Queries to Monitor Datapump Jobs

[Nitish Srivastava](https://databaseinternalmechanism.com/author/nitishanandsrivastava/) / [September 13, 2016](https://databaseinternalmechanism.com/2016/09/13/how-to-monitor-datapump-jobs/)

This post is to cover the queries that could be used to monitor the running Datapump Jobs.

Without wasting anytime just posting the methods/queries which can be used to monitor the details of the datapump jobs:-

1) Using the datapump client (expdp & impdp) STATUS command:-

When the export or import job is running press +C keys to get to the respective datapump client prompt OR you can use another session of datapump client and using the ATTACH clause attach to the running job and then issue the STATUS command:-

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20 | Export> status    Job: SYS\_EXPORT\_FULL\_01    Operation: EXPORT    Mode: FULL    State: EXECUTING    Bytes Processed: 0    Current Parallelism: 1    Job Error Count: 0    Dump File: /u01/app/oracle/dpump/admin.dmp      bytes written: 4,096    Worker 1 Status:    Process Name: DW00    State: EXECUTING    Object Schema: ADMIN    Object Name: TEST\_01    Object Type: DATABASE\_EXPORT/SCHEMA/PACKAGE\_BODIES/PACKAGE/PACKAGE\_BODY    Completed Objects: 78    Worker Parallelism: 1 |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48 | Import> status    Job: SYS\_IMPORT\_SCHEMA\_01    Operation: IMPORT    Mode: SCHEMA    State: EXECUTING    Bytes Processed: 2,788,707,576    Percent Done: 99    Current Parallelism: 6    Job Error Count: 0    Dump File: /apps/keplero/backup/ORA11G/dpump/cishd-34173\_%u.dmp    Dump File: /apps/keplero/backup/ORA11G/dpump/cishd-34173\_01.dmp    Dump File: /apps/keplero/backup/ORA11G/dpump/cishd-34173\_02.dmp    Dump File: /apps/keplero/backup/ORA11G/dpump/cishd-34173\_03.dmp    Dump File: /apps/keplero/backup/ORA11G/dpump/cishd-34173\_04.dmp    Dump File: /apps/keplero/backup/ORA11G/dpump/cishd-34173\_05.dmp    Dump File: /apps/keplero/backup/ORA11G/dpump/cishd-34173\_06.dmp    Worker 1 Status:    Process Name: DW00    State: EXECUTING    Object Schema: XTP\_AC    Object Name: SYS\_C0063284986    Object Type: SCHEMA\_EXPORT/TABLE/CONSTRAINT/CONSTRAINT    Completed Objects: 1,120    Worker Parallelism: 1    Worker 2 Status:    Process Name: DW01    State: WORK WAITING    Worker 3 Status:    Process Name: DW02    State: WORK WAITING    Worker 4 Status:    Process Name: DW03    State: WORK WAITING    Worker 5 Status:    Process Name: DW04    State: WORK WAITING    Worker 6 Status:    Process Name: DW05    State: WORK WAITING    Import> |

So from the output you can see the status of the Master Control Process and Worker process, read my post [Oracle datapump Architecture & Internals](https://databaseinternalmechanism.com/2016/09/09/oracle-datapump-expdp-impdp-internals/) for better understanding of Datapump Architecture.

2) Querying DBA\_DATAPUMP\_JOBS view:-

|  |  |
| --- | --- |
| 1 | select \* from dba\_datapump\_jobs; |

The STATE column of the above view would give you the status of the JOB to show whether EXPDP or IMPDP jobs are still running, or have terminated with either a success or failure status.

3) Querying V$SESSION\_LONGOPS & V$SESSION views:-

|  |  |
| --- | --- |
| 1  2  3  4  5 | SELECT b.username, a.sid, b.opname, b.target,              round(b.SOFAR\*100/b.TOTALWORK,0) || '%' as "%DONE", b.TIME\_REMAINING,              to\_char(b.start\_time,'YYYY/MM/DD HH24:MI:SS') start\_time       FROM v$session\_longops b, v$session a       WHERE a.sid = b.sid      ORDER BY 6; |

4) Querying V$SESSION\_LONGOPS & V$DATAPUMP\_JOB views:-

|  |  |
| --- | --- |
| 1  2  3  4 | SELECT sl.sid, sl.serial#, sl.sofar, sl.totalwork, dp.owner\_name, dp.state, dp.job\_mode       FROM v$session\_longops sl, v$datapump\_job dp       WHERE sl.opname = dp.job\_name       AND sl.sofar != sl.totalwork; |

5) Querying all the related views with a single query:-

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12 | select x.job\_name,b.state,b.job\_mode,b.degree  , x.owner\_name,z.sql\_text, p.message  , p.totalwork, p.sofar  , round((p.sofar/p.totalwork)\*100,2) done  , p.time\_remaining  from dba\_datapump\_jobs b  left join dba\_datapump\_sessions x on (x.job\_name = b.job\_name)  left join v$session y on (y.saddr = x.saddr)  left join v$sql z on (y.sql\_id = z.sql\_id)  left join v$session\_longops p ON (p.sql\_id = y.sql\_id)  WHERE y.module='Data Pump Worker'  AND p.time\_remaining > 0; |

6) Use the following procedure and replace the JOB\_OWNER & JOB\_NAME as per your env. which you fetch from import.log:-

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8 | ;;;  Import: Release 12.1.0.2.0 - Production on Thu Jun 29 00:29:09 2017    Copyright (c) 1982, 2014, Oracle and/or its affiliates.  All rights reserved.  ;;;  Connected to: Oracle Database 12c Enterprise Edition Release 12.1.0.2.0 - 64bit Production  With the Partitioning, OLAP, Advanced Analytics and Real Application Testing options  Master table "SYSTEM"."SYS\_IMPORT\_FULL\_04" successfully loaded/unloaded |

Here the JOB\_OWNER is SYSTEM and JOB\_NAME is SYS\_IMPORT\_FULL\_04.

And below is the procedure:-

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28 | SET SERVEROUTPUT ON  DECLARE    ind NUMBER;    h1 NUMBER;    percent\_done NUMBER;    job\_state VARCHAR2(30);    js ku$\_JobStatus;    ws ku$\_WorkerStatusList;    sts ku$\_Status;  BEGIN  h1 := DBMS\_DATAPUMP.attach('JOB\_NAME', 'JOB\_OWNER');  dbms\_datapump.get\_status(h1,             dbms\_datapump.ku$\_status\_job\_error +             dbms\_datapump.ku$\_status\_job\_status +             dbms\_datapump.ku$\_status\_wip, 0, job\_state, sts);  js := sts.job\_status;  ws := js.worker\_status\_list;        dbms\_output.put\_line('\*\*\* Job percent done = ' ||                             to\_char(js.percent\_done));        dbms\_output.put\_line('restarts - '||js.restart\_count);  ind := ws.first;    while ind is not null loop      dbms\_output.put\_line('rows completed - '||ws(ind).completed\_rows);      ind := ws.next(ind);    end loop;  DBMS\_DATAPUMP.detach(h1);  end;  / |

7) Also for any errors you can check the alert log and query the DBA\_RESUMABLE view.

|  |  |
| --- | --- |
| 1 | select name, sql\_text, error\_msg from dba\_resumable; |

That’s all what I can think of at the moment, would add the queries to this post if I find another view which can be used to get the information of the datapump jobs.