

HP TP Web Connector

Release Notes

December 2007

This document provides the release notes for the HP *TP Web Connector* Version 5.1. These release notes describe known problems with this product, and include hints and suggestions to help you use this product.

Operating System:

Microsoft Windows

Software Version:

HP TP Web Connector
Version 5.1

Hewlett-Packard Company
Palo Alto, California

© Copyright 2007 Hewlett-Packard Development Company, L.P.

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.

The information contained herein is subject to change without notice. The only warranties for HP products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. HP shall not be liable for technical or editorial errors or omissions contained herein.

Microsoft and Windows are US registered trademarks of Microsoft Corporation.

Java is a US trademark of Sun Microsystems, Inc.

Printed in the US

Contents

Preface	v
1 Installation Notes	
1.1 Configuration steps to Run TPware applications on Windows Vista	1-3
1.1.1 Setting up Visual Studio	1-3
1.1.2 Setting up IIS and .NET	1-3
1.1.2.1 Installing IIS Metabase Compatibility	1-3
1.1.2.2 Installing .NET service packs	1-4
1.1.2.3 Enabling ASP.NET v1.1 ISAPI Extension	1-4
1.1.2.4 Adding IgnoreSection Handler	1-4
1.1.2.5 Adding your application to the application pool	1-4
1.1.3 Setting the permissions for the virtual directory in IIS	1-5
1.1.4 Samples	1-5
1.1.4.1 TPware Web Connector: ADD Sample	1-5
1.1.4.2 TPware Web Connector: Employee Sample	1-5
2 Problems Fixed	
2.1 Problems Fixed in Version 5.1	2-1
2.2 Problems Fixed in Version 5.0A	2-1
2.3 Problems Fixed in Version 5.0	2-1
2.4 Problems Fixed in Version 4.5A	2-1
2.5 Problems Fixed in Previous Kits and ECO Releases	2-2
3 Known Problems	
4 Restrictions	
5 General Information	
5.1 General Notes	5-1
5.2 Programming C Clients	5-2
5.3 Programming Automation Clients	5-2
5.4 Programming Java Clients	5-3
5.5 Controlling Client Connections	5-5
5.6 Using the ACMS Gateway Adapter	5-6
5.7 Using the HP TP Web Connector Gateway for ACMS	5-6
5.8 Using the TPware Management Utility	5-6
5.9 Using the Client Build Utility	5-7
5.10 Samples	5-7

6 Troubleshooting Tips

6.1	Registering and Unregistering Automation DLLs	6-1
6.2	Interpreting Runtime Error Codes	6-2
6.2.1	Location of Runtime Error Code Information	6-2
6.2.2	Interpreting Automation Errors	6-2
6.3	Running Oracle Web Request Broker	6-2

Preface

Kit Updates

For notifications on future updates or changes to the HP TP Desktop Connector product kits, refer to the following website:
<http://h71000.www7.hp.com/commercial/tpwebconnector>.

Associated Documentation

In addition to these release notes, also refer to the following TP Web Connector documents:

Online Guides:

HP TP Web Connector Installation Guide

HP TP Web Connector Getting Started

Online help:

TPware Management Utility online help

TPware STDL online help

TPware Client Build Utility Online help

Installation Notes

The following sections describe installation notes.

- When installing this TPware kit the install procedure will detect the presence of a previously installed TPware kit and install the kit in the same directory.
- If you are upgrading from a previously installed TPware kit, all information stored by the TPware management GUI will be retained.
- You must shutdown all TPware processes and applications that use TPware prior to the installation of this kit.
- When upgrading from a previously installed TPware kit, you will not be asked for product license keys for products that are already installed.
- With this release the TPware management information will be migrated from `HKEY_LOCAL_MACHINE\SOFTWARE\HP\TPware` to `HKEY_LOCAL_MACHINE\SOFTWARE\Hewlett-Packard\TPware`. Once the migration is complete the TPware key under `HKEY_LOCAL_MACHINE\SOFTWARE\HP` will be removed.
- If TPware files are lost or damaged once the kit has been installed, reinstall the latest TPware kit in order to restore the files.
- When installing both the *HP TP Desktop Connector Version 5.1* and TP Web Connector on the same system, caution must be used. Both products allow connectivity to HP ACMS. With both products installed, HP TP Desktop Connector applications that use HP ACMS Gateway Connector interface will not work. However, HP TP Desktop Connector applications using the Client Services Interface (formerly known as ACMS Desktop Portable API) will not be affected.
- On rare occasions, a TPware product uninstall does not delete the TPware program folder and/or associated files.
Workaround: Delete the program folder and/or product files using the Windows Explorer.
- 'Error 115' during the installation of a TPware product indicates that a file is locked and will cause the installation to fail.
Workaround: Prior to installing any TPware ECOs or additional TPware products, all TPware applications and TPware product processes must be shutdown. If this error continues, then reboot the system and repeat the installation.
- The `regsvr32.exe` image which is referenced in the product documentation is not supplied with the TPware kit. This is a Microsoft image and must be obtained from Microsoft. This file is typically found in the `WinNT\system32` directory.

- There is only one uninstall procedure. When you run the uninstall procedure, all the TPware products currently installed are removed.
- When installing the TPware products in a directory that differs from a previous installation, check any of the user-defined environment variables to ensure that they are referencing the new installation directory.
- The installation procedure may run out of disk space while copying files due to customized disk configurations.

Workaround: Correct the problem and then perform a reinstall.

- If you need to uninstall or re-install the TPware product and want to save settings defined in the TPware Management Utility, perform the following before you uninstall. Note, this does not have to be done in order to install a TPware ECO.

1. Click on Start -> Run...
2. Enter regedit at the Open prompt.
3. Click Registry on the menu bar.
4. Select Export Registry File... on the Registry Menu.
5. Enter a separate file name in the File Name prompt for each of the settings to be saved.
6. In Export range, click on the Selected branch button, then enter the appropriate registry key:
 - HKEY_LOCAL_MACHINE\SOFTWARE\Hewlett-Packard\TPware\ACMSDA Settings
(for the ACMS settings)
 - HKEY_LOCAL_MACHINE\SOFTWARE\Hewlett-Packard\TPware\Group Settings
(for the Threads settings)
 - HKEY_LOCAL_MACHINE\SOFTWARE\Hewlett-Packard\TPware\Share Settings
(for the Sharing settings)
7. Click on the Save button.
8. Repeat steps 5 through 7 for each of the settings to be saved.

To restore the registry settings after you re-install, do the following:

1. Click on Start -> Run...
2. Enter regedit at the Open prompt.
3. Click Registry on the menu bar.
4. Select Import Registry File... on the Registry Menu.
5. Enter the file name in the File Name prompt. This should be the same file used in step 5 when the registry settings were saved.
6. Click on the Open button. You should see a message box telling you that the information has been successfully entered in the registry.
7. Repeat steps 5 through 6 for each of the settings to be restored.

1.1 Configuration steps to Run TPware applications on Windows Vista

HP TP Web Connector V5.1 is qualified on Windows Vista Enterprise. After installing the required software, complete the configuration steps described in this section to run the TPware applications on Windows Vista.

To run TPware applications on Windows Vista, the following software must be installed on your system:

- Windows Vista Enterprise
- Microsoft Visual Studio 2005
- .NET Framework 1.1
- JDK 1.4
- Web Server IIS 7.0

To run TPware applications on Windows Vista, complete the following configuration steps:

- Setting up Visual Studio
- Setting up IIS and .NET
- Setting the permissions for the virtual directory in IIS

1.1.1 Setting up Visual Studio

To setup Visual Studio, before you build any TPware application, run the following batch file from the command prompt:

```
C:\Program Files\Microsoft Visual Studio8\VC\vcvarsall.bat
```

Note: This setup is applicable only for the current command session.

For information on building any TPware application refer to [ACMSWebSamples.htm](#) and [ACMSTpDeskSamples.htm](#) included in the TPware V5.1 kit.

1.1.2 Setting up IIS and .NET

To setup .NET on an IIS server, follow these steps described in the following section:

1. Installing IIS Metabase Compatibility
2. Installing .NET service packs
3. Enabling ASP.NET v1.1 ISAPI Extension
4. Adding IgnoreSection Handler
5. Adding your Application to the Application pool

1.1.2.1 Installing IIS Metabase Compatibility

To install the "IIS Metabase Compatibility", follow these steps:

1. Click **Start→Control Panel→Programs and Features**.
2. In the left pane, click the "Turn Windows Features on or off" link to open the Windows Features dialog box.
3. Click the plus sign (+) to the left of Internet Information Services, and then click the plus sign (+) to the left of Web Management Tools.

4. Click the plus sign (+) to the left of IIS 6 Management Compatibility and select IIS Metabase and IIS 6 Configuration Compatibility.
5. Click **OK**.

1.1.2.2 Installing .NET service packs

To install the following .NET service packs, double-click the executable and click **Run** in the dialog box:

- .NET Framework Version 1.1 Redistributable Package
- .NET Framework Version 1.1 Service Pack 1
- ASP.NET Security Update for .NET Framework 1.1 SP1

Note: If you fail to install .NET Framework v1.1 service packs, the following data execution prevention error message is displayed:

IIS Worker Process has stopped working.

1.1.2.3 Enabling ASP.NET v1.1 ISAPI Extension

To enable ASP.NET v1.1 ISAPI Extension, follow these steps:

1. Click **Start→Control Panel→Programs and Features**.
2. In the left pane, click the "Turn Windows Features on or off" link to open the Windows Features dialog box.
3. Click the plus sign (+) to the left of Internet Information Services, and then click the plus sign (+) to the left of World Wide Web Services.
4. Click the plus sign (+) to the left of Application Development Features and select all the options.
5. Click **OK**.

1.1.2.4 Adding IgnoreSection Handler

ASP.NET displays runtime exceptions if you have configured IIS in the web.config file that are read by the ASP.NET applications. To make ASP.NET to ignore IIS configuration, add IgnoreSection Handler to the Framework v1.1 machine.config file.

To add IgnoreSection Handler to the machine.config file, follow these steps:

1. To open the Framework v1.1 machine.config file, run the following at the command prompt:

```
%windir%\Microsoft.NET\Framework\v1.1.4322\config\machine.config
```

2. Add the following lines in the <configSections> section:

```
<section name="system.webServer" type="System.Configuration.IgnoreSectionHandler,
System, Version=1.0.5000.0, Culture=neutral, PublicKeyToken=b77a5c561934e089" />
```

1.1.2.5 Adding your application to the application pool

During installation, Framework v1.1 creates an application pool called "ASP.NET 1.1" that is configured to load Framework v1.1 upon startup.

To add your application into this application pool, follow these steps:

1. From the command prompt, set the directory to

```
cd %windir%\system32\inetsrv
```

2. Run the following at the command prompt:

```
appcmd set app "Default Web Site/" /applicationPool:"ASP.NET 1.1"
```

If you want to create a new application pool, follow these steps:

1. From the command prompt, set the directory to

```
cd %windir%\system32\inetsrv
```

2. Run the following at the command prompt:

```
appcmd add apppool /name:"NewPool" /managedRuntimeVersion:"v1.1"
```

1.1.3 Setting the permissions for the virtual directory in IIS

To set the permissions for the virtual directory in IIS that you had created earlier, follow these steps:

1. Click **Start→Control Panel→Administrative Tools**.
2. Double-click Internet Information Services (IIS) Manager to open the IIS Management console.
3. In the left pane, click the virtual directory.
4. In the Features View, double-click Handler Mappings.
5. In the right pane, under Actions, click Edit Handler Permissions...
6. In the Edit Handler Permissions dialog box, select all the options including Execute.
7. Click **OK**.

1.1.4 Samples

1.1.4.1 TPware Web Connector: ADD Sample

For Auto:

After building the automation server component, to register the dll files, embed the manifest file into dll as shown in the following example:

```
mt.exe -manifest add_acms_appl_auto_acmsda.dll.manifest  
-outputresource: add_acms_appl_auto_acmsda.dll;2
```

For information on running the automation sample refer to ACMSWebSamples.htm in the TPware V5.1 kit.

1.1.4.2 TPware Web Connector: Employee Sample

After building automation server component, to register the dll files, embed the manifest file into dll as shown in the following example:

```
mt -manifest employee_acms_auto.dll.manifest  
-outputresource:add_employee_acms_auto.dll;2
```

For information on running the automation sample refer to ACMSWebSamples.htm in the TPware V5.1 kit.

Note

For more information on pre-requisites and installation instructions, refer to the README file.

Problems Fixed

2.1 Problems Fixed in Version 5.1

Following are the problems fixed in this release:

- The STDL compiler fails to compile STDL files using TP Web Connector V5.0. This problem is fixed. (WFM 3216438924-321)
- The TPware application requests use the ACMSDI server irrespective of the GatewayType setting. This problem is fixed. (WFM 3218421599-321)

With this release, the client application can either use the ACMSDI server or the ACMSDA Gateway based on the value of GatewayType set in the

HKEY_LOCAL_MACHINE\SOFTWARE\Hewlett-Packard\TPware\ACMSDA Settings registry.

- If the value of GatewayType is set to 1, then ACMSDI\$SERVER is used on the OpenVMS node. The ACMSDI\$SERVER gateway can be used to run both Desktop Connector and Web Connector samples.
- If the value of GatewayType is set to 0, then ACMSDA\$GATEWAY is used on the OpenVMS node. The ACMSDA\$GATEWAY gateway can be used to run only the Web Connector samples.

2.2 Problems Fixed in Version 5.0A

There are no problems fixed in this release.

2.3 Problems Fixed in Version 5.0

There are no problems fixed in this release.

2.4 Problems Fixed in Version 4.5A

Following were the problems fixed in Release 4.5A:

- Problems while registering EMPLOYEE_ACMS_AUTO.DLL. There were problems in registering the EMPLOYEE_ACMS_AUTO.DLL due to discrepancies in the .asp files. Due to this, the .asp files are modified in the employee sample application.
- The name of the EMPLOYEE_INFO_APPL_XXX sample is changed to EMPLOYEE_WEB_APPL sample.
- The following table lists the files that are replaced by new files in the ACMSDA045A.D saveset:

Existing . . .	Replaced By . . .
EMPLOYEE_INFO_TASK_GROUP.GDF	EMPLOYEE_WEB_GROUP.GDF
TPWARE_ADD_EMPL_INFO.TDF	EMPLOYEE_WEB_ADD.TDF
TPWARE_GET_EMPL_INFO.TDF	EMPLOYEE_WEB_GET.TDF
TPWARE_PUT_EMPL_INFO.TDF	EMPLOYEE_WEB_PUT.TDF

2.5 Problems Fixed in Previous Kits and ECO Releases

The following problems were fixed in releases prior to Release 1.2A:

- The ACMS Gateway output adapter would sometimes incorrectly return the error message -3103 when -3020 should have been returned instead.
- If the length of a text field in a record definition equaled 1024 and the Java input adapter was being used then the compilation of the task group and corresponding record definition(s) would fail.
- The Java adapter was incorrectly converting Unicode characters to UTF-8 instead of multibyte format.
- The TPware Add sample server code was not the same code that was in the ACMSxp V3.1 NT kit.
- The STDL compiler returned an error if the full pathname of the STDL file to be compiled contained embedded spaces.
- When interfacing with HP ACMS, floating point numbers with a high degree of precision were not correctly converted from OpenVMS format to Windows NT format.
- The Output Adapter drop down list for the client build utility did not list the HP ACMS output adapter type.
- A client using the HP ACMS Gateway output adapter would spin if the TP Web Connector Gateway was shutdown or crashed while the client was connected.
- The STDL compiler did not properly compile record definitions that contained embedded records when building the Java adapter.
- A memory corruption problem has been fixed in the HP ACMS Gateway output adapter.
- Failure to define `RPC_DEFAULT_ENTRY` via an environment variable or the TPware Management Utility caused the DCE NSID process to crash.
- If a user was a member of many Windows NT groups, the TPware management GUI would fail to start.
- The MSRPC output adapter did not correctly handle the selection of multiple servers with the same interface if one of the servers was down.
- Binding Timeout, if enabled, did not work.
- The STDL compiler truncated array names within records that exceeded 25 characters in length.
- The STDL compiler allowed numeric characters as the first character in a STDL `#include` preprocessor directive statement.

Known Problems

Following is the known problem from the earlier release:

Unsupported Product traces

The TPWARE installation has been overhauled and many currently unsupported products have been removed from the kit (a notable example is ACMSxp, which is now retired). Still there may be some traces of these products in the installation tree, which are not functional, but still get copied during the Installation.

Restrictions

There are no restrictions in this release of HP TP Web Connector.

General Information

5.1 General Notes

The following general notes apply to TP Web Connector from the earlier releases:

- If you try to use an ISAPI DLL and an Automation DLL in the same process, the second DLL used for task calls returns an error code of 0xFFFFF4444 and an error class of -4.

Workaround: Do not mix ISAPI and Automation DLLs in the same process.

- The number of concurrent connections for the TP Web Connector Gateway for ACMS should be limited in order to reduce the possibility of excessive runtime system resource usage. Set the maximum number of threads on the Threads tab in the Management Utility to limit the number of concurrent gateway connections.
- If you use the Default group to specify threads settings, then any changes made to the Default threads settings do not take effect dynamically. You must restart any application that uses the Default settings in order for the changes to take effect.
- When threads settings are specified for a named group, all processes accessing the group share a common throttle. If a group has "Execute call on a separate thread, but limit maximum threads" semantics, each process that accesses the group will create up to the specified number of worker threads, but only the maximum number of threads are allowed through the throttle.

For example, if a group named "add_task_group" has a limit of 10 threads, and there are two processes that call add_task_group, then each process creates up to 10 worker threads (20 total), but only 10 threads are allowed access to add_task_group's server.

- The error log reporting stdlog -b (before) and -s (since) switches are not working properly.
- Error logging message to the NT event viewer erroneously display binary characters and may not contain all the information returned to the client and information that is reported via the stdlog error logging utility when the STDLOG_LOG_FILE environment variable is set.
- When reading the online version of the product documents with Microsoft Internet Explorer V3.x, some of the hyperlinks do not work.

Workaround: Upgrade to Internet Explorer V5.5 or V6.0.

- The STDL compiler does not report an error when invalid communications adapters are used with the -C switch.
- A thread leak can occur whenever the WWW service is stopped and restarted without stopping the INETINFO process.

Workaround: Issue a NET STOP IISADMIN or alternatively, reboot your system.

5.2 Programming C Clients

In a user program, if `einfo.h` is included before `windows.h` and a message header file is then included, compilation errors occur.

Workaround: Include `windows.h` before `einfo.h`.

5.3 Programming Automation Clients

The following notes apply to Automation clients:

- There are certain times when you must unregister and register an Automation server. Please see the section “Registering and Unregistering Automation DLLs” for further information.
- This version has limited Automation error reporting. Please see the section on “Interpreting Runtime Error Codes” for more information.
- Arrays are base 0 in Automation.
- Automation does not support user-defined datatypes. To handle user-defined datatypes, such as records, each record must be defined as a separate object. For embedded records the software creates an object with the name `<field-name>_RECORD`. If two embedded records in the same task or processing group file have the same field name, then the software will declare two objects with the same name. This causes an error.

Workaround: For X/Open or MIA Syntax: Make sure embedded records in the same task or processing group have unique field names. For X/Open Syntax only: Declare the embedded records as separate datatype, then reference the user-defined type in the field definition.

For example, the following is invalid syntax for Automation adapter:

```
TYPE MOVE_REC1 IS RECORD
  FLD_1 IS INTEGER;
  FLD_2 IS OCTET;
  FLD_3 IS RECORD
    FLD3_1 TEXT SIZE 7;
    FLD3_2 INTEGER;
  END RECORD;
END RECORD;

TYPE MOVE_REC2 IS RECORD
  FLD_1 TEXT SIZE 8;
  FLD_2 DECIMAL STRING SIZE 3;
  FLD_3 IS RECORD
    FLD3_1 TEXT SIZE 7;
    FLD3_2 INTEGER;
  END RECORD;
END RECORD;
```

The following is valid X/Open syntax for Automation adapter:

```
TYPE USER_TYPE IS RECORD
  FLD3_1 TEXT SIZE 7;
  FLD3_2 INTEGER;
END RECORD;
```

```

TYPE MOVE_REC1 IS RECORD
    FLD_1 INTEGER;
    FLD_2 OCTET;
    FLD_3 USER_TYPE;
END RECORD;

TYPE MOVE_REC2 IS RECORD
    FLD_1 TEXT SIZE 8;
    FLD_2 DECIMAL STRING SIZE 3;
    FLD_3 USER_TYPE;
END RECORD;

```

5.4 Programming Java Clients

The following notes apply to Java clients:

- The Java adapter has been tested using Sun Microsystems' J2SE 1.4.2 SDK.
- Visual J++ Version 6.0 is required to run the Visual J++ Automation Add Sample.
- There is no support for bean serialization.
- Arrays (primitives, strings and objects) in the Java adapter are restricted to three dimensions.
- The STDL data type UUID is now converted to and from a Java String. This change to support the UUID data type requires that the Microsoft Visual C library RPCRT4.LIB is included in the link line when building a Java adapter DLL. The RPCRT4.LIB library is needed whether or not UUID types are used in the client application.
- The Java input adapter did not work with the Microsoft virtual machine. With the release of Microsoft SDK for Java, Version 3.1 there is now the option of using the Microsoft compiler and virtual machine. Use <http://www.microsoft.com/java/download.htm> build 3167 2/18/99 or later. When installed this version will provide a development SDK and virtual machine. It will also update the virtual machine for Visual Studio J++ if that product is installed. After the SDK installation, ensure that the executable path is set to locate the new jvc.exe and jview.exe images in preference to images from any earlier Microsoft development environment. The classpath might also need adjusting to locate the new classes.zip file.

When building a Java input adapter from the command console under the Microsoft environment, the environment variable STDL_JAVA_MS has to be defined to be a non-null value. When STDL_JAVA_MS is defined the STDL compiler will use ,code_example>(jvc.exe) instead of the normal javac.exe to compile the Java source files. Microsoft does not provide the Jar utility. The STDL compiler needs the Jar utility to be available in order to archive the class files. The Jar.exe utility can be obtained from the normal Java distribution or one of the IDEs and made available in the executable path.

To run the Microsoft virtual machine from the command console, execute jview.exe instead of java.exe.

It is possible to elect to run the Microsoft virtual machine at runtime while using the normal JDK for development.

- This kit supports a workaround to allow the generated Java classes to be placed into a package. The following is a description of the steps to be taken to allow the STDL compiler to include the classes in a package. These steps

are in addition to the directions for building Java clients provided in the HP TP Desktop Connector Getting Started guide.

You must ensure that the working directory and a sub directory used when STDL compiles the Java files matches the naming of the Java package. For the purpose of this description, assume that the package is to be named `emp.tpw`, the name of the working directory is `C:\dev\emp` and the name of the directory that will hold the Java files during compilation is called `tpw`. The Java compilation directory is created under the working directory, `C:\dev\emp\tpw`. There will be an additional temporary directory automatically created by the STDL compiler alongside the Java compilation directory.

1. Manually create the directories. The terminal directory names in the path have to match the names used in the package definition. For example:

```
C:\dev\emp // Normal development work directory for the make, stdl
              and result files
C:\dev\emp\tpw // Java compilation temporary directory
C:\dev\emp\temp // Automatically generated temporary directory
```

2. Define the environment variable `STDL_JAVA_PACKAGE=<package-name>` in a way that matches the Java package directories. For example:

```
STDL_JAVA_PACKAGE=emp.tpw
```

3. When the java classes are generated they will include the named package. In order for the Java compiler to locate the classes in the package at STDL build time, the working directory's parent needs to be specified in the classpath. For example:

```
SET CLASSPATH=%classpath%;C:\dev
```

4. In addition to generating class files the STDL compiler will also generate C include files to be used when building the adapter DLL. Add the Java compilation directory to the `INCLUDE` environment variable to allow the C compiler to locate the generated include files during the STDL build. For example:

```
SET INCLUDE=%include%;C:\dev\emp\tpw
```

5. When the STDL build completes, the generated DLL and JAR files that are required for the application will have been moved up into the working directory. The normal temporary directory is cleaned up after a STDL build. This is not the case with the Java compilation directory. The directory can be cleaned up by adding to the make file cleanup, a delete of the files in the Java compilation directory.

After the Jar file has been created, client programs can reference the class files in the package by including the Jar file in the class path and importing the package. For example, assume the jar file is named `employee.jar`:

```
SET CLASSPATH=%classpath%;C:\dev\emp\employee.jar
import emp.tpw.*; // In client application source file
```

- Records are defined as separate objects. For embedded records the software creates an object with the name `<field-name>`. If two embedded records in the same task or processing group file have the same field name, then the software will declare two objects with the same name. This causes an error.

Workaround: For X/Open or MIA Syntax: Make sure embedded records in the same task or processing group have unique field names. For X/Open Syntax only: Declare the embedded records as separate datatype, then reference the user-defined type in the field definition.

For example, the following code is invalid syntax for Java adapter:

```
TYPE MOVE_REC1 IS RECORD
  FLD_1 IS INTEGER;
  FLD_2 IS OCTET;
  FLD_3 IS RECORD
    FLD3_1 TEXT SIZE 7;
    FLD3_2 INTEGER;
  END RECORD;
END RECORD;

TYPE MOVE_REC2 IS RECORD
  FLD_1 TEXT SIZE 8;
  FLD_2 DECIMAL STRING SIZE 3;
  FLD_3 IS RECORD
    FLD3_1 TEXT SIZE 7;
    FLD3_2 INTEGER;
  END RECORD;
END RECORD;
```

The following is valid X/Open syntax for Java adapter:

```
TYPE USER_TYPE IS RECORD
  FLD3_1 TEXT SIZE 7;
  FLD3_2 INTEGER;
END RECORD;

TYPE MOVE_REC1 IS RECORD
  FLD_1 INTEGER;
  FLD_2 OCTET;
  FLD_3 USER_TYPE;
END RECORD;

TYPE MOVE_REC2 IS RECORD
  FLD_1 TEXT SIZE 8;
  FLD_2 DECIMAL STRING SIZE 3;
  FLD_3 USER_TYPE;
END RECORD;
```

5.5 Controlling Client Connections

When a client uses the ACMS Gateway adapter the connection established by the client is cached. By default, the connection is cached for up to 20 minutes of inactivity. If after 20 minutes the client has not used the connection, the connection will timeout due to inactivity. This value can be modified by entering a value in the Windows registry by using the registry editor. The registry value that needs to be added is located in the path:

HKEY_LOCAL_MACHINE\SOFTWARE\Hewlett-Packard\TPware\Cache Settings

The value that needs to be added is Binding Timeout the value represents the lifetime in minutes of the connection following a period of inactivity. If the value is set to zero the connection is never removed due to inactivity. To set the Binding Timeout value, do the following:

1. In the registry editor, highlight "Cache Settings".
2. Go to the Edit drop down tab and select "add value".
3. For the value name, enter Binding Timeout (note space in name).

4. For the datatype, select DWORD click on OK and enter a value.
5. Restart the client in order for the change to take affect.

5.6 Using the ACMS Gateway Adapter

The following notes apply to the ACMS Gateway adapter:

- The ACMS Gateway adapter does not support the following features:
 - Exchange Steps
 - Data Compression
 - SHOW_DESKTOP_USERS Utility Program
 - Nonblocking Calls
 - Forced Nonblocking Calls
 - Client Task Cancel Calls
 - non-Windows Client Platforms
 - non-TCP/IP transports
 - variant data types
- A stack overflow may occur when accessing ACMS from clients within web servers. If this happens, einfo will contain the following: eclass = -4, ecode = FFFFF444. (HP TP Web Connector)

Workaround: Use the Management Utility and set either the "Execute call on a separate thread" or "Execute call on a separate thread, but limit maximum threads to" execution semantics for the group.
- EINFO is not being cleared after an error occurs.

5.7 Using the HP TP Web Connector Gateway for ACMS

The release notes for the TP Web Connector Gateway for ACMS component (Version 5.0A) are contained in a separate document within the TP Web Connector Gateway for ACMS installation procedure and are installed in the following OpenVMS system directory:

```
SYS$HELP:ACMSDA050A.RELEASE_NOTES - text version
```

5.8 Using the TPware Management Utility

The following note applies to the TPware Management Utility:

- The TPware Management Utility does not save all changes before connecting to the next registry.

Workaround: Use the Computer tab only at the beginning of a TPware Management Utility session, make registry changes, then exit. If you want to use the Computer tab to connect to and make changes to multiple nodes, it is required that you exit the utility with "Close" or "OK" after managing each node.

5.9 Using the Client Build Utility

The following notes apply to the Client Build Utility:

- The Client Build Utility creates a process that runs `nmake` to create the client files. This process is visible on the desktop as a minimized window icon. If `nmake` runs for 10 minutes without returning, the Client Build Utility will timeout and return a Create Process error. However, the created process will still be running. If the application is unusually complex it may be a legitimately long build, in which case the process can be allowed to complete. Otherwise, the minimized window should be closed, which terminates the process.
- Rebuilding an existing Automation client generates the following error message.

```
Cannot create a file when that file already exists.  
NMAKE: fatal error U1077: 'move': return code '0x1'
```

Workaround: Delete the existing DLL before rebuilding.

5.10 Samples

The following notes apply to the samples provided in the kit:

- The Samples provided in this kit can be found under the "TPware Products" menu item under the "Programs" menu item on the "Start" menu.

Troubleshooting Tips

6.1 Registering and Unregistering Automation DLLs

There are certain times when the user must unregister and re-register an Automation server. If you are adding or removing methods, changing the number or type of arguments, or changing the fields in a record associated with an argument, you should unregister the existing Automation DLL using the `regsvr32 -u` command before you build the new DLL. Failure to issue the unregister command using the exact same DLL that was used when registering can cause the following problems:

- The unregister command will not be able to remove all associated Windows Registry entries.
- The unregister command will signal errors.

If you forgot to unregister before you built the new DLL and receive errors when unregistering, then select to continue if prompted. Next re-register the new DLL. If the Automation DLL registers successfully the new Automation interface should be available. If not try unregistering and registering again.

If you don't re-register an Automation DLL and have changed any of the information mentioned above, then the Automation server will not work correctly because the registered Automation interface will be out of sync with the Automation interface in the DLL.

If you receive the following error messages when registering or unregistering an Automation DLL, then check that the `STDL_SYS_DIR` environment variable is defined and in the current path. If you did not reboot after installing the kit you should do so.

- The dynamic link library `stdl_rtm.dll` could not be found in the specified path

```
LoadLibrary ("<name-of-dll-appears-here>") failed  
Get Last Error returns 0x0000007e
```
- Please check that the `STDL_SYS_DIR` environment variable is defined and in the current `PATH`. If you did not reboot after installing the kit you should do so.

6.2 Interpreting Runtime Error Codes

6.2.1 Location of Runtime Error Code Information

Other than Automation errors as described below, the TP Web Connector returns error information in `info.ecode`. These error codes and their descriptions can be found in the following files:

```
<installed-kit>\stdl\include\stdlrtl_msg.h  
<installed-kit>\stdl\include\acmsda_client_messages.txt
```

The `stdlrtl_msg.h` file contains error codes for TPware runtime exceptions. The range of these errors is 01 to 255 (0x01 to 0xFF).

The `acmsda_client_messages.txt` text files, found in the `TPWARE\stdl\include` directory, document client error codes. The range of these errors is -3000 to -3199 (0xFFFFF448 to 0xFFFFF381).

Please examine these files to evaluate and respond to client and/or gateway errors as logged in the ACMS Software Error log (SWLUP). Also see `SYS$HELP:ACMSDA$GATEWAY_MESSAGES.TXT` on the TP Web Connector Gateway system for ACMS (on OpenVMS) for evaluating gateway error information.

6.2.2 Interpreting Automation Errors

Runtime Automation error values are returned using a 32-bit number known as a result handle (i.e. `HRESULT`). The structure of a `HRESULT` value is defined by Microsoft. Automation errors returned by the TP Web Connector start at 214774989 (0x80041001). These values can be translated to error codes found in `stdlrtl_msg.h` using the following formula:

STDLE error code = <Automation-error-code-value> - 0x80041000

6.3 Running Oracle Web Request Broker

When running the Oracle Web Request Broker samples, the error "The requested access method is not allowed for that object" is returned if the virtual path to the web cartridge does not match the requested path. (Note that the virtual path for web cartridges is case sensitive.)