

Guida introduttiva a vRealize Automation Code Stream

Febbraio 2022

vRealize Automation 8.7

È possibile trovare la documentazione tecnica più aggiornata sul sito Web di VMware all'indirizzo:

<https://docs.vmware.com/it/>

VMware, Inc.
3401 Hillview Ave.
Palo Alto, CA 94304
www.vmware.com

VMware, Inc.
Centro Leoni Palazzo A
Via Spadolini 5
Ground Floor
Milan, MI 20121
tel: +39 02 30412700
fax: +39 02 30412701
www.vmware.com/it

Copyright © 2022 VMware, Inc. Tutti i diritti sono riservati. [Informazioni sul copyright e sui marchi](#)

Sommario

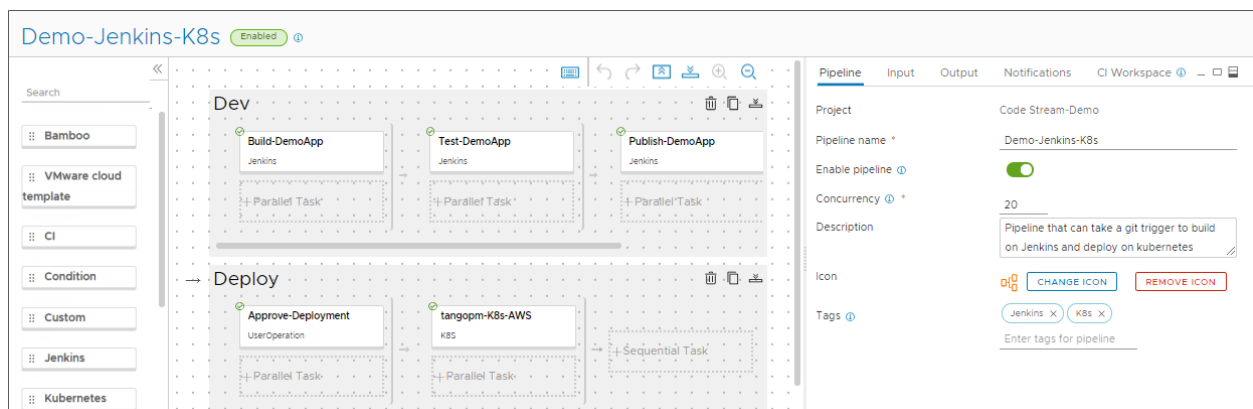
- 1** Cos'è Code Stream 4
- 2** Quali sono le funzioni di Code Stream 5
- 3** Prima di iniziare 8
- 4** Come configurare 10
- 5** Quali sono gli altri obiettivi che è possibile raggiungere 14

Cos'è Code Stream

1

vRealize Automation Code Stream™ è un software CI/CD (integrazione e consegna continue) che consegna software in maniera rapida e affidabile, con un sovraccarico molto contenuto. Code Stream supporta la distribuzione di applicazioni legacy monolitiche e utilizza contenitori Docker e Kubernetes in esecuzione in cloud multipli.

Con Code Stream, è possibile creare pipeline che automatizzano l'intero ciclo di vita di DevOps utilizzando al tempo stesso gli strumenti di sviluppo esistenti come Git e Jenkins.



Code Stream semplifica i passaggi di creazione, test e distribuzione delle applicazioni e aumenta la produttività al rilascio di codice sorgente dal repository di sviluppo, passando per il testing, fino alla produzione. Code Stream supporta applicazioni personalizzate e commerciali e oggetti quali i VMware Cloud Templates.

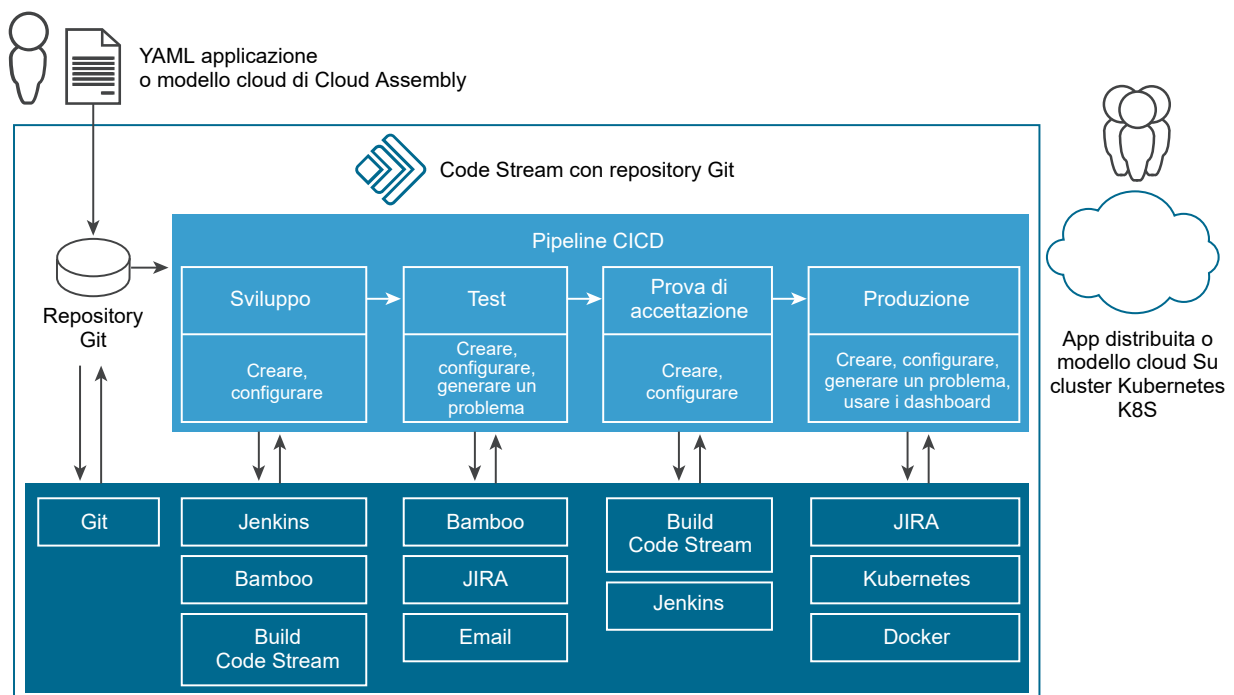
A partire da vRealize Automation 8.2, i blueprint vengono chiamati VMware Cloud Templates.

Quali sono le funzioni di Code Stream

2

Code Stream modella le attività nel processo di rilascio del software, automatizza lo sviluppo e il test del codice dello sviluppatore e lo rilascia nell'ambiente di produzione.

Esso integra il processo di rilascio con gli strumenti dello sviluppatore per realizzare attività specifiche, tracciando tutti gli artefatti e le versioni del codice.



È possibile creare una pipeline che esegue azioni per creare, distribuire, testare e rilasciare il software. Code Stream esegue il software attraverso ogni passaggio della pipeline finché non è pronto per essere rilasciato in produzione.

È possibile integrare la pipeline con uno o più strumenti DevOps, quali ad esempio origini dati, repository e sistemi di notifica, che forniscono alla pipeline i dati per l'esecuzione.

Ad esempio, potrebbe essere necessario utilizzare diversi endpoint in una pipeline che scarica e distribuisce un'applicazione Kubernetes da GitHub a un cluster Kubernetes.



Code Stream si integra con varie tipologie di endpoint.

Tabella 2-1. Code Stream si integra con strumenti DevOps

Endpoint	Funzioni
Git	Estrae codice dello sviluppatore dal repository e lavora con l'attivazione Git per attivare le pipeline quando gli sviluppatori archiviano di codice.
Kubernetes	Automatizza i passaggi per distribuire, scalare e gestire le applicazioni containerizzate.
Code Stream Build	Crea build native per l'integrazione continua senza dover utilizzare integrazioni di terze parti.
Jenkins	Crea artefatti del codice.
Email	Invia notifiche agli utenti.
JIRA	Crea un ticket quando una pipeline non riesce.
Bugzilla	Crea e traccia i bug.

Perché utilizzare Code Stream

Gli sviluppatori e gli amministratori di Code Stream possono utilizzare Code Stream per automatizzare l'intero ciclo di vita delle versioni di DevOps, continuando al tempo stesso a utilizzare gli strumenti di sviluppo esistenti. Code Stream offre i seguenti vantaggi:

- Automazione semplificata
- Plug-in già pronti che funzionano senza open source

- Esperienza di modellazione semplificata e pipeline come codice
- Integrazione totale con VMware Tanzu Kubernetes Grid Integrated Edition (in precedenza denominato VMware Enterprise PKS)
- Creazione di report e approfondimenti
- Visibilità end-to-end con dashboard dettagliati
- Dashboard personalizzati
- Metriche e approfondimenti DevOps
- Governance
- Accesso basato sui ruoli
- Variabili segrete e riservate e approvazioni
- Progetti

Code Stream semplifica la distribuzione di applicazioni software man mano che il codice sorgente attraversa le fasi di sviluppo e test, per poi essere rilasciato in produzione. Esso inoltre semplifica la produttività utilizzando l'attivazione Git. Quando lo sviluppatore archivia del codice, Code Stream può attivare la pipeline e automatizzare la creazione, il test e la distribuzione dell'applicazione.

È possibile utilizzare Code Stream con altri componenti di vRealize Automation.

- Distribuire un modello cloud di Cloud Assembly e utilizzare i valori dei parametri esposti dal modello cloud.
- Pubblicare la pipeline in Service Broker in modo che altri membri del team possano richiederli e distribuirli nelle proprie regioni cloud.

Per gli altri modi di utilizzare Code Stream, vedere [Capitolo 5 Quali sono gli altri obiettivi che è possibile raggiungere con Code Stream](#).

Prima di iniziare con Code Stream

3

Gli amministratori o gli sviluppatori di Code Stream devono registrarsi per creare un account My VMware ed effettuare l'accesso prima di poter iniziare a utilizzare Code Stream.

Accesso

Dopo aver effettuato l'accesso, è possibile creare, eseguire e gestire le pipeline.

Tabella 3-1. Preparazione per l'utilizzo di Code Stream

Pas sag gio	Azione	Operazioni da eseguire
1	Creare un account di My VMware.	Utilizzare il proprio indirizzo e-mail aziendale per creare l'account di My VMware in https://my.vmware.com/ .
2	Accedere a vRealize Automation.	Aprire la porta HTTPS 443 per il traffico in uscita con accesso attraverso il firewall a: <ul style="list-style-type: none">■ *.vmwareidentity.com■ gaz.csp-vidm-prod.com■ *.vmware.com
3	Aprire il servizio Code Stream.	

Guida introduttiva per gli amministratori di Code Stream

L'amministratore dispone del ruolo Code Stream `Administrator` e può creare endpoint per assicurarsi che per gli sviluppatori siano disponibili istanze funzionanti. È possibile creare, gestire e attivare pipeline e molto altro.

Ad esempio, gli sviluppatori potrebbero dover connettere le attività delle loro pipeline a un'origine dati a un repository o a un sistema di notifica. Questi componenti forniscono i dati per l'esecuzione delle pipeline.

È inoltre possibile integrare Code Stream con altri componenti di vRealize Automation.

- Utilizzare Cloud Assembly per distribuire VMware Cloud Templates.
- Utilizzare Service Broker per pubblicare pipeline e attivarle.

Guida introduttiva per gli sviluppatori di Code Stream

Gli sviluppatori possono utilizzare Code Stream per creare ed eseguire pipeline e per monitorare l'attività delle pipeline sui dashboard. Sarà necessario disporre del ruolo `User`.

Dopo aver eseguito una pipeline, si desidererà sapere se:

- Il codice ha avuto esito positivo in tutte le fasi della pipeline. I risultati vengono visualizzati nelle esecuzioni della pipeline.
- La pipeline non è riuscita e per quale motivo. I principali errori vengono visualizzati nel dashboard della pipeline.

Utilizzo dei modelli di pipeline smart

Per risparmiare tempo quando si crea una pipeline che crea, testa e distribuisce l'applicazione in modo nativo, utilizzare i modelli di pipeline smart. Ogni modello di pipeline smart pone una serie di domande e crea una pipeline in base alle risposte alle domande sugli aspetti seguenti:

- Obiettivi della build, ambienti e posizione in cui si trova il codice sorgente.
- Obiettivi della distribuzione e posizione in cui si intende distribuire l'applicazione. Ad esempio, il modello di pipeline smart identifica i cluster Kubernetes esistenti. È quindi possibile selezionare un cluster da utilizzare quando si crea e si distribuisce l'applicazione.

Dopo che il modello di pipeline smart ha creato la pipeline, è possibile modificarla ulteriormente in base alle proprie esigenze.

Impostazione di Code Stream

4

In qualità di amministratore che configura Code Stream, dopo aver effettuato l'accesso, è possibile aggiungere endpoint, creare ed eseguire pipeline e visualizzare i risultati.

Code Stream si connette agli endpoint per ottenere i dati delle pipeline da eseguire. In questo caso d'uso, Code Stream si connette a un repository GitLab in modo che la pipeline possa scaricare un file Kubernetes.

Nell'interfaccia utente di Code Stream è disponibile un processo introduttivo in forma di configurazione guidata. Fare clic sull'icona **Guida** e scegliere **Configurazione guidata**.

Prerequisiti

- Verificare che esista un repository GitLab o GitHub in locale e che contenga il codice la pipeline utilizzerà.

Procedura

- 1 Aggiungere un endpoint Git che connetta Code Stream al repository GitLab locale.
 - a Fare clic su **Endpoint**.
 - b Selezionare il tipo di endpoint Git e inserire nome e descrizione.
 - c Inserire le informazioni rimanenti.
 - d Per testare la connessione all'endpoint, fare clic su **Convalida** e salvare l'endpoint.

- 2 Fare clic su **Pipeline**, creare una pipeline e aggiungere un'attività che utilizza l'endpoint Git. Facoltativamente è possibile aggiungere una notifica via email.

The screenshot displays the vRealize Automation Code Stream interface for a project named "MyKubeApp4" (status: Disabled). The interface is divided into two main sections: a visual pipeline editor on the left and a task configuration panel on the right.

Visual Pipeline Editor:

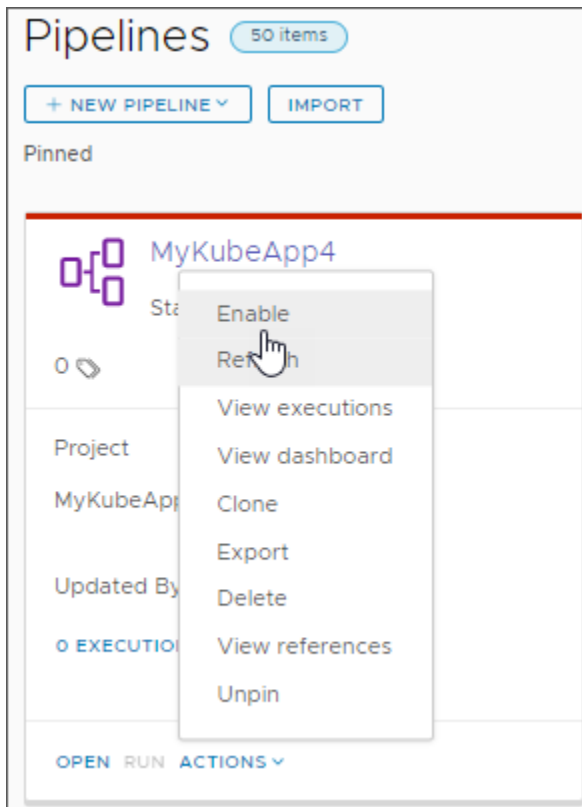
- At the top, there are tabs for "Workspace", "Input", "Model", and "Output".
- The "Model" tab is active, showing a visual representation of the pipeline with stages and tasks.
- Stage0:** Contains a task named "Task0" of type "Kubernetes".
- Stage1:** Contains a task named "Task0" of type "Jenkins".
- Below Stage1, there is a dashed box labeled "Stage" with a plus icon, indicating a new stage can be added.

Task Configuration Panel (Task: Task0):

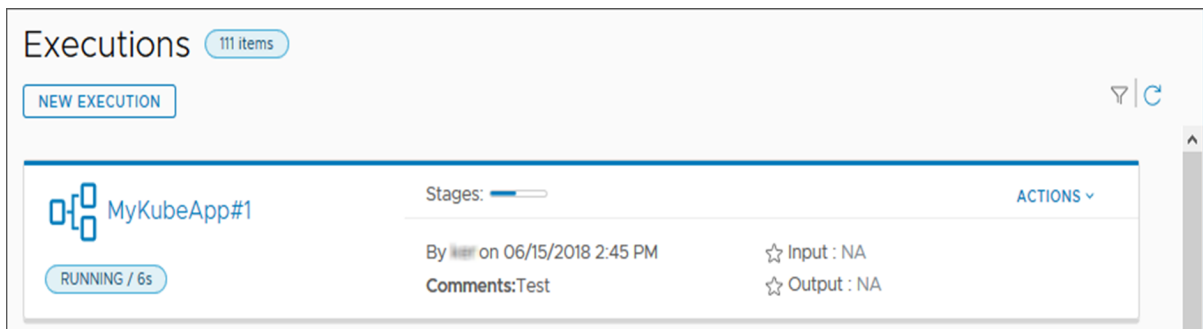
- Task name:** Task0
- Type:** Kubernetes
- Continue on failure:** ☐
- Execute task:** ☒ Always ☐ On condition
- Kubernetes Task Properties:**
 - Kubernetes cluster:** Dev-AWS-Cluster
 - Timeout (in mins):** 5
 - Action:** ☒ Get ☐ Create ☐ Apply ☐ Delete ☐ Rollback
 - Filter by label:** Enter label to filter
 - Source type:** ☒ Source Control ☐ Local definition
 - Git:** GitLab-OnPrem
 - File path:** Enter filepath
- Parameters:** A table with columns "parameter name" and "parameter value". A note below states: "Provide the \$\$ variables used in YAML file as parameters below. Ex: GIT_BRANCH_NAME: \$(input.GIT_BRANCH_NAME) or master".
- Output Parameters:** A button labeled "status".

At the bottom of the interface, there are buttons for "SAVE", "RUN", and "CLOSE", along with a status indicator "Last saved: a year ago". A "VALIDATE TASK" button is also present in the top right of the configuration panel.

- 3 Salvare la pipeline e fare clic su **Abilita** per abilitarne l'esecuzione.



- 4 Dopo aver abilitato la pipeline, fare clic su **Esegui**.
- 5 Fare clic su **Esecuzioni** e osservare la pipeline durante l'esecuzione.



- 6 Se la pipeline non riesce, correggere il problema ed eseguirla nuovamente.
- 7 Fare clic su **Dashboard** e selezionare il dashboard della pipeline in modo che possa monitorare l'attività della pipeline.

Risultati

La pipeline è stata eseguita e ha scaricato il file dello sviluppatore da un'istanza di GitLab. L'attività della pipeline ha distribuito l'applicazione a un cluster Kubernetes ed è stato possibile monitorare personalmente tutta l'attività sul dashboard della pipeline.

Operazioni successive

Se non si trovano le informazioni necessarie, è possibile ottenere ulteriore assistenza nel prodotto.



- Per ottenere informazioni specifiche del contesto, quando e dove sono necessarie, fare clic e leggere le indicazioni e le descrizioni comandi nell'interfaccia utente.
- Aprire il pannello di supporto all'interno del prodotto e leggere gli argomenti visualizzati per la pagina dell'interfaccia utente attiva. Per ottenere risposte alle domande è anche possibile effettuare ricerche nel pannello.

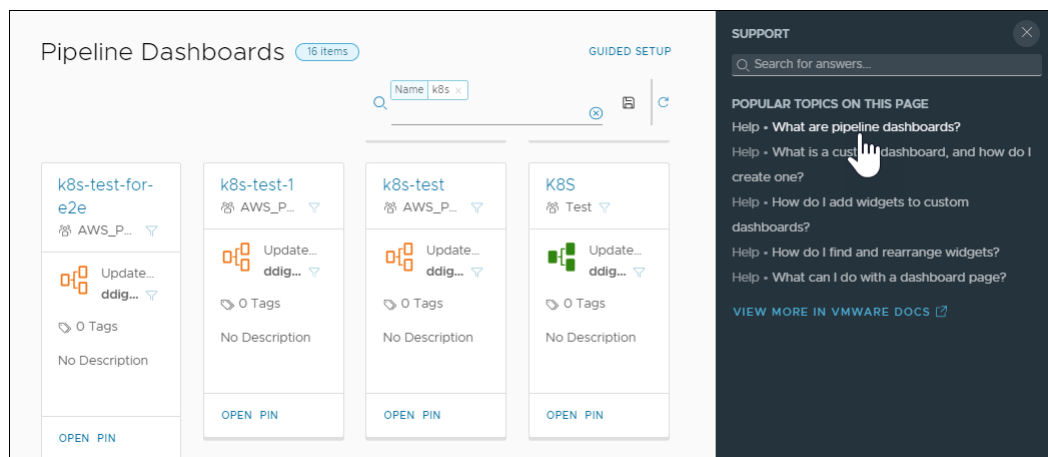
Quali sono gli altri obiettivi che è possibile raggiungere con Code Stream

5

Amministratori e sviluppatori di Code Stream possono creare pipeline complesse multiple che modellano e automatizzano i passaggi nel ciclo di vita delle versioni di DevOps.

Per informazioni su...	Vedere Utilizzo e gestione
Quali altre operazioni è possibile eseguire	Che cos'è e come funziona
Modellazione del processo di rilascio	Impostazione per modellare il processo di rilascio
Configurazione dell'area di lavoro della pipeline	Configurazione dell'area di lavoro della pipeline
Pianificazione di una build nativa CICD, CI o CD e creazione della pipeline	Pianificare nativamente la creazione, integrazione e consegna del codice
Casi d'uso e tutorial	Ad esempio: Come integrare in modo continuativo il codice dal repository GitHub o GitLab nella pipeline E molto altro.

Per ulteriori informazioni sulle funzioni di Code Stream, utilizzare la guida interna del prodotto nel pannello di supporto integrato in Code Stream.



Utilizzando la guida interna al prodotto è possibile conoscere maggiori informazioni su queste funzionalità e su molti altri aspetti:

- Comunicazione con gli endpoint

- Lavorare con risorse limitate e variabili
- Lavorare con i progetti
- Attivazione di una pipeline quando gli sviluppatori aggiornano il codice
- Traccia degli indicatori di prestazioni chiave per le pipeline in dashboard personalizzati

Per ulteriori informazioni sulle funzionalità di Code Stream, vedere [Utilizzo e gestione di VMware Code Stream](#).