

ORACLE®

ORACLE®

Oracle Data Pump Internals

Dean Gagne
Consulting Member of
Technical Staff

DOAG

Program Agenda

- How Oracle Data Pump uses the Master Table
- How to Debug Oracle Data Pump jobs
- Different uses of the Oracle Data Pump interactive command line interface
- How Oracle Data Pump schedules parallelism

How Oracle Data Pump uses the Master Table

Oracle Data Pump Master Table

- Oracle table that is used to store information about the Data Pump job.
 - Export/import job parameters
 - Current status
 - Object information (an index into the dump files)
- Can access the master table in SQLPLUS
- Not deleted if job is stopped or fatal error
- Deleted after job is completed or killed

Master Table Contents

Some Interesting columns for object rows

- Process_order (+ numbers)
- Object_type
- Object_schema
- Object_name
- Processing_state
 - ‘E’ = Estimated
 - ‘R’ = Retrieved (exported)
 - ‘W’ = Written (imported)
 - ‘X’ = Removed (exclude=...)
- Processing_status
 - ‘C’ = Current
 - ‘F’ = Failure

Master Table Contents (cont.)

Some Interesting columns for Data Pump job rows

- Process_order (- numbers)
- Object_schema
- Object_name
- Name
- Value_t
- Value_n
- Value_b

Interesting Process Order Numbers

- Positive process orders describe objects that have been exported.
- Negative process orders describe the Data Pump job
 - -1/-2 Job state row – contains job status
 - -5/-6 completion rows – status for each object type
 - -41/-42 – worker status rows
 - -51/-52 – data filter rows
 - -53/-54 – metadata filter rows
 - -57/-58 – metadata transform rows
 - -59/-60 – job parameter rows

How to Debug Oracle Data Pump Jobs

Example of Positive Rows

What object types are left to import

```
SQL> select unique object_type
      from system.sys_import_full_01
      where process_order > 0 AND
            processing_state = 'R' AND
            processing_status = 'C';
```

OBJECT_TYPE

PROCEDURE

ALTER_PROCEDURE

VIEW

Example of Positive Rows (cont)

Metrics on Exported Data

```
SQL> select sum(dump_orig_length), processing_state
       from "SYSTEM"."SYS_IMPORT_FULL_01"
       where process_order > 0 and duplicate = 0 and
              object_type = 'TABLE_DATA'
       group by processing_state;
```

```
SUM(DUMP_ORIG_LENGTH) P
-----
2525400 R      -- exported
2324944 E      -- estimated
```

Example of Positive Rows (cont)

Metrics on Imported Data

```
SQL> select sum(dump_orig_length), processing_state
       from "SYSTEM"."SYS_IMPORT_FULL_01"
       where process_order > 0 and duplicate = 0 and
             object_type = 'TABLE_DATA'
       group by processing_state;
```

```
SUM(DUMP_ORIG_LENGTH) P
```

```
-----
```

13408128	W	-- already imported
2525400	R	-- to be imported
24944	X	-- excluded

What was my export command

Example of a negative process order row

```
> impdp system/manager directory=dpump_dir  
  dumpfile=mydmp.dmp master_only=y
```

```
SQL> Select name, value_t  
  from SYSTEM.EXPORT_JOB_1  
  where process_order = -59 and  
         name = 'CLIENT_COMMAND';
```

CLIENT_COMMAND

```
system/***** tables=scott.emp directory=dpump_dir  
  dumpfile=ss.dmp reuse_dumpfiles=y
```

Parameters to Consider for Debug

- TRACE
 - Requires privileged user
 - *dw*.trc = worker trace file
 - *dm*.trc = mcp trace file
 - See MOS Note:
**Export/Import DataPump Parameter TRACE
How to Diagnose Oracle Data Pump [ID 286496.1]**
- METRICS =[YES|NO]
 - Indicates whether additional information about the job should be reported to the Data Pump log file.
 - Example:

Processing object type

DATABASE_EXPORT/SCHEMA/ROLE_GRANT

Completed 8943 ROLE_GRANT objects in 134 seconds

Known Performance Issues

- Statistics
 - Very Very slow in 10.1
 - Very Slow since 10.2
 - Exclude=statistics
 - Import them after import is complete
 - Regenerate them after import is complete
- Importing partitions into existing tables with parallel
 - Each partition/subpartition needs table lock
 - Let Data Pump create table and load data in the same job

Different Uses of the Oracle Data Pump Command Line Interfaces

Helpful Interactive Commands

Type ^c when job is running

Import> STATUS

- Job Status
 - Operation, Mode, State, Percent Done, Current Parallelism
- Worker Status
 - Object_schema, object_name, object_type, worker parallelism

Import> PARALLEL=N

- Changes parallelism for current job.
 - Increase is almost immediate
 - Decrease after current assignment is complete

Restart Options

Impdp user/password attach=user.imp_job_1 **trace=xxxxxx**

- Add tracing
 - See MOS Note: **Export/Import DataPump Parameter TRACE - How to Diagnose Oracle Data Pump [ID 286496.1]**

Import>

- Start_job=skip_current
 - Skip in_progress objects for each worker
 - Data – one object per worker
 - Metadata – up to 80 objects per worker

How Oracle Data Pump Determines Table Parallelism

Data Pump TABLE_DATA Work Unit

- Subpartition for subpartitioned tables
- Partition for partitioned tables
- Table for non partitioned tables

Export Case 1

Parallel 6 with lots of data – What processes are running?

- Data consists of 2 large subpartitions and 1 small subpartition
- Master Control Process - does not apply to the parallel 6 value
- Worker1 – Unload metadata – 1 parallel process
- Worker2 – Unload 1 large subpartition
 - user1:tab1:subpart1 - 2 degrees of parallelism
 - Worker2_PQ1 – Unload data – 1 parallel process
 - Worker2_PQ2 – Unload data – 1 parallel process
- Worker3 – Unload small subpartition
 - user1:tab1:subpart2 - 1 degree of parallelism
- Worker4 – Unload 1 large subpartition
 - user1:tab1:subpart3 - 2 degrees of parallelism
 - Worker4_PQ1 – Unload data – 1 parallel process
 - Worker4_PQ2 – Unload data – 1 parallel process

Export Case 1 Summary

Parallel 6 with lots of data

- Sum of the processes doing work
 - MCP - does not count 0
 - Worker 1 unloading metadata 1
 - Worker 2 query coordinator – does not count 0
 - Worker 2 has 2 pq slaves 2
 - Worker 3 unloading small subpartition 1
 - Worker 4 query coordinator – does not count 0
 - Worker 4 has 2 pq slaves 2
- 6 processes working to export job

Export Case 2

Parallel 6 with very little data

- Data consists of 1 small table
- MCP – does not apply to the parallel 6 value
- Worker1 – Unloading metadata – 1 parallel process
- Worker2 – Unload small subpartition

Summary:

- Sum of the processes doing work
 - MCP - does not count 0
 - Worker 1 unloading metadata 1
 - Worker 2 unloading small table 1
- Only 2 workers will be seen even though parallel 6 was specified.

Import Parallelism

- Data parallelism
 - Same algorithm as export
 - No worker processing metadata
 - All degrees of parallelism available
- Metadata parallelism
 - Package bodies **created** in parallel
 - Indexes **built** using parallelism

Metadata Import Parallelism

- Package bodies
 - Divided across workers based on package body size
- Indexes **created** serially **built** using pq slaves
 - Original create index statement
CREATE INDEX "HR"."EMP_ID_PK" ... **PARALLEL 3**;
 - Data Pump create index statement from parallel 22 import job
CREATE INDEX "HR"."EMP_ID_PK" ... **PARALLEL 22**;
 - Data Pump fixup statement
ALTER INDEX "HR"."EMP_ID_PK" **PARALLEL 3**;

Hardware and Software

ORACLE®

Engineered to Work Together

ORACLE®