What's New with Oracle Data Pump in Oracle Database 12c

Mike Dietrich
Senior Principal Technologist
Database Upgrades
Oracle Corporation

HARDWARE AND SOFTWARE ENGINEERED TO WORK TOGETHER

Updated: 4-APR-2014
What's New with Oracle Data Pump in Oracle 12c

- Introduction
- Full Transportable Export
- Views-as-Tables
- Timestamps
- Compression
- Transform
- Wrap Up
$> \texttt{whoami}

Mike Dietrich

6 years
RDBMS Core & Mission Critical Support

6 years
Technology Presales for DataGuard, Upgrades

6 years
ST Upgrade Development Team

50% Reference Projects

50% Workshops Worldwide

+ x% Development Work
Reference Involvement

Customer Success
Your success is our aim!

Non-binding Reference Proposal
for

Oracle Database Upgrade Development: Customer Reference Program

Company Name and Logo

You agree that your company name (where necessary including your logo) will be called "customer" or "reference customer" for relevant products and projects. Typically we will then use your company name during internal and external presentations or events.

Agree: [ ] Yes [ ] No
Comments:

Customer Quote

The quote is a brief statement about how your company has achieved a technological head start or has benefited economically from using Oracle products. Example: "Since we have been using Oracle iPaaS our acquisition costs have fallen by around 20%" - John Doe, CTO, IT Company Inc. With your agreement we will use your quote, for instance, in product brochures.

Agree: [ ] Yes [ ] No
Comments:
Reference Involvement: Results

“The new parallel upgrade script promises to drastically reduce downtime due to planned maintenance. We saw a 37% improvement over the previous upgrade process in our environment.”

Harald Stefan
Lutter Datenbanken
Payback GmbH

>16000 Downloads since June 2013
Slides and Dates?

- Download slides from:
  - blogs.oracle.com/UPGRADE
Data Pump Overview

- The “new” faster export-import
  - Available starting with Oracle 10.1
  - Powerful concept:
  - Master Note for Data Pump: [MOS Note:1264715.1](https://docs.oracle.com/cd/E191030_01/mnt.111/e191030_notes1_1264715.html)
  - For Compatibility and version changes: [MOS Note:553337.1](https://docs.oracle.com/cd/E191030_01/mnt.111/e191030_notes1_553337.html)
Data Pump Best Practices

- **For full exports:**
  - Role `EXP_FULL_DATABASE` is required

- **For export consistency use:**
  - `FLASHBACK_TIME=SYSTIMESTAMP`
    - This will increase UNDO requirements for the duration of the export

- **Always set parameters:**
  - `EXCLUDE=STATISTICS`
  - `METRICS=YES`
Data Pump Best Practices

- Speed up Data Pump:
  - Let Data Pump create the objects
  - `PARALLEL=n`
    - Typically $n = 2 \times \text{number of CPU cores}$
  - `EXCLUDE=STATISTICS` on export
  - `EXCLUDE=INDEXES` on import
    1. Initial `impdp` with `EXCLUDE=INDEXES`
    2. Second `impdp` with `INCLUDE=INDEXES` `SQLFILE=indexes.sql`
    3. Split `indexes.sql` into multiple SQL files and run in multiple sessions
      - Set `COMMIT_WAIT=NOWAIT` and `COMMIT_LOGGING=BATCH` during full imports
Data Pump Best Practices

- Direct import via database link
  - Parameter: `NETWORK_LINK`
    - Run only `impdp` on the target system - no `expdp` necessary
    - No dump file written, no disk I/O, no file transfer needed

- Restrictions of database links apply:
  - Does not work with `LONG/LONG RAW` and certain object types

- Performance: Depends on network bandwidth and target's CPUs

---

$ impdp ... NETWORK_LINK=dblink
Real World Case:
Kaiser Permanente, Medicare (USA)

- `impdp on NETWORK_LINK with`  
  8 vs 16 CPU cores
  
  - 10Gbit connection leveraged up to 8 Gbit
  - 1 TB table copied in ~15 min \(\Rightarrow\) 4 TB/hour

- Network bandwidth and CPU bound
What's New with Oracle Data Pump in Oracle 12c

- Introduction
- Full Transportable Export
- Views-as-Tables
- Timestamps
- Compression
- Transform
- Wrap Up
Full Transportable Export

- Combining:
  - Transportable Tablespaces
  - Data Pump for manual steps
  - Optional: Incremental Backups
- Source: Oracle 11.2.0.3 or newer
- Target: Oracle 12.1.0.1 or newer
- Works with or without Multitenant

Copyright © 2014, Oracle and/or its affiliates. All rights reserved. | Working with Oracle Multitenant in Oracle Database 12c
Traditional Transportable Tablespaces

Rebuild meta information
(views, synonyms, trigger, roles etc)

SOURCE Database 10.2.0.4
- USERS
  - SYSTEM
  - SYSAUX
  - UNDO
  - TEMP
- APP
- HUGO

DESTINATION Database 11.2.0.3
- SYSTEM
- SYSAUX
- UNDO
- TEMP
- APP
- HUGO

Copyright © 2014, Oracle and/or its affiliates. All rights reserved.
Working with Oracle Multitenant in Oracle Database 12c
Full Transportable Export/Import with Copies
Full Transportable Export/Import with Backups

SOURCE Database 11.2.0.3

DESTINATION Database 12.1.0.1

Convert and apply backups

Data Pump
Full Transportable Export/Import into Multitenant

- Create a fresh database/PDB
- Create database link to source
- Tablespaces read-only – *downtime!*
- Copy datafiles to destination
- **Run impdp on NETWORK_LINK**

```sql
impdp oow/passwd@PDB1
  NETWORK_LINK=DB1 VERSION=12 FULL=Y
  TRANSPORTABLE=ALWAYS METRICS=Y
  LOGFILE=oow_dir:src112fullimp.log
  TRANSPORT_DATAFILE='/oradata/ts1.dbf' ...
```
Step by Step: Full Transportable Export/Import

Incremental Backups

1. Enable archive logging in source:
   ALTER DATABASE ARCHIVELOG;

2. Enable block change tracking in source:
   ALTER DATABASE ENABLE BLOCK CHANGE TRACKING USING FILE '<name>' REUSE;

3. Recovery area should be accessible by target instance

4. Simple RMAN script:
   BACKUP INCREMENTAL LEVEL 1 FOR RECOVER OF COPY WITH TAG 'xyz' DATABASE;
   RECOVER COPY OF DATABASE WITH TAG 'xyz';
   - First execution creates level 0 copy of all datafiles
   - Subsequent invocations backup and apply only changed blocks to the datafiles since previous invocation
Step by Step: Full Transportable Export/Import

1 Incremental Backups

2 Setup steps in the new database/PDB

3 – Create a directory object
   CREATE DIRECTORY xyz_dir AS '/reco/data';
   GRANT READ, WRITE ON DIRECTORY xyz_dir TO mike;

4 – Create a database link back into source instance
   CREATE PUBLIC DATABASE LINK v112 USING 'v112';
   • User who runs impdp on target must exist on source with DATAPUMP_EXP_FULL_DATABASE role
Step by Step: Full Transportable Export/Import

1 Incremental Backups
2 Setup steps in the new database/PDB
3 Transport Phase

- Set all data tablespaces read-only on source
  
  ALTERTABLESPACE users READ ONLY;

  >>>>>> DOWNTIME <<<<<

- Final execution of the RMAN backup and apply scripts
  
  BACKUP INCREMENTAL LEVEL 1 FOR RECOVER OF COPY WITH TAG 'xyz' DATABASE;
  RECOVER COPY OF DATABASE WITH TAG 'xyz';
Step by Step: Full Transportable Export/Import

1 Incremental Backups
2 Setup steps in the new database/PDB
3 Transport Phase
4 Data Pump at Work

- impdp mike/<passwd>@V112 NETWORK_LINK=v112
  VERSION=12 FULL=Y TRANSPORTABLE=ALWAYS
  METRICS=Y LOGFILE=xyz_dir:v112fullimp.log
  TRANSPORT_DATAFILES='/reco/data/user1.dbf'
Step by Step: Full Transportable Export/Import

1. Incremental Backups
2. Setup steps in the new database/PDB
3. Transport Phase
4. Data Pump at Work
5. Clean Up
   - Check the logfile
   - Target tablespaces ⇒ read-write
   - Source tablespaces ⇒ read-only
What's New with Oracle Data Pump in Oracle 12c

- Introduction
- Full Transportable Export
- Views-as-Tables
- Timestamps
- Compression
- Transform
- Wrap Up
Exporting Views as Tables

```
expdp system/mgr
views_as_tables=scott.view1 ...
```

```
impdp system/mgr
remap_table=view1:scott.table1 ...
```
Exporting Views as Tables

- Parameter:
  - VIEWS_AS_TABLES=[schema_name.]view_name[:table_name],

- One or more *views* are to be exported *as tables*
  - Exports a table with the same columns as view
  - Row data is fetched from the view
  - Dependent objects also exported, such as grants and constraints

- **TABLE_NAME**: Template table serves as source of metadata
  - Required if the database is read-only
Exporting Views as Tables

Example: export

SQL> CREATE VIEW hr.my_view (first, last, dept) AS
      select e.first_name, e.last_name, d.department_name
      from employees e, departments d
      where e.department_id = d.department_id;

>$ expdp system/manager views_as_tables=hr.my_view ...
Processing object type TABLE_EXPORT/VIEWS_AS.Tables/TABLE ...
exeported "HR"."MY_VIEW" 8.570 KB 106 rows
Exporting Views as Tables

Example: import

```bash
$ impdp system/manager remap_table=my_view:my_table ...
```

Processing object type

```
TABLE_EXPORT/VIEWS_AS_TABLES/TABLE_DATA
```

. . imported "HR"."MY_TABLE" 8.570 KB 106 rows

- Without the `remap_table` parameter, Data Pump creates a table called `my_view`

- `remap_table:source_name:target_name` remaps the name and creates a table with the target name
What's New with Oracle Data Pump in Oracle 12c

- Introduction
- Full Transportable Export
- Views-as-Tables
- Timestamps
- Compression
- Transform
- Wrap Up
LOGTIME Parameter

- Include timestamps in the log file

\[ \text{LOGTIME} = [ \text{NONE} \mid \text{STATUS} \mid \text{LOGFILE} \mid \text{ALL} ] \]

- **NONE**: No timestamps on status or log file messages
- **STATUS**: On status messages only
- **LOGFILE**: On log file messages only
- **ALL**: On both status and log file messages
LOGTIME Parameter

- Without vs With `LOGTIME=ALL`
What's New with Oracle Data Pump in Oracle 12c

- Introduction
- Full Transportable Export
- Views-as-Tables
- Timestamps
- Compression
- Transform
- Wrap Up
Enhanced Compression Algorithm

- **COMPRESSION_ALGORITHM**
  - Defines the compression algorithm when compressing dump files
  - Performance:
    - Compression ratio
    - CPU usage
Enhanced Compression Algorithm

- **COMPRESSION ALGORITHM**
  - **BASIC**: The same algorithm used in previous versions. Good compression, without severely impacting on performance
  - **LOW**: For use when reduced CPU utilization is a priority over compression ratio
  - **MEDIUM**: Recommended option. Similar characteristics to BASIC, but uses a different algorithm
  - **HIGH**: Maximum available compression, but more CPU intensive

```
$ expdp scott/tiger tables=emp directory=mydir
dumpfile=emp.dmp logfile=expdp_emp.log
compression=all compression_algorithm=medium
```
## Enhanced Compression Algorithm

- **Customer evaluation**
  - **BASIC** at 3.5 TB/hour
  - **MEDIUM** at 7.0 TB/hour

### Basic Configuration

<table>
<thead>
<tr>
<th>Name</th>
<th>Disks</th>
<th>AvgBusy</th>
<th>Read/Write-KB/s</th>
<th>TotalMB/s</th>
<th>xfers/s</th>
<th>BlockSizeKB</th>
</tr>
</thead>
<tbody>
<tr>
<td>slot02</td>
<td>6</td>
<td>9.3%</td>
<td>123120.4</td>
<td>0</td>
<td>0.0</td>
<td>120.2</td>
</tr>
<tr>
<td>slot03</td>
<td>6</td>
<td>6.7%</td>
<td>103354.8</td>
<td>0</td>
<td>0.0</td>
<td>100.9</td>
</tr>
<tr>
<td>slot05</td>
<td>6</td>
<td>9.0%</td>
<td>130420.9</td>
<td>7</td>
<td>7.0</td>
<td>127.4</td>
</tr>
<tr>
<td>slot06</td>
<td>6</td>
<td>10.5%</td>
<td>158841.9</td>
<td>7</td>
<td>7.0</td>
<td>155.3</td>
</tr>
<tr>
<td>slot08</td>
<td>6</td>
<td>8.4%</td>
<td>130385.3</td>
<td>0</td>
<td>0.0</td>
<td>127.8</td>
</tr>
<tr>
<td>slot09</td>
<td>6</td>
<td>10.1%</td>
<td>136525.9</td>
<td>0</td>
<td>0.0</td>
<td>133.3</td>
</tr>
<tr>
<td>slot10</td>
<td>6</td>
<td>6.6%</td>
<td>143354.1</td>
<td>0</td>
<td>0.0</td>
<td>137.1</td>
</tr>
<tr>
<td>slot11</td>
<td>6</td>
<td>6.8%</td>
<td>112600.0</td>
<td>2</td>
<td>2.0</td>
<td>110.0</td>
</tr>
<tr>
<td>Groups</td>
<td>3 TOTALS</td>
<td>40</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Medium Configuration

<table>
<thead>
<tr>
<th>Name</th>
<th>Disks</th>
<th>AvgBusy</th>
<th>Read/Write-KB/s</th>
<th>TotalMB/s</th>
<th>xfers/s</th>
<th>BlockSizeKB</th>
</tr>
</thead>
<tbody>
<tr>
<td>slot02</td>
<td>6</td>
<td>14.5%</td>
<td>255770.4</td>
<td>0</td>
<td>0.0</td>
<td>249.8</td>
</tr>
<tr>
<td>slot03</td>
<td>6</td>
<td>16.0%</td>
<td>273037.4</td>
<td>1</td>
<td>1.5</td>
<td>266.6</td>
</tr>
<tr>
<td>slot05</td>
<td>6</td>
<td>15.4%</td>
<td>264851.1</td>
<td>1</td>
<td>1.5</td>
<td>258.7</td>
</tr>
<tr>
<td>slot06</td>
<td>6</td>
<td>13.2%</td>
<td>221160.7</td>
<td>1</td>
<td>2.5</td>
<td>217.4</td>
</tr>
<tr>
<td>slot08</td>
<td>6</td>
<td>15.0%</td>
<td>267156.6</td>
<td>1</td>
<td>1.5</td>
<td>260.9</td>
</tr>
<tr>
<td>slot09</td>
<td>6</td>
<td>14.0%</td>
<td>263140.9</td>
<td>6</td>
<td>6.5</td>
<td>257.0</td>
</tr>
<tr>
<td>slot10</td>
<td>6</td>
<td>14.6%</td>
<td>259603.7</td>
<td>2</td>
<td>2.0</td>
<td>253.5</td>
</tr>
<tr>
<td>slot11</td>
<td>6</td>
<td>14.9%</td>
<td>258113.0</td>
<td>0</td>
<td>0.0</td>
<td>252.1</td>
</tr>
<tr>
<td>Groups</td>
<td>3 TOTALS</td>
<td>48</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

2X enhancement
What's New with Oracle Data Pump in Oracle 12c

- Introduction
- Full Transportable Export
- Views-as-Tables
- Timestamps
- Compression
- Transform
- Wrap Up
Change Table Compression at Import Time

- `TRANSFORM` option to enable Advanced/HCC Compression
  - Example:
    - `TRANSFORM=TABLE_COMPRESSION:"compress for query high"`
Change Table Compression at Import Time

- **But:** Granularity only on the entire import
  - **Workarounds:**
    - Precreate objects
      - **Downside:** Will slow down import!!!
    - *or:*
    - Precreate the tablespace with COMPRESS option
      - `create tablespace ARCHIGH datafile 'archigh.ora' size 100G default compress for archive high;`
    - Then run Data Pump with `TRANSFORM=TABLE_COMPRESSION:N`
      - This will drop all embedded compression attributes associated with the tables
      - Now tablespace compression option will be used for all newly created tables
No Logging Options for Import

- TRANSFORM=DISABLE_ARCHIVE_LOGGING:Y
  - Disable redo logging when loading tables and/or creating indexes
  - Applies to both TABLES and INDEXES
  - Logging attributes restored to original settings after data is loaded
No Logging Options for Import

- Notes:
  - Redo logging for other operations still happens
  - Valid for both file mode imports and network mode imports
  - FORCE LOGGING mode?
    DISABLE_ARCHIVE_LOGGING option will not disable any logging
No Logging Options for Import

- **TRANSFORM=LOB_STORAGE:SECURE_FILE**
  - **Options:** [SECUREFILE | BASICFILE | DEFAULT | NO_CHANGE]
  - Transforms each LOB Segment into a SecureFile LOB
What's New with Oracle Data Pump in Oracle 12c

- Introduction
- Full Transportable Export
- Views-as-Tables
- Timestamps
- Compression
- Transform
- Wrap Up
Real World Checkpoint

- **Payback GmbH**
  - Belongs to Loyalty Partner GmbH which belongs to *American Express*
  - HQ in Munich, Germany 🇩🇪
  - Develops and operates professional customer loyalty programs based on customized IT solutions
    - Provider for Payback
    - Active in Germany, Poland, India and Mexico
Real World Checkpoint

- Migrate 7TB / 1.5TB from HP-UX to Exadata V1
  - Cross platform, cross Endianness, cross version
    - Oracle 9.2.0.7 on HP-UX ⇒ Oracle 11.1.0.7 on OL
  - 4 months planning and migration phase
    - August to November 2009
  - Proposed go-live date
    - 15-NOV-2009
Real World Checkpoint

- Move everything in less than 24 hrs
- Network bottleneck
  - Customer installed InfiniBand hardware into HP box
  \[ \Rightarrow \sim 3\text{GB/sec throughput!} \]
Real World Checkpoint

- Setup:

  PROD
  
  Restore + Upgrade
  
  HP-UX PA-RISC
  
  Prod Load
  
  SWING
  
  HP-UX PA-RISC
  
  IB Hardware
  
  OL 64bit
Real World Checkpoint

- Test migrations:

Customer
Project
Constraints
Preparation
Upgrade
Success?
Remarks

HP-UX PA-RISC
Prod Load

PROD

SWING

HP-UX PA-RISC
IB Hardware

Data Pump on NETWORK_LINK

INSERT APPEND on database links for tables >100 GB

OL 64bit
Real World Checkpoint

- Parallel live loads: Performance tests

Customer
Project
Constraints
Preparation
Upgrade
Success?
Remarks

PROD
HP-UX PA-RISC
Prod Load

SWING
HP-UX PA-RISC
IB Hardware

OL 64bit
Prod Load

Redirect the production load by apps servers
Real World Checkpoint

- Final test became LIVE migration
Real World Checkpoint

- Live? And alive?
  - Yes! Go-live in early November 2009
    - Two weeks earlier than proposed
  - Total upgrade and migration time: ~20 hours
    - ~8 hours: Restore and recovery
    - ~1 hour: Database upgrade to Oracle 11.1.0.7
    - ~10 hours: Data migration to Exadata V1
    - ~1 hour: Smoke testing and final verification
  - Dramatic performance improvements
    - Job runtimes decreased by 80%
    - User complaints about too fast performance … really!!
Real World Checkpoint

- Not a single piece of SQL got changed!!!
  - Most critical job: runtime from 30 hrs to < 2hrs
Further information?

- White Paper: Full Transportable Export/Import

- OTN Page:
  http://otn.oracle.com/goto/datapump

- Documentation:
  Oracle Database Utilities Guide
Resources

- **Download slides as PDF from:**

- **Full Day Upgrade Workshop in Oslo:**
  - **Registration:**
    - [http://eventreg.oracle.com/profile/web/index.cfm?PKWebID=0x705481765&source=EMEAFM13033369MPP059](http://eventreg.oracle.com/profile/web/index.cfm?PKWebID=0x705481765&source=EMEAFM13033369MPP059)
  - **May 20, 2014**
Hardware and Software
Engineered to Work Together