# HP Service Health Reporter

for the Windows® operating system

Software Version: 9.20

## Content Development - Getting Started Guide

Document Release Date: October 2012
Software Release Date: October 2012

# 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 2010-2012 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.

Java is a registered trademark of Oracle and/or its affiliates.

## 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:

**www.hp.com/go/hpsoftwaresupport**

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 HP 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 user 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**

1 Introduction . . . 7
About this Guide . . . 7
Who Should Read this Guide . . . 7
Prerequisites and Reference Documentation . . . 8

2 Content Pack Architecture and Development . . . 9
Content Pack Architecture . . . 9
Components of a Content Pack . . . 9
Domain Component . . . 10
Extraction Transformation Loading (ETL) Component . . . 11
Reports Component . . . 12
Introducing the Content Development Environment . . . 12

3 Creating a Content Pack - Getting Started . . . 13
Preparing for Content Development . . . 14
Study the Business Domain . . . 14
Identify User Personas and Reporting Requirements . . . 14
Extract and Configure CDE . . . 14
Creating and Installing the Domain Component . . . 15
Create the Directory Structure . . . 15
Identify the Grain, Dimensions, and Facts . . . 16
Design the Data Model . . . 16
Implement the Data Model in XML . . . 17
Create the Workflow Streams . . . 18
Use CDE to Generate Domain Component Package . . . 18
Install the Domain Component Package . . . 19
Creating and Installing the ETL Component . . . 20
Custom Data Loading by Using .csv Files . . . 20
Verifying Workflow Streams on Administration Console . . . 21
Creating and Installing the Reports Component . . . 22
Prerequisites . . . 23
Create the Directory Structure . . . 23
Write the Model XML Document . . . 23
Create the Manifest XML File . . . 23
Create a Connection to SAP BusinessObjects . . . 24
Use CDE to Generate SAP BusinessObjects Universe . . . 24
Export the Universe to SAP BusinessObjects Repository . . . 25
Create Web Intelligence Reports . . . 27
Enabling Time Drill Option in Reports . . . 28
Viewing Reports on SAP BusinessObjects InfoView . . . 28
A Demonstration of Retail Sales Report . . . 29
Exporting Reports to SHR System . . . 29
Export Business Intelligence Archive Resource (BIAR) File . . . 29
Create the Manifest XML File . . . 34
Use CDE to Generate Reports Component Package . . . 34
Install the Reports Component Package . . . 35

4 Creating an ETL Component Package . . . 37
Prerequisite . . . 38
Analyze the Data Source . . . 38
Create the Directory Structure . . . 38
Define Collection Policy in XML . . . 38
Define Data Transformation Rules . . . 39
Define Stage Rules . . . 39
Define Workflow Streams . . . 39
Generate the ETL Component . . . 40
Install the ETL component . . . 40
Working with the ETL Component . . . 40
Configure a Generic Database . . . 40
Verify the ETL Component . . . 42
View Reports . . . 42

A Appendix: Creating a Data Source for ETL Component . . . 43
Create a PostgreSQL Database . . . 44
Create Database Tables . . . 44
Insert Data into the Database . . . 45

Glossary . . . 47

We appreciate your feedback! . . . 49

# 1 Introduction

This guide provides an overview of HP Service Health Reporter (SHR) content development and describes the process of creating a content pack. You will use sample files available in your SHR installation media and the instructions in this guide to create a sample content pack.

## About this Guide

This guide contains the following:

1. Introduction (this chapter): Provides the prerequisites and references helpful in understanding content packs and content development terminologies used in this guide.
2. Content Pack Architecture and Development: Describes the architecture of content packs and introduces the Content Development Environment (CDE). The CDE is a set of tools provided by SHR that enable you to develop a content pack.
3. Creating a Content Pack - Getting Started: Provides step-by-step instructions to create a content pack by using an example of Retail Point of Sales. You will use the sample files and templates available in the SHR media to create a working content pack.
4. Creating an ETL Component Package: Provides instructions and sample files to create a generic ETL content pack component, with limited capabilities, for the Retail Point of Sales example.
5. Glossary of terms.

## Who Should Read this Guide

This guide is meant for developers who want to create content packs on SHR or extend existing content packs.

# Prerequisites and Reference Documentation

This guide assumes you have a prior understanding of the following:

| Prerequisite | Reference Documentation |
|---|---|
| HP Service Health Reporter concepts and usage | Read the following documents available at `Start > Programs > HP Software > SH Reporter > Documentation`<br>• ***Concepts Guide*:** This guide explains the key concepts, architecture, and typical workflow of SHR. Read this guide to understand the concept and working of content packs before you start developing any.<br>• ***Installation and Configuration Guide:*** In this guide you find instructions to install content packs and steps to troubleshoot problems during installation of content packs.<br>• ***Online Help for Administrators*:** In this Help you find information about monitoring the installed content packs.<br>• ***Online Help for Users:*** In this Help you find information about the out-of-the-box content packs provided by SHR. |
| Data Warehouse concepts | You can find resources related to data warehouse concepts and examples on the internet. SHR does not recommend any particular resource. |
| SAP BusinessObjects reporting concepts | • ***SAP BusinessObjects Enterprise InfoView User's Guide:*** This guide is available at `Start > Programs > BusinessObjects > BusinessObjects Enterprise > Documentation`. This guide provides instructions to create and work with WebIntelligence reports.<br>• ***SAP BusinessObjects Universe Designer Online Help:*** In this Help you find information about creating, building, and managing Universes. You can launch the Help from the Universe Designer user interface.<br>For additional information and the latest help documents, see **http://help.sap.com/businessobject/product_guides/**. |
| XML concepts and how to create XML documents | You can find resources related to XML concepts and examples on the internet. SHR does not recommend any particular resource. |

# 2 Content Pack Architecture and Development

SHR enables you to create the following content on the performance management database platform:

- **Content pack:** You can create new content packs and extend the out-of-the-box content packs provided by SHR. This guide uses an example to describe the steps to create a content pack.
- **WebIntelligence reports:** Using the SAP BusinessObjects InfoView application interface you can create new reports and customize the out-of-the-box reports provided by SHR.

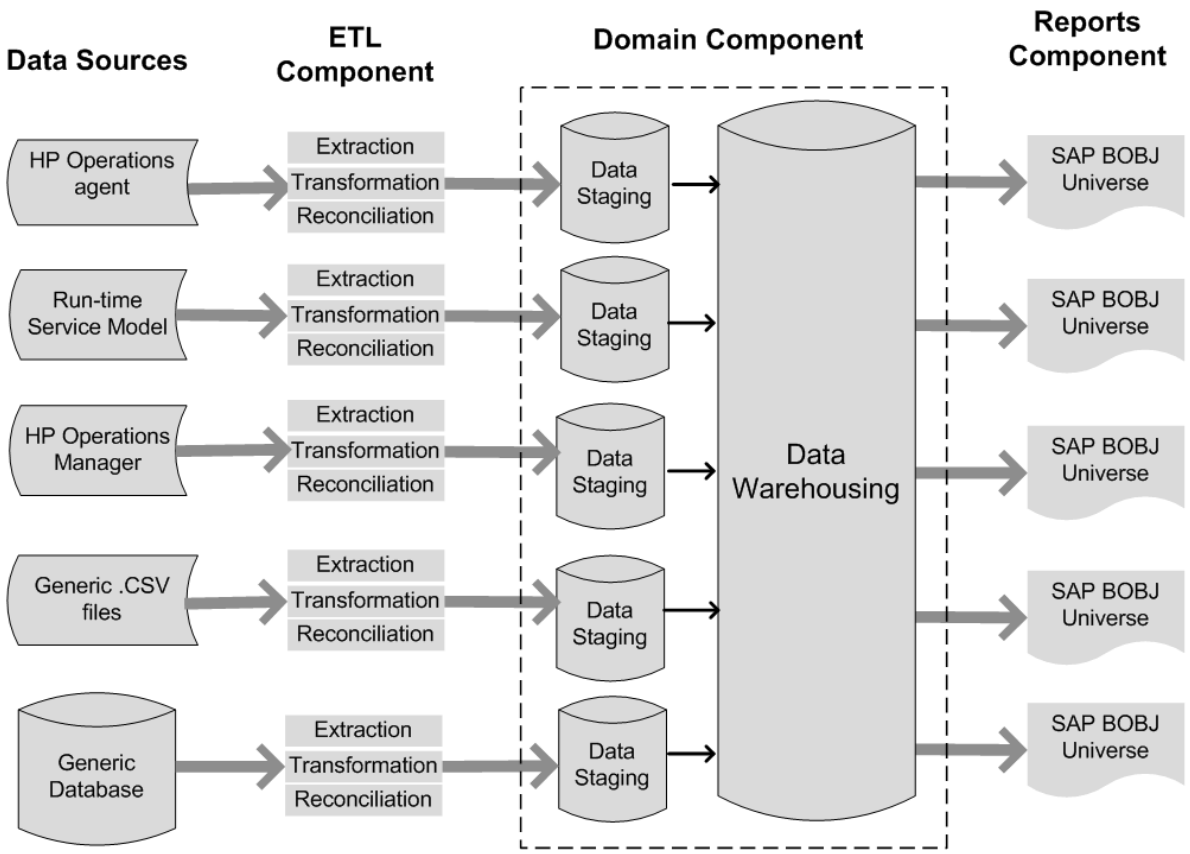
## Content Pack Architecture

A content pack is a domain or application specific data mart deployed on the SHR performance management database platform. Content packs determine the metrics to be collected, how to process and store the metrics, and display the processed data on the reports.

### Components of a Content Pack

A typical content pack consists of three components - the Domain, Extraction Transformation Loading (ETL), and Reports components.

The following figure shows the typical data flow between the components.




**Note:** The Transformation and Reconciliation steps in an ETL component are optional and may not be applicable for all data sources.

## Domain Component

The Domain component defines the data model of the domain you are reporting on along with the logic to perform processing on the data. It requires domain experts to model the data according to the business requirements. This component is independent of the data source. The Domain component includes:

- The data model consisting of the facts and dimensions of the domain you are reporting on and the relationship between them.
- Workflow streams that control and monitor the processing of the data. A stream is made up of steps that are related to one another in a sequential relationship. A content pack contains a set of workflow streams that define and control the flow of data from one step to another. In the Domain component of a content pack, workflow streams are defined in XML files to load the data into tables and perform offline aggregation of the data.
- Optionally defines the dimensions and cubes for the business view to be used by one or more Reports components.

## Extraction Transformation Loading (ETL) Component

The ETL component is data source dependant; it defines the collection of data from the specified data source, followed by transformation and loading of the data into the data warehouse. Therefore, for a particular domain, each data source application has a separate ETL content pack component. Before you start creating the ETL component you must identify the data source that provides the metrics suitable to be fed into the domain data model. The ETL component includes the following:

- **Data Collection (Extraction) rules:** After you identify a data source you must create a collector program or use an existing collector program to collect the required facts and dimensions from the data source. A collection policy must be written in XML to define the metrics to be collected by the collector program from the data source. The collector program collects data as defined in the collection policy and places this data into .csv files.

  SHR supports data collection from a set of known data sources and provides collector programs for each such data source.

  The data sources supported by SHR are:

  — Run-time Service Model (RTSM),

  — HP Operations agent,

  — HP Operations Manager,

  — HP Business Service Management Profile Database,

  — Generic .csv files, and,

  — Databases supporting JDBC.

  Depending upon your business requirements, you can create your own collector program or use the out-of-the-box collectors by writing the corresponding collection policy XML files.

- **Data Transformation rules (optional):** Data transformation rules are required if the collected data, as .csv files, need to be transformed before loading the data into the datawarehouse. For example, you can write a rule to remove the rows that contain an empty value in the 'host name' column. Transformation rules are written in XML files. SHR provides a data transformation utility called the 'mapper' utility for the out-of-the-box transformation rules.

- **Data Reconciliation rules (optional):** Data reconciliation is the process of associating fact data to the corresponding dimension data. In SHR data reconciliation rules are written to associate fact data from one source to the corresponding dimension data from another source using common business keys. For example, in the Service and Operations Bridge (SaOB) deployment, the dimension data is collected from RTSM and the fact data is collected from HP Operations agent. Reconciliation rules are written in XML to reconcile the fact data with the dimension data.

- **Data Staging rules:** After the data (in the form of .csv files) is collected, transformed, and reconciled, it is moved to staging tables. Data staging rules define how the data must be moved to staging tables including the process to merge columns and rows.

- **Workflow Stream definitions:** In the ETL component, workflow streams are defined in XML to control the data movement from collection to stage through transformation and reconciliation steps if required.

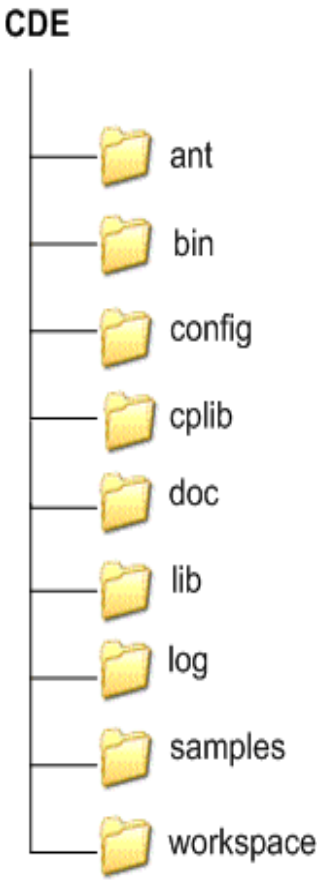
### Reports Component

The Reports component contains the SAP BusinessObjects Web Intelligence reports and universes. A content pack universe provides a business-oriented meaningful mapping of the underlying complex database and simplifies the creation of reports. It is a logical view of the underlying data model that you define in the Domain component. The Reports component imports the dimensions and cubes defined in the corresponding Domain component.

## Introducing the Content Development Environment

The Content Development Environment (CDE) consists of a set of tools that you use during the process of content development. These tools use XML files authored by the content pack developer to generate the installable content pack component packages.

The CDE tools are provided in the SHR media as a self-extracting `CDE.exe` file in the <*installation_directory*>. Upon extraction of the file contents CDE provides a folder structure as shown in the following figure.




**ant**: contains Apache Ant, a Java build tool.

**bin**: contains batch scripts used for creating content packs

**config**: contains configuration files required by CDE tools.

**cplib**: contains out-of-the-box Domain content packs provided by SHR.

**doc**: contains the XML Schema Definition (XSD) files and other reference documentation for content development.

**lib**: contains Java library files for use by CDE.

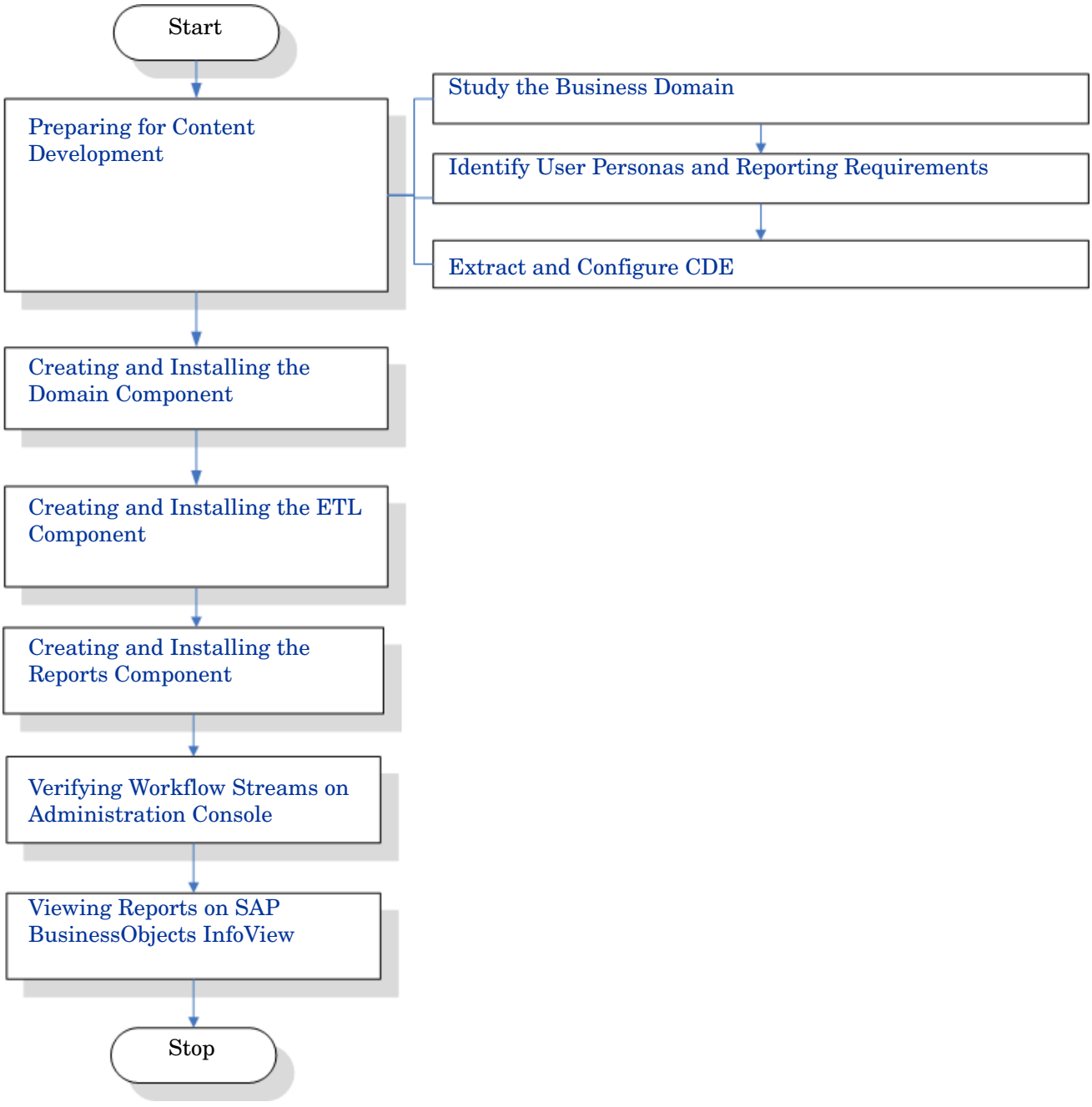
**log**: contains log files generated by CDE tools.

**samples**: contains a source folder within 'RetailPOS Demo Pack' that has the source files to create sample content pack

**workspace**: contains the user-created content pack source files and installable packages.

# 3 Creating a Content Pack - Getting Started

This chapter describes the steps to create a sample content pack by using a real world example related to the retail sales industry. The following flowchart shows the typical steps to create a content pack.



# Preparing for Content Development

## Study the Business Domain

Consider a large electronics retail chain with:

- 100 stores spread across five locations, and,
- each store having about 10,000 individual products on its shelves, the individual products identified by Stock Keeping Units (SKUs).

The retail chain is automated and each product has a scanner label attached to it. The collection of transaction data is done mainly at the Point-of-Sale (PoS) system at the front door of the store where the bar codes are scanned and directly entered into the system. The customer takeaway is measured at this place.

After you understand the retail store chain business you can proceed to determine the business user personas and their respective reporting requirements.

## Identify User Personas and Reporting Requirements

In the Retail Point of Sales example, we consider the business management personnel as the users of the reports.

The management users are interested in a sales summary report that shows the product sales information for different product categories in stores located across different locations. The sales information must be available across time periods such as yearly, quarterly, monthly and daily.

It is recommended to create a design mock-up of the required reports at this stage. This initial mock-up can be created on paper or any design tool of choice. The actual development of the Web Intelligence reports by using SAP BusinessObjects can be done when you create the Reports component package.

## Extract and Configure CDE

1. Logon to the host system where SHR is installed. In the <*installation_directory*> you find an application named `CDE.exe`.
2. Run the file `CDE.exe` to extract the CDE tools into the `CDE` folder.
3. Type **`cmd`** and press **`ENTER`** to open the Command Prompt window.
4. At the command prompt run the **`cd`** command to navigate to the directory where SHR is installed. Then run the following commands to change the directory:
   a. <*installation_directory*> **`cd`** CDE which sets the directory at <*installation_directory*>`\CDE`.
   b. <*installation_directory*>`\CDE`> cd bin which sets the directory at <*installation_directory*>`\CDE\bin`.

Here <*installation_directory*> is the directory where SHR is installed.

5. Run the following command:

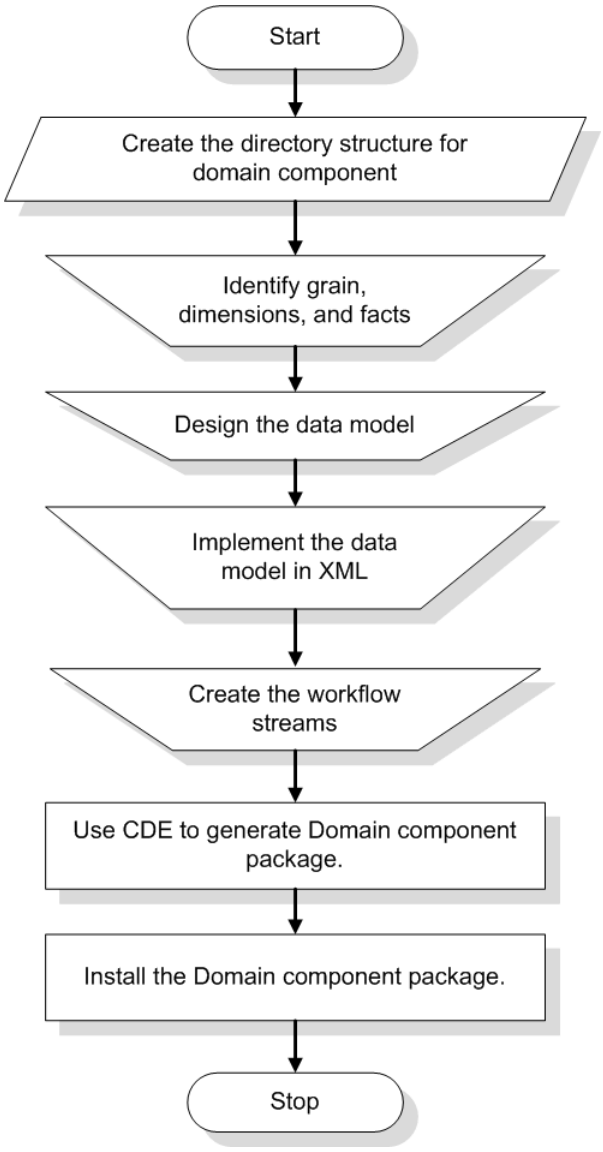
   **`setenv.bat`**

   The following environment variables are set in the path.

```
CDE_HOME
ANT_HOME
JRE_HOME
```

# Creating and Installing the Domain Component

The following flowchart shows the typical steps to create the Domain component of a content pack.




## Create the Directory Structure

To create the directory structure for the Domain component source files, at the command prompt run the following command:

```
<installation_directory>\CDE\bin>CreateCPFolders.bat -package RetailPOS
-subpackage RetailPOSDomain -type domain
```

where,

- *<installation_directory>* is the directory where you have installed SHR,
- **RetailPOS** is the name of the content pack you are developing, and,
- **RetailPOSDomain** is the name of the Domain component within **RetailPOS**.

The directory will contain templates provided by SHR that you will use to create the Domain component source files — the model XML and the workflow streams XML files.

SHR provides sample source files for the Retail POS content pack at the following location. You can use these files as reference to create your own content pack.

```
%CDE_HOME%\samples\RetailPOS_Demo_Content_Pack\Source\
```

## Identify the Grain, Dimensions, and Facts

The grain of the fact table is the most granular data. In the Retail Point of Sales Domain example, the grain is an individual line item on a Point of Sales transaction.

The dimensions are:

- Date
- Product
- Store

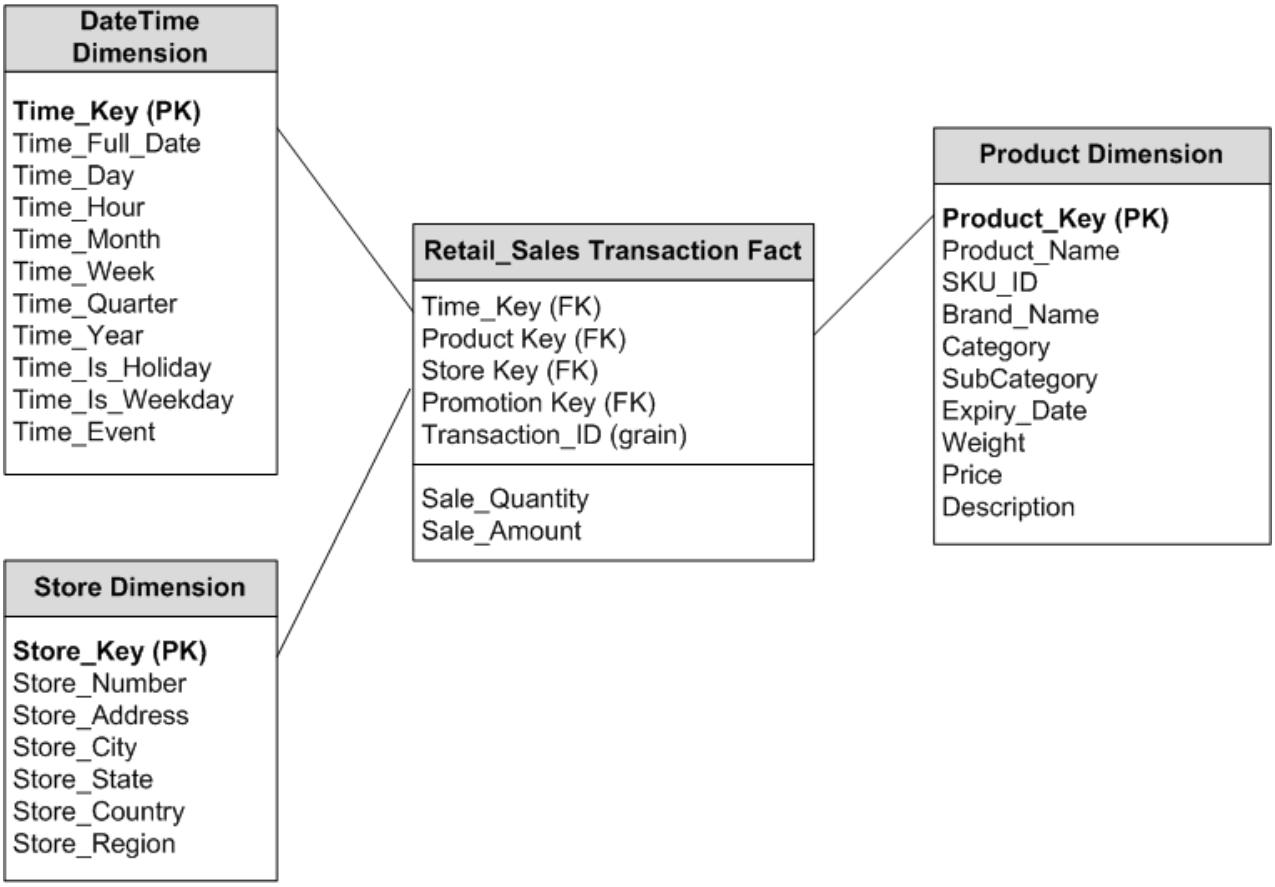
The facts collected by the Retail Point of Sales system are:

- Quantity of Sales
- Amount of Sales

## Design the Data Model

A data model illustrates the relationship between entities (facts and dimensions tables) and their attributes (database table columns). In SHR, a data model is an XML file. To create a data model start by creating a schema diagram and then implement the same into an XML file.

The following figure shows the schema diagram that we will use to create the data model for the **RetailPOS** Content Pack.




## Implement the Data Model in XML

The schema that you designed in the previous step must be implemented using XML to be used by the CDE to create the Domain component package. This XML file is called Model XML.

A typical Model XML file has the following sections:

- **Relational section** defines the fact tables and dimension tables, and the relationship between the facts and dimensions.
- **Logical section** defines the cubes, hierarchies, and levels. You define one cube for every fact table.
- **Aggregate section** defines the aggregation to be performed on the source tables.

After you create the Domain component directory structure you find a template `model_template.xml` at the following path:

```
%CDE_HOME%\workspace\RetailPOS\RetailPOS.ap\source\model
```

You can edit this `.xml` file to author the Model XML.

Also, for reference you can find a sample Model XML file for the **`RetailPOSDomain`** available at the following location:

```
%CDE_HOME%\samples\RetailPOS_Demo_Content_Pack\Source\RetailPOS\RetailPOSDomain.ap\source\model
```

To read the contents of the sample Model XML file, on this PDF document click the **Attachments: View File Attachments** tab and select `RetailPOS_dimension_model.xml`. Double click to open the XML file on your browser window.

## Create the Workflow Streams

SHR provides a workflow framework to control and monitor the workflow of the data processes. This framework is made up of workflow streams. A stream is made up of steps that are related to one another in a sequential relationship. Each content pack contains a set of streams that define and control the flow of data from one step to another.

As an example, the Domain component may have the following steps in the workflow stream:

**`Loading data to rate table`** –> **`Hourly aggregation`** –> **`Daily aggregation`**

You must implement the workflow streams in XML to be used by CDE to create the Domain component package. The streams that you must create using XML are:

- One workflow stream XML to load and aggregate the fact.
- One workflow stream XML to load the **`Store`** dimension.
- One workflow stream XML to load the **`Product`** dimension.

After you create the directory structure for the Domain component you find a workflow stream template file named `ABC_stream_template.xml` at the following path:

`%CDE_HOME%\workspace\RetailPOS\RetailPOS.ap\source\orchestration`

You can edit this `ABC_stream_template.xml` file to author the workflow streams XML files.

Also, as reference you can find sample workflow streams XML files for the **`RetailPOSDomain`** at the following location:

`%CDE_HOME%\samples\RetailPOS_Demo_Content_Pack\Source\`

To read the contents of the sample workflow stream XML files, on this PDF document click the **Attachments: View File Attachments** icon and select `the following .xml` files:

- `Fact_Retail_Sales_stream.xml:` Workflow stream XML to load and aggregate the fact.
- `Dimension_Store_stream.xml:` Workflow stream XML to load the **`Store`** dimension.
- `Dimension_Product_stream.xml:` Workflow stream XML to load the **`Product`** dimension.

Double click to open the XML files on your browser window.

## Use CDE to Generate Domain Component Package

To generate the Domain component package, follow these steps:

1. **Create Manifest XML file**

   The Manifest XML file contains the definitions of the source files to be used by CDE to generate the Domain component package.

   To create the Manifest XML file using CDE:

   a. Using the **`cd`** command change directory to:

      `%CDE_HOME%\workspace\RetailPOS\RetailPOSDomain.ap`

b Run the following command:

ant **createManifestTemplate**

The Manifest XML file named RetailPOSDomain_manifest_template.xml is created in %CDE_HOME%\workspace\RetailPOS\RetailPOSDomain.ap\source

You can see a sample Manifest XML file for the **RetailPOSDomain** at the following location:

%CDE_HOME%\samples\RetailPOS_Demo_Content_Pack\Source\RetailPOS\RetailPOSDomain.ap\source\RetailPOSDomain_manifest_template.xml

To read the contents of the sample Manifest XML file, on this PDF document click the **Attachments: View File Attachments** icon and select RetailPOSDomain_manifest_template.xml. Double click to open the XML file on your browser window.

2 **Build Domain component package**

To create the Domain component package using CDE:

a Using cd command change directory to:

%CDE_HOME%\workspace\RetailPOS\RetailPOSDomain.ap

b Run the following command:

ant

The installable Domain component package is created in

%CDE_HOME%\workspace\RetailPOS\RetailPOSDomain.ap\dist

3 Browse to %CDE_HOME%\workspace\RetailPOS\RetailPOSDomainCP.ap\dist

4 Copy RetailPOS to %PMDB_HOME%\packages. Copying the package makes it available in the Deployment Manager for installation.

It is not necessary to have SHR installed on the machine where you are creating the Domain component package. If you created the Domain component package on another machine, you must copy the package to the SHR machine under %PMDB_HOME%\packages.

## Install the Domain Component Package

SHR provides the Deployment Manager utility on the Administration Console to install the content pack component packages. For instructions on how to install the content pack components, see the *HP Service Health Reporter Installation and Configuration Guide.*

# Creating and Installing the ETL Component

The ETL component consists of data collection, transformation, reconciliation, and staging rules. The creation of the complete ETL component using all the rules can be quite complex. Therefore, to enable you to create a sample content pack this chapter describes a simple alternative method to generate data in the form of .csv files and load into the datawarehouse.

The following chapter Creating an ETL Component Package on page 37 describes the creation of an ETL component package that you can install by using the Deployment Manager on the Administration Console.

## Custom Data Loading by Using .csv Files

In this method you create a set of .csv files in the required format and place the .csv files into the `%PMDB_HOME%\stage` folder to be loaded into the SHR datawarehouse tables.

Perform these steps to create and load the .csv files.

1. **Install the Domain component package:** Before you begin creating the .csv files for loading, make sure you generate the Domain component package you created in Install the Domain Component Package on page 19. The Domain component creates a `stage interface` html file that contains the format in which the .csv files must be created. The stage interface html file is created in `%PMDB_HOME%\packages\RetailPOS\RetailPOSDomainCP.ap\doc.`

   For reference, see the example `RetailPOSDomain_INTERFACE.html` file available in `%CDE_HOME%\samples\RetailPOS_Demo_Content_Pack\RetailPOS\RetailPOSDomain.ap\doc`

2. **Generate .csv files:** SHR provides a simple .csv files generator program to create sample .csv files for the RetailPOS content pack. To create the .csv files in the given format provided in the template .html file by using the `.csv file` generation program follow these steps:

   a. Browse to the folder `%CDE_HOME%\samples\RetailPOS_Demo_Content_Pack\Sample CSV Generator.`

   b. Copy the following files to the specified location as shown in the following table:

| File | Copy to location |
|---|---|
| `retailpos_csvgen.exe` | `%PMDB_HOME%\bin` |
| `retailpos_csvgen.ini` | `%PMDB_HOME%\config\startup` |
| `retailposcsvgen.jar` | `%PMDB_HOME%\lib` |

   c. At the command prompt run `retailpos_csvgen.exe.`

For the sample RetailPOS content pack that you are creating, the .csv file generator program generates two months worth of .csv files and places these files in the `%PMDB_HOME%\stage` folder. The Domain component that you installed earlier loads the `.csv` files to the SHR datawarehouse tables.

# Verifying Workflow Streams on Administration Console

After the `HP_PMDB_Platform_Timer` service is started, you logon to the Administration Console to check the status of the Domain component workflow streams. Follow these steps:

1. On the Administration user interface, click **Internal Monitoring** > **Data Processing**.
2. On the **Stream Details** tab, view the status of streams in the RetailPOSDomain content pack. All streams must show a status of OK to indicate successful completion.

In the RetailPOS example, the Domain component has the following workflow streams with one or more steps within each stream.

- A workflow stream XML to load and aggregate the **`Retail_Sales`** fact.
- A workflow stream XML to load the **`Store`** dimension.
- A workflow stream XML to load the **`Product`** dimension.
- A workflow stream XML to load the **`Promotion`** dimension.

As shown in the following figure, successful completion of the streams is indicated using green color.

**Data Processing**

Stream Details | Historical Stream Overview | Historical Stream Details

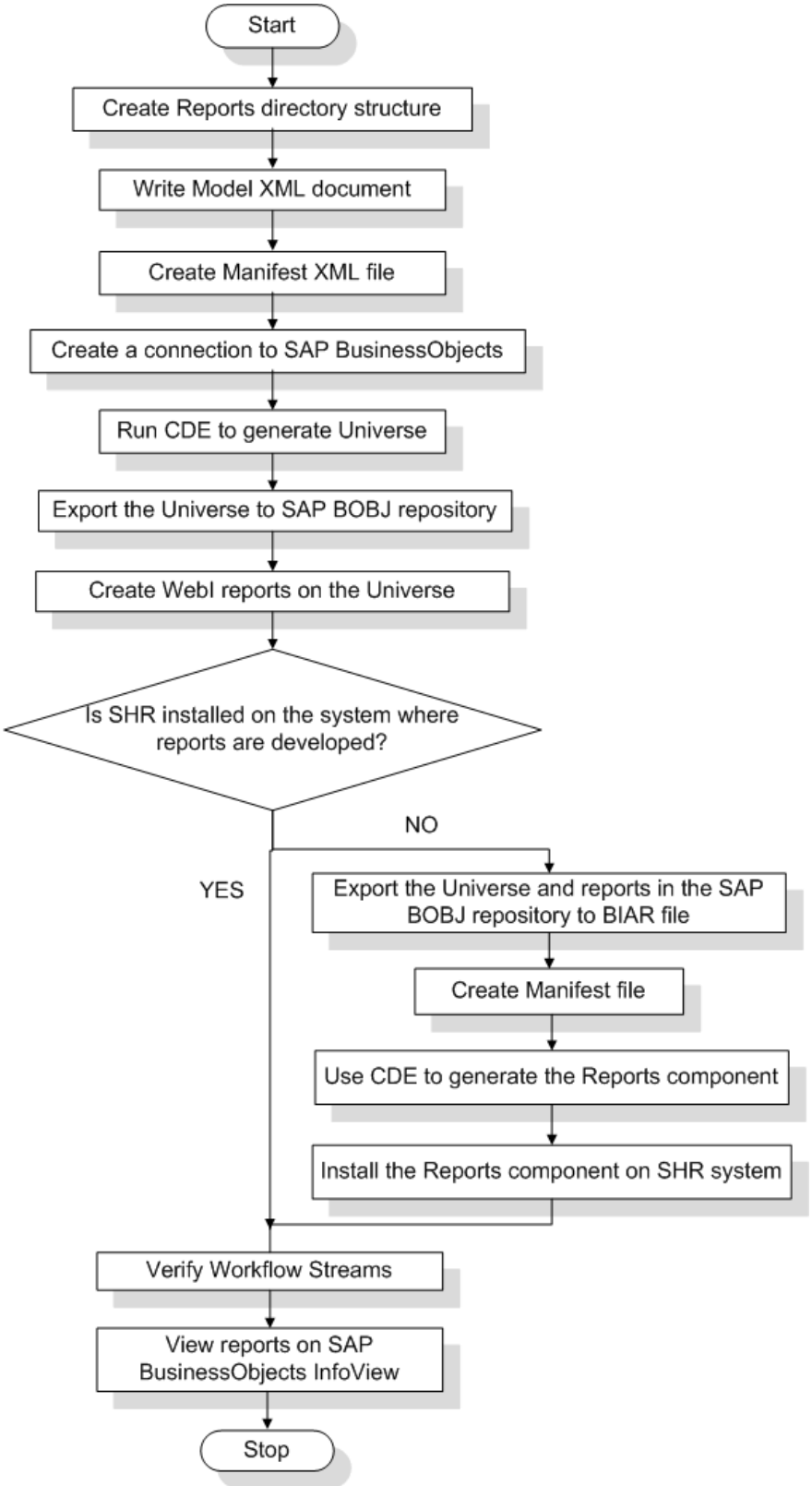
| Content Pack Component name | Number of Streams | Stream Status Details | | | |
|---|---|---|---|---|---|
| | | OK | Warning | Error | Total |
| Core | 26 | 26 | 0 | 0 | 26 |
| RetailPOSDomain | 5 | 5 | 0 | 0 | 5 |
| ETL_SM_VI_Sol_Zones_PA | 0 | 0 | 0 | 0 | 0 |
| ETL_SystemManagement_SiS | 0 | 0 | 0 | 0 | 0 |
| SystemManagement | 0 | 0 | 0 | 0 | 0 |
| MSAppCore | 1 | 1 | 0 | 0 | 1 |

**Stream Detail for Content Pack Component : RetailPOSDomain**

| Stream Name | Step Status(Completed/Total) | Step Status | Start Time |
|---|---|---|---|
| RetailPOSDomain@Product | 1/1 | SUCCESS | May 25, 2012 12:30:11 PM |
| RetailPOSDomain@Store | 1/1 | SUCCESS | May 25, 2012 12:30:11 PM |
| RetailPOSDomain@Downtime | 1/3 | SUCCESS | May 25, 2012 12:30:11 PM |
| RetailPOSDomain@Retail_Sales | 1/4 | SUCCESS | May 25, 2012 12:30:12 PM |
| RetailPOSDomain@Promotion | 1/1 | SUCCESS | May 25, 2012 12:30:11 PM |

# Creating and Installing the Reports Component

The following flowchart shows the steps to create the Reports component.



## Prerequisites

Before you create the Reports component, make sure the following are done:

- CDE is installed on the same machine where you have installed SHR and SAP BusinessObjects.
- The Domain component that you created in Install the Domain Component Package on page 19 is installed. Use the Deployment Manager utility on the Administration Console to install the Domain component. For instructions, see the *HP Service Health Reporter Installation and Configuration Guide*.

## Create the Directory Structure

To create the directory structure for the Reports component source files, at the command prompt run the following command:

```
<installation_directory>\CDE\bin>CreateCPFolders.bat -package RetailPOS
-subpackage RetailPOSApp -type application
```

where,

- <*installation_directory*> is the directory where you have installed SHR,
- **`RetailPOS`** is the name of the content pack you are developing, and,
- **`RetailPOSReporting`** is the name of the Reports component within **`RetailPOS`**.

The directory will contain templates provided by SHR that you will use to create the Reports component source files.

SHR provides sample source files for the Retail POS content pack at the following location. You can use these files as reference to create your own content pack.

```
%CDE_HOME%\samples\RetailPOS_Demo_Content_Pack\Source\
```

## Write the Model XML Document

The Model XML document for the Reports component must have a Logical section. In this section, you provide a reference to the cube that you defined in the Domain component Model XML document.

For reference find a sample Model XML file for the **`RetailPOSReporting`** available at the following location:

```
%CDE_HOME%\samples\RetailPOS_Demo_Content_Pack\Source\RetailPOS\RetailPO
SReporting.ap\source\model
```

To read the contents of the sample Model XML file, on this PDF document click the **Attachments: View File Attachments** tab and select `RetailPOS_App_model.xml`. Double click to open the XML file on your browser window.

## Create the Manifest XML File

The Manifest XML file contains the definition of the Model XML document that you created in Write the Model XML Document to be used by CDE.

To create the Manifest XML file using CDE:

1 Using the **cd** command change directory to:

```
%CDE_HOME%\workspace\RetailPOS\RetailPOSReporting.ap
```

2 Run the following command:

```
ant createManifestTemplate
```

The Manifest XML file named `RetailPOS_manifest_template.xml` is created in `%CDE_HOME%\workspace\RetailPOS\RetailPOSReporting.ap\source`

You can see a sample Manifest XML file for the **RetailPOSReporting** at the following location:

```
%CDE_HOME%\samples\RetailPOS_Demo_Content_Pack\Source\RetailPOS\RetailPOSReporting.ap\source
```

## Create a Connection to SAP BusinessObjects

To generate the SAP BusinessObjects Universe you must create a secured database connection to SAP BusinessObjects by using the Universe Designer. SHR provides a batch script to create the connection. To create a connection by using the script:

1 Using the **cd** command change directory to:

```
%CDE_HOME%\bin
```

2 At the command prompt, run the following:

```
setenv.bat
```

3 At the command prompt, run the following batch script:

```
createUniverseConnection.bat
```

A message `'Default BO Universe connection ("MA") was created successfully'` appears.

## Use CDE to Generate SAP BusinessObjects Universe

To generate the universe by using CDE, at the command prompt:

1 Using the cd command change directory to `%CDE_HOME%\workspace\RetailPOS\RetailPOSReporting.ap`

2 Run the following command:

```
ant
```

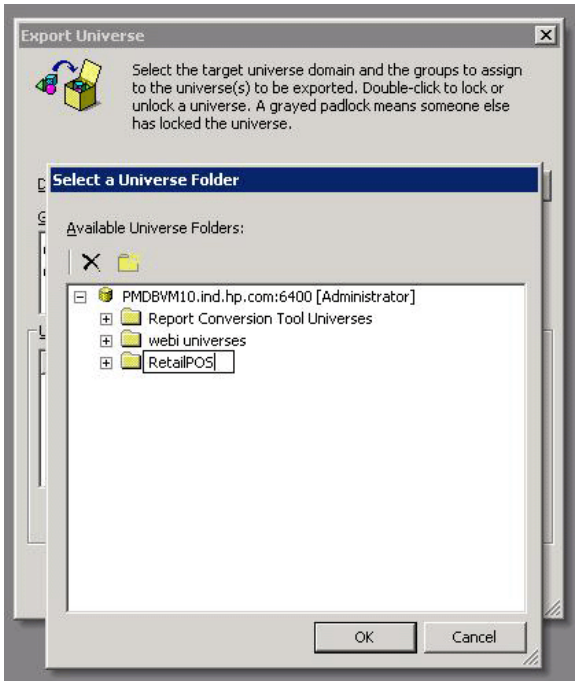
The universe is created with a filename extension of `.unv` and placed under the following folder:
`%CDE_HOME%\workspace\RetailPoS\RetailPoSReporting.ap\dist\RetailPOS\RetailPOSReporting.ap`

You can also edit the Universe to add additional hierarchies. See the *SAP BusinessObjects Universe Designer Online Help* for more information.

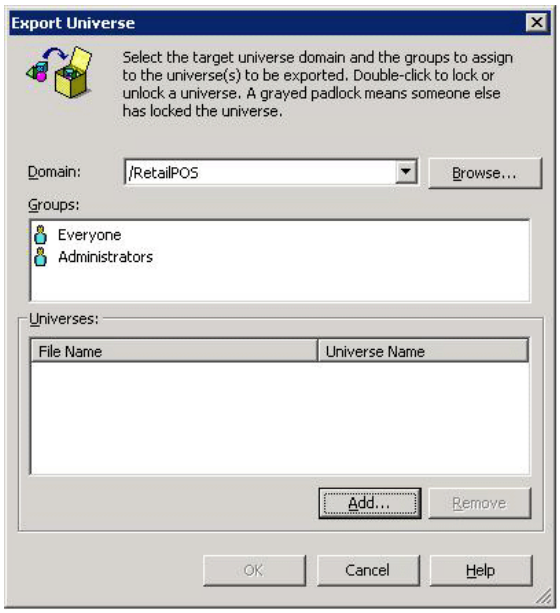
## Export the Universe to SAP BusinessObjects Repository

On the SAP BusinessObjects Universe Designer:

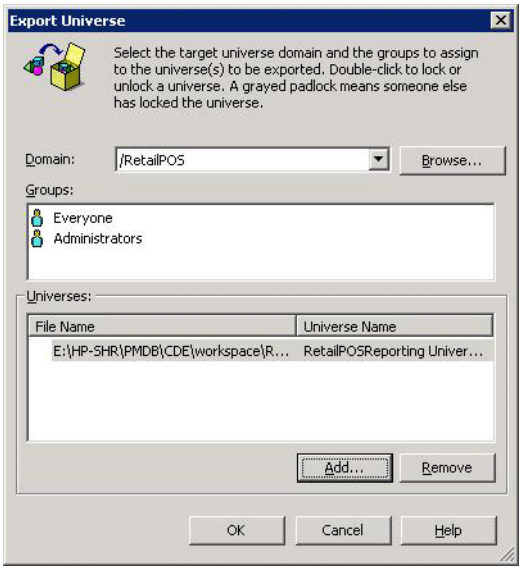
1. Select **File** → **Export**. The Export Universe window opens.
2. Select the required universe folder, in this example the RetailPOS, from the available list. Click **OK**.




3. On the Export Universe window, browse to the location of universe, in this example, the `RetailPOSReporting.unv`. Click **Open**.



The Export Universe window shows the `RetailPOSReporting.unv` added to the list of universes to be exported. Click **Ok**.



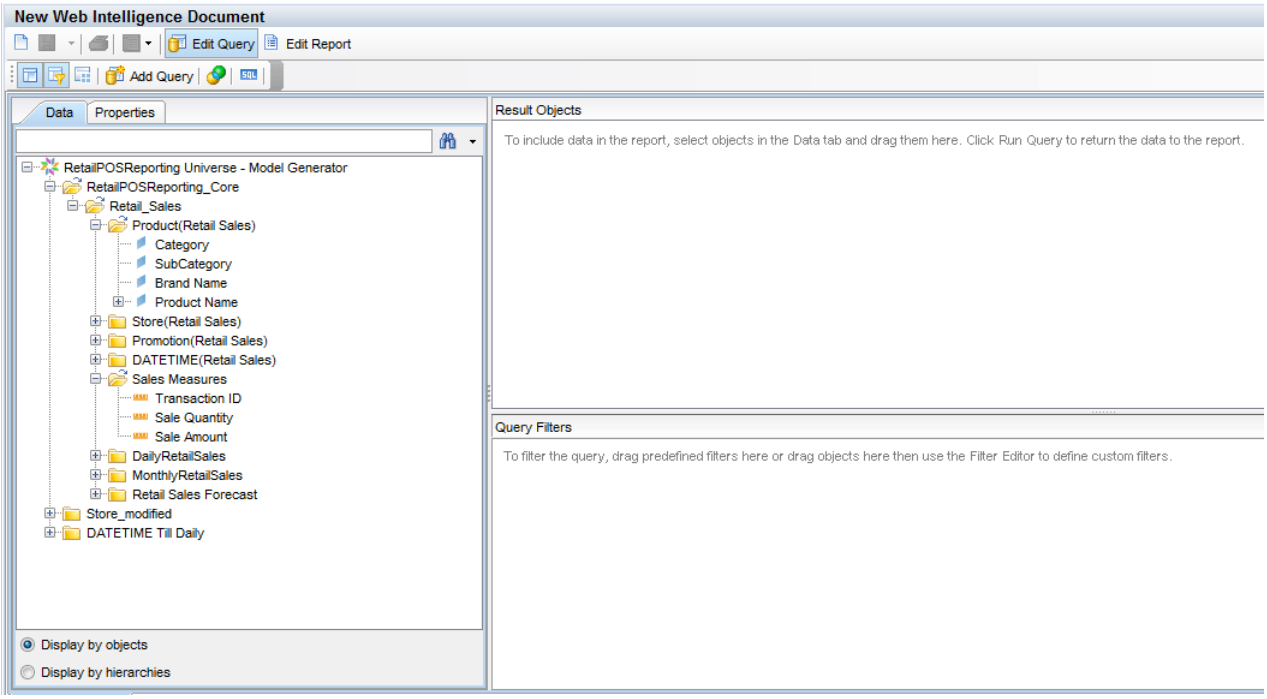

4 The `universe successfully exported` message appears.

## Create Web Intelligence Reports

You can create Web Intelligence reports by selecting the universe in SAP BusinessObjects InfoView and building one or more queries to define the data content of the reports.

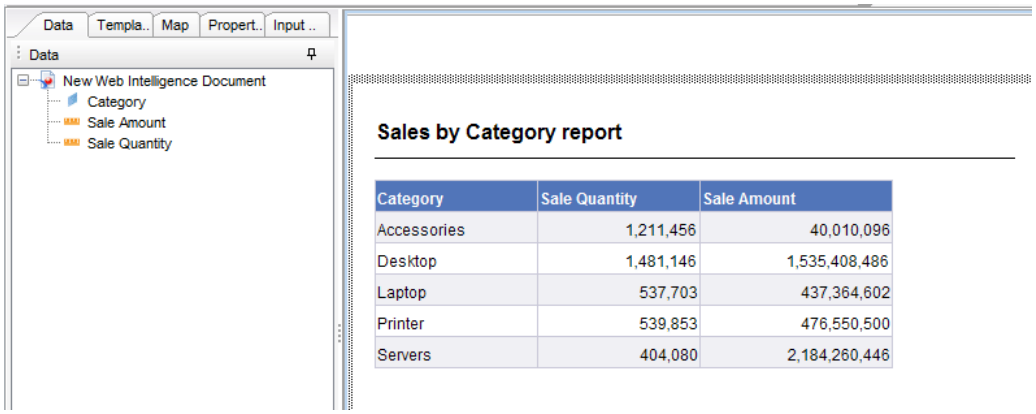
To create a simple sales report containing a table of sales quantity and sales amount per product category, follow these steps:

1. Logon to SAP BusinessObjects InfoView by using one of the following ways:
   - In the address bar of your web browser, type the URL of the SAP BusinessObjects system. The URL of the machine will be in the format: `http://<Host Name or IP address>:PORT NO (8080)/InfoViewApp/logon.jsp`.
   - In the Administration Console, click **Administration** → **SAP BOBJ** and then click **Launch InfoView**. The InfoView login page opens.
2. Click **Document List.**
3. Click **New** → **WebIntelligence Document**. The list of BusinessObjects Universes appears.
4. Select **RetailPOSReporting Universe - Model Generator.** The New Web Intelligence Document window opens. The data tab shows the objects – dimensions and measures – available in the universe as shown in the following figure.




5. To include data in the report, select the following in the data tab and drag them into the Result Objects window. Alternatively, you can double click the objects to place them in the Result Objects window.
   - Dimension: `Category` (under `Product(Retail Sales)`)
   - Measures:
     - `Sale Quantity` (under `Sales Measures`)
     - `Sale Amount` (under `Sales Measures`)
6. Click **Run Query** to return the data to the report.

A table of Sale Quantity and Sale Amount by Product Category is created. You can rename the table to an appropriate title.




## Enabling Time Drill Option in Reports

To enable time drill option in the report, click **Drill** Drill on the InfoView toolbar. You can drill down and roll up by the product category dimension.

**Option 1:** If you developed the Web Intelligence reports on a different system where SHR is not installed, you must do the following:

- Export the BIAR file to the system where SHR is installed, and,
- Generate the Reports component and install it using the Deployment Manager.

See instructions for Exporting Reports to SHR System on page 29.

**Option 2:** If you developed the Web Intelligence reports on the same system where SHR is installed, you can verify the workflow streams and view the reports on the SAP BusinessObjects InfoView as described in the following sections.

# Viewing Reports on SAP BusinessObjects InfoView

Now that you installed the Domain and Reports component packages and the data is loaded into the data warehouse, you can view the reports on the SAP BusinessObjects InfoView interface.

If you installed the sample `RetailPOS_Demo_Content_Pack` from `%CDE_HOME%\samples\` you see a report named `Retail Sales Report` available in the `Document List` in InfoView. For instructions on how to logon to InfoView and view reports, see *HP Service Health Reporter Online Help for Users*.

The Retail Sales Report contains the Sales Summary document which displays the sales revenue information for each of the dimensions that you defined in the data model – location, time, and, product category. You can drill down and roll up on the dimensions to view granular information.

### A Demonstration of Retail Sales Report

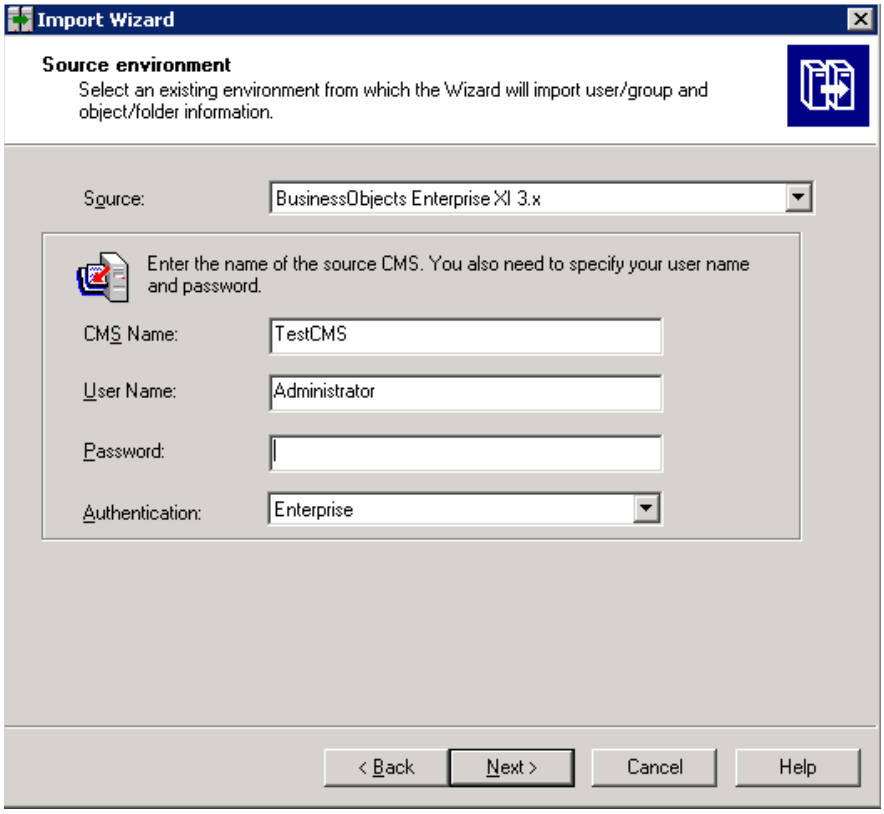
Click to activate the following video. Right-click the video for viewing options.

# Exporting Reports to SHR System

## Export Business Intelligence Archive Resource (BIAR) File

If you developed the reports on a system where SHR is not installed, you must export the BIAR file and install the Reports component on the system where SHR is installed. You select a source, a destination, and the objects that you want to import. Follow these steps:

1. Open the BusinessObjects Import Wizard.
2. On the **Source environment** page, enter the following:
   - CMS Name: The name of the machine on which BusinessObjects is installed.
   - User Name: The username for BusinessObjects user.
   - Password: The password for BusinessObjects user.
   - Authentication: Select **`Enterprise`**.

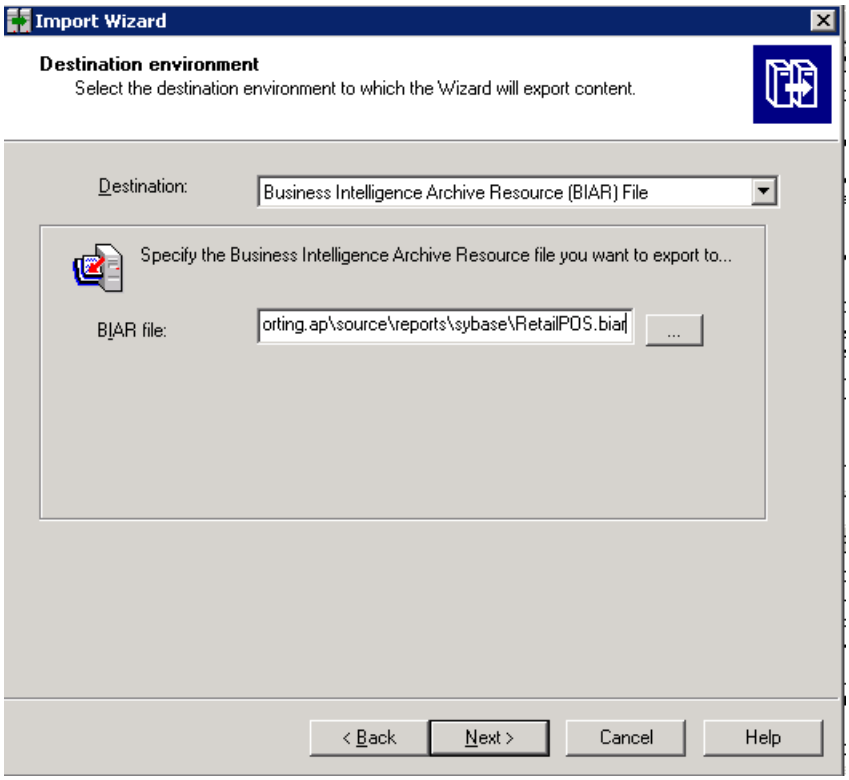



3 On the **Destination environment** page, select the following:

— Destination: Business Intelligence Archive Resource (BIAR) File.

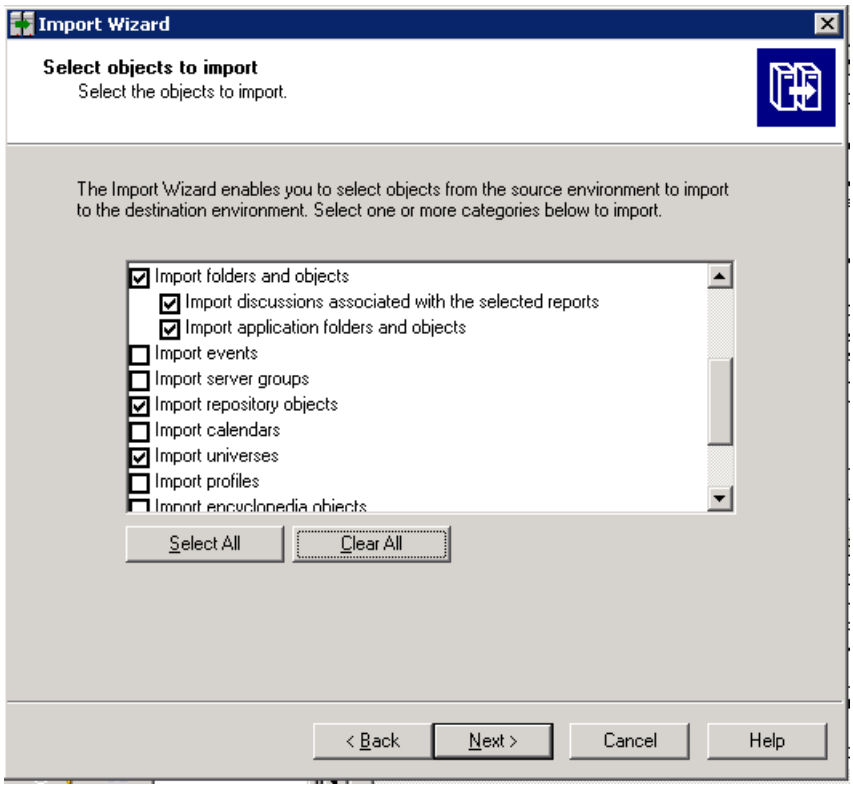
— BIAR file: The .biar file you want to export to. The path of the .biar file is:

```
%CDE_HOME%\workspace\RetailPOS\RetailPOSReporting.ap\source\reports\sybase\
```



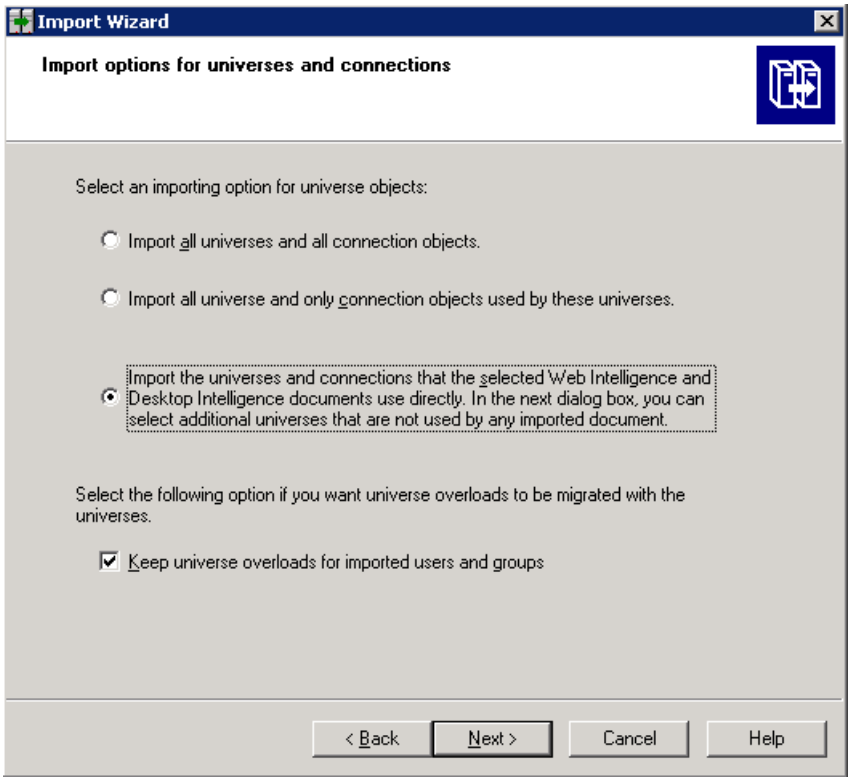

4 On the **Select objects to import** page, select the following:

- Import folders and objects.
- Import repository objects.
- Import universes.

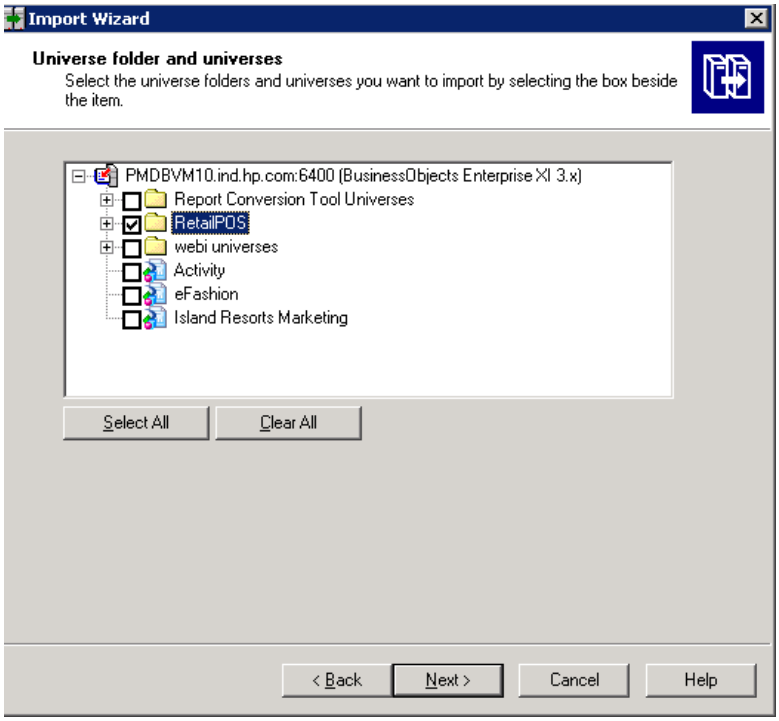



5 On the **Import options for universes and connections** page, select the following option:

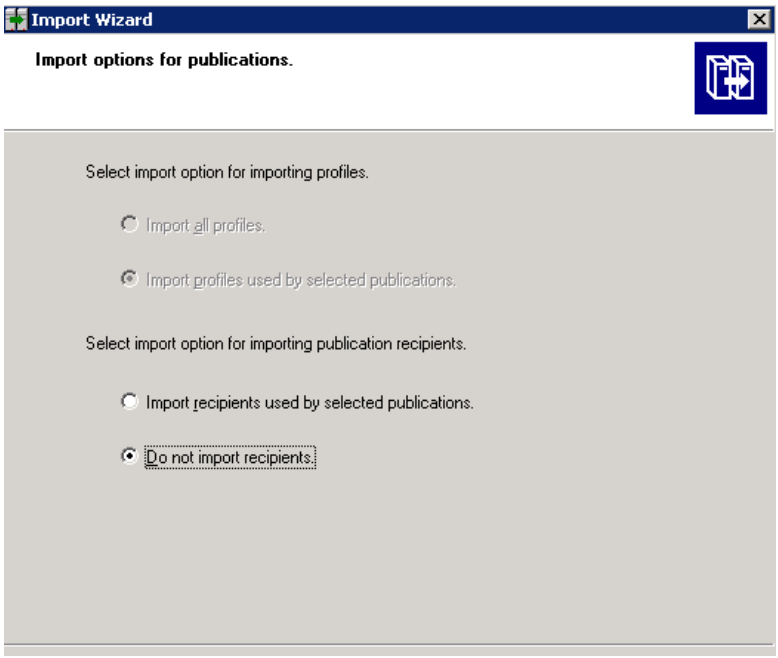
`Import the universes and connections that the selected Web Intelligence and Desktop Intelligence documents use directly.`



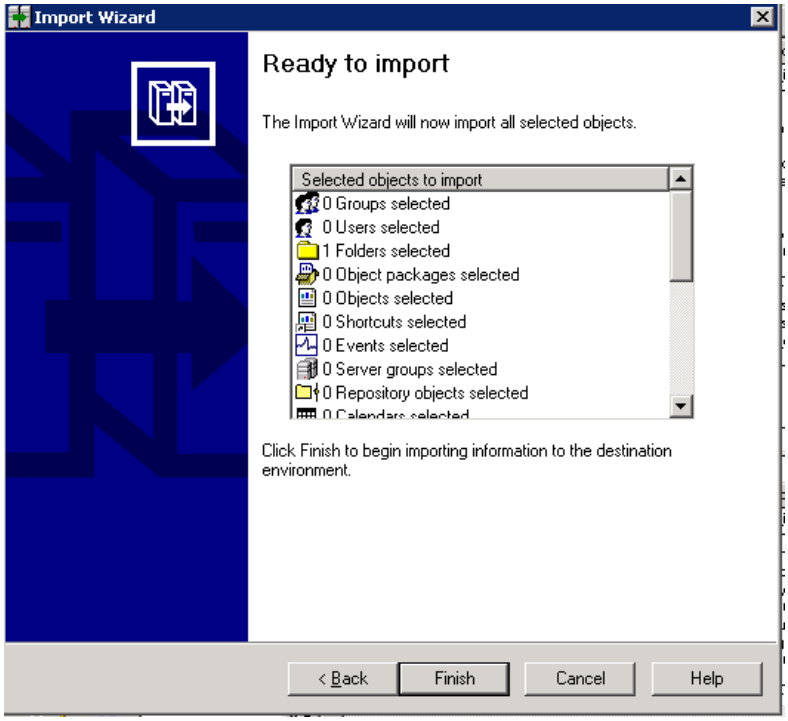
6 On the **Universe folder and universes** page, select the RetailPOS universe. Click **Next**.




7 On the **Import options for publications** page, select the `Do not import recipients` option.l




The **Ready to import page** appears. Click **Finish** to import the universe.




## Create the Manifest XML File

The Manifest XML file contains the definition of the BIAR file that you exported in the previous step to be used by CDE.

To create the Manifest XML file using CDE:

- Using cd command change directory to:

  `%CDE_HOME%\workspace\RetailPOS\RetailPOSReporting.ap`

- `Run the following command:`

  `ant` **`createManifestTemplate`**

The Manifest XML file named `RetailPOSReporting_manifest_template.xml` is created in `%CDE_HOME%\workspace\RetailPOS\RetailPOSReporting.ap\source`

Fore reference, see a sample Manifest XML file for the **`RetailPOSReporting`** at the following location:

`%CDE_HOME%\samples\`

## Use CDE to Generate Reports Component Package

To generate the Reports component package using CDE:

1. Using cd command change directory to:

   `%CDE_HOME%\workspace\RetailPOS\RetailPOSReporting.ap`

2. Run the following command:

   `ant`

The Reports component package named `RetailPOSReporting.ap` is created in

`%CDE_HOME%\workspace\RetailPOS\RetailPOS.ap\dist`

3. Browse to `%CDE_HOME%\workspace\RetailPOS\RetailPOS.ap\dist`
4. Copy `RetailPOSReporting.ap` to `%PMDB_HOME%\packages\RetailPOS`

## Install the Reports Component Package

SHR provides the Deployment Manager utility on the PMDB Platform Administration User Interface to install the content pack component packages.

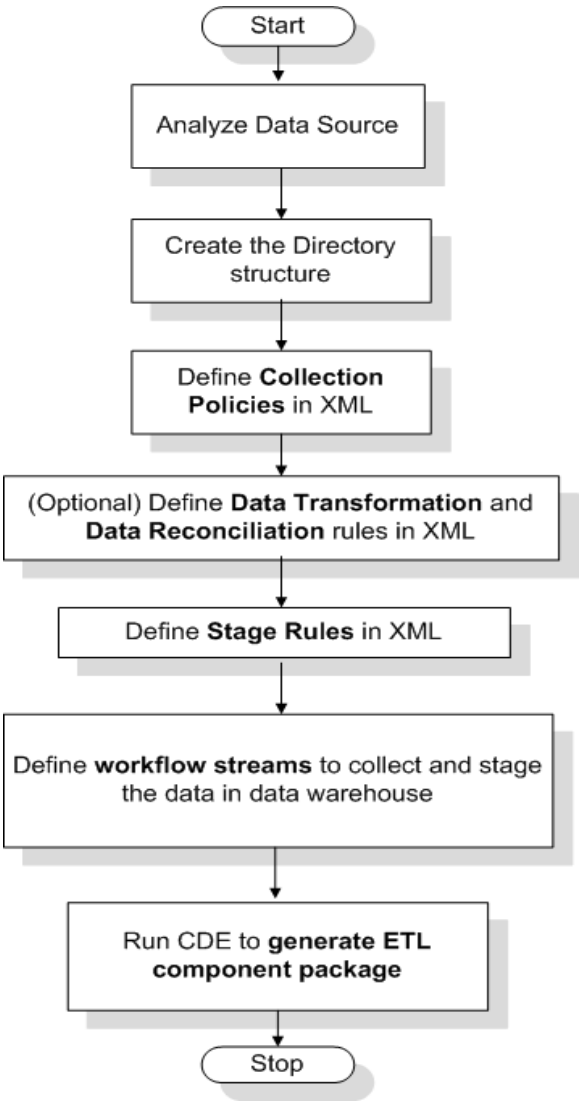
For instructions on how to install the content pack components, see the *HP Service Health Reporter Installation and Configuration Guide.*

After you install the Reports component package, you can verify the workflow streams on the Administration Console and view the reports on SAP BusinessObjects InfoView. See Verifying Workflow Streams on Administration Console on page 21 and Viewing Reports on SAP BusinessObjects InfoView on page 28.

# 4 Creating an ETL Component Package

This chapter describes the steps to create an ETL component package for the existing RetailPOS Domain component package.

Creating an ETL Component package involves the following steps:




This chapter does not describe data reconciliation.

## Prerequisite

### Create and Install the Domain Component Package

Follow the steps described in Creating and Installing the Domain Component on page 15 to create the Domain model and use CDE to generate the Domain component package.

## Analyze the Data Source

As described in the chapter Content Pack Architecture and Development, before you start creating the ETL component, you must identify the data source containing metrics that are suitable to be fed into the Domain data model.

In this guide we consider a simple database as the source of data. The database is created using PostgreSQL software and contains data suitable to be fed into the RetailPOS Domain model. Sample files and scripts are provided in the SHR media to create the database and populate data into the tables. The Appendix: Creating a Data Source for ETL Component describes how to create the PostgreSQL database, the database tables, and how to insert data into the tables.

## Create the Directory Structure

To create the directory structure for the ETL component source files, at the command prompt run the following command:

```
<installation_directory>\CDE\bin>CreateCPFolders.bat -package RetailPOS
-subpackage RetailPOSETL -type etl
```

where,

- *<installation_directory>* is the directory where you have installed SHR,
- **RetailPOS** is the name of the content pack you are developing, and,
- **RetailPOSETL** is the name of the ETL component within **RetailPOS**.

The directory will contain templates provided by SHR that you will use to create the ETL component source files.

SHR provides sample source files for the Retail POS content pack at the following location. You can use these files as reference to create your own content pack.

```
%CDE_HOME%\samples\RetailPOS_Demo_Content_Pack\Source\
```

## Define Collection Policy in XML

To collect the data from the `retailpos` database tables you must define a collection policy in XML. Use the collection policy template named `DB_collection_template.xml` available in the following folder:

```
%CDE_HOME%\workspace\RetailPOS\RetailPOSETL.ap\source\etl\collection\
```

For reference see the sample `RetailPOS_DB_Collection_Policy.xml` in the following folder:

```
%CDE_HOME%\samples\RetailPOS_Demo_Content_Pack\Source\RetailPOS\RetailPO
SETL.ap\source\etl\collection\
```

To read the contents of the sample collection policy XML file, on this PDF document click the **Attachments: View File Attachments** icon and select `RetailPOS_DB_Collection_Policy.xml`. Double click to open the XML file on your browser window.

## Define Data Transformation Rules

In the RetailPOS example, the data transformation rule is used to append the name of the city, state, country, and the zip code to the address column; and use the whitespace character as a delimiter of the column values in the .csv files.

For reference see the sample `RetailPOS_transformation.xml` in `%CDE_HOME%\samples\RetailPOS_Demo_Content_Pack\RetailPOS\RetailPOSETL.ap\doc`

To read the contents of the sample data transformation rule XML file, on this PDF document click the **Attachments: View File Attachments** icon and select `RetailPOS_transformation.xml`. Double click to open the XML file on your browser window.

## Define Stage Rules

Stage rules defined in XML files are used to map the column names in the source .csv files to the column names in the target physical tables, called staging tables, in the database. Use the stage rules template available in the folder `RetailPOS\RetailPOSDomain.ap\source\stagerule_templates` to write the stage rules XML files for each of the dimensions – product, sales, and store.

For reference see the sample stage rule XML files in `%CDE_HOME%\samples\RetailPOS_Demo_Content_Pack\RetailPOS\RetailPOSETL.ap\source\etl\stage_rules`

To read the contents of the sample stage rule XML files, on this PDF document click the **Attachments: View File Attachments** icon and select the following files:

- `Stage_product_stagerule.xml`: Stage rule for Product
- `Stage_retail_sales_stagerule.xml`: Stage rule for Sales
- `Stage_store_stagerule.xml`: Stage rule for Store

Double click to open the XML files on your browser window.

## Define Workflow Streams

Workflow streams defined in XML files are used to collect and stage the .csv files in the data warehouse.

Use the workflow streams template available in the folder `%CDE_HOME%\samples\RetailPOS_Demo_Content_Pack\source\RetailPOS\RetailPOSETL.ap\source\orchestration\stream_definitions` to write the workflow stream XML files, one file for each of the dimensions – product, sales, and store.

For reference see the sample workflow stream XML files in `%CDE_HOME%\samples\RetailPOS_Demo_Content_Pack\RetailPOS\RetailPOSDomain.ap\source\orchestration\stream_definitions`

To read the contents of the sample ETL workflow stream XML files, on this PDF document click the **Attachments: View File Attachments** icon and select the following files:

- `Dimension_Product_ETL_stream.xml:` Workflow stream XML for Product dimension
- `Dimension_Store_ETL_stream.xml:` Workflow stream XML for Store dimension
- `Fact_Retail_Sales_ETL_stream.xml:` Workflow stream XML for the fact

Double click to open the XML files on your browser window.

## Generate the ETL Component

To generate the ETL Component package using CDE:

1. Using cd command change directory to:

   `%CDE_HOME%\workspace\RetailPOS\RetailPOSETL.ap`

2. `Run the following command:`

   `ant`

The ETL component package is created in

`%CDE_HOME%\workspace\RetailPOS\RetailPOSETL.ap\dist`

3. Browse to `%CDE_HOME%\workspace\RetailPOS\RetailPOSETL.ap\dist`
4. Copy `RetailPOS` to `%PMDB_HOME%\packages.` Copying the package makes it available in the Administration Console Deployment Manager for installation.

## Install the ETL component

SHR provides the Deployment Manager feature on the Administration Console to install the content pack component packages. Use the Deployment Manager to install the ETL component package; in the RetailPOS example, the name of the package is RetailPOSETL.

For instructions on how to install the content pack components by using Deployment Manager, see the *HP Service Health Reporter Installation and Configuration Guide.*

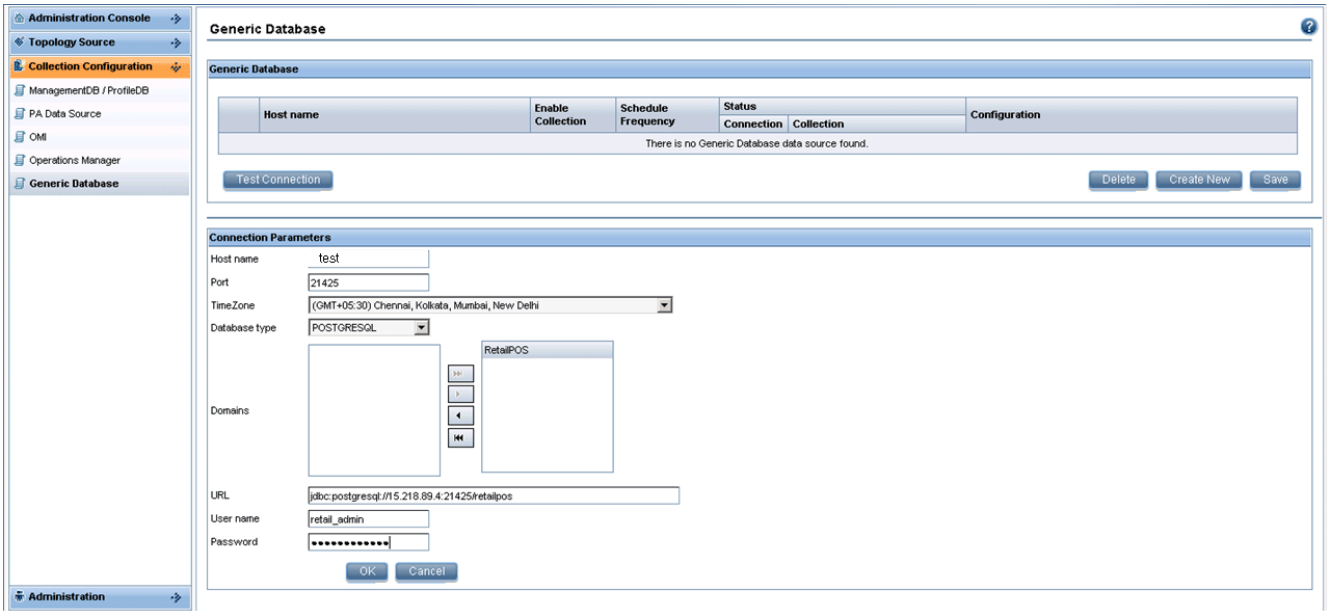
## Working with the ETL Component

### Configure a Generic Database

After you install the RetailPOS Domain and ETL components, you must configure a connection to the `retailpos` database to collect data based on the collection policy XML. You configure the connection on the Administration Console using the Generic Database page. Follow these steps:

1. In the Administrator Console, click **Collection Configuration** → **Generic Database**. The Generic Database page opens.
2. Click **Create New.** The Connection parameters dialog box opens.
3. Type or select the following values:

| Field | Description |
|---|---|
| Host name | Type the IP address or FQDN of the server where you created the retailpos database. |
| Port | Type the port number to query the database server. |
| TimeZone | Select the time zone under which the database instance is configured. |
| Database type | Select POSGRESQL. |
| Domain | Select RetailPOS. |
| URL | Type `jdbc:postgresql//<server>:<port>/retailpos` |
| User Name | Type the name of the generic database user. In this example, the user name is retail_admin. |
| Password | Type the password of the generic database user. In this example, the password is retail_admin. |



## Verify the ETL Component

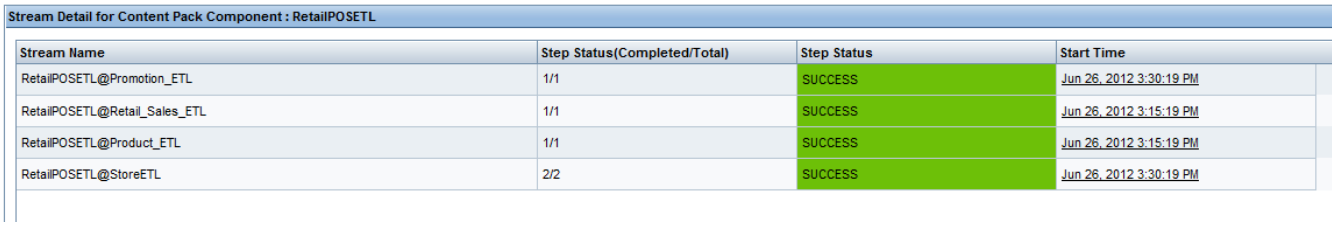
After you install the ETL component package and the `HP_PMDB_Platform_Timer` service is started, you logon to the Administration user interface to check the status of the ETL component workflow streams. Follow these steps:

1. On the Administration user interface, click **Internal Monitoring** > **Data Processing**.
2. On the **Stream Details** tab, view the status of streams in the RetailPOSETL content pack. All streams must show a status of OK to indicate successful completion.

In the RetailPOS example, the ETL component has the following workflow streams with one or more steps within each stream.

- A workflow stream `RetailPOSETL@Retail_Sales_ETL` to move Sales fact to stage tables.
- A workflow stream `RetailPOSETL@Product_ETL` to move Product dimension to stage tables.
- A workflow stream `RetailPOSETL@Store_ETL` to move Store dimension to stage tables.
- A workflow stream `RetailPOSETL@Promotion_ETL` to move Promotion dimension to stage tables.

As shown in the following figure, successful completion of the streams is indicated using green color.



Stream Detail for Content Pack Component : RetailPOSETL

| Stream Name | Step Status(Completed/Total) | Step Status | Start Time |
|---|---|---|---|
| RetailPOSETL@Promotion_ETL | 1/1 | SUCCESS | Jun 26, 2012 3:30:19 PM |
| RetailPOSETL@Retail_Sales_ETL | 1/1 | SUCCESS | Jun 26, 2012 3:15:19 PM |
| RetailPOSETL@Product_ETL | 1/1 | SUCCESS | Jun 26, 2012 3:15:19 PM |
| RetailPOSETL@StoreETL | 2/2 | SUCCESS | Jun 26, 2012 3:30:19 PM |

## View Reports

Now that you installed the Domain and Reports component packages and the data is loaded into the data warehouse, you can view the report on the SAP BusinessObjects InfoView interface. See Viewing Reports on SAP BusinessObjects InfoView on page 28.

# A Appendix: Creating a Data Source for ETL Component

To create a sample PostgreSQL database, SHR provides the following files and scripts:

| Files / Scripts Provided by SHR | Location of File / Script |
|---|---|
| The following .csv files that are copied into the database tables by using scripts:<br>`RetailPOS_Product.csv`<br>`RetailPOS_Promotion.csv`<br>`RetailPOS_Sales.csv`<br>`RetailPOS_Store.csv` | `%CDE_HOME%\samples\RetailPOS_Demo_Content_Pack\RetailPOS_DB_Creation_Scripts\RetailPOS_CSV` |
| `RetailPOS_CreateDatabase.sql`<br>This SQL script creates a PostgreSQL database named RetailPOS for the user named retail_admin. | `%CDE_HOME%\samples\RetailPOS_Demo_Content_Pack\RetailPOS_DB_Creation_Scripts` |
| `RetailPOS_CreateTables.sql`<br>This SQL script creates tables in the RetailPOS database. | `%CDE_HOME%\samples\RetailPOS_Demo_Content_Pack\RetailPOS_DB_Creation_Scripts` |
| `RetailPOS_PopulateTables.sql`<br>This SQL script copies the .csv files from the location `%CDE_HOME%\samples\RetailPOS_Demo_Content_Pack\RetailPOS_DB_Creation_Scripts\RetailPOS_CSV` to the database tables. | `%CDE_HOME%\samples\RetailPOS_Demo_Content_Pack\RetailPOS_DB_Creation_Scripts` |

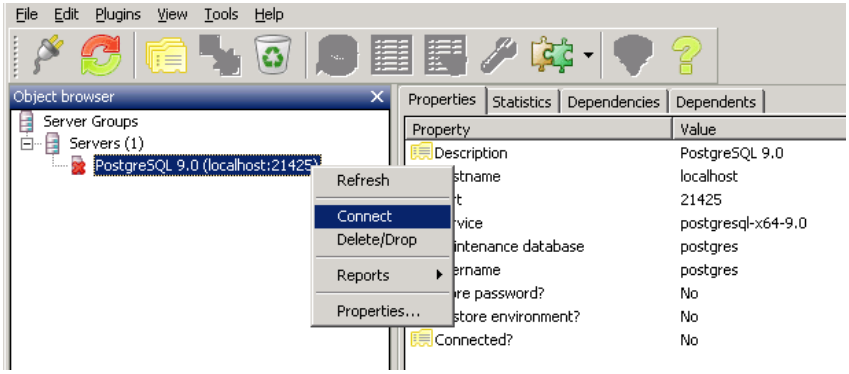
**Prerequisites:** Perform the following tasks before you start creating the PostgreSQL database:

1. Download and install PostgreSQL software from **http://www.postgresql.org/.** You can install PostgreSQL on any system which can be different than the system on which SHR is installed.
2. Copy the following files to the C:\ drive of the system where you installed PostgreSQL.
   - `RetailPOS_CSV`
   - `RetailPOS_CreateDatabase.sql`
   - `RetailPOS_CreateTables.sql`
   - `RetailPOS_PopulateTables.sql`

## Create a PostgreSQL Database

To create the database named RetailPOS, follow these steps:

1. Logon to the system where you installed PostgreSQl as Administrator.
2. Start the PostgreSQL **pgAdmin III** program.
3. Connect to default user **postgre** with the password you configured.




The Object browser pane shows the databases available for the postgre user.

4. On the SQL Query Editor window click **File** → **Open**. Browse to the location on C:\ drive where you copied the script `RetailPOS_CreateDatabase.sql` and click **Open**.
5. Click **Execute pgScript** to run the `RetailPOS_CreateDatabase.sql` script.

   The script creates the retailpos database and retail_admin user as the database owner.
6. Close the SQL Query Editor and click **Refresh**.

   The retailpos database with user retail_admin appear on the Object browser window.

## Create Database Tables

To create tables within the retailpos database, follow these steps:

1. On the Object browser window, select **retailpos** from the list of databases and open the SQL Query Editor.
2. On the SQL Query Editor window click **File** → **Open**. Browse to the location on C:\ drive where you copied the script `RetailPOS_CreateTables.sql` and click **Open**.
3. Click **Execute pgScript** to run the `RetailPOS_CreateTables.sql` script.

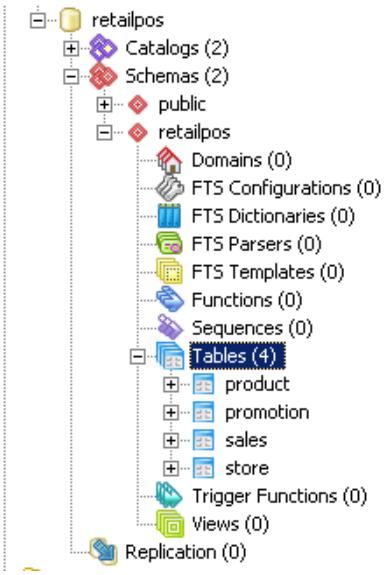
   The following tables are created as shown in the following figure:

   — retailpos.product

   — retailpos.store

   — retailpos.promotion

   — retailpos.sales




## Insert Data into the Database

To insert data from the `.csv` files into the tables, follow these steps:

1. On the SQL Query Editor window click **File** → **Open**. Browse to the location on C:\ drive where you copied the script `RetailPOS_PopulateTables.sql` and click **Open**.
2. Click **Execute pgScript** to run the `RetailPOS_PopulateTables.sql` script.

   The script inserts the .csv files from `RetailPOS_CSV` to the database tables.

# Glossary

**CDE**

CDE is a set of tools provided by SHR for development of content packs.

**Collection policy**

A Collection policy is written in XML to define the metrics to be collected by a collector program from the specified data source.

**Content pack**

Content packs are data marts deployed on the SHR performance management database platform. Content packs enable the platform to collect, store, process, and report the data. A content pack has three components – Domain, ETL, and Reports.

**Data Model**

Data model is a schema diagram that illustrates the relationship between dimension tables (that have attributes) and fact tables (that have measures).

**Domain Component**

Domain component of a content pack defines the data model of the Domain you are reporting on along with the logic to perform processing on the data. It is independent of the data source you collect data from.

**ETL Component**

ETL component of a content pack is data source dependant; it defines the collection of data from the specified data source.

**Loading**

The process of loading data from the stage tables to the data warehouse tables.

**Reconciliation**

Data reconciliation is the technique of associating fact data to corresponding dimension data.

**Reports Component**

The Reports component contains the SAP BusinessObjects Web Intelligence reports and universes.

**Staging**

Data staging is the process of moving the collected, transformed, and reconciled data into the staging tables.

### Transformation

Data transformation is the optional step of cleaning the collected data according to business requirements.

### Workflow Streams

Workflow streams in content packs are used to define and control the movement of data from one step to another.

# We appreciate your feedback!

If an email client is configured on this system, click Send Email

If no email client is available, copy the following information to a new message in a web mail client and send the message to **docfeedback@hp.com**.

**Product name and version**: HP Service Health Reporter 9.20

**Document title**: Content Development Getting Started Guide

**Feedback**: