

Administrator Guide

Amazon CodeCatalyst



Amazon CodeCatalyst: Administrator Guide

Copyright © 2025 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Amazon's trademarks and trade dress may not be used in connection with any product or service that is not Amazon's, in any manner that is likely to cause confusion among customers, or in any manner that disparages or discredits Amazon. All other trademarks not owned by Amazon are the property of their respective owners, who may or may not be affiliated with, connected to, or sponsored by Amazon.

Table of Contents

Who is this guide for?	1
Concepts	2
AWS Builder ID spaces in CodeCatalyst	2
Spaces that support identity federation in CodeCatalyst	2
Identity provider (IdP)	3
Identity source	3
Identity store	3
Single sign-on (SSO)	4
Identity Center application	4
Instance in IAM Identity Center	5
Organization instance in IAM Identity Center	5
Account instance in IAM Identity Center	5
How do spaces work with identity federation and SSO	5
Setting up a space	7
Setting up a Builder ID space	8
Setting up a federated space	12
Planning your space that supports identity federation	14
Prerequisite 1: Setting up an organization in AWS Organizations	15
Prerequisite 2: Enable an instance for identity federation	16
Prerequisite 3: Setting up identity federation in IAM Identity Center	17
Inviting users to a Builder ID space	20
Creating additional Builder ID spaces	21
Deleting a Builder ID space	24
Administering federated spaces	26
Creating a space for identity federation	27
Next steps: Create teams, projects, and resources in CodeCatalyst	30
Configure an existing CodeCatalyst space for identity federation	30
Viewing SSO users and groups for a space	31
Adding SSO groups to a space that supports identity federation	32
Adding the Space administrator role to SSO users in a space	32
Deleting a space that supports identity federation	33
Administering Identity Center applications	34
Viewing Identity Center application details	35
Editing Identity Center application details	36

Associating a space to your Identity Center application	37
Disassociating an Identity Center application from a space	38
Administering connected accounts	40
Adding an account connection for a space (in AWS)	41
Removing an account from a space (in AWS)	41
Tagging account connections	42
Adding tags to account connections	42
Editing tags of account connections	43
Removing tags from account connections	43
Enabling or disabling project-restricted account connections	44
Considerations for project-restricted account connections	44
Enabling project-restricted account connections	45
Disabling project-restricted account connections	46
Enabling generative AI features	47
Enabling or disabling generative AI features	48
Viewing usage of generative AI features	50
Connecting a VPC to a space	51
Use cases	52
How do I get started with VPC connections?	52
Limitations of VPC connections in CodeCatalyst	53
Setting up a VPC	53
VPC requirements	53
Troubleshooting your VPC setup	54
Adding VPC connections for a space	54
Configuring VPC endpoints for a space	57
CodeCatalyst VPC endpoint service names	58
Managing a default VPC connection for a space	59
Setting a default VPC connection	59
Removing a default VPC connection	60
Editing VPC connections for a space	60
Removing VPC connections for a space	61
Administering billing	63
Setting up a billing account	64
Step 1: Add the billing account to your space	64
Step 2: Choose the billing account in CodeCatalyst	64
Changing allowed tiers for a billing account	65

Changing your CodeCatalyst billing tier	66
Changing a billing account	67
Viewing your CodeCatalyst plan and billing	67
Viewing CodeCatalyst accounts for billing in AWS	68
Getting AWS billing information	69
Troubleshooting billing	70
I cannot access the Amazon CodeCatalyst Spaces page in the AWS Management Console	70
I cannot change the billing tier for my space	70
Viewing and configuring your user profile	72
Document history	74

Who is this guide for?

This guide is for users with the **Space administrator** role who want to set up and administer Amazon CodeCatalyst spaces. An *Amazon CodeCatalyst space*, or simply a *space*, is a logical container for one or more related projects, and includes settings not related to a particular project, such as billing and single sign-on (SSO) settings. Spaces are usually named after your company, department, or group.

This guide describes how to perform the following space-related setup and administration tasks:

- Setting up spaces
- Deleting Builder ID spaces
- Inviting users to a Builder ID space
- Performing management activities on federated (SSO) spaces
- Enabling generative AI features in your space
- Connecting an Amazon VPC to your space
- Viewing and changing billing information
- Viewing and configuring your user profile

For all other tasks, see the *Amazon CodeCatalyst User Guide*.

For more information about CodeCatalyst, see [What is Amazon CodeCatalyst?](#) in the *Amazon CodeCatalyst User Guide*.

Concepts

Get familiar with the key concepts for administration in CodeCatalyst. These concepts include terms used in supporting access, user management, and identity federation.

Topics

- [AWS Builder ID spaces in CodeCatalyst](#)
- [Spaces that support identity federation in CodeCatalyst](#)
- [Identity provider \(IdP\)](#)
- [Identity source](#)
- [Identity store](#)
- [Single sign-on \(SSO\)](#)
- [Identity Center application](#)
- [Instance in IAM Identity Center](#)
- [Organization instance in IAM Identity Center](#)
- [Account instance in IAM Identity Center](#)
- [How do spaces work with identity federation and SSO](#)

AWS Builder ID spaces in CodeCatalyst

The **Space administrator** adds users to CodeCatalyst by sending individual emails from the members page. Users who are invited or sign up to CodeCatalyst create their own AWS Builder ID. The profile is managed in AWS Builder ID and displays as the user name and profile information in the user settings in CodeCatalyst.

Spaces that support identity federation in CodeCatalyst

Users who have been added to the SSO users and groups for the IAM Identity Center instance and are managed in the identity store and are added to your space through IAM Identity Center. The **Space administrator** syncs the CodeCatalyst members page for the latest updates. Users sign in using the SSO sign-in portal as set up in the instance in IAM Identity Center for your company. Spaces that support identity federation are connected to the identity store through the Identity Center application and its mapping to the identity store ID. For more information, see [Setting up a space that supports identity federation](#).

Identity provider (IdP)

When you add single sign-on access to an AWS account, IAM Identity Center creates an identity provider. An identity provider helps keep your AWS account secure because you don't have to distribute or embed long-term security credentials, such as access keys, in your application.

Identity source

With IAM Identity Center, you can manage your identities in your preferred identity source. While you can change your identity source, you can have only one directory or one SAML 2.0 identity provider connected to IAM Identity Center at any given time. For information about supported providers, see [Supported identity providers](#).

Important

Dev Environments aren't available for users in spaces where Active Directory is used as the identity provider. When planning a space where the identity provider will be Active Directory, note that users will not be able to use Dev Environments. For more information, see [I can't create a Dev Environment when I'm signed into CodeCatalyst using a single sign-on account](#).

Note

CodeCatalyst spaces with identity federation can support service providers that are supported by IAM Identity Center. CodeCatalyst inherits the identity source that is managed in IAM Identity Center. For more information, see [Manage your identity source](#).

Identity store

Your provider in IAM Identity Center uses an identity store where federated identities are managed. The identity store ID is associated with your enabled IAM Identity Center instance and represents your company's association through your application. The identity store in IAM Identity Center is where your federated identities are managed. Members are shown in the CodeCatalyst member lists for your space. Your Identity Center application is associated with at least one identity store ID

and instance ID in IAM Identity Center. Deleting the IAM Identity Center instance invalidates your Identity Center application.

Single sign-on (SSO)

Single sign-on allows users to sign in to multiple applications by signing in to a single page. SSO support for spaces in CodeCatalyst is managed across domains for each identity store or external provider that is connected as a company to the CodeCatalyst space. A user whose company has been onboarded to a CodeCatalyst space can use SSO to enter their sign-in credentials on a single sign-in page to access CodeCatalyst and all other SaaS applications.

Note

SSO configuration for CodeCatalyst is not associated with an AWS Builder ID. The association is through the AWS account for the space.

Identity Center application

An *Identity Center application* is an association between your CodeCatalyst space and IAM Identity Center. The Identity Center application allows users from your company directory to sign in to CodeCatalyst, so your application name will represent your company and will be visible for selection as an option where users from a workforce directory will access CodeCatalyst. As part of creating a space that supports identity federation, you will choose or create the Identity Center application that will be associated with your space. You can associate multiple spaces with a single Identity Center application. When setting up the Identity Center application for CodeCatalyst, note that the application name must be unique across CodeCatalyst and your IAM Identity Center instances. This uniqueness requirement helps prevent confusion and ensures proper identification of different applications. This unique name is primarily for administrative purposes within IAM Identity Center and doesn't affect the functionality of CodeCatalyst.

Note

The name for your Identity Center application must be globally unique. In addition, since the name will be viewable for signing in and on certain pages in CodeCatalyst, choose a name that will suitably relate to your company for users signing in.

Instance in IAM Identity Center

A single deployment of IAM Identity Center. There are two types of instances available for IAM Identity Center: *organization instances* and *account instances*.

Organization instance in IAM Identity Center

An instance of IAM Identity Center that you enable in the management account in AWS Organizations. Organization instances support all features of IAM Identity Center. We recommend that you deploy an organization instance of IAM Identity Center rather than account instances to minimize the number of management points.

Account instance in IAM Identity Center

An instance of IAM Identity Center that is bound to a single AWS account, and that is visible only within the AWS account and AWS Region in which it is enabled.

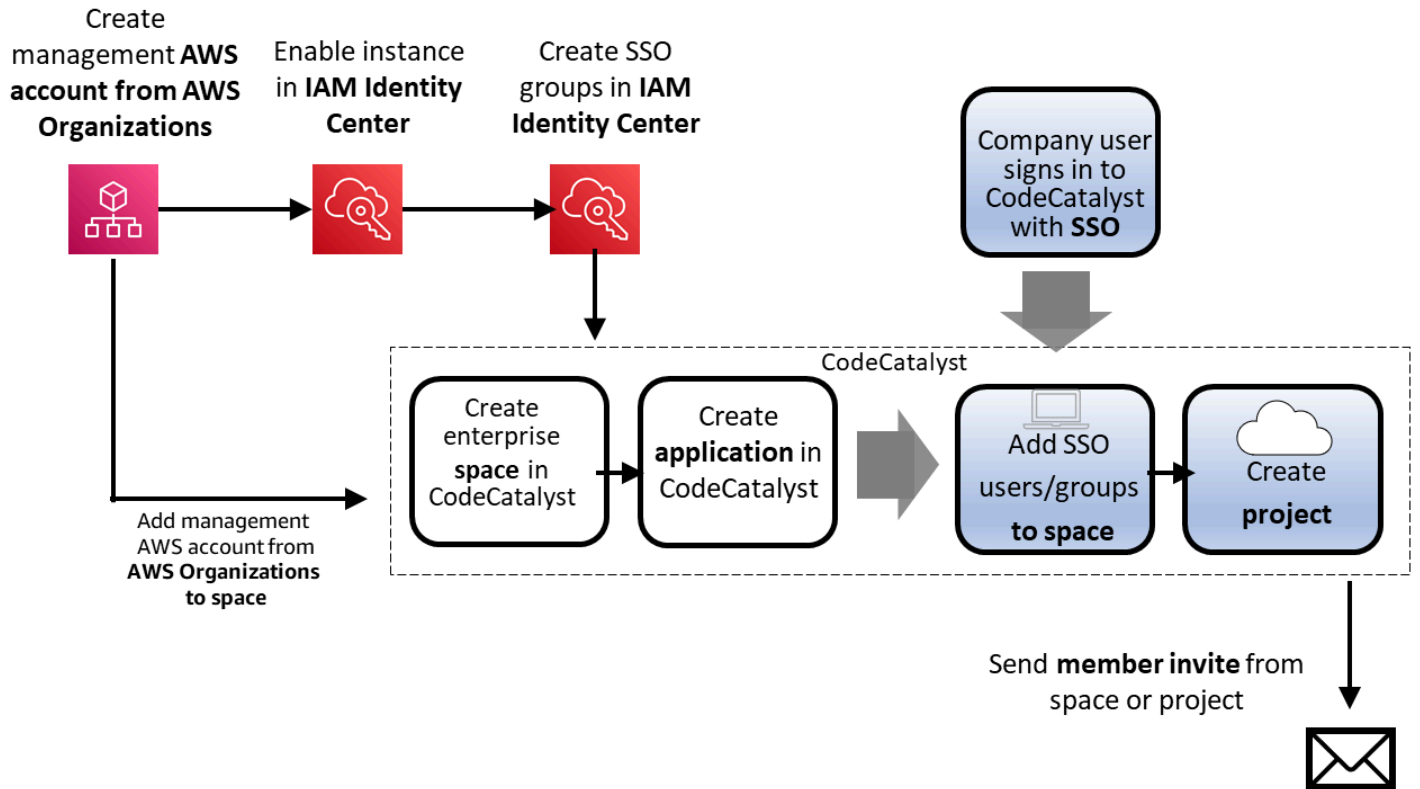
How do spaces work with identity federation and SSO

Companies who want to set up a space for management through identity federation must create a space that supports SSO users and groups.

This guide helps you use CodeCatalyst to perform administration tasks for spaces in CodeCatalyst that support setting up identity federation, managing members, connecting a space to an identity provider and Identity Center application for identity federation, connecting spaces to AWS accounts for billing and access to resources in workflows, or setting user privileges.

The space creation is set up by integrating with AWS tools that support identity federation and SAML configuration. A large company that has a user directory managed in an identity store will want to use these services and tools to allow their users to sign in and be authorized and managed as SSO users with an identity provider such as IAM Identity Center. CodeCatalyst users of the space will be able to use SSO to sign in to CodeCatalyst. This means that the users do not need to sign up individually for CodeCatalyst. The users are added to the identity store and then an email is issued from the space. The users receive an invitation and then go to CodeCatalyst to sign in with their corporate sign-on to a portal. Once the users sign in, they can view the CodeCatalyst space and project in the space where they have access. Users can work in the space, and if they have the **Space administrator** role, they can manage the space and view all projects in the space.

You cannot directly add or remove users in your space in CodeCatalyst. You must work with your Identity federation administrator to manage SSO users and groups in IAM Identity Center. CodeCatalyst syncs on a regular basis with the IAM Identity Center identity store with the latest directory status for your space members.



Setting up a space

Before you start using a space and projects in CodeCatalyst, you must first set up your space and resources in CodeCatalyst.

You can manage your space that supports identity federation in CodeCatalyst after you have set up the space for identity federation.

This guide includes information about administrator tasks for managing spaces and related accounts and federation resources in CodeCatalyst.

For information about the following tasks for managing AWS Builder ID spaces, see the *CodeCatalyst User Guide* :

- Add other space administrators to the space for a AWS Builder ID space
- Change member roles and permissions for a AWS Builder ID space
- Create projects and add members to the project
- View a list of all projects in the space
- View the activity feed for all projects in the space
- Invite users for a AWS Builder ID space

For the steps to set up a CodeCatalyst space without identity federation, an AWS Builder ID space, see [Setting up CodeCatalyst](#) in the *Amazon CodeCatalyst User Guide*.

If you have not already set up a management account in AWS Organizations and set up your identity provider in IAM Identity Center, complete the prerequisites and create your first space as detailed in [Prerequisite 3: Setting up identity federation in IAM Identity Center](#) and [Setting up a space that supports identity federation](#). You work with your Identity federation administrator and AWS account administrator to configure and enable your identity provider (IdP).

Topics

- [Setting up a space that supports AWS Builder ID users](#)
- [Setting up a space that supports identity federation](#)

Setting up a space that supports AWS Builder ID users

You can create a space that manages users with AWS Builder ID access to CodeCatalyst. This is a CodeCatalyst space for AWS Builder ID users.

To get started, you need the following:

- Your AWS Builder ID to sign in to CodeCatalyst
- Your AWS account to provide for the space access to resources and for billing if you go beyond the Free tier.

Before you begin, you must be ready to provide an AWS account ID for an account where you have administrative privileges. Have your 12-digit AWS account ID ready. For information about finding your AWS account ID, see [Your AWS account ID and its alias](#).

To create a space that supports AWS Builder ID users, you start by signing in to CodeCatalyst with your AWS Builder ID. When you create a space that supports AWS Builder ID users, you have the **Space administrator** role by default.

To sign up as a new user

1. Before you start in the CodeCatalyst console, open the AWS Management Console, and then make sure you are signed in with the same AWS account that you want to use to create your space.
2. Open the CodeCatalyst console at <https://codecatalyst.aws/>.
3. On the welcome page, choose **Sign up**. The **Create your AWS Builder ID** page displays. Your AWS Builder ID is an identity you create to sign in. It is not the same as an AWS account.
4. In **Your email address**, enter the email address you want to associate with CodeCatalyst. Then choose **Next**.
5. In **Your name**, provide the first and last name you want displayed in applications where you use your AWS Builder ID. Spaces are allowed. This will be your AWS Builder ID profile name, such as **Mary Major**. You can change the name later.

Choose **Next**. The **Email verification** page displays.

6. A verification code will be sent to the email you specified. Enter this code in **Verification code**, and then choose **Verify**. If you don't receive your code after 5 minutes and cannot find it in your spam or junk folders, then choose **Resend code**.

7. Once we verify your code, enter a password that meets the requirements in **Password** and **Confirm password**.

Select the checkbox confirming your agreement with the AWS Customer Agreement and the AWS Service Terms, and then choose **Create AWS Builder ID**.

8. On the **Create your CodeCatalyst alias** page, enter an alias you want to use for your unique user identifier in CodeCatalyst. Choose a shortened version of your name with no spaces, such as **MaryMajor**. Other CodeCatalyst users will use this to @mention you in comments and pull requests. Your CodeCatalyst profile will contain both your full name from your AWS Builder ID and your CodeCatalyst alias. You cannot change your CodeCatalyst alias later.

Your full name and your alias will display in different areas in CodeCatalyst. For example, your profile name displays for your listed activity in the activity feed, but project members will use your alias to @mention you.

Choose **Next**. The page updates to show the **Create your CodeCatalyst space** section.

9. In **Name your space**, enter the name of your space. You cannot change this later.

 **Note**

Space names must be unique across CodeCatalyst. You cannot reuse names of deleted spaces.

Choose **Next**. The page updates to show the page for adding an AWS account. This account will be used as the billing account for the space.

10. In **AWS Region**, choose the Region where you want to store your space and project data. You cannot change this later.
11. In **AWS account ID**, enter the twelve-digit ID for the account you want to connect to your space.

In **AWS account verification token**, copy the generated token ID. The token is automatically copied for you, but you might want to store it while you approve the AWS connection request.

12. Choose **Verify in AWS**.
13. The **Verify Amazon CodeCatalyst space** page opens in the AWS Management Console. This is the **Amazon CodeCatalyst Spaces** page. You might need to sign in to access the page.

In the AWS Management Console, make sure to choose the same AWS Region where you want to create your space.

To directly access the page, sign in to the Amazon CodeCatalyst Spaces in the AWS Management Console at <https://console.aws.amazon.com/codecatalyst/home/>.

The verification token field in the AWS Management Console is automatically populated with the token generated in CodeCatalyst.


14. Choose **Verify space**.

An **Account verified** success message displays to show that the account has been added to the space.

15. Remain on the **Verify Amazon CodeCatalyst space** page. Choose the following link: **To enable the Standard tier or add IAM roles for this space, view space details**.

The **CodeCatalyst space details** page opens in the AWS Management Console. This is the **Amazon CodeCatalyst Spaces** page. You might need to log in to access the page.


16. To turn on the Standard tier so that the space for the AWS account can update its billing tier, choose **Edit**. Choose the **STANDARD** option, and then choose **Update**.

 **Note**

This does not upgrade the billing tier to the Standard tier. However, this configures the AWS account so that you can change the billing tier for your space at any time in CodeCatalyst. You can turn on the Standard tier at any time. Without making this change, the space is only able to use the Free tier.

17. Return to the CodeCatalyst page, and then choose **Next**.

18. A status message displays while your space is being created. When the space is created, CodeCatalyst opens the page for your space. The view defaults to the **Projects** tab.

 **Note**

If a permissions error or banner is shown, then refresh the page and try to view the page again.

19. (Optional) Proceed to creating a project for your space.

To create and add the CodeCatalyst CodeCatalystWorkflowDevelopmentRole-*spaceName*

1. Before you start in the CodeCatalyst console, open the AWS Management Console, and then make sure you are logged in with the same AWS account for your space.
2. Open the CodeCatalyst console at <https://codecatalyst.aws/>.
3. Navigate to your CodeCatalyst space. Choose **Settings**, and then choose **AWS accounts**.
4. Choose the link for the AWS account where you want to create the role. The **AWS account details** page displays.
5. Choose **Manage roles from AWS Management Console**.

The **Add IAM role to Amazon CodeCatalyst space** page opens in the AWS Management Console. This is the **Amazon CodeCatalyst Spaces** page. You might need to log in to access the page.

6. Choose **Create CodeCatalyst development administrator role in IAM**. This option creates a service role that contains the permissions policy and trust policy for the development role. The role will have a name CodeCatalystWorkflowDevelopmentRole-*spaceName*.

Note

This role is only recommended for use with developer accounts and uses the AdministratorAccess AWS managed policy, giving it full access to create new policies and resources in this AWS account.

7. Choose **Create development role**.
8. On the connection page, under **IAM roles available to CodeCatalyst**, view the CodeCatalystWorkflowDevelopmentRole-*spaceName* role in the list of IAM roles added to your account.
9. To return to your space, choose **Go to Amazon CodeCatalyst**.

To create and add the CodeCatalyst AWSRoleForCodeCatalystSupport

1. Before you start in the CodeCatalyst console, open the AWS Management Console, and then make sure you are logged in with the same AWS account for your space.
2. Navigate to your CodeCatalyst space. Choose **Settings**, and then choose **AWS accounts**.
3. Choose the link for the AWS account where you want to create the role. The **AWS account details** page displays.

4. Choose **Manage roles from AWS Management Console**.

The **Add IAM role to Amazon CodeCatalyst space** page opens in the AWS Management Console. This is the **Amazon CodeCatalyst Spaces** page. You might need to sign in to access the page.

5. Under **CodeCatalyst space details**, choose **Add CodeCatalyst Support role**. This option creates a service role that contains the permissions policy and trust policy for the preview development role. The role will have a name **AWSRoleForCodeCatalystSupport** with a unique identifier appended. .
6. On the **Add role for CodeCatalyst Support** page, leave the default selected, and then choose **Create role**.
7. Under **IAM roles available to CodeCatalyst**, view the **CodeCatalystWorkflowDevelopmentRole-*spaceName*** role in the list of IAM roles added to your account.
8. To return to your space, choose **Go to Amazon CodeCatalyst**.

After you create your AWS Builder ID, create your first space, and add an account, you can then create a project. .

Setting up a space that supports identity federation

You can create a space that supports either of two types of users in CodeCatalyst.

You can create a space that manages users with AWS Builder ID access to CodeCatalyst. This is a CodeCatalyst space for AWS Builder ID users.

Setting up a space in CodeCatalyst includes creating the space, adding users, and assigning CodeCatalyst roles to space members. To set up a AWS Builder ID space and create your first project, use the steps in [Sign up to create your first space and your development role](#) in the *Amazon CodeCatalyst User Guide*.

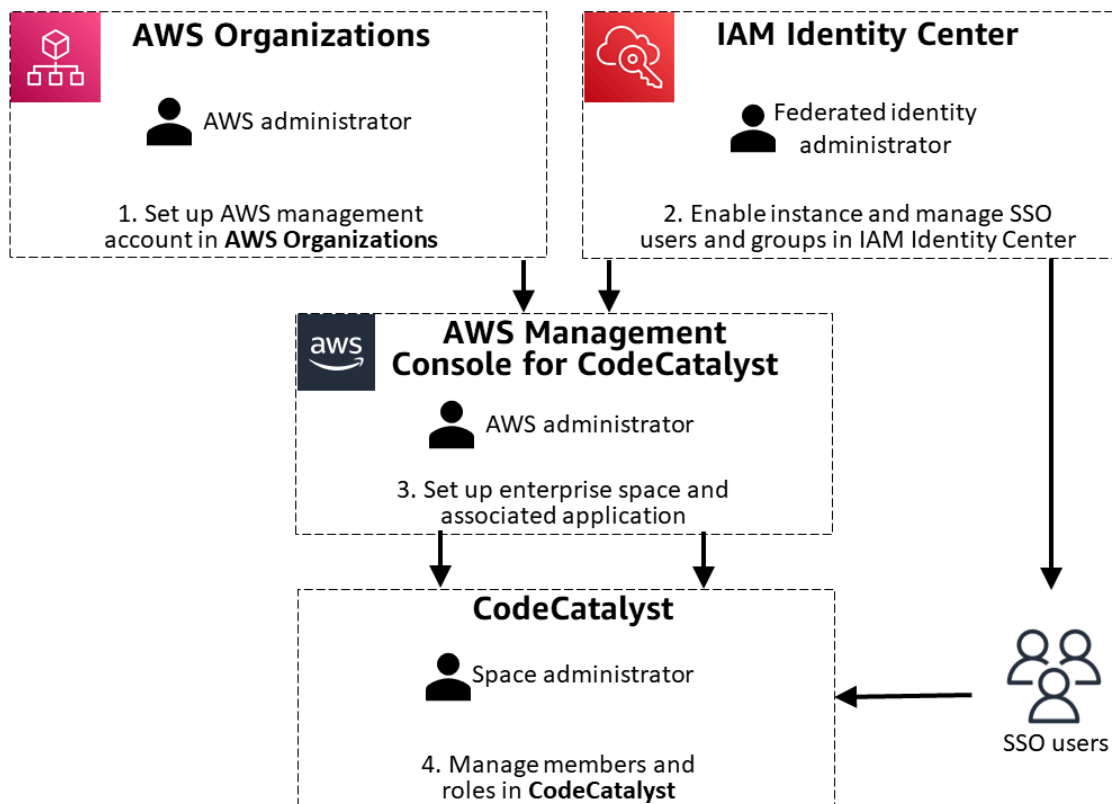
To set up a space that supports identity federation, you must configure prerequisites in the following services before you create or connect a CodeCatalyst space. Use the planning steps in the following section to help you with planning your space.

After completing the prerequisites here, to continue creating your space, see [Creating a space for identity federation](#).

The organization administrator creates the management accounts and Organizational Units for the company in AWS Organizations. After the management account is available, the company identity federation administrator works with IAM Identity Center to enable a provider instance where federated identities will be managed.

The company directory will authorize users to federate through SAML with IAM Identity Center as the SSO provider. Mary coordinates with the company identity federation administrator to set up the users and groups in IAM Identity Center. After this is complete, Mary uses the CodeCatalyst page in the AWS console to create or choose a CodeCatalyst space that will support identity federation. As part of the setup process, Mary creates an SSO application to represent the company and map to the identity store ID in IAM Identity Center. Next, Mary uses the CodeCatalyst page in the AWS Management Console to choose one or more groups to which to grant single sign-on access and to add CodeCatalyst roles. Next, Mary wants to create a team for the space. She incorporates the SSO group into a new team on the **Teams** page in CodeCatalyst.

The following diagram illustrates the flow of tasks for setting up your space.



Topics

- [Planning your space that supports identity federation](#)
- [Prerequisite 1: Setting up an organization in AWS Organizations](#)

- [Prerequisite 2: Enable an instance for identity federation](#)
- [Prerequisite 3: Setting up identity federation in IAM Identity Center](#)

Planning your space that supports identity federation

After you complete the prerequisites here for AWS Organizations and IAM Identity Center, you will use the CodeCatalyst page in the AWS Management Console to create or choose a space and associate it with identity federation. This makes the space a space that supports SSO users and groups. A space that supports identity federation can only manage members through the membership in IAM Identity Center SSO users and groups. These users are maintained in an identity store. When you use the wizard in CodeCatalyst to create or choose a space and enable it for identity federation, you will create an application and give it a name associated with your company.

Important

Dev Environments aren't available for users in spaces where Active Directory is used as the identity provider. When planning a space where the identity provider will be Active Directory, note that users will not be able to use Dev Environments. For more information, see [I can't create a Dev Environment when I'm signed into CodeCatalyst using a single sign-on account](#).

An *Identity Center application* is an association between your CodeCatalyst space and IAM Identity Center. The Identity Center application allows users from your company directory to sign in to CodeCatalyst, so your application name will represent your company and will be visible for selection as an option where users from a workforce directory will access CodeCatalyst. As part of creating a space that supports identity federation, you will choose or create the Identity Center application that will be associated with your space. You can associate multiple spaces with a single Identity Center application. When setting up the Identity Center application for CodeCatalyst, note that the application name must be unique across CodeCatalyst and your IAM Identity Center instances. This uniqueness requirement helps prevent confusion and ensures proper identification of different applications. This unique name is primarily for administrative purposes within IAM Identity Center and doesn't affect the functionality of CodeCatalyst.

Note

The name for your Identity Center application must be globally unique. In addition, since the name will be viewable for signing in and on certain pages in CodeCatalyst, choose a name that will suitably relate to your company for users signing in.

Mary Major is a **Space administrator** who will set up an organization in AWS Organizations that is associated with your company. Mary will need to have the AWS account that is set up a management account for your organization and associate it with the company that will use the space. Additional Organizational Units (OUs) and accounts will be set up for your organization for use with the space.

Mary will work with the Identity federation administrator to set up the directory of users for the company. These are the federated identities that will be set up for the IAM Identity Center instance. The users in the company directory will be set up to sign in to the space using SSO. These users will be defined by the SSO users and groups that the Identity federation administrator sets up in the IAM Identity Center instance.

Note

Users or groups that are added to IAM Identity Center assignments usually appear in CodeCatalyst within two hours. Depending on the amount of data being synchronized, this process might take longer.

Prerequisite 1: Setting up an organization in AWS Organizations

Before you create a space and configure your user membership, complete the following prerequisites for organization and identity federation setup in AWS. You can follow the references in this chapter to get set up.

The AWS Organizations administrator sets up the organization and AWS accounts for your company. The management account for the organization will be specified as the billing account for the space in CodeCatalyst. For more information, see [What is AWS Organizations?](#)

Note

Depending on the type of instance in IAM Identity Center you plan to use in IAM Identity Center, you can choose to create an organization instance or an account instance. If you choose an account instance, then the step to create an organization in AWS Organizations is optional. Choose the instance that best fits your use case. For more information about use cases, see [When to use an organization instance](#) and [When to use an account instance](#) in the *IAM Identity Center User Guide*.

Topics

- [Create an organization in AWS Organizations](#)
- [Add member accounts in AWS Organizations](#)
- [Create organizational units \(OUs\) in AWS Organizations](#)

Create an organization in AWS Organizations

Create an organization for your company. Create an organization with your current AWS account as the management account in AWS Organizations. See the steps in [Creating an organization](#).

Add member accounts in AWS Organizations

Add member accounts to your organization. See the steps in [Add an AWS account to join your organization](#).

Create organizational units (OUs) in AWS Organizations

Create organizational units (OUs) and add member accounts for those OUs in AWS Organizations. See the steps in [Managing organizational units](#).

Prerequisite 2: Enable an instance for identity federation

Before you create a space that supports identity federation, complete the following prerequisites for enabling your instance. You can follow the references in this chapter to get set up.

Note

For IAM Identity Center resources, choose the same Region as your CodeCatalyst space. While you can choose a different Region, this might impact connectivity and latency.

Topics

- [Enable IAM Identity Center](#)

Enable IAM Identity Center

You must have identity federation administrator permissions and the appropriate IAM permissions in IAM Identity Center to complete these steps. Use the steps to enable an AWS IAM Identity Center instance in IAM Identity Center. For more information about this step in IAM Identity Center, see [Step 1: Enable instance](#).

1. Sign in to the IAM Identity Center console.
2. On the welcome page, choose **Enable AWS SSO**. A success banner displays.

Prerequisite 3: Setting up identity federation in IAM Identity Center

IAM Identity Center helps you securely create or connect your identity federation and manage user access centrally across AWS accounts and applications. IAM Identity Center is the recommended approach for identity federation on AWS for organizations of any size and type.

Note

For IAM Identity Center resources, choose the same Region as your CodeCatalyst space. While you can choose a different Region, this might impact connectivity and latency.

The Identity federation administrator sets up the IAM Identity Center instance and SSO users and groups for the company. This represents the identity provider (IdP) for the company. As the identity federation administrator for your company, complete the following tasks in IAM Identity Center.

In IAM Identity Center, you can choose to create an organization instance or an account instance. Choose the type that best fits your use case. For more information about use cases, see [When to](#)

[use an organization instance](#) and [When to use an account instance](#) in the *IAM Identity Center User Guide*.

Topics

- [Set up your provider in IAM Identity Center](#)
- [Set up your portal in IAM Identity Center](#)
- [Create users and groups in IAM Identity Center](#)

Set up your provider in IAM Identity Center

Connect your recently created or existing IAM Identity Center instance to your IdP.

Important

Dev Environments aren't available for users in spaces where Active Directory is used as the identity provider. When planning a space where the identity provider will be Active Directory, note that users will not be able to use Dev Environments. For more information, see [I can't create a Dev Environment when I'm signed into CodeCatalyst using a single sign-on account](#).

Set up your provider in IAM Identity Center. See the steps in [What is IAM Identity Center?](#)

Note

CodeCatalyst spaces with identity federation can support service providers that are supported by IAM Identity Center. CodeCatalyst inherits the identity source that is managed in IAM Identity Center. For more information, see [Manage your identity source](#).

Set up your portal in IAM Identity Center

Connect your recently created or existing instance in IAM Identity Center to your IdP.

Create an SSO portal login for your provider in IAM Identity Center. See [Manage sign-in and attribute use for all identity source types](#).

An *Identity Center application* is an association between your CodeCatalyst space and IAM Identity Center. The Identity Center application allows users from your company directory to sign in

to CodeCatalyst, so your application name will represent your company and will be visible for selection as an option where users from a workforce directory will access CodeCatalyst. As part of creating a space that supports identity federation, you will choose or create the Identity Center application that will be associated with your space. You can associate multiple spaces with a single Identity Center application. When setting up the Identity Center application for CodeCatalyst, note that the application name must be unique across CodeCatalyst and your IAM Identity Center instances. This uniqueness requirement helps prevent confusion and ensures proper identification of different applications. This unique name is primarily for administrative purposes within IAM Identity Center and doesn't affect the functionality of CodeCatalyst.

Note

The name for your Identity Center application must be globally unique. In addition, since the name will be viewable for signing in and on certain pages in CodeCatalyst, choose a name that will suitably relate to your company for users signing in.

Create users and groups in IAM Identity Center

You must create users and groups in IAM Identity Center that you will manage in IAM Identity Center and then specify in CodeCatalyst when you create or view your space. Create and connect groups in IAM Identity Center.

You must have Identity federation administrator permissions and the appropriate IAM permissions in IAM Identity Center to complete these steps. Use the steps to create users and groups who will be the directory users for your space. For more information about this step in IAM Identity Center, see [Manage identities in IAM Identity Center](#).

Note

CodeCatalyst user names have a minimum length of 3 and a maximum length of 100 characters. Provided user names longer than 100 characters will be truncated. This can result in a user name that appears to be a duplicate of another 100-character user name. For more information, see [I can't access my AWS Builder ID space as a new user or can't be added as a new SSO user due to truncated user name](#).

Inviting users to a Builder ID space

Note

This topic describes how to invite users who sign in to CodeCatalyst with a AWS Builder ID. To invite users that are managed as part of SSO users and groups, see [Administering spaces that support identity federation](#).

You can invite users to your CodeCatalyst space (instead of inviting them to a project within a space). Inviting users to a space is useful when you want that user to help you manage the space. Users who will manage a space will also need the **Space administrator** or **Power user** role. Assigning one of those roles to other users can help you distribute the responsibilities of managing the space across more people without having to invite these users to projects.

Note

You must have the **Space administrator** or **Power user** role to invite members.

The **Space administrators** table shows users with the **Space administrator** role. These users are not shown in the **Space members** table because they are automatically (implicitly) assigned to all projects in the space and do not have a role in a project.

Members who accept a project invitation are added to the space by default. The **Project members** table shows all members in the space that have a role in a project.

To invite a user to your space

1. Open the CodeCatalyst console at <https://codecatalyst.aws/>.
2. Navigate to your space.
3. Choose **Settings**, and then choose **Members**.
4. Choose **Invite**.
5. Enter the email of the person you would like to invite to join your space. In **Role**, choose the role you want to assign that user in the space.
6. Choose **Invite**

Creating additional spaces that support AWS Builder ID users

Note

This topic describes how to create spaces that support users who sign in to CodeCatalyst with a AWS Builder ID. To create spaces that support users who are managed as part of SSO users and groups, see [Administering spaces that support identity federation](#).

When you first sign up in Amazon CodeCatalyst, you are required to create a space. You can choose to create additional spaces to meet your business needs.

Note

Space names must be unique across CodeCatalyst. You cannot reuse names of deleted spaces.

To create additional spaces, you must be assigned the Space administrator role.

Note

When you create an additional space, you are not prompted to create a project. .

To create another space

1. In the AWS Management Console, make sure you are signed in with the same AWS account that you want to associate with your CodeCatalyst space.
2. Open the CodeCatalyst console at <https://codecatalyst.aws/>.
3. Navigate to your space.

Tip

If you belong to more than one space, choose a space in the top navigation bar.

4. Choose **Create space**.
5. On the **Create a space** page, in **Space name**, enter a name for the space. You cannot change this later.

 **Note**

Space names must be unique across CodeCatalyst. You cannot reuse names of deleted spaces.

6. In **AWS Region**, choose the Region where you want to store your space and project data. You cannot change this later.
7. In **AWS account ID**, enter the twelve-digit ID for the account you want to connect to your space.

In **AWS account verification token**, copy the generated token ID. The token is automatically copied for you, but you might want to store it while you approve the AWS connection request.

8. Choose **Verify in AWS**.
9. The **Verify Amazon CodeCatalyst space** page opens in the AWS Management Console. This is the **Amazon CodeCatalyst Spaces** page. You might need to sign in to access the page.

In the AWS Management Console, make sure to choose the same AWS Region where you want to create your space.

To directly access the page, sign in to the Amazon CodeCatalyst Spaces in the AWS Management Console at <https://console.aws.amazon.com/codecatalyst/home/>.

The verification token is automatically entered in **Verification token**. A success banner shows a message that the token is a valid token.

10. Choose **Verify space**.

An **Account verified** success message displays to show that the account has been added to the space.

11. Remain on the **Verify Amazon CodeCatalyst space** page. Choose the following link: **To enable the Standard tier or add IAM roles this space, view space details**.

The **CodeCatalyst space details** page opens in the AWS Management Console. This is the **Amazon CodeCatalyst Spaces** page. You might need to log in to access the page.

12. Under **IAM roles available to CodeCatalyst**, choose **Add IAM role**.

The **Add IAM roles available to CodeCatalyst** page displays.

13. Choose **Create CodeCatalyst development administrator role in IAM**. This option creates a service role that contains the permissions policy and trust policy for the development role.

The developer role is an AWS IAM role that enables your CodeCatalyst workflows to access AWS resources such as Amazon S3, Lambda, and AWS CloudFormation. The role will have a name `CodeCatalystWorkflowDevelopmentRole-spaceName` with a unique identifier appended.

14. Choose **Create development role**.
15. On the connection page, under **IAM roles available to CodeCatalyst**, view the developer role in the list of IAM roles added to your account.
16. Choose **Go to Amazon CodeCatalyst**.
17. On the creation page in CodeCatalyst, choose **Create space**.

Deleting a space that supports AWS Builder ID users

Note

This topic describes how to delete spaces that support users who sign in to CodeCatalyst with a AWS Builder ID. To delete spaces that support users who are managed as part of SSO users and groups, see [Administering spaces that support identity federation](#).

You can delete a space to remove access to all of the space's resources. You must have the **Space administrator** role to delete a space.

Note

You cannot undo a space deletion.

After you have deleted a space, all space members will be unable to access space resources. Billing for space resources will also stop, and any workflows that are prompted by third-party source repositories will be stopped.

Note

Space names must be unique across CodeCatalyst. You cannot reuse names of deleted spaces.

To delete a space


1. Open the CodeCatalyst console at <https://codecatalyst.aws/>.
2. Navigate to your space.

Tip

If you belong to more than one space, choose a space in the top navigation bar.

3. Choose **Settings**, and then choose **Delete**.

4. Type **delete** to confirm the deletion.
5. Choose **Delete**.

 **Note**

If you belong to more than one space, you're redirected to the space overview page. If you belong to one space, you're redirected to the space creation page.

Administering spaces that support identity federation

You can manage your space in CodeCatalyst after you have set up the space for identity federation.

This guide includes information about administrator tasks for managing spaces in CodeCatalyst that support identity federation.

For information about the following tasks for managing AWS Builder ID spaces, see the *CodeCatalyst User Guide*:

- Add other space administrators to the space for an AWS Builder ID space
- Change member roles and permissions for an AWS Builder ID space
- Create projects and add members to the project
- View a list of all projects in the space
- View the activity feed for all projects in the space
- Invite users for a AWS Builder ID space

For the steps to set up a CodeCatalyst space without identity federation, an AWS Builder ID space, see [Setting up CodeCatalyst](#) in the *Amazon CodeCatalyst User Guide*.

If you have not already connected the AWS account that will be the specified billing account for your space and set up your identity provider in IAM Identity Center, complete the prerequisites and create your first space as detailed in [Prerequisite 3: Setting up identity federation in IAM Identity Center](#) and [Creating a space for identity federation](#). You work with your Identity federation administrator and AWS account administrator to configure and enable your identity provider (IdP). The AWS account that is specified as the billing account for your CodeCatalyst space has different quotas from other account connections for a space. For more information, see [Quotas](#) in the *CodeCatalyst User Guide*.

Important

Dev Environments aren't available for users in spaces where Active Directory is used as the identity provider. When planning a space where the identity provider will be Active Directory, note that users will not be able to use Dev Environments. For more information, see [I can't create a Dev Environment when I'm signed into CodeCatalyst using a single sign-on account](#).

Creating a space for identity federation

You cannot directly add or remove users in your space that supports identity federation. You must work with your Identity federation administrator to manage SSO users and groups in IAM Identity Center. CodeCatalyst syncs on a regular basis with the IAM Identity Center identity store with the latest directory status for your space members.

Before you start to set up your space, make sure you are signed in to the AWS Management Console with the AWS account that will be the specified billing account for your space.

Before you begin

- Before you begin, you must be ready to provide an AWS account ID for an account where you have administrative privileges as the billing account for your space. Have your 12-digit AWS account ID ready. For information about finding your AWS account ID, see [Your AWS account ID and its alias](#).

You must have completed the prerequisites as follows:

1. Create an organization in AWS Organizations (not required).
2. Set up your billing account or your account in AWS Organizations.
3. Enable IAM Identity Center.
4. Set up your provider in IAM Identity Center.
5. Create users and groups in IAM Identity Center.

To create a space for identity federation

1. Sign in to the Amazon CodeCatalyst page in the AWS Management Console with the AWS account that will be the specified billing account for your space.
2. Open the Amazon CodeCatalyst page in the AWS Management Console at https://us-west-2.console.aws.amazon.com/codecatalyst/home?region=us-west-2#.
3. Choose **IAM Identity Center**. On the **IAM Identity Center** page, under **Application Enabled Spaces**, choose **Connect**.

Tip

Make sure you are signed in to the AWS Management Console with the AWS account that will be the specified billing account for your space.

4. In **AWS Region**, choose the Region for your space. Make sure to choose the same Region as that where your identity resources are created.

Note

For IAM Identity Center resources, choose the same Region as your CodeCatalyst space. While you can choose a different Region, this might impact connectivity and latency.

5. Under **Step 1: Choose application name**, in **Display name**, enter a name that will match your company name for display on login screens and in CodeCatalyst.

Note


Identity Center application names must be globally unique.

Important

Your application name will represent your company and will be visible for selection as an option where users from a workforce directory will access CodeCatalyst.

6. In **AWS Identity Center application name**, provide the name to use when signing in to CodeCatalyst with SSO. This is the name that will represent your company association between your identity provider and your CodeCatalyst space. When you create an application, it is associated with your identity store ID in IAM Identity Center.
7. In **Identity store ID**, the ID for the associated identity store in IAM Identity Center displays. To change this, choose **go to IAM Identity Center**.
8. Choose **Next**.
9. Under **Step 2: Choose or create a CodeCatalyst space**, do one of the following:


- To set up an existing CodeCatalyst to support identity federation and create an application for it, choose **Existing CodeCatalyst**. In the dropdown field for **Choose existing CodeCatalyst**, choose the existing CodeCatalyst space you want to set up.

 **Note**

If you set up an existing space by adding SSO support, only SSO users and groups will be supported. Existing AWS Builder ID users will no longer be supported. This action cannot be undone, and you can't change the space back to an AWS Builder ID space later.

- To set up a new CodeCatalyst, choose **New space**.

In **Space name**, enter a name for your CodeCatalyst space.

 **Note**

Space names must be unique across CodeCatalyst. You cannot reuse names of deleted spaces.

10. Choose **Next**.
11. Under **Step 3: Connect groups**, in **Choose groups**, choose the SSO users and groups you want to add to the space. Choose the box next to each group you want to add. These must be already available in IAM Identity Center for your identity provider.
12. Choose **Next**.
13. Under **Step 4: Assign users to the CodeCatalyst Space administrator role**, choose which users you want to assign the **Space administrator** role. These users will have **Space administrator** permissions in CodeCatalyst for your space, to include removing members and deleting the space. For more information about the role, see [Working with roles in Amazon CodeCatalyst](#).
14. Choose **Next**.
15. In the wizard Step 5 page, review the summary for the space.

 **Note**

Make sure you are ready to create the space with the space name you have chosen. Once you create the space, you will not be able to reuse the space name, even if the

space is deleted. SSO application names can be reassigned to another space, but the space name itself cannot be reused.

Next steps: Create teams, projects, and resources in CodeCatalyst

After you have created your space, you can perform the following tasks.

- Create projects from blueprints in CodeCatalyst. For a tutorial, see [Tutorial: Creating a project with the Modern three-tier web application blueprint](#) in the *Amazon CodeCatalyst User Guide*.
- Work with workflows, repositories, and other resources in CodeCatalyst. See [Build, test, and deploy with workflows in CodeCatalyst](#) in the *Amazon CodeCatalyst User Guide*.
- Create teams in CodeCatalyst and import your users and groups into teams. See [Managing teams](#) in the *Amazon CodeCatalyst User Guide*.
- Create issues and assign tasks to users and teams in CodeCatalyst. See [Issues in CodeCatalyst](#) in the *Amazon CodeCatalyst User Guide*.

Configure an existing CodeCatalyst space for identity federation

You must have the **Space administrator** role and access to the billing account for the space in order to view SSO users and groups for your space.

You must have completed the prerequisites in AWS Organizations and IAM Identity Center for a space. The space can only support members that are managed as federated identities in IAM Identity Center.

You cannot directly add or remove users in your space in CodeCatalyst. You must work with your Identity federation administrator to manage SSO users and groups in IAM Identity Center. CodeCatalyst syncs with the IAM Identity Center on a regular basis to update your space members.

Important

After a space is updated to SSO and associated with an Identity Center application, it is an enabled space in CodeCatalyst for SSO. The space will no longer support AWS Builder

ID users. This action cannot be undone, and you can't change the space back to an AWS Builder ID space later.

1. Open the CodeCatalyst console at <https://codecatalyst.aws/>.
2. Choose **Settings**, and then choose **SSO**.
3. On the **SSO not enabled** message, choose **Set up in AWS**. The wizard page opens for creating a space. To complete the wizard, see the steps in [Creating a space for identity federation](#).

To view information in IAM Identity Center, choose **IAM Identity Center**. You will be taken to IAM Identity Center, where you can work with your Identity federation administrator to configure SSO users and groups for your instance in IAM Identity Center.

Viewing SSO users and groups for a space

You must have the **Space administrator** role and access to the billing account for your space to view SSO users and groups for your space. You cannot directly add or remove SSO users or groups in CodeCatalyst.

Note

Users or groups that are added to IAM Identity Center assignments usually appear in CodeCatalyst within two hours. Depending on the amount of data being synchronized, this process might take longer.

1. Open the CodeCatalyst console at <https://codecatalyst.aws/>.
2. To view users in each group, choose the group. To view application details in the AWS Management Console, choose **View application**.

To view information in IAM Identity Center, choose **IAM Identity Center**. You will be taken to IAM Identity Center, where you can work with your Identity federation administrator to configure SSO users and groups for your instance in IAM Identity Center.

Adding SSO groups to a space that supports identity federation

You can use the Amazon CodeCatalyst page in the AWS Management Console to add SSO groups to your space. You must have already worked with your Identity federation administrator to create the SSO users and groups for your instance in IAM Identity Center. For a high-level reference to the prerequisites steps to configure SSO users and groups in IAM Identity Center, see [Prerequisite 3: Setting up identity federation in IAM Identity Center](#).

Note

Users or groups that are added to IAM Identity Center assignments usually appear in CodeCatalyst within two hours. Depending on the amount of data being synchronized, this process might take longer.

You must have the **Space administrator** role and access to the billing account for your space to view SSO users and groups for your space.

You cannot directly add or remove users to your space in CodeCatalyst. You must work with your Identity federation administrator to manage SSO users and groups in IAM Identity Center. CodeCatalyst syncs on a regular basis with the IAM Identity Center identity store with the latest directory status for your space members.

1. Open the Amazon CodeCatalyst page in the AWS Management Console at <https://us-west-2.console.aws.amazon.com/codecatalyst/home?region=us-west-2#/>.
2. Navigate to the page for your space. Choose **Edit SSO**. Choose the SSO groups you want to add to your space.
3. To view more information in IAM Identity Center, choose **IAM Identity Center**.

Adding the Space administrator role to SSO users in a space

You can use the Amazon CodeCatalyst page in the AWS Management Console to assign the **Space administrator** role to individual users in your SSO groups. You cannot directly add or remove users in your space in CodeCatalyst. You must have already worked with your Identity federation administrator to create the SSO users and groups for your instance in IAM Identity Center. CodeCatalyst syncs on a regular basis with the IAM Identity Center identity store with the latest directory status for your space members.

Note

Users or groups that are added to IAM Identity Center assignments usually appear in CodeCatalyst within two hours. Depending on the amount of data being synchronized, this process might take longer.

You must have the **Space administrator** role and access to the billing account for your space to view SSO users and groups for your space.

1. Open the Amazon CodeCatalyst page in the AWS Management Console at [https://us-west-2.console.aws.amazon.com/codecatalyst/home?region=us-west-2#/.](https://us-west-2.console.aws.amazon.com/codecatalyst/home?region=us-west-2#/)
2. Navigate to the page for your space. Choose **Edit IAM Identity Center**.
3. Choose the individual users that you want to grant the **Space administrator** role for your space.
4. To view more information in IAM Identity Center, choose **IAM Identity Center**.

Deleting a space that supports identity federation

You can delete a space that supports identity federation when you no longer need it.

Note

Deleting the space will delete all projects and resources in the space. Deleting the space will remove the associated SSO users and groups from the space.

You must have the **Space administrator** role and access to the billing account for your space to view SSO users and groups for your space.

To delete a space that supports identity federation

1. Open the Amazon CodeCatalyst page in the AWS Management Console at [https://us-west-2.console.aws.amazon.com/codecatalyst/home?region=us-west-2#/.](https://us-west-2.console.aws.amazon.com/codecatalyst/home?region=us-west-2#/)
2. On the page for your space, choose **Delete**.

Tip

Make sure you are signed in to the AWS Management Console with the AWS account that will be the specified billing account for your space.

3. Choose **Delete**.
4. You can retain the previously associated Identity Center application or delete it.

Administering Identity Center applications

An *Identity Center application* is an association between your CodeCatalyst space and IAM Identity Center. The Identity Center application allows users from your company directory to sign in to CodeCatalyst, so your application name will represent your company and will be visible for selection as an option where users from a workforce directory will access CodeCatalyst. As part of creating a space that supports identity federation, you will choose or create the Identity Center application that will be associated with your space. You can associate multiple spaces with a single Identity Center application. When setting up the Identity Center application for CodeCatalyst, note that the application name must be unique across CodeCatalyst and your IAM Identity Center instances. This uniqueness requirement helps prevent confusion and ensures proper identification of different applications. This unique name is primarily for administrative purposes within IAM Identity Center and doesn't affect the functionality of CodeCatalyst.

Note

The name for your Identity Center application must be globally unique. In addition, since the name will be viewable for signing in and on certain pages in CodeCatalyst, choose a name that will suitably relate to your company for users signing in.

You can manage this application and its association with your space in the Amazon CodeCatalyst page in the AWS Management Console. For information about creating your application, see [Creating a space for identity federation](#).

Important

Dev Environments aren't available for users in spaces where Active Directory is used as the identity provider. When planning a space where the identity provider will be Active Directory, note that users will not be able to use Dev Environments. For more information, see [I can't create a Dev Environment when I'm signed in to CodeCatalyst using a single sign-on account](#).

Topic	Description
Viewing Identity Center application details	This topic describes how to view the space name, display name, and application name of your Identity Center application. You can also view the users to whom you have assigned the Space administrator role, and view the SSO groups that you have added to your space.
Editing Identity Center application details	This topic describes how to edit the SSO groups assigned to your space, how to assign additional administrators to your space, and how to make updates to your connected groups.
Associating a space to your Identity Center application	This topic describes how to connect a CodeCatalyst space to an Identity Center application.
Disassociating an Identity Center application from a space	This topic describes how to disconnect a CodeCatalyst space from an Identity Center application. You can reconnect the application later, or associate it with another space.

Viewing Identity Center application details

You can view the details for the space associated with your Identity Center application.

Note

Users or groups that are added to IAM Identity Center assignments usually appear in CodeCatalyst within two hours. Depending on the amount of data being synchronized, this process might take longer.

To view space and Identity Center application details

1. Open the Amazon CodeCatalyst page in the AWS Management Console at [https://us-west-2.console.aws.amazon.com/codecatalyst/home?region=us-west-2#/.](https://us-west-2.console.aws.amazon.com/codecatalyst/home?region=us-west-2#/)
2. Choose **IAM Identity Center**. On the **IAM Identity Center** page, under **Application Enabled Spaces**, view the spaces enabled for SSO and associated with your application.

Tip

Make sure you are signed in to the AWS Management Console with the AWS account that will be the specified billing account for your space.

3. In **Space name**, view the name for your space.
4. In **Display name**, view the name that displays on the sign-in page for your space.
5. In **Application name**, view the name of your Identity Center application.
6. In **Space administrators**, view the users that you have assigned the **Space administrator** role for your space. These are members of the SSO group who have individual permissions in CodeCatalyst.
7. In **Connected groups**, view the SSO groups that you have added to your space. The users in these groups can be viewed in the member lists in your CodeCatalyst space, projects, and teams.
8. To make updates to your connected groups, choose **Edit Identity Center application**. You will be taken to IAM Identity Center where you can work with your Identity federation administrator to configure SSO users and groups for your instance in IAM Identity Center.

Editing Identity Center application details

You can edit the details for your Identity Center application, such as choosing SSO users and groups that are available in IAM Identity Center.

Note

Users or groups that are added to IAM Identity Center assignments usually appear in CodeCatalyst within two hours. Depending on the amount of data being synchronized, this process might take longer.

To edit space and Identity Center application details

1. Open the Amazon CodeCatalyst page in the AWS Management Console at <https://us-west-2.console.aws.amazon.com/codecatalyst/home?region=us-west-2#/>.
2. Choose **IAM Identity Center**. On the **IAM Identity Center** page, under **Application Enabled Spaces**, view the spaces enabled for SSO and associated with your application.

Tip

Make sure you are signed in to the AWS Management Console with the AWS account that will be the specified billing account for your space.

3. Choose your space from the list, and then choose **Edit space**,
4. On the **Edit assigned users and groups** page, view your application details.
5. In **Groups connected to this space**, choose the SSO groups that you want to add to your space. The users in these groups can be viewed in the member lists in your CodeCatalyst space, projects, and teams.
6. Under **Space administrators**, in **Assign additional administrators**, choose the users to which you want to assign the **Space administrator** role for your space. These are members of the SSO group who have individual permissions in CodeCatalyst.
7. To make updates to your connected groups, choose **Manage in IAM Identity Center**. You will be taken to IAM Identity Center where you can work with your Identity federation administrator to configure SSO users and groups for your instance in IAM Identity Center.

Associating a space to your Identity Center application

You can associate a space with your CodeCatalyst Identity Center application. You must have already completed the prerequisites for setting up identity federation in AWS Organizations and IAM Identity Center.

To associate a space to an Identity Center application

1. Open the Amazon CodeCatalyst page in the AWS Management Console at <https://us-west-2.console.aws.amazon.com/codecatalyst/home?region=us-west-2#/>.
2. Choose **IAM Identity Center**. On the **IAM Identity Center** page, under **Application Enabled Spaces**, view the spaces enabled for SSO and associated with your application.

Tip

Make sure you are signed in to the AWS Management Console with the AWS account that will be the specified billing account for your space.

3. Under **Application Enabled Spaces**, choose **Connect space**. On the **Choose or create a CodeCatalyst space** page, choose the space that you want to associate with your application, or you can choose to create a new space.

Disassociating an Identity Center application from a space

You can disassociate the Identity Center application that is associated with your CodeCatalyst space. You can reassociate the Identity Center application later, or you can associate the Identity Center application to another space.

Note


If you delete the identity store in IAM Identity Center, then the Identity Center application is automatically disassociated from the CodeCatalyst space.

1. Open the CodeCatalyst console at <https://codecatalyst.aws/>.
2. Choose **IAM Identity Center**. On the **IAM Identity Center** page, under **Application Enabled Spaces**, view the spaces enabled for SSO and associated with your application.

Tip

Make sure you are signed in to the AWS Management Console with the AWS account that will be the specified billing account for your space.

3. Under **Application Enabled Spaces**, choose the space that you want to disassociate from your application. Choose **Disassociate space from application**.
4. Enter the confirmation for disassociating the application, and then choose **Disassociate**.

 **Important**

This action will remove all SSO users as members in the CodeCatalyst space.

The Identity Center application will be available to be reassociated with this space or another space when needed.

Administering connected accounts

To access resources in AWS related to your projects in CodeCatalyst, you must connect an AWS account to your space. The connected account can also be used as the billing account for the space if you choose to use a paid tier.

To do so, you must set up a connection between the AWS accounts and your space in CodeCatalyst. Creating a connection like this means that projects and workflows within your CodeCatalyst space can interact with resources in your AWS accounts. You must create one connection for each AWS account you want to use with your CodeCatalyst space.

After you create a connection, you can choose to associate AWS IAM roles with it.

Here is one possible flow for adding an AWS account in CodeCatalyst:

Li Juan has the Project administrator role in a CodeCatalyst project with a workflow that builds and deploys the application to AWS infrastructure in the cloud. To deploy to the AWS infrastructure, CodeCatalyst must use an authorized AWS account to access the AWS resources for the build action in the workflow. Li Juan works with Mary Major, who has the **Space administrator** role, and Mateo Jackson, who has AWS administrator permissions in the AWS account to create a connection between the space and the AWS account. Before creating the connection, Mateo Jackson creates an IAM role in that account called `codecatalyst-build-role` with the IAM permissions policy for the AWS Cloud Development Kit (AWS CDK) stack he wants to use to build the application in the AWS account.

As the next step, Mary Major edits the CodeCatalyst space settings, completes an authorization flow with Mateo Jackson, and adds the AWS account and role to the list of AWS accounts and roles available for the CodeCatalyst space. Li Juan uses the CodeCatalyst environments page to add the account and role to the environment for his CodeCatalyst project. Li Juan also adds the role Amazon Resource Name (ARN) to the `Role` field for the CodeCatalyst workflow YAML.

For steps for managing accounts in the CodeCatalyst console, see [Account connections](#) in the CodeCatalyst User Guide.

Topics

- [Adding an account connection for a space \(in AWS\)](#)
- [Removing an account from a space \(in AWS\)](#)

- [Tagging account connections](#)
- [Enabling or disabling project-restricted account connections](#)

Adding an account connection for a space (in AWS)

For steps for managing accounts in the CodeCatalyst console, see [Account connections](#) in the CodeCatalyst User Guide.

For a space that supports AWS Builder ID users, the space requires that you specify a connected account to be the billing account for the space. For a space that supports identity federation, the space billing account will default to the management account associated with the organization in AWS Organizations.

Removing an account from a space (in AWS)

You can use the page for CodeCatalyst in AWS to remove an account that has been added to a space. For this procedure, using administrative permissions for the specific account you are managing, you sign in the Amazon CodeCatalyst Spaces page in the AWS Management Console to remove an AWS account from your space. To remove an account that is a designated billing account for your CodeCatalyst space, make sure to first specify a new billing account.

An account that has been removed can be added again later, but you must create a new connection between the account and the space. You will need to re-associate any IAM roles to the added account.

A billing account must be designated for your CodeCatalyst space, even if usage for the space will not exceed the Free tier. Before you can remove a space for an account that is a designated billing account, you will need to add another account for your space.

Important

While you can use these steps to remove an account, this is not recommended. The account might also be set up to support workflows in CodeCatalyst.

You must have the **Space administrator** or **Power user** role to manage account connections for your space.

To remove an added account

1. In the AWS Management Console, make sure you are logged in with the same account that you want to manage.
2. Open the CodeCatalyst console at <https://codecatalyst.aws/>.
3. Navigate to your CodeCatalyst space. Choose **Settings**, and then choose **Billing**.
4. View the billing account information on the page to make sure the account you want to remove is not the designated billing account for the space.
5. Choose **Manage billing in AWS**. This opens the Amazon CodeCatalyst Spaces in the AWS Management Console. If you're prompted to log in, log in to AWS, and then choose the button again to load the page.
6. On the **Amazon CodeCatalyst Spaces** page, choose the space with the account that you want to remove. The details page for the space displays.
7. Choose **Remove space**.
8. In **Remove CodeCatalyst space from this account**, enter the space name to confirm. Choose **Remove**.

Tagging account connections

Connections are represented by a connection resource Amazon Resource Name (ARN) that is unique to the connection between a specific AWS account and a specific space in CodeCatalyst.

Using tags, you can organize and control access to your resources with tag-based IAM policies.

Topics

- [Adding tags to account connections](#)
- [Editing tags of account connections](#)
- [Removing tags from account connections](#)

Adding tags to account connections

Use these steps to create and apply tags to an account connection.

To add a tag

1. Make sure you're signed in to the AWS Management Console with the AWS account for the account connection you want to manage.
2. If the **Details** page doesn't display, choose **Spaces**. From the list under **Amazon CodeCatalyst space**, choose the space that corresponds with the account connection you want to tag.

The **Amazon CodeCatalyst space details** page displays.

3. Choose **Manage tags**. The **Manage tags** page displays.
4. Choose **Add tag**. Add a tag by entering a key and a value for the key-value pair.
5. Choose **Save changes**.

Editing tags of account connections

Use these steps to create and apply tags for your account connection resources.

To edit a tag

1. Make sure you're signed in to the AWS Management Console with the AWS account for the account connection you want to manage.
2. Open the Amazon CodeCatalyst page in the AWS Management Console at <https://us-west-2.console.aws.amazon.com/codecatalyst/home?region=us-west-2#/>.
3. If the **Details** page doesn't display, from the side navigation, choose **Spaces**. From the list under **Amazon CodeCatalyst space**, choose the space that corresponds with the account connection you want to tag.

The **Amazon CodeCatalyst space details** page displays.

4. Choose **Manage tags**. The **Manage tags** page displays.
5. In the row for the tag you want to edit, change the key pair value by editing the key, the value, or both.
6. Choose **Save changes**.

Removing tags from account connections

Use these steps to remove tags for an account connection.

To remove a tag

1. Make sure you're signed in to the AWS Management Console with the AWS account for the account connection you want to manage.
2. Open the Amazon CodeCatalyst page in the AWS Management Console at [https://us-west-2.console.aws.amazon.com/codecatalyst/home?region=us-west-2#/.](https://us-west-2.console.aws.amazon.com/codecatalyst/home?region=us-west-2#/)
3. If the **Details** page doesn't display, from the side navigation, choose **Spaces**. From the list under **Amazon CodeCatalyst space**, choose the space that corresponds with the account connection you want to tag.

The **Amazon CodeCatalyst space details** page displays.

4. Choose **Manage tags**. The **Manage tags** page displays.
5. For each tag you want to remove, choose **Remove** on the row for that tag. You can remove multiple tags.
6. Choose **Save changes**.

Enabling or disabling project-restricted account connections

The default in CodeCatalyst is to add an AWS account connection to your space that is then made immediately available for all projects and resources in the space. You can configure account connections so that they are restricted to a specified set of projects. This allows you to restrict which projects have access to connected AWS accounts. The access can be restricted for account connections to workflows and VPC connections.

Connections are represented by a connection resource Amazon Resource Name (ARN) that is unique to the connection between a specific AWS account and a specific space in CodeCatalyst. The connection can be specified as restricted. The account connection will not be available for workflows or default VPCs in CodeCatalyst.

Considerations for project-restricted account connections

The following considerations apply to project-restricted account connections.

- You must have the **Space administrator** or **Power user** role to configure account connections for restriction.

Note

With the **Power user** role, you can enable or disable project restrictions for an account, but you can only configure access for projects where you are a member.

- Any other projects that are using the account, such as workflows in a separate project, will no longer be able to use the account. Make sure to update any projects using the restricted account with an account that is not restricted.
- After specifying an account as enabled for restriction, you must explicitly enable the project or projects that will have access.
- If you create a new project with an account connection that is enabled for project restriction, you will not be able to add the account connection to the new project's workflows until the project is enabled for the restricted account.

Enabling project-restricted account connections

Use these steps to enable an account for project restrictions and to specify projects where access is enabled.

To enable project-restricted account connections

1. Open the CodeCatalyst console at <https://codecatalyst.aws/>.
2. Navigate to your CodeCatalyst space. Choose **Settings**, and then choose **AWS accounts**.

The **Accounts** page displays.

3. Choose the account that you want to restrict for your space. Workflows in projects and VPC connections for the space will not have access to the restricted accounts available and their roles. Choose **Enable project restrictions**.

Note

Only specified projects will be able to access the account connection.

4. Choose the project or projects where you want to enable access, and then choose **Enable**. The account connection is now restricted to the selected projects.

Disabling project-restricted account connections

Use these steps to remove project restrictions for a connected account.

To disable project-restricted account connections

1. Open the CodeCatalyst console at <https://codecatalyst.aws/>.
2. Navigate to your CodeCatalyst space. Choose **Settings**, and then choose **Account connections**.

The **Accounts** page displays.

3. Choose the project where you want to disable access, and then choose **Disable**.
4. To completely disable project restrictions, choose **Disable project restrictions**.

Note

Other projects will be able to access the account connection.

Enabling generative AI features in Amazon CodeCatalyst

Amazon Q Developer in Amazon CodeCatalyst includes generative AI features that can help users in projects in your space develop software faster. Developers frequently have more tasks to do than time to accomplish them. CodeCatalyst integrates with Amazon Q Developer to provide features that can help team members accomplish their tasks more quickly and increase the time they have to focus on the most important parts of their work. These features are only available to users if you enable generative AI features for your space. If you choose to allow access to these features, users can access and use these features to help them accomplish their work more quickly. When these features are enabled, an individual user's usage of and quotas for using Amazon Q features depends on the user's subscription to Amazon Q Developer. For more information, see [Amazon Q Developer Pricing](#).

Important

Generative AI features are only available in the US West (Oregon) Region.

The generative AI features available for your space provide the following functionality:

- **Assign issues to Amazon Q** feature with Amazon Q Developer Agent for software development: Users with the Project administrator or Contributor role in a project can assign issues to Amazon Q. Once assigned, Amazon Q will analyze an issue based on its title and its description, review the code in the specified repository, and attempt to create a draft solution for users to evaluate. Users can assign issues to address problems or feature requests in code. They can also use this feature to create or update workflows for a project. This feature includes interactive commenting between users and Amazon Q in not only the issue, but in any tasks or pull requests it creates. Users can choose to have Amazon Q create one revision of any pull request it creates based on user comments left for Amazon Q.
- **Recommend tasks:** This feature helps users by using Amazon Q to analyze an issue and create tasks based on the issue title, description, and its analysis of the complexity of the issue and the repository code. This can help you and your team to assign individual parts of the work to users in more manageable ways that can be achieved more quickly.
- **Write description for me:** This feature helps users creating pull requests create detailed descriptions of the code changes contained in the pull request by comparing the code in the

source and destination branches and evaluating the impact of the differences on the overall application. This feature is not available for pull requests in linked repositories.

- **Create comment summary:** This feature helps users reviewing pull requests understand the overall direction of the comments left on the code changes by other reviewers by summarizing the requests and sentiments expressed in all comments in the overview of the pull request. This feature is not available for pull requests in linked repositories.

Note

Powered by Amazon Bedrock: AWS implements [automated abuse detection](#). Because the **Write description for me**, **Create content summary**, **Recommend tasks**, **Use Amazon Q to create or add features to a project**, and **Assign issues to Amazon Q** feature with Amazon Q Developer Agent for software development features are built on Amazon Bedrock, users can take full advantage of the controls implemented in Amazon Bedrock to enforce safety, security, and the responsible use of artificial intelligence (AI).

The generative AI features in CodeCatalyst are subject to quotas. For more information, see [Amazon Q Developer Pricing](#), [Enabling or disabling generative AI features for a space](#), and [Administering billing](#).

Enabling or disabling generative AI features for a space

To allow users to use generative AI features in CodeCatalyst, you must enable them for your space. You cannot enable these features for specific projects. If you enable them for the space, all projects in the space will have access to them. For more information about the available generative AI features, see [Enabling generative AI features in Amazon CodeCatalyst](#).

Before you begin

The generative AI features in CodeCatalyst are useful and powerful tools for users in your space. Like all powerful tools, they perform best when well understood by users and administrators. You should be aware of the following considerations before you decide to enable generative AI features for your space.

- **Generative AI features cannot make arbitrary changes to your project.** While Amazon Q can create pull requests with code changes, it cannot merge a pull request. A user must approve the

pull request and merge the changes. However, if you have one or more workflows configured to run based on branch events, those workflows will run immediately when the pull request is created. Additionally, once code is merged, the changes are part of the source repository branch where the pull request was merged. Just like any other merged pull request, any workflows configured to build and deploy pushes to the destination branch will start a run of the changes contained in a pull request created by Amazon Q.

- **Generative AI features might not correctly interpret user statements and existing code structures.** User intent can be misinterpreted by generative AI features, which can result in code generation that does not do what the original requestor expected. The approach Amazon Q creates is based on the title and description of an issue, as well as its analysis of the code in the source repository specified in the issue. If the code in the source repository contains errors, Amazon Q might repeat these errors if it interprets them as part of the desired structure and patterns in the code. As a best practice, encourage users in a space with generative AI features enabled to thoroughly review any code in pull requests created by Amazon Q to ensure that the code both does what the issue title and description suggested, and that the code itself is well-formed and functional. Like any other pull request, users can create a Dev Environment and make changes and fixes to the code in the branch created by Amazon Q. Similarly review a revision created by Amazon Q to help ensure that the changes correctly address any issues raised in comments.
- **Workflows created by Amazon Q might deploy code to production resources immediately once a pull request is merged.** By default, CodeCatalyst does not allow Amazon Q to create or update workflows. However, you can choose to have Amazon Q do so, and it will try to create or update a workflow according to the information in the issue. If you merge a pull request with workflow changes, that workflow will automatically run when its conditions are met. Additionally, like any pull request, pull requests created by Amazon Q will start a workflow run when the specified conditions are met, so if you have a workflow configured to run on branch events, a pull request with workflow changes might start a run of those workflows.

You must have the **Space administrator** role in CodeCatalyst to manage generative feature access for a space.

 **Important**

Generative AI features are only available in the US West (Oregon) Region.

To enable or disable generative AI features for a space

1. Open the CodeCatalyst console at <https://codecatalyst.aws/>.
2. Navigate to your CodeCatalyst space. Choose **Settings**, and then choose **Generative AI**.

The **Generative AI** page displays.

3. To enable generative AI features for all projects in your space, choose the **Projects in this space can access generative AI features** toggle and make sure it is in the on position. Users in projects will immediately have access to the generative AI features available to their roles in those projects.
4. To disable generative AI features for all projects in your space, choose the **Projects in this space can access generative AI features** toggle and make sure it is in the off position.

Warning

Disabling the generative AI features will stop work on all issues and summaries in all projects. The work cannot be restarted even if you immediately re-enable the generative AI features.

Viewing usage of generative AI features in a space

Although generative AI features are enabled for a space, the individual usage of those features is captured at the individual user level. You can view information about your individual usage of generative AI features by going to your **My settings** page.

User usage and quotas for using generative AI features in CodeCatalyst is based on that user's subscription to Amazon Q Developer. For more information, see [Amazon Q Developer Pricing](#).

To view your generative AI feature usage

1. Open the CodeCatalyst console at <https://codecatalyst.aws/>.
2. Choose your profile gravitar, and then choose **My settings**.

Review the information in **Generative AI feature usage** to determine current usage levels and whether your individual user account is close to exceeding any quotas.

Connecting an Amazon VPC to an Amazon CodeCatalyst space

Amazon Virtual Private Clouds (Amazon VPCs) are virtual networks that provide extra security to your application by isolating it from the public internet. By connecting to an Amazon VPC, users can securely run workflow actions and create Dev Environments linked to your VPC in CodeCatalyst. You can set a default VPC connection for a space, so that all workflow runs and Dev Environments will run connected to that VPC.

A *VPC connection* is a CodeCatalyst resource which contains all of the configurations needed for your workflow to access a VPC. Space administrators can add their own VPC connections in the Amazon CodeCatalyst console on behalf of space members. By adding a VPC connection, space members can run workflow actions and create Dev Environments that adhere to network rules and can access resources in the associated VPC.

For more information about setting up a VPC, see the [Amazon VPC User Guide](#).

Considerations for administering VPC connections

- You must have the **Space administrator** role or **Power user** role to manage VPC connections at the space level.
- **Project administrators** can associate a VPC connection to their environment and **Contributors** can associate that VPC-connected environment with a workflow. When administering VPC connections as a **Space administrator**, you are maintaining these resources on behalf of space members.

Topics

- [Use cases](#)
- [How do I get started with VPC connections?](#)
- [Limitations of VPC connections in CodeCatalyst](#)
- [Setting up an Amazon VPC](#)
- [Adding VPC connections for a space](#)
- [Configuring VPC endpoints for a space](#)
- [Managing a default VPC connection for a space](#)

- [Editing VPC connections for a space](#)
- [Removing VPC connections for a space](#)

Use cases

VPC connectivity from CodeCatalyst actions makes it possible to:

- Run a workflow action that follows the network rules of a VPC connection.
- Run a workflow action that accesses resources running in a VPC.
- Deploy an update to an Amazon EKS cluster running in a VPC.

How do I get started with VPC connections?

The high-level steps to add and use a VPC connection are as follows:

1. In the AWS Management Console, **create an Amazon Virtual Private Cloud (VPC)** or use an existing VPC. A VPC is a virtual network that provides extra security to your application by isolating it from the public internet and allows you to securely run your workflow actions in CodeCatalyst. In order for your VPC to work with CodeCatalyst, it must have a certain configuration. For more information, see [Amazon VPC setup requirements](#).
2. In your CodeCatalyst space settings, **create a VPC connection**. A *VPC connection* is a CodeCatalyst resource which contains all of the configurations needed for a workflow to access a VPC. For more information, see [Adding VPC connections for a space](#).
3. Associate this VPC connection with an **environment** to use with your workflow actions. For more information, see [Associating a VPC connection with an environment](#) in the *CodeCatalyst User Guide*.
4. Within a workflow, associate the VPC-connected environment to your **workflow action**. When an action is configured with an environment that has a VPC connection, the action will run connected to the VPC, adhere to the network rules, and access resources specified by the associated VPC. For more information, see [Associating an environment, account connection, and IAM role with a workflow action](#) in the *CodeCatalyst User Guide*.
5. Create a **Dev Environment** associated to your VPC connection. For more information, see [Creating a Dev Environment](#) in the *CodeCatalyst User Guide*.

Limitations of VPC connections in CodeCatalyst

- CodeCatalyst only supports creating VPC connections in the same region. For more information on the available regions, see [CodeCatalyst VPC endpoint service names](#).
- CodeCatalyst does not support VPC connectivity with the Lambda compute type. Instead, use the Amazon EC2 compute type.
- CodeCatalyst does not support VPC connectivity with Windows. Instead, use Linux.
- VPC connectivity may lead to longer action run times.

Setting up an Amazon VPC

Use the following procedure to create a VPC.

To create a VPC

- Follow the instructions in the *Amazon VPC User Guide* for [Creating a VPC](#). While following these instructions, keep in mind the VPC requirements needed to work with CodeCatalyst.

For a tutorial that uses AWS CloudFormation to create a VPC, see [AWS Solution: Amazon Virtual Private Cloud on AWS](#).

Amazon VPC setup requirements

In order for a VPC to work with CodeCatalyst, it must have the following requirements:

- For **Number of public subnets**, make sure that you have at least one [public subnet](#) in any Availability Zone.
- For **Number of private subnets**, make sure that you have one [private subnet](#) in each available Availability Zone in a region.
- Make sure your VPC has access to the internet. This can be done by adding a route with a destination of `0.0.0.0/0` to an [internet gateway](#) and a [NAT device](#).
- Make sure that the routing table for private subnets points to the NAT gateway. For more information, see [Routing to a NAT device](#) in the *Amazon VPC User Guide*.
- Make sure that your internet gateway is attached to the VPC. Public subnets should have a routing table to the internet gateway. For more information, see [Routing to an internet gateway](#) in the *Amazon VPC User Guide*.

- Make sure that your security groups allow outbound traffic.
- Make sure that your IPv4 CIDR block is **not** configured to the 172.16.0.0/12 IP address range. For more information, see [IPv4 VPC CIDR blocks](#) in the *Amazon VPC User Guide*.
- As a best practice, make sure that your security groups have no inbound traffic allowed, unless you specifically require this for other reasons.
- CodeCatalyst does not support assigning a public IP address to the network interfaces that it creates. One way to do this, is to add a NAT device to use CodeCatalyst with your VPC. For more information, see [Connect to the internet or other networks using NAT devices](#) in the *Amazon VPC User Guide*.

Troubleshooting your VPC setup

Use the information that appears in the error message to help you identify, diagnose, and address issues.

The following are some guidelines to assist you when troubleshooting common VPC errors:

1. [Make sure that your internet gateway is attached to VPC.](#)
2. [Make sure that the route table for your public subnet points to the internet gateway.](#)
3. [Make sure that your network ACLs allow traffic to flow.](#)
4. [Make sure that your security groups allow traffic to flow.](#)
5. [Troubleshoot your NAT gateway.](#)
6. [Make sure that the route table for private subnets points to the NAT gateway.](#)
7. [Make sure that your IPv4 CIDR block is not configured to the 172.16.0.0/12 IP address range.](#)

Adding VPC connections for a space

You can add VPC connections in the Amazon CodeCatalyst console.

You must have the **Space administrator** role or **Power user** role to manage VPC connections at the space level.

To add VPC connections

1. Open the CodeCatalyst console at <https://codecatalyst.aws/>.

2. Navigate to your CodeCatalyst space.

Tip

If you belong to more than one space, choose a space in the top navigation bar.

3. Choose **Settings**, and then choose **VPC connections**.

The page lists all VPC connections in your space. You can view the **VPC connection name** name, the **VPC ID**, and the associated **AWS account connection**.

4. Choose **Add VPC connection**.

5. In **AWS account connection**, do the following:

- For **AWS account connection**, choose a connection from the drop-down menu.

For more information about connections, see [Adding an AWS account to a space](#) in the *CodeCatalyst User Guide*.

Note

If you associate your VPC connection with a project-restricted AWS account connection, your VPC connection will only have access to specific projects and cannot be set as default. For more information, see [Enabling or disabling project-restricted account connections](#).

- For **VPC role**, choose a role from the drop-down menu and then choose **Next**.
- We recommend that the `ArnLike` field for your trust policy contains the following:

```
{
  "aws:SourceArn": [
    "arn:aws:codecatalyst:::space/<space-id>",
    "arn:aws:codecatalyst:::space/<space-id>/project/*"
  ]
}
```

Adding this `SourceArn` condition in your trust policy ensures that the VPC role is only used for the specified space.

Note

Understand that VPC connections are a space level resource, meaning that your VPC can be accessed by different projects. You can restrict access by configuring your VPC role trust policy with a specific `projectId` instead of `*`.

- The Action field for your permission policy must contain the following:

```
[
  "ec2:CreateNetworkInterface",
  "ec2:DescribeDhcpOptions",
  "ec2:DescribeNetworkInterfaces",
  "ec2>DeleteNetworkInterface",
  "ec2:DescribeSubnets",
  "ec2:DescribeSecurityGroups",
  "ec2:DescribeVpcs"
]
```

This **VPC role** will be used to populate the **VPC**, **Subnets**, and **Security groups** drop-down menus and establish VPC connectivity with CodeCatalyst actions.

For more information about roles, see [Managing IAM roles for connected accounts](#) in the *CodeCatalyst User Guide*.

6. In **VPC connection details**, do the following:

- For **VPC**, choose a VPC from the drop-down menu.

For more information, see [Create a VPC](#) in the *Amazon VPC User Guide*.

- In **Subnets**, choose *private* subnets to connect to in each availability zone from the drop-down menus. Do not choose public subnets.

For more information, see [Create a subnet](#) in the *Amazon VPC User Guide*.

- In **Security groups**, select the groups from the drop-down menu. You can select up to five security groups.

For more information, see [Security groups](#) in the *Amazon VPC User Guide*.

- In **VPC connection name**, enter the reference name for your VPC connection then choose **Next**.

Note

Each VPC connection name must be unique per space.

7. After you've reviewed your selections, choose **Add VPC connection**.
8. You can now associate this VPC connection with an environment to use with your workflow actions or create a Dev Environment associated to your VPC connection. For instructions, see [Associating a VPC connection with an environment](#) or [Using Dev Environments with a VPC connection](#) in the *CodeCatalyst User Guide*.

Configuring VPC endpoints for a space

VPCs allow you to define a virtual network that isolates AWS resources, securely connects to remote networks, and safely accesses service endpoints through AWS PrivateLink. AWS PrivateLink is used to generate private endpoints which keep all the network traffic within the AWS network. When connected to a VPC, you can create VPC endpoints that will allow CodeCatalyst to communicate directly with certain services rather than through the internet.

For more information about PrivateLink and VPC endpoints, see [What is AWS PrivateLink?](#)

Use the following procedure to configure VPC endpoints for a space.

AWS console

To configure VPC endpoints using the AWS console

1. Open the Amazon VPC console at <https://console.aws.amazon.com/vpc/>.
2. In the navigation pane, choose **Endpoints** and then choose **Create endpoint**.
3. In **Endpoint settings**, do the following:
 - (Optional) For **Name tag**, enter a reference name for your endpoint.
4. In **Services**, enter your specified service name and then select it. For more information, see [CodeCatalyst VPC endpoint service names](#).
5. In **VPC**, choose the VPC in which to create your endpoint.
 - For **Additional settings**, leave the default.

6. In **Subnets**, select the same private subnets that you associated with your VPC connection to connect to in each availability zone:
 - In **IP address type**, select **IPv4**. This enables the endpoint service to accept IPv4 requests.
7. In **Security groups**, select the same security groups that you associated with your VPC connection then choose **Create endpoint**.
8. After your VPC endpoint is created, choose that endpoint, and then choose **Modify private DNS name**.
9. In **Enable private DNS names**, select **Enable for this endpoint**.

AWS CLI

To configure VPC endpoints using the AWS CLI

1. If you haven't done so already, [set up the AWS CLI for CodeCatalyst](#).
2. Run this command to sign-in to Amazon CodeCatalyst using AWS IAM Identity Center:

```
aws sso login --profile codecatalyst
```

3. Create your VPC endpoint:

```
aws ec2 create-vpc-endpoint --vpc-id <vpc-id> --service-name <service-name> --  
subnet-ids <subnet-ids> --security-group-ids <security-group-ids> --private-dns-  
enabled
```

For more information on service names, see [CodeCatalyst VPC endpoint service names](#).

CodeCatalyst VPC endpoint service names

You can create VPC endpoints for these services, if you would prefer for CodeCatalyst to utilize these endpoints.

- Source:
 - Regions: us-west-2, eu-west-1
 - Service name: com.amazonaws.<region>.codecatalyst.git
- API:

- **Regions:** us-west-2, eu-west-1
- **Service name:** aws.api.global.codecatalyst
- **Packages:**
 - **Regions:** us-west-2, eu-west-1
 - **Service name:** com.amazonaws.<region>.codecatalyst.packages

Managing a default VPC connection for a space

You can set a default VPC connection for a space. If you choose to set a default VPC connection, all workflow runs and Dev Environments in your space will run connected to the default VPC connection. You can override this by associating a different VPC connection in your workflow action or Dev Environment.

You must have the **Space administrator** role or **Power user** role to manage VPC connections at the space level.

Topics

- [Setting a default VPC connection](#)
- [Removing a default VPC connection](#)

Setting a default VPC connection

Use the following procedure to set a default VPC connection.

To set a default VPC connection

1. Open the CodeCatalyst console at <https://codecatalyst.aws/>.
2. Navigate to your CodeCatalyst space.


Tip

If you belong to more than one space, choose a space in the top navigation bar.

3. Choose **Settings**, and then choose **VPC connections**.

The page lists all VPC connections in your space. You can view the **VPC connection name**, the **VPC ID**, and the associated **AWS account connection**.

4. Choose the VPC connection name that you want to set as default.

 **Note**

If your VPC connection is associated with a project-restricted AWS account, your VPC connection will only have access to specific projects and cannot be set as default. For more information, see [Enabling or disabling project-restricted account connections](#).

5. Choose **Manage default**, choose **Set as default** from the drop-down menu, then choose **Confirm**.

Removing a default VPC connection

Use the following procedure to remove a default VPC connection.

To remove a default VPC connection

1. Open the CodeCatalyst console at <https://codecatalyst.aws/>.
2. Navigate to your CodeCatalyst space.

 **Tip**

If you belong to more than one space, choose a space in the top navigation bar.

3. Choose **Settings**, and then choose **VPC connections**.

The page lists all VPC connections in your space. You can view the **VPC connection name** name, the **VPC ID**, and the associated **AWS account connection**.

4. Choose the default VPC connection name.
5. Choose **Manage default**, choose **Remove as default** from the drop-down menu, then choose **Confirm**.

Editing VPC connections for a space

You can edit the configuration for a VPC connection, such as the associated subnets or security groups.

You must have the **Space administrator** role or **Power user** role to manage VPC connections at the space level.

Warning

While VPC-connected workflows are in progress, we recommended that you do not edit your VPC connection or your VPC role. If the associated VPC is edited while your workflow is in progress, your workflow will continue to run with the initial VPC connection. If a VPC role is edited to remove necessary permissions while your workflow is in progress, you will not be able to cleanup your resources.

To edit VPC connections

1. Open the CodeCatalyst console at <https://codecatalyst.aws/>.
2. Navigate to your CodeCatalyst space.

Tip

If you belong to more than one space, choose a space in the top navigation bar.

3. Choose **Settings**, and then choose **VPC connections**.

The page lists all VPC connections in your space. You can view the **VPC connection name** name, the **VPC ID**, and the associated **AWS account connection**.

4. Choose the VPC connection name that you want to manage, and then choose **Edit**.
5. Edit your VPC connection as needed, and then choose **Update VPC connection**.

Removing VPC connections for a space

You can remove a VPC connection that is no longer needed or that no longer has an owner.

You must have the **Space administrator** role or **Power user** role to manage VPC connections at the space level.

⚠ Warning

While VPC-connected workflows are in progress, we recommended that you do not delete your VPC connection or your VPC role. If the associated VPC is deleted while your workflow is in progress, your workflow will continue to run with the initial VPC connection.

To remove VPC connections

1. Open the CodeCatalyst console at <https://codecatalyst.aws/>.
2. Navigate to your CodeCatalyst space.

ℹ Tip

If you belong to more than one space, choose a space in the top navigation bar.

3. Choose **Settings**, and then choose **VPC connections**.

The page lists all VPC connections in your space. You can view the **VPC connection name**, the **VPC ID**, and the associated **AWS account connection**.

4. Choose the selector next to the VPC connection you want to manage. Choose **Remove VPC connection**. To confirm, type the VPC connection name, and then choose **Remove**.

Administering billing

Your CodeCatalyst space uses resources, such as Dev Environments and compute, that are charged based on separate subscription tiers. To check your usage and to view pricing for each subscription tier, see the **Billing** tab for your space.

You can view or change your CodeCatalyst subscription for your space. CodeCatalyst has the following tiers available.

- Free
- Standard
- Enterprise

For more information about billing and rates, see the [Amazon CodeCatalyst pricing page](#).

Additional considerations:

- All spaces are required to have a billing account, even if the usage does not exceed the Free tier.
- AWS is the billing provider for CodeCatalyst. The account you use for billing must be an AWS account.
- Billing is configured at the space level. Multiple AWS accounts can be associated with a space, but only one can be active for CodeCatalyst billing at a time. An AWS account can be used as a billing account for more than one space in CodeCatalyst. The AWS account that is specified as the billing account for your CodeCatalyst space has different quotas from other account connections for a space. For more information, see [Quotas](#) in the *CodeCatalyst User Guide*.
- You can view your CodeCatalyst charges on your AWS invoice, including current charges. Billing data is updated regularly on the AWS Management Console page for AWS Billing.

Here is one possible flow to set up billing in CodeCatalyst.

Mary Major is a **Space administrator** for a CodeCatalyst space. Mary works with the AWS administrator for the AWS account that was added to her space in CodeCatalyst when it was created and so defaults as the billing account. Mary can also specify a different account as a billing account for her space. Mary initially set up her subscription to default to the Free tier. Mary knows that hitting the maximum can halt services for the rest of her space on the Free tier, so she periodically reviews usage for her space on the CodeCatalyst console **Billing** tab. Mary views the

pricing subscription for the Standard tier, and after the space grows in size, Mary decides to use the CodeCatalyst billing page to change the subscription tier from the Free tier to the Standard tier. Mary works with her AWS administrator to access her AWS invoice and views the charges for usage beyond the Free tier in CodeCatalyst.

Setting up a billing account

To set up billing, you must add an authorized AWS account to your CodeCatalyst space and configure it for billing. Before you can set up a billing account, you must work with your AWS administrator to complete the prerequisites in the following procedure.

Where necessary, these steps link you to the procedures in this guide for adding the AWS account you want to use for billing to your space by creating a connection.

You must have the **Space administrator** role in CodeCatalyst and have administrator permissions for your account in AWS to manage billing.

Step 1: Add the billing account to your space

To choose a new account for billing, you must have completed the steps to add it to your space. Complete the steps as detailed in [Administering connected accounts](#).

You must have the **Space administrator** role in CodeCatalyst and have administrator permissions for your account in AWS to manage billing.

Step 2: Choose the billing account in CodeCatalyst

Next, specify the connected account as a billing account.

The account ID is now authorized for billing. This is the account that you will add to your space, and it is available in the list of options. Only AWS accounts that have been added to your space will show in the list.

You must have the **Space administrator** role to manage billing and accounts for your space.

To designate an added AWS account to a CodeCatalyst space for billing

1. Open the CodeCatalyst console at <https://codecatalyst.aws/>.

2. Navigate to your CodeCatalyst space. Choose **Settings**, and then choose **Billing**.

The **Billing details** page displays.

3. Choose **Add an AWS account**.

If the space already has a designated billing account that you want to change, choose the **Edit** button next to the **AWS account name** field.

The **Change AWS billing account** page displays.

4. From the **Choose from AWS accounts already associated with your space** dropdown menu, choose the name of another added account for your space that you want to designate as the billing account for your CodeCatalyst space. The account name must be for an account that has been added to your space with a connection and is listed in the **AWS accounts list for your space**.
5. Choose **Change billing account**.

Changing allowed tiers for a billing account

When you set up a billing account, the allowed CodeCatalyst tiers that you want to allow your space to use defaults to the free and paid tiers. You can turn the paid tiers on or off from your space's **Settings**.

You must have the **Space administrator** role in CodeCatalyst and have administrator permissions for your account in AWS to manage billing.

1. In the AWS Management Console, make sure you are signed in with the same account that you want to manage. You will be automatically directed to a page in the AWS Management Console from the CodeCatalyst console.
2. Open the CodeCatalyst console at <https://codecatalyst.aws/>.
3. Navigate to your CodeCatalyst space. Choose **Settings**, and then choose **Billing**.

The **Billing details** page displays.

4. Choose **Manage billing in AWS**. This opens the Amazon CodeCatalyst Spaces in the AWS Management Console. If you are prompted to sign in, sign in to AWS, and then choose the button again to load the page.
5. Choose **Spaces**. The list of each space with a connection to the account you are signed in with, if any, displays.

6. Choose the account link for your CodeCatalyst space. The connection page is shown.
7. Under **Billing details**, choose **Edit**. The **Update allowed billing tiers** page displays.
8. To keep the default, leave the selection on **PAID (STANDARD, ENTERPRISE)**. To turn off paid tiers, deselect the **PAID (STANDARD, ENTERPRISE)** field.
9. Choose **Update**.

Changing your CodeCatalyst billing tier

When you set up a billing account, the allowed CodeCatalyst tiers that you want to allow your space to use defaults to the free and paid tiers. Your current plan and configuration for billing, if set up, is available in the CodeCatalyst console.

You must have the **Space administrator** role in CodeCatalyst and have administrator permissions for your account in AWS to manage billing.

To change your CodeCatalyst billing tier

1. Open the CodeCatalyst console at <https://codecatalyst.aws/>.
2. Navigate to your CodeCatalyst space. Choose **Settings**, and then choose **Billing**.

The **Billing details** page displays.

3. Choose **Change billing tier**. The **Compare tiers** page displays with available plans and the list of charges that apply to each plan.

On the **Tiers** page, choose the tier with the charges that you want to authorize your CodeCatalyst space to use. The following tiers are available:

- **Free**
- **Standard**
- **Enterprise**

For the Enterprise tier, you can enter a value in the **Number of team members** input field to view the total amount of each resource included in your space.

4. Review the details for the change, and then choose **Change tier**.

Changing a billing account

You can change the AWS account that you want to specify as the billing account for your CodeCatalyst space.

Important

You cannot delete the connection for an account that is designated as the billing account for your space.

You must have the **Space administrator** role to manage billing and accounts for your space.

To change the AWS account for a CodeCatalyst space for billing

1. Open the CodeCatalyst console at <https://codecatalyst.aws/>.
2. Navigate to your CodeCatalyst space. Choose **Settings**, and then choose **Billing**.

The **Billing details** page displays.

3. Choose the **Edit** button next to the **AWS account name** field. The **Change AWS billing account** page displays.
4. In **Choose from AWS accounts already associated with your space**, choose the name of another added account that you want to designate as the billing account for your CodeCatalyst space. The account name must be for an account that has been added to your space and is listed in the **AWS accounts** tab for your space.
5. Choose **Change billing account**.

Viewing your CodeCatalyst plan and billing

When you set up a billing account, the allowed CodeCatalyst tiers that you want to allow your space to use defaults to the free and paid tiers. Your current subscription tier and the configured billing account are available in the CodeCatalyst console.

You must have the **Space administrator** role in CodeCatalyst to access billing information.

To access the CodeCatalyst billing information

1. Open the CodeCatalyst console at <https://codecatalyst.aws/>.

2. Navigate to your CodeCatalyst space. Choose **Settings**, and then choose **Billing**.

The **Billing details** page displays.

3. On the **Billing details** page, view your current subscription and usage. In **AWS account status** and **AWS account name**, view the configured AWS account for billing.

To view the CodeCatalyst billing status

1. In the AWS Management Console, make sure you are signed in with the same account that you want to manage. You will be automatically directed to a page in the AWS Management Console from the CodeCatalyst console.
2. Open the CodeCatalyst console at <https://codecatalyst.aws/>.
3. Navigate to your CodeCatalyst space. Choose **Settings**, and then choose **Billing**.

The **Billing details** page displays.

4. Choose **Manage billing in AWS**. You are directed to the connections page.

You can view details about the connection between your AWS account and CodeCatalyst space. Under **Billing details**, view the CodeCatalyst subscriptions allowed for the account.

Viewing CodeCatalyst accounts for billing in AWS

The Amazon CodeCatalyst page in the AWS Management Console shows the account name that you have specified for billing. This page also shows the account connections.

You must have the **Space administrator** role in CodeCatalyst and have administrator permissions for your account in AWS to manage billing.

To view the CodeCatalyst billing status page for an added account

1. Open the CodeCatalyst console at <https://codecatalyst.aws/>.
2. Navigate to your CodeCatalyst space. Choose **Settings**, and then choose **Billing**.

The **Billing details** page displays.

3. In **AWS account status**, view the billing status for the account. In **AWS account name**, view the account name, such as my-space-billing-account.

To view CodeCatalyst connections and billing in AWS

1. In the AWS Management Console, make sure you are signed in with the same account that you want to manage. You will be automatically directed to a page in the AWS Management Console from the CodeCatalyst console.
2. Open the CodeCatalyst console at <https://codecatalyst.aws/>.
3. Navigate to your CodeCatalyst space. Choose **Settings**, and then choose **Billing**.

The **Billing details** page displays.

4. Choose **Manage billing in AWS**. The Amazon CodeCatalyst spaces page in the AWS Management Console displays.

Note

You might need to sign in to AWS before you can open the page in the AWS Management Console.

Getting AWS billing information

If you have the necessary permissions, you can get information about your AWS charges from the AWS Management Console.

For more information about billing, including information about the tiers available, rates, and information included in the monthly AWS billing cycle, see the [Amazon CodeCatalyst pricing page](#).

To get your billing information

1. Sign in to the AWS Management Console and navigate to **Billing**. In the navigation bar, choose your account name.
2. Choose **Billing dashboard**.
3. Use the AWS Billing and Cost Management dashboard to find a summary and a breakdown of your monthly spending. To learn more, see the [AWS Billing User Guide](#).

Troubleshooting problems with billing associated with your space

In CodeCatalyst, you can add an AWS account to your space to grant permissions to resources and for billing purposes. The following information can help you troubleshoot common issues with associated accounts in CodeCatalyst.

Topics

- [I cannot access the Amazon CodeCatalyst Spaces page in the AWS Management Console](#)
- [I cannot change the billing tier for my space](#)

I cannot access the Amazon CodeCatalyst Spaces page in the AWS Management Console

Problem: When I try to access the Amazon CodeCatalyst page in the AWS Management Console to add an account to my CodeCatalyst space or add roles to an account in AWS, I receive a permissions error.

Possible fixes:

For your space, you can add authorized AWS accounts to add them to your project if you have the **Space administrator** role. You must also have an AWS account where you have administrative permissions or can work with your AWS administrator. You must first make sure you are signed in to the AWS Management Console with the same account that you want to manage. After you are signed in to the AWS Management Console, you can open the console and try again.

Open the Amazon CodeCatalyst page in the AWS Management Console at [https://us-west-2.console.aws.amazon.com/codecatalyst/home?region=us-west-2#/.](https://us-west-2.console.aws.amazon.com/codecatalyst/home?region=us-west-2#/)

I cannot change the billing tier for my space

Problem: I am able to view billing for my space but cannot change the subscription tier from Free to Standard.

Possible fixes: For your space, you can change subscription tiers or authorize billing accounts if you have the **Space administrator** role.

Before you can change your CodeCatalyst plan from the Free tier, you must first have an account added to your space. Next, you must set up the account for billing by turning on the Standard tier.

This does not change the billing tier for the space. It authorizes the Standard tier for the account, so the **Space administrator** will be able to upgrade to the Standard tier.

You must have the **Space administrator** role in CodeCatalyst and have administrator permissions for your account in AWS to manage billing.

Viewing and configuring your user profile

Note

This topic describes how to view your user profile if you signed in to CodeCatalyst with a AWS Builder ID. If you signed in as a user who is managed as part of SSO users and groups, see [Administering spaces that support identity federation](#).

Your user profile displays your personal information, and is where you configure user-specific features like notifications.

In your user profile, you can view but not edit:

- The alias you specified during signup.
- The email address you used during signup.
- The user identifier assigned to you during signup.

In your user profile, you can edit:

- Your name.
- Your nickname.

In your user profile, you can configure these features:

- Personal access tokens (PATs). For more information, see [Grant users repository access with personal access tokens](#).
- Personal connections. For more information, see [Accessing GitHub resources with personal connections](#).
- The email and Slack notifications you want to receive. For more information, see [Sending notifications from CodeCatalyst](#).

You provide information at signup that will be used as your credentials to log in to Amazon CodeCatalyst and that will be managed in your profile. This includes your **Name**, **Nickname**, and the **Email address** you use to sign in to CodeCatalyst.

Note

The AWS Builder ID **Nickname** is not your CodeCatalyst alias. You selected your CodeCatalyst alias at signup.

To view and configure your user profile

1. Open the CodeCatalyst console at <https://codecatalyst.aws/>.
2. At the top right, choose the arrow next to the icon with your first initial, and then choose **My settings**. The CodeCatalyst **My settings** page opens.
3. View your user alias and name information. For a user in a CodeCatalyst space that supports AWS Builder ID users, the label **AWS Builder ID** displays above the icon.
4. View and configure features as required, such as your PATs, notifications, and personal connections.

Document history for the Amazon CodeCatalyst Administrator Guide

The following table describes the documentation releases for Amazon CodeCatalyst.

Change	Description	Date
Updated content: Identity Center applications	Added more detailed information about how the Identity Center application name for your space must be globally unique. See Concepts , Setting up a federated space , and Administering Identity Center applications .	September 13, 2024
Update for project-restricted account connections	You can now configure a connected AWS account for restricted use only with specified CodeCatalyst projects in a space. See Enabling and disabling project-restricted account connections .	July 18, 2024
Updates to spaces and account connections	You can now connect the AWS account specified as the billing account to multiple CodeCatalyst spaces. For spaces set up for identity federation, one Identity Center instance can be connected to multiple spaces. See Managing spaces that support identity federation	June 13, 2024

and [Managing Identity Center applications](#).

[Updated content: Managing generative AI features](#)

Updated information for Amazon Q.

April 29, 2024

[New content: Managing generative AI features](#)

Added information about managing the generative AI features available in CodeCatalyst.

November 28, 2023

[Initial release](#)

Initial release of the Amazon CodeCatalyst Administrator Guide

November 15, 2023