

## **(ADB) MV2ADB: move data to Autonomous Database in "one-click" (Doc ID 2463574.1)**

---

### **In this Document**

[Abstract](#)

[History](#)

[Details](#)

[MV2ADB: Migrate to Autonomous Database Cloud within "1-Click"](#)

[Loading data with autonomous database cloud](#)

[MV2ADB Requirements](#)

[MV2ADB features & operations](#)

[MV2ADB - Auto Operation](#)

[MV2ADB - Expdp Operation](#)

[MV2ADB - Impdp Operation](#)

[MV2ADB – createbucket Operation](#)

[MV2ADB – delbucket Operation](#)

[MV2ADB – listbucket Operation](#)

[MV2ADB – putdump Operation](#)

[MV2ADB – getdump Operation](#)

[MV2ADB – deldump Operation](#)

[MV2ADB – listdump Operation](#)

[MV2ADB – advisor Operation](#)

[MV2ADB – report Operation](#)

[MV2ADB – recomp Operation](#)

[MV2ADB – encpass Operation](#)

[MV2ADB Command Option](#)

[MV2ADB Configuration File](#)

[Configuration File Parameters Notes](#)

[MV2ADB Installation](#)

[MV2ADB De-installation](#)

[Managing MV2ADB Privileges and security with sudo](#)

[Allowing Root User Access Using SUDO](#)

[SUDO Example 1: Allow a User to Perform Any mv2adb Operation](#)

[SUDO Example 2: Allow a User to Perform Only Selected mv2adb Operations](#)

[SUDO Example 3: Allow a User to Perform Any mv2adb Operation without password request](#)

[MV2ADB Usage Example](#)

[MV2ADB: Pre-requisite Configuration](#)

[MV2ADB: troubleshooting](#)

[References](#)

---

### **APPLIES TO:**

---

## ABSTRACT

MV2ADB provides support for loading data in the Autonomous Database (ATP-D, ATP-S, ADW)

**NOTE:** In the images and/or the document content below, the user information and environment data used represents fictitious data from the Oracle sample or built-in schema(s), Public Documentation delivered with an Oracle database product or other training material. Any similarity to actual environments, actual persons, living or dead, is purely coincidental and not intended in any manner.

## HISTORY

Author: Ruggero Citton - RAC Pack, Cloud Innovation and Solution Engineering Team

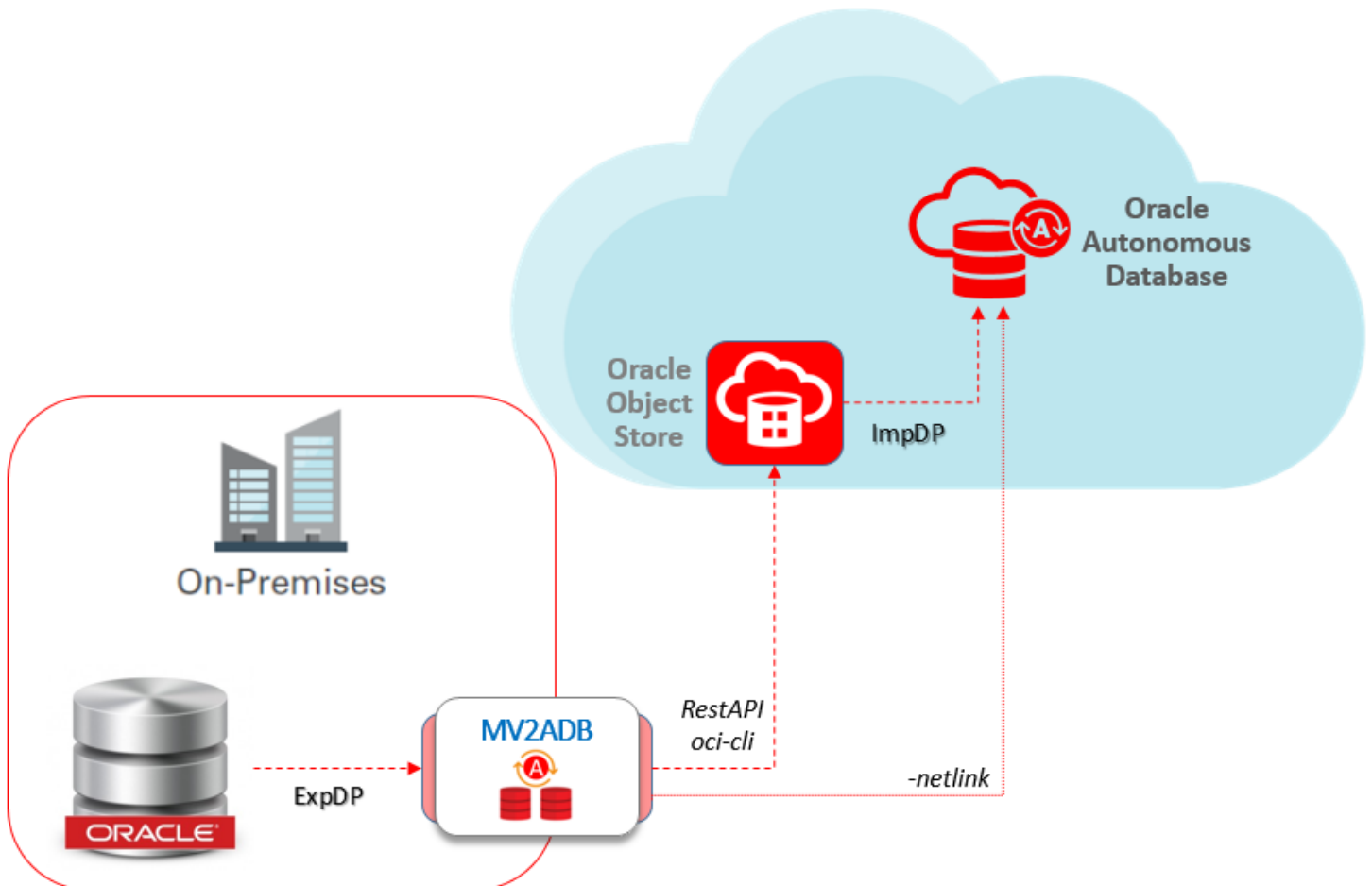
Create Date Sep-2018

Update Date Apr-2020

Latest "[mv2adb](#)" Version: 20200422 - \$Revision: 2.0.1.97 \$

## DETAILS

### MV2ADB: Migrate to Autonomous Database Cloud within "1-Click"



**Loading data with autonomous database cloud**

Traditionally transaction processing systems ingest data through routine transactions or DML operations; you can also bulk load data into Autonomous Transaction Processing or Autonomous Data Warehouse using Oracle Database tools, and Oracle or other 3rd party data integration tools.

In general you load data from files local to your client computer or from files stored in a cloud-based object store. For data loading from files in the cloud, Autonomous Transaction Processing and Autonomous Data Warehouse in general Autonomous Database Cloud provides a new PL/SQL package DBMS\_CLOUD/DBMS\_CREDENTIAL.

For the fastest data loading experience Oracle recommends uploading the source files to a cloud-based object store, such as Oracle Cloud Infrastructure Object Storage, before loading the data into your Autonomous Database Cloud. Oracle provides support for loading files that are located locally in your data center, but when using this method of data loading you should factor in the transmission speeds across the Internet which may be significantly slower.

Move to Autonomous Database ([MV2ADB](#)) is a new tool is permitting the load data and migration from "on premises" to Autonomous Database Cloud leveraging on Oracle Data Pump and within one command. Data Pump Import lets you import data from Data Pump files residing on the Oracle Cloud Infrastructure Object Storage. You can save your data to your Cloud Object Store and to load them to Autonomous Database Cloud using "[mv2adb](#)".

```
### -----
### Disclaimer:
###
### EXCEPT WHERE EXPRESSLY PROVIDED OTHERWISE, THE INFORMATION, SOFTWARE,
### PROVIDED ON AN \"AS IS\" AND \"AS AVAILABLE\" BASIS. ORACLE EXPRESSLY DISCLAIMS
### ALL WARRANTIES OF ANY KIND, WHETHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT
### LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR
### PURPOSE AND NON-INFRINGEMENT. ORACLE MAKES NO WARRANTY THAT: (A) THE RESULTS
### THAT MAY BE OBTAINED FROM THE USE OF THE SOFTWARE WILL BE ACCURATE OR
### RELIABLE; OR (B) THE INFORMATION, OR OTHER MATERIAL OBTAINED WILL MEET YOUR
### EXPECTATIONS. ANY CONTENT, MATERIALS, INFORMATION OR SOFTWARE DOWNLOADED OR
### OTHERWISE OBTAINED IS DONE AT YOUR OWN DISCRETION AND RISK. ORACLE SHALL HAVE
### NO RESPONSIBILITY FOR ANY DAMAGE TO YOUR COMPUTER SYSTEM OR LOSS OF DATA THAT
### RESULTS FROM THE DOWNLOAD OF ANY CONTENT, MATERIALS, INFORMATION OR SOFTWARE.
###
### ORACLE RESERVES THE RIGHT TO MAKE CHANGES OR UPDATES TO THE SOFTWARE AT ANY
### TIME WITHOUT NOTICE.
###
### Limitation of Liability:
###
### IN NO EVENT SHALL ORACLE BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
### SPECIAL OR CONSEQUENTIAL DAMAGES, OR DAMAGES FOR LOSS OF PROFITS, REVENUE,
### DATA OR USE, INCURRED BY YOU OR ANY THIRD PARTY, WHETHER IN AN ACTION IN
### CONTRACT OR TORT, ARISING FROM YOUR ACCESS TO, OR USE OF, THE SOFTWARE.
### -----
```

## **MV2ADB Requirements**

Following are the mv2adb requirements:

- HTTP connectivity between "on-premises" and Oracle Object Store (if "netlink" is not in use) to move expdp dump
- SQL\*Net connectivity between "on-premises" and Autonomous Database

be sure you can connect from "on-premises" your target Autonomous Database using Sql\*Plus before try MV2ADB

- The ADB wallet zip
- Latest instant client to support new impdp cloud command option
  - Basic Package - All files required to run OCI, OCCI, and JDBC-OCI applications - (Version 18.3.0.0.0 or above)
  - SQL\*Plus Package - The SQL\*Plus command line tool for SQL and PL/SQL queries - (Version 18.3.0.0.0 or above)
  - Tools Package - Includes Data Pump, SQL\*Loader and Workload Replay Client - (Version 18.3.0.0.0 or above)
- java executable in the path

- Perl release 5.10 or above
- perl-Data-Dumper

## ***MV2ADB features & operations***

Move to Autonomous Database is supporting multiple operations to manage the load/migration data life-cycle from "on premises" to Autonomous Database Cloud. Possible operations are:

- Auto operation:
  - **auto**
- ExpDP Operations:
  - expdp
- ImpDP Operations:
  - impdp
- Oracle Cloud Infrastructure Object Storage Bucket Operations:
  - createbucket
  - delbucket
  - listbucket
- Oracle Cloud Infrastructure Object Storage Object Operations:
  - deldump
  - getdump
  - listdump
  - putdump
- Database Schema Operations
  - advisor
  - report
  - recomp
- Encrypt password Operations:
  - encpas

MV2ADB is supporting Oracle Cloud Infrastructure APIs and/or Oracle oci-client. The [oci-client](#) must be installed, configured and available in the PATH

## ***MV2ADB - Auto Operation***

"Auto" operation is "one command operation" to perform the entire load data process into Autonomous Database Cloud (ADB):

1. expdp from source (schemas based)
2. upload dump over Oracle Object Store (if "--netlink" is not in use)
3. impdp into Autonomous Database Cloud

```
mv2adb auto {--conf <conf file path> [--netlink] [--norecomp] [--norecomp] [--nosudo] --
swift}} |
    {--ocic
      --ocibucket <bucket name> [--ocinamespace <namespace name>]
      [--ocicsize <size Mb> --ocicparallel <count>]} |
    {--ociregion <oci region>
      --ocinamespace <namespace name>
      --ociId <user ID>
      --ocibucket <bucket name>
      [--proxyHost <host> --proxyPort <port> [--proxyId <user ID> --proxyPass
<password>]]}
      --dbcs <Source DB connect string>
      --schemas <Database schemas to export> | --full y [--exclude <Database schemas
to exclude>]
      {--dumpname <expdp dump file name> --dumppath <expdp dump file path> }
```

```

|
|--netlink}
|--remap <comma separated tablespace remap as SOURCE_TBS:TARGET_TBS]
--ohome <Oracle Home>
--ichome <Instant Client Home>
--adbname <ADB database name>
--adbcfile <ADB credential zip file>
[--adbccores <ADB cores count>]
[--compression ALL | DATA_ONLY | METADATA_ONLY | NONE ]
[--encryption [--enctype AES128 | AES192 | AES256]]
[--goldengate]
[--norecomp]
[--swift]
[--nosudo]}

```

### Execution example (using a configuration file):

```

# ./mv2adb auto -conf conf/mv2atp.cfg
INFO: 2018-09-19 19:39:34: Please check the logfile '/opt/mv2adb/out/log/mv2adb_13040.log

INFO: 2018-09-19 19:39:34: Reading the configuration file 'conf/mv2atp.cfg'
INFO: 2018-09-19 19:39:34: Performing schema expdp for 'SCHEMA_1,SCHEMA_2' from source DB...
INFO: 2018-09-19 19:39:34: Step1 - ...getting ADB parallelism
INFO: 2018-09-19 19:39:39: Step2 - ...getting source DB version
INFO: 2018-09-19 19:39:39: Step3 - ...creating expdp directory 'MV2ATP_EXPDP_DIR'
INFO: 2018-09-19 19:39:39: Step4 - ...executing export datapump
SUCCESS: 2018-09-19 19:40:01: Expdp executed successfully

INFO: 2018-09-19 19:40:01: Performing '4' dump upload to Oracle Object Store...
INFO: 2018-09-19 19:40:01: ...loading '/tmp/expdp_01_01.dmp' into bucket 'BUCKET_NAME'
SUCCESS: 2018-09-19 19:40:02: ...file '/tmp/expdp_01_01.dmp' uploaded on 'BUCKET_NAME'
successfully
INFO: 2018-09-19 19:40:02: ...loading '/tmp/expdp_01_02.dmp' into bucket 'BUCKET_NAME'
SUCCESS: 2018-09-19 19:40:03: ...file '/tmp/expdp_01_02.dmp' uploaded on 'BUCKET_NAME'
successfully
INFO: 2018-09-19 19:40:03: ...loading '/tmp/expdp_01_03.dmp' into bucket 'BUCKET_NAME'
SUCCESS: 2018-09-19 19:40:08: ...file '/tmp/expdp_01_03.dmp' uploaded on 'BUCKET_NAME'
successfully
INFO: 2018-09-19 19:40:08: ...loading '/tmp/expdp_01_04.dmp' into bucket 'BUCKET_NAME'
SUCCESS: 2018-09-19 19:40:09: ...file '/tmp/expdp_01_04.dmp' uploaded on 'BUCKET_NAME'
successfully
SUCCESS: 2018-09-19 19:40:09: Dumps upload over Oracle Object Store complete successfully

INFO: 2018-09-19 19:40:09: Performing impdp into ADB...
INFO: 2018-09-19 19:40:09: Step1 - ...drop Object Store Credential
INFO: 2018-09-19 19:40:14: Step2 - ...creating Object Store Credential
INFO: 2018-09-19 19:40:19: Step3 - ...executing import datapump to ADB
INFO: 2018-10-24 19:40:29: Moving impdp log to Object Store
SUCCESS: 2018-09-19 19:41:23: Impdp to ADB 'RCATP' executed successfully

```

**NOTE:** if your target Autonomous Database is ATP-D (dedicated), you could skip the dump creating/upload to the object store leveraging on '**---netlink**' option. Using '**---netlink**' the data are going directly to ADB through a Database Link created automatically by MV2ADB. Using '**---netlink**' SQL\*Net connectivity from ATP-D to source on-premises database is required.

### MV2ADB - Expdp Operation

"Expdp" operation will perform the expdp schemas based from source DB

```

mv2adb expdp {--conf <conf file path> [--goldengate] [--nosudo]} |
{--dbcs <Source DB connect string>
--schemas <Database schemas to export> | --full y [--exclude <Database schemas
to exclude>]
--dumpname <expdp dump file name>
--dumppath <expdp dump file path>
--ohome <Local ORACLE_HOME> | --ichome <Local Oracle Instant Client Home>
--adbname <ADB database name>
--adbcfile <ADB credential zip file>
[--adbccores <ADB cores count>]
[--compression ALL | DATA_ONLY | METADATA_ONLY | NONE ]
[--encryption [--enctype AES128 | AES192 | AES256]]
[--goldengate]
[--nosudo]
}

```

The command option "--goldengate" should be used only as part of a migration with GoldenGate

#### execution example (using configuration file):

```

# ./mv2adb expdp -conf conf/mv2adb.conf
INFO: 2019-10-28 06:38:31: Please check the logfile '/opt/mv2adb/out/log/mv2adb_11859.log'
for more details

INFO: 2019-10-28 06:38:31: Reading the configuration file 'conf/atp-d.conf'
INFO: 2019-10-28 06:38:31: Using Oracle Home '/u01/app/oracle/product/12.2.0.1/dbhome_1'
INFO: 2019-10-28 06:38:32: Getting ADB parallelism
INFO: 2019-10-28 06:38:33: Getting source DB version
INFO: 2019-10-28 06:38:33: Checking schemas on source DB
INFO: 2019-10-28 06:38:33: Creating expdp directory 'MV2ADB_EXPDP_DIR' for path '/tmp'
INFO: 2019-10-28 06:38:33: Getting latest SCN
INFO: 2019-10-28 06:38:33: Checking Cloud Service Type
INFO: 2019-10-28 06:38:35: Executing Expdp
INFO: 2019-10-28 06:39:01: Following expdp dump has been created:
DUMP_FILES=/tmp/expdp_01.dmp,/tmp/expdp_02.dmp,/tmp/expdp_03.dmp,/tmp/expdp_04.dmp,/tmp/expdp_0
SUCCESS: 2019-10-28 06:39:01: Expdp executed successful

```

#### execution example (using command line options):

```

# ./mv2adb expdp \
--dbcs //<host name/ip address>/DB122H1 \
--schemas SCHEMA_1,SCHEMA_2,SCHEMA_3,SCHEMA_4 \
--dumpname expdp.dmp --dumppath /tmp \
--ohome /u01/app/oracle/product/12.2.0.1/dbhome_1 \
--adbname RCATP \
--adbcfile /opt/mv2adb/source/Wallet_RCATP.zip \

INFO: 2019-10-28 06:36:23: Please check the logfile '/opt/mv2adb/out/log/mv2adb_10029.log'
for more details

INFO: 2019-10-28 06:36:23: Using Oracle Home '/u01/app/oracle/product/12.2.0.1/dbhome_1'

Please enter the 'system' SYSTEM Database password:
Please re-enter the 'system' SYSTEM Database password:

Please enter the 'admin' ADB admin password:
Please re-enter the 'admin' ADB admin password:
INFO: 2019-10-28 06:36:32: Getting ADB parallelism
INFO: 2019-10-28 06:36:34: Getting source DB version
INFO: 2019-10-28 06:36:34: Checking schemas on source DB
INFO: 2019-10-28 06:36:34: Creating expdp directory 'MV2ADB_EXPDP_DIR' for path '/tmp'
INFO: 2019-10-28 06:36:34: Getting latest SCN
INFO: 2019-10-28 06:36:34: Checking Cloud Service Type
INFO: 2019-10-28 06:36:36: Executing Expdp
INFO: 2019-10-28 06:37:04: Following expdp dump has been created:
/tmp/expdp_01.dmp,/tmp/expdp_02.dmp,/tmp/expdp_03.dmp,/tmp/expdp_04.dmp,/tmp/expdp_05.dmp,/tmp/

```

SUCCESS: 2019-10-28 06:37:04: Exdpd executed successfully

Note as the expdp dumps are created based on the target Autonomous Database parallelism (cores). In this example two dumps are in use in the parameter "-dumpfile /tmp/rcatp\_exp01.dmp,/tmp/rcatp\_exp02.dmp", but as the target ADB is having one core, only one dump will be created under '/tmp'

## MV2ADB - Impdp Operation

"Impdp" operation will perform the impdp schemas into Autonomous Database Cloud

```
mv2adb impdp {--conf <conf file path>[--nosudo] --swift}} |
  {--ociregion <oci region>
    --ocinamespace <OCI namespace name>
    --ocibucket <bucket name>
    --ociid <OCI ID>
    --adbname <ADB database name>
    --adbfile <ADB credential zip file>
    --ichome <Oracle Instant Client Home>
    --dumpfile <comma separated expdp dump file name>
    [--schemas <Database schemas to export> | --full y [--exclude <Database schemas
to exclude>]]
    [--remap <comma separated remap SOURCE_TBS:TARGET_TBS>]
    [--adbcores <ADB cores count>]
    [--encryption [--enctype AES128 | AES192 | AES256]]
    [--norecomp]
    [--swift]
    [--nosudo]}
```

**NOTE1:** if your source database binary release is 18c or above, the ICHOME parameter can be the source DB ORACLE\_HOME. If it's 11g or 12c you must install latest Oracle Instant Client

**NOTE2:** if your target Autonomous Database is ATP-D (dedicated), you could skip the dump creating/upload to the object store leveraging on '**--netlink**' option. Using '**--netlink**' the data are going directly to ADB through a Database Link created automatically by MV2ADB.

execution example (using configuration file):

```
# ./mv2adb impdp -conf conf/mv2atp.cfg
INFO: 2018-10-24 15:58:06: Please check the logfile '/opt/mv2adb/out/log/mv2adb_11591.log'
for more details

INFO: 2018-10-24 15:58:06: Reading the configuration file 'conf/RCATP.mv2adb.cfg'
INFO: 2018-10-24 15:58:07: Getting ADB parallelism
INFO: 2018-10-24 15:58:12: Checking Cloud Service Type
INFO: 2018-10-24 15:58:17: Drop Object Store Credential
INFO: 2018-10-24 15:58:22: Creating Object Store Credential
INFO: 2018-10-24 15:58:27: Executing Impdp to ADB
SUCCESS: 2018-10-24 15:59:52: Impdp to ADB 'RCATP' executed successfully
```

execution example (using command options):

```
# ./mv2adb impdp \
-ociregion us-phoenix \
-ocinamespace NAMESPACE_NAME \
-ocibucket BUCKET_NAME \
-ociid mail@corp.com \
-adbname RCATP \
```

```
-cfile /opt/mv2adb/source/Wallet_RCATP.zip \  
-ichome /u01/app/oracle/product/instantclient_18_3 \  
-dumpfile rcatp_exp_01.dmp,rcatp_exp_02.dmp \  
-encryption \  
-enctype AES256  
INFO: 2018-10-24 16:24:50: Please check the logfile '/opt/mv2adb/out/log/mv2adb_12541.log'  
for more details
```

Please enter the 'admin' ADB password:  
Please re-enter the 'admin' ADB password:

Please enter the 'OCI' ID password:  
Please re-enter the 'OCI' ID password:

```
Please enter the 'expdp/impdp' encryption password:  
Please re-enter the 'expdp/impdp' encryption password:  
INFO: 2018-10-24 16:25:05: Getting ADB parallelism  
INFO: 2018-10-24 16:25:10: Checking Cloud Service Type  
INFO: 2018-10-24 16:25:15: Drop Object Store Credential  
INFO: 2018-10-24 16:25:21: Creating Object Store Credential  
INFO: 2018-10-24 16:25:26: Executing Impdp to ADB  
INFO: 2018-10-24 16:25:36: Moving impdp log to Object Store  
SUCCESS: 2018-10-24 16:26:47: Impdp to ADB 'RCATP' executed successfully
```

## ***MV2ADB – createbucket Operation***

It's providing the capability to create a new bucket on Oracle Object Store

```
mv2adb createbucket {--conf <conf file path>[--nosudo]} |  
                    {--ocic--ocicompartId <compartment id> --ocibucket <bucket name> [--  
ocinamespace <namespace name>][--nosudo]} |  
                    {--ociregion <oci region>  
                    --ocinamespace <namespace name>  
                    --ociId <user ID>  
                    --ocibucket <bucket name>  
                    [--proxyHost <host> --proxyPort <port> [--proxyId <user ID> --proxyPass  
<password>]]  
                    [--nosudo]}
```

execution example (using configuration file):

```
# ./mv2adb createbucket -conf conf/mv2atp.cfg  
INFO: 2018-09-20 15:46:50: Please check the logfile '/opt/mv2atp/out/log/mv2adb_3933.log'  
INFO: 2018-09-20 15:46:50: Reading the configuration file 'conf/mv2atp.cfg'  
INFO: 2018-09-20 15:46:50: Creating bucket 'BUCKET_NAME'  
SUCCESS: 2018-09-20 15:46:51: Bucket 'BUCKET_NAME' created successfully
```

execution example (using command options):

```
# ./mv2adb createbucket \  
-ociregion us-phoenix \  
-ocinamespace NAMESPACE_NAME \  
-ociid mail@corp.com \  
-ocibucket BUCKET_NAME  
INFO: 2018-10-24 14:40:14: Please check the logfile '/opt/mv2adb/out/log/mv2adb_5267.log' for  
more details  
  
Please enter the 'mail@corp.com' Auth Tokens password for the OCI account:  
Please re-enter the 'mail@corp.com' Auth Tokens password for the OCI account:  
INFO: 2018-10-24 14:40:22: Creating bucket 'BUCKET_NAME'  
SUCCESS: 2018-10-24 14:40:23: Bucket 'BUCKET_NAME' created successfully
```



## MV2ADB – delbucket Operation

It's providing the capability to delete a bucket on Oracle Object Store

```
mv2adb delbucket [--conf <conf file path>[--nosudo]] |
                [--ocic --ocibucket <bucket name> [--ocinamespace <namespace name>][--
nosudo]] |
                [--ociregion <oci region>
                --ocinamespace <namespace name>
                --ociId <user ID>
                --ocibucket <bucket name>
                [--proxyHost <host> --proxyPort <port> [--proxyId <user ID> --proxyPass
<password>]]
                [--nosudo]}
```

execution example (using configuration file):

```
# ./mv2adb delbucket -conf conf/mv2atp.cfg
INFO: 2018-09-20 15:49:01: Please check the logfile '/opt/mv2atp/out/log/mv2atp_4015.log'
INFO: 2018-09-20 15:49:01: Reading the configuration file 'conf/mv2atp.cfg'
INFO: 2018-09-20 15:49:01: Deleting bucket 'BUCKET_NAME'
SUCCESS: 2018-09-20 15:49:02: Bucket 'BUCKET_NAME' deleted successfully
```

execution example (using command options):

```
# ./mv2adb delbucket \
-ociregion us-phoenix \
-ocinamespace NAMESPACE_NAME \
-ociid mail@corp.com \
-ocibucket BUCKET_NAME
INFO: 2018-10-24 14:45:42: Please check the logfile '/opt/mv2adb/out/log/mv2adb_5425.log' for
more details

Please enter the 'mail@corp.com' Auth Tokens password for the OCI account:
Please re-enter the 'mail@corp.com' Auth Tokens password for the OCI account:
INFO: 2018-10-24 14:45:51: Deleting bucket 'BUCKET_NAME'
SUCCESS: 2018-10-24 14:45:52: Bucket 'BUCKET_NAME' deleted successfully
```

## MV2ADB – listbucket Operation

It's providing the capability to list buckets on Oracle Object Store

```
mv2adb listbucket [--conf <conf file path>[--nosudo]] |
                 [--ocic --ocicompartId <compartment id> [--ocinamespace <namespace name>][--
-nosudo]] |
                 [--ociregion <oci region>
                 --ocinamespace <namespace name>
                 --ociId <user ID>
                 --ocibucket <bucket name>
                 [--proxyHost <host> --proxyPort <port> [--proxyId <user ID> --proxyPass
<password>]]
                 [--nosudo]}
```

execution example (using configuration file):

```
# ./mv2adb listbucket -conf conf/mv2atp.cfg
INFO: 2018-09-20 15:48:01: Please check the logfile '/opt/mv2atp/out/log/mv2atp_3978.log'
INFO: 2018-09-20 15:48:01: Reading the configuration file 'conf/mv2atp.cfg'
INFO: 2018-09-20 15:48:01: Getting the buckets
List the buckets for namespace 'inthgnis':
{"name":"BUCKET_NAME_1","count":0,"bytes":0},
{"name":"BUCKET_NAME_2","count":0,"bytes":0},
```

```
{"name":"BUCKET_NAME_3","count":0,"bytes":0}
```

### execution example (using command options):

```
# ./mv2adb listbucket \  
-ociregion us-phoenix \  
-ocinamespace NAMESPACE_NAME \  
-ociid mail@corp.com  
INFO: 2018-10-24 14:44:00: Please check the logfile '/opt/mv2adb/out/log/mv2adb_5360.log' for  
more details  
  
Please enter the 'mail@oracle.com' Auth Tokens password for the OCI account:  
Please re-enter the 'mail@oracle.com' Auth Tokens password for the OCI account:  
INFO: 2018-10-24 14:44:14: Getting the buckets  
  
List the buckets for namespace 'NAMESPACE_NAME':  
{ "name": "BUCKET_NAME_1", "count": 0, "bytes": 0 },  
{ "name": "BUCKET_NAME_2", "count": 0, "bytes": 0 },  
{ "name": "BUCKET_NAME_3", "count": 0, "bytes": 0 }
```

## MV2ADB – putdump Operation

It's providing the capability to load files into a bucket on Oracle Object Store

```
mv2adb putdump {--conf <conf file path> [--nosudo]} |  
  {--ocic --ocibucket <bucket name> -dumpfiles <comma separated dump name> [--  
ocicsize <size Mb> --ociparallel <count>] [--ocinamespace <namespace name>] [--nosudo]} |  
  {--ociregion <oci region>  
  --ocinamespace <namespace name>  
  --ociId <user ID>  
  --ocibucket <bucket name>  
  [--proxyHost <host> --proxyPort <port> [--proxyId <user ID> --proxyPass  
<password>]]  
  [--nosudo]}
```

### execution example (using configuration file):

```
# ./mv2adb putdump -conf conf/mv2atp.cfg  
INFO: 2018-09-20 15:51:36: Please check the logfile '/opt/mv2atp/out/log/mv2adb_4134.log'  
  
INFO: 2018-09-20 15:51:36: Reading the configuration file 'conf/mv2atp.cfg'  
INFO: 2018-09-20 15:51:36: Loading '/tmp/expdp_01.dmp' into bucket 'BUCKET_NAME'  
SUCCESS: 2018-09-20 15:51:39: File '/tmp/expdp_01.dmp' uploaded on 'BUCKET_NAME' successfully  
INFO: 2018-09-20 15:51:39: Loading '/tmp/expdp_02.dmp' into bucket 'BUCKET_NAME'  
SUCCESS: 2018-09-20 15:51:40: File '/tmp/expdp_02.dmp' uploaded on 'BUCKET_NAME' successfully
```

### execution example (using command options):

```
# ./mv2adb putdump -ociregion us-phoenix \  
-ocinamespace NAMESPACE_NAME \  
-ocibucket BUCKET_NAME \  
-ociid mail@corp.com \  
-dumpfiles /tmp/rcatp_exp01_01.dmp  
INFO: 2018-10-24 14:35:22: Please check the logfile '/opt/mv2adb/out/log/mv2adb_5125.log' for  
more details  
  
Please enter the 'mail@corp.com' Auth Tokens password for the OCI account:  
Please re-enter the 'mail@corp.com' Auth Tokens password for the OCI account:  
INFO: 2018-10-24 14:35:32: Loading '/tmp/rcatp_exp01_01.dmp' into bucket 'RC_Bucket'  
SUCCESS: 2018-10-24 14:35:54: File '/tmp/rcatp_exp01_01.dmp' uploaded on 'RC_Bucket'  
successfully
```

## MV2ADB – getdump Operation

It's providing the capability to get files from bucket on Oracle Object Store

```
mv2adb getdump [--conf <conf file path>[--nosudo]] |
               [--ocic --ocibucket <bucket name> -dumpfiles <comma separated dump name> [--
ocinamespace <namespace name>][--force][--nosudo]] |
               [--ociregion <oci region>
               --ocinamespace <namespace name>
               --ociId <user ID>
               --ocibucket <bucket name>
               --dumpfiles <comma separated dump file name>
               [--proxyHost <host> --proxyPort <port> [--proxyId <user ID> --proxyPass
<password>]]
               [--nosudo]]
```

execution example (using configuration file):

```
# ./mv2adb getdump -conf conf/mv2atp.cfg
INFO: 2018-09-20 15:53:57: Please check the logfile '/opt/mv2atp/out/log/mv2adb_4226.log'
INFO: 2018-09-20 15:53:57: Reading the configuration file 'conf/mv2atp.cfg'
INFO: 2018-09-20 15:53:58: Downloading '/tmp/expdp_01.dmp' from bucket 'BUCKET_NAME'
SUCCESS: 2018-09-20 15:53:59: File '/tmp/expdp_01.dmp' downloaded successfully
INFO: 2018-09-20 15:53:59: Downloading '/tmp/expdp_02.dmp' from bucket 'BUCKET_NAME'
SUCCESS: 2018-09-20 15:54:01: File '/tmp/expdp_02.dmp' downloaded successfully
```

execution example (using command options):

```
# ./mv2adb getdump \
-ociregion us-phoenix \
-ocinamespace NAMESPACE_NAME \
-ocibucket BUCKET_NAME \
-ociId mail@corp.com \
-dumpfiles rcatp_exp01_01.dmp
INFO: 2018-10-24 14:48:38: Please check the logfile '/opt/mv2adb/out/log/mv2adb_5484.log' for
more details

Please enter the 'mail@corp.com' Auth Tokens password for the OCI account:
Please re-enter the 'mail@corp.com' Auth Tokens password for the OCI account:
INFO: 2018-10-24 14:48:46: Downloading 'rcatp_exp01_01.dmp' from bucket 'BUCKET_NAME'
SUCCESS: 2018-10-24 14:48:50: File 'rcatp_exp01_01.dmp' downloaded successfully

# ls -l rcatp_exp01_01.dmp
-rw-rw-rw- 1 root root 2269184 Oct 24 14:48 rcatp_exp01_01.dmp
```

**Note:** the -dumpfiles option is specifying just only the dump file name without path

## MV2ADB – deldump Operation

It's providing the capability to delete a file from bucket on Oracle Object Store

```
mv2adb deldump [--conf <conf file path>[--nosudo]] |
               [--ocic --ocibucket <bucket name> --dumpfiles <comma separated dump name> [--
ocinamespace <namespace name>][--force][--nosudo]] |
               [--ociregion <oci region>
               --ocinamespace <namespace name>
               --ociId <user ID>
               --ocibucket <bucket name>
               --dumpfiles <comma separated dump file name>
               [--force]
               [--proxyHost <host> --proxyPort <port> [--proxyId <user ID> --proxyPass
<password>]]
```

```
[--nosudo]}
```

### execution example (using configuration file):

```
# ./mv2adb deldump -conf conf/mv2atp.cfg
INFO: 2018-09-20 15:57:13: Please check the logfile '/opt/mv2atp/out/log/mv2adb_4331.log'
INFO: 2018-09-20 15:57:13: Reading the configuration file 'conf/mv2atp.cfg'
You are going to delete the objects on bucket 'BUCKET_NAME', are you sure (Y/N)? y
INFO: 2018-09-20 15:57:30: Deleting objects for bucket 'BUCKET_NAME', it will take some
time...
INFO: 2018-09-20 15:57:30: Deleting '/tmp/expdp_01.dmp' from bucket 'BUCKET_NAME'
SUCCESS: 2018-09-20 15:57:31: Objects '/tmp/expdp_01.dmp' removed from bucket 'BUCKET_NAME'
successfully
INFO: 2018-09-20 15:57:31: Deleting '/tmp/expdp_02.dmp' from bucket 'BUCKET_NAME'
SUCCESS: 2018-09-20 15:57:32: Objects '/tmp/expdp_02.dmp' removed from bucket 'BUCKET_NAME'
successfully
```

### execution example (using command options):

```
# ./mv2adb deldump \
-ociRegion us-phoenix \
-ociNamespace NAMESPACE_NAME \
-ociBucket BUCKET_NAME \
-ociId mail@corp.com \
-dumpfiles rcatp_exp01_01.dmp
INFO: 2018-10-24 14:53:23: Please check the logfile '/opt/mv2adb/out/log/mv2adb_7611.log' for
more details

You are going to delete the objects on bucket 'BUCKET_NAME', are you sure (Y/N)? y

Please enter the 'mail@corp.com' Auth Tokens password for the OCI account:
Please re-enter the 'mail@corp.com' Auth Tokens password for the OCI account:
INFO: 2018-10-24 14:53:40: Deleting objects for bucket 'BUCKET_NAME', it will take some
time...
INFO: 2018-10-24 14:53:40: Deleting 'rcatp_exp01_01.dmp' from bucket 'BUCKET_NAME'
SUCCESS: 2018-10-24 14:53:41: Objects 'rcatp_exp01_01.dmp' removed from bucket 'BUCKET_NAME'
successfully
```

**Note:** the `-dumpfiles` option is specifying just only the dump file name without path

## MV2ADB – listdump Operation

It's providing the capability to list a files into a bucket on Oracle Object Store

```
mv2adb listdump {--conf <conf file path>[--nosudo]} |
                {--ocic --ocibucket <bucket name> [--ocinamespace <namespace name>][--
nosudo]} |
                {--ociRegion <oci region>
                --ocinamespace <namespace name>
                --ociId <user ID>
                --ocibucket <bucket name>
                [--proxyHost <host> --proxyPort <port> [--proxyId <user ID> --proxyPass
<password>]]
                [--nosudo]}
```

### execution example (using configuration file):

```
# ./mv2adb listdump -conf conf/mv2atp.cfg
INFO: 2018-09-20 15:52:30: Please check the logfile '/opt/mv2atp/out/log/mv2atp_4177.log' for
more
details
INFO: 2018-09-20 15:52:30: Reading the configuration file 'conf/mv2atp.cfg'
```

```
INFO: 2018-09-20 15:52:30: Getting bucket 'BUCKET_NAME' objects
List dump into bucket 'BUCKET_NAME':
{"name":"expdp_01.dmp","hash":"413229e921e6231c99c10bd7fa09c038","bytes":172032,"last_modified":
09-20T11:51:41.426000","content_type":"binary/octet-stream"},
{"name":"expdp_01_01.dmp","hash":"4806b83e9761062745ca976a0f3826c5","bytes":16384,"last_modified":
09-20T11:11:11.041000","content_type":"binary/octet-stream"},
{"name":"expdp_01_02.dmp","hash":"0a877686e6ad32243f738ff91ca9e88e","bytes":32768,"last_modified":
09-20T11:11:12.414000","content_type":"binary/octet-stream"},
{"name":"expdp_01_03.dmp","hash":"848803b25eeb90306100cc3ebe27e1ac","bytes":8192,"last_modified":
09-20T11:11:13.499000","content_type":"binary/octet-stream"},
{"name":"expdp_01_04.dmp","hash":"7a16f5b55759dfd6a65461d0a24b6366","bytes":335872,"last_modified":
09-20T11:11:17.711000","content_type":"binary/octet-stream"},
{"name":"expdp_02.dmp","hash":"11441ed58977f76a27080cb296778732","bytes":24576,"last_modified":
09-20T11:51:42.751000","content_type":"binary/octet-stream"}
```

### execution example (using command options):

```
# ./mv2adb listdump \
-ociregion us-phoenix \
-ocinamespace NAMESPACE_NAME \
-ocibucket BUCKET_NAME \
-ociid mail@corp.com
INFO: 2018-10-24 14:56:18: Please check the logfile '/opt/mv2adb/out/log/mv2adb_7737.log' for
more details

Please enter the 'mail@corp.com' Auth Tokens password for the OCI account:
Please re-enter the 'mail@corp.com' Auth Tokens password for the OCI account:
INFO: 2018-10-24 14:56:24: Getting bucket 'BUCKET_NAME' objects

List dump into bucket 'BUCKET_NAME':
{"name":"expdp_01.dmp","hash":"413229e921e6231c99c10bd7fa09c038","bytes":172032,"last_modified":
09-20T11:51:41.426000","content_type":"binary/octet-stream"},
{"name":"expdp_01_01.dmp","hash":"4806b83e9761062745ca976a0f3826c5","bytes":16384,"last_modified":
09-20T11:11:11.041000","content_type":"binary/octet-stream"},
{"name":"expdp_01_02.dmp","hash":"0a877686e6ad32243f738ff91ca9e88e","bytes":32768,"last_modified":
09-20T11:11:12.414000","content_type":"binary/octet-stream"},
{"name":"expdp_01_03.dmp","hash":"848803b25eeb90306100cc3ebe27e1ac","bytes":8192,"last_modified":
09-20T11:11:13.499000","content_type":"binary/octet-stream"},
{"name":"expdp_01_04.dmp","hash":"7a16f5b55759dfd6a65461d0a24b6366","bytes":335872,"last_modified":
09-20T11:11:17.711000","content_type":"binary/octet-stream"},
{"name":"expdp_02.dmp","hash":"11441ed58977f76a27080cb296778732","bytes":24576,"last_modified":
09-20T11:51:42.751000","content_type":"binary/octet-stream"}
```

### **MV2ADB – advisor Operation**

MV2ADB is including Oracle Autonomous Database Schema Advisor [Note:2462677.1](#). 'mv2adb advisor' it will install/execute the advisor automatically

The ADB Schema Advisor is a light-weight utility that analyzes your on-premise or cloud Oracle Database schemas for the suitability of migration to the Oracle Autonomous Database. The Advisor discovers the schema objects and performs deep analysis to highlight the differences when the object gets created on Oracle Autonomous Data Warehouse or Oracle Autonomous Transaction Processing database, including the reasons when the object cannot be created.

The Advisor will run on pre-existing Schema and generates a report that highlights :

- The counts of discovered objects and a summary of migration status.
- Objects that cannot be migrated to the Autonomous Database due to the restrictions or lockdowns imposed on certain data types, Oracle Database options and SQL.
- The Objects that will migrate with modifications automatically done during the import or the object creation process
- Informational section containing certain best practice recommendations and guidance

Note1: You may specify a maximum of 30 schemas in a single advisor run. If you like to run more than 30, consider running using "--schemas ALL"

Note2: By default, the output gets truncated when the number of rows exceeds the maximum limit set in the Advisor package (2000). You can reset the number of rows by using "--setmaxrows <rows number>"

## Advisor command options

```
mv2adb advisor {--conf <conf file path>[--nosudo] [--setmaxrows <rows number>]} |
  {--ohome <Local ORACLE_HOME> | --ichome <Local Oracle Instant Client Home>
  --dbcs <Source DB connect string>
  --schemas <comma separated database schemas to report>
  --adbtargt <ADW|ADWD|ATP|ATPD> | --adbname <ADB database name> --adbfile
  <ADB credential zip file>
  [--setmaxrows <rows number>]
  [--nosudo]}
```

Without ADB connectivity ( -adbname <ADB database name> -file <ADB credential zip file>) you can specify the target ADB flavor you are going to test using "-adbtargt <ADW|ADWD|ATP|ATPD>"

## execution example (using configuration file):

```
# ./mv2adb advisor -conf conf/mv2atp.cfg
INFO: 2019-09-01 17:44:41: Please check the logfile '/opt/mv2adb/out/log/mv2adb_27930.log'
for more details

INFO: 2019-09-01 17:44:41: Reading the configuration file 'conf/kvm_atp-d.conf'
INFO: 2019-09-01 17:44:41: using Oracle Home '/u01/app/oracle/product/19.3.0.0/dbhome_1'
INFO: 2019-09-01 17:44:42: Getting advisor report for '///<host name/ip address>/DB193H1'
INFO: 2019-09-01 17:44:42: ...step1 - making the advisor user on '///<host name/ip
address>/DB193H1'
INFO: 2019-09-01 17:44:43: ...step2 - add required grants to the advisor user on '/ip
address/DB193H1'
INFO: 2019-09-01 17:44:44: ...step3 - installing advisor package on '///<host name/ip
address>/DB193H1'
INFO: 2019-09-01 17:44:45: ...step4 - getting advisor report
INFO: 2019-09-01 17:44:45: ...getting advisor report for schema 'SCHEMA_1', it may get some
time...
```

```
=====
== ATPD SCHEMA MIGRATION REPORT FOR SCHEMA_1
=====
```

```
-----
-- ATPD MIGRATION SUMMARY
-----
```

Object Type	Total Objects	Objects		
		Objects Not Migrated	Objects Migrated With Changes	Objects Migrated
CONSTRAINT	48	0	0	48
INDEX	27	0	0	27
PACKAGE	3	0	0	3
PACKAGE BODY	3	0	0	3
TABLE	11	0	0	11
VIEW	2	0	0	2

```
-----
-- ATPD OBJECTS NOT MIGRATED
-----
```

```
-- ATPD OBJECTS MIGRATED WITH CHANGES
```

```
-----  
-----  
-- ATPD MIGRATION ADDITIONAL INFO  
-----
```

```
1. Parallel DEGREE > 1 specified on INDEX (Count=18):  
-----
```

Note: If a PARALLEL clause is specified on the index in your current database, it remains with the index when it gets created, via data pump or manual, in both ATP Serverless and Dedicated. This can lead to SQL statements running in parallel unbeknownst to the end user. To specify serial execution, change the INDEX parallel clause to NOPARALLEL or set the PARALLEL degree to 1.

```
CUST_FUNC_LOWER_NAME_IX CARDDetails_CUST_IX CUST_ACCOUNT_MANAGER_IX  
CUST_DOB_IX CUST_EMAIL_IX INV_PRODUCT_IX  
INV_WAREHOUSE_IX ORD_WAREHOUSE_IX PROD_CATEGORY_IX  
PROD_NAME_IX PROD_SUPPLIER_IX WHS_LOCATION_IX  
ADDRESS_CUST_IX ITEM_ORDER_IX ITEM_PRODUCT_IX  
ORD_CUSTOMER_IX ORD_ORDER_DATE_IX ORD_SALES_REP_IX
```

```
2. Columns defined with BYTE semantics in a single-byte DB character set (Count=37):  
-----
```

Note: Character set used by the Autonomous Database is AL32UTF8 (Multi-byte). But if you create a table with a column that uses BYTE semantics, it may not allow you to fit all characters as "Multi-byte" needs more bytes to store one character. Ensure that your BYTE columns are transformed to CHARACTER semantics before the migration as Oracle Data Pump currently does not do this.

```
CARD_DETAILS (COLS=2) ORDERENTRY_METADATA (COLS=2) PRODUCT_DESCRIPTIONS (COLS=1)  
ADDRESSES (COLS=7) CUSTOMERS (COLS=11) ORDERS (COLS=4)  
PRODUCTS (COLS=3) PRODUCT_INFORMATION (COLS=4) ORDER_ITEMS (COLS=2)  
WAREHOUSES (COLS=1)
```

```
3. DEFAULT user PROFILE will be altered (Count=4):  
-----
```

Note: In ATP and ADW, User's profile will be set to 'DEFAULT', and you are not allowed to create additional PROFILES. Below are the differences in '<USERNAME\_1>' user's current profile and the DEFAULT profile in ADW/ATP.

```
RESOURCE CURRENT LIMITS IN DB NEW LIMITS IN ADW/ATP  
-----
```

```
PASSWORD_LIFE_TIME 180 360  
PASSWORD_REUSE_MAX UNLIMITED 4  
PASSWORD_REUSE_TIME UNLIMITED 1  
PASSWORD_VERIFY_FUNCTION NULL CLOUD_VERIFY_FUNCTION
```

```
4. Database Options currently in use but not available in ADB (Count=2):  
-----
```

Note: The following Database Options are detected as being used. ADB does not have these Options installed. Please verify if the application/schema to be migrated depends on these options.

```
Java in DB Tuning Pack
```

```
5. Database Parameters are set in your database but can't be set in ADB (Count=6):  
-----
```

Note: The following init parameters are set in your database that you would not be able to set in ADB. Please refer to the Oracle Autonomous Database documentation on the parameters that you are allowed to modify/set in the autonomous database.

```
_ipddb_enable filesystemio_options local_listener  
session_cached_cursors thread undo_retention
```

## MV2ADB – report Operation

It's providing the capability to get a schema object report to compare the result of expdp/impdp

```

mv2adb report {--conf <conf file path>[--nosudo]} |
              {--dbcs <Source DB connect string>
              --schemas <Database schemas to report>}
              --ohome <Local ORACLE_HOME> | --ichome <Local Oracle Instant Client Home>
              --adbname <ADB database name>
              --adbfile <ADB credential zip file>
              [--nosudo]}

```

execution example (using configuration file):

```

# ./mv2adb report -conf conf/trgdb.conf
INFO: 2019-03-05 16:30:25: Please check the logfile '/opt/mv2adb/out/log/mv2adb_20630.log'
for more details

```

```

INFO: 2019-03-05 16:30:25: Reading the configuration file 'conf/trgdb.conf'
INFO: 2019-03-05 16:30:26: Checking OCHost '<host name/ip address>' connectivity
INFO: 2019-03-05 16:30:28: Getting Obj report for '//<host name/ip address>/DB122H1'

```

Database Objects REPORT for //<host name/ip address>/DB122H1

SCHEMA_1	INDEX		27
SCHEMA_1	PACKAGE		3
SCHEMA_2	TABLE		11
SCHEMA_2	PACKAGE		3
SCHEMA_3	INDEX		27
SCHEMA_3	PACKAGE		3
SCHEMA_3	PACKAGE BODY		3
SCHEMA_4	PACKAGE BODY		3
SCHEMA_2	PACKAGE BODY		3
SCHEMA_3	VIEW		2
SCHEMA_1	PACKAGE BODY		3
SCHEMA_2	VIEW		2
SCHEMA_4	TABLE		11
SCHEMA_1	VIEW		2
SCHEMA_3	TABLE		11
SCHEMA_4	PACKAGE		3
SCHEMA_4	INDEX		27
SCHEMA_4	VIEW		2
SCHEMA_1	TABLE		11
SCHEMA_2	INDEX		27
		-----	
sum			184

```

INFO: 2019-03-05 16:30:28: Getting Obj report for ADB 'myadb_01'

```

Database Objects REPORT for ADB 'myadb\_011'

SCHEMA_1	INDEX		27
SCHEMA_1	PACKAGE		3
SCHEMA_2	TABLE		11
SCHEMA_2	PACKAGE		3
SCHEMA_3	INDEX		27
SCHEMA_3	PACKAGE		3
SCHEMA_3	PACKAGE BODY		3
SCHEMA_4	PACKAGE BODY		3
SCHEMA_2	PACKAGE BODY		3
SCHEMA_3	VIEW		2
SCHEMA_1	PACKAGE BODY		3
SCHEMA_2	VIEW		2
SCHEMA_4	TABLE		11
SCHEMA_1	VIEW		2
SCHEMA_3	TABLE		11
SCHEMA_4	PACKAGE		3
SCHEMA_4	INDEX		27
SCHEMA_4	VIEW		2
SCHEMA_1	TABLE		11
SCHEMA_2	INDEX		27
		-----	



## MV2ADB – recomp Operation

It's providing the capability to recompile schema objects

```
mv2adb recomp {--conf <conf file path>[--nosudo]} |
  {--schemas <Database schemas to report>}
  --ohome <Local ORACLE_HOME> | --ichome <Local Oracle Instant Client Home>
  --adbname <ADB database name>
  --adbcfile <ADB credential zip file>
  [--nosudo]}
```

execution example (using configuration file):

```
# ./mv2adb recomp --conf conf/rcitton_atps.cfg
INFO: 2020-01-24 14:08:51: Please check the logfile '/opt/mv2adb/out/log/mv2adb_5599.log' for
more details

INFO: 2020-01-24 14:08:51: Reading the configuration file 'conf/rcitton_atps.cfg'
INFO: 2020-01-24 14:08:51: Using Oracle Home '/u01/app/oracle/product/12.2.0.1/dbhome_1'
INFO: 2020-01-24 14:08:53: Recompiling schemas on ADB 'rcatps_high'...
INFO: 2020-01-24 14:08:53: ...recompiling schema 'SCHEMA_1', it may get some time...
INFO: 2020-01-24 14:08:54: ...recompiling schema 'SCHEMA_2', it may get some time...
INFO: 2020-01-24 14:08:57: ...recompiling schema 'SCHEMA_3', it may get some time...
INFO: 2020-01-24 14:09:00: ...recompiling schema 'SCHEMA_4', it may get some time...
```

## MV2ADB – encpass Operation

It's generating an encrypted password to be used into the configuration file

Example:

```
# ./mv2adb encpass

Please enter the password :
Please re-enter the password :
<enc_password>
```

## MV2ADB Command Option

```
mv2adb OPTIONS
--conf: Config file

--dbcs:      Database connect string
--schemas:   Comma separated source Database schemas name
--full:      Performs a full expdp
--exclude:   Comma separated source Database schemas name to be excluded
--remap:     Comma separated remap tablespaces
--ohome:     Oracle RDBMS HOME
--ichome:    Oracle Instant Client HOME
--dumpname:  ExpDP dump file name
--dumppath:  ExpDP dump file path
--dumpfiles: Dump file name
--netlink:   Datapump network link usage
```

```

--encryption: Expdp/impdp encryption
--enctype:      Expdp/impdp encryption type: AES128 | AES192 | AES256 (default: AES128)

--adbname:      Oracle ADB name
--adbcores:     Oracle ADB Core count
--adbtarget:    Oracle ADB tyoe : <ADW | ATP | ATPD

--ocic:         Using oci-client
--ocicsize:     Split file size Mb (Default: 128)
--ocicparallel: Parallel upload count (Default: 3)
--ocicompartId: Object Store compartment id (in use with list/create bucket using oci-cli)

--ociregion:    OCI region name
--ocinamespace: OCI namespace to store backups
--ocibucket:    OCI bucket to store backups
--ociId:        The userid for the Oracle Public Cloud account
--ociPass:      Auth Tokens password for the OCI account
--swift:        Forcing swift URL on OCI_HOST/ocihost in case of impdp
--proxyHost:    HTTP proxy server
--proxyPort:    HTTP proxy server port (Default: 80)
--proxyId:      HTTP proxy server username
--proxyPass:    HTTP proxy server password

--goldengate:  To be used as part of a migration with GoldenGate
--nosudo:      Execution w/o superuser or sudo setup
--setmaxrows   Schema Advisor Maximum Rows Displayed for Output Truncation (default: 2000)
--dpdebug:     Setting datapump debug parameters
--force:       Force the action (deldump, getdump)
--help:        Command help
--version:     Show program's version number and exit

```

## MV2ADB Configuration File

Move to Autonomous Database can be executed using command line options or using a configuration file with all options needed. The passwords in use must be encrypted using "mv2adb encpass" command.

```

#-----#
# DB Parameters                               #
#-----#
DB_CONSTRING=                                # Source DB connect string
SYSTEM_DB_PASSWORD=                          # Source Database 'system' password (with Advisor
command option is the 'sys' password), must be encrypted by 'mv2adb encpass'
SCHEMAS=                                     # Comma separated schemas to export/import
FULL=                                         # FULL=Y for full expdp
EXCLUDE=                                     # Comma separated schemas to be excluded
REMAP=                                        # comma separated remap SOURCE_TBS:TARGET_TBS
DUMP_NAME=                                   # ExpDP dump file name
DUMP_PATH=                                   # ExpDP dump file path
DUMP_FILES=                                  # Comma separated dump files
OHOME=                                       # RDBMS Oracle Home
ICHOME=                                      # Instant Client Oracle Home mandatory with
auto/impdp
#-----#
# Expdp/Impdp Parameters                     #
#-----#
ENC_PASSWORD=                                # Expdp/Impdp encryption password, must be
encrypted by 'mv2adb encpass'
ENC_TYPE=                                    # Expdp/Impdp encryption type { AES128 | AES192 |
AES256 }
#-----#
# ADB Parameters                             #
#-----#
ADB_NAME=                                    # Database name
ADB_PASSWORD=                                # ADB Database 'admin' password
ADB_CORES=                                   # Optional (skip auto check)
ADB_TARGET=                                  # Optional in use with 'advisor' command

```

```

{ADW|ATP|ATPD}
ADB_CFILE=                                # Credential zip file
#-----#
# Object Store Properties                  #
#-----#
OCI_REGION=                                # mandatory if OCIC=false or not defined
OCI_NAMESPACE=                             # mandatory if OCIC=false or not defined
OCI_BUCKET=                                # mandatory
#-----#
OCI_ID=                                    # Oracle Database Backup Cloud Service account
OCI_PASSWORD=                              # Auth Tokens password for the OCI account, must
be encrypted by 'mv2adb encpass'
#-----#
PROXY_HOST=                                # Optional
PROXY_PORT=                                # Optional
PROXY_ID=                                   # Optional
PROXY_PASSWORD=                            # Optional, must be encrypted by 'mv2adb encpass'
#-----#
# OCI-Client Properties                   #
#-----#
OCIC=                                       # Optional, if true oci-client will be used
OCIC_SIZE=                                 # Split file size Mb (Default: 128)
OCIC_PARALLEL=                             # Parallel upload count (Default: 3)
OCIC_COMPART_ID=                           # Oracle Object Store Compartment (required with
list/create bucket using oci-cli)
#-----#
# End Of File                              #
#-----#

```

#### configuration file example:

```

#-----#
# DB Parameters #
#-----#
DB_CONSTRING=//<host name/ip address>/DB122H1
SYSTEM_DB_PASSWORD=<enc_password>
SCHEMAS=SCHEMA_1,SCHEMA_2,SCHEMA_3,SCHEMA_4
REMAP=TBS_1:DATA,TBS_1:DATA,TBS_3:DATA,TBS_4:DATA
DUMP_NAME=expdp.dmp
DUMP_PATH=/tmp
DUMP_FILES=/tmp/expdp_01.dmp,/tmp/expdp_02.dmp
OHOME=/u01/app/oracle/product/12.2.0.1/dbhome_1
ICHOME=/home/RC-SHARED/Software-Tools/Software/instantclient_18_3
#-----#
# Expdp/Impdp Parameters #
#-----#
ENC_PASSWORD=
ENC_TYPE=AES256
#-----#
# ADB Properties #
#-----#
ADB_NAME=myatp
ADB_PASSWORD=<enc_password>
ADB_CFILE=/opt/mv2adb/source/ATP_wallet.zip
#-----#
# Object Store Properties #
#-----#
OCI_REGION=us-phoenix-1
OCI_NAMESPACE=NAMESPACE_NAME
OCI_BUCKET=BUCKET_NAME
OCI_ID=mail@corp.com
OCI_PASSWORD=<enc_password>
#
PROXY_HOST=
PROXY_PORT=
PROXY_ID=
PROXY_PASSWORD=
#-----#

```

```
# OCI-Client Properties #
#-----#
OCIC=true
OCIC_SIZE=10
OCIC_PARALLEL=10
OCIC_COMPART_ID=ocid1.compartment.oc1..aaaaaaaaymakant1lqnlcf5l73trwrndl5wfc5ccxzbed7tuoyc4nnj
#-----#
# End Of File #
#-----#
```

### Configuration File Parameters Notes

Parameters such DUMP\_NAME and DUMP\_PATH are in use by expdp (and auto) functions. MV2ADB is using automatically '%u' in the dump name. Having DUMP\_NAME=expdump.dmp, the dump files will be expdump\_1.dmp, expdump\_2.dmp, ....

DUMP\_PATH: is the path where the expdp dump will be created

DUMP\_FILES: is in used only by "listdump,getdump,putdump,deldump,impdp" funcations. In case of "impdp" the path is not important, only the filename will be used.

REMAP: is describing how the source tablespace will me mappend on the target ADB:

SOURCE\_tablespace:ADB\_tablespace (example: REMAP=TBS\_1:DATA,TBS\_1:DATA,TBS\_3:DATA,TBS\_4:DATA)

ADB\_CORES: if it is not provided, MV2ADB will query the core\_count of the autonomous database setting the impdp parallelism

ADB\_TARGET: is in use by "advisor" funcation and if it not provided, MV2ADB will get the ADB type automatically

ICHOME: is required if the source database is 11g or 12c. Having a source db 18c or above the impdp is supporting the object store

OCI\_ID: if the user is a federated user, the right username will be in the form: 'oracleidentitycloudservice/<email address>'

### MV2ADB Installation

mv2adb can be installed "on premises" using the RPM (RedHat Package Manager) command as following:

```
# rpm -i mv2adb-2.0.1-X.noarch.rpm
```

(\* X=version number

or updating an installed version, issuing:

```
# rpm -Uvh mv2adb-2.0.1-X.noarch.rpm
```

(\* X=version number

Following files are created under '/opt/mv2adb':

```
# tree /opt/mv2adb/
/opt/mv2adb/
├── conf
│   └── DBNAME.mv2atp.cfg
├── lib
│   ├── MV2ADB_LoggingAndTracing.pm
│   ├── MV2ADB_passwd.jar
│   ├── MV2ADB_PFile.pm
│   ├── MV2ADB_Queries.pm
│   └── MV2ADB_Utils.pm
```

```
└─ mv2adb
└─ utils
   └─ install_adb_advisor.sql
   └─ premigration.jar
```

3 directories, 9 files

If for any reason, you can not install MV2ADB using the RPM, you could use the ['mv2adb-2.0.1-X.tar.gz'](#) attached to this article and uncompress it on the target folder:

```
tar xvfz mv2adb-2.0.1.X.tar.gz -C <target directory>
```

Example:

```
$ mkdir -p /home/oracle/mv2adb
$ tar xvfz mv2adb-2.0.1.X.tar.gz -C /home/oracle/mv2adb
```

```
-->
$ tree /home/oracle/mv2adb
/home/oracle/mv2adb
└─ mv2adb-2.0.1.x
   └─ conf
       └─ DBNAME.mv2adb.cfg
   └─ lib
       ├── MV2ADB_LoggingAndTracing.pm
       ├── MV2ADB_passwd.jar
       ├── MV2ADB_PFile.pm
       ├── MV2ADB_Queries.pm
       └─ MV2ADB_Utils.pm
   └─ mv2adb
   └─ utils
       ├── install_adb_advisor.sql
       └─ premigration.jar
```

4 directories, 9 files

Before to execute m2adb you must set the **MV2ADB\_HOME** env variable, example:

```
$ export MV2ADB_HOME=/home/oracle/mv2adb/mv2adb-2.0.1.x
```

and as probably you are executing mv2adb w/o root privileges neither "sudo", you must use "**--nosudo**" command option:

```
$ /home/oracle/mv2adb/mv2adb-2.0.1.x/mv2adb <operation> <options> --nosudo
```

## MV2ADB De-installation

mv2adb can be removed using the RPM (RedHat Package Manager) command as following:

```
# rpm -e mv2adb-2.0.1-X.noarch
```

(\*) X=version number

## Managing MV2ADB Privileges and security with sudo

MV2ADB command-line utility requires root system privileges for most actions. You may want to use SUDO as part of

your system auditing and security policy.

For most tasks, you need to log in as root to use the mv2adb command-line interface if "--nosudo" command option is not provided

### **Allowing Root User Access Using SUDO**

In environments where system administration is handled by a different group than database administration, or where security is a significant concern, you may want to limit access to the root user account and password. SUDO enables system administrators to grant certain users (or groups of users) the ability to run commands as root, while logging all commands and arguments as part of your security and compliance protocol.

A SUDO security policy is configured by using the file /etc/sudoers. Within the sudoers file, you can configure groups of users and sets of commands to simplify and audit server administration with SUDO commands.

Caution: Configuring SUDO to allow a user to perform any operation is equivalent to giving that user root privileges. Consider carefully if this is appropriate for your security needs.

### **SUDO Example 1: Allow a User to Perform Any mv2adb Operation**

This example shows how to configure SUDO to enable a user to perform any mv2adb operation. You do this by adding lines to the commands section in the /etc/sudoers file:

```
Cmnd_Alias MV2ADB_CMD=/opt/mv2adb/mv2adb *
<USERNAME_1> ALL = MV2ADB_CMD
```

In this example, the user name is <USERNAME\_1>. The file parameter setting ALL= MV2ADB\_CMDS grants the user <USERNAME\_1> permission to run all mv2adb commands that are defined by the command alias MV2ADB\_CMDS.

### **SUDO Example 2: Allow a User to Perform Only Selected mv2adb Operations**

To configure SUDO to allow an user to perform only selected mv2adb operations, add lines to the commands section in the /etc/sudoers file as follows:

```
Cmnd_Alias MV2ADB_CMD=/opt/mv2adb/mv2adb putdump
<USERNAME_1> ALL = MV2ADB_CMD
```

### **SUDO Example 3: Allow a User to Perform Any mv2adb Operation without password request**

To configure SUDO to allow an user to perform mv2adb operations without password request, add lines to the commands section in the /etc/sudoers file as follows:

```
Cmnd_Alias MV2ADB_CMD=/opt/mv2adb/mv2adb *
<USERNAME_1> ALL=(root) NOPASSWD:MV2ADB_CMD
```

## **MV2ADB Usage Example**

1. Install latest Oracle Instant Client Basic and Tools Packages
2. Install "mv2adb -2.0.1-X.noarch.rpm"
3. Download the Autonomous Database Cloud Client Credentials (Wallet)



AVAILABLE

# RC\_ATP

DB Connection

Service Console

Scale Up/Down

Stop

Actions

## Autonomous Transaction Processing Database Information

Tags

**Display Name:** RC\_ATP

**Database Name:** RCATP

**Database Version:** 18.0.3.3

**CPU Core Count:** 1

**Storage (TB):** 1

**Created:** Tue, 23 Oct 2018 12:12:47 GMT

**Compartment:** RCitton

**OCID:** ...ax6qpq [Show](#) [Copy](#)

**License Type:** Bring Your Own License

**Lifecycle State:** Available

You will need the client credentials and connection information to connect to your database. The client credentials include the wallet, which is required for all types of connections.

### Download Client Credentials (Wallet)

To download your client credentials, click Download, and supply a password for the wallet.

Download

### Connection Strings

Use the following connection strings or TNS names for your connections. See the [documentation](#) for details.

TNS Name ⓘ	Connection String ⓘ
RCATP_high	...522/yybsyd6qttdmu9q_rcatp_high.atp.oraclecloud.com <a href="#">Show</a>
RCATP_low	...1522/yybsyd6qttdmu9q_rcatp_low.atp.oraclecloud.com <a href="#">Show</a>
RCATP_medium	...2/yybsyd6qttdmu9q_rcatp_medium.atp.oraclecloud.com <a href="#">Show</a>

Showing 3 Item(s)

Close

#### 4. Compile the mv2adb configuration file with all requirements, example (RCATP.mv2adb.cfg)

You can generate the encrypted passwords using "**mv2adb encpass**" command

```
#-----#
# DB Parameters #
#-----#
DB_CONSTRIG=//<host name/ip address>/DB122H1
SYSTEM_DB_PASSWORD=<enc_password>
SCHEMAS=SCHEMA_1, SCHEMA_2, SCHEMA_3, SCHEMA_4
REMAP=TBS_1:DATA, TBS_1:DATA, TBS_3:DATA, TBS_4:DATA
SREMAP=SCHEMA_1:DATA, SCHEMA_2:DATA, SCHEMA_3:DATA, SCHEMA_4:DATA
DUMP_NAME=expdp.dmp
DUMP_PATH=/tmp
DUMP_FILES=/tmp/expdp_01.dmp, /tmp/expdp_02.dmp
OHOME=/u01/app/oracle/product/12.2.0.1/dbhome_1
ICHOME=/u01/app/oracle/product/instantclient_18_3
#
```



```

#-----#
# Expdp/Impdp Parameters #
#-----#
ENC_PASSWORD=<enc_password>
ENC_TYPE=AES256
#-----#
# ADB Properties #
#-----#
ADB_NAME=RCATP
ADB_PASSWORD=<enc_password>
CFILE=/opt/mv2adb/source/Wallet_RCATP.zip
#
#-----#
# Object Store Properties #
#-----#
OCI_REGION=us-phoenix
OCI_NAMESPACE=NAMESPACE_NAME
OCI_BUCKET=BUCKET_NAME
#
OCI_ID=mail@corp.com
OCI_PASSWORD=<enc_password>
#
PROXY_HOST=
PROXY_PORT=
PROXY_ID=
PROXY_PASSWORD=
#-----#
# End Of File #
#-----#

```

in this example we are going to move four database schemas: "SCHEMA\_1","SCHEMA\_2","SCHEMA\_3","SCHEMA\_4" and remapping the tablespaces to "DATA"

The export datapump dumps are encrypted using "AES256" algorithm.

Note: the "OCI\_PASSWORD" is the Auth Tokens password encrypted using mv2adb encpass

The screenshot shows the Oracle Cloud Identity console. At the top, it says 'ORACLE Cloud' and 'Germany Central (Frankfurt)'. The breadcrumb is 'Identity » Users » User Details » Auth Tokens'. The user profile shows a green circle with 'R' and 'ACTIVE' below it. The user's name is redacted with a black bar. The description is 'RAC Pack'. There are buttons for 'Create/Reset Password', 'Enable Multi-Factor Authentication', 'Edit User Capabilities', 'Unblock', 'Delete', and 'Apply Tag(s)'. The 'User Information' tab is selected, showing:
 

- OCID: ..jwjmdq (Show Copy)
- Created: Fri, 05 Jan 2018 03:55:51 UTC
- Multi-factor authentication: Disabled
- Email: [redacted] (Verification Pending)
- Status: Active
- Federated: No

 The 'Capabilities' section shows:
 

- Local password: Yes
- API keys: Yes
- Auth tokens: Yes
- SMTP credentials: Yes
- Customer secret keys: Yes

 The 'Resources' sidebar on the left lists: API Keys (1), Auth Tokens (1), SMTP Credentials (1), Customer Secret Keys (0), and Groups (2). The 'Auth Tokens' section is titled 'Auth Tokens' and shows 'Displaying 1 Auth Tokens'. A 'Generate Token' button is highlighted with a red box. Below it, one token is listed:
 

- OCID: ...ya76ya (Show Copy)
- Description: RC Auth Tokens
- Created: Thu, 19 Sep 2019 06:59:41 UTC

5. Issue the mv2adb "advisor" command to check on-premise for the suitability of migration
6. Issue the mv2adb "auto" command
7. Check the result with the mv2adb "report" command

## MV2ADB: Pre-requisite Configuration

The pre-requisites are dependent on which mv2adb options are used, so they are grouped together with each command option.

### MV2ADB – auto Operation

#### Pre-requisites

- Determine and test the Oracle Net Easy Connect descriptor to the on-premise database, provided in mv2adb.cfg parameter *DB\_CONSTRING*.

```
For example: $ sqlplus system@//sourcedb.example.com/SDB21.example.com
```

- Download the ATP-D database Oracle Net wallet and store in TNS\_ADMIN on the on-premise database host.
- Determine and test the Oracle Net Easy Connect descriptor to the ATP-D database, provided in mv2adb.cfg parameter *OC\_DB\_CONNSTRING*.

```
For example: $ sqlplus system@//exacs_public_ip.example.com/TB01.example.com
```

#### Pre-requisites when not using **--netlink**

- Install and configure OCI client on the on-premise database host. Refer to instructions at <https://github.com/oracle/oci-cli>
- Create directory on on-premise database host, large enough to store the Data Pump export dump files.
- Create an Oracle Object Storage bucket for transfer of the Data Pump dump files.

## Pre-requisites when using **--netlink**

- Configure and test Oracle Net configuration from OCI database to source on-premise database in preparation for automatic database link creation.

## **MV2ADB – expdp Operation**

### Pre-requisites

- Create directory on on-premise database host, large enough to store the Data Pump export dump files.
- Determine and test the Oracle Net Easy Connect descriptor to the on-premise database, provided in mv2adb.cfg parameter *DB\_CONSTRING*.

```
For example: $ sqlplus system@//sourcedb.example.com/SDB21.example.com
```

## **MV2ADB – impdp Operation**

### Pre-requisites

- Download the ADB database Oracle Net wallet and store in TNS\_ADMIN on the on-premise database host. Specify the location and name of the wallet zip file in the mv2adb.cfg parameter *ADB\_CFILE*.
- Test the Oracle Net Easy Connect descriptor to the ADB database, provided in mv2adb.cfg parameter *ADB\_NAME* using the following example:

```
For example: $ sqlplus system@<ADB_NAME>_higg
```

- If ATP-D, pre-create tablespaces. If the tablespace names for each schema being imported are different, use the mv2adb.cfg parameter 'REMAP' to remap the old to new tablespace names.
- Set mv2adb.cfg ICHOME parameter to match the Oracle software version of the database being imported into..

## **MV2ADB – createbucket, delbucket and listbucket Operations**

### Pre-requisites

- Install and configure OCI client on the on-premise database host. Refer to instructions at <https://github.com/oracle/oci-cli>

## **MV2ADB – putdump Operation**

### Pre-requisites

- Install and configure OCI client on the on-premise database host. Refer to instructions at <https://github.com/oracle/oci-cli>
- Create an Oracle Object Storage bucket for transfer of the Data Pump dump files.
- A list of all the Data Pump dump files (created at time of export) has been added to the mv2adb.cfg parameter *DUMP\_FILES*.

## **MV2ADB – getdump Operation**

### Pre-requisites

- Install and configure OCI client on the on-premise database host. Refer to instructions at <https://github.com/oracle/oci-cli>
- Create directory on on-premise database host, large enough to store the Data Pump export dump files.

## **MV2ADB – deldump and listdump Operations**

### Pre-requisites

- Install and configure OCI client on the on-premise database host. Refer to instructions at <https://github.com/oracle/oci-cli>

## **MV2ADB – advisor Operation**

### Pre-requisites

- Determine and test the Oracle Net Easy Connect descriptor to the on-premise database, provided in mv2adb.cfg parameter *DB\_CONSTRING*.
-

```
For example: $ sqlplus system@//sourcedb.example.com/SDB21.example.com
```

## **MV2ADB – report Operation**

### Pre-requisites

- Determine and test the Oracle Net Easy Connect descriptor to the on-premise database, provided in mv2adb.cfg parameter *DB\_CONSTRING*.

```
For example: $ sqlplus system@//sourcedb.example.com/SDB21.example.com
```

- Download the ADB database Oracle Net wallet and store in TNS\_ADMIN on the on-premise database host. Specify the location and name of the wallet zip file in the mv2adb.cfg parameter *ADB\_CFILE*.
- Test the Oracle Net Easy Connect descriptor to the ADB database, provided in mv2adb.cfg parameter *ADB\_NAME* using the following example:

```
For example: $ sqlplus system@<ADB_NAME>_high
```

## **MV2ADB: troubleshooting**

If for any reason, MV2ADB is failing, you could get diagnostics information from the log created under "/opt/mv2adb/out/log"

Note: if you need to debug datapump and you need METRICS=Y LOGTIME=ALL parameters, you could use '**dpdebug**' command option

## **REFERENCES**

[https://cloud.oracle.com/en\\_US/datawarehouse](https://cloud.oracle.com/en_US/datawarehouse)

<https://www.oracle.com/technetwork/database/database-technologies/instant-client/overview/index.html>

<https://docs.oracle.com/en/cloud/paas/autonomous-data-warehouse-cloud/user/load-data.html#GUID-1351807C-E3F7-4C6D-AF83-2AEFADE2F83E>

[https://cloud.oracle.com/en\\_US/atp](https://cloud.oracle.com/en_US/atp)

<https://docs.oracle.com/en/cloud/paas/atp-cloud/atpug/load-data.html#GUID-1351807C-E3F7-4C6D-AF83-2AEFADE2F83E>

[NOTE:2462677.1](#) - Oracle Autonomous Database Schema Advisor

Didn't find what you are looking for?