

## **TUNING CONCURRENT MANAGER**

There are 5 ways to faster your CM

- 1.PMON cycle, queue size, and sleep time.
- 2.Purging Concurrent Requests
- 3.Adjusting the Concurrent Manager Cache Size
- 4.Analyzing Oracle Apps Dictionary Tables for High Performance
5. Number of Standard Managers

### **1.PMON cycle, queue size, and sleep time.**

The ICM performance is affected by the three important Oracle parameters

#### **PMON cycle**

This is the number of sleep cycles that the ICM waits between the time it checks for concurrent managers failures, are having problems with abnormal terminations.

#### **Queue Size**

The queue size is the number of PMON cycles that the ICM waits between checking for disabled or new concurrent managers. The default for queue size of 1 PMON cycle should be used.

#### **Sleep Time**

The sleep time parameter indicates the seconds that the ICM should wait between checking for requests that are waiting to run. The default sleep time is 60, but you can lower this number if you see you have a lot of request waiting (Pending/Normal). However, reducing this number to a very low value many cause excessive cpu utilization.

### **2.Purging Concurrent Requests**

One important area of Concurrent Manager tuning is monitoring the space usage for the subsets within each concurrent manager.

When the space in FND\_CONCURRENT\_PROCESSES and FND\_CONCURRENT\_REQUESTS exceed 50K, you can start to experience serious performance problems within your Oracle Applications. When you experience these space problems, a specific request called "Purge Concurrent Requests And/Or Manager Data" should be scheduled to run on a regular basis. This request can be configured to purge the request data from the FND tables as well as the log files and output files on accumulate on disk.

### **3.Adjusting the Concurrent Manager Cache Size**

Concurrent manager performance can also be enhanced by increasing the manager cache size to be at least twice the number of target processes. The cache size specifies the number of requests that will be cached each time the concurrent manager reads from the FND\_CONCURRENT\_REQUESTS table. Increasing the cache size will boost the throughput of the managers by attempting to avoid sleep time.

#### **4. Analyzing Oracle Apps Dictionary Tables for High Performance**

We can generate statistics that quantify the data distribution and storage characteristics of tables, columns, indexes, and partitions. The cost-based optimization approach uses these statistics to calculate the selectivity of predicates and to estimate the cost of each execution plan. Selectivity is the fraction of rows in a table that the SQL statement's predicate chooses. The optimizer uses the selectivity of a predicate to estimate the cost of a particular access method and to determine the optimal join order and join method.

We should gather statistics periodically for objects where the statistics become stale over time because of changing data volumes or changes in column values. New statistics should be gathered after a schema object's data or structure are modified in ways that make the previous statistics inaccurate. For example, after loading a significant number of rows into a table, collect new statistics on the number of rows. After updating data in a table, you do not need to collect new statistics on the number of rows, but you might need new statistics on the average row length.

It is also very important to run the request Gather Table Statistics on these tables:

**FND\_CONCURRENT\_PROCESSES**  
**FND\_CONCURRENT\_PROGRAMS**  
**FND\_CONCURRENT\_REQUESTS**  
**FND\_CONCURRENT\_QUEUES.**

#### **5. Number of Standard Managers**

Some of environments are copies of other environments, and you may find that the number of Standard Concurrent Managers are just 5. You can increase this to say 10 or 15, this will help the pending requests queue is not getting too long & the Conflict Resolution Manager will have less load.