

Enterprise PeopleTools 8.49 PeopleBook Issues and Corrections

Version 5: Posted March 11, 2009

This documentation enhancement covers known issues and corrections to the Enterprise PeopleTools 8.49 PeopleBooks. While every effort is made to ensure that PeopleBooks contain accurate and comprehensive information, late-breaking changes to the software can make the information in PeopleBooks incorrect or incomplete. This document contains corrections and information that had been omitted previously.

PeopleTools 8.49 PeopleBook Issues and Corrections

This document covers known issues and corrections for these publications:

- PeopleSoft Process Scheduler.
- PeopleSoft Data Management.
- Global Technology.
- System and Server Administration.
- PeopleCode API Reference.
- SQR Language Reference for PeopleSoft.
- PeopleSoft Integration Broker.

PeopleSoft Process Scheduler

This section covers known issues with the Process Scheduler PeopleBook.

Sending Notifications

This issue affects sending email notifications at the process or PSJob level for success and warning.

If you select the **Server Errors** checkbox on the Server Notification page (*PeopleTools, Process Scheduler, Servers, Notification*) for a role or user, only the notifications for errors will be delivered to that role or user. Success and warning will not be delivered.

If you want to receive notifications for warnings and success as well as errors, leave the **Server Error** checkbox cleared on the server definition and select **On Error, On Warning** and **On Success** on the Notification page for the process or PSJob.

There is an error in Chapter “Defining PeopleSoft Process Scheduler Support Information” Defining Process Definitions, Setting Notification Options. The text currently reads:

On Error	Select to notify the user or group when an error occurs. If this option is enabled the notifications at process and PSJob level for success and warning will not be delivered, only the notifications of error.
----------	---

This is incorrect, the sentence if this option is enabled the notifications at process and PSJob level for success and warning will not be delivered, only the notifications of error applies to Server Errors on the Server definition Notification page.

System Load Balancing

This issue affects the **System Load Balancing Option** field used by the Master Scheduler server to determine how to distribute the workload (processes or jobs) among the available Process Scheduler servers.

As delivered, this field is blank. If this value is left blank the Master Scheduler may not be able to determine a suitable Process Scheduler server on which to run processes or jobs that are scheduled without a Server Name assignment, leaving them in ‘queued’ status indefinitely.

To enable system load balancing, select *PeopleTools, Process Scheduler, System Setting, Process System* and change the value for the System Load Balancing Option field to *Assign To Server In Any O/S*.

Note. This field will be ignored if no Master Scheduler server is available to perform the load balancing.

OS/390—TSO Character Set

There is an error in Appendix: *Using the PSADMIN Utility* in the section “Editing the PeopleSoft Process Scheduler Configuration File for UNIX and OS390.” In the table in the “OS390 Config Section” there is a reference to the TSO Character Set, which is incorrect.

The text currently reads:

TSO Character Set	Specifies the codepage for the TSO environment. The default value is CP037 (IBM037: Latin1 code page).
	Note. For example, in Japanese setting, the recommended TSO Character Set is CCSID930.
	See Character Sets in the PeopleSoft Pure Internet Architecture.

The character set mentioned in the note is incorrect and should read CCSID939.

The same character set, CCSID939, should be used for both TSO Character Set and USS Character Set parameters in psprcs.cfg for UNIX and OS390 for Japanese settings.

Data Management

The following issues affect the Data Management PeopleBook.

Data Mover: Valid Commands in Bootstrap Mode

In previous releases, the command REPLACE_VIEW was listed as an invalid command within bootstrap mode. This is incorrect. REPLACE_VIEW is a valid command in bootstrap mode.

Data Mover: UNIX Command Line Syntax

The following examples provided for setting environment variables on UNIX are incorrect for /bin/sh.

```
export PS_DM_DATA=data_path
export PS_DM_SCRIPT=script_path
export PS_DM_LOG=log_path
export PS_SERVER_CFG=$PS_HOME/setup/psdmtx.cfg
```

Modify these to:

```
PS_DM_DATA=data_path;export PS_DM_DATA
PS_DM_SCRIPT=script_path;export PS_DM_SCRIPT
PS_DM_LOG=log_path;export PS_DM_LOG
PS_SERVER_CFG=$PS_HOME/setup/psdmtx.cfg;export PS_SERVER_CFG
```

Global Technology

This section covers these corrections to the *Enterprise PeopleTools 8.49 PeopleBook: Global Technology*:

- Adding translations to existing databases.
- Cloning existing search criteria.
- Selecting email character sets.
- Converting to Unicode on Oracle databases.
- Understanding time zone maintenance.
- Understanding currency-specific settings.
- Installing PeopleSoft-provided translations.

Note. After the title of each correction, an instruction in ***bold-italic*** indicates how the new material fits into the existing PeopleBook.

Adding Translations to Existing Databases

In step 5, the procedure indicates to copy the incorrect project, PPLTLSCURML. The correct project to copy is PPLTLSML.

Cloning Existing Search Criteria

In the Sample Search Criteria section, the first three bullets are changed to read as follows:

Following are examples of possible searches:

- Replacing base language text.

If you enter text in the Search Text - Base and Replacement Text sections, the search and replace is only on the base tables. This is a search and replace because the Search Only (No Replace) check box is clear. Because the search and base languages match, you can enter data in the Search Text - Base and Replacement Text sections of the page.

Notice that all of the check boxes are selected in the Search Objects area. That means that the system searches all of the available search objects for the word *Department*. When the search results are returned, you can replace each instance of the word *Department* with the term *Cost Center*. The text replacement occurs in the base language tables.

- Searching base language text.

For example, enter the word *Product* in the Search Text — Base section only. In this case the system searches for the word *Product* in the base language tables and shows all occurrences of *Product*. You might do this as a preliminary effort to locate the word *Product*, which might have been translated.

- Searching non-base language text.

For example, if you enter the word *Producto* (and select the full word match) in the Search Text - Non Base section, the system searches for all instances of the word *Producto* in the Spanish language tables. Again, you might do this type of search in order to locate all of occurrences of a translation.

Selecting Email Character Sets

Add the following topic at the end of this section:

Using Extended Japanese Characters

To use certain Windows-31J characters (specifically, NEC special characters, NEC-selected IBM extended characters, IBM extension characters, and user-defined characters) in incoming or outgoing email messages with the ISO-2022-JP Japanese character set, your web server and your application server or process scheduler must be using a JRE or JDK version 1.5.0_07 or a 1.5.0_x later than 1.5.0_07.

In addition, you must complete additional configuration of your web server (for incoming email) and application server or process scheduler (for outgoing email).

For incoming email on the web server, the following JVM setting must be added to the JAVA_OPTIONS_WIN32 parameter in the setenv.cmd file:

```
SET JAVA_OPTIONS_WIN32=  
"-Dsun.nio.cs.map=x-windows-iso2022jp/ISO-2022-JP"
```

For outgoing email on the application server or process scheduler, the following JVM option must be added to either the psappsrv.cfg file or the psprcs.cfg file depending on whether the application server or an AE program, respectively, will be handling outgoing email messages. JVM options are set in the PSTOOLS section of the file:

```
[PSTOOLS]  
...  
JavaVM Options=-Dsun.nio.cs.map=x-windows-iso2022jp/ISO-2022-JP
```

Converting to Unicode on Oracle Databases

There are corrections to two subsections:

- Pre-Creating Unicode Database Table Structures Using PeopleSoft Data Mover.
- Specifying Unicode Databases in PeopleSoft Systems.

Pre-Creating Unicode Database Table Structures Using PeopleSoft Data Mover

The Data Mover script to be executed is as follows (bold highlights the new steps):

```
SET NO TRACE;  
SET NO DATA;  
SET INPUT input_file;  
SET LOG log_file;  
SET UNICODE ON;
```

```
SET STATISTICS OFF;  
SET NO INDEX;  
SET ENABLED_DATATYPE 9.0;  
IMPORT *;
```

Specifying Unicode Databases in PeopleSoft Systems

Replace this section of the PeopleBook with the following:

When the import has completed successfully you must specify in PeopleSoft that the database is now a Unicode database. To do this, login to SQL*Plus as the database owner ID, and run the following SQL statement:

```
UPDATE PSSTATUS SET UNICODE_ENABLED=1;
```

Since you set ENABLED_DATATYPE previously, you also need to specify that the database uses Oracle CLOB and BLOB fields, as follows:

```
UPDATE PSSTATUS SET DATABASE_OPTIONS=2;
```

See Also

Enterprise PeopleTools 8.49 PeopleBook: Global Technology, Converting PeopleSoft Systems to Unicode Databases, Converting to Unicode on Oracle Databases, Rerunning Oracle Database Imports

Understanding Time Zone Maintenance

Replace this section of the PeopleBook with the following:

Oracle delivers a subset of time zone data including information about daylight saving time observances in the Time Zone table (PSTIMEZONE). The Oracle-provided subset of time zone data is consistent with current law at the time of the PeopleTools GA release. Oracle does not provide patches or updates to this time zone data once it is released—for example, to bring it up to date with recently enacted legislation. Therefore, you are responsible for two tasks: the creation of all time zone data not included in the Oracle-provided subset that is required by your global operations, *and* the maintenance and update of all time zone data whenever local regulations change the definition for a time zone—for example, changes to the daylight saving time start or end date.

Understanding Currency-Specific Settings

Replace this section of the PeopleBook with the following:

PeopleTools can format a currency amount that appears on a page or a report based on both your numerical format preferences and the currency that the amount represents.

Some display settings, such as the characters that are used for the thousands and decimal separators, are based on your language preference. Others, such as the currency symbol and the decimal precision, are based on the properties of the currency in which the amount appears. Those properties include:

- A currency symbol, such as \$ for the Australian dollar.
- A decimal precision.

For example, the Australian dollar (\$5.00) has two decimal positions, but the Korean Won (500 KRW) has no decimal positions.

Currency information is stored in the Currency Code table (CURRENCY_CD_TBL) record definition along with other information about the currencies that is used in the implementation. The CUR_SYMBOL field controls the currency symbol. The DECIMAL_POSITIONS field controls the decimal precision for a currency.

Note. The PeopleSoft Currency Code table (CURRENCY_CD_TBL) contains sample currency and country code data. The Currency Code table is based on ISO Standard 4217, "Codes for the representation of currencies," and also relies on ISO country codes in the Country table (COUNTRY_TBL). The navigation to the pages where you maintain currency code and country information depends on which PeopleSoft applications you are using. To access the pages for maintaining the Currency Code and Country tables, consult your application PeopleBooks for more information.

Installing PeopleSoft-Provided Translations

Replace this section of the PeopleBook with the following:

Installing Oracle-Provided Translations

This section provides an overview of translation installation and discusses how to:

- Add translations to an existing database on the same PeopleTools version.
- Add translations to an existing database on a different PeopleTools version.
- Use the upgrade process to copy customizations into a multilingual database.

Understanding Translation Installation

In many cases, Oracle provides translations of the user interface and key application data for many PeopleSoft application products. Except for PeopleTools, these translations are shipped on a separate CD for each product line; this CD is known as the "PeopleSoft Global Multi-Language CD." If you did not receive this CD in your kit, you may not have licensed the appropriate translations from Oracle. Contact customer support to obtain the CD. Translations for PeopleTools itself are shipped on the PeopleTools CD; no Global Multi-Language CD is required for PeopleTools.

Typically, you would install these translations when you first install your database. However, if you did not install the Oracle-provided translations during your initial install, this section describes the steps to follow to add translations that you have licensed from Oracle to an already-existing database.

Note. You should install the PeopleSoft Global Multi-Language CD at the same time that you install the English release.

Note. The size impact on the database of installing translations is difficult to predict exactly, but a general rule of thumb is approximately 5% per language.

To add translations licensed from Oracle, you must determine how you will perform your upgrade. Choose the most appropriate of the following three options:

- Add the translations provided on the PeopleSoft Global Multi-Language CD to your existing PeopleSoft database on the same PeopleTools version.

This option is appropriate only if you have not performed any application or PeopleTools maintenance (that is, you have not upgraded, applied patches, or applied maintenance packs), *and* you have not made any customizations to existing PeopleTools-based objects in your existing database, *and* your current PeopleTools version matches the version originally available at the time of the GA release of your PeopleSoft application. PeopleTools-based objects include any objects that you maintain using PeopleSoft Application Designer or PeopleSoft Tree Manager, such as pages, fields, and menus. If you have added only new PeopleTools objects to your database, without modifying existing definitions, then you can use this option.

Adding or customizing system data, such as account codes, departments, and locations, is not considered a PeopleTools object change. If you have performed only these types of transactions against your existing database, *and* you have not modified any objects using PeopleSoft Application Designer or PeopleSoft Tree Manager, then you can use this option.

- Add the translations provided on the PeopleSoft Global Multi-Language CD to an existing database on a different PeopleTools version.

This option is appropriate if you have not made any customizations to existing PeopleTools-based objects, *but* your PeopleTools version is different from the version that was available at the time of the GA release of your PeopleSoft application. You will need to set up a separate database with the translations, upgrade the database to your current PeopleTools version, and then copy the translations from your new database to your existing PeopleSoft database.

- Use the upgrade process to copy your customizations into a multilingual database.

If you have customized the PeopleTools objects in your database, you cannot simply add the translations contained on the PeopleSoft Global Multi-Language CD to your existing database. Instead, you must create a new database containing all of your licensed translations and use the upgrade process to copy customizations from your existing database to your newly installed database.

Adding Translations to an Existing Database on the Same PeopleTools Version

This procedure is appropriate only if you have not performed any application or PeopleTools maintenance (that is, you have not upgraded, applied patches, or applied maintenance packs), *and* you have not made any customizations to existing PeopleTools-based objects in your existing database, *and* your current PeopleTools version matches the version originally available at the time of the GA release of your PeopleSoft application.

To add translations to an existing database on the same PeopleTools version:

1. Read the section “Understanding Translation Installation” presented previously in this chapter.

See *Enterprise PeopleTools 8.49 PeopleBook: Global Technology, Working With PeopleSoft Applications in Multiple Languages, Using Related Language Tables, Installing Oracle-Provided Translations, Understanding Translation Installation*.

2. Back up your database.
3. Load your PeopleSoft Global Multi-Language CD.

See the Enterprise PeopleTools 8.49 installation guide for your database platform, "Setting Up the Windows File Server."

4. Rerun the steps for setting up the batch environment.

If you are running a Process Scheduler on Microsoft Windows, then you must rerun the steps for setting up the batch environment documented in your PeopleTools installation guide.

Because the PeopleSoft Global Multi-Language CD contains translations for your Crystal Reports and PS/nVision layouts, you must transfer those translations to your Windows-based Process Scheduler.

If you run only non-Windows-based Process Schedulers, you do not need to complete this step. This step also does not need to be performed for UNIX batch servers because the only objects that contain translations outside the database are Crystal and PS/nVision reports, both of which are supported on only Windows-based batch servers.

See the Enterprise PeopleTools 8.49 installation guide for your database platform, "Setting Up the Batch Environment."

5. Generate database setup from PeopleSoft Data Mover.

Run PeopleSoft Data Mover in bootstrap mode by logging in with your database's access ID. From PeopleSoft Data Mover, select File, Database Setup, and follow the wizard step by step to generate the database setup script. Be sure to select "Add New Language" in the "Database Type" selection box in the second step of the wizard. The generated database setup script appears in the PeopleSoft Data Mover input view. Be sure to enter the same database parameters that you entered the first time you ran the database setup program to install your English database; do not select any language that already exists in your database.

If you choose a new base language, then the generated script will include the command to swap the base language.

6. Run the PeopleSoft Data Mover script.

Because you have already created your database and set up the appropriate database-specific objects, you must perform only the "Run Data Mover Import Scripts" task.

7. Run the VERSION Application Engine program to update your PeopleTools version numbers.

Run the PeopleTools VERSION Application Engine program against your database. This synchronizes the PeopleTools version numbers, taking the new translations into account.

8. Rerun your database audits.

Your database should now be complete. To ensure that it is complete, rerun the database audits. Refer to the section about creating and checking a database in your PeopleTools installation guide for more information.

See the Enterprise PeopleTools 8.49 installation guide for your database platform, "Creating a Database."

Adding Translations to an Existing Database on a Different PeopleTools Version

This procedure is appropriate only if you have not made any customizations to existing PeopleTools-based objects, *but* your PeopleTools version is different from the version that was available at the time of the GA release of your PeopleSoft application. To add translations to an existing database on a different PeopleTools version than the one that was available at the time of the GA release of your PeopleSoft application, you must set up a separate database with the translations, upgrade it to the required PeopleTools version, and then copy the translations from the new database to your existing database.

To add translations to an existing database on a different PeopleTools version:

1. Read the section “Understanding Translation Installation” presented previously in this chapter.

See *Enterprise PeopleTools 8.49 PeopleBook: Global Technology, Working With PeopleSoft Applications in Multiple Languages, Using Related Language Tables, Installing Oracle-Provided Translations, Understanding Translation Installation*.

2. Back up your database.
3. Install a new database.

Install a new database, including the translations you need at the same time, making sure to perform the steps in the task “Updating Database to Latest PeopleTools Release” in your PeopleTools installation guide. The translations need to be loaded prior to running this task. For more information on installing and upgrading PeopleTools translations, including PPLTLSML, PPLDELETE, PATCH84xML, PT84xTLSxxx.dms, and MSGTLxxx.dms, see your PeopleTools installation guide.

See the Enterprise PeopleTools 8.49 installation guide for your database platform, “Updating Database to Latest PeopleTools Release.”

4. Do a database compare.

Do a database compare of your new database against your other databases, to create a project containing all managed object differences between the two databases. Review the database compare report and the generated database compare project to ensure that only language objects are copied, and they don't overwrite any customizations.

5. Apply this project to the target database.
6. Export application system data.

In Data Mover, export application system data (non managed objects) from the new database to a .dat file. Import the system data into the target database. For application data, the Define System Data menu lists the system data, and the tables are PS_PM_PS_REC and PSRECDEFN.RELLANGRECNAME. PeopleTools system data was already loaded via the PT84xTLYyy.dms scripts.

7. Run the VERSION Application Engine program.

Run the PeopleTools VERSION Application Engine program against your database. This synchronizes the PeopleTools version numbers, taking the new translations into account.

8. Rerun your database audits.

Your database should now be complete. To ensure that it is complete, rerun the database audits. Refer to the section about creating and checking a database in your PeopleTools installation guide for more information.

See the Enterprise PeopleTools 8.49 installation guide for your database platform, “Creating a Database.”

Using the Upgrade Process to Copy Customizations into a Multilingual Database

This section explains how to load translations by using the upgrade process to copy your customizations into a multilingual database. If you have customized the PeopleTools objects in your database, you cannot simply add the translations contained on the PeopleSoft Global Multi-Language CD-ROM to your existing database. Instead, you must create a new database containing all of your licensed translations and use the upgrade process to copy customizations from your existing database to your newly installed database.

To use the upgrade process to copy your customizations into a multilingual database:

1. Read the section “Understanding Translation Installation” presented previously in this chapter.

See *Enterprise PeopleTools 8.49 PeopleBook: Global Technology, Working With PeopleSoft Applications in Multiple Languages, Using Related Language Tables, Installing Oracle-Provided Translations, Understanding Translation Installation*.

2. Back up your database.
3. Load your PeopleSoft Global Multi-Language CD.

See the Enterprise PeopleTools 8.49 installation guide for your database platform, “Setting Up the Windows File Server.”

4. Rerun the steps for setting up the batch environment.

If you are running a Process Scheduler on Microsoft Windows, then you must rerun the steps for setting up the batch environment documented in your PeopleTools installation guide.

Because the PeopleSoft Global Multi-Language CD contains translations for your Crystal Reports and PS/nVision layouts, you must transfer those translations to your Windows-based Process Scheduler.

If you run only non-Windows-based Process Schedulers, you do not need to complete this step. This step also does not need to be performed for UNIX batch servers because the only objects that contain translations outside the database are Crystal and PS/nVision reports, both of which are supported on only Windows-based batch servers.

See the Enterprise PeopleTools 8.49 installation guide for your database platform, “Setting Up the Batch Environment.”

5. Create the new multilingual database.

Create and load your new PeopleSoft global multi-language database, which contains all of the languages you’ve licensed, and apply patches and bundles to bring it to the same level as your source database. Refer to the section on creating a database in your PeopleTools installation guide for more information.

Keep in mind that you must load the English objects provided by PeopleSoft even if you don’t plan to run your system in English. Follow all of the applicable steps in the documentation about creating a database except the swap base language step. You should not swap the base language of your new database until you have copied your customizations into it.

See the Enterprise PeopleTools 8.49 installation guide for your database platform, “Creating a Database.”

6. Copy the customizations into the new multilingual database.

Use the PeopleSoft Application Designer upgrade compare process to compare your existing, customized database against the newly installed database in order to identify your customizations. Once you have identified the objects you have changed, use upgrade copy to copy the objects into your new multilingual database.

If you have maintained all of your customizations in PeopleSoft Application Designer projects, you can easily use these projects to copy only the objects you have modified over to your new PeopleSoft global multilingual database.

7. Create a project with all objects.

This project will be used in the next step to upgrade customizations and translations into the test or production database.

8. Load changes in the test database.

- a. Using a test database (a copy of the production database) as a target, copy the project from the previous step to the database, selecting only the new language.
 - b. Using PeopleSoft Data Mover in user mode, import the language tools .dat files using the data mover script, ppltIs84Xxxx.dms.
 - c. In the PeopleSoft Pure Internet Architecture, sign in in English and ensure that the new language is marked as Enabled in the Maintain Installed Languages pages.
 - d. Run the VERSION Application Engine program to update your PeopleTools version numbers to synchronize the PeopleTools version numbers, taking the new translations into account.
 - e. (Optional) Swap your base language. If you have not already swapped your base language and you decide to run your database in a non-English base language, swap the base language of your database.
9. After testing, repeat the previous steps to load changes in the production database.
 10. Rerun the database audits.

Your database should now be complete. To ensure that it is complete, rerun the database audits, as documented in the PeopleTools installation guide in the chapter on creating a database.

See the Enterprise PeopleTools 8.49 installation guide for your database platform, "Creating a Database."

See Also

Enterprise PeopleTools 8.49 PeopleBook: Global Technology, Working With PeopleSoft Applications in Multiple Languages, Using Related Language Tables, Swapping the Base Language

System and Server Administration

This section discusses errors or omissions from the *Enterprise PeopleTools 8.49 PeopleBook: System and Server Administration*.

Search Server Support (Verity)

In the section *Configuring Search and Building Search Indexes, Configuring PeopleSoft Search, Understanding PeopleSoft Search Configurations* the following table appears:

Operating System	Verity Support	Deployment Options
AIX	32-bit	Type-2, Type-3
HPUX PA64	32-bit	Type-2, Type-3
HPUX IA 64	64-bit	Type-1, Type-2, Type-3
SUSE Linux	32-bit	Type-1, Type-2, Type-3

Operating System	Verity Support	Deployment Options
Red Hat Linux	32-bit	Type-1, Type-2, Type-3
Solaris	32-bit	Type-2, Type-3
Windows	32-bit	Type-1, Type-2, Type-3
Z/Linux	None.	None.
Z/OS	None.	None.

The table above is no longer accurate and is replaced by the following table:

Operating System	Verity Support	Supported Deployment Options
AIX	32-bit	Type-2
HPUX PA64	32-bit	Type-2
HPUX IA 64	None	Type-3
SUSE Linux	32-bit	Type-1
Red Hat Linux	32-bit	Type-1
Solaris	32-bit	Type-2
Windows (32-bit) Windows (64-bit)	32-bit	Type-1
Z/Linux	None.	None.
Z/OS	None.	None.

Note. Search Domains are supported only on Windows and Linux.

Oracle Application Server

In the “Configuring IIS Listener to Use OracleAS Proxy Plug-In” section, make note of the following correction.

Before

12. Using the IIS management console, add oracle_proxy.dll as a filter in your IIS Web site.

The name of the filter should be oproxy and its executable must point to the directory containing oracle_proxy.dll (as in, F:\IIS\proxy\oracle_proxy.dll).

13. Using the IIS management console, add oracle_proxy.dll as a filter in your IIS Web site.

The name of the filter should be oproxy and its executable must point to the directory containing oracle_proxy.dll (as in, d:\proxy\oracle_proxy.dll).

After

12. Using the IIS management console, add a new virtual directory to your IIS Web site with the same physical path as that of oracle_proxy.dll. Name the directory oproxy and give it execute access.

13. Using the IIS management console, add oracle_proxy.dll as a filter in your IIS Web site. The name of the filter should be oproxy and its executable must point to the directory containing oracle_proxy.dll (as in, F:\IIS\proxy\oracle_proxy.dll).

WebSphere

In the "Generating a Certificate for the WebSphere using PeopleSoft pskeyManager" section, the following steps need to be modified:

Before

9. Download the VeriSign TestCA certificate:

Download the VeriSign test CA certificate from <http://digitalid.verisign.com/cgi-bin/getcacert>. When prompted, save getcacert.cer to your WebSphere domain directory. If you need to FTP your certificate to UNIX, you must FTP it in ASCII mode to your WebSphere domain directory.

After

9. Download the VeriSign Trial CA certificates:

First download the VeriSign Trial Root CA certificate from:

https://www.verisign.com/support/verisign-intermediate-ca/Trial_Secure_Server_Root/index.html

From the specified link, click the "Select All" button and copy the contents of the certificate into the verisignRootCA.cer file and save it to your WebSphere domain directory.

Then, download VeriSign's Trial Intermediate CA certificate from:

<https://www.verisign.com/support/verisign-intermediate-ca/trial-secure-server-intermediate/index.html>

Click the "Select All" button and copy the contents of the certificate into the verisignInterCA.cer file and save it into your WebSphere domain directory. You can also append the contents of this Trail Intermediate CA certificate to the Root CA certificate file verisignRootCA.cer.

Note. If you need to FTP your certificate to UNIX, you must FTP it in ASCII mode to your WebSphere domain directory.

Before

10. Import the Verisign test Certificate Authority's certificate into your keystore.

To import the Certificate Authority's public certificate (which you received from Verisign) into your keystore, run `pskeymanager.cmd -import`. When prompted for an alias, specify `VerisignTestCA` as the name to store this CA as. This name is simply an alias for this certificate. When prompted for the certificate file to import, specify the `getcacert.cer` file.

After

10. Import the VeriSign Trial Certificate Authority's certificates into your keystore.

To import the Certificate Authority's public certificate (which you received from VeriSign) into your keystore, run `pskeymanager.cmd -import -trustcacerts`.

When prompted for an alias, specify `VerisignTrialCA` as the name to store this CA as. This name is simply an alias for this certificate. When prompted for the certificate file to import, specify the `verisignRootCA.cer` file.

If the VeriSign Trial Intermediate CA certificate is saved into a different file, run the `pskeymanager.cmd -import -trustcacerts` command to import that certificate also.

PeopleCode API Reference

This section covers these corrections to the *Enterprise PeopleTools 8.49 PeopleBook: PeopleCode API Reference*:

- Cookies property of the `IBInfo` class versus the Cookies property of the `IBConnectorInfo` collection.
- System setup for Java classes.
- Integration with Oracle BPEL Process Manager 10.1.3.x.

Note. After the title of each correction, an instruction in ***bold-italic*** indicates how the new material fits into the existing PeopleBook.

Cookies Property of the `IBInfo` Class versus the Cookies Property of the `IBConnectorInfo` Collection

The Cookies property of the `IBInfo` class was never implemented in PeopleTools even though the PeopleBooks indicated that it was. Instead, the Cookies property of the `IBConnectorInfo` collection was implemented with the functionality described for `IBInfo` Cookies.

The example code presented in the PeopleBooks is invalid and will fail PeopleCode syntax checking. Specifically, the following two lines of code are invalid:

```
&SalesCookies = &SalesResponse.IBInfo.Cookies;
```

```
&SalesRequest.IBInfo.Cookies = &SalesCookies;
```

See *Enterprise PeopleTools 8.49 PeopleBook: PeopleCode API Reference*, Message Classes, IInfo Class Properties, Cookies.

Therefore, the subsection of the PeopleBooks on the Cookies property of the IInfo class replaces the subsection of the PeopleBooks on the Cookies property of the IBCConnectorInfo collection with the exception noted below.

See *Enterprise PeopleTools 8.49 PeopleBook: PeopleCode API Reference*, Message Classes, ConnectorInfo Collection Properties, Cookies.

The code examples for receiving and sending cookies should be as follows. The bold text highlights the portions of code that have been corrected:

The following is an example of receiving cookies:

```
Local Message &SalesRequest, &SalesResponse;
Local Rowset &SALES_ORDER;
Global string &SalesCookies;

&SALES_ORDER = GetLevel0();
&SalesRequest = CreateMessage(OPERATION.SALES_ORDER_SYNC);
&SalesRequest.CopyRowsetDelta(&SALES_ORDER);

/* send the synchronous request; the return value is */
/* the response message object */
&SalesResponse = %IntBroker.SyncRequest(&SalesRequest);

/* Retrieve cookies from the response message */
&SalesCookies = &SalesResponse.IBInfo.IBCConnectorInfo.Cookies;
```

The following is an example of returning cookies:

```
Local Message &SalesRequest, &SalesResponse;
Local Rowset &SALES_ORDER;
Global string &SalesCookies;

&SALES_ORDER = GetLevel0();
&SalesRequest = CreateMessage(OPERATION.SALES_ORDER_SYNC);
&SalesRequest.CopyRowsetDelta(&SALES_ORDER);

/* Insert the cookies in the request message */
&SalesRequest.IBInfo.IBCConnectorInfo.Cookies = &SalesCookies;

/* Send the asynchronous request */
%IntBroker.Publish(&SalesRequest);
```

In addition, the cross-reference from the deprecated Cookies property of the Message class should be to the Cookies property of the IBCConnectorInfo collection.

See Also

Enterprise PeopleTools 8.49 PeopleBook: PeopleCode API Reference, Message Classes, Message Class Properties, Cookies

Enterprise PeopleTools 8.49 PeopleBook: PeopleCode API Reference, Appendix: Quick Reference for PeopleCode Classes; Deprecated Messaging PeopleCode Functions, Methods and Properties; Deprecated Message Class Methods and Properties

System Setup for Java Classes

When PeopleTools loads the Java Virtual Machine (JVM), PeopleTools builds a class path for the JVM from multiple elements. In the 8.49 PeopleBooks, the information on the elements used to construct this class path was incomplete.

The following information applies specifically to the “System Setup for Java Classes” section of the Enterprise PeopleTools 8.49 PeopleBook: PeopleCode API Reference. In addition, this information supersedes information in other PeopleBooks across PeopleTools 8.49 and all PeopleSoft Enterprise application pillars. Specifically, descriptions of the PS_CLASSPATH environment variable, the Add To CLASSPATH parameter of psappsrv.cfg, and those references to the CLASSPATH environment that describe access to third-party or custom Java classes by PeopleTools and PeopleCode are superseded by the following:

If you access only the classes that come defined with PeopleTools, you don't need to do any additional setup.

If you want to access third-party Java classes or your own custom Java classes in the PeopleTools environment (that is, access these classes through PeopleCode), you must place the class files in specified locations or include the class files in JAR files in specified locations.

Note. Oracle recommends using the utility that comes with the Java SDK for creating JAR files.

When PeopleTools loads the Java Virtual Machine (JVM), PeopleTools builds a class path for the JVM from the following elements.

The following numbering indicates the search order that the JVM would use for locating a specified Java class:

1. For each directory listed in the PS_CLASSPATH environment variable or for each JAR file explicitly specified in PS_CLASSPATH, the class files in the specified directory or JAR file.
2. For each directory listed in the “Add To CLASSPATH” parameter of the psappsrv.cfg configuration file:
 - f. Class files in the specified directory.
 - g. Class files in JAR files in the specified directory.
3. Class files in the PS_HOME\class directory.
4. Class files in JAR files in the PS_HOME\class directory.
5. Class files in the PS_HOME\appserv\classes directory.
6. Class files in JAR files in the PS_HOME\appserv\classes directory.

Note. PeopleTools uses the -classpath option when it loads the JVM, which overrides the CLASSPATH environment variable. Therefore, do not use the CLASSPATH environment variable to identify third-party or custom Java classes that you want to access in the PeopleTools environment.

Note. PeopleTools does not guarantee the order in which JAR files within a directory will be added to the class path. If it is necessary to impose a search order on JAR files, then the JAR files must be in separate directories.

Like most environment variables, you can specify more than one entry in PS_CLASSPATH. On Windows, the PS_CLASSPATH entries are separated by semicolons. On Unix, they're separated by colons.

The following PS_CLASSPATH is for Windows:

```
c:\myjava;d:\myjava\com\mycompany\myproduct\myjar.jar; . . .
```

The following PS_CLASSPATH would be for Unix:

```
/etc/myjava:/home/me/myjava/com/mycompany/myproduct/myjar.jar: . . .
```

When developing your own classes you must be aware that most JVMs cache the class definitions. This means that even if you update the class files, a running JVM (inside an application server, for example) has already loaded and is referencing the old versions of the class files. The JVM won't pick up the new versions of the class files. You must restart the application server to make the JVM reload the updated classes.

See Also

Enterprise PeopleTools 8.49 PeopleBook: PeopleCode API Reference, Java Class, System Setup for Java Classes.

PeopleSoft Enterprise Components PeopleBook (for any PeopleSoft Enterprise application pillar): Setting Up the Credit Card Interface.

Integration with Oracle BPEL Process Manager 10.1.3.x

In PeopleTools 8.49-07A (patch 7A), the UpdateConnectorResponseProperties method was added to the BPELUtil class. The documentation for this enhancement contains two subsections:

- UpdateConnectorResponseProperties.
- Integrating Worklist applications with Oracle BPEL Process Manager 10.1.3.x.

See the subsection *"UpdateConnectorResponseProperties"* for a supplement to the Enterprise PeopleTools 8.49 PeopleBook: PeopleCode API Reference.

If you want to integrate a Worklist application with Oracle BPEL Process Manager 10.1.3.x, then you must manually update two delivered methods in two Worklist handler classes. See the subsection "Integrating Worklist Applications with Oracle BPEL Process Manager 10.1.3.x" for instructions on how to do this.

UpdateConnectorResponseProperties

Syntax

```
UpdateConnectorResponseProperties (&Msg)
```

Description

Prior to sending the response message, use this method to update the connector properties for an asynchronous PeopleSoft request/response web service that is consumed by a BPEL process. Among several connector property changes that it makes, this method adds a "content-type" connector property of "text/xml."

Note. The UpdateConnectorResponseProperties method is not required with Oracle BPEL Process Manager 10.1.2 or earlier supported versions of Oracle BPEL Process Manager; however, it can be used with those versions.

Parameters

&Msg The message object to be sent to the BPEL process as the response of an asynchronous PeopleSoft request/response web service.

Returns

None.

Example

The following sample handler PeopleCode program processes requests for an asynchronous PeopleSoft request/response web service. A call is made to UpdateConnectorResponseProperties just before sending the response.

```
import PS_PT:Integration:INotificationHandler;
import PT_BPEL:BPELUtil;

class InboundASyncResponseHandler implements
PS_PT:Integration:INotificationHandler
    method OnNotify(&MSG As Message);
end-class;

method OnNotify
    /* &MSG as Message */
    /* Extends/implements
PS_PT:Integration:INotificationHandler.OnNotify */
    Local PT_BPEL:BPELUtil &bpel;
    Local Message &response;
    Local string &payload;
    Local XmlDoc &xml;
    Local File &MYFILE;
    &bpel = create PT_BPEL:BPELUtil();
    &payload = "<?xml version='1.0'?><PSFTCalcResponseMessage
xmlns='http://xmlns.oracle.com/Enterprise/Tools/schemas/PSFTCALCRESP
ONSEMESSAGE.V1'><result
xmlns=''>9</result></PSFTCalcResponseMessage>";
    &xml = CreateXmlDoc(&payload);
    &response = CreateMessage(Operation.PSFTASYNCCALCULATE,
%IntBroker_Response);
    &response.SetXmlDoc(&xml);
    &response.IBInfo.WSA_MessageID = &MSG.IBInfo.WSA_MessageID;
    &response.IBInfo.WSA_ReplyTo = &MSG.IBInfo.WSA_ReplyTo;
    &bpel.UpdateConnectorResponseProperties(&response);
    %IntBroker.Publish(&response);
end-method;
```

See Also

Enterprise PeopleTools 8.49 PeopleBook: PeopleCode API Reference, BPEL Classes, BPELUtil Class Methods.

Integrating Worklist Applications with Oracle BPEL Process Manager 10.1.3.x

If you want to integrate a Worklist application with Oracle BPEL Process Manager 10.1.3.x through the PeopleSoft Worklist web service, then you must manually update the following two delivered methods in two Worklist handler classes:

- PT_WF_Worklist:WLEntryMarkedWorkedResp.SendCustomResponseWithAdditionalData.
- PT_WF_Worklist:WSWorklistEntry.SendErrorResponse.

Make the following changes to each class and method in order to integrate the PeopleSoft Worklist web service with Oracle BPEL Process Manager 10.1.3.x:

1. Add the following import statement before the beginning of the class definition:

```
import PT_BPEL:BPELUtil;
```

2. Add the following three statements just before the call to the IB Publish method—
%IntBroker.Publish(&respMsg):

```
Local PT_BPEL:BPELUtil &bpel;  
&bpel = create PT_BPEL:BPELUtil();  
&bpel.UpdateConnectorResponseProperties(&response);
```

See Also

Enterprise PeopleTools 8.49 PeopleBook: PeopleSoft Integration Broker, Integrating with BPEL Process-Based Services.

PeopleCode Language Reference

In the *Enterprise PeopleTools 8.49 PeopleBook: PeopleCode Language Reference, "Meta-SQL Reference,"* the %InsertSelectWithLongs meta-SQL construct was omitted.

Use %InsertSelectWithLongs instead of %InsertSelect when the fields to be inserted include long values (LongChar and Image fields).

SQR Language Reference for PeopleSoft

In the section, "Using the PSSQR.INI File and the PSSQR Command Line" the following note should be added:

Note. Avoid using a hyphen (-) or @ sign in the PSSQR.EXE command line, for instance as part of the run control ID. The hyphen (-) and @ sign characters are SQR reserved characters.

See *Enterprise PeopleTools 8.49 PeopleBook: SQR for PeopleSoft Developers, Using the SQR Command Line, Using Reserved Characters*

PeopleSoft Integration Broker

This section covers these corrections to the *Enterprise PeopleTools 8.49 PeopleBook: PeopleSoft Integration Broker*:

- Overriding Synchronous Timeout Intervals at Runtime
- Handling Cookies
- HTTP Connectors – Passing HTTP Parameters

Overriding Synchronous Timeout Intervals at Runtime

The code example in the Enterprise PeopleTools 8.49 PeopleBook: PeopleSoft Integration Broker, Sending and Receiving Messages, Generating and Sending Messages, Overriding Synchronous Timeout Intervals at Runtime section is incorrect.

The following incorrect code example appears in PeopleBooks:

```
&MSG.SetXmlDoc(( &xmlReq );  
&MSG.IBInfo.LoadConnectorPropFromNode(Node.EAI)  
&MSG.IBInfo.SyncServiceTimeout(360000);  
&MSG.IBInfo.ConnectorOverride = true;  
&MSG_Resp = %IntBroker.SyncRequest(&MSG, Node.EAI);  
&xmlResponseDoc = &MSG.GetXmlDoc();
```

In the previous example an extra parenthesis appears in the first line. In addition the SyncServiceTimeout interval value is incorrectly included in parentheses, and instead should be set equal to a value.

The following code snippet shows the corrected example:

```
&MSG.SetXmlDoc(&xmlReq);  
&MSG.IBInfo.LoadConnectorPropFromNode(Node.EAI);  
&MSG.IBInfo.SyncServiceTimeout = 360000;  
&MSG.IBInfo.ConnectorOverride = True;  
&MSG_Resp = %IntBroker.SyncRequest(&MSG, Node.EAI);  
&xmlResponseDoc = &MSG.GetXmlDoc();
```

Handling Cookies

Several code examples in the Enterprise PeopleTools 8.49 PeopleBook: PeopleSoft Integration Broker, Sending and Receiving Messages, Generating and Sending Messages, Handling Cookies are incorrect.

Receiving Cookies

The following incorrect code example appears in the Handling Cookies, Receiving Cookies section:

```
Local Message &SalesRequest, &SalesResponse;  
Local Rowset &SALES_ORDER;  
Global string &SalesCookies;
```

```

&SALES_ORDER = GetLevel0();
&SalesRequest = CreateMessage(OPERATION.SALES_ORDER_SYNC);
&SalesRequest.CopyRowsetDelta(&SALES_ORDER);

/* Send the synchronous request; the return value is the response message
object */
&SalesResponse = &SalesRequest.SyncRequest();

/* Retrieve cookies from the response message */ &SalesCookies =
&SalesResponse.IBInfo.Cookies;

```

The following code snippet shows the corrected example. The code that has been corrected is in **bold** and *emphasis*:

```

Local Message &SalesRequest, &SalesResponse;
Local Rowset &SALES_ORDER;
Global string &SalesCookies;

&SALES_ORDER = GetLevel0();
&SalesRequest = CreateMessage(OPERATION.SALES_ORDER_SYNC);
&SalesRequest.CopyRowsetDelta(&SALES_ORDER);

/* Send the synchronous request; the return value is the response message
object */
&SalesResponse = %IntBroker.SyncRequest(&SalesRequest);

/* Retrieve cookies from the response message */
&SalesCookies = &SalesResponse.IBInfo.IBConnectorInfo.Cookies;

```

Sending Cookies

The following incorrect code example appears in the Handling Cookies, Sending Cookies section:

```

Local Message &SalesRequest, &SalesResponse;
Local Rowset &SALES_ORDER;
Global string &SalesCookies;

&SALES_ORDER = GetLevel0();
&SalesRequest = CreateMessage(Message.SALES_ORDER_SYNC);
&SalesRequest.CopyRowsetDelta(&SALES_ORDER);

/* Insert the cookies in the request message */
&SalesRequest.IBInfo.Cookies = &SalesCookies;

/* Send the asynchronous request */
%IntBroker.Publish(&SalesRequest);

```

The following code snippet shows the corrected example. The code that has been corrected is in **bold** and *emphasis*:

```

Local Message &SalesRequest, &SalesResponse;
Local Rowset &SALES_ORDER;
Global string &SalesCookies;

```

```

&SALES_ORDER = GetLevel0();
&SalesRequest = CreateMessage(Message.SALES_ORDER_SYNC);
&SalesRequest.CopyRowsetDelta(&SALES_ORDER);

/* Insert the cookies in the request message */
&SalesRequest.IBInfo.IBConnectorInfo.Cookies = &SalesCookies;

/* Send the asynchronous request */
%IntBroker.Publish(&SalesRequest);

```

HTTP Connectors – Passing HTTP Parameters

Several codes example for passing HTTP parameters in the Enterprise PeopleTools 8.49 PeopleBook: PeopleSoft Integration Broker, Using Listening Connectors and Target Connectors, Working with HTTP Connectors, Complying with Message Formatting and Transmission Requirements, Passing HTTP Parameters are incorrect.

Passing HTTP Parameters in Message Headers

The format template for passing HTTP message parameters in an HTTP message header in PeopleBooks contains an error. The following example appears in PeopleBooks (required parameters are shown in emphasis):

```

OperationName: OperationName
OperationType: Sync|Async|Ping
From: RequestingNode
Password: Password
OrigUser: OrigUser
OrigNode: OrigNode
OrigProcess: OrigProcess
OrigTimeStamp: OrigTimeStamp
FinalDestination: FinalDestination
To: DestinationNode
SubQueue: SubQueue
NonRepudiation: Y|N

```

The following example shows the correct format. Required parameters are shown in emphasis and the code that has been corrected is underlined:

```

OperationName: OperationName
OperationType: sync|async|ping
From: RequestingNode
Password: Password
OrigUser: OrigUser
OrigNode: OrigNode
OrigProcess: OrigProcess
OrigTimeStamp: OrigTimeStamp
FinalDestination: FinalDestination
To: DestinationNode
SubQueue: SubQueue
NonRepudiation: Y|N

```

Passing HTTP Parameters in Query Strings

The format template for passing HTTP message parameters in an HTTP query string in PeopleBooks contains an error. The following example appears in PeopleBooks (required parameters are shown in emphasis):

```
http://gatewayserver/PSIGW/HttpListeningConnector?&Operation=Operation  
Name&OperationType=[Sync|Async|Ping]&From=RequestingNode&Password=  
Password&OrigUser=OrigUser&OrigNode=OrigNode&OrigProcess=OrigProcess&  
OrigTimeStamp=OrigTimeStamp&FinalDestination=FinalDestination&To=DestinationNode  
&SubChannel=SubChannel&NonRepudiation=[Y|N]&MessageVersion=MessageVersion
```

The following code snippet shows the corrected example. The code that has been corrected is in **bold** and *emphasis*. The code that has been corrected is underlined:

```
http://gatewayserver/PSIGW/HttpListeningConnector?&Operation=Operation  
Name&OperationType=[sync|async|ping]&From=RequestingNode&Password=  
Password&OrigUser=OrigUser&OrigNode=OrigNode&OrigProcess=OrigProcess&  
OrigTimeStamp=OrigTimeStamp&FinalDestination=FinalDestination&To=DestinationNode  
&SubQueue=SubQueue&NonRepudiation=[Y|N]&MessageVersion=MessageVersion
```