

17

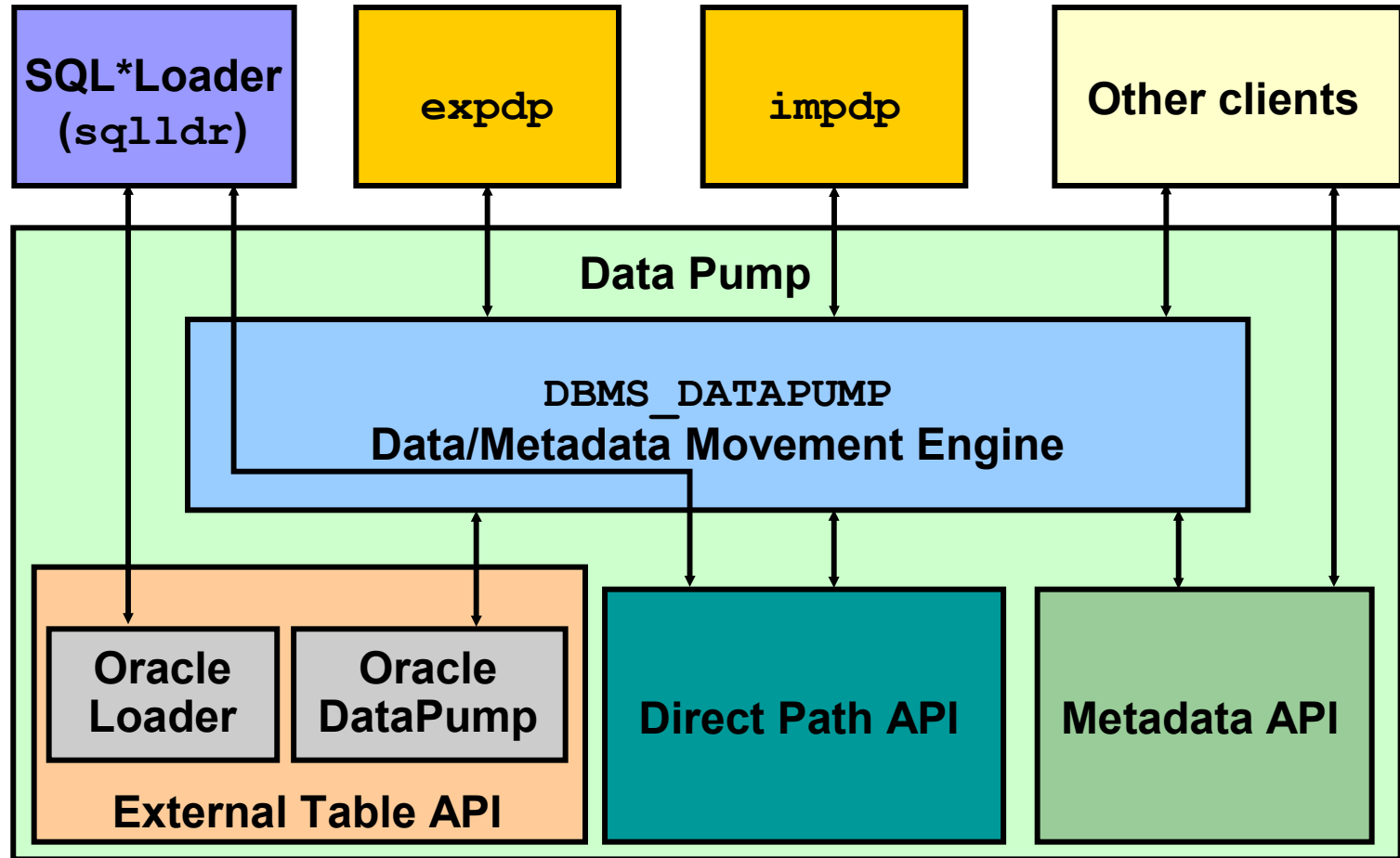
Moving Data

Objectives

After completing this lesson, you should be able to:

- Describe ways to move data
- Create and use directory objects
- Use SQL*Loader to load data from a non-Oracle database (or user files)
- Use external tables to move data via platform-independent files
- Explain the general architecture of Oracle Data Pump
- Use Data Pump Export and Import to move data between Oracle databases

Moving Data: General Architecture



Oracle Data Pump: Overview

As a server-based facility for high-speed data and metadata movement, Oracle Data Pump:

- Is callable via `DBMS_DATAPUMP`
- Provides the following tools:
 - `expdp`
 - `impdp`
 - Web-based interface
- Provides four data movement methods:
 - Data file copying
 - Direct path
 - External tables
 - Network link support
- Detaches from and reattaches to long-running jobs
- Restarts Data Pump jobs



Oracle Data Pump: Benefits

Data Pump offers many benefits and some new features over earlier data movement tools, such as:

- Fine-grained object and data selection
- Explicit specification of database version
- Parallel execution
- Estimation of export job space consumption
- Network mode in a distributed environment
- Remapping capabilities
- Data sampling and metadata compression
- Compression of data during a Data Pump export
- Security through encryption
- Ability to export XMLType data as CLOBs
- Legacy mode to support old import and export files

Directory Objects for Data Pump

ORACLE Enterprise Manager 11g Database Control

Setup Preferences Help Logout

Database

Database Instance: orcl.oracle.com >

Logged in As SYS

Directory Objects

Search

Object Name

By default, the search returns all uppercase matches beginning with the string you entered. To run an exact or case-sensitive match, double quote the search string. You can use the wildcard symbol (%) in a double quoted string.

Selection Mode

Actions

Select	Name	Path
<input checked="" type="radio"/>	DATA_FILE_DIR	/u01/app/oracle/product/11.2.0/dbhome_1/demo/schema/sales_history/
<input type="radio"/>	DATA_PUMP_DIR	/u01/app/oracle/admin/orcl/dpdump/
<input type="radio"/>	LOG_FILE_DIR	/u01/app/oracle/product/11.2.0/dbhome_1/demo/schema/log/
<input type="radio"/>	MEDIA_DIR	/u01/app/oracle/product/11.2.0/dbhome_1/demo/schema/product_media/
<input type="radio"/>	ORACLE_OCM_CONFIG_DIR	/u01/app/oracle/product/11.2.0/dbhome_1/ccr/state
<input type="radio"/>	SS_OE_XMLDIR	/u01/app/oracle/product/11.2.0/dbhome_1/demo/schema/order_entry/
<input type="radio"/>	SUBDIR	/u01/app/oracle/product/11.2.0/dbhome_1/demo/schema/order_entry//2002/Sep
<input type="radio"/>	XMLDIR	/ade/b/1191423112/oracle/rdbms/xml

Database | Setup | Preferences | Help | Logout

Creating Directory Objects

1 Create

Selection Mode **Single**

Edit View Delete Actions **Create Like** Go

Select	Name	Path
<input checked="" type="checkbox"/>	DATA_FILE_DIR	/u01/app/oracle/p

3

General **Privileges**

This page shows the list of users who have privileges for this directory

Add

Select Remove

Select All | Select None

Select	User Name	Read Access	Write Access
<input type="checkbox"/>	HR	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

2

Create Directory Object

General **Privileges**

5

Name ext_tab_logdir

Path /home/oracle/labs/extab1

Test File System

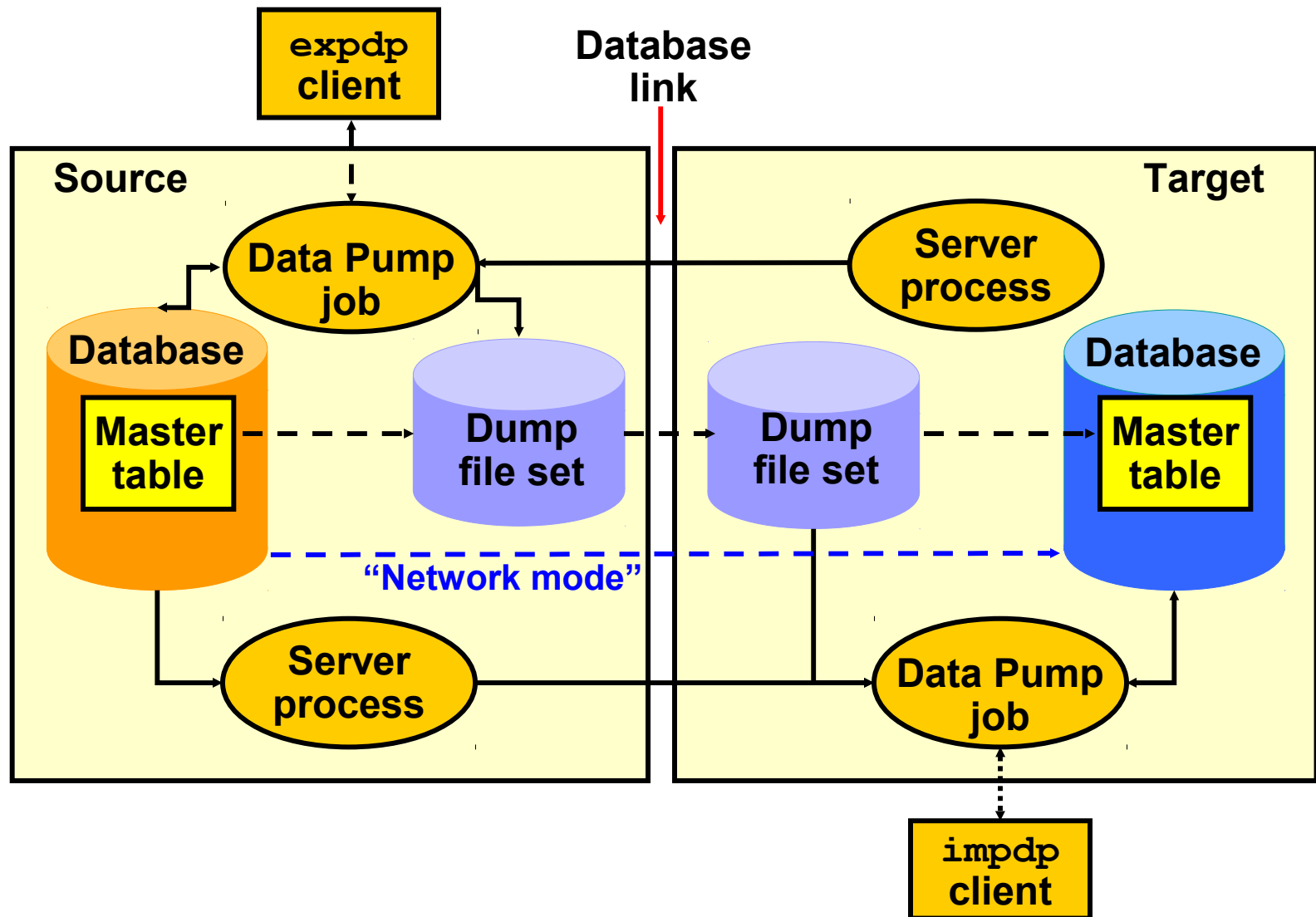
4

Show SQL

Return

```
CREATE DIRECTORY "ext_tab_logdir" AS '/home/oracle/labs/extab1'
GRANT READ ON DIRECTORY "ext_tab_logdir" TO "HR"
GRANT WRITE ON DIRECTORY "ext_tab_logdir" TO "HR"
```

Data Pump Export and Import Clients: Overview



Data Pump Utility: Interfaces and Modes

- Data Pump Export and Import interfaces:
 - Command line
 - Parameter file
 - Interactive command line
 - Enterprise Manager
- Data Pump Export and Import modes:
 - Full
 - Schema
 - Table
 - Tablespace
 - Transportable tablespace



Data Pump Export using Database Control

Database Instance: orcl.oracle.com

Home	Performance	Availability	Server	Schema	Data Movement	Software and Support
----------------------	-----------------------------	------------------------------	------------------------	------------------------	-------------------------------	--------------------------------------

Move Row Data Export to Export Files Import from Export Files Import from Database Load Data from User Files Monitor Export and Import Jobs	Move Database Files Clone Database Transport Tablespaces	Streams Setup Manage Replication Manage Advanced Queues	Advanced Replication Setup Manage
---	---	---	--

Database Instance: orcl.oracle.com >

Export: Export Type

Database orcl.oracle.com Cancel Continue

☒ Schemas
Allows you to export the objects in your schema.

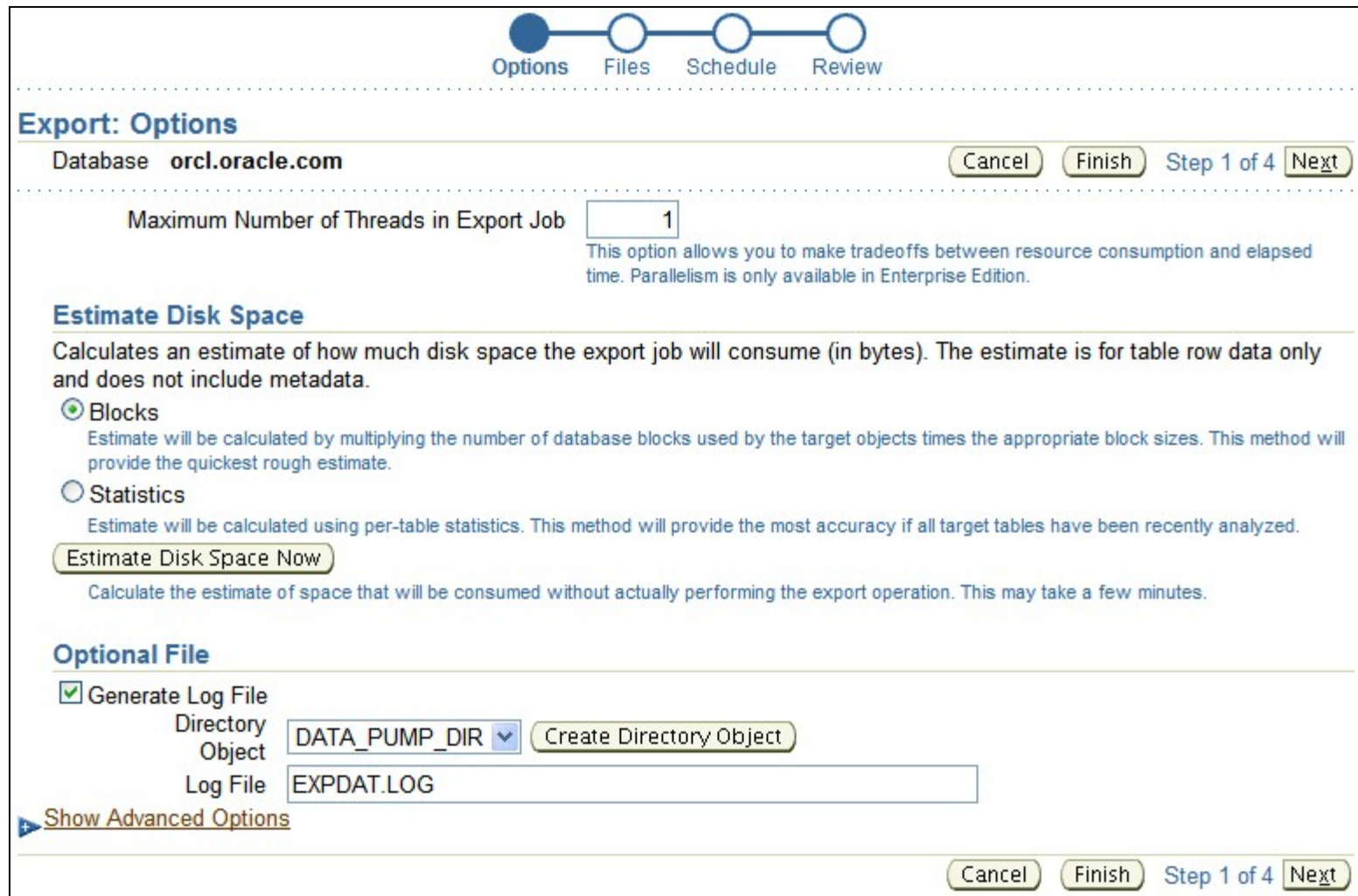
☐ Tables
Allows you to choose one or more tables in your schema to export.

Host Credentials

* Username	<input type="text" value="oracle"/>
* Password	<input type="password" value="....."/>
<input checked="" type="checkbox"/> Save as Preferred Credential	

Cancel Continue

Data Pump Export Example: Basic Options



The image shows a screenshot of the 'Data Pump Export Options' dialog box. At the top, there is a progress bar with four steps: 'Options' (selected), 'Files', 'Schedule', and 'Review'. Below the progress bar, the title 'Export: Options' is displayed. The 'Database' is set to 'orcl.oracle.com'. There are buttons for 'Cancel', 'Finish', 'Step 1 of 4', and 'Next'. The 'Maximum Number of Threads in Export Job' is set to '1'. A note explains that this option allows for tradeoffs between resource consumption and elapsed time, and that parallelism is only available in Enterprise Edition. The 'Estimate Disk Space' section is active, showing two radio buttons: 'Blocks' (selected) and 'Statistics'. The 'Blocks' method description states it will provide the quickest rough estimate. The 'Statistics' method description states it will provide the most accuracy if all target tables have been recently analyzed. There is a button 'Estimate Disk Space Now' and a note that calculating the estimate may take a few minutes. The 'Optional File' section has a checked 'Generate Log File' option. The 'Directory Object' is set to 'DATA_PUMP_DIR' with a 'Create Directory Object' button. The 'Log File' is set to 'EXPDAT.LOG'. At the bottom left, there is a link 'Show Advanced Options'. At the bottom right, there are buttons for 'Cancel', 'Finish', 'Step 1 of 4', and 'Next'.

Options Files Schedule Review

Export: Options

Database **orcl.oracle.com** Cancel Finish Step 1 of 4 Next

Maximum Number of Threads in Export Job This option allows you to make tradeoffs between resource consumption and elapsed time. Parallelism is only available in Enterprise Edition.

Estimate Disk Space

Calculates an estimate of how much disk space the export job will consume (in bytes). The estimate is for table row data only and does not include metadata.

☒ Blocks
Estimate will be calculated by multiplying the number of database blocks used by the target objects times the appropriate block sizes. This method will provide the quickest rough estimate.

☐ Statistics
Estimate will be calculated using per-table statistics. This method will provide the most accuracy if all target tables have been recently analyzed.

Estimate Disk Space Now

Calculate the estimate of space that will be consumed without actually performing the export operation. This may take a few minutes.

Optional File

☒ Generate Log File

Directory Object Create Directory Object

Log File

[Show Advanced Options](#)

Cancel Finish Step 1 of 4 Next

Data Pump Export Example: Advanced Options

Content

What to Export from the Source Database ☒ All

Export both metadata and data

☐ Data Only

Export only table row data

☐ Metadata Only

Export only database object definitions

Export Content ☒ Include All Objects

☐ Include Only Objects Specified Below

☐ Exclude Only Objects Specified Below

Objects to Include or Exclude

Select	Object Type	Object Name Expression
		No items found
Add Another Row		

Object Name Expression example: "IN('EMP','DEPT')" or, to include every object except those of a particular type not beginning with PRO, select EXCLUDE with an expression of "NOT LIKE 'PRO%'"

Flashback

☐ Export read-consistent view of data

☒ As the specified System Change Number (SCN)

SCN

☐ As the SCN which most closely matches the specified time

Date

Time ☒ AM ☐ PM

Query

Specify SELECT statement predicate clauses to be applied to tables being exported. If a Table Name is not supplied for a particular Predicate Clause, the Predicate Clause is applied to (and must make sense for) all tables being exported.

Select	Predicate Clause	Table Name
		No items found
Add		

Data Pump Export Example: Files

Options

Files

Schedule

Review

Export: Files

Database **orcl.oracle.com**

CancelFinishBackStep 2 of 4Next

Specify the directory object and file name, and maximum size for the export files on the database server machine.

Create Directory Object

Remove

Select	Directory Object	File Name	Maximum File Size (MB)
<input checked="" type="radio"/>	DATA_PUMP_DIR	HR_SCHEMA.DMP	

Add Another Row

You can wildcard a set of dump files using '%U' in the filename. A '%D' wildcard will be replaced with the date the job is run using a YYMMDD format.

CancelFinishBackStep 2 of 4Next

[Database](#) | [Help](#) | [Logout](#)

Data Pump Export Example: Schedule

Options

Files

Schedule

Review

Export: Schedule

Database **orcl.oracle.com** Cancel Back Step 3 of 4 Next

Specify a name and description for the export job. Specify a date to start the job.

Job Parameters


Job Name	Export_HR_Schema
Description	Full expora of HR schema

Job Schedule

Time Zone (UTC+00:00) Universal Time

Start

☒ Immediately
☐ Later

Date Jul 11, 2009 
(example: Jul 11, 2009)

Time 5 20 ☐ AM ☒ PM

Repeat


☒ One Time Only
☐ Interval

Frequency 1 Minutes

☐ Monthly
☐ Yearly

Repeat Until

☒ Indefinite
☐ Custom

Date Jul 11, 2009 
(example: Jul 11, 2009)

Time 5 30 ☒ AM ☐ PM
(Ignored except when repeating by minutes or hours.)

ORACLE

Data Pump Export Example: Review

OptionsFilesScheduleReview

Export: Review

Database orcl.oracle.com

CancelBackStep 4 of 4Submit Job

Export Type Schemas

Statistics type Estimate optimizer statistics when data is imported

Parallelism 1

Files to Export DATA_PUMP_DIR HR_SCHEMA.DMP

Log File DATA_PUMP_DIR EXPDAT.LOG

Job Schedule Immediately


Hide PL/SQL

```
declare
  h1 NUMBER;
begin
  h1 := dbms_datapump.open (operation => 'EXPORT', job_mode => 'SCHEMA', job_name =>
'EXPORT_HR_SCHEMA', version => 'COMPATIBLE');
  dbms_datapump.set_parallel(handle => h1, degree => 1);
  dbms_datapump.add_file(handle => h1, filename => 'EXPDAT.LOG', directory => 'DATA_PUMP_DIR',
filetype => 3);
  dbms_datapump.set_parameter(handle => h1, name => 'KEEP_MASTER', value => 0);
  dbms_datapump.metadata_filter(handle => h1, name => 'SCHEMA_EXPR', value => 'IN(''HR'')');
```


Processing

Submit job is progressing. This may take some time.

This may take several minutes. This page will automatically forward to the next page when done.



Process is in progress.

 **TIP** This operation cannot be cancelled. It will continue even if the browser window is closed.

Data Pump Import Example: impdp

Data Pump can be invoked on the command line to allow further command line options to be specified.

```
$ impdp hr DIRECTORY=DATA_PUMP_DIR \  
DUMPFILE=HR_SCHEMA.DMP \  
PARALLEL=1 \  
CONTENT=ALL \  
TABLES="EMPLOYEES" \  
LOGFILE=DATA_PUMP_DIR:import_hr_employees.log \  
JOB_NAME=importHR \  
TRANSFORM=STORAGE:n
```


Data Pump Import: Transformations

You can remap:

- Data files by using `REMAP_DATAFILE`
- Tablespaces by using `REMAP_TABLESPACE`
- Schemas by using `REMAP_SCHEMA`
- Tables by using `REMAP_TABLE`
- Data by using `REMAP_DATA`

```
REMAP_TABLE = 'EMPLOYEES' : 'EMP'
```

Using Enterprise Manager to Monitor Data Pump Jobs

Database Instance: orcl.oracle.com


[Home](#) [Performance](#) [Availability](#) [Server](#) [Schema](#) **Data Movement** [Software and Support](#)

Move Row Data	Move Database Files	Streams	Advanced Replication
Export to Export Files	Clone Database	Setup	Setup
Import from Export Files	Transport Tablespaces	Manage	Manage
Import from Database			
Load Data from User Files			
Monitor Export and Import Jobs			

Export and Import Jobs

Page Refreshed Sep 1, 2008 12:23:20 AM MDT

In database versions 10g and greater, Enterprise Manager uses data pump jobs to do the actual export and import operations. Although Enterprise Manager exports and imports can also be monitored from their corresponding Job Summary pages, data pump jobs defined outside of Enterprise Manager can only be monitored from here.

Select	Data Pump Job	EM Job	Owner	Job Status
	INVENTORY_EXPORT	Yes	DBA1	EXECUTING

Migration with Data Pump Legacy Mode

- Assistance in transitioning from `imp` and `exp` utilities to `impdp` and `expdp` utilities
- Data Pump in legacy mode:
 1. Encounters unique `imp` or `exp` parameter and enters legacy mode
 2. Attempts to map the old syntax to the new syntax
 3. Displays new syntax
 4. Exits legacy mode

Best practice tip: Oracle strongly recommends that you view the new syntax and make script changes as time permits.



Data Pump Legacy Mode

The Data Pump export and import utilities:

- Read and write files only in Data Pump format
- Accept `exp` and `imp` utility commands in legacy mode
- Include legacy mode parameters that:

- Can be identical to the new syntax:

```
FILESIZE=integer[B | K | M | G]
```

- Can be similar:

```
QUERY= query_clause
```

- Are ignored, when the command is superceded by Data Pump defaults

```
BUFFER=integer
```

```
COMPRESS={y|n}
```

```
DIRECT={y|n}
```

- Cause an error when old and new syntax is mixed

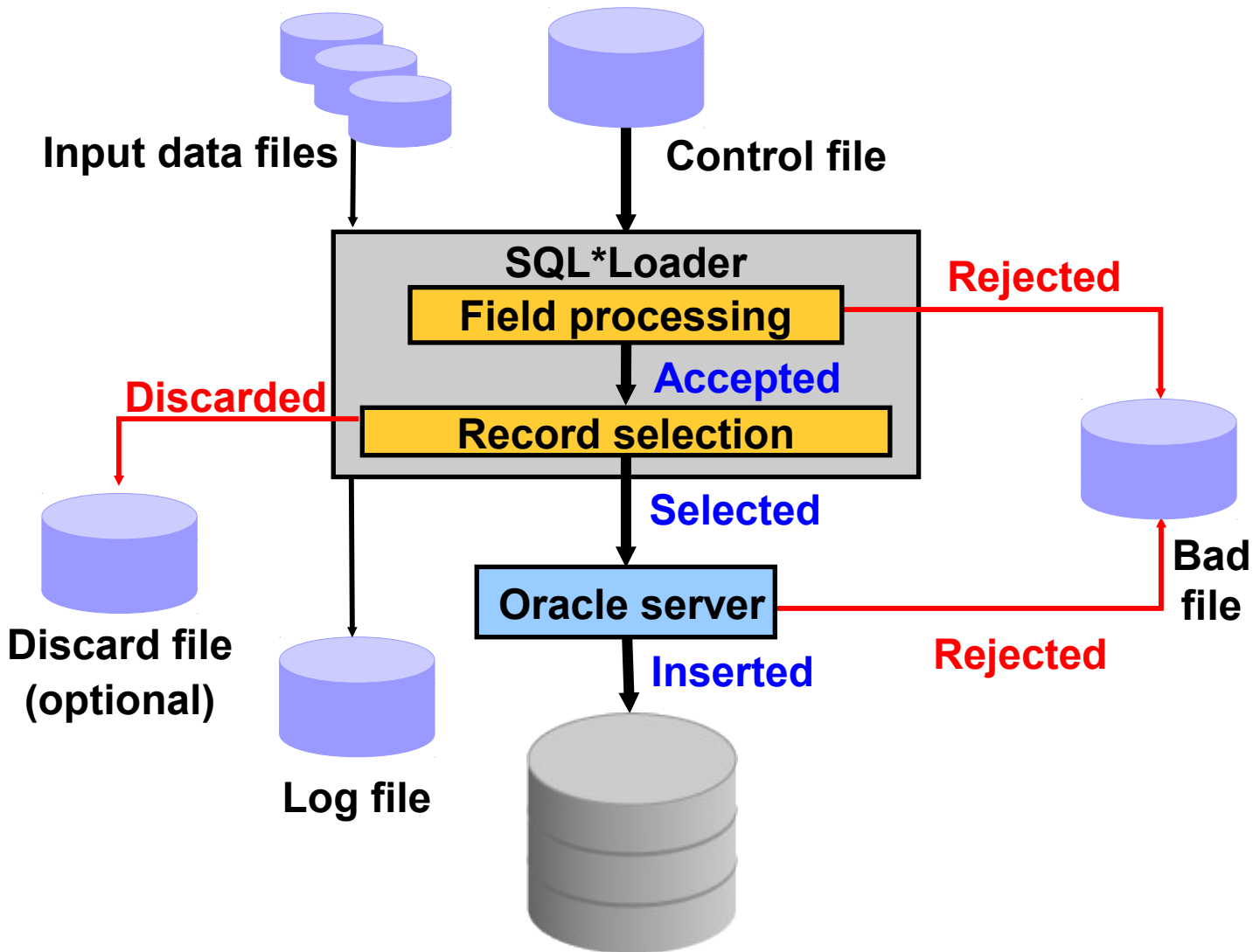
Data Pump Legacy Mode

- Legacy mode parameters:
 - Are mapped to Data Pump parameters, if possible:
`consistent={y|n} -> FLASHBACK_TIME`
`GRANTS=n -> EXCLUDE=CONSTRAINTS`
`INDEXES=n -> EXCLUDE=INDEX`
`LOG=filename -> LOGFILE=filename`
`FILE=filename -> dumpfile=directory-object:filename`
 - Can be similar, but not identical:
`FEEDBACK=integer -> STATUS`
 - Cause an error when incompatible with new Data Pump:
`VOLSIZE=integer`

Managing File Locations

- Original `exp` and `imp` utilities: Fully qualified file names
- Data Pump directory object for file locations
 - Default (in prior versions): `DATA_PUMP_DIR` parameter
 - New optional `DATA_PUMP_DIR_schema-name` directory object
 - Managed with the `CREATE DIRECTORY` and `GRANT SQL` commands
 - Default location (independent of legacy mode), when:
 - Command line without `DIRECTORY` parameter
 - User without `EXP_FULL_DATABASE` privilege

SQL*Loader: Overview



Loading Data with SQL*Loader

Load Data: Generate Or Use Existing Control File

Database **orcl.oracle.com** Cancel Continue

☐ Automatically Generate Control File
A control file will be generated after you define the structure of the data file.

☒ Use Existing Control File
Allows you to use an existing control file that defines the structure of the data file.

Host Credentials

* Username

* Password

☐ Save as Preferred Credential

Control File Data File Load Method Options Schedule Review

Load Data: Control File

Database **orcl.oracle.com** Cancel Finish Step 1 of 6 Next

A control file is used to describe what will be loaded and how. Specify the full path and name of the control file on the database server machine.

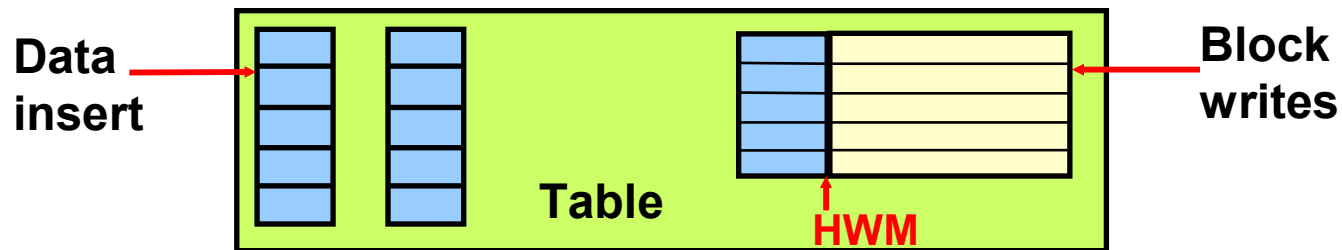
SQL*Loader Control File

The SQL*Loader control file instructs SQL*Loader about:

- Location of the data to be loaded
- Data format
- Configuration details:
 - Memory management
 - Record rejection
 - Interrupted load handling details
- Data manipulation details



Loading Methods



Conventional Load

Uses `COMMIT`

Always generates redo entries

Enforces all constraints

Fires `INSERT` triggers

Can load into clustered tables

Allows other users to modify tables during load operation

Maintains index entries on each insert

Direct Path Load

Uses data saves (faster operation)

Generates redo only under specific conditions

Enforces only `PRIMARY KEY`, `UNIQUE`, and `NOT NULL`

Does not fire `INSERT` triggers

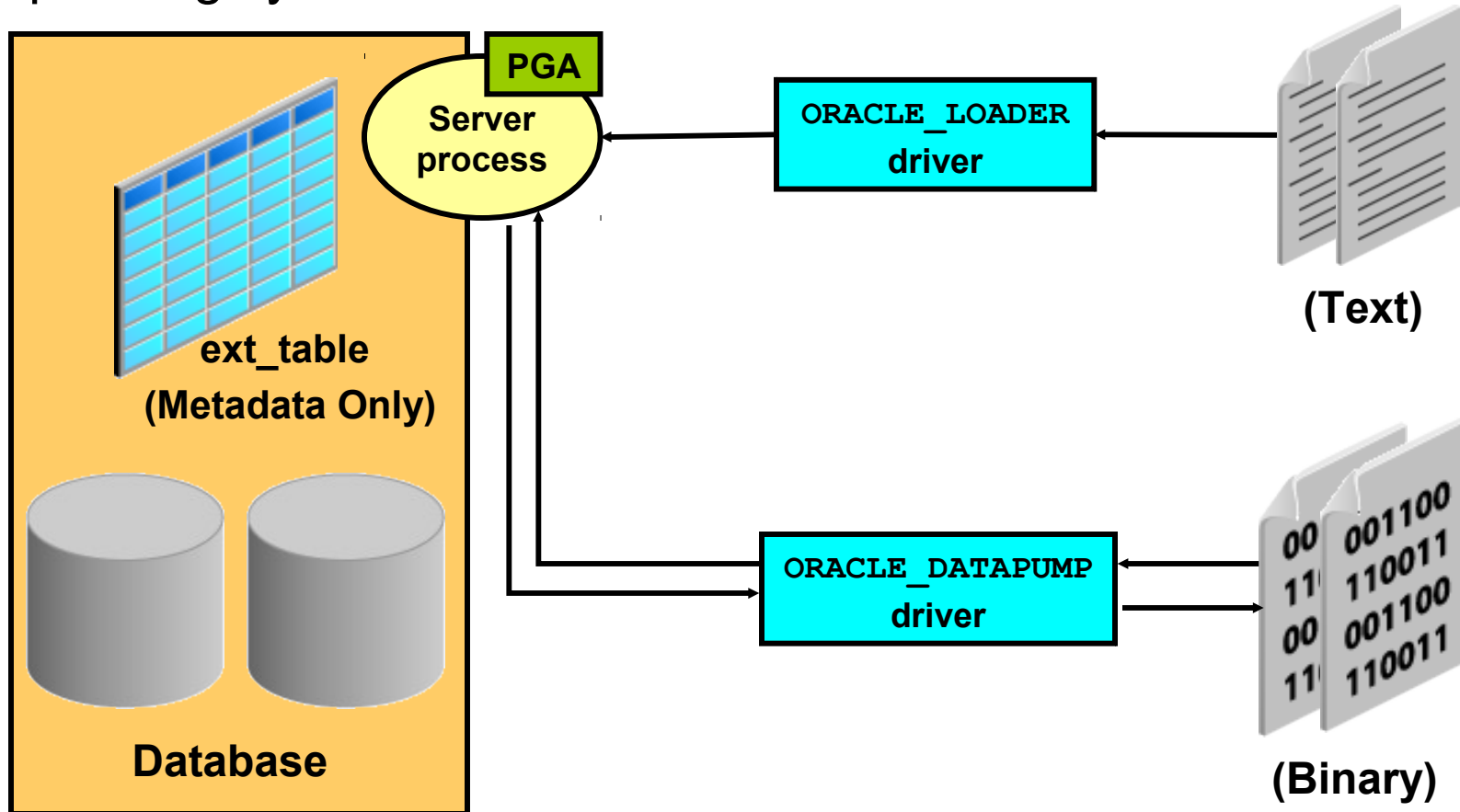
Does not load into clusters

Prevents other users from making changes to tables during load operation

Merges new index entries at the end of the load

External Tables

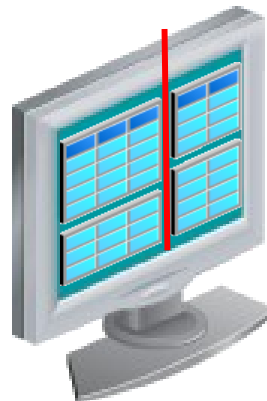
External tables are read-only tables stored as files on the operating system outside of the Oracle database.



External Table Benefits

- Data can be used directly from the external file or loaded into another database.
- External data can be queried and joined directly in parallel with tables residing in the database, without requiring it to be loaded first.
- The results of a complex query can be unloaded to an external file.
- You can combine generated files from different sources for loading purposes.

From Oracle Database



From external file

Defining an External Tables with ORACLE_LOADER

```
CREATE TABLE extab_employees
      (employee_id      NUMBER(4) ,
       first_name       VARCHAR2(20) ,
       last_name        VARCHAR2(25) ,
       hire_date        DATE)
ORGANIZATION EXTERNAL
  ( TYPE ORACLE_LOADER DEFAULT DIRECTORY extab_dat_dir
    ACCESS PARAMETERS
      ( records delimited by newline
        badfile extab_bad_dir:'empxt%a_%p.bad'
        logfile extab_log_dir:'empxt%a_%p.log'
        fields terminated by ','
        missing field values are null
      ( employee_id, first_name, last_name,
        hire_date char date_format date mask "dd-mon-yyyy"))
    LOCATION ('empxt1.dat', 'empxt2.dat') )
PARALLEL REJECT LIMIT UNLIMITED;
```

External Table Population with ORACLE_DATAPUMP

```
CREATE TABLE ext_emp_query_results
  (first_name, last_name, department_name)
ORGANIZATION EXTERNAL
  (
    TYPE ORACLE_DATAPUMP
    DEFAULT DIRECTORY ext_dir
    LOCATION ('emp1.exp', 'emp2.exp', 'emp3.exp')
  )
PARALLEL
AS
SELECT e.first_name, e.last_name, d.department_name
FROM   employees e, departments d
WHERE  e.department_id = d.department_id AND
       d.department_name in
           ('Marketing', 'Purchasing');
```

Using External Tables

- Querying and external table:

```
SQL> SELECT * FROM extab employees;
```

- Querying and joining an external table with an internal table

```
SQL> SELECT e.employee_id, e.first_name, e.last_name,  
           d.department_name FROM  
departments d, extab_employees e  
WHERE d.department id = e.department id;
```

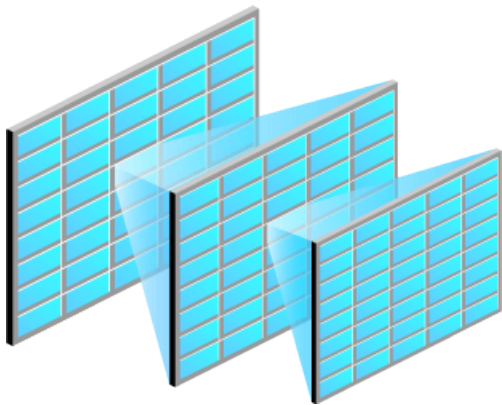
- Appending data to an internal table from an external table

```
SQL> INSERT /*+ APPEND */ INTO hr.employees SELECT * FROM  
extab employees;
```

Data Dictionary

View information about external tables in:

- [DBA | ALL | USER]_EXTERNAL_TABLES
- [DBA | ALL | USER]_EXTERNAL_LOCATIONS
- [DBA | ALL | USER]_TABLES
- [DBA | ALL | USER]_TAB_COLUMNS
- [DBA | ALL]_DIRECTORIES



Quiz

Like other database objects, Directory objects are owned by the user that creates them unless another schema is specified during creation.

1. True
2. False

Quiz

An index can be created on an external table.

1. True
2. False

Summary

In this lesson, you should have learned how to:

- Describe ways to move data
- Create and use directory objects
- Use SQL*Loader to load data from a non-Oracle database (or user files)
- Use external tables to move data via platform-independent files
- Explain the general architecture of Oracle Data Pump
- Use Data Pump Export and Import to move data between Oracle databases

Practice 17 Overview:

Moving Data

This practice covers the following topics:

- Using the Data Pump Export Wizard to select database objects to be exported
- Monitoring a Data Pump Export job
- Using the Data Pump Import Wizard to import tables to your database
- Using the Load Data Wizard to load data into your database
- Loading data by using the command line