SERENA

SERENA®

RELEASE MANAGER

Getting Started Guide

Serena Proprietary and Confidential Information

Trademarks

U.S. Government Rights

Product version: 4.5

Publication date: 2013-05-13

Table of Contents

Welcome to Serena Release Manager

Thank you for choosing Serena Release Manager, the orchestrated solution for application release management that enables you to plan, manage, and automate the deployment of applications into test, pre-production, and production environments across mainframe and distributed systems with start-to-finish traceability and end-to-end visibility.

Audience and Scope

This document is intended for personnel who participate in the processes of managing releases using Serena Release Manager.

For information on installing and configuring Serena Release Manager, see Serena ALM Installation and Configuration.

Before You Begin

See the Readme for the latest updates and corrections for this document.

Procedures and figures in this documentation are examples based on the default implementation of Serena Release Manager. The details may not match your implementation of Serena Release Manager exactly, but provide a reference to get you started with your use of Serena Release Manager.

Accessing the Documentation [page 7]

Accessing the Documentation

You can access all documentation and demonstrations for the current release of Serena Release Manager through the Serena ALM Help system.

When you click the **Help** link in Serena ALM, the Serena ALM online Help appears. You can navigate the contents, search, and view the glossary for information on installing, configuring, and using Serena Release Manager. To view and download readme files and PDFs for all supported releases, select **Demos & All Doc** in the online Help menu.

Serena Release Manager Concepts

Find out the benefits of using Serena Release Manager to plan, control, and automate the deployment of your releases. Gain an understanding of how Serena Release Manager works, including key concepts and terminology.

What is Release Management?

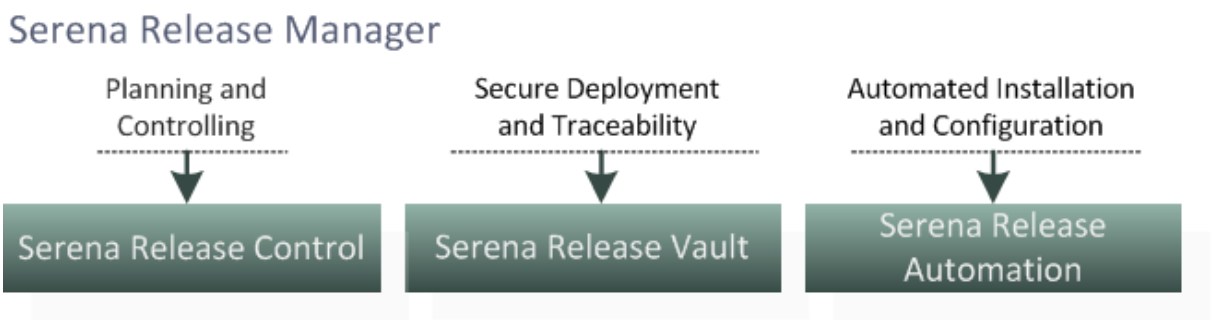
Release management is: "The definition, support, and enforcement of processes for preparing software for deployment into production". (Forrester Research, Inc.)

Release management manages the flow of change into production. It is the hand-off between development, quality assurance, and production operation teams.

The goal of release management is to deploy application changes into production with high quality and without disrupting the business. But this process is often performed manually and is inefficiently connected to the rest of the application lifecycle, leaving a critical gap between application development and operations as well as creating a backlog of changes that must be made.

Control, Secure, and Automate Your Releases

Serena Release Manager is part of the Serena Orchestrated Application Lifecycle Management (ALM) suite. Release Manager helps you manage and deploy releases by enabling you to control, secure, and automate your processes.



- **Serena Release Control**, powered by Serena Business Manager, helps you plan and control your application release processes across your enterprise, from definition to deployment.
- **Serena Release Vault**, powered by Dimensions CM for distributed systems and ChangeMan ZMF for IBM z/OS systems, ensures a secure and auditable path to production.
  - Secures source for multiple releases
  - Prevents unauthorized changes
  - Enables full traceability and audit trail

- **Serena Release Automation** and **Serena Release Automation, Powered by Nolio,** automate application installation and configuration tasks.
  - Handle a high volume of tasks
  - Cut deployment time and cost
  - Reduce deployment errors
  - Handle the complexity of multiple release destinations and configurations

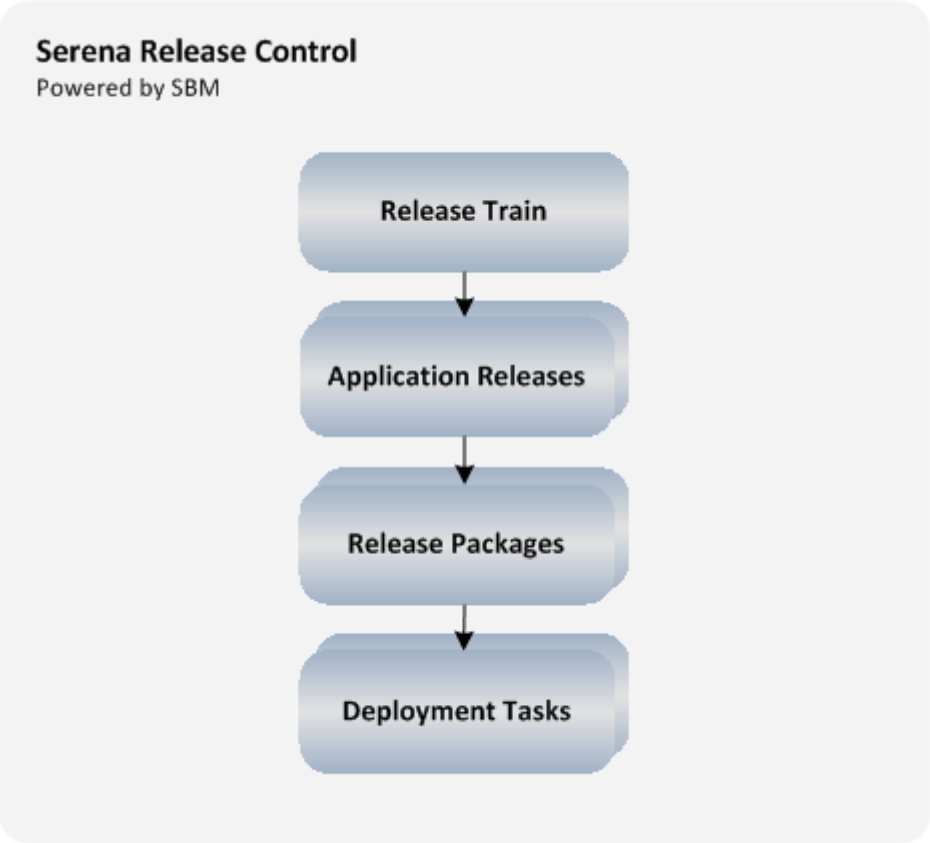
Serena Release Manager Terminology and Relationships

Before you use Serena Release Manager, you should know what the included objects are and how they are related as follows:

NOTE Terms used throughout this documentation may have slightly different meanings depending on the context. For example, the terms **project**, **environment**, and **application** have specific meanings in each of the products that participate in the Serena Release Manager suite. Please see the Glossary [page 76] for more complete contextual information.

Primary Object Hierarchy

Before you use Serena Release Manager, you should know the basic terminology as well as how the primary elements are related. The hierarchical relationships between the primary Serena Release Manager objects are shown and described in the following figure and text.



Release trains provide a published schedule of changes to production. One or more application releases are associated with each release train.

Application releases represent versions of applications or projects, where the application or project architecture is specified by components. One or more release packages are associated with each application release.

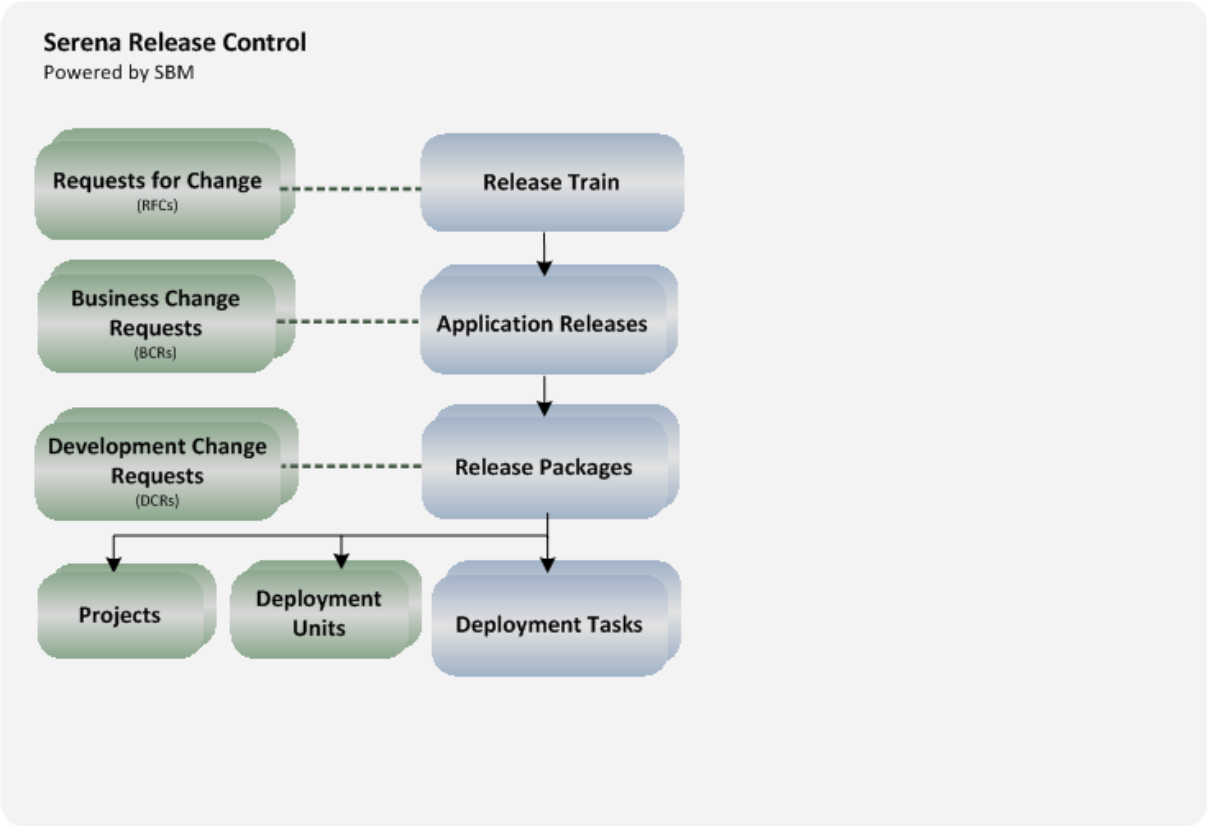
Release packages represent a portion of IT or service infrastructure normally built, deployed, tested, and released together. Release packages define the set of changes to be deployed and drive the deployment processes. One or more development change requests and deployment units are associated with each release package.

Deployment tasks are actions to be executed as part of the process of deploying a release package into a specific stage environment. Deployment task types include manual, vault, and automation.

How Integrating Objects Relate to the Hierarchy

Serena Release Manager brings in information from integrating providers as needed to support your release control processes. Providers implemented in the default implementation include SBM, Serena Service Manager, Dimensions CM, and ChangeMan ZMF. Custom providers can be added by Serena Services or others as documented in *Serena ALM Installation and Configuration*.

Objects that are populated from these integrating providers are as follows.



Requests for Change represent operational changes that may affect multiple applications or implement system infrastructure changes within an enterprise. These are associated with release trains and are typically associated with tickets from service management systems such as Serena Service Manager.

Business Change Requests represent customer or business unit change requests that affect specific application releases. They are associated with application releases and are typically associated with tickets from help desk or incident management systems such as Serena Service Manager or other SBM solutions.

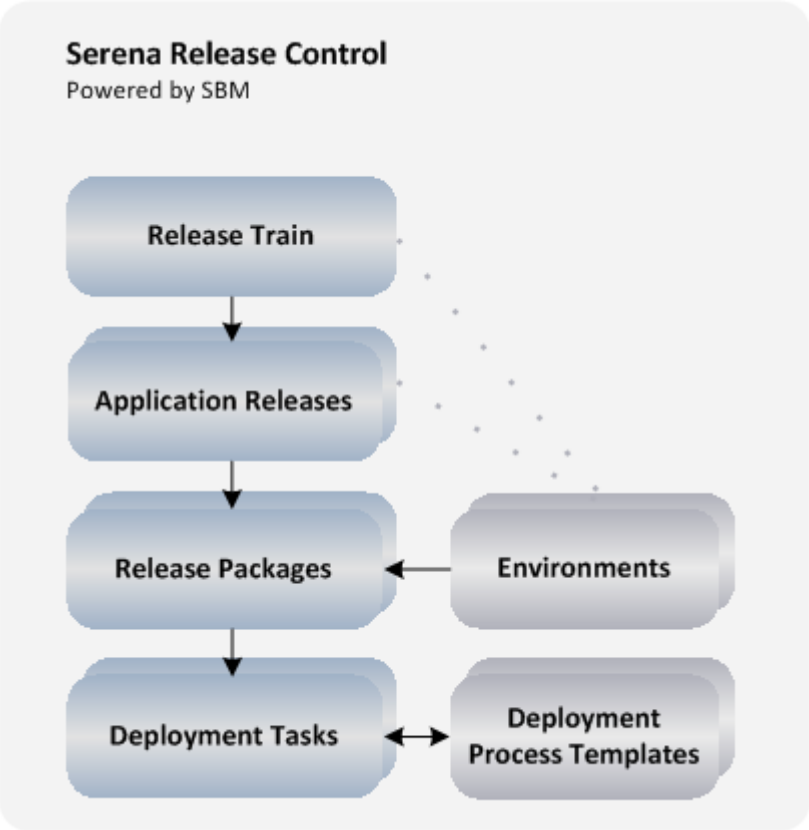
Development Change Requests represent delivered changes from the development process and affect specific release packages. They are associated with release packages and are typically associated with tickets from development management systems such as the SBM Change Manager solution or Dimensions CM request management.

Projects are the organizing entity that Serena Release Manager implementation uses to get a collection of related objects and associate them with the primary Serena Release Manager objects. For example, Dimensions CM, SSM, and SBM projects, and ChangeMan ZMF applications (projects) are used to get information on the other integrating objects.

Deployment Units represent a set of deployable components, such as Dimensions CM baselines with build outputs and ChangeMan ZMF change packages.

Process Support Objects

Serena Release Manager includes objects to support your release control processes. They are not part of the primary hierarchy, but are related to them in key ways.



Environments enable you to determine which physical or logical computers or systems are available for a release package deployment.

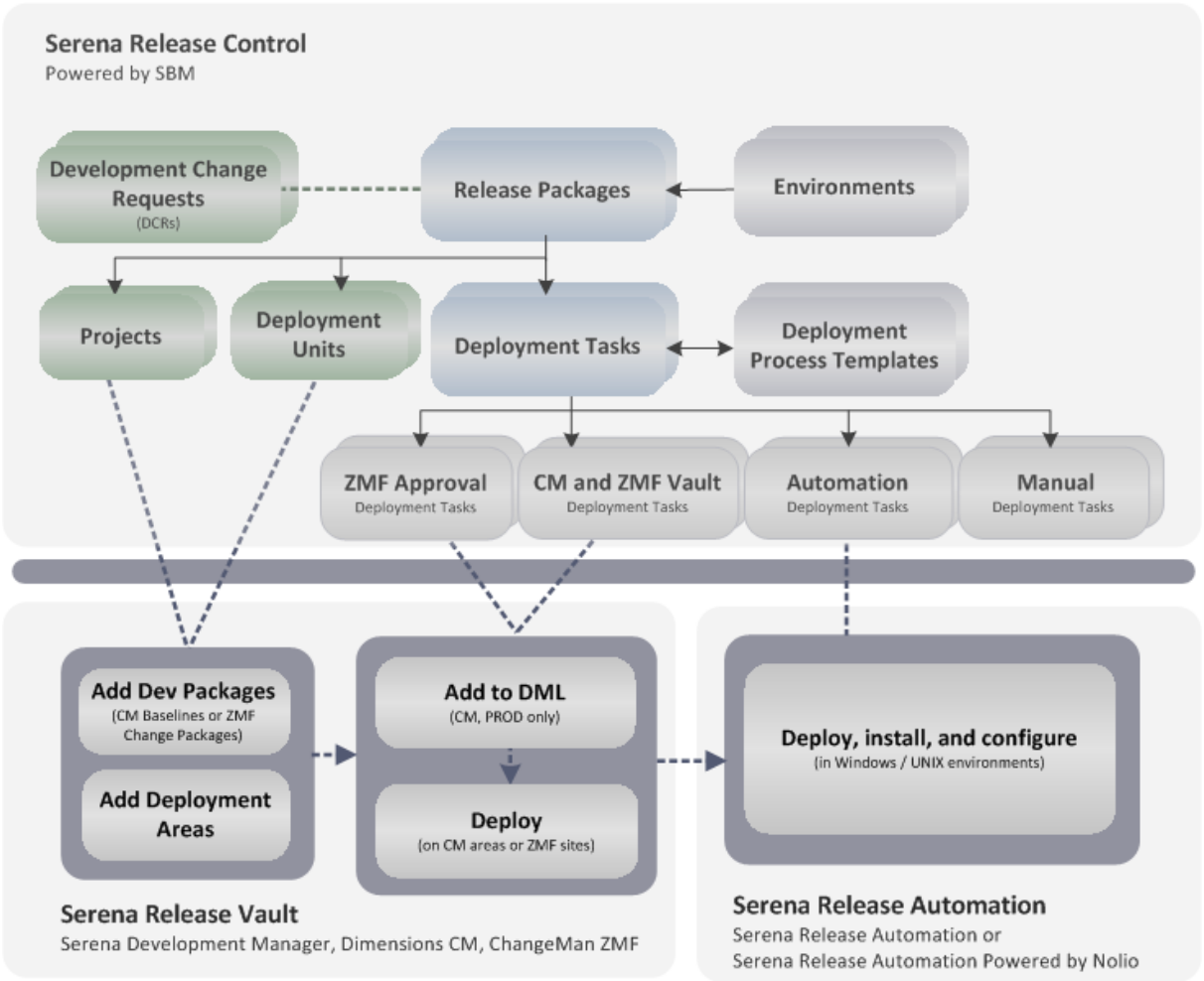
Deployment Process Templates enable you to create and copy sets of deployment tasks for reuse in different stages of release packages in the same or different application releases.

Deployment Relationships

To secure your release deployment and provide an audit trail of changes, Serena Release Manager uses the release vault features of Dimensions CM and ChangeMan ZMF. To automate your release installation and configuration tasks, the default implementation of Serena Release Manager uses Serena Release Automation, powered by Nolio.

Release Packages point to deployment tasks that integrate with release vault and release automation features of integrating products. These deployment tasks automate deployment, installation, and configuration tasks as part of the deployment process.

The flow of information between release control, release vault, and release automation is shown in the following figure.



Serena Release Vault is the location where all development tested versions of media, typically software or documentation, are stored. In the default implementation of Serena Release Manager, the provided release vaults are Dimensions CM and ChangeMan ZMF.

The vault is a single logical storage area and contains components that are not yet seen as definitive, in that they have yet to be approved for production use. For example they have not yet passed UAT.

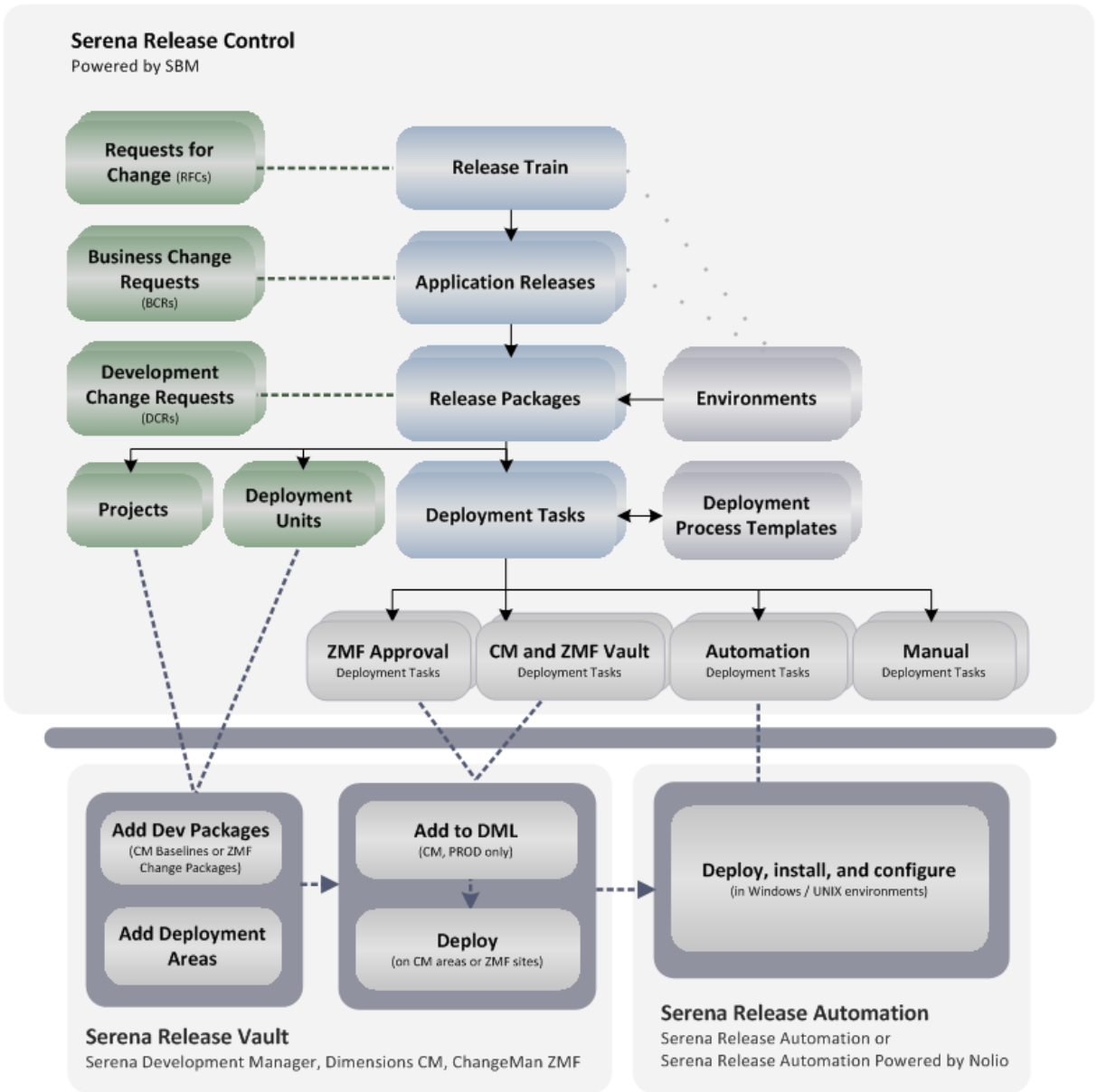
Components delivered to the vault must undertake multiple levels of testing. For example, when using Dimensions CM, if a specific baseline created in the vault fails testing, new versions of components must to be added into the vault and baselined. Once the baseline passes testing and is deployed successfully to the production stage, it is considered definitive.

The **release types** and **release stages** define the purpose of the environments where the **deployment units**, such as Dimensions CM baselines and ChangeMan ZMF change packages, are to be copied. The **deployment tasks** define other actions to be done for the deployment, such as approval, vault tasks, and automation tasks. Deployment tasks point to **deployment areas** in Dimensions CM and **sites** in ChangeMan ZMF to specify the exact destinations for the deployment units.

Serena Release Automation does the installation and configuration tasks once the deployment units are deployed to a specified deployment area or site for a release stage. For example, for each stage, it might update the registries, update configuration files, and restart Web services according to the deployment task definitions. Later, once a release has reached the production deployment stage, Serena Release Automation and/or Serena Release Automation, powered by Nolio, may be used to distribute files from a production environment to a large number of designated servers, updating the registries, updating configuration files, and restarting Web services for all of these.

Putting It All Together

Together, the primary and related objects of Serena Release Manager implement a rich release management solution that is highly configurable and customizable to meet your organization's release management needs. The full default Serena Release Manager hierarchy and related objects are shown in the following figure.



Process and Roles Within Release Manager

Process and roles within Serena Release Manager and the Serena ALM suite vary from one organization to another, but the common goal is to control the processes that get the work developers have done through a defined set of testing stages and over to IT Operations teams with confidence that the requirements and approved changes are addressed and contained in the released production package.

A typical flow of information, tasks, and automations to and from external systems (Dev and IT-OPs) is as follows:

1. Release Managers may work with executives, business analysts, development managers, and IT operations to lay out a schedule of release windows for the organization's IT changes for a future period of time, perhaps planning a year or two ahead. The Release Manager may create

release trains in Serena Release Control so that these release windows are laid out on the Serena Release Control calendar.

2. Organizational technical needs may lead to the creation of a *Requests for Change (RFC)*, which are formal proposals for a change to the IT infrastructure. The Release Manager may associate RFCs with a release train to target these changes for a specific release window, with contingency plans for moving RFCs to earlier or later release trains if necessary.

3. Release Managers may create *application releases* as part of the project or product planning of an application. These are planned to coincide with particular release trains, with contingency plans for moving application releases to earlier or later release trains if necessary.

4. Business Analysts may open *Business Change Requests (BCRs)*. BCRs typically target a specific set of changes for a specific release of a specific application. These are planned to be implemented as part of particular application release, with contingency plans for moving BCRs to a different release of the application if necessary.

5. Release Managers may create *release packages* for sets of executables and supporting files that will be deployed to various testing environments and progressed to production environments. The release engineers will populate these release packages as the development deliverables become available. These release packages are typically associated with an application release and are deployed according to the schedule of the parent release train.

6. Application defects and enhancements may be documented and tracked in *Development Change Requests (DCRs)*. DCRs may be associated with a release package to target a specific set of changes for a specific release package.

7. Release Engineers may create *deployment tasks*. As more information comes available, they may create more deployment tasks, change details, and so on. They may use deployment process templates to help manage and copy the deployment tasks.

8. Development teams resolve DCRs through their work on the software and unit test their work. When the unit testing is done and the software is ready for integrated testing, they create the release-ready baselines that are ready to be picked up by Serena Release Manager. The baselines are added to the deployment tasks as deployment units. Baselines may originate from Dimensions CM, Development Manager (DVM), or other providers defined to Serena Release Manager, or they may be introduced to Serena Release Manager from external systems using *Vault Requests*.

9. Deployment tasks are completed and updated as needed from one test stage to another. Development teams may deliver additional changes and new baselines may be selected for the deployment tasks. Release Managers or Release Engineers may *deploy to each pre-production testing stage* the release packages holding these deployment tasks and baselines. They may repeat a testing stage as many times as necessary or move to the next testing stage.

10. When the baselines have been fully tested in the pre-production testing stages, Release Managers may *deploy the release package to the production stage*.

11. Serena Release Manager maintains the history of the release activities, and Release Managers may copy the existing objects for future similar releases, making it easier to populate and repeat the release processes.

Progressing Releases Through Their Lifecycles

Each major aspect of a release, from the Release Train level to the Deployment Task level, has a lifecycle associated with it. The default release process lifecycles in Serena Release Manager are based on typical release workflows.

The states in the lifecycle guide the actions users take in Serena Release Manager as they work with release information. Your workflows will probably look different than the default implementation because most organizations customize the workflows in some way.

The Serena Release Control user interface leads you through the workflow, so you always know the state of the release objects and what actions are pending. To understand the processes underlying Serena Release Manager objects, see the following:

How the Workflows Interrelate

Serena Release Manager is a hierarchical system, with Release Train as the highest level organizing entity, Application Release next, and so on. Release Packages can optionally participate in the hierarchy or exist standalone. Deployment Tasks are dependent on Release Packages.

The two main dependency relationships among the workflows are restrictions and automations.

Workflow Restrictions

Most of the default workflows allow editing and associating objects to other objects only when items are in the Development or Planning states. For example, you can associate application releases with a release train only when that release train is in the Planning state. You can associate release packages with an application release only when the application release is in the Planning state.

Workflows for Deployment Process Templates are standalone, although deployment process templates populate the deployment tasks for release packages and in that respect have a link to the workflows for both Deployment Tasks and Release Packages. Workflows for environments are completely standalone by default.

Workflow Automations

Many of the workflows are moved to completion states automatically when associated objects are complete. The automation is usually initiated at the lowest level of association and moves up the hierarchy as each level of release activities are completed. For example, after all deployment tasks for a release package reach completion, the release package is marked complete. After all release packages for an application release reach completion, the application release is marked complete. When all application releases for a release train reach completion, the release train is marked complete.

For more details about Serena Release Manager workflow relationships and dependencies, see *Serena ALM Installation and Configuration*, "Workflow Reference".

Example Release Workflows

The workflows that are implemented in the default version of Serena Release Manager are shown in context with the procedures to which they are related. A reference to each of the workflows is given in the following table.

NOTE The implementation of the workflows in SBM include automatic transitions and states. For simplification, the representation of the workflows in this document shows only those transitions (actions) and states that require user interaction.

For the full SBM workflows for the Serena Release Manager process apps, see *Serena ALM Installation and Configuration*, "Workflow Reference".

Release Entity	Workflow Reference
Release Train	See Actioning Release Trains Through the Workflow [page 27].
Application Release	See Actioning Application Releases Through the Workflow [page 30].
Release Package	See Actioning Release Packages Through the Workflow [page 39].

Release Entity	Workflow Reference
Deployment Task	See Actioning Deployment Tasks Through the Workflow [page 51].
Deployment Process Template	See Actioning Deployment Process Templates Through the Workflow [page 58].
Environment	See Actioning Environments Through the Workflow [page 33].
Vault Request	See Actioning Vault Requests Through the Workflow [page 67].
Vault Template	See Actioning Vault Templates Through the Workflow [page 69].

Getting Started with Serena Release Control

Get started by familiarizing yourself with the user interface.

Logging In

You should have received a Serena ALM URL, username, and password from your Serena ALM or Serena Release Manager administrator.

To log in to the Serena ALM web client:

1. Enter the Serena ALM URL. It will look similar to the following:

   `http://almhost/tmtrack/tmtrack.dll?shell=alm`

   The login form appears.

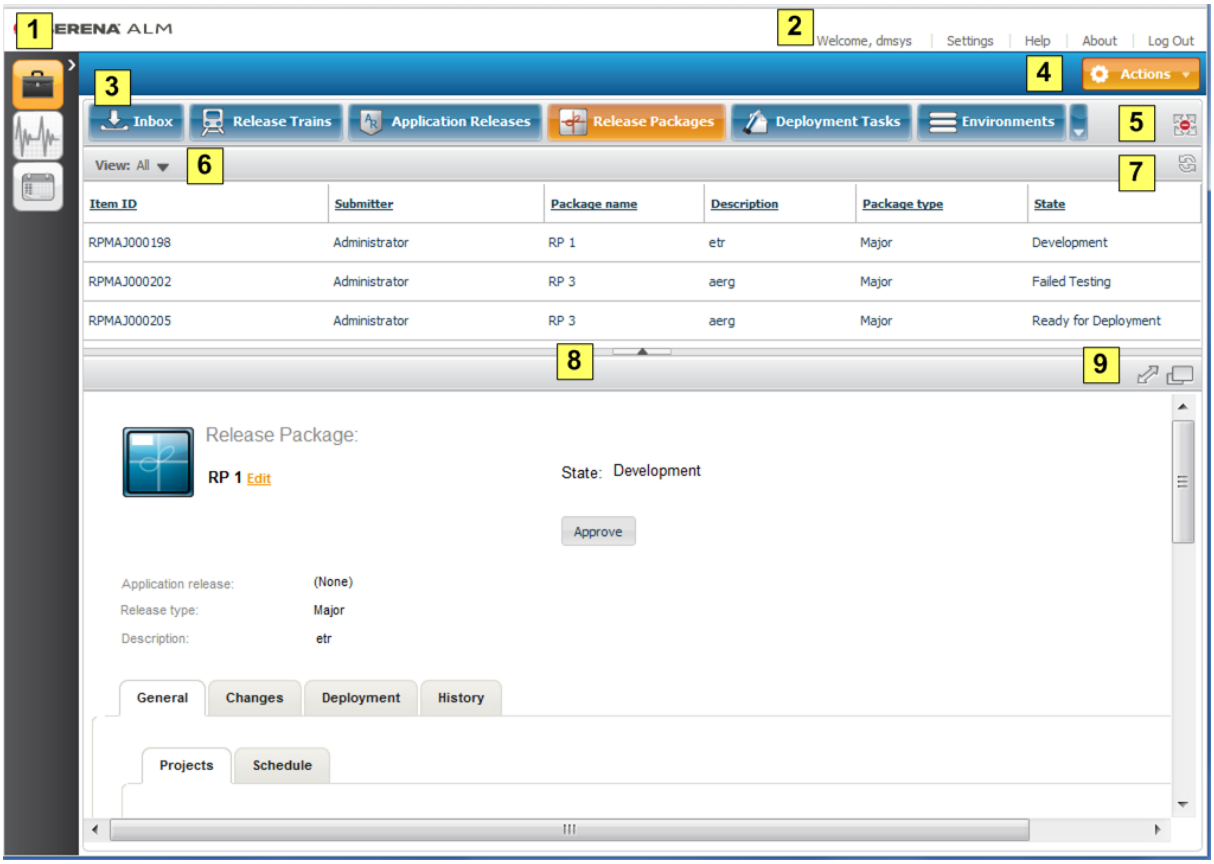
2. Enter your username and password and then click Log In.

Upon successful login, the Serena ALM landing page appears.

If your login is not successful, please contact your Serena ALM or Serena Release Manager administrator.

Navigating Serena Release Manager

The items and options displayed in the Serena ALM user interface are determined by your role and the selections you make in the interface. The default landing page elements for Serena Release Manager are shown in the following figure.



1. The left navigation panel shows top level categories for the ALM suite products. These include:
   - **My Work**: Where you work with your release management or other Serena ALM items according to your role and privileges.
   - **Activity**: Where you view information on release packages with recent activity.
   - **Calendars**: Where you view and select the release trains in context of a Gantt-view schedule or traditional calendar
2. From the menu you can do the following:
   - Click **Settings** to change your personal settings, including e-mail, phone, and password; notification subscriptions; date format, time zone, and calendar; and landing page layout.
   - Click **Help** to open the Serena ALM Help topics in a separate window.
   - Click **About** to see the version and build number of SBM.
   - Click **Logout** to log out of Serena ALM.
3. Click a button in the toolbar to choose the Serena ALM page that you want to work with. For example, click:
   - **Inbox** to list all of the items assigned to you by others.
   - **Release Trains** to view and work with release trains.
   - **Application Releases** to view and work with application releases.
   - **Release Packages** to view and work with release packages.
   - **Deployment Tasks** to view and work with deployment tasks.
   - **Environments** to view and work with environments.

- **Deployment Process Templates** to view and work with deployment process templates.
- **Reports** to view general reports on your release management information.

4. Click the **Actions** button to select from a menu of actions allowed for the current user's role, such as creating a new release train or release package.
5. Use the **Zoom** control ( ) to zoom in and out. For example, if your landing page includes both My Work and the calendar view, when My Work is zoomed in ( ), single-click an item to show the content of the selected item in the lower pane, or preview pane, instead of showing the calendar. When My Work is in zoomed out mode, you must double-click to select an item, and the selected item's form appears in a separate window.
6. Select from the **View All** drop-down list to see reports that you can use to filter the contents of the selected page.
7. Click the **Refresh** control ( ) to reload the contents of the page with the most recent information.
8. Resize the sections of the page to show more detail by dragging the separator line. Close and open the preview pane by clicking the arrow ( ).
9. Use the **Below/Beside** control ( ) to toggle between showing the preview pane below or beside the item list. Use the **Send to Tab** control ( ) to open the form in a separate window.

TIP Use **Settings | Landing Page** to change your landing page layout to show the information you most often work with. For example, change the layout to show only **My Work** on the landing page or change the layout to show only **Calendars** on the landing page. You can view any of the information at any time using the left navigation pane.

For more information on the UI navigation and how to set up the landing page, see *Welcome to Serena ALM*, "Navigating the Serena ALM Web Client".

Getting Help

If you encounter errors in the product or the information you expect does not appear, your user ID may not be assigned the proper role or privileges, or the integrating system may not be communicating properly across the network.

Configuration information is documented in *Serena ALM Installation and Configuration*. If you are unable to find an answer in the documentation or in the readme, check with your Serena Release Manager administrator.

Planning and Controlling Releases

Plan and control application releases with release trains and application releases.

Release Train Hierarchy Overview

Release trains provide a published schedule of changes to production. One or more application releases are associated with each release train.

Release trains have types of major, minor, and emergency. The release types are used to at this level to drive release policies on what types of changes may be delivered.

Release trains enable you to most effectively address common release control challenges.

- **Scheduling Simultaneous Application Releases**: You can schedule multiple application releases for optimal time and resource utilization using release trains.
- **Designating and Adjusting Schedules**: You can designate and adjust the schedules of application releases through their association to release trains.
- **Designating and Adjusting the Scope of Change**: You can designate and adjust the scope of an application release by adding release packages to and removing them from application releases. You can use Request for Change associations to keep visibility of the scope and track scope changes.

Planning Release Trains

Release train planning involves deciding not only the fundamental information of the release train, such as the name, code name, type, and schedule window for each stage of the release, but also designating any infrastructure changes, the application releases involved, and the content that is planned for the release.

Associations with release trains include the following:

- **Requests for Change** represent operational changes that may affect multiple applications or implement system infrastructure changes within an enterprise. These are typically associated with tickets from service management systems such as Serena Service Manager.
- **Application Releases** represent versions of applications or projects, where the application or project architecture is specified by components. One or more release packages are associated with each application release.

- **Release Content** represents other associations that you want to identify and track as part of the release train. Release content may include any number of requests, requirements, or other information relevant to your release train.

Creating Release Trains

You can create a release train to begin your release planning and scheduling.

To create a release train:

1. In the **Actions** menu, select **New Release Train.**

   The **Create Release Trains** page appears.

2. Fill out the form.

   For example, you would fill out the default form as follows:

   - Name the release train.
   - Designate a release manager for the release train.
   - Designate the type of release.
   - Set to and from dates for the release train stages.

3. Click **OK**.

TIP A quick way to set to or from date is to type `Plus n` in the field, where n is number of days. This sets the date to the specified number of days from today's date.

Viewing and Updating Release Trains

You can view and select from lists of release trains from **Inbox**, **Release Trains**, **Calendars**, and **Reports** pages and from child application releases.

From any pages or views that list release trains, you can select a release train to invoke the **Release Train** view. From the **Release Train** view, you can view and edit the release train details and associations.

From the **Release Train** form you can update the dates for stages specified within the release train. You can update schedules only if the release train is in a state in which editing of the schedule is allowed, such as **Planning**.

Selecting Release Trains on a Release Calendar

Release train scheduling information is displayed in the Serena ALM user interface through visual Gantt-style and block calendars.

The release calendars show all the release trains that have been created. Select a release train on the release calendar to view the stage schedule for the release train.

You can do the following:

- View a list of release trains.
- Select a release train to view and work with that particular release train.
- Scroll right in the calendar to see release trains scheduled in the future.
- See the date range for a stage in hover text by moving your cursor over a stage bar in the chart.
- Select time period display format by selecting **Day**, **Week**, or **Month**.

- Switch between the Gantt view and regular calendar view by clicking the **Gantt** and **Calendar** buttons above the calendar. See Using the Calendars for Quick Assessment [page 72].

Associating RFCs with Release Trains

When planning release trains, you can associate requests for change with them. RFCs represent operational changes that may affect multiple applications or implement system infrastructure changes within an enterprise. These are typically associated with tickets from service management systems such as Serena Service Manager.

The default implementation of Serena Release Manager includes a two-way integration with Serena Service Manager (SSM) that is driven by Serena Release Manager.

NOTE Serena Release Manager identifies and lists the requests for change based on the RFC provider information configured for your system. See *Serena ALM Installation and Configuration* for details.

Adding RFCs to Release Trains

You may add existing approved infrastructure change requests from SSM to a release train as RFCs to track them as part of your release planning process.

You can add a request for change only when the release train is in a state for which this action is implemented. In the default workflow, a request for change can be added to the release train only when the release train is in the **Planning** state.

To associate requests for change with a release train:

1. View a release train and select the **Requests For Change** tab.
2. Click **Add requests for change**.

   By default, only change requests in **Approved** and **Approved Changes** states are listed for selection.
3. Select one or more requests for change to add to the release train.
4. Click **OK**.

Creating RFCs from Release Trains

You may find during the release planning process that an infrastructure change is necessary. You can create a request for change in SSM directly from a release train so that you don't need to invoke SSM separately for this purpose.

By default, you can create RFCs only when the release train is in the **Planning** state.

To create a request for change from a release train:

1. View a release train and select the **Requests for change** tab.
2. Click **Create request for change**.

   The SSM user interface form for creating an RFC appears.
3. Create the RFC using the SSM form.

   As you create the RFC, you may assign it to this release train using the **Planned Releases** tab. SSM will submit it to the **Approved Changes** state so that it will then appear under this release train's RFCs.

   If you create the RFC and don't assign it to this release train using the **Planned Releases** tab, it will not appear under this release train. You can later add it to this or another release train using the **Add requests for change** option.
4. Click **OK**.

Removing RFCs from Release Trains

You may find during the release planning process that an infrastructure change needs to be removed from a particular release train.

You can remove a request for change only when the release train is in a state for which this action is implemented. In the default workflow, a request for change can be removed from the release train only when the release train is in the **Planning** state.

To remove requests for change from a release train:

1. View a release train and select the **Requests for change** tab.
2. Select one or more requests for change.
3. Click **Remove request for change**.
4. Click **OK**.

Associating Application Releases with Release Trains

You can add application releases to and remove them from release trains from the release train view. In the default implementation of Serena Release Manager, associations to release trains can be done only when the release train is in the **Planning** state.

To associate application releases with a release train:

1. Navigate to a release train and select the **Application Releases** tab.
2. Click **Add application releases**.
3. Select one or more application releases to add to the release train.
4. Click **OK**.

Associating Release Content with Release Trains

You can add release content and remove it from release trains from the release train view. In the default implementation of Serena Release Manager, associations to release trains can be done only when the release train is in the **Planning** state.

To associate release content with a release train:

1. Navigate to a release train and select the **Release content** tab.
2. Click **Add release content**.
3. Select a process app from which to choose a report.
4. Select a report from which to choose content.
5. Select content items to associate with your release train.
6. Click **OK**.

Prioritizing Release Content

After you have associated release content with a release train, you may update the content items to:

- Specify a different order, or priority, than you originally specified. You can re-order these as needed by overtyping the numbers or by using drag and drop.
- Select a risk assignment, such as Low, Medium, High, or Critical.
- Enter a numeric value to represent the effort or other weighting for the item.

To update release content in a release train:

1. From a release train view, select the **Release content** tab.
2. Click **Update release content**.
3. To re-order the release content items:
   a. To explicitly specify the number, click the number in the sequence number column and overtype with the number you want.
   b. To re-order using drag and drop:
      i. Move your pointer over the **Drag** column icon ( ) for the item you want to re-order.
      ii. When you see the move symbol above the icon (), click and hold the icon as you drag the item to the desired position.
      iii. Release the item in the desired position.

      The items are automatically renumbered according to the new positioning.
4. To select a risk level, from the **Risk** drop-down menu, select one of the available values.
5. To set a value, such as effort or weight, enter a numeric value in the **Value** field.
6. Click **OK**.

Copying Release Trains

In many organizations, release trains with a similar makeup of application releases are initiated at regular intervals over time. To make it easier to create items for these in Serena Release Control, you can make a copy of release train information.

You can copy part or all of the release train hierarchy as follows:

- Release train information only
- Release train and application release information
- Release train, application release, and release package information

This excludes development change requests, deployment units, and deployment tasks.

To copy a release train:

1. Navigate to a release train.
2. Click **Copy**.

   The copy form appears, pre-filled with information from the source release train.
3. Select one of the **Copy Options**, such as:
   - Copy only Release Train
   - Copy all Application Releases
   - Copy all Release Packages
4. Change the information in the form as needed for the new release train.

   For example, you would usually change identifying information and schedule start and end dates.
5. Click **OK**.

The new release train view appears. The initial state is shown as **Copying** and will automatically move to **Planning** when all requested levels of the hierarchy are copied to the new release train.

6. Click the **Reload Item** button () to see the updated state.

NOTE The copy processing is done in the background, or asynchronous mode, to allow you to continue working in Serena Release Control while the system is copying a release train hierarchy with many associations.

Closing Release Trains

Release trains are closed when they reach the **Completed** state in their lifecycle. If for any reason the release trains are not successfully completed, you can move them to the **Release Failed** state through the **Release Fail** action. You can include release failure details to document the reason the release failed.

Publishing and Approving Release Trains

You can require reviews of release trains before they are published to ensure that the content and schedule is agreed upon and controlled.

For example, when release managers and other stakeholders have finished planning a release train, a Release Manager may send it for review and require formal publishing. Once the release train is in review or published, no changes can be made to it unless it is returned to the **Planning** state.

In the default workflow, you may publish a release train after the schedule is agreed upon and all application release associations are made to support the scope and objectives of the release train. This a way to publish the official plan and control any schedule or content changes. After the release train moves to the **Approved** state, the RFC list is recorded. If the release train is returned to the planning state, any RFCs added or removed are tracked so that you can report on the scope changes as needed.

To publish a release train:

1. Navigate to a release train that is in the **Planning** state.
2. Click **Publish**.

The release package enters the **Approved** state and the **Execute** action is available to owners, such as release managers.

Executing Release Trains

After a release train has been published and is in the **Approved** state, it can be executed.

In the default workflow, the release train can still be returned to the **Planning** state from the **Approved** state if you need to change the associations. After you click **Execute**, the release train's application releases' release packages are deployed and installed according to the deployment task definitions that you have previously configured.

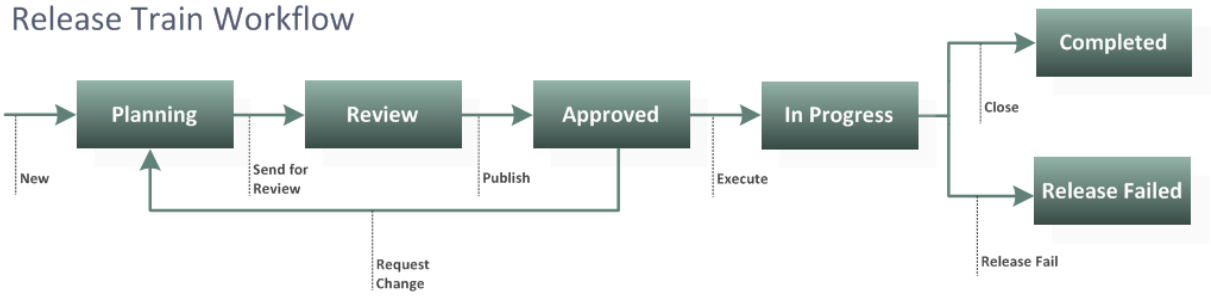
To execute a release train:

1. Navigate to a release train that is in the **Approved** state.
2. Click **Execute**.

The release train enters the **In Progress** state. The release train remains in the **In Progress** state until it progresses to one of the completion states, **Close** or **Release Fail**.

Actioning Release Trains Through the Workflow

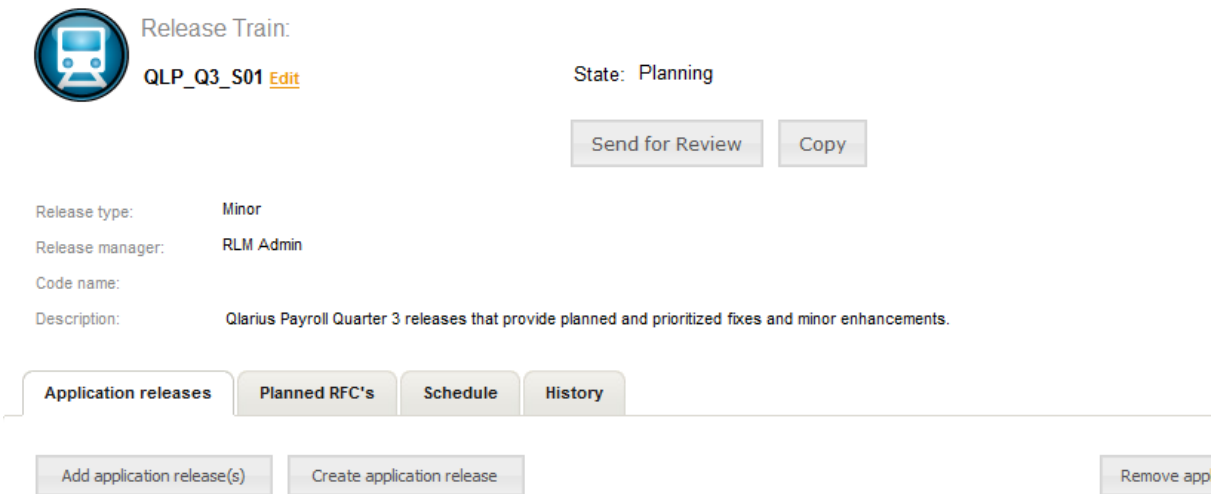
As you work with release trains, you are presented with buttons on the user interface for workflow actions that are available to you. A simplified representation of the default workflow for a release train is shown in the following figure.



The actions that appear here are:

- New (create)
- Send for Review
- Publish
- Request Change
- Execute
- Close
- Release Fail

For example, after you create a new release train, the release train goes into the **Planning** state, and the **Release Train** form shows the next action, **Send for Review**, as shown in the following figure.



While the release train is in **Planning**, you can associate RFCs with it to record the infrastructure or IT level changes to be implemented as part of this release train. You can also associate application releases with it that represent and track information on the versions of applications to be released as part of this release train.

When you click **Send for Review**, the release train goes into the **Review** state. Once the team has agreed upon the release train content, a user with the role responsible for review must then publish the release train so that it progresses to the **Approved** state.

After the release train enters the **Approved** state, any RFCs associated with it are recorded. If at any point the release train is returned to **Planning** and RFCs added or removed, these changes are recorded so that you can track scope changes.

From the **Approved** state, you **Execute** the release train so that it progresses to the **In Progress** state. The release train remains in the **In Progress** state until it is either reverted back to the planning state for additional changes or progressed to one of the completion states.

The progression to **Completed** or **Release Failed** is typically done automatically based on the results of deployment tasks associated with a release package.

For information on how the workflows in Serena Release Manager work together, see Progressing Releases Through Their Lifecycles [page 15].

Planning Application Releases

Application releases represent a version of an application or project, where the application or project architecture is specified by components. An application release may be associated with a single release train and may have one or more release packages associated with it.

Direct associations with application releases include the following:

- **Release Packages** represent a portion of IT or service infrastructure normally built, deployed, tested, and released together. Release packages define the set of changes to be deployed and drive the deployment processes. One or more development change requests and deployment units are associated with each release package.
- **Business Change Requests** represent customer or business unit change requests that affect specific application releases. They are typically associated with tickets from help desk or incident management systems such as Serena Service Manager or other SBM solutions.

Creating Application Releases

You can create an application release for the application you are updating.

To create an application release:

1. In the **Actions** menu, select **New Application Release**.

   The **Create Application Release** form appears.

2. Fill out the form.

   For example, you would fill out the default form as follows:

   - Name an application release.
   - Designate a version for the application release.
   - Designate the status of the release.
   - Describe the application release.
   - Associate the release with the application to which it belongs.

   The combination of official release name and version must be unique.

3. Click **OK**.

Viewing and Updating Application Releases

From release trains and any other pages or views that list application releases, you can select an application release to invoke the **Application Release** view. From this view, you can view and edit the application release details and associations.

Associating BCRs with Application Releases

When planning application releases, you can associate business change requests with them, which represent customer or business unit change requests that affect specific application releases. They are typically associated with tickets from help desk or incident management systems such as Serena Service Manager or other SBM solutions.

Serena Release Manager identifies and lists the business change requests based on the BCR provider information configured for your system. See *Serena ALM Installation and Configuration* for details.

Adding BCRs to Application Releases

You can add a business change request only when the application release is in a state for which this action is implemented. In the default workflow, a business change request can be added to the application release only when the application release is in Planning.

To associate business change requests with an application release:

1. View an application release and select the **Business Change Requests** tab.
2. Click **Add business change requests**.
3. Select one or more business change requests to add to the application release.
4. Click **OK**.

Removing BCRs from Application Releases

You can remove a business change request only when the application release is in a state for which this action is implemented. In the default workflow, a business change request can be removed from the application release only when the application release is in Planning.

To remove business change requests from an application release:

1. View an application release and select the **Business Change Requests** tab.
2. Select one or more business change requests.
3. Click **Remove business change requests**.
4. Click **OK**.

Associating Release Packages with Application Releases

You can add release packages to and remove them from application releases from the application release view.

In the default implementation of Serena Release Manager, associations to application releases can be done only when the application release is in the **Planning** state.

You can add release packages to application releases, making them dependent release packages. You can add multiple release packages to an application release, but a particular release package can be associated with only one release of a particular application.

To associate a release package with an application release:

1. View an application release.
2. Click **Add Release Package**.
3. Select a release package to add to the application release.
4. Click **OK**.

Publishing Application Releases

You can require reviews of application releases before they are published to ensure that the content is agreed upon and controlled.

For example, when release managers and other stakeholders have finished planning an application release, a Release Manager may send it for review and require formal publishing. Once the application release is in review or published, no changes can be made to its associations.

In the default workflow, you may publish an application release after all release package associations are made to support the scope and objectives of the application release. This is a way to publish the official plan and control any content changes.

To publish an application release:

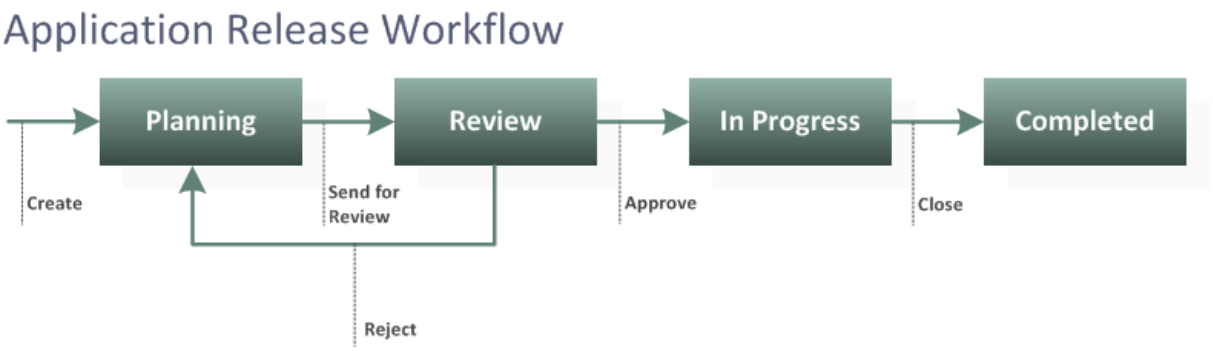
1. Navigate to an application release that is in the **Planning** state.
2. Click **Publish**.

The application release enters the **In Progress** state and the **Close** action is available to owners, such as release managers.

See the simplified default workflow in Actioning Application Releases Through the Workflow [page 30].

Actioning Application Releases Through the Workflow

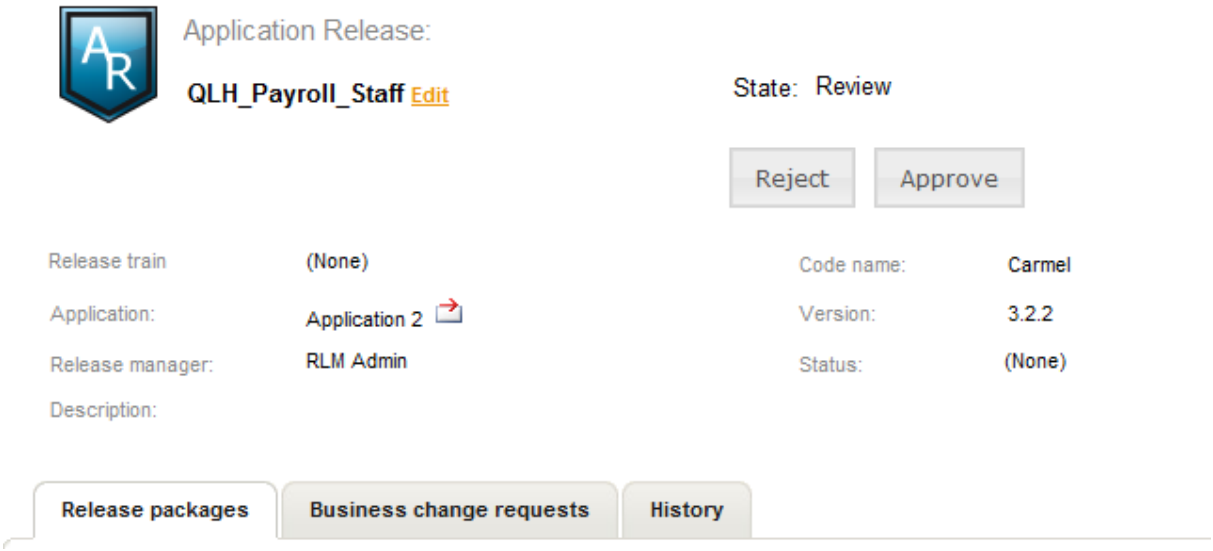
As you work with application releases, you are presented with buttons on the user interface for workflow actions that are available to you. A simplified representation of the default workflow for an application release is shown in the following figure.



The actions that appear here are:

- Create
- Send for Review
- Approve
- Reject
- Close

For example, after you send an application release for review, the application release goes into the **Review** state, and the **Application Release** form shows the next actions to choose from, **Reject** or **Approve**, as shown in the following figure.



The application release remains in the **In Progress** state until it is either reverted back to the planning state for additional changes or progressed to one of the completion states.

The progression to **Completed** is typically done automatically when all of the release packages associated with the application release are complete.

For information on how the workflows in Serena Release Manager work together, see Progressing Releases Through Their Lifecycles [page 15].

Managing Environments

Configure environments to manage the servers associated with your release train and release package stages.

Environment Overview

You can include a list of environments in Serena Release Manager so that release managers and release engineers can manage the availability of servers or areas on servers as you work with release trains and release packages.

Environments enable you to add information about your servers. This can be one or more physical or logical computers or systems to which a release package may be deployed.

By using environments, you can get information about which servers are available to use for your release package stages.

You can do the following:

- Create an environment and associate it with a release stage.
- Modify an environment.
- Commission and decommission an environment.
- Set an environment online or offline.
- Check availability of environments

For information on how to check the availability of environments for your release package stages, see Checking Availability of Environments [page 71].

NOTE Documentation on populating the environment server selection field with your organization's servers is in *Serena ALM Installation and Configuration*.

Commissioning Environments

You should create, or commission, environments as needed to support your release stages. To create an environment you submit it, and it goes into the **Commissioned** state.

To create an environment:

1. In the **Actions** menu, select **New Environment**.

   The **Create Environment** form appears.

2. Fill out the form.

For example:

a. Enter **Name**.

b. Optionally enter a description.

c. Select the owner, such as Release Manager, for the environment.

d. Select the stage for which the environment may be used.

e. Select one or more servers that the environment represents.

3. Click **OK**.

The environment is saved.

Editing an Environment

You can edit an environment only when it is in the Commissioned state. You can change any of the environment fields during editing except stage. Editing the environment has no affect on any servers that it represents.

To edit an environment:

1. Navigate to an environment.
2. Click the edit link beside the view name.
3. Modify fields on the form as needed.
4. Click **OK**.

Setting Environments Online or Offline

In the default workflow, you may set an environment online to indicate that it is available to be used for release package stages, or you may set it offline to indicate that it is not available to be used for release package stages.

See the simplified default workflow in Actioning Environments Through the Workflow [page 33].

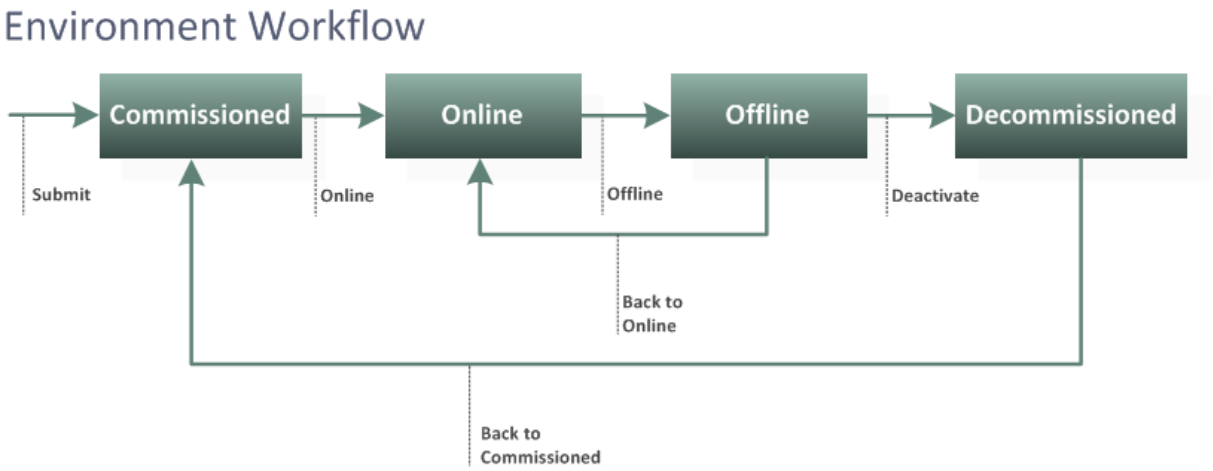
Decommissioning Environments

After the environment is no longer needed, you may decide to decommission it. This indicates that the environment is no longer available to be used.

See the simplified default workflow in Actioning Environments Through the Workflow [page 33].

Actioning Environments Through the Workflow

As you work with environments, you are presented with buttons on the user interface for workflow actions that are available to you. The default workflow for an environment is shown in the following figure.



The actions that appear here are:

- Submit
- Online
- Offline
- Back to Online
- Deactivate
- Back to Commissioned

For example, after you submit, or create, an environment, the environment goes into the **Commissioned** state, and the environment form shows the next action, **Online**.

When the environment is in the **Online** state, the environment is considered available for use. When you click **Offline**, the environment is considered unavailable for use.

From the **Offline** state you can either click **Back to Online** to return it to online status or click **Deactivate** to decommission it and remove it from the list of available environments.

For information on how the workflows in Serena Release Manager work together, see Progressing Releases Through Their Lifecycles [page 15].

Packaging Release Changes

Designate and combine sets of changed components you want to deploy using release packages.

Release Package Overview

Release packages are portions of IT or service infrastructure normally built, deployed, tested, and released together. Release packages define the set of changes to be deployed and drive the deployment processes.

One or more development change requests and deployment units are associated with each release package. Deployment tasks are associated with release packages and are executed as part of the release package workflow.

Release packages can be dependent or independent. Dependent release packages are associated with application releases and are part of the release train hierarchy. Independent release packages are standalone objects.

Dependent release packages are associated with application releases. This ensures that the changes included and deployed as part of the release packages have followed the full workflow and approvals needed for the release train and application release of which they are a part.

Using dependent release packages ensures that complete release scheduling and planning processes are followed before deployment to any environments.

Independent release packages enable you to perform hotfix and emergency releases outside of the scope of a release train and application releases.

For example, you may need to do a data fix to a field that had an incorrect date format applied during an upgrade, and rather than initiate a formal release train and application release for this, you could expedite the change by creating an independent release package and sending it through the simpler workflow to deployment.

Independent release packages enable you to deploy urgent or low-impact changes into production quickly while still leveraging the visibility, automated deployment, and audit trail capabilities of Serena Release Manager.

TIP For independent release packages, consider including schedule information at the release package level since these are not associated with release trains.

Direct associations with release packages include the following:

- **Development Change Requests** represent delivered changes from the development process and are typically associated with tickets from development management systems. Examples include SBM incidents or issues, Dimensions CM requests, and Rally defects.
- **Deployment Units** represent a set of deployable components. Examples include Dimensions CM baselines with build outputs, Serena Development Manager development packages, and ChangeMan ZMF change packages.

- **Deployment Tasks** are actions to be executed as part of the process of deploying a release package into a specific stage environment. Deployment task types include manual, vault, and automation. Within these types are various sub-types.
- **Deployment Process Templates** enable you to create and copy sets of deployment tasks for reuse in different stages, release packages, and applications.

Other attributes that control release packages are as follows:

- **Release Types** are used to determine the stages release packages are deployed through. In the default implementation of Serena Release Manager, release types include major, minor, and emergency.
- **Release Stages** are the gates that the release goes through on its path into production. The default stages provided are integration test (INT), user acceptance test (UAT), and production deployment (PROD). A stage may have one or more environments related to it. Releases may be deployed into an environment based on availability.

Creating Release Packages

You can create a release package by adding identifying details and associating change objects with it. You can enter the details and some of the associations during creation, then add the remaining associations by viewing the release package and selecting the desired options.

To create a release package:

1. In the **Actions** menu, select **New Release Package**.

   The **Create Release Package** form appears.

2. Fill out the form.

   For example, you would fill out the default form as follows:

   - Enter a name for the release package according to your organization's naming standards.
   - Select the package type.
   - Optionally enter a description.
   - Designate a schedule for the release package stages.
   - Designate the release manager, or owner, for the release package.
   - Optionally designate a secondary owner.
   - Select development change request projects for the release package.
   - Select deployment unit projects for the release package.

3. Select one or more development change request projects. These are the projects from which development change requests can be associated with this release package, such as SBM projects for incidents or issues or Dimensions CM projects or streams.
4. Select one or more deployment unit projects. These are the projects or streams from which deployment units can be associated with this release package, such as Dimensions CM baselines.
5. Click **OK**.

IMPORTANT! You must select your projects for the release package associations, such as development change requests and deployment units, during the creation of the release package.

Associating DCRs with Release Packages

When creating a release package, you can associate that release package with development change requests, which represent delivered changes from the development process and are typically associated with tickets from development management systems.

Serena Release Manager identifies and lists the development change requests based on the development change request projects that are selected for the release package and any filters on request status that are defined in configuration files. For example, for Dimensions CM you may see only the requests in IN QA, IN PROGRESS, UNDER WORK, and IN TEST statuses.

Adding DCRs to Release Packages

You can add a development change request only when the release package is in a state for which this action is implemented. In the default workflow, the development change request can be added to the release package only when the release package is in Development.

To associate development change requests with a release package:

1. View a release package and select the **Development Change Requests** tab.
2. Click **Add development change requests**.
3. Select a development change request to add to the release package.
4. Click **OK**.

Removing DCRs from Release Packages

You can remove a development change request only when the release package is in a state for which this action is implemented. In the default workflow, a development change request can be removed from the release package only when the release package is in Development.

To remove development change requests from a release package:

1. View a release package and select the **Development Change Requests** tab.
2. Select one or more development change requests.
3. Click **Remove development change requests**.
4. Click **OK**.

Associating Deployment Units with Release Packages

You can add deployment units to and remove them from release packages. The deployment unit represents a set of deployable components, such as a Dimensions CM baseline with build outputs or Serena Development Manager development packages.

Serena Release Manager identifies and lists the deployment units based on the deployment unit projects that are selected for the release package and any filters on status that are defined in configuration files. For example, for Dimensions CM you may see only the baselines in VERIFIED, DEPLOYED, and CAPTURED statuses.

Adding Deployment Units to Release Packages

You can add a deployment unit only if there is a transition available based on the state of the release package. In the default workflow, the only state where a deployment unit can be added to the release package is when the release package is in Development.

To associate deployment units with a release package:

1. View a release package and select the **Deployment Units** tab.
2. Click **Add deployment units**.

3. Fill out the **Add Deployment Units** form.

   For example:

   a. Select **Source location** of either **Dimensions CM** or **Vault**.

      If you select **Vault** for **Source location**, only baselines in the Dimensions CM product you have defined for your vault will appear in the **Dimensions CM packages** list.

   b. Get and select one or more deployment units. Choices are **Dimensions CM packages**, or baselines, and **ChangeMan ZMF packages**.

4. Click **OK**.

Removing Deployment Units from Release Packages

You can remove a deployment unit only if there is a transition available based on the state of the release package. In the default workflow, the only state where a deployment unit can be removed from the release package is when the release package is in Development.

To remove deployment units from a release package:

1. View a release package and select the **Deployment Units** tab.
2. Select one or more deployment units.
3. Click **Remove deployment units**.
4. Click **OK**.

Associating Deployment Tasks with Release Packages

To automate and ensure all tasks are executed for your release, you can create sets of deployment tasks to be executed when the release package is deployed to each stage environment.

You can save deployment tasks as deployment process templates so that you can use the same sets of tasks across release packages and across release package stages.

For more information see Automating Deployment Processes [page 40].

Approving Release Packages

You can require approvals to release packages to ensure that the content is agreed upon and controlled.

For example, when Release Engineers have finished developing a release package, a Release Engineer or Release Manager may approve it. Once the release package is in review or approved, no changes can be made to it unless it is returned to the **Development** state.

In the default workflow, you may approve a package after all necessary associations are made to support the deployment, installation, and configuration of the release package deployment units. This a way to ensure that the changes are "frozen" and ready to go through the test and pre-production stages to production.

To approve a release package:

1. Navigate to a release package that is in the **Development** state.
2. Click **Approve**.

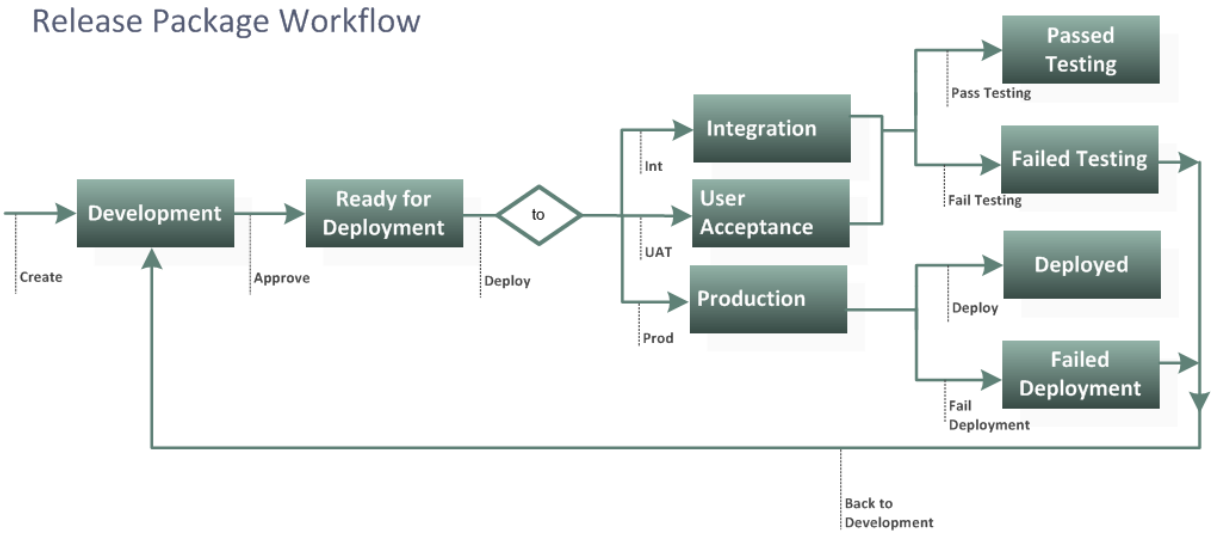
   The release package enters the **Ready for Deployment** state and the **Deploy** action is available to owners, such as release engineers.

See the simplified default workflow in Actioning Release Packages Through the Workflow [page 39].

Actioning Release Packages Through the Workflow

As you work with release packages, you are presented with buttons on the user interface for workflow actions that are available to you. The release package workflow is a conditional workflow that allows for default paths for each release type, Major, Minor, and Emergency. The workflow actions available for each release type are the actions needed to support the release stages for that release type.

A simplified representation of the default workflow path for a major release package is shown in the following figure.



The actions that appear here are:

- Create
- Approve
- Deploy
- Back to Development

For example, after you approve a release package, the release package goes into the **Ready for Deployment** state, and the release package form shows the next action, **Deploy**.

The stage to which the release package is deployed depends on the type of release package and in which stage the release package currently resides.

The progression to **Passed Testing**, **Failed Testing**, **Deployed**, or **Failed Deployment** is typically done automatically based on the results of deployment tasks associated with a release package. If a release package fails testing or deployment, the **Back to Development** action becomes available to allow for rework and re-deployment.

For information on preparing release packages for deployment and deploying them, see Automating Deployment Processes [page 40] and Deploying and Installing Releases [page 60].

For information on how the workflows in Serena Release Manager work together, see Progressing Releases Through Their Lifecycles [page 15].

Automating Deployment Processes

Configure deployment processes to automate your release deployment, installation, and configuration using deployment tasks and deployment process templates.

Deployment Task Configuration Overview

You can automate processes for deploying application release components into environments by configuring sets of deployment tasks for your applications.

Deployment tasks are actions to be executed as part of the deployment process to deploy release packages into specific environments as defined by **release stages**, such as Integrated Testing, User Acceptance Testing, and Production Deployment. Deployment tasks:

- Integrate with Dimensions CM and ChangeMan ZMF release vaults for secure deployment of deployment units.
- Integrate with Serena Release Automation to automate installation.

Types of deployment tasks are as follows:

- **Manual deployment tasks** are actions to be executed by people.
- **Approval deployment tasks** integrate with ChangeMan ZMF to support approvals that must be obtained before change packages are deployed.
- **Vault deployment tasks** integrate with Dimensions CM and ChangeMan ZMF for secure deployment of deployment units. Vault deployment tasks move deployment units (baselines or change packages) securely to deployment areas (environments or sites).
- **Automation deployment tasks** integrate with release automation to automate installation and configuration tasks through the release automation software (such as Serena Release Automation) as part of the deployment process.

NOTE Your Serena Release Manager administrator configures integrating system information for release vault and release automation. Configuration details are documented in *Serena ALM Installation and Configuration*.

Planning Deployment Tasks

As part of the deployment task planning, you can associate deployment tasks to release package stages by creating them as part of the release package associations, or you can create them as part of deployment process templates and then apply them to release package stages.

The specification of the deployment tasks is the same whether you create them in a release package or in a deployment process template view.

Types of Deployment Tasks

You can specify manual, approval, vault, and automation tasks that will be executed upon deployment of your release package to a specific stage.

Scheduling Deployment Task Execution

You can schedule release vault and release automation type deployment tasks to be automatically executed at a future date and time using the **Schedule execution time** field on the deployment task form.

For the scheduling functionality to work, your Serena Release Manager Administrator must enable scheduling of deployment tasks as documented in *Serena ALM Installation and Configuration*.

Sequencing of Deployment Task Execution

Tasks can be assigned sequence numbers to control the order of execution. The sequence numbers are handled as follows:

- By default the sequence number is set to 0. If you do not overtype the sequence number, the task is initiated immediately after the deployment of the release package to the stage with which the task is associated.
- Sequentially ordered tasks are executed in order.
- Tasks with the same sequence number are executed in parallel. For example, if several tasks have sequence number 3, all of those tasks are initiated when all tasks with sequence number 2 are complete.

Deployment task planning includes the following:

- Creating Manual Deployment Tasks [page 41]
- Creating ChangeMan ZMF Approval Deployment Tasks [page 42]
- Creating Dimensions CM Vault Deployment Tasks [page 44]
- Creating ChangeMan ZMF Vault Deployment Tasks [page 45]
- Creating Release Automation Deployment Tasks [page 46]
- Creating Release Automation (Nolio) Deployment Tasks [page 47]
- Deleting Deployment Tasks [page 50]
- Re-ordering Deployment Tasks [page 50]

Creating Manual Deployment Tasks

Certain actions necessary to deployment may need to be executed manually as part of the deployment process.

You can create manual tasks needed to deploy a release package into a specific stage (environment).

You can do the following:

- Associate the task to a release package stage.

- Specify a sequence number for the task.
- Specify which manual task is required.
- Specify owners for the task.

The sequence number is used to specify order of execution of the tasks associated with a request package. The manual deployment task is specific to a single stage.

To create a manual deployment task:

1. From a release package or deployment process template view, navigate to the **Deployment tasks** tab.
2. Select a stage for which this task will be executed.
3. Click **Create Task**.

   The **Create Deployment Task** form appears.
4. In **Task Type**, select **Manual**.

   The **Task Type: Manual** form appears in the form.
5. Fill out the **Manual** form.

   For example:

   a. Enter **Task name** and **Sequence number**.

   b. Select the primary and secondary owners for the task.

   c. Optionally enter a description.
6. Click **OK**.

The manual task is saved. It must be manually executed according to its sequential order when the release package to which it is associated is deployed to the specified stage.

Creating ChangeMan ZMF Approval Deployment Tasks

Approval deployment tasks integrate with ChangeMan ZMF to ensure all necessary approvals are given before a change package is deployed.

You can create an approval deployment task used to approve the deployment of a deployment unit for a release package into a particular stage environment.

You can do the following:

- Associate the task to a release package stage.
- Specify a sequence number for the task.
- Specify which deployment unit, or ZMF change package, to approve.
- Specify which approvers in the ZMF approver list must approve the deployment of the selected deployment unit.
- Specify owners for the task.

The sequence number is used to specify order of execution of the tasks associated with a request package. The vault deployment task is specific to a single stage.

To execute a vault deployment task for deployment of files stored in the ZMF vault, Serena Release Manager needs information on the approvers for each release stage.

Creating Approval Deployment Tasks

To create an approval deployment task:

1. From a release package or deployment process template view, navigate to the **Deployment** tab and then select the **Deployment tasks** tab.
2. Select a stage for which this task will be executed.
3. Click **Create Task**.

   The **Create Deployment Task** form appears.
4. In **Task Type**, select **ChangeMan ZMF approval**.

   The **Task Type: Approval** form appears.
5. Fill out the **Approval** form.

   For example:

   a. Enter **Task name** and **Sequence number**.

   b. Optionally enter a description.

   c. Click **Get units** to refresh the list of deployment units, or ZMF change packages, stored in the vault and then select a deployment unit.

   d. Click **Get approvers** to refresh the list of approvers for the selected deployment unit, and then select an approver.

   e. Select the primary and secondary owners for the task.

   NOTE When you create a deployment task for a deployment process template, you may not be prompted for some of the details. These should be added later when you add the deployment tasks to a release package.
6. Click **OK**.

Example of Approval Deployment Tasks

This example explains how you could set up ZMF deployment tasks using three levels of promotion approvers, one for each of the default release stages in Serena Release Manager.

Promotion Level Approver

10 – INT Integration Test Approver

20 – UAT UAT Approver

30 – PAT PAT Approver

Release Package Deployment Configuration

For every Major release package, you can configure the release packages to have the following deployment tasks.

Stage: Integration Test

1. ZMF Deployment Task for promotion 10 – INT environment
2. Manual validation task
3. ZMF Approval Task for Integration Test Approver

Stage: User Acceptance Test

1. ZMF Deployment Task for promotion 20 – UAT environment

2. Manual validation task
3. ZMF Approval Task for UAT Approver

Stage: Production Acceptance Test

1. ZMF Deployment Task for promotion 30 – PAT environment
2. Manual validation task
3. ZMF Approval Task for PAT Approver

Creating Dimensions CM Vault Deployment Tasks

Vault deployment tasks integrate with Dimensions CM and ChangeMan ZMF to secure deployment of deployment units.

If you are using a Dimensions CM vault, you can create a vault deployment task used to deploy a deployment unit, or baseline, for a release package into a particular stage environment.

You can do the following:

- Associate the task to a release package stage.
- Specify a sequence number for the task.
- Specify which deployment units, or baselines, to deploy.
- Specify the stage in the GSL to which the deployment applies.
- Specify the deployment area, or environments, to which to deploy.
- Specify owners for the task.

The sequence number is used to specify order of execution of the tasks associated with a request package. The vault deployment task is specific to a single stage.

To execute a vault deployment task for deployment of files stored in Dimensions CM, Serena Release Manager needs information on the Dimensions CM stage and deployment area.

To create a Dimensions CM vault deployment task:

1. From a release package or deployment process template view, navigate to the **Deployment** tab and then select the **Deployment tasks** tab.
2. Select a stage for which this task will be executed.
3. Click **Create Task**.

   The **Create Deployment Task** form appears.
4. In **Task Type**, select **Vault**.

   The **Task Type: Vault** form appears.
5. Fill out the **Vault** form.

   For example:

   a. Enter **Task name** and **Sequence number**.

   b. Click **Get units** to refresh the list of deployment units, or baselines, stored in Dimensions CM, and then select a deployment unit.

   c. Click **Get stages** to refresh the list of stages for the selected deployment unit, and then select a stage.

d. Click **Get areas** to refresh the list of deployment areas for the selected stage.

e. Specify the **Schedule execution time**, the date and time at which you want the deployment task to be initiated.

f. Select the primary and secondary owners for the task.

g. Optionally enter a description.

NOTE When you create a deployment task for a deployment process template, you may not be prompted for some of the details. These should be added later when you add the deployment tasks to a release package.

6. Click **OK**.

The vault task is saved. It will be executed according to its sequential order when the release package to which it is associated is deployed to the specified stage.

NOTE If Dimensions CM is used as your deployment unit provider, baselines are displayed for selection only if they are in release mode and are in one of the statuses defined by your Serena Release Manager administrator.

For more information, see *Serena ALM Installation and Configuration*.

Creating ChangeMan ZMF Vault Deployment Tasks

Vault deployment tasks integrate with Dimensions CM and ChangeMan ZMF to secure deployment of deployment units.

If you are using a ChangeMan ZMF vault, you can create a vault deployment task used to deploy a deployment unit, or ZMF change package, for a release package into a particular stage site.

You can do the following:

- Associate the task to a release package stage.
- Specify a sequence number for the task.
- Specify which deployment units, or change packages, to deploy.
- Specify the sites, or LPARS, to which to deploy.
- Specify the deployment area, or promotion level, to which the deployment applies.
- Specify owners for the task.

The sequence number is used to specify order of execution of the tasks associated with a request package. The vault deployment task is specific to a single stage.

To execute a vault deployment task for deployment of files stored in ChangeMan ZMF, Serena Release Manager needs information on the ChangeMan ZMF site and promotion level.

To create a ChangeMan ZMF vault deployment task:

1. From a release package or deployment process template view, navigate to the **Deployment** tab and then select the **Deployment tasks** tab.
2. Select a stage for which this task will be executed.
3. Click **Create Task**.

   The **Create Deployment Task** form appears.
4. In **Task Type**, select **Vault**.

The **Task Type: Vault** form appears.

5. Fill out the **Vault** form.

   For example:

   a. Enter **Task name** and **Sequence number**.

   b. Optionally enter a description.

   c. Click **Get units** to refresh the list of deployment units, or change packages, stored in ChangeMan ZMF, and then select a deployment unit.

   d. Click **Get sites** to refresh the list of sites for the selected deployment unit, and then select a site.

   e. Click **Get areas** to refresh the list of promotion levels for the selected site, and then select an area.

   f. Specify the **Schedule execution time**, the date and time at which you want the deployment task to be initiated.

   g. Select the primary and secondary owners for the task.

   NOTE When you create a deployment task for a deployment process template, you may not be prompted for some of the details. These should be added later when you add the deployment tasks to a release package.

6. Click **OK**.

The vault task is saved. It will be executed according to its sequential order when the release package to which it is associated is deployed to the specified stage.

NOTE If ChangeMan ZMF is used as your deployment unit provider, change packages are displayed for selection only if they are in one of the categories defined by your Serena Release Manager administrator.

For more information, see *Serena ALM Installation and Configuration*.

Creating Release Automation Deployment Tasks

Automation deployment tasks integrate with Serena Release Automation and Serena Release Automation, powered by Nolio, to automate installation and configuration tasks. This section tells how to create a Serena Release Automation deployment task. For information on creating a Serena Release Automation, powered by Nolio, deployment task, see Creating Release Automation (Nolio) Deployment Tasks [page 47] .

You can create a Serena Release Automation deployment task, which is used to deploy a deployment unit for a release package into a particular stage.

You can do the following:

- Associate the task to a release package stage.
- Specify a sequence number for the task.
- Specify the application involved in the tasks.
- Specify components and versions for which installation and configuration tasks are to be executed.
- Specify the target environments on which to run the tasks.
- Specify the processes that define the actions to be taken on the environments.

- Specify owners for the task.

The sequence number is used to specify order of execution of the deployment tasks. The automation deployment task is specific to a single stage. Each set of application parameters is for a specific server.

To execute an automation deployment task for installation and configuration of files through Serena Release Automation, Serena Release Manager needs information on the applications, components and component versions, environments, and processes.

To create a Release Automation deployment task:

1. From a release package or deployment process template view, navigate to the **Deployment** tab and then select the **Deployment tasks** tab.
2. Select a stage for which this task will be executed.
3. Click **Create Task**.

   The **Create Deployment Task** form appears.
4. In **Task Type**, select **Release Automation**.

   The **Task Type: Automation** form appears.
5. Fill out the **Automation** form.

   For example:

   a. Enter **Task name** and **Sequence number**.

   b. Click **Get application** to refresh the list of applications stored in Serena Release Automation and then select an application.

   c. Click **Get component** to refresh the list of components for the selected application and then select the component.

   d. Select the version of the component from the **Version** menu.

   e. Click **Get environment** to refresh the list of environments for the selected application and then select an environment.

   f. Click **Get process** to refresh the list of processes for the selected environments and then select a process.

   g. Specify the **Schedule execution time**, the date and time at which you want the deployment task to be initiated.

   h. Select the primary and secondary owners for the task.

   i. Optionally enter a description.
6. Click **OK**.

The automation deployment task is saved. It will be executed according to its sequential order when the release package to which it is associated is deployed to the specified stage.

Creating Release Automation (Nolio) Deployment Tasks

Automation deployment tasks integrate with Serena Release Automation and Serena Release Automation, powered by Nolio, to automate installation and configuration tasks. This section tells how to create a Serena Release Automation, powered by Nolio, deployment task. For information on creating a Serena Release Automation deployment task, see Creating Release Automation Deployment Tasks [page 46] .

You can create a Serena Release Automation, powered by Nolio, deployment task, which is used to deploy a deployment unit for a release package into a particular stage.

You can do the following:

- Associate the task to a release package stage.
- Specify a sequence number for the task.
- Specify the application involved in the tasks.
- Specify application parameters to pass to specific environments.
- Specify the target environment on which to run the tasks.
- Specify the process to be executed in the environment.
- Specify owners for the task.

The sequence number is used to specify order of execution of the deployment tasks. The automation deployment task is specific to a single stage. Each set of application parameters is for a specific server.

To execute an automation deployment task for installation and configuration of files through Serena Release Automation, Serena Release Manager needs information on the application, application parameters, environment, and process.

To create a Release Automation (Nolio) deployment task:

1. From a release package or deployment process template view, navigate to the **Deployment** tab and then select the **Deployment tasks** tab.
2. Select a stage for which this task will be executed.
3. Click **Create Task**.

   The **Create Deployment Task** form appears.
4. In **Task Type**, select **Release Automation (Nolio)**.

   The **Task Type: Automation** form appears.
5. Fill out the **Automation** form.

   For example:

   a. Enter **Task name** and **Sequence number**.

   b. Click **Get application** to refresh the list of applications stored in Release Automation (Nolio) and then select an application.

   c. Click **Get environment** to refresh the list of environments for the selected application and then select an environment.

   d. Click **Get process** to refresh the list of processes for the selected environments and then select a process.

   e. In the **Release Automation Parameters** section, enter parameters and the list of servers to which you want to pass these parameters. You can specify only servers that are already defined in Release Automation (Nolio).

      Parameters use the following format:

      ```
      <variable_name1>=<variable_data1>&<variable_name2>=<variable_data2>
      ```

      Example parameters:

      ```
      var1=test1&var2=test2
      ```

      Example servers:

```
Server1=lab1_computer

Server2=lab3_computer
```

f. Specify the **Schedule execution time**, the date and time at which you want the deployment task to be initiated.

g. Select the primary and secondary owners for the task.

h. Optionally enter a description.

6. Click **OK**.

The automation deployment task is saved. It will be executed according to its sequential order when the release package to which it is associated is deployed to the specified stage.

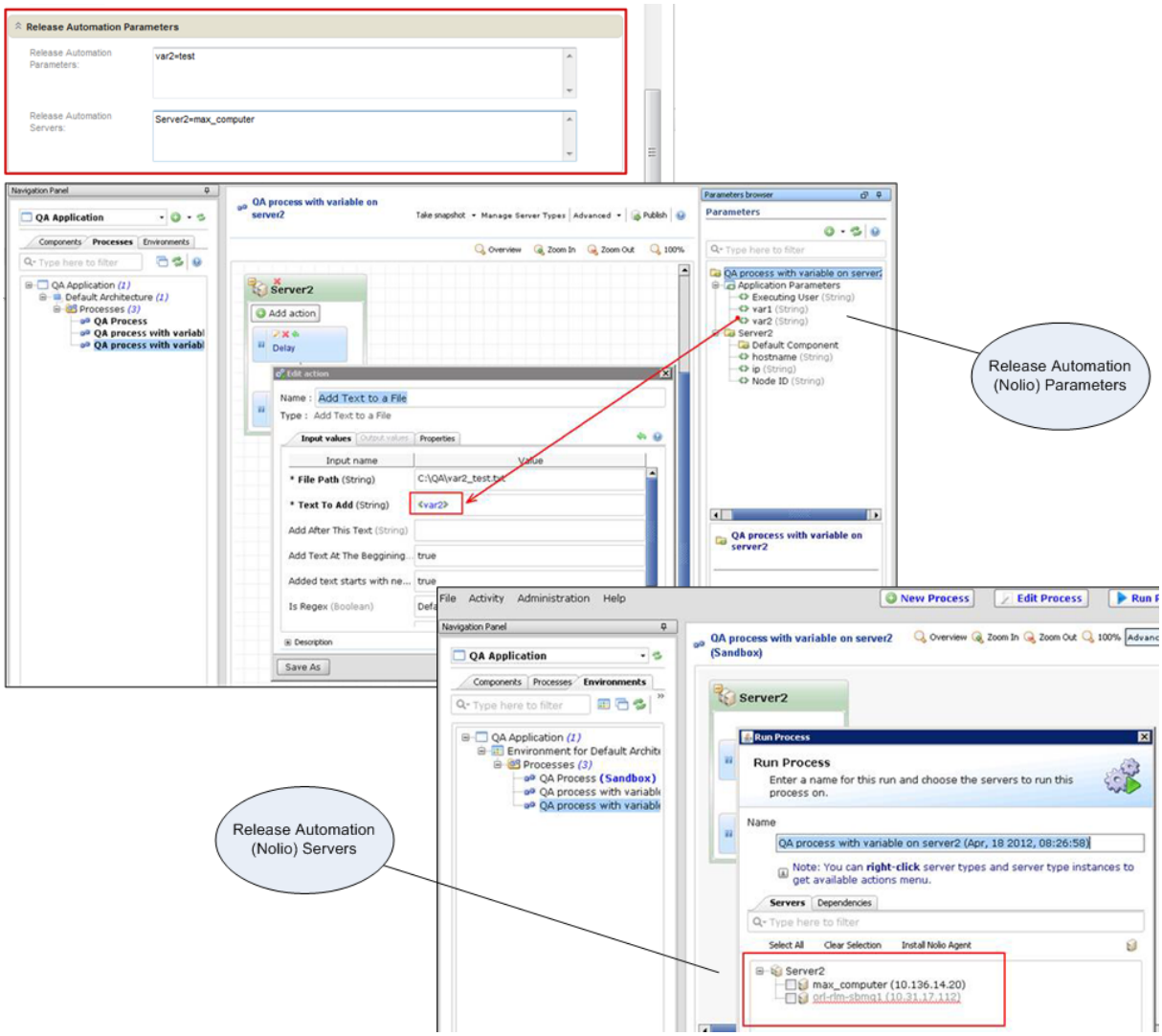
NOTE

- The Release Manager default implementation supports only variables for Release Automation (Nolio) application parameters; it does not support variables for server parameters.
- When you specify clients for your server, only client names that are assigned to your server in Release Automation (Nolio) may be included. For example, if in Release Automation (Nolio) `Server2` is defined to include both `max_computer` and `my_lab_test`, both of these may be specified in the parameter. For example:

  ```
  Server2=max_computer&Server2=my_lab_test
  ```

Example

The Serena Release Manager automation deployment task parameters relate directly to the corresponding parameters and servers in Release Automation (Nolio) as shown in the following figure.



Deleting Deployment Tasks

You can delete one or more deployment tasks from a release package.

To delete deployment tasks from a release package:

1. From a release package or deployment process template view, navigate to the **Deployment** tab and then select the **Deployment tasks** tab.
2. Select a task.
3. Click **Delete task.**

Re-ordering Deployment Tasks

After you have specified the deployment tasks and their sequence numbers, you may decide that the tasks should be done in a different order than you originally specified. You can re-order these as needed by overtyping the numbers or by using drag and drop.

To re-order tasks in a release package:

1. From a release package or deployment process template view, navigate to the **Deployment** tab and then select the **Deployment tasks** tab.
2. Click **Re-order tasks**.
3. To explicitly specify the number, click the number in the sequence number column and overtype with the number you want.

4. To re-order using drag and drop:
   a. Move your pointer over the **Drag** column icon ( ) for the item you want to re-order.
   b. When you see the move symbol above the icon ( ), click and hold the icon as you drag the item to the desired position.
   c. Release the item in the desired position.

   The items are automatically renumbered according to the new positioning.
5. Click **OK**.

Executing Deployment Tasks

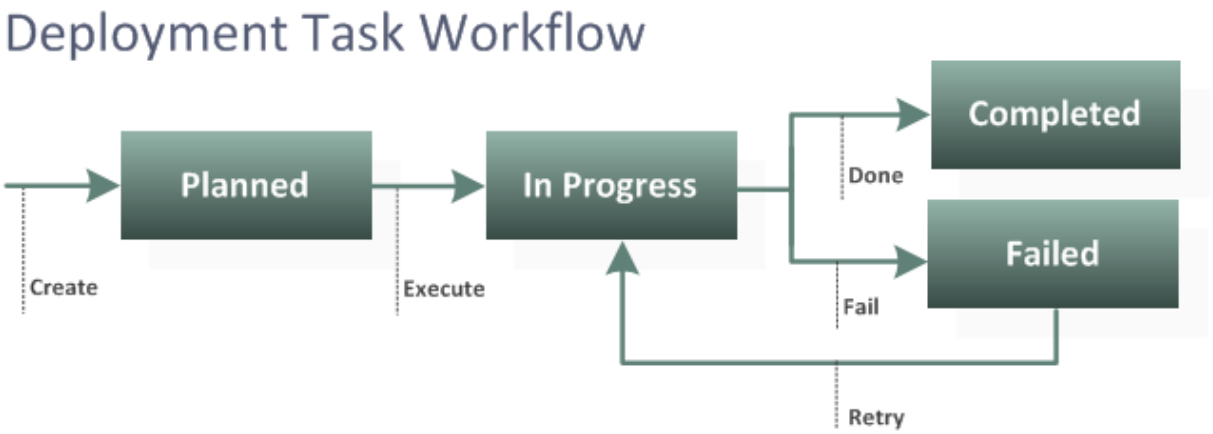
Deployment tasks that are not scheduled are executed automatically when the release package stage to which they are associated is reached and any preceding deployment tasks in the sequentially designated order are completed.

Deployment tasks that are scheduled are executed according to their designated schedule.

Actioning Deployment Tasks Through the Workflow

As you work with deployment tasks, you are presented with buttons on the user interface for workflow actions that are available to you. There are separate workflows for each type of task.

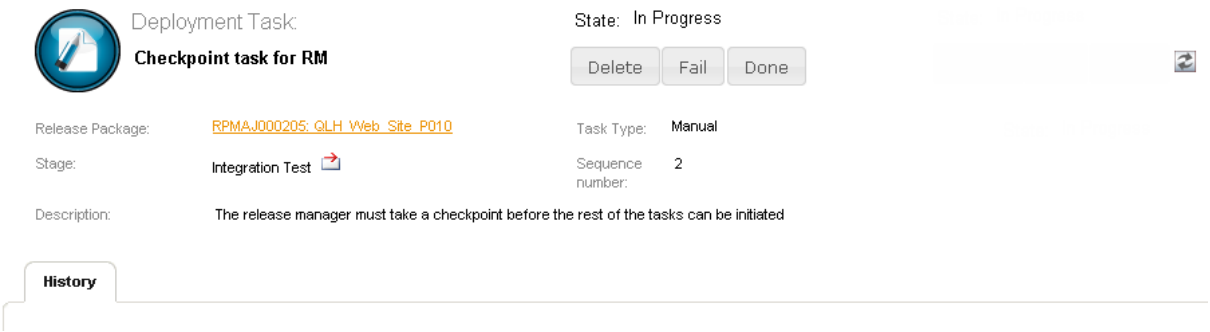
The default workflow for a manual deployment task is shown in the following figure.



The actions that appear here are:

- Create
- Execute
- Done
- Fail

For example, after you execute a manual deployment task, the deployment task goes into the **In Progress** state, and the deployment task form shows the next available actions, **Done** and **Fail**, as shown in the following figure.



You should click **Done** if the manual task as successfully been completed or **Fail** if the manual task has not been successfully completed. **Delete** is available from any state.

Workflows for Non-Manual Deployment Task Types

The workflows for non-manual deployment tasks are similar to that of the manual deployment task except that the **Done** and **Fail** actions are done automatically based on the results of the task execution in Release Vault or Release Automation.

For approval, vault, and automation tasks, the progression to **Completed** or **Failed** is typically done automatically based on the execution results of the deployment tasks.

For information on how the workflows in Serena Release Manager work together, see Progressing Releases Through Their Lifecycles [page 15].

Deployment Process Template Overview

You can copy deployment tasks for reuse through deployment process templates.

Deployment process templates enable you to create and copy deployment tasks for reuse in different stages, release packages, and applications.

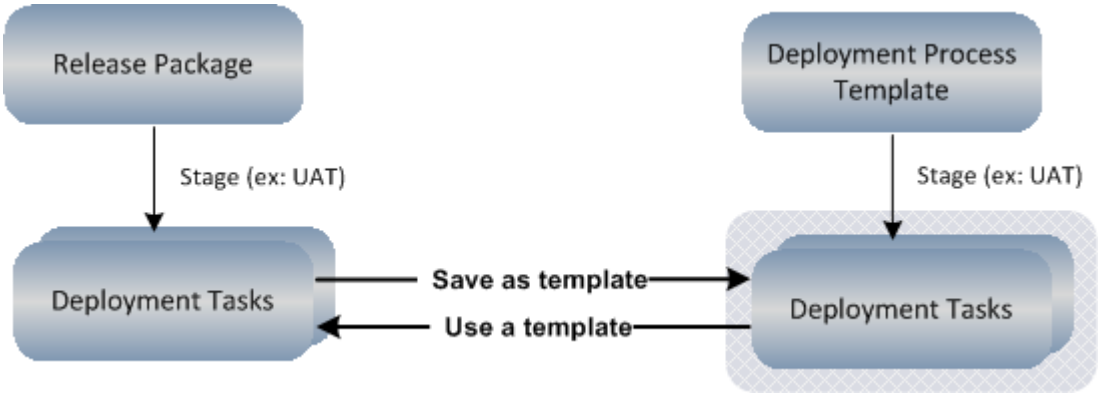
By using deployment process templates, you can more easily implement and standardize release deployment processes and ensure that the required set of deployment tasks are applied at each relevant stage of the release.

By setting up the templates ahead of time, you can more effectively plan the deployment processes. Applying deployment process templates to release packages ensures more consistency across release packages and saves time over setting up the same or similar deployment tasks separately for each release package.

You don’t have to set up the deployment process templates ahead of time. Once you have set up deployment tasks for a release package stage, you can save them to a deployment process template to be used for other stages of that release package or for other similar release packages.

How Are Deployment Process Templates Applied?

Deployment process templates help you plan and configure approved processes and apply them consistently to release package stages from one application release to another. The following figure shows how deployment tasks are saved in deployment process templates and applied to a stage of a release package.



Developing Deployment Process Templates

Deployment process templates enable you to group a number of deployment tasks and make them available for reuse. Reuse may be within the same release package, in which you want to apply the same set of tasks to a different stage, or you may reuse the set of tasks to a different release package.

Deployment tasks in a deployment process template can be re-ordered using a graphical sort function, enabling you to control the order of execution as is appropriate for each stage.

You can do the following:

- Create a deployment process template standalone and then apply it to release package stages.
- Create a deployment process template from the set of deployment tasks associated with particular stage of a release package.
- Save a copy of a deployment process template.
- Modify a deployment process template.
- Re-order deployment tasks within a deployment process template.

Deployment process template development includes the following:

- Creating Deployment Process Templates [page 53]
- Adding Deployment Tasks to a Deployment Process Template [page 54]
- Copying a Deployment Process Template [page 55]
- Saving As a Deployment Process Template [page 55]
- Editing a Deployment Process Template [page 33]
- Retiring Vault Templates [page 69]

Creating Deployment Process Templates

You should create deployment process templates as needed to support your deployment processes. You can create deployment process templates with all new deployment tasks or you can create deployment process templates from existing deployment tasks in a release package.

Creating Deployment Process Templates by Creating New Tasks

To create a deployment process template with new tasks:

1. In the **Actions** menu, select **New Deployment Process Template**.

   The **Create Deployment Process Template** form appears.

2. Fill out the form.

For example:

a. Enter **Template name**.

b. Select the application to which the deployment process template applies.

c. Select the stage for which the template applies.

d. Select the owner, such as Release Engineer, for the template.

e. Optionally enter a description.

3. Click **OK**.

The deployment process template is saved.

Creating Deployment Process Templates Based on Existing Deployment Tasks

You can create a new deployment process template from deployment tasks in a release package.

The save as transfers all existing deployment task field information to the new item. You may edit any field before saving the new item. This enables you to save deployment tasks that you have created for a different stage.

The **Save as** creates new instances of all deployment tasks associated with the original deployment process template and links them to the new deployment process template.

The **Save as** is available from any state of the release package.

To create a deployment process template from existing tasks:

1. Navigate to a release package that has the deployment tasks you want to reuse.
2. Select the stage that has the deployment tasks you want to reuse.
3. Click **Save as template**.

   The **Save as template** form appears.
4. Fill out the **Save as template** form.

   For example:

   a. Enter the new template name in the **Rename Template** field.

   b. Select the application to which the deployment process template applies.

   c. Select the stage for which the template applies.

   d. Select the owner, such as Release Engineer, for the template.

   e. Optionally enter a description.
5. Click **OK**.

The deployment process template is saved.

Adding Deployment Tasks to a Deployment Process Template

You can add one or more deployment tasks to a deployment process template.

To add deployment tasks to a deployment process template:

1. Select the deployment process template to which you want to add a task.
2. Click **Add task.**

3. Enter information to create the task as documented in Planning Deployment Tasks [page 41].

Copying a Deployment Process Template

You can create a new deployment process template from an existing deployment process.

The copy transfers all existing field information to the new item. You may edit any field before saving the new item. This enables you to save deployment tasks that you have created for a different application or for a different stage, whereas an edit only allows you to change the application.

The copy creates new instances of all deployment tasks associated with the original deployment process template and links them to the new deployment process template.

The copy is available from any state of the deployment process template.

To copy a deployment process template:

1. Navigate to a deployment process template that you want to copy.
2. Click **Copy**.

   The **Make a copy** form appears.
3. Fill out the copy form.

   For example:

   a. Enter the new template name in the **Rename Template** field.

   b. Select the application to which the deployment process template applies.

   c. Select the stage for which the template applies.

   d. Select the owner, such as Release Engineer, for the template.

   e. Optionally enter a description.
4. Click **OK**.

   The new deployment process template is saved.

Saving As a Deployment Process Template

You can create a new deployment process template from deployment tasks in a release package. See Creating Deployment Process Templates Based on Existing Deployment Tasks [page 54].

Editing a Deployment Process Template

You can change any of the deployment process template fields during editing except stage.

Editing the deployment process template has no affect on any deployment tasks already applied to release packages.

TIP You can save a copy of the deployment process template for a different stage. See Copying a Deployment Process Template [page 55].

To edit a deployment process template:

1. Navigate to a deployment process template.
2. Click the edit link beside the view name.
3. Modify fields on the form as needed.
4. Click **OK**.

Retiring Deployment Process Templates

After the deployment process template is no longer needed, you may decide to retire it.

To retire a deployment process template:

1. Navigate to a deployment process template that is in the **Available** state.
2. Click **Retire**.

   The deployment process template is no longer available to be used.

Approving Deployment Process Templates

Before you can use a deployment process template to apply deployment tasks to release package stages, you must make the deployment process template available through actions required by the Deployment Process Template workflow.

See the simplified default workflow in Actioning Deployment Process Templates Through the Workflow [page 58].

In the default workflow, you may send a deployment process template for review and have it approved before it is applied to release packages.

To send a deployment process template for review:

1. Navigate to a deployment process template that is in the **Development** state.
2. Click **Send for review**.

   The deployment process template enters the **Review** state and the **Approve** action is available to approvers.

To approve a deployment process template:

1. Navigate to a deployment process template that is in the **Review** state.
2. Click **Approve**.

   The deployment process template enters the **Available** state and the **Retire** action is available to owners, such as release engineers.

Using Deployment Process Templates to Apply Deployment Tasks

Once you have developed your deployment process template and it has been approved and made available for use, you can apply it to release package stages to propagate your deployment tasks.

You can do the following:

- Apply a deployment process template to a release package.
- View the release packages that use a particular deployment process template.
- View the deployment process template to which a deployment task belongs.
- View the stage to which a deployment process template belongs.

Applying a Deployment Process Template to a Release Package

You can use a deployment process template in a release package to implement and standardize your release deployment processes. You first create the deployment process template for a specific stage and then apply it that stage in a release package.

In the default implementation of Serena Release Manager, if a deployment process template is in the **Available** workflow state, it can be accessed from each stage that is available for a release package.

For example, when you attempt to use a deployment process template for the INT stage of a release package, you can see and select from only those deployment process templates that are in the **Available** state and are associated with the INT stage.

See Approving Deployment Process Templates [page 56] .

CAUTION! When you use a template for a release package stage, any existing deployment tasks for that stage are overwritten by the deployment tasks from the deployment process template.

TIP Plan ahead by creating your deployment tasks in deployment process templates first. Consider putting all possible tasks that will be needed for each stage of a particular application in deployment process templates. You can then apply them to the release package stages and tailor the deployment tasks at the release package level as needed.

To use a deployment process template:

1. Navigate to the release package for which you want to use the deployment process template.
2. Select the **Deployment Tasks** tab.
3. Select the stage to which you want to apply the template's deployment tasks.
4. Click **Use a template**.

   The **Use a template** form appears with a list of available deployment process templates.
5. Select a deployment process template from the list.
6. Click **OK**.

Viewing Release Packages That Use this Deployment Process Template

You can view release packages to which a deployment process template is applied.

To view release packages for a deployment process template:

1. Navigate to the deployment process template.
2. Click **Release packages**.

   A list appears with information on all release packages to which this deployment process template has been applied.

Viewing the Deployment Process Template for a Task

A deployment task can belong to only one deployment process template. You can see the deployment process template to which a deployment task belongs.

To see the deployment process template to which a task belongs:

1. Navigate to the deployment task for which you want deployment process template information.

   If the deployment task is associated with a deployment process template, a template name appears in the Deployment Process Template field.

Viewing the Stage to Which a Deployment Process Template Belongs

A deployment process template is associated with a specific release stage, such as UAT or INT. You can see the stage to which a deployment process template belongs through the user interface.

To see the stage to which a deployment process template belongs:

- Navigate to a list of deployment process templates.

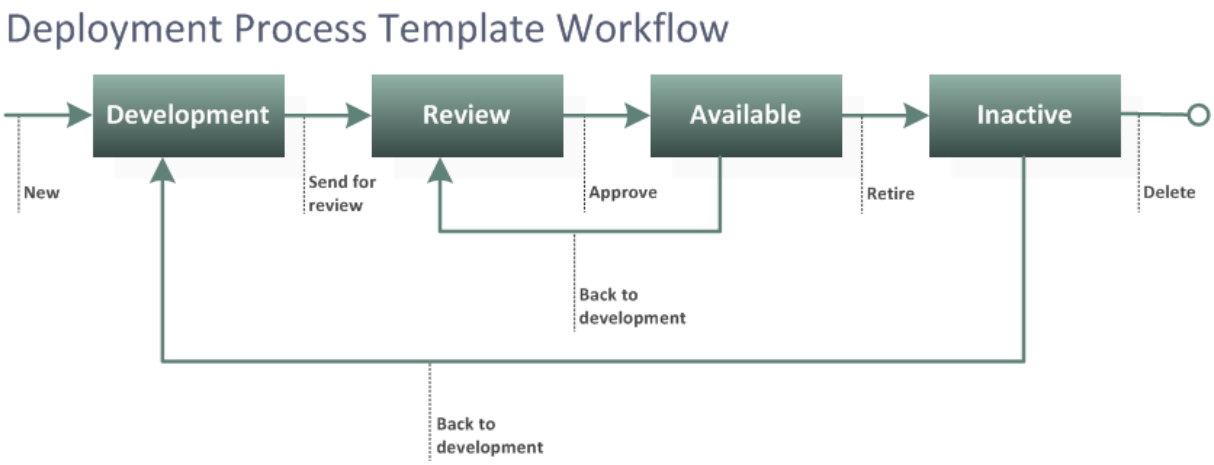
  The stage name appears in the Stage column.

- Navigate to the deployment process template view.

  The stage template name appears in the Stage field.

Actioning Deployment Process Templates Through the Workflow

As you work with deployment process templates, you are presented with buttons on the user interface for workflow actions that are available to you. The default workflow for a deployment process template is shown in the following figure.



The actions that appear here are:

- New
- Send for review
- Approve
- Back to development
- Retire
- Delete

For example, after you create a deployment process template, the deployment process template goes into the **Development** state, and the deployment process template form shows the next action, **Send for review**.

When the deployment process template is in the **Development** state, designated owners create deployment tasks for a stage of the application release. Once you click **Send for review**, the deployment process template is ready to be approved.

If you need to change the deployment tasks associated with the deployment process template, you'll need to click **Back to development**, make the changes while it is in the Development state, and again click **Send for review** to have the additional changes reviewed.

The designated reviewer clicks **Approve** to approve the deployment process template and it becomes **Available**. In this state, the template can be applied to a release package, so that its associated deployment tasks are copied to the corresponding stage of the release package.

When a deployment process template is no longer in use in your organization, you can click Retire to make it **Inactive** and remove it from the list of available templates. You can **Delete** deployment process templates that are in **Inactive** state to remove them from the system.

For information on how the workflows in Serena Release Manager work together, see Progressing Releases Through Their Lifecycles [page 15].

Deploying and Installing Releases

Deploy your release packages to execute defined deployment tasks.

Deploying Release Packages

Deployment includes all the activities that make a software system or application available for use in an environment, including the physical movement of files, execution of commands, and configuration of environment settings.

Deployment in Serena Release Manager is done at the release package level. Release packages are deployed when they reach the state in the lifecycle in which deployment is initiated. In the default workflow, this state is reached when the **Deploy** action is requested.

The deployment tasks associated with a release package stage are executed when you deploy the release package to that stage. The default release package workflow is shown in Actioning Release Packages Through the Workflow [page 39].

NOTE Depending on your organization's processes you may be able to deploy the release package independently of an application release, or you may be required to associate the release package to an application release before you deploy it.

To deploy a release package:

1. Select a release package that is in a state that is ready to deploy.
2. Click **Deploy**.

The release package is deployed and installed into the first stage deployment area. Since this is a major release, the first stage is INT.

This is where integration testing takes place. When the testing is completed, the release is ready to deploy to the next stage.

Upon successful execution, vault and automation deployment tasks move into the Completed state. Manual deployment tasks must be moved into Completed or Failed state by the task owner through the associated user interface actions.

If a vault or automation deployment task process fails for any reason during execution, the deployment task moves into the Failed state and is listed in the current owner's inbox, where the owner can take corrective action.

Deployment Task Execution

Once initiated by deployment of the release package, deployment task execution depends on the sequential ordering and the type of task: vault, automation, or manual. In the default implementation of Serena Release Manager, vault deployment tasks initiate pre-configured actions in your release vault, automation deployment tasks initiate pre-configured actions in Serena Release Automation, and manual tasks document and enforce any tasks that must be done manually before a release package can be deployed.

Vault and Automation Deployment Task Execution

Upon successful execution, vault and automation deployment tasks automatically move into the Completed state.

If a vault or automation deployment task process fails for any reason during execution, the deployment task moves into the Failed state and is listed in the current owner's inbox, where the owner can take corrective action.

Manual Deployment Task Execution

Manual deployment tasks must be moved into **Done** or **Failed** state by the task owner through the associated user interface actions. A manual task must be manually done or deleted before the release package to which it is associated can be deployed to the specified stage or before the next deployment task in the sequence can be executed.

Viewing Release Package Deployment Status

Deployment tasks may initiate actions on many environments, and these actions may involve a number of integrating products. Depending on how many tasks must be executed and how long each takes to complete, the deployment process may take a significant amount of time.

The state of a deployment task tells you information about the progress. By default, the state tells whether the task is in progress, complete, or failed. You can view deployment task **State** information through the **Inbox** page.

Since release packages may include many deployment tasks, it may not be practical to check every deployment task to determine the progress of a release package's deployment. A roll-up of the states of the deployment tasks are used to show release package deployment status.

To view release package deployment status, use the following:

- Monitoring Recent Activity of Release Packages [page 61]
- Viewing the Activity Log [page 62]
- Receiving Status Notifications [page 62]
- Resolving Deployment Problems [page 62]

Monitoring Recent Activity of Release Packages

You can monitor release package deployment progress in real-time using the **Activity** page.

You can do the following:

- You can select **Auto refresh**. This causes an eye-catcher to continuously refresh at the left of the release package if the release package has deployment activities in progress.
- You can designate the interval at which you want the refresh to retrieve status from the deployment tasks. Default values are 5, 10, 30, and 60 seconds.
- You can show release package events that occurred in the following time periods:
  - In last 24 hours
  - In last week
  - In last month
  - In last 3 months
  - In last year

To monitor release package activity:

1. Navigate to the **Activity** page.

2. Select the options you want for your page.
3. View the visual information, such as the eye-catcher showing in-progress activity and the color-coding showing current status.
4. Click an item ID in the list.

   The view of the item is displayed in the lower section of the content pane.

Viewing the Activity Log

Approval information and any additional messages generated by the deployment task processes are shown in the **Activity log** of the release package.

To view the activity log:

1. In a release package, click the **Reload Item** button () to refresh your view and see the latest status of a task.
2. Select the **History** tab and then select the **Activity log** tab.

Receiving Status Notifications

Depending on the Serena Release Manager notifications to which you are subscribed, you may receive e-mail notifications with deployment task progress information, such as state changes for approval or vault deployment tasks you created.

To subscribe to e-mail notifications, see your Serena Release Manager administrator.

Resolving Deployment Problems

If your deployment task fails and you are unable to determine the cause through the **Activity** page, **Activity** log, deployment task details, notification messages, or documentation, see your Serena Release Manager administrator. Information on troubleshooting is in *Serena ALM Installation and Configuration.*

Viewing Release Package Change History

You can view the history of changes to a release package so that you know who changed it, how they changed it, and when they changed it.

To view release package change history:

- View a release package and select the **History** tab, **Change history** sub-tab.
- Create and run custom reports in the **Reports** page. See Reporting On Release Progress and History [page 71].

Adding External Components to the Release Vault

Vault requests and vault templates enable you to easily bring components from disparate SCM and build management systems into Serena Release Manager and provide you with the option to identify components as definitive media.

Vault Request Overview

Vault requests enable you to easily bring components from disparate SCM and build management systems into Serena Release Manager and provide you with the option to identify components as definitive media.

You can create **Vault Templates** to enable the creation of vault requests in Serena Release Manager directly from Serena Development Manager (DVM) Development Packages. See the *Serena Development Manager Getting Started Guide* for information on using this feature.

Vault Requests enable you to:

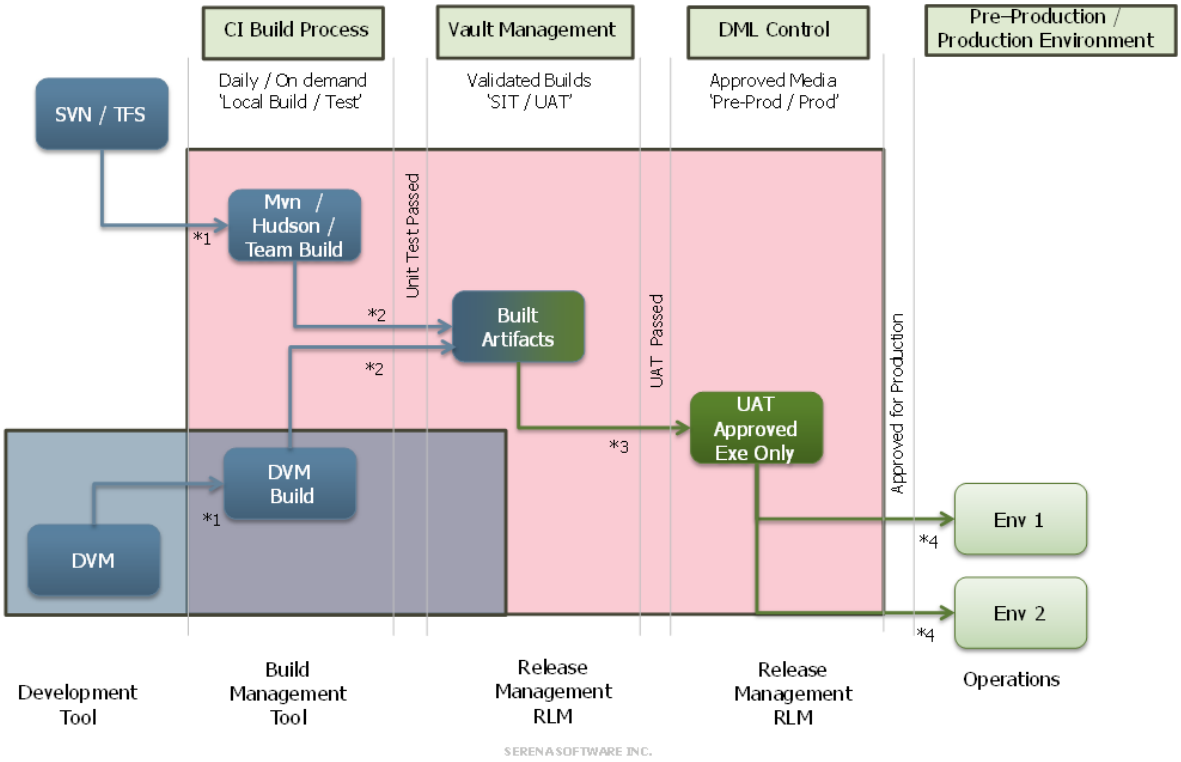
- Introduce components into the release vault from other Serena and non-Serena SCM systems.
- Flag baselines from Dimensions CM and DVM that have passed through User Acceptance Testing (UAT) to be part of the Definitive Media Library (DML).

  Definitive Media Library, or DML, is defined by ITIL as the location where all "definitive and approved" versions of media are stored, including software, documentation, and licenses. The DML is a single logical storage area, even if multiple physical storage areas are included.
- Introduce components into the release vault from build systems such as Jenkins, Hudson, and Maven.

NOTE The introduction of components from external build systems is not provided in the default user interface of Serena Release Manager. This must be done using the Web services provided for vault request functionality.

NOTE Documentation on configuring and extending the vault request functionality is in *Serena ALM Installation and Configuration*.

The following figure shows how bringing your components into Serena Release Manager vault management and adding the objects to your DML by flagging them as definitive media gives you the assurance that your releases contain the true "gold code" components that have passed from development through required testing stages, such as SIT and UAT, are ready for production, and are ready for acceptance by IT operations.



This figure shows how configuration items (CI) that have been built into sets of build artifacts are introduced into Serena Release Manager (RLM) from development tools and build managements tools, in this example Serena Development Manager (DVM) and Hudson / Team Build respectively.

After the built artifacts are tested through the Serena Release Manager release management stages and pass to the final stage, such as **Approved Exe**, **Production**, or **Live**, they are flagged as definitive media. Built artifacts can also brought directly in as definitive media, such as when you want to bring pre-tested vendor software into your Definitive Media Library (DML). Once the definitive media is in the vault, it is considered part of your DML and is ready for IT operations to distribute or release as part of the IT operations processes in your organization.

Adding Vault Requests

You can add vault requests to introduce work items or production-ready items into Serena Release Manager as either vault objects or definitive media. You define the information you want to bring in as the source information and the release vault baseline information as the target information as follows:

1. Source of the component set being brought into the release vault:
    a. Dimensions CM work area
    b. Dimensions CM baseline
    c. Build system such as Jenkins, Maven, or Hudson (if activated through the underlying Web services)
2. Target baseline in the release vault:
    a. New release vault baseline: flagged as vault object
    b. New release vault baseline: flagged as Definitive Media
3. Options let you specify filters to limit the target baseline to a subset of the information in the source baseline

Creating a Vault Request

You can create a vault request to introduce components into your release vault from a Dimensions CM work area or an existing Dimensions CM baseline.

To create a vault request:

1. In the **Actions** menu, select **New Vault Request**.

   The **Create Vault Requests** page appears.

2. Fill out the top, or shared, section of the form.

   For example, you would fill out the default form as follows:

   a. Enter a unique and descriptive **Vault request name**.

   b. For **Flag components as**, select the option as follows:

   - **DML** to flag the components as definitive media and add them to the release vault Definitive Media Library. These are components that should already have been tested externally to Serena Release Manager and are ready to be accepted by IT operations as production-ready components, or "Gold" code.
   - **Vault**: to flag the components as vault objects and add them to the regular release vault to pass through the defined testing stages on the path to production.

3. Select the **Source** tab and fill out the form to select the location from which you are delivering components into the release vault. Choose one of the following based on where the component sets you are adding currently reside.

   Adding from a Dimensions CM Work Area

   If you are introducing component sets to your Serena Release Manager system vault from external systems other than the Dimensions CM that is defined as a deployment unit provider for your Serena Release Manager system:

   a. If you haven't done so already, copy the components to a Dimensions CM work area that is known to your Dimensions CM agent. You should do this outside of Serena Release Manager using your usual procedures.

   b. For **Source type**, select **Dimensions CM work area**.

   c. Select a work area from the list.

   d. Optionally specify the folder in the work area that contains the components you want to deliver. If no folder is specified, all folders in the work area are included.

   Adding from a Dimensions CM Baseline

   If you are introducing component sets to your Serena Release Manager system vault from the Dimensions CM system that is defined as a deployment unit provider for your Serena Release Manager system:

   a. For **Source type**, select **Dimensions CM baseline**.

   b. Select a Dimensions CM product.

   c. Select a Dimensions CM project.

   d. Select a Dimensions CM baseline to use as your source baseline.

   Adding from a build system such as Hudson, Maven, or Jenkins

   If you are introducing component sets directly from a build system into your Serena Release Manager system vault, check with your Serena Release Manager Administrator to find out how

your particular system is implemented. This functionality is not yet provided in the default user interface of Serena Release Manager.

4. Select the **Target** tab and fill out the form to select the location in the release vault to which you are delivering components. For example:

   a. In **Baseline stem**, specify a prefix for the baseline to be created in the target release vault and project specified in configuration files.

      This gives a unique identifier to the baseline name that is automatically generated by the vault request target baseline generation process.

   NOTE In the default implementation of Serena Release Manager, the target project in the release vault to which you are delivering components is defined in your vault provider configuration file. For more information, see *Serena ALM Installation and Configuration*, "Vault Request Configuration".

5. Select the **Options** tab and fill out the form to filter the results selected in the source baseline. This refines the component set that will be created in the target baseline upon execution of the vault request. For example:

   Adding Included and Excluded filters

   a. In the **Included filters** input field, identify the full or partial names of files that you want to include in the target baseline. Type full file names or standard wildcard expressions to identify partial file names. For example `**/*.exe | **/*.bin`.

   b. Click **Add** to add the input field information to the list of filters.

   c. To clear all of the filters at once, click **Clear**.

      TIP To selectively clear, copy and paste the filters become you click **Clear**. Copy the text into the **Included filters** input field, edit it as needed, then add just the information you want to keep as included filters.

   d. In the **Excluded filters** input field, identify the full or partial names of files that you want to exclude from the target baseline. Type full file names or standard wildcard expressions to identify partial file names. For example `**/*.log | **/*.txt`.

   e. Use the same procedure as was used to add **Included filters** to add your **Excluded filters**.

   f. Select a design part for the components that you want to include in the target baseline.

   g. Select a vault stage for the components that you want to include in the target baseline.

6. Click **OK**.

   The vault request is saved.

Editing a Vault Request

You can edit a vault request when it is in any state.

To edit a vault request:

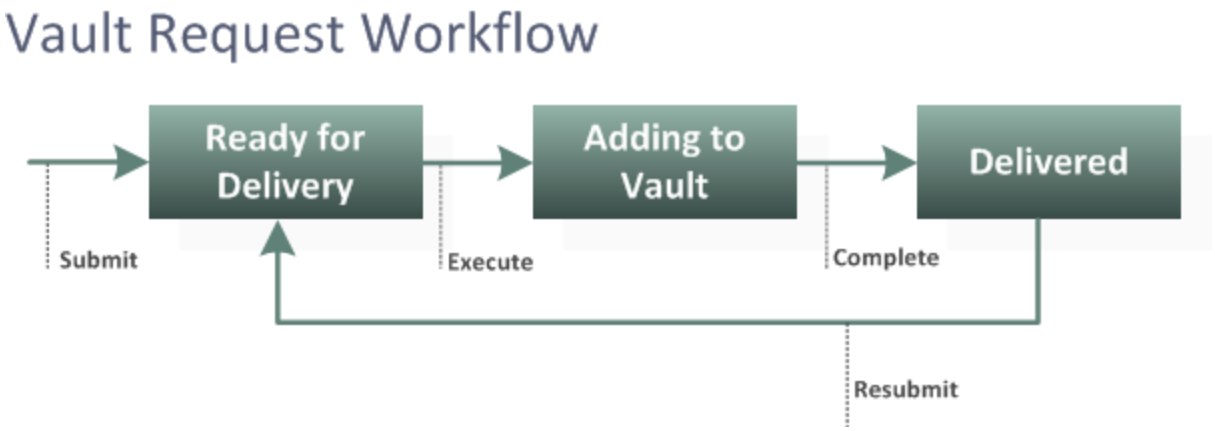
1. Navigate to a vault request.
2. Click the edit link beside the view name.
3. Modify fields on the form as needed.
4. Click **OK**.

Delivering Components to the Vault

In the default workflow, you may deliver baselines to the pre-defined release vault product ID in Dimensions CM by executing a vault request that is **Ready for Delivery**.

Actioning Vault Requests Through the Workflow

As you work with vault requests, you are presented with buttons on the user interface for workflow actions that are available to you. The default workflow for a vault request is shown in the following figure.



The actions that appear here are:

- Submit (create)
- Execute
- Complete
- Resubmit

For example, after you submit, or create, a vault request, the vault request goes into the **Ready for Delivery** state, and the vault request dialog box shows the next action, **Execute**.

When the vault request is in the **Delivered** state, the vault request is considered complete. You can click **Resubmit** to return the vault request to the **Ready for Delivery** state. From that state you can execute the vault request again.

For information on how the workflows in Serena Release Manager work together, see Progressing Releases Through Their Lifecycles [page 15].

Adding Vault Templates

You can create vault templates to enable the creation of vault requests in Serena Release Manager directly from Serena Development Manager (DVM) Development Packages. See the *Serena Development Manager Getting Started Guide* for information on using this feature.

Creating a Vault Template

You may create vault templates to be used to bring component sets into Serena Release Manager in the future. When you create a vault template, you can define the target vault information and filters. These can be used to add baselines directly from DVM through the Serena Development Manager (DVM) to Release Manager integration. Since the source information is selected from an external system, this must be selected when you create the actual vault request after applying the vault template.

To create a vault template:

1. In the **Actions** menu, select **New Vault Template**.

The **Create Vault Template** page appears.

2. Fill out the top, or shared, section of the form.

   For example, you would fill out the default form as follows:

   a. Enter a unique and descriptive **Vault** template **name**.

   b. For **Flag components as**, select the option as follows:

   - **DML** to flag the components as definitive media and add them to the release vault Definitive Media Library. These are components that should already have been tested externally to Serena Release Manager and are ready to be accepted by IT operations as production-ready components, or "Gold" code.
   - **Vault**: to flag the components as vault objects and add them to the regular release vault to pass through the defined testing stages on the path to production.

3. Select the **Target** tab and fill out the form to select the location in the release vault to which you are delivering components. For example:

   a. In **Baseline stem**, specify a prefix for the baseline to be created in the target release vault and project specified in configuration files.

   This gives a unique identifier to the baseline name that is automatically generated by the vault request target baseline generation process.

NOTE In the default implementation of Serena Release Manager, the target project in the release vault to which you are delivering components is defined in your vault provider configuration file. For more information, see *Serena ALM Installation and Configuration*, "Vault Request Configuration".

4. Select the **Options** tab and fill out the form to filter the results selected in the source baseline. This refines the component set that will be created in the target baseline upon execution of the vault request that uses this vault template. For example:

   Adding Included and Excluded filters

   a. In the **Included filters** input field, identify the full or partial names of files that you want to include in the target baseline. Type full file names or standard wildcard expressions to identify partial file names. For example `**/*.exe | **/*.bin`.

   b. Click **Add** to add the input field information to the list of filters.

   c. To clear all of the filters at once, click **Clear**.

   TIP To selectively clear, copy and paste the filters become you click **Clear**. Copy the text into the **Included filters** input field, edit it as needed, then add just the information you want to keep as included filters.

   d. In the **Excluded filters** input field, identify the full or partial names of files that you want to exclude from the target baseline. Type full file names or standard wildcard expressions to identify partial file names. For example `**/*.log | **/*.txt`.

   e. Use the same procedure as was used to add **Included filters** to add your **Excluded filters**.

   f. Select a design part for the components that you want to include in the target baseline.

   g. Select a vault stage for the components that you want to include in the target baseline.

5. Click **OK**.

The vault template is saved.

Editing a Vault Template

You can edit a vault request when it is in any state.

To edit an vault template:

1. Navigate to a vault template.
2. Click the edit link beside the view name.
3. Modify fields on the form as needed.
4. Click **OK**.

Retiring Vault Templates

After the vault template is no longer needed, you may decide to retire it.

To retire a vault template:

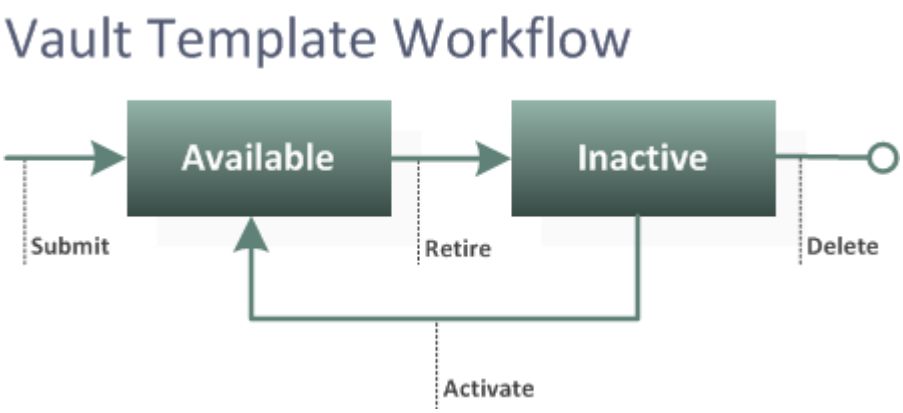
1. Navigate to a vault template that is in the **Available** state.
2. Click **Retire**.

   The vault template is no longer available to be used. However, you can reactivate it by clicking **Activate**.

See the simplified default workflow in Actioning Vault Requests Through the Workflow [page 67].

Actioning Vault Templates Through the Workflow

As you work with vault templates, you are presented with buttons on the user interface for workflow actions that are available to you. The default workflow for a vault template is shown in the following figure.



The actions that appear here are:

- Submit (create)
- Retire
- Delete
- Activate

For example, after you submit, or create, a vault template, the vault template goes into the **Available** state, and the vault template dialog box shows the next action, **Retire**.

When the vault request is in the **Inactive** state, the vault template is still available to use. You can click **Activate** to make the vault template available to use again.

For information on how the workflows in Serena Release Manager work together, see Progressing Releases Through Their Lifecycles [page 15].

Reporting On Release Progress and History

Run reports to track the progress of releases through their lifecycles and review historical release information.

General Reports Overview

The general reports are Serena Business Manager reports that retrieve relevant release management data from Serena Release Manager. At any point in the release management process you can report on current or historic information.

Reports are provided for the following categories:

- Release trains
- Application releases
- Release packages
- Deployment tasks
- Deployment process templates
- Environments

Selecting and Viewing Reports

To select and view reports:

1. In the **Reports** page toolbar, select a category in the **Filter by** drop-down menu, such as **Release Trains**.
2. Select a view in the **View** drop-down menu.

   Typically the report results appear immediately. If the report requires additional selection criteria, enter the criteria and click **Run Report**.

For information on creating reports, see *Serena ALM Installation and Configuration*.

NOTE A list of general reports that are provided in the default implementation of Serena Release Manager and rules for creating and updating reports are given in *Serena ALM Installation and Configuration*.

Checking Availability of Environments

You can check the availability of servers for your release packages through the following:

- List all servers defined to Serena Release Manager through the **Environments** page.

- Report on the number of servers at a particular release train or release package stage.

Using the Calendars for Quick Assessment

Two calendar views are provided in the default version of Serena Release Manager, the Gantt view and the Calendar view. Use these to quickly assess the schedules of your release trains.

Tracking Release Package Activity

You can track release package activity through the **Activity** page. This lets you see past and current release package deployment information.

Closed-loop Change Request Tracking

The Serena Service Manager integration with Serena Release Manager offers closed-loop change request tracking driven from Serena Release Manager.

Serena Service Manager Integration Overview

The default implementation of Serena Release Manager includes a two-way integration with Serena Service Manager (SSM) which is driven by Serena Release Manager. If you are using this integration, when you associate an SSM Change Management change request with a Serena Release Manager RFC in a release train, the change request in SSM will be updated to show the Serena Release Manager release train with which it is associated.

For more information on the integration from within the SSM process app, see the *Serena Service Manager User's Guide*.

For information on configuring the integration in Serena Release Manager, see the *Serena ALM Installation and Configuration Guide*.

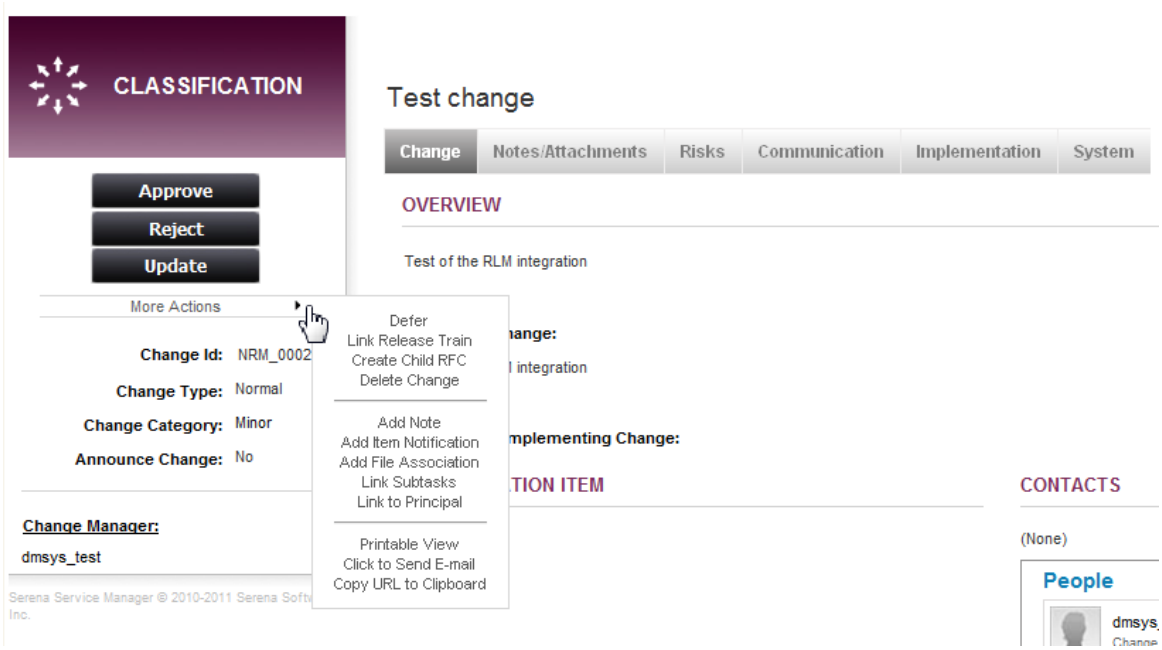
NOTE Serena Service Manager is sold separately from the Serena Release Manager suite.

Associating SSM Requests with Release Trains from SSM

You can associate SSM requests with RFCs from either Serena Release Manager or SSM. In SSM, you can link SSM change management change requests to Serena Release Manager release trains. In Serena Release Manager you associate RFCs to release trains by selecting from a list of SSM change requests. To associate the RFCs from Serena Release Manager, see Associating RFCs with Release Trains [page 23] .

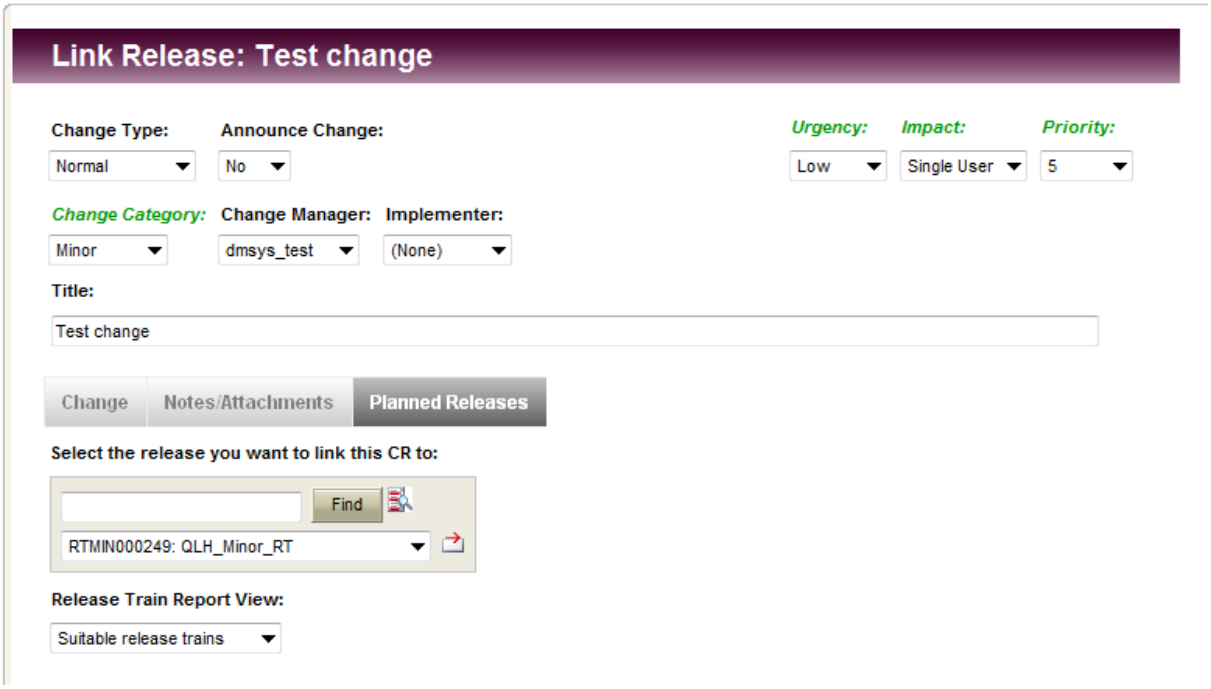
To associate the SSM requests with Release Trains from SSM:

1. Navigate to a request.
2. From the **More Actions** menu, click **Link Release Train** as shown in the following figure.

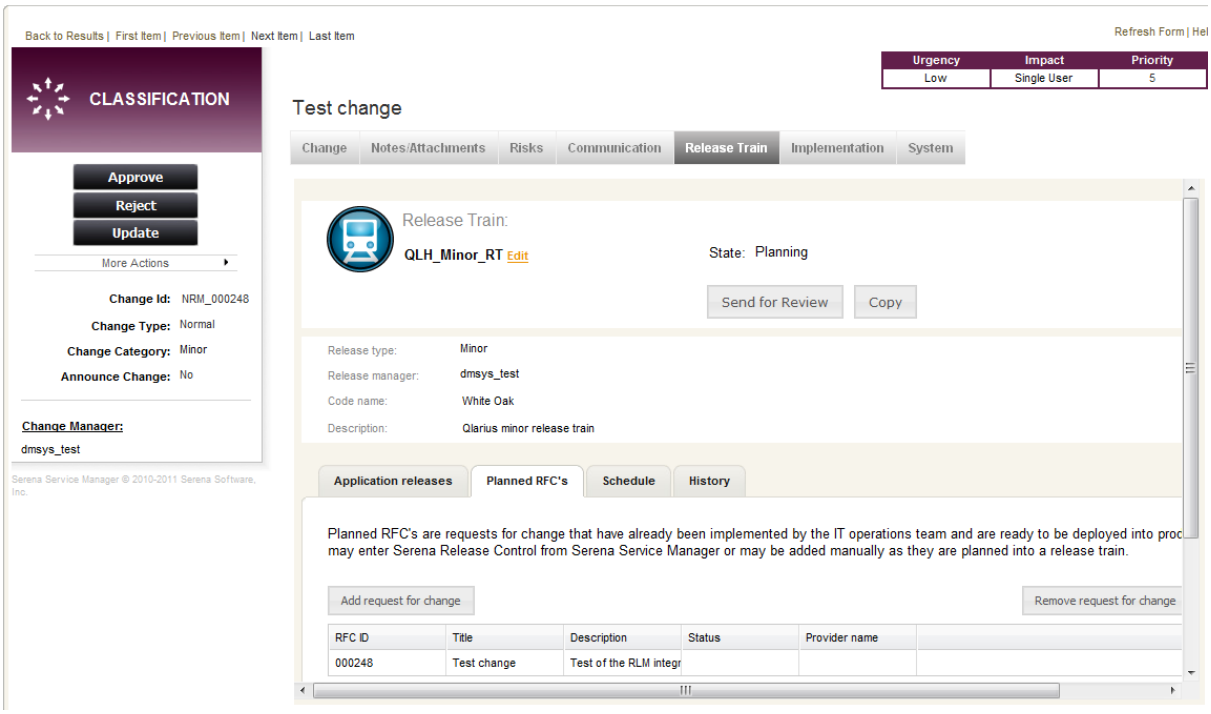


After you click **Link Release Train**, the **Link Release** form appears.

3. In the **Link Release** form, select the **Planned Releases** tab and select the release train to which you want to link the change request as shown in the following figure.



After a release train is linked to the SSM change request, the Serena Release Manager Release Train form appears under the SSM **Release Train** tab, and the release train actions are available directly from SSM as shown in the following figure.



Viewing SSM Requests on the Release Train Calendar

If configured to do so, the non-Gantt calendar displays the implementation dates for SSM requests along with the Production dates for release trains.

Glossary

The following is a glossary of the terminology used in Serena Release Manager.

Application

In Serena Release Manager, a program or system in computers or devices that addresses some problem as a whole. Some types of applications are commercial websites, batch and online business systems, and packaged software. Examples: an online teller system for a bank, an online music store, an accounting system.

In Serena Release Automation, a high-level design construct whose design and ultimate execution allow achieving a fully-automated implementation of a server-based system.

In ChangeMan ZMF, a user software project or domain being implemented. Named and defined by the application administrator; can be a mnemonic. User restrictions are set for it. Type in the exact name of the application in an application, or select from a list of applications by typing a pattern, or use masking. Possible References: APPL, PROJECT, LOCAL.

Application Owner

A possible role in Serena Release Manager that is the business owner of an application. An application owner is responsible for approving an application release during the planning stage and receives notifications of application release status.

Application Release

Represents a version of an application or project, where the application or project architecture is specified by components. One or more release packages are associated with each application release.

Approval

An action taken by an authorized user to progress an entity in a logical workflow. Approvals are typically required at states in the workflow that require expert review before the associated objects proceed to the next state.

Architecture

In Serena Release Automation, a generic design of a possible physical configuration for the implementation of an application.

Area

See Deployment Area [page 77].

Automation Deployment Task

A type of deployment task that integrates with release automation to automate installation and configuration tasks through the release automation software (such as Serena Release Automation) as part of the deployment process.

Build Manager

A possible role in Serena Release Manager that performs builds for a release. This role may also be the Installation Manager [page 78] in some organizations.

Business Change Request (BCR)

Represent customer or business unit change requests that affect specific application releases. BCRs are associated with application releases and are typically associated with tickets from help desk or incident management systems such as Serena Service Manager and other SBM solutions.

Change

In Serena Service Manager, an item submitted into the Change Management process app, which helps ensure that standardized methods and procedures are used for efficient and prompt handling of all changes to controlled IT infrastructure.

If you are using the SSM to Serena Release Manager integration, you associate SSM Changes with RFCs. See Request for Change (RFC) [page 81].

Change Manager	A possible role in Serena Release Manager that is responsible for the IT Operations for the systems where the pre-production and production release environments reside. A change manager approves deployment into pre-production and production environments.
Change Request	See Business Change Request (BCR) [page 76], Development Change Request (DCR) [page 77], and Request for Change (RFC) [page 81].
Change Package	In ChangeMan ZMF, a group of changes to user software. Three levels include simple, super/complex, and participating, made up of members of partitioned data sets such as JCL, source code, copybooks, and so on.
Calendar	The visual representation of release trains and their schedules in Serena Release Manager.
DML	A Definitive Media Library, or DML, is defined by ITIL as the location where all "definitive and approved" versions of media are stored, including software, documentation, and licenses. The DML is a single logical storage area, even if multiple physical storage areas are included.
Deployment	The activities that make a software system or application available for use in an environment, including the physical movement of files, execution of commands, and configuration of environment settings.
Deployment Area	Represents the server or environment to which deployment units are copied during deployment.
Deployment Process Template	Enables you to create and copy sets of deployment tasks for reuse in different stages, release packages, and applications.
Deployment Task	In Serena Release Manager, an action to be executed as part of the process of deploying a release package into a specific stage environment.
Deployment Task Types	Determine the forms presented in the user interface and the processes executed during deployment of release packages. Default deployment task types include manual, vault, and automation.
Deployment Unit (DU)	Represents a set of deployable components. DUs are associated with release packages, and are typically associated with deployable components held in systems used to manage development processes, such as Serena Dimensions CM, Serena Development Manager, and Serena ChangeMan ZMF. Examples of sets of deployable components include Dimensions CM baselines with build outputs and ChangeMan ZMF change packages.
Development Change Request (DCR)	Represent delivered changes from the development process. DCRs are associated with release packages and are typically associated with change requests from systems used to manage development processes, such as Serena Service Manager, the SBM Incident Management solution, Serena Development Manager, and Serena Dimensions CM.
Development Manager	A possible role in Serena Release Manager that is responsible for and approves development activities for a release. A development manager would typically be consulted during the release management process and provide approval on the content of a release package. This is a primary role in Serena Development Manager (DVM).

Emergency Change Package

In ChangeMan ZMF, unplanned change packages that are scheduled for installation outside of normal business hours. Emergency change packages are automatically assigned a shorter approval list.

Emergency Release Package

In Serena Release Manager, a release type for unplanned release trains or release packages that must be expedited through the minimum required release stages. In the default implementation, emergency release packages are deployed only to the Production Deployment stage.

See Release Type [page 81].

Environment

In general, a physical computer and operating system on which software resides; for example, a Windows or UNIX server.

In Serena Release Manager, represents one or more physical or logical computers or systems to which a release package may be deployed.

In Serena Release Automation, one of any number of real, physical implementations of a specific architecture that was designed for the application.

Global Stage Lifecycle

In Dimensions CM, the lifecycle that items follow that controls which versions are included in configurations and builds of the project or stream. This lifecycle is defined for the base database. Deployment areas for a project/stream can be associated with these stages so that item files are copied to those areas when they are deployed to the corresponding stage.

Incident

An item in the Incident Management SBM solution. An incident may be mapped to a Serena Release Manager Development Change Request (DCR) [page 77].

Installation

In ChangeMan ZMF, the process of either updating the production libraries, or in the case of temporary change packages, concatenating the temporary production libraries before the current production libraries.

Change packages are turned over to the ZMF installation scheduler after the final approval. For change packages that have a manual scheduler, ZMF submits the first installation job as soon as the last approval is entered, regardless of the installation date on the ZMF calendar.

Installation Manager

A possible role in Serena Release Manager that ensures that the deployment, or installation, of a release is done correctly and completely. An installation manager is assigned manual deployment tasks in Serena Release Manager and is responsible for deployment of request packages into environments.

This role may also be the Build Manager [page 76] in some organizations.

Integration Test (INT)

Testing of all application release components together to ensure that the application release's functions properly as a whole. This testing typically follows unit testing in a testing lifecycle.

In the default implementation of Serena Release Manager, Integration Test is also a stage in release train and release package schedules.

Issue

An item in the Issue Defect Management (IDM) SBM solution. An issue may be mapped to a Serena Release Manager Development Change Request (DCR) [page 77].

Item In general, a software entity or object that can be listed or selected.

In SBM, primary items, such as issues and incidents, are records that are tracked by an application. Primary items are organized in projects, which are assigned to a workflow. You submit primary items into a specific project, and those items follow the project's workflow to completion.

In Dimensions CM, an object that represents the physical implementation of an application component. For example, an item in a software application could be source code, an executable file, a specification, or a user guide. An item in a website could be an HTML file, a script, or an image file.

Lifecycle The set of states and rules for transitions between states defined for a particular object type.

Major Release Package In Serena Release Manager, a release type for release trains or release packages that require the maximum release stages. In the default implementation, major release packages are deployed through the INT, UAT, and Production Deployment stages.

See Release Type [page 81].

Manual Deployment Task A type of deployment task. An action to be executed by a person as part of the deployment process.

Minor Release Package In Serena Release Manager, a release type for release trains or release packages that require only some of the release stages. In the default implementation, minor release packages are deployed only to the UAT and Production Deployment stages.

See Release Type [page 81].

Pre-Production The environments to which production-ready application releases are deployed and held before deployment to production. Pre-production can be thought of as final staging or production readiness.

In the default implementation of Serena Release Manager, Pre-Production is also a stage in release train and release package schedules.

Process In Serena Release Automation, a high-level sequence of planned activity involving one, multiple, or all components of an application.

Production The environments to which production-ready application releases are ultimately deployed. Production can be thought of as live or real-time environments.

Production Deployment A stage in release train and release package schedules in the default implementation of Serena Release Manager.

Product Test Testing of a product’s features and functionality. This testing typically follows unit testing in a testing lifecycle.

In the default implementation of Serena Release Manager, Product Test is also a stage in release train and release package schedules.

Project In general, a project is an organizing entity made up of a collection of related objects.

Promotion Level In ChangeMan ZMF, manages test libraries. For example, promotion Level 10 could be defined for Integration Testing, promotion Level 20 could be defined for User Acceptance Testing, and promotion Level 30 could be defined for Production Readiness Testing.

QA Manager A possible role in Serena Release Manager that is responsible for and approves testing activities for a release.

Release See Application Release [page 76].

Release Automation The part of release management that automates application installation and configuration tasks. In the default implementation of Serena Release Manager, the release automation is done using Serena Release Automation.

Release Calendar See Calendar [page 77].

Release Control The part of release management that controls the processes. In the default implementation of Serena Release Manager, release control is powered by Serena Business Manager (SBM).

Release Engineer The default role in Serena Release Manager that is responsible for the automating the release deployment and resolving any deployment failures. For example, a release engineer may create deployment tasks for release packages.

This role may be the Installation Manager [page 78] or Build Manager [page 76] in some organizations.

Release Management The management of defining and deploying versions of applications into production environments. The activities of release management include defining, scheduling, monitoring, changing, provisioning, and analyzing history of multiple releases deployed into production environments.

Release Manager The default role in Serena Release Manager that is responsible for releases within a particular organization. A release manager manages and monitors releases, plans releases in collaboration with development managers, and responds to successes, failures, and other statuses.

Release Package A portion of IT or service infrastructure normally built, deployed, tested, and released together. Release packages define the set of changes to be deployed and drive the deployment processes.

Release Runbook Represents a set of tasks and activities to be performed in a specific order to implement an application release. The release runbook is implemented in association with a release train.

Release Schedule See Release Train [page 80].

Release Stage Gate that the release goes through on its path into production. Typical stages in a release are integration test (INT), user acceptance test (UAT), and production deployment (PROD). A stage may have one or more environments related to it. Releases may be deployed into an environment based on availability.

Release Train Provides a published schedule of changes to production. One or more application releases are associated with each release train.

Release Type Drives release policies on what types of changes may be delivered for release trains and determines the stages release packages are deployed through. In the default implementation of Serena Release Manager, release types include major, minor, and emergency.

See also Release Stage [page 80].

Release Vault The location where all development tested versions of media, typically software or documentation, are stored. In the default implementation of Serena Release Manager, the provided release vaults are Dimensions CM and ChangeMan ZMF.

Request In Dimensions CM, an object used to report a defect, suggest an enhancement, or detail other work for a particular product. Requests can include external files (such as requirements or specification documents) as attachments. A request may be mapped to a Serena Release Manager Development Change Request (DCR) [page 77].

See also Request for Change (RFC) [page 81], Business Change Request (BCR) [page 76], and Development Change Request (DCR) [page 77].

Request for Change (RFC) Represent operational changes that may affect multiple applications or implement system infrastructure changes within an enterprise. RFCs are associated with release trains and are typically associated with tickets from service management systems such as Serena Service Manager.

Server A computer that manages centralized data storage or network communications resources. A server provides and organizes access to these resources for other computers linked to it. (American Heritage Science Dictionary)

In Serena Release Automation, the actual physical machines on which a Web-based application will ultimately run.

Server Type In Serena Release Automation, an abstract representation of a server specified for a particular architecture.

Site In ChangeMan ZMF, locations where change packages are stored. These are typically IBM z/OS LPARs, either local or remote.

Software Application See Application [page 76].

Stage In Serena Release Manager, a Release Stage [page 80].

In Dimensions CM, a stage in the Global Stage Lifecycle [page 78].

State A key part of a Serena Business Manager (SBM) workflow. States are positions that a primary item, such as a release package, resides in while moving through a process workflow. States can also be considered a stopping point along a workflow's path.

Status A measure of where an item is in relation to a preset goal. For a release train, status is the lowest status of the application releases associated with it. Release managers manually set the status for application releases. Release package status is automatically set by Serena Release Manager based on movement of the release package through underlying SBM states.

Status Indicator A visual representation of status, such as green for on schedule, yellow for schedule at risk, and red for behind schedule.

System	Depending on the context, can refer to a software application, to an environment, or both.
Unit Testing	Testing of individual or groups of similar components of a software application as part of the development process. This testing is typically done by software engineers who are developing the components and is done before the components are considered complete and packaged for deployment (baselined).
User Acceptance Test (UAT)	Testing of an application release by key users to evaluate the application release's compliance with its specified user requirements. This testing typically follows system testing in a testing lifecycle. In the default implementation of Serena Release Manager, UAT is also a stage in release train and release package schedules.
Vault	See Release Vault [page 81].
Vault Deployment Task	A type of deployment task that integrates with a release vault, such as Dimensions CM or ChangeMan ZMF, for secure deployment of deployment units. For example, Dimensions CM vault deployment tasks move baselines securely to environments.
Vault Request	Enables you to easily bring components from disparate SCM and build management systems into Serena Release Manager and provides you with the option to identify components as definitive media.
Vault Template	Created either standalone or from vault requests to more easily create similar vault requests.
Workflow	In Serena Release Manager and SBM, a collection of states, transitions, and fields that define an organization's tracking process.