# Cluster Health Monitor Readme

## I. INTRODUCTION

Oracle Cluster Health Monitor (CHM) - OS Tool (fomerly known as IPD/OS)

This tool is designed to detect and analyze operating system (OS) and cluster resource related degradation and failures in order to bring more explanatory power to many Oracle Clusterware and Oracle RAC issues such as node eviction. It continuously tracks the OS resource consumption at node, process, and device level. It collects and analyzes the cluster-wide data. In real time mode, when thresholds are hit, an alert is shown to the operator. For root cause analysis, historical data can be replayed to understand what was happening at the time of failure.

## II. INSTALLATION

On Linux, the tool requires Linux kernel version greater than or equal to 2.6.9 and architecture is x86. The install will work on x86_64 as well if the kernel is configured to run 32-bit binaries.

NOTE: If you already have IPD/OS installed on your cluster, you will have to remove the current version before installing the new version.

NOTE: When we refer to install home in the README, we mean /usr/lib/oracrf on Linux and "C:\Program Files\oracrf" on Windows.

For windows, Windows Server 2003 with service pack 2 is required.

1. In Linux, create user '<username>:<group>' (e.g. crfuser:oinstall) on all the nodes where tool is being installed. Make sure username's home is the same on all nodes. Typically, on most systems, you will issue:

   ```
   useradd -d /opt/crfuser -s /bin/sh -g oinstall crfuser
   ```

   while logged in as root.

   In windows, the administrator user should be used to install the tool and no additional user is required for that.

2. In Linux, setup passwordless ssh for the user created in step 1. Test that the '<username>' can ssh to all nodes (including the local node) using hostname (without domain) without password and without any user intervention like acknowledging prompts. This can be done by generating the key at one node for '<username>' adding public key to ~/.ssh/authorized_keys and copying the whole ~/.ssh directory over to all nodes under home dir of the '<username>'. Now, test by ssh'ing into all nodes from all nodes.

   For ssh-key details, please refer: http://fedoranews.org/dowen/sshkeys/

   In case of windows, make sure that UNC (uniform naming convention, such as \\Server\Volume\File) path is configured on all nodes where OSTool is to be installed.

3. If you have a previous install of this tool, delete it from all nodes. Login as privileged user (root on Linux/admin on Windows):

   a. Disable the tool

   ```
   "/etc/init.d/init.crfd disable"
   ```

"stopcrf" from a command prompt on Windows.

b. Uninstall

```
"/usr/lib/oracrf/install/crfinst.pl -d" on Linux
"perl C:\programm files\oracrf\install\crfinst.pl -d" on Windows
```

c. Make sure all BDB databases are deleted from all nodes.

d. Manually delete the install home if it still exists.

4. Login as '<username>' on Linux. Login as admin user on Windows. Unzip the crfpack.zip file.

5. Run crfinst.pl (see below for usage details) script on a node with desired node list, specified as comma separated list, for cluster-wide install. You will find this script in the install subdirectory. An example invocation will look like this:

   ```
   $ ./crfinst.pl -i node1,node2,node3 -b /opt/oracrfdb -m node1
   ```

6. Once the step 5 finishes, it will instruct you to run crfinst.pl script with -f and optionally -b <bdb location> on each node while logged in as root/admin to finalize the install on that node.

7. Enable the tool on all nodes. Once the finalize operation is complete, run the following while logged in as privileged user:

   ```
   # /etc/init.d/init.crfd enable, on Linux
   > runcrf, on Windows
   ```

   to enable the tool.

DO NOT bypass any of above steps or try other ways to install because the daemons will not work correctly, and you will not be supported.

Also remember that only root/admin can de-install. DO NOT attempt re-install without doing de-install with 'crfinst.pl -d' option.

All node names in the nodelist should be specified without the domain name to maintain uniformity and ease of use. Entering node names with domain names may lead to a failed install.

```
Usage: crfinst.pl -a [<nodelist>]
                  -c [<nodelist>]
                  -d
                  -f [-b <bdb loc>]
                  -g <ui install dir>
                  -h
                  -i <nodelist> -b <bdb loc> [-m <master>]
                  -N ClusterName
```

-a : Add nodes for OS resource metrics collection by the tool from a configured node. The supported mechanism for node addition is to stop IPD/OS on all the nodes, add a new node and then restart IPD/OS on all nodes.

-b : Specify the path <bdb loc> where a Berkeley DB can be created to store OS metrics. This location MUST be outside of the location

```
            where you unzipped the ZIP file because all the directories
            under that location which were created by unzip will be removed.
            BDB files can be kept as it is for later usage. The location should
            be a path on a volume with at least 5GB per node space available
            and writable by privileged user only. It cannot be on root
            filesystem in Linux. This location is required to be same on all
            hosts. If that can not be done, please specify a different location
            during finalize (-f) operation on each host, following the above
            size requirements. The path MUST not be on shared disk. If a shared
            BDB path is provided to multiple hosts, BDB corruption will happen.

       -c : Perform the user equivalence and other pre-install checks for the
            nodes.

       -d : De-install software on the node where the script is run (Only
            'root/admin' can perform this operation).

       -f : Finalize the install on a node (Only 'root/admin' can perform this
            operation).

       -g : Standalone UI installation on current node. Oracle recommends to
            not install the UI on the servers. You can use this option to
            install the UI-only client on a separate machine outside of
            cluster.

       -h : Show this help

       -i : Install software on nodes listed in the <nodelist>.

       -N : Specify the name of the cluster

       <bdb loc> : Path for Berkeley DB to store OS metrics.
       <master>  : Name of the node for master logger daemon.
       <nodelist>: Comma separated list of nodes to install software on
       <ui install dir> : Path for the standalone UI install directory.
```

## III. THE DAEMONS

The OS Tool consists of three daemons: ologgerd, oproxyd and osysmond. There is one ologgerd master daemon on only one node in the installed set of nodes and there is one osysmond on every node. If there is more than 1 node in the installed set of nodes, another node is chosen to house the standby for the master ologgerd. If master daemon suffers a death (because daemon is not able come up after a fixed number of retries or node where master was running is down), standby takes over as master and selects a new standby. Master manages the OS metric database in Berkeley DB and interacts with the standby to manage a replica of the master OS metrics database.

osysmond is the monitoring and OS metric collection daemon that sends the data to ologgerd. ologgerd receives the information from all the nodes and persists in a Berkeley DB based database.

oproxyd is the proxy daemon which handles connections on the public interface. If the tool is configured with private node names, only orpoxyd is listening on the public interface for external clients (like oclumon and crfgui). This serves as a security measure against attacks on ologgerd master daemon. It runs on all the nodes and is highly avaliable.

## IV. USING THE TOOL

IV.1 Start the daemons

The install does not enable/run the daemons by default.

On linux, after the finalize operation has finished on all nodes, enable the daemons on each node by running:

```
# /etc/init.d/init.crfd enable
```

On windows, type 'runcrf' from windows command prompt.

This needs to be executed on all nodes. This will make init start the osysmond daemon if it is not running. init also restarts the osysmond daemon if it dies.

osysmond will spawn the master ologgerd if the node is master node. It will start replica ologgerd daemon if the node is configured as replica.

If ologgerd dies for some reason, osysmond restarts it by trying upto a maximum number of times, after which replica takes over.

oproxyd daemon is spawned by the sysmond daemon.

IV.2 Run the GUI

You need to install the GUI with -g separately. You can invoke the GUI with:

```
$ crfgui
```

from any node which has a osysmond running. Oracle recommends you run the gui from outside the cluster you are monitoring. If you are running GUI from a node outside of the set of installed nodes, you MUST use the '-m' option as below,

```
$ crfgui -m <nodename>
```

where <nodename> is any node where the tool is installed and is part of the cluster install provided to crfinst.pl script.

This command opens up a window showing the Cluster View. Cluster view shows top consumer processes on each node for CPU%, memory, file descriptors (FDs), and thread resources. This is followed by system wide statistics for each node, which includes statistics such as CPU%, CPU Queue Length, Memory free, IO operations, and read and write rates, Network read and write rates, number of processes, number of FDs, number of disks and NICs seen on the node.

The default refresh rate of the GUI is 1 second.  To change the refresh rate, use -r with number of seconds (i.e. -r 5 for a 5 second refresh rate)

```
$ crfgui -r 5 -m <nodename>
```

Inside the GUI, you can use 'node <nodename>' command to open a view which gives more detailed information about a node in a Node View. Alternatively, you can double click a node to get the Node View. A Node View presents the detailed statistics on interesting processes, disks and NICs based on heuristics.

Both Cluster View and Node View show text alerts at the bottom. These alerts are generated when the sampled value of a resource metric either goes above or falls below a threshold that could lead to potential problems on the node and hence on the cluster.

One can drill down the details on partitions for the disks listed in Node View by double clicking the disk. The information is presented in the Disk View. The Disk View provides a detailed list of partitions and corresponding stats for each one of them. It also clearly marks partitions which are found to belong to certain categories like Voting/OCR/SWAP/ASM disks.

Invoking the GUI with '-d' option starts it in historical mode.

```
$ crfgui -d  "<hh>:<mm>:<ss>" -m <nodename>
```

where -d is used to specify hours (<hh>), minutes (<mm>) and seconds (<ss>) in the past from the current time to start the GUI from e.g. crfgui -d "05:10:00" starts the GUI and displays information from the database which is 5 hours and 10 minutes in the past from the current time.

Invoking the GUI with '-i' option provides the same shell at the command prompt as is seen in the GUI windows with a prompt of 'toprac>'. You can use '?' at this prompt to get detailed information about available commands and options.

Alternatively,

```
$ crfgui -h
```

provides help about more options.

When using crfgui, there are a few interactive mouse operations to simplify the operation:

* double clicking on a cell with a node name, will send out a request for a node view.
* double clicking on a cell with a device name, will send out a request for a device view.
* clicking on column header, it will sort rows by the values in this column in descending order.  Any other sort criterion will be reset.
* shift-clicking on column header, will sort rows by the values in this column in ascending order.  Any other sort criterion will be reset.
* control-clicking on a column header has several functions:

  a) if this column has sort activated, it will toggle the sort order.
  b) if another column has sort activated, it will add a sub-sort order to a column by preserving the existing sorts.

A sort order in a column is indicated by a small triangle pointing up or down in the direction of the sort.

## IV.3 Oclumon

A command line tool is included in the package which can be used to query the Berkeley DB backend to print out to the terminal the node specific metrics for a specified time period. The tool also supports a query to print the durations and the states for a resource on a node during a specifed time period. These states are based on predefined thresholds for each resource metric and are denoted as red, orange, yellow and green indicating decreasing order of criticality. For example, you could ask to show how many seconds did the CPU on node "foo" remain in RED state during the last 1 hour. Oclumon can also be used to perform miscellaneous administrative tasks such as changing the debug levels, querying version of the tool, changing the metrics database size, etc.

```
$ oclumon -h
```

provides information about this tool and its options.

To get detailed information about all nodes in the cluster use verbose (includes process and device stats) mode:

```
$ oclumon dumpnodeview -v -allnodes -last "00:30:00"
```

which will dump all stats for all nodes for last 30 minutes from the current time.

To limit the scope of dump, use start and end times:

```
$ oclumon dumpnodeview -allnodes -s "2008-11-12 12:30:00"
                                 -e "2008-11-12 13:30:00"
```

which will dump stats for all nodes from 12:30 to 13:30 on Nov 12th, 2008.

Note that you don't specify the timezone in these clocks. To find the timezone on the servers in the cluster, use oclumon dumpnodeview -allnodes. All your time specification should be in that timezone, without actually referring the timezone string (like UTC or PST8PDT) in the time.

Alternatively, specify a TZ variable in the environment profile (.login/.profile/.bash_profile) of 'oracle' or 'crfuser' user and the daemons will use that. Then, a uniform TZ will be used by clients and servers.

You can use '-n' to specify a node name:

```
$ oclumon dumpnodeview -v -n mynode -last "00:10:00"
```

will dump all stats for 'mynode' for last 10 minutes.

To use oclumon to query for alerts only, use the '-alert' option:

```
$ oclumon dumpnodeview -v -allnodes -alert -last "00:30:00"
```

which will dump all records for all nodes for last 30 minutes, which contains at least one alert. It should be noted that if '-v' is not used you may see records which apparently don't have any system wide alerts. (NOTE: system wide alerts are the alerts on the data that is shown in dumpnodeview output without the -v option.) The alert in those cases may come from one of processes or the devices.

## IV. Misc. debug utilities

Included in the package are tools, which can help manage the Berkeley DB. These tools are supported only when you are working with Oracle Support. The tools are located in bin directory of the install home. The BDB tools have names starting with db_. Many of these tools will require you to setup the environment manually.

ologdbg: This utility provides a debug mode loggerd daemon. It is used to run loggerd in a special mode where an existing BDB database is specified on the command line and the daemon starts to serve oclumon and crfgui clients against that database. This can be used to analyze a complete backup of OS Tool BDB offline on a separate machine. It is useful for scenarios where BDB expiration may be an issue and the full BDB backup is made available offline for analysis. Note that there are restrictions on running debug mode loggerd and you should familiarize yourself with those before running loggerd in this mode. To get more information on this utility, run 'ologdbg -h' from command line.

## V. FAQ/ Known Limitations

>> How does the tool manage timezone? What is the Clock in oclumon output?

All Cluster Health Monitor  daemons use the TZ variable which is retrieved in the following
order:

'oracle' user's profile, 'crfuser' user's profile, system wide TZ environment variable, UTC if none of the previous sources yielded a valid value for TZ.

The "Clock:" in the oclumon output is printed in the timezone which the master daemon is running with. As mentioned above, you can change that by setting TZ in one of two places (TZ variable in the environment profile of 'oracle' or 'crfuser' user).

>> What does CPU value in GUI represent?

The CPU value, whether at system level or process level, represents the CPU usage in the sampling interval, which dynamically changes on the collecting node. This sampling interval is independent of the refresh rate of the GUI. CPU values are based on a cpu for processes i.e. if a process consumes all of one CPU on a 4 CPU system , the value reported is 100% for this process, and aggregated system wide.

>> Why do we need to be root/admin to deploy this tool?

Many OS metrics (e.g. number of open FDs for a process) can not be gathered as a normal user. Moreover, we need to tie in with initializing daemons to bring the daemons up at boot time and use their re-startability to make the sysmond daemon highly available.

>> How to interpret the alerts on memory metrics for Oracle processes?

Current version of the tool treats the Oracle processes same as any other process for alerting memory consumption. So if there is large SGA allocated to the instance you may see continuously more alerts because the SGA remains pinned for the Oracle processes.

>> How much history of OS metrics is kept in Berkely DB?

By default the database retains the node views from all the nodes for the last 24 hours in a circular manner. However this limit can be increased to 72 hours by using oclumon command : 'oclumon manage -bdb resize 259200'.

>> What rate does the tool sample the data?

The sampling rate of the tool depends on the currently active processes and the devices on the system. Up to a total of 1000 active processes and disks with ideal system, the sampling interval is approximately 1 second.

>> What is "MyCluster" in the GUI title?

It is the default name of the cluster this GUI is running for. It can be chosen using 'crfinst.pl -N' during install as mentioned in section-II.

>> Why does the tool not mark special disk devices correctly on 64-bit?

It is a known issue with 64-bit kernels running 32-bit binaries. This will be fixed when we have a native 64-bit port of the tool. You can refer bug #7651604 for more details.

>> How do I choose BDB location?

The BDB database location is recommended to be on local storage. Strictly observe the following rules when selecting BDB location:

```
    * It must not be on the root FS.
    * It must not be on shared disk.
    * It must meet the size guidelines specified above in usage.
```

If you insist on putting it on shared disk, please make sure that all hosts get a different path for this location in the finalize(-f) step. If you don't follow this guideline, you may end up with a corrupt database and loggerd daemon may not come up. The installer will check for common shared storage FSs and reject the install.

In Linux, if the node only has root filesystem, an option is to install a loopback filesystem. An example of doing it would be:

```
    a) # dd if=/dev/zero of=<bdb_file> count=<5000*N> bs=1M
    b) # /sbin/mke2fs -F -j -L "ipdosbdb" <bdb_file>
    c) # mkdir <bdb_loc>
    d) # mount -o loop <bdb_file> <bdb_loc>
    e) add the line in /etc/fstab
       <bdb_file>      <bdb_loc>    ext3   rw,loop=/dev/loop0 0 0

    <N>        : The number of nodes IPD will run on.
    <bdb_file> : The file that will be mounted as a loop device.
    <bdb_loc>  : The location where BDB files become visible. This
                 is the location that you will specify during IPD-OS
                 install with -b argument.
```

Adjust /dev/loop0 to a free loopback device node on your machine.

Note that you will need to do this on all cluster nodes.

>> What does the PRIORITY of a process mean?

The linux priorities range from -20 to 19. There is static priority and there is nice value. We report the dynamic nice value only. We report +ve priority in the range 0-39 for non-RealTime processes. Processes in the RT class are reported to have priorities from 41 to 139. This way a consistent "high number means high priority" priority is reported across platforms. The math used is (19 - nice_val) for non-RT and (40 + rtprio) for RT processes, where nice_val and rtprio are corresponding fields in the /proc/<pid>/stat. This is consistent with the Unix utility 'ps'. Also note that, Unix utility 'top' reports priority and nice as two different values, and are different from what IPD-OS reports.

>> Why is procfdlimit is shown as 1024 but openfds for the process are higher?

When we can not find /proc/<PID>/limits, which happens to be the case on some kernel versions, we fallback to limits governed by sysconf. Note that the process or shell that started the process has changed the limit dynamically in this case.

>> Some disk devices are missing from the device view.

This can happen for two reasons:

* We only collect and show top (decided by wait time on the disk) 127 devices in the output. OCR/VOTING/ASM/SWAP devices are pinned forever. So, the missing device may have just fallen off of this list if you have more than 127 devices (luns).

* The disks were added after the Cluster Health Monitor was started. In this case, just restart the Cluster Health Monitor stack. Future versions of Cluster Health Monitor will be able to handle this case without restart.