Salesforce DevOps Center Developer Guide

Version 68.0, Winter ’27

Last updated: July 22, 2026

© Copyright 2000–2026 Salesforce, Inc. All rights reserved. Salesforce is a registered trademark of Salesforce, Inc., as are other names and marks. Other marks appearing herein may be trademarks of their respective owners.

CONTENTS

Chapter 1: DevOps Center Developer Guide 1

Release Notes 2
Understand the DevOps Center Data Model 2
- How DevOps Center Uses Asynchronous Operations 3
- How DevOps Center Keeps Track of User Changes 4
- How Promotions Work 6

DevOps Center Custom Objects 13
- Async Operation Result (sf_devops__Async_Operation_Result__c) 16
- Back Sync (sf_devops__Back_Sync__c) 18
- Branch (sf_devops__Branch__c) 22
- Change Bundle (sf_devops__Change_Bundle__c) 24
- Change Bundle Install (sf_devops__Change_Bundle_Install__c) 25
- Change Submission (sf_devops__Change_Submission__c) 27
- Deploy Component (sf_devops__Deploy_Component__c) 31
- Deployment Result (sf_devops__Deployment_Result__c) 33
- Environment (sf_devops__Environment__c) 36
- Hidden Remote Change (sf_devops__Hidden_Remote_Change__c) 40
- Merge Result (sf_devops__Merge_Result__c) 41
- Object Activity (sf_devops__Object_Activity__c) 43
- Pipeline (sf_devops__Pipeline__c) 47
- Pipeline Stage (sf_devops__Pipeline_Stage__c) 48
- Project (sf_devops__Project__c) 52
- Remote Change (sf_devops__Remote_Change__c) 54
- Repository (sf_devops__Repository__c) 57
- Source Member Reference (sf_devops__Source_Member_Reference__c) 59
- Submit Component (sf_devops__Submit_Component__c) 60
- VCS (sf_devops__Vcs__c) 62
- VCS Synch State (sf_devops__Vcs_Synch_State__c) 64
- Work Item (sf_devops__Work_Item__c) 65
- Work Item Promote (sf_devops__Work_Item_Promote__c) 72

DevOps Center Custom Field on the User Standard Object 75
DevOps Center Custom Platform Events 76
- Deployment (sf_devops__Deployment__e) 76
- Work Item Commit (sf_devops__Work_Item_Commit__e) 78
- Work Item Merged Change Request (sf_devops__Work_Item_Merged_Change_Request__e) 80
- Work Item Open Change Request (sf_devops__Work_Item_Open_Change_Request__e) 82
- Work Item State Change (sf_devops__Work_Item_State_Change__e) 84

CHAPTER 1 DevOps Center Developer Guide

In this chapter ...

- DevOps Center Release Notes
- Understand the DevOps Center Data Model
- DevOps Center Custom Objects
- DevOps Center Custom Field on the User Standard Object
- DevOps Center Custom Platform Events

This guide describes the DevOps Center object model and provides reference information for the custom objects.

Salesforce DevOps Center provides an improved experience around change and release management that brings DevOps best practices to your development team, regardless of where you fall on the low-code to pro-code spectrum. All developers can work together to deliver value to customers in a repeatable and scalable way, whether you use declarative builders, write code in Salesforce Extensions for VS Code, leverage the power and flexibility of Salesforce CLI, or all three.

SEE ALSO:

Salesforce Help: Manage and Release Changes Easily and Collaboratively with DevOps Center

Salesforce Help: Install and Configure DevOps Center

EDITIONS

Available in: Lightning Experience in **Enterprise**, **Performance**, **Professional**, **Unlimited**, and **Developer** Editions.

Inoperable in: **Government Cloud**. Although DevOps Center appears in the Setup menu, don't enable or install it because it won't operate properly.

Not available in: **EU Operating** Zone. EU Operating zone is a special paid offering that provides an enhanced level of data residency commitment. DevOps Center *is* supported in orgs in the EU that are not part of EU OZ, per standard product terms and conditions.

DevOps Center Release Notes

The DevOps Center team releases new features, product enhancements, and bug fixes on a regular basis.

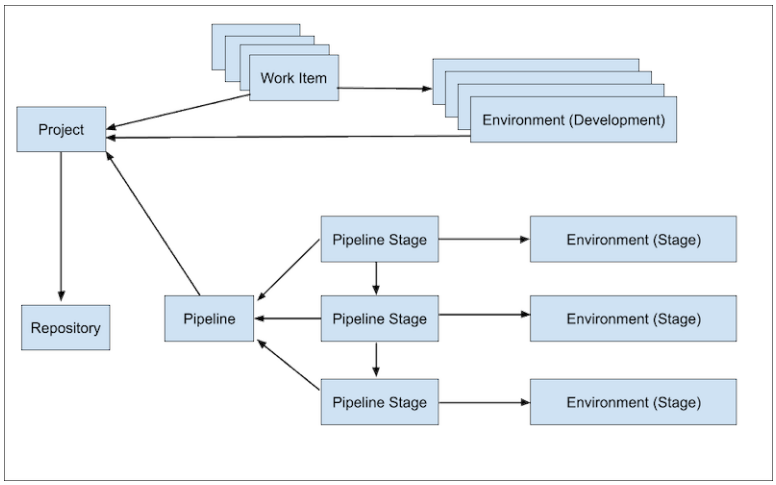
To read about the goodness in each package version, see DevOps Center Releases.

Understand the DevOps Center Data Model

The DevOps Center object model consists of custom objects that are created when you install the DevOps Center managed package in an org. The top object is Project, which is the direct or indirect parent of almost all other objects. The only exception is Repository, which one or more projects reference.

Note: For easier reading, this guide typically refers to objects and fields by their labels rather than their API names. For example, the guide refers to the object Async Operation Result rather than `sf_devops__Async_Operation_Result__c` and to its field Error Details rather than `sf_devops__Error_Details__c`. See the custom objects reference section on page 13, which lists the objects by label, with their API name in parentheses. Also, when referring to the object, the name is capitalized (Project). When referring to the DevOps Center feature that the object represents, the guide uses lower case (project).

This diagram shows many of the foundational objects in the DevOps Center object model; not all objects are shown.



These three objects have direct relationships to the Project object:

- Work Item—Represents a collection of metadata changes. When the user connects the work item to a development environment in DevOps Center (rather than developing and committing externally in the source control repository), the Work Item object references an Environment object. When the work item is promoted, the referenced environment is where the development work was previously done for that item. A single project can have multiple work items.
- Pipeline—Represents the release pipeline. A Pipeline object has one or more Pipeline Stage child objects that represent the stages of the pipeline, such as integration, UAT, and release. The pipeline stages are in a specific order, with the production stage always occurring last. Currently, a project can have only one pipeline.
- Environment—Represents an environment in which development work or testing of the promoted work is done. Projects, work items, and pipeline stages all have direct relationships with environments. Projects have multiple child environments, but work items and pipeline stages each reference only one environment. Currently, the only type of environment is a Salesforce org.

The DevOps Center object model uses validation rules to enforce that all references to work items, environments, pipeline stages, and so on, are contained within a single project. For example, when a work item is promoted to a new pipeline stage, the work item and

pipeline stage must both be part of the same project. You can't connect a work item in one project to an environment that's connected to a different project.

How DevOps Center Uses Asynchronous Operations
The custom objects that make up the DevOps Center object model live in the org in which you installed the DevOps Center managed package. But many of the operations that DevOps Center performs happen outside this org.

How DevOps Center Keeps Track of User Changes
When a DevOps Center user is connected to a development environment, it tracks the changes the user makes to the environment. This tracking allows DevOps Center to later commit the changes to the source control repository for review.

How Promotions Work
Show the objects involved when a user does an unbundled or bundled promotion.

How DevOps Center Uses Asynchronous Operations

The custom objects that make up the DevOps Center object model live in the org in which you installed the DevOps Center managed package. But many of the operations that DevOps Center performs happen outside this org.

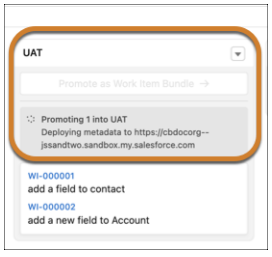
These operations include:

- Merging branches on your source control repository.
- Deploying metadata to the development and staging environments.
- Pulling metadata from a development environment.

DevOps Center delegates these operations to a Heroku application, which is created and configured as part of the initial DevOps Center installation.

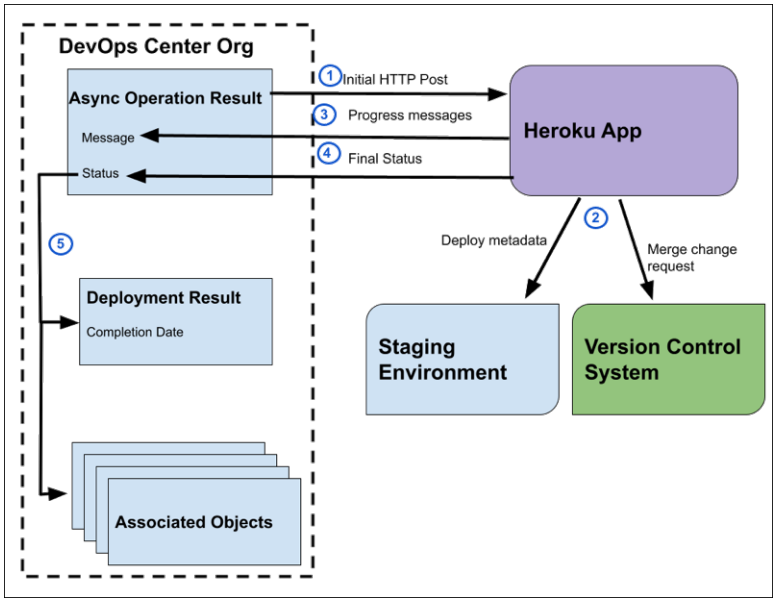
For example, when DevOps Center deploys metadata to a staging environment, it packages up the related data and sends an HTTP POST to the Heroku application. The Heroku application parses this request payload and performs the required task. However, Heroku applications can keep an HTTP connection open for a maximum of 30 seconds, and sometimes a metadata deployment can take longer. For this reason, DevOps Center runs these operations asynchronously.

DevOps Center manages these asynchronous operations by creating a record in the Async Operation Result custom object each time it interacts with the Heroku application. The request payload that DevOps Center sends to Heroku includes the ID of this new record; Heroku uses the ID to update the Messages field of the Async Operation Result record with its progress. DevOps Center posts these progress messages to the UI, such as this message about deploying metadata during a promotion:



When Heroku finishes the operation, it changes the Status field of Async Operation Result to either `Completed` or `Error`.

While Heroku is performing these operations, DevOps Center watches the Async Operation Result record for the change in status. When it happens, an Apex trigger updates other objects in the model to reflect the final state. For example, when the async operations associated with a promotion complete successfully, DevOps Center updates the Completion Date field of the Deployment Result. This update in turn lets the rest of the objects know that the promotion succeeded. Here's a diagram to show the flow.



When DevOps Center performs a remote operation, all objects that are part of the operation are associated with the Async Operation Result. This design serves two purposes:

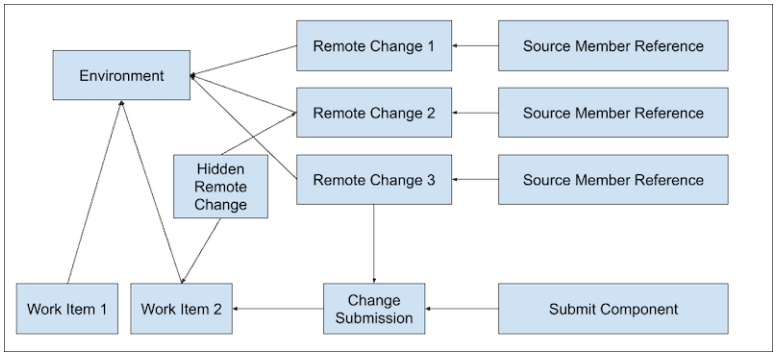
- When the remote operation completes, an Apex trigger can execute a single SOQL query to update all the associated objects to their final status, ensuring that the updates happen in a single transaction.
- Other potential remote operations can determine whether there's an operation already in progress, and wait for it to complete if necessary.

For example, let's say a user initiates a promotion from the UAT pipeline stage to Staging. The new Async Operation Result record sets a flag on the two associated Pipeline Stage records to indicate that they're associated with a remote operation (promotion) that's in progress. Then let's say a different user initiates a promotion from Integration to UAT. DevOps Center first checks whether either of these stages is already involved in a remote operation. Because the UAT stage is involved in the first operation, DevOps Center waits for it to complete before initiating a new operation.

How DevOps Center Keeps Track of User Changes

When a DevOps Center user is connected to a development environment, it tracks the changes the user makes to the environment. This tracking allows DevOps Center to later commit the changes to the source control repository for review.

This graphic shows the DevOps Center custom objects involved in tracking user changes. In the graphic, a single development environment has two work items. The org associated with the development environment has three metadata changes that DevOps Center is tracking.



Let's break this process down a bit more.

Org Source Tracking and Remote Changes

As we know, the only type of DevOps Center environment is a Salesforce org, such as a sandbox or a scratch org. When a user makes a change in their development environment, they're actually making a change to the metadata in their org. Examples of a change include creating an Apex class or a new custom field on a standard object. To track these changes, DevOps Center uses a Salesforce feature called "source tracking", which in turn depends on the Tooling API object Source Member. While many metadata types support source tracking, not all of them do; check the Metadata Coverage Report for details. For metadata types that support source tracking, Source Member records contain information about each piece of metadata in the org that has changed. Each record includes how the metadata was changed (add, change, remove), who changed it, and when it was changed.

But DevOps Center doesn't use the Source Member Tooling API object directly. Instead, DevOps Center mirrors this information from Source Member in the custom object Source Member Reference.

When a user makes a metadata change in their development environment, DevOps Center creates a Source Member Reference record, which is then paired with a new Remote Change record. The Remote Change object represents an accumulation of operations on a particular piece of metadata that will be committed to the remote source control repository at some point. What does this mean? Here are a few examples.

- Let's say a source member reference includes a new piece of metadata (add), and then multiple changes to the same metadata. The associated remote change is still an add operation, regardless of the number of subsequent changes, because when the user commits the changes, the source control repository performs an add operation.
- If a source member reference first includes a change to a piece of metadata and then removes it, the remote change is simply a remove operation.

In the graphic on page 4, remote changes are linked to environments and not work items. The reason is that multiple work items can be associated with the same environment. As a result, all work items associated with the same environment always show the same set of files that can be committed to the source control repository. So how does DevOps Center determine the list of available changes in a work item? Read on!

Determine the List of Available Changes in a Work Item

After a DevOps Center user makes and tests changes in their development environment, they pull them into their work item. This action allows the user to see the changes before committing them to the source control repository. In the graphic, Remote Change 1, 2, and 3 represent these changes.

When determining the list of changes to display in the work item, DevOps Center starts with the list of files from all the Remote Changes, but then modifies the list based on these circumstances:

- **The `.forceignore` file on the work item's associated feature branch changes.** This change can mean that some of the remote changes aren't visible in the associated work item anymore.

  But a change to the `.forceignore` file in one feature branch doesn't affect the list of available changes in a work item associated with a different feature branch. DevOps Center manages this scenario with the Hidden Remote Change object.

  In the graphic, Work Item 1 can see Remote Change 2, but Work Item 2 can't see it because it's hidden. This scenario implies that the feature branch associated with Work Item 2 has a `.forceignore` file that excludes the metadata represented by Remote Change 2.

- **A user commits one or more files to the feature branch associated with a work item.** When a user performs this action, DevOps Center retrieves the metadata from the org and commits it to the source control repository. DevOps Center then creates a Change Submission record to model the commit and associates all the remote changes that were part of the commit to it. Because the metadata change is now in the source control repository, no other work item associated with the environment can commit this

particular change. So DevOps Center excludes all the remote changes associated with a change submission from the list of available changes.

In the graphic, Remote Change 3 isn't listed in either Work Item 1 or 2 because it's been committed to the source control repository.

Determine the Components to Submit

DevOps Center uses the Submit Component custom object to track all the changes that have been committed to a feature branch associated with a work item. These changes can come from two places:

- When a user commits using the DevOps Center UI, DevOps Center abstracts all of the relevant metadata from these remote commits into records of the Submit Component object.
- DevOps Center gets notified every time a user commits to a feature branch outside of the UI, such as using VS Code. In this case DevOps Center creates Submit Component records to track these commits.

In sum, regardless of how metadata changes are committed to a feature branch, the Submit Component object represents them. DevOps Center uses these objects when the work item is eventually promoted.

What Happens When a Work Item Status Is Changed to Never

If the user changes the status of a work item to Never, DevOps Center deletes its associated Change Submission and Submit Component records. These deletions can have ripple effects in what's displayed in the UI.

For example, let's look at this graphic again. If a user sets the status of Work Item 2 to Never, then DevOps Center deletes its associated Change Submission record. DevOps Center then returns the changes listed in Remote Change 3 to the available changes list and resets the source tracking for the changes. Work Item 1 now displays these changes and the user can recommit them from this work item.

How Promotions Work

Show the objects involved when a user does an unbundled or bundled promotion.

DevOps Center supports two types of promotions:

- Unbundled promotion—Users select one or more work items and promote (move) them to the next pipeline stage.
- Versioned promotion—Work items are combined into a single bundle represented by the Change Bundle object. The bundle is promoted as a single unit to the next pipeline stage. Change bundles have version identifiers, such as 1.0.

The Pipeline Stage object has a Boolean field called Versioned that DevOps Center uses to control how promotions come into a pipeline stage. The value of the field determines what type of pipeline stage it is:

- Unbundled Stage—A stage that work items are promoted into by users selecting them individually.
- Bundling Stage—The last unbundled stage in a pipeline. Work items have been promoted individually into this stage, but new change bundles are promoted out of it.
- Versioned Stage—A stage where change bundles are promoted into.

When an admin creates a pipeline, DevOps Center provides a template of a typical release pipeline that includes a bundling stage. However, you have complete flexibility to configure it based on your preferences. This setup allows users to individually select the work items they want to promote from the Approved Work Items list to the first pipeline stage. Admins can configure subsequent stages as either unbundled or versioned. But if an admin makes a stage versioned, all the stages to the right in the pipeline must also be versioned, and they can't go back to being unbundled.

Common Promotion Custom Objects
Before we drill down into the details of promotions, let's first review the common custom objects that both unbundled and bundled promotions share.

Unbundled Promotions: A Deeper Look
Now that we know the main custom objects involved in an unbundled promotion, let's step through the different phases of the promotion and check the status of these objects at each phase.

Common Promotion Custom Objects

Before we drill down into the details of promotions, let's first review the common custom objects that both unbundled and bundled promotions share.

Deployment Result Custom Object

The Deployment Result object tracks the requests and results of a metadata deployment to the environment associated with the pipeline stage to which work items or change bundles are being promoted. Currently these environments are always Salesforce orgs.

When a user initiates a promotion, DevOps Center creates a Deployment Result record with all the request properties. These properties include whether the deployment is full or partial and the set of metadata components from all associated work items to be deployed. Each metadata component is stored as a record of the Deploy Component object, which is a child of Deployment Result.

DevOps Center computes the set of metadata components to be deployed like this:

- For unbundled promotions, the list comes from all of the Submit Components records associated with all the work items that the user selected to promote.
- For versioned promotions, the list comes from the Deploy Components of the previous versioned promotion.

When the promotion completes, the Heroku application writes result properties to the same Deployment Result record, such as the deployment ID, completion date, and number of tests completed.

Merge Result Custom Object

The Merge Result object tracks the request and the results of merging branches in the source control repository.

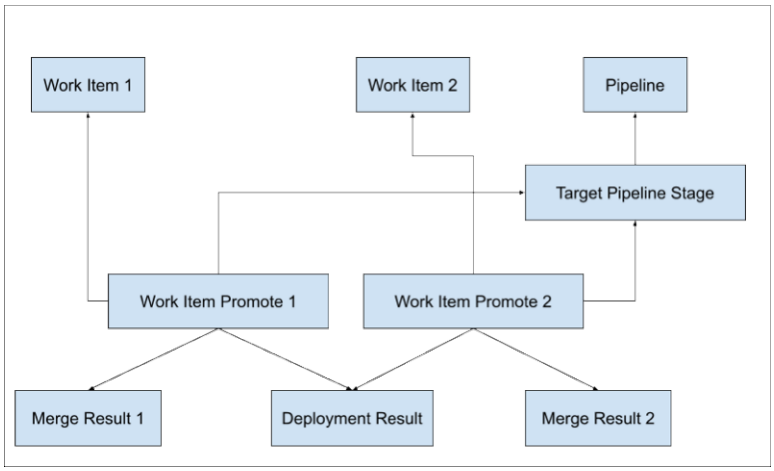
A promotion doesn't always require a merge, such as when a user synchronizes their development environment from their source control repository. Sometimes a promotion requires more than one merge, such as when a user does an unbundled promotion of multiple work items.

When a user initiates a promotion, DevOps Center creates a Merge Result record for every merge that is required. When the promotion completes, the Heroku application writes the merge IDs (or SHA as it's called in GitHub) to the same Merge Result records.

Unbundled Promotions: A Deeper Look

Now that we know the main custom objects involved in an unbundled promotion, let's step through the different phases of the promotion and check the status of these objects at each phase.

Here's a graphic that shows an unbundled promotion of two work items and how the associated objects are connected.



When a user initiates an unbundled promotion, DevOps Center creates a record of Work Item Promote for every work item in the promotion. Each work item is still associated with a feature branch in the user's source control repository, so each of these branches must be merged into the branch associated with the target stage environment. This relationship means that every Work Item Promote record has an associated Merge Result record. But there's only one metadata deployment for the entire promotion (to the associated stage environment), so each Work Item Promote record points to a shared record of Deployment Result.

Let's start! For our example, assume a user promotes a single work item (WI-000003) from the Integration stage to UAT. In this example, the promotion is initiated in the DevOps Center UI.

State Before a Promotion

Here's WI-000003 in DevOps Center, ready to be promoted.

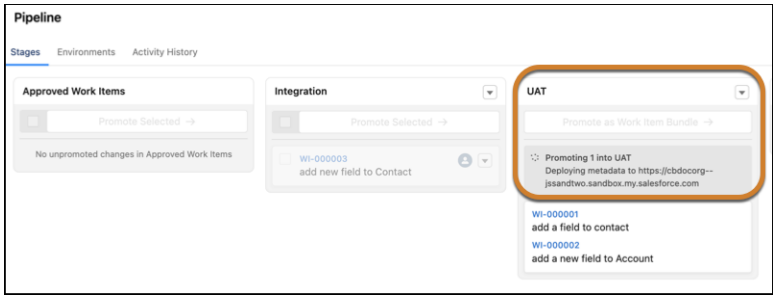
This table lists the state of the existing custom object records associated with the unbundled promotion before it begins. Not all objects and fields are listed, only the ones that are interesting and relevant.

Custom Object	Relevant Field	Value	Notes
Work Item	Status	NULL	See How DevOps Center Uses Asynchronous Operations for more information.
	Promoted	TRUE	
	Review Remote Reference	Unique ID of the change request between this work item's feature branch and the target stage.	

Custom Object	Relevant Field	Value	Notes
Pipeline Stage	Status	NULL	See How DevOps Center Uses Asynchronous Operations for more information.
Work Item Promote DevOps Center created this record when the work item was previously promoted from the Approved Work Item stage to Integration.	Work Item	WI-000003	
	Pipeline Stage	Integration	
	Status	A reference to an Async Operation Result record in the Completed status.	

State During a Promotion

The user now promotes the work item from the Integration stage to the UAT stage by selecting the item and clicking **Promote Selected**. Here's the promotion in progress:



Let's further split this phase into two subphases:

- Immediately after a user clicks **Promote Selected**.
- The moment when the Heroku application takes over.

Immediately after the user clicks **Promote Selected**, DevOps Center queues the promotion and updates the object model, as shown in this table. The **New or Existing?** column specifies whether DevOps Center creates a record of the object in this promotion phase or if a record exists.

Custom Object	New or Existing?	Relevant Field	Value	Notes
Async Operation Result	New	Status	In Progress	
		Operation	AD_HOC_PROMOTE	
		Message		The asynchronous process regularly updates this field with the current status.
Work Item	Existing	Status	ID of the new Async Operation Result record.	
Pipeline Stage	Existing	Status	ID of the new Async Operation Result record.	

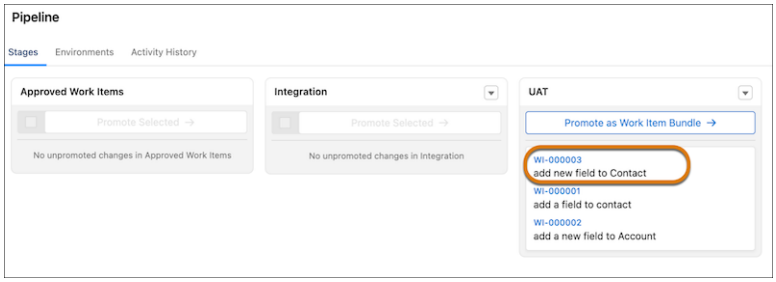
Custom Object	New or Existing?	Relevant Field	Value	Notes
Work Item Promote This new record represents the current promotion from the Integration stage to UAT.	New	Work Item	WI-000003	
		Pipeline Stage	UAT	
		Status	ID of the new Async Operation Result record.	
Deployment Result	New	n/a		
Merge Result	New	Source Branch	The feature branch associated with the work item.	
		Target Branch	The branch associated with the UAT stage.	

DevOps Center then fires off a request to Heroku to perform the async processing, such as deploying metadata to the appropriate environment and merging branches. Here's what the object model looks like when Heroku takes over.

Custom Object	Relevant Field	Value (Changed by Heroku)	Notes
Async Operation Result	Message	Status messages based on the state of the promotion.	
Deployment Result	Deployment Id	The metadata deployment ID for this promotion.	DevOps Center sets this value as soon as the user initiates the promotion.
Merge Result	Remote Reference	Merge ID (called a SHA in GitHub) of the merge commit from the feature branch to the target stage's branch.	
	Previous Remote Reference	The SHA of the target stage's branch before this work item was merged.	

State When a Promotion Succeeds

Hurray, the promotion succeeded, and our work item has been promoted to the UAT stage!



Here's the final state of the objects after a successful promotion.

Note: After a successful promotion, DevOps Center also performs a rebase of the feature branch associated with the work item. This table doesn't go into detail about rebasing; we plan to provide this information in the future. After a successful promotion, DevOps Center often performs other operations, such as rebasing or deleting feature branches. These details aren't discussed in this section; stay tuned for more information at a later date.

Custom Object	Relevant Field	Value	Notes
Async Operation Result	Status	Completed	
	Message	Promotion Completed	
Deployment Result	Completion Date	The date and time when the deployment completed.	
Merge Result	Merge Date	The date and time when the merge was completed.	
Work Item	Status	NULL	
	Branch	Updated with start and end SHAs of the branch after it has been rebased.	
	Rebase Branch	Updated with the start and end SHAs of the feature branch after the initial promotion into the pipeline	This is only created after the initial promotion.
Pipeline Stage	Status	NULL	

State When a Promotion Fails

Sadly, promotions occasionally encounter an error and fail. Let's see what the state of the custom objects look like in this case

Custom Object	Relevant Field	Value
Async Operation Result	Status	ERROR
	Message	Error message from the async processing.
	Error Details	If available, a stack trace or other detailed error information.
Deployment Result	Deployment Id	NULL
	Completion Date	NULL
Merge Result	Merge Date	NULL
	Remote Reference	NULL
	Previous Remote Reference	NULL
Work Item	Status	NULL

Custom Object	Relevant Field	Value
Pipeline Stage	Status	NULL

Variation: Multiple Work Items

In the previous unbundled promotion flow, only one work item was promoted. But you can also perform an unbundled promotion of multiple work items at the same time. The resulting flow is similar to the single work item flow, but with these differences:

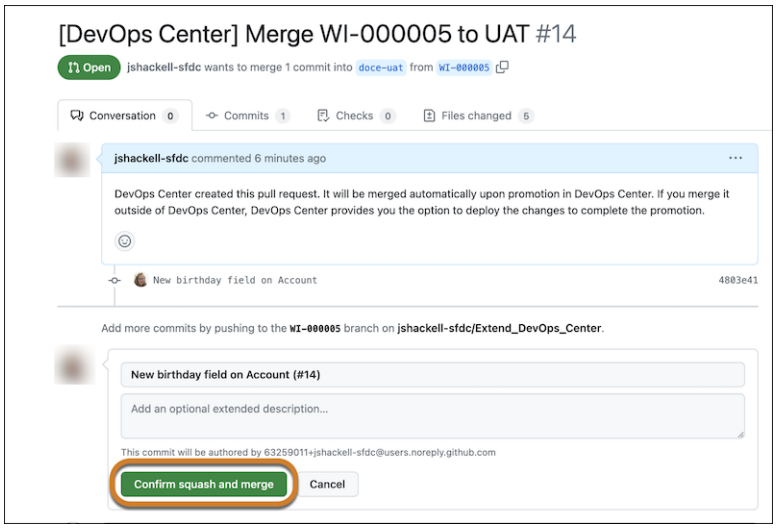
- For each Work Item record, DevOps Center creates a unique Work Item Promote record and a unique Merge Result record.
- But all Work Item Promotes records that are being promoted together share a single Deployment Result and Async Operation Result.

Variation: Externally Merged Change Request

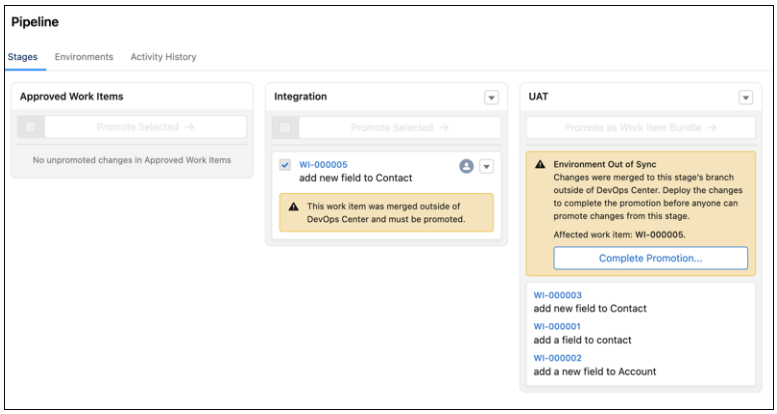
In the previous unbundled promotion flow, DevOps Center merged the change request in the source control repository. But a developer can also merge a change request externally, such as through the GitHub UI. The resulting flow is similar to the non-external merge flow, but with a few key differences.

The beginning state of the custom objects, before the promotion, is the same as described in this section on page 8. But then the user decides to externally merge the change (pull) request in the source control repository.

This screenshot shows the user merging changes from the GitHub feature branch for WI-000005 to the `doce-uat` branch (which is associated with the UAT pipeline stage) using the recommended Squash and Merge option. DevOps Center created this pull request in GitHub when WI-000005 was promoted to the Integration stage.



At this point, the changes are in a partially promoted state because they've been merged but not yet deployed. DevOps Center alerts the user both in the work item itself and in the pipeline:



This table shows the state of the object model after the external merge but before the user clicks **Complete Promotion** in DevOps Center. Both records are new.

Custom Object	Relevant Field	Value
Work Item Promote This new record represents the current undeployed promotion from the Integration stage to UAT.	Work Item	WI-000005
	Pipeline Stage	UAT
Merge Result	Source Branch	The feature branch associated with the work item.
	Target Branch	The branch associated with the UAT stage.
	Remote Reference	Merge ID (called a SHA in GitHub) of the merge commit from the feature branch to the target stage's branch.
	Previous Remote Reference	ID of the commit in the target stage's branch before the merge happened.
	Remote Reference Date	The date the external merge occurred.

After the user clicks Complete Promotion, DevOps Center continues processing as described in State During a Promotion on page 9, except that some records exist, such as Work Item Promote and Merge Result. The rest of the promotion is the same.

DevOps Center Custom Objects

The DevOps Center data model includes several custom objects.

Async Operation Result (sf_devops__Async_Operation_Result__c)
Represents the communication bridge between the Heroku app and DevOps Center. DevOps Center creates an instance of Async Operation Result when it delegates certain asynchronous operations to the Heroku app. These operations include merging branches in the source control repository (also called version control system, or VCS) and deploying metadata to environments. This object is available in orgs that have DevOps Center installed.

Back Sync (sf_devops__Back_Sync__c)

Represents the synchronization between a DevOps Center user's development environment and the first pipeline stage's branch. In particular, this object tracks when the synchronization happened and the records of the Source Member Reference object that the synchronization can ignore. This object is available in all orgs that have DevOps Center installed.

Branch (sf_devops__Branch__c)

Stores the state of a branch in the source control repository (also version control system, or VCS). In the DevOps Center object model, this object is a child of the Repository where the branch lives. Work Item and Pipeline Stage records refer to this object. This object is available in all orgs that have DevOps Center installed.

Change Bundle (sf_devops__Change_Bundle__c)

A collection of Work Item records that are promoted as a single unit to the next pipeline stage. This collection helps ensure a consistent merge and deployment of the metadata as it moves through the release pipeline. This object is available in all orgs that have DevOps Center installed.

Change Bundle Install (sf_devops__Change_Bundle_Install__c)

Represents the deployment of the metadata components associated with a change bundle into an environment. This object is available in all orgs that have DevOps Center installed.

Change Submission (sf_devops__Change_Submission__c)

Represents a change that was submitted (committed) to the work item feature branch. The change includes relevant metadata files. This object is available in all orgs that have DevOps Center installed.

Deploy Component (sf_devops__Deploy_Component__c)

Stores the aggregated set of metadata components that must be deployed as part of a promotion. Includes the metadata components of all the work items associated with the promotion. The Deploy Component object is a child of Deployment Result. This object is available in all orgs that have DevOps Center installed.

Deployment Result (sf_devops__Deployment_Result__c)

Contains information from DevOps Center to the Heroku application about how to execute a metadata deployment to an environment, such as the Apex tests and test level. After the deployment completes, this object then stores information about the deployment, such as the deployment ID and completion date. See How Promotions Work for more information. This object is available in all orgs that have DevOps Center installed.

Environment (sf_devops__Environment__c)

Represents a connection from DevOps Center to an environment, which currently can be only a Salesforce org. Developers use development environments to do their work. Each pipeline stage has an associated environment. This object is available in all orgs that have DevOps Center installed.

Hidden Remote Change (sf_devops__Hidden_Remote_Change__c)

Used to hide a Remote Change record from a work item. A sample use case is when a feature branch associated with the work item has a `.forceignore` file that excludes the metadata represented by the Remote Change record. See How DevOps Center Keeps Track of User Changes for more information. This object is available in all orgs that have DevOps Center installed.

Merge Result (sf_devops__Merge_Result__c)

Contains information from DevOps Center to the Heroku application about the source control branch to merge as part of a promotion. When the merge completes, this object then stores information about the merge, such as the ID of the merge and when it happened. This object is available in all orgs that have DevOps Center installed.

Object Activity (sf_devops__Object_Activity__c)

Represents an activity by one of the DevOps Center custom objects. Object Activity records determine the items that are listed in the Activity Histories of work items and pipelines. When DevOps Center performs an operation, it creates an activity record and inserts it in the appropriate user interface view; each activity references the associated custom object. For example, when a user promotes a work item, DevOps Center inserts a promotion initiation activity in the work item Activity History; the activity references the Work Item and Pipeline Stage objects. When the promotion terminates, DevOps Center inserts a second activity that references the same two objects and Async Operation Result. This object is available in all orgs that have DevOps Center installed.

Pipeline (sf_devops__Pipeline__c)

Represents a collection of Pipeline Stage records that together make up the DevOps Center release pipeline in a project. This object is available in all orgs that have DevOps Center installed.

Pipeline Stage (sf_devops__Pipeline_Stage__c)

Represents a connection from a pipeline to an environment; the collection of all pipeline stages in a project make up the release pipeline. This object is available in all orgs that have DevOps Center installed.

Project (sf_devops__Project__c)

Represents the parent of all DevOps Center custom objects. See Understand the DevOps Center Data Model for more information. This object is available in all orgs that have DevOps Center installed.

Remote Change (sf_devops__Remote_Change__c)

Represents a change to an environment that's connected to DevOps Center. In particular, a Remote Change record represents an accumulation of operations on a particular piece of metadata. DevOps Center presents this change to the user so they can pull it into their work item and commit it to the associated feature branch. See How DevOps Center Keeps Track of User Changes for more information. This object is available in all orgs that have DevOps Center installed.

Repository (sf_devops__Repository__c)

Represents a specific location in a source control system where the metadata for a project is stored. Multiple projects can reference the same repository. This object is available in all orgs that have DevOps Center installed.

Source Member Reference (sf_devops__Source_Member_Reference__c)

Represents a copy of the relevant information from a SourceMember Tooling API record in a development environment. DevOps Center copies this data to track the metadata changes that a user hasn't yet pulled into their work item. Copying the data also makes computing Remote Change records more efficient. See How DevOps Center Keeps Track of User Changes for more information. This object is available in all orgs that have DevOps Center installed.

Submit Component (sf_devops__Submit_Component__c)

Represents a metadata component that was committed to a feature branch in the source control repository. The commit can be initiated in one of two ways, either from within DevOps Center or directly in the source control repository. Each metadata component that's part of the commit must also be deployed to an environment, and DevOps Center uses the Submit Component object to model the metadata. The Submit Component object is a child of Change Submission. This object is available in all orgs that have DevOps Center installed.

VCS (sf_devops__Vcs__c)

Represents a supported source (version) control system. This object is available in all orgs that have DevOps Center package version 8.2 and later. Available in API version 62.0 and later.

VCS Synch State (sf_devops__Vcs_Synch_State__c)

Represents the synchronization state between DevOps Center and the source (version) control system. DevOps Center uses this object to track all synchronization events to ensure that DevOps Center is working with the latest version of the code in the source control repository. This object is available in all orgs that have DevOps Center package version 8.2 and later. Available in API version 62.0 and later.

Work Item (sf_devops__Work_Item__c)

Represents a collection of metadata changes in a project. A work item can be associated with an environment in which the work is performed. If it's not connected to an environment, the VCS Event object handles the changes. A work item goes through a number of development lifecycle stages until all development work is complete and the work item is part of the release pipeline. This object is available in all orgs that have DevOps Center installed.

Work Item Promote (sf_devops__Work_Item_Promote__c)

Represents the unbundled promotion of a work item to the next stage in a pipeline. See Unbundled Promotions: A Deeper Look for more information. This object is available in all orgs that have DevOps Center installed.

Async Operation Result (sf_devops__Async_Operation_Result__c)

Represents the communication bridge between the Heroku app and DevOps Center. DevOps Center creates an instance of Async Operation Result when it delegates certain asynchronous operations to the Heroku app. These operations include merging branches in the source control repository (also called version control system, or VCS) and deploying metadata to environments. This object is available in orgs that have DevOps Center installed.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
`Name`	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** Name of the Async Operation Result record.
`OwnerId`	**Type** reference **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** ID of the user who owns this record. This field is a polymorphic relationship field. **Relationship Name** Owner **Relationship Type** Lookup

Field	Details
	Refers To Group, User
`sf_devops__Dependent_Records__c`	**Type** textarea **Properties** Create, Nillable, Update **Description** JSON representation of all the related object records that must be updated as a result of this asynchronous operation.
`sf_devops__Error_Details__c`	**Type** textarea **Properties** Create, Nillable, Update **Description** If the remote operation encounters an error, this field contains the stack trace from the Heroku app.
`sf_devops__In_Terminal_State__c`	**Type** boolean **Properties** Defaulted on create, Filter, Group, Sort **Description** If `true`, indicates that this async operation is in a terminal state. If `false`, the operation is in progress. This field is a calculated field.
`sf_devops__Log_Id__c`	**Type** string **Properties** Create, Filter, Group, Sort, Update **Description** Unique ID for this remote operation. Used internally by Salesforce to tie together various logging systems when investigating customer cases.
`sf_devops__Message__c`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Status message that's displayed in DevOps Center as the remote operation progresses.

Field	Details
`sf_devops__Operation__c`	**Type** string **Properties** Create, Filter, Group, Sort, Update **Description** Type of remote operation that this record is being used in.
`sf_devops__Remote_Commands__c`	**Type** textarea **Properties** Create, Nillable, Update **Description** List of remote commands that were executed as part of this remote operation. Includes `git` and `sf` (Salesforce CLI) commands.
`sf_devops__Status__c`	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Status of the remote operation. Possible values are: • `Completed` • `Error` • `Ignored` • `In Progress` The default value is `In Progress`.

SEE ALSO:

How DevOps Center Uses Asynchronous Operations

Back Sync (sf_devops__Back_Sync__c)

Represents the synchronization between a DevOps Center user's development environment and the first pipeline stage's branch. In particular, this object tracks when the synchronization happened and the records of the Source Member Reference object that the synchronization can ignore. This object is available in all orgs that have DevOps Center installed.

Let's set up an example to see how this works. Assume that:

- The user has already pulled two metadata changes into their development environment, represented by two Remote Change records. The Revision Counter fields for these two metadata files in the Source Member Reference object are 5 and 6.
- There are two unpulled metadata changes in the Source Member Reference object, with revision counters 7 and 8.

- The user clicks **Sync** in their Pipeline Environment. DevOps Center sets the Start Revision Counter field of this Back Sync record to 8.
- The synchronization generates 3 new rows in the Source Member Reference table with Revision Counter values of 9, 10, and 11.
- The synchronization completes, and DevOps Center sets the End Revision Counter field of this Back Sync record to 11.
- The user makes two more metadata changes in their development environment: the corresponding with Revision Counter values in the Source Member Reference object are 12 and 13.

The next time the user clicks **Pull Changes** from within their work item, DevOps centers pulls metadata changes that correspond to Source Member References with revision counters 7 through 13.

DevOps Center then must convert relevant Source Member References into Remote Change records. It first checks other Back Sync records associated with this development environment. In our example, DevOps Center ignores the revision counters 9, 10, and 11 and creates Remote Change records for only revision counters 7, 8, 12, and 13.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
`Name`	**Type** string **Properties** Create, Defaulted on create, Filter, Group, idLookup, Nillable, Sort, Update **Description** Name of the Back Sync record.
`OwnerId`	**Type** reference **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** ID of the user who owns this record. This field is a polymorphic relationship field. **Relationship Name** Owner **Relationship Type** Lookup **Refers To** Group, User
`sf_devops__Deployment_Result__c`	**Type** reference

Field	Details
	Properties Create, Filter, Group, Sort, Update **Description** A reference to the Deployment Result record that was sent to the Heroku application to perform the metadata deployment to the development environment. This field is a relationship field. **Relationship Name** sf_devops__Deployment_Result__r **Relationship Type** Lookup **Refers To** sf_devops__Deployment_Result__c
sf_devops__Destination_Environment__c	**Type** reference **Properties** Create, Filter, Group, Sort, Update **Description** A reference to the development environment that was synchronized. This field is a relationship field. **Relationship Name** sf_devops__Destination_Environment__r **Relationship Type** Lookup **Refers To** sf_devops__Environment__c
sf_devops__End_Revision_Counter__c	**Type** double **Properties** Create, Filter, Nillable, Sort, Update **Description** The Source Member Reference revision counter in the development environment after the synchronization completes.
sf_devops__Operation_Result__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** A reference to the Async Operation Result that was used in this remote operation.

Field	Details
	This field is a relationship field. **Relationship Name** sf_devops__Operation_Result__r **Relationship Type** Lookup **Refers To** sf_devops__Async_Operation_Result__c
sf_devops__Source_Pipeline_Stage__c	**Type** reference **Properties** Create, Filter, Group, Sort, Update **Description** A reference to the Pipeline Stage whose branch was used to deploy metadata from. This stage is always the first one in the pipeline. This field is a relationship field. **Relationship Name** sf_devops__Source_Pipeline_Stage__r **Relationship Type** Lookup **Refers To** sf_devops__Pipeline_Stage__c
sf_devops__Start_Revision_Counter__c	**Type** double **Properties** Create, Filter, Sort, Update **Description** The Source Member Reference revision counter in the development environment before the synchronization was initiated.

SEE ALSO:

- *Salesforce Help*: Synchronize Your Development Environment
- Async Operation Result (sf_devops__Async_Operation_Result__c)
- Source Member Reference (sf_devops__Source_Member_Reference__c)
- Environment (sf_devops__Environment__c)
- Pipeline Stage (sf_devops__Pipeline_Stage__c)
- Deployment Result (sf_devops__Deployment_Result__c)

Branch (sf_devops__Branch__c)

Stores the state of a branch in the source control repository (also version control system, or VCS). In the DevOps Center object model, this object is a child of the Repository where the branch lives. Work Item and Pipeline Stage records refer to this object. This object is available in all orgs that have DevOps Center installed.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
`Name`	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** Name of the Branch record.
`OwnerId`	**Type** reference **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** ID of the user who owns this record. This field is a polymorphic relationship field. **Relationship Name** Owner **Relationship Type** Lookup **Refers To** Group, User
`sf_devops__Ignore_Rules__c`	**Type** textarea **Properties** Create, Nillable, Update **Description** Serialized JSON representation of the preprocessed contents of the `.forceignore` file on the branch.

Field	Details
sf_devops__Name__c	**Type** string **Properties** Create, Filter, Group, Sort, Update **Description** Name of the branch.
sf_devops__Parent_Remote_Reference__c	**Type** string **Properties** Create, Filter, Group, Sort, Update **Description** Unique ID that represents where this branch diverged from its parent branch. In GitHub, this ID is called a SHA. DevOps Center uses this ID to recreate the branch if necessary. For example, if a user deletes the branch on GitHub, but DevOps Center isn't finished using the branch, then DevOps Center can recreate it using this ID.
sf_devops__Remote_Reference_Date__c	**Type** dateTime **Properties** Create, Filter, Nillable, Sort, Update **Description** Last date and time when the branch identified by the Remote Reference field was updated.
sf_devops__Remote_Reference__c	**Type** string **Properties** Create, Filter, Group, Sort, Update **Description** Unique ID of the tip of this branch. In GitHub, this ID is called the HEAD SHA.
sf_devops__Repository__c	**Type** reference **Properties** Create, Filter, Group, Sort, Update **Description** A reference to the source code repository in which this branch lives. This field is a relationship field. **Relationship Name** sf_devops__Repository__r

Field	Details
	Relationship Type Lookup **Refers To** sf_devops__Repository__c

SEE ALSO:

- Repository (sf_devops__Repository__c)
- Work Item (sf_devops__Work_Item__c)
- Pipeline Stage (sf_devops__Pipeline_Stage__c)

Change Bundle (sf_devops__Change_Bundle__c)

A collection of Work Item records that are promoted as a single unit to the next pipeline stage. This collection helps ensure a consistent merge and deployment of the metadata as it moves through the release pipeline. This object is available in all orgs that have DevOps Center installed.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
`Name`	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** Name of the Change Bundle record.
`sf_devops__Project__c`	**Type** reference **Properties** Create, Filter, Group, Sort **Description** A reference to the project in which this change bundle lives. This field is a relationship field. **Relationship Name** sf_devops__Project__r

Field	Details
	Relationship Type Master-detail **Refers To** sf_devops__Project__c
`sf_devops__Version_Name__c`	**Type** string **Properties** Create, Filter, Group, Sort, Update **Description** Unique name of this bundle within the project.

SEE ALSO:

Project (sf_devops__Project__c)

Change Bundle Install (sf_devops__Change_Bundle_Install__c)

Represents the deployment of the metadata components associated with a change bundle into an environment. This object is available in all orgs that have DevOps Center installed.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
`Name`	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** Name of this Change Bundle Install record.
`sf_devops__Change_Bundle__c`	**Type** reference **Properties** Create, Filter, Group, Sort

Field	Details
	Description Reference to the Change Bundle record that's being deployed. This field is a relationship field. **Relationship Name** sf_devops__Change_Bundle__r **Relationship Type** Master-detail **Refers To** sf_devops__Change_Bundle__c
sf_devops__Deployment_Result__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Reference to the Deployment Result record that controls the metadata deployment associated with this record. This field is a relationship field. **Relationship Name** sf_devops__Deployment_Result__r **Relationship Type** Lookup **Refers To** sf_devops__Deployment_Result__c
sf_devops__Environment__c	**Type** reference **Properties** Create, Filter, Group, Sort, Update **Description** Reference to the environment in which the change bundle is being deployed. This field is a relationship field. **Relationship Name** sf_devops__Environment__r **Relationship Type** Lookup **Refers To** sf_devops__Environment__c
sf_devops__Merge_Result__c	**Type** reference

Field	Details
	Properties Create, Filter, Group, Sort, Update **Description** Reference to the Merge Result record that lists the source control branches that were merged as part of this deployment. This field is a relationship field. **Relationship Name** sf_devops__Merge_Result__r **Relationship Type** Lookup **Refers To** sf_devops__Merge_Result__c
`sf_devops__Status__c`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Reference to the Async Operation Result record that's associated with this record. This field is a relationship field. **Relationship Name** sf_devops__Status__r **Relationship Type** Lookup **Refers To** sf_devops__Async_Operation_Result__c

SEE ALSO:

- Change Bundle (sf_devops__Change_Bundle__c)
- Deployment Result (sf_devops__Deployment_Result__c)
- Environment (sf_devops__Environment__c)
- Merge Result (sf_devops__Merge_Result__c)
- Async Operation Result (sf_devops__Async_Operation_Result__c)

Change Submission (sf_devops__Change_Submission__c)

Represents a change that was submitted (committed) to the work item feature branch. The change includes relevant metadata files. This object is available in all orgs that have DevOps Center installed.

The commit can occur in one of these ways:

- A user clicked **Commit Changes** in a work item within DevOps Center.
- A user committed the change outside of DevOps Center, such as through the GitHub UI, and DevOps Center detects the event.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
`Name`	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** Name of this Change Submission record.
`sf_devops__Comment__c`	**Type** textarea **Properties** Create, Nillable, Update **Description** The commit message that the user entered.
`sf_devops__Creation_Result__c`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** When the commit comes from a user using DevOps Center, this field references the Async Operation Result record that was used for the commit. This field is a relationship field. **Relationship Name** sf_devops__Creation_Result__r **Relationship Type** Lookup **Refers To** sf_devops__Async_Operation_Result__c
`sf_devops__Inspection_Result__c`	**Type** reference

Field	Details
	Properties Create, Filter, Group, Nillable, Sort, Update **Description** When the commit comes from outside DevOps Center, this field references the Async Operation Result record that was used when DevOps Center detected the commit. DevOps Center must inspect the commit to determine the relevant metadata components. This field is a relationship field. **Relationship Name** sf_devops__Inspection_Result__r **Relationship Type** Lookup **Refers To** sf_devops__Async_Operation_Result__c
sf_devops__Remote_Reference__c	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Unique ID of the feature branch before the commit. In GitHub, this ID is called a SHA. DevOps Center uses this ID when the Inspection Result field contains an error and DevOps Center must inspect the commit again.
sf_devops__Repository__c	**Type** reference **Properties** Create, Filter, Group, Sort **Description** Reference to the repository in which the commit is made. This field is a relationship field. **Relationship Name** sf_devops__Repository__r **Relationship Type** Master-detail **Refers To** sf_devops__Repository__c
sf_devops__Submitted_By_Name__c	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update

Field	Details
	Description Name of the user that performed the commit when it was initiated outside of DevOps Center.
`sf_devops__Submitted_By__c`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Reference to the user that performed the commit when it was initiated in DevOps Center. This field is a relationship field. **Relationship Name** sf_devops__Submitted_By__r **Relationship Type** Lookup **Refers To** User
`sf_devops__Submitted_On__c`	**Type** dateTime **Properties** Create, Filter, Sort, Update **Description** Date and time of the commit.
`sf_devops__Work_Item__c`	**Type** reference **Properties** Create, Filter, Group, Sort, Update **Description** Reference to the work item that the commit is associated with. This work item owns the feature branch. This field is a relationship field. **Relationship Name** sf_devops__Work_Item__r **Relationship Type** Lookup

Field	Details
	Refers To sf_devops__Work_Item__c

SEE ALSO:

- Async Operation Result (sf_devops__Async_Operation_Result__c)
- Repository (sf_devops__Repository__c)
- Work Item (sf_devops__Work_Item__c)

Deploy Component (sf_devops__Deploy_Component__c)

Stores the aggregated set of metadata components that must be deployed as part of a promotion. Includes the metadata components of all the work items associated with the promotion. The Deploy Component object is a child of Deployment Result. This object is available in all orgs that have DevOps Center installed.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
`Name`	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** Name of this Deploy Component record.
`sf_devops__Deployment_Result__c`	**Type** reference **Properties** Create, Filter, Group, Sort **Description** Reference to the parent Deployment Result of this component. This field is a relationship field. **Relationship Name** sf_devops__Deployment_Result__r **Relationship Type** Master-detail

Field	Details
	Refers To sf_devops__Deployment_Result__c
`sf_devops__File_Path__c`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Full pathname of the `-meta.xml` source file for this component. The Heroku application uses this field to determine if the `.forceignore` file on the branch references the component. If it does, this component is ignored during the deployment to the environment.
`sf_devops__Force_Ignored__c`	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Set to `true` by the Heroku application when the component wasn't deployed to the environment because it matched a `.forceignore` rule. The default value is `false`.
`sf_devops__Operation__c`	**Type** string **Properties** Create, Filter, Group, Sort, Update **Description** Specifies the operation, either `UPSERT` or `DELETE`.
`sf_devops__Source_Component__c`	**Type** string **Properties** Create, Filter, Group, Sort, Update **Description** Type and name of this metadata component.

SEE ALSO:

Deployment Result (sf_devops__Deployment_Result__c)

Deployment Result (sf_devops__Deployment_Result__c)

Contains information from DevOps Center to the Heroku application about how to execute a metadata deployment to an environment, such as the Apex tests and test level. After the deployment completes, this object then stores information about the deployment, such as the deployment ID and completion date. See How Promotions Work for more information. This object is available in all orgs that have DevOps Center installed.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
`Name`	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** Name of this Deployment Result record.
`OwnerId`	**Type** reference **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** ID of the user who owns this record. This field is a polymorphic relationship field. **Relationship Name** Owner **Relationship Type** Lookup **Refers To** Group, User
`sf_devops__Check_Deploy_Date__c`	**Type** dateTime **Properties** Create, Filter, Nillable, Sort, Update **Description** Specifies the date and time that a validate-only deployment completed. DevOps Center uses this value to prompt the user when a validate-only deployment is about to expire.

Field	Details
sf_devops__Check_Deploy_Status__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Reference to the Async Operation Result record that was used to perform the validate-only deployment. This field is a relationship field. **Relationship Name** sf_devops__Check_Deploy_Status__r **Relationship Type** Lookup **Refers To** sf_devops__Async_Operation_Result__c
sf_devops__Check_Deploy__c	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** DevOps Center sets this field to `true` to alert the Heroku application to perform a validate-only deployment. The default value is `false`.
sf_devops__Completion_Date__c	**Type** dateTime **Properties** Create, Filter, Nillable, Sort, Update **Description** Date and time when the promotion of this Deployment Result completed.
sf_devops__Deployment_Id__c	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** ID of the deployment. The Heroku application sets this field after a successful deployment to the environment.
sf_devops__Full_Deploy__c	**Type** boolean

Field	Details
	Properties Create, Defaulted on create, Filter, Group, Sort, Update **Description** DevOps Center sets this field to `true` to alert the Heroku application to deploy all of the metadata from the branch, rather than just the metadata components in the Deploy Component records. The default value is `false`.
`sf_devops__Number_Tests_Completed__c`	**Type** double **Properties** Create, Defaulted on create, Filter, Nillable, Sort, Update **Description** Number of tests that completed as part of the deployment to the environment. The Heroku application sets this field after the deployment. DevOps Center uses this field to determine if a validate-only deployment is available for a subsequent quick deploy.
`sf_devops__Run_Tests__c`	**Type** textarea **Properties** Create, Nillable, Update **Description** If the Test Level field is set to `RunSpecifiedTests`, this field contains the comma-separated list of Apex tests to run during the deployment.
`sf_devops__Status__c`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Reference to the Async Operation Result record that was used for the associated promotion or quick deploy. This field is a relationship field. **Relationship Name** sf_devops__Status__r **Relationship Type** Lookup **Refers To** sf_devops__Async_Operation_Result__c
`sf_devops__Test_Level__c`	**Type** string

Field	Details
	Properties Create, Filter, Group, Nillable, Sort, Update **Description** Specifies the level of Apex tests to run during the deployment. The Heroku application uses this field when it executes the deployment. Valid values are: • `NoTestRun` • `RunAllTestsInOrg` • `RunLocalTests` • `RunSpecifiedTests`

SEE ALSO:

Metadata API Developer Guide: deploy()

Async Operation Result (sf_devops__Async_Operation_Result__c)

How Promotions Work

Environment (sf_devops__Environment__c)

Represents a connection from DevOps Center to an environment, which currently can be only a Salesforce org. Developers use development environments to do their work. Each pipeline stage has an associated environment. This object is available in all orgs that have DevOps Center installed.

Warning: As DevOps Center grows to support additional types of promotions in the Salesforce ecosystem, this object might change. For example, current fields that are specific to orgs might migrate to a new org-centric custom object. And additional custom objects might be created to represent MuleSoft, Tableau, and so on.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
`Name`	**Type** string **Properties** Create, Defaulted on create, Filter, Group, idLookup, Nillable, Sort, Update **Description** Name of this Environment record.

Field	Details
sf_devops__Can_Track_Changes__c	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** If `true`, specifies that the org that this environment references has source tracking enabled. The default value is `false`.
sf_devops__Expired__c	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** If `true`, specifies that the org that this environment references has expired. Examples of expired orgs include deleted or expired scratch orgs, or sandboxes that have been refreshed. The default value is `false`.
sf_devops__last_Revision_Counter__c	**Type** double **Properties** Create, Filter, Nillable, Sort, Update **Description** The last revision counter that was pulled from the org that this environment references. DevOps Center uses this value to ensure that it doesn't pull duplicate records from Source Member Reference.
sf_devops__Named_Credential__c	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Name of the named credential that DevOps Center uses to connect to the org referenced by this environment.
sf_devops__Operation_Result__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Reference to the Async Operation Result record for the current ongoing operation associated with this environment. DevOps Center sets this value when the operation begins and clears it when the operation terminates.

Field	Details
	This field is a relationship field. **Relationship Name** sf_devops__Operation_Result__r **Relationship Type** Lookup **Refers To** sf_devops__Async_Operation_Result__c
`sf_devops__Org_Id__c`	**Type** string **Properties** Create, Filter, Group, idLookup, Sort, Update **Description** The 15-character ID of the Salesforce org that this environment is connected to. DevOps Center connects to an org using a URL. When it connects, it compares its org ID with the value in this field. If the two IDs differ, DevOps Center determines that the connected org is a sandbox that has been refreshed.
`sf_devops__Project__c`	**Type** reference **Properties** Create, Filter, Group, Sort **Description** Reference to the project of this environment. This field is a relationship field. **Relationship Name** sf_devops__Project__r **Relationship Type** Master-detail **Refers To** sf_devops__Project__c
`sf_devops__Refresh_Date__c`	**Type** dateTime **Properties** Create, Filter, Nillable, Sort, Update **Description** Date and time that this sandbox was refreshed from its parent. DevOps Center uses this field and Refresh Source to determine which work items are missing from a recently swapped environment (sandbox).

Field	Details
sf_devops__Refresh_Source__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Reference to the environment in DevOps Center that this sandbox was cloned from. DevOps Center uses this field and Refresh Date to determine which work items are missing from a recently swapped environment (sandbox). This field is a relationship field. **Relationship Name** sf_devops__Refresh_Source__r **Relationship Type** Lookup **Refers To** sf_devops__Environment__c
sf_devops__Replaces__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Reference to the environment that this environment replaces. This field is a relationship field. **Relationship Name** sf_devops__Replaces__r **Relationship Type** Lookup **Refers To** sf_devops__Environment__c
sf_devops__Test_Environment__c	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** If `true`, specifies that DevOps Center uses `test.salesforce.com` to log in to the org. If `false`, DevOps Center uses `login.salesforce.com`.

Field	Details
	The default value is `true`.

SEE ALSO:

- Source Member Reference (sf_devops__Source_Member_Reference__c)
- Async Operation Result (sf_devops__Async_Operation_Result__c)
- Project (sf_devops__Project__c)
- *Salesforce Help*: Manage Environments

Hidden Remote Change (sf_devops__Hidden_Remote_Change__c)

Used to hide a Remote Change record from a work item. A sample use case is when a feature branch associated with the work item has a `.forceignore` file that excludes the metadata represented by the Remote Change record. See How DevOps Center Keeps Track of User Changes for more information. This object is available in all orgs that have DevOps Center installed.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
`Name`	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** Name of this Hidden Remote Change record.
`sf_devops__Hidden_by_Force_Ignore__c`	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** If `true`, specifies that this Remote Change is hidden from the work item because it matched a `.forceignore` rule. The default value is `false`.
`sf_devops__Remote_Change__c`	**Type** reference

Field	Details
	Properties Create, Filter, Group, Sort **Description** Reference to the Remote Change record that's hidden. This field is a relationship field. **Relationship Name** sf_devops__Remote_Change__r **Relationship Type** Master-detail **Refers To** sf_devops__Remote_Change__c
`sf_devops__Work_Item__c`	**Type** reference **Properties** Create, Filter, Group, Sort **Description** Reference to the work item that can't see the Remote Change record because it's hidden. This field is a relationship field. **Relationship Name** sf_devops__Work_Item__r **Relationship Type** Master-detail **Refers To** sf_devops__Work_Item__c

SEE ALSO:

- Work Item (sf_devops__Work_Item__c)
- Remote Change (sf_devops__Remote_Change__c)

Merge Result (sf_devops__Merge_Result__c)

Contains information from DevOps Center to the Heroku application about the source control branch to merge as part of a promotion. When the merge completes, this object then stores information about the merge, such as the ID of the merge and when it happened. This object is available in all orgs that have DevOps Center installed.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
`Name`	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** Name of this Merge Result record.
`OwnerId`	**Type** reference **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** ID of the user who owns this record. This field is a polymorphic relationship field. **Relationship Name** Owner **Relationship Type** Lookup **Refers To** Group, User
`sf_devops__Merge_Date__c`	**Type** dateTime **Properties** Create, Filter, Nillable, Sort, Update **Description** Date and time that the merge happened.
`sf_devops__Previous_Remote_Reference__c`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Unique ID of the branch before the merge. In GitHub, this ID is called the HEAD SHA.
`sf_devops__Remote_Reference__c`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update

Field	Details
	Description Unique ID of this merge. In GitHub, this ID is called the SHA.
sf_devops__Source_Branch_Name__c	**Type** string **Properties** Create, Filter, Group, Sort, Update **Description** Name of the branch that's being merged.
sf_devops__Target_Branch_Name__c	**Type** string **Properties** Create, Filter, Group, Sort, Update **Description** Name of the branch into which the source branch is being merged.

Object Activity (sf_devops__Object_Activity__c)

Represents an activity by one of the DevOps Center custom objects. Object Activity records determine the items that are listed in the Activity Histories of work items and pipelines. When DevOps Center performs an operation, it creates an activity record and inserts it in the appropriate user interface view; each activity references the associated custom object. For example, when a user promotes a work item, DevOps Center inserts a promotion initiation activity in the work item Activity History; the activity references the Work Item and Pipeline Stage objects. When the promotion terminates, DevOps Center inserts a second activity that references the same two objects and Async Operation Result. This object is available in all orgs that have DevOps Center installed.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
Name	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** Name of this Object Activity record.

Field	Details
sf_devops__Activity_Date__c	**Type** dateTime **Properties** Create, Filter, Sort, Update **Description** Date and time that the activity occurred.
sf_devops__Activity_Type__c	**Type** string **Properties** Create, Filter, Group, Sort, Update **Description** Type of activity that was performed.
sf_devops__Change_Submission__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Reference to the Change Submission record that was associated with this activity. This field is a relationship field. **Relationship Name** sf_devops__Change_Submission__r **Relationship Type** Lookup **Refers To** sf_devops__Change_Submission__c
sf_devops__Environment__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Reference to the Environment record that was associated with this activity. This field is a relationship field. **Relationship Name** sf_devops__Environment__r **Relationship Type** Lookup **Refers To** sf_devops__Environment__c

Field	Details
`sf_devops__Hidden__c`	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** If `true`, this activity is for internal bookkeeping only, and DevOps Center doesn't include it in the Activity History views. The default value is `false`.
`sf_devops__Operation_Result__c`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Reference to the Async Operation Result record that was associated with this activity. This field is a relationship field. **Relationship Name** sf_devops__Operation_Result__r **Relationship Type** Lookup **Refers To** sf_devops__Async_Operation_Result__c
`sf_devops__Parent_Activity__c`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Reference to the parent activity, if the parent activity triggered this activity. For example, the promotion of a change bundle triggers child activities that represent all the work items in the change bundle that were promoted. This field is a relationship field. **Relationship Name** sf_devops__Parent_Activity__r **Relationship Type** Lookup **Refers To** sf_devops__Object_Activity__c
`sf_devops__Pipeline__c`	**Type** reference

Field	Details
	Properties Create, Filter, Group, Nillable, Sort, Update **Description** Reference to the Pipeline record that's associated with this activity. This field is a relationship field. **Relationship Name** sf_devops__Pipeline__r **Relationship Type** Lookup **Refers To** sf_devops__Pipeline__c
`sf_devops__Project__c`	**Type** reference **Properties** Create, Filter, Group, Sort **Description** Reference to the parent project that this activity is in. This field is a relationship field. **Relationship Name** sf_devops__Project__r **Relationship Type** Master-detail **Refers To** sf_devops__Project__c
`sf_devops__Summary__c`	**Type** string **Properties** Create, Filter, Group, Sort, Update **Description** Summary displayed in the Activities History view for this activity.
`sf_devops__Target_Pipeline_Stage__c`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Reference to the Pipeline Stage record that's associated with this activity. This field is a relationship field.

Field	Details
	Relationship Name sf_devops__Target_Pipeline_Stage__r **Relationship Type** Lookup **Refers To** sf_devops__Pipeline_Stage__c
`sf_devops__Work_Item__c`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Reference to the Work Item record that's associated with this activity. This field is a relationship field. **Relationship Name** sf_devops__Work_Item__r **Relationship Type** Lookup **Refers To** sf_devops__Work_Item__c

SEE ALSO:

- Pipeline (sf_devops__Pipeline__c)
- Pipeline Stage (sf_devops__Pipeline_Stage__c)
- Work Item (sf_devops__Work_Item__c)
- Async Operation Result (sf_devops__Async_Operation_Result__c)
- Change Submission (sf_devops__Change_Submission__c)

Pipeline (sf_devops__Pipeline__c)

Represents a collection of Pipeline Stage records that together make up the DevOps Center release pipeline in a project. This object is available in all orgs that have DevOps Center installed.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
`Name`	**Type** string **Properties** Create, Defaulted on create, Filter, Group, idLookup, Nillable, Sort, Update **Description** Name of this Pipeline record.
`sf_devops__Activated__c`	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** If `true`, the admin has clicked Activate for the pipeline in DevOps Center and users can start promoting work items. If `false`, the admin is still building the pipeline. The default value is `false`.
`sf_devops__Project__c`	**Type** reference **Properties** Create, Filter, Group, Sort **Description** Reference to the parent project of this pipeline. This field is a relationship field. **Relationship Name** sf_devops__Project__r **Relationship Type** Master-detail **Refers To** sf_devops__Project__c

SEE ALSO:

- Pipeline Stage (sf_devops__Pipeline_Stage__c)
- Project (sf_devops__Project__c)

Pipeline Stage (sf_devops__Pipeline_Stage__c)

Represents a connection from a pipeline to an environment; the collection of all pipeline stages in a project make up the release pipeline. This object is available in all orgs that have DevOps Center installed.

DevOps Center doesn't use the Pipeline Stage object to represent the left-most Approved Work Items column in a pipeline; for this reason the column is referred to as a pseudo stage. Instead, DevOps Center computes the items in this column from all Work Item records whose Development Approved field is `true` but haven't yet been promoted.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
`Name`	**Type** string **Properties** Create, Defaulted on create, Filter, Group, idLookup, Nillable, Sort, Update **Description** Name of this Pipeline Stage record.
`sf_devops__Branch__c`	**Type** reference **Properties** Create, Filter, Group, Sort, Update **Description** Reference to the Branch record that contains information about the branch associated with this pipeline stage. This field is a relationship field. **Relationship Name** sf_devops__Branch__r **Relationship Type** Lookup **Refers To** sf_devops__Branch__c
`sf_devops__Environment__c`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Reference to the Environment record associated with this pipeline stage. Currently all environments are Salesforce orgs. This field is a relationship field.

Field	Details
	Relationship Name sf_devops__Environment__r **Relationship Type** Lookup **Refers To** sf_devops__Environment__c
sf_devops__Next_Stage__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Pointer to the next stage in the pipeline. This field is a relationship field. **Relationship Name** sf_devops__Next_Stage__r **Relationship Type** Lookup **Refers To** sf_devops__Pipeline_Stage__c
sf_devops__Operation_Status__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** If this pipeline stage is part of a remote operation, this field references the associated Async Operation Result record. When the operation terminates, DevOps Center clears this field value. This field is a relationship field. **Relationship Name** sf_devops__Operation_Status__r **Relationship Type** Lookup **Refers To** sf_devops__Async_Operation_Result__c
sf_devops__Pipeline__c	**Type** reference **Properties** Create, Filter, Group, Sort

Field	Details
	Description Reference to the Pipeline record that this stage belongs to. This field is a relationship field. **Relationship Name** sf_devops__Pipeline__r **Relationship Type** Master-detail **Refers To** sf_devops__Pipeline__c
`sf_devops__Prerelease__c`	**Type** boolean **Properties** Defaulted on create, Filter, Group, Sort **Description** Specifies whether this stage is the bundling stage. See How Promotions Work for more information. This field is a calculated field.
`sf_devops__Promote_Review_Remote_Reference__c`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Unique ID of the change request from this stage to the next stage. In GitHub, a change request is called a pull request (PR) and the ID is the PR number. DevOps Center creates this change request whenever changes are merged into the branch associated with this pipeline stage.
`sf_devops__Swap_Status__c`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Reference to the Async Operation Result record for the remote operation that's deploying metadata to this stage's environment after it has been swapped. This field is a relationship field. **Relationship Name** sf_devops__Swap_Status__r **Relationship Type** Lookup

Field	Details
	Refers To sf_devops__Async_Operation_Result__c
`sf_devops__Versioned__c`	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** If `true`, this pipeline stage accepts only change bundles for promotion. If `false`, this pipeline stage accepts only work items for promotion. The default value is `false`.

SEE ALSO:

- Pipeline (sf_devops__Pipeline__c)
- Branch (sf_devops__Branch__c)
- Environment (sf_devops__Environment__c)
- Async Operation Result (sf_devops__Async_Operation_Result__c)

Project (sf_devops__Project__c)

Represents the parent of all DevOps Center custom objects. See Understand the DevOps Center Data Model for more information. This object is available in all orgs that have DevOps Center installed.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
`Name`	**Type** string **Properties** Create, Defaulted on create, Filter, Group, idLookup, Nillable, Sort, Update **Description** Name of this Project record.
`OwnerId`	**Type** reference

Field	Details
	Properties Create, Defaulted on create, Filter, Group, Sort, Update **Description** ID of the user who owns this record. This field is a polymorphic relationship field. **Relationship Name** Owner **Relationship Type** Lookup **Refers To** Group, User
`sf_devops__Description__c`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Description of the project. DevOps Center uses this field only to display the description in the UI.
`sf_devops__Hidden__c`	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** If `true`, this project doesn't show up in the DevOps Center list of projects. The default value is `false`.
`sf_devops__Package_Directories__c`	**Type** textarea **Properties** Create, Nillable, Update **Description** List of filepaths that DevOps Center uses when inspecting feature branches for metadata. DevOps Center first loads the list from the `sfdx-project.json` file when the admin creates the Salesforce DX project. DevOps Center subsequently updates the list whenever the `sfdx-project.json` file changes on the branch associated with the final pipeline stage.
`sf_devops__Platform_Repository__c`	**Type** reference

Field	Details
	Properties Create, Filter, Group, Sort, Update **Description** Reference to the project's source control repository. This field is a relationship field. **Relationship Name** sf_devops__Platform_Repository__r **Relationship Type** Lookup **Refers To** sf_devops__Repository__c

SEE ALSO:

Repository (sf_devops__Repository__c)

Remote Change (sf_devops__Remote_Change__c)

Represents a change to an environment that's connected to DevOps Center. In particular, a Remote Change record represents an accumulation of operations on a particular piece of metadata. DevOps Center presents this change to the user so they can pull it into their work item and commit it to the associated feature branch. See How DevOps Center Keeps Track of User Changes for more information. This object is available in all orgs that have DevOps Center installed.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
`Name`	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** Name of this Remote Change record.
`sf_devops__Change_Submission__c`	**Type** reference

Field	Details
	Properties Create, Filter, Group, Nillable, Sort, Update **Description** If this remote change has already been committed to the feature branch, this field contains a reference to the associated Change Submission record. This field is a relationship field. **Relationship Name** sf_devops__Change_Submission__r **Relationship Type** Lookup **Refers To** sf_devops__Change_Submission__c
`sf_devops__Environment__c`	**Type** reference **Properties** Create, Filter, Group, Sort **Description** Reference to the environment that this remote change is associated with. This field is a relationship field. **Relationship Name** sf_devops__Environment__r **Relationship Type** Master-detail **Refers To** sf_devops__Environment__c
`sf_devops__Member_Name__c`	**Type** string **Properties** Create, Filter, Group, Sort, Update **Description** Name of the metadata component that this remote change represents.
`sf_devops__Member_Type__c`	**Type** string **Properties** Create, Filter, Group, Sort, Update **Description** Metadata type that this remote change represents.

Field	Details
sf_devops__Remote_Change_By__c	**Type** string **Properties** Create, Filter, Group, Sort, Update **Description** Username of the user who made the metadata change in the environment.
sf_devops__Remote_Change_On__c	**Type** dateTime **Properties** Create, Filter, Sort, Update **Description** Date and time when the metadata change was made.
sf_devops__Remote_Change_Type__c	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Restricted picklist, Sort, Update **Description** How the metadata was changed. Possible values are: • ADD • CHANGE • MANUAL • REMOVE • RENAME The default value is ADD.
sf_devops__Source_Component__c	**Type** string **Properties** Filter, Nillable, Sort **Description** Combination of the Member Type and Member Name fields.

Field	Details
	This field is a calculated field.

SEE ALSO:

- How DevOps Center Keeps Track of User Changes
- Change Submission (sf_devops__Change_Submission__c)
- Environment (sf_devops__Environment__c)

Repository (sf_devops__Repository__c)

Represents a specific location in a source control system where the metadata for a project is stored. Multiple projects can reference the same repository. This object is available in all orgs that have DevOps Center installed.

Warning: Many of the fields on this object are specific to GitHub. As DevOps Center supports additional source control systems, these fields will likely change or move.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
`Name`	**Type** string **Properties** Create, Defaulted on create, Filter, Group, idLookup, Nillable, Sort, Update **Description** Name of this Repository record.
`OwnerId`	**Type** reference **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** ID of the user who owns this record. This field is a polymorphic relationship field. **Relationship Name** Owner **Relationship Type** Lookup

Field	Details
	Refers To Group, User
sf_devops__Default_Branch__c	**Type** string **Properties** Create, Filter, Group, Sort, Update **Description** Branch in the repository that contains all the changes you plan to release. In GitHub, this branch is typically called `main`.
sf_devops__GitHub_Owner__c	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Owner of the repository. In GitHub, this string is used in the URL for the repository.
sf_devops__GitHub_Repo__c	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Name of the repository. In GitHub, this string is used in the URL for the repository.
sf_devops__Last_Event__c	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** ID of the most recent VCS Event that DevOps Center processed from GitHub. DevOps Center uses this ID to ensure it doesn't duplicate downloading or reprocessing events.
sf_devops__Named_Credential__c	**Type** string **Properties** Create, Filter, Group, Sort, Update **Description** Name of the named credential that DevOps Center uses to connect to this repository.
sf_devops__Polling_Interval__c	**Type** double

Field	Details
	Properties Create, Filter, Nillable, Sort, Update **Description** Specifies how often DevOps Center polls for VCS Events.
`sf_devops__Provider__c`	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Restricted picklist, Sort, Update **Description** Provider of this repository. Currently, the only valid value is `GitHub`.

SEE ALSO:

Project (sf_devops__Project__c)

Source Member Reference (sf_devops__Source_Member_Reference__c)

Represents a copy of the relevant information from a SourceMember Tooling API record in a development environment. DevOps Center copies this data to track the metadata changes that a user hasn't yet pulled into their work item. Copying the data also makes computing Remote Change records more efficient. See How DevOps Center Keeps Track of User Changes for more information. This object is available in all orgs that have DevOps Center installed.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
`Name`	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** Name of this Source Member Reference record.
`sf_devops__Remote_Change__c`	**Type** reference **Properties** Create, Filter, Group, Sort

Field	Details
	Description Reference to the Remote Change record that this Source Member Reference record is linked to. This field is a relationship field. **Relationship Name** sf_devops__Remote_Change__r **Relationship Type** Master-detail **Refers To** sf_devops__Remote_Change__c
`sf_devops__Revision_Counter__c`	**Type** double **Properties** Create, Filter, Sort, Update **Description** Value of the RevisionCounter field of the associated SourceMember Tooling API record.
`sf_devops__Source_Member_Id__c`	**Type** string **Properties** Create, Filter, Group, Sort, Update **Description** ID of the associated SourceMember Tooling API record.

SEE ALSO:

- How DevOps Center Keeps Track of User Changes
- Remote Change (sf_devops__Remote_Change__c)
- *Tooling API Developer Guide*: SourceMember

Submit Component (sf_devops__Submit_Component__c)

Represents a metadata component that was committed to a feature branch in the source control repository. The commit can be initiated in one of two ways, either from within DevOps Center or directly in the source control repository. Each metadata component that's part of the commit must also be deployed to an environment, and DevOps Center uses the Submit Component object to model the metadata. The Submit Component object is a child of Change Submission. This object is available in all orgs that have DevOps Center installed.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
`Name`	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** Name of this Submit Component record.
`sf_devops__Change_Submission__c`	**Type** reference **Properties** Create, Filter, Group, Sort **Description** Reference to the parent Change Submission record that this Submit Component record is a child of. This field is a relationship field. **Relationship Name** sf_devops__Change_Submission__r **Relationship Type** Master-detail **Refers To** sf_devops__Change_Submission__c
`sf_devops__Empty__c`	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** If `true`, indicates that the metadata component didn't change anything on the associated feature branch. This scenario occurs when a user pulls removed metadata from a development environment, but the metadata source doesn't yet exist on the feature branch. DevOps Center still removes the metadata from the environment but doesn't change the feature branch. The default value is `false`.
`sf_devops__File_Path__c`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Full pathname of the `-meta.xml` source file for this component on the feature branch.

Field	Details
`sf_devops__Operation__c`	**Type** string **Properties** Create, Filter, Group, Sort, Update **Description** Specifies the operation that was executed on this component. Either `ADD`, `CHANGE`, or `DELETE`.
`sf_devops__Source_Component__c`	**Type** string **Properties** Create, Filter, Group, Sort, Update **Description** Combination of the metadata name and type.
`sf_devops__Source_Remote_Change__c`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** If a user in DevOps Center initiated the commit, this field is a reference to the associated Remote Change record that displayed the component in the user interface. This field is a relationship field. **Relationship Name** sf_devops__Source_Remote_Change__r **Relationship Type** Lookup **Refers To** sf_devops__Remote_Change__c

SEE ALSO:

Change Submission (sf_devops__Change_Submission__c)

Remote Change (sf_devops__Remote_Change__c)

VCS (sf_devops__Vcs__c)

Represents a supported source (version) control system. This object is available in all orgs that have DevOps Center package version 8.2 and later. Available in API version 62.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
`LastReferencedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** Not used by DevOps Center.
`LastViewedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** Not used by DevOps Center.
`Name`	**Type** string **Properties** Create, Defaulted on create, Filter, Group, idLookup, Nillable, Sort, Update **Description** Name of the source control system.
`OwnerId`	**Type** reference **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Not used by DevOps Center. **Relationship Name** Owner **Refers To** Group, User
`sf_devops__Base_URL__c`	**Type** url

Field	Details
	Properties Create, Filter, Group, Sort, Update **Description** Base URL for the source control system, for example, `https://github.com`.
`sf_devops__Named_Credential__c`	**Type** string **Properties** Create, Filter, Group, Sort, Update **Description** The developer (API) name of the named credential used to access this source control system.
`sf_devops__Service_Provider__c`	**Type** string **Properties** Create, Filter, Group, Sort, Update **Description** Name of the provider for this source control system. This field must reference an existing developer (API) name for a Service Provider CMT that implements the Vcs Provider interface.

VCS Synch State (sf_devops__Vcs_Synch_State__c)

Represents the synchronization state between DevOps Center and the source (version) control system. DevOps Center uses this object to track all synchronization events to ensure that DevOps Center is working with the latest version of the code in the source control repository. This object is available in all orgs that have DevOps Center package version 8.2 and later. Available in API version 62.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
`Name`	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** Name of the synchronization state.

Field	Details
`OwnerId`	**Type** reference **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Not used by DevOps Center. **Relationship Name** Owner **Refers To** Group, User
`sf_devops__Project__c`	**Type** reference **Properties** Create, Filter, Group, Sort, Update **Description** Reference to a DevOps Center project that this state belongs to. This field is a relationship field. **Relationship Name** sf_devops__Project__r **Refers To** sf_devops__Project__c
`sf_devops__Synch_Name__c`	**Type** string **Properties** Create, Filter, Group, Sort, Update **Description** Name of the sychronization state used for logging.

Work Item (sf_devops__Work_Item__c)

Represents a collection of metadata changes in a project. A work item can be associated with an environment in which the work is performed. If it's not connected to an environment, the VCS Event object handles the changes. A work item goes through a number of development lifecycle stages until all development work is complete and the work item is part of the release pipeline. This object is available in all orgs that have DevOps Center installed.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
`Name`	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** Name of this Work Item record.
`sf_devops__Assigned_To__c`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Reference to the DevOps Center user that this work item is assigned to. DevOps Center uses this field only to display the user's name in the UI. This field is a relationship field. **Relationship Name** sf_devops__Assigned_To__r **Relationship Type** Lookup **Refers To** User
`sf_devops__Branch__c`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Reference to the Branch record that contains all the information about the feature branch associated with this work item. DevOps Center initially sets this value when the work item transitions to `IN_PROGRESS`. The value stays set until the work item becomes a member of the change bundle. This field is a relationship field. **Relationship Name** sf_devops__Branch__r **Relationship Type** Lookup **Refers To** sf_devops__Branch__c

Field	Details
sf_devops__Change_Bundle__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** If this work item is part of a bundle, this field references the Change Bundle record that this work item belongs to. This field is a relationship field. **Relationship Name** sf_devops__Change_Bundle__r **Relationship Type** Lookup **Refers To** sf_devops__Change_Bundle__c
sf_devops__Combine_Status__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Reference to the Async Operation Result record associated with the most recent attempt to combine other work items with this work item. This field is a relationship field. **Relationship Name** sf_devops__Combine_Status__r **Relationship Type** Lookup **Refers To** sf_devops__Async_Operation_Result__c
sf_devops__Combined_With__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Reference to the other work item that this work item has been combined with. This field is a relationship field. **Relationship Name** sf_devops__Combined_With__r **Relationship Type** Lookup

Field	Details
	Refers To sf_devops__Work_Item__c
`sf_devops__Concluded__c`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** If this field is set to a value, the work item has finished being used for active development or promotion. A work item is done when it's either reached the last stage of the pipeline or a user has set its status to `NEVERED`.
`sf_devops__Cross_Environment_Combination__c`	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** If `true`, this work item is the result of a combination operation where metadata came from multiple development environments. The default value is `false`.
`sf_devops__Description__c`	**Type** textarea **Properties** Create, Nillable, Update **Description** Description of the work item. DevOps Center uses this field only to display the description in the UI.
`sf_devops__Development_Approved_By__c`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Reference to the user who clicked the **Ready to Promote** button for this work item in DevOps Center. This field is a relationship field. **Relationship Name** sf_devops__Development_Approved_By__r **Relationship Type** Lookup

Field	Details
	Refers To User
sf_devops__Development_Approved__c	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** If `true`, a user has clicked the **Ready to Promote** button for this work item in DevOps Center and the work item shows up in the Approved Work Items column in the pipeline. The default value is `false`.
sf_devops__Development_Environment__c	**Type** string **Properties** Filter, Nillable, Sort **Description** Environment in which the work for this work item was developed. This field is a calculated field.
sf_devops__Environment__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Environment in which active development work for this work item is happening. DevOps Center clears this field value when the work item's status is `PROMOTED`. This field is a relationship field. **Relationship Name** sf_devops__Environment__r **Relationship Type** Lookup **Refers To** sf_devops__Environment__c
sf_devops__Operation_Status__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update

Field	Details
	Description If this work item is part of a remote operation, this field references the Async Operation Result record for that operation. DevOps Center clears this field value when the operation terminates. This field is a relationship field. **Relationship Name** sf_devops__Operation_Status__r **Relationship Type** Lookup **Refers To** sf_devops__Async_Operation_Result__c
`sf_devops__Project__c`	**Type** reference **Properties** Create, Filter, Group, Sort **Description** Reference to the parent project of this work item. This field is a relationship field. **Relationship Name** sf_devops__Project__r **Relationship Type** Master-detail **Refers To** sf_devops__Project__c
`sf_devops__Promoted_From_Environment__c`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Reference to the development environment that this work item was originally promoted from. When a user promotes a work item, DevOps Center clears the Environment field value of its associated Work Item record. However, DevOps Center must still know which environment the work item was originally promoted from so it can do further processing. So when the first promotion of a work item completes, DevOps Center sets this field to the value of the Environment field before it's cleared. This field is a relationship field. **Relationship Name** sf_devops__Promoted_From_Environment__r **Relationship Type** Lookup

Field	Details
	Refers To sf_devops__Environment__c
`sf_devops__Promoted__c`	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** If `true`, then this work item has been promoted at least one time. The default value is `false`.
`sf_devops__Rebase_Branch__c`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Reference to the Branch record that this work item is using as its rebase branch. This field is a relationship field. **Relationship Name** sf_devops__Rebase_Branch__r **Relationship Type** Lookup **Refers To** sf_devops__Branch__c
`sf_devops__Review_Remote_Reference__c`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Unique ID of the change request for this work item. In GitHub, a change request is called a pull request (PR) and the ID is the PR number. DevOps Center clears this field when the work item's status changes to `IN_REVIEW`. As the work item moves through the pipeline stages, DevOps Center creates change requests and updates this field with the new ID. When the work item becomes a member of a change bundle, DevOps Center clears this field and no longer sets it.
`sf_devops__State__c`	**Type** string **Properties** Filter, Nillable, Sort

Field	Details
	Description Current state of this work item. Valid values: • NEW • IN_PROGRESS • IN_REVIEW • APPROVED • PROMOTED • CLOSED • NEVERED
sf_devops__Subject__c	**Type** string **Properties** Create, Filter, Group, Sort, Update **Description** Subject of the work item. DevOps Center uses this field only to display the subject in the UI.

SEE ALSO:

- Branch (sf_devops__Branch__c)
- Environment (sf_devops__Environment__c)
- Change Bundle (sf_devops__Change_Bundle__c)
- Async Operation Result (sf_devops__Async_Operation_Result__c)
- Project (sf_devops__Project__c)

Work Item Promote (sf_devops__Work_Item_Promote__c)

Represents the unbundled promotion of a work item to the next stage in a pipeline. See Unbundled Promotions: A Deeper Look for more information. This object is available in all orgs that have DevOps Center installed.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
Name	**Type** string

Field	Details
	Properties Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** Name of this Work Item Promote record.
sf_devops__Deployment_Result__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Reference to the Deploy Result record that DevOps Center used to control the deployment of the work item. This field is a relationship field. **Relationship Name** sf_devops__Deployment_Result__r **Relationship Type** Lookup **Refers To** sf_devops__Deployment_Result__c
sf_devops__Merge_Result__c	**Type** reference **Properties** Create, Filter, Group, Sort, Update **Description** Reference to the Merge Result record that DevOps Center used to control how the feature branch associated with the work item was merged as part of the promotion. This field is a relationship field. **Relationship Name** sf_devops__Merge_Result__r **Relationship Type** Lookup **Refers To** sf_devops__Merge_Result__c
sf_devops__Pipeline_Stage__c	**Type** reference **Properties** Create, Filter, Group, Sort **Description** Reference to the pipeline stage that the work item was promoted to.

Field	Details
	This field is a relationship field. **Relationship Name** sf_devops__Pipeline_Stage__r **Relationship Type** Master-detail **Refers To** sf_devops__Pipeline_Stage__c
sf_devops__Rebase_Status__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Reference to the Async Operation Result record associated with a potential remote rebase operation. After DevOps Center performs an unbundled promotion of a work item, it checks whether another unbundled promotion is pending. If there is, DevOps Center must rebase the feature branch of the original work item to the branch in the next stage of the pipeline. This field references the Async Operation Result that's doing this work. This field is a relationship field. **Relationship Name** sf_devops__Rebase_Status__r **Relationship Type** Lookup **Refers To** sf_devops__Async_Operation_Result__c
sf_devops__Status__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Reference to the Async Operation Result record associated with the remote operation of promoting this work item. This field is a relationship field. **Relationship Name** sf_devops__Status__r **Relationship Type** Lookup **Refers To** sf_devops__Async_Operation_Result__c

Field	Details
sf_devops__Work_Item__c	**Type** reference **Properties** Create, Filter, Group, Sort **Description** Reference to the work item that was promoted. This field is a relationship field. **Relationship Name** sf_devops__Work_Item__r **Relationship Type** Master-detail **Refers To** sf_devops__Work_Item__c

SEE ALSO:

- Deployment Result (sf_devops__Deployment_Result__c)
- Async Operation Result (sf_devops__Async_Operation_Result__c)
- Merge Result (sf_devops__Merge_Result__c)
- Pipeline Stage (sf_devops__Pipeline_Stage__c)
- Work Item (sf_devops__Work_Item__c)

DevOps Center Custom Field on the User Standard Object

The DevOps Center data model uses the User standard object and adds this custom field.

Field	Details
sf_devops__GitHub_Primary_Email_Address__c	**Type** email **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Primary email address of the user's GitHub account.

SEE ALSO:

- *Object Reference for the Salesforce Platform*: User

DevOps Center Custom Platform Events

DevOps Center generates platform events for work items as they move through the development and promotion lifecycle.

In the development phases, DevOps Center generates a platform event every time a work item changes state (New, In Progress, and so on). DevOps Center also generates a platform event for every commit on the work item's feature branch and when the change request is opened.

In the promotion lifecycle, DevOps Center generates a platform event whenever the work item's metadata is merged into a pipeline stage. DevOps Center also generates a platform event whenever a work item's metadata is deployed to a pipeline stage's org.

You can subscribe to these events using all the methods supported by Salesforce (Apex Triggers, Flows, LWCs, APIs, and so on).

Deployment (sf_devops__Deployment__e)
Notifies subscribers when a work item's metadata is deployed to a pipeline stage. This object is available in API version 62.0 and later.

Work Item Commit (sf_devops__Work_Item_Commit__e)
Notifies subscribers whenever a commit occurs on a work item's feature branch. This object is available in API version 62.0 and later.

Work Item Merged Change Request (sf_devops__Work_Item_Merged_Change_Request__e)
Notifies subscribers when a work item's metadata is merged into a pipeline stage's branch. This object is available in API version 62.0 and later.

Work Item Open Change Request (sf_devops__Work_Item_Open_Change_Request__e)
Notifies subscribers whenever a change request (pull request) is opened for a work item. This object is available in API version 62.0 and later.

Work Item State Change (sf_devops__Work_Item_State_Change__e)
Notifies subscribers when the State__c field of a work item changes. This object is available in API version 62.0 and later.

SEE ALSO:

GitHub Repository: DevOps Center Extension Showcase

Deployment (sf_devops__Deployment__e)

Notifies subscribers when a work item's metadata is deployed to a pipeline stage. This object is available in API version 62.0 and later.

Supported Calls

`create()`, `describeSObjects()`

Supported Subscribers

Subscriber	Supported?
Apex Triggers	✔
Flows	✔
Processes	✔

Subscriber	Supported?
Pub/Sub API	✔
Streaming API (CometD)	✔

Streaming API Subscription Channel

`/event/sf_devops__Deployment__e`

Event Delivery Allocation Enforced

Yes

Special Access Rules

DevOps Center package version 8.2 or later must be installed in the org.

Fields

Field	Details
`EventUuid`	**Type** string **Properties** Nillable **Description** The unique ID of the event, which is shared with the corresponding storage object. For example, `0a4779b0-0da1-4619-a373-0a36991dff90`. Use this field to correlate the event with its storage object.
`ReplayId`	**Type** string **Properties** Nillable **Description** Represents an ID value that is populated by the system and refers to the position of the event in the event stream. Replay ID values aren't guaranteed to be contiguous for consecutive events. A subscriber can store a replay ID value and use it on resubscription to retrieve missed events that are within the retention window.
`sf_devops__Change_Bundle_Id__c`	**Type** string **Properties** Create, Nillable

Field	Details
	Description The ID of the work item change bundle that was deployed. This field is empty if an individual work item was deployed.
`sf_devops__Deployment_Id__c`	**Type** string **Properties** Create, Nillable **Description** The ID for the deployment in the target org.
`sf_devops__Target_Stage_Id__c`	**Type** string **Properties** Create, Nillable **Description** The ID of the target stage to which the changes were deployed.
`sf_devops__Work_Item_Id__c`	**Type** string **Properties** Create, Nillable **Description** The ID of the work item that was deployed. This field is empty if a work item change bundle was deployed.

Work Item Commit (sf_devops__Work_Item_Commit__e)

Notifies subscribers whenever a commit occurs on a work item's feature branch. This object is available in API version 62.0 and later.

The event isn't generated if a commit occurs on a feature branch after a work item has been promoted.

Supported Calls

`create()`, `describeSObjects()`

Supported Subscribers

Subscriber	Supported?
Apex Triggers	✔
Flows	✔

Subscriber	Supported?
Processes	✔
Pub/Sub API	✔
Streaming API (CometD)	✔

Streaming API Subscription Channel

`/event/sf_devops__Work_Item_Commit__e`

Event Delivery Allocation Enforced

Yes

Special Access Rules

DevOps Center package version 8.2 or later must be installed in the org.

Fields

Field	Details
`EventUuid`	**Type** string **Properties** Nillable **Description** The unique ID of the event, which is shared with the corresponding storage object. For example, `0a4779b0-0da1-4619-a373-0a36991dff90`. Use this field to correlate the event with its storage object.
`ReplayId`	**Type** string **Properties** Nillable **Description** Represents an ID value that is populated by the system and refers to the position of the event in the event stream. Replay ID values aren't guaranteed to be contiguous for consecutive events. A subscriber can store a replay ID value and use it on resubscription to retrieve missed events that are within the retention window.
`sf_devops__Remote_Reference__c`	**Type** string

Field	Details
	Properties Create **Description** The unique ID of the commit in the source control system.
`sf_devops__Work_Item_Id__c`	**Type** string **Properties** Create **Description** The ID of the work item associated with the commit on the work item's feature branch.

Work Item Merged Change Request (sf_devops__Work_Item_Merged_Change_Request__e)

Notifies subscribers when a work item's metadata is merged into a pipeline stage's branch. This object is available in API version 62.0 and later.

Supported Calls

`create()`, `describeSObjects()`

Supported Subscribers

Subscriber	Supported?
Apex Triggers	✔
Flows	✔
Processes	✔
Pub/Sub API	✔
Streaming API (CometD)	✔

Streaming API Subscription Channel

`/event/sf_devops__Work_Item_Merged_Change_Request__e`

Event Delivery Allocation Enforced

Yes

Special Access Rules

DevOps Center package version 8.2 or later must be installed in the org.

Fields

Field	Details
`EventUuid`	**Type** string **Properties** Nillable **Description** The unique ID of the event, which is shared with the corresponding storage object. For example, `0a4779b0-0da1-4619-a373-0a36991dff90`. Use this field to correlate the event with its storage object.
`ReplayId`	**Type** string **Properties** Nillable **Description** Represents an ID value that is populated by the system and refers to the position of the event in the event stream. Replay ID values aren't guaranteed to be contiguous for consecutive events. A subscriber can store a replay ID value and use it on resubscription to retrieve missed events that are within the retention window.
`sf_devops__Remote_Reference__c`	**Type** string **Properties** Create, Nillable **Description** The unique ID of the change request in the source control system.
`sf_devops__Source_Stage_Id__c`	**Type** string **Properties** Create, Nillable **Description** The ID of the source pipeline stage if the changes were merged from a pipeline stage to the next pipeline stage. Not applicable if the changes were merged from a work item's feature branch.
`sf_devops__Target_Stage_Id__c`	**Type** string

Field	Details
	Properties Create, Nillable **Description** The ID for the target pipeline stage.
`sf_devops__Work_Item_Id__c`	**Type** string **Properties** Create, Nillable **Description** The ID of the work item that contained the changes that were merged.

Work Item Open Change Request (sf_devops__Work_Item_Open_Change_Request__e)

Notifies subscribers whenever a change request (pull request) is opened for a work item. This object is available in API version 62.0 and later.

This event occurs whether the change request was initiated in DevOps Center or directly in the source control system. For changes made in feature branches, the change request is associated with the first pipeline stage. After a promotion to a pipeline stage, the event is generated for the next pipeline stage. This event is generated after a promotion as well as when a change request is opened for the next pipeline stage.

Supported Calls

`create(), describeSObjects()`

Supported Subscribers

Subscriber	Supported?
Apex Triggers	✔
Flows	✔
Processes	✔
Pub/Sub API	✔
Streaming API (CometD)	✔

Streaming API Subscription Channel

`/event/sf_devops__Work_Item_Open_Change_Request__e`

Event Delivery Allocation Enforced

Yes

Special Access Rules

DevOps Center package version 8.2 or later must be installed in the org.

Fields

Field	Details
`EventUuid`	**Type** string **Properties** Nillable **Description** The unique ID of the event, which is shared with the corresponding storage object. For example, `0a4779b0-0da1-4619-a373-0a36991dff90`. Use this field to correlate the event with its storage object.
`ReplayId`	**Type** string **Properties** Nillable **Description** Represents an ID value that is populated by the system and refers to the position of the event in the event stream. Replay ID values aren't guaranteed to be contiguous for consecutive events. A subscriber can store a replay ID value and use it on resubscription to retrieve missed events that are within the retention window.
`sf_devops__Remote_Reference__c`	**Type** string **Properties** Create, Nillable **Description** The unique ID of the change request in the source control system.
`sf_devops__Target_Stage_Id__c`	**Type** string **Properties** Create, Nillable **Description** The ID of the pipeline stage where the change request will be merged.

Field	Details
`sf_devops__Work_Item_Id__c`	**Type** string **Properties** Create, Nillable **Description** The ID of the work item associated with the newly created change request.

Work Item State Change (sf_devops__Work_Item_State_Change__e)

Notifies subscribers when the State__c field of a work item changes. This object is available in API version 62.0 and later.

This event is empty if the current state is `New`. In some cases, a work item can revert to a previous state, for example, if it's not approved for promotion.

Supported Calls

`create()`, `describeSObjects()`

Supported Subscribers

Subscriber	Supported?
Apex Triggers	✔
Flows	✔
Processes	✔
Pub/Sub API	✔
Streaming API (CometD)	✔

Streaming API Subscription Channel

`/event/sf_devops__Work_Item_State_Change__e`

Event Delivery Allocation Enforced

Yes

Special Access Rules

DevOps Center package version 8.2 or later must be installed in the org.

Fields

Field	Details
`EventUuid`	**Type** string **Properties** Nillable **Description** The unique ID of the event, which is shared with the corresponding storage object. For example, `0a4779b0-0da1-4619-a373-0a36991dff90`. Use this field to correlate the event with its storage object.
`ReplayId`	**Type** string **Properties** Nillable **Description** Represents an ID value that is populated by the system and refers to the position of the event in the event stream. Replay ID values aren't guaranteed to be contiguous for consecutive events. A subscriber can store a replay ID value and use it on resubscription to retrieve missed events that are within the retention window.
`sf_devops__New_State__c`	**Type** string **Properties** Create **Description** The current state of the work item.
`sf_devops__Previous_State__c`	**Type** string **Properties** Create, Nillable **Description** The previous state of the work item.
`sf_devops__Work_Item_Id__c`	**Type** string **Properties** Create **Description** The ID of the work item that changed state.