

HP Service Health Reporter

Software Version: 9.30

Performance Insight Data Migration Guide



Table of Contents

Introduction	2
Why the Migration?	2
Goal of Migration	2
Benefits of Migration	2
Scope of Migration	3
Ideal Conditions for Migration	4
Data Migration Architecture	5
Assumptions	5
Data Migration – HPOM Deployment Scenario	5
Data Migration – BSM Deployment Scenario	6
Data Migration Approach	6
Known Issues and Limitations	7
Data Migration Procedure	9
Prerequisites	9
PI Migration Content – Packaging	9
Migrating Data from PI	9
Typical Use cases/Scenarios	11
Frequently Asked Questions (FAQ)	12
Acronyms	14
References	15
Data Mapping between System Resource Report Pack of PI and System Performance Content in SHR	15

Introduction

HP Performance Insight (PI) is a performance management and reporting application. Migrating Performance Insight customers to other reporting solutions in the HP Business Service Management (BSM) portfolio is planned in a phased approach.

Why the Migration?

The migration program will help Performance Insight customers (on a valid support contract) to transition to the more advanced BSM reporting tools.

PI customers must be migrated to one or more of the following products based on their use case.

- HP Service Health Reporter (SHR)
- HP Network Node Manager i Software (NNMi) + Network Performance Server (NPS)

Goal of Migration

- To enable PI customers to have near-equivalent functionality as HP moves to consolidate the BSM reporting tools.
- To leverage advanced features like cross-domain analysis and reporting provided by HP Service Health Reporter.

This document provides details of the process for migrating data from the PI System Resource Report Pack on to SHR to leverage the SHR content for System Performance.

Benefits of Migration

The benefits of migrating to Service Health Reporter (Phase 1) are listed as follows:

- Upgrade to the next generation cross-BSM reporting solution.
 - SHR performs a model driven reporting; tying application performance to the underlying infrastructure.
 - Provides a holistic view of your environment.
 - Tracks topology updates automatically.
 - Single pane of glass for viewing the performance of applications, infrastructure, database, and network.
- Brings business intelligence to data center management.
 - SAP BusinessObjects Enterprise is the reporting framework.
 - Allows easier customization.
- Additional functionality.
 - Downtime considerations, custom grouping, customizable shifts, and such features are available in reporting.
 - Provides future analytics considerations such as virtualization capacity planning.
- Scalability
 - A single instance of SHR can directly collect data from 5000 system nodes.

The purpose of this document is to explain the:

- Approach adopted for migrating data from PI to SHR.
- Steps for migrating PI data into SHR content for System Performance.
- Assumptions.
- Known limitations.
- Scalability and performance considerations.

Scope of Migration

The scope of content for data migration from HP Performance Insight (Phase 1) is restricted to System Resource report pack and the following sub-packages:

- SystemResourceCPU
- SystemResource_Disk
- SystemResource_NetInterface

What SHR will do	What SHR will not do
Supports continuous service improvement and historical reporting use cases <ul style="list-style-type: none"> - Consolidated reporting - BSM and non-BSM scenarios Collect from intermediate data stores <ul style="list-style-type: none"> - except performance agent data 	Reports in SHR are historical and high-level; not near-real-time and not intended for troubleshooting use cases. <ul style="list-style-type: none"> - No out-of-the-box reports for diagnostics or troubleshooting.
One time data migration from PI <ul style="list-style-type: none"> - Once data is migrated, live collection from the data sources (agents for System Performance content and HPOM/RTSM for topology) is expected to be configured in SHR 	One time data migration does not include: <ul style="list-style-type: none"> - HPOM SPIs (including Virtualization SPI) - InterfaceReporting/DeviceResource, Traffic, MPLS, QoS, IPTelephony, Service Assurance and other NPS metrics
Only aggregated data (hourly & daily grain), not rate data, will be migrated from PI. Configurations will not be migrated.	Will not use PI as a source for continuous data collection.
Supports customization & extension of content via Content Development Environment (CDE) <ul style="list-style-type: none"> - Individual reports in PI will not be migrated as such to out-of-the-box SHR content. However, custom reports can be created to address additional use cases. - Out-of-the-box content can be customized. 	Will not collect low latency data <ul style="list-style-type: none"> - No real time data in the reports.
Creating custom groups for reporting is supported.	Not supported on the Linux/Unix platform or Oracle/Sybase ASE. Sybase IQ is the only supported database.

Ideal Conditions for Migration

The following information will help identify the ideal conditions for data migration:

- Predominantly uses PI for:
 - Systems (HPOM messages, Database SPI, Virtualization, and SiteScope)
 - Network Data (addressed by NPS)
- Scale of environment
 - ~5000 nodes for SHR
- Possesses the expertise to customize SAP BusinessObjects based reports.
- Understands the underlying domain well (to create custom reports).
- Currently an HP Operations Manager (HPOM) or BSM user.
- Looking for newer capabilities in the reporting solution than what PI provides, such as shifts, downtime, custom grouping, and so on.

Data Migration Architecture

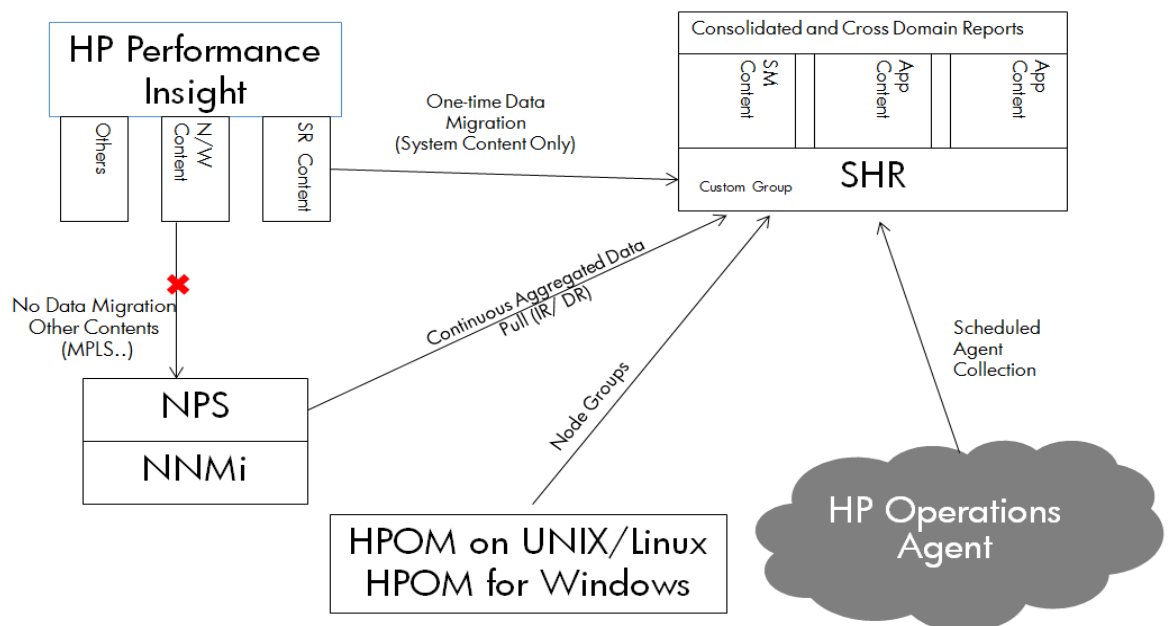
During Phase 1, PI customers will be migrated to one or more BSM products (based on the use case). This section details the architectural approach adopted for data migration, assumptions, scenarios, and known limitations.

Assumptions

- PI is not a data source for continuous data collection.
- The PI customer is a current HPOM or BSM user.
- The size of the reporting solution must be lesser than the scale supported by SHR 9.30.
- SHR will be installed on a different server than PI.
- The customers have PI 5.3 and 5.4x versions; earlier versions are not supported.
- The topology source configuration in SHR is a one-time configuration and needs to stay consistent before or after migration.
- The data migration package brings in the dimension and fact data from PI. The user is expected to install an SHR ETL package, for example, SysPerf_ETL_PerformanceAgent for
 - Bringing in topology data (from the respective topology sources, such as Node groups in HPOM and Business Service based topology in RTSM)
 - On-going live collection from agents because it is a one-time data migration.

The following sections describe the migration approach (Phase 1).

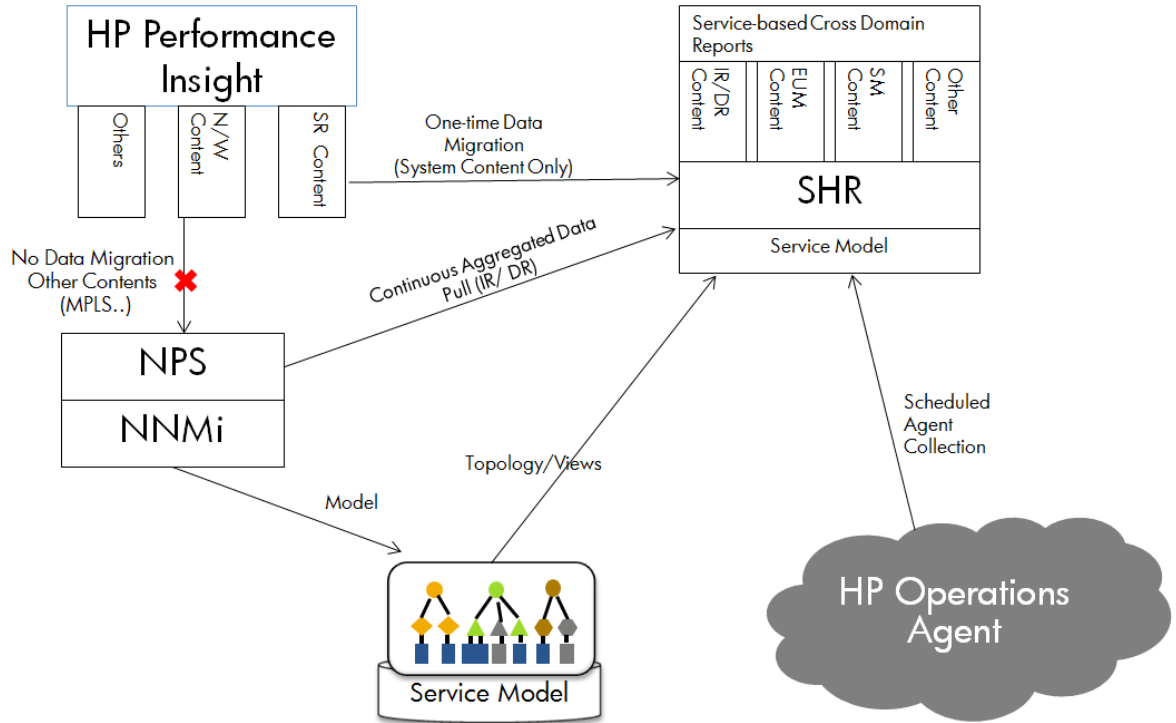
Data Migration – HPOM Deployment Scenario



The data migration package sources topology information from the HPOM (Unix/Linux/Windows) sources that are configured for topology collection in SHR. Data is migrated from the hourly and daily tables of PI System Resource content onto SHR for the nodes monitored by the HPOM configured as the topology source.

When the data is migrated, SysPerf_ETL_PerformanceAgent must be installed to bring in topology/node group information and live rate data from agents into SHR.

Data Migration – BSM Deployment Scenario



The data migration package sources topology information from the RTSM instance configured for topology collection in SHR. Data is migrated from the hourly and daily tables of PI System Resource content onto SHR for the nodes present in SM_PA view in RTSM.

When the data is migrated, SysPerf_ETL_PerformanceAgent must be installed to bring in service based topology information (from SHR views like SM_PA deployed in RTSM) and live rate data from agents into SHR.

Data Migration Approach

The data migration approach makes use of the ETL and orchestration (Audit Balance and Control (ABC)) framework of SHR. Following is a synopsis of the data flow occurring through each ABC Stream:

- Collect topology data.
- Build reconciliation registry.
- Extract data (dimensions/hourly and daily fact data) from PI (both Oracle & Sybase ASE) using DB Collector. The associated PI tables are mentioned in the References section.
- Reconcile data from PI against topology source.
- Load data to SHR aggregate tables using loader.

Collecting Topology Data

Data is collected from configured topology sources in SHR (RTSM and HPOM). The topology collection helps build the reconciliation registry to leverage reconciliation of fact data from PI.

Note: Detailed dimension and topology collection is covered in the post-migration package when live data flow is initiated.

Extracting data from PI

Data extraction from the Oracle/Sybase ASE database used by PI is supported. The migration package leverages the SHR database collection mechanism.

Migration content pack needs to extract data starting from hourly table only from PI database. Since SHR only has hourly and daily tables for System Performance, the data migration occurs from their counterparts in System Resource Report Pack of PI and its sub-packages. The yearly and monthly aggregated data and rate data from PI is not migrated to SHR.

After the data migration, initiate rate data collection in SHR with adequate initial history to bring in raw metrics from HP Performance Agents.

Processing data from PI

The SHR DB Collector dumps collected data in the `%PMDB_HOME%\collect` folder that must be copied to the stage folder using the 'collect' step for further processing. The data collected from PI does not have any topology information, and is it not enriched with CI_UID. Hence, the collected data must be reconciled against the data collected from a topology source.

Loading PI data to SHR

SHR's loader module is used to load the hourly and daily aggregated fact data and also the dimension and location/customer bridge data collected from PI.

Known Issues and Limitations

- Items that cannot be migrated to SHR:
 - The scope of data (metrics and dimensions) brought in from PI includes only those that can be mapped to the data model in SHR. For example, file system data and forecast metrics are not migrated. Detailed list of mapping between PI and SHR schema is provided in the *References* section.
 - Configuration details (collection configuration, default hardcoded shifts in PI, report configurations like schedules, and so on).
 - The planned downtime feature was not supported in PI; hence, it is not available in the migrated data set.
 - SHR 9.30 does not have out-of-the-box reports by customer. However, the customer dimension data and its associations to node dimension is migrated to SHR.

- As PI did not support consistent time zones, the migrated data set has the time zone of its source. In SHR 9.30, users can set the time zone to GMT or to local time. However, the migrated data does not convert differing time zones between PI and SHR. The migrated data has the time zone of its source and the newly collected data, after installing the SysPerf_ETL_PerformanceAgent content, has the time zone configured in SHR.
- The dimension attributes brought in from PI are likely to be updated with those brought in by SHR live collection.
- All hourly and daily data available in PI is collected. Customizing an interval or filtering of such data is not currently available in the migration content.
- Historical Reports in SHR shows migrated data from PI as per the defined retention policy. If the retention of hourly data was seven days in PI, drilling down from daily data prior to seven days will not yield hourly grain data.
- The data migration package brings data of only those nodes from PI that are available as part of the configured HPOM or RTSM topology source.
- The Exception and Grade of Service metrics brought into SHR from PI are computed against the threshold values configured earlier in PI. The live collection in SHR brings in data that leverages threshold values configured in SHR for computation of these metrics. So, the thresholds configured in PI and SHR might not be consistent.
- Only default shifts are applied to the migrated data. Shifts are not expected to be configured in SHR prior to data migration (the shift enrichment could affect the sanity of data being migrated). After the migration, shifts can be configured and the data collected thereafter will have relevant shifts applied.
- If NPS database (or any other external database) is already configured as a generic DB data source and so is PI. During migration, there are chances that data collection is attempted from both sources for the collection policies. In such a case, data collection occurs fine from the right source. However, for the other data source, log failure errors can be encountered and ABC stream status is displayed as WARNING.

Data Migration Procedure

Prerequisites

Prior to initiating data migration of System Performance content from PI to SHR, the following must be ensured:

- The PI Timer Service is stopped (and remains so during the migration period).
- The SHR Collection and Timer services are disabled.
- No shifts are configured in the SHR Administration Console (only the default shifts should be available). This is very significant because the presence of configured shifts in SHR might affect the sanity of aggregate data loaded from PI.

PI Migration Content – Packaging

The PI Data Migration package (`HPSHRSmPIMgr.msi`) is available on Media as well as on Live Network. The same package caters to both RTSM and HPOM deployment scenarios and will have a dependency on the domain content of System Management (`SysPerf_Domain`).

Migrating Data from PI

Task 1: Installing the Migration Package

1. Install the PI Migration package available on the SHR DVD.
(`HPSHRSmPIMgr-9.30.000-Win5.2_64.msi` for Windows and `HPSHRSmPIMgr-9.30.000-Linux2.6_64.rpm` for Linux).

Navigate to the packages folder in the DVD and execute the following command to install the RPM file on Linux:

```
rpm -ivh HPSHRSmPIMgr-9.30.000-Linux2.6_64.rpm
```

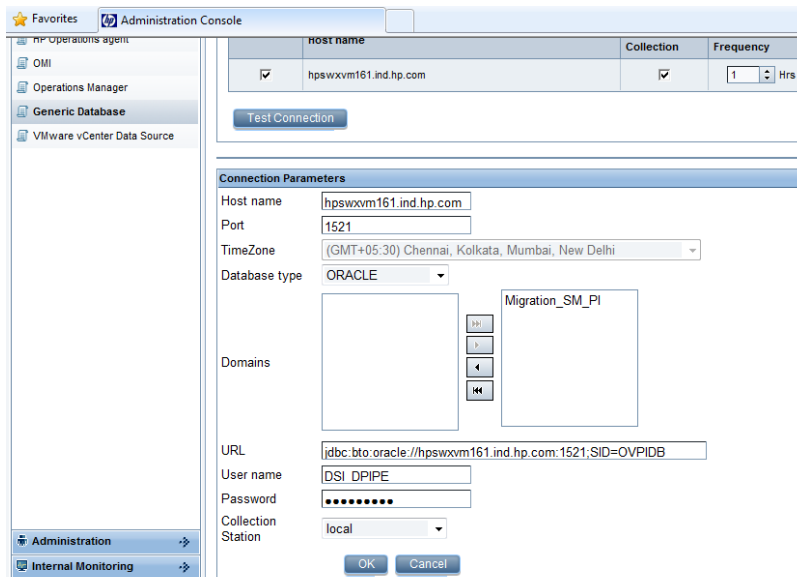
This extracts the data migration package (`Migration_SM_PI\ Migration_SM_PI.ap`) onto the `%PMDb_HOME%/packages` folder on Windows, and the `/$PMDb_HOME/packages` folder on Linux.

2. In the SHR Administration Console, click **Administration > Deployment Manager**.
3. From the **Content Pack Component Name** column, select the check box for `SysPerf_PerformanceInsight_Migration` and click **Install/Upgrade**.
When the installation completes, you must see *Installation Successful* in the **Status** column.

Task 2: Configuring the Data Sources

Configure the topology and PI database sources for data collection as follows:

1. Configure topology source details required for topology collection using RTSM/HPOM topology source configuration page in SHR Administration Console.
2. Configure PI database details as a 'Generic database configuration' in SHR Administration Console.



3. Select the domain name **Migration_SM_PI** to ensure data collection from PI database.
4. Offset the collection station to Local.

Note: See [Typical Use Cases/Scenarios](#) for additional configuration options for deploying PI in a distributed environment.

Task 3: Executing the Data Migration Script

To start data migration, execute the data migration script from command line as follows:

On Windows:

```
trend_proc -f %PMDB_HOME%/packages/Migration_SM_PI/Migration_SM_PI.ap/migrate_pi_shr.pro
```

On Linux:

```
trend_proc -f $PMDB_HOME/packages/Migration_SM_PI/Migration_SM_PI.ap/migrate_pi_shr.pro
```

Since the data migration sequence is facilitated through SHR's data flow orchestration (ABC) framework, the status of the steps involved in migration is available in the Administration Console.

Pertinent Log Files


The following log files would contain details logged during install/execution of the data migration package. The log files are available in the %PMDB_HOME%/log folder on Windows and the \$PMDB_HOME/log folder on Linux systems.

- pagemanager.log – Deployment details of migration package (and its dependency, SysPerf_Domain package) using the Deployment Manager.
- Topologycollector.log – HPOM/RTSM topology collection details.
- reconcileStep.log – Details of reconciliation performed against the dimension data from the configured topology source.
- dbcollector.log – Details of query execution against PI database and join queries involved in the extract phase before generating the initial CSV file.
- dwablauncher.log – Details of ABC stream execution.
- loader.log – Details of the last step of loading hourly/daily aggregated data to SHR tables.

Task 4: Uninstalling the Migration Package

After the data is successfully migrated, the migration package must be uninstalled.

1. In the SHR Administration Console, click **Administration > Deployment Manager**.

- From the **Content Pack Component Name** column, locate the row that contains *SysPerf_PerformanceInsight_Migration* and click the  icon.
When the uninstallation completes, you must see *Uninstallation Successful* in the **Status** column
- Uninstall the PI Migration package from the host:

On Windows:

Run the HPSHRSmPIMgr-9.30.000-Win5.2_64.msi file and follow instructions on the wizard to remove the content pack. Alternatively, you can launch **Control Panel > Add/Remove Programs** and remove it.

On Linux:

To uninstall the HPSHRSmPIMgr-9.30.000-Linux2.6_64.rpm file, execute the following command:

```
rpm -ev HPSHRSmPIMgr-9.30.000-1
```

The migration pack is no longer available for selection in Deployment Manager.

- Delete the PI database details that were configured as a generic database in the SHR Administration Console.

Task 5: Restarting SHR Services and Install System Performance ETL Package

SHR services, including the Collection and Timer Services, can now be enabled and started.

For System Performance content, install the ETL package (SysPerf_ETL_PerformanceAgent) to collect topology data and real time data from performance agents.

Note: The PI Migration content pack performs migration of only the fact and dimension data from PI; the topology/dimension data collection from configured topology sources occur only after the relevant ETL package is installed and a round of topology data collection runs. Start the timer and collection services before data can be viewed in reports.

Typical Use cases/Scenarios

Distributed Environment – Satellite and Central Servers

Execute the migration stream multiple times after configuring the PI satellite/central server details in the SHR Administration Console. So, if there are 2 satellite servers and 1 central server, you must configure the details of the first satellite server in the Administration Console and execute the migrate_pi_shr.pro file. When the stream executes, configure the second satellite server and execute the stream. Follow the same procedure for each satellite and central server. In case of smaller scale deployments, you can configure all sources – central and satellite servers in the Administration Console and execute the migrate_pi_shr.pro file.

Note: If hourly data is available in the central server (depending on the copy policy defined in PI) you do not need to fetch data from individual satellite servers.

Frequently Asked Questions (FAQ)

This section attempts to provide answers to some common queries regarding the data migration process.

Q1. Is it possible to conduct data migration from PI to SHR after SHR has been installed and operational for few months? If other report packs are deployed, should the migration activity be postponed until all the required migration packages are available?

A1. Migration of data from PI can be initiated even after SHR has been installed and operational for some time. But, if SHR has already collected and aggregated data for say one week, then PI will load the data in SHR prior to the week.

Report packs can be migrated in a phased manner. Hence, it is not necessary to wait until all data migration packages are available.

Q2. Can custom groups of nodes be created to use for reporting in SHR?

A2. Custom groups can be defined in SHR in an XML format as follows:

```
<groups>
  <group name="System_customgroup" type="CUSTOMGROUP">
    <instances type="K_CI_System">
      <instance>
        <attribute name="Node_Name" value="node1.ind.hp.com" operator="EQUALS"
          relation="OR" />
        <attribute name="Node_Name" value=" node2.ind.hp.com " operator="EQUALS"
          relation="OR" />
        <attribute name="Node_Name" value=" node3.ind.hp.com " />
      </instance>
    </instances>
  </group>
</groups>
```

- *group name* – Name of the Custom group
 - *type* - preferably "CUSTOMGROUP" or another distinct type (to differentiate it from other group types like VIEWS and NODEGROUPS in HPOM)
 - *instances type* – name of dimension table, dimension instances of which are intended to be grouped.
 - *attribute_name* - the column name of the dimension table, for which instance values would be provided.
 - *value* - the actual data values of the columns for which we want to create a group.
- The XML file name must have the format *customgroup.xml and it should be placed in the folder %PMDB_HOME%/config (on Windows)/\$PMDB_HOME/config (on Linux).
 - Execute the following commands to process the .xml (generate .csv files and stage them):
 - `abcloadNrun -loadBatch -streamId CustomGroup@Platform`
 - `abcloadNrun -runStream -streamId CustomGroup@Platform`
 - Verify that data is loaded into stage tables using the query:
 - `select * from K_CI_Group_Bridge_`
 - The dimension stream in Core CP will execute the loader step (upon invocation via timer) and load it to the data warehouse table K_CI_Group_Bridge.
 - Ensure that the custom group values appear in group selection prompts in SHR reports.

Q3. Is it possible to configure the data migration for a select set of Nodes/Node Groups?

A3. Yes, it is possible only in the HPOM topology source scenario of SHR. By default, node filtering is turned off.

- To enable node filtering, you must set the following property in collection.properties file located at %PMDB_HOME%\config(Windows) or \$PMDB_HOME/config (Linux):
om.filtering.enabled=true
- Create a file containing the list of nodes that must be selectively imported into SHR, say %PMDB_HOME%\config\filterednodes.conf (on Windows)/\$PMDB_HOME/config/filterednodes.conf (on Linux). It must contain the Node Names/FQDNs as they appear in HPOM console.
- Create a file containing the list of groups from HPOM that need to be selectively imported into SHR (say %PMDB_HOME%\config\filteredgroups.conf (on Windows)/\$PMDB_HOME%/config/filteredgroups.conf (on Linux)).
- If both the files exist and have valid values, all nodes belonging to each of the groups under filteredgroups.conf are imported to SHR along with nodes specified in the filterednodes.conf file.

Acronyms

Acronym	Expansion
ABC	Audit, Balance, and Control
BSM	Business Service Management
CDE	Content Development Environment
CP	Content Pack
ETL	Extract, Transform, and Load
HPOM	HP Operations Manager
NNMi	HP Network Node Manager i Software
NPS	Network Performance Server
PI	Performance Insight
RP	Report Pack
RTSM	Run-time Service Model
SHR	Service Health Reporter
SPI	Smart Plug-In

References

Data Mapping between System Resource Report Pack of PI and System Performance Content in SHR

This section provides a mapping between database tables and their columns in PI with those in SHR. Columns that remain unused in SHR reports, even though data is migrated from PI, are displayed in *italics*.

Dimension Table for System	
PI: K_Node	SHR: K_CI_System
NODE_NAME	DNS_Name Host_Name Name Display_Name
OPERATING_SYS	OS
MODEL	Model
MAKE	Vendor
MAKE	Manufacture
NODE_TYPE	HyperVisor_Type
SERIAL_NUM	Serial_Number
NODE_ID	
IP_ADDRESS	
DEPARTMENT	
SYSOBJECTID	
IP_STATE	
SYSOBJECTID	
	Host_Key
	OS_Version
	Internal_Name
	Server_Type
	isVirtual (set to "FALSE")
	Processor_Architecture
	Phys_Mem_GB
	CPU_Num
	CPU_Num_Core
	CPU_Speed_MHz
	Disk_Num
	Network_Num
	Node_Type
	CPUUtil_Threshold
	SwapUtil_Threshold
	MemUtil_Threshold
	RunQ_Threshold
	PageOut_Threshold
	StaticThresholdFlag

Dimension Table for System	
PI: K_Node	SHR: K_CI_System
	CPUFamily
	CI_UID (Retreived from topology source – NODE_NAME from PI is used for reconciliation)
	Creation_time
	Created_by
	Update_time
	Updated_by
	Description
	User_Key
	CPUUtil_STH1
	CPUUtil_STH2
	CPUUtil_STH3
	SwapUtil_STH1
	SwapUtil_STH2
	SwapUtil_STH3
	MemoryUtil_STH1
	MemoryUtil_STH2
	MemoryUtil_STH3
	RunQ_STH1
	RunQ_STH2
	RunQ_STH3
	CIT_Key
	Managed_BY
	Disk_Capacity_GB
	Total_Network_Speed
	State
	CPU_Unreserved
	Role
	HyperVisor_Type
	UUID
	ProcessorModel
	CPUCapacityGHZ
	Datacenter
	Standalone_View (set to "1")
	OSPatch
	Memory_Unreserved
	Cluster_Name
	Hypervisor_View

Fact Table for Node Resources – Hourly Data	
PI: SH_SR_SysXcep	SHR: SH_SM_Node_Res
	dsi_key_id(auto-generated)
ta_period	ta_period
K_Node.K_Location.dsi_key_id	LocationRef
	ShiftRef (auto-generated)
AVGrung	avgRunQ

Fact Table for Node Resources – Hourly Data	
PI: SH_SR_SysXcep	SHR: SH_SM_Node_Res
P95runq	P95RunQ
TOTrunq_grade	totRunQ_Grade
AVGrunq_grade	avgRunQ_Grade
AVGcpuutil	avgCPUUtil
TCTcpuutil	tctCPUUtil
P95cpuutil	P95CPUUtil
TOTcpuutil_grade	totCPUUtil_Grade
AVGcpuutil_grade	avgCPUUtil_Grade
AVGmemutil	avgMemUtil
TCTmemutil	tctMemUtil
P95memutil	P95MemUtil
TOTmemutil_grade	totMemUtil_Grade
AVGmemutil_grade	avgMemUtil_Grade
AVGswapUtil	avgSwapUtil
TCTswapUtil	tctSwapUtil
P95swapUtil	P95SwapUtil
TOTswapUtil_grade	totSwapUtil_Grade
AVGswapUtil_grade	avgSwapUtil_Grade
AVGmemPageOutRate	avgMemPageOutRate
TCTmemPageOutRate	tctMemPageOutRate
P95memPageOutRate	P95MemPageOutRate
TOTavgNumProcs	totNumProcs
AvgNumProcs	avgNumProcs
PI: SH_SR_SysVolXcep	SHR: SH_SM_Node_Res
TOTInPackets	totInPackets
TOTOutPackets	totOutPackets
TOTvolume	totVolume
	maxVolume
WAVvolume	avgVolume
	avgPacketRate
	maxPacketRate
AVGCollisionRate	avgCollisionRate
	maxCollisionRate
AVGErrorRate	avgErrorRate
	maxMemUtil
	P90MemUtil
	avgFreeMemGB
	maxRunQ
	P90RunQ
	maxCPUUtil
	P90CPUUtil
	maxSwapUtil
	P90SwapUtil
	P90MemPageOutRate
	avgDiskPhysIORate
	maxDiskPhysIORate
	P90DiskPhysIORate

Fact Table for Node Resources – Hourly Data	
PI: SH_SR_SysXcep	SHR: SH_SM_Node_Res
	avgNetIORate
	maxNetIORate
	P90NetIORate
	totNumStartedProcs
	totNumActiveProcs
	maxErrorRate
	avgSysModeUtil
	maxSysModeUtil
	avgUsrModeUtil
	maxUsrModeUtil
	avgCSwitchRate
	maxCSwitchRate
	avgInterruptRate
	maxInterruptRate
	avgByteRate
	ubsavgMemoryUtil
	lbsavgMemoryUtil
	ubsavgRunQ
	lbsavgRunQ
	ubsavgCPUUtil
	lbsavgCPUUtil
	ubsavgSwapUtil
	lbsavgSwapUtil
	ubsavgDiskPhysIORate
	lbsavgDiskPhysIORate
	ubsByteRate
	lbsByteRate
	avgNetworkUtilMbps
	NetInByteRate
	NetOutByteRate
	avgDiskPhysreadbyteRate
	avgDiskPhyswritebyteRate
	avgReadlatency
	avgWritelatency
	VMNum
	DiskSpaceUtil

Fact Table for Node Resources – Daily Data	
PI:SD_SR_SysXcep	SHR:SD_SM_Node_Res
	dsi_key_id(auto-generated)
ta_period	ta_period
K_Node.K_Location.dsi_key_id	LocationRef
	ShiftRef (auto-generated)
AVGrunq	avgRunQ
P95runq	P95RunQ
TOTrunq_grade	totRunQ_Grade

Fact Table for Node Resources – Daily Data	
PI:SD_SR_SysXcep	SHR:SD_SM_Node_Res
AVGrunq_grade	avgRunQ_Grade
AVGcpuutil	avgCPUUtil
TCTcpuutil	tctCPUUtil
P95cpuutil	P95CPUUtil
TOTcpuutil_grade	totCPUUtil_Grade
AVGcpuutil_grade	avgCPUUtil_Grade
AVGmemutil	avgMemUtil
TCTmemutil	tctMemUtil
P95memutil	P95MemUtil
TOTmemutil_grade	totMemUtil_Grade
AVGmemutil_grade	avgMemUtil_Grade
AVGswapUtil	avgSwapUtil
TCTswapUtil	tctSwapUtil
P95swapUtil	P95SwapUtil
TOTswapUtil_grade	totSwapUtil_Grade
AVGswapUtil_grade	avgSwapUtil_Grade
AVGmemPageOutRate	avgMemPageOutRate
TCTmemPageOutRate	tctMemPageOutRate
P95memPageOutRate	P95MemPageOutRate
TOTavgNumProcs	totNumProcs
AvgNumProcs	avgNumProcs
PI: SD_SR_SysVolXcep	SHR: SD_SM_Node_Res
TOTInPackets	totInPackets
TOTOutPackets	totOutPackets
TOTvolume	totVolume
	maxVolume
WAVvolume	avgVolume
	avgPacketRate
	maxPacketRate
AVGCollisionRate	avgCollisionRate
	maxCollisionRate
AVGErrorRate	avgErrorRate
	maxMemUtil
	P90MemUtil
	avgFreeMemGB
	maxRunQ
	P90RunQ
	maxCPUUtil
	P90CPUUtil
	maxSwapUtil
	P90SwapUtil
	P90MemPageOutRate
	avgDiskPhysIORate
	maxDiskPhysIORate
	P90DiskPhysIORate
	avgNetIORate
	maxNetIORate

Fact Table for Node Resources – Daily Data	
PI:SD_SR_SysXcep	SHR:SD_SM_Node_Res
	P90NetIORate
	totNumStartedProcs
	totNumActiveProcs
	maxErrorRate
	avgSysModeUtil
	maxSysModeUtil
	avgUsrModeUtil
	maxUsrModeUtil
	avgCSwitchRate
	maxCSwitchRate
	avgInterruptRate
	maxInterruptRate
	avgByteRate
	ubsavgMemoryUtil
	lbsavgMemoryUtil
	ubsavgRunQ
	lbsavgRunQ
	ubsavgCPUUtil
	lbsavgCPUUtil
	ubsavgSwapUtil
	lbsavgSwapUtil
	ubsavgDiskPhysIORate
	lbsavgDiskPhysIORate
	ubsByteRate
	lbsByteRate
	avgNetworkUtilMbps
	NetInByteRate
	NetOutByteRate
	avgDiskPhysreadbyteRate
	avgDiskPhyswritebyteRate
	avgReadlatency
	avgWritelatency
	VMNum

Fact Table for Node Availability – Hourly Data	
PI: SH_SR_SysUp	SHR: SH_SM_Node_Avail
	dsi_key_id (auto-generated)
ta_period	ta_period
K_Node.K_Location.dsi_key_id	LocationRef
	ShiftRef (auto-generated)
TOTUPTIME	totuptime
TOTDOWNTIME	totdowntime
	Totplandtime
	Totexcdtime
	Totunknowntime
	Totavailability

Fact Table for Node Availability – Daily Data	
PI: SD_SR_SysUp	SHR: SD_SM_Node_Avail
	<i>dsi_key_id (auto-generated)</i>
ta_period	ta_period
K_Node.K_Location.dsi_key_id	LocationRef
	ShiftRef (auto-generated)
TOTUPTIME	totuptime
TOTDOWNTIME	totdowntime
	Totplandtime
	Totexcdtime
	Totunknowntime
	Totavailability

Dimension Table for CPU	
PI: K_System_CPU	SHR: K_SM_CPU
	<i>dsi_key_id (auto-generated)</i>
CPUID	cpu_id
(K_Node.node_name(K_System_CPU.node_fk)	node_name
	Systemref(auto-gen (using ciid - reconciled using node_name))
	cpu_vendor
	cpu_speed

Fact Table for CPU – Hourly Data	
PI: SH_SR_CPU	SHR: SH_SM_CPU
	<i>dsi_key_id (auto-generated)</i>
ta_period	ta_period
K_Node.K_Location.dsi_key_id	LocationRef
	ShiftRef (auto-generated)
AVGCPUUTIL	avgTotUtil
AVGCPUSYSTEMMODE	avgSysModeUtil
AVGCPUUSERMODE	avgUsrModeUtil
AVGINTRATE	avgInterruptRate
AVGCSRATE	avgCntxtSwitchRate
	maxTotUtil
	maxSysModeUtil
	maxUsrModeUtil
	maxInterruptRate
	maxCntxtSwitchRate

Fact Table for CPU – Daily Data	
PI: SD_SR_CPU	SHR: SD_SM_CPU
	<i>dsi_key_id (auto-generated)</i>
ta_period	<i>ta_period</i>
K_Node.K_Location.dsi_key_id	<i>LocationRef</i>
	<i>ShiftRef (auto-generated)</i>
AVGCPUUTIL	<i>avgTotUtil</i>
AVGCPUSYSTEMMODE	<i>avgSysModeUtil</i>
AVGCPUUSERMODE	<i>avgUsrModeUtil</i>
AVGINTRATE	<i>avgInterruptRate</i>
AVGCSRATE	<i>avgCntxtSwitchRate</i>
	<i>maxTotUtil</i>
	<i>maxSysModeUtil</i>
	<i>maxUsrModeUtil</i>
	<i>maxInterruptRate</i>
	<i>maxCntxtSwitchRate</i>

Dimension Table for Physical Disk	
PI: K_Disk_Dsk	SHR: K_SM_PhysicalDISK
	<i>dsi_key_id (auto-generated)</i>
prop_disk_name	<i>disk_name</i>
(K_Node.node_name(K_System_CPU.node_fk))	<i>node_name</i>
	<i>Systemref(auto-gen (using ciid - reconciled using node_name))</i>
	<i>Dir_name</i>

Fact Table for Physical Disk – Hourly Data	
PI: SH_SR_Disk	SHR: SH_SM_Disk
	<i>dsi_key_id (auto-generated)</i>
ta_period	<i>ta_period</i>
K_Node.K_Location.dsi_key_id	<i>LocationRef</i>
	<i>ShiftRef (auto-generated)</i>
AVGDISKUTIL	<i>avgPctUtil</i>
AVGPHYSICALIORATE	<i>avgPhyIORate</i>
AVGPHYSREADRATE	<i>avgPhyReadRate</i>
AVGPHYSWRITERATE	<i>avgPhyWriteRate</i>
AVGSYSTEMIORATE	<i>avgSysIORate</i>
AVGVMIORATE	<i>avgVMIORate</i>
	<i>avgPhyByteRate</i>
	<i>avgPhyReadByteRate</i>
	<i>avgPhyWriteByteRate</i>
	<i>avgRawReadRate</i>
	<i>avgRawWriteRate</i>
	<i>maxPctUtil</i>
	<i>maxPhyIORate</i>
	<i>maxPhyReadRate</i>
	<i>maxPhyWriteRate</i>

Fact Table for Physical Disk – Hourly Data	
PI: SH_SR_Disk	SHR: SH_SM_Disk
	maxSysIORate
	maxVMIORate
	maxPhyByteRate
	maxPhyReadByteRate
	maxPhyWriteByteRate
	maxRawReadRate
	maxRawWriteRate

Fact Table for Physical Disk – Daily Data	
PI: SD_SR_Disk	SHR: SD_SM_Disk
	<i>dsi_key_id (auto-generated)</i>
ta_period	<i>ta_period</i>
K_Node.K_Location.dsi_key_id	<i>LocationRef</i>
	<i>ShiftRef (auto-generated)</i>
AVGDISKUTIL	<i>avgPctUtil</i>
AVGPHYSICALIORATE	<i>avgPhyIORate</i>
AVGPHYSREADRATE	<i>avgPhyReadRate</i>
AVGPHYSWRITERATE	<i>avgPhyWriteRate</i>
AVGSYSTEMIORATE	<i>avgSysIORate</i>
AVGVMIORATE	<i>avgVMIORate</i>
	<i>avgPhyByteRate</i>
	<i>avgPhyReadByteRate</i>
	<i>avgPhyWriteByteRate</i>
	<i>avgRawReadRate</i>
	<i>avgRawWriteRate</i>
	<i>maxPctUtil</i>
	<i>maxPhyIORate</i>
	<i>maxPhyReadRate</i>
	<i>maxPhyWriteRate</i>
	<i>maxSysIORate</i>
	<i>maxVMIORate</i>
	<i>maxPhyByteRate</i>
	<i>maxPhyReadByteRate</i>
	<i>maxPhyWriteByteRate</i>
	<i>maxRawReadRate</i>
	<i>maxRawWriteRate</i>

Dimension Table for System Net Interface	
PI: K_NetInterface_NetIf	SHR: K_SM_NetInterface
	<i>dsi_key_id (auto-generated)</i>
prop_netif_name	<i>Interface_name</i>
(K_Node.node_name(K_System_CPU.node_fk))	<i>Node_Name</i>
	<i>Systemref(auto-gen (using ciid - reconciled using node_name))</i>
	<i>Network_speed</i>

Fact Table for System Net Interface – Hourly Data	
PI: SH_SR_NetInterface	SHR: SH_SM_NetInterface
	dsi_key_id (auto-generated)
ta_period	ta_period
K_Node.K_Location.dsi_key_id	LocationRef
	ShiftRef (auto-generated)
	avgPktRate
AVGINPACKETS	avgInPktRate
AVGOUTPACKETS	avgOutPktRate
	avgByteRate
AVGINBYTES	avgInByteRate
AVGOUTBYTES	avgOutByteRate
AVGCOLLISIONRATE	avgCollisionRate
AVGERRORRATE	avgErrorRate
	NetworkUtil
	maxPktRate
	maxInPktRate
	maxOutPktRate
	maxByteRate
	maxInByteRate
	maxOutByteRate
	maxCollisionRate
	maxErrorRate

Fact Table for System Net Interface – Daily Data	
PI: SD_SR_NetInterface	SHR: SD_SM_NetInterface
	dsi_key_id (auto-generated)
ta_period	ta_period
K_Node.K_Location.dsi_key_id	LocationRef
	ShiftRef (auto-generated)
	avgPktRate
AVGINPACKETS	avgInPktRate
AVGOUTPACKETS	avgOutPktRate
	avgByteRate
AVGINBYTES	avgInByteRate
AVGOUTBYTES	avgOutByteRate
AVGCOLLISIONRATE	avgCollisionRate
AVGERRORRATE	avgErrorRate
	NetworkUtil
	maxPktRate
	maxInPktRate
	maxOutPktRate
	maxByteRate
	maxInByteRate

Fact Table for System Net Interface – Daily Data	
PI: SD_SR_NetInterface	SHR: SD_SM_NetInterface
	maxOutByteRate
	maxCollisionRate
	maxErrorRate

Property Table for Customer Metrics	
PI: K_Customer	SHR: K_Customer
	<i>dsi_key_id (auto-generated)</i>
cust_id	Customer_ID
cust_name	Name Display_Name
	User_Key
	Address1
	Address2
	City
	State
	ZIP_Code
	Phone_Number
	Description

Dimension Table for CI Customer Bridge	
PI: K_SR_System, K_Node, K_Customer	SHR: K_CI_Cust_Bridge
	<i>dsi_key_id (auto-generated)</i>
K_SR_System.node_fk(reconciled to get CI_UID)	CI_Key
K_Customer.cust_name	Cust_Key
	Valid_Period_Start
	Valid_Period_End

Dimension Table for Location	
PI: K_Location	SHR: K_Location
	<i>dsi_key_id (auto-generated)</i>
	Country ('Default')
	State ('Default')
	City ('Default')
location_name	Region
	Address ('Default')
	Building ('Default')
	Floor ('Default')
location_name	Name

Dimension Table for CI Location Bridge	
PI: K_SR_System, K_Node, K_Location	SHR: K_CI_Loc_Bridge
	<i>dsi_key_id (auto-generated)</i>
K_SR_System.node_fk(reconciled to get CI_UID)	CI_Key

Dimension Table for CI Location Bridge	
PI: K_SR_System, K_Node, K_Location	SHR: K_CI_Loc_Bridge
K_Location. location_name	Loc_Key
	Valid_Period_Start
	Valid_Period_End

© 2013 Hewlett-Packard Development Company, L.P. The information contained herein is subject to change without notice. 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.

