Siebel

Approval Manager Guide

January 2020



Siebel
Approval Manager Guide

January 2020

Part Number: F12795-04

Copyright © 2020, Oracle and/or its affiliates. All rights reserved

Authors: Siebel Information Development Team

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display in any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this is software or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, the following notice is applicable:

U.S. GOVERNMENT END USERS: Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are "commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

Oracle and Java are registered trademarks of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

This software or hardware and documentation may provide access to or information about content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services unless otherwise set forth in an applicable agreement between you and Oracle. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services, except as set forth in an applicable agreement between you and Oracle.

The business names used in this documentation are fictitious, and are not intended to identify any real companies currently or previously in existence.

Contents

Preface i

1 What's New in This Release 1

What's New in Siebel Approval Manager Guide, Siebel CRM 20.1 Update 1
What's New in Siebel Approval Manager Guide, Siebel CRM 19.11 Update 1
What's New in Siebel Approval Manager Guide, Siebel CRM 19.1 Update 1

2 Siebel Approval Manager Overview 3

Siebel Approval Manager Overview 3
About Siebel Approval Manager 3
Siebel Approval Manager Workflow 3
Siebel Approval Manager Integration 4

3 Installing Siebel Approval Manager 5

Installing Siebel Approval Manager 5
Prerequisites and Requirements 5
Steps Used to Install Siebel Approval Manager 5
Uninstalling Siebel Approval Manager 8

4 Using Siebel Approval Manager 11

Using Siebel Approval Manager 11
Overview of the Creating Approval Request Process 11
Changing the Password in Siebel Approval Manager 12
Creating and Submitting Approval Requests 13
Reviewing Requests 15

5 Configuring Siebel Approval Manager 17

Configuring Siebel Approval Manager 17
Adding Users 17
Adding User Roles and Assigning Users to User Roles 18

Defining the Fields for Use Throughout the SAM Application 20
Setting up the Third-Party Systems 21
Defining the Approval Request Statuses 26
Specifying Messages 27
Configuring Emails 28
Configuring Approval Profiles 30
Configuring Multiple Issue Approval Mode 42
Deploying and Configuring Workspaces 43

6 REST APIs for SAM 51

REST APIs for SAM 51
REST API GET URL Sample Format for Issue Tracking Systems 51
REST API GET Method's JSON Format for Issue Tracking Systems 52
REST API POST URL Sample Format for Issue Tracking Systems 52
REST API POST Method's JSON Format for Issue Tracking Systems 53
Importing System Fields 53
REST API GET URL Sample Formats for SCM Systems 54
REST API GET Method's JSON Format for SCM Systems 54
POST REST API Exposed by SAM for Workflow Actions 55
GET REST API Exposed by SAM to Retrieve Status Types 56
GET REST API Exposed by SAM to Retrieve User Types 56
GET REST API Exposed by SAM to Retrieve List of Profiles 56
REST API Exposed by SAM to Retrieve Approval Requests 57

Preface

This preface introduces information sources that can help you use the application and this guide.

Using Oracle Applications

To find guides for Oracle Applications, go to the Oracle Help Center at *http://docs.oracle.com/*.

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the *Oracle Accessibility Program website*.

Contacting Oracle

Access to Oracle Support

Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit *My Oracle Support* or visit *Accessible Oracle Support* if you are hearing impaired.

Comments and Suggestions

Please give us feedback about Oracle Applications Help and guides! You can send an e-mail to: *oracle_fusion_applications_help_ww_grp@oracle.com*.

1 What's New in This Release

What’s New in Siebel Approval Manager Guide, Siebel CRM 20.1 Update

No new features have been added to this guide for this release. This guide has been updated to reflect only product name changes.

What’s New in Siebel Approval Manager Guide, Siebel CRM 19.11 Update

The following information lists the changes described in this version of the documentation to support this release of the software.

Topic	Description
Uninstalling Siebel Approval Manager	New topic. This topic describes the steps to uninstall the SAM application.
Serialized Delivery of Workspaces	New Topic. This topic describes the configurations for serialized delivery of workspaces from Siebel Application Manager.
Query Based on Multi-Query Request Tokens	New topic. This topic describes the SAM APIs used for searching SAM requests based on multiple query tokens.

What’s New in Siebel Approval Manager Guide, Siebel CRM 19.1 Update

No new features have been added to this guide for this release. This guide has been updated to reflect only product name changes.

2 Siebel Approval Manager Overview

Siebel Approval Manager Overview

This chapter provides an overview of Siebel Approval Manager. It includes the following topics:

- *About Siebel Approval Manager*.
- *Siebel Approval Manager Workflow*.
- *Siebel Approval Manager Integration*.

About Siebel Approval Manager

Siebel Approval Manager (SAM) provides users an approval process to manage approvals of development work. For example, developers can use SAM with an associated issue tracking system (such as Bugzilla, Jira) to fix issues (or bugs). When developers have bug fixes or customizations to deliver, they need to check the fixes or changes into their Software Configuration Management for source code systems (or SCM systems), so they would use SAM to create an approval request record, and that approval request record gathers approval manager, approval committee, QA approvals, or the whole approval process. When the request approval is complete, SAM can initiate the check-in process to check the bug fix into the SCM system.

Within Siebel applications, SAM is most useful when used along with Siebel Workspace. If users make workspace changes, they can make the workspace changes locally and then use SAM to submit an approval request. After the request is approved, workspace changes are delivered into the main Siebel system. In addition to Siebel Workspace, SAM can be used with other third-party tools such as ClearCase, SVN, github, and so on.

You can use SAM for Siebel CRM development, but SAM can also work independently with other applications. It integrates with several third-party issue tracking and software configuration systems (or Software Configuration Management of source code), including the email integration that provides both notifications and approvals across the development team.

SAM is a stand-alone application from Oracle Siebel. This application runs on Tomcat, and its user interface is consistent with Siebel's products and shares the same CSS implementation. This application can be displayed on desktops, tablets, or handheld devices.

You can configure almost all fields in the SAM system, including the fields that SAM populates from the defect tracking system. SAM stores the copy of issue tracking data and subsequent workflow activities in a database for future reporting. This application also supports multiple approval workflows and validates fields using regular expression matching when the data is populated from the associated issue tracking system.

Siebel Approval Manager Workflow

SAM approval workflow follows specific steps in the SAM approval process. These steps can be defined specifically for an approval profile or for all approval profiles.

The SAM approval workflow includes the following steps:

1. Developer enters the unique issue identifier from the associated issue tracking system (for example, the bug number from Bugzilla) into the SAM application.
2. SAM populates all detailed information from the associated issue tracking system to the SAM application itself.
3. Developer enters the source code location such as branch, workspace name, and so on.
4. SAM validates the source code branch or workspace.
5. Developer submits the approval request.
6. Depending on the approval process that is set up by the SAM administrators, approval managers or approval committees are notified.
7. Approvers review the request and then either approve or deny the requests.
8. SAM delivers the source code or workspaces information to the SCM (Software Configuration Management) system.

Siebel Approval Manager Integration

SAM provides a REST interface for the integrations that are listed in the following table:

System	Activity	Example Connector
Issue Tracking	Input ID and list of fields. Output list of field values.	Bugzilla, Jira
Issue Tracking	Update a defect at various events in the application (developer submission, approvals, and so on).	Bugzilla, Jira
Code Review	Pull in code review details for a particular change.	Review Board
Software Configuration	Pull in list of files on the given branch, private branch validation, or rebase validation.	ClearCase, SVN, GitHub, Siebel Workspace
Software Configuration	Possibly automatically check in code after it is approved.	ClearCase, SVN, GitHub, Siebel Workspace
Email	Deliver emails upon specific application events (approval required or approvals completed).	SMTP-compliant email systems and more
Email	Accept manager and committee approvals using email.	IMAP-compliant email systems and more

Note: It is simple to develop the REST APIs to connect to your issue tracking and software configuration systems. For more information on how to develop the REST APIs for Siebel Approval Manager and information about the working examples, see *REST APIs for SAM*.

3 Installing Siebel Approval Manager

Installing Siebel Approval Manager

This chapter provides the information on how to install Siebel Approval Manager (SAM). It includes the following topics:

- *Prerequisites and Requirements*.
- *Steps Used to Install Siebel Approval Manager*.
- *Uninstalling Siebel Approval Manager*

Prerequisites and Requirements

Before you install the Siebel Approval Manager, you must review the guidelines described in the README file. In addition, you will:

- Ensure that Siebel Server is installed along with the application container.
- Set up email integration by the installer.
- Set up REST APIs for the tracking system.
- Optionally, set up REST APIs for the SCM (Software Configuration Management) system.



Note: You can have only one instance of SAM installation in each machine. When you upgrade SAM, you can either upgrade the existing version or delete the existing version from your machine and then install the new version.

Steps Used to Install Siebel Approval Manager

You can use the following steps to install the SAM application.

To install Siebel Approval Manager

1. Browse the location of the SAM build.

   This example shows the location of the SAM build in a Microsoft Windows environment:

   ```
   \\sam-image\SAM-673\Disk1\InstData\Windows\VM\Siebel Approval Manager.exe.
   ```

   This example shows the location of the SAM build in a Unix environment:

   ```
   //net/sam-image/SAM-2.17.8/Disk1/InstData/Linux/ NoVM/ Siebel Approval
   Manager.bin
   ```

2. Invoke the SAM installer using either one of the following options:
   - Double-clicking the Siebel Approval Manager.exe file in a Microsoft Windows environment, or
   - Entering the command "./Siebel Approval Manager.bin" in a Unix environment.

   The Siebel Approval Manager Installation window appears displaying the installation process.
3. Select a language from the list of available options and click the OK button.

   The Introduction window appears.
4. Review the introduction information and click the Next button.

   The Installation Folder window appears.
5. Select a destination folder to install the SAM application.

   You can accept the default destination folder, enter a new folder location into the Where would you like to Install text field, or click the Choose button to select a folder.
6. Click the Next button.

   The Web Server Application Folder window appears.
7. Select an Apache TomcatHome folder to deploy the WAR file.

   You can accept the default destination folder, enter a new folder location into the Please Choose a Folder text field, or click the Choose button to select a folder.

   > **Note:** Oracle Siebel recommends selecting the location where you install Tomcat; for example, C:\Siebel\ses\applicationcontainer.

8. Click the Next button.

   The Database Details window appears.
9. Select a database.

   The available options are Oracle Database Enterprise Edition and Microsoft SQL Server.
10. Click the Next button.

    If you select the Oracle Database Enterprise Edition option, the Oracle Database User Inputs window appears, enabling you to enter the Oracle database information.

    If you select the Microsoft SQL Server option, the MSSQL Database User Inputs window appears, enabling you to enter the Microsoft SQL database information.
11. Enter the required database information.

Field	Example Value in the Oracle Database Server	Example Value in the MSSQL Server
Database Server	slc09urt	slc09uiu
Database Name	<Not applicable>	MSUIU128
Database Instance Name	qaurt1	BIN_INST1

Field	Example Value in the Oracle Database Server	Example Value in the MSSQL Server
Port Details	1551	34100
Username	ORAURT108	MSUIU128
Password	ORAURT108	MSSQL
Table Owner Name	ORAURT108	dbo
Siebel Server Location	C:\Siebel\ses\siebsrvr	C:\Siebel\ses\siebsrvr

12. Click the Next button.
    The SAM Email Configuration Inputs window appears.
13. Enter the SAM email configuration information.

Field	Example Value
SMTP Hostname	internal-mail-router.oracle.com
SMTP Port	25
Username	DoNotReply@sam.com
Password	changeme
Noreply	noreply@oracle.com
Mailto	firstname.lastname@oracle.com

14. Click the Next button.
    The Email REST API Inputs window appears.
15. Enter the email REST API information.

Field	Example Value
Email UserID	samuel.siebel@oracle.com

Field	Example Value
Email Password	siebel
Email Folder	inbox
Email server	stbeehive.oracle.com
Email server port number	993
Web server port number	8080 (This port number is also the Tomcat port number.)

16. Click the Next button.

    The Pre-Installation Summary window appears.
17. Review the summary page and click the Install button.

    The installation process begins. When the installation process is successfully completed, the Install Complete window appears.
18. Accept the default selection to restart your system and click the Done button.

    Alternatively, select the option to restart your system manually. Click the Done button and then restart your system manually.
19. After your system is restarted, verify the WAR files to ensure they are deployed under the SAM application container.
    a. Access this location: <database>\C:\Siebel\ses\applicationcontainer\webapps\.
    b. Check to ensure that sam.war and emailapprover.war files are in the webapps folder.

    For information on how to deploy and configure workspace, see *Deploying and Configuring Workspaces*.
20. Verify that the SAM URL is valid.
    a. Launch the SAM URL using this format: http://<localhost>:<tomcatport>/sam/login.

       For example: http://slc01qlu:8080/sam/login.
    b. Enter admin and admin for the user name and password to log into the SAM application.

       The SAM Home page appears.

Uninstalling Siebel Approval Manager

This topic provides the information on how to uninstall Siebel Approval Manager (SAM):

- *Steps to Uninstall Siebel Approval Manager*

- *Cleaning Up Files*

Steps to Uninstall Siebel Approval Manager

You can use the following steps to uninstall the SAM application.

To uninstall Siebel Approval Manager

1. Navigate to the location of the SAM installation folder.

   For example, the location of the SAM installation folder in a Microsoft Windows environment could be,

   C:\Program Files (x86)\SAM_SAM_installation\SAM_unInstallation.exe

   The location for SAM installation folder in a UNIX environment could be,

   /export/home/SAM/_SAM_installation/SAM_unInstallation
2. Start the SAM uninstaller using one of the following options,
   - Double-click the SAM_unInstallation.exe file in a Microsoft Windows environment.
   - Enter the command "./SAM_unInstallation" in a UNIX environment.

   The Configure SAM window appears displaying the Maintenance mode selector screen.
3. Select the Uninstall Production option and click the Next button.
4. In the Uninstall SAM window that appears with a warning message, click Next to start the uninstall process.

   The uninstallation process begins and on completion, a confirmation message appears.
5. Click Done in the final screen to complete the process.

Cleaning Up Files

In the uninstall process, there may be few files and folder left out that you can remove manually.

For example, in the location of the SAM installation folder in a Microsoft Windows environment, the following files and folders may be removed.

- < Apache TomcatHome folder>\conf\database.properties
- < Apache TomcatHome folder>\conf\email.properties
- < Apache TomcatHome folder>\conf\mail.properties

The following is the example of the location of the SAM installation folder in a UNIX environment.

- /export/home/SAM
- /export/home/SAM_unInstallation
- /export/home/com.zerog.registry.xml

Note: SAM uninstallation does not modify any of the existing database schema.

4 Using Siebel Approval Manager

Using Siebel Approval Manager

After administrators configure the Siebel Approval Manager (SAM) application, SAM users and developers can use the SAM application to create and approve the approval requests.

This chapter describes how to perform the tasks used by SAM users, developers, and approvers. This chapter includes the following topics:

- *Overview of the Creating Approval Request Process*.
- *Changing the Password in Siebel Approval Manager*.
- *Creating and Submitting Approval Requests*.
- *Reviewing Requests*.



Note: If you are authorized as a SAM administrator, you can also select the Administration option from the Main Menu button to perform SAM administrative tasks. For more information, see *Configuring Siebel Approval Manager*.

Overview of the Creating Approval Request Process

After you log into the SAM application, the SAM Home page appears listing all requests that you created, all requests that are pending your approval, or all requests across the system.

- There is a list of approval profiles and the Create button that you can use to create new approval requests.
- After you click the button used to create the approval request, the Create New Approval Request page appears, enabling you to enter the unique identifier from your issue tracking system, such as Bugzilla bug number, Jira ID, and so on.
- After you enter an ID and tab out of that ID field, the fields are populated using the information from your issue tracking system (such as the Bugzilla, Jira, and so on) as defined in the profile configuration.

  If a field has been configured for validation, it is displayed with a checkmark if its value is valid or it is displayed with an alert symbol if its value is not valid.
- Other than the fields from the issue tracking systems, the Create New Approval Request page displays some additional internally generated fields (such as Approval ID, Approval Status, Last Approver) along with optional Comments and Approval Manager Override fields.
- You can specify one or more software configuration locations (ClearCase branch, Workspace name, and so on) that contain the changes to be approved.
- When you click one of the available action buttons to create a request, the approval request is saved and an email notification can optionally be sent to the assigned approvers.

Explanation of fields in the Home page

- Main Menu button.

  This button appears next to the Home title. Clicking this button displays these options: Home, Administration, and About.

  You use the Administration option to configure and manage the SAM application.

- Create New Approval Request drop-down list.

  This section enables you to select a profile from a list of all available profiles used to create new approval requests.

  The Profile values on the drop-down list are defined by the SAM administrator. For more information, see *Managing the Approval Profiles* and *Creating New Approval Profiles*.

- List of Approval Requests.

  This section displays a list of existing approval requests. You can filter the approval request list by selecting one option from the Approval Request drop-down list.

  The available options are My Approval Requests (requests that you created), Pending My Approval, and All Requests.

- Search text box and button.

  Enter an issue tracking ID in the search text box and click the Search button (or click the Enter key on your keyboard) to view the approval request for the corresponding issue tracking record, such as Bugzilla bug number.

- User icon.

  Click this icon to access the User Settings page or Log Out of the SAM application.

Changing the Password in Siebel Approval Manager

You can use the Change Password page to change the password of your user ID in the SAM application.

To change the password in Siebel Approval Manager

1. From the SAM Home page, click the User icon.

   A dialog box appears displaying the user ID, Settings button, and Log Out button.
2. Click the Settings button.

   The Settings page appears.
3. Enter a new password and reenter it to confirm the change.
4. Click the Save button.

Creating and Submitting Approval Requests

You can use the Create Approval Request page to create and submit approval requests. The following table describes the fields in the Create Approval Request page.

Field/Column	Description
Back button	Click this button to return to the Home page.
Configured buttons	Click the appropriate button to follow the next step in your organization's configured approval workflow.
Issue ID text box and Refresh button (two curved arrows)	Enter an issue ID into the Issue ID text box and click the Refresh button (or tab out of the text box) to enable the SAM application to populate the issue's detailed information from the issue tracking system into the SAM system. If the data from the issue tracking system has changed since the first time it was initially populated or the data needs to be refreshed, clicking this button refreshes the information on the page. For example, when a populated field fails validation and you use the drilldown button to access the issue tracking system to fix the issue, when you came back to the SAM system, you should click this Refresh button to reload the data and remove the validation error.
Drilldown to the issue tracking system (down arrow).	Click this button to open the issue in its own tracking system where you can view the issue's detailed information.
Tracking System fields (view-only)	The values in these fields are transferred over from the issue tracking system.
SAM internal fields (view-only)	The values in these fields are specific to the SAM system. These values are not transferred from the issue tracking system.
Comments field	Use this field to add any additional comments that might help the approver or next owner. This field is configurable (may or may not be on the page) and optional. These comments will be available in the Approval History section of the page.
Override Manager field	Use this field to specify a different approver other than the current user's manager (as defined in the Administration \ Users page). This field is configurable (may or may not be on the page) and optional. Note that the value in this field only overrides the approver where the approver is set to Approving Manager (as opposed to a different role).
SCM Details section	Use this section to add the SCM entries to identify the SCM location(s) of the changes that are involved in the approval request.
Approval History section	This section displays the approval history of the request, including the date and time of each approval, the user who approved the request, and so on.

Note: The Back button along with any configured action buttons are available when the status of the approval request is New.

To create an approval request

1. From the SAM Home page, select a profile from the Create New Approval Request drop-down list.
2. Click the Create button.

   The Create Approval Request page appears. The Back button is displayed along with other action buttons that are configured for new requests.
3. Enter an existing unique identifier from your issue tracking system and click the Enter key on your keyboard.

   The field values are populated using the information from the tracking system (the review sources such as the Bugzilla, Jira, and so on). For example, if you enter a Bugzilla bug ID from the Bugzilla system, all detailed information of that bug is displayed in the Create Approval Request page.

   Note: If a field has been configured for validation, it is displayed with a checkmark if its value is valid or it is displayed with an alert symbol if its value is not valid.
4. Correct any validation errors in the tracking system and click the Refresh button to repopulate the data.
5. Optionally, enter a name of an approver to send the approval request directly to that approver instead of the current user's manager (as defined in the Administration\Users page) if the approving manager is specified as an approver for the current profile.
6. Use the SCM Details section to add the source code locations for the changes that are involved in the approval request.

   Note: This step is optional or required depending on how the approval profile is configured. If an action (for example, Submit) is configured with the Validate SCM option selected, then that action button is disabled when there is not at least one validated SCM entry in an approval request. If another action (for example, Save Draft) is configured with the Validate SCM option cleared, then this is an optional step.

   a. Click the Add button on the SCM Details section.
   b. Select an appropriate SCM system from the SCM Profile drop-down list.
   c. Enter the source value (such as ClearCase private branch or Workspace name) and click the Validate button.

      The Validation dialog box appears.
   d. Click the OK button on the Validation dialog box to validate the source value and return to the Create Approval Request page.

      The Validation Results column displays a link that you can click to access and view the validation results.
   e. Optionally, repeat steps a to d to add another SCM entry.

      Note: You can add multiple SCM entries if needed.
7. Click one of the available action buttons.

   The request is sent to the next reviewer(s) in the approval workflow. Email notifications are also sent to these reviewer(s) if email notification is configured in the SAM application.
8. Return to the SAM Home page to view the approval status of the submitted request.

The status is changed to the next status in the approval workflow.

Reviewing Requests

You can use the SAM Home page to view and approve the requests that are submitted to you for approvals.

To review a request through the Approval Request Details page

1. From the SAM Home page, select the Pending My Approval option from the Approval Requests drop-down list.

   The Approval Requests section is filtered and it displays only requests that have an approval action pending by the current user.
2. Click the approval ID link of the request that you want to review.

   The Approval Request Details page appears displaying the action buttons, including the Cancel button and others that are configured for this profile's workflow.
3. Review the request.
4. Click an appropriate action button based on your review (for example, Approve button or Reject button).

   If email notification is configured, then the SAM application sends an email to the next reviewer in the approval workflow as well as the request owner and/or manager.

   The status of the request is also changed to the next status in the approval workflow.

To review a request through email

1. Open the Submitted for Approval email that was sent from Siebel Approval Manager.

   When you receive the Submitted for Approval email as an approver, all action buttons that are on the Approval Request Details page are also on the email.
2. Review the email and click an appropriate action button on the email to send the subsequent email.

   The SAM application is updated accordingly as if you reviewed the request using the Approval Request Details page.

5 Configuring Siebel Approval Manager

Configuring Siebel Approval Manager

If you are assigned the Siebel Approval Manager Administrator role, you can use the Administration option from the Main Menu to perform Siebel Approval Manager (SAM) administrative tasks.

This chapter describes how to perform SAM administrative tasks. This chapter includes the following topics:

- *Adding Users*.
- *Adding User Roles and Assigning Users to User Roles*.
- *Defining the Fields for Use Throughout the SAM Application*.
- *Setting up the Third-Party Systems*.
- *Defining the Approval Request Statuses*.
- *Specifying Messages*.
- *Configuring Emails*.
- *Configuring Approval Profiles*.
- *Configuring Multiple Issue Approval Mode*.
- *Deploying and Configuring Workspaces*

Adding Users

Use the Users page to add new users into the SAM application. The following table describes the fields in the Users page.

Field/Column	Description
Add (plus) button	Click this button to add a new user.
Name column	This column contains the displayed name for the given SAM user.
Email Address column	This column lists the email address for the given SAM user. Note that the email address must be unique for each user ID
Manager column	This column lists the default approving manager for the given SAM user. The named manager is the approver for requests that are created by the given user when the approver is configured as Approving Manager.
Roles column	This column lists the roles that the given SAM user has. You can click the values here to modify the roles of that user.

Field/Column	Description
Reset Password column	Click the Reset Password button to reset the password of the corresponding user and send the same user an email notification with a new, random password.
Delete button	Click the Delete button to delete the corresponding user from the SAM application

Note: SAM is shipped with one user that uses user name admin and password admin; this user is assigned the role of an administrator. Make sure to change the administrator password upon first login. You must create additional user IDs and passwords for all other users.

To add a new user

1. From the SAM Home page, select the Main Menu button.
2. Select the Administration option.
3. Select the Users option from the Administration drop-down list.

   The Users page appears.
4. Click the Add button in the Users page.
5. Enter the user name and email address.
6. Select an approval manager from the Manager drop-down list.
7. Tab out of the Manager drop-down list.

   The user record is created.

   The user and email address that you just added to the SAM system receives a notification email with a new system-generated password. That user can change the password from within the SAM application.
8. Add roles to the new user.
   a. In the Users page, click the role value of the user that you want to add role.

      The User Roles dialog box appears.
   b. In the Available Columns box, select the role that you want to assign to the current user and click the move-right icon.

      The selected role is moved to the Selected Columns box.
   c. Optionally, repeat the previous step to add more roles to the current user.

      Alternatively, you can click the Move All button to assign all available roles to the current user.
   d. Click the OK button.

Adding User Roles and Assigning Users to User Roles

A role is way of grouping one or more responsibilities and assigning them to particular users. Roles are useful for managing large collections of responsibilities. You can assign approver responsibilities and send notification emails to an individual role.

You can use the Roles page to add new user roles to the SAM application and assign user IDs to the user roles. The following table describes the fields in the Roles page.

Field/Column	Description
Add (plus) button	Click this button to add new user roles.
Role Name column	This column lists the display name of the given role.
# of Members column	This column lists the number of users that have the given role. You can click the member count link to view the members of each role
Delete button	Click the Delete button to delete the corresponding user role from the SAM application



Note: Administrator is the default role in the SAM application and this role cannot be deleted. Only users with the administrator role can access the Administration menu from the Main Menu and perform the administrative tasks. Other roles can be added to the SAM application by the SAM administrators.

To add a new user role

1. From the SAM Home page, select the Main Menu button.
2. Select the Administration option.
3. Select the Roles option from the Administration drop-down list.

   The Roles page appears.
4. Click the Add button in the Roles page.
5. Enter a role name.
6. Tab out of the Role Name field.

   The new role is listed under the Role Name section. Notice that the newly created role has zero members (users).

To assign a user to a user role

1. From the SAM Home page, select the Main Menu button.
2. Select the Administration option.
3. Select the Roles option from the Administration drop-down list.

   The Roles page appears.
4. In the # of Members column, click the value of the role for which you want to add users.

   The User Roles dialog box appears.
5. Select a user from the Select User drop-down list.
6. Click the Add button.
7. Optionally, repeat Steps 5 and 6 to add more users to the current user role.
8. Optionally, click the Remove button to remove any user from the current user role.
9. Click the Close (X) button to finish adding or moving users, and return to the Roles page.

Defining the Fields for Use Throughout the SAM Application

Oracle Siebel delivers numerous defined internal fields with the SAM application. You can use the Fields page to add the defined fields to be displayed throughout the SAM application and modify the default labels for each of the fields.

In other words, the Fields page enables you to reuse many of the shipped fields to store the data that is populated from your issue tracking system. The fields that are shipped, such as CUSTOMER or PRIORITY, can be reused for any purpose by changing the default label. This new label is then displayed throughout the SAM application instead of the old default label. The following table describes the fields in the Fields page.

Field/Column	Description
Field column	This column contains the internal system fields in the Fields page.
Default Label column.	This column contains the default label of the current field. To change any label, click that label and type the new label over the current one. If it is left blank, the system name is displayed.
List Order column	This column contains the order of the current field in the Approval Requests section on the SAM Home page. If a field has a List Order set to Hidden, that field is not displayed in a table on the Approval Requests section.

To define the fields that will be displayed on the SAM Home page

1. From the SAM Home page, select the Main Menu button.
2. Select the Administration option.
3. Select the Fields option from the Administration drop-down list.

   The Fields page appears.
4. Optionally, modify the default labels.
   a. In the Default Labels column, click the label of the field that you want to modify.
   b. Type the new label.
   c. Repeat step 4a and step 4b to modify additional labels.
5. Optionally, modify the list order of the fields.
   a. In the List Order column, click the list order of the field that you want to modify and type the new list order number.
   b. To set a field to Hidden, remove the current value of the field in the List Order column and click (or tab) out of that field.

      The value automatically changes to Hidden.
   c. Repeat step 5a and step 5b to modify the list order of other fields.

Setting up the Third-Party Systems

You can use the Systems page to register and set up the third-party systems that integrate with the SAM application.

- Examples of the issue tracking systems that integrate with the SAM application are Bugzilla, Jira, or Review Board.
- Examples of the Software Configuration Management (SCM) systems are ClearCase, SVN, and Siebel Workspace.

The following table describes the fields in the Systems page.

Field/Column	Description
Add (plus) button	Click this button to add new system.
System Type column	This column lists the type of the third-party system. Two system types are available: Issue Tracking and SCM (Software Configuration Management).
System column	This column lists the name of the integrated system and this is a user-entered display name.
Read/Update column	Use the button in this column to display the Read/Update Method dialog box. • Issue Tracking Systems: When users enter an issue tracking system's unique identifier on the Create Approval Request page, the SAM application invokes the REST API whose URL is defined here to pull information from the associated issue tracking system and populate the fields on the page. • SCM Systems: When the user enters an SCM location (such as a ClearCase branch or Workspace name) and clicks the Validate button, the SAM application invokes the REST API whose URL is defined here to confirm that the given location is valid. SAM will prompt for the user to enter a value for the source field on the Approval Request page. That source value will be stored along with the approval request and can be used to deliver the changes once the request is fully approved. SAM can prompt for additional fields if they are required for validation but they will not be stored. Examples: Simple single parameter scenario: http://hostname:8080/worskpace/wsname/{source} The element {source} is available in the Field drop-down list to select and add. The source captures the workspace name when an SCM is added for delivery during the Create Request process. This source is stored for future delivery. Two parameter scenario: http://hostname:8080/github/repo/{repository}/pbname/{source} This example uses two different parameters that are required for validation and both are prompted for on the Approval Request page. Again, the additional {repository} parameter is not stored for future use and is not available as an item in the Field drop-down list so it must be entered into the URL manually.

Field/Column	Description
Write/Delivery column	Use the button in this column to display the Write/Delivery Method dialog box. • Issue Tracking Systems: After an action is taken on a request, the SAM application can invoke the REST API whose URL is defined here to update the issue tracking system. For example, if a request is approved, the SAM application can add a comment and update the status of the record of the issue tracking system record. • SCM Systems: After an action is taken on a request, the SAM application can invoke the REST API whose URL is defined here to deliver the changes defined in the SCM location(s). For more information on how to invoke these REST API calls in the Updates table, refer to *Setting Up Approval Profile Actions in the SAM Workflow*. **Note:** This feature for SCM systems is only supported in 2.17.10 and onwards.
Drilldown URL column	Use the button in this column to display the Drilldown URL dialog box where you can add or modify the URL that the SAM application uses to drill down to the tracking systems. **Note:** This is only for Issue Tracking systems. This is an example of the Drilldown URL for issue tracking system: http://hostname:8880/bugzilla/show_bug.cgi?id={ISSUE_ID}. This example points to the URL of Bugzilla. When users click the Drilldown URL button next to the bug ID on the Create Approval Request page, the SAM application opens the bug in Bugzilla.
Fields column	Use the button in this column to display the Config Fields dialog box where you can configure this system's fields that are available to use throughout SAM. To load the list of fields, enter the URL for the system's field JSON file and click the Get Fields JSON button. For more information regarding the field JSON format, refer to the section *Importing System Fields*. The following properties can be configured for each field: • Name – The internal name of the field (this property should not be edited). • Label – The label of the field that is displayed within SAM. • Active – When selected, this field is available to be selected throughout SAM when selecting fields for the system. • Default Status – Selecting this identifies this field as the status field for this system. This is only used for Issue Tracking systems. • Key Field – Selecting this identifies this field as the unique identifier field for this system. This is only used for Issue Tracking systems. • List Order – The order that that this field is displayed in any field lists in SAM. Issue Tracking systems: These are the issue tracking system fields that are available within SAM to be displayed on the Approval Request page and updated when an approval status changes. SCM systems: These are the fields that SAM can pass to the SCM system's Write/Delivery REST API when an approval status changes (for example, delivering workspace changes when a request is approved). When data is sent to these REST APIs, each field is mapped in the Workflow>Actions>Update section for that profile and is sent with dynamic data bound with the help of Messages. For more information, see the *Fields in the Update section* table.

Field/Column	Description
	Note: This feature for SCM systems is only supported in 2.17.10 and onwards.
Delete button	Click the Delete button to delete the corresponding system type from the SAM application.

To set up an Issue Tracking third-party system

1. From the SAM Home page, select the Main Menu button.
2. Select the Administration option.
3. Select the Systems option from the Administration drop-down list.
   The Systems page appears.
4. Click the Add button in the Systems page.
5. Select Issue Tracking from the System Type drop-down list.
   Two system types are available: Issue Tracking and SCM.
6. Enter the name of the third-party system into the System field.
7. Tab out of the System field.
8. Add the REST API URL that the SAM application will use to read the data from the tracking system.
   Suppose that in the Administration\Fields page, you have set the default label of the ISSUE_ID field to Bug # and plan to store the Bugzilla Bug Number value in that field. For the Read/Update REST API to work properly, you will need to pass the Bug ID into the URL.
   a. Locate the tracking system that you just added and click its corresponding Read/Update button.
      The Read/Update Method dialog box appears.
   b. Enter the hardcoded portion of the URL.
      For example, http://hostname:<port number>/bugzillarest/bugs/.
   c. Place the cursor where you want to insert the Bug Number field value.
      In this case, that is at the very end of the string.
   d. From the Insert Field Value into URL drop-down list, select the name of the field whose value you want to insert into URL, which is Bug # in this example.
      The text {ISSUE_ID} is inserted into the URL where you placed your cursor so the resulting URL is http://hostname:<port number>/bugzillarest/bugs/{ISSUE_ID}.
   e. Click the OK button.
      When a user accesses the Create Request page, enters the value 1234 into the Bug ID field, and clicks the Enter key, the SAM application invokes http://hostname:<port number>/bugzillarest/ bugs/1234 to pull data from the issue tracking system into the page.
9. Add the REST API URL that the SAM application will use to write the data back from the SAM application to the tracking systems.
   a. Locate the tracking system that you just added and click its corresponding Write/Delivery button.
      The Write/Delivery Method dialog box appears.

b. Enter the hardcoded URL into the Write/Delivery URL field.
This is an example of the Write/Delivery URL: http://hostname:<port number>/BugzillaRESTService/bug/.
c. Click the OK button.

10. Add the URL that the SAM application will use to drilldown to the tracking systems.
a. Locate the tracking system that you just added and click its corresponding Drilldown URL button.
The Drilldown URL dialog box appears.
b. Add a URL into the Drilldown URL field.
This is an example of the drilldown URL: http://hostname:<port number>/bugzilla/ show_bug.cgi?id={ISSUE_ID}.

> **Note:** You can use the Insert Field Value into URL drop-down list to select the name of the field and insert it into URL. For more information on how to insert field value into URL, see step 8.

c. Click the OK button.

11. Configure the issue tracking fields that are used by the SAM application.
a. Locate the system that you just added and click its corresponding Fields button.
The Config Fields page appears.
b. In the URL field, enter the URL of the tracking system's field JSON file. For more information on the expected JSON format, see *Importing System Fields*.
c. Click the Get Fields JSON button.
The SAM application lists all available fields that are populated from the tracking system.
d. In the Label column, modify the labels as needed.
e. Select the Active option for the fields that you want to be used within SAM.
f. In the Default Status column, select the radio button in the Default Status column for the issue tracking system's status field.
For example, this would be the status field in Bugzilla.
g. In the Key Field column, select the radio button in the Key Field column for the issue tracking system's unique identifier field.
For example, this would be the bug number field in Bugzilla.
h. Modify the listing order of the fields using the values in the List Order column.
i. Click the OK button.

To set up an SCM third-party system

1. From the SAM Home page, select the Main Menu button.
2. Select the Administration option.
3. Select the Systems option from the Administration drop-down list.
The Systems page appears.
4. Click the Add button in the Systems page.
5. Select SCM from the System Type drop-down list.
Two system types are available: Issue Tracking and SCM.

6. Enter the name of the third-party system into the System field.
7. Tab out of the System field.
8. Add the REST API URL that the SAM application will use to validate the approval request's SCM source (private branch or workspace name for example).

    a. Locate the SCM system that you just added and click its corresponding Read/Update button.

    The Read/Update Method dialog box appears.

    b. Enter the hardcoded portion of the URL.

    For example, http://hostname:<port number>/workspace/wsname/.

    c. Place the cursor where you want to insert the private branch name or workspace name

    In this case, that is at the very end of the string.

    d. From the Insert Field Value into URL drop-down list, select the field source.

    > **Note:** This is an in-built field to store private branch name or workspace name that is also used when delivery is triggered from SAM.

    The text {source} is inserted into the URL where you placed your cursor so the resulting

    URL is http://hostname:<port number>/workspace/wsname/{source}

    > **Note:** You can configure a more complex URL with multiple parameters.

    For example,

    http://hostname:<port number>/github/repo/{repository}/pbname/{source}

    In this case, the element {repository} is not available as an item in the Field drop-down list and must be manually entered. When an SCM entry is added to an approval request, these two elements are prompted to get entered.

    e. Click the OK button.

    When a user creates a request and adds an SCM entry for delivery, a prompt for {source} will be displayed. On clicking the Validate button, a pop-up screen appears where {source} and the similar elements like {repository}, if configured, will get populated for input.

    On clicking the OK button in the pop-up screen, if the user entered dev_mypb as their private branch, for example, the SAM application invokes http://hostname:<port number>/workspace/wsname/dev_mypb to validate the SCM entry.

    For the complex scenario, if the user entered staging for the {repository} prompt, the following URL is invoked:

    http://hostname:<port number>/github/repo/staging/pbname/dev_mypb

9. Add the REST API URL that the SAM application will use to deliver the SCM entry changes.

    > **Note:** The Write/Delivery method of SCM system is only supported in 2.17.10 and onwards.

    a. Locate the SCM system that you just added and click its corresponding Write/Delivery button.

The Write/Delivery Method dialog box appears.

b. Enter the hardcoded URL into the Write/Delivery URL field. The SAM application will use this REST API URL to deliver the SCM entry changes.
For example, http://<servername>:<port number>/workspacedelivery/workspace/
For more information refer to *Deploying and Configuring Workspaces*.

c. Click the OK button.

10. Configure the SCM fields that are used by the SAM application for delivery parameters.
    a. Locate the SCM system that you just added and click its corresponding Fields button.
    The Config Fields page appears.
    b. In the URL field, enter the URL of the SCM system's field JSON file. For more information on the expected JSON format, see *Importing System Fields*.
    c. Click the Get Fields JSON button.
    The SAM application lists all available fields that are populated from the SCM system.
    For example, for the URL entered in the step b, the two fields that display are requestid and wsname.
    Select Key Field option button for requestid and Default Status option for wsname.
    d. In the Label column, modify the labels as needed.
    e. Select the Active option for the fields that you want to be used within SAM.
    f. Modify the listing order of the fields using the values in the List Order column.
    g. Click the OK button.

Defining the Approval Request Statuses

You can use the Statuses page to create, edit, or remove the approval request statuses based on the desired workflow at your corporation. The following table describes the fields in the Statuses page.

Field/Column	Description
Add (plus) button	Click this button to add new approval request statuses.
Status Name column	This column contains the approval request statuses in the SAM application.
Delete button	Click the Delete button to delete the corresponding approval request status from the SAM application.

To add an approval request status

1. From the SAM Home page, select the Main Menu button.
2. Select the Administration option.
3. Select the Statuses option from the Administration drop-down list.
   The Statuses page appears.

4. Click the Add button in the Statuses page.
5. In the Status field, enter a name for the new approval request status.
6. Tab out of the Status field.

To modify an approval request status

1. From the SAM Home page, select the Main Menu button.
2. Select the Administration option.
3. Select the Statuses option from the Administration drop-down list.

   The Statuses page appears.
4. In the Status Name column, click the status that you want to modify.
5. Enter the new name for the approval request status over the current name.
6. Tab out of the Status Name field.

Specifying Messages

You use the Messages page to create and customize the messages that are used in validation error messages, email communication, or tracking system updates. For example, if you want a validation error when an invalid value occurs for the Target Release field, you could create a message with the text Invalid value for {TARGET_RELEASE}. The following table describes the fields in the Messages page.

Field/Column	Description
Add (plus) button	Click this button to add new messages.
Message Name column	This column contains the name of the given message. Notice that the message name must be unique for each message.
Text column	This column contains the text of the given message. You can click the link to open the Message Details dialog box where you can edit the corresponding message.
Delete button	Click the Delete button to delete the corresponding message from the SAM application.

To add a new message

1. From the SAM Home page, select the Main Menu button.
2. Select the Administration option.
3. Select the Messages option from the Administration drop-down list.

   The Messages page appears.
4. Click the Add button in the Messages page.

   The Message Details dialog box appears.
5. In the Message Name field, enter the name of the message.
6. In the Message Text field, enter a message text.

7. Optionally, use the Insert Field Value into Text drop-down list to insert the field value from the approval request record.
    a. Place the cursor in the message text where you want to insert the field value.
    b. From the Insert Field Value into Text drop-down list, select the name of the field whose value you want to insert into the message text.
8. Click the OK button.

To modify an existing message

1. From the SAM Home page, select the Main Menu button.
2. Select the Administration option.
3. Select the Messages option from the Administration drop-down list.

   The Messages page appears.
4. In the Text column, click the text link for the message that you want to modify.

   The Message Details page appears.
5. Modify the message name and message text as required.
6. Optionally, use the Insert Field Value into Text drop-down list to insert the field value from the approval request record.
    a. Place the cursor in the message text where you want to insert the field value.
    b. From the Insert Field Value into Text drop-down list, select the name of the field whose value you want to insert into the message text.
7. Click the OK button.

Configuring Emails

This topic describes how to configure SAM for sending email and approving by email.

- Configuration for sending email: This configuration is in the installer screen SAM Email Configuration Input. This screen captures the SMTP server, port number, User Id, and password of SMTP server authentication for sending email. Configurations also include capturing a no reply email Id and an email Id to receive reply action emails.
- Configuration for approval by email: The screen Email REST API Input is used to configure inbound emails for approval actions. This includes the configuration of the email Id to receive approval emails, the password, the default folder for Email REST API to look for the incoming emails, such as Inbox, the IMAP server URL, IMAP server port number, SAM being hosted in http or https, port number of the SAM hosting.

Reconfiguring Emails

This topic describes how to reconfigure the email sending mechanism and the Approval by Email feature.

Reconfiguring Send Email

To reconfigure the email sending mechanism, modify the following files:

- The configuration of SMTP is defined in the file, sam\resources\spring\mail\mail.xml

> **Note:** By default SMTP authentication is disabled.

- Values of this configuration can be modified in the file, applicationcontainer\conf\mail.properties

  Following items are available for modification,

  mail.smtp.hostname=<SMTP host URL>

  mail.smtp.port=<SMTP server port number>

  mail.username=<email id for SMTP auth>

  mail.password=<password of the email id>

  mail.noreply=<noreply@sam.com>

  mail.mailto=<email id of the inbox that will receive approval emails>

Reconfiguring Approval Email

To reconfigure the Approval by Email feature

- The configuration for IMAP settings are stored in the file, emailapprover\resources\spring\context\context-integration.xml

  Setting can be found under the section, <mail:inbound-channel-adapter id="imapAdapter" ...
- The values for these settings can be modified in applicationcontainer\conf\email.properties

  Following items are available for modification,

  email.id=<email id of the inbox that will receive approval emails>

  email.pwd=<encrypted password>

  email.folder=<inbox>

  email.uri=<server name>:<port number>

  email.samURL=http://<server name>:<port number>

  email.samUserName=<email id of the SAM user which is needed for approval actions via API calls>

  email.samPassword= <encrypted password>

  To update the passwords, use the following URL to get encrypted password,

  http://<server name>:<port number>/emailapprover/encrypt/<new password>

Configuring Approval Profiles

An approval profile describes how the Approval Request pages appear, which systems are used, and defines the Approval workflow. You should create different approval profiles for different types of approval requests. For example, you should create different profiles for different approval workflows, different issue tracking or SCM systems, different sets of fields that are required to be displayed, different validation rules, and so on.

This topic describes how to manage and create the approval profiles in the SAM application. It includes the following information:

- *Managing the Approval Profiles*.
- *Creating New Approval Profiles*.
- *Identifying Fields Displayed on the Approval Request Page*.
- *Defining Approval Workflows*.
- *Enabling Conditional Workflows in the SAM Application*.
- *Setting Up Approval Profile Actions in the SAM Workflow*.

Managing the Approval Profiles

You can use the Approval Profiles page to manage approval profiles such as activating, deactivating, or deleting the existing approval profiles. This page also enables you to access the Approval Profiles detail page where you can view the detailed information of the existing approval profiles or create new approval profiles. The following table describes the fields in the Approval Profiles page.

Note: You may create different approval profiles that enable you to manage a different set of fields, approval workflows, software configuration systems, and tracking systems for each profile.

Field/Column	Description
Add (plus) button	Click this button to add new approval profiles into the SAM application.
Profile Name column.	This column contains the name of the corresponding approval profile. Click on the name to access the Approval Profiles detail page where you can edit the selected approval profiles.
Issue Tracking System(s) column	This column displays the tracking systems for the corresponding approval profile.
SCM System(s) column	This column displays the SCM systems for the corresponding approval profile.
Active column	This column indicates whether the approval profile is active. Notice that only active approval profiles can be used to create approval requests.
Delete button.	Click the Delete button to delete the corresponding approval profile from the SAM application.

Creating New Approval Profiles

You can use the Approval Profiles detail page to create an approval profile that will be used to identify which fields are displayed on the Approval Request page. You can also use this page to define the approval workflow and map which software configuration systems are used in the approval profiles.

You can define the following options for an approval profile:

- Profile names.

  The names that are displayed in the Create Approval Request drop-down list on the Home page.
- Specific issue tracking system.

  For example, Bugzilla or Jira.
- Set of fields that are identified to populate the data from the issue tracking system along with what their display labels, display order, and acceptable (valid) values.
- A set of allowed software configuration connectors for this specific source.

  For example, ClearCase, SVN, or Workspaces.
- A specific set of approval workflow steps and approval workflow.

The following table describes the fields in the Approval Profiles detail page.

Field/Column	Description
Profile Name	The name that is displayed in the Create Approval Request drop-down list on the Home page.
Active?	On selection, the approval profile gets activated. Only active profiles can be used to create approval requests.
Issue Tracking System(s)	The systems that help track and fix issues. For example, Bugzilla or Jira.
Primary Tracking System	The issue tracking system that is selected as the primary tracking system.
Issue Tree Open Statuses	Specifies a comma-delimited list of status values that indicate an Open status with the primary issue tracking system.
Issue Tree Open Statuses	Specifies a comma-delimited list of status values that represent a Closed status.
SCM System(s)	The SCM systems where SAM initiates the check-in process to deliver the associated changes. For example, ClearCase, SVN, or Workspaces.
Save button	Saves the approval profile. The Fields and Workflow sections are active and available. The newly created approval profile is listed under the Create New Approval Request drop-down list on the SAM Home page if the profile is Active. The Save button only saves the first section of the page. The data within the Fields and Workflow sections are saved once you exit a row you are editing/adding.

Field/Column	Description
Back button	The page reverts to the Approval Profiles page. On clicking the Back button without saving approval profile, the profile details are lost.

To create a new approval profile

1. From the SAM Home page, select the Main Menu button.
2. Select the Administration option.
3. Select the Approval Profiles option from the Administration drop-down list.

   The Approval Profiles page appears.
4. Click the Add button in the Approval Profiles page.
5. In the Profile Name field, enter the name for the new approval profile.

   Note: Each approval profile must have a unique name.
6. Select the Active option to activate the approval profile.

   Note that only active profiles can be used to create approval requests.
7. Click the search (magnifier) icon next to the Issue Tracking Systems field.
8. Select one or more issue tracking systems from the list of available options.

   Note: If multiple issue tracking systems are chosen, they must share a common unique key.
9. Click the OK button.
10. Click the search icon next to the Primary Tracking System field.
11. Select one primary tracking system from the list of available options, which are displayed based on your selection in the Issue Tracking Systems field.

    For example, if Bugzilla and a Code Review system are chosen, then Bugzilla is the primary issue tracking system and the Bugzilla Bug number (#) field would be the unique key for both Bugzilla and the Code Review system.
12. Optionally, enter the values in the Issue Tree Open Statuses field and the Issue Tree Closed statuses field.
    - Issue Tree Open Statuses: Specifies a comma-delimited list of status values that indicate an Open status with the primary issue tracking system.
    - Issue Tree Closed Statuses: Specifies a comma-delimited list of status values that represent a Closed status.
13. Click the search icon next to the SCM System(s) field and select one or more SCM systems from the list of available options.
14. Click the OK button.
15. Click the Save button.

    The Fields and Workflow sections are active and available after you save the approval profile. The newly created approval profile is listed under the Create New Approval Request drop-down list on the SAM Home page if the profile is Active.

    For more information on the Fields section, see following section.

Identifying Fields Displayed on the Approval Request Page

Use the Fields section to determine which fields are displayed on the Approval Request page. Thus, the Fields section on the Approval Profiles detail page enables you to identify which fields are displayed on the Approval Request page and map which internal fields the issue tracking data is stored in. The following table describes the fields in the Fields section on the Approval Profiles detail page.

Field/Column	Description
Add (plus) button	Click this button to add new fields that will be displayed on the Approval Requests page.

Order column	This column displays the number denoting the order in which to place the corresponding field.
	
	On the larger screens fields are displayed in two columns and ordered left, right, left, right, and so on based on this field.
	
	For example, they are displayed like this:
	
	`1 2`
	`3 4`
	`5 6`
	
	On smaller screens (i.e. mobile devices), there is only one column of fields and they are ordered by this field value.

Source column	The column contains the system from where the corresponding field originates. This can either be one of the issue tracking systems defined for this profile or Internal to denote that this field is a SAM internal field.
Source Field column	This column contains the name of the field (from the system selected in the source column) whose value is used to populate the corresponding field. You can optionally select the <DIVIDER> option in the System fields to add a line to the approval profile to separate the fields on the page.
Mapped System Field column	This column contains the SAM field where the external system data, as identified in the two previous columns, will be stored. **Note:** The Mapped System Field is inactive if the Source column is set to Internal. This column value is optional. If it is not specified, then the external field value is only displayed and will not be stored within SAM.
Override Label column	This column contains the override field label that is optional and will be displayed next to the corresponding field on the Approval Request page. If no value is entered here, then the default field label is used as the label. These default labels are specified in the Field Administration page. For example, suppose that you select the following field values: • The Source is Sample Bug System.

Field/Column	Description
	• The Source Field is ID. • The Mapped System Field is Bug Number. • Override Label is Bug #. That means that the SAM application will pull the ID field from Sample Bug System, display it in a field labeled Bug # on the Approval Request page, and then store it in Bug Number internal field.
Default View column	If this check box is selected, then the field is displayed in the expanded record.  **Note:** If Multiple Issue Approval is not enabled, then this field can be ignored.
Header column	If this check box is selected, then the field is displayed in the header of the collapsed record.  **Note:** If Multiple Issue Approval is not enabled, then this field can be ignored.
Enable URL column	If this check box is selected, then the drilldown icon will be displayed next to this field on the Approval Request page.
Validation Values and Message columns	These columns display the validation values and corresponding messages. When no value is added, the column displays None. A regular expression can be optionally entered in the Validations Values field. When the Approval Request page is loaded, the value of the field is compared with this regular expression and the following happens: 1. If the value matches with the regular expression, then the field is displayed with a check mark. 2. If the value does not match, then: o The field is displayed with an alert symbol o The buttons for any actions that have the Validate? flag selected will be disabled o The message selected for Validation Message is displayed for the field. The following are a few examples of regular expressions: • Regular Expression: 1.([0-9]|1[0-9]) Description: Range from 1.0 to 1.19 Examples: 1.2, 1.18 • Regular Expression: 1.2 Description: Equals the value 1.2 exactly Examples: 1.2 • Regular Expression: 1.2|1.3 Description: Either of the two values, 1.2 or 1.3 Examples: 1.2,1.3 • Regular Expression: Siebel \s+Engineering Description: Two strings, Siebel and Engineering, separated by a white space

Field/Column	Description
	Examples: Siebel Engineering • Regular Expression: Siebel \s+Engineering* Description: Two strings, Siebel and Engineering, separated by a white space and followed by any other characters (wildcard) Examples: Siebel Engineering Tools • Regular Expression: .*(Engineering|Installer).* Description: Any string that contains either the string Engineering or the string Installer Examples: Siebel Installer for IP For more information on how to add new message into the SAM application, see *Specifying Messages*.
Delete button	Click the Delete button to delete the corresponding field from the Fields list. The deleted field will not be displayed on the Create Approval Requests page.

Defining Approval Workflows

You use the Workflow section on the Approval Profiles detail page to define the SAM approval workflows. In this section, you can identify which conditions must be met for the approval actions to be available, including the starting statuses, action owners, and additional configurable fields with matching conditions.

For example, when the status of a request is New and the logged in user is the Owner of the request, then the request can be submitted to be reviewed by their manager. When the status is Pending Approval and the logged in user is the manager of the owner of the request, then the request can either be Approved or Rejected.

The following table describes the fields in the Workflow section on the Approval Profiles detail page.

Field/Column	Description
Add (plus) button	Click this button to add new starting statuses.
Starting Statuses column	This column contains the status or statuses that the request must be in for this workflow step to apply.
Action Owners column	This column displays the role responsible for the completion of the corresponding workflow step. The logged in user must be in at least one of the roles for this workflow step to apply. The Role can either be Owner (the creator of the request), Approving Manager (the manager of the request's creator as defined in the User administration page) or one of the roles defined in the Role administration page.
Conditions column	This column displays any additional conditions that must be met for this workflow step to apply. Click on the link to open the Workflow Steps Conditions dialog box where you can enable conditional workflows in the SAM application. For more information, see *Enabling Conditional Workflows in the SAM Application*.

Field/Column	Description
Actions column	This column lists the actions that can be taken by the logged in user when all of the conditions defined in the previous columns have been met. Click on the link to access the Approval Profile Actions dialog box where you can modify the existing approval actions. For more information, see *Setting Up Approval Profile Actions in the SAM Workflow*.
Delete button	Click the Delete button to delete the corresponding starting status from the SAM application.

To add a new workflow row

1. From the SAM Home page, select the Main Menu button.
2. Select the Administration option.
3. Select the Approval Profiles option from the Administration drop-down list.

   The Approval Profiles page appears.
4. Click the profile where you want to add a new workflow row.
5. Scroll down to the Workflow section and click the Add button.
6. In the Starting Statuses field, select the starting statuses.

   a. Click the search (magnifier) icon next to the Starting Status field.

      The Choose Starting Statuses dialog box appears.
   b. Select one or more starting statuses from the list of available options.
   c. Click the OK button.
7. In the Action Owners field, select the action owners.

   a. Click the search (magnifier) icon next to the Action Owners field.

      The Choose Approvers dialog box appears.
   b. Select one or more approvers from the list of available options.
   c. Click the OK button.
8. Tab out of the Action Owners field.

   Locate the starting status that you just created. Notice that its condition is None and its action value is No Actions Defined.

For more information on how to enable conditional workflow in the SAM application, see *Enabling Conditional Workflows in the SAM Application*.

For more information on how to set up approval profile actions in the SAM application, see *Setting Up Approval Profile Actions in the SAM Workflow*.

Enabling Conditional Workflows in the SAM Application

You can use the Workflow Step Conditions dialog box to add additional conditions to the expressions of each step within the SAM workflow. These conditions compare a field with a regular expression (that is similar to the field validation). However, adding additional conditions is optional.

The SAM application evaluates the conditions grouped together by the operator ANDs. In other words, the SAM application puts parentheses around each set of operator ANDs (without displaying the parentheses). For example, if you define the following conditions:

1. Status matches 25
2. AND Fix By matches IP2016
3. OR Status matches 16
4. AND Fix By matches CIR4
5. AND Assignee matches SAMUEL
6. OR Status matches 39|40
7. AND Assignee matches SADMIN

Then SAM evaluates the conditions as following:

```
(Status = 25 AND Fix By = IP2016) OR (Status = 16 AND Fix By = CIR4 AND Assignee =
SAMUEL) OR (Status = 39|40 AND Assignee = SADMIN)
```

The following table describes the fields in the Workflow Step Conditions dialog box.

Field/Column	Description
Add (plus) button	Click this button to add new conditions to the expressions of each step within the SAM workflow.
Order column	This column displays a number denoting the order in which to list the corresponding condition.
AND/OR column	This column displays the values of AND and OR and it is not available for the first condition.
Field column	This column contains the field that is used to match against for this condition. The field can be selected from all fields that are mapped to system fields and some additional system fields such as Owner ID and Manager ID.
Expression field	Use this text field to enter the regular expression to match against the field's value. For more information, see Validation Values and Message Columns in the *Fields section on the Approval Profile details* table.
Delete button	Click the Delete button to delete the corresponding workflow step condition from the workflow step.

To add new conditional workflow steps to the SAM application

1. From the SAM Home page, select the Main Menu button.
2. Select the Administration option.
3. Select the Approval Profiles option from the Administration drop-down list.

   The Approval Profiles page appears.
4. Scroll down to the Workflow section and click the condition link for the starting status that you want to add condition.

   The Workflow Step Conditions dialog box appears.
5. Click the Add button.
6. Enter a condition list order number using the Order field.
7. Select the operator AND, OR from the AND/OR drop-down list.

This list is not available for the first workflow step condition.

8. Select a field from the Field drop-down list.
9. Enter an expression in the Expression text field to match against the field's value.
10. Tab out of the Field drop-down list.
11. Click the Close button to return to the Approval Profiles detail page.

To modify an existing condition within the SAM workflow step

1. From the SAM Home page, select the Main Menu button.
2. Select the Administration option.
3. Select the Approval Profiles option from the Administration drop-down list.
   The Approval Profiles page appears.
4. Scroll down to the Workflow section and click the condition link for the starting status that you want to modify.
   The Workflow Step Conditions dialog box appears.
5. Click the field values that you want to modify and type the new values over the current ones.
6. Tab out of the field.
7. Click the Close button to return to the Approval Profiles detail page.

Setting Up Approval Profile Actions in the SAM Workflow

You use the Approval Profile Actions dialog box to set up the approval profile actions for the SAM workflow steps. In this dialog box, you can define the actions and their results when the listed conditions are met. The results include:

- Update the Approval Request's status
- Send notification email
- Update fields on the associated issue tracking system record
- Execute an SCM delivery script

The following table describes the fields in the Approval Profiles Action dialog box.

Field/Column	Description
Action Name	The action name will be the name of the action button that is displayed in the Approval Request page.
Result Status	The status of the approval request. When a user clicks on this action button, the status of the Approval Request page will be updated to this value. For example, assume that the value in the Action Name field is Save Draft and the value in the Result Status is Draft. When users click the Save Draft button on the Approval Request page, the approval request is saved and its status changes to Draft.
Validate?	Selection indicates that the approval request must be validated against the values in the Validation Values column. If the approval request does not pass the validation, this action button will be disabled.
Default Submit?	When in Multiple Issue Approval mode, selection indicates that this action will be executed for the child request when the Submit for the original record is executed.
Validate SCM?	Selection indicates that the approval request must be validated against the values that users entered in the SCM Profile section on the Approval Request Details page.

Field/Column	Description
	Note: There must be at least one entry in the SCM Profile section. If there are no entries, the button will still be disabled even though there are no SCM validation errors.
Approve?	Selection indicates that the action that is set in the Action Name field is an Approval action. The approval requests that have an action that has Approve? selected and meet the conditions in this workflow step for the currently logged in user will be displayed in the Pending My Approval list on the Home page.
Email To	The role or roles to receive the notification emails once this action is executed.
Email Cc	The role or roles of any other recipient to receive the notification emails once this action is executed.
Email Subject	The template of email subject to be used to send email to the owner. For more information, see *Specifying Messages*.
Email Body	The template of email body to be used to send email to the owner. For more information, see *Specifying Messages*.
Action Id	The REST API action ID as a unique identifier for this profile's action. This field is read-only and populated after the action has been initially saved. It can be used by external systems to execute an action through a SAM REST API. For example, if the current action is Save Draft, the administrator can call a REST API that executes the Save Draft action on a particular request. For more information, see *REST APIs for SAM*.

The following table describes the fields in the Update section.

Field/Column	Description
Update Type	The type of system that will be updated – either Issue Tracking or SCM. • For Issue Tracking systems, SAM will update the specified field to a given value. • For SCM systems, SAM will execute the REST API defined in this system's Write/Delivery Method. Note that if you have multiple update entries for the same system, it will only execute once, passing the arguments specified in the different update entries in that single call.
External System	The name of the system that will be updated. Only systems that have a Write/Delivery Method defined in the System Administration page are shown here.
Field	• Issue Tracking systems: The field (belonging to the selected External System) that will be updated. • SCM systems: The name of the SCM field/argument (as defined in the System Administration page) to pass to the Write/Delivery Method.
Value Type	The Exact Value or Message (see the following for Value).

Field/Column	Description
Value	• Issue Tracking systems: If the Value Type is set to Exact Value, then the value used to update the selected Field is the exact string that is entered here is. If the Value type is set to Message, then the value used to update the selected Field is the evaluated message (as defined in Message Administration) that is selected here. For example, this section enables users to write comments or status back to the third-party issue tracking systems. For example, if, o The action name is Submit for Approval. o The update type is Issue Tracking. o The external system is Sample Bug System. o The field is Description. o The value type is Exact Value. o The value text is comment This bug has been submitted. When users click the Submit for Approval button on the Approval Request Details page, the SAM application executes the Sample Bug System's Write/Delivery REST API to update the Description field to This bug has been submitted. • SCM systems: If the Value Type is set to Exact Value, then the value that is sent as an argument is the exact string that is entered here is. If the Value type is set to Message, then the value that is sent as an argument is the evaluated message (as defined in Message Administration) that is selected here. For SCM delivery using SCM entry details (as collected in the Approval Request page), the built in field called source is to be defined in Message Administration screen. This Message is then selected to make the value passed to the Field selected. **Note:** This feature is only supported in 2.17.10 and onwards. • For example, to create deliver a workspace change upon a request's approval, do the following: Prerequisites: o You have configured the Write/Delivery Method and Fields as described in the System Administration example. o You have created the message Source for Workspace Delivery with the text {source}. a. Add a new update row. b. In the new row, for Update Type, select SCM. c. For External System, select your SCM system. d. For Field, select the SCM field that will be mapped to the source. e. For Value Type, select Message. f. For Value, select Source. **Note:** Note that if you have more than one update entry specified for a single SCM system, that system's Write/Delivery API will be invoked only once, passing all of the fields specified in the various update entries in a single URL call.

To add a new approval profile action or modify an existing one

1. From the SAM Home page, select the Main Menu button.
2. Select the Administration option.
3. Select the Approval Profiles option from the Administration drop-down list.

   The Approval Profiles page appears.
4. Scroll down to the Workflow section and click the Add Action button (the plus sign that is next to the action in the Actions column) for the starting status that you want to add actions.

   Alternatively, to modify an existing approval action, click the action link for the starting status that you want to modify.

   The Approval Profile Actions dialog box appears.
5. Enter an action name in the Action Name field.

   The action name will be the name of the action button that is displayed in the Create Approval Request page.
6. In the Result Status drop-down list, select an option from the list of available statuses.
7. Optionally, define the following values on the Approval Profile Actions page.

   a. Select the Validate option to indicate that the approval request must be validated against the values in the Validation Values column. If the approval request does not pass the validation, this action button will be disabled.

      Notice that the Validation Values and the Validation Messages columns are in the Fields section on the Approval Profiles page.
   b. Select the Default Submit option when in Multiple Issue Approval mode to indicate that this action will be executed for the child request when the Submit for the original record is executed.
   c. Select the Validate SCM option to indicate that the approval request must be validated against the values that users entered in the SCM Profile section on the Approval Request Details page.

      > **Note:** There must be at least one entry in the SCM Profile section. If there are no entries, the button will still be disabled even though there are no SCM validation errors.

   d. Select the Approve option to indicate that the action that is set in the Action Name field is an Approval action.
   e. Enter the role names into the Email To and Email CC fields so the users with those roles will receive the notification emails once this action executed.
   f. Select the email subject and email body using the template in the Email Body and Email Subject fields. For more information, see *Specifying Messages*.
   g. For existing actions, take note of the Action Id value if you will be using the SAM API to execute this action via a REST call.
   h. Define the values in the Update section to specify updates to the issue tracking systems or run scripts on the SCM systems that are executed once the users perform this action. For more information on Update section fields, see *Setting Up Approval Profile Actions in the SAM Workflow*.
8. Click the Save button.

Configuring Multiple Issue Approval Mode

This is a special request submission mode that is available within SAM. The Multiple Issue Approval allows users to submit requests for multiple issues at once if they are all in the same hierarchy.

Note: The Multiple Issue Approval mode is applicable only if there is a value in the Issue Tree Open Statuses field. Otherwise, it will operate in the normal mode of submitting one issue at a time.

Identifying issues in the same hierarchy

On the Profile Admin page,

- In the Issue Tree Open Statuses field, a comma-delimited list of status values is specified that indicate an Open status with the primary issue tracking system.
- In the Issue Tree Closed Statuses field, a comma-delimited list of status values is specified that represent a Closed status.

There is a special field in an issue tracking system to hold a Parent Id that is the Id of its parent issue.

In the Create Approval Request page when an issue Id is entered, REST API connected to SAM will use the Parent Id field and find all issues in the primary issue tracking system that are in the same hierarchy and have one of the statuses defined in the Issue Tree Open Statuses field. These issues are further filtered in SAM.

For example, if there are 6 issues identified,

Issue Id	Parent Id	Status
1	2	Open
2	3	Open
3		Closed
4	3	Awaiting review
5	6	Open
6		Open

and if the Issue Tree Open Statuses is set to Open, Awaiting review,

in the Create Request page when the issue Id 2 is entered, then REST API connected to SAM would find all issues in that hierarchy:

- It would find issue 1 because 1 is a child of 2

- It would find issue 3 because 3 is 2's parent
- It would find issue 4 because 4 shares the same parent as 2
- It would not find issues 5 or 6 since they are not related at all to 2

So 1, 2, 3 and 4 are found by REST API. However only 1, 2 and 4 have a status value listed in Issue Tree Open Statuses, so 3 is ignored. Hence, REST API connected to SAM finds issues from 1-4, SAM filters out issue 3 and SAM shows issues, 1, 2 and 4. For more information, see *REST APIs for SAM*.

Submitting Multiple Issues

- On the Create Approval Request page, all issues are displayed in the tree with an open status.
- Each issue is collapsed in its own section on the page.
- Only the fields that are flagged as Header on the Profile administration page are displayed in the head of the collapsed record.
- When a collapsed section is expanded, only the fields with Default View set on the Profile administration field are displayed.
- A Close button is displayed on each of the collapsed records that can be used to close that issue.
- On clicking the Close button, it allows to pick one of the status values defined in the Issue Tree Closed Statuses on the Profile administration page. After selecting a value, when the submit button is clicked, the status field of that issue tracking system record is updated to the selected closed value.
- The Refresh icon can be used to refresh the page. The closed record no longer appears on the page.
- A drop-down list is displayed for each record that will allow to specify which profile to use for that approval request since they may be different for each issue in the hierarchy.

Events after Submission for Multiple Issues

- When the approval request is submitted, SAM will submit approval requests for all of the records on the page (assuming none have validation errors).
- For each of the issues found (not including the issue that was initially entered on the page), SAM will look at its selected profile and find the available action that has the Default Submit flag set and execute that action for the request.

  Note: The issue initially entered on the page is not included because, for the original issue, it will use the profile that was selected when launching this page.

- Approval requests are created for each of the issues involved and approvals are to be done individually for each issue. Multiple Issue Approval mode is only for bulk submission and not for bulk approval.

Deploying and Configuring Workspaces

Deploying workspaces is a two-step process.

1. Configuring SAM to setup and deliver workspace

2. Deploying workspace file

Configuring SAM

The following information describes the process of configuring SAM.

To setup workspace delivery

1. Set up your workspace system in SAM.

   Refer to *To set up an SCM third-party system*.
2. Configure messages. Add two new messages for the SCM system. .

   Refer to *Specifying Messages*

   - For the first message, create message Approval ID for Workspace delivery with text {APPROVAL_ID}.
   - For the second message, create message Source for Workspace delivery with text {source}.
3. Add one new status for Pending Delivery and optionally two additional statuses for Delivery Completed and Delivery Failed.

   Refer to *Defining the Approval Request Statuses*.
4. Open the Approval Profile screen and locate the profile that you want to configure workspace delivery.

   a. Add a workflow according to your process step, for example an approval for delivery.
   b. Add an action by clicking the Plus (+) sign under Action(s) header for the newly created workflow.
   c. In the Approval Profile Actions pop-up window, select Pending Delivery in the Result Status drop-down list. Configure other fields.

      Refer to *Setting Up Approval Profile Actions in the SAM Workflow*.
   d. Add update entries for Action(s) as following in the next part of the Approval Profile Actions to initiate the delivery.

      Create the following update entry to pass the approval ID to the workspace API:

      - Update Type: SCM
      - External System: The Workspace system you configured
      - Field: requestid
      - Value Type: Message
      - Value: Approval ID for Workspace delivery

      Create the following update entry to pass the workspace name to the workspace API:

      - Update Type: SCM
      - External System: The Workspace system you configured
      - Field: wsname
      - Value Type: Message
      - Value: Source for Workspace delivery
5. This is an optional step for receiving a post-delivery action in SAM. You can add a success action and failure action to flip a success and failure status in SAM request. These actions can be additionally configured to record a message or flip a status on the Issue Tracker.

To receive a delivery status update, add a new workflow with Starting Status as Pending Delivery and two Action Result Status as Delivery Successful and Delivery Failed.

The Action IDs of the configured status Approval Profile Actions must be used when configuring the workspace REST API. For more information, refer to Step 4.

> **Note:** Make sure you have a SAM user and role defined for the user who has the action ownership. Refer to *Adding User Roles and Assigning Users to User Roles*. This user must also be configured in workspace.properties as value for jdbc.samUsername. In this example user is deliv@oracle.com and role is Delivery Agent. Refer to Step 4.

Deploying Workspaces

The following information describes the process of deploying workspaces.

To deploy workspace file

1. Find the workspace.war file from the SAM installed folder and rename it to workspacedelivery.war.
2. Copy the workspacedelivery.war file to Tomcat Webapps folder. This deployment is done on the server where Siebel Tools is installed.

   > **Note:** If SAM is not setup in this server, then copy US_export_policy.jar and local_policy.jar from the bin folder in the SAM installed folder and place it in lib/security under your Java home.

3. Change the content of <Siebel Tomcat home>\webapps\workspacedelivery\resources\properties \workspace.properties to match the workspace database connection. Choose from the following properties based on your database type:

   Oracle DB

   ```
   jdbc.driverClassName=oracle.jdbc.driver.OracleDriver
   jdbc.url=jdbc:oracle:thin:@<server name>:<portnumber>/<instance name>
   jdbc.username=<TBLO>
   jdbc.password=<encrypted password>
   jdbc.tblo=<TBLO>
   ```

   DB2 UDB

   ```
   jdbc.driverClassName=com.ibm.db2.jcc.DB2Driver
   jdbc.url=jdbc:db2://<server name>:<portnumber>/<DB Alias>
   jdbc.username=<TBLO>
   jdbc.password=<encrypted password>
   jdbc.tblo=<TBLO>
   ```

   DB2 390

   ```
   jdbc.driverClassName=com.ibm.db2.jcc.DB2Driver
   jdbc.url=jdbc:db2://<server name>:<portnumber>/<DB Alias>
   jdbc.username=<database user>
   jdbc.password=<encrypted password>
   jdbc.tblo=<TBLO>
   ```

MS SQL DB

```
jdbc.driverClassName=com.microsoft.sqlserver.jdbc.SQLServerDriver
jdbc.url=jdbc:sqlserver://<server name>\\<instance name>:<portnumber>;databaseName=<TBLO>
jdbc.username=<TBLO>
jdbc.password=<encrypted password>
jdbc.tblo=<TBLO>

jdbc.intBranchName=<workspace branch where you want to deliver your changes>
jdbc.siebelUsername=<Siebel user name>
jdbc.siebelPassword=<Encrypted Siebel password>

jdbc.logPath=<Create a local folder for API to write log files to, like C:/workspace>
jdbc.toolsPath=<Siebel Tools path, eg: C:/Siebel/18.0.0.0.0/18_03_Tools>
jdbc.siebelDSN=<Siebel DSN>
```

Note: Place a copy of your DB2 license jar file (db2jcc_license_cisuz.jar) to your Siebel Tomcat lib folder and bounce Tomcat server in case your Siebel DB is DB2 UDB or DB2 390 (z/OS).

4. This step is optional. Use this for workspacedelivery REST API to call back SAM to record success and failure actions.

```
jdbc.samURL=http://<Server name>:<portnumber>/sam
jdbc.samUsername=<A user who has a role which has ownership to do success and failure actions for
example delivery@<emaildomain.com>
jdbc.samPassword=<Encrypted SAM password>
jdbc.samSuccessActionID=<Action Id for post successful delivery>
jdbc.samFailureActionID=<Action Id for post failed delivery>
```

Note: To encrypt password, use the link http://<server name>:<port number>/emailapprover/encrypt/<password to encrypt>

5. Restart Tomcat server and open the link http://<servername:portnumber>/workspacedelivery/ in any browser to see the welcome message, Welcome to Siebel Workspace Delivery Adaptor!!!

Note: Restart Tomcat server only after finalizing the configuration or making any changes during configuration and testing.

To test the workspace delivery REST API

You may use Postman or any similar utility to test the API functionality before configuring SAM. After bringing up Workspacedelivery application as mentioned in step 1 through 4 (You can leave the optional section for the testing), use the Postman to do a POST action on the URL http://<server name>:<port number>/workspacedelivery/workspace/. Provide the following raw JSON content after altering the value according to your workspace name and request id.

```
{
 "wsname":"test",
 "requestid":"test"
 }
```

This would start a Siebel Tools instance and it would try to deliver the workspace. You will get a successful delivery message or a failure message accordingly. If you find any issues, check the workspace.properties, make corrections, and bounce Tomcat before you retest delivery.

Serialized Delivery of Workspaces

Delivery of workspaces can be serialized from Siebel Application Manager using a few steps and scripts. This will allow serial delivery of more than one workspace concurrently submitted for delivery, not requiring manual submission of each delivery or wait for the ongoing deliveries.

Note: You can set up the delivery schedule outside of SAM from a scheduled Windows task, a Cron job, or any other method to run a scheduled process. Amend the script if there is a delivery failure.

To set up asynchronous serialize workspace delivery,

To set up asynchronous serialize workspace delivery

1. Set up a delivery schedule.
2. Query for approval requests by Profile ID and Status. For example:

   Profile ID: “All requests in Release 19.10”

   Status: “Pending Delivery”
3. Call the following API that is in-built to query the list of requests and the associated workspaces.

   http://<sam server:port number>/sam/api/approvals/

   This is a POST end point.
4. You can POST the query criteria in JSON format. Use the following steps to find the values for the query.
   a. Make a sample request in SAM using a SAM profile and take the approval process request to the desired status of delivery.
   b. To retrieve Row ID, make a POST call to the API in Step 2 using the sample Request Id as ‘row_id’ value as in the following JSON,

      ```
      {
      "row_id":"88-5E29P"
      }
      ```

      Tip: You can find the Request Id from the browser address bar by drilling down in your sample request, for example, http://<sam server:port number>/sam/api/approvals/88-5E29P.

   c. Make a note of the Profile ID that the JSON output brings up on successful POST made in step 4b. For example:

      ```
      {
      "message": "",
      "data": [
      {
      "approvalId": "88-5E29P",
      "seedIssue": "Y",
      "profileId": "88-5E1NS",
      "defaultActionId": null,
      "closeStatus": null,
      "fieldsAndData": [
      .
      .
      ```

```
.
],
"buttons": null,
"scmSystems": [
],
"approvalAuditTrail": [
]
}
]
}
```

d. To retrieve the list of statuses and their Ids, make a GET call to http://<sam server:port number>/sam/status/ for the status of your sample request. For example:

```
{
"statusId": "88@30993086",
"statusName": "Ready for delivery",
"active": "Y",
"lastUpdated": 1484249924000
},
```

e. Construct a final POST query JSON for the main API, http://<sam server:port number>/sam/api/approvals/ with the following values,

```
{
"PROFILE_ID":"88-5E1NS ",
"APPROVAL_STATUS_ID":"88@30993086"
}
```

This will return an array of JSON data as following,

```
{
"message": "",
"data": [
{
"approvalId": "88-5E29P",
"seedIssue": "Y",
"profileId": "88-5E1NS",
"defaultActionId": null,
"closeStatus": null,
"fieldsAndData": [
.
.
.
],
"buttons": null,
"scmSystems": [
{
"sdId": "88-5E7NM",
"approvalId": "88-5E29P ",
"systemId": "88@30993101",
"systemName": "Workspace",
"source": "\"dev_test_1\"",
"valid": "Y",
"validMsg": null,
"deliveryStatus": null,
"deliveryMsg": null
},
{
"sdId": "88-5E7NM",
"approvalId": "88-5E29P ",
"systemId": "88@30993101",
"systemName": "Workspace",
```

```
"source": "\"dev_test_2\"",
"valid": "Y",
"validMsg": null,
"deliveryStatus": null,
"deliveryMsg": null
}
],
"approvalAuditTrail": [
]
},
{
(2nd array element of JSON representing the second request.)
},
{
(3rd array element of JSON representing the third request.)
}
]
}
```

5. Iterate through each of the requests for workspace information.

   a. Locate the element “scmSystems” and copy the “source-value” in the array of JSON for requests queued up for delivery. The “scmSystems” may have multiple workspaces in JSON array if your request has multiple workspaces such as dev_test_1 and dev_test_2 in the sample JSON data in step 4.

   b. For each node of requests (a node has multiple calls within each request with more than one workspace in the same request) make POST calls to http://<workspace delivery API hosted server:port number>/ workspacedelivery/workspace/ and pass the following JSON,

```
{
“wsname”: “workspace name found from ‘source: value’ in step above”,
“requestid”: “approvalid found from the step above”
}
```

   For example:

```
{
“wsname”: “dev_test_1”,
“requestid”: “88-5E29P”
}
```

6 REST APIs for SAM

REST APIs for SAM

Siebel Approval Manager (SAM) has a plug-n-play architecture using REST APIs. REST APIs are used to feed data from external systems into SAM as well as write data back to those external systems. These external systems can be either Issue Tracking systems like Bugzilla or SCM (Software Configuration Management) systems like GitHub or Siebel Workspace.

GET REST API is used for reading (specified as the Read/Update property in SAM System administration), providing the query data element part of the URI. POST REST API is used for writing (specified as the Write/Deliver property) of any data change that need to be pushed to external systems.

Note: The GET/POST end-points should be open without any authentication.

This chapter describes sample REST APIs. This chapter includes the following topics:

- *REST API GET URL Sample Format for Issue Tracking Systems*
- *REST API GET Method's JSON Format for Issue Tracking Systems*
- *REST API POST URL Sample Format for Issue Tracking Systems*
- *REST API POST Method's JSON Format for Issue Tracking Systems*
- *Importing System Fields*
- *REST API GET URL Sample Formats for SCM Systems*
- *REST API GET Method's JSON Format for SCM Systems*
- *POST REST API Exposed by SAM for Workflow Actions*
- *GET REST API Exposed by SAM to Retrieve Status Types*
- *GET REST API Exposed by SAM to Retrieve User Types*
- *GET REST API Exposed by SAM to Retrieve List of Profiles*
- *REST API Exposed by SAM to Retrieve Approval Requests*

REST API GET URL Sample Format for Issue Tracking Systems

The sample format is as follows:

http://<server ip>/services/bugzilla/<issue ID>

REST API GET Method's JSON Format for Issue Tracking Systems

The GET method exposed by an external system's REST API needs to follow a certain JSON format, which is readable by SAM. The JSON must have a 'key identifier' property representing an issue's unique identifier (such as Bugzilla's bug number) and a 'key status' property which represents the lifecycle state of the issue (such as Bugzilla's status field).

GET JSON needs to be an array with single or multiple elements.

JSON sample:

```
[
 {
 "issueId": "17788411",
 "status": "Approved, waiting for codeline to open",
 "priority": "1",
 "customer": "INTERNAL",
 "severity": "2",
 "publish": "N",
 "statusId": "39"
 },
 {
 " issueId ": "22809179",
 "status": "Closed, Not a Bug",
 "priority": "5",
 "customer": "INTERNAL",
 "severity": "2",
 "publish": "N",
 "statusId": "92"
 },
 {
 " issueId ": "22809190",
 "status": "Closed, Not a Bug",
 "priority": "5",
 "customer": "INTERNAL",
 "severity": "2",
 "publish": "N",
 "statusId": "92"
 }
]
```

Note: "issueId" is the 'key identifier' and "statusId" is the 'key status' property in the previous sample.

REST API POST URL Sample Format for Issue Tracking Systems

The sample format is as follows:

http://<server ip>/services/issue/

REST API POST Method's JSON Format for Issue Tracking Systems

The JSON format expected by POST should also follow a certain format. The JSON must have a 'key identifier' representing the issue's key that GET. POST JSON need to be a simple structure without any array.

JSON sample:

```
{
" issueId ": "22809190",
"publish": "N",
"statusId": "92"
}
```

In the previous sample "issueId", which is the 'key identifier', will help the POST end point identify which Issue/Bug is going to get updated. The other two fields, "publish" and "statusId", are the two fields that are going to get updated with values supplied in the JSON.

Importing System Fields

A static GET end point has to be used to import field metadata that the 'Issue Tracking System' supports. This end point will bring two properties namely, 'fieldName' and 'fieldId'. 'fieldName' represents the name of the field and 'fieldId' represents a unique identifier which is also used for order of the fields while being imported in SAM. All the available fields in GET and POST mentioned earlier need to be represented in the JSON output.

URL: http://<server ip>/services/bugzilla/fields/

JSON sample:

```
[
 {
 "fieldName": "issueId",
 "fieldId": "0"
 },
 {
 "fieldName": "status",
 "fieldId": "1"
 },
 {
 "fieldName": " priority ",
 "fieldId": "2"
 },
 {
 "fieldName": " customer ",
 "fieldId": "3"
 },
 {
 "fieldName": " severity",
 "fieldId": "4"
 },
 {
 "fieldName": " publish",
 "fieldId": "5"
```

```
 },
 {
 "fieldName": " statusId ",
 "fieldId": "6"
 }
]
```

REST API GET URL Sample Formats for SCM Systems

Configured URL in SAM: http://<server ip>/services/github/<source>

URL runtime formation sample:

```
http://<server ip>/services/github/dev_fix_for_inline_edit
```

source = dev_fix_for_inline_edit

Note: For the branch name, SAM has a built in placeholder called 'source'. If there are additional data elements like version or repository, those can be configured by providing static elements.

For example if the URL needs 'repository' and 'version' (in addition to the built-in branch name placeholder for 'source') then providing the element in the following format is required.

Sample URL:

Configured URL in SAM: http://<server ip>/github/repos/<repository>/<version>/commits/<source>

URL runtime formation sample:

```
http://<server ip>/github/repos/siebelsam/sam-demo/commits/dev_fix_for_inline_edit
```

repository = siebelsam

version = sam-demo

source = dev_fix_for_inline_edit

REST API GET Method's JSON Format for SCM Systems

GET method for SCM systems is a strictly formatted set of properties. The information required for GET URI is passed via URI format.

```
{
 "valid": "N",
 "message": "Workspace status is invalid."
}
```

The property 'valid' can have either 'Y', which stands for a valid branch, or 'N', which stands for an invalid or an already closed branch. 'Message' can contain a string with the reason for declaring the branch invalid or any message for valid branch (such as list of affected files, etc.).

POST REST API Exposed by SAM for Workflow Actions

SAM provides a built-in POST REST API to do Approval Request actions.

Sample:

http://<server id>/sam/api/approvals/88@93?actionId=88@85

'88@93' in the previous URL represent 'Approval Id' and '88@85' represents the 'Action Id', which will be performed. The 'Action Id' can be found in the Action Id field of the Workflow Actions popup on the Profile administration screen within SAM.

Output:

The 'message' contains a string explaining the outcome, either that the action is completed or was not able to be performed because of some reason. The 'data' property will always be empty. The positive output will return 200 as its status code.

```
{
 "message": "Approval Request status changed to Draft",
 "data": null
}
```

The negative output will return 409 or 500 as status code.

```
{
 "message": "Approval Request status cannot be changed to null",
 "data": null
}
```

For more examples, see *https://github.com/OracleSiebel/ConfiguringSiebel/tree/master/ExampleCode/SAM*.

Perform the following to update the status on workspace delivery.

1. Find the action Id required for status update. To find,
    a. Login to SAM as administrator.
    b. Go to Main menu, Administrator.
    c. Click to drill down to the profile.
    d. Find the Action under Workflow section.
    e. Click to bring up the Action pop-up screen to find the Action Id in the Approval Profile Actions screen.
2. Flip SAM status post-delivery using the Action Id found in step 1. Do the following,
    a. Make the content type as application/json.
    b. Provide authentication details of the user who is part of the action owner role.

    Make a POST call to http://<SAM server>/sam/api/approvals/88@30994552?actionId=88@30991159. Output appears as below in Postman.

```
{
"message": "Approval Request status changed to Ready for Build",
"data": null
}
```

GET REST API Exposed by SAM to Retrieve Status Types

SAM provides a built-in GET REST API to access a list of all of the configured approval statuses.

To get list of statuses, make a GET call to http://<SAM server>/sam/status using the Postman application.

The JSON output will contain information such as,

```
{
"data": [
{
"statusId": "88@30993086",
"statusName": "Draft",
},
{
"statusId": "88@30993089",
"statusName": "Pending Manager Approval",
},
{
"statusId": "0CX@32325368",
"statusName": "New",
}
]
}
```

GET REST API Exposed by SAM to Retrieve User Types

SAM provides a built-in GET REST API to access a list of all of the configured users.

To get list of user types, make a GET call to http://<SAM server>/sam/users.

The JSON output will contain information with their Ids in the property named ‘uld’ such as,

```
{
"uId": "0CX@32325363",
"userId": null,
"userName": "Admin",
},
```

GET REST API Exposed by SAM to Retrieve List of Profiles

SAM provides a built-in GET REST API to access a list of all of the configured approval request profiles.

To get the list of profiles, make a GET call to http://<SAM server>/sam/profiles.

The JSON output will contain information such as

```
{
```

```
"profileName": "2.17.x",
"profileId": "88@30990605",
…
```


REST API Exposed by SAM to Retrieve Approval Requests

APIs are built for searching SAM requests. The SAM requests could be the following:

- *Query based on a single approval request* by its approval Id.
- *Query based on multi-query request tokens* like Profile Id, Status and so on.

Query Based on a Single Approval Request

The SAM provides a built-in GET REST API to access Approval Request information.

Sample:

Note: For readability only two fields are provided in the output.

http://<server ip>/sam/approvals/88@93

'88@93' in the previous URL is Approval Id.

Output:


```
{
 "message": "",
 "data": [
 {
 "approvalId": "88@93",
 "seedIssue": "Y",
 "profileId": "88@30990610",
 "defaultActionId": null,
 "closeStatus": null,
 "fieldsAndData": [
 {
 "fieldmapId": "88@30990653",
 "genericFieldId": null,
 "genericFieldLabel": null,
 "genericFieldName": "APPROVAL_ID",
 "extFieldName": "APPROVAL_ID",
 "overrideLabel": "Approval Id",
 "fieldValue": "88@93",
 "valid": null,
 "drillDownURL": null,
 "defaultView": "N",
 "header": "N",
 "validationText": null,
 "keyField": "N",
 "systemId": "OCX@32325372",
 "systemname": "System",
 "displayOrder": 1
 },
 {
```

```
"fieldmapId": "88@30990654",
"genericFieldId": null,
"genericFieldLabel": null,
"genericFieldName": "APPROVAL_STATUS",
"extFieldName": "APPROVAL_STATUS",
"overrideLabel": "Approval Status",
"fieldValue": "Draft",
"valid": null,
"drillDownURL": null,
"defaultView": "N",
"header": "N",
"validationText": null,
"keyField": "N",
"systemId": "0CX@32325372",
"systemname": "System",
"displayOrder": 2
},
],
"buttons": null,
"scmSystems": null,
"approvalAuditTrail": [
{
"atId": "88@94",
"approvalId": "88@93",
"approvalStatusId": "88@30993086",
"approvalStatus": "Draft",
"comments": null,
"overrideManagerId": "null",
"overrideManager": null,
"updatedById": "88@30993489",
"updatedByEmail": "thirupathi.rajendran@oracle.com",
"updatedBy": "Thirupathi Rajendran",
"updated": 1501064367000,
"version": null
}
]
}
]
}
```


Query Based on Multi-Query Request Tokens

APIs are built for searching SAM requests based on multiple query tokens like Profile Id, Status and so on. You can use Google Chrome app, Postman or a similar HTTP client to test the REST APIs mentioned below.

To make a query request

To make a query request

1. Find the query token data that you need.

   The following could be some of the query tokens and the way to find them. The details are in the following topics.

   - STATUS_ID: Make a GET call on the *Status API* to find the statusId value that you want for a particular status.
   - MANAGER_ID and OWNER_ID: Make a GET call on the *Users API* to get all the users with their Ids in the property named 'uId'.
   - PROFILE_ID: Make a GET call on the *Profile API* to get the profileId of the profile to query on.

- ISSUE_ID: This is the actual Issue Id generated by your Issue Tracker.

2. Query using the data. Make a POST call to http://<SAM server>/sam/api/approvals.
   - Make the content type as application/json.
   - Provide authentication details of the user who has administrative privileges.
   - Provide the JSON content, like the sample here according to your requirement, in the request body.

     Sample of a single query criteria. This will return all of the approval requests created using the provided Profile.

     ```
     {
     "PROFILE_ID":"88@30990605"
     }
     ```

     Sample of a dual query criteria. This will return all of the approval requests created using the provided Profile and are in the provided Status.

     ```
     {
     "PROFILE_ID":"88@30990610",
     "APPROVAL_STATUS_ID":"88@30993086"
     }
     ```

     The output format is identical to the one listed for the single approval request.