

RELEASING HIGH-QUALITY
APPLICATIONS AND
INFRASTRUCTURE FASTER
WITH vREALIZE CODE STREAM

Table of Contents

| | |
|--|----|
| Abstract | 3 |
| The Need for Speed | 3 |
| How to Accelerate Application Release Delivery | 4 |
| Understanding Pipeline Automation | 5 |
| Tool Chain Integration | 6 |
| Artifact Management Is Critical | 7 |
| Process Governance and Control | 8 |
| Dashboards and Reporting Provide Visibility | 10 |
| Enabling IT to Apply DevOps Best Practices to Infrastructure | 10 |
| Bridging the Gap Between Development and Operations | 11 |

Abstract

If your company relies on applications to enable new business opportunities and drive competitive advantage, getting faster at deploying infrastructure or applications for developers and test engineers might be solving only part of your problem. For companies that are looking to shorten their software delivery process, see why VMware® vRealize Code Stream™ empowers development, infrastructure, and operations teams to release software more frequently and efficiently, thereby driving greater business agility.

The Need for Speed

“Business agility” is probably one of the most overused marketing terms hyped by enterprise technology companies. When companies say they want to become more agile, what does that mean? In most cases, getting faster at provisioning infrastructure and applications is not what they mean by becoming more agile. Getting faster and more efficient at these tasks is just the means to the end. That end is delivering value to their customers. For companies that rely on technology, new and updated applications are the engine that drives business opportunity and competitive advantage.

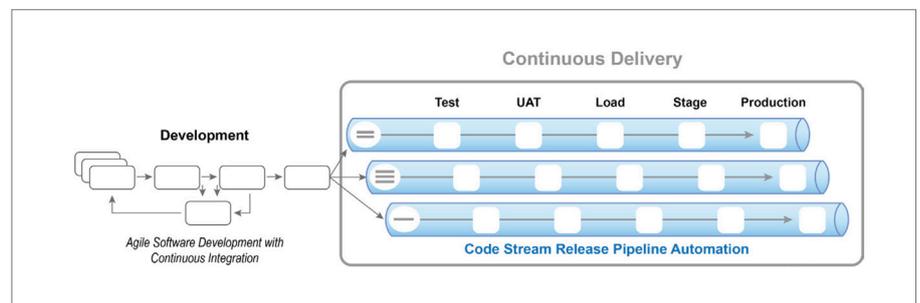
For these companies, “business agility” means getting faster and more efficient at the application delivery process. In recent years, many companies have implemented agile development processes with continuous integration and smaller, more frequent releases. Although developers are getting faster at turning out code, getting that code through all the different stages of development and into production remains a huge challenge for most companies. Manual processes that worked with 6-to-12-month release cycles can’t stand up to the demands of quarterly, monthly, or even more frequent releases.

Companies leverage a variety of software development lifecycle tools to coordinate the build deployment and testing at each stage. With more frequent releases, the problem becomes not only performing these tasks faster, but coordinating across these different environments to make sure that you are deploying and testing the same software versions and configurations at each stage in the process.

How to Accelerate Application Release Delivery

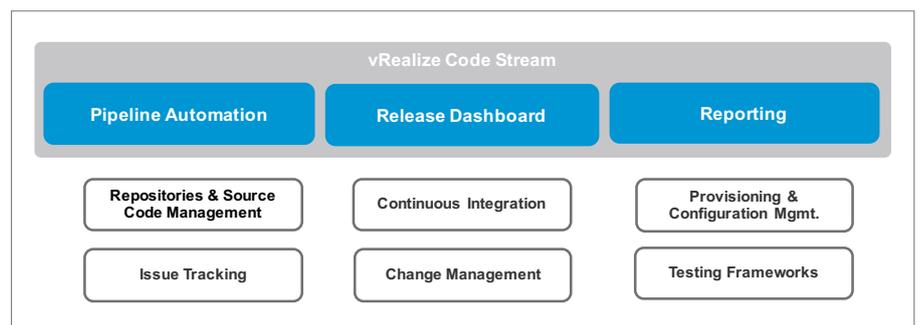
VMware vRealize Code Stream is a release automation solution targeted at organizations that use agile development approaches and that want to automate a manual release process. It allows developers and operations teams to release software more frequently, more efficiently, and at lower operational risk, all while leveraging their existing investments in developer and operations tools.

vRealize Code Stream enables continuous delivery by automating the tasks required to build, deploy, and test at each stage in the release delivery pipeline, including gating rules between stages.



To automate the delivery, vRealize Code Stream integrates with and orchestrates the release process by leveraging existing software development lifecycle tools. vRealize Code Stream has three core functional modules and leverages six supporting functions within those modules (see the following diagram). The three core modules are:

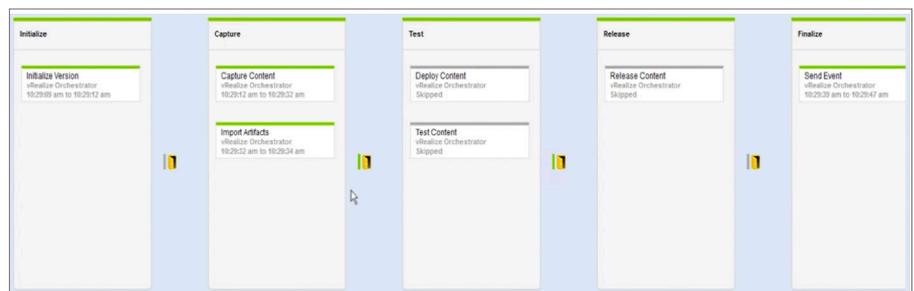
- **Pipeline automation** – Model any release process for any kind of software.
- **Release dashboard** – Get real-time visibility into your release process.
- **Reporting** – Identify problem areas to help improve release quality and efficiency over time.



Understanding Pipeline Automation

In a release delivery process, after development makes enhancements or fixes defects, several additional groups need to test, stage, and accept the release prior to putting it into production. Today, this is largely accomplished using processes in silos with tools that might not be consistent across groups. Not only does the process to deploy and test at each stage take too long, coordination and consistency at each stage also suffer.

vRealize Code Stream allows application release teams to model any kind of release process for any kind of application from simple, single-node, on-premises applications to complex, multitiered, cloud-based, or hybrid next-generation applications. Pipeline models configure the workflow tasks and governance policies used to build, deploy, and test the software at each stage in the delivery process as well as the gating rules between stages. An example of a pipeline is shown here.



Furthermore, as organizations move from waterfall to agile development to DevOps and true continuous delivery, vRealize Code Stream adapts to an organization's evolving release automation maturity. It allows release teams to model manual tasks and approvals, or it can automatically promote builds from one stage to the next based on test results or predefined rules. So as organizations' release processes mature, vRealize Code Stream pipeline templates can be modified from partial to full automation to evolve with them.

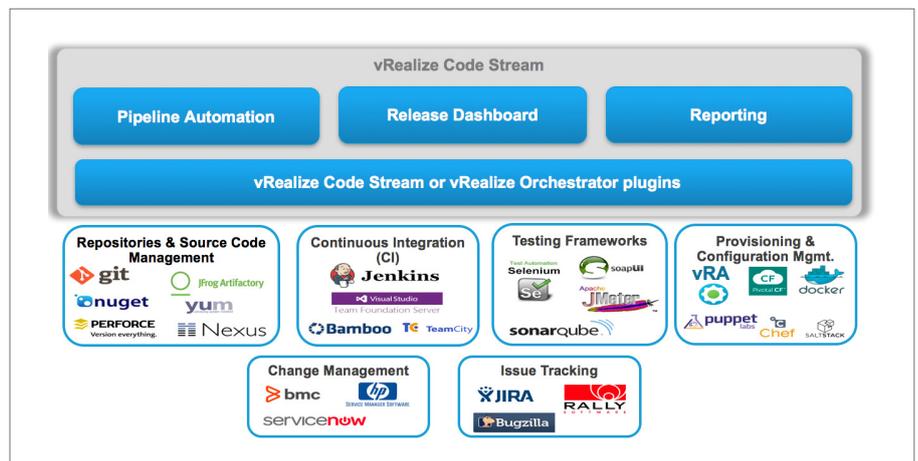
vRealize Code Stream helps add order, speed, and reliability to the release process by:

- Automating the different tasks needed to provision, deploy, test, monitor, and decommission the software targeted for a specific release
- Ensuring standardized deployment configurations, by coordinating the deployment of the right blueprints and artifacts at each release delivery stage
- Providing governance and control across the end-to-end process, ensuring process consistency at each phase in the delivery pipeline
- Leveraging existing tools and processes to minimize disruption and leverage prior investments

This last point is the key to understanding how vRealize Code Stream automates and provides governance and control over the release delivery process. It does not replace your existing tools and processes. Rather, it leverages and works with your collection of independent multivendor tools to orchestrate the end-to-end software delivery process to ensure consistency and predictably across all stages of the release delivery pipeline.

Tool Chain Integration

vRealize Code Stream includes out-of-the-box support for a variety of different development, test, provisioning, and operations tools. In addition, a robust integration framework allows companies to leverage almost any of their existing tools, custom scripts, and process investments. The product's integration framework consists of native plug-ins and the ability to leverage VMware vRealize Orchestrator™, the VMware general-purpose automation and orchestration engine.

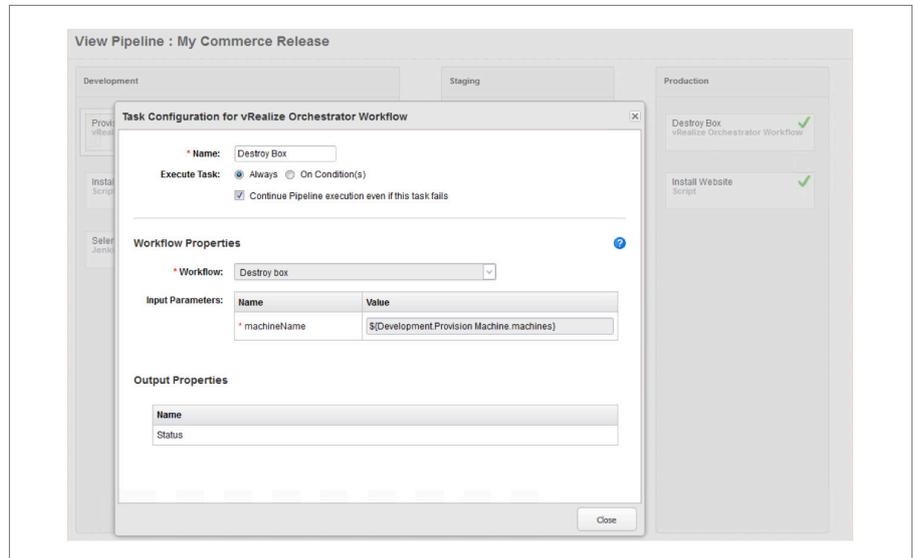


Native vRealize Code Stream Plug-Ins

vRealize Code Stream provides direct plug-ins to popular continuous integration tools (e.g., Jenkins, Microsoft Team Foundation Server, Atlassian Bamboo), repositories (e.g., JFrog Artifactory), provisioning solutions (VMware vRealize Automation™, Cloud Foundry), defect tracking tools (e.g., JIRA, Bugzilla), and other systems. Additionally, the vRealize Code Stream plug-in SDK allows organizations to create their own custom plug-ins.

vRealize Orchestrator

vRealize Orchestrator is a general-purpose workflow and runbook automation tool that can be used to automate almost any task. vRealize Orchestrator workflows can be invoked from various tasks within vRealize Code Stream pipelines. It can be integrated with almost any software development, provisioning, or deployment tool, as well as storage, networking, security, and many other management systems. You can create custom workflows, invoke existing scripts, or leverage a library of VMware and partner-developed vRealize Orchestrator workflows and plug-ins that are available in the [Cloud Management Marketplace \(https://marketplace.vmware.com/vsx/\)](https://marketplace.vmware.com/vsx/) of the VMware Solutions Exchange. The following screenshot contains an example of how to invoke a vRealize Orchestrator workflow from a vRealize Code Stream pipeline task.



Integration with vRealize Automation

vRealize Code Stream offers tight integration for infrastructure provisioning and application deployment. When deployed together, the combined solution offers a single platform to handle IT service provisioning **and** accelerate the application delivery lifecycle. The governance policies and automated delivery of vRealize Automation ensure that each group in the application delivery pipeline receives the right size application in the right configuration for the task to be performed.

Artifact Management Is Critical

One of the most critical aspects of being able to release software faster and more efficiently is to make sure everyone in the release delivery process is using the same software binaries. All the improvements in development and testing will be wasted if different groups are using different binary versions from different repositories when they provision or deploy an application. Not only do you need to coordinate these binary artifacts across multiple groups, but you also need to coordinate the different components across the multiple versions of an application that might be active at any point in time. As release windows continue to shrink, keeping track of these software artifacts can become a daunting task.

vRealize Code Stream supports the modeling and resolution of artifacts so that the right artifact versions are automatically retrieved and then fed to deployment scripts or tools when a particular build version of an application is deployed. This means organizations can easily control, store, and manage binary artifacts throughout the software release cycle, and ensure that the right artifact version is deployed every time.

Process Governance and Control

Automation is what makes the software delivery process go faster, but faster does not guarantee that the software will be more stable and reliable. Governance policies are the rails that ensure that your release train stays on track and that the software delivered will run reliably.

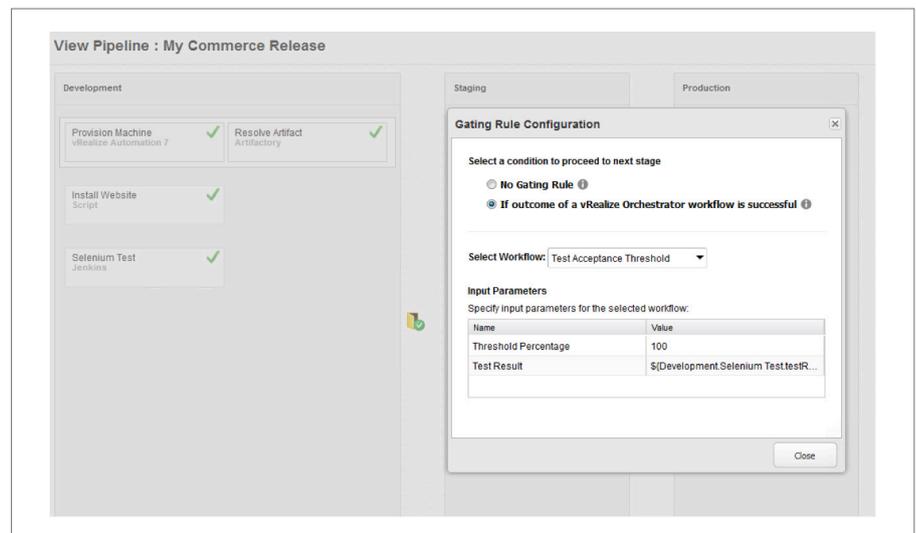
Throughout the release delivery process, you also want visibility into the status of your release to allow authorized users to rapidly take corrective action, keeping your release train on schedule. Enhanced visibility enables better communication and collaboration across teams and less finger pointing if something goes wrong.

Governance of the Release Delivery Process

Let's explore how vRealize Code Stream provides the governance controls necessary to complement an automated delivery process. The product has two types of governance policies: native and inherited. Native governance policies are specified directly in vRealize Code Stream. Inherited governance policies are specified in the tools that vRealize Code Stream invokes.

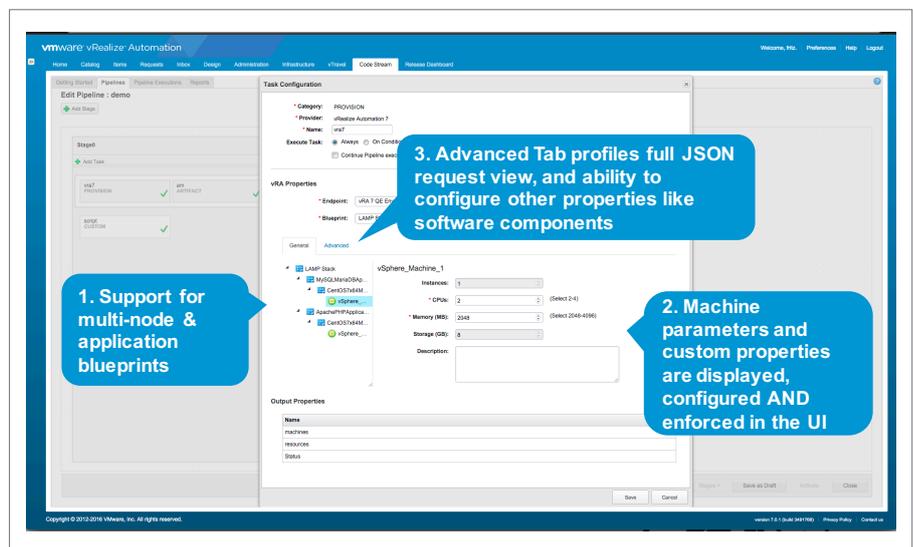
Native Governance Policies

The primary governance policy is the workflow specified in the release pipeline. This describes the steps and tasks that need to be performed at each stage in the release process. vRealize Code Stream task definitions include information and attributes that need to be passed to the tools performing the tasks. Gating rules determine if and when the software release can move to the next stage and what conditions (e.g., percentage of test passed, manual approvals) need to be satisfied before proceeding to the next stage.



Inherited Governance Policies

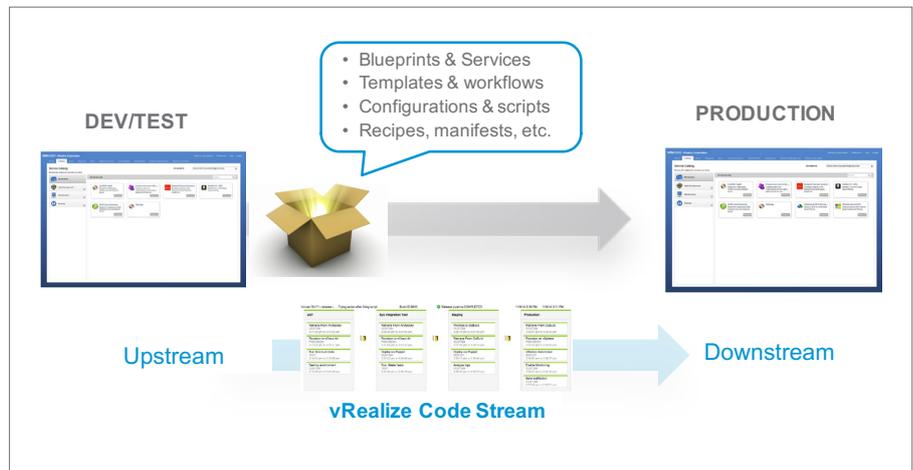
When vRealize Code Stream invokes another tool to perform a task such as build, deploy, or test, it inherits the governance policies of that application. For example, consider a provisioning task, in which vRealize Code Stream invokes vRealize Automation to provision infrastructure for the new release's test environment. The administrator defining the task can select only infrastructure services that they are authorized to use; they will be provisioned on resources reserved for their group, and each virtual machine will only be able to use the maximum amount of resources specified in the machine's blueprint. The same principles that apply to inheritance of governance policies from vRealize Automation apply to vRealize Code Stream invoking other tools.



From a governance perspective, vRealize Code Stream adds additional governance policies and works with your existing software development lifecycle tools by inheriting their governance policies.

LEARN MORE

Additional information can be found at <http://www.vmware.com/products/vrealize-code-stream>.



Bridging the Gap Between Development and Operations

If your company wants to get faster and more efficient at delivering value to your customers through new and updated applications, development and operations need to work better together. For companies that have a DevOps or continuous delivery initiative, vRealize Code Stream is a release automation solution that empowers development and operations teams to become more efficient and effective at releasing software.

- Task automation and process standardization reduce manual effort and, more important, accelerate completion of every task at each stage in the software delivery pipeline.
- Artifact management and governance policies help ensure consistency at each stage in the process.
- Out-of-the-box integration with a broad range of application development tools, along with an SDK that makes it easy to integrate additional tools, promote maximum developer productivity.
- Reporting and release dashboards allow all groups to monitor the status and foster greater collaboration among teams.



VMware, Inc. 3401 Hillview Avenue Palo Alto CA 94304 USA Tel 877-486-9273 Fax 650-427-5001 www.vmware.com

Copyright © 2017 VMware, Inc. All rights reserved. This product is protected by U.S. and international copyright and intellectual property laws. VMware products are covered by one or more patents listed at <http://www.vmware.com/go/patents>. VMware is a registered trademark or trademark of VMware, Inc. and its subsidiaries in the United States and other jurisdictions. All other marks and names mentioned herein may be trademarks of their respective companies. Item No: 65016vmw-twp-FY18Q3SDDCLaunchvRCS-A4-101