

HP Asset Manager Software License Optimization Best Practice Package

For the Windows[®] operating system

Software Version: 9.5.12.3

Technical Guide

Document Release Date: April 2016

Software Release Date: April 2016



Legal Notices

Warranty

The only warranties for HP products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. HP shall not be liable for technical or editorial errors or omissions contained herein.

The information contained herein is subject to change without notice.

Restricted Rights Legend

Confidential computer software. Valid license from HP required for possession, use or copying. Consistent with FAR 12.211 and 12.212, Commercial Computer Software, Computer Software Documentation, and Technical Data for Commercial Items are licensed to the U.S. Government under vendor's standard commercial license.

Copyright Notice

© Copyright 1994 - 2016 Hewlett-Packard Development Company, L.P.

Trademark Notices

Adobe® is a trademark of Adobe Systems Incorporated.

Microsoft® and Windows® are U.S. registered trademarks of Microsoft Corporation.

UNIX® is a registered trademark of The Open Group.

Documentation Updates

The title page of this document contains the following identifying information:

- Software Version number, which indicates the software version.
- Document Release Date, which changes each time the document is updated.
- Software Release Date, which indicates the release date of this version of the software.

To check for recent updates or to verify that you are using the most recent edition of a document, go to: <http://h20230.www2.hp.com/selfsolve/manuals>

This site requires that you register for an HP Passport and sign in. To register for an HP Passport ID, go to: <http://h20229.www2.hp.com/passport-registration.html>

Or click the **New users - please register** link on the HP Passport login page.

You will also receive updated or new editions if you subscribe to the appropriate product support service. Contact your HP sales representative for details.

Support

Visit the HP Software Support Online web site at: <https://softwaresupport.hpe.com/>

This web site provides contact information and details about the products, services, and support that HP Software offers.

HP Software online support provides customer self-solve capabilities. It provides a fast and efficient way to access interactive technical support tools needed to manage your business. As a valued support customer, you can benefit by using the support web site to:

- Search for knowledge documents of interest
- Submit and track support cases and enhancement requests
- Download software patches
- Manage support contracts
- Look up HPE support contacts
- Review information about available services
- Enter into discussions with other software customers
- Research and register for software training

Most of the support areas require that you register as an HP Passport user and sign in. Many also require a support contract. To register for an HP Passport ID, go to:

<http://h20229.www2.hp.com/passport-registration.html>

To find more information about access levels, go to:

http://h20230.www2.hp.com/new_access_levels.jsp

HP Software Solutions Now accesses the HPSW Solution and Integration Portal Web site. This site enables you to explore HP Product Solutions to meet your business needs, includes a full list of Integrations between HP Products, as well as a listing of ITIL Processes. The URL for this Web site is

<http://h20230.www2.hp.com/sc/solutions/index.jsp>

Contents

Contents	3
Introduction	9
SAM business model	9
Phase 1. Preparing data	9
Phase 2. License consumption calculation	10
Phase 3. Result presentation	11
Normalization and Non-normalization	11
Why normalizing software installation model?	11
Normalization solution (recommended)	12
Non-normalization solution	15
How does software counters work	16
Simple process	16
Advanced process	17
Technical details of software counters (Per supplier)	21
Microsoft	21
Microsoft software counter overview	21
Terminologies used in algorithm	21
Using Microsoft Windows Server software counters	22
Pre-condition: Involved SLO business data during calculation	22
License type	22
Workflow for pre-calculation	22
Windows Server license rules	23
Algorithm of Windows Server pre-calculation	24
Limitations	28
Cluster examples	29
Scenario 1. Infrastructure cluster	29
Scenario 2. OS cluster	30
Using Microsoft SQL Server software counters	31
Pre-condition: Involved SLO business data during calculation	31
License type	31

Workflow for core factor population	31
Workflow for pre-calculation	32
Core factor table	33
SQL Server license rules	33
Algorithm of SQL Server pre-calculation	37
General process	38
Calculation process on standalone physical computers	39
Calculation process in infrastructure layer cluster	40
Limitations	41
Cluster examples	42
Infrastructure cluster	42
Application cluster	43
Using Microsoft Office/Visio/Project software counters	46
Pre-condition: Involved SLO business data during calculation	46
License type	46
Workflow for pre-calculation	46
License rules	46
Using Microsoft desktop OS software counters	47
Pre-condition: Involved SLO business data during calculation	47
License type	47
Workflow for pre-calculation	47
License rules	47
Additional information 1. Acquiring the number of VM hosts (Microsoft)	48
Storing the VM migration log in Asset Manager	48
Algorithm	48
Additional information 2. SQL Server core factor table	49
Oracle	49
Oracle software counter overview	49
Oracle DB counter	50
Oracle DB option counter and Enterprise Management Pack counter	50
Oracle Weblogic	51

Terminologies used in algorithm	51
Using Oracle software counters	52
Pre-condition: Involved SLO business data during calculation	53
License type	53
Workflow for core factor population	53
Core factor table	54
Hardware partitioning type list	54
Oracle processor license rule	54
Algorithm of pre-calculation	55
General process	55
Recognizing invalid installations	56
Calculation process on standalone physical computer	57
Calculation process in infrastructure layer cluster	59
Limitations	60
Using Oracle cluster with Asset Manager	60
Additional information 1. Algorithm for Oracle software product on LPAR	63
Algorithm	63
The fields that stand for core capacity of LPAR	64
Additional information 2: Oracle core factor table	64
Customization on amSAMFactorType:\$DOUBLE_FACTOR (the CPU core factor table)	65
Additional information 3. Oracle hard partitioning technology list	67
Customization on amSAMVMPartition:Hard	67
IBM	67
IBM software counter overview	67
Two solutions for IBM software counters	67
Using IBM software counters	68
Pre-condition: Involved SLO business data during calculation	68
License type	68
Workflow for core factor population	68
Workflow for pre-calculation	69
Core factor table	69

IBM license rules	70
Algorithm of pre-calculation	70
Error handling	72
How to validate the IBM PVU counter result	73
Using IBM software counters (ILMT based)	75
Additional information 1. Algorithm for IBM PVU software product on LPAR	75
Additional information 2. IBM PVU core factor table	75
Adobe	78
Autodesk	78
BMC	78
HP	79
Using HP software counters	79
Pre-condition: Involved SLO business data during calculation	79
License rules	79
Red Hat	80
SAP	80
Using SAP counters	80
Symantec	81
Tibco	81
VMWare	81
Appendix 1. Data model and definition	82
Infrastructure data model	82
Inventory data used by software counter	82
License asset	82
Software installation	83
Computer	83
Cluster	85
VM migration log	88
Inventory data used by Oracle LMS solution	88
Software counter data model	90
Analysis result tables for consumption details	90

Table: amSoftLicCounter	90
Table: amSAMNodeCompAnalysis	90
Table: amSAMClusterCompAnalysis	92
Table: amSAMInstCompAnalysis	93
Table: amSAMPoolCompAnalysis	93
Table: amSAMIssueAnalysis	94
Consumption detail table for IBM PVU products	95
amCoreFactorResult table	97
Health check data model	97
Table: amHealthCheckRule	97
Table: amHealthCheckItem	98
OOB check rules	99
Appendix 2: Developer's reference	100
Functions used to generate statistics	100
Software usage statistics	100
Software assurance renewal statistics	100
Appendix 3: Automation mechanism	101
Pre-calculation workflows	101
Software counters batch calculation	101
Dashboard generation	102
License version update according to maintenance contract	102
Automatic removal of invalid links between license models and servers	103
Clean old calculation result	103
Clean old missing installations	104
Sambp Archival	104
Appendix 4: How to build a cluster	105
Maintain cluster information manually	105
Cluster examples	105
Example 1: VMware vSphere Cluster	106
Example 2: Microsoft Windows Failover Cluster	107
Example 3: Microsoft SQL Server Mirroring Cluster	108

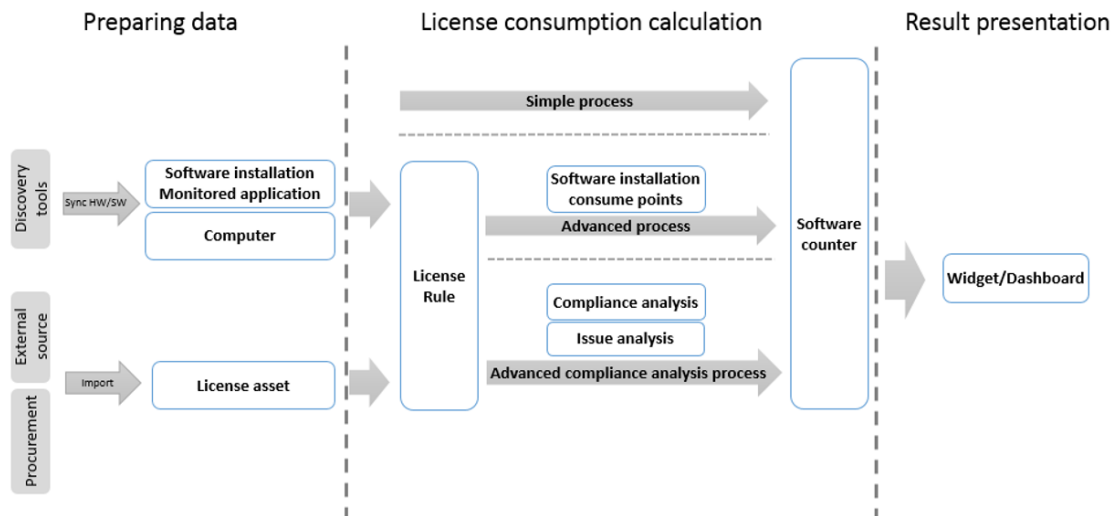
We appreciate your feedback!	110
------------------------------------	-----

Introduction

Software counter is an important component of the Asset Manager SLO Best Practice package, which calculates and stores the acquired and consumed software licenses in the environment. The Asset Manager SLO Best Practice package implements some out-of-the-box algorithms to calculate the compliance of the supported software products which have complicated license rules, such as Microsoft SQL Server and Oracle Database.

This document is intended for technical people who want to understand the technical details of the software counters and perform customization.

SAM business model



The figure above shows the standard process of software asset management (SAM). The inventory data including software installation and computers is imported into Asset Manager from discovery tools. Once license assets are in place, the software counters will calculate the license consumption and show compliance result in widgets, dashboards, or reports.

For complex licensing models such as Microsoft and Oracle, additional pre-calculation scripts (AM license rules) are required. Before software counters calculate the license consumption, license rules will perform pre-calculation and input temporary results to software counters.

The standard process includes 3 phases.

Phase 1. Preparing data

In this phase, Inventory data such as software installation, hardware information and license information needs to be provided. Usually, these information needs to be integrated with external systems.

- **Discovery tools**

Usually, software installation and computer information is inventory data that comes from discovery tools.

SLO Best Practice package or Asset Manager can integrate with the following discovery tools:

HPE UD, HPE DDML, Microsoft SCCM, Microsoft MAP, IBM ILMT.

- **External source**

The external source can be .txt file, .xml file, RESTFUL API, or database. It is optional to import license assets through the Connect-It tool.

- **Procurement**

Procurement is a key module in Asset Manager. You can create license assets in Asset Manager through the procurement process. For more information about the Procurement module, see the *Procurement* guide shipped with Asset Manager.

Phase 2. License consumption calculation

The software asset management module of Asset Manager provides 3 solutions for license consumption calculation: **Simple process**, **Advanced process**, and **Advanced compliance analysis process**.

- **Simple process**: This process is used for simple licensing rules such as "Per installation", "Per processor", "Per device", and so on. Objects including context table, query statement and license/installation calculation formula can all be defined in the software counter.
- **Advanced process** and **Advanced compliance analysis process**: Those two processes are more complex than the **Simple process** because they involve the pre-calculation scripts (AM license rules). The pre-calculation scripts will implement complex licensing rules published by software suppliers. They takes the following factors into consideration.
 - Software installation
 - Hardware information including cluster and virtualization technology
 - License asset
 - Core factor values
 - License downgrade (using the license of higher version to entitle lower versions)

The differences between these two solutions are:

- The **Advanced compliance analysis process** performs compliance analysis and keeps consumed license points per node, while the **Advanced process** directly writes consumed license points to the **IUseCount** field in the **amSoftInstall** table. That is, software counters

aggregate results from different places.

- The **Advanced compliance analysis process** performs issue analysis when running license rules, it writes the results to the issue analysis table. The **Advanced process** does not perform issue analysis.

The **Advanced compliance analysis process** allows you to know how the license were consumed per node or per cluster and helps analyze and track compliance result.

Phase 3. Result presentation

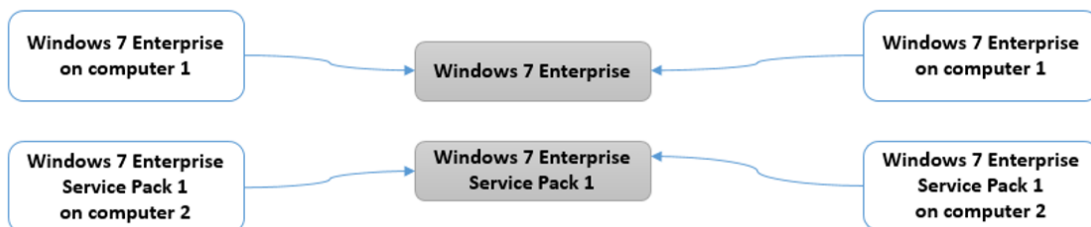
The SLO Best Practice package provides SLO widget, dashboard, business homepage and reports to show the license consumption results.

Normalization and Non-normalization

SAM supports normalization and non-normalization solutions when importing software installations into Asset Manager. The SLO Best Practice package does that as well.

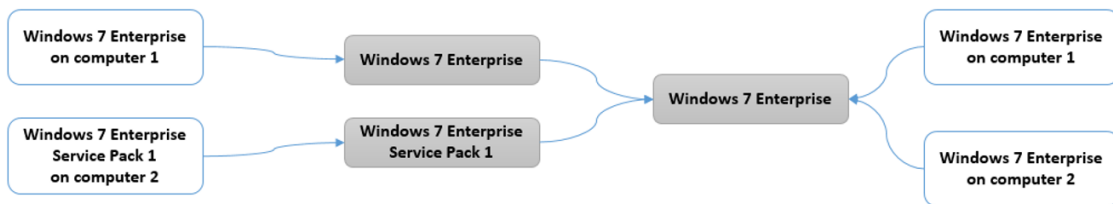
Why normalizing software installation model?

Take Windows 7 Enterprise as an example.

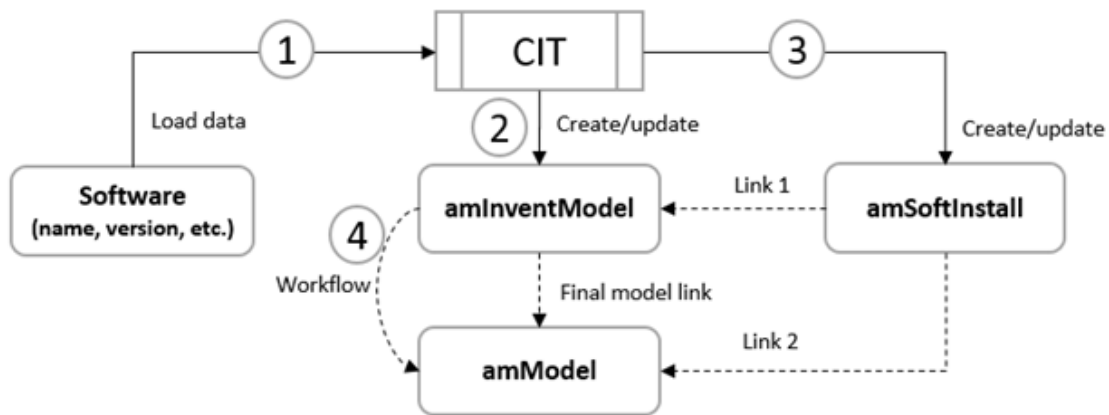


As shown in the figure above, the discovery tools detects Windows 7 Enterprise installation on Computer 1 and Windows 7 Enterprise Service Pack 1 on Computer 2. If you import the software installations into Asset Manager directly, there will be 2 software installation models created in the **amModel** table and the installations link to different models. Since one software title may have multiple software installation models, it is not convenient to manage software installations and license compliance in Asset Manager.

To avoid this issue, Asset Manage introduces inventory model, as shown in the figure below. First, an inventory model is created or updated in Asset Manager according to the discovered software installation. Then, the inventory model is linked to a definitive model in **amModel**. At last, it creates software installations linking to the definitive model.



Normalization solution (recommended)



As shown in the figure above, the normalization solution can be broken down into 4 steps.

Step 1. Connect-It loads a software installation from the network environment.

Step 2. Asset Manager creates or updates the inventory model (**amInventModel**) based on brand, name, and version. If the inventory model is new, the definitive model (links to **amModel**) will be empty and the temporary model (links to **amModel**) is shown as **Unknown software installation**. The user needs to resolve it by linking inventory model to definitive model. See Step 4.

Note: If the user uses the out-of-box integration solution such as Universal Discovery (UD) or Discovery and Dependency Mapping Inventory (DDMI), the relation between inventory model and definitive model is already set up.

Step 3. Asset Manager creates the software installation with the link to the model in the **amSoftInstall** table.

If the inventory model (in Step 2) has a link to the definitive model, the software installation will be linked to the definitive model. Otherwise, it will be linked to the temporary model named **Unknown software installation**. Once you specify a definitive model for the inventory model, the temporary model will be replaced with definitive model.

Step 4. There are two back-end workflows running to support the normalized solution.

- Workflow 1: **Normalize models: 'amInventModel' added** (reference name is **syslvtExtModelNew**)

This workflow helps you create new software installation models in the **amModel** table and assign it to the inventory model. There are two methods to do it.

Method 1. Model creation by wizard (manual and recommended)

If the connection is “((bResolved = 0) AND (bCreatedOnTheFly = 1)) AND (seType = 0)”, the workflow will add a workflow task to run a wizard that assists to resolve the inventory model.

Creation	Reference	Activity
Wednesday, January 20, 2016 12:38:48 P...	001541	Management of counters and new ...
Wednesday, January 20, 2016 05:59:39 ...	001542	Select definitive model

Display selections 0 selected records ☒ Action on the selections in current page Page 29 of 29 Total records: 580

Detail of task '001542'

Workflow:

Activity:


General | Comment | Assignment | Administration | History | Documents

Description

New inventoried models have been added to the 'amInventModel' table. These models must be resolved. To do this, you need to create a link between these models and models in the 'amModel' table.

[Detail...](#)
[Wizard...](#)

Click **Wizard**.... A wizard pops up.

 **Select software installation models to reconcile**

< Previous > Next ✓ Finish ✕ Cancel

Inventoried brand: Microsoft

Inventoried name: SharePoint Portal Server

Inventoried version: 2007 sp1 x64

Inventoried key: PD\I30444

Select an action...

- ☒ Resolve with an existing model
- ☐ Create a new corresponding model and map it
- ☐ Update without resolving

< Previous > Next ✓ Finish ✕ Cancel

Note: The “((bResolved = 0) AND (bCreatedOnTheFly = 1)) AND (seType = 0)” condition indicates that:

- seType = 0: It is an inventory model.
- bResolved = 0: The inventory model has not been linked to a final model.
- bCreatedOnTheFly = 1: It is possible to create records on the fly.
- bCreatedOnTheFly is the monitored field. Now it can not be updated manually on either web client or windows client. It is updated using Connect-It scenario or push adapter when creating inventory model.

By default, the user in the **Normalize models** group will receive a workflow task. If there is no user in the group, by default, the supervisor of the group will receive the task. To maintain the group, open the **Employee groups** screen and find **Normalize models**.

Method 2. Automatic model creation

If the condition is “(((bCreatedOnTheFly = 1) AND (bResolved <> 0)) AND (IModelFinalId = 0)) AND (seType = 0)”, the workflow will automatically create a new model with the necessary inventory information such as inventory name, version, and so on. Then, it sets the definitive model to the new model.

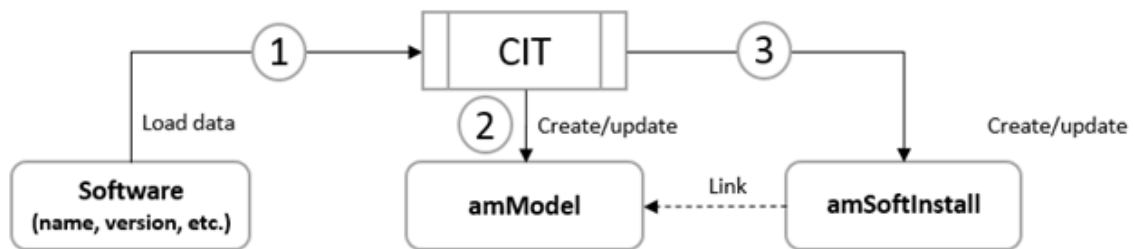
Note: The “(((bCreatedOnTheFly = 1) AND (bResolved <> 0)) AND (IModelFinalId = 0)) AND (seType = 0)” condition indicates that:

- seType = 0: It is an inventory model.
- (bResolved <> 0) AND (!ModelFinalId = 0): The inventory model has not been linked to a final model but the option of the resolve flag is checked.
- bCreatedOnTheFly = 1: It is possible to create records on the fly.
- bCreatedOnTheFly is the monitored field. Now it can not be updated manually on both web client and windows client. It is updated using Connect-It scenario or push adapter when creating inventory model.

- Workflow 2: **Normalize models: 'amInventModel' resolved** (Reference name is **syslvExtModelEnd**)

If in step 3, software installation is created with the link to the temporary model (unknown software installation), once the definitive model is specified, the workflow will automatically update related software installation with the definitive model. The software installation in **amSoftInstall** with the link to the inventory model will be replaced with the definitive model.

Non-normalization solution



As shown in the figure above, the non-normalization solution can be broken down into three steps.

Step 1. Connect-It loads software installations from the network environment.

Step 2. Asset Manager creates or updates software models in the **amModel** table according to the reconciliation rule defined in Connect-It mapping.

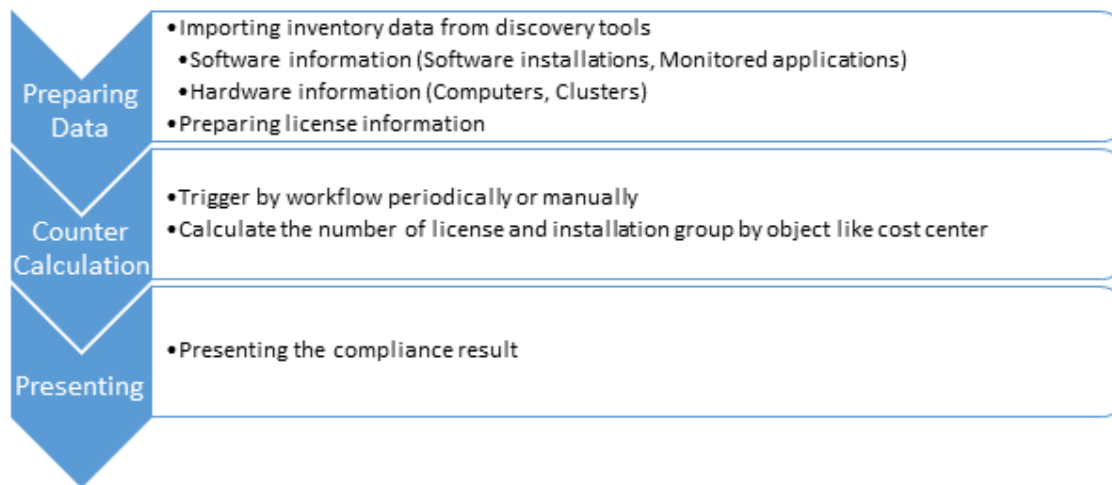
Step 3. Asset Manager creates software installations with the link to the model.

Note: No matter which solution you choose, you must check if the software counter has links to appropriate license models and installation models before calculating license compliance.

How does software counters work

Simple process

The **Simple process** is described as follows.



Phase 1. Preparing data

- Software installation and hardware information imported from discovery tools.
- License information.
- Using health check to check inventory data quality. Add the **Health check report** widget to the home page. Before running counter, fix the issues found by the health check rule.

Phase 2. Counter Calculation

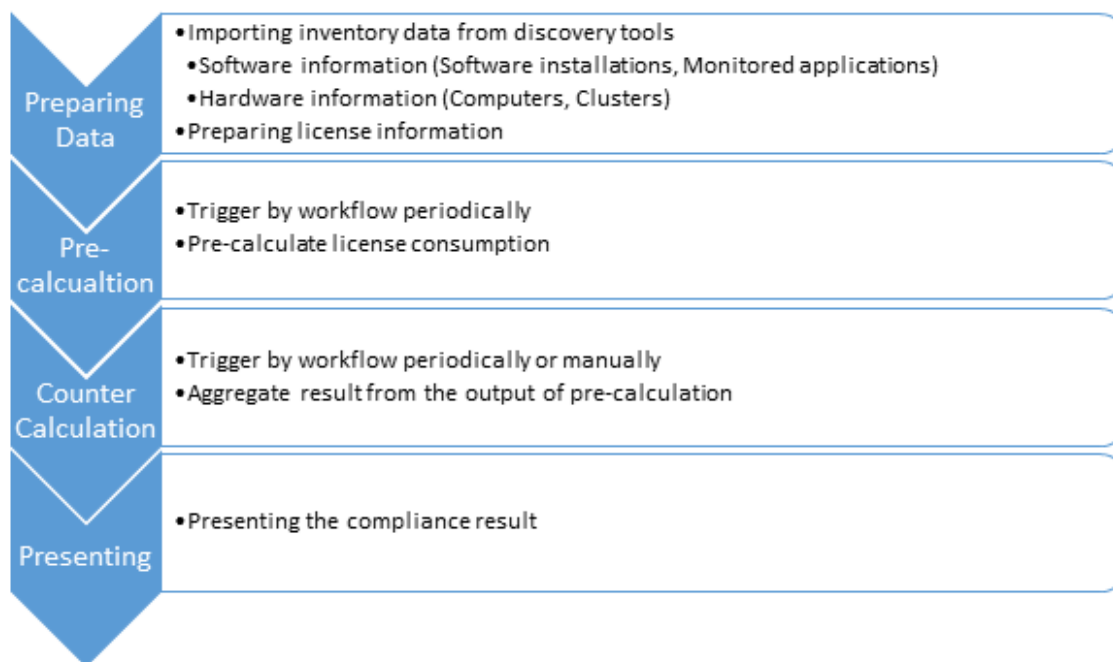
- Two ways to trigger the phase:
 - Use the **Software counters batch calculation** workflow (**sysSAM4CALCULATE_WK**) to batch calculate software counter. You can set the scheduler to meet the requirement.
 - Manually calculate the software counter by clicking the **Calculate** button in the software counter screen (Windows client) or choosing the **Recalculate the rights and the installations for the selected counters** action in the software counter screen (web client).
- Make sure that counters have linked the license models and installation models.
- Counter will calculate the number of consumed license points and the number of installations.

Phase 3. Presenting

- Add the SLO widget **Software licensing compliance report** to the home page to show compliance summary.
- Business homepage of SLO Best Practice package shows generic statistics.
- Dashboards show supplier related statistics.
- Some out-of-box Crystal Reports.

Advanced process

The **Advanced process** and **Advanced compliance analysis process** are described as follows.



Phase 1. Preparing data

In this phase, you need to prepare all required data and ensure the accuracy.

- Software installation and hardware information imported from discovery tools.
- License information.
- Using health check to check inventory data quality. Add the **Health check report** widget to the home page. Before running counter, fix the issues found by the health check rule.

Phase 2. Pre-calculation

In this phase, there are some supplier-related workflows running periodically to pre-calculate license consumption and the result will be aggregated by counter calculation.

- Supplier-related workflows to calculate core factors (if licensing model needs core factors). For example:
 - **Calculate Oracle core factor periodically (sysSAM4ORA_CF_WK)**
 - **Calculate Microsoft core factor periodically (sysSAM4MS_CF_WK)**
 - **IBM core factor detector (sysSAM4IBM_CF_WK)**
- Supplier-related workflows to calculate license consumption. For example:
 - **Oracle Installation pre-count (sysSAM4ORA_ORA_DB)**
 - **Microsoft installation pre-count (sysSAM4MS_MS_WK)**
 - **IBM installation pre-count (sysSAM4IBM_CNT_WK)**
- The differences between **Advanced process** and **Advanced compliance analysis process** are:
 - **Advanced process** will directly save consumed license points to the **IUseCount** field in **amSoftInstall**. Software counter will count the **IUseCount** field and group by object.
 - **Advanced compliance analysis process** will do compliance analysis and save consumed points into generic compliance analysis tables. Later, software counter will aggregate results from node analysis and cluster analysis table.
 - **amSAMClusterCompAnalysis**: keep consumed license points at cluster level.
 - **amSAMNodeCompAnalysis**: keep consumed license point at node level.
 - **amSAMInstCompAnalysis**: keep the software installations (monitored applications) which were covered by counter.
 - **amSAMPoolCompAnalysis**: keep capacity to be licensed at resource pool level. Currently, only IBM processor pool with PowerVM technology is supported.
 - **Advanced compliance analysis process** will do issue analysis during the calculation.
amSAMIssueAnalysis: keep detected issues during calculating. If issues were found, it is recommended to fix them before the next time you run counter calculation.
 - The **bRelyOnAnalysisTable** field tells you if the counter is using **Advanced compliance analysis process**. In that process, you do not need to define **Calculation** (on the **Installation/Utilizations** Tab) for installations. Currently, you cannot modify the **bRelyOnAnalysisTable** field.

Tip: There are 2 parameters that enable showing more consumption details during the pre-calculation. You can find these parameters by following these steps:

1. Log on to Asset Manager with an administrator user account.
2. On the web client, enter “scripts” in the search box on the top, and then find and open the **Scripts** screen. On the Windows client, on the **Administration** menu, click **Script libraries**.
3. Find and choose the **SamPreCalcUtilLib** record.
4. In the detail pane, look into the **Script** field, find the follow lines:

```
CONST DEBUG_ENABLED = x
```

and

```
CONST INSTANALYSIS_ENABLED = x
```

The parameter **DEBUG_ENABLED** responds to the output debug log when you manually run the pre-calculation actions. By default, it is set to 1, which means the debug log (log information which starts with “[DEBUG]”) will be generated. Change the value to 0 and save the record to get cleaner log.

The parameter **INSTANALYSIS_ENABLED** responds to store installation analysis details. By default, it is set to 1, which means for each physical computer’s consumption detail record, a set of installation analysis records will be saved. They represent the installations which were involved in pre-calculation on that physical server. If you change the value to 0 and save the record, the installation analysis record will not be saved and the performance will be improved for pre-calculation process.

Phase 3. Counter Calculation

- Two ways to trigger the phase:
 - Use the **Software counters batch calculation** workflow (**sysSAM4CALCULATE_WK**) to batch calculate software counter. You can set the scheduler to meet the requirement.
 - Manually calculate the software counter by clicking the **Calculate** button in the software counter screen (Windows client) or choosing the **Recalculate the rights and the installations for the selected counters** action in the software counter screen (web client).
- Make sure that counters have linked the license models and installation models.
- Counter will aggregate compliance results from temporary results.

Phase 4. Presenting

- Add the SLO widget **Software licensing compliance report** to the home page to show compliance summary.

- Business homepage of SLO Best Practice package shows generic statistics.
- Dashboards show supplier related statistics.
- Some out-of-box Crystal Reports.

Technical details of software counters (Per supplier)

This chapter explains how the software counters are implemented for different software suppliers.

Microsoft

This section describes the technical details of Microsoft software counters.

Microsoft software counter overview

Microsoft licensing rules are described on the Microsoft Product Use Rights (PUR) website <http://www.microsoft.com/licensing/about-licensing/product-licensing.aspx>.

SLO Best Practice package provides counters for multiple Microsoft software products (see the *Microsoft software titles in scope* section in the *Supported Software Titles* manual).

The counters of the following products use the **Advanced process**.

- Office
- Visio
- Project
- Windows Desktop OS

The counters of the following products use the **Advanced compliance analysis process**.

- Windows Server
- SQL Server

Other counters use the **Simple process**.

The following sections mainly discuss the counters with the **Advanced process** and the **Advanced compliance analysis process**.

Terminologies used in algorithm

Standalone physical computer

A standalone physical computer should meet the following criteria:

The computer is not an **Active** component of an **In Use** infrastructure cluster. (ComputerId NOT IN (SELECT IComputerComponentId FROM amClusterComponent WHERE bActive = 1 AND Cluster.PortfolioItem.seAssignment=0 AND Cluster.seClusterLayer = 2).

Individual OSE

An operating system environment that has one or more Microsoft software installations. An individual OSE can be hosted on a physical computer or a virtual machine.

Using Microsoft Windows Server software counters

This section describes the pre-calculation workflow and algorithm of Microsoft Windows Server software counters.

You can leverage the uCMDB push adapter to push the cluster information from uCMDB to Asset Manager. You need to define your own TQL and map them. For more information about uCMDB push adapter configuration, see **HP Universal CMDB Discovery and Integration Content Guide > Integrations > HP Asset Manager Push Integration**.

Pre-condition: Involved SLO business data during calculation

License type

The following two Windows Server license matrices are mapped to two out-of-box license types in the SLO Best Practice package.

- **Server License** - The code of license type is LICTYPE_MS_SERVER
- **Processor License** - The code of license type is LICTYPE_MS_PROCESSOR

Workflow for pre-calculation

Asset Manager SLO Best Practice package provides a workflow to pre-calculate the consumed license points of Microsoft products automatically. And pre-calculation for Windows Server license points is one action of this workflow.

Workflow

Reference	Name of workflow	Responsibility
sysSAM4MS_MS_ACT	Microsoft pre-Count Activity	Workflow to pre-calculate consumed license points of Microsoft products.

Action

Reference	Context table	Responsibility
sysSAM4MS_WIN_ENT_CNT_Action	amSoftInstall	Core action to calculate the consumed license points of Windows Server.

Windows Server license rules

The following table lists the license rules of Windows Server software products.

Rule type	License matrices	Rule description	Product name
Individual calculation	Server license (License OSE)	One OSE consumes one server license	Windows Server 2003 Std
			Windows Server 2003 Ent
			Windows Server 2003 Web
			Windows Server 2003 R2 Std
			Windows Server 2003 R2 Ent
			Windows Server 2003 R2 Web
			Windows Server 2008 Web
			Windows Server 2008 R2 Web
	Processor license	License needed for each physical processor	Windows Server 2003 DC
		You can have 1 physical OSE and 2 virtual OSEs with each license.	Windows Server 2012 Std
			Windows Server 2012 R2 Std

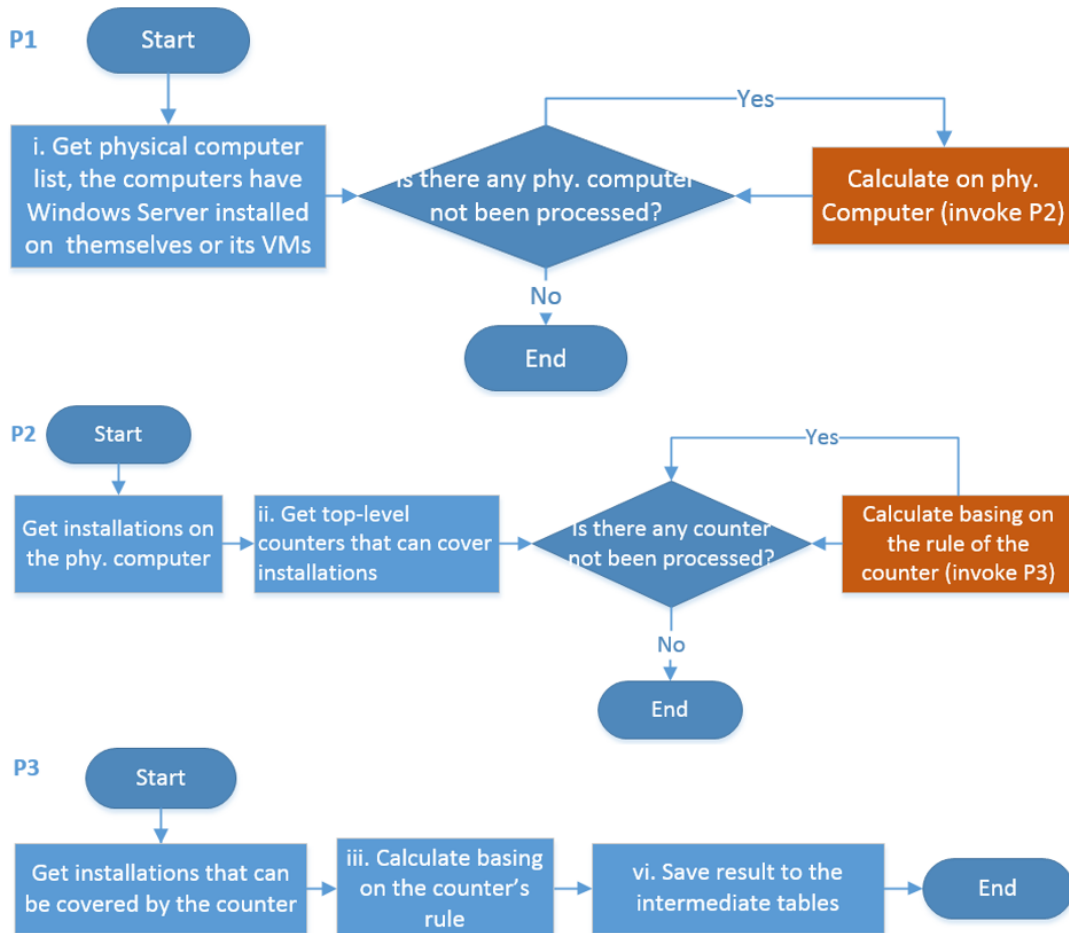
Rule type	License matrices	Rule description	Product name
Support additional virtual installation (1:1)	Server license (License OSE)	You can have 1 physical OSE and 1 virtual OSEs with each license.	Windows Server 2008 Std
			Windows Server 2008 R2 Std
Support additional virtual installation (1:4)	Server license (License OSE)	You can have 1 physical OSE and 4 virtual OSEs with each license.	Windows Server 2008 Ent
			Windows Server 2008 R2 Ent
Support additional virtual installation (1:unlimited)	Processor license	Licensed needed for each physical processor. The license allows unlimited virtual OSEs.	Windows Server 2003 R2 DC
			Windows Server 2008 DC
			Windows Server 2008 R2 DC
			Windows Server 2012 DC
			Windows Server 2012 R2 DC

Algorithm of Windows Server pre-calculation

The algorithm will pre-calculate the consumed license points of Microsoft Windows Server, it handles two scenarios of installation, infrastructure cluster and non-infrastructure cluster (including standalone and other types of clusters). Depending on the amount of virtual machines on physical servers, Asset Manager SLO Best Practice package uses different rules to calculate the license points. The core process is similar, the algorithm will calculate the license points of the installations with the highest version or edition, and these licenses can also be used to cover the installations of lower version or edition.

In detail, the algorithm:

1. Clear the intermediate compliance result in **amSAMNodeCompAnalysis**, **amSAMClusterCompAnalysis**, **amSAMIssueAnalysis**, and **amSAMInstCompAnalysis**.
2. Divide all Windows Server installations into two parts.
3. Calculate the license points for the installations that are not in an infrastructure cluster.



4. Find all the physical computers that have Windows Server installed on itself or on its VM(s). These computers are marked as **In Use** (PortfolioItem.seAssignment=0)) and are not an **Active** component of an **In Use** infrastructure clusters. (IComputerId NOT IN (SELECT IComputerComponentId FROM amClusterComponent WHERE bActive = 1 AND Cluster.PortfolioItem.seAssignment=0 AND Cluster. seClusterLayer = 2))
5. Calculate the license consumption for all these computers one by one.
6. Find the highest version/edition license model of the Windows Server installations on this physical server based on the information in the **amLicDowngrade** table, and then use its license rule to do calculation.

7. If more than one highest version/edition license model are found, all the highest license models will be taken into account on this physical computer. For example, Windows Server 2012 Standard license and Windows Server 2008 Datacenter license cannot cover lower versions or editions. In fact, this is not a normal case since Microsoft does not suggest such deployment.
8. Calculate the consumed license points for all the installations of highest version/edition. See the license rule table for details.
9. If the processor number of a physical computer is 0, a record will be created in **amSAMIssueAnalysis**.

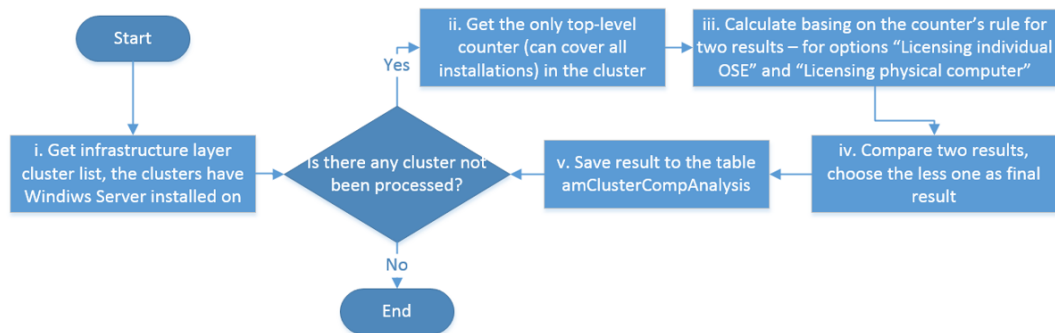
The **amLicPointItem** table shows the key parameters for license rules.

Field name	Description
ILicCounterId	Foreign key of the link a license counter.
IVirtual	Support additional virtual installation or not 1 means YES, 0 means NO.
IVMPoints	The number of additional virtual installations. For example: 4 means one license supports up to 4 free virtual installations.
IProcessorRatio	The number of processors that one license can support. For example, 2 mean one license can support up to 2 processors.
SpecialRule	Only works for Windows Server 2012 STD and Windows Server 2012 R2 STD because the license rule is more complex than the others.

10. Generate one intermediate compliance record for each physical computer in **amSAMNodeCompAnalysis** with the **bCountable** set to 0.
11. Generate intermediate compliance result in **amSAMInstCompAnalysis**. The record number equals to the number of counted Windows server installations on this physical computer.
12. Calculate the license points for the installations that are in an infrastructure cluster.

Note: If different versions/editions of Windows Server installations exist in the cluster, only the license for the higher version/edition is taken into account. For example, you have both Windows Server 2008 STD and Windows Server 2008 R2 ENT installed in the cluster, Asset Manager calculates the license points by assuming that both the installations are entitled by the Windows Server 2008 R2 ENT license.

In a cluster environment, Asset Manager does the calculation using the server license/processor licenses over the server licenses.



13. Find all the **Active** cluster components of the **In Use** infrastructure clusters whose components have Windows Server installed on itself or on its VM(s). (bActive = 1 AND Cluster.seClusterLayer=2 AND Cluster.Portfolio.seAssignment=0)
14. Find the highest version/edition license model for the cluster. One cluster can only use one single type of license. Therefore, in a cluster, there must be one license type that can be used to cover all lower versions/editions of Windows Server. For instance, both Windows Server 2008 DC and Windows Server 2012 STD are installed in one cluster. The license of Windows Server 2012 DC will be used to entitle the while cluster even for the physical/virtual machines that do not have Windows Server 2012 DC installed. This is because the license of Windows Server 2012 DC is the lowest version/edition that can be used to cover both Windows Server 2008 DC and Windows Server 2012 STD.
15. Calculate the license points for the **License individual operating system environment** and **License physical computer** options respectively. Generate intermediate compliance result in **amSAMClusterCompAnalysis**. Two records will be created for each cluster. One is for **Licensing individual OSE**, the other is for **Licensing physical host**.

Note: When processing the **License individual operating system** option, for each VM containing Windows Server instances, Asset Manager calculates the license consumption of the current VM according to its software/hardware information. Then, Asset Manager multiplies the license consumption by the number of the physical servers that have hosted this VM in the past 90 days. For more information about how to obtain the number of the physical servers that have hosted this VM, see ["Additional information 1. Acquiring the number of VM hosts \(Microsoft\)" on page 48](#).

When processing the **License physical host** option, for each physical server containing Windows Server instances, Asset Manager calculates the license consumption of the current physical server according to its software/hardware information.

As for the **License individual operating system** option:

- a. Consider all OSEs in the cluster as individual physical computers, and calculate license consumption with these OSEs' information. The calculation process is the

same as the non-infrastructure calculation.

- b. The license consumption of a cluster equals to the sum of that of all OSEs.

As for the **License physical host** option:

- a. The calculation process is the same as the non-infrastructure calculation.
- b. The license consumption of a cluster equals to the sum of that of all physical computers.

16. Compare the two results, and use the smaller value if neither of them equals to 0. If one of the values is 0, use the other value. Mark **bCountable** to 1 for the final result in **amSAMClusterCompAnalysis**.
17. Update the final result in amClusterCompAnalysis. License consumption points will not be saved on software installation records. The summary compliance data comes from the intermediate tables. So, the intermediate records have to be cleared before calculation.

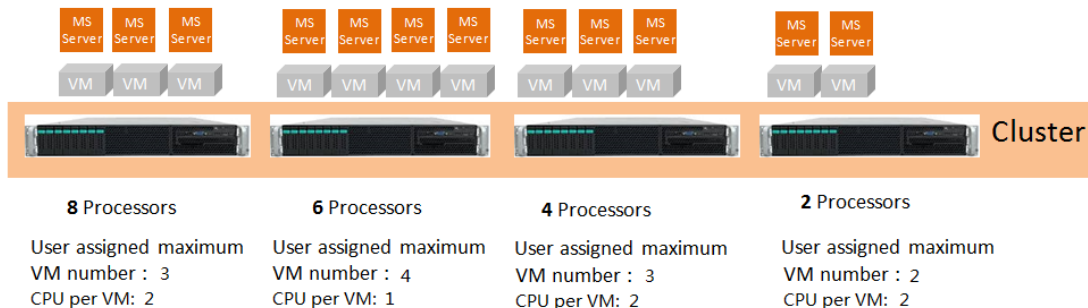
Limitations

- The Windows server pre-calculation script can only count up to 1 million physical computers with Windows Server installations.
- The calculation result comes from a screenshot of your Windows Server deployment.
- Windows Server upgrade license are not supported in this version of the SLO package.
- According to the license rule, in a virtualization environment, Windows Server instance on the physical server should only be used to manage the virtual machines on this physical server, if not, additional license should be assigned to this physical server. However, Asset Manager SLO Best Practice package cannot determine if the instance is just used as a manage server, so in these cases, Asset Manager SLO Best Practice package considers all Windows servers on the physical server are only used as a management server, no additional licenses need to be assigned to these physical servers. For more information, see http://www.highergroundtech.com/downloads/WS2012_Licensing-Pricing_FAQ.pdf.
- Asset Manager SLO Best Practice package does not support mobility cases because it does not have the Software Assurance (SA) information and frequency of license transfer in a farm, so every physical server will be assigned the license as long as there is Windows Server instance running on it or its virtual machine.
- Asset Manager SLO Best Practice package cannot verify if the customer has SA covering Windows Server products. By default, it is assumed that all Windows Server products are not covered by SA.

Cluster examples

The Asset Manager SLO Best Practice package now supports Windows Server installations in a cluster.

Scenario 1. Infrastructure cluster



As shown in the figure above, a vSphere cluster includes 4 servers with different virtual machine number and CPU number. Each virtual machine runs a Windows Server 2012 Standard instance.

According to the license rule of Windows Server 2012 Standard edition, Asset Manager SLO Best Practice package compares the license number based on the virtual number and CPU number, then takes the higher value.

Therefore, the total licenses required for Windows Server 2012 Standard edition is:

Server 1 :

Consumption based on individual VM:

1 Processor(1 license) * 3 = 3 licenses

Consumption based on physical host:

4 VMs (2 licenses) < 8 Processors (4 licenses) => 4 licenses

Therefore, server1 needs 3 licenses.

Server 2 :

Consumption based on individual VM:

1 Processor (1 license) * 4 = 4 licenses

Consumption based on physical host:

4 VMs (2 licenses) < 6 Processors (3 licenses) => 3 licenses

Therefore, server2 needs 3 licenses.

Server 3 :

Consumption based on individual VM:

2 Processors(1 license) * 3 = 3 licenses

Consumption based on physical host:

3 VMs (2 licenses) = 4 Processors (2 licenses) => 2 licenses

Therefore, server3 needs 2 licenses.

Server 4 :

1. Consumption based on individual VM

2 Processors(1 license) * 3 = 3 licenses

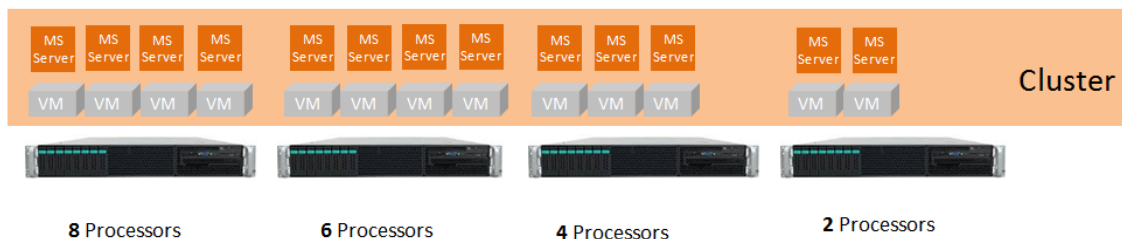
2. Consumption based on physical host

2 VMs (1 license) = 2 Processors (1 license) => 2 licenses

Therefore, server4 needs 2 licenses.

Total license number : 3 + 3 + 2 + 2 = 10

Scenario 2. OS cluster



As shown in the figure above, the Windows Server cluster includes 4 servers with different virtual machine number and CPU number. Each virtual machine runs a Windows Server 2012 Standard instance.

According to the license rule of Windows Server 2012 Standard edition, Asset Manager SLO Best Practice package compares the license number based on the virtual number and CPU number, then takes the higher value.

Therefore, the total licenses required for Windows Server 2012 Standard edition is:

Server 1 : 4VMs (2 license) < 8Processors (4 license) => 4 license

Server 2 : 4VMs (2 license) < 6Processors (3 license) => 3 license

Server 3 : 3VMs (2 license) = 4Processors (2 license) => 2 license

Server 4 : 2VMs (1 license) = 2Processors (1 license) => 1 license

Total license number : 4 + 3 + 2 + 1 = 10

Using Microsoft SQL Server software counters

This section describes the pre-calculation workflow and algorithm of Microsoft SQL Server software counters.

For the “Microsoft SQL Server 2014 Standard (Per Core)”, “Microsoft SQL Server 2014 Enterprise (Per Core)”, “Microsoft SQL Server 2012 Standard (Per Core)”, “Microsoft SQL Server 2012 Enterprise (Per Core)”, and “Microsoft SQL Server 2012 Web” software counters, licenses are assigned in pairs (because one SKU contains two per core licenses). Therefore, consumed license points must be a multiple of two. For each SQL server license asset purchased on the Per Core license model, the number of rights must be two.

Pre-condition: Involved SLO business data during calculation

License type

The following three SQL Server license matrices are mapped to three out-of-the-box license types in Asset Manager SLO Best Practice package.

- **Server License** - The code of license type is **LICTYPE_MS_SERVER**.
- **Processor License** - The code of license type is **LICTYPE_MS_PROCESSOR**.
- **Core License** - The code of license type is **LICTYPE_PER_CORE**.

Workflow for core factor population

The out-of-box workflow **Calculate Microsoft core factor periodically** (SQL name: **sysSAM4MS_CF_WK**) is responsible for populating the Microsoft core factor of the physical computers that have Microsoft software installed on. By default, it is scheduled every quarter.

Once the workflow is triggered, the action **sysSAM4MS_CF_Action** is invoked to update the **amCoreFactorResult** table (see the *The amCoreFactorResult table* section in ["Appendix 1. Data model and definition" on page 82](#)) regularly. The algorithm is as follows:

1. Scan the **amCoreFactorResult** table, delete any record whose **bConfirm** field is 0 and the link points to the license type record whose value of the **Code** field is **LICTYPE_MS_PER_CORE**.
2. Search for any physical computer which matches the following criteria:
 - There is Microsoft software product installed on the physical computer directly or on the virtual machine running on the physical computer.

Note: It is assumed that all Microsoft software has the same characteristics. The

company of **SoftInstall** model is Microsoft, the conditional expression is:

```
SoftInstall.Model.Brand.Company.Code='DEMO-S27'.
```

- The installation can be filtered out by installation query of the software counter whose **LicType** link points to the OOB license type **Core License** (the value of the **Code** field is **LICTYPE_MS_PER_CORE**).
 - There is not any core factor result exists with:
 - The core factor result links to the physical computer.
 - The core factor result links to the OOB license type **Core License** (the value of the **Code** field is **LICTYPE_MS_PER_CORE**).
 - The core factor result has value 1 for the **bConfirmed** field.
3. For the selected physical computers, match them with the core factor table, save the corresponding core factor in the **amCoreFactorResult** table.
 4. For all physical computers that do not have the core factor value assigned because of missing/incorrect hardware information, set the core factor value to the default value (1), and then set the value of **bEstimate** to 1.

With this workflow, you can check the core factor value of the physical computer before or after counter calculation so that you can get to know what value is used for calculation.

Workflow for pre-calculation

Asset Manager SLO Best Practice package provides a workflow to pre-calculate the consumed license points of Microsoft products automatically. And pre-calculation for SQL Server license points is one action of this workflow.

Workflow

Reference	Name of workflow	Responsibility
sysSAM4MS_MS_ACT	Microsoft pre-Count Activity	Workflow to pre-calculate consumed license points of Microsoft products.

Action

Reference	Context table	Responsibility
sysSAM4MS_SQL_CNT_ Action	amSoftInstall	Core action to calculate the SQL Server consumed license points.

Core factor table

See "Additional information 2. SQL Server core factor table" on page 49. Also see http://download.microsoft.com/download/4/4/5/445627B4-9AB0-4AED-BCCD-C7AC5ADAF6B2/CoreFactorTable_4_1_2014.pdf.

SQL Server license rules

The brief SQL Server license rules are as follows.

Rule type	License matrices	Rule description	Product name
Individual calculation	Server license (License OSE)	One OSE consumes one server license	SQL Server 2005 STD
			SQL Server 2005 WKG
			SQL Server 2008 STD
			SQL Server 2008 WKG
			SQL Server 2008 R2 STD
			SQL Server 2008 R2 WEB
			SQL Server 2012 STD
			SQL Server 2012 BI
			SQL Server 2014 BI

Rule type	License matrices	Rule description	Product name
	Processor license	One CPU consumes one processor license	SQL Server 2005 STD
			SQL Server 2005 WKG
			SQL Server 2008 STD
			SQL Server 2008 WKG
			SQL Server 2008 WEB
			SQL Server 2008 R2 STD
			SQL Server 2008 R2 WKG

Rule type	License matrices	Rule description	Product name
Support additional virtual installation (1:unlimited)	Core license	License all cores	SQL Server 2012 STD
			SQL Server 2014 STD
			SQL Server 2012 WEB
	Server license (License OSE)	Only license physical OSE	SQL Server 2005 ENT
			SQL Server 2008 ENT
	Processor license	Only license physical processor	SQL Server 2005 ENT
		<ul style="list-style-type: none"> License all physical processors if the physical OSE has SQL Server installations. It covers unlimited number of installations on virtual machines. 	SQL Server 2008 ENT
		<ul style="list-style-type: none"> License all virtual processors if the physical OSE has no installation, but need to compare with the consumption on the physical processor and use the smaller result. 	SQL Server 2008 R2 DC

Rule type	License matrices	Rule description	Product name
Support additional virtual installation (1:4)	Server license (License OSE)	One server license can entitle 4 installations at most, for instance, 1 physical installation and 3 virtual installations, or 4 virtual installations.	SQL Server 2008 R2 ENT
			SQL Server 2012 ENT
	Processor license	<ul style="list-style-type: none"> License all physical processors if the physical OSE has SQL Server installations License all virtual processors if physical OSE has no installation, but need to compare with the consumption on physical processor and use the smaller result. One processor license can entitle up to 4 OSEs (physical or virtual). 	SQL Server 2008 R2 ENT
			SQL Server 2012 ENT
	Core license	<ul style="list-style-type: none"> All physical cores on the server must be licensed if physical server has SQL Server installations. All virtual cores (v-cores) supporting virtual OSEs that are running instances of SQL Server 2012 software must be licensed. When licensing individual OSE, core factors do not apply. There is a minimum of 4 core licenses required for each virtual machine. With all physical cores on the server licensed, you can run an unlimited number of instances of the software in a number of OSEs (physical and/or virtual) equal to the number of core licenses assigned to the server. Additional VMs need additional Core licenses. 1 additional core license allows 1 additional VM. 	SQL Server 2014 ENT

Algorithm of SQL Server pre-calculation

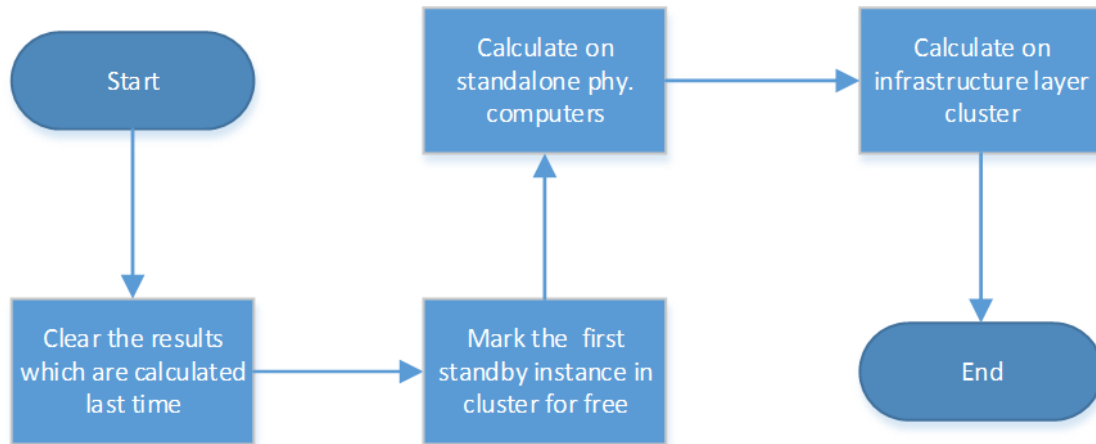
The algorithm will pre-calculate the consumed license points of Microsoft SQL Server, it handles two scenarios of installation, infrastructure cluster and non-infrastructure cluster.

SQL Server instance can be entitled by different license types, to assign a license type for installation, open the installation screen, choose the tab **Software inst**, and then assign value for

the field **License Metric** (SQLName is **LicenseMetric**). If you do not set the license type, Asset Manager SLO Best Practice package will use the default setting to do the calculation. You can change the default setting in the script library **Sam4MSSqlServerLicLib**, constant field **DEFAULT_LIC_TYPE**.

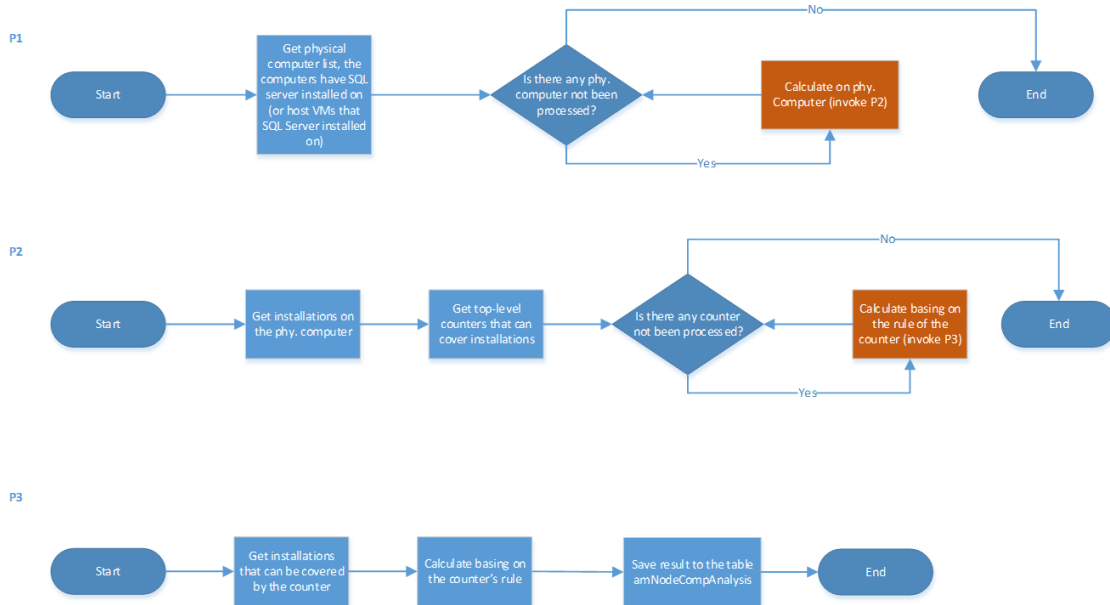
General process

In general, the algorithm is as follows.



1. Delete the compliance analysis results which were populated in the last time pre-calculation was running.
2. Find the first standby instance in each SQL Server cluster (application layer cluster), put them into the **amSoftInstExclTmp** table. Those instances will be excluded for licensing.
3. Categorize all physical computers into 2 types (standalone physical computers and physical computers in infrastructure cluster), and then calculate consumptions. For details, see ["Calculation process on standalone physical computers" on the next page](#) and ["Calculation process in infrastructure layer cluster" on page 40](#).

Calculation process on standalone physical computers

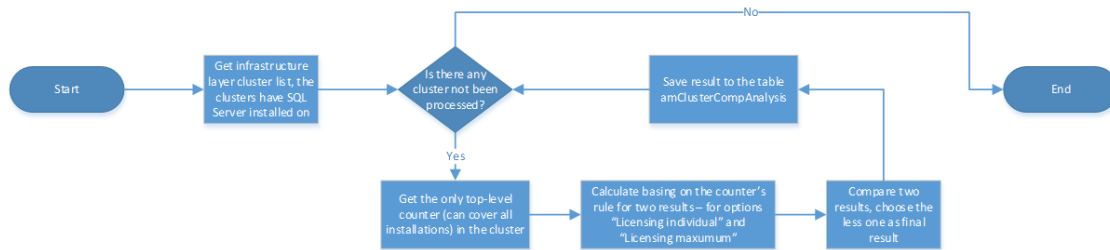


First, Asset Manager gets the standalone physical computer list. Those computers must have SQL Server product installed on the computer (or installed on the VMs that are hosted by the computer).

Then, Asset Manager scans through the computer list, for each physical computer on the list, Asset Manager does the following calculation.

1. Find all installations on the physical computer according to the counter's installation query, then, find the most advanced software editions(those multiple editions cannot cover each other) based on the data in the **amLicDowngrade** table (this data is initialized during the SLO package installation and cannot be customized yet). By identifying the most advanced editions, Asset Manager can figure out the licensing rules to be used for calculation on the computer.
2. For each license rule, find the installations which can be covered by the license rule, calculate their consumption points. Then, save the result to the **amNodeCompAnalysis** table, with the flag **bCountable** set to 1.

Calculation process in infrastructure layer cluster



First, Asset Manager gets the infrastructure layer cluster list. Those clusters should meet the following criteria.

- The cluster is infrastructure layer (amCluster.seClusterLayer=2), and status is **In Use** (amCluster.Portfolio.seAssignment=0).
- The cluster has concerned SQL Server product installed on its computer component directly (or installed on the VMs hosted by the computer).
- The computer acts as an **Active** component of the cluster (amCluster.ClusterComponents.bActivated=1).

Then, Asset Manager scans through the cluster list, for each cluster on the list, Asset Manager does the following calculation.

1. Find the **ONLY** top-level counter whose corresponding license rule can cover all installations in the cluster.

Note: If different versions/editions of SQL Server installations exist in the cluster, only the license for the higher version/edition is taken into account. For example, you have both SQL Server 2008 STD and SQL Server 2008 R2 ENT installed in the cluster, Asset Manager calculates the license points by assuming that both the installations are entitled by the SQL Server 2008 R2 ENT license. And Asset manager requires all installations in the cluster can be covered by the most advanced version/edition.

In a cluster environment, Asset Manager does the calculation using the core license/processor licenses over the server licenses.

2. Calculate the license points for the **License individual OSE** option and **License for maximum virtualization** option respectively.

Note: When considering for the "license individual OSE" option, for each VM containing SQL Server instances, Asset Manager calculates the license consumption of the current VM according to its software/hardware information. Then, Asset Manager multiplies the

license consumption by the number of the physical servers that have hosted this VM in the past 90 days. For more information about how to obtain the number of the physical servers that have hosted this VM, see [Appendix 1. Acquiring the number of VM hosts \(Microsoft\)](#).

Asset Manager assumes that all physical servers in the cluster have the same hardware information.

3. Compare the results in step 2, use the smaller value if neither of them equals to 0. If one of the values is 0, use the other value.
4. Save the consumption results in the table **amSAMClusterCompAnalysis**, and set the flag **bCountable** to 1. Meanwhile, for each physical computer in the cluster, the corresponding consumption result of computer is saved in the table **amSAMNodeCompAnalysis**, with the flag **bCountable** set to 0.

Limitations

- Asset Manager SLO Best Practice package does not support hyper thread information. It is assumed that all servers (physical/virtual machine) with SQL Server instances do not use hyper thread.
- Asset Manager SLO Best Practice package cannot identify if a SQL Server instance is 64-bit application or 32-bit application. It is assumed that all SQL Server instances are 32-bit application. The special installation rules for 64-bit SQL Server are not supported now.
- Asset Manager SLO Best Practice package uses one license type to cover multiple SQL Server instances running on one server (physical/virtual machine). The license type depends on the default setting or customized setting.

For example, if one server contains two SQL Server instances (one SQL Server 2005 Standard edition and one SQL Server 2008 Enterprise edition), and you link the license type to the server as “by server”. All instances will have the “by server” license type.

If a SQL Server instance has only one license type per Microsoft's license rule, only that license type is taken into account even you assign another license type to the server.

For example, if one server contains SQL Server 2012 Business Intelligence, even you assign “by CPU core” license type to the server, it will always have the “by server” license type because SQL Server 2012 Business Intelligence only has the “by server” license type per Microsoft's license rule.

- It is assumed that all SQL Server instances in an infrastructure cluster should be the same version, same edition, and same license type (linked license type to the server of the instance).
- “AlwaysOn Availability Groups” introduced in SQL Server 2012 Enterprise is not supported currently.

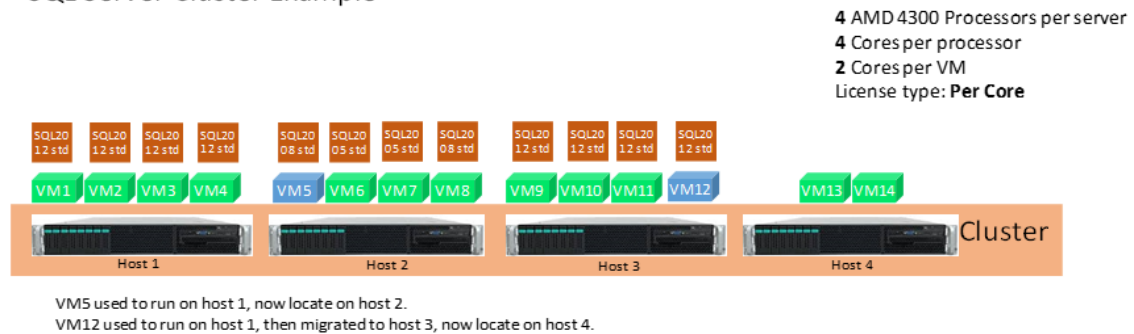
- When licensing with “Core license” in an infrastructure cluster, SLO package always assumes that there is an active SA that can be applied.

Cluster examples

The Asset Manager SLO Best Practice package now supports SQL Server installations in a cluster.

Infrastructure cluster

SQL Server Cluster Example



Scenario 1: As shown in the figure above, a vSphere cluster includes 4 servers with different virtual machine number. Different versions of SQL Server Standard instances are installed on the virtual machines.

Asset Manager uses SQL Server 2012 Standard license for license calculation, it can be used to entitle the less advanced versions.

For the "license individual virtual machine" option:

- VM5: points = $2(\text{core number}) * 2(\text{touched 2 hosts}) = 4$
- VM12: points = $2(\text{core number}) * 3(\text{touched 3 hosts}) = 6$
- Other VMs: points = $2(\text{core number})$, 10 VMs in total with 20 points
- For the entire cluster, points = $4 + 6 + 20 = 30$; purchased pack = $30 / 2 = 15$

For the "license for maximum virtualization" option:

- Host 1, Host 2 and Host 3: points = $4 * 4 = 16$, in total, $16 * 3 = 48$
- Host 4: no SQL Server instance, points = 0
- For the entire cluster, points = $48 + 0 = 48$; purchased pack = $48 / 2 = 24$

In summary, Asset Manager chooses the smaller point number, that is 30 points for the cluster and 15 core pack.

Application cluster

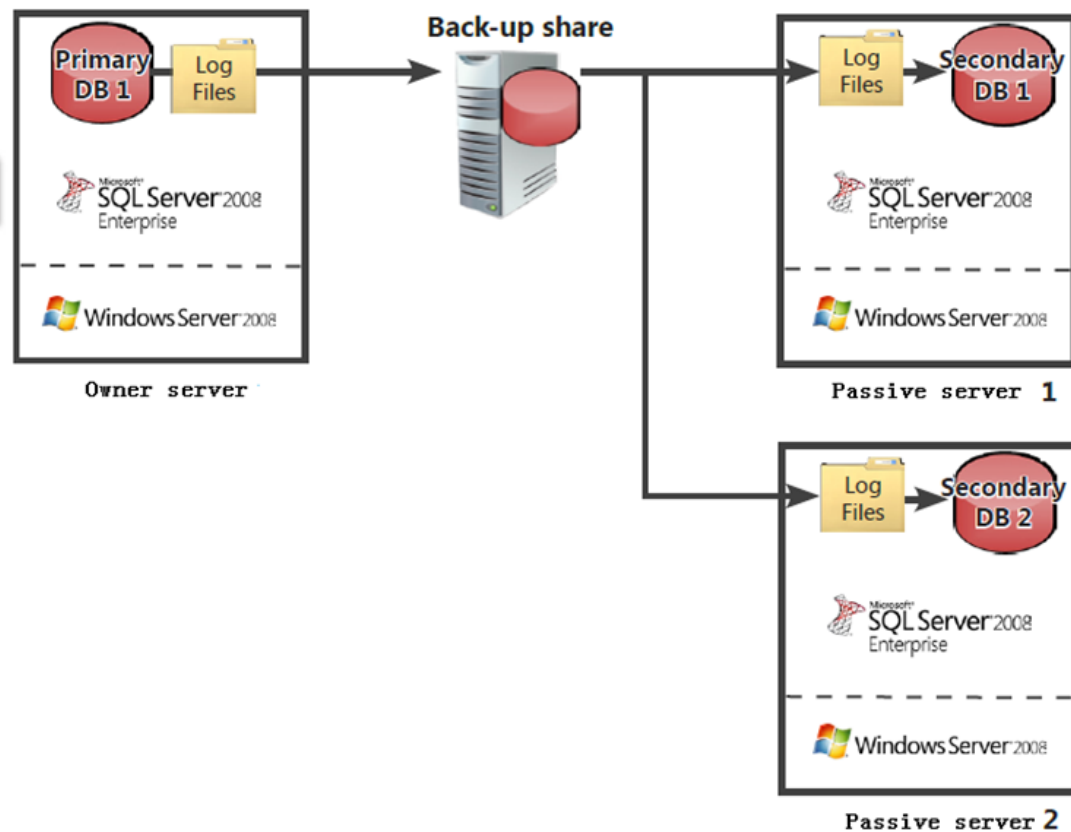
There are four types of application layer cluster to ensure high availability for SQL Server:

- Backup log shipping
- Database mirroring
- Failover clustering
- Always On (introduced in SQL Server 2012)

For each properly licensed instance of SQL Server, you can run a supporting standby instance in a separate operating system environment (OSE) for temporary support. A standby SQL Server instance is one that is not serving SQL Server data to clients, running active SQL Server workloads, or spreading the serving power of SQL server, this standby failover instance can run on a server other than the licensed server.

In the current application cluster for SQL Server, one cluster can contain only one “owner” role component. If a cluster contains more one owner components according to the real scenario, we will split it into some sub-clusters.

Only the first passive SQL Server instance is free for SQL Server application cluster. The serial number is sorted in the **StandbyOrder** field of the **amClusterComponent** table.

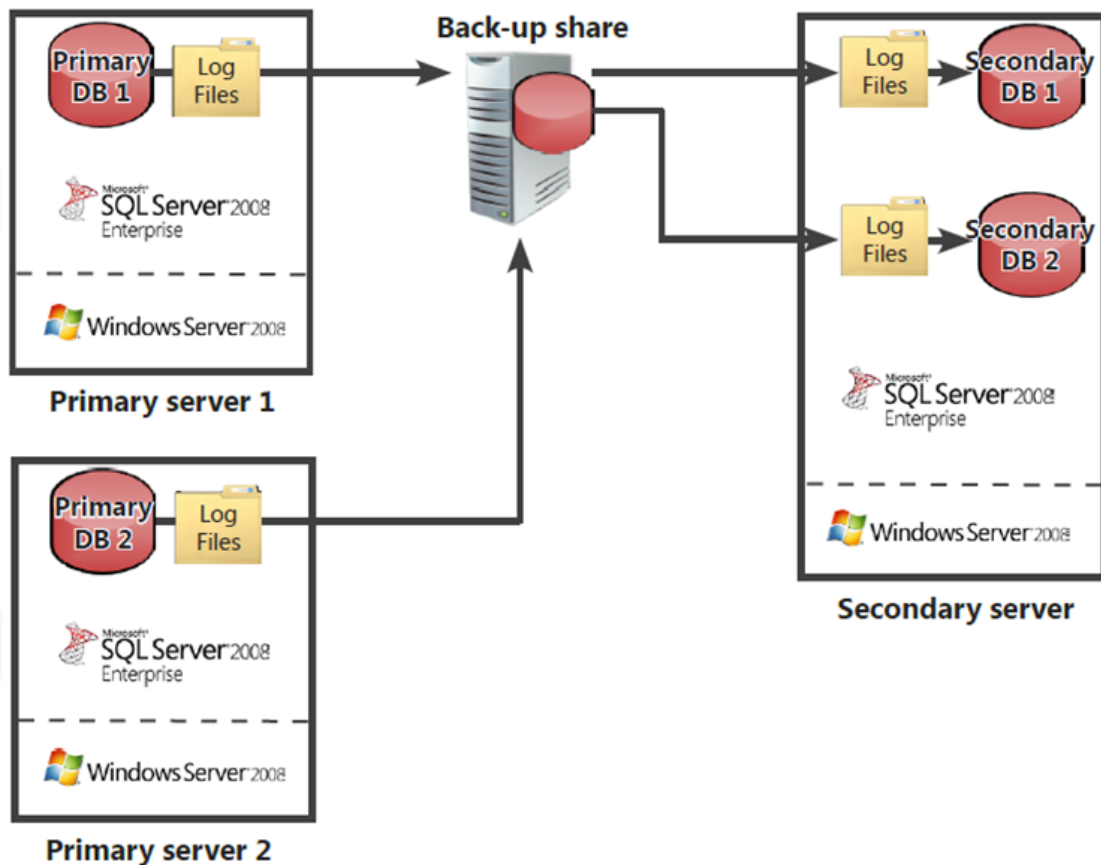


Scenario 2: As shown in the figure above, an application cluster includes 3 installation components: Primary DB1, Secondary DB1, and Secondary DB2.

Primary DB1 takes owner role; Secondary DB1 takes standby role; Secondary DB2 takes standby role and StandbyOrder is bigger than Secondary DB1.

According to the license rule of SQL Server 2008 Enterprise edition, each server license can support unlimited instances in unlimited OSEs (VM / physical) on a server.

Therefore, Secondary DB1 is free according to current scenario as it is the first standby instances in an application cluster. Primary DB1 and Secondary DB2 need licenses according to real scenario.



Scenario 3: As shown in the figure above, an application cluster includes 4 installation components: Primary DB1, Primary DB2, Secondary DB1, and Secondary DB2.

Primary DB1 takes owner role; Primary DB2 takes owner role; Secondary DB1 takes standby role; Secondary DB2 takes standby role.

Because one application cluster can contain only one owner component, we must split the current cluster into two sub-clusters:

- Sub-cluster 1 contains two components: Primary DB1 and Secondary DB1
- Primary DB1 takes owner role
- Secondary DB1 takes standby role
- Sub-cluster 2 contains two components: Primary DB2 and Secondary DB2
- Primary DB2 takes owner role
- Secondary DB2 takes standby role

According to the license rule of SQL Server 2008 Enterprise edition, each server license can support unlimited instances in unlimited OSEs (VM / physical) on a server.

Therefore, Secondary DB1 is free according to current scenario as its first standby instances in sub-cluster 1. Secondary DB1 is free according to current scenario as its first standby instances in sub-cluster 2. Primary DB1 and Primary DB2 need licenses according to real scenario.

Using Microsoft Office/Visio/Project software counters

This section describes the pre-calculation workflow and algorithm of Microsoft Office/Visio/Project software counters.

Pre-condition: Involved SLO business data during calculation

License type

The following license matrix is mapped to the out-of-box license type in Asset Manager SLO Best Practice package. For more details, see the Microsoft Product Use Rights (PUR) website <http://www.microsoft.com/licensing/about-licensing/product-licensing.aspx>.

- **Per Device** - The code of license type is **LICTYPE_MS_PER_DEVICE**.

Workflow for pre-calculation

Asset Manager SLO Best Practice package provides a workflow to pre-calculate the consumed license points of Microsoft products automatically. And pre-calculation for Office/Visio/Project license points is one action of this workflow.

Workflow

Reference	Name of workflow	Responsibility
sysSAM4MS_MS_ACT	Microsoft pre-Count Activity	Workflow to pre-calculate consumed license points of Microsoft products.

Action

Reference	Table that contains installation data	Responsibility
sysSAM4MS_MOB_DEV_CNT_Action	amSoftInstall	Core action to calculate Office/Visio/Project consumed license points.

License rules

Microsoft desktop applications are licensed per workstation, with the possibility to install a second copy on a portable device (such as laptop) for the user.

Using Microsoft desktop OS software counters

This section describes the pre-calculation workflow and algorithm of Microsoft desktop OS software counters.

Pre-condition: Involved SLO business data during calculation

License type

The following license matrix is mapped to the out-of-box license type in Asset Manager SLO Best Practice package.

- **Per Copy Per Device** - The code of license type is **LICTYPE_MS_DESK_OPERA_SYS**.

Workflow for pre-calculation

Asset Manager SLO Best Practice package provides a workflow to pre-calculate the consumed license points of Microsoft products automatically. And pre-calculation for Windows Desktop OS license points is one action of this workflow.

Workflow

Reference	Name of workflow	Responsibility
sysSAM4MS_MS_ACT	Microsoft pre-Count Activity	Workflow to pre-calculate consumed license points of Microsoft products.

Action

Reference	Table that contains installation data	Responsibility
sysSAM4MS_DSKOS_CNT_Action	amSoftInstall	Core action to calculate the Windows Desktop OS consumed license points.

License rules

For licensing details, see the Microsoft Product Use Rights (PUR) website
<http://www.microsoft.com/licensing/about-licensing/product-licensing.aspx>.

Additional information 1. Acquiring the number of VM hosts (Microsoft)

In an infrastructure cluster, virtual machines can be migrated from one host (physical server) to another. Based on Microsoft license rules, if a virtual machine is migrated between two physical servers within 90 days, customers must assign licenses to both physical servers that have hosted the virtual machine. Therefore, Asset Manager needs to acquire the number of these physical servers.

Storing the VM migration log in Asset Manager

In Asset Manager with SLO Best Practice package, the VM migration log is stored in the **amHistory** table. An out-of-the-box scenario (scenario\MAP\am95vCenter\vCenter_Evt_log.scn) is provided to analyze the VM migration event from vCenter event log and then import the migration data to Asset Manager.

The record in the **amHistory** table looks like:

Field	Value
memPreviousCmt	Always be "The info is extracted from vCenter event log."
Field	Always be "Parent.Computer.TcpipHostName"
HistObjTable	Always be "amPortfolio"
IHistObjId	The portfolio item ID of the VM in Asset Manager
dtLastModif	The time when the migration event occurred
PreviousVal	The name of the physical server that hosted the VM before the migration event
NewVal	The name of the physical server that hosts the VM after the migration event

To view the migration log, open the portfolio screen, select the corresponding VM item, and then click the History tab.

Algorithm

Asset Manager uses the following algorithm to acquire the number of the physical servers which have hosted the VM in 90 days.

1. For a given VM, search in the amHistory table with the following conditions.
 - HistObjTable = "amPortfolio"
 - The value of the IHistObjId field is the portfolio item ID of the VM.

- Field = "Parent.Computer.TcpipHostName"
 - The value of the dtLastModif is no earlier than 90 days ago.
2. From the selected records, find out the values of the following fields.
 - PreviousVal
 - NewVal

These values stand for the physical servers that have hosted the virtual machine. You also need to take the current physical server's host name into consideration so that you can get the number of the physical servers that have hosted the virtual machine.

Additional information 2. SQL Server core factor table

Currently, the SLO Best Practice package does not support configurable Microsoft core factor table. If customization is needed, customers may have to update the corresponding script code as follows:

1. Log on to AM with the an administrator user account.
2. On the web client, enter "scripts" in the search box on the top, then open the "scripts" screen. On the Windows client, on the **Administrator** menu, click **Script libraries**.
3. Find and choose the **SAM4MS_Generic** record.
4. In the detail pane, look into the script field, find the **GetCoreModelFactor** function. Update the code according to your requirements and save.

Oracle

This section describes the pre-calculation workflow and algorithm of Oracle software counters.

Oracle software counter overview

Oracle provides various licensing models for different products. Currently SLO Best Practice package only supports the per processor licensing model, which is more cost effective for an organization with many Oracle software users.

The Asset Manager Software License Optimization (SLO) Best Practice package uses 4 types of Oracle software counters:

- Oracle Database
- Oracle Database Option

- Oracle Enterprise Management Pack
- Oracle Weblogic

Oracle DB counter

SLO Best Practice Package supports the following Oracle Database products:

- Oracle Database Enterprise Edition
- Oracle Database Standard Edition
- Oracle Database Standard Edition One
- Oracle Database Standard Edition Two

Oracle DB counter contains two types of counters for installed instance and running instance respectively, and categorized by edition.

The records in the **amSoftlInstall** table represent the installed instance and the records in the **amMonitoredApp** table represent the running instance.

Oracle DB option counter and Enterprise Management Pack counter

The Oracle DB option and Enterprise Management Pack are both extended components of Oracle Database Enterprise edition. The records in the **amMonSWComp** table represent these installations.

SLO package also offers two sets of counters as shown in the figure below. One is for installation-based options and the other is for in-use based options/packs. For more information about how to use these counters, see the User Guide > Using the SLO Best Practice package on top of the Asset Manager Software Asset Management module > Counters for Oracle products.

<input type="checkbox"/>	Name ↕
<input type="checkbox"/>	Oracle DB option: Active Data Guard (Active)
<input type="checkbox"/>	Oracle DB option: Advanced Analytics (Active)
<input type="checkbox"/>	Oracle DB option: Advanced Compression (Active)
<input type="checkbox"/>	Oracle DB option: Advanced Security (Active)
<input type="checkbox"/>	Oracle DB option: Audit Vault (Active)
<input type="checkbox"/>	Oracle DB option: Advanced Analytics (In use)
<input type="checkbox"/>	Oracle DB option: Advanced Compression (In use)
<input type="checkbox"/>	Oracle DB option: Active Data Guard (In use)
<input type="checkbox"/>	Oracle DB option: Advanced Security (In use)
<input type="checkbox"/>	Oracle DB option: Audit Vault (In use)

Why two sets of Oracle database counters?

Normally Oracle audit will consider those installed options/packs and/or used options, so by default, installation-based counters will be included in the compliance report which include all installed Options/Packs, no matter used or not. SLO Best Practice package also offers in-use based counters which include the options/packs that are currently used or were used in 366 days. This helps you know the current utilization of Options/Packs.

Oracle Weblogic

The SLO Best Practice package supports the following three Weblogic editions.

- Standard
- Suit
- Enterprise

Terminologies used in algorithm

Standalone physical computer

A standalone physical computer should meet the following criteria:

- The computer is not an **Active** component of an **In Use** infrastructure cluster. (IComputerId NOT IN (SELECT IComputerComponentId FROM amClusterComponent WHERE bActivated = 1 AND Cluster.PortfolioItem.seAssignment=0 AND Cluster. seClusterLayer = 2).

Using Oracle software counters

This section describes the pre-calculation workflow and algorithm of Oracle software counters.

The counters serve for two integration solutions:

Traditional integration

In this solution, all software deployment data is imported into the **amSoftInstall** table, which stores the products that have been installed in the customer environment. In the HP out-of-the-box content package, usually the software deployment data is discovered/recognized by HP Universal Discovery (UD) and imported by push adapter (or discovered /recognized by HP Discovery and Dependency Mapping Inventory (DDMi) and imported by Connect-It scenario). The counters serving for this solution include:

- Oracle database counters whose name does not contain the postfix “(Running Instance)”
- Weblogic counters

Oracle LMS integration

In this solution, software deployment data is discovered /recognized by HP UD with a special technology (a set of scripts certificated by Oracle) and imported into AM by LMS push adapter. The target tables in AM are **amMonitoredApp** and **amMonSWComp**, which store the products that are running in customer environment. The counters serving for this solution include:

- Oracle database counters whose name has the postfix “(Running Instance)”
- Oracle Database Option counters
- Oracle Enterprise Management Pack counters

Why two sets of Oracle database counters?

Oracle officially uses the License Management Services (LMS) tool for Oracle database/option audit. However, LMS can only collect information from servers that are running Oracle database instances/options. Yet, there might be some installed database instances/options, which are not running and could lead to potential license overspending. Therefore, the Asset Manager SLO Best Practice package provides two sets of database instance counters to check the compliance, one of which counts consumed licenses based on physical installations (traditional mechanism) and the other counts consumed licenses based on running database (data stored in the **amMonitoredApp** table).

Pre-condition: Involved SLO business data during calculation

License type

Two out-of-box license types in Asset Manager SLO Best Practice package are mapped to Oracle's processor matrix.

- **Processor license (socket based)** - The code of license type is **LICTYPE_ORA_PROC_PERSOCKET**.
- **Processor license (core based)** - The code of license type is **LICTYPE_ORA_PROC_PERCORE**.

Workflow for core factor population

The OOB workflow **Oracle Installation pre-count** (SQL name is **sysSAM4ORA_ORA_DB**) is responsible for the pre-calculation of the Oracle software license consumption points. In this workflow, the following 5 actions are run to populate the consumed points.

SQL name of the action	Table that contains installation data	Responsibility
sysSAM4ORA_ORA_DB_CNT_Action	amSoftInstall	Calculating the consumed points for all installed database instances.
sysSAM4ORA_DB_MON_PRECALC	amMonitoredApp	Calculating the consumed points for running database instances (Calc for LMS data).
sysSAM4ORA_OPT_MON_PRECALC	amMonSWComp	Calculating the consumed points for database options that are running on the database instances (Calc for LMS data).
sysSAM4ORA_ENT_MGRPACK_PRECALC	amMonSWComp	Calculating the consumed points for enterprise management packs that are running on the database instances (Calc for LMS data).
sysSAM4ORA_ORA_WLC_CNT_Action	amSoftInstall	Calculating the consumed points for all installed Weblogic instances.

Once the workflow is triggered, the 5 actions will be executed one by one (the process may take a long time to complete depending on the amount of data). That means, before any Oracle counter calculation, the workflow needs to be run only once.

Core factor table

See "Additional information 2: Oracle core factor table" on page 64. Also see <http://www.oracle.com/us/corporate/contracts/processor-core-factor-table-070634.pdf>.

Hardware partitioning type list

See <http://www.oracle.com/us/corporate/pricing/partitioning-070609.pdf>.

Oracle processor license rule

The brief Oracle license rule for Oracle Database, Option, Management pack, and Weblogic is as follows.

- There is no license downgrade rights among different editions, this means if more than one Oracle Database edition are running on the same computer, each edition should be calculated separately.
- The CPU core factor table needs to be applied when calculating licenses for the Enterprise edition.
- For the products in an infrastructure layer cluster, the VM on which has the product installed can be migrated from one physical computer to another in the same cluster, therefore, all physical CPUs/Cores in the cluster need to be licensed.
- On a certain computer, the following formula is used to get the consumption of core-based licenses.

`Required licenses = Roundup(Physical Core count * Oracle Processor Core Factor)`

- If the software is on a physical computer, the core count is the physical core count of the physical computer.
- If the software is on a software-partitioning VM, the core count is the physical core count of the physical computer.
- If the software is on a hardware-partitioning VM, the core count is the actually used core count of the VM.

To get the consumption of socket-based licenses, use the following formula, no matter installation locates on VM or physical computer.

`Required licenses = Physical CPU count`

For Oracle Database SE2, consider its consumption of socket-based licenses as described below.

- If the software is on a physical computer or a software-partitioning VM, the formula is:

Required licenses = Physical CPU count

- If the software is on a hardware-partitioning VM, the formula is:

Required licenses = Roundup (Physical Core count/Core count per socket)

Notice that the core count is the actually core count used by the VM.

Note: The SLO package has the following assumptions for pre-condition of calculation.

- The VMs hosted by the same physical computer use the same virtualization technology. That is, the same value should be assigned to the VMType field of all those VMs. In the algorithm, for those VMs hosted by the same physical computer, we DISTINCT the value of the field VMType without any order, and use the first non-null value for calculation.
- We assume that software-partitioning and hardware-partitioning VMs cannot co-exist on a same host. When we find a software-partition VM on a host, we assume that all VMs on that host are software-partitioning VMs.

For official doc for license rule of Oracle Processor Metric, please go to <http://www.oracle.com/us/corporate/pricing/software-investment-guide/index.html> and download the Software Investment Guide pdf for Processor Metric description (page 14). Also, see rule details from <http://www.oracle.com/us/corporate/pricing/databaselicensing-070584.pdf>.

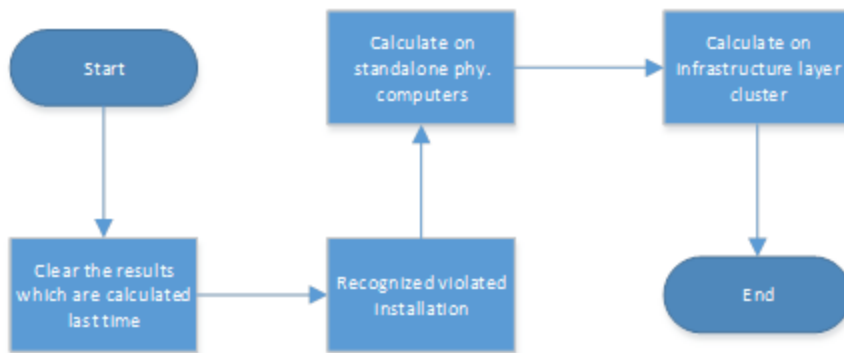
For more information about Oracle Database SE2, see https://blogs.oracle.com/UPGRADE/entry/se2_some_questions_some_answers.

Algorithm of pre-calculation

The pre-calculation workflow contains 5 actions as described above. AM uses a general algorithm in all these 5 actions. That is, for Oracle Database, Database Option, Database Management Pack and Weblogic, the same license rule is used to calculate the license consumption. As for Oracle Database, although AM has two sets of counters (traditional installation and running instance), those two sets of counters calculate with the same algorithm, the only difference is the context table used.

General process

For each product, the algorithm is as follows.



1. Delete the compliance analysis results generated by the last pre-calculation.
2. Identify installations that lead to license violation.

Note: This step is only required for Oracle Database pre-calculation and Weblogic pre-calculation.

3. Categorize all physical computers into 2 types (standalone computers and computers in infrastructure cluster), and then calculate consumptions. For details, see [Calculation process on standalone physical computer](#) and [Calculation process in infrastructure layer cluster](#)

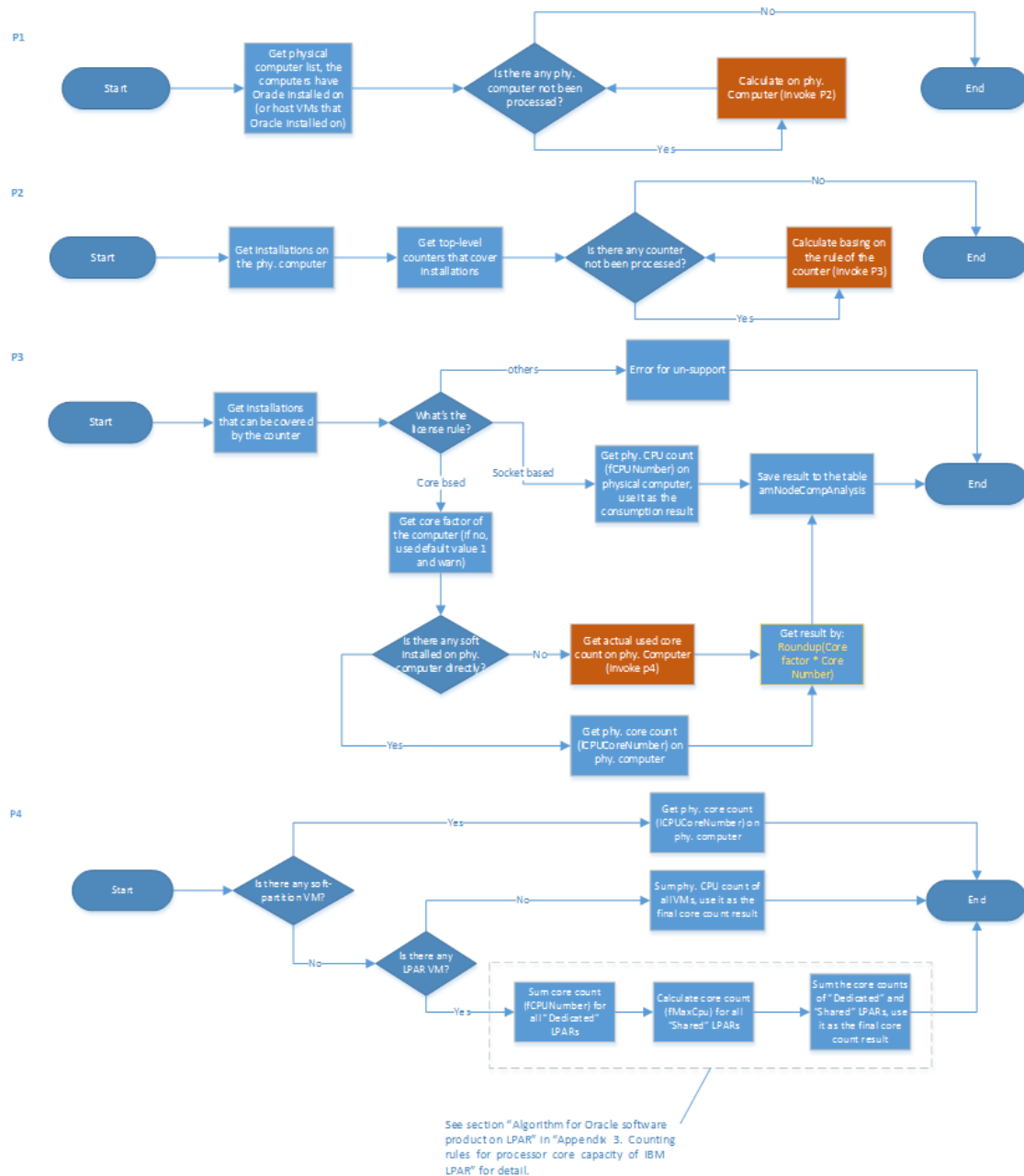
Recognizing invalid installations

Oracle claims that SE and SE1 only permit a maximum of 4 and 2 sockets respectively –occupied or not – on the physical hardware where the database is installed.

When considering the software product in an infrastructure layer cluster, the total physical CPU number of all physical computers in the cluster should not exceeds the limits.

SLO package will recognize those installations whose hardware capacity exceeds the limits, and create issue record in the **amSAMIssueAnalysis** table.

Calculation process on standalone physical computer

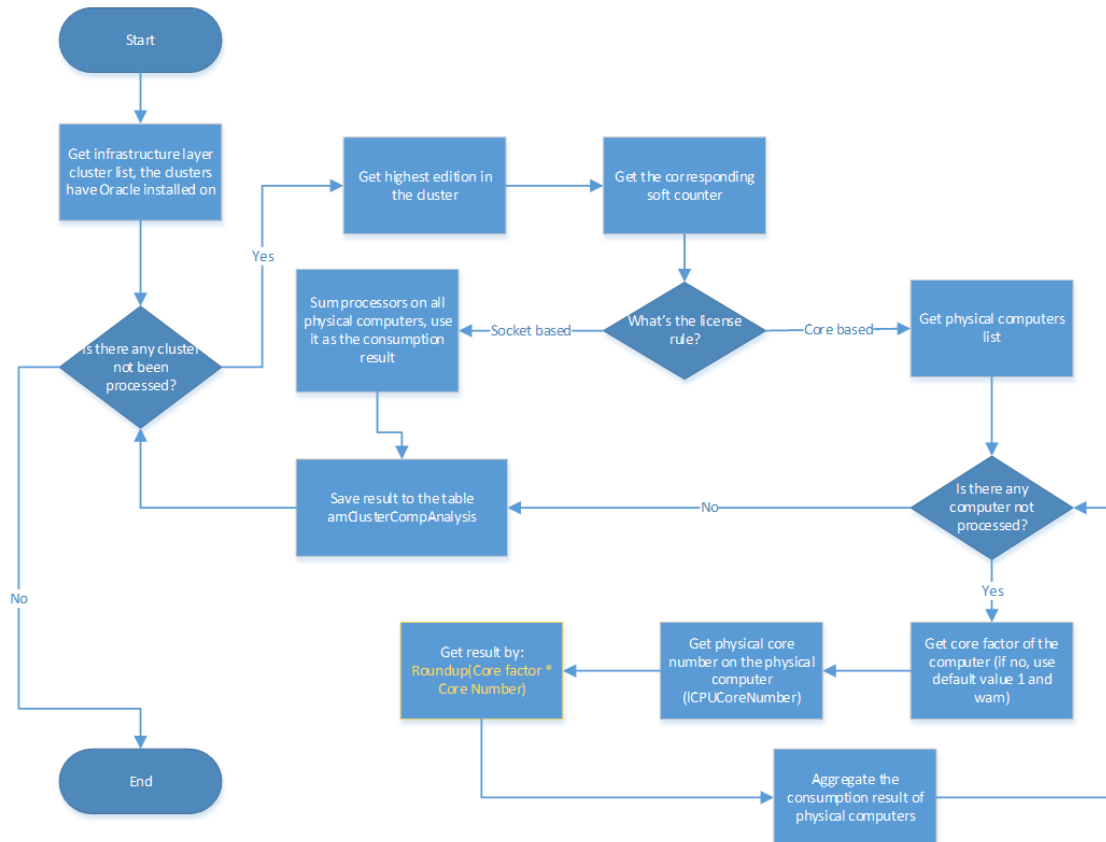


First, Asset Manager gets the standalone physical computer list. Those computers **MUST** have concerned Oracle product installed on the computer (or installed on the VMs that are hosted by the computer).

Then, Asset Manager scans through the computer list, for each physical computer on the list, Asset Manager does the following calculation.

1. Find all installations on the physical computer according to the counter's installation query, then, find the most advanced software edition based on the data in the **amLicDowngrade** table (this data is initialized during the SLO package installation and cannot be customized yet). By identifying the most advanced edition, Asset Manager can figure out the licensing rule (core based or socket based) to be used for calculation on the computer.
2. Get the core factor based on hardware information of the physical computer. In this step, Asset Manager tries to fetch value from the **amCoreFactorResult** table first. If no value can be found in the **amCoreFactorResult** table, Asset Manager uses the core factor table (stored in the itemized list) to detect core factor value, meanwhile, the value will be stored in the **amCoreFactorResult** table.
3. Check if there is any products installed directly on the physical computer. If yes, all cores (or sockets) on the physical computer need to be licensed. Then, Asset Manager calculates the consumption with the formulas above and go to step 7. If not, go to step 4.
4. Check if there is any product installed on a software-partitioning VM. If yes, all cores (or sockets) on the physical computer need to be licensed. Then, Asset Manager calculates the consumption with the formulas above and go to step 7. If not, go to step 5.
5. Check if there is any product installed on a LPAR VM. If yes, Asset Manager populates the actual used core count according to the [Oracle on Power System](#) documentation, and then calculates the consumption according to the formulas above and go to step 7. If not, go to step 6.
6. Sum up the CPU count of hardware-partitioning VMs (**amComputer.fCPUNumber**) on the physical computer, the total CPU number should be no greater than the physical core count of the physical computer. Asset Manager uses the summed number as the actually used core count, and then calculates the consumption according to the formulas above and go to step 7.
7. Save the consumption result in the **amSAMNodeCompAnalysis** table, and set the **bCountable** to 1.

Calculation process in infrastructure layer cluster



First, Asset Manager gets the infrastructure layer cluster list. Those clusters should match the following criteria.

- The cluster is infrastructure layer (`amCluster.seClusterLayer=2`), and status is “Inused” (`amCluster.Portfolio.seAssignment=0`).
- The cluster has concerned Oracle product installed on its computer component directly (or installed on the VMs hosted by the computer).
- The computer acts as an “Active” component of the cluster (`amCluster.ClusterComponents.bActivated=1`).

Then, Asset Manager scans through the cluster list, for each cluster on the list, Asset Manager does the following calculation.

1. Find all installations in the cluster according to the counter’s installation query, and then find the most advanced edition based on the data in the **amLicDowngrade** table (this data is initialized during the SLO package installation and cannot be customized yet). By identifying the most advanced edition, Asset Manager can figure out the licensing rule (core based or

socket based) to be used for calculation on the computer.

2. For socket-based rule, sum up the total physical CPU number of all active computer components in the cluster, return the value as the consumption result of the cluster. Then, go to step 4.
3. For core-based rule, scan through each active computer component in the cluster, get the Oracle core factor and physical core count (The SQL name of the field is **ICPUCoreNumber**) of the physical computer, and then use the formula above to get the consumption of the physical computer. Then, sum up the license consumption of all active computer components and return the value as the consumption result of the cluster.
4. Save the consumption results in the **amSAMClusterCompAnalysis** table and set the **bCountable** to 1. Meanwhile, for each physical computer in the cluster, the corresponding consumption result of computer is saved in the **amSAMNodeCompAnalysis** table with the **bCountable** flag set to 0.

Limitations

- The SLO Best Practice package cannot distinguish if the Oracle database is in a testing environment or a development environment.
- The SLO Best Practice package cannot build the link between an installed Oracle database instance (from the **amSoftInstall** table) and a running Oracle database instance (from the **amMonitoredApp** table).
- The SLO Best Practice package cannot detect the installed Oracle database Standard One edition.
- The push adapter cannot push running Oracle instances if the host machine's data is incomplete.

Using Oracle cluster with Asset Manager

How to build a cluster environment in Asset Manager for Oracle compliance

Asset Manager can support Oracle Real Application Cluster (RAC), to create a cluster with Asset Manager, following these steps.

1. Log on to Asset Manager with the **Admin** account.
2. Create a cluster.
3. On the **Portfolio** tab, set the **Assignment** field to **In use**.

4. On the **Cluster** tab, set the following fields:
 - **Cluster layer:** **Application Clustering**
 - **Software table:** **Running software (amMonitoredApp)**
5. In the **Cluster components** box, add **Oracle Database Enterprise Edition (Running)**, and set the **IsActive** property to **Yes**.
6. Double-click the **Oracle Database Enterprise Edition (Running)** you added, click the magnifier of the **Monitored application** item.
7. On the **Component** table, add **OracleDB option: Real Application Cluster (Running)**.
8. Repeat step 5 and step 7 to add multiple Oracle Database Enterprise Edition with RAC.

Calculation examples for infrastructure cluster

The following example shows how the license points are calculated for an infrastructure cluster.

You have a vSphere cluster that includes 4 servers with 4 VMs per server. One Oracle Database Enterprise edition instance runs on the VMs as shown in the following figure.



Hardware:

- 2 processors per server
- 4 cores per processor
- 2 cores per VM
- $2 \times 4 = 8$ total cores per server
- The factor is 0.5

Therefore, the total license points required for Oracle Database Enterprise edition is:

$$4 \text{ servers} * (2 \text{ processors} * 4 \text{ cores} * 0.5 \text{ factor}) = 16$$

Calculation examples for failover cluster

The following example shows how the license points are calculated for a failover cluster.

As shown in the following figure, you have a failover cluster that includes 4 servers (1 primary node and 3 failover nodes).



Hardware:

- 2 processors per server
- 4 cores per processor
- $2 \times 4 = 8$ total cores per server
- The factor is 0.5

Therefore, the total license points required for Oracle Database Enterprise edition is:

$4 \text{ servers} \times (2 \text{ processors} \times 4 \text{ cores} \times 0.5 \text{ factor}) = 16$

If one failover node up time is less than 10 days per year:

$3 \text{ servers} \times (2 \text{ processors} \times 4 \text{ cores} \times 0.5 \text{ factor}) = 12$

Note: In an Oracle failover cluster, if the up time of a failover node is less than 10 days, you do not need license for that computer. Currently, Asset Manager cannot obtain the up time of a computer and cannot identify if a node needs license. Therefore, an option named **License excluding** (**amSoftInstall.seLicenseExcluding** or **amMonitorApp.seLicenseExcluding**) is added. It allows you to choose if the installation needs license. If this option is selected, Asset Manager will not calculate the license points for this installation.

Classification:	
No of points:	8
Status:	
<input type="checkbox"/> Missing from monitored application	
<input checked="" type="checkbox"/> License excluding	
Role:	

Additional information 1. Algorithm for Oracle software product on LPAR

SLO package calculates the core capacity on LPAR by referencing the doc [IBM Power Systems \(AIX, iOS, Linux\)](#). Yet the doc only demonstrate how to calculate the core capacity when there is default processor pool on physical server ONLY. For the scenario that multiple processor pools on same physical computer, the algorithm is not clear.

A consultant vendor gives their algorithm based on their LPAR management experience. SLO package implements that for Oracle product counters. It is described below.

Algorithm

The count of cores to be licensed is defined as follows:

Step 1

Define the following parameters which will be used subsequently.

dblCore_ActivedInFrame: The value is equal to the value of the **fCpuNumber** field of the physical computer.

dblCore_InDedicatedFixed: The value is equal to the sum of the value of the **fCpuNumber** field of all **non-shared** dedicated LPARs.

dblCore_DefaultPoolAvailable: The value is equal to **dblCore_ActivedInFrame** minus **dblCore_InDedicatedFixed**.

Step 2

Calculate the core count to be licensed for all dedicated LPARs (including “shared” and “non-shared” dedicated LPARs) which have concerned software running on. The core count is the sum of the **fCpuNumber** field of all these LPARs. Then, define a parameter named **dblCoreLic_Dedicated** to store the value.

Step 3

Calculate the core count to be licensed for the default shared-processor pool.

Note: With the SLO Best Practice package, processor pool is mapped into the **amResourcePool** table, and the pool ID is stored in the **Code** field. For the IBM PowerVM virtualization technology, the default shared-processor pool always has the pool ID as 0, SLO Best Practice package will set the **bDefault** of the default shared-processor pool to 1.

To get the core count,

1. Sum the **fMaxCpu** field of all micro-partition LPARs on the default shared-processor pool. These LPARs must have concerned software running on.
2. Compare the result of the previous step with the value of **dblCore_DefaultPoolAvailable**, choose the smaller one.

Step 4

Calculate the core count to be licensed for other shared-processor pools.

Other than the default one, for each shared-processor pools, the core count on that pool can be calculated by the following steps.

1. Sum the **fMaxCpu** field of all micro-partition LPARs on that shared processor pool. Those LPARs must have concerned software running on.
2. Compare the result of the previous step, the value of **dbiCore_DefaultPoolAvailable**, and the pool's capacity (the **MaxCapability** field), choose the smallest one.

Step 5

Calculate core count to be licensed for the physical computer.

To get the core count,

1. Sum the core count of dedicated LPARs, default shared-processor pool, and other shared-processor pools (core counts in step 2, 3 and 4).
2. Choose the lower value of the result in the previous step and **dbiCore_ActivedInFrame**, and round up to next greater integer.

The fields that stand for core capacity of LPAR

The **fMaxCpu** field is used when calculating the core capacity of a shared LPAR. Other two fields **fEntitledCapacity** and **fOnlineVirtualCpu** also can be used instead of the **fMaxCpu** field. By default, SLO package uses the **fMaxCpu** field. If you use UD content pack whose version is equal or greater than CP18, you can choose to use the **fEntitledCapacity** and **fOnlineVirtualCpu** fields for calculation. That can be done by assigning value for a new switch in the script library. To do this:

1. Log on to AM with an administrator user account.
2. On the web client, enter "scripts" in the search box on the top, open the **Scripts** screen. On the Windows client, on the **Administration** menu, click **Script libraries**.
3. Find and choose the **SamPreCalcUtilLib** record.
4. In the detail pane, find the following line:

```
CONST LPAR_ECVP_ENABLED = x
```

The **LPAR_ECVP_ENABLED** variable is the exact switch. By default, it is set to 0, which means the algorithm of core capacity on LPAR relies on the **fMapCpu** field. Change the value to 1 and save the record so that the algorithm will rely on the **fEntitledCapacity** and **fOnlineVirtualCpu** fields.

Additional information 2: Oracle core factor table

The Itemized lists used in Oracle pre-calculation algorithm are as follows:

Identifier	Description
amSAMFactorType:\$DOUBLE_FACTOR	This is for the hardware information list which corresponds to a given CPU core factor (\$DOUBLE_FACTOR). It comes from the Oracle CPU core factor table. If the host computer's hardware information matches any listed value, the given CPU core factor should be applied. For more information, refer to http://www.oracle.com/us/corporate/contracts/processor-core-factor-table-070634.pdf .

Customization on amSAMFactorType:\$DOUBLE_FACTOR (the CPU core factor table)

Currently, Oracle has 4 CPU core factors: 0.25, 0.5, 0.75, and 1.0. You do not need to define for the factor 1.0 because it will be used automatically if:

- No other factors can be matched.
- Virtual machine using the software partitioning technology cannot find the host computer.
- The CPU type of the host compute is null.
- The CPU number of the host computer is 0.
- The core number of the host computer is 0.

For each listed value, it can be combined by multi-type hardware information with the following format: "AAA:BBB:CCC:...". For example: ULTRASPARC-T1:LCPUSPEEDMHZ:1000,1200.

The character ":" divides the value into several parts. The first part should always be the CPU type (related to the **amComputer.CPUType** field). The second part is a keyword that represents the hardware information type, and the third part is the value to be matched. All these information can be explained by following the same rule. The acceptable hardware information type keywords are as follows:

Keyword	Description
MODNO	The value should be matched to the amComputer.CPUInternal field of the host computer.
ICPUSpeedMHz	The value should be matched to the amComputer.ICPUSpeedMHz field of the host computer.
CPUNum	The value should be matched to the amComputer.fCPUNumber field of the host computer.

Keyword	Description
PerCoreNum	The value should be matched to the host computer's CPU core number per processor value. CPU core number per processor= amComputer.ICPUCoreNumber/ amComputer.fCPUNumber
HWMDL	The value should be matched to the amComputer.Model.Name field of the host computer.
dtPURCH	The value should be matched to the amComputer.Portfolio.Asset.dAcquisition field of the host computer.

For the value part, it also can be combined by multiple values with the following format: "AAA-BBB,%CCC%DDD%,EEE". The match rule is:

- The character ":" is the separator for values, each value segment needs to be extracted before processing.
- For each value segment, if it contains the character "%", it will be recognized as a separator, each part will be checked if it is contained in the host computer's corresponding hardware information. Only when every part of it is matched, the value segment can be matched. Otherwise, you must check the next value in the itemized list. For example:

Source string to be checked	Value rule in core factor itemized list	Result
ULTRAPARC T1	%ULTRAPARC-T1%	Cannot match. "ULTRAPARC-T1" failed to match.
ULTRAPARC T1	%ULTRAPARC%T1%	Match. Both "ULTRAPARC" and "T1" match successfully.
ULTRAPARC T1	%ULTRAPARC%T2%	Cannot match. "ULTRAPARC" match successfully but "T2" failed.
XEON E5-2629	%XEON%E5-26%	Match. Both "XEON" and "E5- 26" match successfully.
XEON E5-2629	%XEON%E5-2600%	Cannot match. "XEON" match successfully but "E5- 2600" failed.

- For each value segment, if it contains relational operators ("<", ">", ">=", "<=", "<>") and the value to be matched is numeric, the value segment will be recognized as a data comparison and the value will be checked if it satisfy the comparing condition. Notice that the only acceptable date format is YYYY-MM-DD, and it needs to be surrounded with "#". For example, you can define a date comparison statement like ">#2010-07-06#".
- For each value segment, if the above rules do not apply, the equivalent check takes place.

For a listed value, only when all combined multi-type hardware information is matched, the given CPU core factor can be applied to the host computer.

Additional information 3. Oracle hard partitioning technology list

SLO package stores Oracle hard partitioning technology list in the **amItemizedList** table. The Itemized lists used in algorithm are as follows:

Identifier	Description
amSAMVMPartition:Hard	This is for the hardware partitioning VM type list. The VM type value is stored in the amComputer.VMType field. If an IT equipment is a VM and its VMType value matches one of the listed VM types, that means the VM is built with the hardware partitioning technology. For more information, refer to http://www.oracle.com/us/corporate/pricing/partitioning-070609.pdf .

Customization on amSAMVMPartition:Hard

Each listed value is one of hardware partitioning VMType. The value should come from the **amComputer.VMType** field and exactly be same (case sensitive) as the value of that field.

IBM

This section describes the pre-calculation workflow and algorithm of IBM software counters.

IBM software counter overview

For supported IBM software product list, refer to the *Supported Software Titles* doc.

Currently, only the PVU licensing is covered by the SLO Best Practice package. The number of PVUs required is based on the licensing offering used, the processor technology and the number of processor cores available to the software to license.

Two solutions for IBM software counters

In the SLO Best Practice package, there are two sets of counters for IBM PVU software products:

- The counters work by applying the **Advanced process**. These counters are calculated basing on the inventory data in AM and the calculation algorithm is implemented by SLO Best Practice package by aligning with IBM docs.
- The counters which have the “ILMT” label in the name. Those counters work by applying the **Simple calculation process**. Those counters do not involve inventory data in AM, they rely on the data integrated from ILMT.

Using IBM software counters

The SLO Best Practice package follows the IBM licensing guide and accurately calculates the consumed IBM licenses. By generating the consumption data that is needed by IBM auditing, this solution enables the customers to track the intermediate calculation parameters and to validate the final result.

Currently, the following virtualization technologies are supported:

- PVU license counting for LPAR/DLPAR (Linux on Power System is not supported). See [IBM Power Systems \(AIX, iOS, Linux\)](#).
- PVU license counting for X86 platform. See [x86 \(Intel / AMD\) based servers](#).

The following enhancements for the IBM PVU counters are introduced in Asset Manager SLO Best Practice package:

- To align with the way that IBM audits, the IBM license counters calculate the compliance result on the IBM software edition level.

Note: The Asset Manager SLO Best Practice package still provides both sub-capacity and full-capacity counters for IBM software.

- The schema is extended to store the historic consumption result of the IBM PVU counters. This enables the customers to get the peak value of the consumption result for IBM auditing.

The IBM software counters work by applying Advanced process.

Pre-condition: Involved SLO business data during calculation

License type

An out-of-box license type in Asset Manager SLO Best Practice package is mapped to IBM's Processor Value Unit matrix.

- **Processor Value Unit** - The code of license type is **SAM4BP_T101**.

Workflow for core factor population

By default, every quarter, a workflow named **sysSAM4IBM_CF_WK** invokes the **sysSAM4IBM_CF_Action** action to update the **amCoreFactorResult** table regularly. For more information about the **amCoreFactorResult** table, see ["Appendix 1. Data model and definition" on page 82](#).

1. Scan the **amCoreFactorResult** table, delete any record whose **bConfirm** field is 0 and **ILicTypeld** is "By PVU".
2. Search for any physical computer that has IBM software installed. The IBM software can be installed directly on the physical computer or on the virtual machine running on the physical computer.

Note: It is assumed that all IBM software has the same characteristics. The brand in its **SoftInstall** model is IBM, the conditional expression is `SoftInstall.Model.Brand.BarCode='IBM'`.

3. For the selected physical computers, match them with the configuration information of the core factor table, save the corresponding core factor in the **amCoreFactorResult** table.
4. For all physical computers that do not have the core factor value assigned because of missing/incorrect hardware information, set the core factor value to the default value (0), and then set the value of **bEstimate** to 1.

Workflow for pre-calculation

Asset Manager SLO Best Practice package provides a workflow to pre-calculate the consumed license points of IBM PVU products automatically.

Workflow

SQL name	Name of workflow	Responsibility
sysSAM4IBM_CNT_WK	IBM Installation pre-count	Workflow to pre-calculate consumed license points of IBM PVU products.

Action

SQL name	Context table	Responsibility
sysSAM4IBM_CNT_ACT	amSoftInstall	Core action to calculate the consumed license points of IBM PVU products.

Core factor table

See "Additional information 2. IBM PVU core factor table" on page 75. Also see https://www-01.ibm.com/software/passportadvantage/guide_to_identifying_processor_family.html.

IBM license rules

The current implementation of the SLO Best Practice Package enables calculation for both sub-capacity and full-capacity.

On a certain computer, the following formula is used to get the consumption of full-capacity:

Required licenses = Physical core count * PVU Table per Core

The following formula is used to get the consumption of sub-capacity:

Required licenses = Actual used physical core count * PVU Table per Core

For more information about sub-capacity and full-capacity, see [Processor Value Unit \[PVU\] licensing for Distributed Software](#).

For sub-capacity, IBM defines counting rules for different virtualization technologies. For more information, see [Virtualization Capacity License Counting Rules](#). However, the current version of the SLO Best Practice Package only supports LPAR and x86 virtualization technologies.

For more information about the fields of the amComputer table that are used to calculate the LPAR virtualization capacity, see ["Additional information 1. Algorithm for IBM PVU software product on LPAR" on page 75](#).

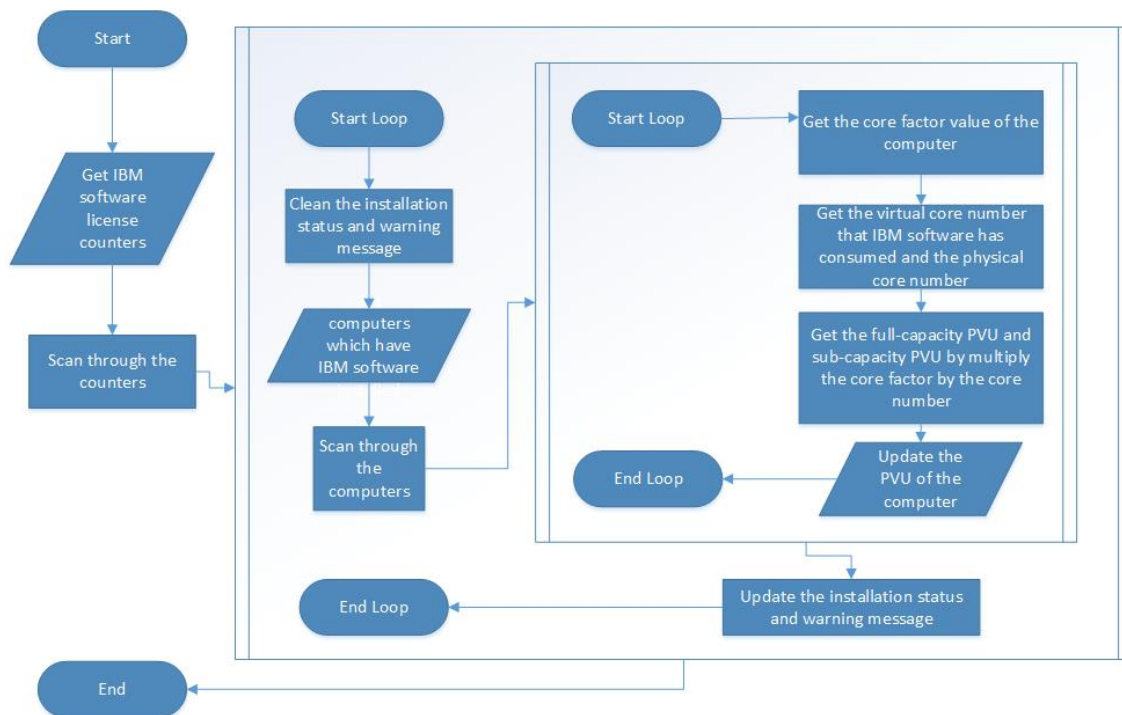
Algorithm of pre-calculation

The algorithm of the pre-calculation action **sysSAM4IBM_CNT_ACTION** is as follows:

1. Find all IBM software license counters for sub-capacity.
2. Scan through the counters. Calculate the PVU consumption for each counter.
 - a. Clear the sub-capacity consumption, full-capacity consumption, warning information, and status flag of the previous calculation.
 - b. Use the license counter to find:
 - All physical computers that have IBM software installed directly.
 - All physical computers running virtual machines that have IBM software installed.
 - c. Invoke the **VerifyAndReturnCoreFactor** function to obtain the core factors of all the physical computers selected in step b.

Note: The data of some physical computers may be synchronized to Asset Manager after the **sysSAM4IBM_CF_WK** workflow is run. Therefore, the core factor values of these physical computers are not stored in the **amCoreFactorResult** table. In this case, the **VerifyAndReturnCoreFactor** function also calculates these core factor values and stores them to the **amCoreFactorResult** table. For more information about the **amCoreFactorResult** table, see ["Appendix 1. Data model and definition" on page 82](#).

- d. Invoke the **calculateCoreNumberByVMList** function to obtain the core number that is used to calculate the sub-capacity consumption. If the IBM software is installed directly on the physical computer, the returned value is the core number of the physical computer. If the IBM software is installed on the virtual machine, the returned value is the maximum core number used by the virtual machine. The maximum core number used by the virtual machine cannot be larger than the core number of the physical computer.
- e. Query the core number of the physical computer (core number used to calculate full-capacity consumption).
- f. Multiply the core factor by the core number of sub-capacity consumption to get the PVU value of sub-capacity; multiply the core factor by the core number of full-capacity to get the PVU value of full-capacity.
- g. Invoke the **saveOrUpdateIBMConsPoints** function to store the two PVU values to the **amIBMConsPoints** table. For more information about the **amIBMConsPoints** table, see the *Consumption detail table for IBM PVU product* section in ["Appendix 1. Data model and definition" on page 82](#).
- h. Update the **IUseCount** field and the **dUtilization** field of the **amSoftInstall** table with the two PVU values.
- i. Update the warning message and status flag of the current license counter of the IBM software installation.



Error handling

The following two fields of the **amSoftInstall** table are used to track IBM PVU warning information.

SQL name	Description	Field type	Comment
ErrorInfo	Error information of the license rule calculation	Long text field	
seCalculateFlag	Calculate flag for license rule	System itemized list	Calculated successfully 0 Calculated with risk 1 Estimated result 2 Cannot be calculated 3 Other 99

The following table lists the error scenarios and corresponding warning information.

Scenario	seCalculateFlag	ErrorInfo
LPAR VM does not belong to a physical computer.	Cannot be calculated 3	Cannot find the physical computer that hosts this VM.
The core number of the VM is 0. Or, for a LPAR VM, the value of the fMaxCpu field is 0.	Cannot be calculated 3	The CPU number of the VM is 0.
The core number of the physical computer is 0.	Cannot be calculated 3	The core number of the physical computer is 0.
The physical computer contains more than one processor pool and donating LPAR VMs exist.	Cannot be calculated 3	The physical computer contains more than one processor pool and donating LPAR VMs exist.
For a LPAR VM, the value of VmLPARType or VmMode is invalid.	Cannot be calculated 3	Invalid information for LPAR VM.

Scenario	seCalculateFlag	ErrorInfo
For a shared LPAR VM, the processor pool is not assigned or the pool capacity is 0.	Cannot be calculated 3	Invalid information for processor pool.
Cannot confirm the core factor value of the physical computer due to missing hardware information.	Cannot be calculated 3	<p>Cannot populate the correct core factor value of this physical computer because some hardware information is missing or mismatched in the core factor table configuration items.</p> <div> <p>Note: When you see this warning message, you need to check the Model, CPU type, Number of sockets, and the Cores per socket fields on the physical computer. If the CPU type shows the processor is Intel Xeon, you also need to check the CPU version field and make sure that its value contains the processor model number, for example, "E5500".</p> </div>

Note: If multiple errors exist, all warning messages are stored in the **ErrorInfo** field.

How to validate the IBM PVU counter result

Basically, IBM uses the following formula to calculate the PVU consumption:

PVU consumption = core factor * core number

Therefore, to validate the result of the IBM PVU counter, you need to check both the core factor and the core number.

Step 1. Check the core factor

In the **Computer** screen of the physical computer, click the **Core factor result** tab, find the record with the **License type** field set to "By PVU". If the **Factor value** field is "0" and the **Estimate** field is "Yes", some hardware information of the physical computer is missing or mismatched in the core factor table configuration items, and the SLO Best Practice package cannot detect the core factor value based on these information. In this situation, you must correct the hardware information and make sure that the hardware information matches the configuration item in the core factor table.

Step 2. Check the core number

The consumption results for each physical computer are stored in the **amIBMConsPoints** table. To access the screen of this table, click **List of screen** on the **Administration** menu and then open the selected item in the following figure.

Table	Screen	Domain	Screenset
IBM consumption points (amIBMConsPoints)	IBM consumption points of product per computer (amIBMConsPoints_Full)	SAM	Full
IBM consumption points (amIBMConsPoints)	IBM consumption points of product per computer (amIBMConsPoints_Simple)	SAM	Simple

The following figure shows an example of the records in the **amIBMConsPoints** table.

Computer	Product model	License type	Value unit
CPU000048		By PVU	50
CPU000049		By PVU	50

Description: <input type="text"/> Product model: <input type="text"/> Aggregation start date: 7/21/2014 2:37:39 PM License type: By PVU Cores in full capacity: 64 Cores in subcapacity: 24 Cores limit: 0		Computer: CPU000048 Software license counter: IBM WebSphere Application Server Aggregation end date: 7/31/2014 5:22:14 PM Value unit: 50 Consumption points in full capacity: 3200 Consumption points in subcapacity: 1200 Consumption points in limit mode: 0	
--	--	--	--

To verify the core number, follow these steps:

1. Filter the records by using the **Software license counter** field and the **Computer** field.
2. Check the record that has the latest date in the **Aggregation end date** field. Make sure that the date is the same day when the latest counter calculation occurs. If it is not the same day, that means during the latest calculation, the SLO Best Practice package did not get any result on this computer. For more information of the cause of this issue, see the error information table in ["Error handling" on page 72](#) ["How to validate the IBM PVU counter result" on the previous page](#)

Note: The calculation of the counter MUST run on the same day when the pre-calculation action runs. For example, the pre-calculation action starts running on August 20, and then the records in the **amIBMConsPoints** table are stored with this time stamp (August 20). One day later, the counter does the calculation and the counter result is stored in the **amSoftLicCounter** table with this time stamp (August 21). The offset between these two time stamps may lead to mismatch between the peak value and the consumption details in the IBM PVU report. Therefore, it is recommended that you schedule the pre-calculation action after the midnight and then schedule the counter's calculation on the same day.

3. The **Value unit** represents the core factor used during the calculation which works out the current result.
4. The **Cores in subcapacity** field and the **Cores in fullcapacity** field represent the core numbers that are used in sub-capacity and full-capacity calculation. The consumption points represent the consumption result based on **Value unit** and **Cores**.

Note: The value of the **Cores in full capacity** field is equal to the total core number of the physical computer (the **Core CPU count** field in the **amComputer** table). The calculation of the **Cores in subcapacity** field is more complex. For more information, see *Counting rules for IBM PVU processor core capacity*.

In addition, the hardware information of the virtual machine and the physical computer reflects the latest network infrastructure. Therefore, when you check the information of the virtual machine or the physical computer, make sure that no such hardware information is refreshed in Asset Manager after the counter did calculation .

Using IBM software counters (ILMT based)

The SLO Best Practice package can also leverage the results from IBM License Metric Tool and import the results to the Asset Manager database (**amIBMConsPoints** table), and then generate the corresponding dashboard and report. Therefore, a set of OOTB counters – which have the label “ILMT” in the name – calculate the compliance result basing on the data in **amIBMConsPoints** table.

An OOB scenario - whose name is ILMT.scn - is used to import the consumed PVU in the environment to the Asset Manager database. This scenario is located in ILMT folder of the installation package.

The IBM software counters (ILMT based) work by applying the **Simple process**.

Additional information 1. Algorithm for IBM PVU software product on LPAR

Refer to the doc [IBM Power Systems \(AIX, iOS, Linux\)](#).

Note: The implementation supports single processor pool only.

Additional information 2. IBM PVU core factor table

The itemized lists used in the algorithm are as follows.

Identifier	Description
amSAM4PVUType:SafeList	Define the processor name. The model number of the processor needs to be checked through the CPUInternal field.
amSAM4PVUType:0	Define the default PVUs per core when the processor cannot be matched to the item option in the IBM "PVU table per core".
amSAM4PVUType: \$ INTEGER_PVU_TYPE	These identifiers are for the hardware information list that corresponds to the given PVUs per core (\$INTEGER_PVU_TYPE). These identifiers are from the IBM "PVU table per core". If the host computer's hardware information matches any listed value, the given PVUs per core should be applied. For more information, see http://www-01.ibm.com/software/passportadvantage/pvu_licensing_for_customers.html

Customize amSAM4PVUType: \$ INTEGER_PVU_TYPE (the PVUs per core)

Currently, IBM has 6 PVUs per core types: 30, 50, 70, 80, 100 and 120. If the hardware information cannot match one of these 6 types, Asset Manager returns the default PVUs per core type, which is defined in the **amSAM4PVUType:0** identifier.

- For each listed value, it can be combined by multi-type hardware information with the following format “AAA:BBB:CCC:...”. For example: SPARC%T4:Model:T4-4:Sockets:<= 4:xCores:> 1.
- The character “:” divides the value into several parts. The first part should always be the processor name (related to the **amComputer.CPUType** field) or a specific flag. The second part is a keyword that represents the hardware information type, and the third part is the value to be matched.
- Only when all listed hardware information options are matched, the processor matches the current PVUs per core type. Otherwise, the algorithm will continue to check the next listed value line.

All these information can be explained by following the same rule. The acceptable hardware information type keywords are as follows.

Keyword	Description
Model	The value should be matched to the amComputer.Portfolio.Model.Name field of the host computer.
ModNo	The value should be matched to the amComputer.CPUInternal field of the host computer.
Sockets	The value should be matched to the amComputer.INumSockets field of the host computer.
xCores	The value should be matched to the host computer’s core number per processor value, amComputer.INumCorePerSocket .
AAA	The flag to define the following settings. It contains processor names that need to be excluded.

When the itemized line starts with “AAA”, the following keywords are allowed.

Keyword	Description
xCores	The value should be matched to the host computer’s core number per processor value (amComputer.INumCorePerSocket).
xCPU	The value should NOT be matched to the amComputer.CPUType field of the host computer.

For the value part, it also can be combined by multiple values with the following format “AAABBB,%CCC%DDD%,EEE”. The matching rules are as follows.

- The character “,” is the separator of values, each value segment must be extracted before processing.
- Specifically, for the keyword **ModNo**, separator is “;” instead of “,”.

For example,

Source information to be checked	Value rule in the itemized lists	Result
CPUType: ULTRAPARC T1 INumCorePerSocket: 4	ULTRAPARC%T1:xCores:> 1	Match. Both “ULTRAPARC” and “T1” are matched. Core number per processor is matched as well.
CPUType: ULTRAPARC T2 INumCorePerSocket: 4	ULTRAPARC%T1:xCores:> 1	Dismatch. “ULTRAPARC” is matched successfully. However, “T2” is not matched.
CPUType: ULTRAPARC T1 INumCorePerSocket: 1	ULTRAPARC%T1:xCores:> 1	Dismatch. Both “ULTRAPARC” and “T1” are matched. However, the core number per processor is not matched.
CPUType: INTEL XEON CPUInternal: INTEL XEON E6-3300 INumCorePerSocket: 8	INTEL%XEON:ModNo:3000-3399;5000-5499;7000-7499:xCores:> 1	Match. All conditions are matched successfully.
CPUType: Pentium 4 Extreme Edition INumCorePerSocket: 1	AAA:xCores:= 1:xCPU:POWERXCELL,CELL/B.E.8I	Match. The core number per processor is matched. The CPUType (not on the list) is matched as well.
CPUType: POWERXCELL INumCorePerSocket: 1	AAA:xCores:= 1:xCPU:POWERXCELL,CELL/B.E.8I	Dismatch. The core number per processor is matched. However, the CPUType is not matched.

- For each value segment, if it contains relational operators (“<”, “>”, “>=”, “<=”, “<>”) and the value to be matched is numeric, the value segment is recognized as a data comparison. The value will be checked if it meets the comparing condition.
- Range operators (“-”, “in (xxx, xxx)”) are also allowed.

For operator "-", only when both sides of "-" and the value segment are all numeric, the "-" is recognized as a range operator. Otherwise, the "-" is only recognized as a normal character.

- For each value segment, if it contains the "%" character, it is recognized as a separator, each part will be checked if it is contained in the host computer's corresponding hardware information. Only when every part of it is matched, the value segment can be matched. Otherwise, you must check the next value in the itemized list.
- Leading and ending "%" means that the value segment must be matched in the attribute value.
- If the rules above do not apply to the value segment, the equivalent check takes place.

For a listed value, only when all combined multi-type hardware information is matched, the given PVUs per core can be applied to the host computer.

Adobe

For supported Adobe software product list, refer to the *Supported Software Titles* manual.

The Adobe products are licensed per installation, regardless of whether they are installed on physical or virtual platforms.

All Adobe software counters work by applying the **Simple process**.

Autodesk

For supported Autodesk software product list, refer to the *Supported Software Titles* manual.

All Autodesk software counters only support the stand-alone license type.

Stand-alone licenses are called "node-locked" licenses, because each license is bound to a single user and computer. For some Autodesk products, you can install a second copy of a stand-alone product for non-concurrent use by a single user. For more information, see the Software License Agreement (SLA) for your product.

The Autodesk licensing rule is described in detail on the Autodesk website:

1. Go to <http://usa.autodesk.com/support/>.
2. Click the **Licensing, Registration & Activation** link.

All Autodesk software counters work by applying the **Simple process**.

BMC

For supported BMC software product list, refer to the *Supported Software Titles* manual.

BMC Performance Manager has a base license that is licensed per installation of the application and per node for monitored objects.

All BMC software counters work by applying the **Simple process**.

HP

For supported HP software product list, refer to the *Supported Software Titles* manual.

HP Asset Manager software counters work by applying the **Advanced process**, other HP software counters work by applying the **Simple process**.

Using HP software counters

This section describes the pre-calculation workflow and algorithm of Asset Manager software counters.

Pre-condition: Involved SLO business data during calculation

Workflow for pre-calculation

Asset Manager SLO Best Practice package provides a workflow to pre-calculate the consumed license points of HP products automatically. That includes a pre-calculation for Asset Manager license points.

Reference	Name of workflow	Responsibility
sysSAM4HP_AM	HP AM installation pre-count	Workflow to pre-calculate consumed license points of HP product.

Action

Reference	Table that contains installation data	Responsibility
sysSAM4HP_AM_CNT_Action	amEmplDept	Core action to calculate the Asset Manager consumed license points.

License rules

Asset Manager can be used by three types of users: Named, Floating or Guest users.

Named users have an account that gives guaranteed full access to the application. Floating users have a full access to the application and the number of floating users is evaluated on a concurrent basis. For instance, if a company acquired 20 floating users, it can have more than 20 floating user accounts, but the 21st user logging concurrently to the application will be rejected.

Guest users have a limited (mostly read-only) access to the Asset Manager data and are aimed for casual access processes such as creating an internal request, validating a request, accessing a report or dashboard, and browsing data.

Named, Floating and Guest user licenses can be acquired for each of the modules (Portfolio, Procurement, Contract Management, Financial Management, Software Asset Management). The

number of users of the Portfolio (base) module must be equal or superior to the max number of users of each of the other modules. In addition, each user using any of the modules must be covered by a portfolio module user license.

Red Hat

For supported Red Hat software product list, refer to the *Supported Software Titles* manual.

Red Hat uses the software subscription model for the licensing of JBoss Enterprise Middleware products. The JBoss subscription is based on the total number of processor cores, no matter the type of the processor core and whether it's virtual or physical core on which the product is installed.

See <http://www.jboss.com/pdf/JBossSubscriptionGuide.pdf>.

All Red Hat software counters work by applying the **Simple process**.

SAP

For supported SAP software product list, refer to the *Supported Software Titles* manual.

The licensing model for SAP is described on the SAP website at <http://www.sap.com/buy-now/index.epx> (the *Learn about SAP's Licensing Model* link).

Of the two elements of the SAP licensing model, only the Named user licensing is covered by the SLO Best Practice package. The package licensing is out of the scope.

All SAP software counters work by applying the **Simple process**.

Using SAP counters

The Asset Manager SLO Best Practice package only covers the Named user license compliance for SAP titles.

Due to the large number of the SAP named user types and categories, only three SAP counters have been provided as examples. You should create counters to cover all the actual user types and categories of named user licenses purchased by your organization, in line with the way the named user licensing is defined by SAP.

Usually software counters reconcile software installation points with license points.

However, the SAP's named user licensing rule does not concern the number of installed instances, rather it concerns the number of users who use certain SAP functionalities. Such usage information is not available by the means of discovery tools.

The user count can be collected by an SAP tool - License Administration Workbench (LAW), which is usually used during the SAP license audit process. The consolidated user data is stored in the LAW system, and you can download the data to a local file. Refer to SAP documentation for detailed instructions.

Then you can use the provided HP Connect-It scenario **named_user_inst.scn** to import the user data to Asset Manager. The user data should be exported as a delimited text file, in a format that conforms to that indicated in the Connector description file **consolidation_result.dsc**

accompanying the scenario. The user count is imported as software installations which enable SAP counters to perform its function like other normal counters.

Symantec

For supported Symantec software product list, refer to the *Supported Software Titles* manual.

Symantec EndPoint Protection, Symantec Antivirus, and Symantec Endpoint Encryption have a per-workstation licensing mode.

All Symantec software counters work by applying the **Simple process**.

Tibco

For supported Tibco software product list, refer to the *Supported Software Titles* manual.

Tibco software is licensed based on the number of CPU cores. The license rules vary with products on a contract by contract basis. The SLO Best Practice package currently uses a simple core count approach to the calculation of TIBCO software compliance.

All Tibco software counters work by applying the **Simple process**.

VMWare

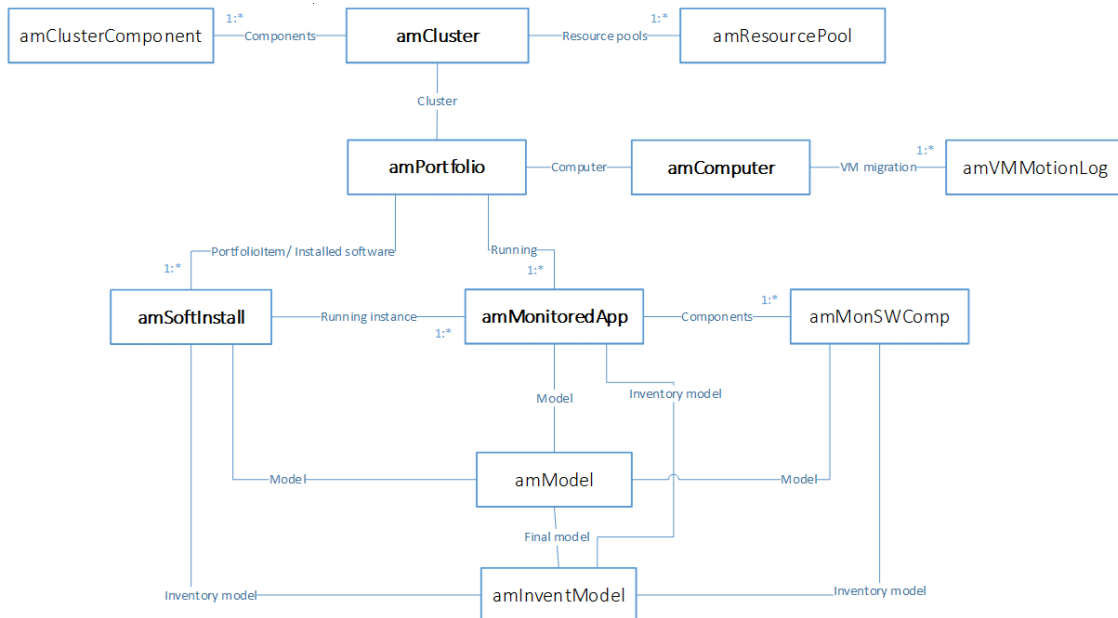
For supported VMWare software product list, refer to the *Supported Software Titles* manual.

All VMWare software counters work by applying the **Simple process**.

Appendix 1. Data model and definition

This section describes the definition and technical details of the SAM data model.

Infrastructure data model



The calculation of the software counter is based on qualified infrastructure data, so it is important to make sure that your inventory data is correct and sufficient. You can use the “Health check” feature to validate the data.

Inventory data used by software counter

Typically, software counter involves the following inventory data during the calculation.

License asset

Used to calculate the amount of license points you own. (Main table: amAsset)

Attributes involved in calculation (SQLName)	Description	Remark
amAsset.Model	Indicate the type of license you own	From model name, AM can recognize the title/edition/version of the product.
amAsset.Portfolio.seAssignment	Indicate the status of the license asset	Only value 0 or 1 will be treated as valid for license and will be taken into account during calculation.

Attributes involved in calculation (SQLName)	Description	Remark
amAsset.ISoftLicUseRights	Indicate the amount of entitled license points of the license asset	

Software installation

Inform the counter of the products you have deployed (Main table: amSoftInstall)

Attributes involved in calculation(SQLName)	Description	Remark
amSoftInstall.Model	Indicate the type of the software products that you have deployed	From model name, AM can recognize the title/edition/version of the product.
amSoftInstall.bMissing	Indicate if the software is missing from computer	Only the installation with value 0 of this field will be taken into account during calculation.
amSoftInstall.seType	Indicate the installation type	Only the installation with value 0 of this field will be taken into account during calculation.
amSoftInstall.ParentPortfolio	The computer where the software is installed	Could be either a VM or a physical computer.
amSoftInstall.seLicenseExcluding	Indicate if the installation can be ignored for calculation	By default, the value is 0, which means it is NOT excluded. Customer can set it to 1 manually if he/she can confirm the installation does not need to be entitled.

Computer

Provide hardware information which is required by license rules (Main table: amComputer)

For physical computer:

Attributes involved in calculation (SQLName)	Description	Remark
amComputer.Model	The computer's model name provided by the manufacturer	For example: SunFire 280.
amComputer.Portfolio.seAssignment	Mention the computer's status	Only the computer with value 0 of this field will be taken into account during calculation.
amComputer.CPUType	CPU type	Used for core factor detection.
amComputer.CPUInternal	CPU version	
amComputer.ICPUSpeedMHz	CPU speed	
amComputer.fCPUNumber	Physical CPU count	
amComputer.ICPUCoreNumber	Total physical core count	

For virtual machine:

Attributes involved in calculation (SQLName)	Description	Remark
amComputer.Portfolio.seAssignment	Indicate the status of the computer	Only the computer with value 0 of this field will be taken into account during calculation.
amComputer.VmType	The virtualization technology used to create the VM	This is used when recognizing VM with hardware partitioning.
amComputer.ICPUCoreNumber	Total core count used by the VM	Generally, it equals to the value of the amComputer.fCPUNumber field of the same VM

Attributes involved in calculation (SQLName)	Description	Remark
amComputer.Portfolio.Parent	The physical host where the VM is located	The attributes are used for core count calculation for LPAR VM. See appendix for more information.
amComputer.VmLPARType	LPAR Type	
amComputer.VmMode	VM mode	
amComputer.ProcessorPoolCode	Processor pool in which the VM is defined	
amComputer.ProcPoolCapacity	The capacity of the processor pool	
amComputer.fMaxCpu	Maximum number of the processors that can be assigned to the VM	

Cluster

Special calculation is needed when software is deployed in an infrastructure layer cluster (Main table: **amCluster**)

Attributes involved in calculation (SQLName)	Description	Remark
amCluster.Portfolio.seAssignment	Indicate the status of the cluster	Only the cluster with value 0 of this field will be taken into account during calculation
amCluster.Portfolio.seClusterLayer	Cluster layer	

The definition of the **amCluster** table can be found in the following table.

Field/Link	Comment
Model	As a type of portfolio, the Model link is mandatory for a cluster.

Field/Link	Comment
Assignment	In the out-of-the-box implementation, cluster license rules are applied when the software is installed on a computer that is part of an In use cluster. If the value of the Assignment field is not In use , SLO Best Practice package calculates license points by applying non-cluster license rules.
Cluster layer	<p>SLO Best Practice package defines the following three types of clusters:</p> <ul style="list-style-type: none"> • Application clustering: This type of clustering is typically built into applications that hand over replication and failover to another server independently without involving the operating system. By installing the application on two different servers, the clustering feature can be configured in the application. An example of this type of clustering is Oracle RAC. • OS clustering: This type of clustering is handled by the operating system that is responsible for synchronizing and cutting over an application from one server to another. The typical architecture for this type is two servers that use a shared disk where the application resides. An example of this type of clustering is Microsoft Cluster Server (MSCS). • Infrastructure clustering: This type of clustering runs not on its own kernel instead of a third-party operating system. It can consist of different hardware components inside and outside a server that prevents a single hardware component failure from crashing a server. An example of this type of clustering is VMWare ESX Server.
Technology	The technology used in a cluster. Reserved for extension and not used in the out-of-the-box implementation.
Cluster type	Cluster type. This value can be Loadbalancing, Mirroring, Failover, and Standby .

Field/Link	Comment
Resource pools	The resource pools created in the cluster. Reserved for extension and not used in out-of-the-box implementation.
Software table	<p>When the value of the cluster layer is Application Clustering, you need to set this field to mention which table the component comes from.</p> <p>This value can be amSoftInstall or amMonitoredApp. If you want to build an Oracle RAC cluster and use Oracle LMS package, this field should be amMonitoredApp. Otherwise the value should be amSoftInstall.</p>

A record in the **amClusterComponent** table represents component information in a cluster. For an application layer cluster, the component may be a software installation or a monitored application. For an OS/infrastructure layer cluster, the component could be either a virtual machine or a physical server. The definition of the **amClusterComponent** table can be found in the following table.

Field/Link	Comment
Computer	When the cluster layer is OS clustering or Infrastructure clustering, this field represents the computer as the component.
Software installation	When the cluster layer is Application clustering and software table is amSoftInstall , this field represents the component of the cluster.
Monitored application	When the cluster layer is Application clustering and software table is amMonitoredApp , this field represents the component of the cluster.
Component role	<p>This field represents the role of the component in the current cluster.</p> <p>For a failover or a mirroring cluster, the role can be owner or standby.</p>
IsActive	This field identifies if the component plays the defined role in the cluster. If it is not active, it means that the component is not available in the cluster, and the component should not be counted while calculating in the cluster, but it needs to be considered as in a non-cluster environment.

Field/Link	Comment
Maximum VM number	This field represents the capability of running a number of virtual machines in the component. It is meaningful when the component is a physical server and the cluster is set up for failover.

VM migration log

In cluster, when VM is migrated from one physical computer to another, the cluster management application can log the migration event. Such kind of information would impacts the software compliance. Asset Manager stores those migration logs in the **amVMMigrationLog** table.

The record in the **amVMMigrationLog** table looks like:

Field	Value
Description	The description of the migration event.
IVirtualMachineId	The computer ID of the VM in Asset Manager.
dtMigrationTime	The time when the migration event occurred.
SourceHostFQDN	The fully qualified domain name of the physical server that hosted the VM before the migration event.
TargetHostFQDN	The fully qualified domain name of the physical server that hosts the VM after the migration event.

Inventory data used by Oracle LMS solution

For Oracle databases, HP UD has a special content called Oracle LMS. It is certificated by Oracle for collecting Oracle database product information. That solution is different from the traditional one, the collected software information will be push to the following tables in AM:

1. amMonitoredApp – Similar to the table amSoftInstall, it stores Oracle database information collected by HP UD Oracle LMS package.

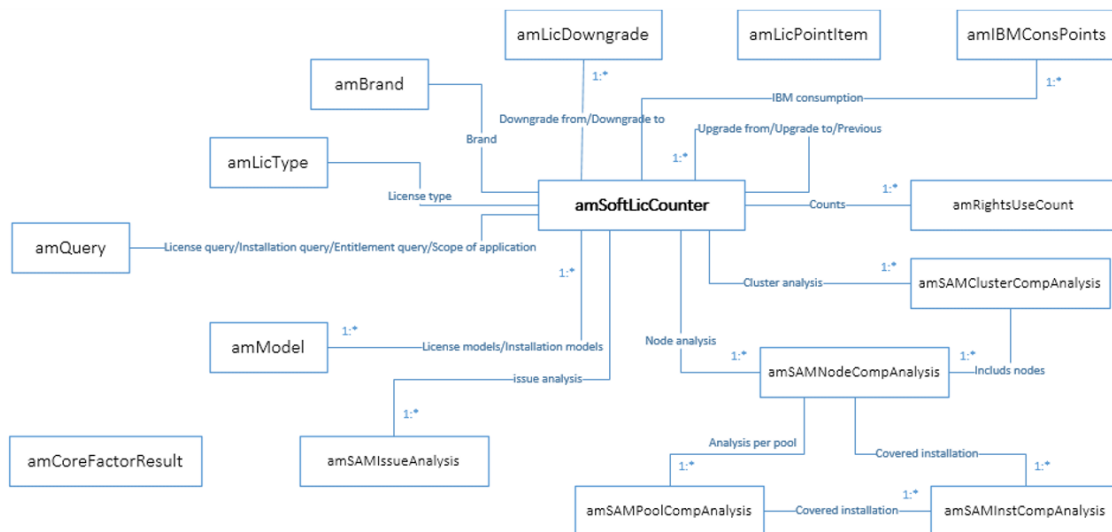
Attributes involved in calculation (SQLName)	Description	Remark
amMonitoredApp.Model	Indicate the type of the software product you have deployed	From model name AM can recognize the title/edition/version of the product

Attributes involved in calculation (SQLName)	Description	Remark
amMonitoredApp.bMissing	Indicate if the software is missing from computer	Only the installation with value 0 of this field will be taken into account during calculation
amMonitoredApp.MonParentPortfolio	The computer where the software is installed	Could be either a VM or a physical computer
amMonitoredApp.seLicenseExcluding	Indicate if the installation can be ignored for calculation	By default, the value is 0, which means it is NOT excluded. Customer can set it to 1 manually if he/she can confirm the installation does not need to be entitled.

2. amMonSWComp – It stores Oracle database option and management pack information collected by HP UD Oracle LMS package.

Attributes involved in calculation (SQLName)	Description	Remark
amMonSWComp.Model	Indicate the type of the software product you have deployed	From model name, AM can recognize the title/edition/version of the product
amMonSWComp.seStatus	The status of option/management pack	Only the installation with value 2 or 3 of this field will be taken into account during calculation
amMonSWComp.MonApp	The monitored application that the option/management pack is related to	Database option and management pack are collected along with the database product

Software counter data model



Analysis result tables for consumption details

Table: amSoftLicCounter

This table is used to save counter definition and compliance result. New field in this table:

Field name	Label	Data type	Description
bRelyOnAnalysisTable	Rely on compliance analysis result	System itemized list 0: NO 1: Yes	If check the option of Rely on compliance analysis result , the calculation script (Recalculate the rights and the installations for the selected counters (sysCoreSoftLicCountCalc)) will automatically aggregate result from compliance analysis result tables and do not ask user to set the Calculation field on the Installations/Utilizations tab. Otherwise, result is aggregated from software installation table.

Table: amSAMNodeCompAnalysis

This table is used to save license consumption result at node level. If there are different algorithms used for a node, multiple compliance analysis result records will be saved and only one of them will be aggregated to the consumption result. Take Microsoft SQL server as an example. There are two ways to count license: per individual virtual OSE or physical cores on a server. By default, the minimum value will be used.

	Field name	Label	Data type	Description
Fields				
	CalcMethod	Calculation method	Customer itemized list	It indicates the type of algorithm that is used at this time.
	IProcessorNumber	Processor count	Double	The total number of processors used by node.
	ICoreNumber	Processor core count	32 Integer	The total number of processor cores used by node.
	fFactorVal	Factor value	Float	The value of factor is based on a specific physical processor.
	TierVal	Tier value	Text	Tier value.
	dLicConsCount	Consumed points	Double	The number of consumed points.
	bCountable	Can be aggregated	System itemized list 0: NO 1: Yes	It indicates which result will be taken into account in case of several algorithms used in license count.
	Remark	Remark	Text	Detail information.
Links				
	Node	Node	1:1 link to amComputer	Node associated with the compliance analysis record.
	LicCounter	Software counter	1:1 link to amSoftLicCounter	It specifies the related counter which calculates and produces the current result.
	InstallCompAnalysis	Software installation compliance analysis	1:m link to amSAMInstCompAnalysis	Software installation compliance analysis.
	Tenant	Tenant	1:1 link to amTenant	Tenant.

Table: amSAMClusterCompAnalysis

This table is used to save the license consumption results at cluster level where multiple nodes contribute to the license consumption. If there are different algorithms used for nodes in the cluster (apply same algorithms for all node in cluster), multiple compliance analysis records will be saved and only one of them will be aggregated to the consumption result.

	Field name	Label	Data type	Description
Fields				
	CalcMethod	Calculation method	Customer itemized list	It indicates the type of algorithm that is used at this time.
	IProcessorNumber		Double	The total number of processors used in the entire cluster.
	ICoreNumber		32 Integer	The total number of processor cores used in the entire cluster.
	dLicConsCount	Consumed points	Double	The number of consumed points.
	bCountable		System itemized list 0: NO 1: Yes	It indicates which result will be taken into account in case of several algorithms used in license count.
	Remark	Remark	Text	Detail information.
Links				
	Cluster		1:1 link to amCluster	Cluster associated with the compliance analysis record.
	LicCounter		1:1 link to amSoftLicCounter	It specifies the related counter which calculates and produces the current result.

	Field name	Label	Data type	Description
	NodeCompAnalysis		1:m link to amSAMNodeCompAnalysis	Consumption analysis of cluster elements.
	Tenant		1:1 link to amTenant	Tenant.

Table: amSAMInstCompAnalysis

This table shows how the software installations on a node contribute to the final licensing consumption.

	Field name	Label	Data type	Description
Fields				
	Remark		Text	Note how the software installations contribute to the compliance result. For example, the installation will be ignored or will be covered by license, and so on.
Links				
	MonApp		1:1 link to amMonitoredApp	Monitored application that is located on the computer and calculated by the counter.
	MonSWComp		1:1 link to amMonSWComp	Monitored software component that is located on the computer and calculated by the counter.
	SoftInstall		1:1 link to amSoftInstall	Software installation that is located on the computer and calculated by the counter.

Table: amSAMPoolCompAnalysis

This table shows how the consumption is calculated for each resource pool.

Note: Currently, only IBM processor pool is supported for pool analysis.

	Field name	Label	Data type	Description
Fields				

	Field name	Label	Data type	Description
	Remark		Text	Note how the software installations contribute to the compliance result. For example, the installation will be ignored or will be covered by license, and so on.
	fLicCapacity		Double	The capacity number to be licensed.
Links				
	InstallCompAnalysis		1:n link to amSAMInstCompAnalysis	Installation analysis records that relate to the calculation on the pool.
	NodeCompAnalysis		n:1 link to amSAMNodeCompAnalysis	The node analysis record whose node hosts the resource pool.
	ResourcePool		n:1 link to amResourcePool	The resource pool being analyzed.

Table: amSAMIssueAnalysis

Pre-calculation will log issues found during the calculating for hardware and software installation. This will help to fix issues and get accurate result.

	Field name	Label	Data type	Description
Fields				
	Category		Customer itemized list	The category of the issue.
	IssueDesc		Text	Describe the details of the issues.
	bResolved		System itemized list	Flag if the issue has been resolved.
	RefObjTable		Table of field name	Table of the record that has issues when calculation licensing compliance.
	RefObjDesc		Text	Referenced object short description.
Links				

	Field name	Label	Data type	Description
	LicCounter		1:1 link to amSoftLicCounter	The counter that raises the issue.
	Product		1:1 link to amBrand	The software product that has the issue.
	RefObj		Typed link	Referenced object.
	Tenant		1:1 link to amTenant	Tenant

Consumption detail table for IBM PVU products

The **amIBMConsPoints** table stores the detailed calculation results for IBM PVU products, each record represents an aggregated calculation result of the physical computer during a given period.

SQL name	Type	Description
ICoresSubCap	Integer	The core number of the partition where the licensed program is used.
ICoresFullCap	Integer	The core number of the entire physical computer that is available to the licensed program.
IConsSubCap	Integer	PVU licensing consumption of the partition where the licensed program is used.
IConsFullCap	Integer	PVU licensing consumption of the entire physical computer that is available to the licensed program.
IValueUnit	Integer	The PVU core factor value.
Computer	Link	The physical computer which consumes the PVU.
SoftLicCounter	Link	IBM PVU counter.
LicType	Link	The license type. The license type of IBM PVU products is always "By PVU".
dtAggrStart	Date and time	The start date of the aggregation.
dtAggrEnd	Date and time	The end date of the aggregation.

Note: Within a day, there may be several PVU values for the specific product and specific physical computer. Asset Manager only stores the record of the maximum PVU calculation result for sub-capacity.

After the PVU consumption details of a physical computer are calculated, the result (*Record A*) is stored in the **amIBMConsPoints** table by following the rules below:

1. Find the existing records that are linked to this physical computer and have the same **Computer**, **LicType**, and **SoftLicCounter** values, then, select the record (*Record B*) that has the latest **dtAggrStart** date.
2. *Record A* is stored according to one of the following four scenarios.

- **Condition:**

Record B is null.

Result:

The pre-calculation workflow inserts *Record A* into the table and sets the **dtAggrStart** field and the **dtAggrEnd** field to the current date.

- **Condition:**

Record B is not null and the **ICoresSubCap**, **ICoresFullCap**, **IConsSubCap**, **IConsFullCap**, and **IValueUnit** fields of *Record A* and *Record B* are all respectively equal.

Result:

The pre-calculation workflow updates *Record B* by setting the **dtAggrEnd** field to the current date.

- **Condition:**

Record B is not null and the **ICoresSubCap**, **ICoresFullCap**, **IConsSubCap**, **IConsFullCap**, and **IValueUnit** fields of *Record A* and *Record B* are not respectively equal. The value of the **dtAggrEnd** field of *Record B* is not the current date.

Result:

The pre-calculation workflow updates the **dtAggrEnd** field of *Record B* to one day before the current date and inserts *Record A* into the table. The **dtAggrStart** field and the **dtAggrEnd** field of *Record A* are set to the current date.

- **Condition:**

Record B is not null and the **ICoresSubCap**, **ICoresFullCap**, **IConsSubCap**, **IConsFullCap**, and **IValueUnit** fields of *Record A* and *Record B* are not respectively equal. The values of the **dtAggrStart** field and **dtAggrEnd** field of *Record B* are both the current date.

Result:

The pre-calculation workflow compares the **IConsSubCap** field of *Record A* and *Record B*.

If the value of the **IConsSubCap** field of *Record B* is larger than or equal to the value of the **IConsSubCap** field of *Record A*, the pre-calculation workflow keeps *Record B* as is and does not insert *Record A* to the table.

If the **IConsSubCap** field of *Record B* has a smaller value, the pre-calculation workflow updates the **ICoresSubCap**, **ICoresFullCap**, **IConsSubCap**, **IConsFullCap**, and **IValueUnit** fields of *Record B* with the values of these fields of *Record A*.

amCoreFactorResult table

The **amCoreFactorResult** table stores the core factor information of the software vendors. It contains following attributes:

Column name	Data type	Description
IComputerId	Integer	The ID of the physical computer, which is associated with the amComputer table. This column represents a physical computer.
ILicTypeId	Integer	The license type ID, which is associated with the amLicType table.
fFactor	Double	The core factor value.
bEstimate	Integer	System itemized list. If no core factor value matches the given CPU model, an estimated core value is provided by SLO, and bEstimate is set to 1 (Yes).
bConfirm	Integer	System itemized list. This column indicates whether the core factor (fFactor) is specified by the customer. If the customer wants to use a core factor value other than the one specified in the IBM Core Factor Value table, the customer must set the value of this field to 1 (Yes). Then, Asset Manager will not calculate the core factor for this computer and will always use the core factor specified by the customer.

Health check data model

The tables and relations of the health check data model are as shown in the following figure.



Table: amHealthCheckRule

A check rule is a simple AQL query used to check data status, for example, computers are missing key information. The **IThreshold** field decides whether to show the rule result in the widget.

	Field name	Label	Data type	Description
Fields				
	Code	Code	Text	Unique code.
	Name	Name	Text	Check rule name.
	Columns	Columns	Text	The columns (in the contextual table) that will be displayed in the widget.
	Description	Description	Text	Description of the check rule.
	IThreshold	Threshold	32-bit integer	Only when the number of selected records is equal to or greater than this value, the result is displayed in the widget.
Links				
	RuleQuery	Condition	1:1 link to amQuery	The conditions of the AQL query.
	Tenant	Tenant	1:1 link to amTenant	Tenant.

Table: amHealthCheckItem

	Field name	Label	Data type	Description
Fields				
	Code	Code	Text	Unique code.
	Name	Name	Text	Check rule name.
	Description	Description	Text	Short description.
	Category	Category	Text	Category of the check item.
	bPublish	Show in widget	System itemized list 0: No 1: Yes	Determine whether the rules of the check item will be displayed in the widget. Note that if the number of selected records is less than the Threshold field of the rule, the rule will not be displayed in the widget even if you enable this option.
Links				
	CheckRules	Check rules	m:n link to amHealthCheckRule	Check rules belong to the check item.
	Tenant	Tenant	1:1 link to amTenant	Tenant.

OOB check rules

The following list shows OOB check rules that examine common data integrity and utilization.

List of check rules

Category	Check item	Check rule
Inventory data	Inventory data	Virtual machines without a host relationship
		Computers with incomplete CPU information
		Virtual machines without virtualization technology
		Software installations missing computer information
Utilization	Utilization	Computers not scanned in the past 90 days
		Software not used in the past 3 months
Software counter	Counter common check	Counters without brand or company information
		Counters (created by template) without link to license or installation models
		Counters whose installation models are not used by any software installation

Appendix 2: Developer's reference

This section introduces the functions that can be reused by the advanced users when they are creating software counters out of the scope of the SLO Best Practice package.

Functions used to generate statistics

The statistics displayed in the SLO Best Practice package dashboard are generated using functions. There are two types of statistics: software usage statistics and software assurance renewal statistics, which are generated using different functions.

Software usage statistics

Software usage statistics are generated by calling the **SAMStatistic** function from the **Sam4ComLib** script library. The following takes the VMware Workstation statistics as an example.

```
RetVal = AmEvalScript("Sam4ComLib","SAMStatistic","", "SAM4VMW_WS%" , "Code ASC"  
," License:Vmware Workstation ")
```

See Asset Manager *Programmer's Reference* guide for more information about the **AmEvalScript()** function.

Software assurance renewal statistics

Software assurance renewal statistics are generated by calling the **AmGetMaintDashboard** function from the **Sam4ComLib** script library. The following takes the VMware License Maintenance renewals statistics as an example.

```
RetVal = RetVal & AmEvalScript("Sam4ComLib","AmGetMaintDashboard","", strBarcode  
(intInit), strProduct(intInit), strColor(intInit) )
```

Appendix 3: Automation mechanism

Pre-calculation workflows	101
Software counters batch calculation	101
Dashboard generation	102
License version update according to maintenance contract	102
Automatic removal of invalid links between license models and servers	103
Clean old calculation result	103
Clean old missing installations	104
Sambp Archival	104

The automation mechanism of the SLO Best Practice package consists of several workflows. These workflows are launched periodically by Asset Manager Automated Process Manager to improve the performance of the software counters and dashboards, ensure the software application is updated to the latest version authorized by the maintenance contract and remove the invalid links caused by uninstallation or upgrade.

Caution: The SLO Best Practice workflows are disabled by default. You should manually enable them each time after installing the package.

The workflows are enabled by setting valid **Start** (dStart) and **End** (dEnd) dates.

Pre-calculation workflows

The SLO Best Practice package comes with the pre-calculation workflows in catering to their complex licensing rules. These workflows are executed periodically to count the installation points of these software; thus, ensure that the installation points are always up-to-date and you can immediately view the installation count without waiting for the calculation process to be carried out each time when requested.

For more information about the workflows, refer to "[Technical details of software counters \(Per supplier\)](#)" on page 21.

The result returned by the pre-calculation workflows will be used to generate the software compliance statistics (when you click **Calculate** to refresh the software counter result).

Software counters batch calculation

The functionality of the **Software counters batch calculation** (Reference: sysSAM4CALCULATE_WK) workflow is equivalent to clicking **Calculate** for each software counter in batch. It obtains the results returned by pre-calculation workflows and generates the software compliance statistics. It enables that:

- User can immediately view the latest software license usage statistics (without having to click the **Calculate** button for each software counter) upon opening the software counters screen.
- Dashboard can be generated efficiently based on the latest software counter results.

Dashboard generation

Dashboard displays the associated statistics, and the statistics uses the data obtained from the result returned by the **Software counters batch calculation** workflow and generates the chart with the function introduced in "[Appendix 2: Developer's reference](#)" on page 100. Therefore, dashboards can be rendered quickly upon requested.

Because the cycle time of the pre-calculation workflows and **Software counters batch calculation** workflow is one week, and the system relies on the results of these workflows to render the dashboards; therefore, if you want to view the dashboard within one week after installing the SLO Best Practice package, you have to manually execute first the three pre-calculation workflows then the **Software counters batch calculation** workflow.

License version update according to maintenance contract

This workflow is introduced in catering to the common rule for the different software publishers covered by the SLO Best Practice package: if a license is under maintenance, the user is authorized to upgrade the application covered by the maintenance contract to the latest released versions; if a license is under maintenance for a period, the license entitles the application to upgrade to the last version released before the maintenance contract end date.

This workflow checks periodically the latest application version that the purchased license (which associates with a maintenance contact) gives entitlements to, and update the license model to the latest version within the contract entitlement.

For example, if an Office Standard 2003 license was purchased in 2004 with a three years' software assurance from 2004 to 2007; and if the software assurance was not renewed for the license, this license will give entitlement to install Microsoft Office Standard 2007. Therefore, the workflow will upgrade the license to Office Standard 2007 as it is entitled by the initially purchased license. The software counters will count the licenses according to their actual entitlements and the downgrade right which can be retrieved from the **View Linked Counters** wizard, and will roll down the right in excess of the latest version to the previous one.

The historical release dates of each software title are stored as values of the corresponding itemized list in the SLO Best Practice package. These itemized lists can be identified by the records whose **Identifier** follows this pattern:

```
amSAM4<Vendor Name><Software Name><Edition>_US
```

You can customize the provided release dates with dates specific to the location where the software licenses were purchased. You can also create new itemized lists to extend this workflow to licenses that are not covered by the SLO Best Practice package.

- The proper working of the workflow requires that the **BarCode** of your license model meets certain criteria. Namely, the barcode should contain the string populated in the **Description** field of the itemized list which covers the concerned model.
If you are not using the SLO Best Practice license models, make sure your barcode meets this criteria.
You should also update the corresponding itemized list value for that model. The values are made up of two parts which are separated by colon (for example, SAM4MS_OFFICE_2007_ENT_LIC:2006/11/30). Replace the string before the colon with the actual barcode of the license model.
- Make sure you have activated the historization on the **amAsset.IModelID** field to track the license update footprint.
- This workflow only applies to software licenses whose **Assignment** is **In use** or **In stock**.
- You can choose to deactivate this workflow if you want to avoid changing your license models. However, you will need to check that the workflow is still deactivated after each new import of the package.

Automatic removal of invalid links between license models and servers

The **Automatic removal of invalid links between license models and servers** (Reference: sysSAM4REMOVELINK_WK) workflow is launched automatically to remove the link between Oracle/SQL/VMware Server license models and servers when the server has been uninstalled or perhaps replaced by another version. In the latter circumstance, user has to rebind the license models and servers according to the new installation using the **Link / unlink servers with license editions** wizard.

Clean old calculation result

Each time you click **Calculate** in the detail of the counter, a record is created in the **Rights / Utilizations count** table (amRightsUsesCount) for each line displayed in the **Results** tab of the counter detail.

This workflow periodically deletes old records (those records whose creation date is earlier than 180 days from the current date) in the **Rights / Utilizations count** table, to avoid the table becoming too large.

Tip: You can configure the days by updating the following line in the script **Clean old calculation result action**.

```
1Days = -180
```

Remember to make changes on workflow and scripts duplicated from the **Clean old calculation result** workflow to avoid having your changes overridden by future installations.

Caution: The workflow is disabled by default.

You can enable it by setting a valid **End** (SQL: dEnd) date.

Clean old missing installations

This workflow periodically deletes software installation records that have been undiscoverable by HP Discovery and Dependency Mapping Inventory on their host computers for certain days (by default it is 180 days).

Tip: You can configure the days by updating the following line in the script **Clean old missing installations action**.

```
lDays = -180
```

Remember to make changes on workflow and scripts duplicated from the **Clean old missing installations** workflow to avoid having your changes overridden by future installations.

Caution: The workflow is disabled by default.

You can enable it by setting a valid **End** (SQL: dEnd) date.

Sambp Archival

A workflow named **Sambp Archival** is used to archive all the records that have not been modified than 2 years. To enable this workflow, you need to open Asset Manager Application Designer, click the **Enable archival** option on the **Database** menu. After this option is enabled, a new connection (for example, **AMDemo94en - with archival**) is created. To run the workflow, you need to use this connection in the APM. For more information about Asset Manager archive, see the Asset Manager **Administration** guide, chapter **Archival**.

Appendix 4: How to build a cluster

Currently, the Asset Manager push adapter in Universal Discovery (UD) content package can synchronize some cluster information. However, it cannot fulfill all of above fields/links. You may maintain the cluster information manually.

Maintain cluster information manually

To manually maintain cluster information, follow these steps:

1. Create a nature with the overflow table **amCluster**.

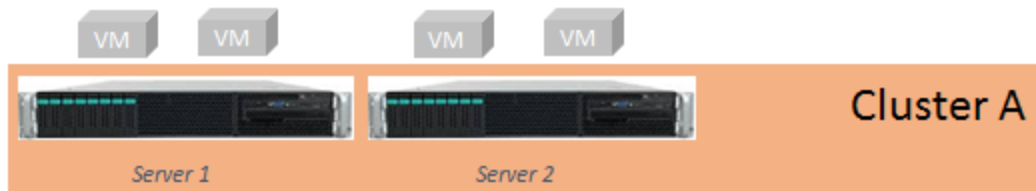
SLO Best Practice package publishes an out-of-the-box nature with the barcode **ITCLUSTER** to align with the out-of-the-box UD integration for cluster. We recommend that you use such nature for your cluster model.

2. Create a model which refers to the nature above.
3. On the navigator bar, click **Asset lifecycle > Software asset management > Cluster** to open the **Cluster** screen, click **New** to open the **Detail** panel for a new cluster, and then enter the cluster name.
4. On the **General** tab, choose the model you created.
5. On the **Cluster** tab, assign a value for the cluster layer.
6. Select the cluster type and save the settings.
7. On the **Cluster** tab, add component record in cluster component.
8. Select the computer, and assign the component role.
9. Assign value for the maximum virtual machine number if applicable.
10. Save the component record.

Cluster examples

This section shows some typical cluster types working with Asset Manager.

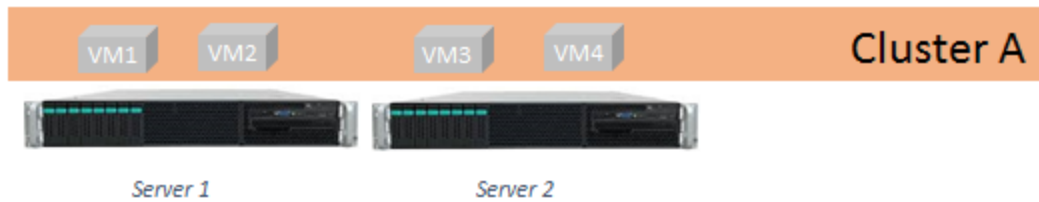
Example 1: VMware vSphere Cluster



As shown in the figure above, the cluster includes 2 physical servers, and each physical servers has 2 virtual machines running on it. The VMware vSphere cluster is an infrastructure cluster. Therefore, the cluster data in Asset Manager tables is similar to the following table.

Table	Field/Link	Value	Comment
amNature	Code	ITCLUSTER	This is the out-of-the-box nature published by SLO Best Practice package.
amModel	Name	VMWare vSphere Cluster	
	Nature.Code	ITCLUSTER	
amCluster	Name	Cluster A	
	Model.Name	VMWare vSphere Cluster	
	Assignment	In use	
	Cluster layer	Infrastructure Clustering	
	Cluster type	Failover	
amClusterComponent (for server 1)	Computer.Name	Server 1	
	Component role	Owner	
	IsActive	Yes	
amClusterComponent (for server 2)	Computer.Name	Server 2	
	Component role	Owner	
	IsActive	Yes	

Example 2: Microsoft Windows Failover Cluster



As shown in the figure above, there are 2 physical servers, and each physical server has 2 virtual machines running on it. The total 4 virtual machines form a Windows Failover Cluster. This is an OS layer cluster. There, the cluster data in Asset Manager tables is similar to the following table.

Table	Field/Link	Value	Comment
amNature	Code	ITCLUSTER	This is the out-of-the-box nature published by SLO Best Practice package.
amModel	Name	Windows Failover Cluster	
	Nature.Code	ITCLUSTER	
amCluster	Name	Cluster A	
	Model.Name	Windows Failover Cluster	
	Assignment	In use	
	Cluster layer	OS Clustering	
	Cluster type	Failover	
amClusterComponent (for VM1)	Computer.Name	VM1	
	Component role	Owner	
	IsActive	Yes	
amClusterComponent (for VM2)	Computer.Name	VM2	
	Component role	Owner	
	IsActive	Yes	

Table	Field/Link	Value	Comment
amClusterComponent (for VM3)	Computer.Name	VM3	
	Component role	Standby	
	IsActive	Yes	
amClusterComponent (for VM4)	Computer.Name	VM4	
	Component role	Standby	
	IsActive	Yes	

Example 3: Microsoft SQL Server Mirroring Cluster



As shown in the figure above, there are 2 physical servers, and each physical server has 2 virtual machines running on it. On VM1 and VM3, you have SQL Server 2012 Enterprise edition installed, and they are set up as a SQL Server mirroring cluster. This is an application layer cluster. Therefore, the cluster data in Asset Manager tables is similar to the following table.

Table	Field/Link	Value	Comment
amNature	Code	ITCLUSTER	This is the out-of-the-box nature published by SLO Best Practice package.
amModel	Name	SQL Server Cluster	
	Nature.Code	ITCLUSTER	
amCluster	Name	Cluster A	
	Model.Name	SQL Server Cluster	
	Assignment	In use	
	Cluster layer	Application Clustering	
	Cluster type	Mirroring	

Table	Field/Link	Value	Comment
amClusterComponent (for instance 1)	SoftInstall	SQL 2012 ENT 1	
	Component role	Owner	
	IsActive	Yes	
amClusterComponent (for instance 2)	SoftInstall	SQL 2012 ENT 2	
	Component role	Standby	
	IsActive	Yes	

We appreciate your feedback!

If you have comments about this document, you can [contact the documentation team](#) by email. If an email client is configured on this system, click the link above and an email window opens with the following information in the subject line:

Feedback on Technical Guide (Asset Manager SLO Best Practice package 9.5.12.3)

Just add your feedback to the email and click send.

If no email client is available, copy the information above to a new message in a web mail client, and send your feedback to ovdoc-itsm@hp.com.