Libbard

Smart Plugin for HP Service Manager (User Manual)
Version: 1.5.0
Date: April 22, 2014

Table of Contents

Smart Plugin for HP Service Manager

The Smart Plugin for HP Service Manager (SMSPI) is an add on component for HP Operations that allows advanced monitoring of HP Service Manager (SM) without developing custom policies or scripts. An average installation takes less than an hour and requires minimal tuning.

Overview of SMSPI Functionality

- Automatically discovers vertically scaled installations of HP Service Manager

- Monitors for common log-file, process failures, and availability

- Alerts and reports on application specific metrics:

    ◦ User load per SM process in addition to composite usage (example, Total System Usage)

    ◦ Advanced application memory alarms that cause degraded performance for SM users

- Dynamic discovery and automatic monitoring of scaling changes.

- SCAuto and Load Balancer monitoring

- Outage control for on a per process basis (See outage control)

System Requirements

HP Operations Management Server

OS: Windows 2003, Windows 2008, Windows 2008R2

HP Operations for Windows: 9.0+

HP Service Manager for Windows

OS: Windows 2003, Windows 2008, Windows 2008R2

HP Service Manager: 9.20 +

If your business is running another mixture of MS Windows and HP Service Manager that is not listed above, please contact us at support@libbard.com *to determine if the SM SPI will function in your environment.*

Installation

HP OMW Console Server

1. Download the *SmSpi_Installer.msi* from www.libbard.com/smspi and put it somewhere on the HP Operations server (preferably *on a local drive*)

2. Click to accept the license terms and click on "Install"

3. A series of programs will execute and the next screen will follow when installation completes
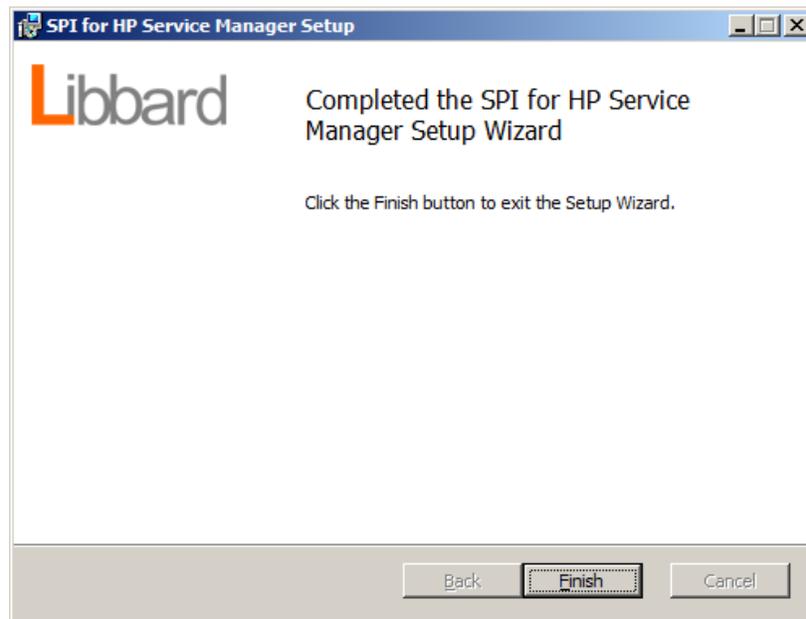
4. Confirm that the SMSPI has been installed by opening the OMW console and verifying the new policies are on the system. (Policy management → Policy groups → SPI for Service Manager)
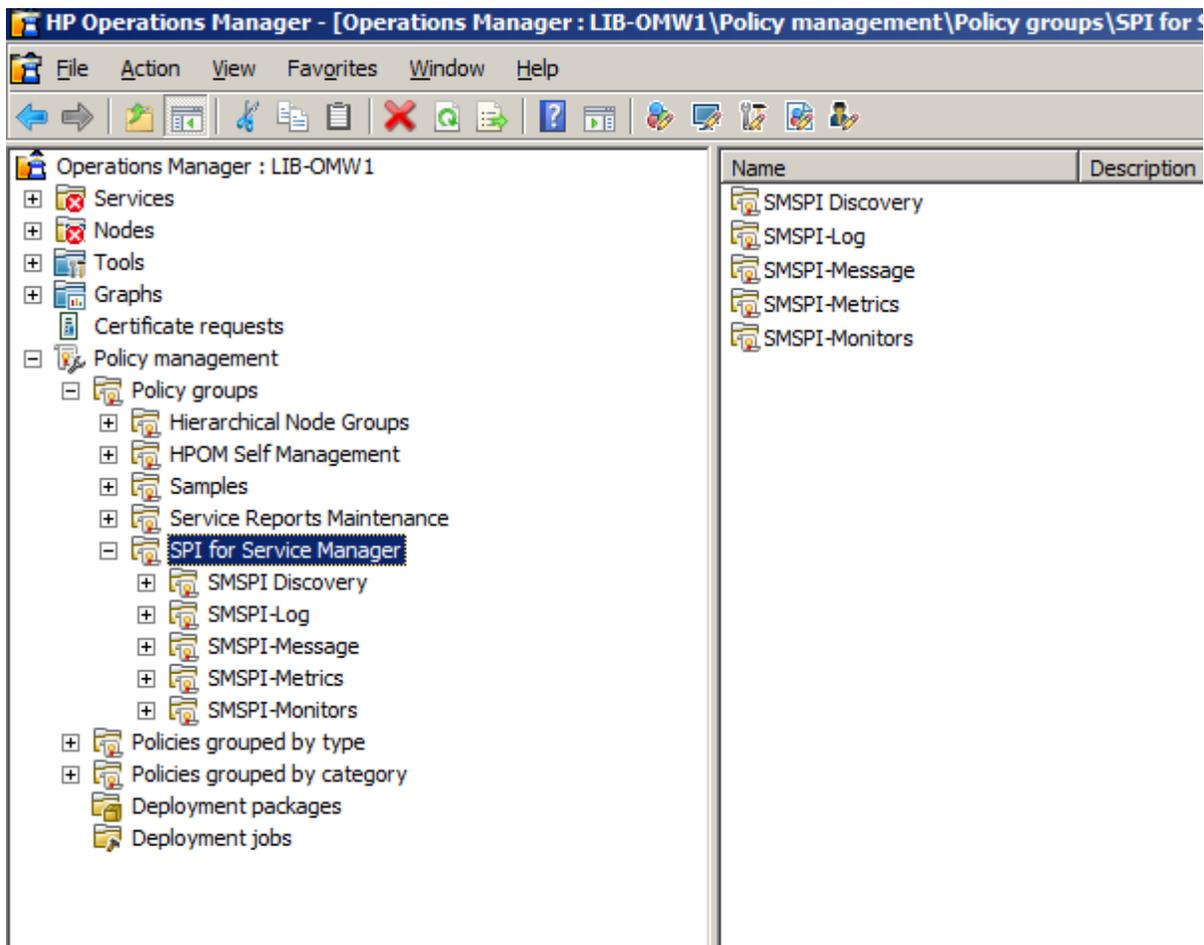
HP Service Manager 9.2+ Server

1. Make a backup copy of your *sm.cfg* and *sm.ini* file on each SM application server that will host the SMSPI. These are contained with the "RUN" directory of Service Manager

    Example Location: C:\Program Files (x86)\HP\Service Manager 9.30\Server\RUN

2. Open the sm.ini and add the following JVM Options ("**JVMOption1**" is skipped intentionally)

    JVMOption0:-Dcom.sun.management.jmxremote

    JVMOption2:-Dcom.sun.management.jmxremote.ssl=false

    JVMOption3:-Dcom.sun.management.jmxremote.authenticate=false

    Warning: If you have already added JVMOptions to the sm.ini file please make sure the numbers do not overlap. For example, if the last JVMOption used before the SMSPI was *JVMOption4* the resulting *sm.ini* file will look like this:

    JVMOption5:-Dcom.sun.management.jmxremote

    JVMOption7:-Dcom.sun.management.jmxremote.ssl=false

JVMOption8:-Dcom.sun.management.jmxremote.authenticate=false

3. The final *sm.ini* configuration file should look something like this when complete:

```
# ServiceManager Initialization file
# (c) Copyright 2008 Hewlett-Packard Development Company, L.P.
# Configuration Modified Date:2/3/14 8:27 PM
shared_memory:32000000
JVMOption0:-Dcom.sun.management.jmxremote
JVMOption2:-Dcom.sun.management.jmxremote.ssl=false
JVMOption3:-Dcom.sun.management.jmxremote.authenticate=false



log:../logs/sm.log
system:13080
httpPort:13080
sslConnector:1
httpsPort:13443
sqldictionary:sqlserver
[sqlserver]
sqldb:SM_32
sqllogin:smadmin/Password123
plugin0:kmplugin.dll
```

4. Save the sm.ini file and open the sm.cfg file in either MS Notepad or another text editor.

5. For each SM Engine (lines that begin with "sm"), a single port for each engine will need to be defined. The particular port number choice is not important, but it must not be used by another process on the system. The only different with each entry is the port number. Feel free to copy/paste the red section from the example below.

sm -loadBalancer -httpPort:13080 **-JVMOption1:-Dcom.sun.management.jmxremote.port=9004**

sm -httpPort:13081 -httpsPort:13082 **-JVMOption1:-Dcom.sun.management.jmxremote.port=9005**

sm -httpPort:13083 -httpsPort:13086 **-JVMOption1:-Dcom.sun.management.jmxremote.port=9006**

sm -scautolistener:12690 **-JVMOption1:-Dcom.sun.management.jmxremote.port=9007**

6. An example of a fully configured sm.ini file is shown below for reference.

```
#
# HP Service Manager Server Configuration File
#
# Used by HP Service Manager service on Windows and smstart script on Unix
# to start the Service Manager server processes.
#
#######################################################################################
#
# Copyright 1994-2010 Hewlett-Packard Development Company, L.P.
# All Rights Reserved
#
#######################################################################################

#
# start a Service Manager listener
#
sm -loadBalancer -httpPort:13080  -JVMOption1:-Dcom.sun.management.jmxremote.port=9004
sm -httpPort:13081 -httpsPort:13082 -JVMOption1:-Dcom.sun.management.jmxremote.port=9005
sm -httpPort:13083 -httpsPort:13086 -JVMOption1:-Dcom.sun.management.jmxremote.port=9006
sm -scautolistener:12690 -JVMOption1:-Dcom.sun.management.jmxremote.port=9007



# When unable to start the server as a service
# uncomment the following line to produce an output
# file which may contain some information from the
# Java JVM.
#cmd /c "sm" > C:\\process_stdout 2>&1
```

7. Restart HP Service Manager and check the sm.log to ensure that no problems have been introduced.

Deploy Policies and Instrumentation

1. In the OMW console, right click on the SM application server and enter the "All Tasks → Deploy Instrumentation" section.

2. Select the *ServiceManager* Instrumentation and select "OK"
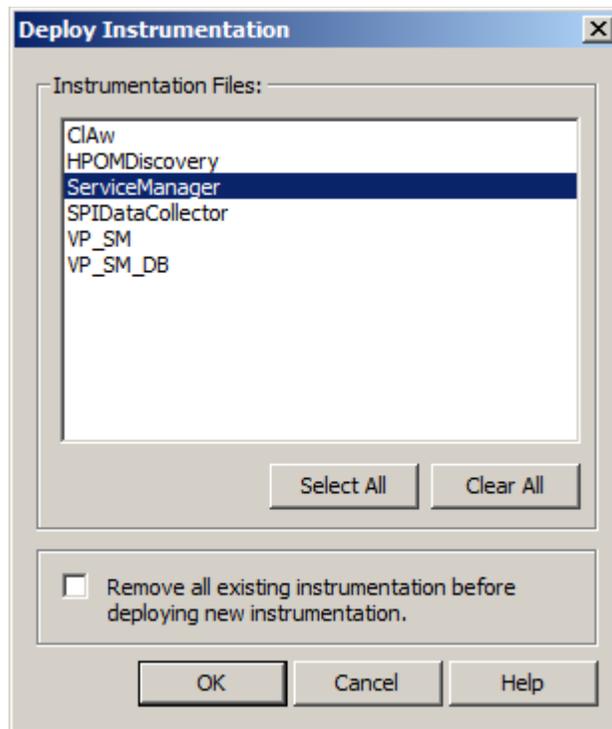
3. After the instrumentation has been successfully deployed, navigate to "Policy groups->SPI for Service Manager" in OMW console. Right click and use the "Deploy on.." option.
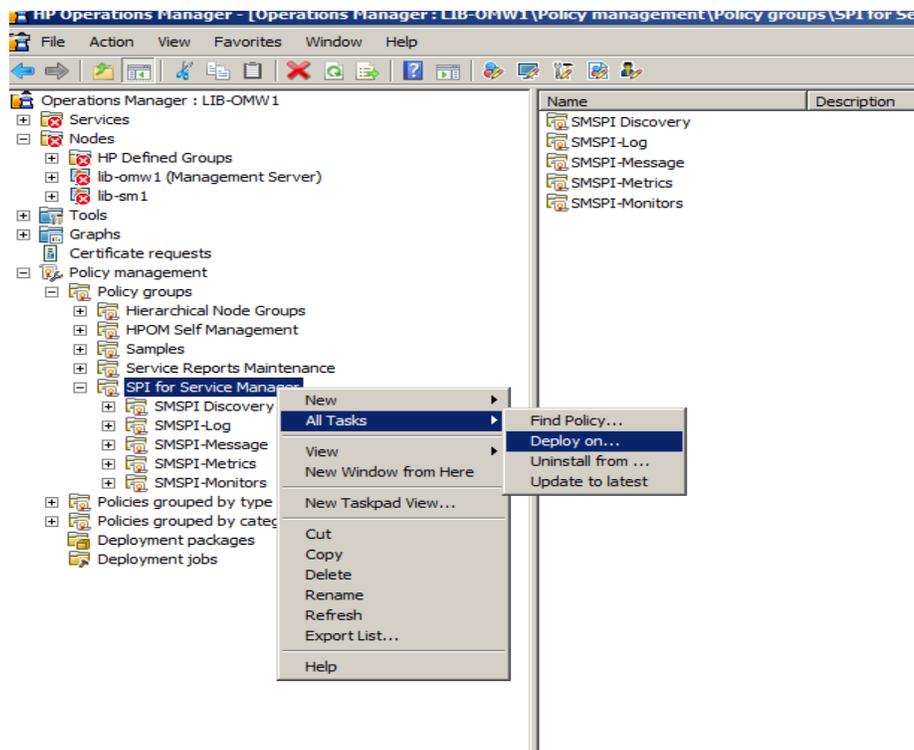
4. Select the node you want to apply the SM SPI and click "OK"
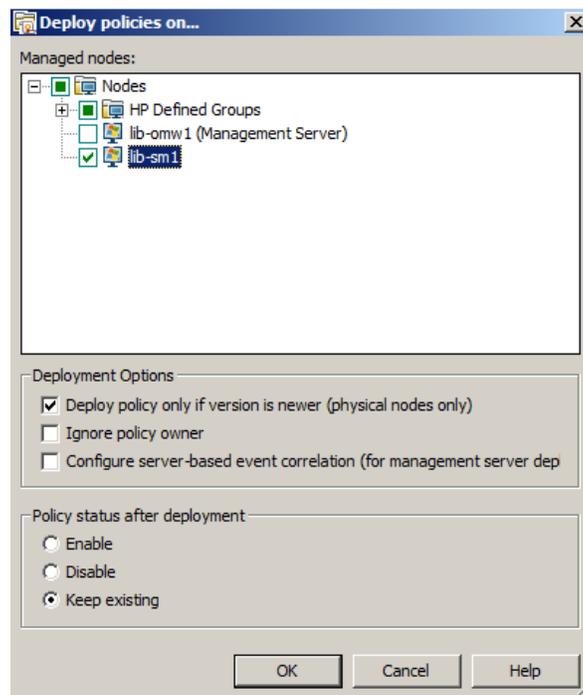
5. Two normal messages will appear in the OMW message console indicating that the deployment has been successful. They will appear like the defined messages below:

Node: <TargetNodeName> Message group: SM_SPI_Config Application: SM SPI Object: SMSPI_Discovery Severity: Normal Text: **SMSPI has started discovery**	Node: <TargetNodeName> Message group: SM_SPI_Config Application: SM SPI Object: SMSPI_Discovery Severity: Normal Text: **SMSPI has successfully finished discovery. See annotations for engine details**

6. Installation is now complete. If there are any errors encountered during the installation process they will be presented in the OMW console linked to the node the SMSPI was deployed on.

Policies and Metrics

SM-SPI-0001 (Up/Down Engine Status)

Description: For each standard NON-SSL engine port that is configured the SM SPI will generate an alert when the port becomes unavailable. When this error is received this means that the specified HTTP port is not responding and users are most likely being impacted.

Metric Type: Boolean (True or False)

SM-SPI-0002 (HTTPS/SSL Up/Down Engine Status)

Description: For each of the SSL/HTTPS ports that are configured in the *sm.cfg* file, an alert will be generated when that port becomes unavailable. When this error is received this indicates that the specified SSL port is not responding and users are most likely being impacted. If SSL is not configured within the *sm.cfg* file than this alert will never be triggered.

Metric Type: Boolean (True or False)

SM-SPI-0003 (Load Balancer Up/Down Engine Status)

Description: For each of the SSL/HTTPS ports that are configured in the *sm.ini* file, an alert will be generated when that port becomes unavailable. When this error is received this indicates that the specified SSL port is not responding and users are most likely being impacted. If the loadbalancer is not configured within the *sm.cfg* file than this alert will never be triggered.

Metric Type: Boolean (True or False)

SM-SPI-0004 (Maximum User Engine Load by Percentage)

Description: Each engine can host a number of users (default max of 50), but this setting can vary by customer and the SMSPI dynamically determines this setting for each SM engine that is configured. Once an SM app engine starts to reach maximum capacity, based on **percentage** values, this alert will be triggered.   The settings can be altered/added in the policy based on the needs of the customer. The default settings are:

Critical > 99%

Major > 97%

Metric Type (Percentage): Integer (0 to 100)

SM-SPI-0005 (Maximum User Engine Load by Remaining Threads)

Description: Each engine can assigned one thread to each user (default max of 50), but this setting can vary by customer and the SMSPI auto resolves the setting for each SM engine. Once an SM application engine utilizes its available threads then users will start being rejected.   The settings can be altered/added in the policy based on the needs of the user. The delivered default settings are:

Critical <   5 Remaining Threads (Example: Engine can handle 5 additional users before failing)

Major   > 10 Remaining Threads

Metric Type (Threads): Integer (Max Engine Capacity to 0)

SM-SPI-0006 (Maximum TOTAL User Engine Load by Remaining Threads)

Description: With vertically/horizontally scaled installations of HP Service Manager it is common to have multiple *sm.exe* processes running simultaneously. This metric takes the consolidated load of ALL the engines that are currently running and determine the total remaining threads available. If this alert is generated the SM administrator should consider re-scaling and adding additional SM engines.

Critical <   0 Remaining Threads (Users will be denied, entire system is at full capacity)

Major   < 10 Remaining Threads for entire system

Warning < 20 Remaining Threads for entire system

Metric Type (Threads): Integer (Total Server Capacity to 0)

SM-SPI-0007 (Maximum TOTAL User Engine Load by Percentage)

Description: With vertically/horizontally scaled installations of HP Service Manager it is common to have multiple *sm.exe* processes running simultaneously. This metric takes the consolidated load of ALL the engines that are currently running and determine the total percentage consumed.

Critical =   100% (Users will be denied, entire system is at full capacity)

Major   > 95% Total System Consumption

Warning > 92% Total System Consumption

Metric Type (Percentage): Integer (0 to 100)

SM-SPI-0010 (Engine Heap Usage in Megabytes)

Description: Since Service Manager is mainly a Java VM the SMSPI extracts the current JVM HEAP usage in MB and determines if it is currently running out of memory or if the value is high enough to impact user experience. Based upon past results, we have seen users impacted by SM engines that have reached the predefined thresholds. Reasons for reaching such high values can be attributed to many unique issues. But most likely it is due to either a tailoring problem or how the installation was designed and configured.

Critical > 800MB (Individual Memory Heap)

Major > 700MB (Individual Memory Heap)

Minor > 400MB (Individual Memory Heap)

Metric Type (Megabytes): Integer (0 to Max Memory)

Note: More information can be found on JVM Memory Management via Google

SM-SPI-0011 (Engine NON-Heap Usage in Megabytes)

Description: The SMSPI extracts the current JVM Non-HEAP usage in MB and determines if it is currently running out of memory or if the value is high enough to impact user experience. When Service Manager does not release memory properly the result is gradual performance degredation. The reason for reaching such high values can be attributed to many unique things that can occur. If this alert becomes frequent something is wrong with the programming within Service Manager or you are running a version containing "leaked loaders".

Critical > 800MB (Individual Memory Heap)

Major > 700MB (Individual Memory Heap)

Minor > 400MB (Individual Memory Heap)

Metric Type (Megabytes): Integer (0 to Max Memory)

Note: More information can be found on JVM Memory Management via Google

> ### SM-SPI-0020 (SC Auto Listener)
>
> **Description:** If a SCAuto listener is configured within the *sm.cfg* file this alert will be generated when the listener port is non-responsive.   The SMSPI currently only monitors one port per defined SCAuto listener.
>
> **Metric Type: Boolean (True or False)**

Tools

Within the OMW console a new tool-set will be installed that can be used to manage the SMSPI and its configurations. While the SMSPI is designed to require minimal tailoring after installation, some tools have been provided to allow the product to provide more flexibility. Some of the things that can be done with the SMSPI tools are:

- Add and Remove certain Service Manager processes from monitoring (aka *Engines)*
- Display the status of a vertically or horizontally scaled installation of Service Manager. This is especially useful for operations personnel that do not have OS credentials to SM Servers
- Install purchased licenses (*See Installation Instructions on License Invoice)*

Status Tool

The status tool will display all the discovered Service Manager engines on a particular server. To run simply right-click->Launch and select an active Service Manager application server. Below is an example of vertically scaled installation of Service Manager.

```
Tool Status                                                                       _ □ X

Launched Tools:
Status       Action          Start/Finish Time    Node           Command
No Result    SM SPI Status   04/22/14 17:48...    lib-sm1        SM_Controller.exe --spi_args -status


Tool Output:
                    Service Manager Smart Plugin Status
                          Libbard Software(2014)
          _____

                        Load Balancer Section
          ---------------------------------------------------------------
ENGINE NAME      JMX PORT            HTTP PORT           HTTPS PORT
LB1              9004(ONLINE)        13080(ONLINE)       0(UNDEFINED)
                        Standard Engine Section
          ---------------------------------------------------------------
ENGINE NAME      JMX PORT            HTTP PORT           HTTPS PORT
Engine1          9005(ONLINE)        13081(ONLINE)       13082(ONLINE)
Engine2          9006(ONLINE)        13083(ONLINE)       13086(ONLINE)
                       SC Auto (Listener) Section
          ---------------------------------------------------------------
ENGINE NAME      JMX PORT            LISTENER PORT
Scauto1          9007(ONLINE)        12690(ONLINE)


    Save                                              Rerun    Close    Help
```

Ignore Engine

If a particular engine does not require monitoring use the "Ignore Engine" tool in the OMW console. Simply enter the engine name as it appears in the status screen in the *Parameters* section. The example below adds "Engine2" to the ignore list:

```
Edit Parameters                                       X

Parameters

    [icon]        Ignore Engine

Tool Description:
Enter the SM SPI engine name to add it to the ignore list. No messages or
metrics will come from the specified engine. Enter SM engine name (as it
appears in status or messages) into the parameters section of the tool to

Command:
SM_Controller --spi_args -ignore_add

Parameters:
Engine2



                           Launch...    Cancel    Help
```
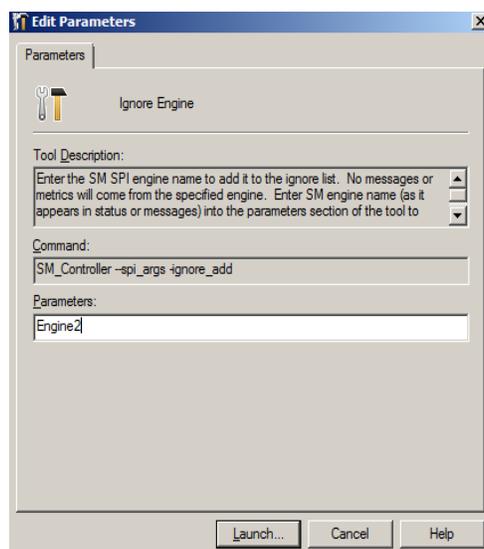
After the engine has had its monitoring disabled an alert will appear in the console indicating that monitoring is no longer active for that engine.

Note: Do not add more than one engine at a time to the parameters section.

Remove Engine from Ignore List

After an engine is ignored it can be re-activated by applying this alternate tool. It works the same as the *Ignore Engine* (see above). In the parameters section enter the engine's name in the *Parameter* section of the tool and click "Launch".   Once the engine has been processed a normal event will appear in the console indicating that the engine is now being monitored.

Add License

Instructions for this tool will be provided with a valid purchase agreement. It is not necessary to use this tool for the 20 day trial period.

Graphs and Performance Data

Since all SMSPI performance data is contained within the OM Performance Agent it can be accessed in a variety of ways and used for troubleshooting or scaling assessments. The most common method is using HP Performance Manager to compare SMSPI data with other fundamental OS metrics to determine where and when certain problems are occurring. To dynamically graph SMSPI performance data an active version of HP Performance Manager must be installed in your environment; preferably on the same server as OMW.

The SMSPI provides graphing functionality for all the metrics that are listed Policy and Metrics section of this document.

Available Performance Metrics

Table of SMSPI Specific Metrics

Metric Name		Description
EngineUserLoadRemainingThreads		The remaining threads (user slots) for a particular SM engine.
EngineUserLoadPercent		Percentage of engine consumption based on active users
TotalSystemRemainingThreads		Remaining threads for ALL engines on a particular server
TotalSystemUserLoadPercentage		Percentage of user threads (user slots) that are consumed on the entire system
EngineHeapInMegaBytes		Engine's consumed JVM heap size
EngineNonHeapMegaBytes		Engine's consumed JVM NON-HEAP size
LoadBalancerUptime		Availability of SM load balancer
SslUptime		Availability of SSL SM engine HTTPS ports
NonSslUptime		Availability of standard SM engine HTTP ports
ScAutoUptime		Availability of the SCAuto engine (aka, SM Listener)

Create Graph Using Design Wizard

1. Select the Service Manager application server in the left column and then "Graphs->Design a Graph"
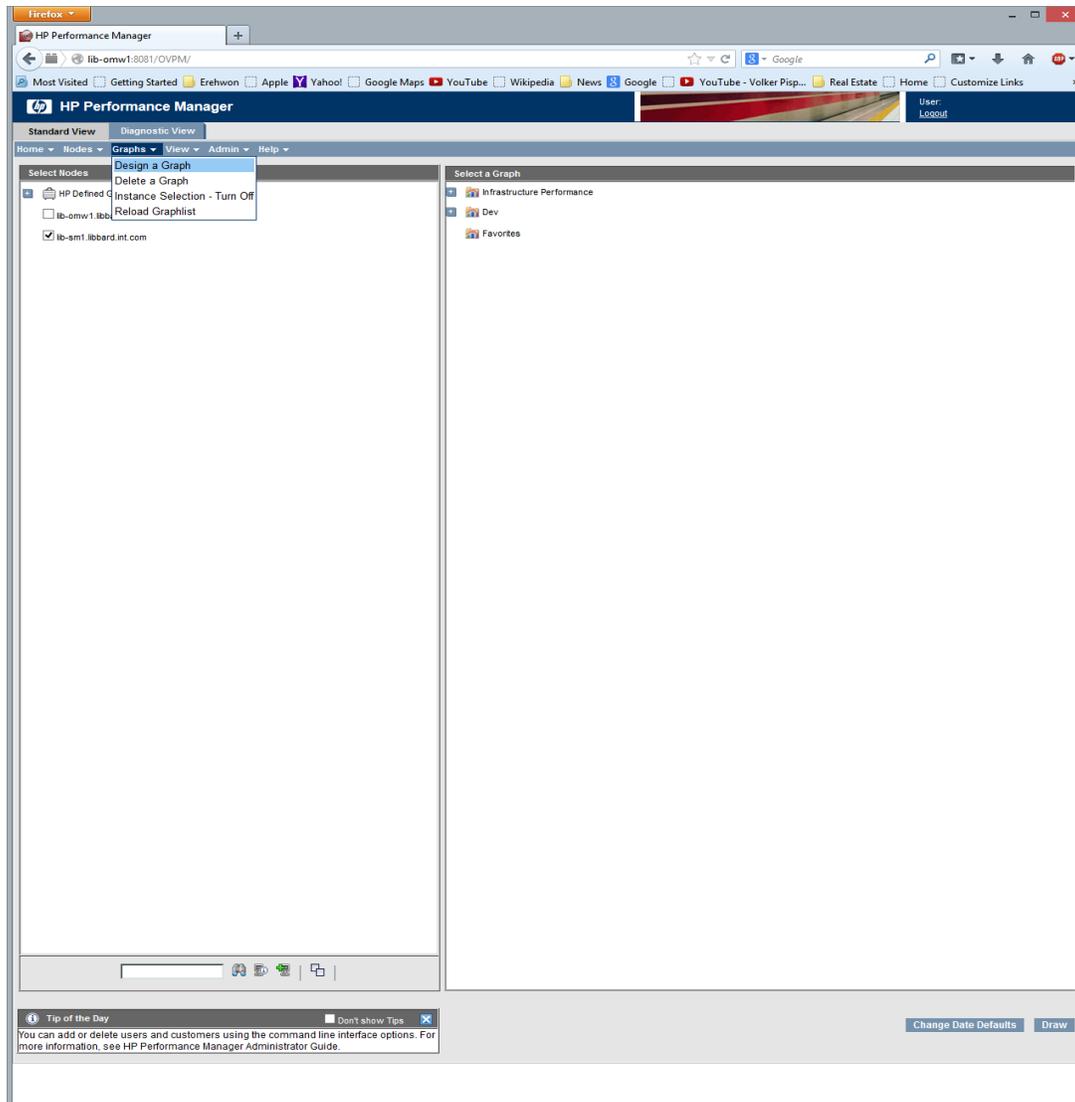
2. Select "Design a new Graph"
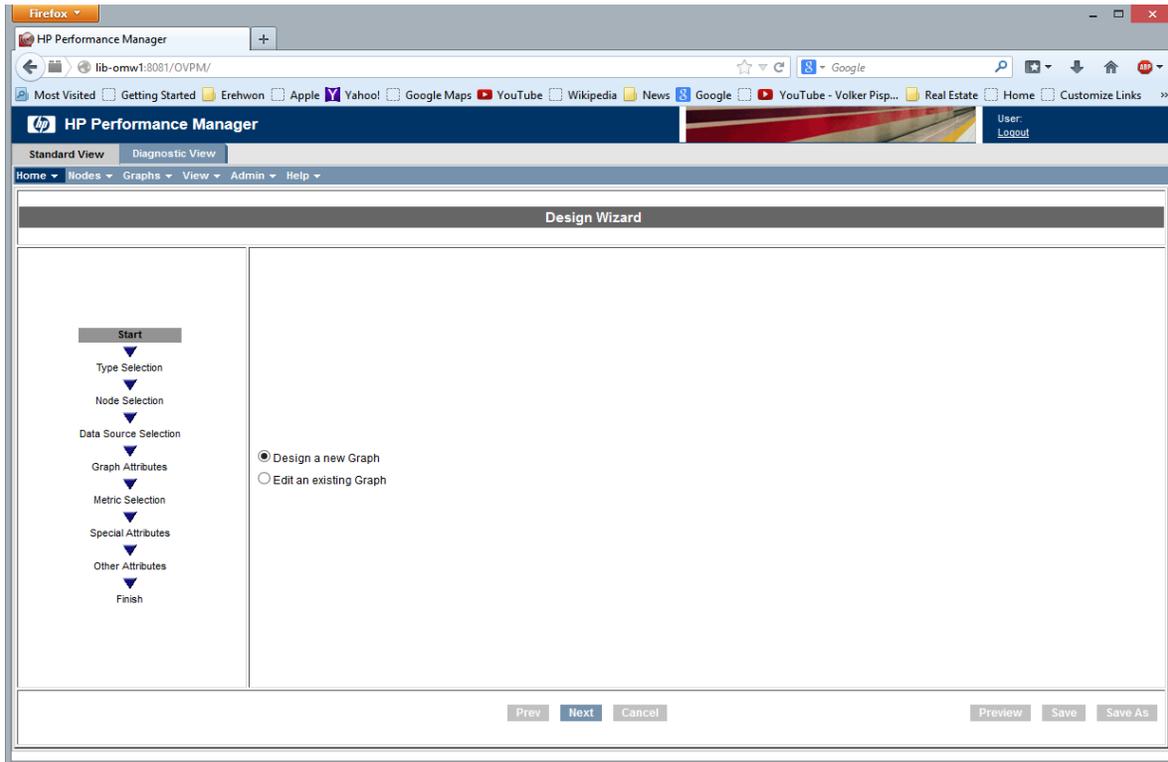
3. On the next screen, select "Normal Graph"

4. When selecting the node, select only the SM Application server. For this example, *lib-sm1.libbard.int.com* is our Service Manager application server containing SMSPI data.
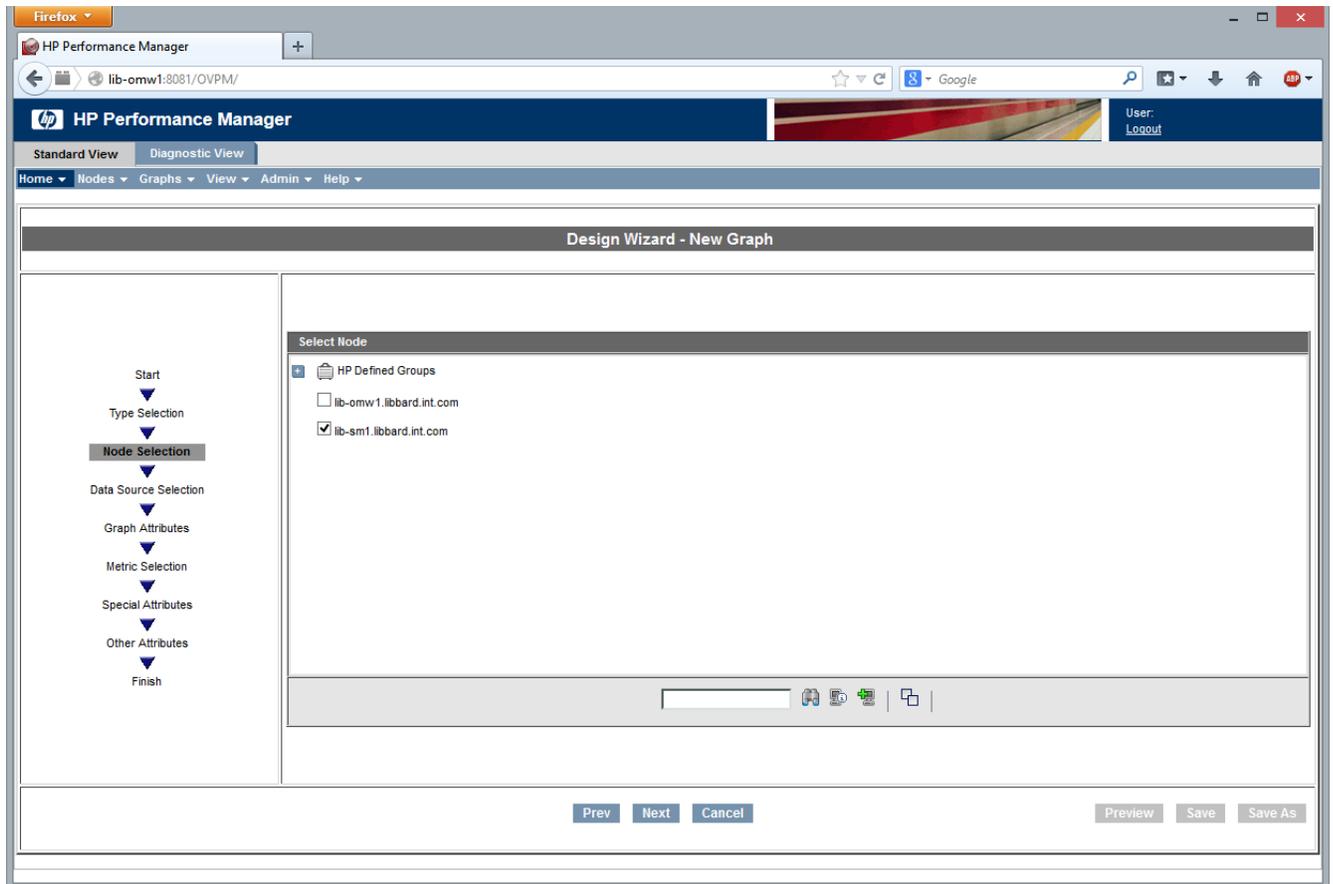
5. Select "PA(Performance Agent)" on the following screen.

6. The screen where you select the time range is based on the user needs. If the SMSPI was just installed you will not have summarized data for at least twenty minutes. For this example we will leave it set to its defaults.
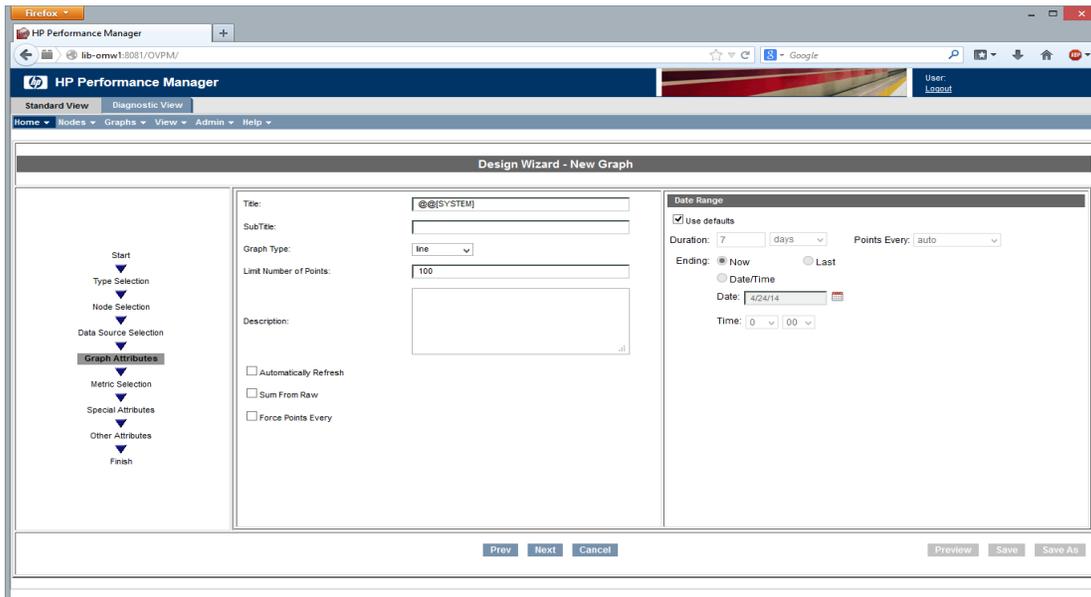
7. The next screen will allow the user to choose specific metrics for graphing. You may select as many metrics as you like, but for this example we will only be using SMSPI metrics. After you click the "Add Metrics" button you will see the following window:
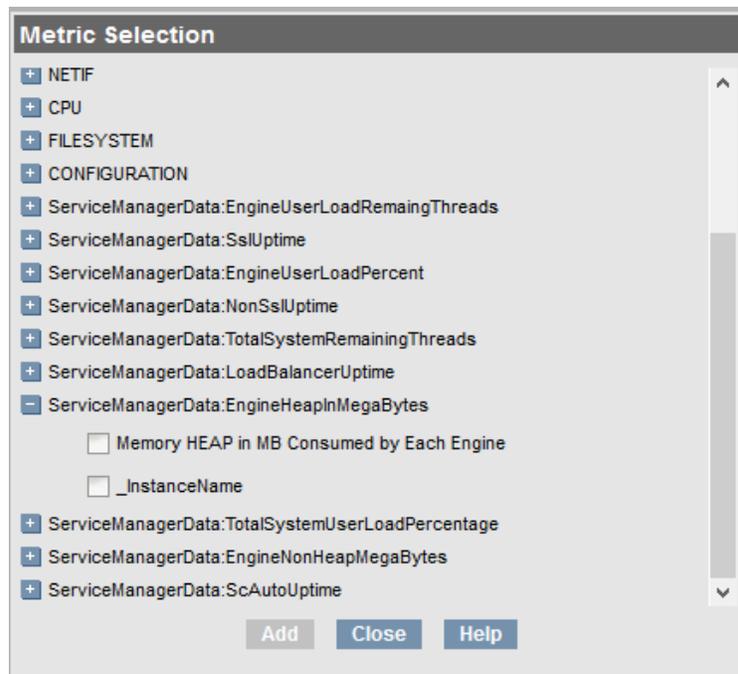
8. Select the metric that will be displayed in the graph. For this example we will display the "*ServiceManagerData:EngineHeapInMegaBytes*"data to see the fluctuation in memory usage. **Note:** It is not necessary to select the "_InstanceName"

9. Click the "Preview" button and you will see a screen allowing you to select which SM instances that you would like to see. For this example we will select engine 1 and 2 for graphing.
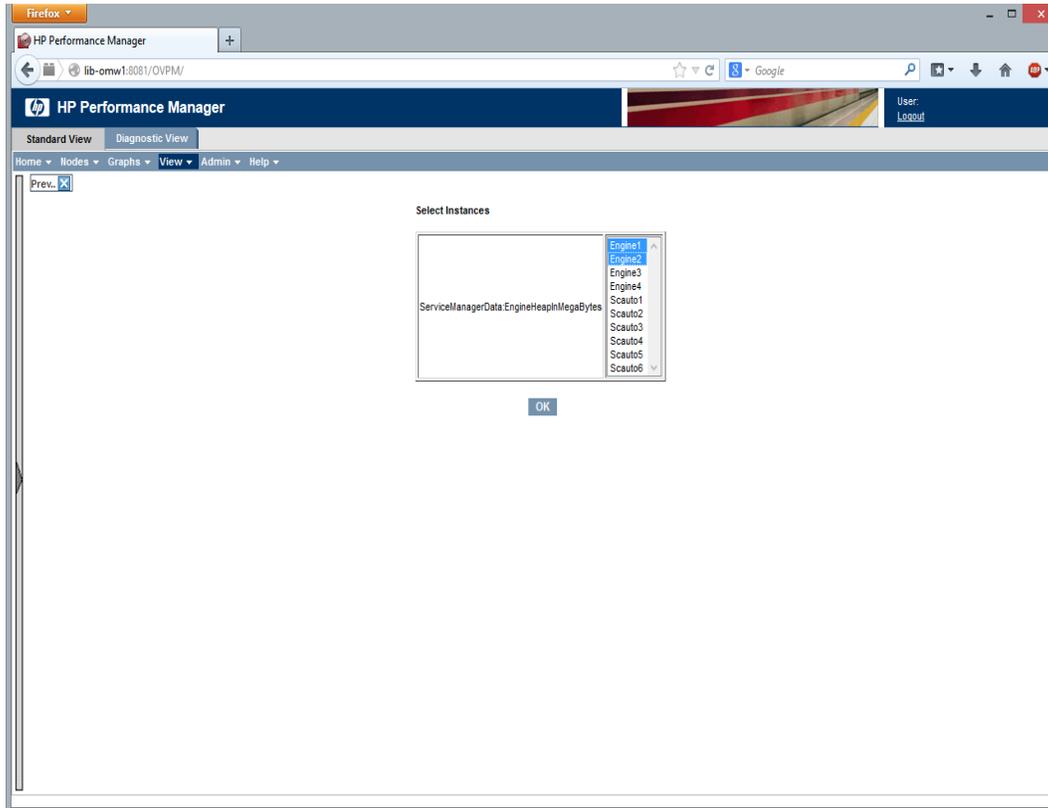
10. After clicking "OK" two lines will be displayed showing each engine's memory consumption:
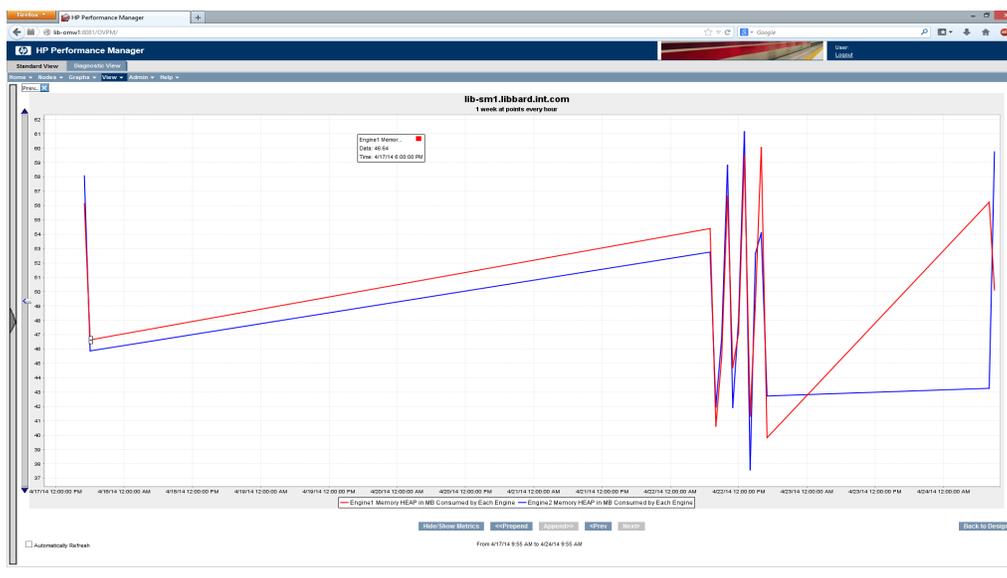
11. Click "Back to Design" to finish saving your graph for future use. The next two screens in Performance Manager (*Special Attributes* and *Other Attributes*) allow you to change the cosmetic features of the graphs for other users.

Troubleshooting Graphs

If blank graphs are appearing after the SMSPI has been installed or the metrics are not visible it might be necessary to restart the Operations Agent installed on the Service Manager servers (Agent Restart Command : *opcagt -kill & opcagt -start)*. After restarting the offending agents wait for at least ten minutes to allow the Performance Agent data to populate its tables. If problems persist please contact technical support at: support@libbard.com

Customizing

The SMSPI is primarily customized through its policies so administrators are not required to modify scripts and alter complex configuration files. Realizing that each organization has different requirements, there are number of features that have been designed into the SMSPI to make customization easier.

Changing Alert Thresholds

Most of the SMSPI's core functionality uses the standard *Measurement Threshold* policy type. They use basic numeric comparisons that can be altered by changing the predefined values within the policy.

Disable Thresholds

If one of the SMSPI metrics is not desired than the policy can be removed from OMW or it can be disabled on a particular server using the following OMW menu option:
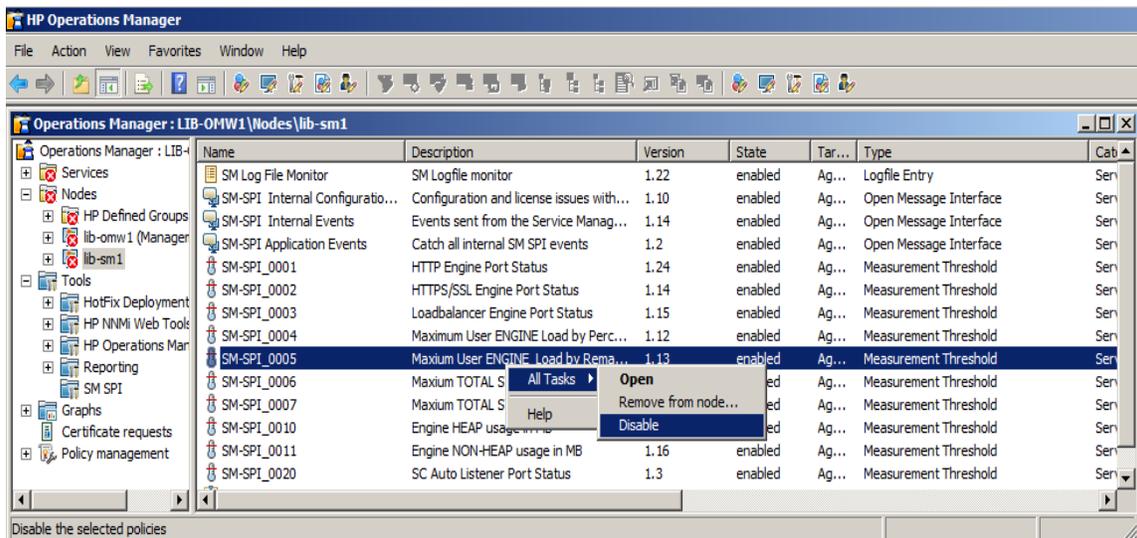
Since all threshold monitors are externally populated, it is not necessary to make adjustments anywhere else within OMW. If the policy is removed or disabled the performance metrics will not be visible in HP Performance Manager. To keep the metrics available in the Performance Manager it is advisable to adjust the thresholds to a different level or suppress the outputted message within the policy.

Changing Threshold Policy Names

Normally changing the policy names on a prepackaged *Smart Plugin* will have some undesired results. Usually customers want to discern modified policies from the set that were provided by the vendor. We actually recommend this technique so that patches and upgrades can be better managed. For example, if the customer changed ALL the SMSPI measurement policies to the following format:

MyLLC-**SM-SPI_00##**

The SMSPI collector would continue sending the monitors to original format (SM-SPI_00##) and events would be lost.

This is easily remedied by modifying the main collection policy (*SM-SPI-StatusPoller)* and adding the prefix definition to the command (*-polprefix <Policy Prefix>).* An example is given on the next page for the *MyLLC* example given above.
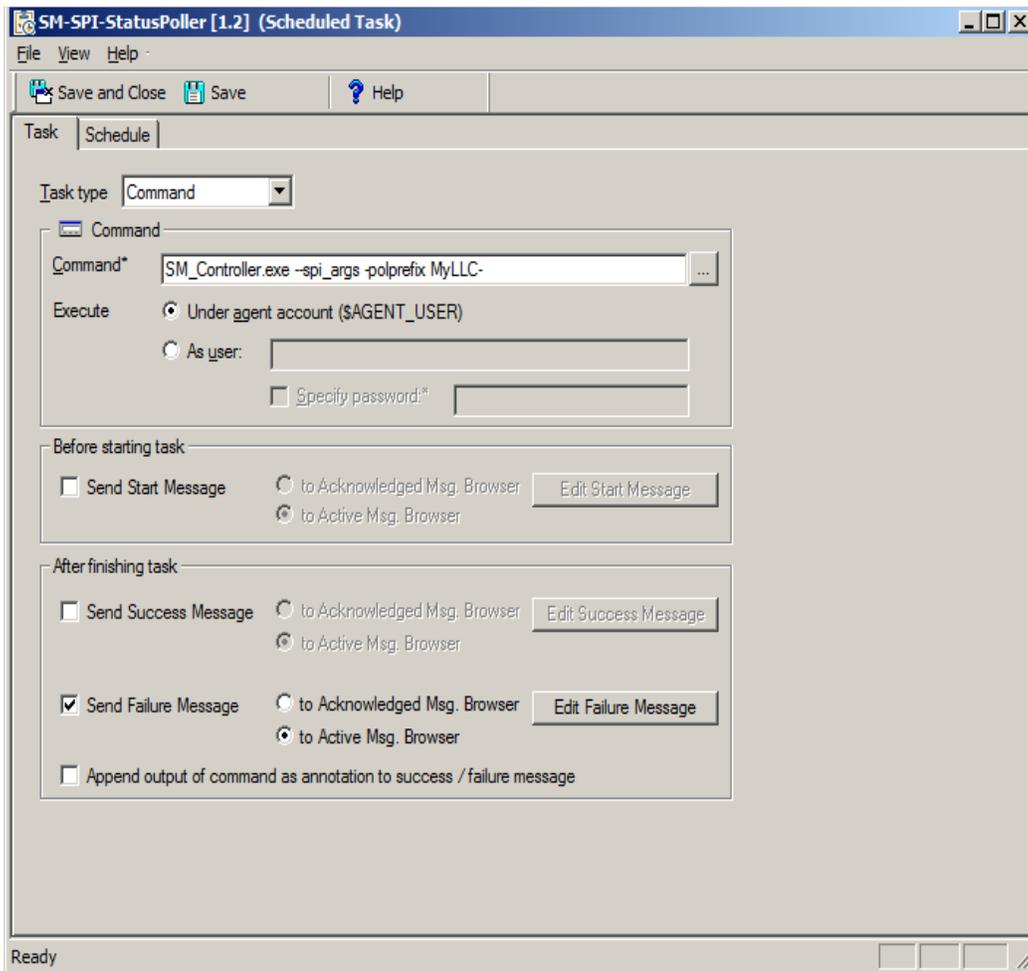
Save the policy change and redistribute the policy to the effected nodes.

Note: Spaces and non-alphanumeric are not permitted in the prefix. Adding them will cause errors and potentially missed alerts from the SMSPI.

Changing Non-Threshold Policy Names

The other policies located outside the "SMSPI-Monitors" folder can be changed without impacting SMSPI functionality. It is still recommended that the same naming convention be applied to all policies for easier management of patches and upgrades.

Changing Polling Frequency

The polling frequency is defined in the **SM-SPI-StatusPoller** policy under the *SMSPI-Monitors* policy folder.   By default the collection runs every 2 minutes, but this number can be changed to higher or lower value based on user preference.   On average, the SMSPI takes less than ten seconds to run for a moderately sized installation of HP Service Manager. For example, if a one minute polling cycle was desired a simple change to the polling policy would resemble the following:
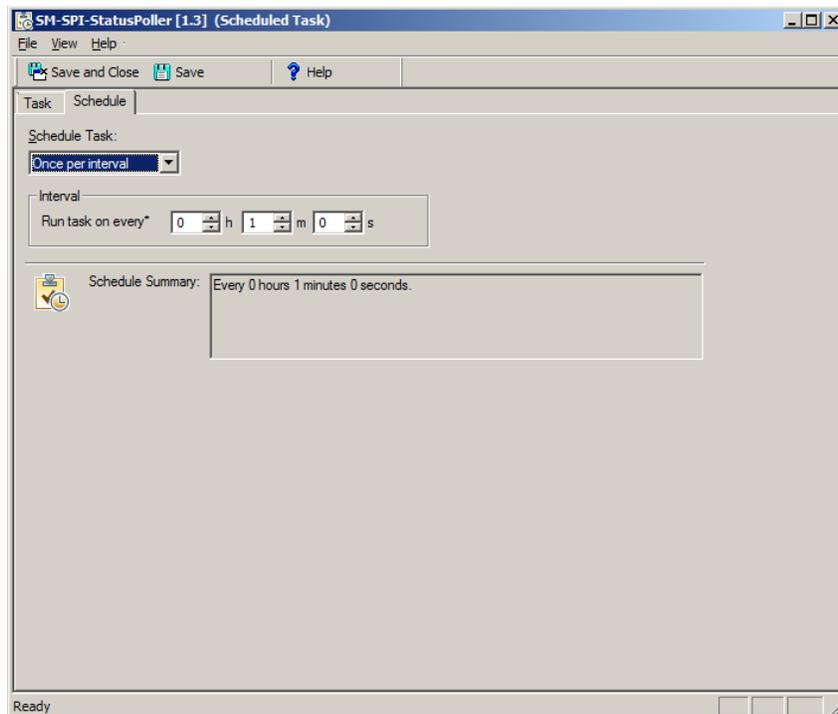
Security

Securing JMX Ports

By entering the settings in the sm.cfg file the administrator is enabling JMX access to the SMSPI. Security administrators may have concerns that these ports could be accessed from outside the system. If so, simply use the Windows firewall to prevent the port range that was defined from external access. The SMSPI will always access these ports locally so it is not a requirement that they be exposed to other systems.

Service Manager User Accounts

The SMSPI does not require an HP Service Manager account for monitoring.

Legal

Copyright Notices

© Copyright 2002-2014 Libbard LLC

Acknowledgments

Product contains software developed by SLF4J (www.slf4j.org)

Product uses software developed by MinGW (www.mingw.org)

Product contains software developed by Eclipse (www.eclipse.org)

Product contains software developed by Apache Commons Project (commons.apache.org)

Product contains software written and developed by Scott J. Spencer (2013-2014)

Trademark Notices

- HP Operations and HP Service Manager are a registered trademark of Hewlett-Packard

- Java is a registered trademark of Oracle Corporation

Support

The trial version comes with 20 days of free support. After a valid license for the SMSPI is obtained there is six months of included support (7AM-6PM EST). After the six month period an additional support agreement can be purchased based upon the size of the installation.

Customers with valid support agreements can reach us at support@libbard.com and placing your customer ID in the subject line. Or call directly at 216.539.5302 to speak to sales or technical support.