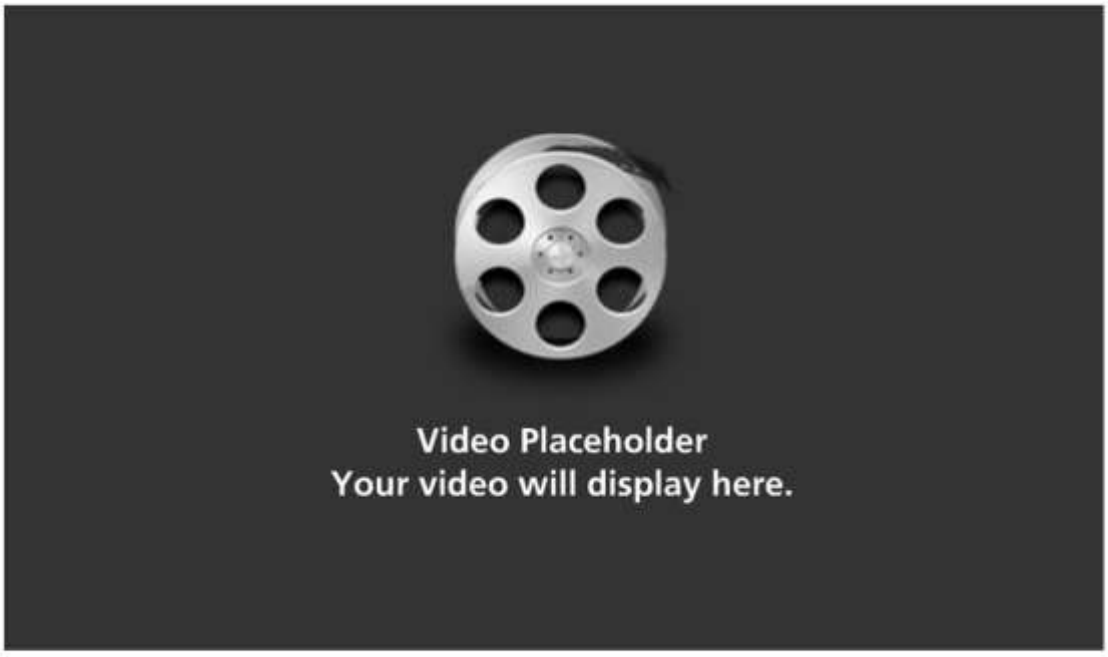
[2]After you create the formatter class, the NoSQL data is ready to be loaded to the external table. The data is backed up in a file that contains the access information. This backup file is called a location file. A location file does not contain data. It contains query restrictions, formatting, and configuration information about connecting to Oracle Database and Oracle NoSQL Database.




A preprocessor is a program that the Oracle Database server invokes when a SQL statement is executed. The `nosql_stream` script is the preprocessor program that ships with Oracle NoSQL Database. This script contains the system settings, such as `PATH`, `KVHOME`, and `CLASSPATH`, that are required for executing the formatter class. This preprocessor is later registered to the external table. So, when you run a SQL query on an external table, it looks for the preprocessor script that sets the classpath, and then the .dat file is used to run the appropriate formatter class. This file is editable and you need to make sure it has execute permissions. The preprocessor is invoked once for the External Table location file, and the number of location files determines the level of parallelism when retrieving records.

Because you use Java Database Connectivity (JDBC) to connect to Oracle Database, you should install `ojdbc6.jar`. If you are using Oracle Wallet Manager as an external password store, make sure that `oraclepki.jar` is installed.

## Demonstration

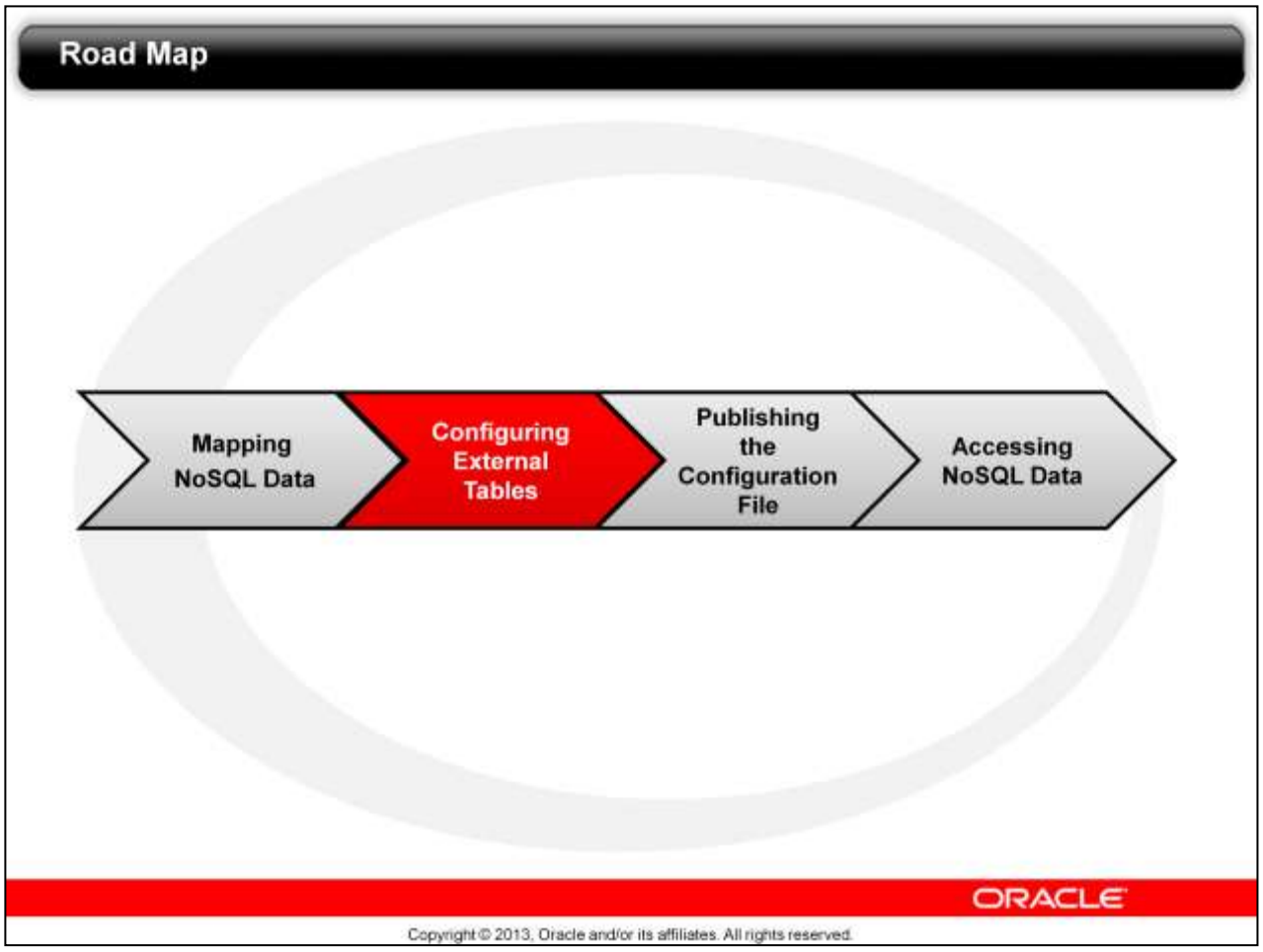



ORACLE

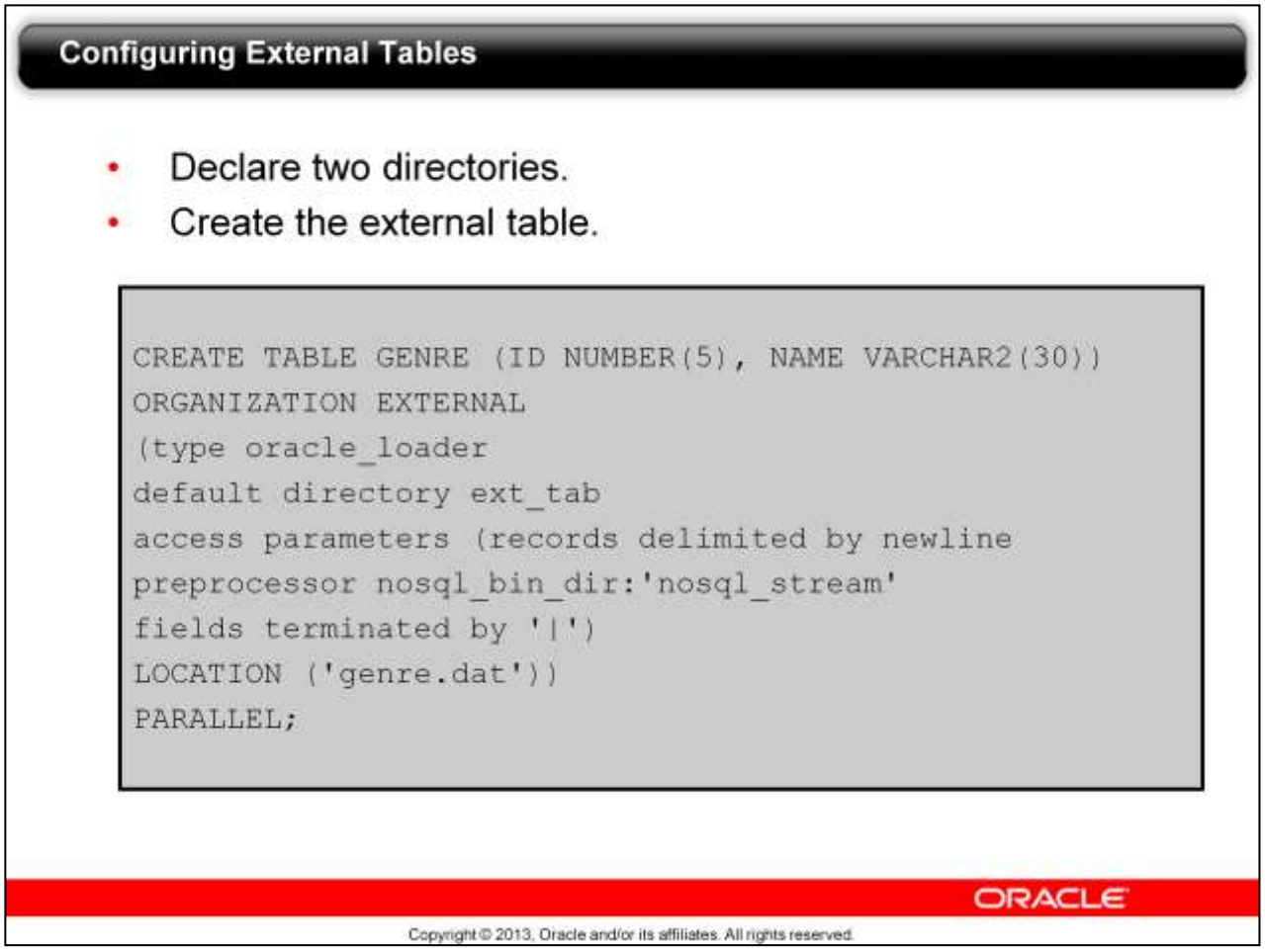


Take a look at this demonstration to understand how to:

- Start an Oracle NoSQL Database instance
- Compile the formatter classes
- Configure the `nosql_stream` script



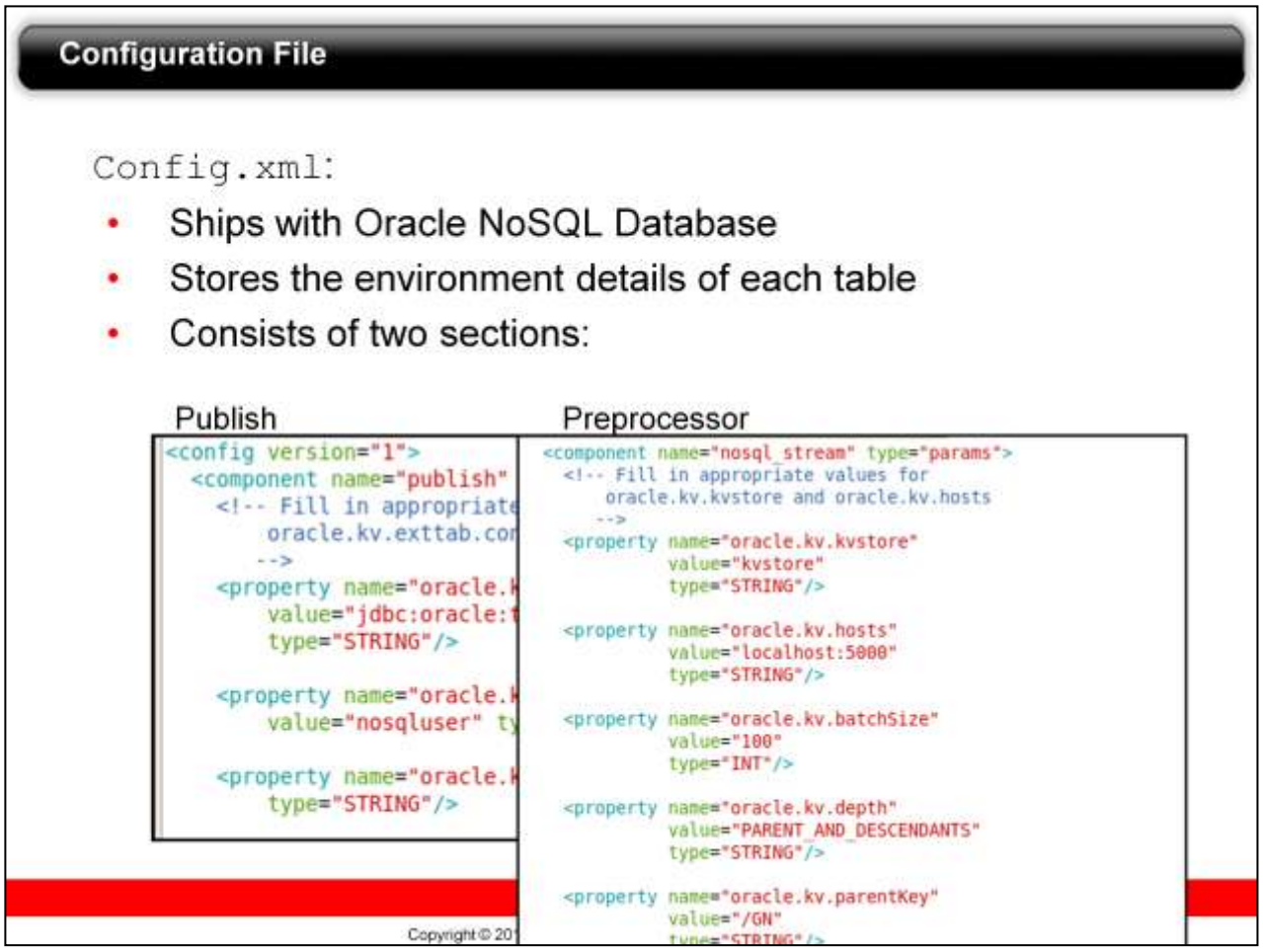

Let's move on to the next topic, "Configuring External Tables," where you learn about the steps to expose the data stored in Oracle NoSQL Database as external tables. You also learn to create the configuration file and edit the environment variables.




To configure the Oracle Database external table, you need to set up two directories, one to store the External Table location files and another to store the `nosql_stream` script. Then, you create the external table and specify the location files and the `nosql_stream` preprocessor, as shown in the slide.

So, when you run a SQL query on an external table, it looks for the preprocessor script that sets the classpath and then the .dat file is used to run the appropriate formatter class.

**Note:** Grant appropriate permissions to the Oracle Database users who will access the external table.




Before you can write the external tables to Oracle Database, you must create a configuration file for such values as connection URL, user, location files, KVStores, and hosts based on your Oracle Database and Oracle NoSQL Database installations. The configuration file is an XML document, and a template is available at `nosql/kv-2.1.15/examples`. The `config.xml` file consists of a publish section and a preprocessor section. Click each section to learn about the parameters.

*Note to producer: This slide needs to be converted into an interactive slide, when the user clicks on the appropriate section, its explanation comes up.*

## Publish (Hidden Slide)

Publish

```
<config version="1">
  <component name="publish" type="params">
    <!-- Fill in appropriate values for
        oracle.kv.exttab.connection.url and oracle.kv.exttab.connection.user
      -->
    <property name="oracle.kv.exttab.connection.url"
        value="jdbc:oracle:thin:/@localhost:1521:XE"
        type="STRING"/>

    <property name="oracle.kv.exttab.connection.user"
        value="nosqluser" type="STRING"/>

    <property name="oracle.kv.exttab.tableName" value="genre"
        type="STRING"/>
```

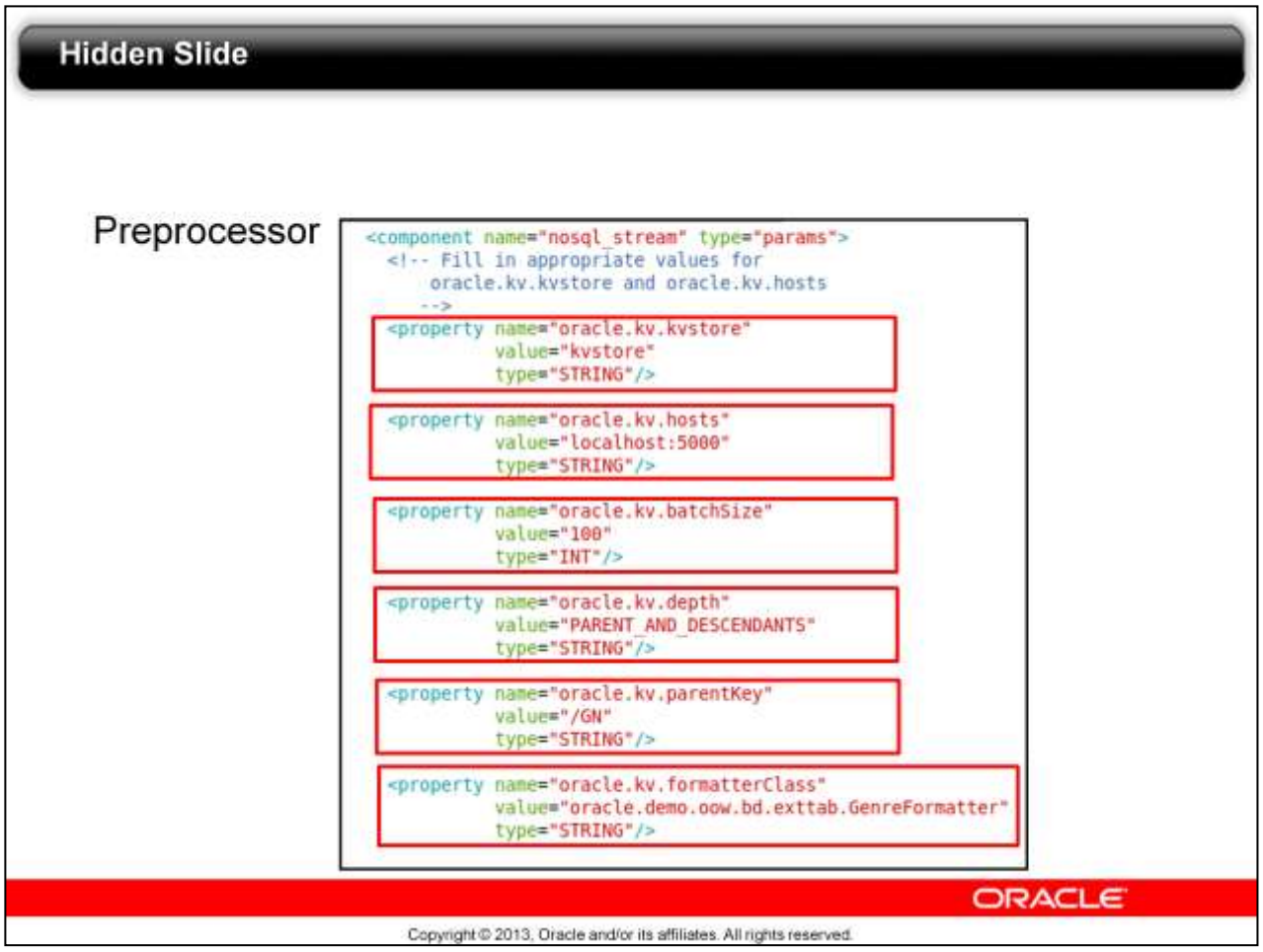
ORACLE



[1] `oracle.kv.exttab.connection.url`: The connection URL used to connect to the Oracle instance that contains the external table

[2] `oracle.kv.exttab.connection.user`: A required password prompt when a wallet is not used for connections

[3] `oracle.kv.exttab.tableName`: The schema-qualified table name of the external table in the format of `schemaName.tableName`




In the `nosql_stream` section, the following parameters are required:

[1]`oracle.kv.kvstore`: The name of the NoSQL database

[2]`oracle.kv.hosts`: One or more hostname:port pairs, separated by commas (The utility uses these hostname:port pairs to contact Rep Nodes and obtain information about the organization of Oracle NoSQL Database.)

The following parameters are optional:

[3]`oracle.kv.batchSize`: The number of records to fetch during each round trip

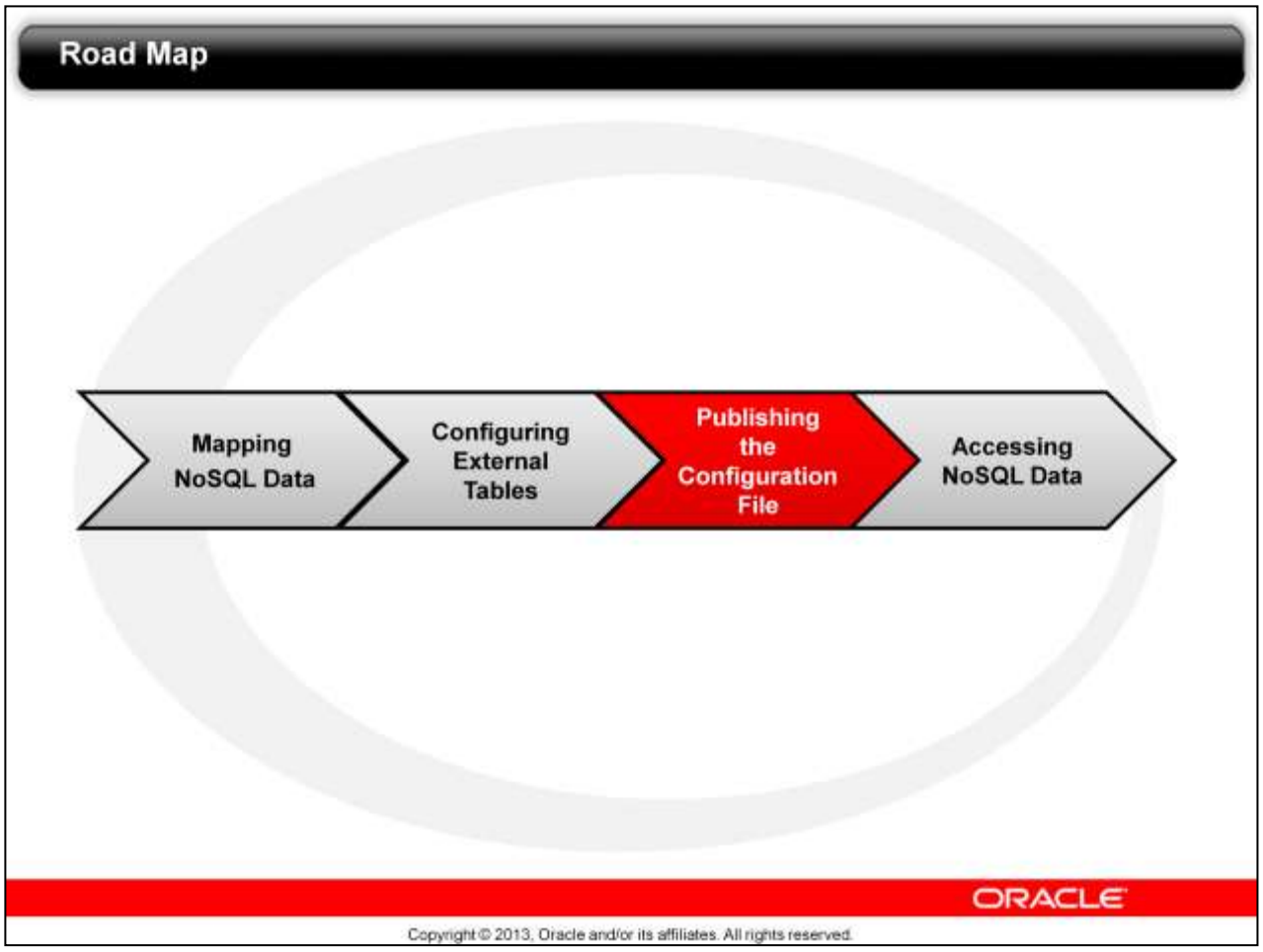
[4]`oracle.kv.depth`: One of `CHILDREN_ONLY`, `PARENT_AND_CHILDREN`, `DESCENDANTS_ONLY`, or `PARENT_AND_DESCENDANTS`

[5]`oracle.kv.parentKey`: The parent key whose child key-value (KV) pairs are to be fetched, specified in the canonical key format in `Key.toString()`

[6]`oracle.kv.formatterClass`: The name of a class that can appropriately format KV pairs for the Oracle Loader


Take a look at this demonstration to learn how to:

- Configure the information in Oracle NoSQL Database as external tables
- Edit the available configuration template to reflect the environment variables




Let's move on to the next topic, "Publishing the Configuration File." This topic focuses on publishing the configuration file.

## Publish Utility

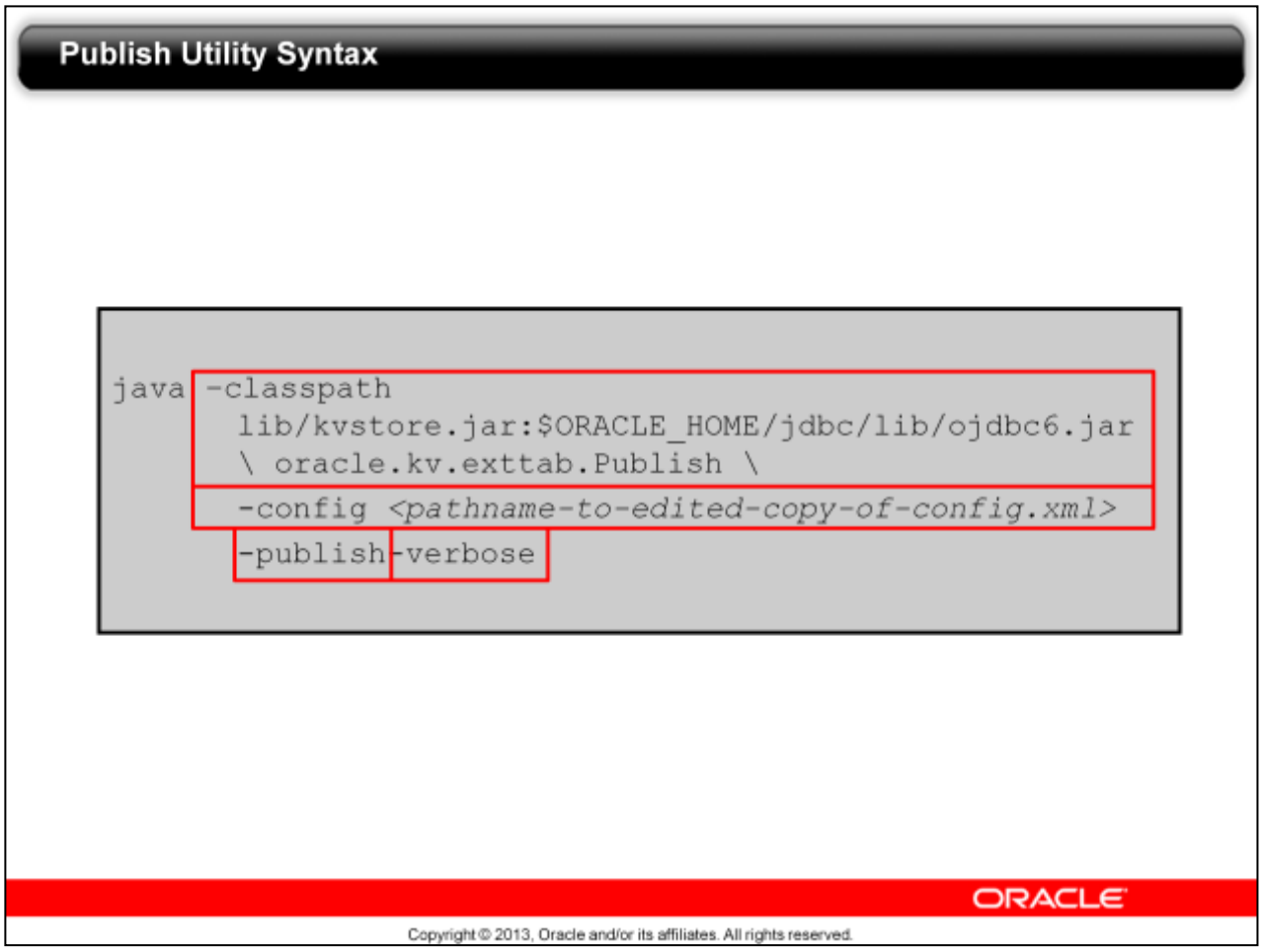
Publish Utility :

- Is shipped with Oracle NoSQL Database
- Writes connection information to connect to Oracle NoSQL Database from an Oracle Database external table
- Uses JDBC to connect between the two databases

ORACLE



After configuring the external tables and creating the config.xml file, you must invoke the Publish Utility, a program that ships with Oracle NoSQL Database. When you run the Publish Utility, a connection is established with the Oracle Database instance and the schema name and the information in the external table are extracted. Next, the preprocessor establishes a connection with Oracle NoSQL Database, reads the data according to the restrictions specified in the configuration, and formats the data by using the default formatting or the user-supplied binding class. And finally, the Oracle server reads the output and imports it into the external table.




The slide shows the syntax to run the `oracle.kv.exttab.Publish` utility to publish the configuration to the External Table location files.

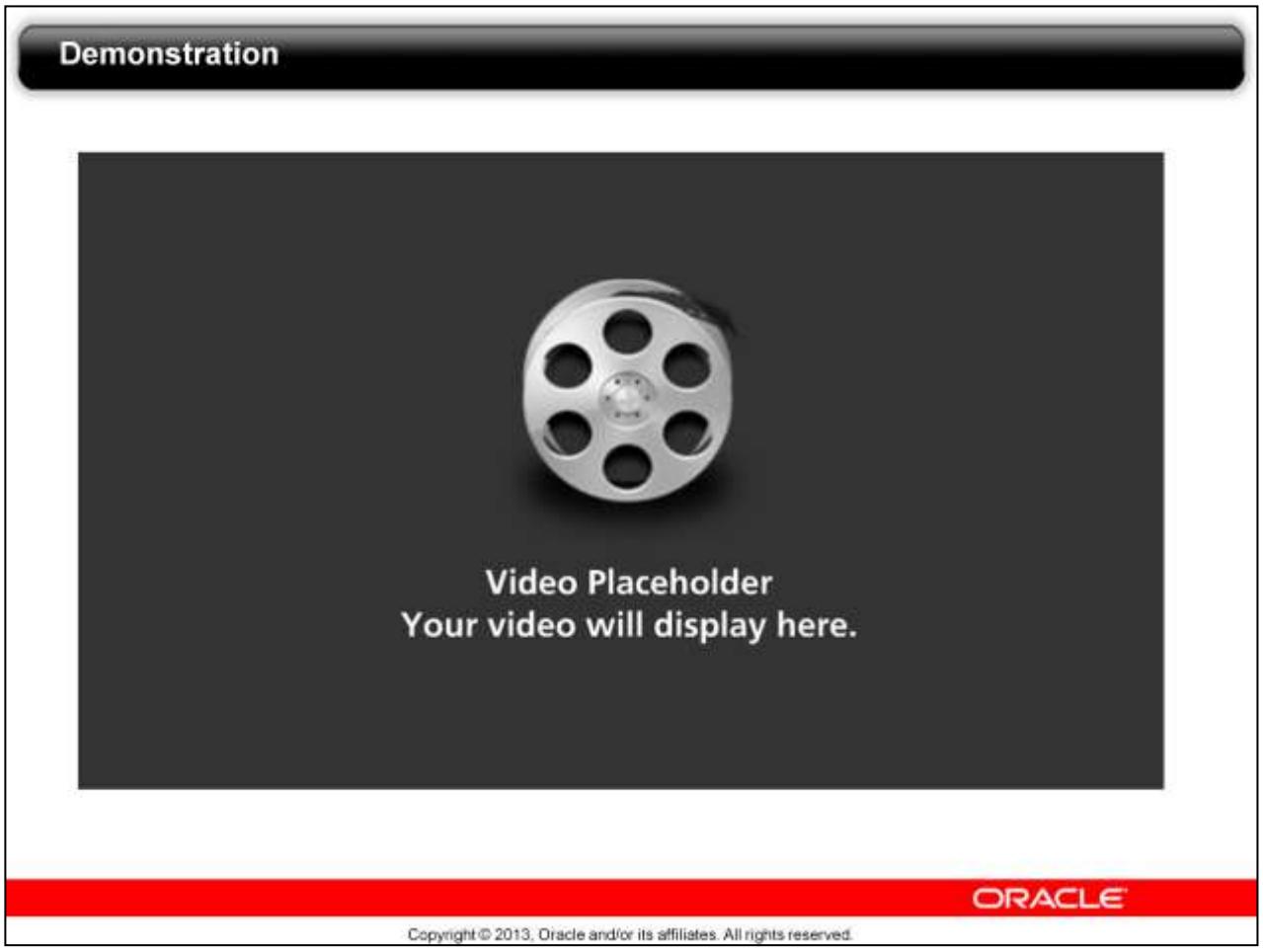
The Publish Utility accepts the following command-line parameters:

`[1]-classpath`: Specifies the path to the required JAR files

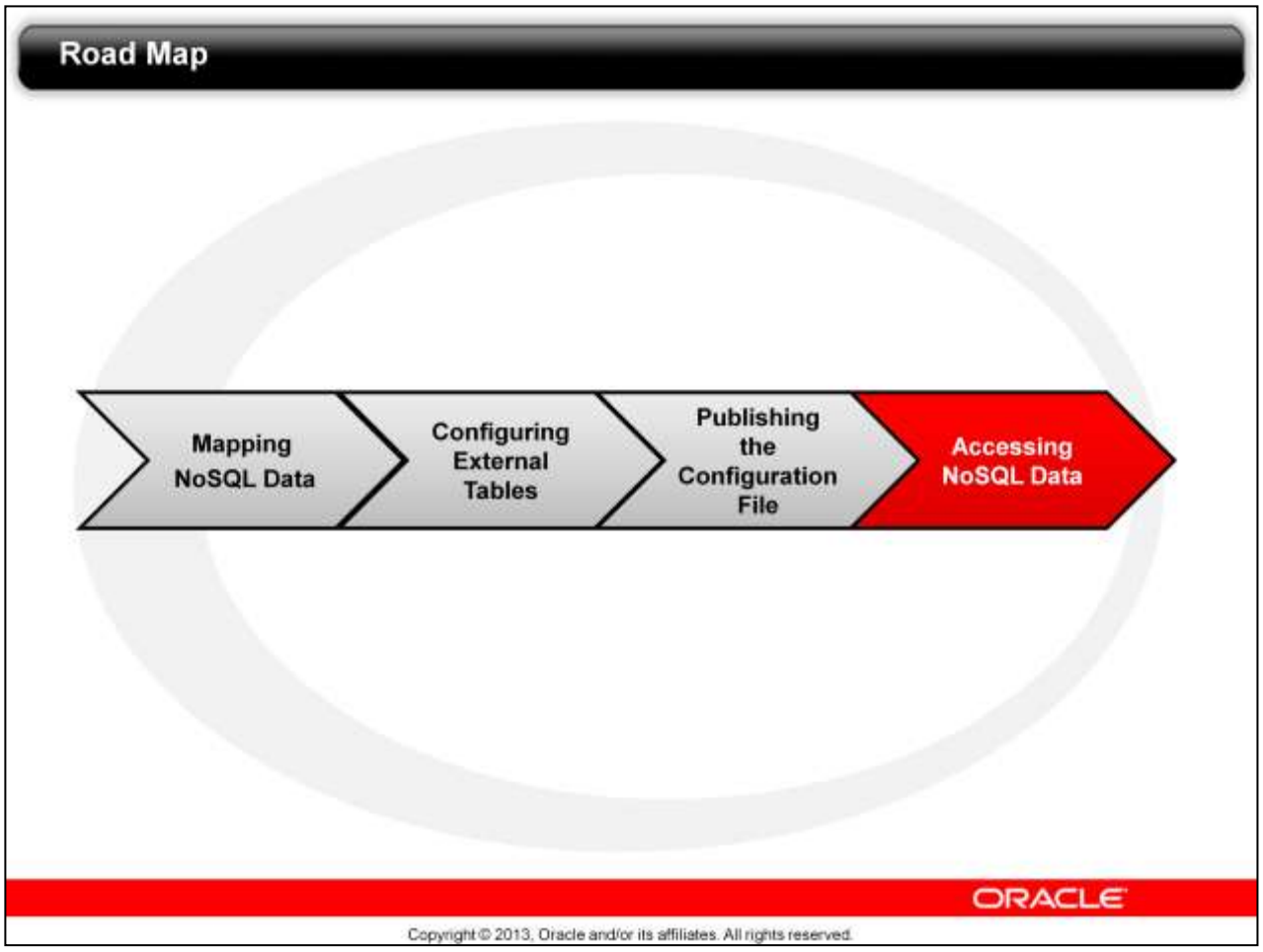
`[2]-config` <file>: Specifies the XML configuration file

`[3]-publish`: Writes the relevant information to the External Table location files

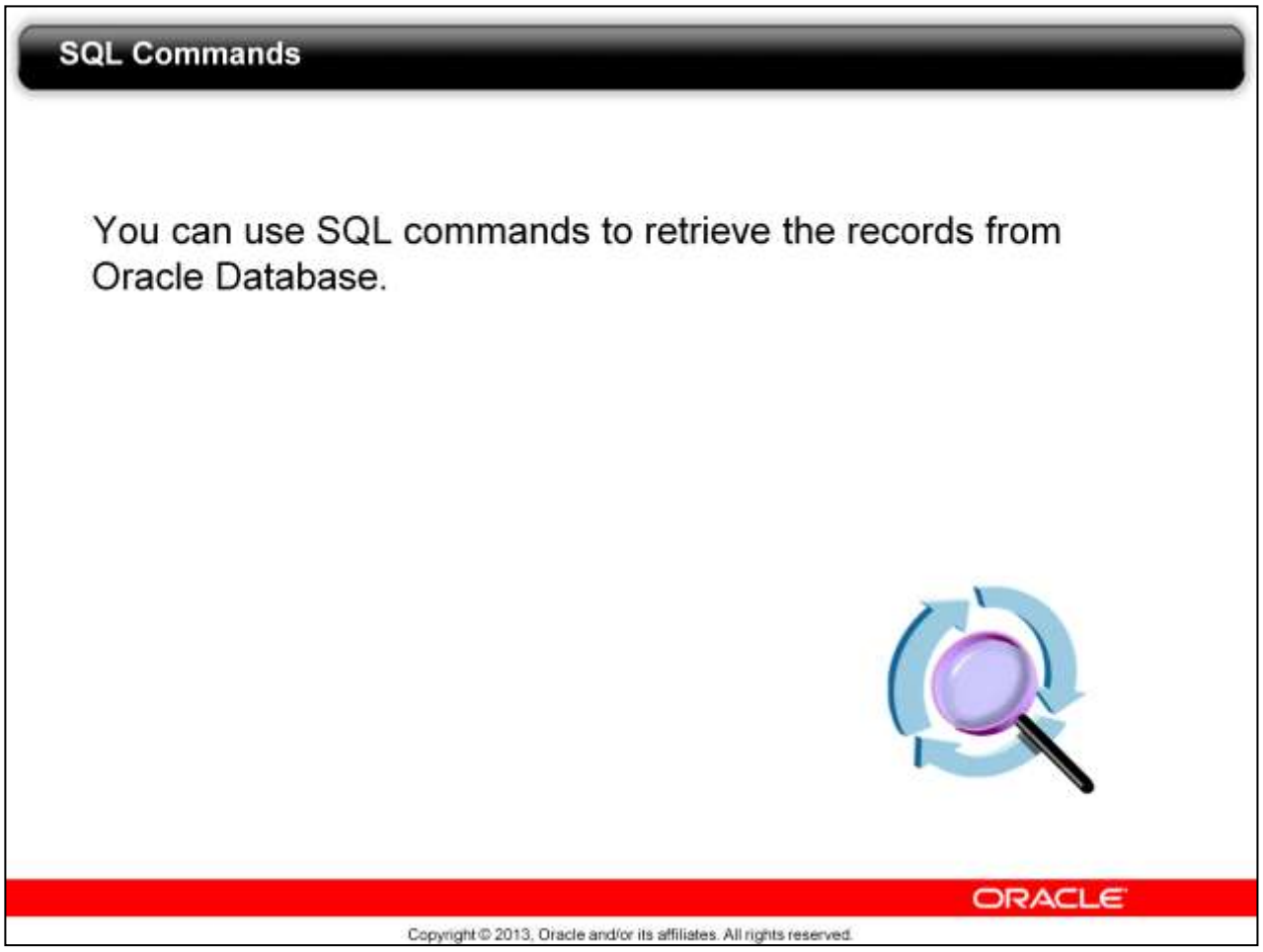
`[4]-verbose`: Writes debugging output to the console




Take a look at this demonstration to learn how to publish the configuration file.

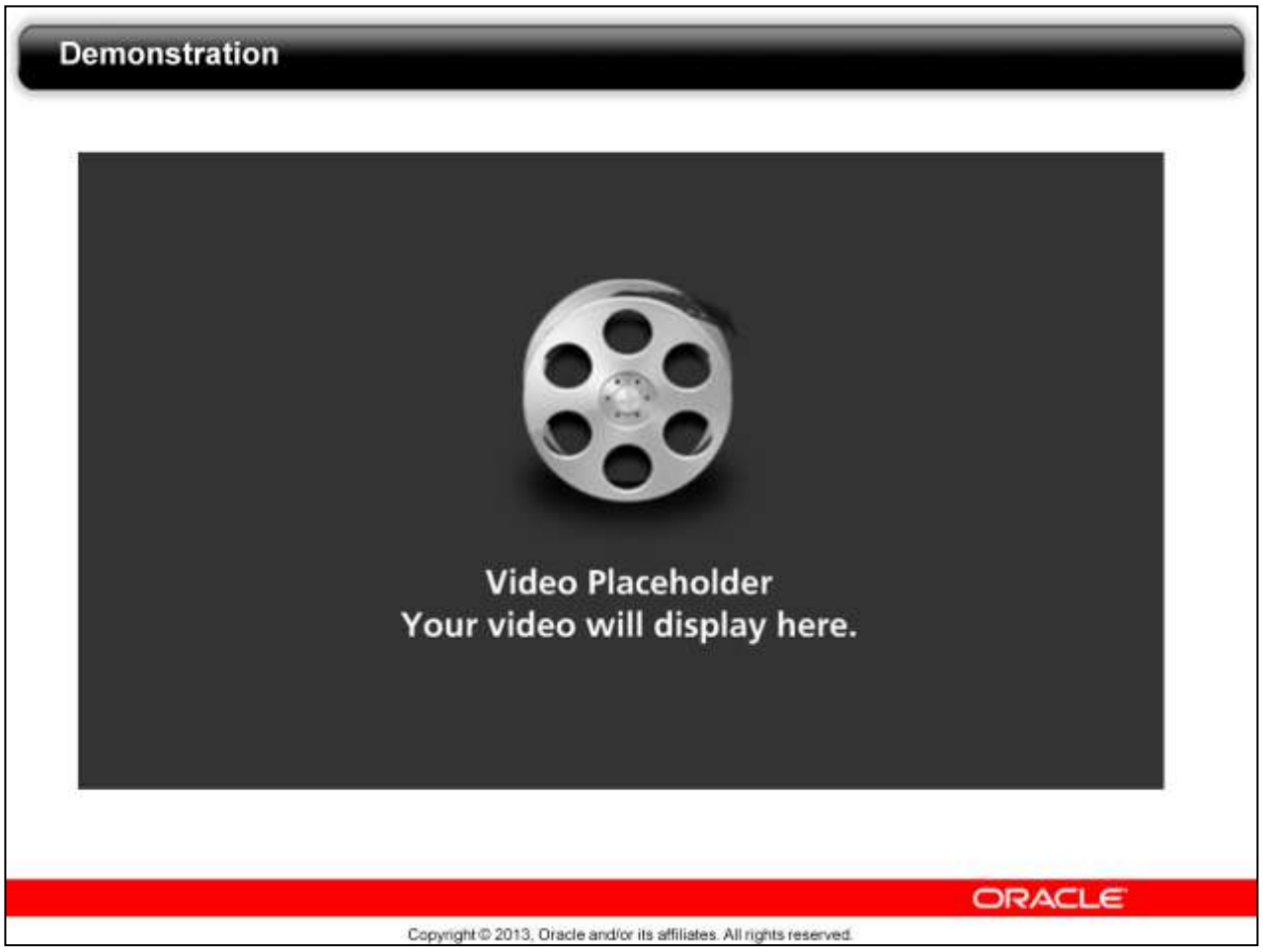



Let's move on to the next topic, "Accessing NoSQL Data." This topic focuses on using SQL commands to retrieve the NoSQL data.

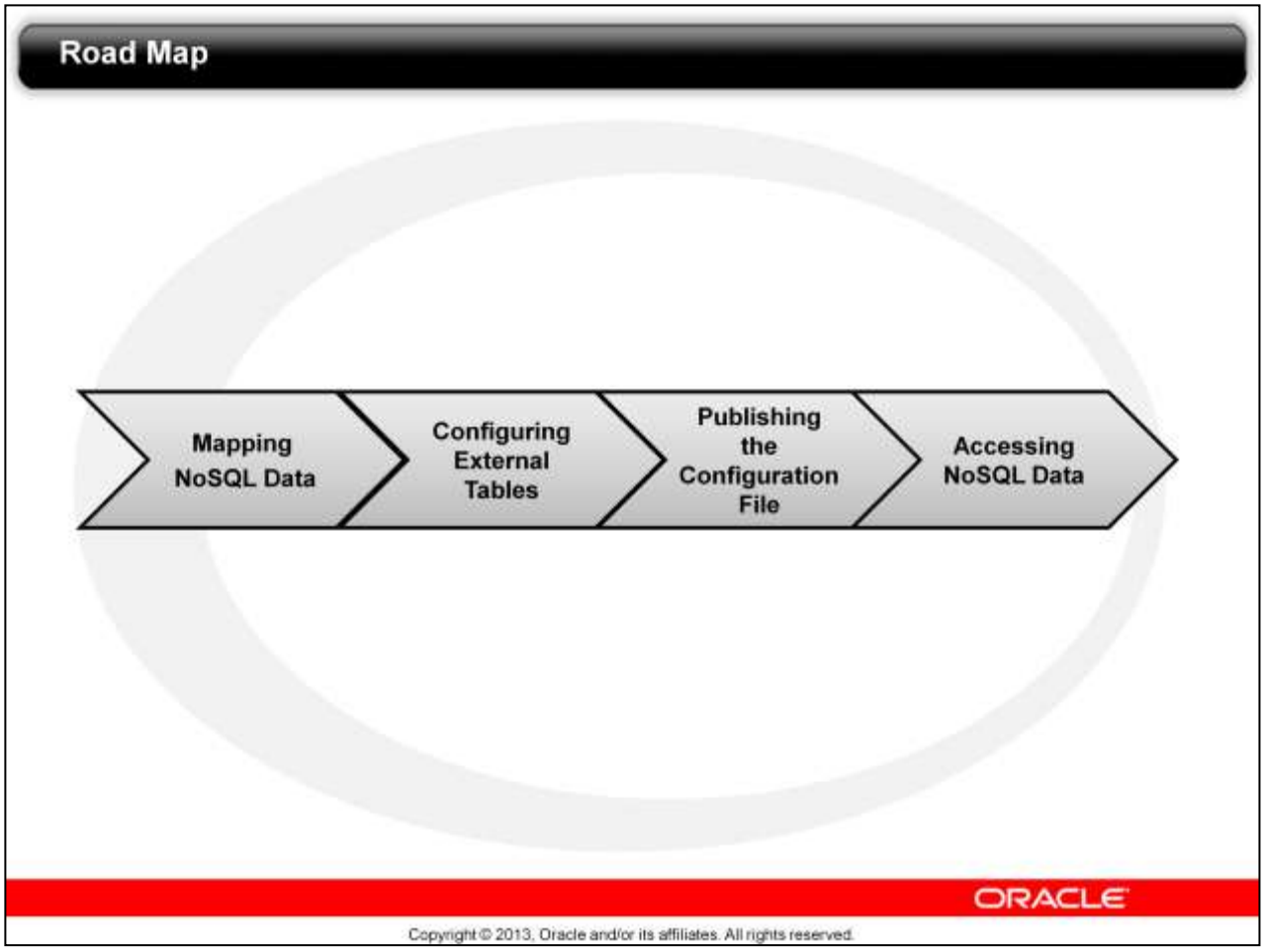



After configuring and publishing the external tables, you can use simple `SELECT` statements to access NoSQL data from Oracle Database.

While the `SELECT` statement is executing, the output of encountered exceptions is written to a file. The default log file in the External Table location file directory is `tablename_pid.log`.



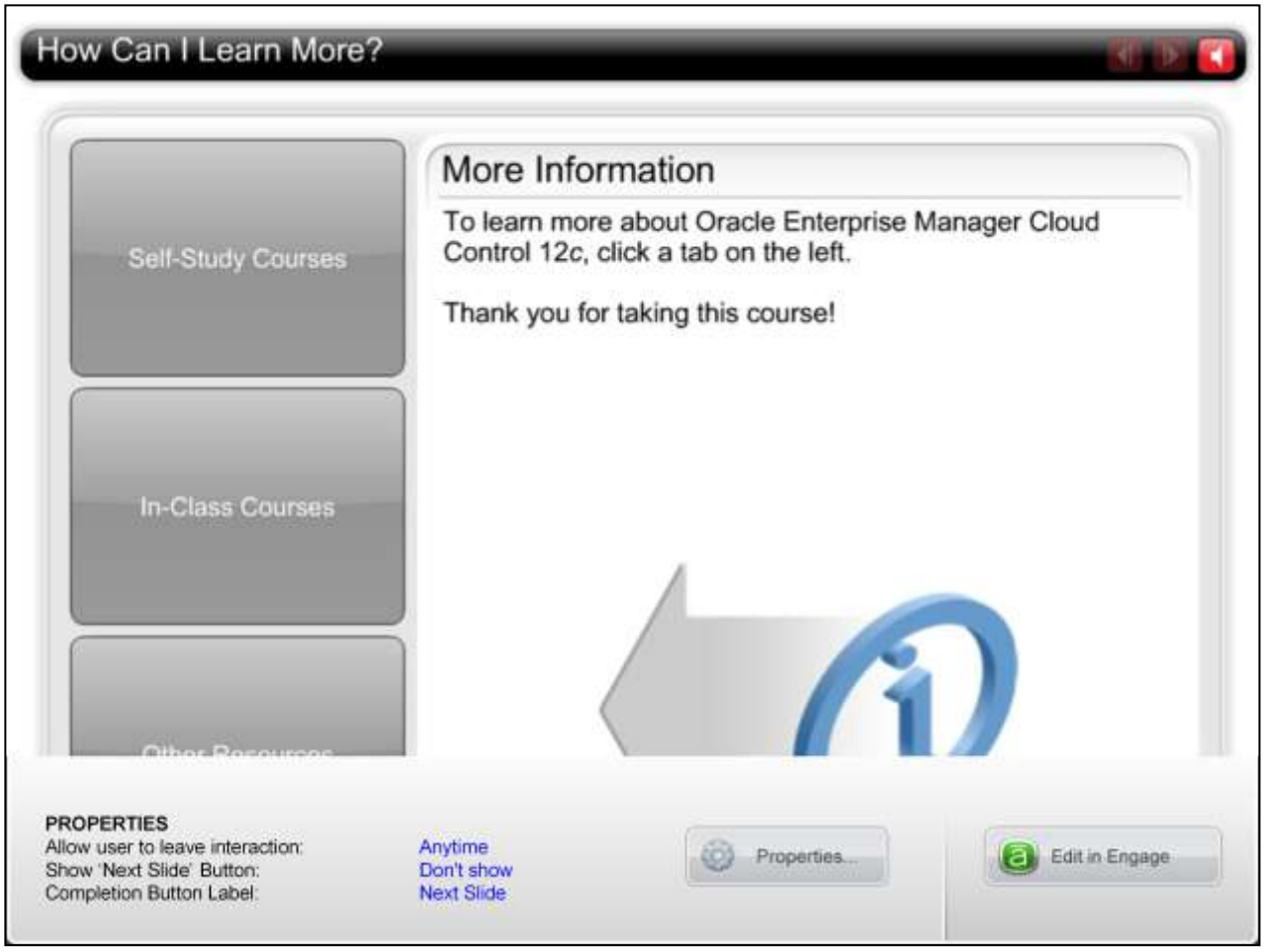

Take a look at this demonstration to learn how to use SQL statements to retrieve NoSQL data.




[1]In this course, we discussed [1] Mapping NoSQL Data to Oracle Database, [2] Configuring External Tables, [3]Publishing the Configuration File, and [4] Accessing NoSQL Data.

You should now be able to:

- Display data in Oracle NoSQL Database
- Configure external tables
- Create and publish the configuration file
- And, use SQL to access NoSQL data from Oracle Database




You can learn more about Oracle NoSQL Database from a variety of channels. Click a tab on the left to learn about just a few of the channels.

We know your time is valuable, and so we thank you for participating in this self-paced course, and hope you found it informative and useful.

*Note to Producer: The slide text needs to be changed to match this course.*

*To learn more about Oracle NoSQL Database, click a tab on the left.*

## Self-Study Courses (Hidden Slide)

More Oracle NoSQL Database self-study training is available from Oracle Learning Library:

- Hands-on with Oracle NoSQL Database

ORACLE



Oracle University provides additional self-paced courses like this one. Visit Oracle University to find more information about them.

## In-Class Courses (Hidden Slide)

Several In-Class courses are also available through Oracle University:

- Using Oracle NoSQL Database
- Introduction to Big Data

ORACLE



Oracle University provides In-Class courses, such as *Using Oracle NoSQL Database* and *Introduction to Big Data*. Visit Oracle University to find more information about them.

## Other Resources (Hidden Slide)

The Oracle Learning Library offers free demonstrations and tutorials.

Also view the Oracle NoSQL Database documentation and online help embedded within the product.

ORACLE



The Oracle Learning Library offers many free demonstrations and tutorials.

And, of course, the Oracle NoSQL Database documentation and online help embedded within the product are also valuable resources.

## Credits

Author and Narrator
Prathima Trivedi

Technical Contributor & Reviewer
Anuj Sahni

**Thanks for watching!**

Editors
Susan Moxley

Producers
Swarnapriya Shridhar

ORACLE