



# **SERENA<sup>®</sup>**

# **Development Manager 1.2**

**Getting Started Guide**

Serena Proprietary and Confidential Information

Copyright © 2011-2012 Serena Software, Inc. All rights reserved.

This document, as well as the software described in it, is furnished under license and may be used or copied only in accordance with the terms of such license. Except as permitted by such license, no part of this publication may be reproduced, photocopied, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, recording, or otherwise, without the prior written permission of Serena. Any reproduction of such software product user documentation, regardless of whether the documentation is reproduced in whole or in part, must be accompanied by this copyright statement in its entirety, without modification.

This document contains proprietary and confidential information, and no reproduction or dissemination of any information contained herein is allowed without the express permission of Serena Software.

The content of this document is furnished for informational use only, is subject to change without notice, and should not be construed as a commitment by Serena. Serena assumes no responsibility or liability for any errors or inaccuracies that may appear in this document.

Third party programs included with the Dimensions product are subject to a restricted use license and can only be used in conjunction with Dimensions.

### **Trademarks**

Serena, StarTool, PVCS, Comparex, Dimensions, Mashup Composer, Prototype Composer, and ChangeMan are registered trademarks of Serena Software, Inc. The Serena logo and Meritage are trademarks of Serena Software, Inc. All other products or company names are used for identification purposes only, and may be trademarks of their respective owners.

### **U.S. Government Rights**

Any Software product acquired by Licensee under this Agreement for or on behalf of the U.S. Government, its agencies and instrumentalities is "commercial software" as defined by the FAR. Use, duplication, and disclosure by the U.S. Government is subject to the restrictions set forth in the license under which the Software was acquired. The manufacturer is Serena Software, Inc., 1900 Seaport Boulevard, 2nd Floor, Redwood City, California 94063-5587.

Publication date: February 2012

---

# Table of Contents

---

	<b>Welcome to Serena Development Manager . . . . .</b>	<b>5</b>
	Contacting Technical Support . . . . .	5
	Platform Support . . . . .	5
	Demonstrations . . . . .	5
	Documentation . . . . .	6
<i>Chapter 1</i>	<b>Introduction to Serena Development Manager . . . . .</b>	<b>7</b>
	Welcome to Serena ALM . . . . .	8
	Key Benefits of Development Manager . . . . .	8
	Using the Development Manager Components . . . . .	9
	Usage Overview . . . . .	9
	Detailed Example Use Case . . . . .	12
	User Roles . . . . .	12
	Process Example . . . . .	13
<i>Chapter 2</i>	<b>Understanding the Out-of-the-Box Workflows . . . . .</b>	<b>17</b>
	About the Out-of-the-Box Development Workflows . . . . .	18
	Object Types and Relationships . . . . .	18
	ALM Projects Workflow . . . . .	19
	Dev Change Requests Workflow . . . . .	20
	SBM Workflow for Change Requests . . . . .	20
	Integrating Change Requests with HP Quality Center / ALM . . . . .	21
	Dev Tasks Workflow . . . . .	21
	Dev Packages Workflow . . . . .	23
	Dev Packages Workflow Diagram . . . . .	24
	Supporting Your Development Process . . . . .	25
	What Types of Development Projects Will You Manage? . . . . .	25
	Is Your Development Process Agile? . . . . .	26
	What Are Your Workflow States? . . . . .	26
	How Do You Manage Test Cases? . . . . .	27
	What Reports Does Your Organization Need? . . . . .	27
<i>Chapter 3</i>	<b>Getting Started Using Development Manager . . . . .</b>	<b>29</b>
	Introduction . . . . .	30
	Logging In to Development Control . . . . .	30
	Creating and Managing Projects . . . . .	30
	Creating a Project . . . . .	31
	Example Project in ALM Projects . . . . .	34
	Displaying and Creating Related Change Requests . . . . .	36
	Creating and Relating Development Packages . . . . .	36
	Creating and Working on Change Requests . . . . .	36
	Finding Change Requests . . . . .	37

---

Creating New Change Requests . . . . .	37
Working on Change Requests . . . . .	39
Creating and Working on Tasks . . . . .	40
Finding Tasks . . . . .	40
Creating New Tasks . . . . .	41
Working on Tasks . . . . .	42
Creating and Working with Development Packages . . . . .	42
Finding Packages. . . . .	42
Creating New Packages . . . . .	43
Working on Packages. . . . .	44
Creating New Dimensions CM Baselines . . . . .	44
Starting a Build. . . . .	47
Displaying Project Metrics with the Serena® Dashboard . . . . .	48
Included KPI & Metric Reports. . . . .	48
Getting to Know Dimensions CM . . . . .	51
Integrating with HP Quality Center. . . . .	52

*Chapter 4*

<b>Using the Requirements Manager Integration . . . . .</b>	<b>53</b>
Introduction . . . . .	54
Creating Dev Change Requests from Requirement Distribution Tasks. . . . .	55
Viewing a Distribution Task . . . . .	55
Creating a Single Dev Change Request from One or More Requirements	57
Creating one or More Dev Change Requests From Requirements. . . . .	59
Creating Dev Change Requests for the Requirements in an ALM Project . . . . .	61
Viewing the Requirements Related to a Project . . . . .	61
Viewing and Associating Requirements for a Dev Change Request . . . . .	63
Viewing the Requirements for a Dev Package . . . . .	65

# Welcome to Serena Development Manager

---

Thank you for choosing Development Manager to plan and control your releases.

Serena® Development Manager (Powered by Serena Business Manager) enables you to plan, control, and automate all your development processes from definition to quality assurance and release approval with start-to-finish traceability and end-to-end visibility across distributed environments.

**Audience and Scope** This document is intended for personnel who participate in the processes of managing releases using Serena Development Manager.

**Before You Begin** See the Readme for the latest updates and corrections for this document.

## Contacting Technical Support

Serena provides technical support for all registered users of this product, including limited installation support for the first 30 days. If you need support after that time, contact Serena Support at the following URL and follow the instructions:

<http://www.serena.com/support>

Language-specific technical support is available during local business hours. For all other hours, technical support is provided in English.

## Platform Support

For details of supported server and client platforms, third party integrations, and Serena Integrations, see the Serena Release Plan for Serena Development Manager at:

<http://support.serena.com/roadmap/index.aspx>

and select the product and release version.

## Demonstrations

Demonstrations of Serena product features can be viewed at the following public Web site:

<http://courseware.serena.com>

---

# Documentation

The Development Manager documentation set includes the following manuals and tutorials.

Manual or Tutorial	Description
<i>Serena Development Manager Installation and Configuration Guide</i>	Describes how to install and configure the Serena Dimensions CM suite of products. (PDF manual)
<i>Serena Development Manager Getting Started Guide</i>	Gives an overview of Serena Development manager (PDF manual).
<i>Serena Development Manager Connector for HP Quality Center / ALM</i>	Describes how to configure and use the Development Manager web services that interact with HP Quality Center.
<i>Serena Development Manager Web Services Reference</i>	Provides Information on the web services provided for Development Manager.
<i>Serena Development Manager Readme</i>	Provides information about platform requirements and known issues for Development Manager.

Access the documentation and tutorials for the Serena Dimensions CM suite through the Serena Help server at <http://help.serena.com/alm/dvm/index.html>.

---

## Chapter 1

---

# Introduction to Serena Development Manager

Welcome to Serena ALM	8
Key Benefits of Development Manager	8
Using the Development Manager Components	9
Detailed Example Use Case	12

## Welcome to Serena ALM

Serena Development Manager is a key component of the Serena Application Lifecycle Management (ALM) product line, providing software version and issue traceability from requirements definition through to deployment. With Serena ALM, you can rest assured that all changes to your critical business and commercial systems are managed and visible.

Serena ALM orchestrates your change management process through the powerful lifecycle engine in Serena Business Manager (SBM). SBM centralizes all of your related software lifecycle workflows, orchestrating events across diverse systems using Web services. You can define and customize your requirements management, application development, and software deployment lifecycles. Customize the stages, approvals, and events based on your own organizational needs and best practices.

## Key Benefits of Development Manager

Serena Development Manager enables you to orchestrate and monitor your key software development efforts, tracking source code changes and approvals through a central workflow engine. Development Manager uses Serena Business Manager (SBM) to coordinate events across your systems using Web services, integrating application project definition, source code management, test management, and release approvals. Track and report on development progress using the included dashboard solution, providing comprehensive decision-making support to managers and directors who need the latest information at all times on key performance indicators (KPIs).

Serena Development Manager helps you to conform with CMMI standards by enabling you to establish and maintain an organizational policy for planning and performing your processes and make them visible to relevant members of the organization. It helps you to create organizational expectations for establishing and maintaining baselines, tracking and controlling changes to work products under configuration management, and establishing and maintaining integrity of those baselines.

Serena Development Manager provides the following benefits. You can choose which of these options would be helpful for your organization, and configure and customize the workflow as needed based on your work practices.

- **Development project lifecycle management.** Promote development activities through lifecycles stages such as planning, code, test, and release. You can customize the development stages according to your organizational practices, define user roles and owners for each stage, and set up notifications.
- **Source code management.** Development Manager includes Serena Dimensions CM, the powerful, enterprise-scale source code configuration management application.
- **Test management system integration.** Monitor your QA team's progress on defects and requirements by integrating Development Manager with HP Quality Center / ALM.
- **Graphical business intelligence dashboard.** As an option, Serena provides a rich, graphical reporting tool, Serena<sup>®</sup> Dashboard that allows you to carefully monitor key performance indicators (KPIs). The reports are fully configurable, and can provide graphical information on any aspects of your projects that managers or executives need to track.

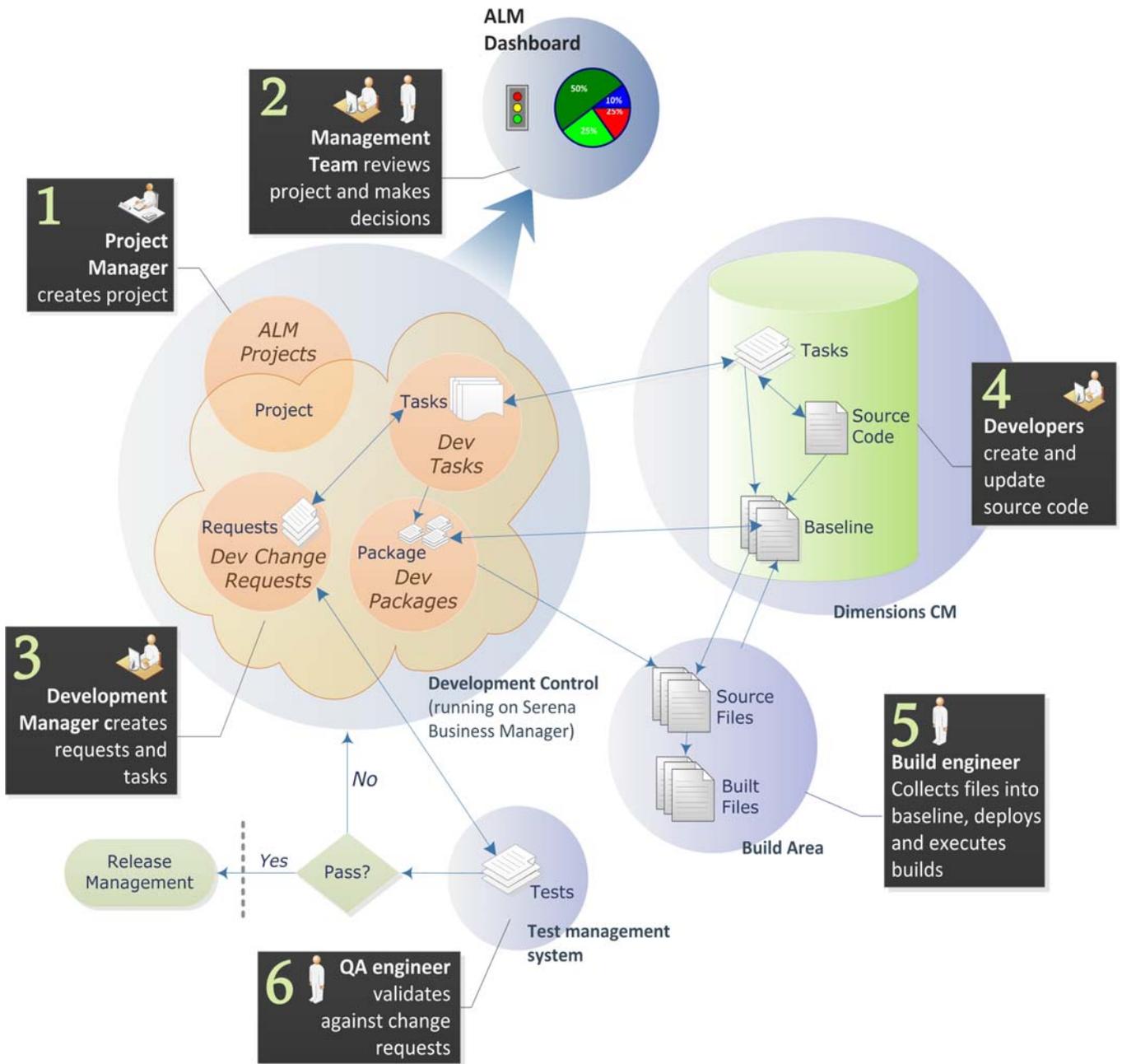
# Using the Development Manager Components

Serena Development Manager includes the following:

- **Development Control**, a collection of process apps implemented on the Serena Business Manager platform. A process app is a Web application, hosted on a Business Manager server, that you can log into and use from any supported Web browser. With the included process apps, Development Control orchestrates your development work across all of your inter-related systems. The included process apps are:
  - **ALM Projects**: Use this app to define and track development projects. This process app sets the workflow for all projects, including states for planning, development, testing, and release. As a project progresses through the workflow, different users work with it using a combination of their own systems.
  - **Dev Change Requests**: Define and track development requests. You can relate requests tasks in the Development Tasks process app, which you can then relate to source code assets in Dimensions CM. You can also relate specific types of requests, such as defects, to test cases in your test management system.
  - **Dev Tasks**: Define and track development tasks. Manage tasks by relating them to change requests. Track source code changes by relating tasks to source files in Dimensions CM.
  - **Dev Packages**: Define and manage development packages. Integrate with your SCM tool to compile and build source code, and integrate with your test management system to test builds before preparing them for release.
- **Serena Dimensions CM**, a rich enterprise-class version and source code configuration management solution. Dimensions CM provides a wide variety of clients and usage models, with Web and desktop clients, as well as complete scripting and API support. Development Control integrates with Dimensions CM via Web services, enabling you to associate source code assets in the Dimensions CM repository with your tasks in Development Control.
- **Serena Requirement Control (optional)**, a collection of process apps implemented on the Serena Business Manager platform that interfaces with Serena Dimensions RM. This enables you to manage and approve requirements. The additional process app it provides is:
  - **Req Approval**: This is used in Development Manager to associate requirements with dev change requests.
- **Serena Dashboard (optional)**, a powerful, highly configurable reporting solution built on the IBI WebFocus platform. With Serena Dashboard, you can build dashboards of graphs that provide precisely the information you, your managers, and your executives need, when they need it.

## Usage Overview

The components included with Serena Development Manager work together as defined in the following diagram. This describes a standard configuration, as well as the order in which users interact with it. This only focuses on the end user scenario; configuration is addressed in the *Serena Development Manager Installation and Configuration Guide*.



Lets step through this diagram.

Step	Description
1	<p>At the beginning of a project, a project manager uses the ALM Projects process app in Development Control (running on Serena Business Manager) to create a new project. This is typically in response to incoming demand, such as requirements from a business analyst for a new feature, or a customer request or defect. For details on the Development Project workflow, please see <a href="#">"ALM Projects Workflow" on page 19</a>. For details on creating projects, please see <a href="#">"Creating and Managing Projects" on page 30</a>.</p> <p>If you are using the additional separately licenced Serena Requirements Manager, the requirements that drive the development process will be managed, approved and assigned using this application. For details see <a href="#">"Using the Requirements Manager Integration" on page 53</a>.</p>
2	<p>At any point during a project's lifecycle, leads, managers, executives, and others may consult Serena Dashboard to review project status and key performance indicators (KPIs). The reports displayed here may help decision makers choose the correct path forward when work must be prioritized or re-evaluated. For more on viewing and customizing Serena Dashboard views, see <a href="#">"Displaying Project Metrics with the Serena® Dashboard" on page 48</a>.</p>
3	<p>The Development Manager creates change requests using the Dev Change Requests process app - or from the project in ALM Projects. The change requests describe the features and other work to be implemented. The change requests are related back to the project, ensuring complete traceability of work. The Development Manager and others also create tasks using the Dev Tasks process app. Tasks can be used to split the work into more manageable units, that can be assigned to individual developers. When you create a development task, the task is synchronized to Dimensions CM, and a new request of type Task is created.</p> <p>To learn about the detailed workflow for change requests and tasks, please see <a href="#">"Dev Change Requests Workflow" on page 20</a> and <a href="#">"Dev Tasks Workflow" on page 21</a>. For information on creating change requests, please see <a href="#">"Creating and Working on Change Requests" on page 36</a>. For information on working with tasks, please see <a href="#">"Creating and Working on Tasks" on page 40</a>.</p>
4	<p>Developers update the source code using their source control environment. Serena Development Manager includes Serena Dimensions CM. Tasks in Development Control are synchronized to Dimensions CM, and information on all work on files in Dimensions CM is stored in Dimensions CM tasks. This information is then synchronized back to the originating tasks (which are in turn related back to the originating change requests) in Development Control, ensuring a complete audit path of all work completed in context of a project.</p>

Step	Description
5	As work progresses on the project, the build engineer sets up packages using the Dev Packages process app in Development Control. The build engineer relates the packages to baselines in Dimensions CM that collect all of the files associated with change requests in the project and deploy them to a build area. Using these files, the build engineer runs a build and installs it for testing purposes. To learn more about the workflow for the Development Baselines process app, see <a href="#">"Dev Packages Workflow" on page 23</a> . For details on creating and working with development packages, see <a href="#">"Creating and Working with Development Packages" on page 42</a> .
6	Using HP Quality Center / ALM, the QA staff tests the builds, both the nightly builds and release candidate builds. Defects may be tracked in the Dev Change Requests process app, and related to defects in Quality Center. Failed tests are returned to Development Control and the original change requests are returned to developers to fix. When a release candidate build passes testing, the build is turned over to the release engineer who will use a release management solution, such as Serena Release Manager, to deploy the build into all of the required environments. For details on implementing the SBM Connector to Quality Center, see the <i>SBM Connector for HP Quality Center / ALM Implementation Guide</i> .

## Detailed Example Use Case

We will walk through a scenario that demonstrates what Development Manager can do for you. Keep in mind that this is just an example, and that these steps can all be customized to satisfy the goals of your organization. Customization is one of the key benefits of Serena solutions built on Business Manager; you can adapt the workflows, forms, integrations, and reports to meet your business needs.

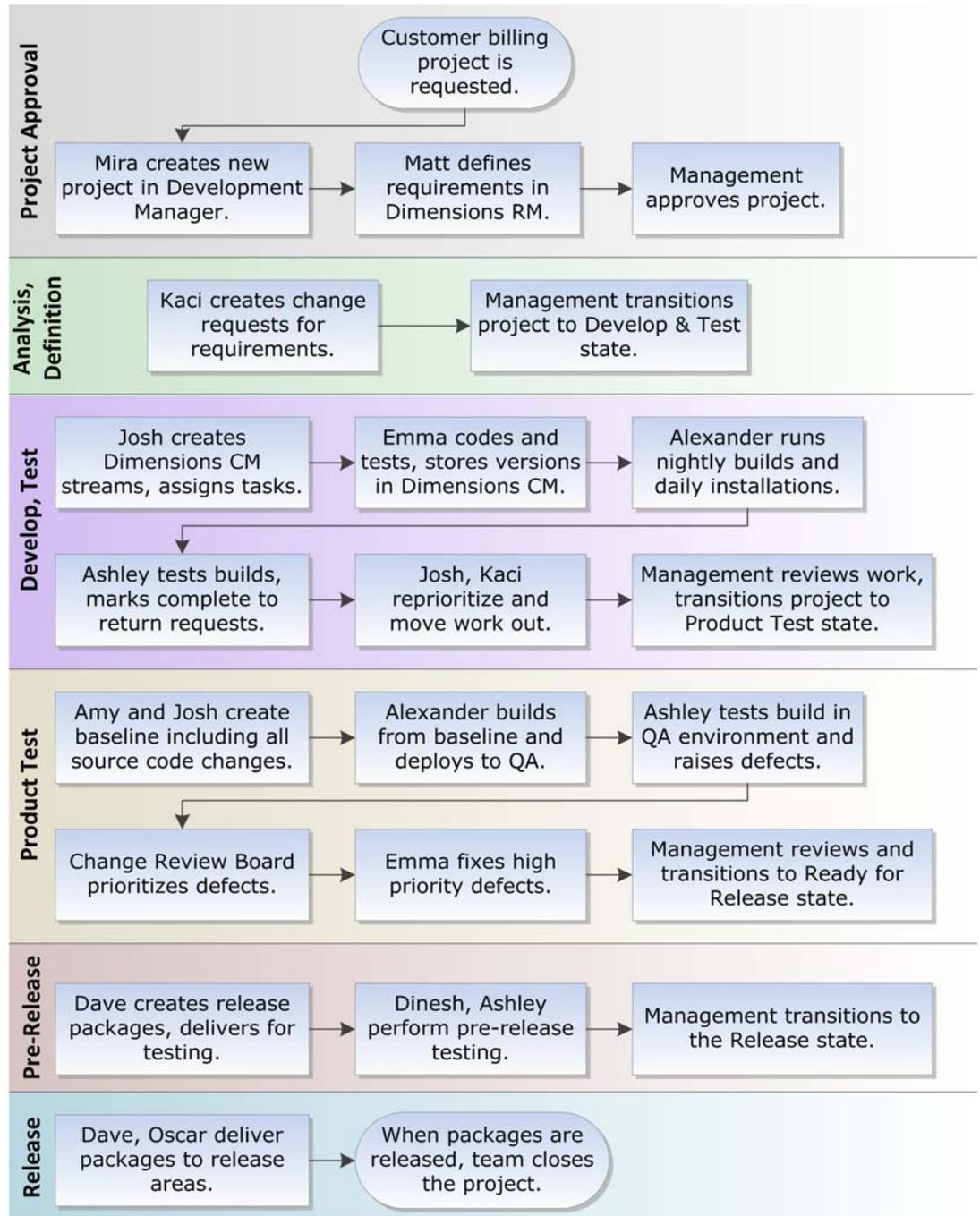
### User Roles

Serena Development Manager is designed to meet the needs of key participants in a standard development effort. Roles in your organization may vary. This example scenario includes the following users:

- Mira: Project Manager
- Matt: Business Analyst
- Kaci: Product Owner
- Josh: Development Manager
- Emma: Development Engineer (member of Development Team)
- Amy: Build Manager (member of Build Team)
- Alexander: Build Engineer (member of Build Team)
- Dinesh: QA Manager (member of QA Team)
- Ashley: QA Engineer (member of QA Team)
- Dave: Release Manager (member of Release Team)
- Oscar: Release Engineer (member of Release Team)

## Process Example

The following flowchart presents an example scenario using the personas described above. Details about each step follow the diagram.



Project Phase	Description	Tool
<b>Project Approval</b>	Matt ( <i>Business Analyst</i> ) asks Mira ( <i>Project Manager</i> ) to create a new project for the next release of the customer billing system. Mira creates the project in the ALM Projects process app in SBM. This project will be used to coordinate development activities.	ALM Projects
	Matt is notified that the project has been created, and uses his requirements management tool to define detailed requirements. He adds requirement links to the project.	External requirements management tool
	The management team (Mira, Matt, Kaci, Josh, Dinesh, Amy, and Dave) reviews the requirements and works through any open questions. When the requirements are approved, they promote the project to the next phase, <i>Analysis and Definition</i> .	ALM Projects
<b>Analysis and Definition</b>	Kaci ( <i>Product Owner</i> ) reviews the detailed requirements and creates software change requests for each of them. Her goal is to break the requirements into small, clearly defined units of work.	ALM Projects
	The management team reviews the requirements, stories, and testing requirements. When the requirements and change requests are approved, they promote the project to the next state, <i>Develop and Test</i> .	ALM Projects
<b>Develop and Test</b>	Josh ( <i>Development Manager</i> ) reviews the requests and works with Kaci to prioritize the work. Josh also creates the development streams in Dimensions CM where the source code will be managed.	ALM Projects
	Josh, Kaci, Emma ( <i>Software Engineer</i> ), and Ashley ( <i>QA Engineer</i> ) break the requests into development tasks. As they are created, the tasks are synchronized to Dimensions CM requests. Once the requests and tasks are ready to work on, Josh moves them into development. Ashley and Emma can now begin work.	ALM Projects
	Emma develops and tests code changes for each of her tasks. She works with the tasks in Dimensions CM via the Dimensions CM integration to Microsoft Visual Studio. She stores new versions of source files in Dimensions CM. Meanwhile, Ashley defines test cases that can be used to test each story as it is completed.	Dimensions CM
	Alexander (Build Engineer) manages nightly builds and daily installation on the QA test environments.	External build tool
	When she completes all of the tasks for a change request, Emma assigns that request to Ashley for testing.	Dimensions CM
	Ashley runs the test cases for the request that Emma finished coding. Ashley uses the installation that Alexander runs from the nightly build. Ashley returns any request that fails testing back to Emma to fix.	HP Quality Center / ALM
	Once she has finished testing a request, Ashley marks it as complete. This delegates the request back to Josh.	HP Quality Center / ALM
	Josh and Kaci continually review the list of requests in progress, and move work out of the project as needed based on remaining time and business priorities.	ALM Projects
	Once the requests have all been developed and tested, the management team reviews the requirements, requests, and test results. When they approve the completed work, they move the project to the next phase, <i>Product Test</i> .	ALM Projects

Project Phase	Description	Tool
<b>Product Test</b>	Amy and Josh create a package in the Dev Packages process app, and then relate that package to a baseline in Dimensions CM. The baseline assembles all of the source code changes associated with completed requests. Amy delegates the package to Alexander who then builds the source associated with the Dimensions CM baseline. He collects the build outputs back into Dimensions CM, and deploys them to the QA testing environments.	ALM Dev Packages, Dimensions CM
	Josh assigns the turnover package to Dinesh ( <i>QA Manager</i> ) to coordinate testing.	ALM Projects
	Dinesh prioritizes testing requirements for the project.	HP Quality Center / ALM
	Ashley runs test cases against the QA environments and raises defects as she finds issues.	HP Quality Center / ALM
	The change review board determines which defects need to be fixed. Defects that need to be fixed are delegated to Emma. Deferred defects are moved out of the project to be fixed at a later date.	ALM Projects
	As Emma fixes defects, new packages that include the fixes are forwarded to QA. Ashley re-tests the defects.	HP Quality Center / ALM
	Once all of the requirements and defects have been tested, the management team reviews the requirements, stories, package(s), and test results. Once they approve the completed work, they promote the project to the next phase, <i>Ready for Release</i> .	ALM Projects
<b>Ready for Release</b>	Using Serena Release Manager, Dave ( <i>Release Manager</i> ) integrates the packages into the release cycle. He works with Oscar ( <i>Release Engineer</i> ) to get the package delivered to the appropriate pre-release areas for testing.	Dev Packages
	Dinesh coordinates the pre-release testing and Ashley raises defects as necessary.	HP Quality Center / ALM
	Once the packages have been successfully tested, the management team does a final review of the requirements, completed requests, packages, and test results. Once they approve the release, they promote the project to the next phase, <i>Release</i> .	ALM Projects
<b>Released (inactive)</b>	Using Serena Release Manager, Dave works with Oscar to deliver the package to the release team.	Release Control
	Once the packages have been successfully released and the management team approves the final release, they close the project.	ALM Projects



---

## Chapter 2

---

# Understanding the Out-of-the-Box Workflows

About the Out-of-the-Box Development Workflows	18
Object Types and Relationships	18
ALM Projects Workflow	19
Dev Change Requests Workflow	20
Dev Tasks Workflow	21
Dev Packages Workflow	23
Supporting Your Development Process	25

## About the Out-of-the-Box Development Workflows

Serena Development Manager includes rich out-of-the-box workflows as part of Development Control, a collection of process apps that run on Serena Business Manager. Development Control integrates development processes with versioning capabilities provided by Serena Dimensions CM, and a set of out-of-the-box reports provided by Serena Dashboard. Development Control satisfies use cases specific to the development states of a project, including analysis and design, development, testing, and release preparation.

Out-of-the-box, Development Control provides workflows for the following:

- Development projects, implemented by the ALM Projects process app.
- Development change requests, implemented by the Dev Change Requests process app.
- Development tasks, implemented by the Dev Tasks process app.
- Development baselines and packages, implemented by the Dev Packages process app.

## Object Types and Relationships

The SBM process apps included with Development Control enable you to manage projects, change requests, tasks, and release packages. Each of these object types can be related to specific items in external systems.

- **Projects:** You manage projects using the ALM Projects process app. You can relate projects as follows:
  - By relating a project to Dimensions CM streams and projects, you enable the association of source code and other managed assets related to the project.
  - By relating a project in ALM Projects to change requests in the Dev Change Requests process app, you ensure that all work to be completed in context of the project is associated with the project. You also provide scope for the creation of Dimensions CM tasks and items that are needed for this project.
  - By relating a project to packages in the Dev Packages process app, you ensure that all completed work included in releases are associated with the project.
- **Change Requests:** You manage change requests using the Dev Change Requests process app. You can relate change requests as follows:
  - Relate change requests to tasks in the Dev Tasks process app. You can then manage all development work from the related tasks.
  - If you use HP Quality Center / ALM to manage defects, associate requirements and defects to defect type change requests in Development Control.
  - If you use Serena Requirements Manager, associate requirements to dev change requests. You can associate multiple requirements to a single dev change request or multiple dev change requests to a single requirement.

- **Tasks:** You manage tasks using the Dev Tasks process app. You can relate tasks as follows:
  - Relate tasks to change requests in the Dev Change Requests process app to maintain a history of which tasks were associated with the requests.
  - Relate tasks to requests in Dimensions CM in order to track specific code changes. All updates to files stored in Dimensions CM are tracked in Dimensions CM tasks; this information is then synchronized to tasks (and their associated requests) in the Dev Tasks process app.
- **Development packages:** You manage development packages using the Dev Packages process app. Packages define the scope of a release by collecting all included change requests and tasks into one place. You can relate packages as follows:
  - Relate packages to baselines in Dimensions CM. Dimensions CM baselines collect all source code (or compiled deliverables) associated with a particular project or release. Those baselines can be deployed, using Dimensions deployment features or using Serena Release Manager.
  - Packages are also related to specific change requests. The change requests define the scope of the code included in the baseline.

## ALM Projects Workflow

ALM Projects provides a high level view of the overall project workflow and status.

### ALM Projects Workflow



#### The states include the following:

- 1 **Inception:** During this state, the Project Manager creates the project.
- 2 **Elaboration:** During this state, the Business Analyst defines change requests for the project based on the approved requirements.
- 3 **Construction:** At this point, the Development Manager or Lead prioritizes the change requests and creates tasks related to the requests. The tasks are assigned to developers to implement. Developers work on the tasks, storing new versions of files in their configuration management system (such as Dimensions CM). Regular builds are compiled and installed for testing, and requests are assigned to QA to test. Once all of the requests have been implemented, the Development Manager can transition the project to the next state.
- 4 **Transition:** During this state, QA perform robust testing of the completed product or features, recording defects as they find them. QA may use a test management system such as HP Quality Center to track test case execution. QA submits defects to the Change Request process app, and the defects are assigned to developers to fix. QA validates fixes and closes defects as they are resolved. When all critical defects are complete, QA can transition the project into the Complete state.

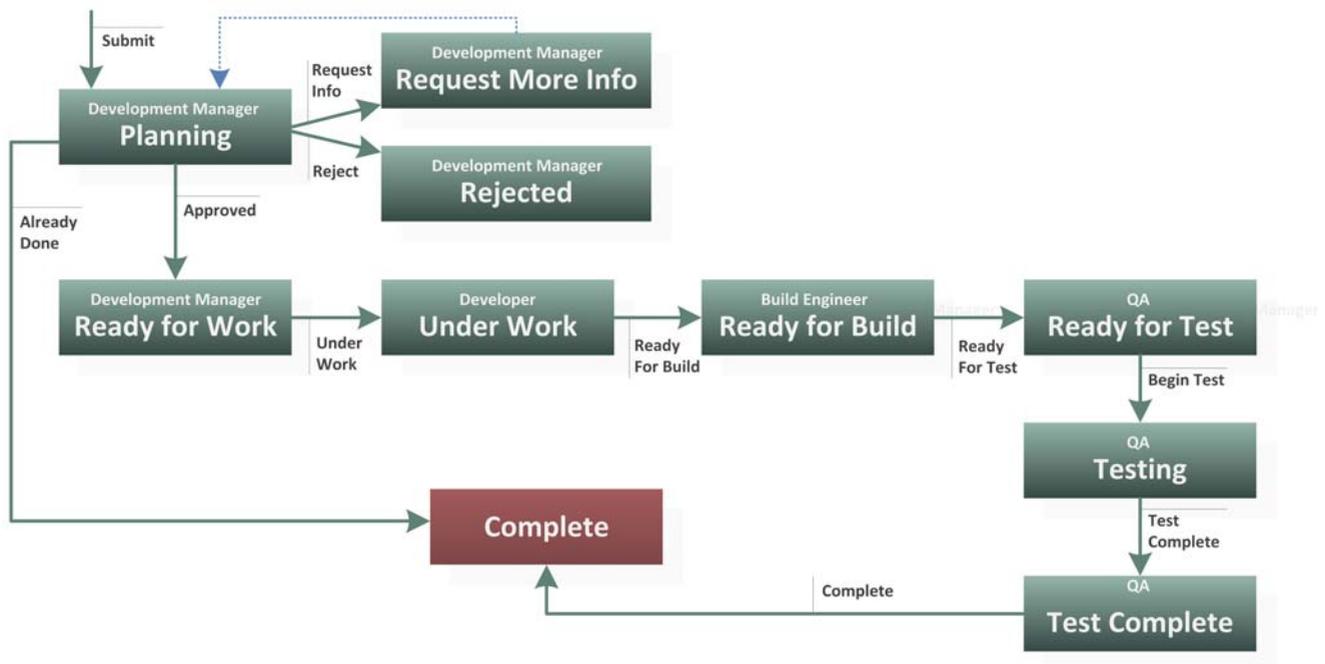
- 5 **Complete:** During this state, the finished product is prepared for release. The final builds are collected and packaged for deployment, perhaps using Serena Release Manager. When the product is ready to deploy or release, the project can move to the final state.

## Dev Change Requests Workflow

Change requests follow the workflow illustrated below. This workflow is implemented in the Dev Change Requests process app. This illustration is exported directly from SBM. Note that most work on change requests is expect to happen during the **Develop & Test** state in a project.

### SBM Workflow for Change Requests

Development Change Requests Workflow



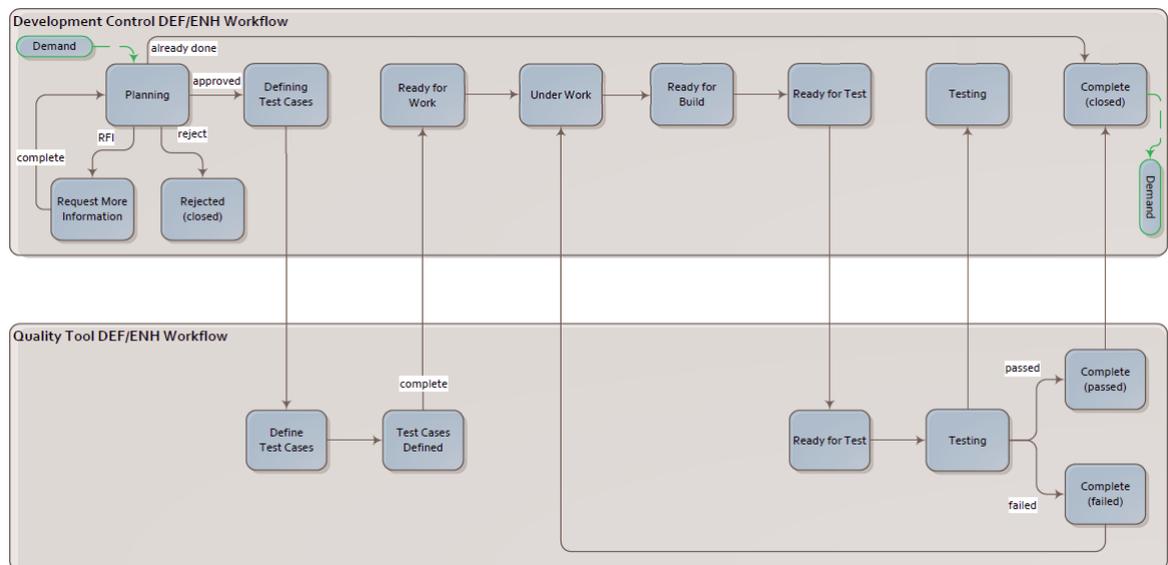
- 1 **Planning:** During this state, the Product Owner and Development Manager define the content of change requests based on the requirements for the project. If more information is needed, the request can be moved to the **Request More Info** state and a request for more information sent directly from SBM. Once the request is approved, it can be moved into the **Ready for Work** state.
- 2 **Under Work:** Developers can now work on tasks associated with the request. As work is complete the developer checks updated source code files into Dimensions CM. Once this work associated with the request (and its related tasks) is ready to be compiled and tested, the request is transitioned by the developer to the next state.
- 3 **Ready for Build:** At this state, the build engineer starts a build using the code that was created or updated during the **Under Work** state. When the build successfully

completes and the request is ready to test, the build engineer transitions the request to the next state.

- 4 **Ready for Test:** The QA staff can now test the code that was developed and compiled to satisfy this change request. The QA manager or responsible QA engineer transitions the request to the **Testing** state. Testing is then managed in HP Quality Center. If a test fails in the test case management system, the request is returned to the **Under Work** state. When testing is complete, the request can be transitioned to the **Test Complete** state, and then to the **Complete** state.

## Integrating Change Requests with HP Quality Center / ALM

The following diagram illustrates the out-of-the-box integration points between Dev Change Requests and HP Quality Center / ALM.

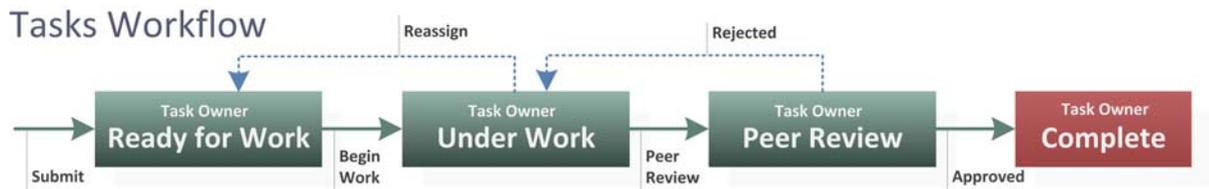


The test states in the change request workflow are mapped to workflow states in the Quality Center / ALM workflow. When the change request enters the *Ready for Test* state, it is associated with test cases in Quality Center.

## Dev Tasks Workflow

Manage specific developer tasks using the Tasks process app. You start and complete work on a task during the **Under Work** state of the change request that owns it. Development Managers create tasks to break work associated with a change request down into more manageable units that can be assigned to individual developers.

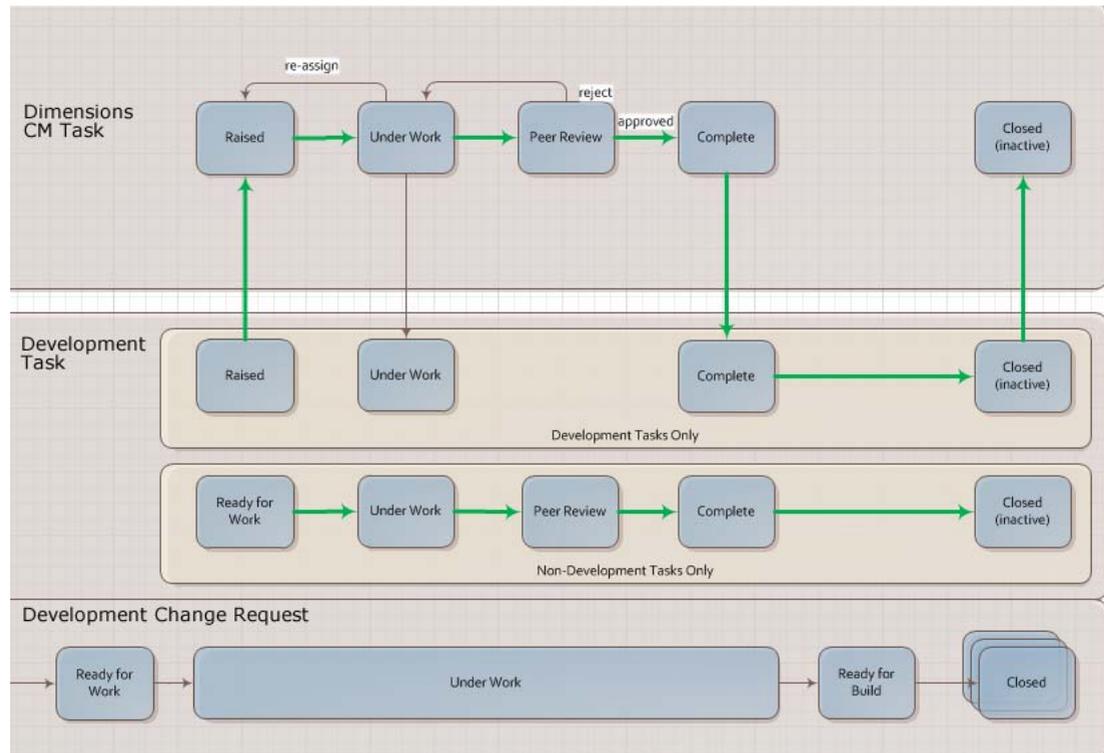
Tasks follow the workflow illustrated below. This illustration is exported directly from SBM. If you are using Dimensions CM, these states map to parallel states for change requests in Dimensions CM.



The states include the following:

- 1 Raised:** The task is submitted. If you are using Dimensions CM, a related change request is also created in Dimensions CM. To start work, the Development Manager assigns an owner and transitions to the **Under Work** state. The related item in Dimensions CM is also transitioned to the parallel **Under Work** state in Dimensions CM.
- 2 Under Work:** As work is complete the developer checks updated source code files into Dimensions CM. Information about the new and updated files is stored in the requests in Dimensions CM, and that data is synchronized back to the task in Dev Tasks. All information about affected files is therefore captured in context of the change request that the task belongs to.
- 3 Peer Review:** A software development best practice is to include review of changes by another developer before turning the new or updated code over for testing. Once the peer has approved the changes, the task can be transitioned to the **Complete** state. If the task is related to a request in Dimensions CM, the Dimensions CM item is also transitioned to the **Complete** state.

The following illustration clarifies the relationship between tasks in Dev Tasks and related tasks in Dimensions CM. Note that this is just one example of how tasks in Development Control might be linked to tasks in Dimensions CM for purposes of tracking work.



In this example, the associated task in Dimensions CM follows a parallel workflow. The request may be a specific type that is designed to synchronize with tasks in the Dev Tasks process app. In this example, the task in Dev Tasks is transitioned to the completed state once the related task in Dimensions CM is completed, requiring all work completed in Dimensions CM to be peer reviewed and approved before the task can be closed. Transitioning the request in Dimensions CM to the **Under Work** and **Complete** states in turn automatically transitions the related task in Dev Tasks.

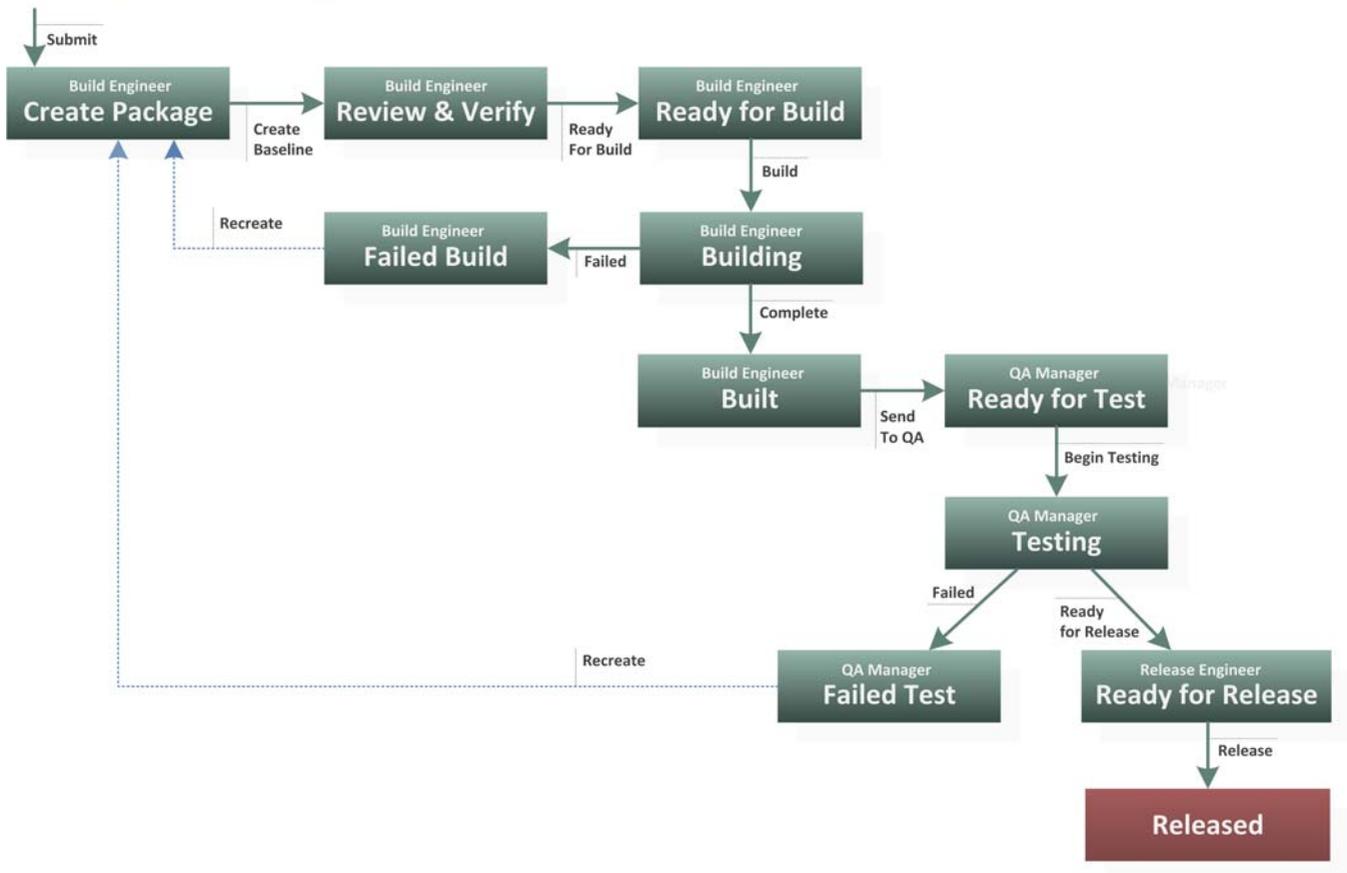
## Dev Packages Workflow

Use the Dev Packages process app to manage the creation and validation of release packages and baselines, including testing builds and approving for final release. Most of the work managed within the Dev Packages process app takes place during the **Product Test** project state.

The Dev Packages workflow provides integration points into Dimensions CM, as well as Quality Center. Use Dev Packages to fully automate the build and validation process. Start by building from baselined sources and finish by passing the final build off to your release management system, such as Serena Release Manager.

## Dev Packages Workflow Diagram

### Development Packages Workflow



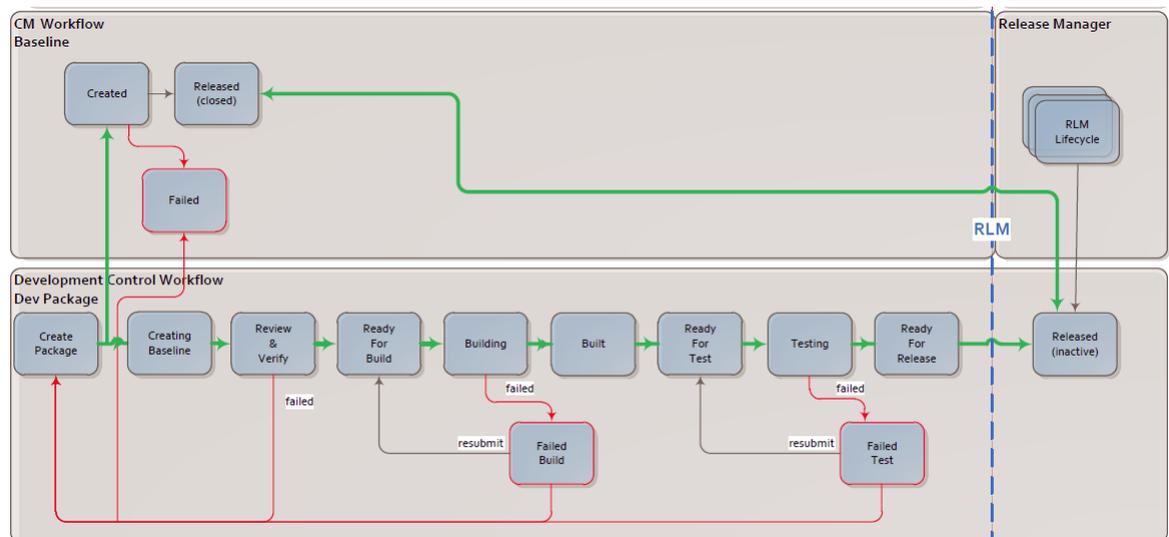
The workflow states include the following:

- 1 **Create Package:** The release engineer creates a package that defines the scope of a release. The package includes links to change requests that define the content of the package.
- 2 **Creating Baseline and Review and Verify:** A baseline is created in Dimensions CM that includes all of the files that are to be included in the release. The baseline is related to the package in Dev Packages, and the baseline follows a parallel workflow. Once the baseline and package are approved for testing, the package is transitioned to the next state.
- 3 **Ready for Build and Building:** The build engineer compiles the build from the package that the release engineer turned over. If the build fails, the release engineer reviews and updates the package as needed. The baseline may need to be recreated and the build run again. If the build succeeds, the build engineer transitions the package to the **Built** state, and then deploys the compiled deliverables for QA to test.
- 4 **Ready for Test and Testing:** QA staff test the deployed build. Information about the packages is synchronized to requirements in the test management tool. Test cases for validating those requirements are associated with the requirements. QA run their tests and update with pass / fail information, which is then synchronized back to the packages in Dev Packages. If a test fails, then either the build, baseline, or source code will need to be updated to correct the defect. The team must review the

results and decide who will fix any issues that the testing reveals. If the source code is updated, the release engineer must re-generate the baseline, and the build engineer must re-run the build and deploy it. When testing is complete and the build is approved for release, QA transitions the package to the **Ready for Release** state.

- 5 Ready for Release:** At this state, the release management process takes over. For example, you may use Serena Release Manager to coordinate deployment.

The following diagram illustrates the integration points between the Dev Packages process app and Dimensions CM. This is based on the out-of-the-box configuration. In this example, a baseline in Dimensions CM follows a separate workflow. As the package progresses through its workflow states, information and state are synchronized to the related baseline in Dimensions CM.



## Supporting Your Development Process

Serena Development Manager is a powerful, highly customizable lifecycle management and reporting application. To make the best use of it, spend some time evaluating your own development processes. It is important to decide what processes and issues Serena Development Manager can help you solve, to set expectations appropriately and begin to plan roll-out and adoption. Consider creating a diagram of your development lifecycle if you do not already have one, as this can be an valuable resource when mapping your workflows in Serena Business Manager.

### What Types of Development Projects Will You Manage?

In Serena Development Manager, you can define project types that correspond to specific development management needs in your organization. For example, you can define different project types for different applications, or for different departments. Consider the following when deciding what types of project you will support:

- Will Serena Development Manager support projects for different departments? Do the departments have their own lifecycles?

- What applications does your organization use, and which of these will you integrate with Serena Development Manager? For example, does your development team use Serena Dimensions CM for source code management, or another system? What about Requirements and Test Case Management?

### ***Best Practices for Development Project Management***

- Use the Dev Change Request process app to manage defect reports alongside other change requests. If you also use HP Quality Center for defect management, you can synchronize your defects between the two systems.

## **Is Your Development Process Agile?**

Determine what type of development practice your project types follow. Do they follow iterative Agile methodologies? Is their process closer to traditional "waterfall" methodologies, with heavier upfront analysis and planning?

### ***Best Practices for Agile Projects:***

- Capture user stories with change requests using the Dev Change Requests process app. You can simply word change requests as you would word a user story.

## **What Are Your Workflow States?**

Identify the states in your development workflow. You can replicate your existing states, or refine your current practices when configuring the lifecycle in Serena Business Manager. Questions you might hear include:

- Who submits new requests for different types of work (such as features, defects, etc.) to your development team, and in what form?
- What is involved with design and analysis of new features, before developers actually begin work? Who is responsible?
- How is testing managed? Is it part of the development effort, completed afterwards, or both? Who manages the test states?
- What steps are required in order to certify new work for release? Consider such things as stabilization, defect burndown sprints, and deploying to staging. Who manages these steps?

### ***Best Practices for Defining Your Workflow***

- You can customize the workflows in Serena Development Manager as needed. Study and work with the out-of-the-box workflows, itemize the changes that you need to make, and update them using Serena Business Manager Composer.
- Integrate testing into your development lifecycle. Out-of-the-box, the Serena Development Manager workflow includes a separate state for testing that occurs after development is complete. However, we strongly advocate for testing in context of development. Developers should test their own work, as well as peer review each other's work.

## How Do You Manage Test Cases?

Development Manager provides a Web Services integration to HP Quality Center, enabling you to synchronize information from your development requests in Business Manager to defects in Quality Center. Please see the *Serena Business Manager Connector for Quality Center Implementation Guide* for implementation details.

## What Reports Does Your Organization Need?

Serena provides as an option, Serena Dashboard, a powerful dashboard reporting solution that allows you to display rich graphical data on key performance indicators (KPIs). This enables your stakeholders, including team members, managers, and executives, to quickly review project status and make critical project decisions.

However you can define additional reports as needed. Consider what reports you need to define. Who needs to review project information? Which systems are involved? For more information on the out-of-the-box metrics included with Serena Dashboard, see ["Displaying Project Metrics with the Serena® Dashboard" on page 48](#). To learn more about configuration Dashboard views, please see the *Serena Development Manager Installation and Configuration Guide*.



---

## Chapter 3

---

# Getting Started Using Development Manager

Introduction	30
Logging In to Development Control	30
Creating and Managing Projects	30
Creating and Working on Change Requests	36
Creating and Working with Development Packages	42
Creating and Working with Development Packages	42
Displaying Project Metrics with the Serena® Dashboard	48
Getting to Know Dimensions CM	51
Integrating with HP Quality Center	52

## Introduction

After installing the Development Manager components, you can log in right away to the out-of-the-box configurations in Development Control and Serena Dashboard. You can then start learning your way around the default process apps. A process app is an application running in Serena Business Manager with a Web-based user interface and workflow logic that coordinates the features and integrations across all included platforms. Process apps can be customized as needed to meet the needs of your organization, however right out-of-the-box you can step through the default process apps to see a mature example of a Development Manager implementation.

## Logging In to Development Control

To work with the Development Control process apps, log in to Serena Business Manager using one of the default user accounts provided as part of your Development Manager Installation.

### To log into Development Control:

- 1 In a supported browser application, open the URL to your Serena Business Mashups server. By default, this is:

```
http://server:port/tmtrack/tmtrack.dll?shell=dvm
```

For example, if the server is called dev and the port number is 8085, and you are using the Serena-supplied custom shell:

```
http://dev:8085/tmtrack/tmtrack.dll?shell=dvm
```

Don't forget the question mark at the end of the .dll.

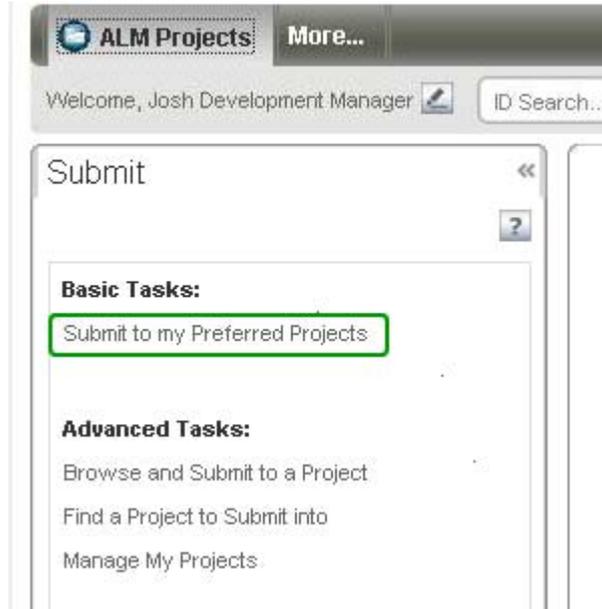
- 2 Enter your login credentials.

## Creating and Managing Projects

Use the ALM Projects process app to create and manage your development projects. As new requests and requirements come in, create projects to manage development work. Once you create a development project, you can associate change requests and packages with it.

## Creating a Project

Click **Submit to my Preferred Projects** from the **Submit** navigation area on the left to create a new development project.



Note that in the default process application you will need to be logged in as Mira, the Project Manager, to submit a project.

A Submit form appears:

**Submit into:** ALM Projects

OK Cancel

\* Project Name:  Project Status: (None) Project Type: (None)

Description:

**Roles**

\* Project Manager: (None) \* Product Owner: (None) QA Manager: (None) Development Manager: (None)

Build Manager: (None) Business Analyst: (None) Release Manager: (None)

When you create a new project, you set a number of core properties for the project, including:

- The type of project from the **Item Type** list. Out-of-the-box, you can choose **Innovation** or **Operational**.
- The title and description.

- Key users involved with the project, such as the Project Manager, Product Owner, Business Analyst, and Development Manager.
- Milestone dates for the project. This includes target start and end dates for each stage in the project, including Inception, Elaboration, Construction, and Transition. You can set actual start and end dates later on as the actual dates are passed

**Dates**

**Project Phase**

	Target Start Date:	Target End Date:
<b>Inception:</b>	<input type="text"/> mm/dd/yyyy	<input type="text"/> mm/dd/yyyy
<b>Elaboration:</b>	<input type="text"/> mm/dd/yyyy	<input type="text"/> mm/dd/yyyy
<b>Construction:</b>	<input type="text"/> mm/dd/yyyy	<input type="text"/> mm/dd/yyyy
<b>Transition:</b>	<input type="text"/> mm/dd/yyyy	<input type="text"/> mm/dd/yyyy

- Dimensions CM products and projects or streams to associate with the project. Under **CM Settings**, click **Get CM Products** or **Get CM Projects/Streams** to list available products and projects / streams.

Note that in the default process application you may need to log in as Josh, the Development Manager, after the ALM Project has been created to associate it with a

Dimensions CM products product and project/stream

**Configuration Management Settings**

**1. CM Product:**

Product

Get CM Products

**2. CM Project or Stream:**

Project or Stream

Get Projects or Streams

Submit into: ALM Projects

OK Cancel Actions: [dropdown] [grid] [help]

\* Project Name: R1.0.1 Enhancements Project Status: Project  
(None) (None)

Description: Updates for Dietary Features & GUI

**Roles**

* Project Manager: Mira Project Manager	* Product Owner: Kaci Product Owner	QA Manager: Dinesh QA Manager	Development Manager: Josh Development Manager
Build Manager: Amy Build Manager	Business Analyst: (None)	Release Manager: (None)	

**Dates**

**Project Phase**

	Target Start Date:	Target End Date:
Inception:	[mm/dd/yyyy]	[mm/dd/yyyy]
Elaboration:	[mm/dd/yyyy]	[mm/dd/yyyy]

## Example Project in ALM Projects

Here we see an example project in ALM Projects. This has three dev change requests associated with it.

ALM Projects ALM Projects - 000195: 1704prj1 (Inception)

Project: **1704prj1** [Edit](#) State: Inception

Description:

Project Manager: Administrator

Project Status: (None)

Item Id: 000195

**Change Requests** Packages Tasks Schedule History Log CM

Create Dev CR

Item Id	Title	State	Severity	Business Priority	Target Due Date	List of Child Tasks
<a href="#">000207</a>	4cm1	Planning	(None)	(None)		
<a href="#">CHG000213</a>	3req	Planning	(None)	(None)		
<a href="#">CHG000212</a>	2req	Planning	(None)	(None)		
<a href="#">000208</a>	cr6	Planning	(None)	(None)		DEV000209: 4cmtask112 (Under Work)
<a href="#">000204</a>	cr4cm	Planning	(None)	(None)		DEV000206: task2 (Raised)
<a href="#">CHG000198</a>	3req	Planning	(None)	(None)		
<a href="#">000197</a>	cr2	Planning	(None)	(None)		
<a href="#">000196</a>	0704cr1	Planning	(None)	(None)		

You can transition a project to the next state in the lifecycle by pressing the appropriate transition button, such as **Begin Elaboration** or **Begin Construction**. Note that you need the required role, for example Project Manager, to transition the project, otherwise the required button will not be displayed.

ALM Projects ALM Projects - 000195: R1.0.1 Enhancements (Inception)

Actions:

Project: **R1.0.1 Enhancements** [Edit](#) State: Inception

Description: Updates for Dietary Features & GUI

Project Manager: Mira Project Manager

Project Status: (None)

Item Id: 000195

**Begin Elaboration**

You can update details for the project by clicking the **Edit** link.

You can display transition history for a project or other Development Manager item by selecting its **History Log** tab.



Below the state change history you can view the history of activities by expanding the Activity Log.

You can expand an entry to reveal more details by clicking the + icon.

Changed Value	Prior Value	New Value
Dev Packages	DVPK000201: 1704pack1 (Review & Verify), DVPK000203: pack2 (Review & Verify)	DVPK000201: 1704pack1 (Review & Verify), DVPK000203: pack2 (Review & Verify), DVPK000204: 4cnpack1 (Built)
Last Modified Date	*****	*****

To view the workflow diagram of any item, click the Display Workflow button. This will display a graphic of the workflow for that type of item.



## Displaying and Creating Related Change Requests

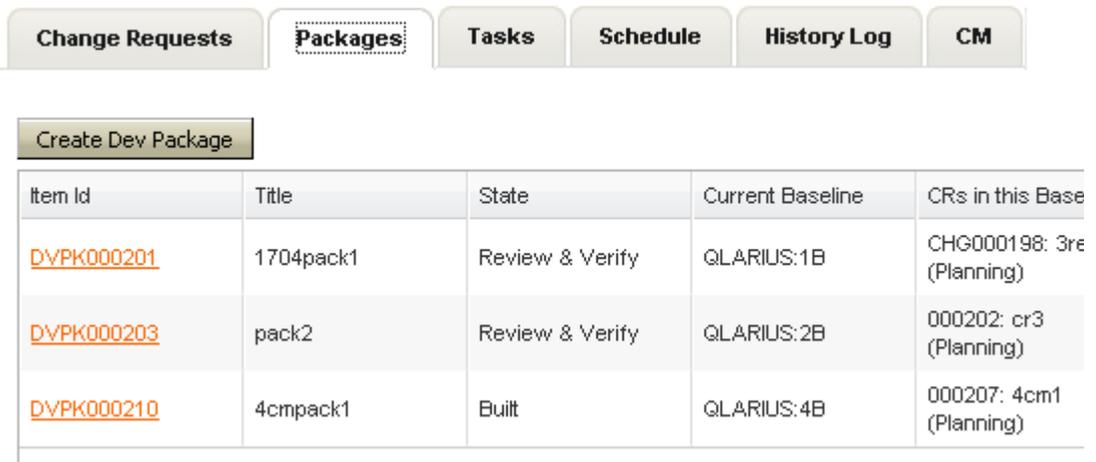
Under **Change Requests**, you can list and display details on the change requests associated with the project, as well as create new change requests. To create new change requests, click the **Create Dev CR** button.

You can also create dev change requests from the Dev Change Requests process app, but in this case you will need to select the ALM project to which it belongs when you create it.

See ["Creating New Change Requests" on page 37](#) for details on creating change requests.

## Creating and Relating Development Packages

Under **Dev Packages**, you can list and display details on the development packages associated with the project.



Click the item id of any dev package to display more information about it. Click the **Create Dev Package** button to create a new package.

See ["Creating New Packages" on page 43](#) for details.

## Creating and Working on Change Requests

From the Dev Change Request process app, you can create, update, and report on development change requests. You can also relate change requests to ALM projects in the ALM Projects process app.

## Finding Change Requests

Display the **Search** tool from the left navigation pane. Here you can submit simple keyword searches, or search based on specific criteria. For details on searching in Serena Business Manager, see the Serena Business Manager online help. To display all change requests in the system, select **Search by Keyword** from the **Search** area, leave the keyword field blank, and click **Search**.

The screenshot shows the 'Search by Keyword(s)' interface. At the top is a grey header with the text 'Search by Keyword(s)'. Below this is a 'Keyword(s)' label followed by an empty text input field, which is highlighted with a green border. Underneath the input field are two radio buttons: 'Match All' (selected) and 'Match Any'. Below these is a 'Project(s)' label followed by a dropdown menu showing 'Dev Change Requests Project'. Under the dropdown are two links: 'Show My Projects' and 'Manage My Projects'. Below these links is a checked checkbox labeled 'Search in Sub-projects'. At the bottom left of the form area is the 'Active/Inactive' label with three radio buttons: '(Both)' (selected), 'Active', and 'Inactive'. At the bottom center is a 'Search' button.

You can also find and track specific defects by running and creating reports from the **Reports** view. For more on reporting in SBM, see the SBM online help.

## Creating New Change Requests

You can submit a new change request by clicking the **Submit** area on the left navigation pane, and then clicking one of the submit links. For example, click **Browse and Submit to a Project** under **Advanced Tasks** to browse the project tree and choose the correct project to submit to.



**NOTE** The project in the project tree to which you are submitting in this case is an SBM Project and not an ALM Project. You choose the ALM Project to associate with the dev change request in the **Associate to Project** field when you are creating it. Alternatively, do this by first selecting the ALM Project as described next.

You can also submit change requests from the ALM project in the ALM Projects app. For example, select **Search for an existing ALM Project**, complete the search page and

click the **Search** button. Select the required ALM project, click the **Change Requests** tab, and click the **Create Dev CR** button.

When you submit a new change request, a form appears. Note that users with different privileges may see more fields on the submit form.

**Submit into:** Dev Change Requests Project

OK Cancel

\* **Change Request Name:**  #BC ✓

Change Request Type:  #BC ✓

Target Due Date:  mm/dd/yyyy

\* **Associate to Project:**  Find

Business Priority:  #BC ✓

Severity:  #BC ✓

Customer Found:  #BC ✓

Estimated Effort:  hours

Description:

Acceptance Criteria:

Status Log:

These are comments you may leave for others to see and respond to.

**Roles**

\* **Development Manager:**  #BC ✓

Developer:  #BC ✓

Build Engineer:  #BC ✓

QA Engineer:  #BC ✓

You set a number of core properties for the request. You can set most of these later on; only the fields with red labels are required. You can set options such as:

- From the **Item Type** list, the type of request such as **Defect**.
- The title and description.
- Business priority, severity, and acceptance criteria for resolving the request.
- Specific users in the various user roles, such as the **Development Manager** or **Build Engineer**.

## Working on Change Requests

Open a change request to update and work on it. A change request with some associated tasks might look something like this.

Back to Results

**Dev Change Requests Project Dev Change Requests - CHG000199:** UnderWriter Smart System (Planning)



**Change Request**  
**UnderWriter Smart System** [Edit](#)

Description: Technology plays an important role in an underwriter's job. Underwriters use computer applications called "smart systems" to manage risks more efficiently and accurately.

Owner: Administrator       Parent Project: [000195: 1704pr1 \(Inception\)](#)

Submitter: serviceuser      Target Due Date:

Business Priority: (None)      Item Id: CHG000199

Severity: (None)

**State:** Planning

[Approved](#) [Reject](#) [Request F](#)

**Tasks** | Packages | Status Log | History Log | Properties

[Create Task](#)

Item Id	CM_ID	Title	State
DEV000405	QLARIUS_TASK_18	Update user documentation for diet feature	Raised
DEV000406	QLARIUS_TASK_19	Cope with diet information in the XML data store	Raised
DEV000407	QLARIUS_TASK_20	Add diet info to pending quote storage	Raised

From a change request, you can:

- Review information about the change request, including its change history, Quality Center relationships, associated tasks, associated change requests, and associated development packages by selecting the relevant tabs.
- Update the title, description, assignment, and other properties by clicking the **Edit** link.
- Transition the request to a different state, for example click the **Under Work** button to transition it to the Under Work state.
- Click the **Create Task** button on the **Tasks** tab to create a task that will then be related to this request, and that you can then assign to the specific resource who will do the work.

**Tasks** | Packages | Status Log | Quality | State Change History | Activi

[Create Task](#)

- View the Activity Log on the **History Log** tab to see the actions that have been performed on the change request. Click the arrows at the left of a row to expand the details.

12/28/2011 08:37:48 AM, 'Submit' by Josh Development Manager  
 12/28/2011 08:37:49 AM, 'Approved' by Josh Development Manager

Changed Value	Prior Value	New Value
Quality Item Status	Ready for Test	Define Test Cases
State	Planning	Ready for Work
Last Modified Date	12/28/2011 08:37:48 AM	12/28/2011 08:37:49 AM

12/28/2011 08:37:49 AM, 'Under Work' by Josh Development Manager  
 12/28/2011 08:38:34 AM, 'Update' by dmsys  
 12/28/2011 08:38:37 AM, 'Update' by dmsys

## Creating and Working on Tasks

From the Dev Tasks process app, you can create, update, and report on development tasks. You can relate tasks to projects in the ALM Projects process app, and relate tasks to Dimensions CM tasks.

### Finding Tasks

Display the **Search** tool from the left navigation pane. Here you can submit simple keyword searches, or search based on specific criteria. For details on searching in Serena Business Manager, see the Serena Business Manager online help. To display all tasks in the system, select Search by Keyword from the **Search** view, leave the **keyword(s)** field blank, and click **Search**.

Search by Keyword(s)

Keyword(s)

Match All  Match Any

Project(s)

Show My Projects | Manage My Projects

Search in Sub-projects

Active/Inactive  (Both)  
 Active  
 Inactive

Search

You can also find and track specific tasks by running and creating reports from the **Reports** view. For more on reporting in SBM, see the SBM online help.

## Creating New Tasks

You can submit a new task from the Dev Tasks process app by clicking the **Submit** area on the left navigation pane, and then clicking one of the submit links. For example, click **Browse and Submit to a Project** under Advanced tasks to browse the project tree and choose the correct project to submit to. You can also create new tasks from an open dev change request in the Dev Change Requests process app. When you create new development tasks, the tasks are synchronized to new requests in Dimensions CM of type *Task*. Development engineers will relate these tasks to the code changes they make in Dimensions CM in response to them.

When you submit a new task, a form like the following appears. Note that users with different privileges may see more fields on the submit form.

**Submit into:** Dev Tasks Project

OK Cancel

\* Task Name:  REC

\* Task Type: (None) ▾

Target Due Date:  mm/dd/yyyy

\* Task Owner: (None) ▾

Estimated Effort:  hours

Description:   
.....

Status Log:

These are comments you may leave for others to see and respond to.

When you create a new task, you set core properties for the task, including:

- From the **Item Type** field, the type of task, such as **Design** or **Development**.
- The title and description.
- The owner and estimated effort.

## Working on Tasks

Open a task to update and work on it. A typical dev task might look something like this.

The screenshot displays a task detail view. At the top left is a task icon and the name 'task2' with an 'Edit' button. The state is 'Raised'. Below this is a description field. The main section contains metadata: Owner (Emma Development Engineer (Deleted)), Submitter (Administrator), Estimated Effort (0 hours), and Actual Dev Effort (0 hours). It also shows a Parent Change Request (000204: cr4cm (Planning)) and an Item Id (DEV000206). A navigation bar includes 'Status Log', 'CM', 'History Log', and 'Properties'. The 'History Log' section shows a state change from 'Subtask Submit' to 'Raised'.

From a task, you can:

- Review information about the task, including its change history, Dimensions CM relationships, Quality Center relationships, associated change requests, and attachments.
- Transition a task from one state to the next. For example, to start working on a new task, click the **Begin Work** button.
- Click the **Edit** button to modify properties such as the owner of the task, estimated work, and attachments.

## Creating and Working with Development Packages

From the Dev Packages process app, you can create, update, and report on development packages. Development packages collect together all work associated with a release. You can gather all change requests related to a project, and then associate them with a baseline in Dimensions CM. This allows you to deploy the specific files in Dimensions CM that are related to the change requests, and run any build processes required to test the changes.

### Finding Packages

Display the **Search** tool from the left navigation pane. Here you can submit simple keyword searches, or search based on specific criteria. To display all packages in the

system, select **Search by Keyword** from the **Search** view, leave the **keyword(s)** field blank, and click **Search**.

You can also find and track specific packages by running and creating reports from the **Reports** view. For more on reporting in SBM, see the SBM online help.

## Creating New Packages

You can submit a new development package by clicking the **Submit** area on the left navigation pane, and then clicking one of the submit links. For example, click **Browse and Submit to a Project** under **Advanced Tasks** to browse the project tree and choose the correct project to submit to. Remember that you can also create new packages directly from the ALM Projects process app.

When you create a new package in the out-of-the-box Dev Packages process app, a screen like the following appears.

When you create a new package, you set core properties for the package, including:

- The title and description.
- The Release Engineer, Build Engineer, and QA Manager to assign to the package.
- Associated project from the ALM Projects process app. This then supplies the name of the Dimensions product and project / stream.

## Working on Packages

Open a package to update and work on it. A package might look something like this.

Package: **pack2** [Edit](#) State: Review & Verify

Description:

Owner: Administrator Parent Project: [000195:1704pr1 \(Inception\)](#)

Current Baseline: QLARIUS:2B

Package Iterations: 1

Item Id: DVPK000203

**Baselines** **Associated CR's** **Associated Tasks** **Status Log** **History Log** **Properties**

Item Id	Title	State	Severity	Business Priority	List of Child Tasks
<a href="#">000202</a>	cr3	Planning	(None)	(None)	

From a package, you can:

- Review information about the package, including its change history, associated Dimensions CM baselines, and associated ALM project.
- Transition the package to the next state in the workflow, for example if the package is in the **Ready To Build** state, you can click the **Build** button to transition it to the **Building** state. Note that you need the appropriate role, such as Build manager, to transition the package, otherwise the required button will not appear.
- Click the **Edit** button to change core properties for the package, including associated change requests, title and description, and the project from the ALM Projects process app that is related to the package.

## Creating New Dimensions CM Baselines

You can create new Dimensions CM baselines from the Dev Packages process app. When you create new baselines, you choose which change requests from the Dev Change Requests process app to associate with the new baseline. You can also create a new revised baseline.

**To create new Dimensions CM baselines:**

- 1 From a development package that is in the **Create Package** state, click the **Create Baseline** button.

Create Baseline Dev Package Project Dev Packages - DVPK000403: Enhancements for R1.0.1 ((Auto))



\* **Baseline Name:**  CM Project: QLARIUS:JAVA\_BRANCHA\_STR

**Baseline Name must be all caps, and "QLARIUS" will be prepended.**

\* **1. Selected Design Part:**

CmDesignParts

\* **2. Selected Baseline Type:**

CmBaselineTypes

\* **3. Selected Baseline Template:**

CmBaselineTemplates

Change Requests for this Baseline:

Enter value to find here

- 2 Enter the baseline name in the **Baseline Name** field. This is the name that will appear on the baseline in Dimensions CM. This must be defined in all capital letters.
- 3 In the **Selected Design Part** field, click the **Get Design Parts** button to list all options. Select a design part from which to scope items selected for the baseline..

\* **1. Selected Design Part:** QLARIUS:QLARIUS.A;1

CmDesignParts

QLARIUS:QLARIUS.A;1

- 4 In the **Selected Baseline Type** field, click the **Get Baseline Types** button and select a baseline type.

\* **2. Selected Baseline Type:** BASELINE

CmBaselineTypes

BASELINE

LABEL

- In the **Selected Baseline Template** field, click the **Get Baseline Templates** button and select a baseline template.

\* 3. Selected Baseline Template: ALL\_ITEMS\_LATEST

CmBaselineTemplates
ALL_ITEMS_APPROVED
ALL_ITEMS_LATEST

Create Baseline Dev Package Project Dev Packages - DVPK000207: Fixes for R1.0.1 ((Auto))

OK Cancel

\* Baseline Name:

R101FIXES

CM Project:

QLARIUS:JAVA\_BRANCHA\_STR

Baseline Name must be all caps, and "QLARIUS" will be prepended.

\* 1. Selected Design Part: QLARIUS:QLARIUS.A;1

CmDesignParts
QLARIUS:QLARIUS.A;1

Get Design Parts

\* 2. Selected Baseline Type: BASELINE

CmBaselineTypes
BASELINE
LABEL

Get Baseline Types

\* 3. Selected Baseline Template: ALL\_ITEMS\_LATEST

CmBaselineTemplates
ALL_ITEMS_APPROVED
ALL_ITEMS_LATEST

Get Baseline Templates

Change Requests for this Baseline:

Enter value to find here Find

--	--

- Search for change requests that you want to associate with the baseline, by selecting them in the **Change Requests for this Baseline** field.

Tasks associated with these change requests are in turn related to corresponding tasks in Dimensions CM; and files that have been updated in Dimensions CM in response to those tasks will be included in the baseline.

To use this field, enter part of the name of the change requests and click the **Find** button to locate them. Then select the required change request and click the arrow to move them to the right-hand list

Change Requests for this Baseline:

Find

DEF000402: AutoQuote Fixes (Planning)	FNC000400: Dietary Updates (Under Work)
FNC000400: Dietary Updates (Under Work)	
FNC000401: GUI Changes (Planning)	

If that doesn't work, click the  button. On the dialog box that appears, click the **Lookup** button to display a list of all available change requests, or enter a specific ID or title in the Item Id or Title field.

- 7 If you are revising an existing baseline, you must complete the options under Revised Baseline Settings:
  - Under **Previous Baseline Project**, locate the project in ALM Projects that the package and baseline belong to.
  - Then, locate the dev package to which the baseline is associated, under **Previous Baseline Dev Package**.
  - Finally, locate the existing baseline in the **Previous Baseline** field.
  - Based on the above searches, you can now update the change requests associated with this baseline. This will create a new revised baseline in Dimensions CM. The changes you make to associated change requests will in turn modify the tasks and associated versioned files related to the baselines in Dimensions CM.
- 8 Once you create the new baseline, it will be associated with the current package.
- 9 Once the baseline has been successfully created in Dimensions CM, the dev package will be transitioned to *Review and Verify*. At this point the Build Manager will verify the baseline and either select **Failed Verification**, meaning that another baseline will need to be created for the dev package, or select **Verified**, meaning that it is *Ready for Build*.



## Starting a Build

You can start a build from development packages that have been verified and are in the *Ready for Build* state.

### To start a build:

- 1 From a package in the **Ready for Build** state, click the **Build** button. Normally you will need to be logged in as the Build Manager to do this.

A form like the following appears.

The screenshot shows a configuration dialog box with the following elements:

- Buttons: OK, Cancel
- Baseline to be Built: QLARIUS:R101FIXES
- Collect Build Outputs: Yes (dropdown)
- Auto-Revise Baseline with Build Outputs: No (dropdown)
- Section: Configuration Management Settings
  - CM Product: QLARIUS
  - CM Project or Stream: QLARIUS:JAVA\_BRANCHA\_STR
  - \* 1. Selected Build Configuration: CmBuildConfigurations (list box with empty space)
  - \* 2. Selected Work Area: CmBuildAreas (list box with empty space)
  - Buttons: Get CM Build Configurations, Get CM Work Areas
- Section: Selected Baseline Data (collapsible header)

- 2 The build will be based on the baseline in the **Current Baseline** field, and will include the change requests listed in the **CRs in this Baseline** list.
- 3 Click the **Get CM Build Configurations** button and select a build configuration
- 4 Click the **Get CM Work Areas** button and select a work area.
- 5 Click **OK**.
- 6 The package is transitioned to the **Building** state while the build task is executed by Dimensions CM. Once the build task is complete, the package is transitioned either to the **Failed Build** state (if the build failed), or the **Built** state (if the build succeeded). If the build failed, you must start again with a fresh package, once the issues have been addressed. If the build succeeded, you can transition the package to the next state, **Ready For Test**, by clicking the **Send to QA** button.

## Displaying Project Metrics with the Serena<sup>®</sup> Dashboard

Serena Development Manager can be used with Serena Dashboard, a rich, graphical reporting tool that allows you to carefully monitor key performance indicators (KPIs). The Dashboard is fully configurable, and can provide graphical information on any aspects of your projects that managers or executives need to track.

Documentation is provided separately for Serena Dashboard through the Serena Help server at <http://help.serena.com/alm/sdb/index.html>.

### Included KPI & Metric Reports

Serena Dashboard includes graphs on several Key Performance Indicators (KPIs) and metrics that are relevant to most development organizations. These graphs are

configured to work directly out-of-the-box with the default Development Manager configuration. The Dashboard pulls data from the following sources:

- The Development Control process apps running on SBM, such as ALM Projects, Dev Change Requests, and Dev Packages.
- Dimensions CM, to display data on build success

The following KPI and metric graphs are included.

### Build Success Rate

This graph displays the percentage of builds that completed successfully. This data is pulled from Dimensions CM. For example:



You can click the graph to see information about specific builds, including the name of the Dimensions CM build configuration, when the build stopped, and whether it succeeded.

Build Success Rate

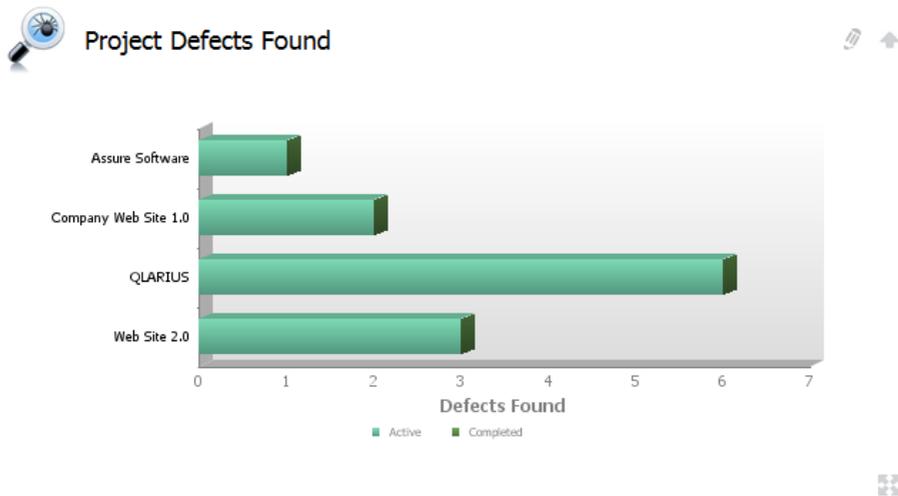
Success	Start Time	Project	Baseline	Config Name	User
	2011/08/29 23:45:19	QLARIUS	WEBSITE_2.0-A_BLD	ANT_JAVA_BUILD	dmsys
	2011/08/29 23:28:46	QLARIUS	WEBSITE_2.0-A	ANT_JAVA_BUILD	dmsys
	2011/08/29 23:23:36	QLARIUS	WEBSITE_2.0-A	ANT_JAVA_BUILD	dmsys
	2011/08/29 22:01:38	QLARIUS	WEBSITE_2.0-A	ANT_JAVA_BUILD	dmsys
	2011/08/29	QLARIUS	WEBSITE_2.0-A	ANT_JAVA_BUILD	dmsys

### Defect Escape Rate

This graph displays the percentage of defects that are escaped. These are defects that were reported by users or customers, that were not found by internal testing.

### Project Defects Found / Project Defects By Month

These graphs (*Project Defects Found* and *Project Defects by Month*) display, in different colors, the number of active and inactive defects either for particular projects, or found on from month to month. The following is an example of Projects Defects Found.



### Development Packages

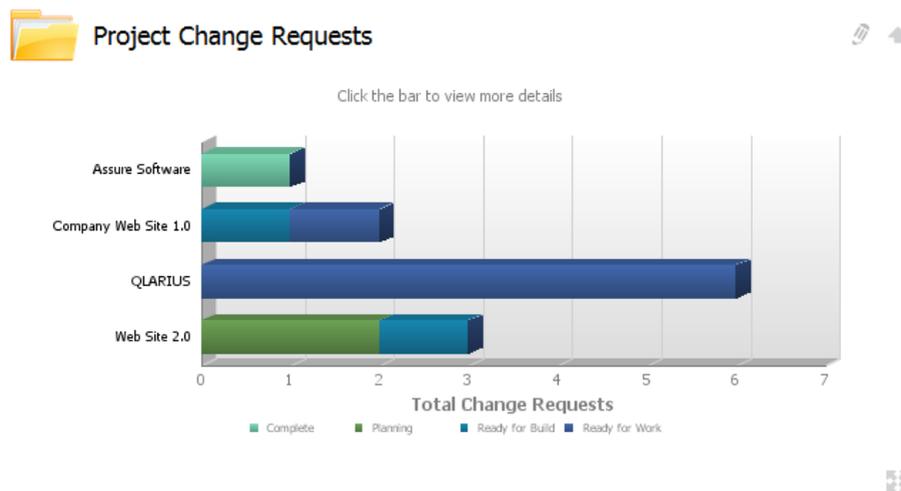
This graph lists the number of development packages in each project defined in ALM Projects. These packages are associated from the Dev Packages process app.



### Project Change Requests

This graph displays the number of open change requests against projects defined in ALM Projects. The change request count is pulled from the Dev Change Requests process app.

The change requests are color-coded according to their current state, such as Planning, Ready for Build, Ready for Work, and Complete.



### Project Status

The project status graph lists all current projects in the ALM Projects process app and displays their status as green, red, or yellow. This graph also lists the state that the project is currently in. You can click a project name to drill down into project state, start date, and end date.

 **Project Status**  

Project	Status	State	Project Manager	Last Modified
<a href="#">Assure Software</a>		Inception	Mira Project Manager	08/30/2011
<a href="#">Company Web Site 1.0</a>		Construction	Administrator	08/29/2011
<a href="#">QLARIUS</a>		Elaboration	Administrator	08/27/2011
<a href="#">Web Site 2.0</a>		Inception	Mira Project Manager	08/30/2011

## Getting to Know Dimensions CM

With Dimensions CM, you can manage your source code assets by collaborating on shared development projects and streams with your team members. You can synchronize your local workspace with Dimensions CM, automatically delivering your changes to the shared repository, as well as copying other users' changes to your workspace. With Dimensions CM, you can collect files related to a milestone, such as a release build, into a baseline that can be deployed for testing or release.

With Development Manager, you can synchronize tasks from the Dev Tasks process app running on Serena Business Manager to Dimensions CM requests. Developers can then work on tasks in context of their favorite integrated development environment (IDE). Dimensions CM provides a number of different clients to choose from, to best suite the different working styles on your team. You can use the Dimensions Web client, desktop client, IDE clients, or even command-line clients depending on what best suits you.

To get started learning more about Dimensions CM, please see *Getting Started with Dimensions CM*.

## Integrating with HP Quality Center

Use the Development Manager Connector for HP Quality Center to synchronize defects between the Dev Change Request process app and Quality Center. When users submit and update defects in one system, the defects are automatically submitted and updated in the other system. In this way, if your QA teams use Quality Center to record test results, you can ensure that all work against a project is tracked in Serena Development Manager.

The heart of the Connector for HP Quality Center is a set of Web Services that provide an API to key Quality Center features, such as submitting and updating defects. SBM process apps can interact with Quality Center using these Web services. For instructions on installing and configuring the Connector, please see the *Development Manager Connector for HP Quality Center / ALM*.

---

## Chapter 4

---

# Using the Requirements Manager Integration

Introduction	54
Creating Dev Change Requests from Requirement Distribution Tasks	55
Creating Dev Change Requests for the Requirements in an ALM Project	61
Viewing and Associating Requirements for a Dev Change Request	63
Viewing the Requirements for a Dev Package	65

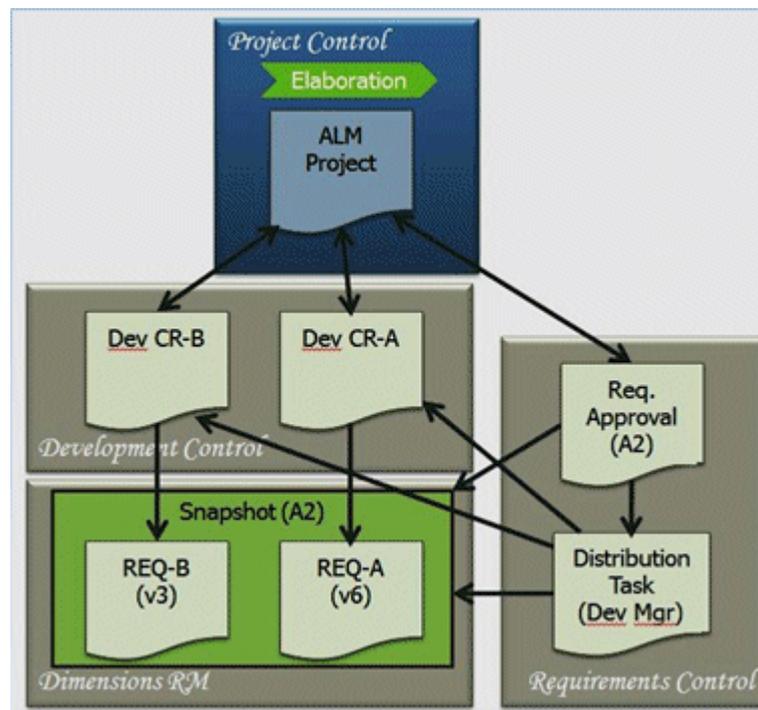
---

## Introduction

If you have installed the Requirements Manager option in addition to the Development Manager components, you will have an additional *Req Approval* tab containing the Requirement Approval process app. For details of installing and using Serena RQM, consult the documentation provided with that product, in particular the:

- Serena Requirement Manager Installation and Configuration Guide
- Serena Requirement Manager Getting Started Guide

In RQM, the Business Analyst(s) will use an RM collection related to the ALM Project (and an RM document based on it) to break down and organize the project's requirements as needed. Once a Requirements Approval ticket has been approved, the Business Analyst distributes the requirements to one or more teams (i.e. Development, QA, Documentation) by issuing distribution tasks.



The Requirements Approval process app consists of three sub-projects:

- Approval Process
- Approval Polling
- Distribution Tasks

Serena Development Manager is largely concerned with the Distribution Tasks sub-project which is concerned with processing a distribution task. The other sub-projects are concerned with the approval and polling processes that results in the requirement approval ticket being approved. The distribution process creates a Distribution Task ticket corresponding to the requirements approval that the Development Manager will then own and can use to satisfy the individual requirements by creating dev change requests associated with them.

Requirements are created in RQM in relation to an ALM project. This project will be visible within the ALM Projects process app.

When working with requirements, you can

- Select a distribution task and view and create dev change requests based on the requirements related to it. See ["Creating Dev Change Requests from Requirement Distribution Tasks" on page 55](#).
- Select a project and view and create dev change requests based on the requirements related to it. See ["Creating Dev Change Requests for the Requirements in an ALM Project" on page 61](#).
- Select a dev change request and view and add or remove requirements related to it. See ["Viewing and Associating Requirements for a Dev Change Request" on page 63](#).
- View the requirements that are included in a selected dev package. See ["Viewing the Requirements for a Dev Package" on page 65](#).

## Creating Dev Change Requests from Requirement Distribution Tasks

Use the Req Approval process app to view requirement distribution tasks that are assigned to you, and to create dev change requests related to their requirements. A single dev change request can be created for multiple requirements or multiple dev change requests can be created for a single requirement.

When a requirement approval has been approved in RQM, a requirement distribution task is generated and assigned to the development manager. From a selected distribution task, you can:

- Create a single dev change request and assign one or more requests
- Create a separate dev change request for each requirement you select.
- View the requirements belonging to the distribution task and any dev change requests that have already been assigned to them.

### Viewing a Distribution Task

- 1 Select the Req Approval tab.
- 2 Click Search for an existing requirement approval.

- 3 For Project(s) select **Distribution Task Project**.

Search by Keyword(s)

Keyword(s)

Match All  Match Any

Project(s)

- RQM
- Approval Process Project
- Approval Polling Project
- Distribution Tasks Project**

Show My Projects | Manage My Projects

Search in Sub-projects

Active/Inactive  (Both)  Active  Inactive

- 4 Select the search criteria and click **Search**.
- 5 Select the distribution task.

6 To view the requirements, select the Requirements tab.

MANAGER

Back to Results

**Distribution Tasks Project Requirement Approval - DTSK000003:** 1704ra1 - Create Dev CRs (In Progress)

Requirements Distribution Task:  
1704ra1 - Create Dev CRs

ALM Project: [000195: 1704prj1 \(Inception\)](#) Business Analyst: Administrator

Requirements Document Version: [drdocument1\\_1.0](#) Task Owner: Administrator

Requirements | History Log | Properties

Create Single Dev. Change Request | Create Multiple Dev. Change Requests

All Requirements

Title	Requirement ID	Version	Dev. Change Requests
<a href="#">Clarius UnderWriter</a>	BR_5	2	000204: cr4cm (Planning), 000208: cr6 (Planning)
<a href="#">1req</a>	BR_6	1	000196: 0704cr1 (Planning), 000202: cr3 (Planning), 000207: 4cm1 (Planning)
<a href="#">2req</a>	BR_7	1	000197: cr2 (Planning), 000202: cr3 (Planning)
<a href="#">3req</a>	BR_8	1	CHG000198: 3req (Planning)
<a href="#">UnderWriter Smart System</a>	UR_4	2	CHG000199: UnderWriter Smart System (Planning)
<a href="#">UnderWriter Data</a>	UR_5	2	
<a href="#">UnderWriter Access</a>	UR_6	2	

You can select:

- **All Requirements** to view all the requirements belonging to the distribution task
- **With Development Change Requests** to view only requirements that have a dev change request assigned to them.
- **Without Development Change Requests** to view only requirements that do not have a dev change request assigned to them.
- Click a requirement ID to view the requirement.

## Creating a Single Dev Change Request from One or More Requirements

- 1 Select the distribution task as above
- 2 Select the Requirements tab

Click the **Create Single Dev Change Request** button.

Alternatively, you can select the ALM Project to which the requirements belong, and on the Change Requests tab, click the **Create Dev CR** button

Create Single Dev. Change Request Distribution Tasks Project Requirement Approval - DTSK000003: 1704ra1

OK Cancel

\* Select the Requirements to assign to a single Development Change Requests:

BR_5 v2 - Glarius UnderWriter	→	
BR_6 v1 - 1req		
BR_7 v1 - 2req	←	
BR_8 v1 - 3req		
UR_4 v2 - UnderWriter Smart System		

Requirements

All Requirements

Title	Requirement ID	Version	Dev. Change Requests
<a href="#">Glarius UnderWriter</a>	BR_5	2	000204: cr4cm (Planning), 000208: cr6 (Planning)
<a href="#">1req</a>	BR_6	1	000196: 0704cr1 (Planning), 000202: cr3 (Planning), 000207: 4cm1 (Planning)
<a href="#">2req</a>	BR_7	1	000197: cr2 (Planning), 000202: cr3 (Planning)
<a href="#">3req</a>	BR_8	1	CHG000196: 3req (Planning)
<a href="#">UnderWriter Smart System</a>	UR_4	2	CHG000199: UnderWriter Smart System (Planning)
<a href="#">UnderWriter Data</a>	UR_5	2	
<a href="#">UnderWriter Access</a>	UR_6	2	

Select the requirements and move them to the right-hand list.

Click OK.

A Submit Dev Change Request page is displayed.

Submit into: Dev Change Requests Project

OK Cancel

\* Change Request Name:  ABC ✓

Change Request Type:  mm/dd/yyyy

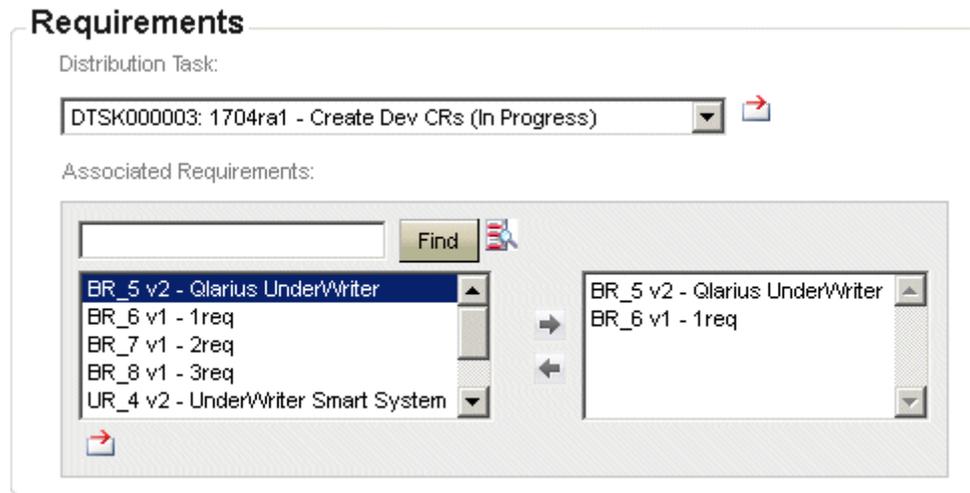
Target Due Date:  mm/dd/yyyy

Associate to Project:

Business Priority:  Severity:  Customer Found:  Estimated Effort:  hours

Enter a name for the dev change request.

At the bottom of the page, the requirements you selected are listed, but you can add or remove requirements from the list.

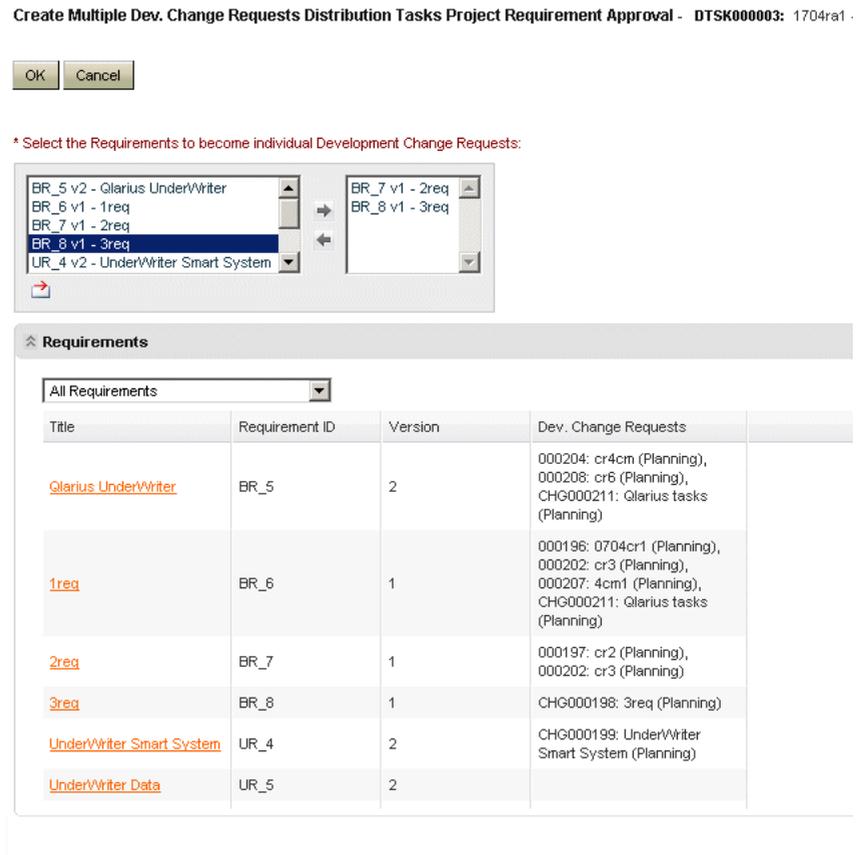


Click OK and the dev change request will be created.

## Creating one or More Dev Change Requests From Requirements

- 1 Select the distribution task as above.
- 2 Select the Requirements tab.
- 3 Click the **Create Multiple Dev Change Requests** button.

- On the Create Multiple Dev CRs form, select the requirements. You can use the shift key to select multiple requirements.



- Click **OK**.

After some processing, the dev CRs will be created. If you go into the Dev Chg Reqs process app to look for the items owned by you, you will see that there is a dev change

request created for each selected requirement with a title that is based on the title of the requirement and in the initial state *Planning*.

Search Results

Now showing Dev Change Requests 1 - 11 of 11    Sorted by: Project (Hierarchy)

Dev Change Requests Project	
Item Id	Title (State)
<input type="checkbox"/> 000196	0704cr1 (Planning)
<input type="checkbox"/> 000197	cr2 (Planning)
<input type="checkbox"/> CHG000198	3req (Planning)
<input type="checkbox"/> CHG000199	UnderWriter Smart System (Planning)
<input type="checkbox"/> 000202	cr3 (Planning)
<input type="checkbox"/> 000204	cr4cm (Planning)
<input type="checkbox"/> 000207	4cm1 (Planning)
<input type="checkbox"/> 000208	cr6 (Planning)
<input type="checkbox"/> CHG000211	Glarius tasks (Planning)
<input type="checkbox"/> CHG000212	2req (Planning)
<input type="checkbox"/> CHG000213	3req (Planning)

Check All | Uncheck All | Requery

(Project=Dev Change Requests Project)

Update All Checked    Create Link in -->

The ALM project to which they belong is the same one to which the requirements belong in RQM.

## Creating Dev Change Requests for the Requirements in an ALM Project

From the ALM Projects process app, you can:

- View the requirements related to the project
- View the Requirement Approvals related to the project.

### Viewing the Requirements Related to a Project

To view the requirements related to an ALM project:

- 1 Select the project

2 Select the Requirements tab.

Title	Requirement ID	Version	Distribution Tasks	Dev. Change Request
<a href="#">Glarus UnderWriter</a>	BR_5	2	DTSK000003: 1704ra1 - Create Dev CRs (In Progress), DTSK000004: 1704ra1 - Create QA Requirements (In Progress)	000204: cr4cm (Planning), 000208: cr6 (Planning)

To view the requirements:

- To view all the requirements for all requirement approvals related to the project, select **All**.
- To view only requirements that have a dev change request assigned to them, select **With Dev CRs**.
- To view only requirements that do not have a dev change request assigned to them, select **Without Dev CRs**.
- To view an individual requirement, click the **Title** link.

Note that the requirements displayed here are only ones that have been distributed for this ALM project, i.e. have a distribution task created for them. It does not show all requirements related to the project.

**To view the distribution tasks for an ALM project:**

- Click the Tasks tab.
- Select Distribution Tasks from the drop down list

Name	Item Id	Task Owner	State	Requirements Document
<a href="#">1704ra1 - Create Dev CRs</a>	DTSK000003	Administrator	In Progress	drdocument1 1.0
<a href="#">1704ra1 - Create QA Requirements</a>	DTSK000004	Administrator	In Progress	drdocument1 1.0

To view a distribution task, click its item ID.

**To view the RM collections for an ALM project:**

Click the Dimensions RM tab.

RM Collection Name:	drcollection1
RM Collection ID:	26

**To view the requirements related to dev change requests for an ALM project:**

Click the Change Requests tab.

Item Id	Title	State	Severity	Business Priority	Target Due Date	List of Child Tasks	Associated Requirem
000207	4cm1	Planning	(None)	(None)			BR_5 v1 - 1req
000208	cr6	Planning	(None)	(None)		DEV000209: 4cmtask112 (Under Work)	BR_5 v2 - Gtarius UnderWriter
000204	cr4cm	Planning	(None)	(None)		DEV000206: task2 (Raised)	BR_5 v2 - Gtarius UnderWriter
00000000	...	Planning	(None)	(None)			BR_5 v1 - 2req

Any requirement(s) that are related to a dev change request are shown in the Requirements column.

**To view the Requirement Approvals related to an ALM project:**

Click the Approvals tab.

Name	Item Id	Business Analyst	State	Requirements Docume
1704cm1	RA000001	Administrator	Distribution in Progress	rdocument1 1.0

Any requirement approvals that are related to the project are listed. You can click the name to view the requirement approval.

## Viewing and Associating Requirements for a Dev Change Request

For a selected dev change request, you can:

- View the requirements that are already associated with it
- Edit the dev change request to add or remove associated requirements.

**To view the requirements for a dev change request:**

- 1 Select the dev change request.

- 2 Click the Requirements tab.

Actions:



**Change Request**  
cr2 [Edit](#)

Description:

**State: Planning**

Owner: Administrator 

Submitter: Administrator 

Business Priority: (None)

Severity: (None)

Parent Project: 000195: 0204prj1 (Inception) 

Target Due Date:

Item Id: 000198

Requirement ID	Version	Title	Distribution Tasks
UR_4	2	UnderWriter Smart System	DTSK000004: 1ra - Create Dev CRs (In Progress), DTSK QA Requirements (In Progress), DTSK000009: 2ra - Create Dev CRs (In Progress), DTSK000010: 2ra - Create QA Requirements (In Progress), DTSK000015: 4ra - Create Dev CRs (In Progress), DTSK QA Requirements (In Progress)
UR_5	2	UnderWriter Data	DTSK000004: 1ra - Create Dev CRs (In Progress), DTSK QA Requirements (In Progress), DTSK000009: 2ra - Create Dev CRs (In Progress), DTSK000010: 2ra - Create QA Requirements (In Progress), DTSK000015: 4ra - Create Dev CRs (In Progress), DTSK QA Requirements (In Progress)

- 3 To view a requirement, click the requirement ID.

**To change the associated requirements for a dev change request:**

- 1 Select the dev change request.
- 2 Click the **Edit** link.
- 3 Scroll to the Requirements section.
- 4 Select a distribution task from the drop down list and click the **Find** button.

**Requirements**

Distribution Task:



Associated Requirements:

Enter value to find here  

BR\_5 v2 - Qlarius UnderWriter

UR\_4 v2 - UnderWriter Smart System

UR\_5 v2 - UnderWriter Data

UR\_6 v2 - UnderWriter Access

→

←



- 5 Add or remove the requirements to the right-hand list that you want to associate with the dev change request.

# Viewing the Requirements for a Dev Package

Select the dev package.

- Click the **Requirements List** tab to view the requirements and their associated dev change requests.

Dev Package Project Dev Packages - DVPK000203: pack2 (Review & Verify)

Package: **pack2** [Edit](#) State: Review & Verify  
[Verified](#) [Failed Verification](#)

Description:

Owner: Administrator Parent Project: [000195: 1704pr1 \(Inception\)](#)

Current Baseline: QLARIUS:2B

Package Iterations: 1

Item Id: DVPK000203

Baselines Associated CR's Associated Tasks Status Log History Log Properties **Requirements List**

Requirement(s) associated to this Dev Package  
Now showing Dev Change Requests 1 - 2 of 2 Sorted by: Project (Hierarchy)

Dev Change Requests Project	
Title (Requirements)	Dev. Change Requests (Requirements)
1req	000196: 0704cr1 (Planning), 000202: cr3 (Planning), 000207: 4cm1 (Plann
2req	000197: cr2 (Planning), 000202: cr3 (Planning), CHG000212: 2req (Planning)

You can click the icon to the left of the requirement title and select:

- Requirements** to view the requirement
- Dev Change Reqs** to view the dev change requests.

Dev Change Requests Project

Title (Requirements)	Dev. Change Rec
1req	000196: 0704cr1 (P
	000197: cr2 (Plann

Dev Change Reqs  
Requirements

Requery

- Click the **Associated CRs** tab to view the associated dev change requests and any requirements that are assigned to them

Baselines **Associated CR's** Associated Tasks Status Log History Log Properties Requirements List

Item Id	Title	State	Severity	Business Priority	List of Child Tasks	Associated Requirem
<a href="#">000202</a>	cr3	Planning	(None)	(None)		BR_6 v1 - 1req, BR_7 v1 - 2req

