

Author – A.Kishore/Sachin
<http://appsdba.info>

Oracle AD Controller

AD Controller is ad utilities used to monitor/ control the workers execution.

- **How to run AD controller.**

Step 1 : Login as Applications Tier user & run the environment file.

```
su applmgr  
  
cd /do1/oracle/prodappl  
  
./APPSORA.env
```

Step 2 : Run the following AD controller command.

```
[applmgr@erp ~]$ adctrl
```

You will be prompted for the location of APPL_TOP location , password of APPLSYS and APPS. After providing the above information the AD controller menu will appear as shown below.

AD Controller Menu

1. Show worker status
2. Tell worker to restart a failed job
3. Tell worker to quit
4. Tell manager that a worker failed its job
5. Tell manager that a worker acknowledges quit
6. Restart a worker on the current machine
7. Exit

Author – A.Kishore/Sachin
<http://appsdba.info>

- **How to check the status of the workers?**

After adctrl is started, we have to choose the first option "Show worker status".

Enter your choice [1] :

| Worker | Control Code | Context | Filename | Status |
|--------|--------------|--------------|----------|-----------|
| 1 | Run | Generic R115 | US.msb | Running |
| 2 | Run | Generic R115 | US.msb | Running |
| 3 | Run | Generic R115 | US.msb | Assigned |
| 4 | Run | Generic R115 | US.msb | Completed |
| 5 | Run | Generic R115 | US.msb | Running |
| 6 | Run | Generic R115 | US.msb | Running |

Review the messages above, then press [Return] to continue. █

Please Note: If there is no session ,used by the workers, then the following message will appear:

Error: The FND_INSTALL_PROCESSES table does not exist.

This table is used for communication with the worker processes, and if it does not exist, it means that the workers are not running, because the ad utility has not started them yet.

We should check the file adctrl.log for errors.

This is because the FND_INSTALL_PROCESSES table is created when AD parallel jobs start (not the AD utility) and is dropped when the task is completed.

Author – *A.Kishore/Sachin*
<http://appsdba.info>

- **The meaning of each worker status.**

| STATUS | Description |
|----------------|--|
| Waiting | The worker is idle. |
| Assigned | A job was assigned by the manager to a worker but the worker didn't start the job. |
| Running | The worker is running a job. |
| Failed | The job failed due to an error. |
| Fixed, Restart | When a jobs restart after the error has been fixed (during this time the worker run the failed job). |
| Restarted | After the error has been fixed, the worker will have the status "Fixed, Restart" and after that "Restarted". (The status will not change to "Running") |
| Completed | The job was completed and the manager did not yet assigned another job to that worker. |

- **Database Processing Phases concept**

When a database patch/ operation will run, the tasks are divided into functions of the kind of modification. This is done by Oracle when the patch is created. Suppose a patch will create 4 tables and 4 sequences. In this case the patch driver contains 2 phases, one for tables creation and one for sequences creation. Because the sequences could be created in the same time, this will be done in parallel by using more workers.

Here are some Database Processing Phases:

seq = create sequence

tab = create tables, synonyms, grants privileges on tables

pls = create package specification

plb = create package body

vw = create views

- **Fixing a "Failed" worker**

When a job fails for the 1st time, the job is deferred at the end of the phase and another job is assigned to that worker.

Author – A.Kishore/Sachin
<http://appsdba.info>

If the job fails 2nd time,

- If the run time of the job was < 10 min => the job is deferred at the end of the phase and another job is assigned to that worker.

- if the run time of the job was >= 10 min => the job status will be "Failed".

If the job fails 3rd time => the job status will be "Failed".

To review the worker log information you have to check into

`APPL_TOP/admin/<SID>/log/adworkNNN.log`

Example: adwork001.log will be the log file for the worker number 1.

After fixing the error we have to start (if is not already started) AD Controller and to use the option 2 "Tell worker to restart a failed job". When prompted we have to specify the worker which must be restarted. If all the workers are failed, we can type all to restart all the workers.

- **Restarting a Failed Patch Process**

During a patch process (or adadmin process) if a job fails and cannot be restarted the patch must be restarted.

Here are the steps for doing this:

3. Tell worker to quit (for all workers) => to manually shutdown/ quit the workers
4. Tell manager that a worker failed its job
5. Tell manager that a worker acknowledges quit => the manager will stop, the AutoPatch will stop.

restart the patch

PLEASE NOTE: When the patch will restart all the information in the database about this session must be accurate.

Author – A.Kishore/Sachin
<http://appsdba.info>

- **How to determine if a process is Hanging or not**

- a) We can check the log file to see if some information is added or not to the log file.

- b) We can determine if the worker process is consuming CPU by issuing below command.

- ```
$ ps -eo pcpu,pid,user,args | grep workerid
```

- (c) We check if there are any child processes, which are consuming CPU by issuing following command:

- ```
$ ps -eo pcpu,pid,ppid,user,args | grep <Parent Process> | grep -v grep
```

- **Restarting a Hanging Worker Process**

- a) kill at the OS level the processes associated with the Hanging Worker Process.

- ```
$ kill -9 ProcessNumber
```

- b) fix the problem

- c) Restart the worker (or the job)

- **Restart an AD utility after a Node Crash**

- a) Start AD Controller

- b) Choose "4. Tell manager that a worker failed its job"

- c) Choose "2. Tell worker to restart a failed job"

- d) Restart the AD utility that was running when the node crashed.

- **Shutting down the Manager**

- a) Start AD Controller

- b) Choose "3. Tell worker to quit"

- c) Verify that no worker processes are running