

MarkView

MarkView Reintegration after
an Oracle R12 Upgrade



Portions © 2004-2011 Kofax, Inc., 15211 Laguna Canyon Road, Irvine, California 92618, U.S.A. Portions © 2002-2006 Kofax Development GmbH. Portions © 1997-2006 Kofax Development UK Ltd. All Rights Reserved. Use is subject to license terms.

This product is protected by U.S. Patent Nos. 5,159,667 and 7,478,332.

Third-party software is copyrighted and licensed from Kofax's suppliers.

THIS SOFTWARE CONTAINS CONFIDENTIAL INFORMATION AND TRADE SECRETS OF KOFAX, INC. USE, DISCLOSURE OR REPRODUCTION IS PROHIBITED WITHOUT THE PRIOR EXPRESS WRITTEN PERMISSION OF KOFAX, INC.

Kofax, the Kofax logo, and MarkView are trademarks or registered trademarks of Kofax, Inc. in the U.S. and other countries. All other products names are the trademarks or registered trademarks of their respective owners.

U.S. Government Rights Commercial software. Government users are subject to the Kofax, Inc. standard license agreement and applicable provisions of the FAR and its supplements.

You agree that you do not intend to and will not, directly or indirectly, export or transmit the Software or related documentation and technical data to any country to which such export or transmission is restricted by any applicable U.S. regulation or statute, without the prior written consent, if required, of the Bureau of Export Administration of the U.S. Department of Commerce, or such other governmental entity as may have jurisdiction over such export or transmission. You represent and warrant that you are not located in, under the control of, or a national or resident of any such country.

DOCUMENTATION IS PROVIDED "AS IS" AND ALL EXPRESS OR IMPLIED CONDITIONS, REPRESENTATIONS AND WARRANTIES, INCLUDING ANY IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT, ARE DISCLAIMED, EXCEPT TO THE EXTENT THAT SUCH DISCLAIMERS ARE HELD TO BE LEGALLY INVALID.

This product includes iText Library - Copyright (c) 1999-2005 by Bruno Lowagie and Paulo Soares. The contents of iText Library are subject to the Mozilla Public License Version 1.1 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at <http://www.mozilla.org/MPL/>. Software distributed under the License is distributed on an "AS IS" basis, WITHOUT WARRANTY OF ANY KIND, either express or implied. See the License for the specific language governing rights and limitations under the License.

This product includes software developed by:

- Apache Software Foundation (<http://www.apache.org/>). Licensed under the Apache License, Versions 1.1 and 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at <http://www.apache.org/licenses/LICENSE-1.1> or <http://www.apache.org/licenses/LICENSE-2.0>. Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
- Jean-Loup Gailly and Mark Adler - Copyright (c) 1995-2004.
- The JDOM Project (<http://www.jdom.org/>).
- Jaxen Project (<http://jaxen.org/>). Copyright © 2001 werken digital. All Rights Reserved.

Copyright (c) 2000 – 2006 The Legion Of The Bouncy Castle (<http://www.bouncycastle.org>)

All other third-party products or brands mentioned in this documentation are or may be trademarks or registered trademarks and are the property of their respective owners.

LibTiff.Net

Copyright (c) 2008-2011, Bit Miracle

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- Neither the name of the Bit Miracle nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL BIT MIRACLE BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Contents

CHAPTER 1	Reintegrating MarkView with Oracle	7
	Re-create Grants and Synonyms	7
	Re-create Grants and Synonyms for Advisor (Advisor only)	7
	Compile All Invalid Objects in the MarkView Schema	8
	Re-enable Triggers in the MarkView Schema	8
	Re-create Advisor Triggers (Advisor only)	8
	Migrate MarkView Data to R12 Format	8
	Re-create MarkView Design Time Data	9
	What to Do Next	9
CHAPTER 2	Reinstalling MarkView Oracle Objects	11
	Redefining the c_MARKVIEW_TOP Environment Variable	11
	Setting Up c_MARKVIEW_TOP in Oracle E-Business Suite	12
	Running the MarkView Oracle Objects Installer	12
CHAPTER 3	Configuring Oracle Forms	15
	Testing the c_MARKVIEW_TOP Variable	15
	Using the Oracle Forms Builder	16
	Setting Up the Forms Builder Environment	16
	Launching the Oracle Forms Builder	17
	Updating the Oracle Attachments Form (FNDATTCH.fmb)	18
	Step 1: Open the FNDATTCH and ATTCH170 Forms	18
	Step 2: Paste the GET_MV_DOCUMENT_ID Item into the FNDATTCH Form	18
	Step 3: Paste the MarkView Package into the FNDATTCH Form	19
	Step 4: Edit the PRE-FORM Trigger	19
	Step 5: Edit the PRE-QUERY Trigger	19

Step 6: Edit the POST-QUERY Trigger	20
Step 7: Edit the PRE-INSERT Trigger	20
Step 8: Edit the WHEN-NEW-RECORD-INSTANCE Trigger	20
Step 9: Create the KEY-NEXT-ITEM Trigger	21
Step 10: Edit the WHEN-VALIDATE-ITEM Trigger for the DATATYPE_NAME Item	21
Step 11: Edit the WHEN-VALIDATE-ITEM Trigger for the CATEGORY_DESCRIPTION Item	21
Step 12: Add an Alert to the FNDATTCH Form	22
Step 13: Edit the WHEN-BUTTON-PRESSED Trigger	22
Step 14: Save Your Changes	22
Updating the Custom Library	23
Step 1: Locate and Copy the Modified CUSTOM.pll File (Customized CUSTOM.pll Only)	23
Step 2: Open CUSTOM.pll for Editing	24
Step 3: Attach Libraries (Customized CUSTOM.pll Only)	24
Step 4: Edit the zoom_available, style, and event Procedures	25
Step 5: Compile and Copy CUSTOM.pll	25
Configuring Oracle Forms for Multi-Language Support (Optional)	26
Enabling Invoice Workbench	26
Image-Enable the Default Invoice Workbench Form	27
Modify SFXINWKB for GetNext and Open Work Item	27
Reintegrating the Quick Invoices Form	29
CHAPTER 4	Configuring MarkView for Expense Management 33
Configuring Oracle Expenses Workflow	33
Setting Administrator Access Level	33
Opening Expenses Workflow	34
Adding Functions to the Workflow Process	35
Configuring the Workflow Background Process	42
Updating the Fax Number	43
Configuring the Missing-Receipt Timeout	44

Extract the zip file for your version of MarkView. You can place the files on any machine with access to the database server. Using the scripts provided in the zip file, complete the tasks in this section.

Re-create Grants and Synonyms

- 1 Change to the directory: **1_grants_synonyms**.
- 2 Execute the script:
`@mvoa_create_grants_synonyms.sql`
- 3 Follow the prompts for entering the MarkView and Apps schema information.
- 4 When prompted for the Oracle E-Business Suite version, enter **11.5** (this is version to select for R12).

Re-create Grants and Synonyms for Advisor (Advisor only)

- 1 Change to the directory: **2_grants_synonyms_advisor**.
- 2 Log in to SQL*Plus as the Oracle E-Business Suite schema owner (APPS).
- 3 Execute the script:
`@grantAccessToMV.sql`
- 4 When prompted for the secondary_user, enter the MarkView schema name.

- 5 Log in to SQL*Plus as the MarkView schema owner.
- 6 Execute the script:
`@markview_synonyms.sql`
- 7 When prompted for the secondary_user, enter the Oracle E-Business Suite schema name.

Compile All Invalid Objects in the MarkView Schema

- 1 Change to the directory: `3_gencomp`.
- 2 Log in to SQL*Plus as the MarkView schema owner.
- 3 Execute the script:
`@Gencomp.sql`

Re-enable Triggers in the MarkView Schema

- 1 Change to the directory: `4_triggers`.
- 2 Log in to SQL*Plus as the MarkView schema owner.
- 3 Execute the script:
`@Genenable.sql`

Re-create Advisor Triggers (Advisor only)

- 1 Change to the directory: `5_triggers_advisor`.
- 2 Log in to SQL*Plus as the MarkView schema owner.
- 3 Execute the script:
`@recreate_triggers.sql`

Migrate MarkView Data to R12 Format

- 1 Change to the directory: `6_r12_patches`.
- 2 Log in to SQL*Plus as the Oracle E-Business Suite schema owner (APPS).

- 3 Execute the script:
`@oracle-integ-apps-1000-update-datatype-category.sql`
- 4 Execute the script:
`@oracle-integ-apps-1001-update-file-name.sql`
- 5 Execute the script:
`@oracle-integ-apps-1002-update-negative-filenames.sql`

Re-create MarkView Design Time Data

- 1 Change to the directory: `7_apps_seed_data`.
- 2 Log in to SQL*Plus as the Oracle E-Business Suite schema owner (APPS).
- 3 Execute the script:
`@mvoa_seed_all_actions_apps.sql`
- 4 Execute the script:
`@SFXINWKB.sql`

What to Do Next

- Reinstall Oracle Forms—see [Reinstalling MarkView Oracle Objects on page 11](#).
- Configure MarkView for Expense Management, if installed—[Configuring MarkView for Expense Management on page 33](#).
- Reconfigure MarkView Self-Service Invoice (SSI) for Oracle R12:
 - If you are changing to Regime to Rate tax entry, open MarkView Administration and change the preference value for SSI_TAX_ENTRY_MODE to **REGIME_TO_RATE**.
 - If you use SSI Templates that contain tax information, re-create the templates using the SSI interface. Create new invoice requests that contain all of the information to be in the template including tax. Save the new request and overwrite the existing templates.

Redefining the c_MARKVIEW_TOP Environment Variable

- 1 Open Oracle E-Business Suite with the System Administrator responsibility.
- 2 Navigate to **System Administration > Oracle Applications Manager > OAM Setup > Site Map > AutoConfig**.

The Applications Dashboard opens the Context Files screen.

- 3 Go to the Related Links section at the bottom of the window and click **Manage Custom Parameters**.
- 4 On the Customized Parameters page, click **Add**.
- 5 Step 1: Select **Applications Tier** and click **Next**.
- 6 Step 2: Set up the next page as follows:

Field	Description and Setting
OA_VAR	The <i>name</i> of the environment variable. Select c_MARKVIEW_TOP .
Default Value	The <i>c_MARKVIEW_TOP directory</i> . %s_at%/<your TOP name>/12.0.0 (Use 12.0.0 for R12.1.)
Title	MarkView Top
Description	MarkView for Oracle Forms Top
OA Type	PROD_TOP

If you encounter the error: "Customization Is Not Allowed On Different Versions of Configuration Files," see Metalink Note 762590.1.

- 7 Click **Next**.
- 8 On the Step 3 confirmation page, verify your values and click **Next**.
Step 4 summarizes the files that change when you change the TOP variable. You can read the XML file in your shell by viewing \$CONTEXT_FILE.
- 9 Click **Finish**.
- 10 Go to [Setting Up c_MARKVIEW_TOP in Oracle E-Business Suite](#). AutoConfig will load the contents of \$CONTEXT_FILE into your shell and the Oracle Forms environment through \$ORACLE_CONFIG_HOME/forms/server/default.env.

Setting Up c_MARKVIEW_TOP in Oracle E-Business Suite

Run AutoConfig

Running AutoConfig loads the c_MARKVIEW_TOP variable into your environment and ensures that future runs of AutoConfig will not revert the setting. See Oracle documentation for more information about AutoConfig.

Log in to the database server as the APPS Manager and run the following script:

```
$INST_TOP/admin/scripts/adautocfg.sh
```

Restart Oracle E-Business Suite

After AutoConfig finishes running, stop and start Oracle Applications.

Confirm that the Variable Is Set

- 1 Log out of your shell and close your remote connection.
- 2 Log in to the shell and display your variable by issuing the following shell command:

```
echo $c_MARKVIEW_TOP
```

If after running AutoConfig and logging into your shell, the variable is unavailable, see MetaLink Note: 461326.1 for a bug that may apply to your system.

Running the MarkView Oracle Objects Installer

- 1 On the Oracle E-Business Suite server, open a command shell and log in as the Oracle E-Business suite owner. The default account for this user is **applmgr**.
- 2 Locate your current version of the MarkView installer in the distribution directory, for example, \$HOME/mvforms_dist.bin.

- 3 Identify the target MarkView install directory for Oracle Objects modules. This directory was created during the initial MarkView installation, for example, `$HOME/markview/`.
- 4 Make sure that all environment variables are correctly set up for this account, check:
 - `$PATH`—Confirm Java version 1.5 comes first (or run `java -version`).
 - `$APPL_TOP`—Confirm this is the correct instance of Oracle E-Business Suite.
 - `$FORMS_PATH` (in 11i, this path variable is called `FORMS60_PATH`)—Confirm that it includes `$AU_TOP/resource` and `$AU_TOP/forms/US`.
- 5 Ensure that the application server is running.
- 6 Navigate to the bin directory in the MarkView installer distribution directory and start the installer by running the following command:

```
./install.sh
```

Note: You must run the installer twice—once to reinstall existing components and again to install the R12 integration.

 - The first time you run the installer, select **Partial Install** and select **Oracle Forms Integration**. If you have MarkView for Expense Management installed, also select **OA Framework Integration for Expense**.
 - The second time you run the installer, select **Standard Full Install** and select **OA Framework Integration for AP (R12 Only)**.
- 7 Click **Next**.
- 8 Select the installation type and click **Next**.
- 9 Click **Select Folder**, navigate to the target install directory that was used for the initial installation, and click **Next**.
- 10 If prompted, enter or select the Target Registry Services URL and click **Next**. For example, enter `http://my_oas_host.company.com:7777/mvstrs/markview`.
- 11 Enter your MarkView Schema Password and click **Next**.
- 12 Enter your Oracle E-Business Password and click **Next**.
The order and appearance of the next few steps varies depending on your selected installation options.
- 13 Review the values displayed and click **Next**:
 - `APPL_TOP` value—Confirm that this is the instance of Oracle E-Business Suite on which to install Forms.

- AU_TOP and FND_TOP values—Confirm that this is the instance of Oracle E-Business Suite on which to install Forms (both directories should be under \$APPL_TOP).
- \$c_MARKVIEW_TOP value—Confirm that this is the value that you set up in [Redefining the c_MARKVIEW_TOP Environment Variable on page 11](#).

14 Review the Summary and click **Next**.

15 Click **Install**.

The installation finishes with a message indicating whether the installation is successful.

The installation script creates log files in the target MarkView install directory.

Once the installation is complete, follow the instructions in [Chapter 3, "Configuring Oracle Forms,"](#) to configure the Oracle Forms.

Configuring Oracle Forms

Configuring Oracle Forms involves completing the following processes, which this chapter describes:

- 1 [Testing the c_MARKVIEW_TOP Variable](#) (this page)
- 2 [Using the Oracle Forms Builder](#) on page 16
- 3 [Updating the Oracle Attachments Form \(FNDATTCH.fmb\)](#) on page 18
- 4 [Updating the Custom Library](#) on page 23
- 5 [Configuring Oracle Forms for Multi-Language Support \(Optional\)](#) on page 26
- 6 [Enabling Invoice Workbench](#) on page 26

All installations require the completion of the first four processes. If your system uses a language other than US English, configure Oracle Forms for multi-language support.

Testing the c_MARKVIEW_TOP Variable

Set the MARKVIEW_TOP environment variable for MarkView Oracle Forms before you install Forms. See [Redefining the c_MARKVIEW_TOP Environment Variable](#) on page 11 for more information.

To test whether the MARKVIEW_TOP variable is set properly:

- 1 Log in to Oracle Applications and open any form except Invoice Workbench.
- 2 Select **Help > Diagnostics > Examine**.

- 3 When prompted, enter the apps password.
- 4 Navigate to **Block** and press **CTRL-L**. An LOV appears.
- 5 Select **\$ENVIRONMENT\$**.
- 6 In **Field**, type **C_MARKVIEW_TOP** and press **Tab**. Use *all* uppercase letters in this field.

Caution: If you do not use all uppercase letters, you may receive a false positive result when you verify that the variable is set.

Caution: Do not press **CTRL-L** to display the list of values (LOV). If you press **CTRL-L**, **C_MARKVIEW_TOP** does not appear in the LOV.

- 7 Press **Tab** to navigate to **Value**.
 - If the pathname of **c_MARKVIEW_TOP** appears in the field, the variable is configured correctly. Proceed to [Using the Oracle Forms Builder on page 16](#).
 - If the pathname does not appear in **Value**, Oracle Forms returns an error. Complete the following steps.
 - a Verify that you configured **c_MARKVIEW_TOP** properly. (See [Reinstalling MarkView Oracle Objects on page 11](#).)
 - b Restart the application server and repeat steps 1 through 7.

Using the Oracle Forms Builder

Use Oracle Forms Builder to edit Oracle Forms. Before launching Oracle Forms Builder you must set environment variables.

Log in to the Forms host as a user with write privileges in the **\$APPL_TOP** directory, for example, **applmgr**.

Setting Up the Forms Builder Environment

X-Windows Client

Because Oracle Forms Builder runs through a windowing system, you must initiate this session on a desktop environment or from an X Windows client.

If running from an X Windows client, set the **DISPLAY** environment variable to the IP address of your local machine. For example, **export DISPLAY=255.255.255.255:0**.

Set Environment Variables

The \$APPL_TOP directory contains configuration files that have the suffix .env and set environment variables. These environment variables must be set properly for you to compile forms and libraries.

Before invoking Oracle Forms Builder, invoke the appropriate configuration file to establish the proper working environment. For example, when using AutoConfig, the configuration filename is

APPSsid_hostname.env

where *sid* is the database SID and *hostname* is the server name.

To invoke the configuration file, use the shell command that runs scripts or batch files in your current environment. For example, depending on your shell, use one of the following commands to invoke the configuration file:

- C shell—source APPSsid_hostname.env
- KornShell—. APPSsid_hostname.env

The configuration file establishes several environment variables, including the following:

Table 3-1. Key Environment Variables

Variable	Variable Content
APPL_TOP	The name of the directory where the configuration files reside.
FORMS_PATH in R12	One or more directories required to compile forms and libraries.
AU_TOP	The source code for all forms.

Oracle Forms Path

The Oracle Forms Builder uses the variable FORMS_PATH.

- 1 Verify that the FORMS_PATH includes **\$AU_TOP/resource**.
- 2 Verify that the Oracle Forms Builder has **\$AU_TOP/forms/US** in its path by doing one of the following:
 - Launch Oracle Forms Builder from the directory **\$AU_TOP/forms/US**.
 - Set the variable FORMS_PATH to include **\$AU_TOP/forms/US** at the end.

Launching the Oracle Forms Builder

Enter the following command in a shell:

frmbld

Updating the Oracle Attachments Form (FNDATTCH.fmb)

By default, Oracle Applications support attachments on most standard Oracle Applications forms. A user clicks the Attachments (paper clip) button on the Oracle Applications toolbar, which starts the FNDATTCH form. The FNDATTCH form searches for and returns any attachments to the current Oracle Applications record. Modify the FNDATTCH form so that it shows attachments with MarkView categories.

Many of the changes to the FNDATTCH form are encapsulated in a source form that is installed by the MarkView Installer. The source form is named 170ATTCH. As you update the FNDATTCH form, you will cut-and-paste some of the MarkView modifications from 170ATTCH to FNDATTCH.

Complete the steps in the following sections to update the FNDATTCH form.

Note: This procedure assumes that the MarkView base path is the default `$c_MARKVIEW_TOP`.

Step 1: Open the FNDATTCH and ATTCH170 Forms

- 1 Copy `$AU_TOP/forms/US/FNDATTCH.fmb` to `$c_MARKVIEW_TOP/forms/US`.
- 2 Launch the Oracle Forms Builder and select **File > Open** to open `$c_MARKVIEW_TOP/MVOA/<version_number>/forms/170ATTCH.fmb`, where *version_number* is the current version of MarkView for Oracle Applications. The name of the form, ATTCH170, appears at the top of the navigation tree.
- 3 Select **File > Open** and open `$c_MARKVIEW_TOP/forms/US/FNDATTCH.fmb`. The name of the form, FNDATTCH, appears at the top of the navigation tree.

Step 2: Paste the GET_MV_DOCUMENT_ID Item into the FNDATTCH Form

To paste the GET_MV_DOCUMENT_ID item into the FNDATTCH form:

- 1 Copy the following item:
`ATTCH170>Data Blocks>DOCUMENT_CONTROL>Items>GET_MV_DOCUMENT_ID`
- 2 Navigate to the following FNDATTCH window and paste GET_MV_DOCUMENT_ID into the FNDATTCH form:
`FNDATTCH>Data Blocks>DOCUMENT_CONTROL>Items`

Step 3: Paste the MarkView Package into the FNDATTCH Form

- 1 In the Object Navigator, expand **ATTCH170 > Program Units**.
- 2 Right-click the following packages and select **Copy**:
 - MARKVIEW* (Package Spec)**
 - MARKVIEW* (Package Body)**
- 3 In the Object Navigator, right-click **FNDATTCH > Program Units** and select **Paste**.

Step 4: Edit the PRE-FORM Trigger

- 1 In the Object Navigator, select **FNDATTCH > Triggers > PRE-FORM**.
- 2 Right-click and select **PL/SQL Editor**.
- 3 Make the following changes to the PRE-FORM trigger:
 - a Change the version number to indicate that this form was modified for MarkView integration by appending **.170** to the version number. For example, change the form version number from:

```
FND_STANDARD.FORM_INFO('$Revision: 115.155.11591.2')
```

to:

```
FND_STANDARD.FORM_INFO('$Revision: 115.155.11591.2.170')
```
 - b Add the following code to the end of the trigger:

```
MarkView.Event('PRE-FORM');
```
- 4 Click **Compile** and **Close**.

Step 5: Edit the PRE-QUERY Trigger

- 1 In the Object Navigator, select:
FNDATTCH > Data Blocks > DOCUMENT_HEADER > Triggers > PRE-QUERY
- 2 Right-click and select **PL/SQL Editor**.
- 3 In the PL/SQL Editor, add the following line of code to the end of the trigger:

```
MarkView.Initialize;
```
- 4 Click **Compile** to save your changes.

Step 6: Edit the POST-QUERY Trigger

- 1 In the Object Navigator, select:
FNDAATCH > DataBlocks > DOCUMENT_HEADER > Triggers > POST-QUERY
- 2 Right-click and select **PL/SQL Editor**.
- 3 Add the following line of code to the end of the trigger:
MarkView.Event('POST-QUERY');
- 4 Click **Compile** and **Close**.

Step 7: Edit the PRE-INSERT Trigger

- 1 In the Object Navigator, select:
FNDAATCH > Data Blocks > DOCUMENT_HEADER > Triggers > PRE-INSERT
- 2 **Right-click** and select **PL/SQL Editor**.
- 3 To the beginning of the trigger, add the following code:
If MarkView.IsMarkView then
MarkView.Event('PRE-INSERT');
End if;
- 4 Click **Compile** and **Close**.

Step 8: Edit the WHEN-NEW-RECORD-INSTANCE Trigger

- 1 In the Object Navigator, select:
FNDAATCH > DataBlocks > DOCUMENT_HEADER > Triggers >
WHEN-NEW-RECORD-INSTANCE
- 2 Right-click and select **PL/SQL Editor**.
- 3 In the PL/SQL Editor, add the following code to the end of the trigger:
MarkView.Default_Show_Document;
- 4 Click **Compile** and **Close**.

Step 9: Create the KEY-NEXT-ITEM Trigger

- 1 In the Object Navigator, select:
FNDATTCH > Data Blocks > DOCUMENT_HEADER > Items > DATATYPE_NAME > Triggers
- 2 Click the plus sign (+) in the toolbar to create a new trigger. A list of trigger names appears.
- 3 From the list, select **KEY-NEXT-ITEM**.
- 4 Add the following code to the new KEY-NEXT-ITEM trigger:
MarkView.Event('KEY-NEXT-ITEM');
- 5 Click **Compile** and **Close**.

Step 10: Edit the WHEN-VALIDATE-ITEM Trigger for the DATATYPE_NAME Item

- 1 In the Object Navigator, select:
FNDATTCH > DataBlock > DOCUMENT_HEADER > Items > DATATYPE_NAME > Triggers > WHEN-VALIDATE-ITEM
- 2 Right-click and select **PL/SQL Editor**. A PL/SQL pane appears, containing the code for this trigger.
- 3 Append the following code to the end of the trigger:
MarkView.Event('WHEN-VALIDATE-ITEM');
- 4 Click **Compile** and **Close**.

Step 11: Edit the WHEN-VALIDATE-ITEM Trigger for the CATEGORY_DESCRIPTION Item

- 1 In the Object Navigator, select:
FNDATTCH > DataBlock > DOCUMENT_HEADER > Items > CATEGORY_DESCRIPTION > Triggers > WHEN-VALIDATE-ITEM
- 2 Right-click and select **PL/SQL Editor**. A PL/SQL pane appears containing the code for this trigger.
- 3 Append the following code to the end of the trigger:
MarkView.Event('WHEN-VALIDATE-ITEM');
- 4 Click **Compile**. Your changes are saved automatically.

Step 12: Add an Alert to the FNDATTCH Form

- 1 In the Object Navigator, select **ATTCH170 > Alerts > MVOA_MESSAGE**.
- 2 Right-click and select **Copy**.
- 3 In the Object Navigator, select **FNDATTCH > Alerts**.
- 4 Right-click and select **Paste**.

Step 13: Edit the WHEN-BUTTON-PRESSED Trigger

- 1 In the Object Navigator:
 - a Select:
FNDATTCH > Datablocks > DOCUMENT_CONTROL > Items >
OPEN_OTHER_DOCUMENT > Triggers > WHEN-BUTTON-PRESSED
 - b Right-click and select **PL/SQL Editor**.
- 2 Replace any existing code with the following code:

```
if MarkView.IsMarkView then
copy(:document_header.file_name, 'GLOBAL.ATCHMT_FILE_NAME');
APPCORE_CUSTOM.EVENT('OPEN-DOCUMENT');
else
ATCHMT_OPEN_OTHER_DOCUMENT.open_document('document_header');
end if;
```
- 3 Click **Compile** and **Close**.

Step 14: Save Your Changes

- 1 In Object Navigator, save FNDATTCH.fmb by selecting **File > Save**.
- 2 Close Oracle Forms Builder.
- 3 Using Oracle Forms generator, issue the following command to compile the form into an .fmx file, substituting the appropriate userid information.

```
frmcmp userid=apps/apps-pw@connectstring module=$c_MARKVIEW_TOP/forms/US/  
FNDATTCH.fmb module_type=FORM compile_all=YES
```

The compiled form is now at the following pathname:

```
$c_MARKVIEW_TOP/forms/US/FNDATTCH.fmx
```

Note: Kofax encountered problems when compiling forms in some versions of Forms Builder. For the best results, compile from the command line as a safeguard.

- 4 In the host operating system, copy **FNDATTCH.fmx** to the following directory:
\$FND_TOP/forms/US

For example, from the UNIX shell, you would issue the following command:

```
$ cp $c_MARKVIEW_TOP/forms/US/FNDATTCH.fmx $FND_TOP/forms/US
```

Note: You do not need to copy **FNDATTCH.fmb** to **\$FND_TOP/forms/US**; at run-time, the system uses the **.fmx** file, not the **.fmb** file.

Updating the Custom Library

The *custom library* (CUSTOM.pll) is an Oracle Forms library that ships with Oracle Applications. The custom library is attached to all forms in the E-Business Suite. As shipped by Oracle, this library performs no functions, but does provide callouts for custom logic. You must modify this library to integrate it with your MarkView software.

If you customized the custom library (CUSTOM.pll) that ships with Oracle applications, restore the version of CUSTOM.pll that contains your modifications before adding the MarkView Integration code. If you made no customizations, skip this section.

Step 1: Locate and Copy the Modified CUSTOM.pll File (Customized CUSTOM.pll Only)

If you customized the CUSTOM.pll, restore the version of the file that contains your changes. If you did not customize the library, skip this step and go to [Step 2: Open CUSTOM.pll for Editing](#).

The MarkView installer created restore directories at the following path:

```
$c_MARKVIEW_TOP/MVOA/version-number/Restore_timestamp
```

where *version-number* is the version number of the MarkView software, and *timestamp* is the date and time when the MarkView installer was run.

Copy the CUSTOM.pll file from the restore directory to the following directory:

```
$c_MARKVIEW_TOP/MVOA/version-number/libraries
```

Note: If you find multiple restore directories, use the oldest, because this contains the original files. The restore directory contains both CUSTOM.pll and CUSTOM.plx.

Step 2: Open CUSTOM.pll for Editing

- 1 Open Oracle Forms Builder and connect to the Oracle Apps database schema.
- 2 Select **File > Connect**.
- 3 Enter the Oracle Apps database user, password, and connect string (the same credentials specified during MarkView for Oracle Applications installation).
- 4 Open the copy of CUSTOM.pll that is in the following directory:

```
$c_MARKVIEW_TOP/MVOA/version-number/libraries/
```

Step 3: Attach Libraries (Customized CUSTOM.pll Only)

If you customized the CUSTOM.pll, manually attach the following MarkView libraries to CUSTOM.pll. If you did not customize the library, skip this step and proceed to [Step 4: Edit the zoom_available, style, and event Procedures on page 25](#).

Attach the following libraries to the CUSTOM.pll file:

- MVOAUTIL.pll
- MVVIEWER.pll
- MVFOLDER.pll
- SFAPI.pll

To attach the libraries manually:

- 1 In the Object Navigator, navigate to the Attached Libraries section and click the **green +** button (Create).
- 2 In the dialog box that opens, click **Find**.
- 3 Navigate to the **\$AU_TOP/resources** directory and select **MVOAUTIL.pll**:
- 4 Repeat steps 1-3 for each of the other libraries, selecting **MVVIEWER.pll**, **MVFOLDER.pll**, and **SFAPI.pll** in step 3.

The attached libraries appear in the Attached Libraries section.

Step 4: Edit the zoom_available, style, and event Procedures

The following steps change the procedures to call the MVOAUTIL library.

- 1 Open the package body of the CUSTOM program unit. The PL/SQL editor appears.
- 2 In the zoom_available procedure, add the following code at the end:

```
return(MVOA_Util.Zoom_Available);
```
- 3 In the style procedure, add the following code at the end:

```
return(MVOA_Util.Zoom_Style);
```
- 4 In the event procedure, add the following code at the end:

```
MVOA_Util.Event(event_name);

if ( (GET_APPLICATION_PROPERTY(Current_Form_Name) ='APXXEER'
or GET_APPLICATION_PROPERTY(Current_Form_Name) ='APXINWKB')
and event_name = 'ZOOM') then
execute_trigger('WHEN-WINDOW-ACTIVATED');
end if;
```
- 5 Save the changes to CUSTOM.pll.

Step 5: Compile and Copy CUSTOM.pll

After editing CUSTOM.pll, compile and convert your compiled copies of CUSTOM.pll and CUSTOM.plx as described next. Copying the files to an active directory activates the files.

- 1 Using Oracle Forms generator, issue the following command to compile the library into a .plx, substituting the appropriate userid information:

```
frmcmp userid=apps/apps-pw@connectstring module=$c_MARKVIEW_TOP/MVOA/
version-number/libraries/CUSTOM.pll module_type=LIBRARY compile_all=YES
```
- 2 Copy

```
$c_MARKVIEW_TOP/MVOA/version-number/libraries/CUSTOM.pll
```

to

```
$AU_TOP/resource
```
- 3 Copy

```
$c_MARKVIEW_TOP/MVOA/version-number/libraries/CUSTOM.plx
```

to

```
$AU_TOP/resource
```

Configuring Oracle Forms for Multi-Language Support (Optional)

To configure Oracle Forms to support other languages in addition to US English, edit language-specific versions of the forms to include MarkView integration code.

- 1 Locate the appropriate FNDATTCH.fmb file. For example, the US English version of this file is in the following location:
`$AU_TOP/forms/US/FNDATTCH.fmb`
while the French version of the file is in the following location:
`$AU_TOP/forms/FR/FNDATTCH.fmb`
- 2 Create a new language-specific subdirectory under `$c_MARKVIEW_TOP/forms`. For example, if you are adding support for French, your subdirectory should be:
`$c_MARKVIEW_TOP/forms/FR`
Make sure the .fmx file is in the correct language-specific subdirectory.
- 3 Modify the file to include the MarkView integration code, as described in [Updating the Oracle Attachments Form \(FNDATTCH.fmb\) on page 18](#).
- 4 Copy the APXINWKB.fmb file from `$AU_TOP/forms/language-specific-subdirectory` to the subdirectory that you created in Step 2.
- 5 Rename the APXINWKB.fmb file to SFXINWKB.fmb.
- 6 Edit the SFXINWKB.fmb file to include the MarkView integration changes.
- 7 Use Oracle Forms Generator to compile SFXINWKB.fmb into SFXINWKB.fmx.

Repeat these steps for each language to support.

Enabling Invoice Workbench

To use Oracle forms with the MarkView Process GetNext or Open Work Item operations, modify copies of the original Oracle forms as described next.

To enable Invoice Workbench functionality for MarkView Process workflow processing, you create a modified version of the Invoice Workbench form (APXINWKB to SFXINWKB), which will be enabled for both GetNext and Open Work Item.

Note: The following instructions assume that the MarkView base path is the default
`$c_MARKVIEW_TOP`

Image-Enable the Default Invoice Workbench Form

Two versions of Invoice Workbench exist:

- **SFXINWKB.fmb**—the integrated version. Use this version to enable GetNext and Open Work Item workflow processing from the Invoice Workbench. If using this version, go to [Modify SFXINWKB for GetNext and Open Work Item](#).
- **APXINWKB.fmb**—the non-integrated version. Use this version if you do not need GetNext or Open Work Item workflow processing from the Invoice Workbench. If using this version, image-enable the default invoice workbench form as follows:

Run the **APXINWKB.sql** script, which ships with MarkView for Oracle Applications database components. The script is located under the installation directory in the directory `mvoadbobj/seed_data/oracle_forms/`.

Modify SFXINWKB for GetNext and Open Work Item

Complete this procedure if you are using the integrated version of the Invoice Workbench (SFXINWKB):

- 1 From `$AU_TOP/forms/US/`, open Oracle Forms Builder. See [Using the Oracle Forms Builder on page 16](#) for details.
- 2 From `$c_MARKVIEW_TOP/forms/US`, open the form **SFXINWKB.fmb**.
- 3 Select **File > Connect** to connect to the Oracle APPS database schema. Enter the Oracle APPS database user, password, and connect string (the same credentials specified during MarkView for Oracle Applications installation).
Note: The displayed name in Forms Builder Object Navigator is APXINWKB. Do not change this name to SFXINWKB. Only the FMB and FMX file names change.
- 4 Choose **Data Blocks > INV_SUM_FOLDER > Items** and click the + sign in the tool bar to add a new item.
 - a Rename the new item **IMG_IMAGE_ID**.
 - b Right-click and select **Property Palette**.
 - c In Item Type enter 'Text Item'.
 - d Select **Property Class**.
 - e In Subclass Information, select **TEXT_ITEM** from the **Property Class** drop-down menu.
 - f Navigate to **Data** and set Maximum Length to 240.
 - g Navigate to **Database** and set Database Item to No.

- 5 Choose **Program Units > INV_SUM_FOLDER_BLOCK_INSERT1(Package Body)**.
 - a Right-click and select **PL/SQL Editor**.
 - b After the call to AP_INVOICES_PKG.Insert_Row, add the following code immediately before the END Insert_Row line:


```
IF :INV_SUM_FOLDER.IMG_IMAGE_ID IS NOT NULL THEN
    APPCORE_CUSTOM.EVENT(' INVOICE-IMAGE-ASSOCIATED ');
END IF;
```
 - c Click **Compile** to save changes.
- 6 Select **DataBlocks > INV_SUM_FOLDER > Triggers > KEY-DUPREC**.
 - a Right-click and select **PL/SQL Editor**.
 - b Replace the existing text with the following:


```
-- inv_sum_folder.event('KEY-DUPREC');
Declare
    cur_image_id Varchar2(240);
Begin
    cur_image_id := :inv_sum_folder.img_image_id;
    inv_sum_folder.event('KEY-DUPREC');
    :inv_sum_folder.img_image_id := cur_image_id;
End;
```
 - c Click **Compile** and **Close**.
- 7 Select **Triggers** and click the + sign in the toolbar and then select WHEN-WINDOW-ACTIVATED in the dialog box.
 - a Right-click and select **PL/SQL Editor**.
 - b Add the following text:


```
APPCORE_CUSTOM.EVENT('MVOA-WHEN-WINDOW-ACTIVATED');
```
 - c Click **Compile**.
- 8 Select **Program Units > APXINWKB (Package_Body)**.
 - a Right-click and select **PL/SQL Editor**.
 - b Modify the version number to indicate that it was modified for MarkView integration by appending .170 to the version number. For example:


```
FND_STANDARD.FORM_INFO('$Revision: 115.564.11591.4')
to:
FND_STANDARD.FORM_INFO('$Revision: 115.564.11591.4.170')
```
 - c Click **Compile**.

- 9 Select **File > Save** to save the .fmb version.
- 10 Exit Form Builder.
- 11 Using Oracle Forms generator, issue the following command to compile the form into a .fmx.

```
frmcmp userid=apps/apps-pw@connectstring module=$c_MARKVIEW_TOP/forms/US/SFXINWKB.fmb module_type=FORM compile_all=YES
```

Substituting the appropriate userid information. The compiled form is now at the following pathname:

```
$c_MARKVIEW_TOP/forms/US/SFXINWKB.fmx
```

Note: Kofax encountered issues with forms compiled in some versions of Forms Builder. For the best results, compile from the command line as a safeguard against these issues.

Reintegrating the Quick Invoices Form

If your company also uses MarkView Connector with the Quick Invoices Form, reintegrate Quick Invoices as described next.

- 1 Open `$c_MARKVIEW_TOP/forms/US/SFXIISIM.fmb` in Oracle Forms Designer.
frmblld SFXIISIM.fmb
- 2 Connect to the Oracle APPS database schema and do the following:
 - a Select **File > Connect**.
 - b Enter the Oracle APPS database user, password, and connect string (the same credentials specified during MarkView for Oracle Applications installation).
- 3 Navigate to **Oracle Forms Designer > Property Palette**, and change the name of the form from `APXIIISIM.fmb` to `SFXIISIM.fmb`.
- 4 Add the item `IMG_IMAGE_ID` to the data block `INVOICES_FOLDER`.
 - a In the Property Palette, open **IMG_IMAGE_ID**.
 - b Select **ITEMS** then click the **+** sign on the tool bar.
 - c Set Item Type to **text_item**.
 - d Set Maximum length to **240**.
 - e Set Database Item to **No**.
 - f Close the window.

- 5 Add the following code to the ON-INSERT trigger on the INVOICES_FOLDER data block. Insert the code after the call to INVOICES_FOLDER.INSERT_ROW.

```
IF:INVOICES_FOLDER.IMG_IMAGE_ID IS NOT NULL THEN
  APPCORE_CUSTOM.EVENT('INTERFACE-INVOICE-IMAGE-ASSOCIATED');
END IF;]
```
- 6 Change the trigger KEY-DUPREC on the INVOICES_FOLDER block to the following:

```
--invoices_folder.key_duprec;
Declare
  cur_image_idVarchar2(240);
Begin
  cur_image_id:=invoices_folder.img_image_id;
  invoices_folder.key_duprec;
  :invoices_folder.img_image_id:=cur_image_id;
End;
```
- 7 Add the trigger WHEN-WINDOW-ACTIVATED at the form level with the code:

```
default_value(null, 'global.mvoa_open_opening_work_item');
if :GLOBAL.MVOA_OPEN_OPENING_WORK_ITEM = 'Y' then
  clear_block(no_validate);
end if;
APPCORE_CUSTOM.EVENT('MVOA-WHEN-WINDOW-ACTIVATED');
```
- 8 Update the form trigger WHEN-NEW-FORM-INSTANCE.
 - a Comment out the following line:

```
EXECUTE_TRIGGER('QUERY_FIND');
```
 - b Add the following line:

```
set_window_property('INVOICES_FOLDER_WINDOW', title, 'MarkView
Quick Invoices');
```
- 9 Navigate to **Data Blocks > QF_INVOICES > Items > source_DSP**, open the **Properties Palette**, and in the Data section, set Required to **No**.
- 10 Navigate to **Data Blocks > QF_INVOICES > Items > group_ID**, open the **Properties Palette**, and in the Data section, set Required to **No**.
- 11 Right-click the **INVOICES_FOLDER** Data Block and use the Properties Palette to set the Query Data Source Name to **MVCN_AP_INVOICES_INTERFACE_V**.
- 12 In **Canvases**, select the **Layout Editor** to open **INVOICES_CONTENT**.
 - a In the lower right corner, right-click **Save and Next** to open the **Properties Palette**.
 - b In the Functional section, set **Enabled** to **No**.

- c** In the Physical section, set **Visible** to **No**.
- d** Close the **Properties Palette** and **INVOICES_CONTENT**.
- 13** To save the .fmb version of the form, select **File > Save**.
- 14** Exit **Forms Designer**.
- 15** Using the Oracle Forms generator, issue the following command to compile the form into a .fmx.

Caution: Run the following command on a single command line. If you cut and paste from this document, remove line breaks.

```
frmcmp userid=apps/apps-pw@connectstring module=$c_MARKVIEW_TOP/forms/US/  
SFXIISIM.fmb module_type=FORM compile_all=YES
```


Configuring MarkView for Expense Management

Configuring Oracle Expenses Workflow

To configure MarkView for Expense Management, you must have Oracle Workflow Builder installed and your system must include data.

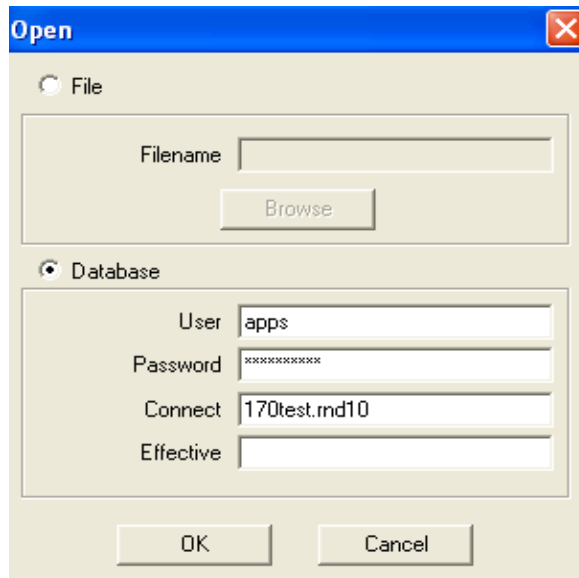
The screen shots in this chapter may differ from those on your Oracle interface. Screen appearance depends on the version of Oracle Internet Expenses to which you are integrating. Regardless of the screen appearance, the integration steps are identical.

Setting Administrator Access Level

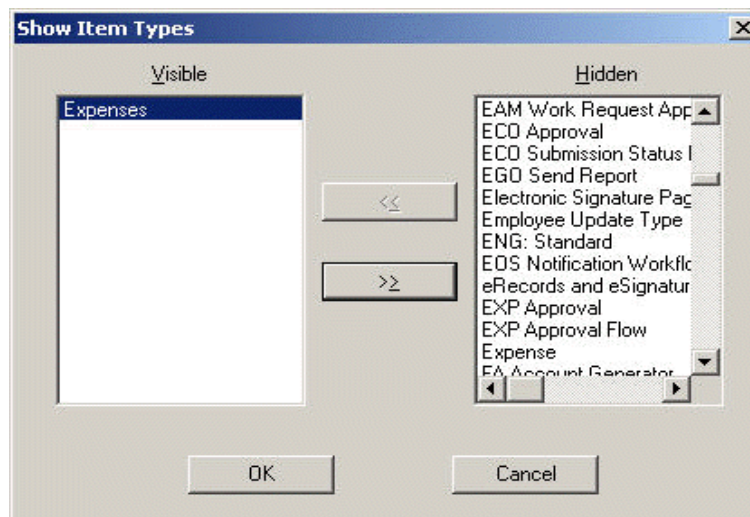
- 1 Open Oracle Workflow Builder and select **Help > About Oracle Workflow Builder**.
- 2 In the About Oracle Workflow Builder window, set the following values:
 - Set Access Level to **0**.
 - Enable **Allow modification of customized objects**.
- 3 Click **OK**.

Opening Expenses Workflow

- 1 From the Oracle Workflow Builder, select **File > Open**.



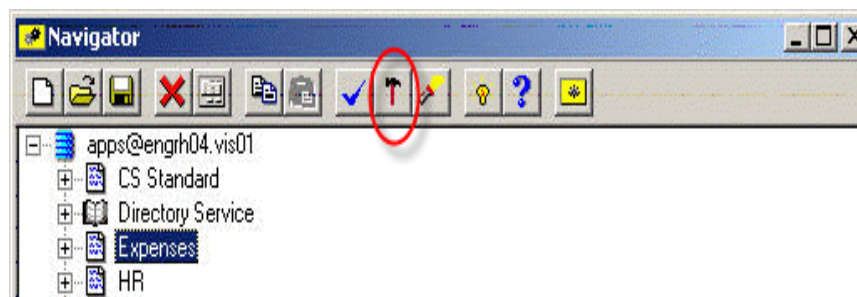
- 2 In the Open dialog box, complete the following fields:
 - Database Enable
 - User Enter the database *username*
 - Password Enter the *password* associated with APPS database username
 - Connect Enter the *connect string* information
 - Effective Leave blank
- 3 Click **OK**.



- 4 In the Show Item Types dialog box, move **Expenses** from the Hidden column to the Visible column, and click **OK**.

The expenses item type appears in a tree structure.

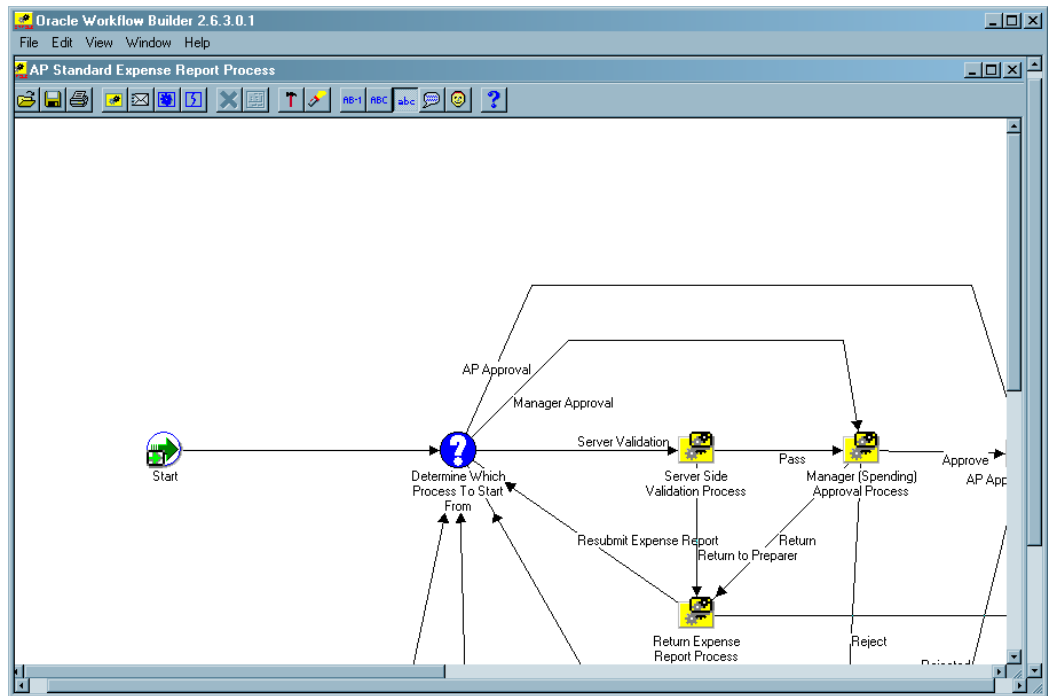
- 5 Expand the tree to display all the options under the Expenses menu tree. The Expenses workflow is now visible.
- 6 Ensure that Developer mode is **off** in the Oracle Workflow Builder. The hammer icon in the button bar should be deselected.



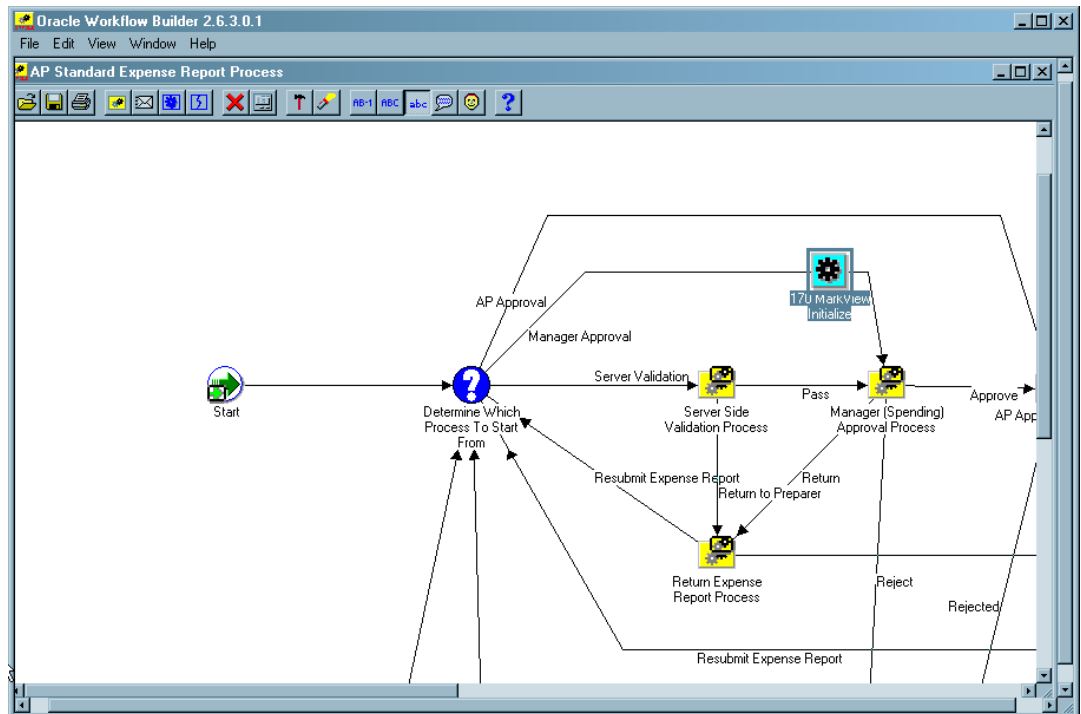
Adding Functions to the Workflow Process

- 1 Open Oracle Workflow Builder.
- 2 Log in with the APPS database *username* and *password*.
- 3 In the Expenses item type, expand the **Processes** section.
- 4 Double-click **AP Standard Expense Report Process**.

The main workflow opens in an edit window with the starting process diagram displayed.



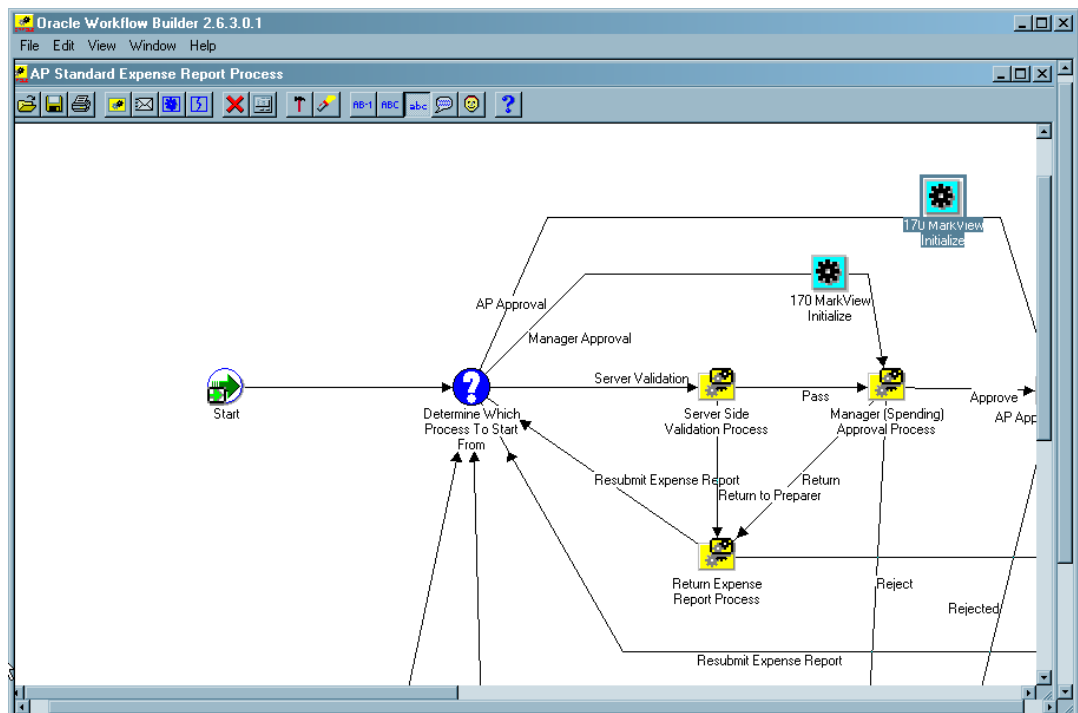
- From the Navigator window, drag the **MarkView Process Initialize** function onto the Manager Approval transition.



- 6 From the Navigator window, select and drag the **MarkView Process Initialize** function onto the AP Approval transition.

The process diagram contains two instances of MarkView Process Initialize.

Note: The instances of MarkView Process Initialize may be labeled SQLFLOW_INIT or 170 MarkView Process Initialize.

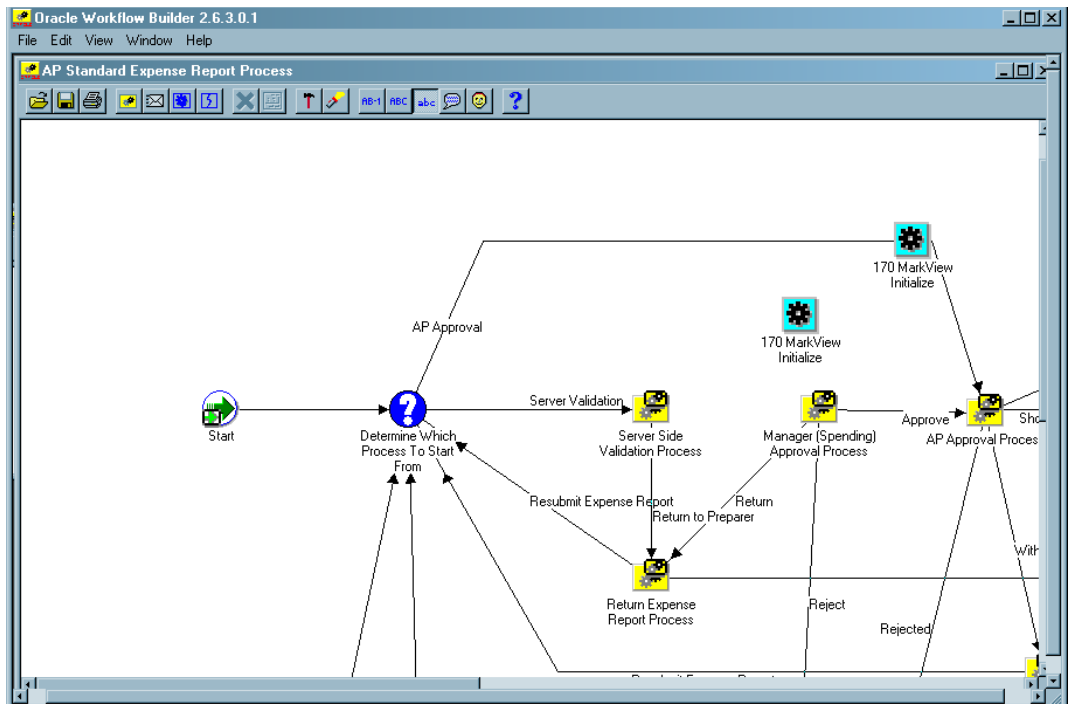


- 7 From between Determine Which Process To Start From and Manager (Spending) Approval Process, delete the **Manager Approval** transition.

Right-click the arrow and select **Delete Selection**.

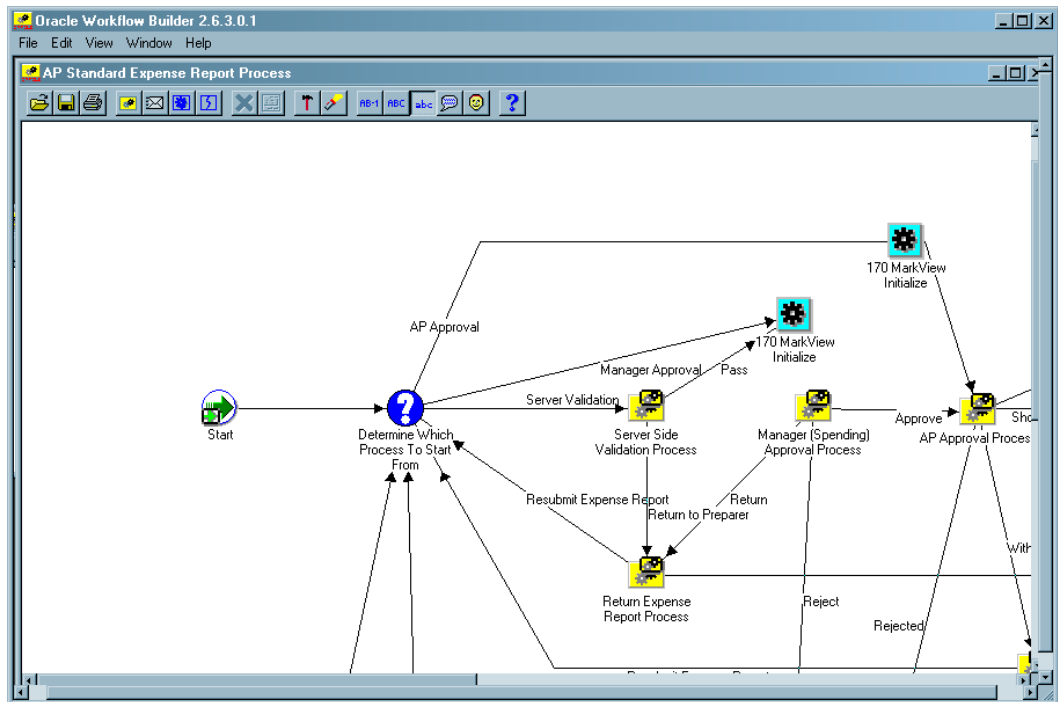
- 8 From between Server Side Validation Process and Manager (Spending) Approval Process, delete the **Pass** transition line.

Right-click the arrow and select **Delete Selection**.

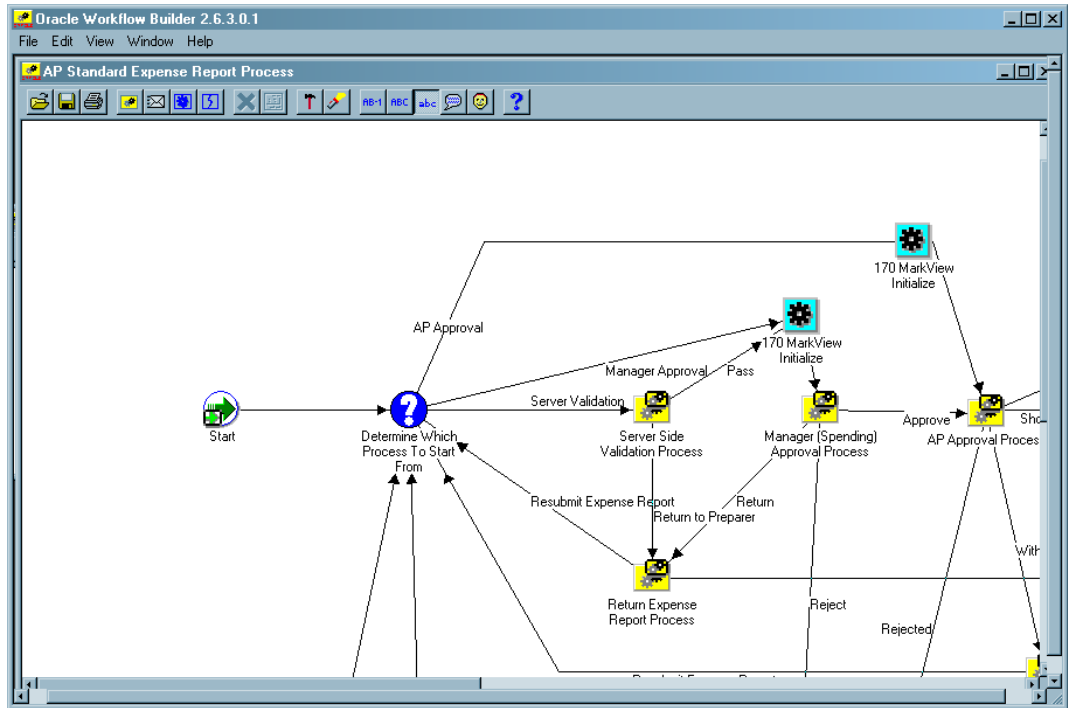


- 9 Create a transition between the Determine Which Process To Start From and MarkView Process Initialize:
 - a Right-click and hold **Determine Which Process To Start From**.
 - b Drag the cursor over **MarkView Process Initialize**.
 - c Release the mouse button.
 - d Select **Manager Approval**.

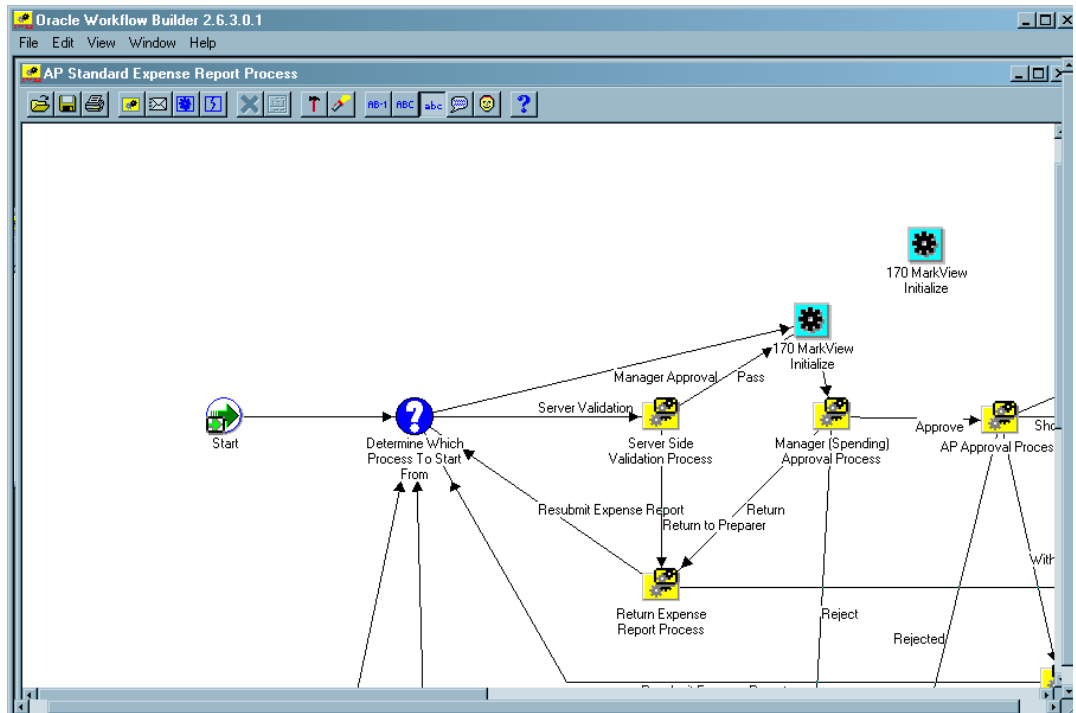
- 10 Create a transition between Server Side Validation Process and MarkView Process Initialize:
 - a Right-click and hold **Server Side Validation Process**.
 - b Drag the cursor over **MarkView Process Initialize**.
 - c Release the mouse button.
 - d Select **Pass**.



- 11 Create a transition between MarkView Process Initialize and Manager Approval Process:
 - a Right-click and hold **MarkView Process Initialize**.
 - b Drag the cursor over **Manager Approval Process**.
 - c Release the mouse button.



- 12 From between Determine Which Process To Start From and AP Approval Process, delete the **AP Approval transition line**:
 - a Position the cursor on the arrow that connects them.
 - b Right-click and select **Delete Selection**.



- 13 Create a transition between Determine Which Process to Start From and the second MarkView Process Initialize function:
 - a Right-click and hold **Determine Which Process to Start From**.
 - b Drag the cursor over **MarkView Process Initialize**.
 - c Release the mouse button.
 - d Select **AP Approval**.
- 14 Create a transition between MarkView Process Initialize and AP Approval Process:
 - a Right-click and hold **MarkView Process Initialize**.
 - b Drag the cursor over **AP Approval**.
 - c Release the mouse button.
- 15 Click **Save**. Saving your changes can generate multiple validation failed for activity errors such as:

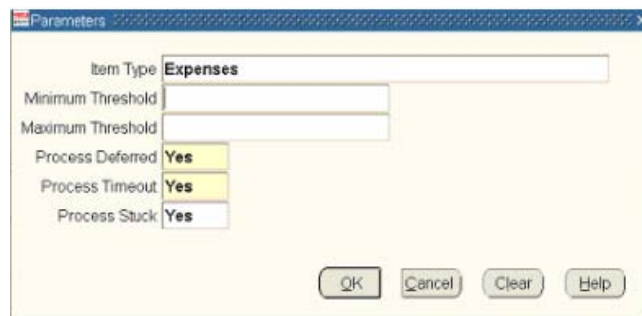

```
'SAVE validation failed for activity 'HRSSA/HR_SAVE_FOR_LATER'.
```

 Ignore errors that do not relate to your changes.

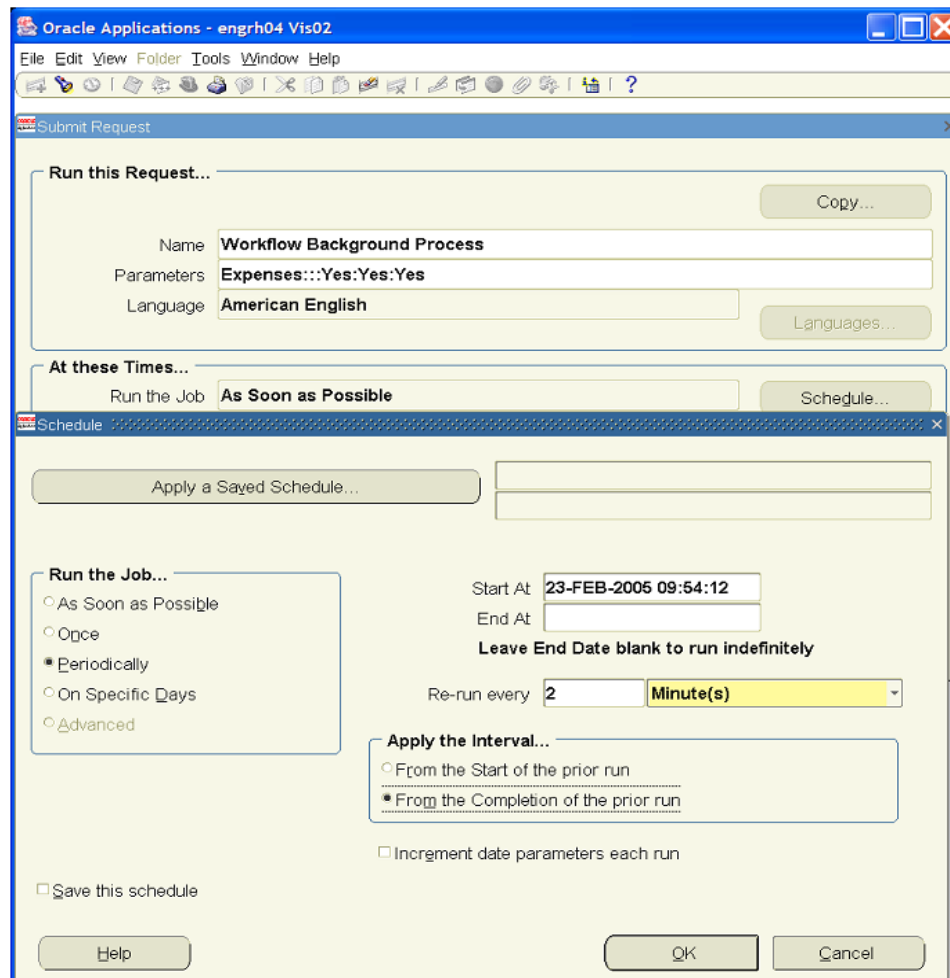
Configuring the Workflow Background Process

Schedule the Workflow Background Process concurrent request to run periodically in the Oracle Concurrent Manager. This concurrent request handles items in the Oracle workflow that are timed out, stuck, or notified. The following steps describe the settings to use with your MarkView system. For information about setting up this process, see Oracle MetaLink note 182936.1, the *Oracle Workflow Developer's Guide*, and the *Oracle Applications User's Guide*.

- 1 Log into Oracle Concurrent Manager as a system administrator.
- 2 Follow the instructions in MetaLink note 182936.1 and the *Oracle Applications User's Guide* for submitting the Workflow Background Process as a concurrent program.
- 3 Complete the Parameters dialog box as follows:
 - Item Type: **Expenses**
 - Minimum Threshold: Leave blank
 - Maximum Threshold: Leave blank
 - Process Deferred: **Yes**
 - Process Timeout: **Yes**
 - Process Stuck: **Yes**



- 4 Click **OK**.
- 5 In the Schedule dialog box, set the job to run at 2-minute intervals.



Updating the Fax Number

The MarkView preference `MVEXP_EXPENSES_FAX_NUMBER` includes the value for the fax number that employees use to fax expense report packets.

In MarkView Administration, set the new value for the preference `MVEXP_EXPENSES_FAX_NUMBER`. The format of the value does not matter; the fax number is for display purposes only.

For more information about creating submission instructions, see the MarkView administration guide.

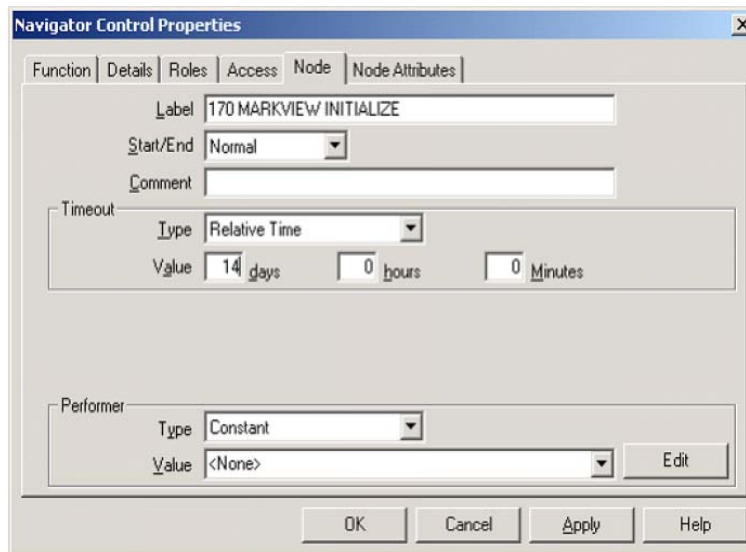
Configuring the Missing-Receipt Timeout

When the MarkView Initialize function starts, the system creates the MarkView work item and waits for faxed receipts.

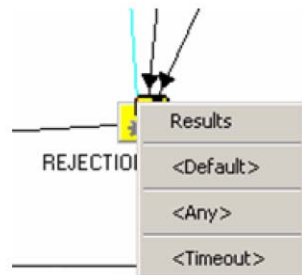
If employees take too long to submit receipts, you can automatically move their expense reports into the Oracle Applications rejection process by configuring the system to reject expense reports after a specified number of days.

To configure the missing-receipt timeout:

- 1 Open the Oracle Workflow Builder.
- 2 Connect as apps/<appspassword>.
- 3 Open the Expenses (APEXP) Item Type.
- 4 Expand the Processes list, and double-click the **AP Standard Expense Report** process. The process diagram opens.
- 5 Right-click the **MarkView Initialize** function and select **Properties**.
- 6 In the Navigator Control Properties dialog box, select the **Node** tab.
- 7 Under Timeout, change the Type to **Relative Time**.
- 8 In the Value fields, enter the total amount of time after which an expense report times out. Use whole numbers.



- 9 Right-click the MarkView Initialize function and draw a line to the REJECTION_PROCESS function. A menu opens.



- 10 Select **<Timeout>** and save your changes.
- 11 Repeat this procedure for all other functions that have names similar to MarkView Initialize, such as MarkView Initialize-1.

For the timeout subprocess to work, schedule the Workflow Background Process job to execute on a regular basis.

