HP Helion Codar

Software Version: 1.00

Concepts Guide

Document Release Date: December 2014
Software Release Date: December 2014

Legal Notices

Warranty

The only warranties for HP products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. HP shall not be liable for technical or editorial errors or omissions contained herein.

The information contained herein is subject to change without notice.

Restricted Rights Legend

Confidential computer software. Valid license from HP required for possession, use or copying. Consistent with FAR 12.211 and 12.212, Commercial Computer Software, Computer Software Documentation, and Technical Data for Commercial Items are licensed to the U.S. Government under vendor's standard commercial license.

Copyright Notice

Trademark Notices

Adobe® is a trademark of Adobe Systems Incorporated.

Microsoft® and Windows® are U.S. registered trademarks of Microsoft Corporation.

UNIX® is a registered trademark of The Open Group.

Documentation Updates

The title page of this document contains the following identifying information:

- Software Version number, which indicates the software version.
- Document Release Date, which changes each time the document is updated.
- Software Release Date, which indicates the release date of this version of the software.

To check for recent updates or to verify that you are using the most recent edition of a document, go to: **http://h20230.www2.hp.com/selfsolve/manuals**

This site requires that you register for an HP Passport and sign in. To register for an HP Passport ID, go to: **http://h20229.www2.hp.com/passport-registration.html**

Or click the **New users - please register** link on the HP Passport login page.

You will also receive updated or new editions if you subscribe to the appropriate product support service. Contact your HP sales representative for details.

Support

Visit the HP Software Support Online web site at: **http://www.hp.com/go/hpsoftwaresupport**

This web site provides contact information and details about the products, services, and support that HP Software offers.

HP Software online support provides customer self-solve capabilities. It provides a fast and efficient way to access interactive technical support tools needed to manage your business. As a valued support customer, you can benefit by using the support web site to:

- Search for knowledge documents of interest
- Submit and track support cases and enhancement requests
- Download software patches
- Manage support contracts
- Look up HP support contacts
- Review information about available services
- Enter into discussions with other software customers
- Research and register for software training

Most of the support areas require that you register as an HP Passport user and sign in. Many also require a support contract. To register for an HP Passport ID, go to:

http://h20229.www2.hp.com/passport-registration.html

To find more information about access levels, go to:

http://h20230.www2.hp.com/new_access_levels.jsp

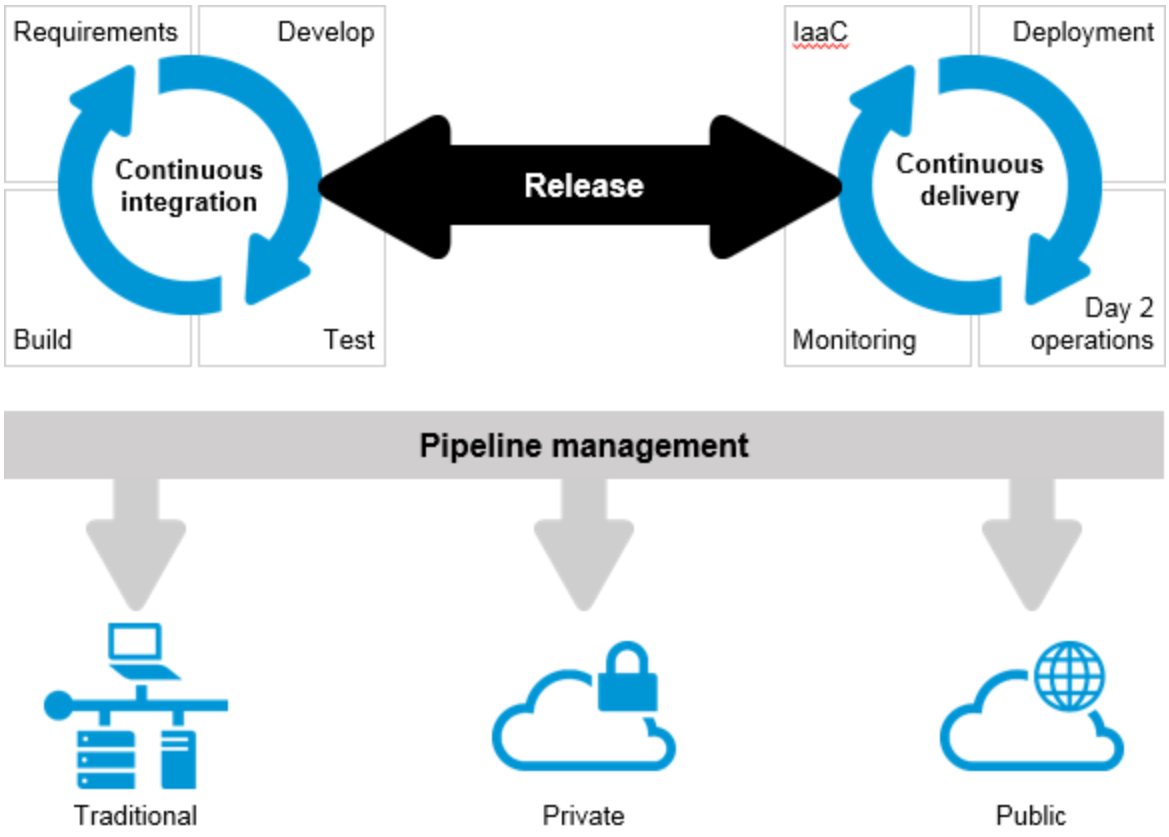
HP Software Solutions Now accesses the HPSW Solution and Integration Portal Web site. This site enables you to explore HP Product Solutions to meet your business needs, includes a full list of Integrations between HP Products, as well as a listing of ITIL Processes. The URL for this Web site is **http://h20230.www2.hp.com/sc/solutions/index.jsp**

Contents

HP Helion Codar

Organizations are facing new challenges when extending continuous integration into continuous delivery. Challenges include consistently deploying applications through development to production environments while considering the differences in those environments.

DevOps provides a framework to bridge the gaps between the development (Dev) and operations (Ops) environments by using a set of principles, methods, and practices around collaboration, automation, and governance. The goal is to extend continuous build or assembly integration to repeatable and consistent application deployment across heterogeneous environments. The following diagram illustrates the continuous integration and continuous delivery cycle in a DevOps environment.



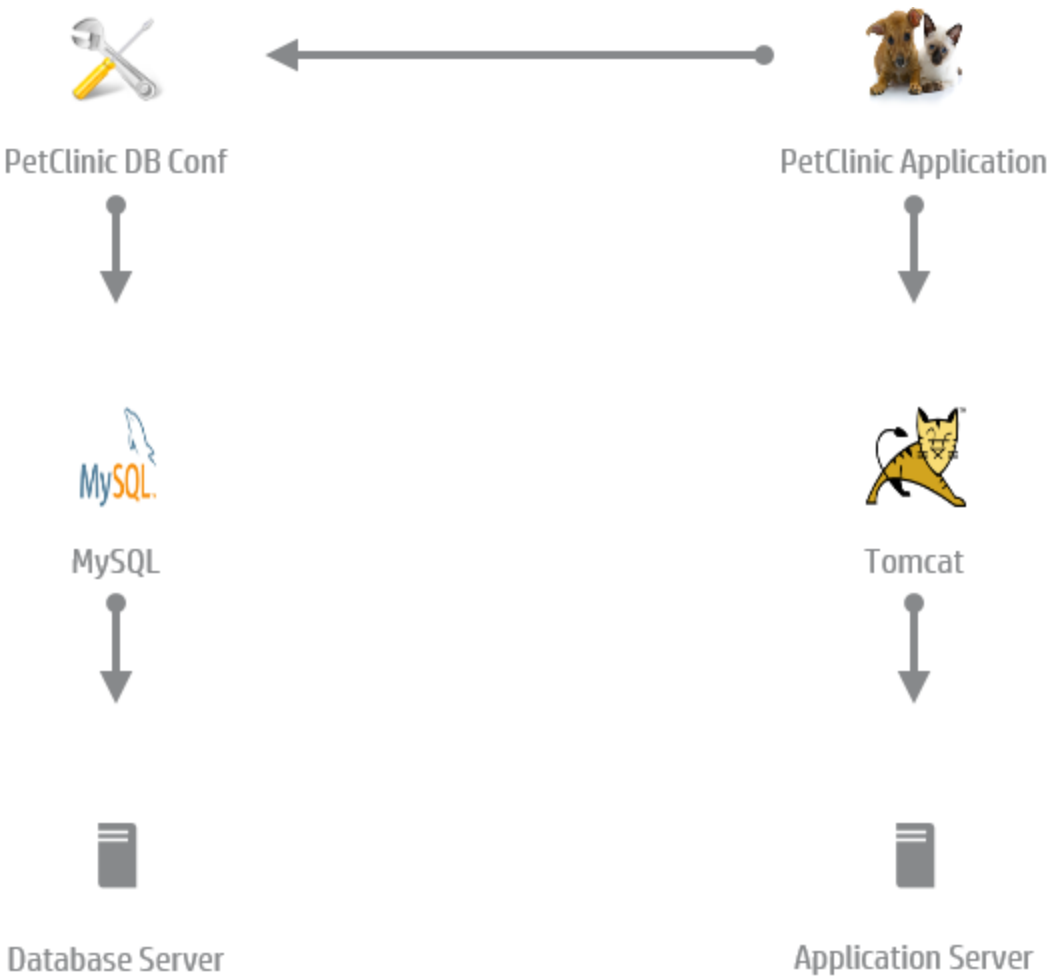
HP Helion Codar overview

HP Helion Codar facilitates continuous delivery in which every change to the system is releasable and that every code change can be deployed in production. It enables automation of continuous delivery where every code change triggers a build, which is deployed, automated unit tests are executed, and the application is automatically deployed to an environment based on policies that are defined in a runbook automation flow. Continuous delivery aims to deliver frequently and get fast feedback from users. Elements of the core value proposition include the following:

Declarative based modeling

Automating the deployment of applications using declarative based modeling allows the user to declare the end state of the application deployment (the application components and the dependencies between them) while the process to get to that state is triggered in the background. This allows the user to focus on what is deployed rather than how it gets deployed, which results in a shorter time to automate the deployment of multi-tier applications and greater simplicity in managing them over time.

HP Helion Codar supports declarative-based model development that involves creation, integration, and maintenance of complex designs through a user interface. A model consists of a topology design and its properties. HP Helion Codar provides flexibility for the user to modify the properties during the time of realization (similar to late binding).



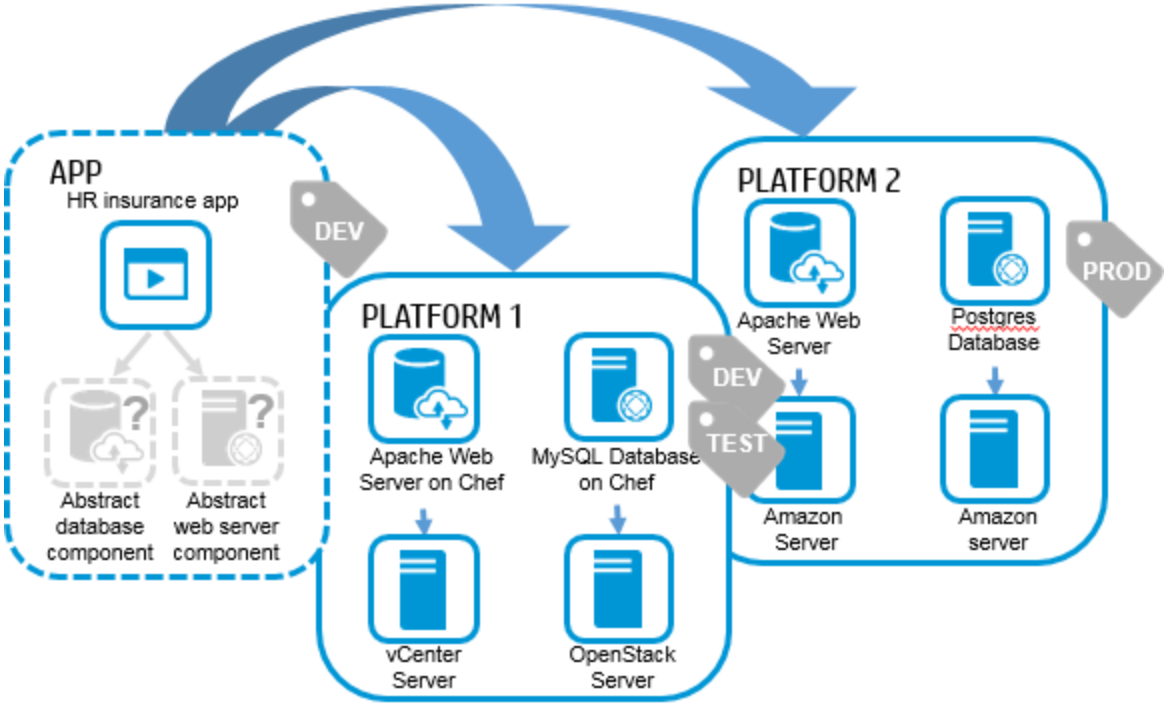
Topology composition

Topology composition is used to compose the application design with the infrastructure design at run time. During the application deployment, the infrastructure need varies for each deployment, the topology composition helps in defining these variable infrastructure needs in the application design and allows to compose with different infrastructure design in deploy time.

The capabilities and characteristics are used to describe the components. The application design will define the requirements using the capability components and characteristics in the design. The application design cannot be provisioned on its own and requires the selection of a compatible infrastructure design. The infrastructure design components are matched for their capability and characteristics to check the compatibility and the matching designs are chosen as compatible infrastructure design during the deployment.

The following illustration shows the topology composition for an HR insurance app. The app requires a database component and web server component, which are defined in the application design APP. This is fulfilled by PLATFORM1 as it has the Apache Web Server which has the web server capabilities and

its characteristics and MySQL database, which has the database capability and its characteristics. Similarly the PLATFORM2 also matches the APP requirements.



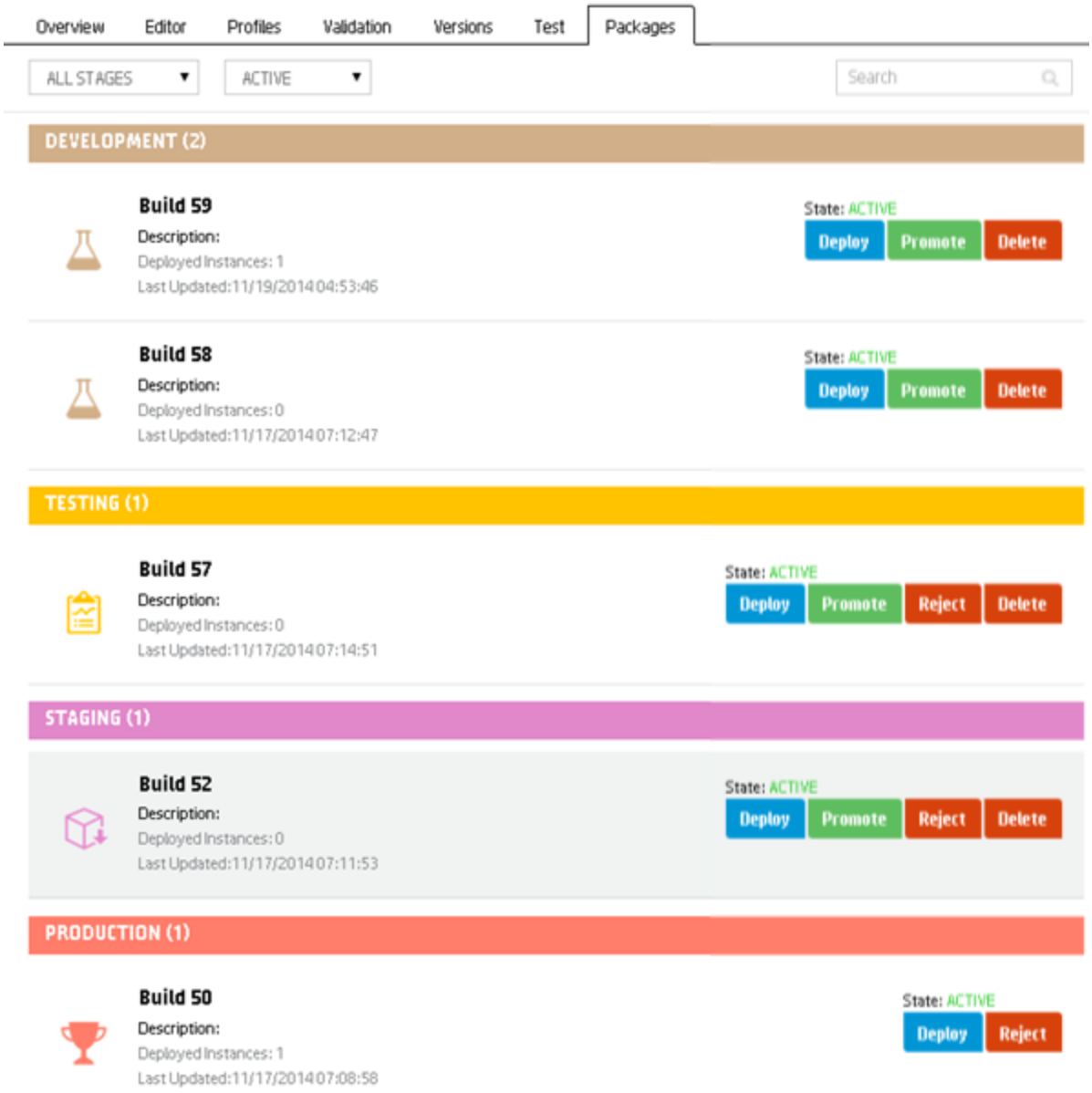
Infrastructure as code (IaaC)

Managing Infrastructure as Code (IaaC) allows IT teams to leverage the best practices for developing code such as code reviews and unit testing for how infrastructure and applications get provisioned.

HP Helion Codar has the ability to manage infrastructure as code. Topology designs that can contain server configurations, networks, volumes, relationships, and application-specific details like the application version and package information can be exported in JSON format and managed with the application in the source control system. Developers can make changes to the model using a text editor and use it for automation. The modified model can also be imported back into HP Helion Codar.

Application pipeline management

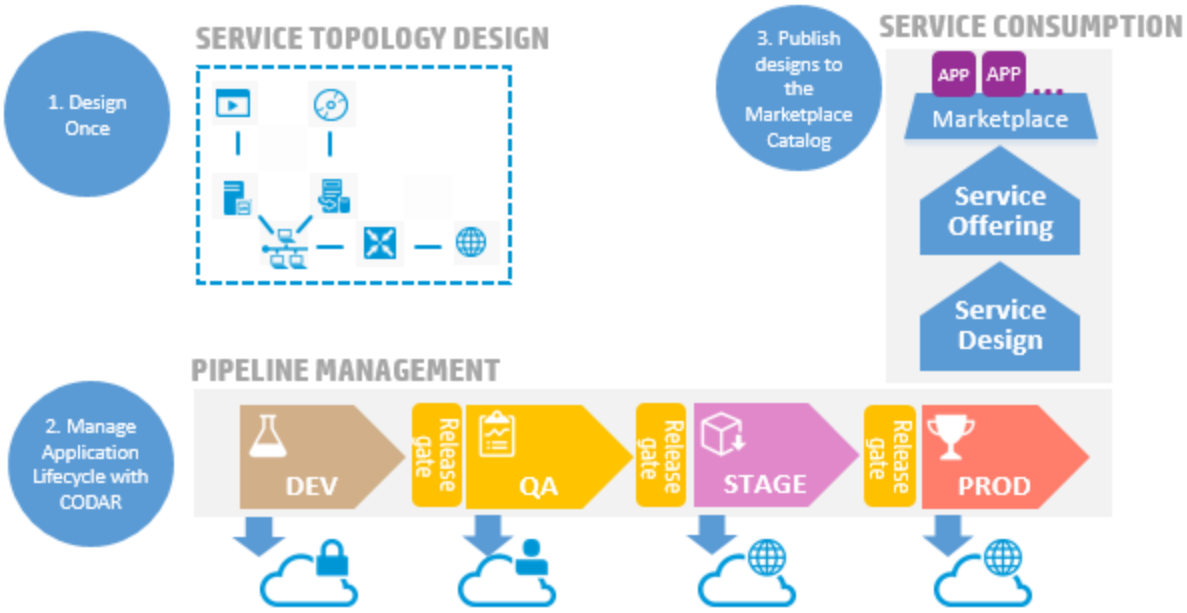
In order to increase visibility into the application release process, cross development, and operations, HP Helion Codar facilitates application pipeline management capabilities, as show in the following screenshot.



Packages are promoted from one stage to another in a consistent and repeatable manner. This ensures visibility to the developers when their applications are pushed into production. HP Helion Codar is a centralized structure for implementing a DevOps environment.

Jenkins

The following illustration shows pipeline management with service consumption. The package is built in Jenkins and then taken through various life cycle stages, such as Dev, QA, Staging, and Production. When the package is production ready, the design and production-ready package can be published for consumption and fulfillment.



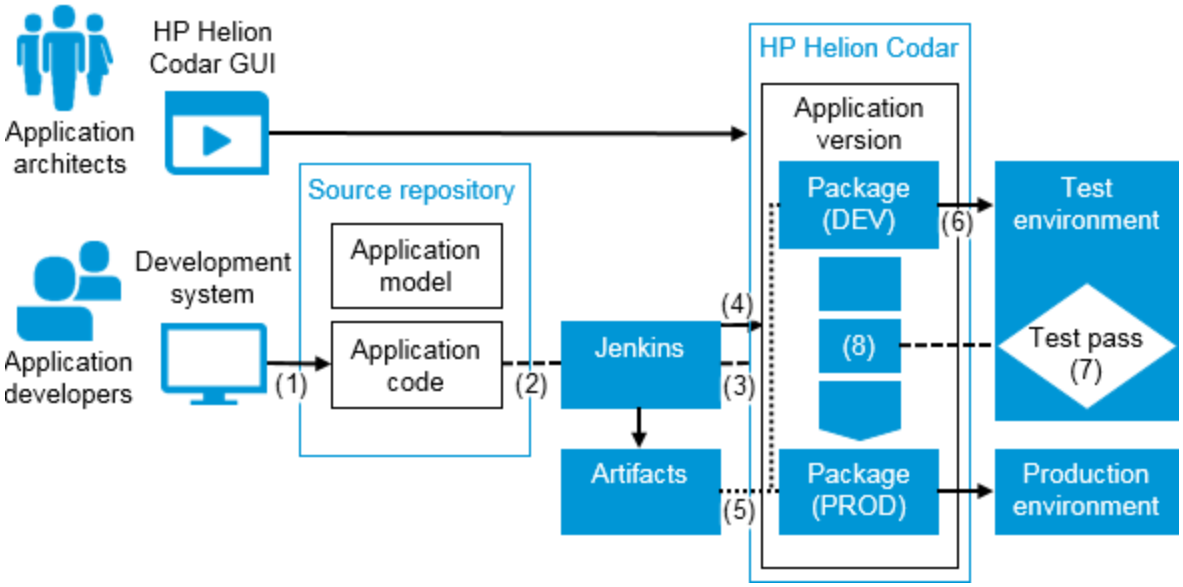
Open and extensible APIs

HP Helion Codar is open and extensible, and can be integrated with different build systems such as Jenkins, Hudson, etc. A comprehensive set of REST APIs can be used with other external tools to achieve continuous integration, deployment, and delivery. The HP Helion Codar architecture also provides options for you to hook into customized flows for DevTest and DevOps.

Use case – continuous integration, deployment, and delivery

The goal is for an application to be enabled for continuous integration (CI) and continuous deployment. An application developer codes the application and an application architect models the application in the HP Helion Codar interface and then exports the application model as code (IaaC). When the application developer checks in the code, a Jenkins build is triggered and the application is deployed using the application model on a specific environment. After the application is deployed, the continuous deployment process is extended to continuous delivery whereas application-specific tests can be automatically run on the deployed instance, with the application possibly being deployed to a different environment dependent on the outcome of the tests.

The following section describes how HP Helion Codar achieves this scenario.



Application modeling

Application architects model applications graphically by including the necessary components of the design in the designer interface and connecting them via relationships. HP Helion Codar contains a palette of standard components, and components can be imported (embraced) from various deployment engines such as HP Operations Orchestration and Chef. Such designs, called application models, are representations of the methods in which applications are to be deployed. An application model can be exported in JSON format and managed in an external source repository, achieving infrastructure as code (IaaC).

Continuous integration and deployment

In continuous integration, the code for the sample application and the model for the deployment of the application (in JSON format) is available in a source repository.

When an application developer makes a code change to the application and checks it into the source repository (1), Jenkins triggers a build (2).

HP Helion Codar provides a Jenkins plug-in which has details such as the IP address, user name, and password for HP Helion Codar. It establishes a connection and invokes an API as part of a post-build step (3). The API then invokes a workflow that executes various actions for achieving continuous deployment and continuous delivery as described in "Continuous deployment and DevOps " on page 13.

Importing an application design

If the application model has not already been imported into HP Helion Codar or if it has changed, the continuous deployment workflow imports it, in JSON format (IaaC), into HP Helion Codar (4) as a new version of the application design. This allows changes that have been made by application developers and architects to be taken into consideration at the time of deployment.

It is important to note that if the application model has already been imported or if there is no change in the application design, then this import operation does not take place and the application version within HP Helion Codar remains the same. You can view the application model in the Topology tile in the designer.

Managing packages

Packages represent a snapshot of an application design and allow properties to be parameterized within the design. We can also say that the package represents a particular build of an application.

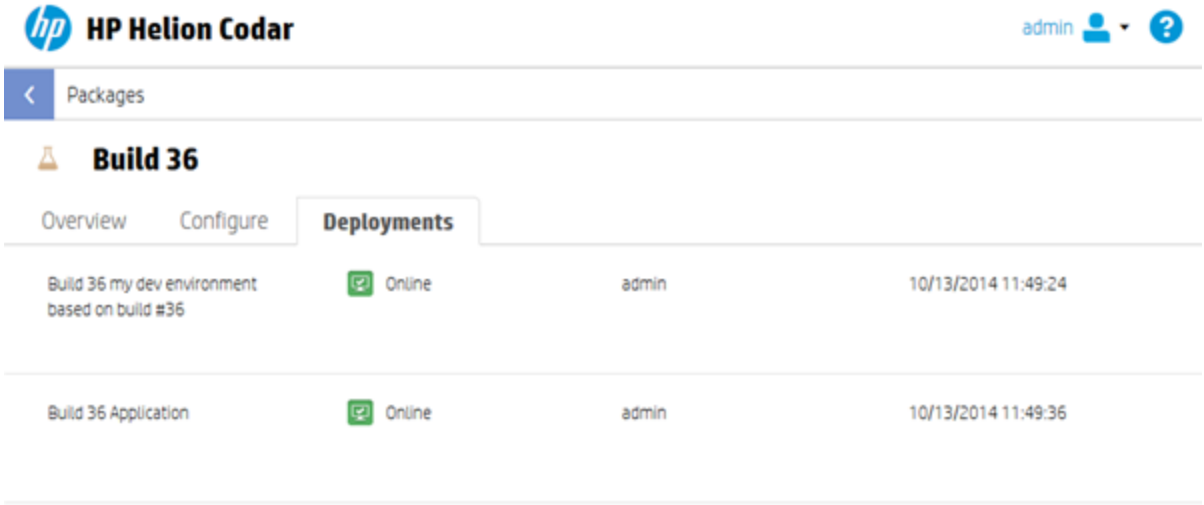
It is the smallest unit that can be deployed for an application. It represents both the implementation artifacts (the manner in which an application should be deployed) and deployment artifacts(the location of libraries like war, ear, etc., that should be deployed). From the HP Helion Codar perspective, a package comprises the following features:

- You can create a package from a specific application version. An application version can consist of multiple packages.
- Packages can be deployed. In this case the corresponding state of an application design along with the properties of the design specified in the package will be fulfilled.
- Packages are associated with a lifecycle stage. A package can belong to Development, Testing, Staging, or Production stages.

- Packages are associated with pipeline management. They can be moved across lifecycle stages through promotion. For example, a user with the QA role can reject a package.
- Packages in production can be published as offerings.

Deploying on an environment

After the package is created, the continuous deployment workflow fulfills the application design based on the environment (6). You can view deployments for the package on the Deployments tab:



A runbook automation engine creates an execution plan based on the design that fulfills the infrastructure layer, platform layer, and application layer. You can monitor the status of the deployment of a particular package and view a graphical representation of the deployed application, which includes component-level properties and actions.

Publishing a design

Publishing a design makes it available as an offering to service consumers. You must have an HP CSA license installed before you can publish a design.

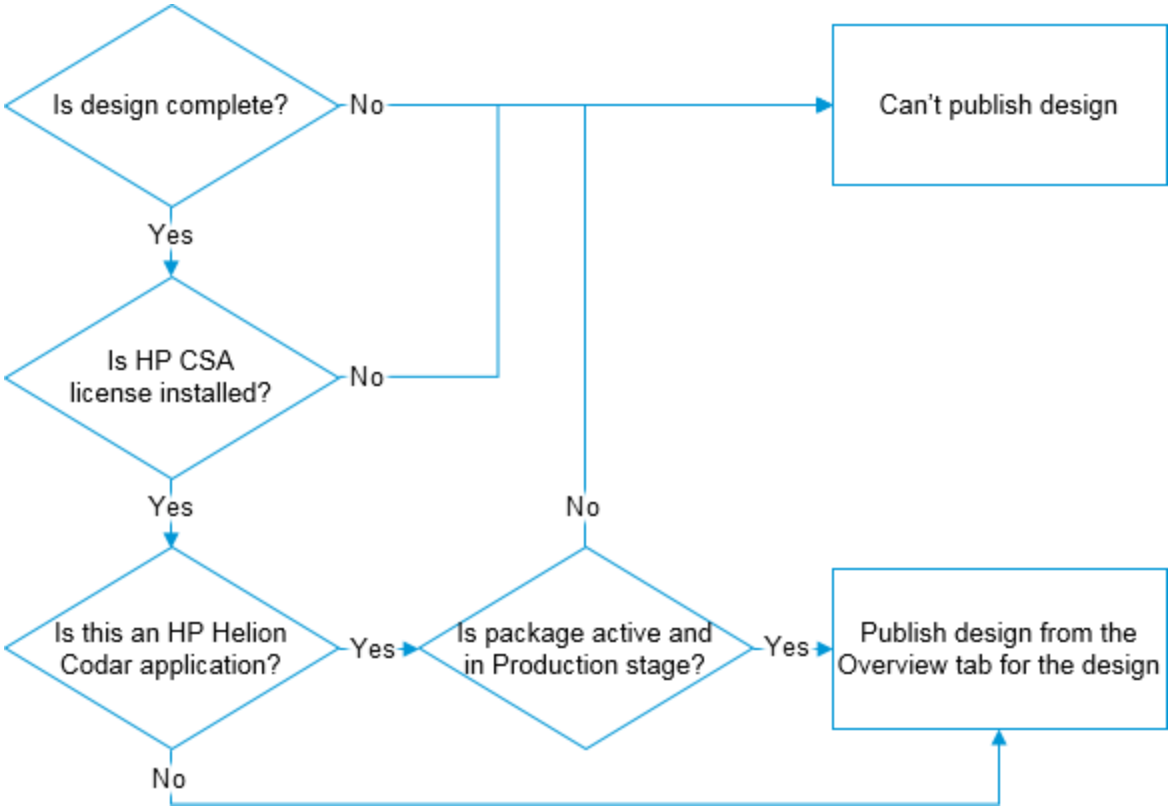
A complete design with an active package in the Production stage contains package-specific properties as part of the design and can be published.

A partial design with an active package in the Production stage contains package-specific properties as part of the design, but it cannot be published until a final composed design is created by deploying the production package.

Publishing a partial design is different depending on which licenses you have installed:

- An HP Helion Codar application design that has been advanced to the Production stage is deployed on a production infrastructure, and then the composed production design is made visible on successful production deployment. The design can then be published to service consumers.
- A design that is not an HP Helion Codar application design must be saved as a composed design from the Test tab. The design can then be published to service consumers.

The following figure illustrates when a design can be published based on the license used:



Continuous deployment and DevOps

The following sections explain continuous delivery using HP Helion Codar.

Running tests on an environment

After deploying the application on an environment using the continuous deployment flow, users can manually run their tests on this deployed instance. It is also possible to extend this continuous deployment flow so that tests are run automatically after the deployment is successful.

Lifecycle states, stages, and actions

Packages have the following lifecycle stages and corresponding actions:

Stage	Promote	Deploy	Edit	Delete	Reject	Refresh
Development	Yes	Yes	Yes	Yes	No	Yes
Testing	Yes	Yes	No	Yes	Yes	Yes
Staging	Yes	Yes	No	Yes	Yes	Yes
Production	No	Yes	No	Yes	Yes	Yes

Note: You may not have access to some or all of this functionality, depending on your role.

You can find the lifecycle stages on the Packages tab, and the current stage and state is shown on the Overview tab for a design.

Use the following actions to deploy or move a package through the stages:

- **Promote**: Moves the package to the next lifecycle stage. The package state remains Active.
- **Deploy**: Deploys the package.
- **Edit**: Change the properties of a package.
- **Reject**: Stops the package from advancing to another stage. The package will remain in its current stage, its state will be set to Rejected, and the action buttons will no longer be available.
- **Delete**: Delete a package.
- **Refresh**: Retrieves current package status.

Package states

Packages have the following states:

- Active
- Rejected

If you reject a package, then it remains in its current stage, its state is set to Rejected, and no further actions can be applied.

When a package is promoted, it moves to the next stage and remains in the active state. Packages are always created in the Development stage. If the HP Helion Codar Jenkins plug-in is configured, then after a successful build the Jenkins plug-in talks to HP Helion Codar and creates a package. Packages

will be in the Development stage. The package get deployed based on the environment configured in the Jenkins plug-in.

Roles in HP Helion Codar

Like HP CSA, organization roles provide authorization for members to perform tasks. These roles are configured and assigned by the administrator.

The Administrator role is the same as that for HP CSA, and users with this role have access to all areas in HP Helion Codar.

Application Architect Role

Users with this role can

- Create packages.
- View packages in any stage.
- Deploy, update, and delete packages in Development stage only.
- Embrace components.
- Create, update, and delete applications and application versions.

Users with this role cannot

- Promote or reject packages in any stage.

Application Developer Role

Users with this role can

- Create packages.
- View packages in any stage.
- Deploy, update, and delete packages in Development stage only.
- Promote packages from Development to Testing stage.

Application QA Role

Users with this role can

- View packages in any stage.
- Deploy, update, reject, and delete packages in Testing stage.

- Promote packages from Testing to Staging stage.
- Deploy, update, reject, and delete packages in Staging stage.

Application Release Manager Role

Users with this role can

- View packages in any stage.
- Deploy, update, reject, and delete packages in Staging stage.
- Promote packages from Staging to Production stage.
- Deploy, update, reject, and delete packages in Production stage.

Summary of Access by Role

The following table shows the predefined roles and the tasks to which each has access.

	Application Architect	**Application Developer**	**Application QA**	**Application Release Manager**
Embrace components	Yes			
View packages	Yes	Yes	Yes	Yes
Create packages	Yes	Yes		
View application and application versions		Yes	Yes	Yes
Create applications and application versions	Yes			
Update applications and application versions	Yes			
Delete applications and application versions	Yes			
Update packages in Development stage	Yes	Yes		
Delete packages in Development stage	Yes	Yes		

Deploy packages in Development stage	Yes	Yes		
Promote packages to Testing stage		Yes		
Update packages in Testing stage			Yes	
Delete packages in Testing stage			Yes	
Deploy packages in Testing stage			Yes	
Promote packages to Staging stage			Yes	
Reject packages in Testing stage			Yes	
Update packages in Staging stage			Yes	Yes
Delete packages in Staging stage			Yes	Yes
Deploy packages in Staging stage			Yes	Yes
Promote packages to Production stage				Yes
Reject packages in Staging stage			Yes	Yes
Update packages in Production stage				Yes
Delete packages in Production stage				Yes
Deploy packages in Production stage				Yes
Reject packages in Production stage				Yes

Moving from one environment to another

After successful deployment, you can execute tests and if the tests are successful then the packages can be promoted to next stage. Packages can be deployed at every stage and you can either promote

or reject the packages. However, packages cannot be rejected at the Development stage, and your user role determines what you can do at each stage.

Next steps

HP Helion Codar Installation and Configuration Guide explains how to download, install, and configure the software, and *HP Helion Codar API Quick Start* provides a brief introduction to the REST APIs and explains how to obtain detailed information for each API. You can also access online help from the application for task-oriented assistance.

Send Documentation Feedback

If you have comments about this document, you can contact the documentation team by email. If an email client is configured on this system, click the link above and an email window opens with the following information in the subject line:

Feedback on Concepts Guide (Helion Codar 1.00)

Just add your feedback to the email and click send.

If no email client is available, copy the information above to a new message in a web mail client, and send your feedback to csadocs@hp.com.

We appreciate your feedback!


Java™
COMPATIBLE