ACTIVE Governance™

ACTIVE Governance/ Identity Manager Integration Kit

Software Version 7.2

LogicalApps®

Restricted Rights Legend

LogicalApps Provides on-site support as well as remote phone and web support to ensure quick and effective product implementation. To request support, to suggest product enhancements, or to comment on LogicalApps software or documentation, send email to support@logicalapps.com, or contact us at the address or phone number given below.

Document Version AG012-720A

8/29/07

LogicalApps
15420 Laguna Canyon, Suite 150
Irvine, CA 92618
949.453.9101

Contents

Chapter 1

Integration Overview

Sun Identity Manager consolidates user administration in ERP systems (and other resources). It establishes a single account for each user and, through that account, defines the rights the user has in any number of systems or resources used by a company.

ACTIVE Governance from LogicalApps includes ACTIVE Access Governor™, which enforces segregation-of-duties (SOD) rules in Oracle ERP systems.

The ACTIVE Governance/Identity Manager Integration Kit from LogicalApps enables Access Governor to enforce SOD rules when administrators use Identity Manager to assign responsibilities to users of Oracle ERP systems.

In Identity Manager, three workflows — CREATE_sp_USER, UPDATE_sp_USER, and APPROVAL — route the tasks that must be completed when administrators create or modify user accounts. The ACTIVE Governance/Identity Manager Integration Kit adds LogicalApps plug-in processes to the Identity Manager workflows. There are two possible implementations:

- If a company has not modified the default Identity Manager workflows, then integration with the Access Governor SOD capability involves installing workflows that overwrite the defaults. The new workflows contain the LogicalApps plug-in processes, but are otherwise identical to the default Identity Manager workflows.
- If a company has customized its Identity Manager workflows, it must edit those workflows to insert the LogicalApps plug-in processes.

Chapter 2

Performing the Integration

To integrate ACTIVE Governance with Identity Manager, you can either replace .xml files that implement default Identity Manager workflows with LogicalApps versions of the files, or you can edit customized workflows in the Identity Manager Business Process Editor to add LogicalApps "activities" to them. In either case, you also install some necessary files, then complete some configuration steps from within Identity Manager.

In either case, it's assumed that Identity Manager is installed at *WSHOME*\webbapps\idm (where *WSHOME* is the path to the home directory for your web server). You'll need to know the host name and port number of the server on which ACTIVE Governance is installed, the URL and port number your site uses to connect to Oracle ERP, and the Oracle database user name and password.

Integration with Default Workflows

To begin integrating ACTIVE Governance with an instance of Identity Manager that uses default workflows, complete these steps:

1. Stop Identity Manager by stopping the web server on which it runs.
2. Create an LA_IDM_InstallationSetup folder. In it, download the file LA_IDM_InstallationSetup.zip from a site provided by LogicalApps, and extract its contents.

3. Create the folder *WSHOME*\webapps\idm\WEB-INF\bkp.
4. Copy the following files from LA_IDM_InstallationSetup\idmXMLs\configuration to *WSHOME*\webapps\idm\WEB-INF\object\Configuration:
   - LOGICALAPPS_sp_SOD_sp_ANALYSIS_sp_SUB-PROCESS.xml
   - LOGICALAPPS_sp_POST_sp_APPROVAL_sl_REJECT_sp_SUB-PROCESS.xml
5. Back up Identity Manager versions of the following three files. Copy them to *WSHOME*\webapps\idm\WEB-INF\bkp:
   - *WSHOME*\webapps\idm\WEB-INF\object\configuration\APPROVAL.xml
   - *WSHOME*\webapps\idm\WEB-INF\object\TaskDefinition\UPDATE_sp_USER.xml
   - *WSHOME*\webapps\idm\WEB-INF\object\ProvisioningTask\CREATE_sp_USER.xml
6. Copy LogicalApps versions of these files to the directories from which you backed up the Identity Manager versions. The LogicalApps versions are located in the following subdirectories of LA_IDM_InstallationSetup\idmXMLs:
   - .\configuration\APPROVAL.xml
   - .\TaskDefinition\UPDATE_sp_USER.xml
   - .\ProvisioningTask\CREATE_sp_USER.xml
7. For each pair of files from steps 5 and 6, open the backup Identity Manager version and copy its header information, which includes everything that comes before the <Extension> tag. Then open the LogicalApps version of the file; in it, replace the header information with the header copied from the backup Identity Manager version.
8. Replace an Identity Manager version of a file called workflowRegistry.xml with a LogicalApps version. To do so, copy workflowRegistry.xml from LA_IDM_InstallationSetup\idmXMLs\config to *WSHOME*\webapps\idm\config
9. Navigate to LA_IDM_InstallationSetup\classes. In that folder, edit the file idm.properties. Replace the value *MAC_NAME.whq.logicalapps.com* with the host name of the server on which your ACTIVE Governance instance is installed. Replace the value *PORT* with the port number (typically 8080) used by ACTIVE Governance.
10. Copy the LA_IDM_InstallationSetup\classes folder to *WSHOME*\webapps\idm\WEB-INF
11. Copy all the .jar files from LA_IDM_InstallationSetup\lib to *WSHOME*\webapps\idm\WEB-INF\lib

To continue, skip ahead to "Completing the Integration" (page 7).

Integration with Customized Workflows

To integrate ACTIVE Governance with an instance of Identity Manager that uses already-customized workflows, begin by copying subprocesses that will be assigned to new activities in two of the workflows. Then use the Business Process Editor, an Identity Manager tool, to modify existing workflows.

- For each of CREATE_sp_USER and UPDATE_sp_USER, you create three activities, associate LogicalApps subprocesses with two of them, and configure transitions between the activities you've created and other activities that already exist in your workflows. (For the CREATE and UPDATE workflows, the configuration steps are identical.)
- In the existing APPROVAL subprocess, you edit a variable.

To copy the LogicalApps subprocesses, complete these steps:

1. Stop Identity Manager by stopping the web server on which it runs.
2. Create an LA_IDM_InstallationSetup folder. In it, download the file LA_IDM_InstallationSetup.zip from a site provided by LogicalApps, and extract its contents.
3. Copy the following files from LA_IDM_InstallationSetup\idmXMLs\configuration to *WSHOME*\webapps\idm\WEB-INF\object\Configuration:
   - LOGICALAPPS_sp_SOD_sp_ANALYSIS_sp_SUB-PROCESS.xml
   - LOGICALAPPS_sp_POST_sp_APPROVAL_sl_REJECT_sp_SUB-PROCESS.xml

To associate these subprocesses with your workflows, complete the following steps. You are presumed to know how to use the Identity Manager Business Process Editor. If you need information about it, see Sun Identity Manager documentation.

1. Start the Identity Manager Business Process Editor. In it, open the CREATE_sp_USER workflow.
2. Create three activities with the following names:
   - LogicalApps SOD Analysis Plugin
   - LogicalApps Post Approval/Reject Plugin
   - divert
3. Open the Start activity. Create a transition from it to the LogicalApps SOD Analysis Plugin activity; position it first in the list of transitions. Then create the following condition for this transition:

```
<gt>
  <length>
    <ref>accounts</ref>
  </length>
  <i>1</i>
</gt>
```

4 Open the LogicalApps SOD Analysis Plugin activity. Link it to *WSHOME*\ webapps\idm\WEB-INF\object\Configuration\LOGICALAPPS_sp_ SOD_sp_ ANALYSIS_sp_SUB-PROCESS.xml.

5 Create three transitions from the LogicalApps SOD Analysis Plugin activity to other activities, in the order shown in the following table. For each, create the condition shown in the following table:

	Transition To	Condition

1	End	`<eq>`
		`  <ref>laError</ref>`
		`  <s>true</s>`
		`<eq>`

2	End	`<eq>`
		`  <ref>laApproval</ref>`
		`  <s>false</s>`
		`</eq>`

3	LogicalApps Post Approval/Reject Plugin	`<eq>`
		`  <ref>approved</ref>`
		`  <s>false</s>`
		`</eq>`

6 Open the LogicalApps Post Approval/Reject Plugin activity and link it to *WSHOME*\ webapps\idm\WEB-INF\object\Configuration\LOGICALAPPS_ sp_POST_ sp_APPROVAL_sl_REJECT_sp_SUB-PROCESS.xml.

From the LogicalApps Post Approval/Reject Plugin activity, create a single transition to the End activity. The transition should have no condition.

7 Open the divert activity. Create two transitions from it to other activities, in the order shown in the following table. For the first, create the condition shown in the following table. (Note: there is no subprocess associated with the divert activity.)

	Transition To	Condition

1	LogicalApps Post Approval/Reject Plugin	`<gt>`
		`  <length>`
		`  <ref>accounts</ref>`
		`  </length>`
		`  <i>1</i>`
		`<gt>`

2	End	None

8 Identify activities in the workflow which, prior to this integration, had transitions to the End activity. Open each activity and reroute its End transitions to the divert activity.

9 Save and close the CREATE_sp_USER workflow. Open the UPDATE_sp_USER workflow, repeat steps 2–7 in it, and save and close UPDATE_sp_USER.

10 Under Workflow Subprocesses, open APPROVAL. In the list of variables, locate Description, and, in its Variable field, enter the following:

```
$(approvalDescription)
```

Save and close the workflow subprocess.

11 Open the file *WSHOME*\webapps\idm\config\workflowRegistry.xml and add the following at the bottom of the file:

```
<WorkflowApplication name='LA_APPLICATION' class='com.logicalapps.
integration.idm.sun.LaSunIDMIntegrationApplicationImp'>
  <Comments>
    Removes a deferred task definition from an object.
  </Comments>

  <ArgumentDefinition name='user'>
    <Comments>
      Accounts Information
    </Comments>
  </ArgumentDefinition>

</WorkflowApplication>

<WorkflowApplication name='LA_ACK_APPLICATION' class='com.
logicalapps.integration.idm.sun.LaSunIDMIntegrationApproverAck'>
  <Comments>
    Removes a deferred task definition from an object.
  </Comments>

  <ArgumentDefinition name='user'>
    <Comments>
      Accounts Information
    </Comments>
  </Argument Definition>

</WorkflowApplication>
```

Next, prepare and install some additional files:

1 Navigate to LA_IDM_InstallationSetup\classes. In that folder, edit the file idm.properties. Replace the value *MAC_NAME.whq.logicalapps.com* with the host name of the server on which your ACTIVE Governance instance is installed. Replace the value *PORT* with the port number (typically 8080) used by ACTIVE Governance.

2 Copy the LA_IDM_InstallationSetup\classes folder to *WSHOME*\webapps\idm\WEB-INF

3 Copy all the .jar files from LA_IDM_InstallationSetup\lib to *WSHOME*\webapps\idm\WEB-INF\lib

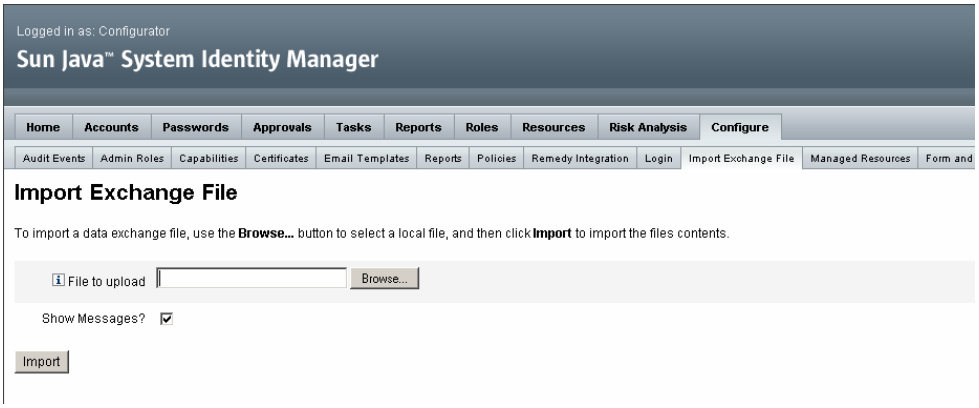
To continue, see the next section, "Completing the Integration."

Completing the Integration

No matter whether you are integrating ACTIVE Governance with an Identity Manager instance that uses default or customized workflows, complete the integration by performing the following configuration within Identity Manager:

1 Start your web server and log on to Identity Manager. Use *configurator* as both the user ID and the password.

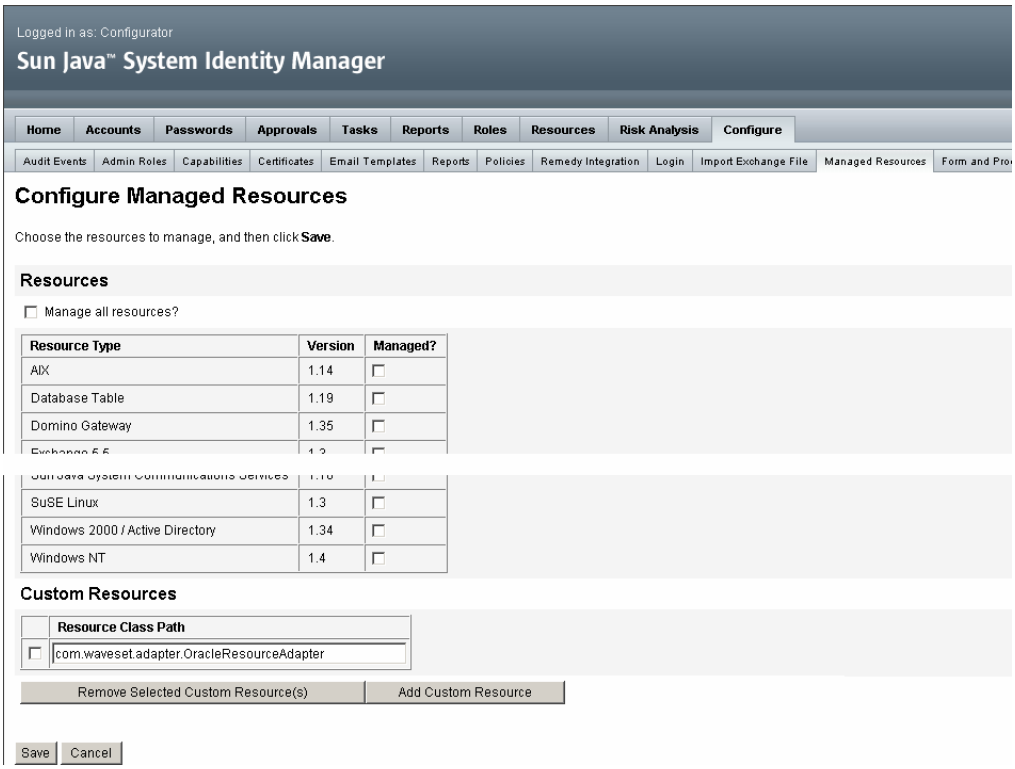
2 Click on the Configure tab, and then on the Import Exchange File subordinate tab.



3 Click on the Browse button. In a Choose File dialog, navigate to LA_IDM_InstallationSetup\idmXMLs\oracleERP\LA-OracleERPUserForm.xml and click on the Open button. The file name and path appear in the File to Upload field. Click on the Import button.

4 Click on the Browse button again. In the Choose File dialog, navigate to LA_IDM_InstallationSetup\idmXMLs\user form\LA-TabbedUserForm.xml and click on the Open button. The file name and path appear in the File to Upload field. Click on the Import button.

5 Click on the Managed Resources subordinate tab.



6 Beneath the Custom Resources heading, click on the Add Custom Resource button. An empty field appears beneath the Custom Resources heading. In it, type the following entry:

```
com.waveset.adapter.OracleResourceAdapter
```

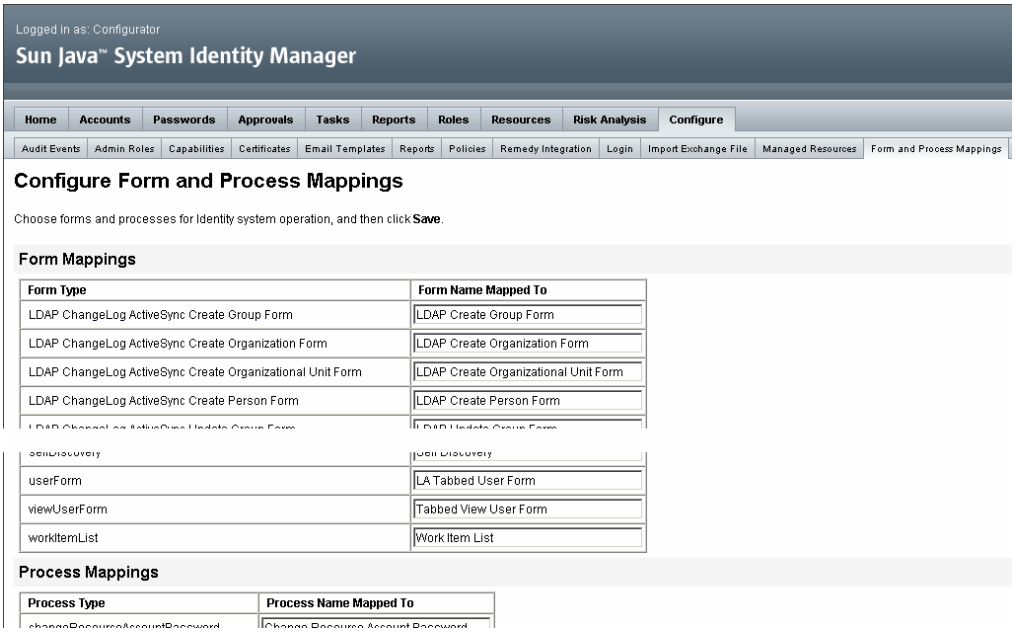
7 Click on the Add Custom Resource button again. In a second empty field, type the following entry:

```
com.waveset.adapter.OracleERPResourceAdapter
```

8 Click on the Save button.

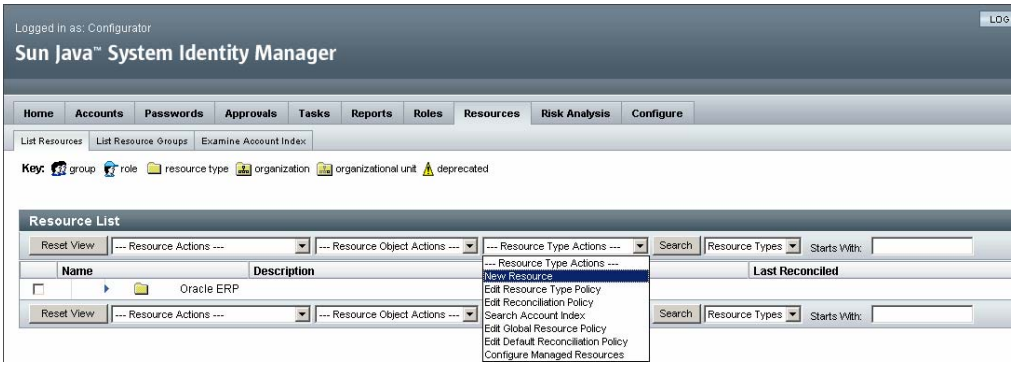
9 Click on the Form and Process Mappings subordinate tab.

10 In the Form Type column of the Form Mappings grid, locate the entry for userForm. In the corresponding field of the Form Name Mapped To column, change the entry to *LA Tabbed User Form*. Then click on the Save button.

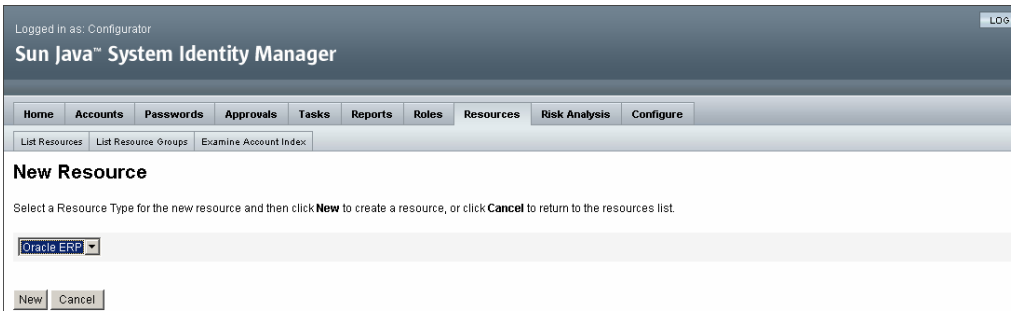


11 Click on the Resources tab.

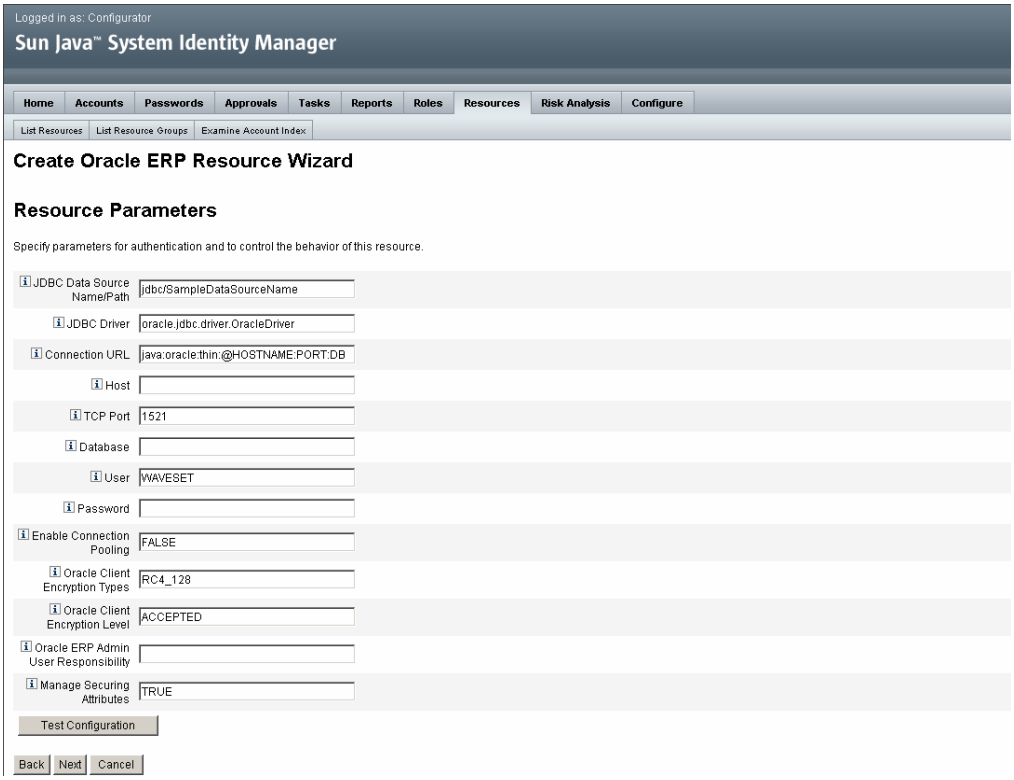
12 In the third field from the left under Resource List (it's labeled Resource Type Actions by default), select the value *New Resource*.



13 A New Resource form opens. Select Oracle ERP in its list box, and then click on its New button.



14 A Create Oracle ERP Resource Wizard begins to run. Click on its Next button. A Resource Parameters form opens:

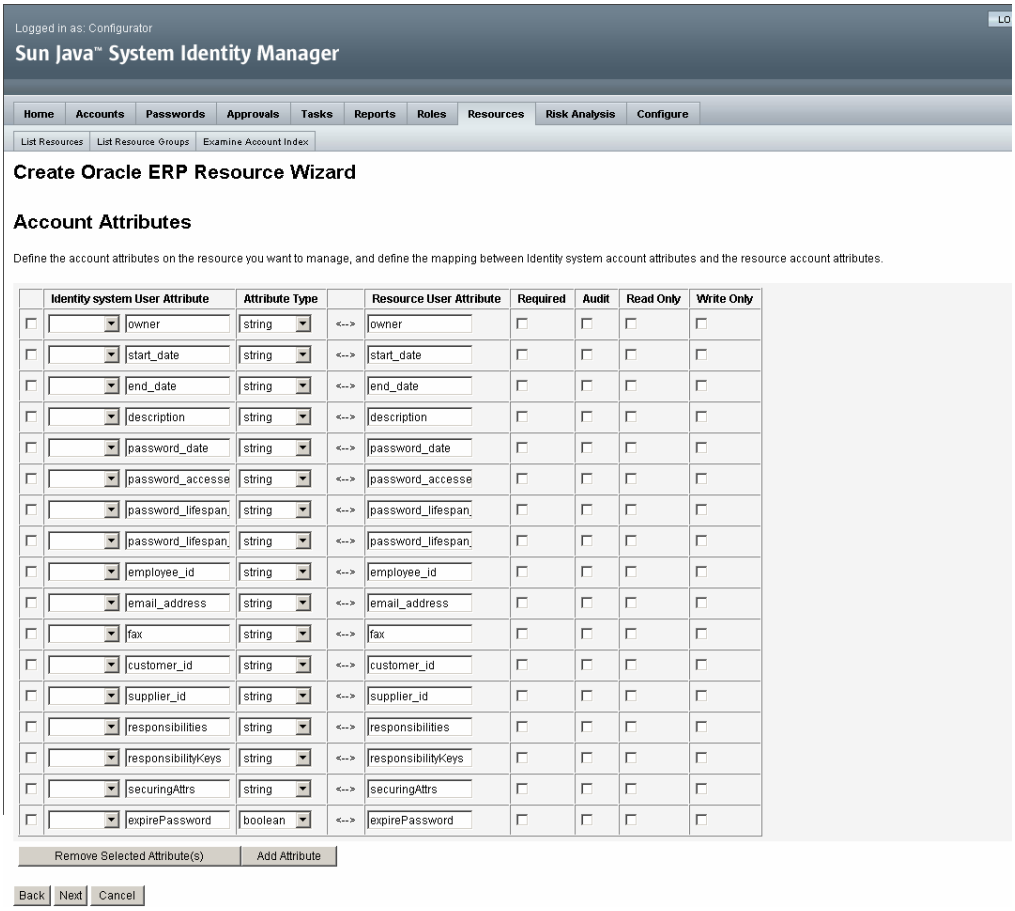


15 Edit values in the following fields. Each must match a corresponding value that has been set for an Oracle EBusiness Suite instance configured as a data source for ACTIVE Governance. (These values can be seen in the View Data Source panel of the ACTIVE Governance Platform.) Leave other fields at their default values.

- Connection URL: Enter the URL your site uses to log on to an Oracle ERP instance (for example: java:oracle:thin:@aspen:1532:vis11512)
- TCP Port: Enter the port number used by the Oracle instance.
- User: Enter the username for your Oracle database schema (for example, APPS).
- Password: Enter the password configured for the Oracle database user.

16 Click on the Test Configuration button to ensure that Identity Manager can "talk to" the Oracle resource. A green message at the top of the form indicates success; a red message reports an error.

17 Click on the Next button. An Account Attributes form appears:



If you use Oracle ERP version 11.5.9, accept the default values.

If you use Oracle ERP version 11.5.10, make these modifications:

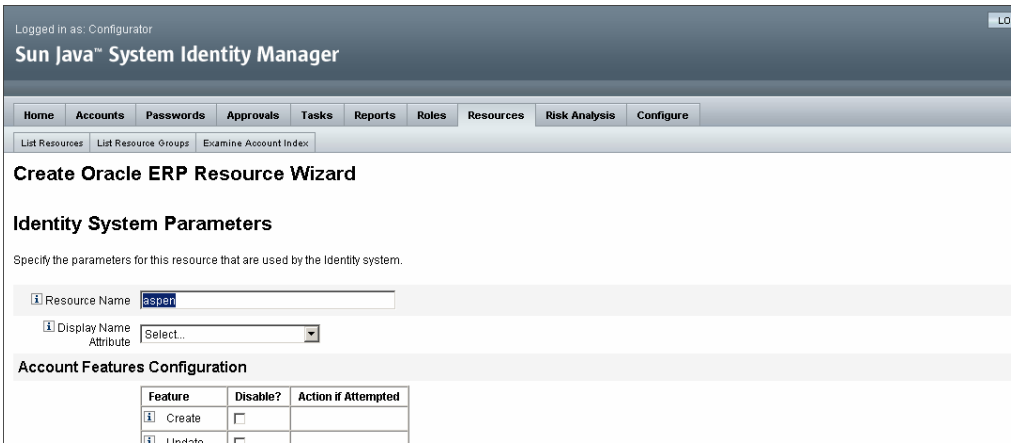
- Remove the *responsibilities* entry. Click on its check box, and then click on the Remove Selected Attribute(s) button.
- Add two new entries — *directResponsibilities* and *indirectResponsibilities*. For each, click on the Add Attribute button, and a new row appears in the grid. Type the attribute name in both the second field of the Identify System User Attribute column and in the Resource User Attribute column. Be sure the Attribute Type is set to *string*. Leave the other field and check boxes blank.

Then click on the Next button.

18 In an Identity Template form, accept the default and click on the Next button.

19 In an Identity System Parameters form, replace the default value in the Resource Name field with the name configured for the Oracle instance whose URL, port,

user, and password you supplied in step 15. Accept default values for the other fields, and click on the Save button.



Chapter 3

Using the Integrated System

To understand how the configuration of users in Identity Manager is affected by its integration with ACTIVE Governance, you need to know a bit about how ACTIVE Access Governor works. (For complete information, see the *ACTIVE Access Governor User's Guide*.)

Users of Access Governor create segregation-of-duties rules, each of which specifies two or more Oracle responsibilities or functions that should not be assigned simultaneously to an individual person. Each SOD rule applies one of four "control types":

- A Prevent rule denies access to conflicting responsibilities or functions.
- An Allow with Rules SOD rule permits access to conflicting responsibilities or functions if one or more additional rules, written in an application called Form Rules, mitigate the conflict by modifying Oracle forms. This control type (like the Prevent type) requires no approval, and does not send approval requests.
- An Approve with Rules SOD rule also permits access to conflicting responsibilities or functions on condition that one or more Form Rules mitigate the conflict. In this case, however, the SOD rule designates an "owner," and access to the conflicting entities is granted only if the owner approves.
- An Approval Required rule also designates an owner who can either approve a conflict or reject it. In this case, no Form Rule is attached to the SOD rule, and for access to be granted to conflicting responsibilities or functions, no condition need be met other than that the owner approve the conflict.

When SOD rules are satisfactorily configured, an Access Governor user "generates conflicts" — causes Access Governor to note Oracle ERP users whose work assignments violate the rules. If Oracle users have been assigned responsibilities or functions before SOD rules were created to define them as conflicting, Access Governor records the conflicts, but administrators would need to resolve them manually within the Oracle ERP instance.

After conflicts have been generated, the SOD rules remain in force. Thus when a user is created or updated in Identity Manager, and assigned responsibilities in an Oracle ERP instance, the rules are evaluated, with the following results:

- If a responsibility assignment violates one or more Prevent rules, a new user is not created, or an existing user is not updated. Even responsibilities uninvolved with Prevent-rule conflicts are withheld. No approval requests are distributed.
- If a responsibility assignment violates no Prevent rules, but triggers one or more Approve with Rules or Approval Required rules, Identity Manager sends approval requests to the rule owners. All must approve for access to be granted. If even one owner rejects, a new user is not created, or an existing user is not updated; even responsibilities other than the rejected one are withheld.
- If only Allow with Rules SOD rules are triggered, or if no SOD rules are triggered, a new user is created or an existing user is updated. No approval requests are distributed.

Note, however, that you may have configured approval conditions in Identity Manager that have nothing to do with segregation of duties. If so, these conditions may distribute approval requests regardless of the outcome of segregation-of-duties analysis.

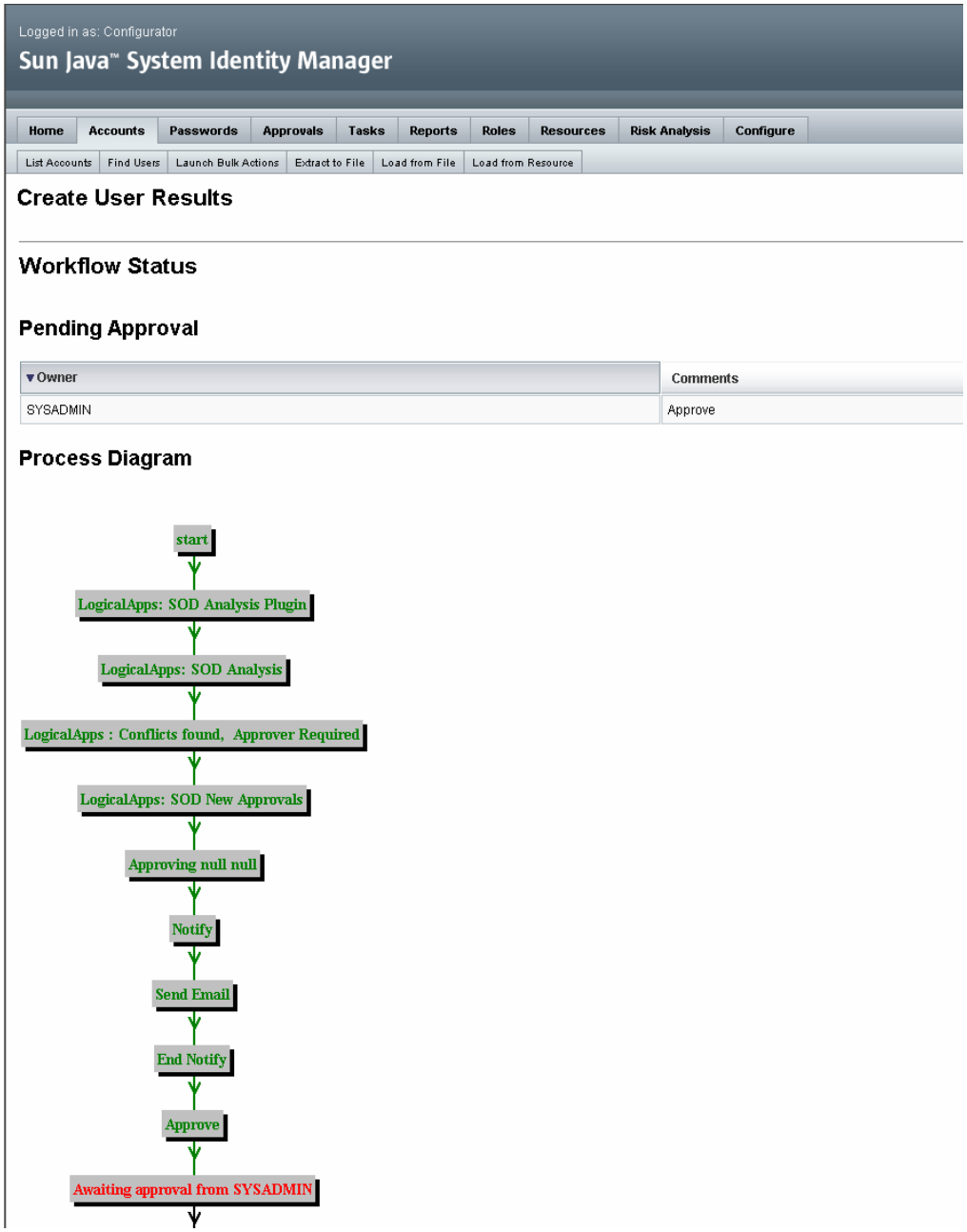
Configuring Users

When ACTIVE Governance is integrated with Identity Manager, you would configure users essentially as you would if you were using Identity Manager on its own. For complete instructions, see Identity Manager documentation. Steps typically include the following:

- In Identity Manager, select the Accounts tab. Then either select New User in the New Actions list box, or click on a user's account ID in the Name list.
- In the Create User or Edit User panel, ensure that the Identity tab is selected. Then enter or edit an account ID and the user's name, email address, and password, and select an organization.
- Click on the Assignments tab and select Oracle ERP instances for the user: In the Individual Resource Assignments area, move any number of Oracle data sources from the Available Resources field to the Current Resources field.
- Click on the Attributes tab and select the Add Direct Responsibilities check box. For each responsibility you want to assign, make selections in the Oracle ERP Responsibilities, Oracle ERP Applications, and Oracle ERP Security Groups fields and click on the Add Direct Responsibility button.

For the last of these steps, the integration of Identity Manager with Access Governor alters the procedure somewhat. When Identity Manager is used on its own and when more than one Oracle ERP instance has been selected for a user, the Attributes tab presents a block of fields that applies to all Oracle ERP instances. When Identity Manager is integrated with Access Governor, the Attributes tab presents duplicate blocks of fields, one for each Oracle ERP instance. A user can therefore select distinct sets of responsibilities in each instance, and SOD rules are evaluated in each instance.

When you save your selections, Identity Manager presents a process diagram showing that LogicalApps SOD analysis has taken place; if approval requests have been generated, it shows records of those requests as well:



It is assumed that when you implement Identity Manager, you will run its synchronization utility so that pre-existing Oracle users will be created as Identity Manager users. If not, using Identity Manager to assign additional responsibilities to a user that exists in Oracle will cause that user to lose the responsibilities he had already been assigned.

In order for approvals to take place, every SOD rule owner must be created as a user in Identity Manager. If you've set up Identity Manager so that approvers are subject to any specialized configuration, you must ensure that appropriate configuration steps are completed for users who are SOD rule owners. Moreover, owners must be able to use Access Governor to set status for conflicts that occur because users had been assigned responsibilities or functions before rules were created to define them as conflicting. So it's assumed that an owner would be created as a user not only in Identity Manager (and thus in Oracle), but also in ACTIVE Governance.

If you use Identity Manager to create a user and assign her responsibilities in an Oracle resource, and later delete the user in Identity Manager, an Action table retains records of the responsibilities being approved. If you subsequently use Identity Manager to re-create the user, the Action table records prevent SOD analysis from being performed if you reassign any of the responsibilities the user had earlier. (SOD analysis is carried out, however, for any responsibilities she did not have earlier.)

Creating SOD Rules

When ACTIVE Governance is integrated with Identity Manager, you would create SOD rules essentially as you would if you were using Access Governor on its own. See the *ACTIVE Access Governor User's Guide* for details.

There is, however, one significant exception: When Access Governor is used on its own, one can write an SOD rule that designates an approval group as the conflict reviewer. Moreover, one can select a workflow role or a responsibility as an SOD rule owner. (If, for example, a rule designates Purchasing Super User as its owner and does not specify an approval group, everyone assigned the Purchasing Super User responsibility would receive approval requests.)

Identity Manager does not accommodate these features. In an integrated system, when an SOD rule designates an approval group, Identity Manager ignores the group and sends approval requests to the rule owner. Moreover, the owner must be a person, not a role or responsibility (and, as noted above, that person must have a user account in Identity Manager).

Approving Users

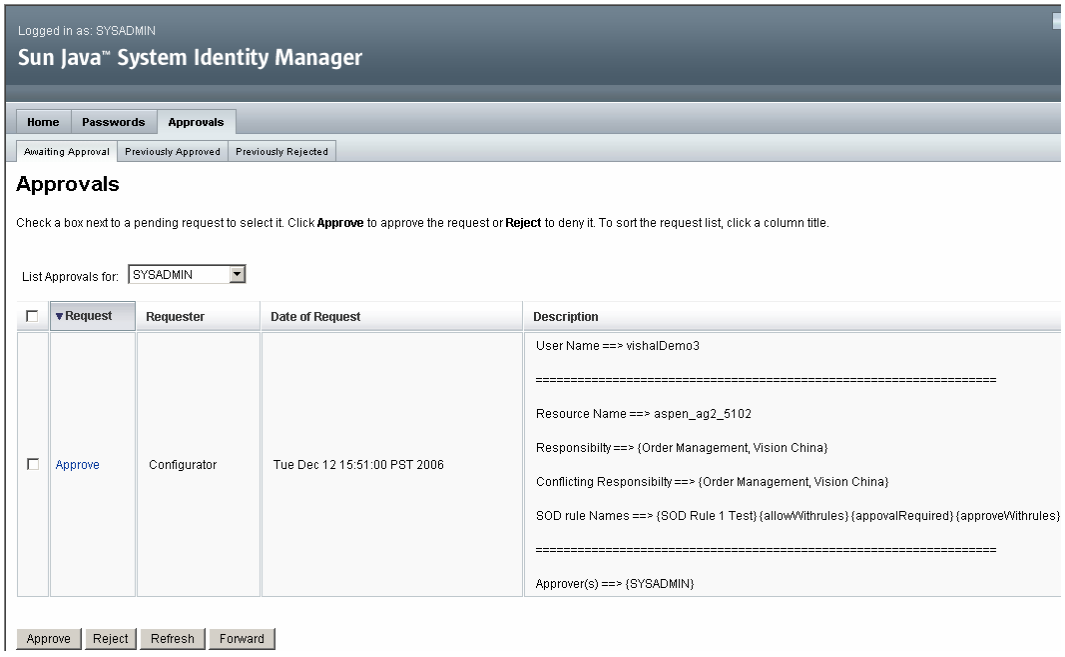
When Access Governor is used on its own, and when responsibility assignments violate Approve with Rules or Approval Required SOD rules, approval requests are distributed in the Oracle ERP instance. The reviewer (owner or approval-group member) approves or rejects the assignment of the particular responsibilities involved in a conflict, rather than the user himself.

When Access Governor is integrated with Identity Manager, requests to approve users with conflicts generated by SOD rules are distributed in Identity Manager; the Oracle notifications no longer occur. In Identity Manager, one approves or rejects the user — that is, all the responsibilities she is being assigned, regardless of whether they are involved in conflicts.

As is standard in Identity Manager, an owner would review approval requests by clicking on the Approvals tab. The first three columns in each request show the type (Approve), the ID of the requester (the person who modified a user account in a way that triggered one or more SOD rules), and the time and date of the request. The final column, labeled Description, provides the following information:

- The User Name (account ID) of the user whose account is being created or modified.
- The name of the resource (Oracle instance) in which the user is being assigned responsibilities.
- The responsibilities which, by being assigned to the user, have triggered one or more SOD rules.
- The names of the SOD rules that these responsibilities have triggered.
- The approvers — the names of the owners designated by the SOD rules.

If the user has been assigned responsibilities in more than one Oracle ERP instance, distinct sets of Resource, Responsibilities, Conflicting Responsibilities, and SOD Rule Names fields display data for each instance. In this case, the rejection of a conflict in any instance prevents the assignment of responsibilities in all instances.



To render a judgment on users, select the check box for approval requests you want to review, and click on the Approve or Reject button.