Oracle Enterprise Manager 12c Facts and Concepts

Author: Rob Zoeteweij  
Date: 19/01/13  
http://oemgc.wordpress.com

Oracle Enterprise Manager 12c was released October 2011 and introduced a complete rewritten framework for monitoring and managing Data Centre components.

Where Oracle Enterprise Manager evolved from just a Database Monitoring tool (EM 10.1) to include Middleware targets and targets within the complete Oracle Software Stack and even non-Oracle targets (like MS SQL Server, IBM Websphere, BEA Systems, Storage and Network devices and others...), to a Grid supporting and Database Lifecycle support tool. Enterprise Manager 12c now supports Lifecycle Management throughout the complete Oracle Software and Hardware Stack (Enterprise Manager OPS Center 12c), delivering “Application to Disk” monitoring, advanced management for Oracle Hardware, Virtualized environments and fully supporting Oracle’s Cloud concepts including PaaS (Platform as a Service), IaaS (Infrastructure as a Service), DBaaS (Database as a Service), MWaaS (Middleware as a Service) and several more being announced (Schema as a Service, Testing as a Service).

Even more than before, encouraging Partners to embrace the new Plug-in framework and to deliver additional features to support non-Oracle environments.

This paper will introduce several Facts and Concepts to give a good overview of available features, challenges and ideas coming up when starting implementation of what has become a key part of Oracle’s strategy to deliver hardware and software systems engineered together as appliances and integrated systems.

Features discussed in this paper

• Framework  
• Administration Groups  
• Dynamic Groups  
• Template Collections  
• Lifecycle Management  
• Incident Management  
• Configuration Management  
• Compliance Management  
• Cloud Management  
• Mobile
Framework
The heart of the Enterprise Manager framework consists out of the following components:

- **Oracle Management Service (OMS)**
  Stores data that was collected by a Management Agent and uploaded to the OMS in the Management Repository. This data next will be input to several sub-systems that will interpret and react accordingly. An example of this could be the Incident Notification system that will send out notifications to an Administrator if a certain series of events occurs.
  The Management Service will also take care of UI rendering in case an Administrator connects to the Cloud Control Console.
  The OMS is implemented as a Java Application running in a Weblogic Server environment

- **Oracle Management Repository (OMR)**
  The Management Repository is stored in the SYSMAN schema in a dedicated Oracle Database.

- **Oracle Management Agent (OMA)**
  The Management Agent will, based on a given schedule, collect data of monitored targets on a Host and upload this data in xml files to the Management Service.

- **Oracle Management Plugins**
  Specific Target type related functionality that can be deployed as integrated code on a Management Agent. Any code that is dedicated to a certain Target Type (like a Database Instance, Weblogic Server, Host, etc.) is implemented as a Plug-in.

- **Enterprise Manager Console**
  The User Interface layer (ADF based) that allows an Administrator to browse through the data that is stored in the Management Repository.

Working together as in the following picture:
Please note that both Load Balancer and second OMS are included to show a possible setup for organizations working with environments including many targets or because of availability demands.

Also note that you might consider Oracle Enterprise Manager 12c as just another mission critical Application with its specific availability requirements that necessitates for an High Availability Architecture as described in “Oracle Maximum Availability Architecture – MAA” – checkout: http://www.oracle.com/technetwork/database/features/availability/maa-090890.html

Also note that the EM12c architecture does not include an (in this example “shared”) upload file system as used in previous releases (EM10 and EM11g). Agent uploaded files would be stored by the OMS prior to actually processing the files. Agent uploaded files are now processed by the OMS instantly.

Figure 1 shows the architecture where data being collected by Management Agents is uploaded to the Management Service that will store the data in the Management Repository (the SYSMAN schema in just another Oracle Database). The Management Service includes several sub-systems that will start actually processing the data that was uploaded and for instance is input to the Incident Notification sub-system. The OMS will also take care of rendering data from the Repository to the Cloud Control Console when
any Administrator accesses it. For more information on this, please checkout: Oracle Enterprise Manager Cloud Control Introduction -12c Release 2 (12.1.0.2)

**Oracle Management Plugins**
Some special remarks on the Oracle Management Plugins Framework.

We remember Plugins as they already existed in previous releases (EM 10 and 11) where they specifically supported the monitoring of non-Oracle Targets.

In EM12c, Plugins are here to provide any functionality on any type of Target, including obvious Target types like Database Instance, Weblogic Server etc.

Because of this the footprint of a Management Agent will be as minimum as possible as it will include only that code necessary to monitor and manage Target types that are actually on a Host!

![Figure 2 Agent Footprint based on managed Target Types](image)

As Figure 2 shows, the size of the Management Agent’s footprint is based on the number of different Target Types monitored on a Host.

**Extensibility**
Plugins are a component within the Extensibility Framework that is in EM12c.

The Framework make sure that you as an Administrator will be aware of the availability of any new version of a Plugin, Agent Software and Management Connectors.
Figure 3 The Self Update page within EM12C

Figure 3 shows the Self Update page as part of the Extensibility Framework. From this Page Administrators are able to download any new version of Plugin, Management Connector, Virtual Assembly, VM Template, Monitoring Standards etc. provided by Oracle.

Figure 4 Plug-ins Page

Figure 4 shows the Plug-ins Page. Please notice that the page shows the latest available version of a Plugin, the version of the latest downloaded Plugin, the version of the Plugin that is deployed to the Management Server and the number of Agents the Plugin is deployed.

**Administration Groups**

Groups were here already when working with Enterprise Manager 10 and 11g and provide us with means to logically group Targets.
One major advantage of working with Groups is that they appear as just another Target Type, allowing us to perform several ‘standard’ actions with it like: include in Reports, Jobs, Monitoring Template applications etc.

Figure 5 Group Homepage in EM11g

Figure 5 shows the Homepage of a Group in EM11g. In this example the Group “All Database Instances” includes only Database Instance Target Types. A Group however can include any combination of Target Types.

Having said that, you might imagine it would be an interesting idea to include Groups in a Group... and by this creating a Hierarchy.

Figure 6 Hierarchy of Groups
Figure 6 shows such a Hierarchy of Groups, in this example based on what you might call the Lifecycle phase of a Target.

Next you could start using this Hierarchy to use it when:
- Applying Monitoring Templates
- Running Jobs (for instance Database Backups)
- Running Reports

Although this sounds like a very good idea and this example is seen at several organizations using Oracle Enterprise Manager 10/11g, this approach implies several challenges including but not limited to:
- How do we know what Lifecycle phase a target is in?
- How do we make sure only Targets of the proper Type are included?
- How do we make sure all Targets are in the right Group?
- How do we make sure this Hierarchy stays limited and therefore manageable?
- How do we make sure Targets are not in two different Groups at the same time?
- When using a Group within this Hierarchy for the application of Monitoring Templates, how can we make sure that we won’t forget to reapply a Template if one of it’s Metric Thresholds has been modified?

In too many situations Group Hierarchies end up in chaos because of unlimited growth and undefined completeness of any member in the Hierarchy.

In order to address most of these challenges, EM12c introduces the concept of ‘Administration Groups’.

Figure 7 shows the Administration Group DTAP_GROUP divided into three ‘Sub’ Groups:
- Production – PRD_GROUP
- Staging – ACC_GROUP
- Development or Test – DEV_TST_GROUP
These Groups are based on the members Lifecycle Phase, where you might notice that Lifecycle Phases ‘Development’ and ‘Test’ are combined in one Group. Enterprise Manager supports the Lifecycle Concept and recognizes the following Phases:

- Mission Critical
- Production
- Staging
- Test
- Development

Also notice that each of the Groups includes members.

**Figure 8 Lower Level Branches in Administration Group**

Figure 8 shows the expansion of Group ‘PRD_GROUP’ into 4 sub Groups:

- SALES – SALES_PRD_GROUP
- FINANCE – FINANCE_PRD_GROUP
- CRM – CRM_PRD_GROUP
- IT – IT_PRD_GROUP

Each of these Groups is based on the members Line of Business (SALES, FINANCE, CRM and IT).

Whereas the Lifecycle Phase in EM12c has 5 distinct valid values (Mission Critical, Production, Staging, Test, Development), the Line of Business domain can include any number of values.

The relationship between the Administration Groups and the Lifecycle Phase and Line of Business is encapsulated by means of the Properties of a Target.

Each Target has got some additional Properties that integrate with the Administration Group model.
Figure 9 Target Properties

Figure 9 shows the Target Properties that can be used to setup an Administration Group Hierarchy.

Notice that setting one of these Properties to a specific value will result in inclusion of the Target in the related Administration Group.

So, if for instance one of your Administration Groups is based on Property 'Cost Center' then setting this Property to value 'FINANCE' would automatically include this Target in the proper Administration Group.

Template Collections
Administration Groups are specially designed to support the setup of Monitoring Settings. As the previous section described the Administration Groups concept and setup, this section will describe the usage of Template Collections.

Monitoring Settings
Monitoring of Targets is done by means of Metrics and Thresholds and Compliance Settings.
When looking at Metrics and Thresholds we know that each Target has a set of its own. So for instance if we would take a look at a Database Target and navigate to the Metric and Collection Settings we would see the following (or something similar):

![Figure 10 Metrics and Collection Settings](image)

Figure 10 Metrics and Collection Settings

Figure 10 shows an example of the Metrics and Collection settings for a specific Database Instance.

This page would allow us to modify any of the settings and by that modify the monitoring of this specific Target (orclBTM.mycorpdomain.com in our example). This approach of setting Thresholds for specific Metrics dates from Enterprise Manager release 10 and therefor is something we are familiar with.

We should also be familiar with the fact that we would prefer to use Monitoring Templates to modify Metric Thresholds for all of our databases, or for all of the databases in a specific Group. By this we were able to enforce Monitoring Standards and monitor our Databases all in the same way.

New in EM12c is the concept of Template Collections, where one Template Collection can contain several Monitor Templates. A limitation here is that you can only have one Monitoring Template per Target Type as member of a Template Collection.
Figure 11 shows an overview of available (in this example) Template Collections.

Figure 12 shows that Template Collection 'PRD Template Collection' includes 3 Monitoring Templates for 3 different Target Types.

The next thing you need to do is associate your Template Collections to an Administration Group. By doing so, the Monitoring Settings that are included in the Template Collection by means of Monitoring Templates will be applied to all members within the associated Administration Group!

It is also true that if a target becomes a new member within an Administration Group, the Monitoring Settings will be applied automatically based on the Template Collection association!

It is also true that if any of the Metric Thresholds within any of the Monitoring Templates is modified, the modified settings are applied automatically to all members within associated Template Collections, containing the Monitoring Templates that was modified!
Synchronization Schedule
All these automatic application of Monitoring Settings based on Template Collection / Administration Group association is controlled by the ‘Synchronization Schedule’.

Figure 13 Synchronization Status in Administration Group
Figure 13 show the Synchronization Status in Administration Group PRD_GROUP. In this Status overview you will be able to see how many Targets are synchronized, how many Synchronization Jobs are still pending or running or have actually failed during execution.

Figure 14 Administration Groups and Template Collections Page
Figure 14 shows the Administration Groups and Template Collection Page. In this page we can see the Synchronization Status for the Group as well. Notice that the page says, “All targets under the administration group are synchronized with Template Collections”. The page includes a button ‘Start Synchronization’ that will only be enabled if not all targets would have been synchronized yet.

![Administration Groups and Template Collections](image)

**Figure 15 Unassigned Targets Report**

Figure 15 shows another helpful tool to manage your Administration Groups and Monitoring Settings, the “Unassigned Targets Report”. This Report will show you all Targets that are not in any Administration Group yet.

![Unassigned Targets Report](image)

**Figure 16 Unassigned Targets Report**
Figure 16 shows the Report “Unassigned Targets”. With this Report you can make sure all Targets have the proper Properties set to get included in the Administration Group Hierarchy.

**Dynamic Groups**

Groups as we know them since Enterprise Manager 10, are static. This means that we need to add Members to a Group manually. As described in Paragraph “Administration Groups”, this static Group behavior results in several challenges as described in the same Paragraph.

Enterprise Manager 12c introduces the concept of “Dynamic Groups” that allow us to specify criteria; Targets should meet to automatically become a Member of the Group. The criteria mentioned must be entered when the Dynamic Group is created.

---

Figure 17 shows an example of the creation of a Dynamic Group based on Target Type ‘Database Instance’ and Lifecycle Status ‘Production’.

As ‘Administration Groups’ are the best approach (Oracle best practice) to do the setup of Monitoring Settings, ‘Dynamic Groups’ would be your best bet when setting up Groups to support Jobs and Reports.

Having said this, we might want to say that the usage of ‘Static’ Groups should be avoided as much as possible.
Incident Management

Starting with Enterprise Manager 12c, the monitoring of Datacenter components isn’t focused on Events anymore but on Incidents instead.

Where Enterprise Manager 10 and 11g focused on the monitoring of Events, where an Event could be: ‘Not enough available space in Tablespace’ or ‘CPU usage exceeds 80% for more than 30 minutes’, Enterprise Manager 12c introduces the model where Events can be logically bundled in a single Incident.

You might not interested in the fact that CPU consumption on a Host is more than 80% for a certain time period, but you might be interested in the fact that CPU consumption on a Host is more than 80% and that at the same time MEM consumption is more that 70% and Disk I/O is way above acceptable limits. In this example we say that we want to be notified as soon as the Incident ‘Server Overload’ occurs. This Incident would be created automatically when all three Events mentioned occur.

Figure 18 Incident created based on single Event

Figure 18 shows the situation where a single Event “Database DB1 is down” result in the creation of an Incident. Please notice that the Incident includes information like: Status, Severity, Priority and a Comment made by the Administrator who is currently working on it.

Figure 19 Incident created based on multiple Events

Figure 19 shows the situation where multiple Events result in the creation of an Incident.
Lifecycle – Workflow of an Incident
As soon as an Incident is created several features become available that allow you to manage the complete lifecycle of the Incident
- Assign incident ownership.
- Track the incident resolution status.
- Set incident priority.
- Set incident escalation level.
- Ability to provide a manual summary.
- Ability to add user comments.
- Ability to suppress/unsuppress
- Ability to manually clear the incident.
- Ability to create a ticket manually.

Incident Rules and Rule Sets
Based on Incident Rules and Rule Sets, automated actions can be defined based on conditions like ‘Severity’ or the time an Incident has not been acknowledge for longer than 2 hours, etc.

Incidents can be created based on Event, Incident of Problem related Criteria. Some examples of these Criteria:
- Rule applies if the newly raised incident does not have an owner.
- Rule applies to a specific lifecycle status for a target. Lifecycle status is a target property that specifies a target's operational status.
- Rule applies to a specific target type.
- Each problem has a problem key, which is a text string that describes the problem. It includes an error code (such as ORA 600) and in some cases, one or more error parameters.
- Rule applies when the problem escalation level matches the selected level. Available escalation levels: None, Level 1, Level 2, Level 3, Level 4, Level 5

Events, Incidents and... Problems
If a critical error is encountered within the Oracle Software, a Problem is created. A Problem therefor represents a root cause for an Oracle Software related Incident. To support the resolution of Problems you can use the Support Workbench that is integrated with Enterprise Manager to gather and package diagnostic information from the ADR and to create a SR in My Oracle Support.

Lifecycle Management
When looking at Enterprise Manager 12c one remark that can be made is that it is increasingly supporting the complete lifecycle of Targets.

Actually, when thinking of an Organization starting up a Project to create new marketable Services (Products) for their Customers, we could say that Enterprise Manager 12c support the complete Lifecycle of this Service (in terms of IT related activities).
Provisioning
First introduced in EM10.2, the Provisioning Framework has evolved enormously, considering the number of ‘out of the box’ Deployment Procedures and extensions to the Framework.

When trying to think of the added value of automatic deployment of Datacenter components, try thinking of the enormous growth of Datacenters in terms of Servers, Storage Boxes, Network Components, and Engineered Systems etc.

The creation, configuration, monitoring and management of hundreds or even thousands of components is getting complex just because of the number of it.

The Provisioning Framework in EM12c allows Administrators to provision components like:

- Databases (Single Instance, RAC, One Node RAC)
  - Including Scale Up / Down RAC Nodes
- Cluster Infrastructure
  - Including Scale Up / Down Cluster Nodes
- Automatic Storage Management
  - Single Node and Cluster ASM
- Weblogic Domain
  - Including Scale Up / Down
- SOA Composites
- SOA Artifacts
- Coherence Nodes
- J2EE Applications
- BPEL Process
- Oracle Homes (Database and Middleware)
• Assemblies
  o Scale Up / Down Assembly Tier
• Virtual Guests
• Bare Metal

Apart from that, patching and upgrading of most of these components is supported as well.

Figure 21 System Lifecycle

Figure 20 shows the Lifecycle of a System (or Service what's in a name...) as starting with the installation of Software, rolling out Patches, configuring Monitoring Settings, administering and finally the removal at the end of the Lifecycle (decommissioning).
Figure 22 The principle of Golden Images

Figure 21 shows the principle of Golden Images as a Clone based on a Reference System (could be an Oracle Home (DB or MW), or a DBCA Template, etc.)

The Golden Image will then be stored in the Software Library and from here deployed to a System (for instance a Production System).

Figure 23 Principle of Automated Patching
Figure 22 shows the principle of automated patching of Targets. First the Patch(es) is (are) identified within My Oracle Support (which is fully integrated with Enterprise Manager 12c) and included in a Patch Plan. Next the Patch Plan is deployed to one or more Targets.

**Software Library**

The EM Software Library is a single location where Software Entities (Components) like patches, virtual appliance images, reference golden images, application software and associated directive scripts (Directives) are stored.

The Entities referred to are also known as 'Components'. You might compare Components and Directives with Ingredients and Recipes.

When focusing on Provisioning and taking the provisioning of a Database as an example we would have the a ‘Database Oracle Home as Component’ and several Directives like: ‘Create spfile’, ‘Deploy Database’, ‘Run Database Config Tools’, ‘Run Prereq and Fixups’ that are combined in a Deployment Procedure ‘Provision Oracle Database’.

**Deployment Procedures**

As mentioned in this chapter, the Provisioning Framework comes with many “Out of the Box” Deployment procedures. Approximately 40 (depending on what Plug-ins are installed).

As mentioned, these Deployment Procedures consists out of Directives that are bundled together in a Procedure and perform actions on / with Components like Database, Middleware and even Applications.

Directives are based on scripts where most of the scripts delivered “out of the box” are Perl scripts. You are however allowed to use other scripting like bash or korn shell or Windows batch scripts. Using Perl script gives us the benefit of multi-platform support.

**Procedural logic**

![Deployment Procedure Logic]

*Figure 24 Deployment Procedure Logic*
Figure 24 shows the Procedural Steps in the “Provision Oracle RAC Database” Procedure. Each of the Steps has an “Error Handling” logic that can be: “Inherit (Stop on Error)”, “Stop on Error”, “Continue on Error” or “Skip Target”, indicating what should happen in case the Step end in an Error.

Road to Self Service
Apart from the fact that using the Provisioning Framework is the way to go when it comes to enforcing standardization, efficiency and optimization of “Time to Market”, the Provisioning Framework is “the Road” to Self Service.

Read more on this in paragraph “Cloud Management”.

Patching
Another major feature within Lifecycle Management is “Automated Patching”.

This feature works based on integration with My Oracle Support.

Based on the integration with My Oracle Support the EM12c Console will show you Patch Recommendations if there are any patches available that applies to one of your Targets. From the EM12c you would then have direct access to the available Patches in MOS and from there start installing them.
Figure 25 shows the Patch Recommendations in the EM12c Console. Selecting the All Recommendations link would navigate the user to the MOS Patch Recommendations Page.

Figure 26 Patch Recommendations Page in MOS

Patch installation can than be done using a Patching Plan. A Patching Plan allows you to include multiple Patches, analyze for possible conflicts and next deploy on any number of Targets.

Figure 27 Deploying Patches using a Patch Plan

**Configuration Management**

*According to Gartner, “Through 2015, 80% of outages impacting mission-critical services will be caused by people and process issues, and more than 50% of those outages will be caused by change/configuration/release integration and hand-off issues.”* ¹

---

¹ Gartner Research, Top Seven Considerations for Configuration Management for Virtual and Cloud Infrastructures, 2010
Configuration management (CM) is the detailed recording and updating of information that describes an enterprise’s hardware and software. It is a process for establishing and maintaining consistency of a product’s performance and functional and physical attributes with its requirements, design and operational information throughout its life. (Wikipedia)

A configuration management database (CMDB) is a repository of information related to all the components of an information system. It contains the details of the configuration items (CI) in the IT infrastructure. Although repositories similar to CMDBs have been used by IT departments for many years, the term CMDB stems from ITIL. In the ITIL context, a CMDB represents the authorized configuration of the significant components of the IT environment. A CMDB helps an organization understand the relationships between these components and track their configuration. The CMDB is a fundamental component of the ITIL framework’s Configuration Management process. (Wikipedia)

Getting grip on your Configurations therefore should be one of your priorities. Getting grip includes:
1. Knowing what the configuration of a specific Target Type should look like (Baseline)
2. Knowing what the configuration of a specific Target looks like
3. Knowing what differences there are between the current configuration of a Target and the configuration Baseline
4. Knowing when a change has been made in the current configuration and by whom it was made
5. Knowing if any change that has been made in the current configuration is an known and approved change

Yes, we are talking ITIL now!

When discussing Configuration Management we are talking about a CMDB (Configuration Management Database) and as accustomed we have gotten to Oracle Enterprise Manager we know that the OMR (Oracle Management Repository) as one of the base components within the EM architecture, actually is our CMDB.

Tenth, to hundreds of different properties (like name, version, vendor etc.) that make out the Configuration of our Targets, are gathered on frequent base and stored in Management Repository (CMDB). This indicates the presence of a “Latest Configuration” that will be refreshed based on a “Current Configuration”.


Figure 28 Latest Configuration EMGC_GDomain

Figure 28 shows the Latest Configuration Page of EMGC_GDomain.

Enterprise Manager 12c comes with many standard insights (queries and reports) to search the CMDB, including associations like “is used by...”, “uses...” and in graphical presentation data like Topology.

“Out of the Box” CMDB searches include:
- Database Tablespaces
- Initialization Parameter Settings
- Database Datafiles
- Oracle Weblogic Server: Web Modules
- Oracle Weblogic Server: Ports
- Oracle Weblogic Server: Data sources
- Oracle Weblogic Server: EJB Modules
- Oracle Weblogic Server: Deployed Applications
- Applied Patches on Oracle Products
- Etc.

Creation of custom or customized Searches is fully supported.
Like mentioned earlier, the configuration of a Target is refreshed on frequent base. As the configuration data of a Target gets overwritten with every refresh, you might want to preserve the configuration at a specific moment in time.

Figure 29 Configuration Topology

Saved Configurations
An example could be the moment when a newly created WebLogic Domain is created, you might want to save the configuration in a “Configuration Baseline”, so that you can make comparisons with this Baseline on any moment in time and by that be able to indicate configuration drifting.

**Compare Configurations**

As discussed in the previous paragraph, you might want to compare the configuration of a Target with a Configuration Baseline or maybe the configuration of multiple Targets with one specific Configuration Baseline of the configuration of one specific Target.

EM12c allows you to do so and also allow you to configure notifications when certain differences are detected.

![Figure 30 Comparison results](image)

Figure 30 shows the results of a comparison between the Configurations of two Hosts.

**Comparison Templates**

As you don’t want to be bothered by obvious differences in properties like “Name” you can use a “Comparison Template” to exclude such properties during comparisons.

**Compliance Management**

*Compliance means conforming with stated requirements. At an organizational level, it is achieved through management processes which identify the applicable requirements (defined for example in laws, regulations, contracts, strategies and policies), assess the state of compliance, assess the risks and potential costs of non-compliance against the*
projected expenses to achieve compliance, and hence prioritize, fund and initiate any corrective actions deemed necessary. (Wikipedia)

So we know we need to be compliant to several regulations, like a European Bank would need to be compliant to regulations like:

- MiFID (Markets in Financial Instruments Directive)
- EMIR (European Market Infrastructure Regulation)
- SEPA (Single Euro Payment Area)
- CRD / Basel (Capital Requirements Directive / Basel Accord)
- Solvency
- SOX (Sarbanes – Oxley)
- Brussels regulation
- Local Governmental
- Local Bank
- Etc.

Several of these have impact on IT infrastructure, like accessibility of binaries, database accounts / contents etc.

Let’s see what features are available in EM12c to check our systems on Compliancy.

**Compliance Frameworks**

A Compliance Framework can be seen as a set of specific Compliance Standards that associate to Compliancy Rules.

An example of a Compliance Framework might be **PCI DSS**.

*The Payment Card Industry Data Security Standard (PCI DSS) is a widely accepted set of policies and procedures intended to optimize the security of credit, debit and cash card transactions and protect cardholders against misuse of their personal information. Four major credit-card companies created the PCI DSS jointly in 2004: Visa, MasterCard, Discover and American Express.*

PCI DSS is one of the Compliance Frameworks available ‘out of the box’ in EM12c.

![Figure 31 PCI DSS as a Compliance Framework](image)
Figure 31 shows PCI DSS as one of the ‘out of the box’ Compliance Frameworks. Furthermore we see several Oracle ‘Best practice’ Compliance Frameworks.

A Compliance Framework in EM12c implies one or more Compliance Standards. If we again take a look at PCI DSS we would see the following Standards:

<table>
<thead>
<tr>
<th>Standards</th>
</tr>
</thead>
<tbody>
<tr>
<td>Build and Maintain a Secure Network (PCI 1.2)</td>
</tr>
<tr>
<td>Establish firewall and router configuration standards (PCI 1.1)</td>
</tr>
<tr>
<td>Restrict untrusted network access to Cardholder Data environment (PCI 1.2)</td>
</tr>
<tr>
<td>Restrict internet network access to Cardholder Data environment (PCI 1.3)</td>
</tr>
<tr>
<td>Do not use default configurations (PCI 2.1)</td>
</tr>
<tr>
<td>Security configuration standards for all components (PCI 2.2)</td>
</tr>
<tr>
<td>Storage Best Practices for ASM</td>
</tr>
<tr>
<td>Storage Best Practices for Oracle RAC Database</td>
</tr>
<tr>
<td>Weblogic Cluster Configuration Compliance</td>
</tr>
<tr>
<td>Storage Best Practices for Oracle Database</td>
</tr>
<tr>
<td>Weblogic Server Configuration Compliance</td>
</tr>
<tr>
<td>Configuration Best Practices for Oracle Database</td>
</tr>
<tr>
<td>Weblogic Cluster Configuration Compliance</td>
</tr>
<tr>
<td>Configuration Monitoring for Networking Linux Packages</td>
</tr>
<tr>
<td>Configuration Monitoring for Exadata Compute Node Networking</td>
</tr>
<tr>
<td>Weblogic Domain Configuration Compliance</td>
</tr>
<tr>
<td>Encrypt all administrative access using SSH, VPN, or SSL/TLS (PCI 2.3)</td>
</tr>
</tbody>
</table>

- Protect Cardholder Data (PCI3)
  - Mask account: numbers when shown (PCI 3.3)
  - Render account numbers unreadable when stored (PCI 3.4)
  - Protect crypto keys (PCI 3.5)
  - Follow processes with managing keys (PCI 3.6)

- Maintain a Vulnerability Management Program (PCISS, PCISG)
  - Deploy anti-virus software on all systems (PCI 5.1)
  - Ensure anti-virus software is up to date, running and capable of reporting violations (PCI 5.2)

- Ensure all software is at latest patch level (PCI 6.1)
  - Patchable Configuration For Cluster
  - Patchable Configuration For ASM
  - Patchable Configuration For RAC Database
  - Patchable Configuration For Oracle Database
  - Address threats and vulnerabilities for public-facing web applications (PCI 6.6)

- Implement Strong Access Control Measures (PC17, PC18)
  - Configuration Monitoring for Security Linux Packages

- Limit user access to cardholder data (PCI 7.1)
  - Configuration Monitoring for User Access Linux Packages
  - Restrict user access to a need-to-know basis; deny all by default (PCI 7.2)
  - Proper authentication and password management for non-consumer users and administrators (PCI 8.5)

- Regularly Monitor and Test Networks (PC10, PC11)
  - Link all administrative access to individual users (PCI 10.1)
  - Implement automated audit trails (PCI 10.2)
  - Record many attributes of audit trail entities for reconstruction (PCI 10.3)

- All clocks must be synchronized (PCI 10.4)
  - Configuration Monitoring for Exadata Compute Node Time
  - Configuration Monitoring for Network Time Linux Packages
  - Secure audit trails (PCI 10.5)
  - Retain audit history for one year. Three months immediately available (PCI 10.7)

- Deploy file integrity monitoring for all critical files (PCI 11.5)
  - Configuration Monitoring for Exadata Compute Node
  - File Integrity Monitoring for important Linux packages
  - Configuration Monitoring for Core Linux Packages
  - File Integrity Monitoring for Exadata Compute Node

Figure 32 A detailed overview of Compliance Standards in PCI DSS
Figure 32 shows an overview of all Compliance Standards (and Compliance Framework Subgroups) that make up the PCI DSS Compliance Framework.

Let's take a closure look to some of the Standards included:

![Figure 33](image1.png)

**Figure 33 A closer look at one of the Compliance Standards**

Figure 33 shows a detail of Compliance Standard “Weblogic Domain Configuration Compliance”

If we would search for this Compliance Standard in the Library we would find:

![Figure 34](image2.png)

**Figure 34 Compliance Standard “Weblogic Domain Configuration”**

Selecting the Standard will show us in detail what Standard Rules are included
Figure 35 Compliance Rule Details

Figure 35 shows the details of Compliance Rule ‘Administration Port Enabled’. Apart of a description of the rule we also see the SQL statement that is executed to check the rule.

- **Repository Rules**

  Used for checking the configuration state of one or multiple targets. A rule is said to be compliant if it is determined that the configuration items do in fact meet the desired state; that is, the rule test failed to identify any violations. Otherwise, a rule is said to be non-compliant if it has one or more violations. The data source that is evaluated by a compliance standard rules test condition can be based on a repository query. A compliance standard rules test condition can be implemented using a
threshold condition based on the underlying metrics (or queries) column value or SQL expression or a PLSQL function.

• **WebLogic Server Signature Rules**

Contain a check condition, which will be checked on the WebLogic Server. Guardian signature rules are examples of WebLogic Server signature based rules.

Guardian signature rules describe potential problems based on information about WebLogic Servers and the environment in which they are deployed, including Java Virtual Machines (JVMs), operating systems, and databases. Signature rules contain executable logic that can identify specific versions of these products, as well as their configuration settings.

• **Real-time Monitoring Rules**

Monitor low-level configuration objects in real-time to determine exactly what changes took place, when, and by whom. These rules monitor Process, OS User, Database tables, views, index, user, Windows Registry key, Active Directory Group, and so on. These rules contain configuration parameters specifying what entities they will be monitoring, for instance, what files to monitor, how to monitor (for example, operations (read/write)), when to do the monitoring (time-period), who to monitor (user name).

*Source Enterprise Manager Online Help*

![Diagram](image)

**Figure 36 Compliance Framework Hierarchy**

Figure 36 shows the Compliance Framework Hierarchy, where a Compliance Framework may include multiple Compliance Standards that each might include other

**Real-time Monitoring**

Real-time Monitoring allows for monitoring of specific Target Types. A Target Type is made of entities, like files, processes, users or database tables.

Each of these entities are described by means of a Facet that can be created based on a Target Type like: Host, Database Instance, Agent, Cluster and even JVM.

Real-time Monitoring facets based on target types are used to specify the entities to monitor in real-time monitoring rules. As an example, if monitoring a host for file changes, a facet can be a list of distinct single files, patterns with wildcards that would include many files, or simply an entire directory. These patterns can also include parameters that have a default, but can be overridden as needed for each target. Built-in parameters, such as ORACLE_HOME will be dynamically filled in for each target. If you wanted to specify monitoring the database configuration file tnsnames.ora, your pattern may be `{ORACLE_HOME}/network/admin/tnsnames.ora`.

Checkout [Oracle® Enterprise Manager Lifecycle Management Administrator’s Guide](http://docs.oracle.com/cd/E24628_01/em.121/e27046/compliance_lcm.htm#CIAIICAF)

<table>
<thead>
<tr>
<th>Entity Types supported by Real-Time monitoring</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>OS File</td>
<td>Oracle Database Table</td>
</tr>
<tr>
<td>OS Process</td>
<td>Oracle Database View</td>
</tr>
<tr>
<td>OS User</td>
<td>Oracle Database Procedure</td>
</tr>
<tr>
<td>Microsoft Windows Registry</td>
<td>Oracle Database User</td>
</tr>
<tr>
<td>Microsoft Active Directory User</td>
<td>Oracle Database Index</td>
</tr>
<tr>
<td>Microsoft Active Directory Computer</td>
<td>Oracle Database Sequence</td>
</tr>
<tr>
<td>Microsoft Active Directory Group</td>
<td>Oracle Database Function</td>
</tr>
<tr>
<td>Oracle Database Dimension</td>
<td>Oracle Database Profile</td>
</tr>
<tr>
<td>Oracle Database Synonym</td>
<td>Oracle Database Public Synonym</td>
</tr>
<tr>
<td>Oracle Database Type</td>
<td>Oracle Database Role</td>
</tr>
</tbody>
</table>
Figure 37 Real-Time Monitoring Facets

Figure 37 shows the ‘Out of the box’ Real-Time monitoring facets.

Figure 38 Kernel Boot Config Files

Figure 38 shows a more detailed view on Facet ‘Kernel Boot Config Files’. This facet supports the detection of for instance modification of any of the files included.
The Compliance Dashboard now shows in one page the exact situation for a selected Compliance Framework. Figure 39 shows an example of the 'PCI DSS' Compliance Framework.

Another thing you should notice is the ‘Average Compliance Score’ that ideally should be 100% and gives the percentage your targets are compliant.

Cloud Management

Cloud computing offers convenient, on-demand access to a shared pool of computing resources including networks, servers, storage, applications, and services—that can be rapidly provisioned and released with minimal management effort. Typically, cloud computing is made available in one of three deployment scenarios—Infrastructure-as-a-Service (IaaS) offering physical and virtual systems, including an operating system, hypervisor, raw storage, and networks; Platform-as-a-Service (PaaS) providing the infrastructure plus a solution stack including program language execution environment, Database, and Web server; and Software-as-a-Service (SaaS) offering a platform plus applications. (Oracle White paper “Making Infrastructure-as-a-Service in the Enterprise a Reality”)
Instead of purchasing Serves, Software, Datacenter floor space and Network equipment, resources are claimed on an “as and when needed base”. Business units will need Datacenter resources to startup projects, development of new Business Services, testing of new releases of Application before deploying them in the Production environment.

And when a Business unit needs a certain quantity of resources it might need these resources only for 2 weeks and only during normal office hours between 08:00 AM and 05:00 PM. When considering this example from a ‘Cloud Computing’ perspective we would want to make sure that during this given timeframe the resources are available and that: after 2 weeks the test can be removed. By this resources would be released and available for new resource requests.

In previous discussed example where a Business unit would want to have a complete Platform for two weeks to test a new release of an Application, other requests might just concern the creation of a Weblogic Domain or a Database or even just a Database Schema.

So apart from Cloud Services like IaaS, PaaS and SaaS we also need to consider DBaaS, Schema-aaS and MWaaS.

When talking about Cloud Services we should take in account that there is a difference between Public and Private Clouds. We will continue discussing the Private Cloud concept where we will be responsible for the complete management of resources in the Cloud and provisioning of these resources on user request.

Let’s now take a look how Oracle Enterprise Manager 12c supports the management of the Cloud Lifecycle.
Planning

One of the questions we need to answer during planning of our Cloud would be “What would be the size of resources we need to provide in the Cloud to support all Business needs?”

As we might have the luxury situation where we can use brand new hardware, software or even a new datacenter, we might want to reuse hardware and software. A very interesting question than would be “If we want to move from currently used hardware to ‘new’ hardware and want to optimize overall utilization, how many new servers do we need?”

EM12c provides us with the Consolidation Planner that helps us to answer just that question. The Consolidation Planner is able to advise us to move from Physical to Virtual (P2V), Physical to Physical (P2P), or Physical to an Exadata solution.

Figure 41 Cloud Lifecycle

Figure 42 Creation of a Consolidation Project
Setup
As EM12c offers features like Bare metal provisioning, setting up of servers and storage pools in Zones that are based on functional (i.e. Line of Business or Application) or technical (i.e. Database version) aspects and supports the modeling of infrastructure-as-a-service (IaaS), middleware-as-a-service (MWaaS), and database-as-a-service (DBaaS) clouds. Oracle Virtual Assembly Builder allows for packaging of a Multi Tier platform into a single Meta driven Cloud Service. Think of a complex Siebel Platform that includes multiple servers with Databases and Middleware components and the Application Stack.

Such an assembly can then be published to the Enterprise Manager Software Library where it can be made available for developers as a Cloud service.

Building
Using EM12C, entire Applications can be published to the Cloud as a service. Testing teams, Business analysts or Production teams can now deploy these pre-build applications with just a mouse click.

Testing and Deploying
Enterprise Manager 12c provides tooling for testing both Application and Database.

The testing solution provides the ability to capture a production load and replay in a test environment, so that the results are predictable.

After services have been published to the Cloud, end-users can request these services using the Self Service Portal. For each request made from the Self Service Portal the amount of necessary resources (like CPU, MEM and Storage) can be specified. Based on available quota end-users can than request for a service. Based on Cloud policies resources can be scaled out or back based on a specific schedule or performance metrics.
Monitoring and Managing

Many facilities are available in EM12c to manage an entire Software Stack and detail monitoring of Targets in a “Bottom Up” or a complete Business Function (such as HR or Sales) in a “Top Down” (aka “Application to Disk”) approach.

Using the Service Level Management Pack the availability of a complete Business Service can be defined by means of ‘Syntactical Test’ and Performance Thresholds, so that Service Level Agreements can be monitored.

Using REUI and Business Transaction Management performance and availability of Applications and Transactions can be monitored.

Metering, Charging and Optimization

As part of the Cloud computing model, users should only pay for what they have actually been using. Also referred to as the “Cafeteria Model”.

Based on Charge Plans, dynamic resources like CPU, MEM and Storage but also static resources like Datacenter facilities and Software Licenses can be charged to individual or grouped Targets.

Figure 44 Chargeback Report in EM12c
Self Service Portal

Users with the SSA_USER roles are able to login to the EM Self Service Portal and request for

The Self Service Template that is available for selection was created in an earlier stage based on the 'out of box' Create Oracle Database Procedure

Mobile

For those Administrators that are constantly on the move, Oracle provides a mobile app for iPhone.