An Oracle White Paper
September 2009

Oracle Enterprise Manager Implementation

Managing Many Environments as One

Disclaimer

The following is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, and timing of any features or functionality described for Oracle’s products remains at the sole discretion of Oracle.

Executive Overview

Enterprise Manager provides many ways to implement and proactively monitor your critical environments. With all applications it is best to plan before an implementation is started. This document will outline specifics around planning, implementation and target management used by Oracle Applications IT. These practices have proven to streamline process and assist management as well as Database and Application administrators in ensuring maximum uptime for critical environments.

Introduction

With the execution of suggested practices in this document, the implementer can expect to implement Enterprise Manager, discover various targets, implement suggested practices and in the end, have a fully operational Enterprise Manager model that will provide proactive notifications on critical environments as well as assist in day to day management.

This document will also outline how to use various features in Enterprise Manager that will streamline the management and provide suggestions on how to interact with external custom tools (e.g., outage task tracking to assist in the management of Enterprise Manager.

When deploying, there is a specific flow that should be used. This document follows that flow once the Enterprise Manager infrastructure is installed. The following is an example of that flow that in general, should be used for all target types discovered:

- Install the agent
- Discover the target
- Setup Standard & User Defined Metrics (UDMs)Metrics
- Setup & apply metrics templates
- Setup required notification groups

- Implement Notification Rules
- Update roles and privileges with newly discovered targets

By following the above flow for each target type discovered, you will ensure that all critical targets are discovered and monitored properly with accurate notifications.

As well, this basic template will assist ensure standards are maintained when expanding the enterprise.

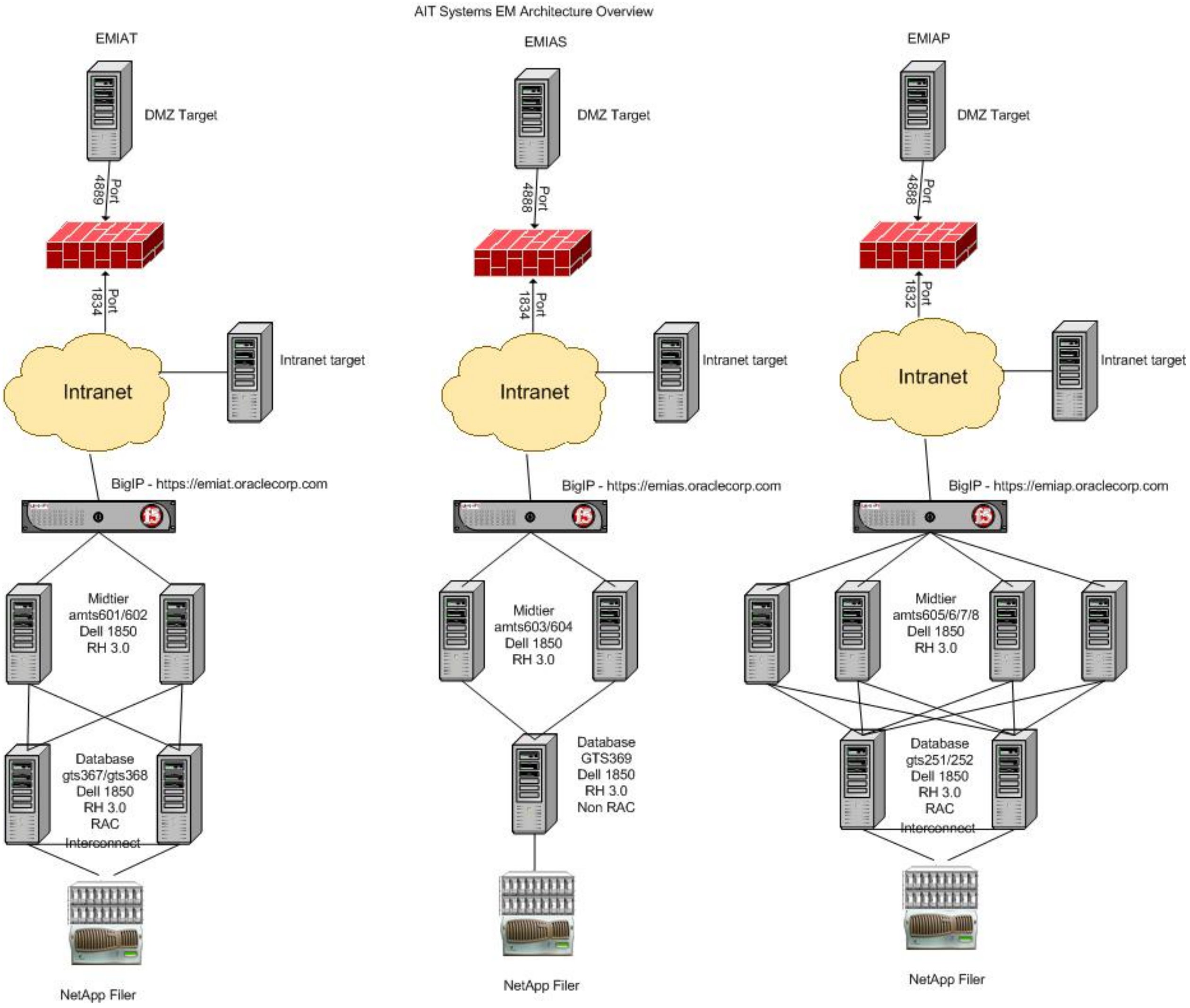
Installation

Before the implementation of Enterprise Manager begins, it is important to review requirements around capacity. The following details are from a real world implementation and have proven to be very reliable for an environment that currently supports over 5000 targets.

Also when implementing Enterprise Manager in is important to review architecture before the installation starts to ensure that the architecture you deploy today will support tomorrows requirements. Finally, it is also important to ensure that you have proper test and stage environments where you can validate new versions or patches to the application or database repository. In our case, we follow a standard architecture setup, Test, Stage and Production. Additional information specific to architecture considerations can be found in the Oracle® Enterprise Manager Advanced Configuration 10*g* Release 5 (10.2.0.5)

When we created the test, stage and production environments, we discovered a subset of targets on the test and stage environments with their own agents on each target discovered. That is to say, on the non production environments, we installed 3 agents, one for test, one for stage and one for production on separate ports. This enabled us to discover the same targets on all consoles with no conflicts.

The following is an example of our architecture:


AIT Systems EM Architecture Overview
EMIAT
DMZ Target
Port 4889
Port 1834
Intranet
Intranet target
BigIP - https://emiat.oraclecorp.com
Midtier amts601/602 Dell 1850 RH 3.0
Database gts367/gts368 Dell 1850 RH 3.0 RAC
Interconnect
NetApp Filer
EMIAS
DMZ Target
Port 4888
Port 1834
Intranet
Intranet target
BigIP - https://emias.oraclecorp.com
Midtier amts603/604 Dell 1850 RH 3.0
Database GTS369 Dell 1850 RH 3.0 Non RAC
NetApp Filer
EMIAP
DMZ Target
Port 4888
Port 1832
Intranet
Intranet target
BigIP - https://emiap.oraclecorp.com
Midtier amts605/6/7/8 Dell 1850 RH 3.0
Database gts251/252 Dell 1850 RH 3.0 RAC
Interconnect
NetApp Filer

Environment Capacity

In most organizations, there is a requirement to support a test, stage and production environment. The following is an example of this setup:

- Test – This environment is, for the most part, used by the team that manages Enterprise Manager to validate patches and execute upgrades to Enterprise Manager. This environment infrastructure should mirror the production environment so that proper comparison of feature can be complete in test with the same results as you would expect in production. This environment should include the discovery of only the test environments against a test agent on the give targets. The specifics around the agent deployment in support a setup to the test console will be explained in greater detail in the agent deployment section of this document. No notifications to the wider team should be sent from the test environment. We would recommend the following hardware setup that would again, mirror production. With this setup, there would be a load balancer in front of the midtiers to ensure maximum availability:

  - Midtier: 4 (CPU: 2; Mem: 8 GB) - OS: Oracle Enterprise Linux (OEL) 5
  - Database: 2 (CPU: 2: Mem: 16 GB) – Real Application Cluster (RAC) - OS: Oracle Enterprise Linux 5

- Stage – The stage environment is used by the wider technical group to review features, validate patches and upgrades against features used by the various personnel in the group. As an example, a new feature is introduced in the latest version of Enterprise Manager, the stage environment is where the group team members would exercise this new feature against the test environments to ensure that it works as expected before it goes to production. This environment should only have the test environments discovered for all applications managed that wouldn't be effected in the production console specific to metrics. The specifics around the agent deployment in support a setup to the stage console will be explained in greater detail in the agent deployment section of this document. We would recommend the following hardware setup that would not mirror production. Again, this environment would primary be used to validate application functionality by the wider technical team that uses Enterprise Manager on a regular basis. With this setup, there would be a load balancer in front of the midtiers to ensure maximum availability:

  - Midtier: 2 (CPU: 2; Mem: 8 GB) - OS: Oracle Enterprise Linux 5

    - Database: 1 (CPU: 2: Mem: 16 GB) - OS: Oracle Enterprise Linux 5

- Production – The production environment is used for all critical proactive monitoring of all production environments and non production environments. Again, the hardware should reflect the setup in test. Hardware should be as follows:
    - Midtier: 4 (CPU: 2; Mem: 8 GB) - OS: Oracle Enterprise Linux 5
    - Database: 2 (CPU: 2: Mem: 16 GB) – Real Application Cluster (RAC) OS: OEL 5

Additional information specific to capacity can be found in the Oracle® Enterprise Manager Grid Control Installation Guide 10*g* Release 5 (10.2.0.5.0) in section 3, Preinstallation Requirements.

Finally, additional information specific to architecture can be found in the Oracle® Enterprise Manager Advanced Configuration 10*g* Release 5 (10.2.0.5) under section 11, Sizing and Maximizing the Performance of Oracle Enterprise Manager.

Database Repository

When implementing Enterprise Manager it is critical to review proper sizing for the database repository. Enterprise Manager creates the “sysman” schema automatically when installing the Oracle Management Server (OMS) or midtier. We chose to install the OMS on a different server than the database repository to simplify configuration management.

Our current database version is 11g (11.1.0.6). It is a Real Application Cluster (RAC) environment with the use of NAS disk (Network Application) subsystem. Our repository size to support over 5000 targets is 118 GIG, at this time. There is no purging of data beyond the normal cleanup schedule, standard within Enterprise Manager. When implementing on a RAC environment additional considerations should be reviewed in the Oracle® Enterprise Manager Grid Control Installation Guide 10*g* Release 5 (10.2.0.5.0) in section 8, Prerequisites for Installing Enterprise Manager Grid Control on Oracle RAC.

Management Server

When implementing the midtier for any Enterprise Manager environment, load balancing routers in front of the midtiers for a given Enterprise Manager environment should be considered. Behind the load balancer, multiple midtiers should be installed to ensure maximum uptime for the Enterprise Manager console. All OMS midtiers where installed under an "oracle" user at the host Linux level.

In our case, for production, we have 4 midtiers behind a F5 BigIP Router that will route traffic in a round robin fashion to each midtier to ensure maximum uptime.

As well, we have implemented a "shared" upload area that is mounted via NFS mount across all midtiers for a given environment to ensure that, in the event any a midtier goes down, the XML files received from the agents that contain the critical status and metrics data from the agent, is processed in a timely manner. If the "shared" upload mount isn't used and any one midtier server goes down, you would effectively have "orphaned" XML files that would sit in limbo unto the midtier server in question is brought back on line. This has a negative effect on metrics and alerts as well as the alerting latest status of targets in the repository for Enterprise Manager and therefore, will impact the timely alerts required to ensure maximum uptime.

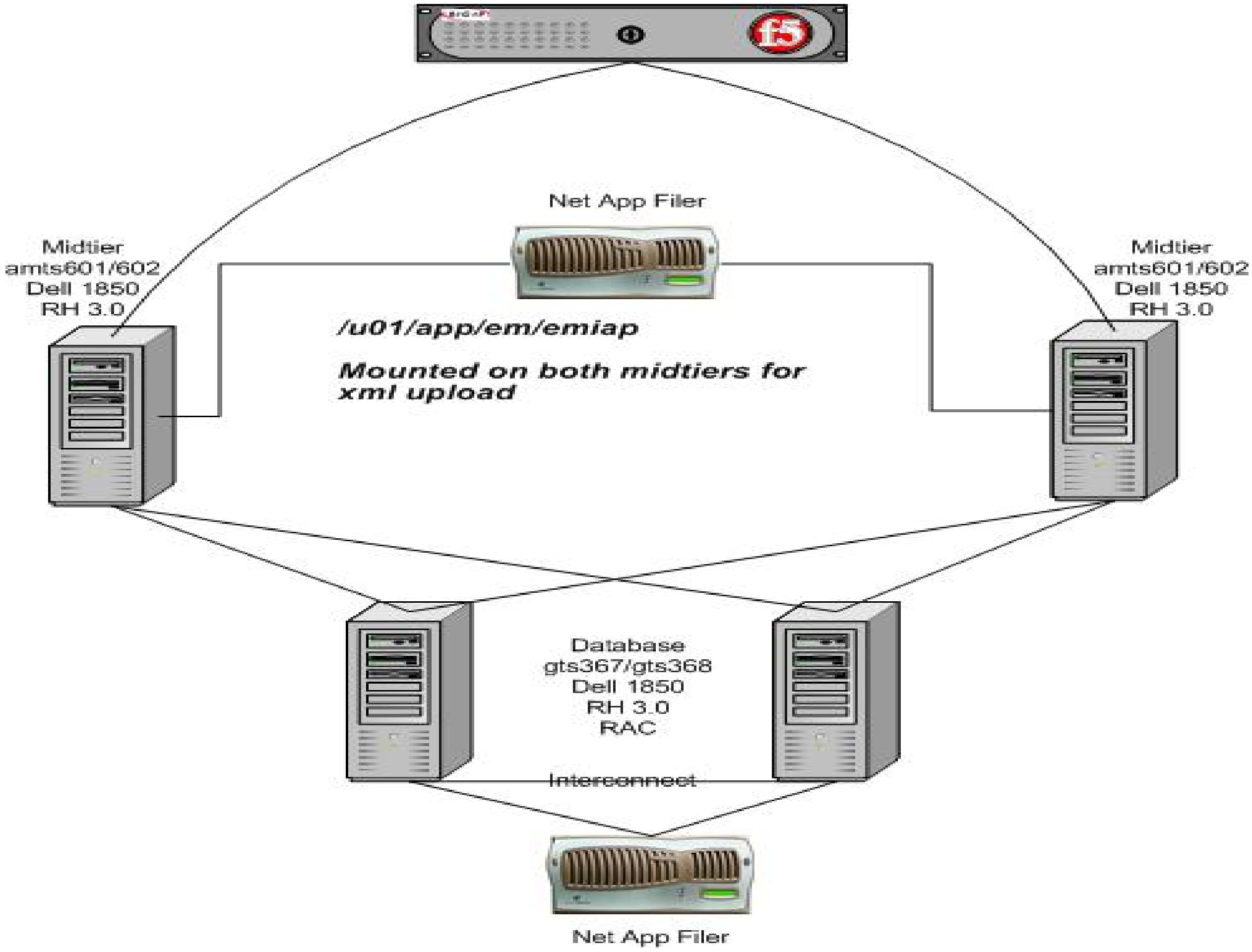
The following is an example of the "emoms.properties" file used with multiple midtiers using a shared upload area:

```
oracle.sysman.emSDK.svlt.ConsoleServerName=amts125.us.oracle.com\:4889_Management_Service

oracle.sysman.eml.mntr.emdRepConnectDescriptor=(DESCRIPTION\=(ADDRESS_LIST\=(ADDRESS\=(PROTOCOL\=TCP)(HOST\=dbs510-crs.us.oracle.com)(PORT\=1521))(ADDRESS\=(PROTOCOL\=TCP)(HOST\=dbs509-crs.us.oracle.com)(PORT\=1521)))(CONNECT_DATA\=(SERVICE_NAME\=emiap.us.oracle.com)))

ReceiveDir=/u01/app/em/emiap

oracle.sysman.db.isqlplusWebDBAUrl=%ISQLPLUS_PROTOCOL%\://%ISQLPLUS_HOST%\:%ISQLPLUS_PORT%/isqlplus/dba/dynamic

em_oob_shutdown=false
```

```
oracle.sysman.db.isqlplusUrl=%ISQLPLUS_PROTOCOL%\://%ISQLPLUS_HOST%\:%ISQLPLUS_PORT%/isqlplus/dynamic
oracle.sysman.eml.mntr.emdRepUser=sysman
proxyPort=80
em_from_email_name=_NOT_AVAILABLE_
oracle.sysman.emkeyfile=/u01/app/oracle/product/oms10g/sysman/config/emkey.ora
oracle.sysman.eml.mntr.emdRepPort=1521
oracle.sysman.emSDK.svlt.ConsoleServerPort=4889
oracle.sysman.eml.mntr.emdRepPwdSeed=6992450084234464576
em_from_email_address=_NOT_AVAILABLE_
em.loader.threadPoolSize=2
oracle.sysman.eml.mntr.emdRepServer=dbs509-crs.us.oracle.com
em_email_gateway=
oracle.sysman.emSDK.svlt.ConsoleServerHTTPSPort=4888
em_oob_startup=false
oracle.sysman.eml.mntr.emdRepSID=NO_VALUE
oracle.sysman.eml.mntr.emdRepPwdEncrypted=TRUE
oracle.sysman.eml.mntr.emdRepPwd=8dea07703da4a212
oracle.sysman.eml.maxInactiveTime=125
proxyHost=www-proxy.us.oracle.com
emdrep.ping.pingCommand=/bin/ping -c 3 -w 30 <hostname>
LargeRepository=false
dontProxyFor=*.oracle.com, *.oraclecorp.com
oracle.sysman.emSDK.svlt.PublicServletEnabled=true
em.oms.dumpModules=omsThread,repos
oracle.sysman.emSDK.svlt.ConsoleServerHost=amts125.us.oracle.com
em_oob_crash=false
```

```
em.loader.maxDirectLoadFileSz=80485760
oracle.sysman.emRep.dbConn.statementCacheSize=50
em.ip.ui.enable=true
em.loader.coordinationMethod=sharedFilesystem
em_email_address=
```

The following is an example of a shared upload area:



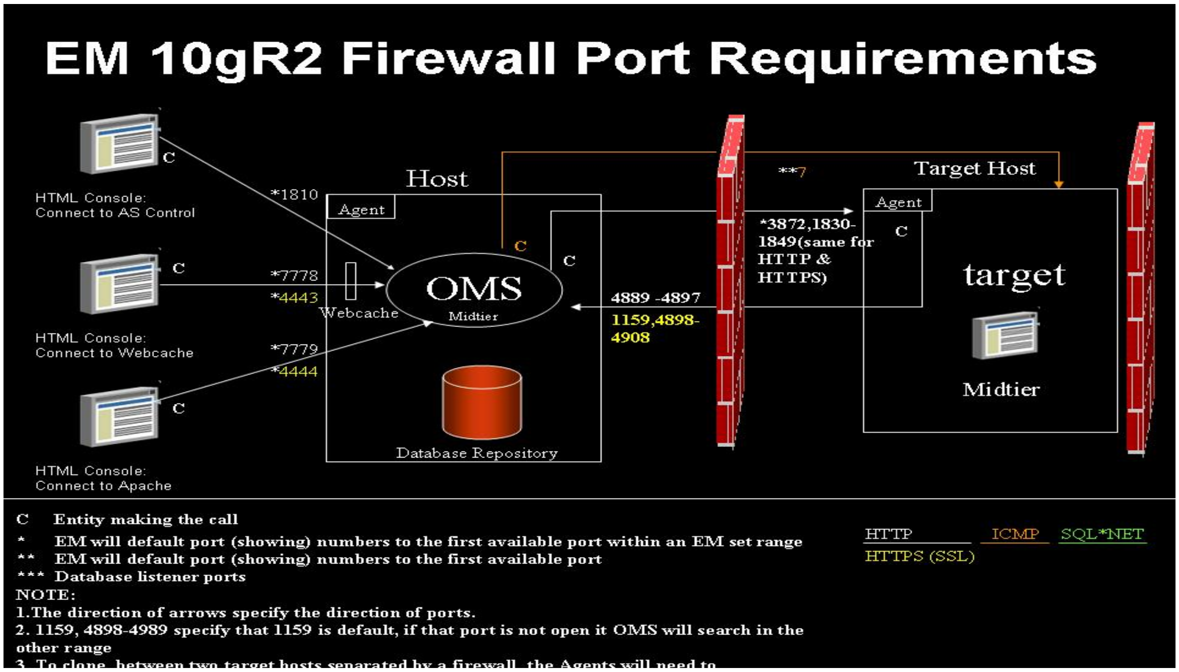
Additional information around the installation of the Enterprise Manager application can be found in the Oracle® Enterprise Manager Grid Control Installation Guide 10*g* Release 5 (10.2.0.5.0) under section 7, Installing

Enterprise Manager Grid Control. This section of the document also outlines the type of installation specific to an new installation vs. an upgrade of an existing environment. Also, it outlines the process for installation when using an existing database. Finally, this section also outlines how to add additional management servers.

As well, additional information can be found specific to multiple management server installation in the Oracle® Enterprise Manager Advanced Configuration
10*g* Release 5 (10.2.0.5) under section 3, Grid Control Common Configurations.

Firewall Considerations

From a firewall perspective, it is important to review requirements all requirements early to ensure there are no issues during implementation and as well, no issues with monitoring after implementation. In our case, we have midtier applications on the intranet as well as the internet and therefore, we must configure beacons for ASLM to ensure proper access. This is done via proxy setup. Additional information on the proxy setup as well as other topics can be found in the Oracle® Enterprise Manager Advanced Configuration
10*g* Release 5 (10.2.0.5) under section 6, Configuring Enterprise Manager for Firewalls. The following is an example of a firewall setup:



EM 10gR2 Firewall Port Requirements

C Entity making the call
* EM will default port (showing) numbers to the first available port within an EM set range
** EM will default port (showing) numbers to the first available port
*** Database listener ports
NOTE:
1.The direction of arrows specify the direction of ports.
2. 1159, 4898-4989 specify that 1159 is default, if that port is not open it OMS will search in the other range
3. To clone, between two target hosts separated by a firewall, the Agents will need to

HTTP ICMP SQL*NET
HTTPS (SSL)

Agent

The agent resides on the client or target side and gathers data specific to the real-time status as well as historical data specific to all targets on a given host. It is critical that the host remains up to ensure proper proactive notifications. On scheduled outages at the OMS level, the XML files that hold the real-time and historical data will backup on the target side and when the OMS is brought back up, the OMS midtier will process the targets XML files. Depending on the outage, this could take some time.

Outside of the XML files, the OMS pings the agent and host that the agent resides on a regular basis to ensure that the agent and host are reachable. There could be a case where the agent would be down but the host and it's targets would be up. In this case, it is critical that notifications are setup so that a team member is notified and the agent is restarted.

As well, we have installed a number of agents at the target level that address the test, stage and production OMS. This enables us to completely test and validation patches against the test and stage environment before we go to production.

From an agent owner perspective, it is good to assign the agent owner to a specific user rather than to the "oracle" user, as an example. This will assist with auditing and ensure a proper responsibility split for security purposes. This is not always required as in many cases the database administrator is also the person who would be managing Enterprise Manager.

From an agent resource overhead perspective, we have seen no negative impact with having one or many agents on a single host. That being said, if you have a small enterprise and have hosts that limited resources, it is a good idea to setup standard metrics to monitor the resources used by the agent and notify required individuals to ensure that the agent has no negative impact on the host. As the agent gathers info at the host level specific to CPU and memory, proactive monitoring the agent is important. All metrics specific to the agent can be setup under the "Metrics & Policy" section. As well, one can use the template feature to setup the same metrics across all agents in your EM Enterprise.

There are two ways that we have agents deployed, a stand alone and shared agent. The following are details around each deployment:

Stand Alone Agent

In a stand alone agent setup, the entire setup of the agent is limited to a given host and can only be administered from that host. As indicated, we have installed a number of agents at the target level to address the requirement to communicate with the test, stage and production OMS, depending on the targets on a given hosts. The setup for each agent in a stand alone setup is as follows:

- Test – With this setup, we have specified a port for the test agent to listen on which doesn't conflict with other ports being used on the host.
    - Binary Location: /u01/app/em/test/agent10g
    - Port Used: 1834
- Stage - With this setup, we have specified a port for the test agent to listen on which doesn't conflict with other ports being used on the host.
    - Binary Location: /u01/app/em/stage/agent10g
    - Port Used: 1836
- Production - With this setup, we have specified a port for the test agent to listen on which doesn't conflict with other ports being used on the host.
    - Binary Location: /u01/app/em/prod/agent10g
    - Port Used: 1832

The installation of the stand alone agent was completed via command line with the following setup requirements:

- Setup:
    - Log in to the host target to be installed as your userid "aitem".
    - cd /home/aitem
    - Create directory: mkdir /home/aitem/<test, stage,

prod>

  - cd /home/aitem<test, stage, prod>
  - Download the agentDownload.linux script from the OMS. In this case, we will use emiat.oraclecorp.com. This will download the agentDownload.linux from the OMS and put in the /home/aitem/<test, stage, prod> directory:
  - sftp userid@amts122.us.oracle.com
  - please ensure that you pull from the right environment. These are the midtiers for each Enterprise Manager console.
    - amts122 – test
    - amts124 – stage
    - amts125 - prod
  - cd /u01/app/oracle/product/oms10g/sysman/agent_download/10.2.0.4
  - get agentDownload.linux
  - quit
- Installation
  - mkdir /u01/app/em/<test, stage, prod>
  - cp /home/aitem/<test, stage, prod> agentDownload.linux /u01/app/em/<test, stage, prod>
  - cd /u01/app/em/<test, stage, prod>
  - chmod 755 agentDownload.linux
  - Execute the following environment requirements:

- export ORACLE_HOME=/u01/app/em/<test, stage, prod>/agent10g
- export PATH=$PATH:/u01/app/java/jdk1.3.1_08/bin - Note: the path to the "jar" executable may not be as above, please find the executable and set the proper path.
- export ORACLE_HOME=/u01/app/em/prod/agent10g
- export PATH=$PATH:/u01/app/java/jdk1.3.1_08/bin
- export AGENT_INSTALL_PASSWORD=<password>
- chmod 755 *
- "pwd" and ensure that you are under the /u01/app/em/<test, stage, prod>
- Install with the following - ./agentDownload.linux –i /u01/app/em/<test, stage, prod>/oraInventory
- This will start the installation process and install under /u01/app/em/<test, stage, prod>/agent10g
- The process can be monitored via tail –f /u01/app/em/<test, stage, prod>/agent10g/ agentDownload.linux<#>.log. Run root.sh when installation is complete.
-

Shared Agent

With a shared agent setup, the binaries for a given OS platform (i.e., Linux, Solaris, HP) are shared on multiple hosts from a single shared NFS mount. This assists with upgrades and patch apply down the road. As well, there is a local “state” mount point where all local files for that specific agent reside. Before installing the shared agent, you must first decide where the “master” agent will reside. The host that is chosen, should be a host that is stable, has little downtime and will be around for some time. The “master” host should also support required C libraries for the lowest version of the operating system. As an example, in a Linux environment for a shared agent, you may have two versions (i.e., Red Hat 3; Oracle Enterprise Linux 4) running on the same shared agent. If the master is OEL 5 and the children are RH 3, a recompile of the executable on OEL 5 could break the RH 3 agents as the libraries on the OEL 5 target are at a higher version and therefore, RH 3 won’t be able to support the recompiled executable. On this master host binary tech stack is mount point mount READ/WRITE. This is the only place where the agent will be mounted READ/WRITE. All other shared hosts for a given binary setup will be mounted READ ONLY. This ensures proper configuration management and as well, makes it simple to manage where upgrades take place. It is critical that this setup is properly executed as you don’t want multiple READ/WRITE mounts for the binaries. This will confuse things and down the road, cause major problems, again around configuration management.

As well, when using a shared agent additional considerations specific to the type of disk subsystem that is used for the NFS mounts. It is critical to ensure that the disk subsystem is reliable from a performance and availability perspective. In the event, there are issues with the NFS disk subsystem, it will effect multiple agents and therefore, multiple targets.

Also, when using a shared agent, it is important to proactively monitor the shared file systems from a space perspective. Again, in the event

As well, prior to installation, setup is required at the local “home” account for the “aitem” OS agent owner. This is where the “oraInts.loc and the inventory file will reside for that master binary installation. Under the /home/aitem account location, create the following:

- /home/aitem/prod
- /home/aitem/test
- /home/aitem/stage

The agent installation file system setup is as follows:

- Test – With this setup, we have specified a port for the test agent to listen on which doesn't conflict with other ports being used on the host.

  - Binary Location: /u01/app/em/test/agent10g
  - State Location: /u01/em_state
  - Port Used: 1834

- Stage - With this setup, we have specified a port for the test agent to listen on which doesn't conflict with other ports being used on the host.
  - Binary Location: /u01/app/em/stage/agent10g
  - State Location: /u01/em_state
  - Port Used: 1836
- Production - With this setup, we have specified a port for the test agent to listen on which doesn't conflict with other ports being used on the host.
  - Binary Location: /u01/app/em/prod/agent10g
  - State Location: /u01/em_state
  - Port Used: 1832

The installation of the shared agent was completed via command line with the following setup requirements:

- Master Agent Installation Steps:
  - Log into the Enterprise Manager UI console as your own user
  - On the master agent host, ensure that there is an oraInst.loc file as follows; the file is to be located under /home/aitem/<test or stage or prod>/oraInst.loc:
    - inventory_loc=/u01/app/em/<test or stage or prod>/oraInventory
    - inst_group=aitem
  - Fresh Install on the "master" node from the Enterprise Manager UI:

- Navigate to the following:
  - Deployments -> Install Agent ->Fresh Install
  - Installation screen should be filled out as follows:
    - Source Software = Default
    - Version = As required (i.e., 10.2.0.1)
    - Platform = As required (i.e. linux)
    - Provide Host List (must be fully qualified domain i.e., gts271.us.oracle.com)
    - OS Credentials – aitem/<passwd>
    - Root.sh = Turn off – This will be run by web or database services afterwards.
    - Installation Base Directory = /u01/app/em/<test, stage or prod> – agent10g directory will be created under this base directory
      - Port – 1834 for test; 1836 for stage and 1832 for prod
      - Additional Parameters –
      - -z PST8PDT -i /u01/app/em/<test, stage or prod>/oraInst.loc
      - Management Server Repository Passwd – Same as aitem passwd
      - Additional Scripts – N/A
  - Click Continue – This will start the verification process and installation.
  - Accept License – if shown
  - Turn Off Registration for Metalink – if shown
  - Turn off Proxy – if shown
  - Click Continue
    - This will start the Connectivity Verification as well as execute the prerequisite installation review. Log is located here: /u01/app/em/<test or prod>/prereqLogs
    - You can also tail the install.log under the /u01/app/em/<test or stage or prod>/tmp area to get status.

        - Ensure that the proper root.sh after the 10.2.0.1 install has been run to preclude errors on the upgrade.
    - PATCH Installation/Upgrade:
        - The Master agent must now be upgraded to 10.2.0.2 prior to converting to a shared agent, select the patches be loaded in $ORACLE_HOME/EMStagedPatches/<patch_number>/<zip_file>.
        - Ensure that the master agent is up and available
        - Under “Related Links” select “Patch Oracle Software”
        - Select “Search by number”
        - Use patch number 3822442 – Platform Linux x86 – Search
        - Select the proper target and apply
        - This will schedule a job that can be reviewed for status
    - The following patches are MANDITORY and have to be applied to the master agent code tech stack else the master agent can’t be properly converted to a shared agent, patches are loaded in /home/aitem/stage_patch on gts271, the master agent host for Linux.
        - SHUTDOWN the master agent before applying these patches
        - ***Application of these agent patches will be manual . Copy the patches to the Stage directory [/ORACLE_HOME/EMStagedPatches/<patch_number>]. Before applying these patches check if ORACLE_HOME is set and if the PATH contains opatch. Set the ORACLE_HOME and PATH. Unzip the patches and follow the instructions in the readme.txt file***

        - 5231067 – for 10.2.0.2
        - 5230945 – for 10.2.0.2

        - 4761818 – for 10.2.0.2

- NFS Agent – Master Code Migration to NFS state
    - Shutdown Master Agent from /u01/app/em/<test or stage or prod>/agent10g/bin
    - ./emctl stop agent
    - Create backup file of the /u01/app/em/<test or stage or prod>/agent10g/sysman/install/AgentNFS.pl as follows:
    - cp AgentNFS.pl AgentNFS.pl.bck
    - Update the /u01/app/em/test/agent10g/sysman/install/AgentNFS.pl script with the following:
        - system("$OraHome/oui/bin/runInstaller **-invPtrLoc \"/home/aitem/<test or stage or prod>/oraInst.loc\"** -attachHome ORACLE_HOME=$OraHome ORACLE_HOME_NAME=\"nfsagent\" -slient ");
    - Copy the oraInst.loc_shared_<test or stage or prod> file from /u01/app/em to the /home/aitem/<test or stage or prod>/oraInst.loc. Create the directory if it doesn't already exist.
    - Update the **/home/aitem/<test or state or prod>/oraInst.loc as follows:**
        - inventory_loc=/u01/em_state/<host>/<test or state or prod>/oraInventory
        - inst_group=aitem
    - Execute the following to convert the master to the NFS agent:
    - cd /u01/em_state/<host>/<test or state or prod>
    - Execute the following:
        - Test: /u01/app/em/test/agent10g/sysman/install/nfsagentinstall -s /u01/em_state/<host>/test -p 1834
        - Stage: /u01/app/em/stage/agent10g/sysman/install/nfsagentinstall -s /u01/em_state/<host>/stage -p 1836

        - Prod: /u01/app/em/prod/agent10g/sysman/install/nfs agentinstall -s /u01/em_state/<host>/prod -p 1832
    - Answer the request for the registration passwd for the OMS
    - Once completed - Review the targets and update the dbsnmp passwd for database environments

Additional information can be found in the Oracle® Enterprise Manager Grid Control Installation Guide 10*g* Release 5 (10.2.0.5.0) under section 10, Installing Management Agent.

Agent Implementation Comparison

Single Agent

- Pros
    - Simplified installation
    - Good for small enterprise environments
    - Can be installed on local system disk
- Cons
    - In a large environment with a large number of targets, managing agents in an individual basis can be time consuming and difficult from a configuration management perspective
    - Upgrades/Patching implementation timelines can be extensive as each and every agent must be upgraded one by one.

Shared Agent

- Pros
    - Central location for administration of code
    - Excellent for large enterprise environments specific to administration and troubleshooting
    - Central location for upgrading and patching
- Cons
    - NFS disk subsystem required
    - Depending on the setup, central location for logs specific to the group of agents installed. In the event this area is filled up, all agents on that mount point will suspend monitoring and uploading of XML files.
    - Not supported for cluster database environments

Implementation

Once the repository, midtiers (OMS) and initial agents are installed, follow on implementation setup are required when you start adding additional targets over the basic agent installation. This section will outline the use of various configuration steps that will ensure proper configuration management and simplify administration as you expand the use of Enterprise Manager.

Agent

Once the stand alone and shared agents are installed, administration is minimal. In our case, as we added new hosts, we rolled them into the existing shared agent specific to the host OS being provisioned. As far as the stand alone agent, there isn't much administration required outside the requirement to apply patches. Upgrades on the stand alone agent aren't required as the stand alone agents are on the RH 2.1 OS and the Enterprise Manager agent 10.1.0.5.1 is the latest supported version.

From a shared agent perspective, the following outlines how to add a new host to an existing shared agent installation. This should be completed when adding a new host to your enterprise. This is done via command line with the aitem user. Prior to installation, please ensure that the proper /u01/app/em shared binary file system is mounted in the host where the agent will be installed.

- Copy the oraInst.loc file from an existing host under /u01/app/em to the /home/aitem/<test or stage or prod>/oraInst.loc. Create the /home/aitem/<test or stage or prod> directory if it doesn't already exist on the new host.
- Update oraInst.loc as follows:
    - inventory_loc=/u01/em_state/<host>/<test or stage or prod>/oraInventory
    - inst_group=aitem

  - Create the /u01/em_state/<host> directory if not already there.
  - Create the /u01/em_state/<host>/<test, stage and prod> directory if not already there.
  - cd /u01/em_state/<host>/<test or stage or prod>
  - Execute the following to add the host the master agent:

Test: /u01/app/em/test/agent10g/sysman/install/nfsagentinstall -s /u01/em_state/<host>/test -p 1834 –i /home/aitem/test/oraInst.loc

Stage: /u01/app/em/stage/agent10g/sysman/install/nfsagentinstall -s /u01/em_state/<host>/stage -p 1836 –i /home/aitem/stage/oraInst.loc

Prod: /u01/app/em/prod/agent10g/sysman/install/nfsagentinstall -s/u01/em_state/<host>/prod –p 1832 –i /home/aitem/prod/oraInst.loc

- Input the Administration Agent passwd.
- Once completed - Review the targets and update the dbsnmp passwd for database envs

Target Configuration

Database

When installing the agent, the database is automatically discovered. This is usually discovered under a name that isn't very useful when searching targets. Therefore, a removal and re discovery of an automatically discovered database with the proper name is required. This is completed via the agent page.

When re discovering a single instance database, the name should contain the host name, the instance name and the domain that the instance resides on.

If this naming convention is maintained, one can query any host or instance name from the Enterprise Manager home page via the "Target Search" feature.

The following naming convention should be used: <instance name>_<host name>.us.oracle.com

When this naming convention is maintained, you can search on the host or the instance and all occurrences will show up in the search results.

Listener

When discovering the listener via the agent level discovery, a proper naming convention should be maintained to ensure again, a search of any host or database instance will product the proper results.

The following naming convention should be used:

- LISTENER_<instance name>_<host name>.us.oracle.com

Again, if this naming convention is maintained, a search of the instance name or the host name will result in a list of all databases and listeners. As well, if a search is executed on the host name, the database instances and the listeners on the host searched will be produced.

Beacons

Beacons are used for the Application Service Level Dashboard (ASLM) module and are servers located in a location away from the primary Enterprise Manager console that plays transactions from a given location to simulate the end user experience.

We have Beacons located in the United Kingdom (UK), Singapore (SG), India (IDC), Austin Data Centre, TX (ADC) and Rocky Mountain Data Centre, CO (RMDC) . A beacon is simply an agent installed as a stand alone agent with the latest version of Enterprise Manager. It is important that the latest version is used to take advantage of all features available.

Also when creating a beacon on the agent installed, the naming convention is very important. We use the following to ensure that the standard search criteria can be used:

- <location>_<hostname>.us.oracle.com - As well, you can add a suffix to any beacon name for specific usages such using a specific beacon for a simple login/logout transaction. As follows: <location>_<hostname>.us.oracle.com-log

Application Server (iAS)

When discovering the Application Server, it is important to note that as the discovery process is driven by naming conventions setup in the XML configuration files for any iAS tech stack, changes to the name before discovery and as well, post discovery, are not possible.

In our case, this is an acceptable setup as the name has the application environment and host name in the iAS configuration files. The following is an example of the naming convention:

- <environment name>.<hostname>.us.oracle.com
- Example: aitinfra01.amts517.us.oracle.com

Again, this can be setup in any way required when installing the iAS tech stack. Once the installation iAS is complete, the name cannot be changed for specific Enterprise Manager discovery requirements

Discovery starts at the agent level where the iAS tech stack was installed. If the operating system user that owns the iAS tech stack to be discovered is different than the operating system owner of the Enterprise Manager agent, permissions changes are required in the iAS tech stack level before the discovery can take place. If the permissions changes are not completed, the discovery will fail, indicating that there where no targets found.

File permissions changes for the iAS tech stack are as follows using the “chmod” OS command:

- IAS_ORACLE_HOME/sysman/* world read
- IAS_ORACLE_HOME/config/ias.properties world read
- IAS_ORACLE_HOME/sysman/emd/targets.xml world read
- IAS_ORACLE_HOME/opmn/conf/opmn.xml world read
- IAS_ORACLE_HOME/opmn/bin/* world read/execute
- IAS_ORACLE_HOME/dcm/bin/* world read/execute
- IAS_ORACLE_HOME/dcm/config/dcm.conf world read
- IAS_ORACLE_HOME/Apache/Apache/conf/httpd.conf world

read

- IAS_ORACLE_HOME/Apache/Apache/bin/* world read/execute
- IAS_ORACLE_HOME/bin/* world read/execute
- /etc/emtab and/or /var/opt/oracle/emtab world read
- /etc/oratab.oem and/or /var/opt/oracle/oratab.oem world read
- /etc/oratab and/or /var/opt/oracle/oratab world read
- /etc/oraInst.loc and/or /var/opt/oracle/oraInst.loc world read

Once all required permissions changes are complete, the discovery will be successful.

Groups

When all required targets are discovered, the next step is to create groups to facilitate ease of administration for the targets in question.

It is critical that the grouping of targets is completed in a very organized fashion. In our case, we separated the groups by midtier hosts and database hosts.

For the database groups the naming convention includes the application supported with the suffix "_dbs" (e.g. gsiap_dbs). This group includes the host, agent, database cluster (if applicable), database instances and listeners for that environment.

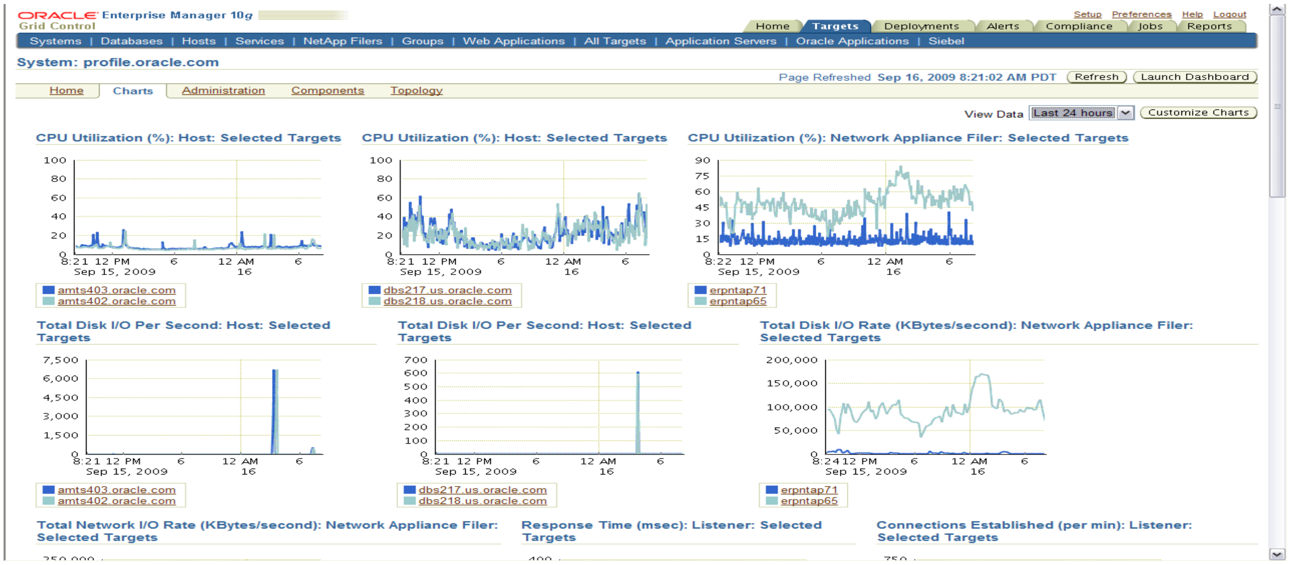
For the midtier groups we use the same naming convention specific to the environment with the suffix "_mts". This group includes the host, agent, Application Service Level Management (ASLM) transactions and iAS targets.

Once the groups are created, this will facilitate the use of Group Level Dashboards with button located on the right side of the group home page. As well, under the administration tab, there are features that will enable you to compare targets in the group against user defined templates. Templates will be discussed in section 3.5. As well, under the group administration tab, one can search the configuration for the targets in that group. Finally, one can also obtain a deployment summary around hardware, OS levels and Oracle Homes.

Additional information can be found in the Oracle® Enterprise Manager Grid Control Quick Start Guide 10*g* Release 2 (10.2) under section 4, Managing Groups.

System Group

Another type of group would be a System Group where one would create a group with all of the targets that make up the infrastrucuture that supports a given application. This can later be used to create a “chart” view of specific metrics across the targets added to the System Group. As well, this System Group will later be used to assign to Application Service Level Management (ASLM) Web Transaction to further expand the use of root cause analaysis for a given applicaion downtime. An example would be the creation of a System Group of disk subsystem, database host and database, middleware hosts and middleware application, traffic routing targets and web application transactions. Once this is created, the chart tab can be configured to display CPU usage at the disk subsystem level, hosts CPU & Memory on the database and middleware hosts, database wait times, database listener connections refused, middleware OC4J statistics and finally, the CPU usage on the traffic routing target (e.g., F5 BigIP Router). This will provide an overview of the environments infrastructure that could highlight “hot spots” that may be causing performance issues and overall application instability. The following is an example of a possible view:


ORACLE Enterprise Manager 10g
Grid Control
Setup Preferences Help Logout
Home Targets Deployments Alerts Compliance Jobs Reports
Systems | Databases | Hosts | Services | NetApp Filers | Groups | Web Applications | All Targets | Application Servers | Oracle Applications | Siebel
Home Charts Administration Components Topology
Refresh
Launch Dashboard
Customize Charts
CPU Utilization (%): Host: Selected Targets
100
80
60
40
20
0
8:21 12 PM 6 12 AM 6
Sep 15, 2009 16
amts403.oracle.com
amts402.oracle.com
CPU Utilization (%): Host: Selected Targets
100
80
60
40
20
0
8:21 12 PM 6 12 AM 6
Sep 15, 2009 16
dbs217.us.oracle.com
dbs218.us.oracle.com
CPU Utilization (%): Network Appliance Filer: Selected Targets
90
75
60
45
30
15
0
8:22 12 PM 6 12 AM 6
Sep 15, 2009 16
erpntap71
erpntap65
Total Disk I/O Per Second: Host: Selected Targets
7,500
6,000
4,500
3,000
1,500
0
8:21 12 PM 6 12 AM 6
Sep 15, 2009 16
amts403.oracle.com
amts402.oracle.com
Total Disk I/O Per Second: Host: Selected Targets
700
600
500
400
300
200
100
0
8:21 12 PM 6 12 AM 6
Sep 15, 2009 16
dbs217.us.oracle.com
dbs218.us.oracle.com
Total Disk I/O Rate (KBytes/second): Network Appliance Filer: Selected Targets
200,000
150,000
100,000
50,000
0
8:24 12 PM 6 12 AM 6
Sep 15, 2009 16
erpntap71
erpntap65
Total Network I/O Rate (KBytes/second): Network Appliance Filer: Selected Targets
Response Time (msec): Listener: Selected Targets
Connections Established (per min): Listener: Selected Targets

System: profile.oracle.com

Page Refreshed Sep 16, 2009 8:21:02 AM PDT

View Data Last 24 hours

Notification Rules

The notification rules are the drive for all alerts received when a metric setting is violated. It is critical that the rules are setup properly as in the event there are issues with rules, notifications aren't sent and proactive information to parties concerned are lost.

Notification rules are based on target type, metric and level of notification (i.e., P1 or P2). Under this setup, one can assign a group with multiple target types and specify the target type that you want to notify against.

As notification rules are created and administered at the Enterprise Manager user level, we created a single Enterprise Manager user that is used to administer all notification rules. This user has super administer and is only accessed by the Enterprise Manager administration team.

Naming convention and notification definition is critical. As you move forward with the setup, you will find that multiple notification rules are required to satisfy different requirements from database and midtier application owners. As well, the use of multiple notification methods is available in the setup

The naming convention we used consists of the notification level rule number for that section. That is to say, we setup different rules for database and midtier environments. Therefore, if we have a requirement for a database rule that notifies as a P1, we would create a rule that consists of the following:

- Name: P1rule1_DBA
- Owner: Rule_Admin
- Description: Stage and Critical Database: p1's for alert, p2 for warning and error

As we create additional rules, we would increment the rule number (i.e., P1rule2, P1rule3,…). For the midtier rules, we use the same naming convention for the rule name with the suffix "_MTS" and provide the proper description to ensure that when a team member is reviewing, it is clear what the rule is used for.

When creating the rule, you can then specify the target type, metrics you want to notify on and the notification method desired. In most cases, one would use the e-mail notification method. That being said, one can also create custom notification methods in the Enterprise Manager data base repository and interface with external notification systems. Enterprise Manager also provides interfaces to other standard call tracking systems such as Remedy.

Prior to using the e-mail notification, you must also update the e-mail configuration under the Notification Methods section under the "Setup" link, located in the top right hand corner of the Enterprise Manager home page. Complete the "Mail Server" setup as well as the "Repeat Notifications", as required.

Once the rules are created, you can then associate a group that was previously created, the rule. This simplifies the administration of notification rules. When groups are created with various targets, the group can be associated with a notification rule and as notification rules are driven by target type, only the target types that are addressed in the rule will be notified against. This facilitates simply additions and removal of targets. When a new target is discovered, simply add the new target to the required group and after the metrics template is applied, the rule will then notify on the required metrics. As well, when it is required to stop metrics or remove a target, you simply remove the target in question from the group and the notifications will stop.

Finally, additional information around notifications can be found in the Oracle® Enterprise Manager Advanced Configuration 10*g* Release 5 (10.2.0.5) under section 13, Configuring Notifications.

Monitoring Templates

When deploying any target type, it is important that the metrics on like target types have a basic standard setup. We use templates to ensure that when we discover a new target, the same basic standard and User Defined Metrics (UDM) are applied across all like targets. As an example, before discovering the database targets, review all standard metrics that are available out of the box in Enterprise Manager and define the basic metrics that you would like to see on ALL like targets discovered. You should then review any requirements for UDMs. Before adding to the template, simply discover the first database target, add the UDMs and then create the standard template. Once this is complete, you will have all standard metrics as well as UDMs in the same “baseline” template.

This process should be used for any target type to ensure a standard baseline is created. Templates can be created by any user so it is also important to designate the single user that will administer baseline templates. This also ensures that templates aren’t changed outside of the standard process.

Specific to the naming convention for templates, it is also important to maintain standards. In our case, we setup user administrators for the database and midtier templates. Therefore, when working with the database templates, the database template/notification rule is used to administer.

The naming convention we used for templates was as follow:

- Admin User: DBA_CR_TS_NOTIFY
- Template Name: RDBMS-ALL-11g
- Target Type: Database

Once the template is created, it can be applied to multiple targets.

As well, templates can be used against to compare metrics between different targets. This assists in configuration management, ensuring that like targets have the same baseline standard and User Defined Metrics.

Security Policies

When reviewing the various targets, it is a good idea to review the “out of the box” policies when provisioning targets. Under the “Compliance” tab, there is a list of policies under the “Library” link.

For our deployment, we focused on the security policies but there are also policies around “config” and “storage”.

The process to deploy the required security policies is very simple, we distributed the list of security policies to the Security and Database Services’ teams, requesting review and input as to the policies that they would like to deploy.

Once we received the feedback, create a “Security Policy Monitoring Template” and applied the policies to the designated database targets. We then created a notification rule to address the policy violations. Alerts are then routed to the Security team for review. After the Security team completed a review of the alerts and ensured that action was required, they would then forward the alert to the Database Services group for action/correction. Once the issue is corrected on the target with the violation, the alert would clear in Enterprise Manager and the issue would auto close. Therefore, when the policy was reviewed in the future, no new alert would be fired.

As well, once all security policies are applied and violations are addressed, you can review the status around security for any target type via the link on the target home page under the security section. Click on the “Security at a Glance” and it will summarize violations and overall percentage of compliance.

Roles

Roles allow you to group Enterprise Manager target privileges, and grant these to administrators or to other roles. Privileges give the administrator rights to perform management actions within Enterprise Manager. With roles, one can separate the access and administration of target types as a view, operator or administrator level.

For our purposes, we created roles based on database, midtier and performance organizations with in our management structure. Within this setup, we also setup roles for view, operator and administrator.

As an example, we created a role called “AITSYS_PERF_ASLM_A” that enables the user to administer Application Service Level Management (ASLM) transactions for the Performance Team. Within this role, they can only see those transactions and cannot administer or update any other target types.

The following is the basic flow that can be used when a new user is created:

- User Created
- Role assigned
- Subscribe to notification rules

- Target Subtab Setup

Additional information can be found in the Oracle® Enterprise Manager Grid Control Quick Start Guide 10*g* Release 2 (10.2) under section 2, Getting Started with Enterprise Manager.

Application Service Level Management (ASLM)

The Application Service Level Management module enables you to proactively monitor the application from the end user perspective. With ASLM along with the beacons in the global locations, you can simulate the end user experience from a given global location.

We have beacons located in India, United Kingdom, Singapore, Texas and Colorado. This gives a good overview of performance for our general employee population.

The transactions we recorded are simple login/logout transactions that ensure that we can access the application via the standard URL. Once the transactions are completed, it is critical to review the “success strings” for each transactions step.

In the UI for a given transactions, under the key test, you click on the “Edit Service Test” at the bottom of the page. Under this section, you will see the steps for each transaction. Edit each step and add the success string for each step. The success string should be something that you would expect would be returned by that step. When creating the transaction, Enterprise Manager will auto fill the success string field with the required data, this should be validated.

Success strings are critical as this will ensure that the transaction is working properly. If you don’t have success strings, the transaction will assume that any return of data from the step in the transaction is valid and in some cases for specific applications, this could give you a false positive.

From a naming convention perspective, as always, thought should be put in to the way things are named. We used the following format:

- <Application>-Application Login-<Sub Application>

The sub application would be the specific application for a given environment. That to say, some applications have separate sections that you would want to drill down into to ensure a successful login.

Additional information around Services and ASLM can be found in the Oracle® Enterprise Manager Grid Control Installation Guide 10*g* Release 5 (10.2.0.5.0) under section 7, Configuring Services.

Target Management

Jobs

Enterprise Manager provides a job system that enables the administrator to execute various administrative scripts across multiple target types.

Most administrators today execute various administrative scripts via the host level crontab. This is a huge administrative nightmare as you have to go to each host to administer and from a notification perspective, this is a distributed setup.

With the Enterprise Manager job queue, one can execute any host level UNIX/Linux script across multiple hosts and provide status from a central location.

This also can be done for database and midtier application tech stack targets. As well, you can save the job in the “Job Library” and schedule for execution against additional like target types, down the road.

There are various job types that can be executed from log rotation to patch applying to start-up and shutdown of database and midtier envs. Finally, there is a specific job that executes against My Support (Formally Metalink) that pulls details specific to Critical Patch Updates (CPU) for the tech stacks discovered in ones console.

Also, one can create a job that has multiple steps. During the run, each step can be dependent on the success of the previous step. In the event any previous step fails, notification can be sent out and the entire job would be suspended until all steps are successful.

From our perspective, we use the Enterprise Manager job queue for core file cleaning, application session killing when they are spinning out of control, performance analyze, stats gathering and review of additional security setup.

Once these jobs run, if there are failures we are notified and action can then be taken to correct. As well, one can be notified on a successful run to ensure things remain clean.

Additional information can be found in the Oracle® Enterprise Manager Grid Control Quick Start Guide 10*g* Release 2 (10.2) under section 6, Job System

Information Publishing (Reporting)

Another powerful feature is the reporting module. There are a number of "standard" reports in Enterprise Manager that are owned by "sysman". These can be used as is, or can be used as "templates" with the "create like" feature. One can then customize the report for your specific requirements. Standard "canned" reports can be found specific to Deployment and Configuration, Enterprise Manager Setup, Monitoring, Security and Storage.

From a custom creation perspective, we have created various custom reports to provide additional insight into status, capacity and configuration. Draft reports are created by the individual users and then migrated to the "production support" user for the reporting module. The "AITSYS_REPORT" user is used to administer and configure the production reports. As well, various reports we run are set to "public access". This enables us to provide report access to management groups without having to log into the Enterprise Manager console.

The publishing of a public report can be configured by simply setting the proper privileges for the report to run and then setting the report public under the "Access" section for any report created.

Once this is completed, you then can link the report in other web pages for centralized access.

The following are some of the custom reports we have created:

- Hardware, Operating System Version & Oracle Database Version inventory summary
- Disk Subsystem (i.e., NetApps) reports that summarize details specific to quarterly growth capacity.
- Database Target User Defined Metric (UDM) Listing Report
- Host File system Mount Summary
- Monitoring Template Metric Summary by Target Type (i.e.,

Oracle_Database)

- Host Performance Summary (Daily, Weekly, Monthly)
- Database Capacity Overview by Quarter
- Database Tablespace Capacity Report – By Target

Additional information can be found in the Oracle® Enterprise Manager Grid Control Quick Start Guide 10*g* Release 2 (10.2) under section 9, Information Publisher.

Enterprise Manager Command Line Interface (EMCLI)

The command line interface enables one to interact with the Enterprise Manager console at the OS level from any host. This interface must be configured with the required access to the console so that it can execute various commands. Via the command line interface, one could administer targets when working with a large number of targets.

In the EMCLI, you can get help with any specific command with the following:

Type the following to get a list of EMCLI commands: emcli

Then type the following to get help for each: emcli help <command>

In our case, we use the EMCLI to update passwords for Siebel targets as well as create blackouts via a custom application that tracks blackout tasks.

From a security perspective, one should limit where the EMCLI is deployed. During the setup of EMCLI, one must configure to user a userid/password in the Enterprise Manager console that has massive privileges. Although the password is stored in an encrypted fashion, it is still good practice to limit the deployment of EMCLI clients to specific authorized users at the OS level to facilitate future maintenance and upgrade of the EMCLI client binaries. In later versions (i.e. 10.2.0.5) of Enterprise Manager, there will no longer be a requirement to store the encrypted password when configuring EMCLI.

Siebel CRM Usage

From a Siebel perspective, we are required to change the password for all of our Siebel CRM environments to maintain compliance with Sarbanes-Oxley Act of 2002. As the Siebel enterprise and Siebel Servers have hundreds of components, it isn’t practical to update the password via the UI. Therefore, the use of the EMCLI is required. Detailed code can be found in the appendix of this document.

Blackout Usage

From a blackout perspective, we use the EMCLI to create blackouts for scheduled outages around maintenance for our critical environments.

When the blackout is started in the custom application, the custom application interacts with Enterprise Manager to create the blackout for all required targets. Before we start the blackout, we pull a list of targets from Enterprise Manager via the EMCLI, save in a flat file and then execute a UNIX/LINUX script in a loop to create blackout for the targets in question. See appendix at the end of this document for code details.

User Administration Usage

When administering users, one can use the EMCLI to create, update and end date users as well as assign roles. Using the command line interface enables the Enterprise Manager administrator to provide self services support to the end user. In our case, we use an external application that manages users for the corporation. By interfacing with Enterprise Manager via the EMCLI, and end user can request an account on our Enterprise Manager console with no additional intervention required.

Additional information on the use of EMCLI can be found in the Oracle® Enterprise Manager Advanced Configuration 10*g* Release 5 (10.2.0.5) under section 10, Using Enterprise Manager For Grid Automation With Deployment Procedures.

Configuration Comparison

One of the features under the Deployment tab in Enterprise Manager is the Configuration Comparison feature that enables the user to compare a baseline specific to the setup on a given target against additional like targets.

This feature is very useful when reviewing the configuration for each instance in a cluster database, as an example. One can compare real-time from one target to another or compare one to many via the job queue. If you choose to compare one to many, Enterprise Manager will start a job that will complete the comparison in the background and then provide the results in a report.

This type of comparison will reveal all differences from the baseline system to the compared targets and ensure that all instances in a cluster are the same.

As well, one could create a snapshot of a given target, save it in the library and use for trouble shooting down the road. As an example, take a snapshot on Monday of all critical environments. This will enable one to compare the configuration on Monday with the configuration on a day down the road to see what has changed from Monday to let's say, Friday. This is a very useful tool when trouble shooting and trying to understand what changed in the life cycle of an environment.

Additional information can be found in the Oracle® Enterprise Manager Grid Control Quick Start Guide 10*g* Release 2 (10.2) under section 7, Configuration Management.

User Defined Metrics (UDM)

Where possible, we make every attempt to use standard, out of the box metrics when monitoring our targets. That being said, there are some cases where a "custom" solution is required. This is where the UDM comes on. The User Defined Metric enables the administrator to create scripts that can be scheduled at the target level with thresholds that will notify when a specific issue is seen. This can be at the host level, the database level or the application server level. Additional information on now to configure and when to use can be found in the Oracle® Enterprise Manager Advanced Configuration 10*g* Release 5 (10.2.0.5) under section 14, User-Defined Metrics.

Conclusion

When implementing Enterprise Manager, it is critical to plan ahead and understand what is required in the initial deployment. As well, it is also important to plan for expansion and setup the environment so that major changes to your enterprise don't require major changes to the configuration of Enterprise manager. As well, major changes wouldn't be required when adding targets.

From a console perspective, it is also important to create a test environment so that validation of patches and the roll out of new features can be executed before moving to production. As well, the capacity required at the console/repository level is important. As you add targets to the console, this also effects the space requirements at the repository level.

At the agent level, deciding which type of agent is critical there are no currently supported methods to migrate from a stand alone agent to a shared agent.

Finally, with the proper planning and implementation, Enterprise Manager will enable the administrator to manage many targets as one via the standard Enterprise Manager console. As well, as you expand the enterprise you manage, Enterprise Manager will enable you to roll in new targets as well as new target types while maintaining standards, providing proactive reporting, notifications and management for your critical applications.

Appendix

The following are specific references to scripts and documentation we have used to implement various features discussed in this document.

Enterprise Manager Command Line Interface - Siebel CRM Password Change Script

This script should be run and then the post product from this script needs to be run twice to validate the password change was completed. Then the agent on the target in question should be bounced. Once this is completed, the passwd will be changed on the environment it was run against.

As well, before running this, you will have to adjust environment variables and also ensure that the script to setup the java environment is complete.

The JAVA_SETUP script should look like this:

```
#!/usr/bin/ksh
export JAVA_HOME=/usr/java/jdk1.4.2
export
PATH=/u01/app/system/scripts/gsaoem/emcli/emcli_10.2.0.4:$JAV
A_HOME/bin:$PATH
```

Core Script:

```
#!/bin/ksh
#
#  siebel_pass.sh
#
# rbrinega 18jul08 – created
CMD_NAME=$0
CMD_BASE=`basename ${CMD_NAME}`
CMD_DIR=`dirname ${CMD_NAME}`
TARGET=$1
SIEBEL_PASS=$2
EXECUTE_COMMAND=$3
TIMESTAMP=`date +%d%b%y_%H:%M`
TEMP_COMMAND_TEMPLATE=${CMD_BASE}_temp
TEMP_COMMAND=${CMD_DIR}/${TEMP_COMMAND_TEMPLAT
E}_${TIMESTAMP}_${TARGET}
```

```
JAVA_SETUP=/u01/app/system/scripts/gsaoem/emcli/emcli_10.2.0
.4/java_setup
if [ "X"${EXECUTE_COMMAND} != "X-run" -a
"X"${EXECUTE_COMMAND} != "X-norun" ]
then
echo
echo "usage $0 <TARGET> <NEW_SIEBEL_PASSWORD> [-run|-
norun]"
echo
echo "  where TARGET is the individual target"
echo "      (eg.,
AsgnSrvr_AsgnMgmt_amts114_GCMAT_amts112.us.oracle.com)"
echo "    or any group of targets with a similar string"
echo "      (eg., gcmat or GCMAT - we ignore case)"
echo
echo "  The -run flag means run the script now and the"
echo "      -norun flag means only create the script file to be run
later"
echo
exit
fi
#clean up old temporary commands
find ${CMD_DIR} -type f -name
"${TEMP_COMMAND_TEMPLATE}*" -mtime +15 -exec rm {} \;
${JAVA_SETUP}
emcli get_targets -format=name:script|egrep
'siebel_component|siebel_component_group|siebel_server'|sed -e
's/  /~/g' | awk -F~ '{print "emcli modify_target -name=" $4 " -type="
$3 " -credentials=SIEBEL_PASSWORD:${PASSO} -
on_agent"}'|grep -i ${TARGET}> ${TEMP_COMMAND}
chmod 700 ${TEMP_COMMAND}
export PASSO=${SIEBEL_PASS}
if [ ${EXECUTE_COMMAND} == "-run" ]
then
  ${TEMP_COMMAND}
  echo " ${TEMP_COMMAND} has been run"
  echo
fi
echo "if you want to execute the temporary command (which may
be deleted after two weeks):"
echo "   ${TEMP_COMMAND}"
echo "you will need to issue the following commands first:"
echo "   . ${JAVA_SETUP}"
echo "   export PASSO=${SIEBEL_PASS}"
echo
if [ ${EXECUTE_COMMAND} == "-norun" ]
then
echo "NOTE: the temporary command ${TEMP_COMMAND} has not yet been run"
echo
fi
```

Enterprise Manager Command Line Interface – Blackout Script

Before running, you will have to adjust environment variables and also ensure that the script to setup the java environment is complete.

The JAVA_SETUP script should look like this:

```
#!/usr/bin/ksh
export JAVA_HOME=/usr/java/jdk1.4.2
export PATH=/u01/app/system/scripts/gsaoem/emcli/emcli_10.2.0.4:$JAVA_HOME/bin:$PATH
```

As indicated, before you start the blackout, you can list the targets in a group or by target type to locate the exact name of the target with the following

Get Groups: emcli get_groups

Get Group members: emcli get_group_members -name=my_hosts -depth=0

Or Get Target by Target Type: emcli get_targets -targets="oracle_database"

Command Line Execution that can be wrapped in a UNIX/Linux shell script to execute against custom applications.:

```
emcli create_blackout -name=b1 -add_targets=database2:oracle_database -schedule="duration::30" -reason="good reason1"
```

Documentation

Enterprise Manager Concepts

Enterprise Manager Grid Control Quick Start Guide

Enterprise Manager Product Main Page

Oracle Technology Network Enterprise Manager Documentation

Enterprise Manager Extension/Plugins

Enterprise Manager Configuration Management

Enterprise Manager Architecture Best Practices

Enterprise Manager User Group

Enterprise Manager Grid Control Installation Guide

Enterprise Manager Advanced Configuration

ORACLE

White Paper Title
September 2009

Oracle Corporation
World Headquarters
500 Oracle Parkway
Redwood Shores, CA
94065
U.S.A.

Worldwide Inquiries:
Phone:
+1.650.506.7000
Fax: +1.650.506.7200
oracle.com

Oracle is committed to developing practices and products that help protect the environment