

# HP Service Health Reporter

ソフトウェアバージョン: 9.20

## Performance Insight データ移行ガイド



### 目次

はじめに .....	2
移行する理由 .....	2
移行の目的 .....	2
移行の利点 .....	2
移行の範囲 .....	3
データ移行に適した状況の性質 .....	4
用語の定義 .....	4
データ移行 - アーキテクチャー .....	5
前提条件 .....	5
Performance Insight からのデータ移行 - OM デプロイメントのシナリオ .....	6
Performance Insight からのデータ移行 - BSM デプロイメントのシナリオ .....	7
データ移行アプローチ .....	7
既知の問題および制限事項 .....	8
データ移行手順 .....	10
前提条件 .....	10
PI 移行コンテンツ - パッケージ .....	10
PI からのデータ移行で実行される手順 .....	10
一般的な使用事例/シナリオ .....	11
よくある質問 [FAQ] .....	12
参考文献 .....	14
PI の System Resource Report Pack と SHR の System Performance コンテンツ間のデータマッピング .....	14

# はじめに

HP Performance Insight は、パフォーマンス管理およびレポートアプリケーションです。既存の Performance Insight カスタマーを BSM ポートフォリオの他のレポートソリューションに移行する場合、段階的なアプローチとして計画されます。

## 移行する理由

HP PI はいずれ廃止されます。移行プログラムを使用すると、(有効なサポート契約のある) 現在の Performance Insight カスタマーを他の BSM レポートツールに移行しやすくなります。

(使用事例に基づいて) 1 つ以上の製品に移行する HP PI カスタマー

- Service Health Reporter
- NNMi+Performance iSPI

## 移行の目的

- HP は BSM レポートツールを統合しようとしており、これと同等の機能を HP Performance Insight カスタマーが得られるようにする
- HP PI カスタマーが Service Health Reporter に移行して、x ドメイン分析およびレポートなどの高度な機能を活用できるようにする

このドキュメントでは、SHR の System Performance コンテンツを活用するために、HP PI System Resource Report Pack から SHR にデータを移行するプロセスの詳細について説明します。

## 移行の利点

Service Health Reporter への移行 (フェーズ 1) の利点を以下に示します。

- 次世代クロス BSM レポートソリューションへのアップグレード
  - SHR によるモデル駆動レポート => アプリケーションパフォーマンスと基盤インフラストラクチャーの結合
  - 環境の全体的なビュー
  - トポロジ更新の自動追跡
  - サービスのアプリケーション、インフラストラクチャー、データベース、およびネットワークのパフォーマンスを確認するための単一ペイン
- ビジネスインテリジェンスからデータセンター管理への移行
  - レポートフレームワークとしての SAP Business Objects Enterprise
  - 容易なカスタマイズ
- 追加機能
  - レポートにおけるダウンタイムの考慮/カスタムグループ化/カスタマイズ可能なシフトなど
  - 将来の分析プラットフォームの考慮 (仮想化キャパシティプランニングなど)
- 拡張性
  - 単一の SHR インスタンスで 5000 個のシステムノードから直接収集

このドキュメントの目的は、以下の事項を説明することです。

- Performance Insight から SHR へのデータ移行で採用されるアプローチ
- Performance Insight の System Performance コンテンツのデータ移行で実行される手順

- 前提条件
- 既知の制限事項
- 拡張性およびパフォーマンスの側面

## 移行の範囲

Performance Insight からのデータ移行 (フェーズ 1) のコンテンツ範囲は、System Resource レポートパックと以下のサブパッケージに限定されています。

- SystemResourceCPU
- SystemResource\_Disk
- SystemResource\_NetInterface

SHR で行われること	SHR で行われなかったこと
<p>継続的なサービスの改善と履歴レポートの各使用事例に対応</p> <ul style="list-style-type: none"> <li>- 統合レポート</li> <li>- BSM および BSM 以外のシナリオ</li> </ul> <p>中間データストアから収集</p> <ul style="list-style-type: none"> <li>- Performance Agent によるデータ収集を除く</li> </ul>	<p>SHR のレポートは、リアルタイムレポートでなく履歴および高レベルレポートで、トラブルシューティングの使用事例を意図していない</p> <ul style="list-style-type: none"> <li>- 診断またはトラブルシューティング用の標準レポートはない</li> </ul>
<p>PI から一括データ移行</p> <ul style="list-style-type: none"> <li>- データが移行されてから、ソースからのライブ収集 (System Performance コンテンツの場合はエージェント、トポロジの場合は HP OM/RtSM) が SHR で設定されることが想定される</li> </ul>	<p>一括データ移行には以下は含まれない</p> <ul style="list-style-type: none"> <li>- HP OM SPI (Virtualization SPI など)</li> <li>- InterfaceReporting/DeviceResource、Traffic、MPLS、QoS、IPTelephony、Service Assurance および他の Network Performance SPI メトリック</li> </ul>
<p>PI からは集計データ (時間別および日別グレン) のみが移行され、割合データは移行されない</p> <p>設定は移行されない</p>	<p>PI は連続データを収集するときのソースとして使用されない</p>
<p>コンテンツ開発環境 (CDE) を介したコンテンツのカスタマイズおよび拡張をサポート</p> <ul style="list-style-type: none"> <li>- PI の個々のレポート自体を標準の SHR コンテンツに移行することはできないが、カスタムレポートを作成してその他の使用事例 (ある場合) に対応できる</li> <li>- 標準のコンテンツをカスタマイズすることも可能</li> </ul>	<p>遅延の少ないデータは収集されない</p> <ul style="list-style-type: none"> <li>- レポートにリアルタイムデータはない</li> </ul>
<p>レポート用のカスタムグループの作成がサポートされる</p>	<p>Linux/Unix または Oracle/Sybase ASE ではサポートされず、Sybase IQ データベースのみがサポートされる</p>

## データ移行に適した状況の性質

以下はデータ移行に適した状況を特定するのに役立ちます。

- PIの主な用途:
  - システム (OMメッセージ/Database SPI/Virtualization および Sitescope)
  - ネットワークデータ (Perf/iSPI 対応)
- 環境の規模
  - ~5000ノード (SHR)
- SAP Business Objects ベースのレポートをカスタマイズするための専門知識
- 基盤ドメインの深い理解 (カスタムレポートを作成するため)
- 現在の OM または BSM ユーザー
- PI を超えるレポートソリューションの新機能
- (シフト、ダウンタイム、カスタムグループ化など) を求めているユーザー

## 用語の定義

用語	定義
CP	コンテンツパック
RP	レポートパック
DWH	データウェアハウス
HP PI	HP Performance Insight
BSM	Business Service Management
RtSM	Run time Service Model
OM	Operation Manager
CDE	コンテンツ開発環境
SPI	Smart Plug-In
SHR	Service Health Reporter
ETL	抽出、変換と読み込み
ABC	バランスとコントロールを監査

# データ移行 - アーキテクチャー

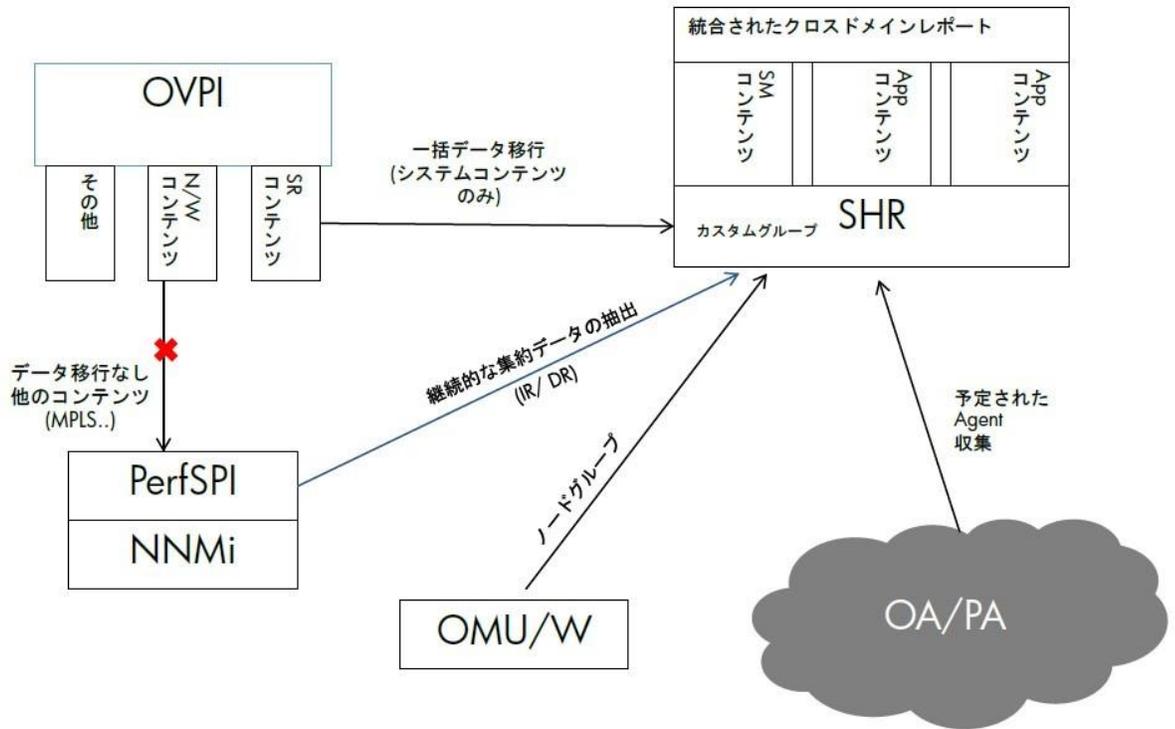
フェーズ 1 では、PI カスタマーは (使用事例に基づいて) 1 つ以上の BSM 製品に移行されます。このセクションでは、データ移行で採用されるアーキテクチャーアプローチ、作成される前提条件、カバーされるシナリオ、および既知の制限について詳細に説明します。

## 前提条件

- PI は、連続データを収集するためのデータソースではない (一括データ移行アプローチ)
- PI カスタマーは現在の OM または BSM ユーザー
- レポートソリューションのサイズは SHR v9.20 でサポートされる規模よりも小さい
- SHR は個別のサーバーにインストールされる (PI サーバーでの共存は想定されていない)
- HP PI の各バージョン (5.3 および 5.4x) のみがサポートされる – 以前のバージョンの PI (およびサポートされていない HP PI の設定) は移行のデータソースとしてサポートされない
- SHR のトポロジソースの設定は一括設定で、移行の前後の整合性を確保する必要がある
- データ移行パッケージにより、PI からディメンションおよびファクトデータが取り込まれる。以下のためにユーザーが SHR ETL パッケージ (SysPerf\_ETL\_PerformanceAgent など) をインストールすることが想定されます。
  - 各トポロジソース (RtSM の OM およびビジネスサービスベーストポロジのノードグループなど) からのトポロジデータの取り込み
  - エージェントからの継続的なライブ収集 (一括データ移行であるため)

以下のセクションでは、移行アプローチ (フェーズ 1) について説明します。

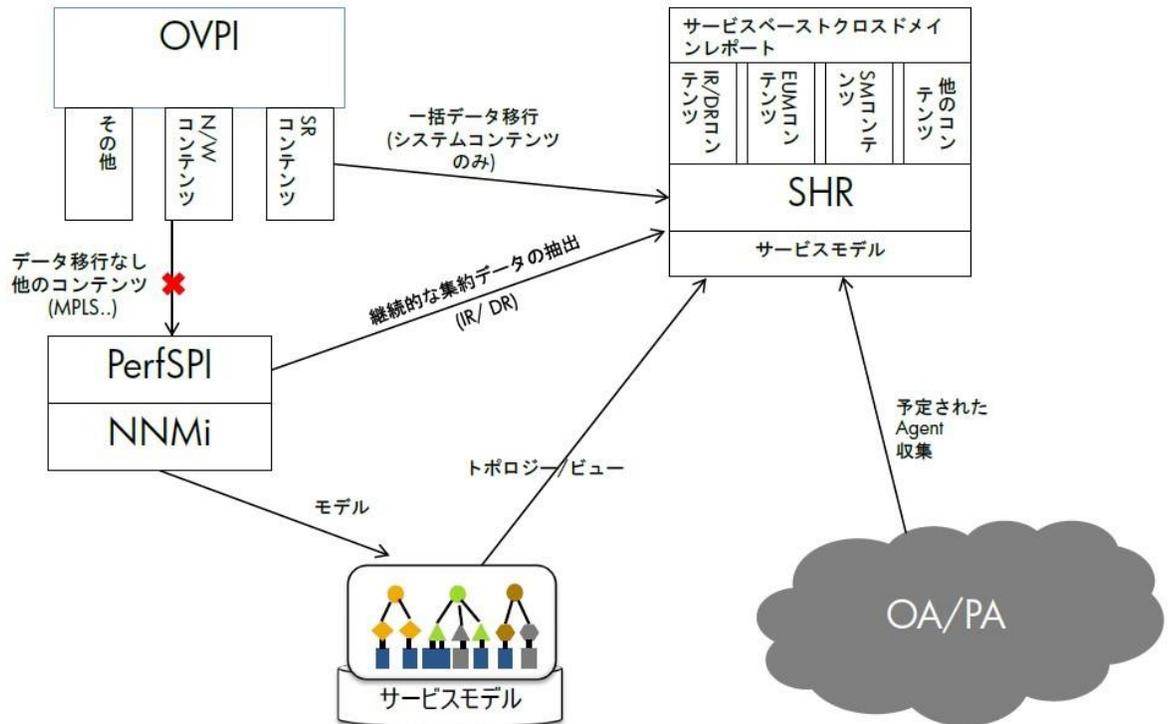
## Performance Insightからのデータ移行 – OMデプロイメントのシナリオ



OM デプロイメントのシナリオでは、SHR でトポロジ収集用に設定された OM (Unix/Linux/Windows) ソースのトポロジ情報がデータ移行パッケージのソースになります。トポロジソースとして設定されたノード (OM によって監視されるノード) を対象として、PI System Resource コンテンツの時間別および日別テーブルから SHR にデータが移行されます。

データ移行が完了したら、SysPerf\_ETL\_PerformanceAgent をインストールして、トポロジ/ノードグループ情報およびライブ割合データをエージェントから SHR に取り込む必要があります。

## Performance Insightからのデータ移行 – BSMデプロイメントのシナリオ



BSM デプロイメントのシナリオでは、SHR でトポロジ収集用に設定された RiSM インスタンスのトポロジ情報がデータ移行パッケージのソースになります。RiSM の SM\_PA ビューにあるノードを対象として、PI System Resource コンテンツの時間別および日別テーブルから SHR にデータが移行されます。

データ移行が完了したら、SysPerf\_ETL\_PerformanceAgent をインストールして、(RiSM にデプロイされる SM\_PA などの SHR ビューの) サービスベーストポロジ情報およびライブ割合データをエージェントから SHR に取り込む必要があります。

### データ移行アプローチ

データ移行アプローチでは、SHR の ETL およびオーケストレーション (バランスとコントロールを監査 (ABC)) フレームワークを利用します。各 ABC ストリームで発生するデータフローの概要を以下に示します。

- トポロジデータを収集する
- 調整レジストリを構築する
- データベースコレクターを使用して PI (Oracle と Sybase ASE の両方) からデータ (ディメンション/時間別および日別ファクトデータ) を抽出する (関連する PI テーブルの詳細については、このドキュメントの「参考文献」セクションを参照してください)
- トポロジソースに対して PI データを調整する
- ローダーを使用して SHR 集計テーブルにデータをロードする

## トポロジデータの収集

SHR (RtSM および OM) で設定されたトポロジソースからデータが収集されます。PI のファクトデータの調整に活用される調整レジストリを構築するためにトポロジ収集が必要になります。

注: 詳細なディメンションおよびトポロジ収集は、移行パッケージでカバーされません。これは、移行後にライブデータフローが開始されてから行われます。

## PI からのデータの抽出

HP PI で使用される Oracle/Sybase ASE データベースでは、データ抽出がサポートされています。移行パッケージでは、このために SHR データベース収集メカニズムを活用します。

移行コンテンツパックでは、PI データベースの時間別テーブルのみを起点としてデータを抽出する必要があります。SHR には System Performance の時間別および日別テーブルしかないため、データは PI の System Resource Report Pack (およびそのサブパッケージ) の対応するテーブルから移行されます。PI の年別および月別集計データや割合データは SHR に移行されません。

データ移行の後に、HP Performance Agent から未処理のメトリックを取り込める十分な初期履歴を使用して、SHR で割合データの収集を開始できます。

## HP PI のデータの処理

収集されたデータは SHR のデータベースコレクターによって '%PMDB\_HOME%\collect' フォルダにダンプされます。さらに処理するには、「収集」手順を使用してこのデータをステージフォルダにコピーする必要があります。PI から収集されたデータにはトポロジ情報がありません (CL\_UID で強化されてもいません)。したがって、PI から収集されたデータは、トポロジソースから収集されたデータに対して調整される必要があります。

## SHR への HP PI データのロード

SHR のローダーモジュールを使用して、時間別および日別集計ファクトデータや、PI から収集されたディメンションおよび場所/カスタマーブリッジデータをロードします。

## 既知の問題および制限事項

- 移行されない項目:
  - PI から取り込まれたデータ範囲 (メトリックおよびディメンション) には、SHR のデータモデルにマップできるもののみが含まれます (たとえば、ファイルシステムデータ/予測メトリックは移行されません)。PI および SHR スキーマ間のマッピングの詳細なリストについては、このドキュメントの「参考文献」セクションを参照してください。
  - 設定の詳細 (収集設定、PI のハードコードされているデフォルトのシフト、スケジュールなどのレポート設定など) は SHR に移行されません。
  - PI では計画ダウンタイム機能はサポートされていなかったため、移行されたデータセットでこの機能は使用できません。
  - SHR 9.20 には、カスタマー用のすぐに使える標準レポートはありません。ただし、カスタマーのディメンションデータおよびノードのディメンションへの関連付けは SHR に移行されます (カスタムユニバース/レポートは同様に作成できます)。

- PI では整合性のあるタイムゾーンはサポートされていなかったため、移行されたデータセットはソースのタイムゾーンで使用できます。SHR 9.20 では、ユーザーはタイムゾーンを GMT またはローカルタイムゾーンに設定できます。ただし、データ移行では PI と SHR 間でタイムゾーンが異なる場合にタイムゾーンの変換は行われません。移行されたデータはソースのタイムゾーンで使用でき、新しく収集されたデータ (SysPerf\_ETL\_PerformanceAgent コンテンツのインストール後) は SHR で設定したタイムゾーンになります。
- 多くの場合、PI から取り込まれたディメンション属性は、SHR のライブ収集で取り込まれたディメンション属性で更新されます。
- PI で使用可能なすべての時間別および日別データが収集されます。現在、移行コンテンツで間隔のカスタマイズやそのようなデータのフィルタリングを行うことはできません。
- SHR の履歴レポートには、PI で設定された保持期間の範囲でのみ、移行された PI のデータ (時間別/日別) が表示されます。たとえば、PI の時間別データの保持期間が 7 日間しかなかった場合、7 日間より前の日別データからドリルダウンしても、時間別グレーンのデータは表示されません。
- データ移行パッケージでは、設定した OM/RISM トポロジソースの一部として使用可能な PI のノードのファクトデータのみが取り込まれます。
- PI から SHR に取り込まれる例外およびサービスのグレードメトリックは、PI で設定されたしきい値に対して計算されています。SHR のライブ収集で取り込まれるデータでは、これらのメトリックを計算するために SHR で設定されたしきい値が活用されます。PI で設定されたしきい値と SHR で設定されたしきい値は一致しない場合があります。
- デフォルトシフトのみが移行されたデータに適用されます。データ移行前にシフトが SHR で設定されることは想定されていません (シフトエンリッチメントは移行されるデータの正常性に影響する可能性があります)。移行後にシフトの設定とデータ収集ができるようになり、関連するシフトが適用されます。
- 現在、管理 UI の汎用データベース設定に関する既知の問題があります。Perf iSPI データベース (または他の外部データベース) がすでに汎用 DB データソースとして設定されていて、PI が移行用に設定されている場合、デプロイされた収集ポリシーで両方のソースからデータが収集される可能性があります。そのような場合、適切なソースで正常に収集が行われます。ただし、そうならなかった場合は、エラーがログファイルに記録され、ABC ストリームのステータスが [WARNING] として表示されます。

# データ移行手順

## 前提条件

PI から SHR への System Performance コンテンツのデータ移行を開始する前に、以下を確認する必要があります。

- PI Timer Service が停止している (移行期間中も停止した状態になっている)。
- SHR Collection および Timer Service が無効になっている。
- SHR 管理 GUI で設定されたシフトがない (デフォルトシフトのみが使用可能)。これは非常に重要です。SHR で設定されたシフトがあると、PI からロードされる集計データの正常性に影響する可能性があります。

## PI移行コンテンツ - パッケージ

PI データ移行パッケージ (HPSHRSmPIMgr.msi) は、メディアまたは Live Network にあります。同じパッケージで RiSM と OM の両方のデプロイメントシナリオに対応します。このパッケージは、System Management のドメインコンテンツ (SysPerf\_Domain) に依存します。

## PIからのデータ移行で実行される手順

### 移行パッケージのインストール

- SHR DVD にある PI 移行パッケージ (HPSHRSmPIMgr.msi) をインストールします。これにより、データ移行パッケージ (Migration\_SM\_PI\Migration\_SM\_PI.ap) が %PMDB\_HOME%\packages フォルダに抽出されます。
- デプロイメントマネージャーを使用してパッケージをデプロイします(データ移行に進む前に Collection Service および Timer Service が実行されていないことを確認します)。

### データソースの設定

次の手順では、データ収集で使用するトポロジおよび PI データベースソースを設定します。

- トポロジ収集に必要なトポロジソースの詳細は、SHR 管理 UI の RiSM/OM トポロジソース設定ページを使用して設定する必要があります。
- HP PI データベースの詳細は、SHR 管理 UI で汎用データベースとして設定する必要があります。

(注: このような場合に設定を追加する「PI の分散デプロイメント」については、このドキュメントの使用事例のセクションを参照してください)

### データ移行スクリプトの実行

データ移行を開始するには、以下のようにコマンド行からデータ移行スクリプトを実行します。

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

SHR のデータフローオーケストレーション (ABC) フレームワークによってデータ移行シーケンスが円滑化されているため、移行に伴う手順のステータスを管理 UI で確認できます。

### 移行されたデータの検証

SHR にインポートされた行数に対して PI で使用可能な行数を検証するユーティリティを使用できます。データを検証するには、以下の手順を実行します。

PI システム (%DPIPE\_HOME%/data) から SHR のパス (%PMDB\_HOME%/data) に systems.xml をコピーします。

```
ovpiupgrade -ovpi <PI *.csv パス> -shr92 <SHR *.csv パス> -xmlPath <マッピング xml への絶対パス> -report <概要レポートのフォルダへの絶対パス>
```

場合によっては、移行された行数が異なることがあります。以下のような原因が考えられます。

- i) PI のノードがトポロジソースで使用できないために調整が失敗する
- ii) PI とトポロジソースの自然キー値 (ノード名など) が一致しない
- iii) ツールで PI の合計ノード数は比較されるが、カスタムノードのフィルタリングが考慮されない (制限事項)

### 関連するログファイル

%PMDB\_HOME%/log フォルダの以下のログファイルには、データ移行パッケージのインストール/実行時に記録された詳細が含まれます。

- packagemanager.log – デプロイメントマネージャーを使用する移行パッケージ (およびこれに依存する SysPerf\_Domain パッケージ) のデプロイメントの詳細
- Topologycollector.log – OM/RtSM トポロジ収集の詳細
- reconcileStep.log – 設定したトポロジソースのディメンションデータに対して行われた調整の詳細
- dbcollector.log – HP PI データベースに対するクエリーの実行と最初の.csv ファイルを生成する前の抽出フェーズに含まれるクエリーの結合に関する詳細
- dwablauncher.log – ABC ストリームの実行の詳細
- loader.log – SHR テーブルに時間別/日別集計データをロードする最後の手順の詳細

### 移行パッケージのアンインストール

データが正常に移行されたら、移行パッケージをアンインストールする必要があります。デプロイメントマネージャーを使用して、データ移行パッケージをアンインストールできます。さらに、PI 移行パッケージ (HPSHRSmPIMgr.msi) もアンインストールする必要があります。これは、データ移行プロセスが完了した場合に、デプロイメントマネージャーで移行パッケージが選択されないようにするためです。

### SHR サービスの再起動と System Performance ETL パッケージのインストール

この時点で、Collection および Timer Service などの SHR サービスを有効にして開始できます。System Performance コンテンツの場合、リアルタイムデータ収集と並行して、Performance Agent からのレポートに必要なトポロジデータを収集できるように ETL パッケージ (SysPerf\_ETL\_PerformanceAgent) をインストールします。

注: PI 移行コンテンツパックでは、ファクトおよびディメンションデータのみが PI から移行されます。設定したトポロジソースからのトポロジ/ディメンションの収集は、関連する ETL パッケージがインストールされて、一連のトポロジ収集が実行された後でのみ行われます。そのため、レポートでデータを表示できるようにするには、Timer および Collection Service を開始しておく必要があります。

## 一般的な使用事例/シナリオ

### 分散環境 – サテライトサーバーおよび中央サーバー

この場合、ユーザーは SHR 管理 UI で PI サテライト/中央サーバーの詳細を設定してから移行ストリームを複数回実行する必要があります。2 つのサテライトサーバーと 1 つの中央サーバーがある場合、ユーザーはまず 1 つ目のサテライトサーバーの詳細を管理 UI で設定し、migrate\_pi\_shr.pro ファイルを実行する必要があります。ストリームの実行が完了したら、2 つ目のサテライトサーバーの詳細を設定してストリームを実行する必要があります。各サテライトおよび中央サーバーで同じ手順を実行します (小規模なデプロイメントの場合、管理 UI で最初に中央およびサテライトサーバーのすべてのソースを設定してから migrate\_pi\_shr.pro ファイルを実行することもできます)。

注: 中央サーバー自体で時間別データを使用できる場合 (HP PI でカスタマーが定義したコピーポリシーによって決まる)、個々のサテライトサーバーからデータをプルする必要はありません。

## よくある質問 [FAQ]

このセクションでは、データ移行プロセスに関する一般的な質問に回答します。

**Q1.** SHR をインストールして数か月間実行した後で PI からデータを移行できますか?他のレポートパックがデプロイされている場合、必要な移行パッケージがすべて使用できるようになるまで移行作業を延期する必要がありますか?

**A1.** PI からのデータの移行は、SHR をインストールして一定期間実行した後でも開始できます。SHR でデータが収集されていない期間のデータが PI から移行されます。たとえば、SHR で過去 1 週間のデータを収集および集計されている場合、それ以前の期間のデータのみが PI から SHR にロードされます。

レポートパックは、段階的に確実に移行できます。したがって、すべてのデータ移行パッケージが使用できるようになるまで待つ必要はありません。

**Q2.** SHR のレポートに使用できるノードのカスタムグループを作成できますか?

**A2.** カスタムグループは、SHR で以下のような xml 形式で定義できます。

```
<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 – カスタムグループの名前
  - type – "CUSTOMGROUP" (推奨) または別の区別型 (OM の VIEWS や NODEGROUPS などの他のグループタイプと区別するため)
  - instances type – グループ化するディメンションテーブルのディメンションインスタンスの名前
  - attribute\_name – インスタンス値が表示されるディメンションテーブルの列名
  - value – グループを作成する列の実際のデータ値
- xml ファイル名の形式は\*customgroup.xml で、xml ファイルは%PMDB\_HOME%/config フォルダに配置される必要があります。
  - .xml を処理するには、以下のコマンドを実行します (.csv ファイルが生成されてステージに移動します)。
    - abcloadNrun -loadBatch -streamId CustomGroup@Platform
    - abcloadNrun -runStream -streamId CustomGroup@Platform
  - 以下のクエリーを使用して、データがステージテーブルにロードされていることを確認します。
    - select \* from K\_CI\_Group\_Bridge\_
  - Core コンテンツパックのディメンションストリームでローダー手順を実行して (Timer 経由で起動)、データウェアハウステーブル K\_CI\_Group\_Bridge に同じデータをロードします。

- SHR レポートの [グループの選択] プロンプトにカスタムグループ値が表示されていることを確認します。

**Q3.** 選択した一連のノード/ノードグループのデータが移行されるように設定できますか？

**A3.** 現在、これは SHR の OM トポロジソースのシナリオでのみ可能です。デフォルトでは、ノードフィルタリングはオフになっています。

- ノードフィルタリングを有効にするには、%PMDB\_HOME%\config\collection.properties ファイルで以下のプロパティを設定する必要があります。  
om.filtering.enabled=true
- SHR に選択的にインポートするノードのリストを含むファイル%PMDB\_HOME%\config\filterednodes.conf を作成します。これには、OM コンソールで表示されるようにノード名/FQDN を含める必要があります。
- SHR に選択的にインポートする OM のグループのリストを含むファイル%PMDB\_HOME%\config\filteredgroups.conf を作成します。
- 両方のファイルが存在していて値も有効な場合、%PMDB\_HOME%\config\filterednodes.conf ファイルで指定したノードとともに、%PMDB\_HOME%\config\filteredgroups.conf の各グループに属するすべてのノードが SHR にインポートされます。

## 参考文献

### PIのSystem Resource Report PackとSHRのSystem Performanceコンテンツ間のデータマッピング

このセクションでは、PI と SHR 間のデータベーステーブル/列のマッピングについて説明します。SHR レポートで使用されない列 (データはPIから移行される) は、濃い赤で表示されています。

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 ("FALSE"に設定)
	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
	CPUFamily
	CI_UID (トポロジソースから取得 - PI の

	NODE_NAME は調整に使用される)
	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 ("1"に設定)
	OSPatch
	Memory_Unreserved
	Cluster_Name
	Hypervisor_View

PI: SH_SR_SysXcep	SHR: SH_SM_Node_Res
	dsi_key_id(自動生成)
ta_period	ta_period
K_Node.K_Location.dsi_key_id	LocationRef
	ShiftRef (自動生成)
AVGrunq	avgRunQ
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
<b>PI: SH_SR_SysVolXcep</b>	<b>SHR: SH_SM_Node_Res</b>
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
	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

PI: SD_SR_SysXcep	SHR: SD_SM_Node_Res
	dsi_key_id(自動生成)
ta_period	ta_period
K_Node.K_Location.dsi_key_id	LocationRef
	ShiftRef (自動生成)
AVGrunq	avgRunQ
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
<b>PI: SD_SR_SysVolXcep</b>	<b>SHR: SD_SM_Node_Res</b>
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
	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

<b>PI: SH_SR_SysUp</b>	<b>SHR: SH_SM_Node_Avail</b>
	dsi_key_id(自動生成)
ta_period	ta_period
K_Node.K_Location.dsi_key_id	LocationRef
	ShiftRef (自動生成)
TOTUPTIME	totuptime
TOTDOWNTIME	totdowntime
	Totplandtime
	Totexcdtime
	Totunknowntime
	Totavailability

<b>PI: SD_SR_SysUp</b>	<b>SHR: SD_SM_Node_Avail</b>
	dsi_key_id(自動生成)
ta_period	ta_period
K_Node.K_Location.dsi_key_id	LocationRef
	ShiftRef (自動生成)
TOTUPTIME	totuptime
TOTDOWNTIME	totdowntime
	Totplandtime
	Totexcdtime
	Totunknowntime
	Totavailability

PI: K_System_CPU	SHR: K_SM_CPU
	dsi_key_id(自動生成)
CPUID	cpu_id
(K_Node.node_name(K_System_CPU.node_fk))	node_name
	Systemref(自動生成 (ciid を使用 - node_name を使用して調整))
	cpu_vendor
	cpu_speed

PI: SH_SR_CPU	SHR: SH_SM_CPU
	dsi_key_id(自動生成)
ta_period	ta_period
K_Node.K_Location.dsi_key_id	LocationRef
	ShiftRef (自動生成)
AVGCPUUTIL	avgTotUtil
AVGCPUSYSTEMMODE	avgSysModeUtil
AVGCPUUSERMODE	avgUsrModeUtil
AVGINTRATE	avgInterruptRate
AVGCSRATE	avgCntxtSwitchRate
	maxTotUtil
	maxSysModeUtil
	maxUsrModeUtil
	maxInterruptRate
	maxCntxtSwitchRate

PI: SD_SR_CPU	SHR: SD_SM_CPU
	dsi_key_id(自動生成)
ta_period	ta_period
K_Node.K_Location.dsi_key_id	LocationRef
	ShiftRef (自動生成)
AVGCPUUTIL	avgTotUtil
AVGCPUSYSTEMMODE	avgSysModeUtil
AVGCPUUSERMODE	avgUsrModeUtil
AVGINTRATE	avgInterruptRate
AVGCSRATE	avgCntxtSwitchRate
	maxTotUtil
	maxSysModeUtil
	maxUsrModeUtil
	maxInterruptRate
	maxCntxtSwitchRate

PI: K_Disk_Dsk	SHR: K_SM_PhysicalDISK
	dsi_key_id(自動生成)
prop_disk_name	disk_name
(K_Node.node_name(K_System_CPU.node_fk))	node_name
	Systemref(自動生成 (ciid を使用 - node_name を

	使用して調整))
	Dir_name

PI: SH_SR_Disk	SHR: SH_SM_Disk
	dsi_key_id(自動生成)
ta_period	ta_period
K_Node.K_Location.dsi_key_id	LocationRef
	ShiftRef (自動生成)
AVGDISKUTIL	avgPctUtil
AVGPHYSICALIORATE	avgPhyIORate
AVGPHYSREADRATE	avgPhyReadRate
AVGPHYSWRITERATE	avgPhyWriteRate
AVGSYSTEMIORATE	avgSysIORate
AVGVMIORATE	avgVMIORate
	avgPhyByteRate
	avgPhyReadByteRate
	avgPhyWriteByteRate
	avgRawReadRate
	avgRawWriteRate
	maxPctUtil
	maxPhyIORate
	maxPhyReadRate
	maxPhyWriteRate
	maxSysIORate
	maxVMIORate
	maxPhyByteRate
	maxPhyReadByteRate
	maxPhyWriteByteRate
	maxRawReadRate
	maxRawWriteRate

PI: SD_SR_Disk	SHR: SD_SM_Disk
	dsi_key_id(自動生成)
ta_period	ta_period
K_Node.K_Location.dsi_key_id	LocationRef
	ShiftRef (自動生成)
AVGDISKUTIL	avgPctUtil
AVGPHYSICALIORATE	avgPhyIORate
AVGPHYSREADRATE	avgPhyReadRate
AVGPHYSWRITERATE	avgPhyWriteRate
AVGSYSTEMIORATE	avgSysIORate
AVGVMIORATE	avgVMIORate
	avgPhyByteRate
	avgPhyReadByteRate
	avgPhyWriteByteRate
	avgRawReadRate
	avgRawWriteRate

	maxPctUtil
	maxPhyIORate
	maxPhyReadRate
	maxPhyWriteRate
	maxSysIORate
	maxVMIORate
	maxPhyByteRate
	maxPhyReadByteRate
	maxPhyWriteByteRate
	maxRawReadRate
	maxRawWriteRate

PI: K_NetInterface_NetIf	SHR: K_SM_NetInterface
	dsi_key_id(自動生成)
prop_netif_name	Interface_name
(K_Node.node_name(K_System_CPU.node_fk)	Node_Name
	Systemref(自動生成 (ciid を使用 - node_name を使用して調整))
	Network_speed

PI: SH_SR_NetInterface	SHR: SH_SM_NetInterface
	dsi_key_id(自動生成)
ta_period	ta_period
K_Node.K_Location.dsi_key_id	LocationRef
	ShiftRef (自動生成)
	avgPktRate
AVGINPACKETS	avgInPktRate
AVGOUTPACKETS	avgOutPktRate
	avgByteRate
AVGINBYTES	avgInByteRate
AVGOUTBYTES	avgOutByteRate
AVGCOLLISIONRATE	avgCollisionRate
AVGERRORRATE	avgErrorRate
	NetworkUtil
	maxPktRate
	maxInPktRate
	maxOutPktRate
	maxByteRate
	maxInByteRate
	maxOutByteRate
	maxCollisionRate
	maxErrorRate

PI: SD_SR_NetInterface	SHR: SD_SM_NetInterface
	dsi_key_id(自動生成)
ta_period	ta_period
K_Node.K_Location.dsi_key_id	LocationRef
	ShiftRef (自動生成)
	avgPktRate
AVGINPACKETS	avgInPktRate
AVGOUTPACKETS	avgOutPktRate
	avgByteRate
AVGINBYTES	avgInByteRate
AVGOUTBYTES	avgOutByteRate
AVGCOLLISIONRATE	avgCollisionRate
AVGERRORRATE	avgErrorRate
	NetworkUtil
	maxPktRate
	maxInPktRate
	maxOutPktRate
	maxByteRate
	maxInByteRate
	maxOutByteRate
	maxCollisionRate
	maxErrorRate

PI: K_Customer	SHR: K_Customer
	dsi_key_id(自動生成)
cust_id	Customer_ID
cust_name	Name Display_Name
	User_Key
	Address1
	Address2
	City
	State
	ZIP_Code
	Phone_Number
	Description

PI: K_SR_System, K_Node, K_Customer	SHR: K_CI_Cust_Bridge
	dsi_key_id(自動生成)
K_SR_System.node_fk(CI_UID を取得するように調整)	CI_Key
K_Customer.cust_name	Cust_Key
	Valid_Period_Start
	Valid_Period_End

PI: K_Location	SHR: K_Location
	dsi_key_id(自動生成)
	Country ('Default')
	State ('Default')
	City ('Default')
location_name	Region
	Address ('Default')
	Building ('Default')
	Floor ('Default')
location_name	Name

PI: K_SR_System, K_Node, K_Location	SHR: K_CI_Loc_Bridge
	dsi_key_id(自動生成)
K_SR_System.node_fk(CI_UIDを取得するように調整)	CI_Key
K_Location.location_name	Loc_Key
	Valid_Period_Start
	Valid_Period_End