Skip Headers

Oracle Workflow Developer's Guide
Release 12.2
Part Number E22011-18

Contents Previous Next

Configuring Messages for Oracle Mobile Approvals for Oracle E-Business Suite

This chapter explains how to configure message metadata for approval notifications that you want to expose in the Oracle Mobile Approvals for Oracle E-Business Suite app.

This chapter covers the following topics:

Configuring Messages for Oracle Mobile Approvals for Oracle E-Business Suite

Oracle Mobile Approvals for Oracle E-Business Suite (Approvals) is a smartphone app that lets users respond on the go to pending approval notifications sent by workflow processes. This app serves as an alternative interface through which users can respond to approval notifications, in addition to the Worklist web pages and e-mail notifications. By providing access to approval notifications on mobile devices, the app enables users to take quick action on these notifications to help complete business processes that may be time-sensitive.

The Approvals app displays approval notifications from approval types seeded by Oracle E-Business Suite. If you want approval notifications from custom workflows to appear in the app as well, you can optionally add custom approval types. You can also optionally modify the configuration metadata for seeded approval notifications that are defined to allow customizations.

The Approvals app is intended to provide a streamlined interface, so it does not automatically show all the content that would appear for notifications in the Worklist pages. You can configure the content that will appear in the app based on what information is most important to help approvers make a decision.

The Approvals app leverages the Approvals Data Services Framework to retrieve data for approval notifications in a standard format and display those notifications to approvers using a consistent look and feel in the mobile app user interface. The Approvals Data Services Framework is a component of Oracle Workflow that draws on the Oracle Workflow Notification System infrastructure as well as on specifically configured metadata to make approval notifications available to the Approvals app. The framework provides a configuration tool that lets you define the metadata required to retrieve approval notification data from the underlying application tables. It also provides the REST services that use that metadata to communicate the approval notification data to the Approvals app. By using the approvals data services configuration tool, you can expose approval notifications from your custom workflows in the Approvals app without needing to create custom Web services.

Note: Approvals data services are sometimes also known as worklist data services.

The configuration tool provides an easy-to-use interface to let you do the following:

- Configure approval message metadata, drawing on existing workflow definitions as much as possible.
- Preview the message configuration before saving it.
- Verify the message configuration by viewing the data returned for an existing notification ID.
- Update an existing message configuration.
- Test how the metadata you defined works with the approvals data REST services.

Additionally, you can use the Generic Loader (FNDLOAD) concurrent program to download and upload approvals data services metadata so that you can transfer the metadata between Oracle E-Business Suite instances or store a copy for source control purposes.

The approvals data services consume the metadata created by the configuration tool and produce notification content in a canonical XML format. The approvals data services offer the following features:

- All the approvals data services are secured using an Oracle E-Business Suite session cookie associated with a valid Oracle E-Business Suite user session. This method ensures that the user can only see notifications to which he or she has access.
- The services accept XML requests and provide XML responses with notification content in a canonical XML format. This XML format remains the same regardless of the business object being approved.
- The services use the metadata from the configuration tool to build the appropriate notification content XML response for a given notification ID.
- The response is focused on providing the approval notification content, which is structured as attributes in simple name - value pairs. The approvals data services do not address the presentation or formatting of the content; instead, the presentation is handled entirely by the client Approvals app.
- Oracle Workflow provides and exposes the approvals data services as well as the Approvals app that uses those services to show the approval notification content. You do not need to build or expose any custom Web services or build a custom app in order to make your custom approval notifications available for display; you only need to configure the appropriate metadata for the approvals data services to consume.

Prerequisites

- To implement the Approvals Data Services Framework in your Oracle E-Business Suite installation, apply the current ATG_PF Consolidated Patch for Mobile Applications Foundation as shown in *Oracle E-Business Suite Mobile Foundation Release Notes*, My Oracle Support Knowledge Document 1642431.1.
- You must perform the server-side implementation steps for the Approvals app for your users to have access to approval notifications through the app. See: *Oracle Mobile Approvals for Oracle E-Business Suite Release Notes*, My Oracle Support Knowledge Document 1642423.1.

Guidelines for Configuring Messages as Approval Notifications

The Approvals app is intended to let approvers access their important approval notifications in a streamlined interface while on the go. Follow these guidelines when selecting approval messages to appear in the app:

- Select approval messages that are critical for your business flows. It is not recommended to configure every workflow message as an approval notification in the Approvals app.
- The Approvals app is designed only for notifications that require a response. Do not configure FYI notifications for the app.
- Due to the limited display area on a smartphone, you should configure only essential message content to appear in the app to help approvers make a decision. It is not recommended to include all the content from the full workflow message as it appears in the Notification Details page.
- The app supports only two possible response actions: Approve or Reject. You can use different labels for these actions, and you can configure messages for the app that have additional possible results as part of their workflow definitions, but you must select only two results to map to the Approve and Reject actions within the app. To provide other responses, users must access messages through the Worklist web pages or through e-mail.

  Note: The app provides an option for approvers to request more information before taking action. However, the approval notification is not completed until the approver chooses either the Approve or the Reject response action.

  An approver can also choose to reassign a notification.

To configure messages to appear as approval notifications in the Approvals app, follow this overall process:

1. Identify the approval notifications and the content for those notifications that you want to display in the Approvals app.
2. Use the approvals data services configuration tool to configure the message metadata.
3. Use the approvals data services approval type tool to define approval types.
4. Use the approvals data services test client to test how the metadata you defined works with the approvals data REST services.
5. Use the Generic Loader (FNDLOAD) concurrent program to download and upload approvals data services metadata.

Identifying Approval Notification Content

To display custom approval notifications in the Approvals app, you must first review your workflow definitions to identify the approval notifications that you want to include and the content that will appear for those notifications in the app. For notifications that include an embedded Oracle Application Framework region, you must also review the BC4J object implementation that retrieves notification content, using Oracle JDeveloper with Oracle Application Extension. For notifications that include PL/SQL document content, you must also review the PL/SQL procedures that retrieve notification content.

In some cases you might need to make changes in your workflow definitions, Oracle Application Framework view objects, or PL/SQL procedures to prepare the approval notification content to appear in the app. If so, you should take into consideration how such changes will affect the appearance of your approval notifications in the Worklist pages and in e-mail as well.

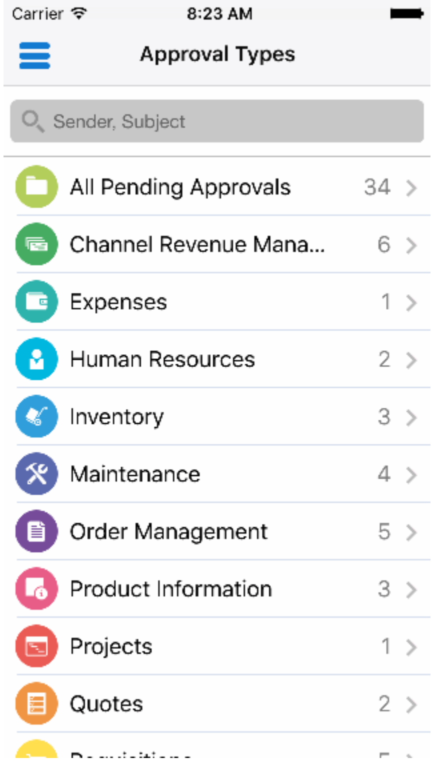
The content that appears in the Approvals app, in the order in which users see the content in the approval flow, is as follows:

- Approval types
- Approval notification information shown in the Pending Approvals list
- Approval notification details
  - Header
  - Body
    - Body details
    - Action history
    - Attachments
  - Approval response actions

Approval Types

An *approval type* is a logical grouping of approval notifications that belong to a business flow in an Oracle E-Business Suite application. For example, seeded approval types include Expenses, Purchase Orders, Requisitions, and so on.

Approval Types Screen



The name you assign to an approval type appears in the Approvals app on the Pending Approvals by Type screen, which approvers can access from the app's springboard. This screen lets approvers quickly filter the approval notifications they want to view. For example, an approver can select the Expenses type to view all approval notifications for expenses, or the Requisitions type to view all approval notifications for purchase requisitions.

Note: Approval types in the Approvals app are not the same as workflow item types. Unlike the Worklist pages, which show the workflow item type to which a notification belongs, the Approvals app uses approval types to provide a higher-level grouping of approval notifications for business transactions.

An approval type can include notifications for multiple related *approval objects*. For example, purchase requisitions and contract requisitions are two different approval objects whose approval notifications are included in the Requisitions approval type. Similarly, vacancies and offers are two different approval objects whose approval notifications are included in the Human Resources approval type.

Additionally, an approval object can have multiple approval notifications associated with it. For example, notifications can be sent to request approval for creating a vacancy and for updating a vacancy. Notifications from different workflow item types and message definitions can be included in one approval type to group together all the relevant approval notifications for related approval objects.

Follow these steps to determine what custom approval types you want to define. You can then use the approvals data services configuration tool to define the approval types you have planned.

1. Identify all the approval objects in your business flows. For example, an approval object could be a contract requisition, a purchase requisition, creating a vacancy, updating a vacancy, and so on.

2. Identify all the workflow notifications that request approvals associated with these objects. Check for notifications that may belong to different workflow item types but are associated with the same approval objects. Make a note of the workflow item type and message name that uniquely identify each message used to send these notifications.

3. Decide which related approval objects and associated notifications you want to group together into an approval type.

4. Select a name for the new approval type. Ensure that the name represents all the approval objects grouped within that type. To let approvers review and filter their approval notifications more easily, use a name related to the business flow. For example, an approval type can be named after a business object, such as Expenses or Purchase Orders, or after a functional area, such as Human Resources.

5. Select an icon to identify the approval type visually within the Approvals app. The icon should be in PNG format. It should be monochromatic white and should consist of an image with a size of 48 x 48 pixels, surrounded by padding so that the full icon size is 64 x 64 pixels. Copy the icon file to the $OA_MEDIA directory on your Oracle E-Business Suite application server.

   After copying the file, verify that you can view the icon image by accessing the following URL:

   ```
   http://<ebshost>:<port>/OA_MEDIA/icon_filename.png
   ```

   The Approvals app uses this URL to load the image from the server.

   Note: The mapping of $OA_MEDIA is completed as part of the Oracle E-Business Suite installation and setup steps. For more information, see the installation documentation for your release.

6. In some cases, a message definition is used by multiple notification activities in a workflow process. If all the notifications that use a particular message will be exposed in the Approvals app and belong to the same approval type, then you do not need to perform any further configuration. However, if you want to expose only some approval notifications sent by a workflow message in the Approvals app, or if the approval notifications sent by a workflow message should belong to different approval types, then you must perform additional steps to distinguish these notifications from each other.

   - If a message is used by multiple notification activities, define a special message attribute for that message with the internal name `#APPROVALS_GROUP`.

- For each notification activity that uses this message, include logic in your workflow process to set the value of the `#APPROVALS_GROUP` attribute to a value representing the approval type to which the notification belongs. If the notification should not be exposed in the Approvals app, leave the value of this attribute blank.

  For example, suppose the same standard message definition is used to send approval notifications for medical leaves of absence, vacations, hiring, and training, and only the first three notifications should be exposed in the app. In this case, the `#APPROVALS_GROUP` attribute could be set to `ABSENCE` for the notification activities that send the approval notifications for medical leaves of absence and for vacations, grouping these notifications into the same approval type. The `#APPROVALS_GROUP` attribute could be set to `HIRING` for the notification activities that send the approval notifications for hiring, indicating that these notifications belong to a different approval type. Finally, the attribute could be left blank for the notification activities that send the approval notifications for training, indicating that these notifications do not belong to any approval type and are not available in the app.

  When you define your approval types in the configuration tool, you will specify the value you set for the`#APPROVALS_GROUP` attribute as part of the approval type definition to indicate which notifications belong to the approval type.

  Additionally, if you will define an approvals group value for an approval type, then you must define the `#APPROVALS_GROUP` message attribute for all approval notifications that belong to this approval type and set the attribute value to the specified approvals group value. That is, if you use the `#APPROVALS_GROUP` message attribute for any approval notifications within an approval type, then you must define it for all other messages that belong to that type as well, even if all notifications sent by those messages belong to this same approval type.

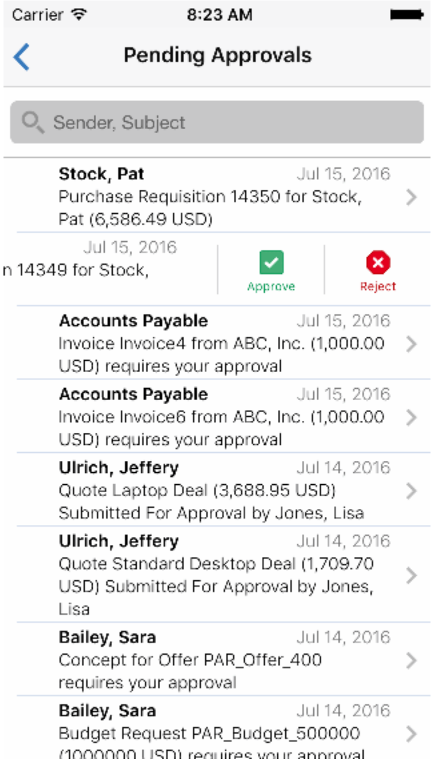
  Note: If you do not specify an approvals group value for the approval type, then the approval type will include all notifications sent by the messages associated with the type, regardless of whether those notifications have any value specified in an `#APPROVALS_GROUP` message attribute. That is, an approval notification can have its `#APPROVALS_GROUP` message attribute set to a value matching the approvals group value for one approval type, and the same approval notification can also belong to another approval type with no approvals group value specified.

Pending Approvals Information

The Approvals app provides the following screens to list approval notifications:

- Pending Approvals - A list of pending approval notifications for the current user from all approval types. The approver can access this screen directly from the springboard by selecting the Pending Approvals menu item.
- Pending Approvals by Type - A filtered list of pending approval notifications for the current user from only one approval type. The approver can access this screen by selecting the Pending Approvals by Type menu item from the springboard and selecting an approval type.
- Past Approvals - A list of notifications in the approver's queue that are closed. The approver can access this screen directly from the springboard by selecting the Past Approvals menu item.

Pending Approvals Screen



Stock, Pat Jul 15, 2016
Purchase Requisition 14350 for Stock, Pat (6,586.49 USD)

Jul 15, 2016
n 14349 for Stock,

Accounts Payable Jul 15, 2016
Invoice Invoice4 from ABC, Inc. (1,000.00 USD) requires your approval

Accounts Payable Jul 15, 2016
Invoice Invoice6 from ABC, Inc. (1,000.00 USD) requires your approval

Ulrich, Jeffery Jul 14, 2016
Quote Laptop Deal (3,688.95 USD) Submitted For Approval by Jones, Lisa

Ulrich, Jeffery Jul 14, 2016
Quote Standard Desktop Deal (1,709.70 USD) Submitted For Approval by Jones, Lisa

Bailey, Sara Jul 14, 2016
Concept for Offer PAR_Offer_400 requires your approval

Bailey, Sara Jul 14, 2016
Budget Request PAR_Budget_500000 (1000000 USD) requires your approval

These screens are analogous to the Worklist page in Oracle E-Business Suite. They display information about the approval notification information including the user from whom the notification was sent, the sent date, and the notification subject line. The approvals data services retrieve this information from the WF_NOTIFICATIONS table in the same way that it is retrieved for display in the Worklist.

The approval notification information is included automatically; you do not need to perform any steps for this information in the approvals data services configuration tool. However, you should follow these steps to prepare this information to be displayed in the Pending Approvals, Pending Approvals by Type, and Past Approvals screens.

1. Ensure that the subject line defined for each notification in the workflow definition clearly summarizes the key information that an approver needs in order to understand the notification. The recommended format is as follows:

   ```
   <Object Type> <Object ID> for <Requester> (<Main Value>)
   ```

For example:

```
Requisition 12345 for Walker, John (564.26 USD)
Expense 56789 for Smith, Karen (345.65 USD)
```

2. Ensure that the #FROM_ROLE message attribute is defined for each notification to identify the user from whom the notification is sent. See: #FROM_ROLE Attribute.

Note that if you update the message definitions for notifications that you intend to display in the Approvals app, those changes will appear in the Worklist pages as well.

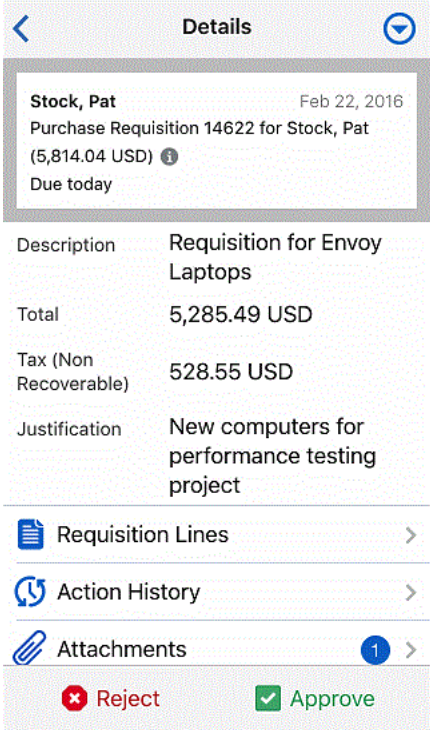
Approval Notification Details

The approval details screens display the content of an approval notification for the approver to review and make a decision. Approval details include the following:

- Header
- Body
  - Body details
  - Action history
  - Attachments
- Approval response actions

These screens are analogous to the Notification Details page in Oracle E-Business Suite. However, the Notification Details page displays all the notification content in a single page. By contrast, due to the limited display area on a smartphone, the Approvals app displays the notification content in multiple screens, progressing from higher-level information to lower-level details. The first-level details screen, which appears when the approver selects a notification from the Pending Approvals list or one of the other lists, shows the header attributes for the notification. From that screen, the approver can drill down to the second-level details screen to view the body content of a specific region associated with the approval object. For example, expense report headers are shown in the first-level details screen for Expenses approvals, while expense report lines are shown in the second-level details screen.

First-Level Details Screen



Details

Stock, Pat — Feb 22, 2016
Purchase Requisition 14622 for Stock, Pat (5,814.04 USD)
Due today

Description	Requisition for Envoy Laptops
Total	5,285.49 USD
Tax (Non Recoverable)	528.55 USD
Justification	New computers for performance testing project

Requisition Lines

Action History

Attachments 1

Reject Approve

You can optionally define a third-level details screen to show even more granular details. For example, for a requisition approval notification, the approver can drill down from the header information in the first-level details screen to the requisition lines in the second-level details screen. From that screen, the approver can drill down to the third-level details screen to view distributions corresponding to a given line.

The first-level details screen also lets the approver drill down to additional screens to view the action history for a notification and any attachments.

All the approval details screens include the response section, from which the approver can choose either the Approve or the Reject response action for the notification. This section also includes a Request Information option for the approver to request more information before making a decision. However, the approval notification is not completed until the approver chooses either the Approve or the Reject response action.

Header

The first-level details screen for an approval notification displays the names and values of the header attributes, which are high-level attributes related to the approval object. Their values can be retrieved either from message attributes in the workflow definition or from the application tables that correspond to the approval object. For example, the header attributes for Expenses approval notifications include Person, Cost Center, Report Total, and Purpose. The header attributes for Purchase Orders approval notifications include Supplier, Site, Freight Terms, Amount, and Tax.

Follow these steps to prepare approval notification header information for display. You can then use the approvals data services configuration tool to configure the header information you have identified.

1. Identify all the attributes related to the approval object that you want to display as headers in the first-level details screen. Make a note of the workflow message attribute or application table and column in which each attribute's value is stored.

   If you have defined special header attributes for a message whose internal names begin with `#HDR_`, then those attributes are automatically included as headers in the first-level details screen. See: Header Attributes. Additionally, if you have defined the special message function WF_NOTIFICATION(ATTRS,...) to include a table of message attributes in the message, then those attributes are automatically included as headers in the first-level details screen as well. See: WF_NOTIFICATION() Message Function. You can also use the approvals data services configuration tool to configure other message attributes and details stored in application tables as headers to appear in this screen.

   You can use Oracle Workflow Builder to review the attributes defined for a message in the workflow definition. You can also use the following SQL command to retrieve a list of attributes for a message:

   ```
   select name, type, subtype, value_type, text_default
   from wf_message_attributes
   where message_type = '<item_type_name>'
   and message_name = '<message_name>';
   ```

   Note: If the approval object has a finite number of related attributes for the approver to review, it is recommended that you define them all as header attributes. In this way the approver can view all the relevant information in the first-level details screen without needing to drill down further.

   For example, an approval notification for a requisition can contain a variable number of requisition lines. In such a case, it is recommended to use a separate drill-down region to show the lines. By contrast, an approval notification for absences only has a given number of attributes to display, which does not vary at runtime. Consequently, for such a notification it is better to show all the attributes in the headers screen without requiring the approver to drill down.

   Additionally, if the message body for an approval notification is generated simply through text substitution of message attribute tokens, without any PL/SQL document content or embedded Oracle Application Framework regions, then you can define all the relevant message attributes as header attributes. In this case you do not need to define second-level or third-level details screens.

2. Review the display name for each header attribute. It is recommended to keep these names short for better display on a smartphone screen. For example, use the name "Report Total" rather than "Expense Report Total". Update the display name for workflow message attributes in the workflow definition if necessary. For attributes drawn from application tables, choose an appropriate display name that you will enter later when you define the attributes in the configuration tool.

3. Determine the order in which the header attributes should appear.

4. If a header attribute requires a unit of measure, it is recommended that you concatenate the value and the unit of measure into one attribute. For example:

   ```
   Amount   1,400 USD
   Quantity 1 EA
   ```

5. For attributes that you want to display as headers and that are not defined as workflow message attributes but are stored in the underlying application tables, you must perform similar steps to those required for retrieving data from the tables for display in the body details.

   - Identify an existing view object that can retrieve the approval object data from the tables, or use Oracle JDeveloper to create a suitable view object if no existing view object meets your needs.
   - Incorporate this view object into an application module, either existing or new.
   - Create a method in the AM implementation class that can initialize the view object for execution.

   For more information, see: Body Details.

Body Details

The second-level and third-level details screens can optionally be used for an approval notification to display more granular body content about the approval object, which is retrieved from the underlying application tables. For example, a second-level details screen can show requisition lines, expense lines, purchase order lines, and so on. A third-level details screen that is a child of a root or parent second-level details screen can show further details, such as distributions for each requisition line.

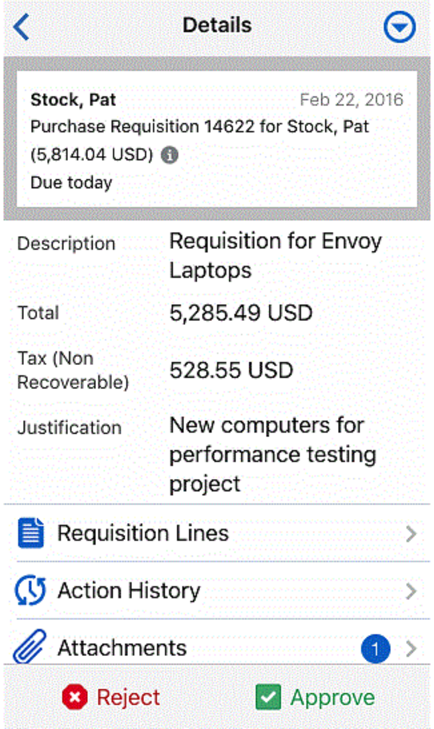
Note: To streamline the content presented to approvers, only two levels of body details screens are supported. That is, you cannot drill down any further from a third-level details screen.

You should configure body details to be displayed in the second-level and third-level details screens only if the number of records to display can vary at runtime. If the approval object has a finite number of related attributes for the approver to review, it is recommended that you define them all as header attributes so that the approver does not need to drill down to another details screen.

The Approvals app organizes body details for an approval object in the following hierarchy:

1. Region - A *region* is the highest level of approval object details content. In the app, a root or parent region is available through a link in the first-level details screen that the approver can tap to drill down to view more details. A non-root or child region is available through a drill-down icon within the second-level details screen for its associated root region. An approval object can have one or more detail regions.

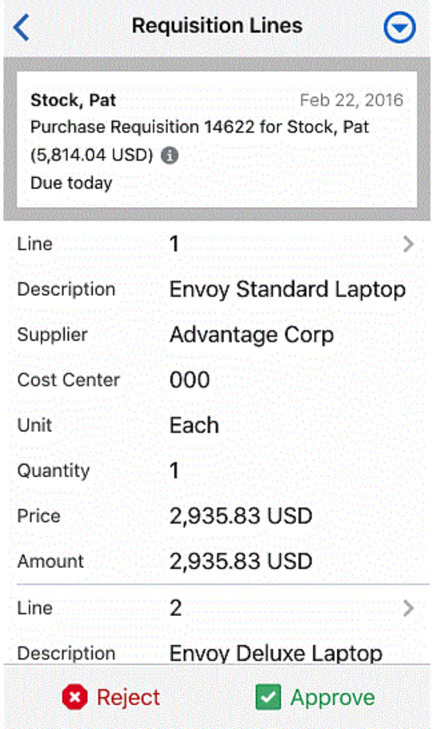
   Example of a first-level details screen for a requisition notification with a link to a root region for Requisition Lines



Stock, Pat Feb 22, 2016
Purchase Requisition 14622 for Stock, Pat
(5,814.04 USD)
Due today

Description	Requisition for Envoy Laptops
Total	5,285.49 USD
Tax (Non Recoverable)	528.55 USD
Justification	New computers for performance testing project

Example of a second-level details screen for a requisition notification with drill-down icons to child regions for requisition distributions



Stock, Pat Feb 22, 2016
Purchase Requisition 14622 for Stock, Pat
(5,814.04 USD)
Due today

Line	1
Description	Envoy Standard Laptop
Supplier	Advantage Corp
Cost Center	000
Unit	Each
Quantity	1
Price	2,935.83 USD
Amount	2,935.83 USD
Line	2
Description	Envoy Deluxe Laptop

2. View - A *view* is a group of records. A region is a collection of one or more views. Each view maps to an underlying view object from which approval object details records are retrieved. Each view within a region can optionally have a view title that is displayed before the view content.

3. Row - A *row* is an approval object detail record that contains a collection of one or more attributes.

4. Attribute - An *attribute* is an approval object data detail. An attribute is represented as a name - value pair in the Approvals app.

The approvals data services rely on the Oracle Application Framework BC4J implementation to retrieve data for an approval object from the underlying application tables . When you configure the body details for an approval notification to appear in the Approvals app, you must select the view object through which the content is retrieved from the tables.

- If the message body in the workflow definition is generated by an embedded Oracle Application Framework region, then you can use an existing view object for that region. See: Embedding Oracle Application Framework Regions in Messages.

- If the message body in the workflow definition is generated by a PL/SQL or PL/SQL CLOB document, then you must create a view object that can access the application tables to retrieve the corresponding data. See: To define a document attribute and Defining Messages.

The full notification body for a workflow notification can include several types of content when it is displayed in the Notification Details page in Oracle E-Business Suite. Content shown in the following formats can be easily converted to be displayed in the Approvals app:

- Details that are displayed in a table format as rows of attributes in the Notification Details page
- Details that are displayed as name - value pairs of attributes in the Notification Details page

However, the Approvals app cannot include the following:

- Complex data such as images or graphs
- Links to web pages
- Respond attributes to capture additional user response values in addition to the response action.

  Note: The only exception is the WF_NOTE attribute. If the WF_NOTE respond attribute is defined for a message, then the Approvals app stores any comments that the approver entered while responding in the WF_NOTE attribute.

Follow these steps to prepare your approval object body details for display in the app. You can then use the approvals data services configuration tool to configure the body details you have identified.

To determine what body details to display in the app

1. Review the notification body content displayed in the Notification Details page in Oracle E-Business Suite, and identify the regions needed to display body details for the approval object. To streamline the content presented to approvers in the app, it is recommended that you include only the most important body details necessary to allow approvers to make a decision.
2. For each region, select a region name. The region name for a region displayed in a second-level details screen will appear as a link in the first-level details screen. It is recommended to keep the region name short for better display, such as "Expense Lines" or "Vacancy Details".
3. If you plan to define more than one region for an approval object, determine the order in which the regions should appear.
4. Identify the views to display within each region.
5. For each view, select a view title that describes the content in the view to the approver. To streamline the content, it is recommended to keep the view title short, such as "Credit Card Expenses" or "Cash and Other Expenses".

   Note: If a region contains only one view, then you do not need to provide a view title.
6. If a region contains more than one view, determine the order in which the views should appear.
7. Identify the attributes to display in each row. To streamline the information presented to approvers, you should select only the most important details to appear as attributes in each row. For each attribute, select an attribute name to appear as a label before the attribute value. It is recommended to keep the attribute name short for better display.
8. Determine the order in which the attributes should appear within the row. The key attribute that identifies the row should appear first. For example, in the row for an expense line, the key attribute could be "Expense Type".
9. If a region displayed in a second-level details screen will have a child region displayed in a third-level details screen, determine which attribute in the root or parent region should have the drill-down icon displayed next to it for approvers to navigate to the child region details.
10. If an attribute requires a unit of measure, it is recommended that you concatenate the value and the unit of measure into one attribute. For example:

    ```
    Amount   1,400 USD
    Quantity 1 EA
    ```
11. After you have identified the body details that you want to display in the app, you must determine how to retrieve the corresponding data from the underlying application tables. Begin by reviewing the message definition in Oracle Workflow Builder to determine how the message body is generated. For body content that is included through a message attribute token, including embedded Oracle Application Framework regions and PL/SQL documents, check how the message attribute value is set. You may need to review your related application code if you have implemented logic that sets specific values at runtime. Then prepare the objects needed to retrieve the data, depending on how the message body content is generated.
    - For an embedded Oracle Application Framework region, see: To use body details defined through an embedded Oracle Application Framework region.
    - For a PL/SQL document, see: To use body details defined through a PL/SQL document.
12. If you want a particular region to be displayed in the apps only in certain circumstances, prepare the attribute you will use to configure that region for conditional display. See: To prepare a region for conditional display.

To use body details defined through an embedded Oracle Application Framework region

1. For body content defined through an embedded Oracle Application Framework region, the message attribute value should ultimately resolve to a Java Server Page (JSP) call in the following format that references that region.

   ```
   JSP:/OA_HTML/OA.jsp?OAFunc=<Function_Name>&<ParameterName1>=<ParameterValue1>&<ParameterName2>=<ParameterValue2>...
   ```

   Identify the JSP call for each Oracle Application Framework region embedded in the message body. Note that a message body can include multiple embedded Oracle Application Framework regions. If so, ensure that you identify the JSP call for each region.
2. For each region, use Oracle JDeveloper to review the Oracle Application Framework region implementation referenced in the JSP call. In Oracle JDeveloper, identify the view objects used to retrieve the approval object data. The region may extend other regions that each use a separate view object. Select the view objects that you want to include in the approval notification body details. It is recommended that you include only the most important content to streamline the content displayed in the app.

   Note: If the existing view objects do not meet your needs for approvals data services configuration, you can optionally create new view objects to retrieve content specifically for approvals data services. However, in this case you must maintain both implementations separately going forward. In most cases, you can reuse the existing view objects to retrieve content for the Approvals app as well.

3. Verify that the view attributes you want to reference are defined with the appropriate attribute types so that the client device can format their values according to the client locale. For example, the client device can display values marked as Date and Number attributes with the appropriate format masks for the locale that the user selected for the device. However, the client device cannot format date and number values contained in String attributes.

   - View attributes that contain dates should have either the Date attribute type to display the date value without a time value, or the Timestamp attribute type to display a date value that includes a time value.
   - View attributes that contain numbers should have the Number attribute type.
   - View attributes that contain text strings should have the String attribute type.

   For an attribute that contains a currency value, it is recommended that you format the amount and the currency code in the view object's SQL before returning the value as a String. For example:

   ```
   SELECT ... TO_CHAR((quantity * unit_price),
                      FND_CURRENCY.GET_FORMAT_MASK(currency_code, 40),
                      'NLS_NUMERIC_CHARACTERS='''||FND_PROFILE.VALUE('ICX_NUMERIC_CHARACTERS')||'''')
              || ' ' || currency_code as line_amount, ...
   FROM ...
   ```

   This SQL code can perform rounding of the currency amount based on the number (in this example, the number resulting from the `quantity * unit_price` calculation) and the currency code. For example, suppose the number resulting from the `quantity * unit_price` calculation is 123.45678 and the currency code is USD, which has a precision of 2 as defined in the FND_CURRENCIES table. In this case the SQL would return the amount as follows:

   ```
   select TO_CHAR(1234.56789,
                  FND_CURRENCY.GET_FORMAT_MASK('USD', 40),
                  'NLS_NUMERIC_CHARACTERS='''||FND_PROFILE.VALUE('ICX_NUMERIC_CHARACTERS')||'''')
   from dual;
   ```

   The resulting rounded amount would be: `1,234.57`

   Next, suppose the number is 123.45678 but the currency is JPY, which has a precision of 0 as defined in the FND_CURRENCIES table. In this case the SQL would return the amount as follows:

   ```
   select TO_CHAR(1234.56789,
                  FND_CURRENCY.GET_FORMAT_MASK('JPY', 40),
                  'NLS_NUMERIC_CHARACTERS='''||FND_PROFILE.VALUE('ICX_NUMERIC_CHARACTERS')||'''')
   from dual;
   ```

   The resulting rounded amount would be: `1,235`

4. Identify the BC4J application module that includes the view objects you have selected.

5. Identify a method in the AM Implementation class (AMImpl) that can initialize the selected view objects, from which approvals data services can retrieve records. If no existing method meets your needs, add a new method in the AMImpl class to initialize the view objects.

   To be used for the approvals data services, the initialization method must accept one or more of the following input parameters and use those parameters to initialize and execute the selected view objects.

   - Workflow item type
   - Item key
   - Notification ID
   - Activity ID
   - Any message attribute defined for the workflow message, such as the expense report ID attribute for an expense message or the requisition number attribute for a requisition message

   For example, the `oracle.apps.ap.oie.workflow.apexp.server.NotifAMImpl` class used by embedded Oracle Application Framework regions for Expense approvals includes the following method, which accepts the report header ID and initializes all the view objects to retrieve expense lines:

   ```
    public void initExpQuery(Number nReportHeaderId);
   ```

   This method can be used in the message configuration without requiring any changes because the report header ID value is available as a message attribute.

   If you require certain business logic to be executed before the view objects are executed, you can implement that business logic within the initialization method as well.

   Note: It is recommended that you use the same initialization method for your existing embedded Oracle Application Framework regions in notifications displayed in the Worklist pages and for your approvals data services message configuration.

   For a view in a non-root or child region, the initialization method should include one or more parameters that will be mapped to view attributes from the related view in the parent or root region. That is, you must select at least one view attribute from the parent to pass to the initialization method for a child view. Other input parameters for the initialization method for the child can be drawn from the workflow attributes listed above or from constant values.

6. When you configure the message in the approvals data services configuration tool, you will specify the view objects you selected and the AMImpl method that initializes each view object. At runtime, the approvals data services execute the initialization method which in turn executes the view object. The approvals data services then retrieve the records from the view object and return the XML response containing the approval object data to the client.

To use body details defined through a PL/SQL document

1. For body content defined through a PL/SQL document or PL/SQL CLOB document, the message attribute value should ultimately resolve to a PL/SQL procedure reference in the following formats, respectively:

   plsql:*<procedure>/<document_identifier>*

   plsqlclob:*<procedure>/<document_identifier>*

Identify the PL/SQL procedure and review the SQL queries within it to determine how they retrieve approval object data from the underlying application tables.

2. Ensure that the PL/SQL procedure includes a SQL query that accepts one or more of the following input parameters:

   - Workflow item type
   - Item key
   - Notification ID
   - Activity ID
   - Any message attribute defined for the workflow message, such as the expense report ID attribute for an expense message or the requisition number attribute for a requisition message

   If necessary, rewrite the SQL with appropriate bind parameters to accept at least one of these parameters.

3. Use Oracle JDeveloper to create a view object that implements this SQL query and retrieves the approval object data. Then incorporate this view object into an application module, either existing or new, and create a method in the AM implementation class that can initialize the view object for execution as described in step 14.

4. When you configure the message in the approvals data services configuration tool, you will specify the view objects you selected and the AMImpl method that initializes each view object. At runtime, the approvals data services execute the initialization method which in turn executes the view object. The approvals data services then retrieve the records from the view object and return the XML response containing the approval object data to the client.

To prepare a region for conditional display

In some cases, you may want a particular region to be displayed in the apps only in certain circumstances. For example, if the same message definition is used as a boilerplate template to send approval notifications for different approval objects, you may want to include different detail regions for the different types of approval notification content and display only the relevant regions for each approval notification in the app. When you configure the message in the approvals data services configuration tool, you can define rules for the regions to control whether the regions are displayed.

A rule compares a value that is set dynamically at runtime with a constant value called an operand to determine whether to display the region. You can use a message attribute to obtain the runtime value to use in the comparison.

- You can use an existing message attribute that stores a value representing the approval object for a particular approval notification.
- Alternatively, if no existing message attribute meets your needs, you can define a new special message attribute named #APPROVALS_OBJECT and include logic in your workflow process to set the message attribute value dynamically at runtime to a value that represents the approval object for the notification.

For example, the workflow message PO_REQ_APPROVE_JRAD in the item type REQAPPRV is used to send Purchase Requisition approvals and Contract Requisition approvals. This message has a message attribute named "Is Contractor Requisition" that stores a value of either Y or N. If one region is defined to display purchase requisition details, and another region is defined to display contract requisition details, this message attribute can be used in region-level rules specifying that the first region should be displayed only when the "Is Contractor Requisition" attribute is set to N, and the second region should be displayed only when the "Is Contractor Requisition" attribute is set to Y.

Alternatively, the #APPROVALS_OBJECT attribute could be used to differentiate Purchase Requisition approvals and Contract Requisition approvals. In this case, the #APPROVALS_OBJECT attribute would need to be defined for the message and the workflow would include logic to set the #APPROVALS_OBJECT attribute to PURCHASE_REQ for purchase requisitions or to CONTRACT_REQ for contract requisitions. The #APPROVALS_OBJECT attribute can then be used in region-level rules to determine at runtime which region to display.

1. To prepare to define a rule for a region, determine which message attribute can retrieve a value representing the approval object for the approval notifications at runtime.
2. If you plan to use the #APPROVALS_OBJECT message attribute, update the workflow definition to add this attribute and include appropriate logic to set its value.

Action History

The action history for a workflow notification shows what actions users have performed on the notification to date. See: Action History.

Oracle Workflow provides a default action history that records when the following actions have occurred:

- A user initially submits the notification.
- A user transfers or delegates the notification.
- The notification times out.
- A user requests more information about the notification.
- Another user provides the requested information.
- A user approves or rejects the notification.

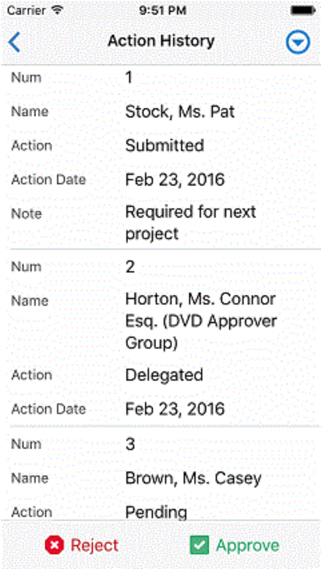
To use the default action history, an approval object must be processed by a single workflow notification activity throughout its entire lifecycle. That is, if multiple levels of approval are required, the workflow process should loop back to the same notification activity to send a notification to each approver until all approvals are completed. In this way Oracle Workflow can record all the approvals as actions for the same notification activity.

If an approval object uses the default Oracle Workflow action history, then the Approvals app automatically includes that action history in the approval notification details. In this case you do not need to perform any further configuration for the action history. The approver can drill down from the first-level details screen to view the action history, including the following details:

- A sequence number indicating the order in which the actions on this notification took place, beginning at one (1) for the initial submission of the request.
- The user who performed the action.
- The action performed on the notification.

- The date that a user acted on the notification.
- Any additional note from the user who performed the action.

Action History Screen



Num	1
Name	Stock, Ms. Pat
Action	Submitted
Action Date	Feb 23, 2016
Note	Required for next project
Num	2
Name	Horton, Ms. Connor Esq. (DVD Approver Group)
Action	Delegated
Action Date	Feb 23, 2016
Num	3
Name	Brown, Ms. Casey
Action	Pending

You can optionally define a custom action history for an approval notification to replace the standard action history, using the special message attribute named #HISTORY. For example, if an approval object is routed from one notification activity to another or from one item type to another during its lifecycle, then the default action history does not apply, but you can track the actions performed on the approval object using a custom action history retrieved from the underlying application tables.

To display the custom action history in the Approvals app, review its contents and identify the attributes that you want to display in the app. Then you must perform similar steps to those required for retrieving data from the application tables for display in the body details.

- Identify an existing view object that can retrieve the approval object's action history data from the application tables, or use Oracle JDeveloper to create a suitable view object if no existing view object meets your needs.
- Incorporate this view object into an application module, either existing or new.
- Create a method in the AM implementation class that can initialize the view object for execution.

For more information, see: Body Details.

You can then use the approvals data services configuration tool to configure the custom action history region with the content you identified.

Attachments

The Approvals app can display two types of attachments for approval notifications:

- PL/SQL document attachments
- Oracle E-Business Suite attachments specified in the #ATTACHMENTS message attribute

If these types of attachments are defined for a notification in its workflow definition, then the Approvals app automatically includes them in the approval notification details. You do not need to perform any further configuration for the attachments. The approver can drill down from the first-level details screen to view the attachment contents.

Approval Response Actions

In the response section, which appears in all the approval details screens, the approver can choose either the Approve or the Reject response action for the notification. The Approvals app supports only these two response actions. You can optionally configure different labels to be displayed for these actions, such as `Yes` and `No` instead of `Approve` and `Reject`, respectively. However, one action should still generally indicate approval or acceptance, while the other action should generally indicate disapproval or rejection. The app displays the same icon images regardless of the labels you specify: a green check mark icon for the Approve response action and a red X or cross icon for the Reject response action.

You can configure messages for the app that have additional possible results as part of their workflow definitions, but you must select only two of the results to map to the Approve and Reject actions within the app. To respond with other results, users must access messages through the Worklist pages or through e-mail.

After choosing the Approve or Reject response action, the approver can optionally enter a note with comments about the decision.

The response section also includes a Request Information option for the approver to request more information before making a decision. However, the approval notification is not completed until the approver chooses either the Approve or the Reject response action.

An approver can also choose to reassign a notification.

Follow these steps to prepare approval notification results for use as responses within the Approvals app. You can then use the approvals data services configuration tool to configure the responses you have identified.

1. If the workflow notification definition includes more than two possible results, select only two to use within the app. For example, if the possible results are `Approve`, `Reject`, and `Return for Correction`, the `Approve` and `Reject` results would be most appropriate to use within the app, while approvers would need to access the notifications through the Worklist pages or e-mail to respond with the `Return for Correction` result.

You can use Oracle Workflow Builder to review the lookup type used in the result for the message and the lookup codes available as possible results in that lookup type.

2. Determine which result indicates approval or acceptance, to map to the Approve response action, and which result indicates disapproval or rejection, to map to the Reject response action. Also, decide what labels you want to display for these actions in the app. The most commonly used labels are `Approve` and `Reject`. You can optionally configure other labels appropriate to your approval notification, such as `Yes` and `No`.

Configuring Messages for Approvals Data Services

After you have identified and prepared all the approval notification content that you want to display in the Approvals app, use the approvals data services configuration tool to configure the message metadata for use with the approvals data services. For each message, follow this configuration process:

1. Search for the message.
2. Configure header attributes based on workflow message attributes and response actions.
3. Configure regions for header attributes based on data from underlying application tables, for body details, and for action history.
4. Review and verify the message configuration.

Note: The configuration that you perform in the approvals data services configuration tool does not affect the appearance of the workflow notifications in the Worklist pages or in e-mail. However, if you made changes in your workflow definitions, Oracle Application Framework view objects, or PL/SQL procedures to prepare the approval notification content, then those changes can affect the appearance of the workflow notifications in the Worklist pages and in e-mail.

Customization Levels

You can use the configuration tool to define your custom message configurations and to modify message configurations seeded by Oracle E-Business Suite that are defined to allow customizations. Most components of a message configuration are assigned a customization level that determines whether you can update the component's definition. Oracle Workflow uses the customization levels to protect Oracle E-Business Suite seed data and to preserve your custom configurations and customizations in an upgrade.

Components in a message configuration can have one of the following customization levels:

- Core - No changes can be made to components with this level. You can view the component's properties, but you cannot update them. This level is used only in configurations seeded by Oracle E-Business Suite.
- Limit - You can enable or disable the component for display in the Approvals app, but no other changes can be made to the component. This level is used only in configurations seeded by Oracle E-Business Suite.
- User - You can update the properties for components with this level. This level is automatically set in custom configurations that you define.

The following components of a message configuration have customization levels assigned to them:

- Header attributes based on workflow message attributes
- Regions
- Views
- View attributes

In seeded message configurations, for regions, the views that belong to those regions, and the view attributes that belong to those views, the customization level of the higher-level component determines the possible customization levels for the lower-level components. That is, a region's customization level determines the possible customization levels for its views:

- If a seeded region has a customization level of Core, then its views - and their view attributes - can only have customization levels of Core.
- If a seeded region has a customization level of Limit, then its views can have customization levels of Limit or Core.
- If a seeded region has a customization level of User, then its views can have customization levels of User, Limit, or Core.

Similarly, a view's customization level determines the possible customization levels for its attributes:

- If a seeded view has a customization level of Core, then its attributes can only have customization levels of Core.
- If a seeded view has a customization level of Limit, then its attributes can have customization levels of Limit or Core.
- If a seeded view has a customization level of User, then its attributes can have customization levels of User, Limit, or Core.

When you view a seeded message configuration, the configuration tool displays some fields as read-only fields and hides some options according to the customization levels of the components.

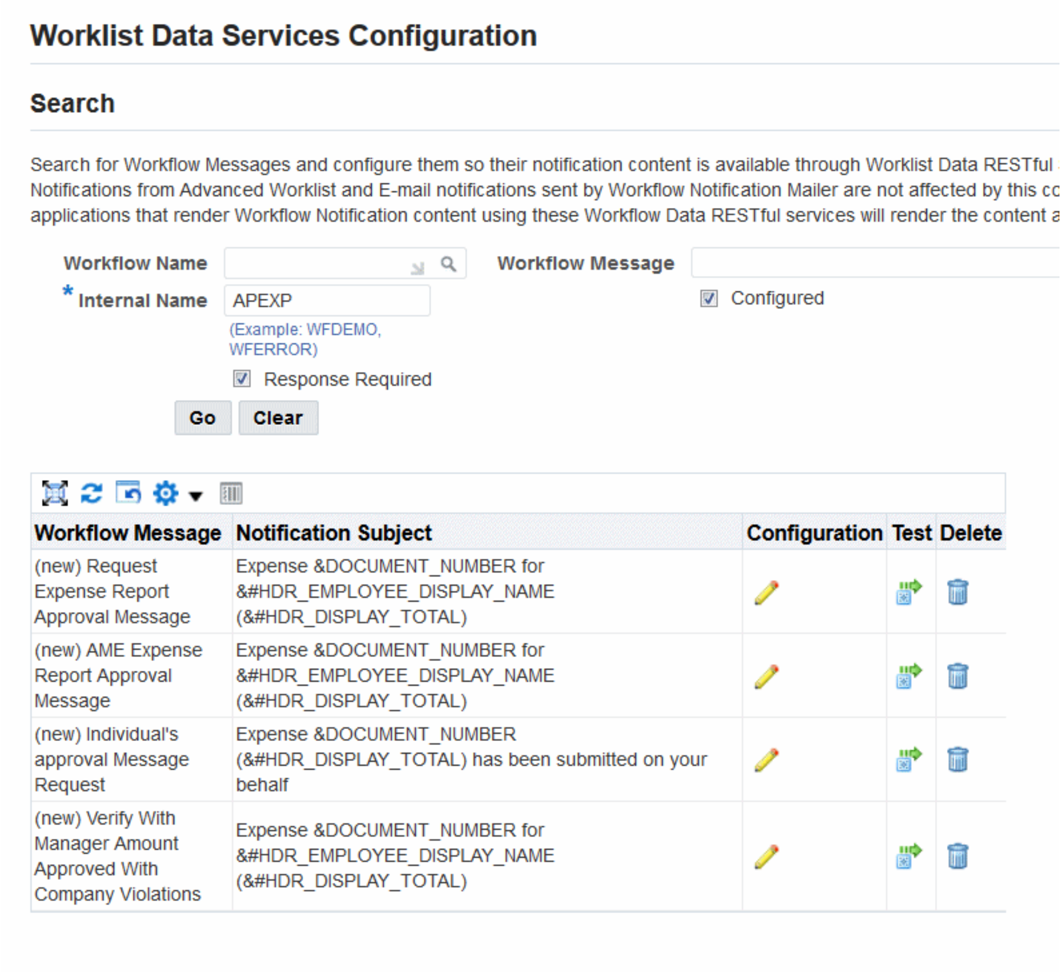
In custom message configurations that you define, all components are always assigned a customization level of User.

Note: You can add a custom region in the configuration for any message, including seeded configurations. For example, you can add a custom details region to a seeded message to show content from a custom view object. You can also hide a seeded details region that has a customization level of Limit and replace it with a custom details region.

To manage message configurations

1. Navigate to the Worklist Data Services Configuration - Search page from the Oracle E-Business Suite Navigator by choosing either Workflow Administrator Web Applications: Approvals Data Services Manager > Configuration or a custom responsibility and menu path specified by your system administrator.

   Worklist Data Services Configuration - Search Page



Worklist Data Services Configuration

Search

Search for Workflow Messages and configure them so their notification content is available through Worklist Data RESTful
Notifications from Advanced Worklist and E-mail notifications sent by Workflow Notification Mailer are not affected by this cc
applications that render Workflow Notification content using these Workflow Data RESTful services will render the content a

Workflow Name

* Internal Name APEXP
(Example: WFDEMO, WFERROR)

☑ Response Required

Workflow Message

☑ Configured

Workflow Message	Notification Subject	Configuration	Test	Delete
(new) Request Expense Report Approval Message	Expense &DOCUMENT_NUMBER for &#HDR_EMPLOYEE_DISPLAY_NAME (&#HDR_DISPLAY_TOTAL)	✎	➜	🗑
(new) AME Expense Report Approval Message	Expense &DOCUMENT_NUMBER for &#HDR_EMPLOYEE_DISPLAY_NAME (&#HDR_DISPLAY_TOTAL)	✎	➜	🗑
(new) Individual's approval Message Request	Expense &DOCUMENT_NUMBER (&#HDR_DISPLAY_TOTAL) has been submitted on your behalf	✎	➜	🗑
(new) Verify With Manager Amount Approved With Company Violations	Expense &DOCUMENT_NUMBER for &#HDR_EMPLOYEE_DISPLAY_NAME (&#HDR_DISPLAY_TOTAL)	✎	➜	🗑

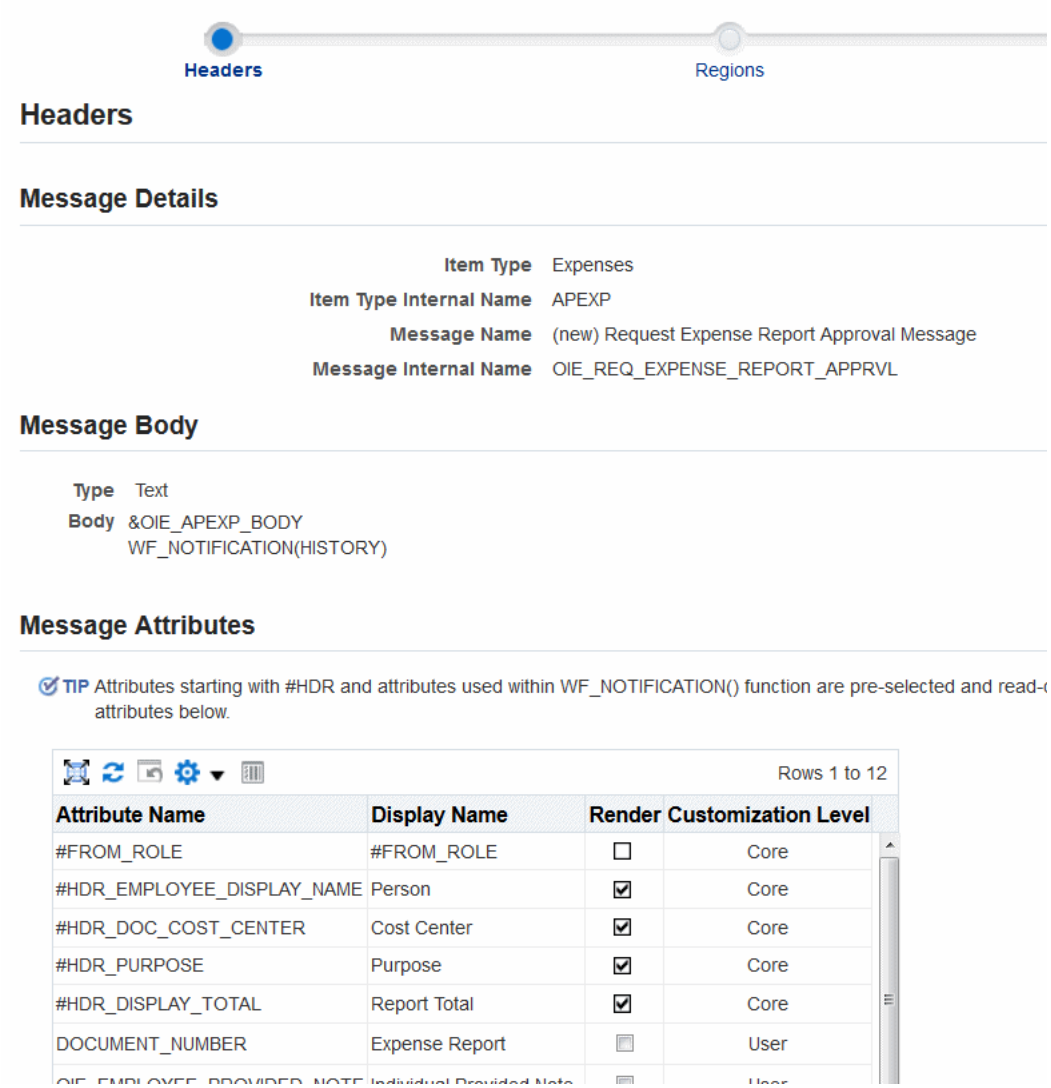
2. Search for the messages you want to configure by selecting the workflow item type that contains those messages. You can enter the internal name for the item type directly, or select the workflow item type display name, in which case the internal name is populated automatically. You can also specify the following optional search criteria:
    - Workflow Message - Enter the display name of a specific message to configure.
    - Configured - Select this option to display messages that have already been configured for approvals data services. Leave this option deselected to display message that have not yet been configured.
    - Response Required - Select this option to display only messages that require a response. It is recommended that you select this option, because the Approvals app is designed only for notifications that require a response.
3. To configure a message for use with approvals data services, or to review and update an existing configuration, choose the configuration icon for that message. Perform the following tasks to configure a message:
    - Configure header attributes based on workflow message attributes and response actions. See: To configure headers and response actions for a message.
    - Configure regions for header attributes based on data from underlying application tables, for body details, and for action history. See: To configure regions for a message.
    - Review and verify the message configuration. See: To review and verify a message configuration.
4. To test an existing configuration for a message using a notification ID, choose the test icon for that message. See: To review and verify a message configuration.
5. To delete a message configuration that you no longer require, choose the delete icon for that message.

    Note: Do not delete message configurations seeded by Oracle E-Business Suite.

To configure headers and response actions for a message

1. Navigate to the Headers page by choosing the configuration icon for a message from the Worklist Data Services Configuration - Search page. The page shows the display name and internal name for the item type to which the message belongs, as well as the display name and internal name for the message. The page also displays the message body definition as specified in Oracle Workflow Builder.

    Headers Page: Message Details, Message Body, and Message Attributes



Headers

Message Details

Item Type	Expenses
Item Type Internal Name	APEXP
Message Name	(new) Request Expense Report Approval Message
Message Internal Name	OIE_REQ_EXPENSE_REPORT_APPRVL

Message Body

Type Text

Body &OIE_APEXP_BODY
WF_NOTIFICATION(HISTORY)

Message Attributes

TIP Attributes starting with #HDR and attributes used within WF_NOTIFICATION() function are pre-selected and read-
attributes below.

Rows 1 to 12

Attribute Name	Display Name	Render	Customization Level
#FROM_ROLE	#FROM_ROLE	□	Core
#HDR_EMPLOYEE_DISPLAY_NAME	Person	☑	Core
#HDR_DOC_COST_CENTER	Cost Center	☑	Core
#HDR_PURPOSE	Purpose	☑	Core
#HDR_DISPLAY_TOTAL	Report Total	☑	Core
DOCUMENT_NUMBER	Expense Report	□	User
OIE_EMPLOYEE_PROVIDED_NOTE	Individual Provided Note	□	User

2. Review the list of message attributes as defined for the message in Oracle Workflow Builder. Any header attributes whose internal names begin with `#HDR_` are automatically marked to be included as headers in the first-level details screen in the Approvals app. Additionally, if you defined the special message function WF_NOTIFICATION(ATTRS,...) to include a table of message attributes in the message, then those attributes are automatically marked to be included as headers in the app as well. You can optionally select the Render option for any additional message attributes with a customization level of Limit or User that you want to display as headers in the app.

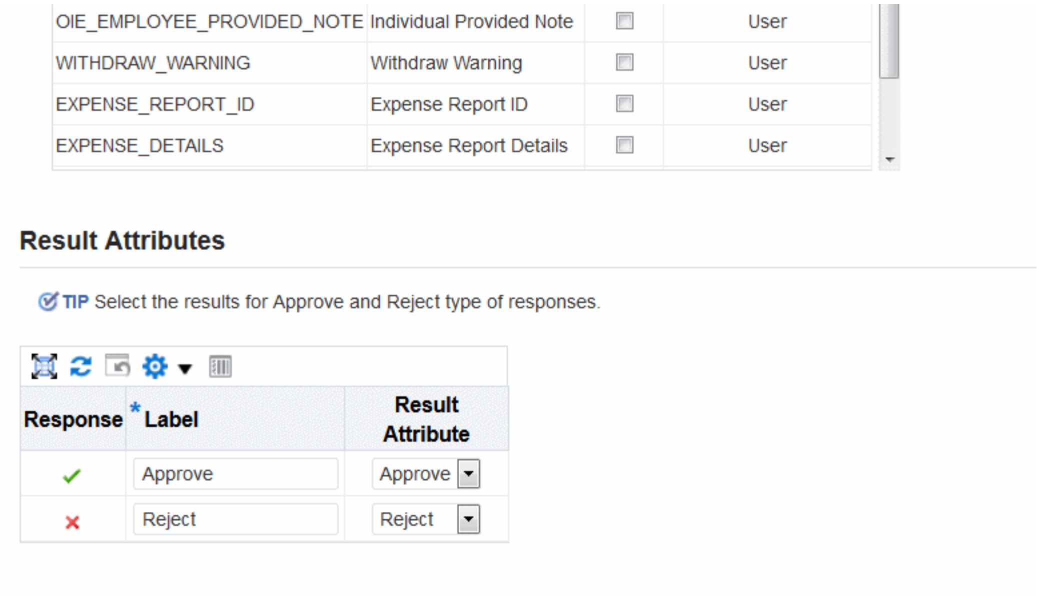
   Note: You cannot change the Render option setting for message attributes with a customization level of Core.

   Note: You can also display data from application tables as headers in the app. However, you perform the configuration for such data in the Regions page rather than in this page.

3. Select the result attributes for the message that you want to map to the Approve and Reject response actions in the app. You can optionally update the labels that will appear for these actions in the app.

   Note: You can only choose two results to use as response actions within the app. To respond with other results, users must access messages through the Worklist web pages or through e-mail.

   Headers Page: Result Attributes



OIE_EMPLOYEE_PROVIDED_NOTE	Individual Provided Note	☐	User
WITHDRAW_WARNING	Withdraw Warning	☐	User
EXPENSE_REPORT_ID	Expense Report ID	☐	User
EXPENSE_DETAILS	Expense Report Details	☐	User

Result Attributes

TIP Select the results for Approve and Reject type of responses.

Response	* Label	Result Attribute
✓	Approve	Approve
✗	Reject	Reject

4. Choose Next to navigate to the Regions page.

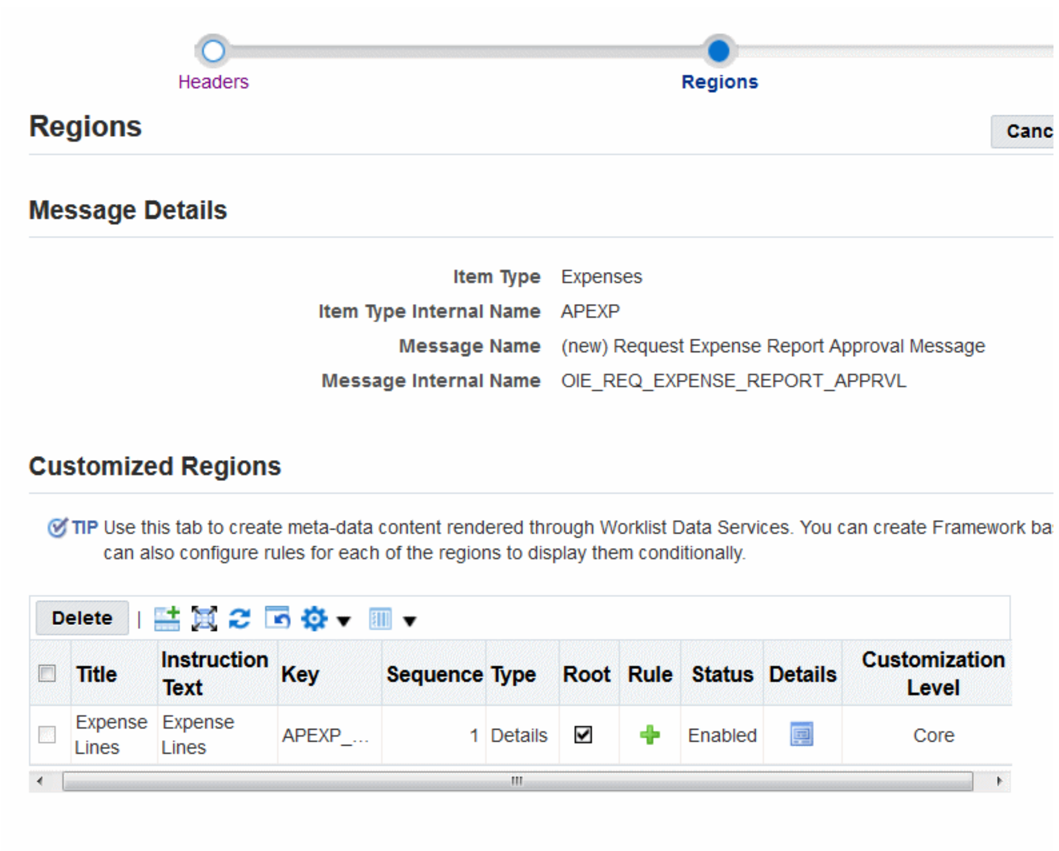
To configure regions for a message

1. To begin defining a new region in the Regions page, choose the Add a Region icon.

   For existing regions, the customization level determines what updates you can make.

   - Core - You cannot update any region properties or any details of the region's views or their view attributes.
   - Limit - You can update the status for the region to enabled or disabled, but you cannot update any other region properties. You may be able to update the status for the region's views and their view attributes depending on their respective customization levels.
   - User - You can update all region properties except for the customization level. You may be able to update the region's views and their view attributes depending on their respective customization levels.

   Regions Page



Regions

Message Details

Item Type: Expenses

Item Type Internal Name: APEXP

Message Name: (new) Request Expense Report Approval Message

Message Internal Name: OIE_REQ_EXPENSE_REPORT_APPRVL

Customized Regions

TIP Use this tab to create meta-data content rendered through Worklist Data Services. You can create Framework ba can also configure rules for each of the regions to display them conditionally.

☐	Title	Instruction Text	Key	Sequence	Type	Root	Rule	Status	Details	Customization Level
☐	Expense Lines	Expense Lines	APEXP_...	1	Details	☑	+	Enabled		Core

2. Enter a title for the region. In the app, the approver can tap the region title for body and action history regions to drill down to the associated details.

3. Enter optional instruction text to provide more guidance to the approver.

4. The configuration tool automatically generates a unique key to identify the region. You can optionally update the key.
5. Enter a sequence number to specify the position in which the region appears relative to other regions of the same type.

   Note: Each region type has a separate sequence. For instance, you can define a sequence number of 1 for a header region, sequence numbers of 1, 2, and 3 for details regions, and a sequence number of 1 for a history region.
6. Select the region type:
   - Header - Header attributes retrieved from application tables for display in the first-level details screen in the app
   - Details - Body details for display in the second-level or third-level details screens in the app
   - History - A custom action history
7. Select the Root check box if this region should appear in a second-level details screen in the app. Deselect this check box if this region should appear in a third-level details screen in the app. A region that is not a root region must be associated with a view in a parent region that is a root region, from which the approver can drill down to the third-level details screen for the non-root or child region.
8. You can optionally define a rule to control whether this region appears in the app based on a runtime condition. To do so, select the rule icon for the region.
   - To begin defining a new rule, choose the Add a Rule icon.
   - Enter a display name for the rule.
   - The configuration tool automatically generates a unique key to identify the rule. You can optionally update the key.
   - The Message Attribute rule type is automatically selected.
   - In the criterion field, select the message attribute used to retrieve a value dynamically at runtime to determine whether the region should be displayed. If you defined a message attribute named #APPROVALS_OBJECT for this purpose, select that attribute here. You can also select another message attribute whose runtime value can be used in the comparison.

     The criterion and the operand are compared using the `Is` operator. That is, the region is displayed only if the values of criterion and the operand match.
   - Enter a constant operand value against which the criterion is compared.

     For example, suppose a rule is defined for a detail region in a contract requisition approval notification, with the criterion set to the message attribute "Is Contractor Requisition", the operator "Is", and the operand set to the constant value "Y". At runtime, the Approvals app will render this region only for contract requisition notifications, which have the "Is Contractor Requisition" attribute set to "Y".

     If the same message also sends purchase requisition approval notifications, a purchase requisition detail region should also be defined for the message. A rule should be defined for that region with the criterion set to the message attribute "Is Contractor Requisition", the operator "Is", and the operand set to the constant value "N". At runtime, the Approvals app will render this region only for purchase requisition notifications, which have the "Is Contractor Requisition" attribute set to "N".
   - To delete a rule that you no longer require, select the delete icon.
9. In the Customized Regions list, select the status for the region, either Enabled or Disabled. Only enabled regions appear in the Approvals app.
10. To define the views for the region that retrieve data from the underlying application tables, select the details icon for the region.

    You can define new views only for regions with a customization level of User.

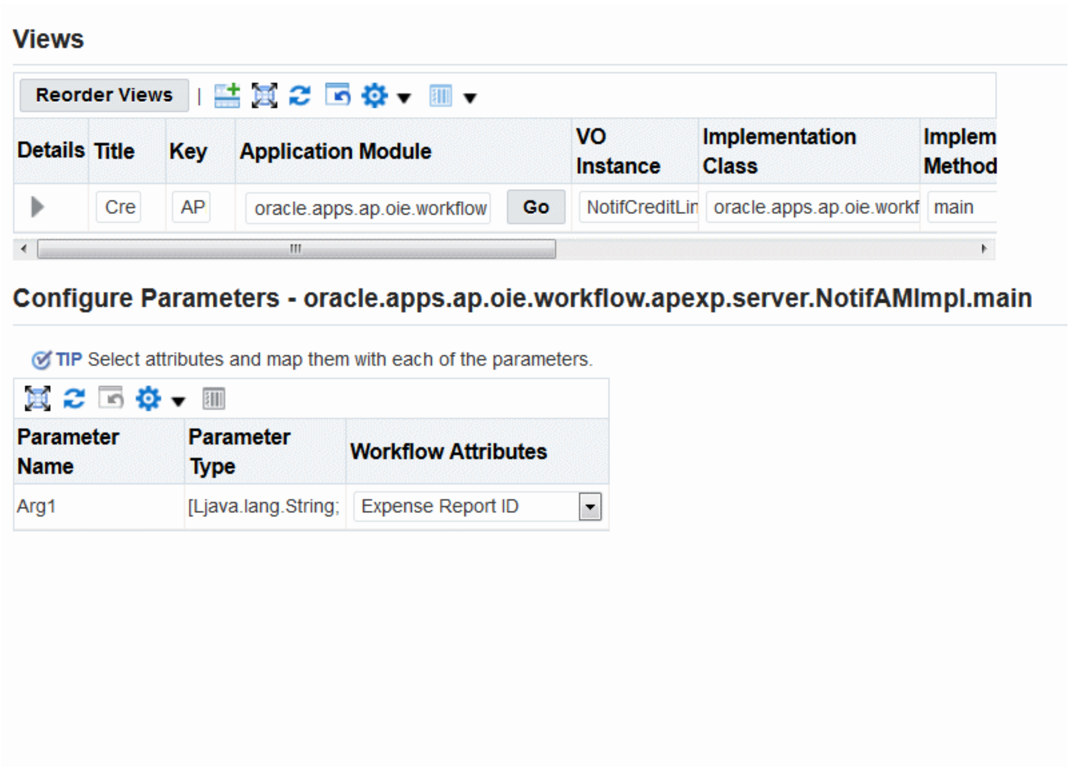
    For existing views, the customization level for the view determines what updates you can make.
    - Core - You cannot update any view properties or any details of the view attributes.
    - Limit - You can update the status for the view to enabled or disabled, but you cannot update any other view properties. You may be able to update the rendering status for the view attributes depending on their respective customization levels.
    - User - You can update the view properties, except for the Application Module, VO Instance, Implementation Class, and Implementation Method properties and the customization level. You may be able to update the view attributes depending on their respective customization levels.
11. For a child or non-root region, select the root region that is the parent of this child region. Then select the view within the parent region from which approvers should drill down to this child region.

    Note: Each view in a parent region can allow drilldown only to one child region. Consequently, if you define multiple child regions for the same parent region, the list of values in the Parent Region View field includes only those views that are not already assigned to another child region.
12. To begin defining a new view, choose the Add a View icon in the Views page. Then enter information for the view as follows:
    - Enter a title for the view to help differentiate the content of this view from the content of other views when displayed in the app. If the region contains only one view, you can optionally omit the title.
    - The configuration tool automatically generates a unique key to identify the view. You can optionally update the key.
    - Enter the Oracle Application Framework application module (AM) that contains the view objects to retrieve data from the application tables. Then select Go to load the view objects for that AM into the list of values for the following field.
    - Select the view object that contains the attributes you want to retrieve from the application tables.
    - Select the AM implementation class (AMImpl) that contains the method used to initialize and execute the view object.

      Note: The list of values may also include view object implementation classes. However, for Approvals app purposes you should use an AMImpl class, because an AMImpl class is common to all view objects, and it is a best practice to initialize view objects in the AMImpl method to manage them better.
    - Select the initialization method within that implementation class.

13. To configure the input parameters to be passed to the initialization method, select the Configure Parameters icon for the view.

    Views Page: Configure Parameters

    ## Views

Details	Title	Key	Application Module		VO Instance	Implementation Class	Implem Method
▸	Cre	AP	oracle.apps.ap.oie.workflow	Go	NotifCreditLin	oracle.apps.ap.oie.workf	main

    ## Configure Parameters - oracle.apps.ap.oie.workflow.apexp.server.NotifAMImpl.main

    TIP Select attributes and map them with each of the parameters.

Parameter Name	Parameter Type	Workflow Attributes
Arg1	[Ljava.lang.String;	Expense Report ID

    - For a view in a root region, for each input parameter that the method accepts, select the value to be passed to that parameter. You can choose from the following possible values:
        - Workflow item type
        - Item key
        - Notification ID
        - Activity ID
        - Any message attribute defined for the workflow message, such as the expense report ID attribute for an expense message or the requisition number attribute for a requisition message
    - For a view in a child or non-root region, the initialization method should include one or more parameters that will be mapped to view attributes from the related view in the parent or root region. That is, you must select at least one view attribute from the parent to pass to the initialization method for a child view. Other input parameters for the initialization method for the child can be drawn from the workflow attributes or from constant values. For each input parameter that the method accepts, first select the value type for the parameter, and then specify the value to be passed to that parameter.
        - Product Attributes value type - Select the attribute of the parent view whose value should be passed as the value for this input parameter in the initialization method for the child view.

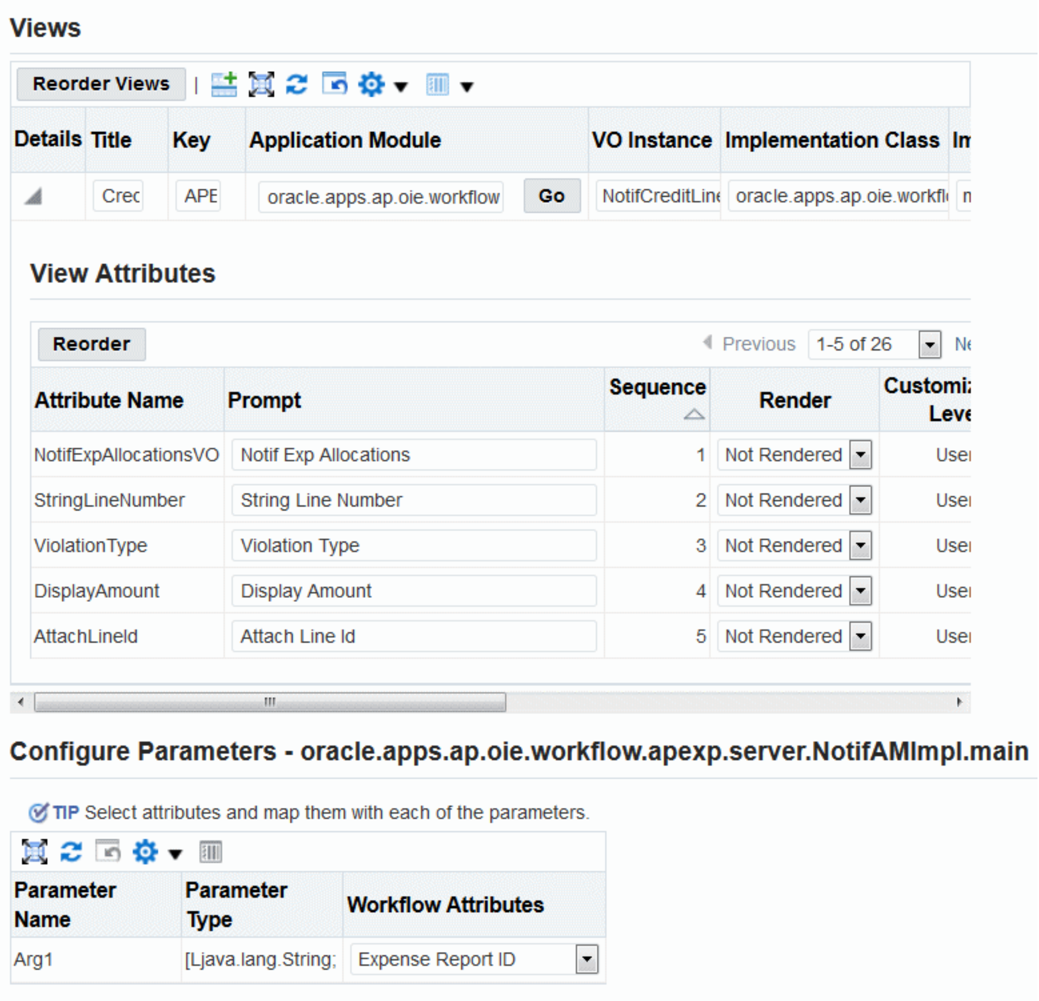
          Note: The parent view attribute must have a rendering status of Passed to have its value passed to the child view. See step 15.
        - Workflow Attributes value type - Select the workflow attribute whose value should be passed to this input parameter.
        - Constant Value value type - Enter a constant value to be used as the value for this input parameter.

14. In the Views list, select the status for the view, either Enabled or Disabled. Only enabled views appear in the Approvals app.
15. Next, expand the details for the view to configure the attributes for the view. The list displays all the attributes associated with the view object.

    The customization level for a view attribute determines what updates you can make.

    - Core - You cannot update any view attribute properties.
    - Limit - You can update the rendering status for the view attribute, but you cannot update any other view attribute properties.
    - User - You can update all display-related view attribute properties.

    Views Page: View Attributes



Views

Reorder Views

Details	Title	Key	Application Module		VO Instance	Implementation Class	Im
	Crec	APE	oracle.apps.ap.oie.workflow	Go	NotifCreditLine	oracle.apps.ap.oie.workfl	n

View Attributes

Reorder ◂ Previous 1-5 of 26 Ne

Attribute Name	Prompt	Sequence	Render	Customi: Leve
NotifExpAllocationsVO	Notif Exp Allocations	1	Not Rendered	Usei
StringLineNumber	String Line Number	2	Not Rendered	Usei
ViolationType	Violation Type	3	Not Rendered	Usei
DisplayAmount	Display Amount	4	Not Rendered	Usei
AttachLineId	Attach Line Id	5	Not Rendered	Usei

Configure Parameters - oracle.apps.ap.oie.workflow.apexp.server.NotifAMImpl.main

TIP Select attributes and map them with each of the parameters.

Parameter Name	Parameter Type	Workflow Attributes
Arg1	[Ljava.lang.String;	Expense Report ID

- Select the Reorder button and use the arrow buttons to move the attributes you want to include in the app to the top of the list. Move those attributes into the order in which you want them to appear in the app.
- Enter the prompt label that is displayed in the app next to the attribute value. By default, the configuration tool sets the prompt label to a value based on the underlying view object attribute's name. Ensure that you update the default value to a user-friendly value that is easy to read and describes the attribute clearly.
- Select the rendering status for the attribute.
  - Rendered - The attribute is displayed in the app.
  - Highlight - The attribute is displayed in the app. Additionally, if this view belongs to a root region and is assigned as the parent of a child region, then this attribute has the drill-down icon displayed next to it to let approvers drill down to the child region.

    You should select the Highlight rendering status only for one attribute in a parent view. Do not select this status for an attribute in a child view.
  - Not Rendered - This attribute is not displayed in the app.
  - Passed - This attribute is not displayed in the app. However, if this view belongs to a root region and is assigned as the parent of a child region, then this attribute is passed to the child region's views. You should define at least one attribute in the parent view to be passed to the child region.

    Do not select this status for an attribute in a child view.

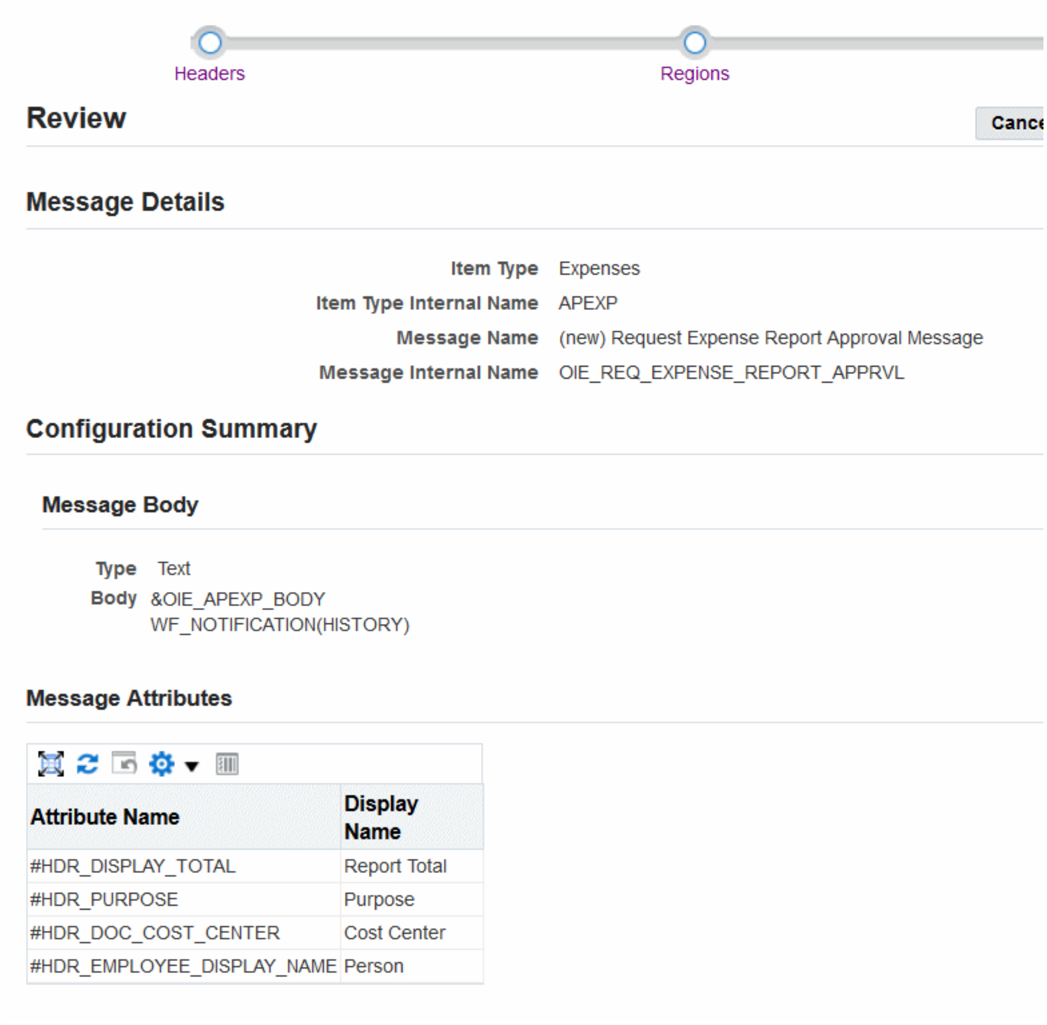
    Note: Only attributes with a rendering status of Passed are passed to a child region. Attributes with a rendering status of Rendered, Highlight, or Not Rendered are not passed.

16. To specify the order in which the views appear in the details screen in the app, select the Reorder Views button. Use the arrow buttons to move the views into the order you want.
17. To delete a view that you no longer require, select the delete icon. You can only delete views with a customization level of User.
18. After you finish defining views, choose Return to navigate back to the Regions page.
19. To delete a region that you no longer require, select the region and select the Delete button. You can only delete regions with a customization level of User.
20. After you finish defining regions, choose Next to navigate to the Review page.

To review and verify a message configuration

1. In the Review page, review the message configuration you defined. You can return to previous pages to update any details you want to change. When you are satisfied with the message configuration, choose Submit to save it.

Review Page: Message Details, Message Body, Message Attributes



Review

Message Details

Item Type	Expenses
Item Type Internal Name	APEXP
Message Name	(new) Request Expense Report Approval Message
Message Internal Name	OIE_REQ_EXPENSE_REPORT_APPRVL

Configuration Summary

Message Body

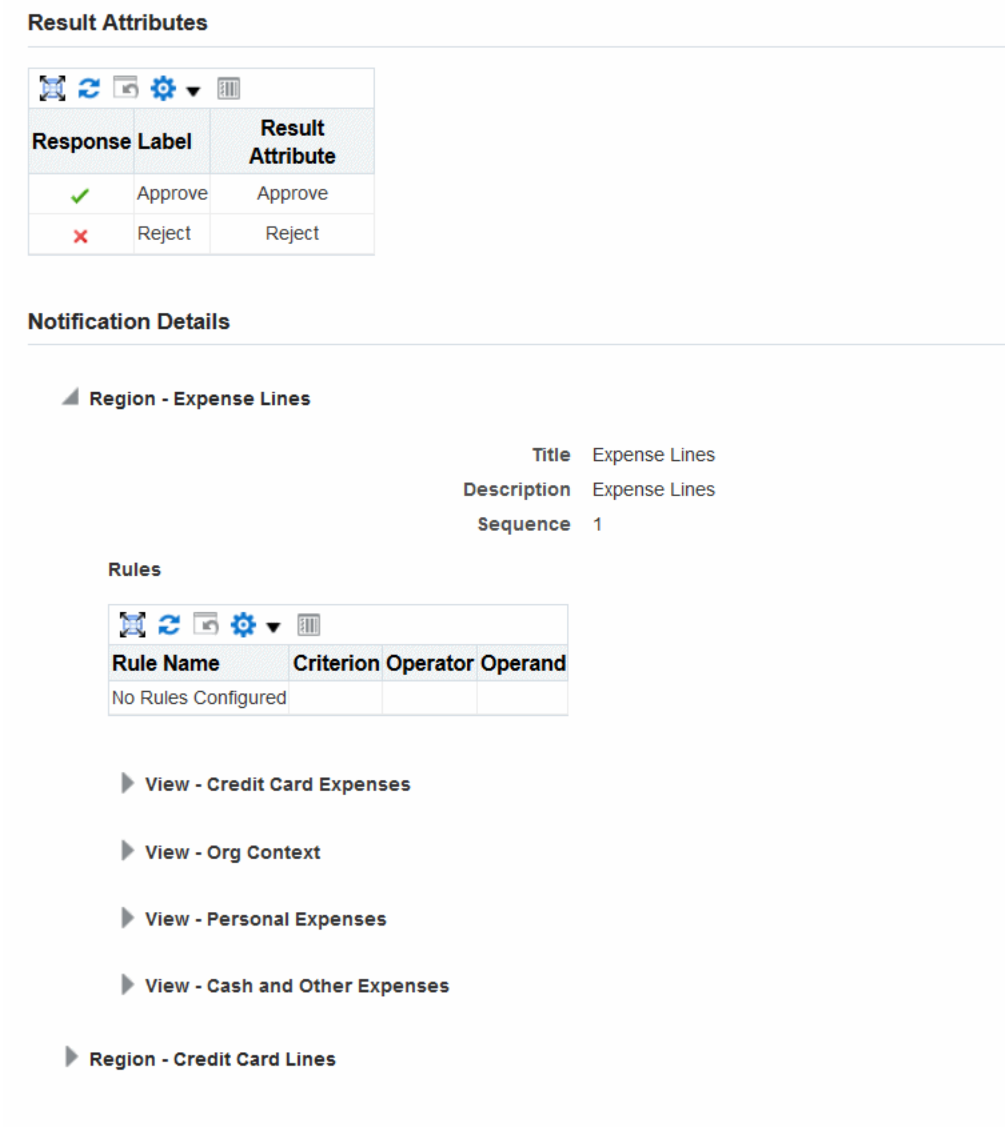
Type Text

Body &OIE_APEXP_BODY
WF_NOTIFICATION(HISTORY)

Message Attributes

Attribute Name	Display Name
#HDR_DISPLAY_TOTAL	Report Total
#HDR_PURPOSE	Purpose
#HDR_DOC_COST_CENTER	Cost Center
#HDR_EMPLOYEE_DISPLAY_NAME	Person

Review Page: Result Attributes, Notification Details



Result Attributes

Response	Label	Result Attribute
✓	Approve	Approve
✗	Reject	Reject

Notification Details

Region - Expense Lines

Title Expense Lines

Description Expense Lines

Sequence 1

Rules

Rule Name	Criterion	Operator	Operand
No Rules Configured			

View - Credit Card Expenses

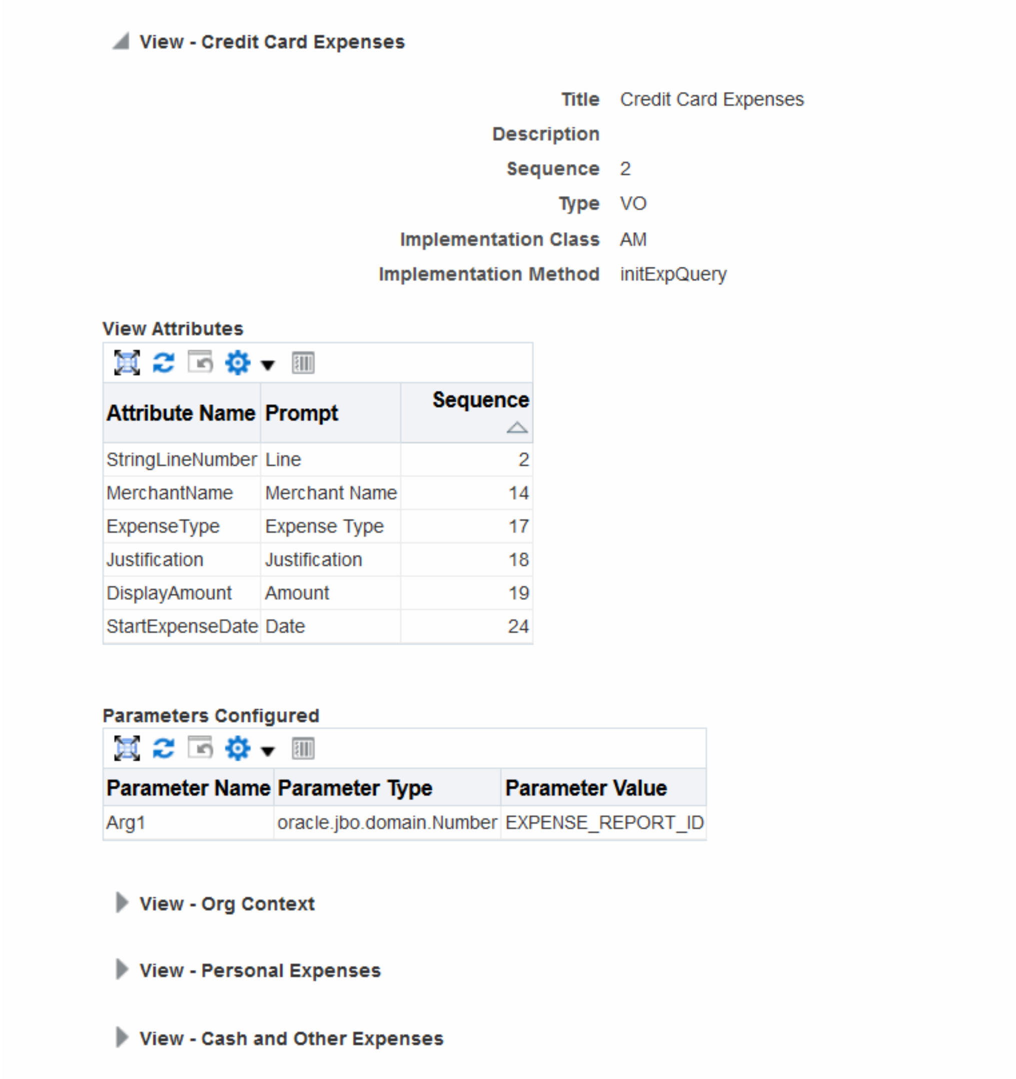
View - Org Context

View - Personal Expenses

View - Cash and Other Expenses

Region - Credit Card Lines

Review Page: Views, View Attributes, Parameters Configured



◢ View - Credit Card Expenses

Title	Credit Card Expenses
Description	
Sequence	2
Type	VO
Implementation Class	AM
Implementation Method	initExpQuery

View Attributes

Attribute Name	Prompt	Sequence
StringLineNumber	Line	2
MerchantName	Merchant Name	14
ExpenseType	Expense Type	17
Justification	Justification	18
DisplayAmount	Amount	19
StartExpenseDate	Date	24

Parameters Configured

Parameter Name	Parameter Type	Parameter Value
Arg1	oracle.jbo.domain.Number	EXPENSE_REPORT_ID

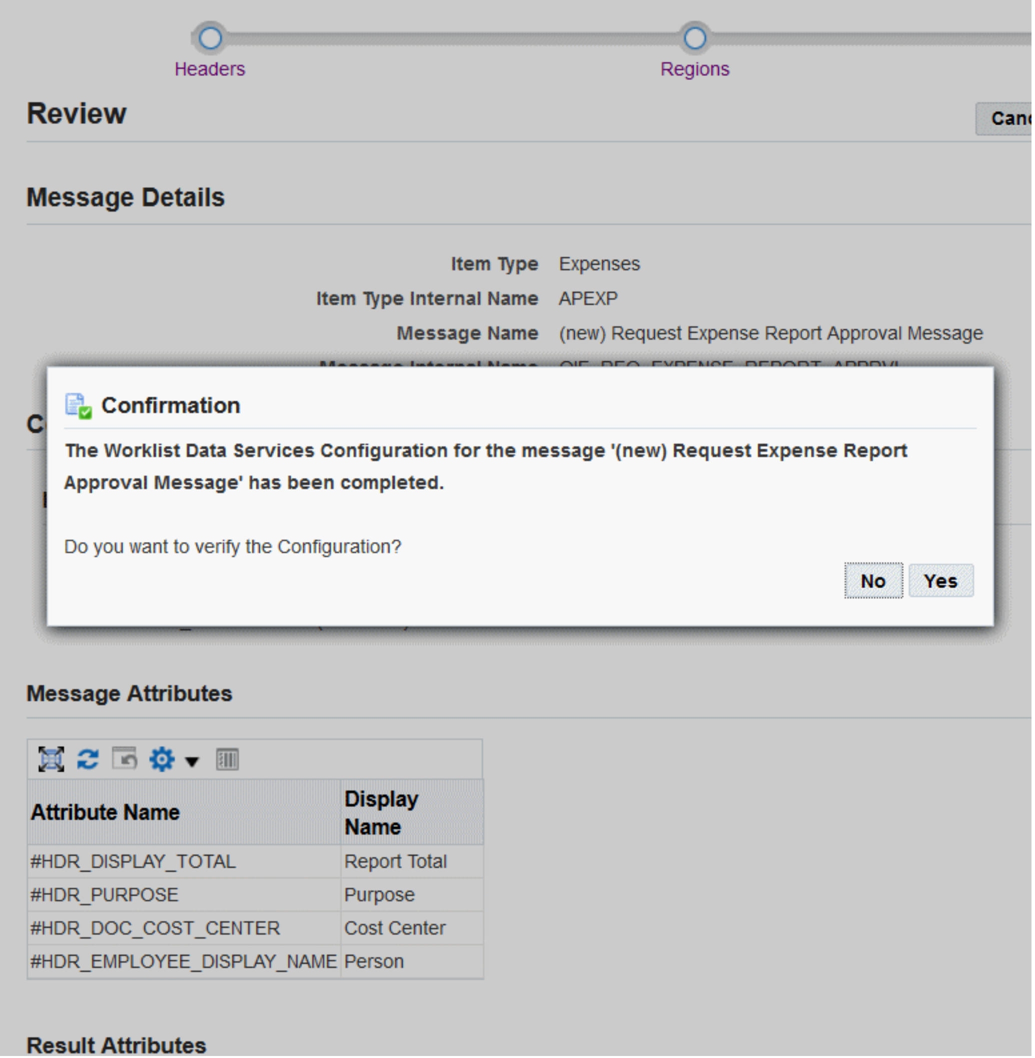
▶ View - Org Context

▶ View - Personal Expenses

▶ View - Cash and Other Expenses

2. After you save a message configuration, the confirmation dialog lets you choose whether to verify the configuration. Choose Yes to proceed to the Verify page.

Review Page: Confirmation Dialog



Review

Message Details

Item Type	Expenses
Item Type Internal Name	APEXP
Message Name	(new) Request Expense Report Approval Message

Confirmation

The Worklist Data Services Configuration for the message '(new) Request Expense Report Approval Message' has been completed.

Do you want to verify the Configuration?

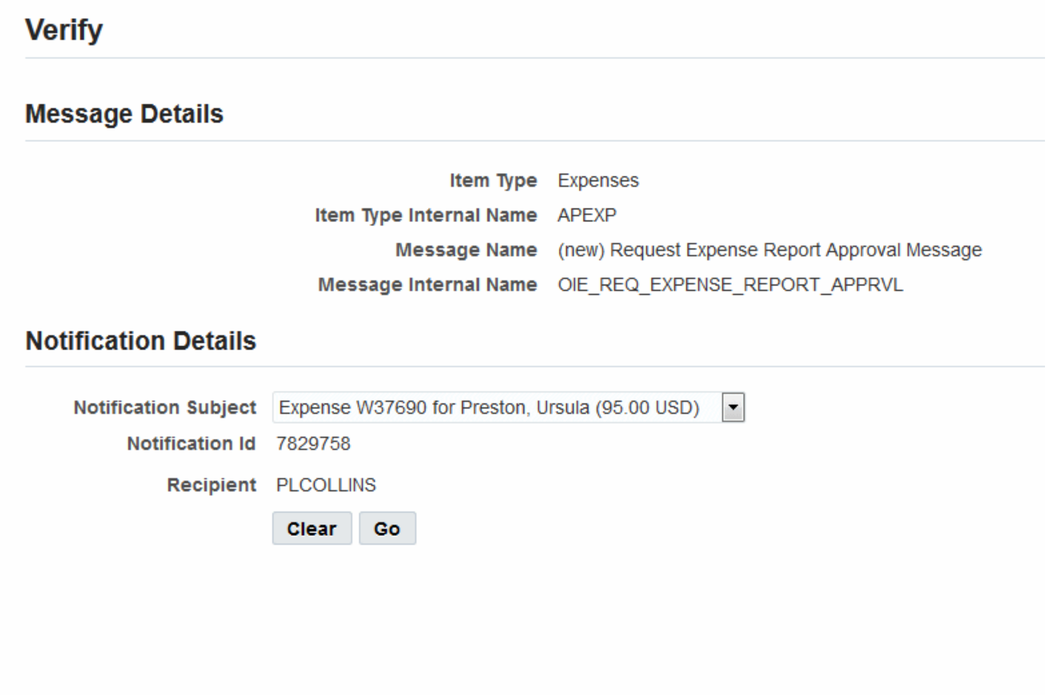
Message Attributes

Attribute Name	Display Name
#HDR_DISPLAY_TOTAL	Report Total
#HDR_PURPOSE	Purpose
#HDR_DOC_COST_CENTER	Cost Center
#HDR_EMPLOYEE_DISPLAY_NAME	Person

Result Attributes

3. The Verify page shows the display name and internal name for the item type to which the message belongs, as well as the display name and internal name for the message. The list of values for the Notification Subject field includes the subject lines of any existing open notifications in your Oracle E-Business Suite instance that were sent using this message. Select the notification you want to use to verify the message configuration. The Verify page displays the notification ID and recipient for the selected notification. Choose Go to display the test output.

Verify Page: Message Details, Notification Details



Verify

Message Details

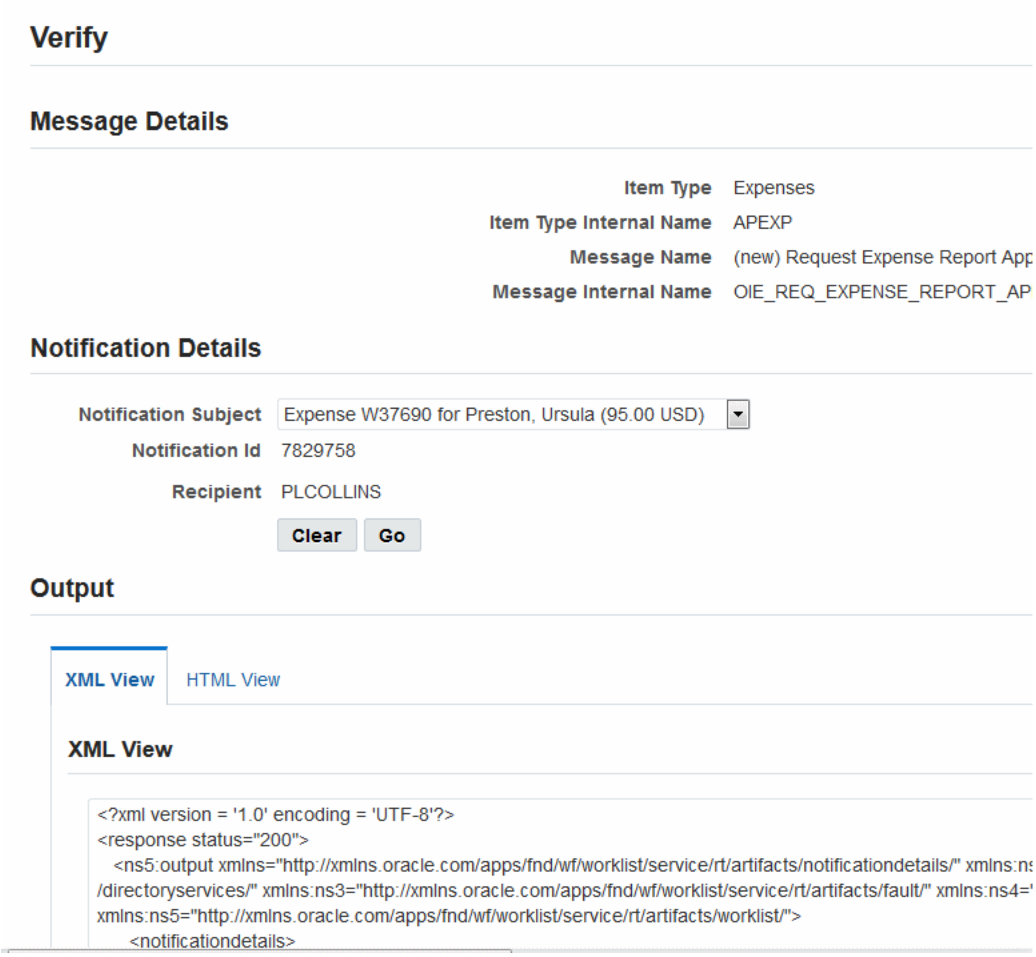
Item Type	Expenses
Item Type Internal Name	APEXP
Message Name	(new) Request Expense Report Approval Message
Message Internal Name	OIE_REQ_EXPENSE_REPORT_APPRVL

Notification Details

Notification Subject	Expense W37690 for Preston, Ursula (95.00 USD)
Notification Id	7829758
Recipient	PLCOLLINS

4. The Output region displays the approval notification content that the approvals data services return for that notification using the message configuration you defined. This output is the same content that is returned to the Approvals app. Review the XML content, and verify that it matches your expectations. You can update the message configuration to adjust the output if necessary.

   Verify Page: Output



Verify

Message Details

Item Type Expenses

Item Type Internal Name APEXP

Message Name (new) Request Expense Report App

Message Internal Name OIE_REQ_EXPENSE_REPORT_AP

Notification Details

Notification Subject Expense W37690 for Preston, Ursula (95.00 USD)

Notification Id 7829758

Recipient PLCOLLINS

Output

XML View

```
<?xml version = '1.0' encoding = 'UTF-8'?>
<response status="200">
  <ns5:output xmlns="http://xmlns.oracle.com/apps/fnd/wf/worklist/service/rt/artifacts/notificationdetails/" xmlns:ns
/directoryservices/" xmlns:ns3="http://xmlns.oracle.com/apps/fnd/wf/worklist/service/rt/artifacts/fault/" xmlns:ns4='
xmlns:ns5="http://xmlns.oracle.com/apps/fnd/wf/worklist/service/rt/artifacts/worklist/">
    <notificationdetails>
```

   Note: The HTML View is reserved for future use.

Defining Approval Types

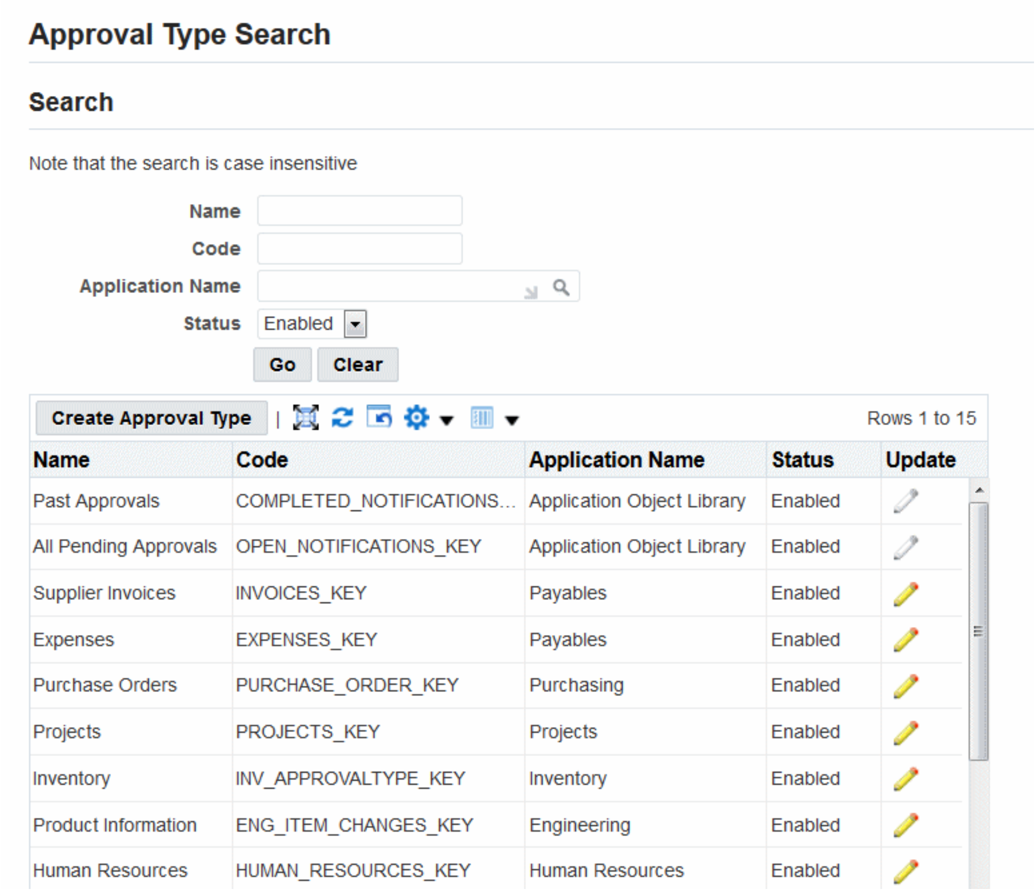
After you complete the configuration for all messages for your application, define approval types to group together the approval notifications that belong to the same business flow so that approvers can easily filter their pending approvals in the app to view these related notifications.

Use the Approval Type Search page to view and manage the approval types defined for your Oracle E-Business Suite installation.

To search for approval types

1. Navigate to the Approval Type Search page from the Oracle E-Business Suite Navigator by choosing either Workflow Administrator Web Applications: Approvals Data Services Manager > Approval Types or a custom responsibility and menu path specified by your system administrator.

   Approval Type Search Page



Approval Type Search

Search

Note that the search is case insensitive

Name

Code

Application Name

Status Enabled

Name	Code	Application Name	Status	Update
Past Approvals	COMPLETED_NOTIFICATIONS...	Application Object Library	Enabled	✎
All Pending Approvals	OPEN_NOTIFICATIONS_KEY	Application Object Library	Enabled	✎
Supplier Invoices	INVOICES_KEY	Payables	Enabled	✎
Expenses	EXPENSES_KEY	Payables	Enabled	✎
Purchase Orders	PURCHASE_ORDER_KEY	Purchasing	Enabled	✎
Projects	PROJECTS_KEY	Projects	Enabled	✎
Inventory	INV_APPROVALTYPE_KEY	Inventory	Enabled	✎
Product Information	ENG_ITEM_CHANGES_KEY	Engineering	Enabled	✎
Human Resources	HUMAN_RESOURCES_KEY	Human Resources	Enabled	✎

2. Search for the approval types you want to display. You can enter the following search criteria:

   - Name - Enter the display name of the approval type. You can enter a partial value to search for types whose names contain that value.
   - Code - Enter the unique internal name of the approval type. You can enter a partial value to search for types whose names contain that value.
   - Application Name - Select the application that owns the approval type.
   - Status - Select whether to display enabled or disabled approval types.

   The search criteria fields are all case-insensitive.

3. To define a new approval type, select the Create Approval Type button. See: To define an approval type.

4. To update an approval type, choose the update icon for that approval type. See: To define an approval type.

   Note: Update only custom approval types.

To define an approval type

1. In the Approval Type Creation page, enter a display name for the approval type. This name appears in the Pending Approvals by Type screen in the Approvals app.

2. In the Code field, enter a unique internal name for the approval type. The internal name can include only uppercase letters from A through Z and the underscore character (_).

3. Enter a brief description for the approval type.

4. Select the parent application to which the approval type belongs.

5. Select the status for the approval type, either `Enabled` or `Disabled`. Only enabled approval types are displayed in the Approvals app.

6. Enter the name of the .png file for the icon that visually identifies the approval type within the Approvals app. The icon must be stored in the physical directory that your Web server's /OA_MEDIA/ virtual directory points to. If you do not specify an icon, or if there is no file matching the specified icon name in the /OA_MEDIA/ directory, then the app displays a default folder icon instead.

7. If required, enter the approvals group value that identifies this approval type. This value must match the value specified in the `#APPROVALS_GROUP` message attribute for the notifications that belong to this approval type. You should enter an approvals group value if you want to expose only some approval notifications sent by a workflow message in the Approvals app, or if the approval notifications sent by a workflow message should belong to different approval types.

   Note: If you define an approvals group value for an approval type, then you must define the `#APPROVALS_GROUP` message attribute for all approval notifications that belong to this approval type and set the attribute value to the specified approvals group value. That is, if you use the `#APPROVALS_GROUP` message attribute for any approval notifications within an approval type, then you must define it for all other messages that belong to that type as well, even if all notifications sent by those messages belong to this same approval type.

If you do not specify an approvals group value for the approval type, then the approval type will include all notifications sent by the messages associated with the type, regardless of whether those notifications have any value specified in an `#APPROVALS_GROUP` message attribute. That is, an approval notification can have its `#APPROVALS_GROUP` message attribute set to a value matching the approvals group value for one approval type, and the same approval notification can also belong to another approval type with no approvals group value specified.

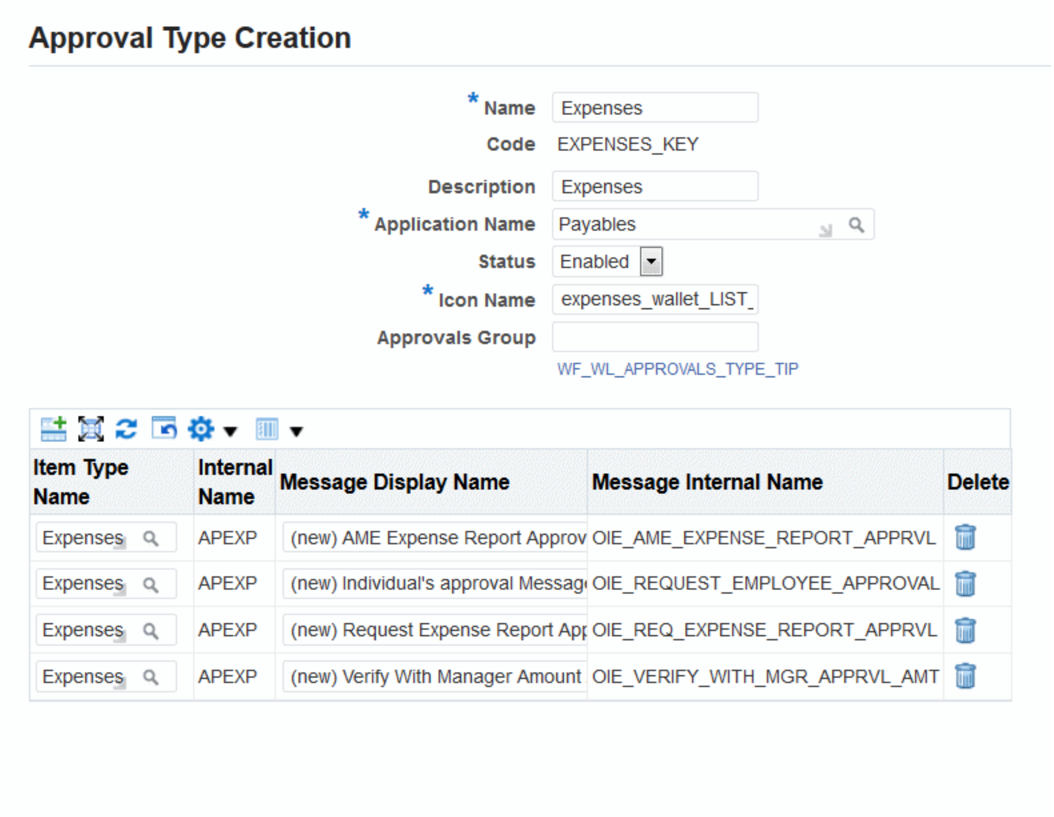
8. In the messages table, add the messages that belong to this approval type. To add a message, perform the following steps:

   - Choose the Add Item Type icon.
   - Select the item type to which the message belongs.
   - Select the message within that item type that you want to include in this approval type within the app.

     Note: Ensure that you select only messages for which you have configured the required metadata to allow the approval notification content to be displayed in the app.

     If you specified an approvals group value for this approval type, then ensure that you have defined the `#APPROVALS_GROUP` message attribute for each message that you add, to identify the approval notifications that belong to this approval type.

Approval Type Creation Page



Approval Type Creation

* Name	Expenses
Code	EXPENSES_KEY
Description	Expenses
* Application Name	Payables
Status	Enabled
* Icon Name	expenses_wallet_LIST_
Approvals Group	
	WF_WL_APPROVALS_TYPE_TIP

Item Type Name	Internal Name	Message Display Name	Message Internal Name	Delete
Expenses	APEXP	(new) AME Expense Report Approv	OIE_AME_EXPENSE_REPORT_APPRVL	🗑
Expenses	APEXP	(new) Individual's approval Messag	OIE_REQUEST_EMPLOYEE_APPROVAL	🗑
Expenses	APEXP	(new) Request Expense Report Ap	OIE_REQ_EXPENSE_REPORT_APPRVL	🗑
Expenses	APEXP	(new) Verify With Manager Amount	OIE_VERIFY_WITH_MGR_APPRVL_AMT	🗑

9. To remove a message from the approval type, choose the delete icon for that message.
10. Choose Apply.

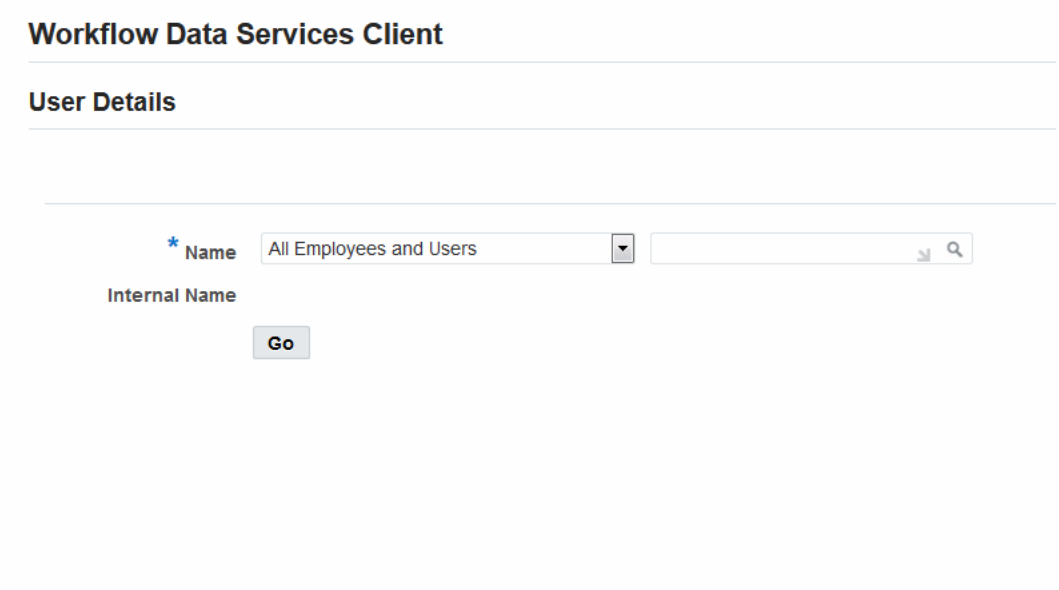
Testing Approvals Data Services with the Test Client

Use the approvals data services test client to verify the approval notification content that the approvals data services return using the metadata you configured.

To test approvals data services

1. Navigate to the Workflow Data Services Client page from the Oracle E-Business Suite Navigator by choosing either Workflow Administrator Web Applications: Approvals Data Services Manager > Test or a custom responsibility and menu path specified by your system administrator.

   Workflow Data Services Client Page: User Details



Workflow Data Services Client

User Details

* Name: All Employees and Users

Internal Name

2. In the User Details region, select the user under whose context you want to execute the approvals data services for this test.
3. In the Service Invocation Details region, select the service you want to test in the Function field.

   You can invoke the approvals data services in the following order to approximate the order in which they are used in the Approvals app. When you invoke the services in this order, you can use information from the output of each service as input for the next service in the list.

   a. getLists

   b. searchByFilter

   c. getFullBody

   d. getDetail

   e. getMoreInfoParticipants

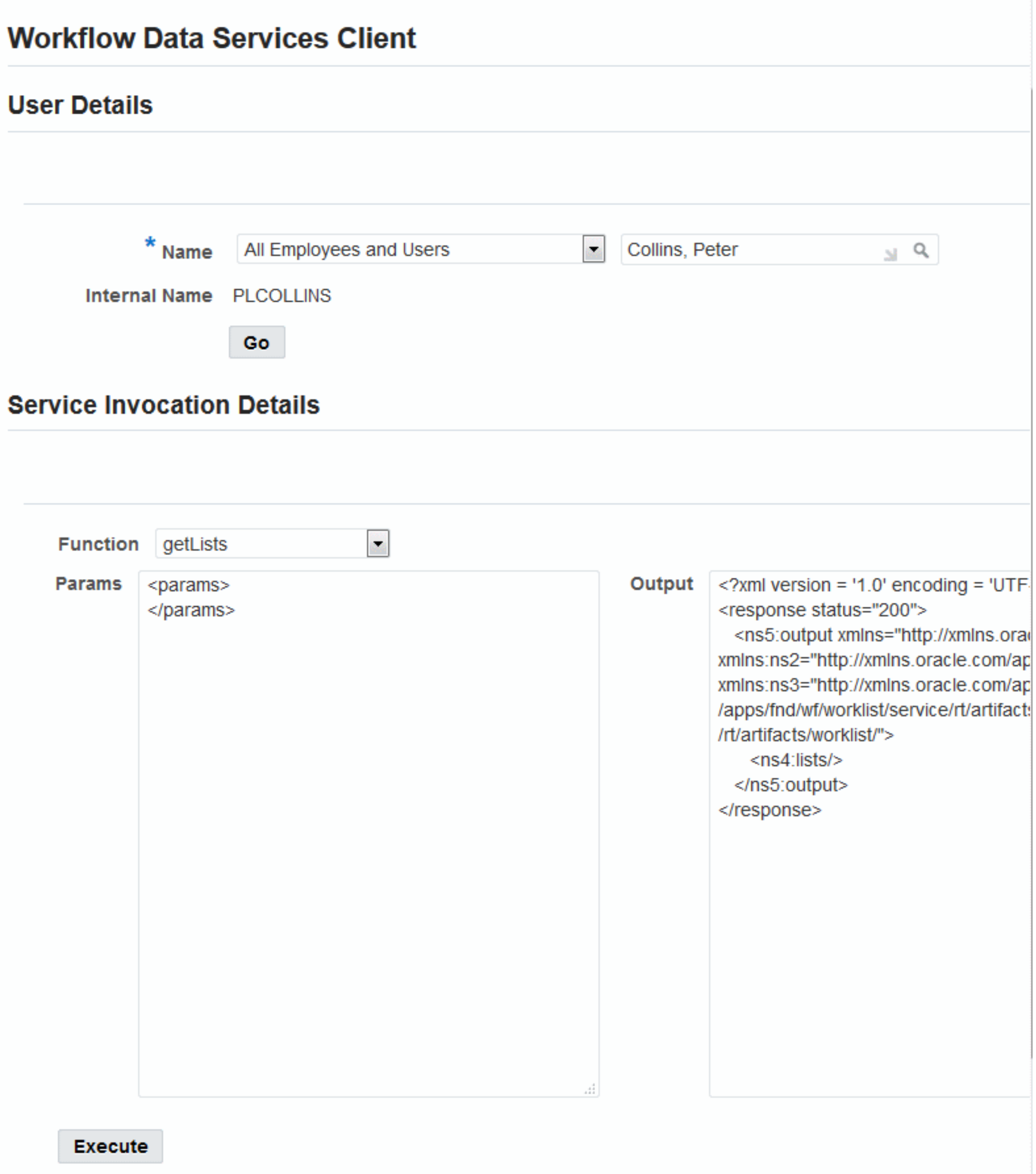
   f. requestMoreInformation

   g. provideMoreInformation

   h. respond
4. The Params field displays the input payload template for the selected service. Enter values for the parameters in the template corresponding to the approval notifications you want to test. Then choose Execute.

   For more information about the input parameters for each service, see: Testing Web Service Requests on the Server, Oracle Mobile Approvals for Oracle E-Business Suite Release Notes, My Oracle Support Knowledge Document 1642423.1.
5. Review the output payload to verify that the service returned the expected response.

   Workflow Data Services Client Page: Service Invocation Details



Workflow Data Services Client

User Details

* Name All Employees and Users Collins, Peter

Internal Name PLCOLLINS

Service Invocation Details

Function getLists

Params

```
<params>
</params>
```

Output

```
<?xml version = '1.0' encoding = 'UTF
<response status="200">
  <ns5:output xmlns="http://xmlns.orac
xmlns:ns2="http://xmlns.oracle.com/ap
xmlns:ns3="http://xmlns.oracle.com/ap
/apps/fnd/wf/worklist/service/rt/artifact
/rt/artifacts/worklist/">
     <ns4:lists/>
  </ns5:output>
</response>
```

Loading Approvals Data Services Configuration Metadata

You can use the Generic Loader (FNDLOAD) concurrent program to download approvals data services configuration metadata, including message configurations and approval type definitions, from an Oracle E-Business Suite instance to text files. You can also use the Generic Loader to upload the metadata from a text file to an Oracle E-Business Suite instance. For example, you can download a message configuration and approval type definition from a development or test instance and upload them to a production instance. You can also download and store text file representations of your message configurations and approval type definitions for source control purposes.

Note: Before downloading the configuration metadata for a message, ensure that you have defined an approval type that includes that message. The Generic Loader does not download any data for a message that does not belong to an approval type.

Downloading Approvals Data Services Configuration Metadata

1. To download message configurations using the Generic Loader, specify the `wfwlsvcs.lct` configuration file, the name of an LDT file in which to save the downloaded data, and the item type to which the messages belong. If you have configured messages in multiple item types, you must download a separate message configuration LDT file for each item type.

   ```
   FNDLOAD apps 0 Y DOWNLOAD $FND_TOP/patch/115/import/wfwlsvcs.lct <LDT_FILE_NAME>.ldt ITEM_TYPE_NAME=<ITEM_TYPE_NAME>
   ORACLE Password:
   ```

   For example:

   ```
   FNDLOAD apps 0 Y DOWNLOAD $FND_TOP/patch/115/import/wfwlsvcs.lct MYITT.ldt ITEM_TYPE_NAME=MYITT
   ORACLE Password:
   ```

2. To download an approval type definition using the Generic Loader, specify the `wfwlsvcs.lct` configuration file, the name of an LDT file in which to save the downloaded data, the approval type code, and the application short name for the application to which the approval type belongs.

   ```
   FNDLOAD apps 0 Y DOWNLOAD $FND_TOP/patch/115/import/wfwlsvcs.lct <LDT_FILE_NAME>.ldt WF_WL_LISTS LIST_KEY=<APPROVAL_TYPE_CODE> APPL_SHORT_NAME=<A
   ORACLE Password:
   ```

   For example:

```
    FNDLOAD apps 0 Y DOWNLOAD $FND_TOP/patch/115/import/wfwlsvcs.lct MYITT_APP_TYPE.ldt WF_WL_LISTS LIST_KEY=APP_TYPE_KEY APPL_SHORT_NAME=XXX
    ORACLE Password:
```

Uploading Approvals Data Services Configuration Metadata

To upload either message configurations or approval type definitions using the Generic Loader, specify the `wfwlsvcs.lct` configuration file, the name of the LDT file that contains the metadata to upload, and `-` to upload the entire contents of the LDT file.

```
FNDLOAD apps 0 Y UPLOAD $FND_TOP/patch/115/import/wfwlsvcs.lct <LDT_FILE_NAME>.ldt -
ORACLE Password:
```

For example:

```
FNDLOAD apps 0 Y UPLOAD $FND_TOP/patch/115/import/wfwlsvcs.lct MYITT.ldt -
ORACLE Password:
```

Transferring Custom Approvals Data Services Configurations between Oracle E-Business Suite Instances

When you transfer custom approvals data services configurations between Oracle E-Business Suite instances, such as from a development or test instance to a production instance, ensure that you apply all the related files to the target instance, including the following:

- Message configuration LDT files
- Approval type definition LDT files
- The .png files for the icons associated with the approval types
- Any workflow definition files (.wft files) that you updated to prepare the messages for Approvals configuration
- Any application module and view object implementations that you created or updated to prepare the associated messages for Approvals configuration
- Any .pls files containing PL/SQL or PL/SQL CLOB APIs that you created or updated to prepare the associated messages for Approvals configuration

To ensure that your custom configurations are handled appropriately in conjunction with the Online Patching feature introduced in Release 12.2, follow the instructions in *Developing and Deploying Customizations in Oracle E-Business Suite Release 12.2*, My Oracle Support Document 1577661.1.

Related Topics

Generic Loader, *Oracle E-Business Suite Setup Guide*

Contents | Previous | Top of Page | Next