AWS Black Belt Online Seminar

Amazon CodeCatalyst

Identity, permissions, and access 編

国兼 周平

Solutions Architect
2023/12

自己紹介

国兼 周平 / Kunikane Shuhei

Amazon Web Services Japan ソリューションアーキテクト

経歴:

1. 国内 SIer にて、アプリケーションエンジニアとして
   金融/公共の分野で開発
2. インターネットサービス企業にて、バックエンド開発者として
   ネット証券取引サービスやネットバンキングサービスの立ち上げ
3. AWS にて、Professional Services (*) のコンサルタントとして
   モビリティのお客様を支援
4. ソリューションアーキテクトに転向

得意分野:

- 高負荷 WEB サービスの設計/開発
- CI/CD, IaC, Serverless, Container

* Professional Services: AWS の提供する有償のコンサルティングサービス

My favorite AWS Services

- Amazon ECS
- AWS Lambda
- Amazon CodeCatalyst

AWS Black Belt Online Seminar とは

- 「サービス別」「ソリューション別」「業種別」などのテーマに分け、アマゾン ウェブ サービス ジャパン合同会社が提供するオンラインセミナーシリーズです
- AWS の技術担当者が、 AWS の各サービスやソリューションについてテーマごとに動画を公開します
- 以下の URL より、過去のセミナー含めた資料などをダウンロードすることができます
  - https://aws.amazon.com/jp/aws-jp-introduction/aws-jp-webinar-service-cut/
  - https://www.youtube.com/playlist?list=PLzWGOASvSx6FIwIC2X1nObr1KcMCBBlqY

ご感想は X (Twitter) へ！ハッシュタグは以下をご利用ください
#awsblackbelt

内容についての注意点

- 本資料では資料作成時点のサービス内容および価格についてご説明しています。AWS のサービスは常にアップデートを続けているため、最新の情報は AWS 公式ウェブサイト (https://aws.amazon.com/) にてご確認ください
- 資料作成には十分注意しておりますが、資料内の価格と AWS 公式ウェブサイト記載の価格に相違があった場合、AWS 公式ウェブサイトの価格を優先とさせていただきます
- 価格は税抜表記となっています。日本居住者のお客様には別途消費税をご請求させていただきます
- 技術的な内容に関しましては、有料の AWS サポート窓口へお問い合わせください
- 料金面でのお問い合わせに関しましては、カスタマーサポート窓口へお問い合わせください (マネジメントコンソールへのログインが必要です)

Black Belt Amazon CodeCatalyst シリーズ

- Overview 編
- Spaces 編
- Projects, Blueprints 編
- Source repositories 編
- Dev Environments 編
- Workflow 編
- Issues 編
- **Identity, permissions, and access 編**
- Extensions 編

シリーズ構成

- 全体像をお伝えする Overview 編
- 各機能の詳細についてお伝えする各機能編

シリーズの対象読者

- チーム開発をするすべてのアプリケーション開発者

アジェンダ

1. Amazon CodeCatalyst のセキュリティ
2. Space とお客様 AWS アカウントの関係
3. ユーザーおよび権限
4. AWS アカウントへの接続
5. モニタリング
6. クォータ

Amazon CodeCatalyst のセキュリティ

AWS 責任共有モデル

セキュリティは AWS とお客様の間で共有される責任

クラウドのセキュリティ / Security of the Cloud

AWS は、AWS クラウドで提供されるすべてのサービスを実行するインフラストラクチャの保護について責任を負います。

クラウド内のセキュリティ / Security in the Cloud

お客様の責任は、お客様が選択した AWS クラウドサービスによって異なります。これにより、お客様がセキュリティの責任の一部として実行する必要がある設定作業の量が決まります。

Amazon CodeCatalyst の利用における責任共有モデルの適用について詳しくは公式ドキュメントを参照

https://docs.aws.amazon.com/codecatalyst/latest/userguide/security.html
https://aws.amazon.com/jp/compliance/shared-responsibility-model/

CodeCatalyst でのお客様データの保護

- CodeCatalyst 内のすべてのお客様データは保存中も転送中も暗号化される
  - Space ごとに別々の Key で暗号化する
  - Key は AWS によって自動的にローテーションなどの管理がされる
- Workflow は他のお客様の環境と隔離されたシングルテナントの環境で実行される
- Space 作成時にお客様が指定したリージョン以外にお客様のデータが移動することは基本的にない
  - CodeCatalyst のナビゲーションのために、例外として Space, Project, User の限られたメタデータが同一パーティションのリージョンにレプリケートされる
  - リージョンのパーティションおよびレプリケートされるデータ項目については公式ドキュメントを参照

https://codecatalyst.aws/explore/faq
https://docs.aws.amazon.com/codecatalyst/latest/userguide/data-protection.html
https://docs.aws.amazon.com/ja_jp/whitepapers/latest/aws-fault-isolation-boundaries/partitions.html

注意事項 / 検討事項

- 他のサービスとの連携機能を利用する場合、連携先サービスでのデータ管理については CodeCatalyst とは別で検討する
  - GitHub や JIRA, Slack などの Extension の他に、後述の「紐付けされた AWS アカウント」も含む

- MFA (多要素認証) で認証のセキュリティを強化する
  - CodeCatalyst で利用できる AWS Builder ID および AWS IAM Identity Center はともに MFA をサポートしている
  - 外部 IdP を IAM Identity Center の ID ソースとしている場合は、MFA は外部 IdP での設定となる

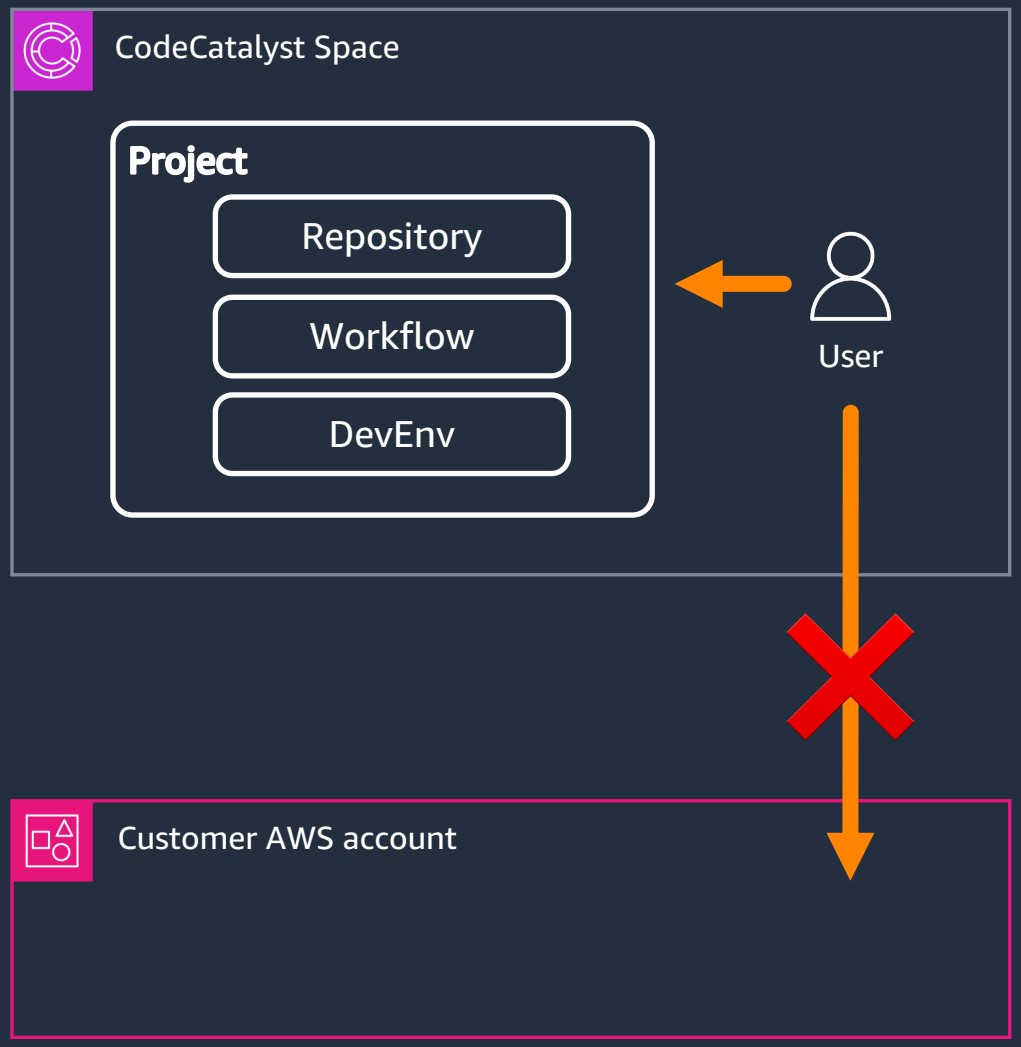
- リソースの名前や Tag に機密情報を含めない

利用開始前に公式ドキュメントのセキュリティのパートを確認

https://docs.aws.amazon.com/codecatalyst/latest/userguide/security.html

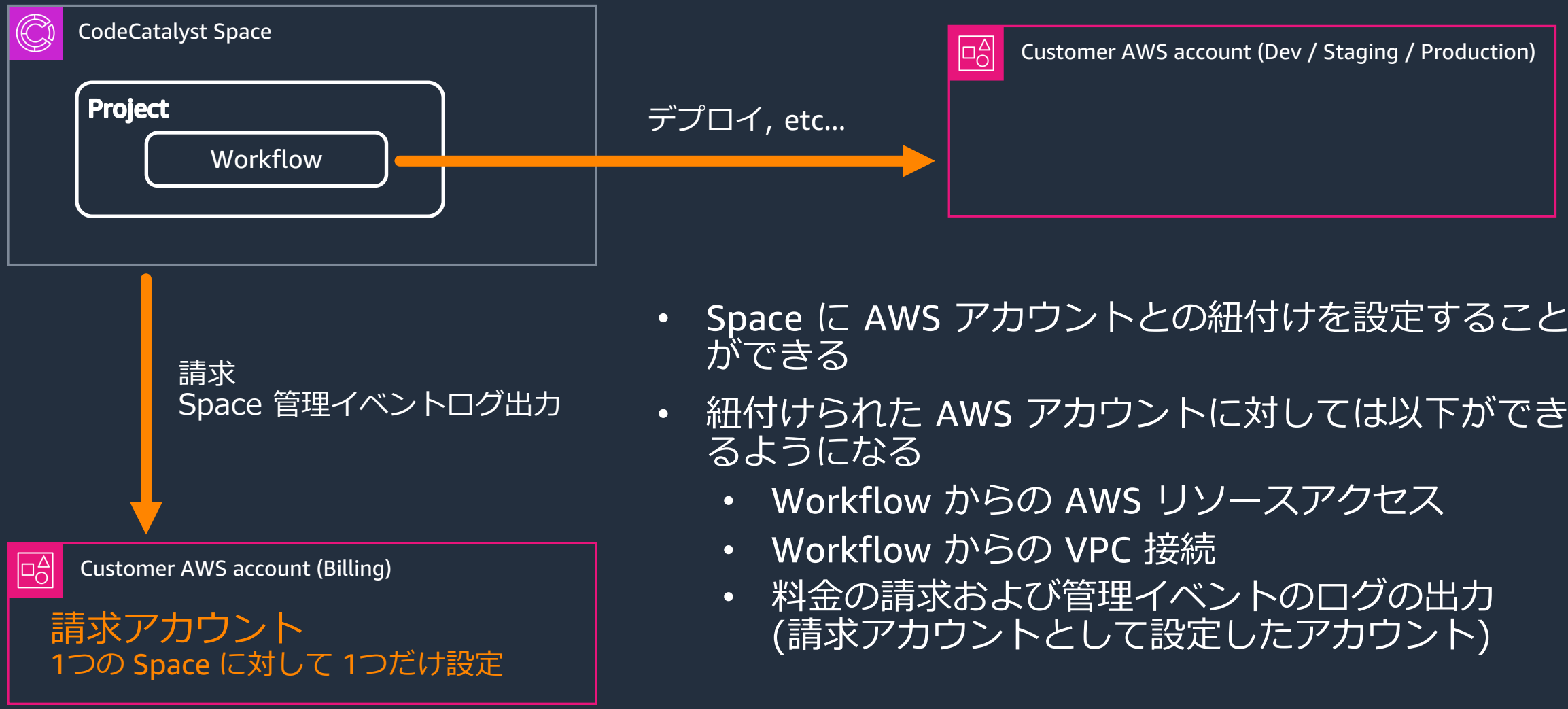
Space とお客様 AWS アカウントの関係

お客様の AWS アカウントとの分離



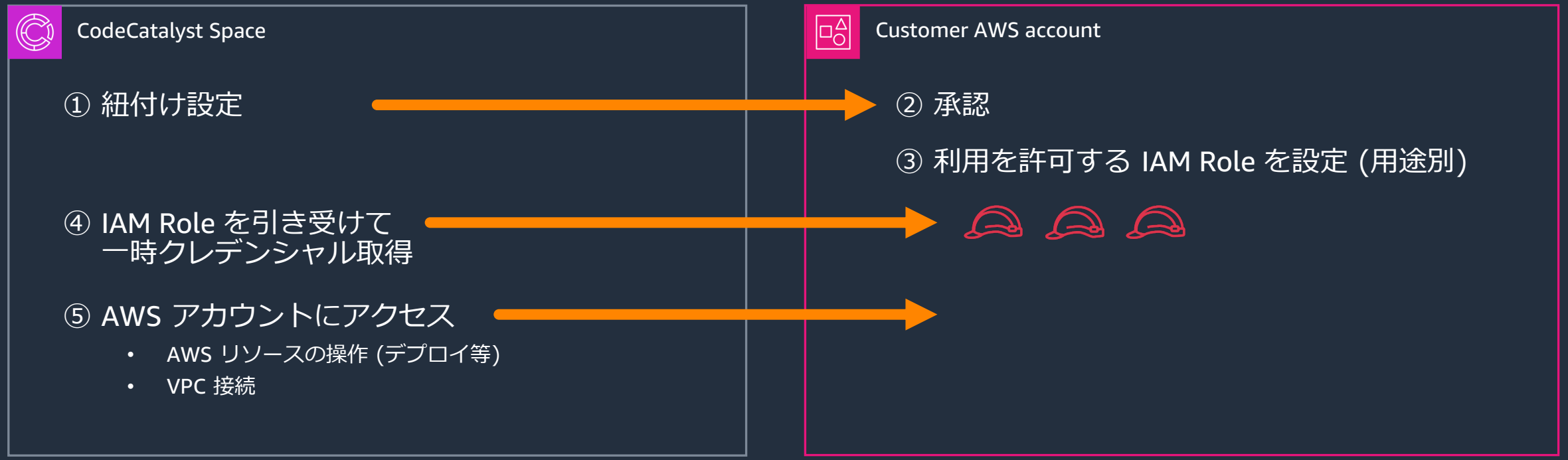
- CodeCatalyst の Space はお客様の AWS アカウントの外部に存在する
- Space の ユーザーはお客様の AWS アカウントのリソースに直接アクセスすることはできない（IAM のクレデンシャルをもたない）

お客様の AWS アカウントとの紐付け



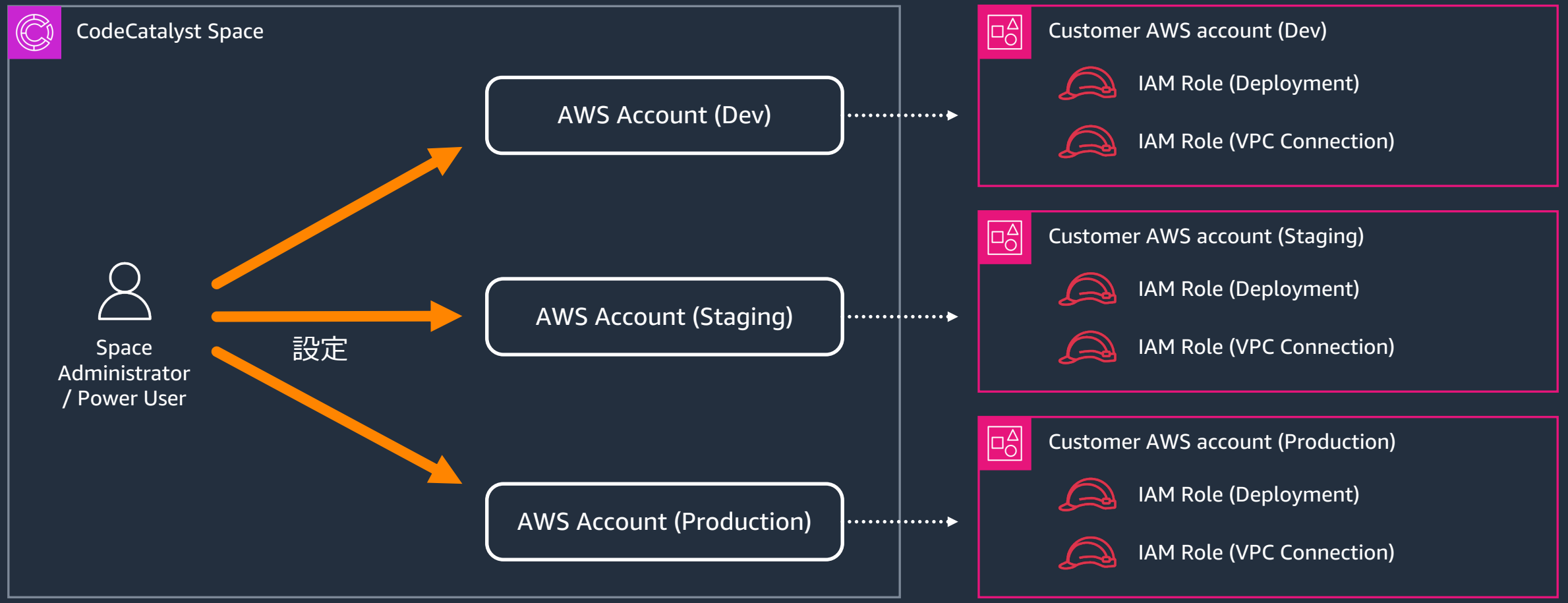
- Space に AWS アカウントとの紐付けを設定することができる
- 紐付けられた AWS アカウントに対しては以下ができるようになる
  - Workflow からの AWS リソースアクセス
  - Workflow からの VPC 接続
  - 料金の請求および管理イベントのログの出力
    (請求アカウントとして設定したアカウント)

お客様の AWS アカウントとの紐付け



- 紐付け設定の完了には AWS アカウント側の承認が必要
- Space では AWS アカウントに許可された IAM Role の中から必要なものを選択して引き受け、AWS リソースにアクセスする

お客様の AWS アカウントとの紐付け



- Space と AWS アカウントの紐付けは Space レベルで管理する
- 設定には Power User 以上の権限が必要

ユーザーおよび権限

CodeCatalyst のユーザー

Space ごとにどちらをサポートするかを選択

- AWS Builder ID
- AWS IAM Identity Center を利用したフェデレーション

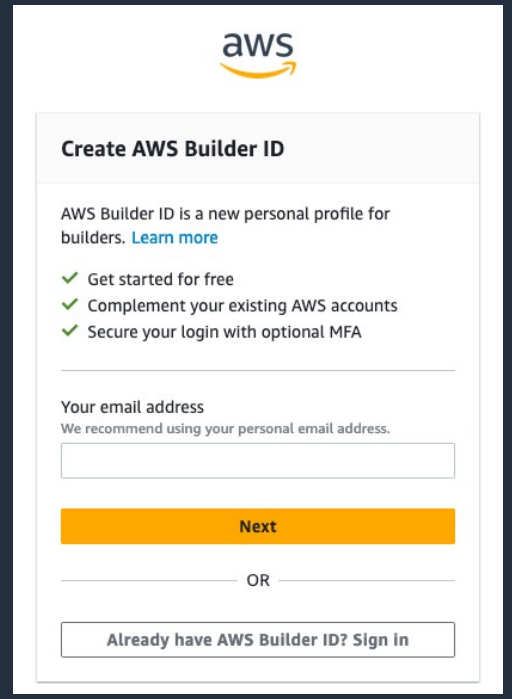
両方を同時に有効にすることはできない

Space 作成時に選択、作成後に変更も可能

AWS Builder ID

- AWS 上の開発者のための個人 ID (無料)
  - CodeCatalyst のほか、Amazon CodeWhisperer や AWS Training and Certification で利用可能
  - MFA (多要素認証)にも対応

- CodeCatalyst では複数の Space に関連づけることが可能



- 同じ開発者が複数の Space を利用する場合でも Builder ID は 1 つでよい
- CodeCatalyst では、Global でユニークな alias を Builder ID に関連付け、User 同士は alias でお互いを識別しあう

https://docs.aws.amazon.com/ja_jp/signin/latest/userguide/sign-in-aws_builder_id.html

内容の都合上
IAM Identity Center を利用したフェデレーションの前に
Role / Team の説明をします

Role

- Role = User に割り当てる権限セット
- CodeCatalyst では Space レベルと Project レベルの両方で User に Role を付与して権限を管理する
- すべての User は Space レベルのいずれかの Role を付与される

Space Role	Project Role
Space Administrator	Project Administrator
Power User	Contributor
Limited Access	Reviewer
	Read Only

Space Role

Role	Description
Space Administrator	• CodeCatalyst で最も強力な権限をもつ Role • Space 全体に対するすべての権限をもつ • Space Administrator Role に User を追加/削除できる唯一の Role
Power User	• Space の User やリソースの管理を支援する User 向けの Role • Project の作成はできるが、Project 内のリソースに対するアクセスはできない
Limited Access	• Space リソースの管理を行なう必要がない User にはこの Role を付与する • Space リソースに対する権限はほとんど持たない • 特定の Project 内から招待された User は、Space レベルではデフォルトでこの Role が付与される

https://docs.aws.amazon.com/codecatalyst/latest/userguide/ipa-roles.html

aws

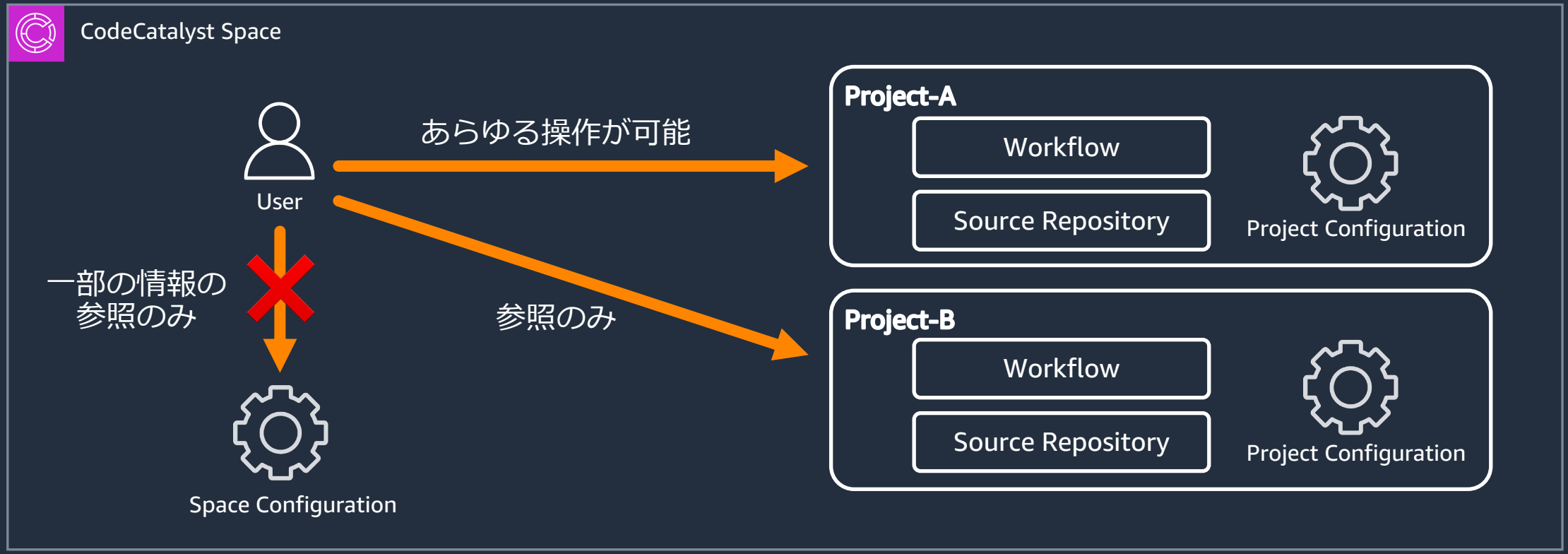
Project Role

Role	Description
Project Administrator	• Project に対して最も強力な権限をもつ Role • Project 設定の編集、Project への User 招待、User の Project 権限管理、Project の削除など あらゆる操作が可能
Contributor	• Project に参加する多くのメンバーが持つことを想定された Role • コード、Workflow、Issue などを操作できる
Reviewer	• Pull Request のレビューや Issue の管理を目的として Project にアクセスする User が持つことを想定された Role • コードや Workflow に対しては参照のみ可能
Read Only	• Project リソースの参照のみ可能な Role

https://docs.aws.amazon.com/codecatalyst/latest/userguide/ipa-roles.html

Role - ある User の例

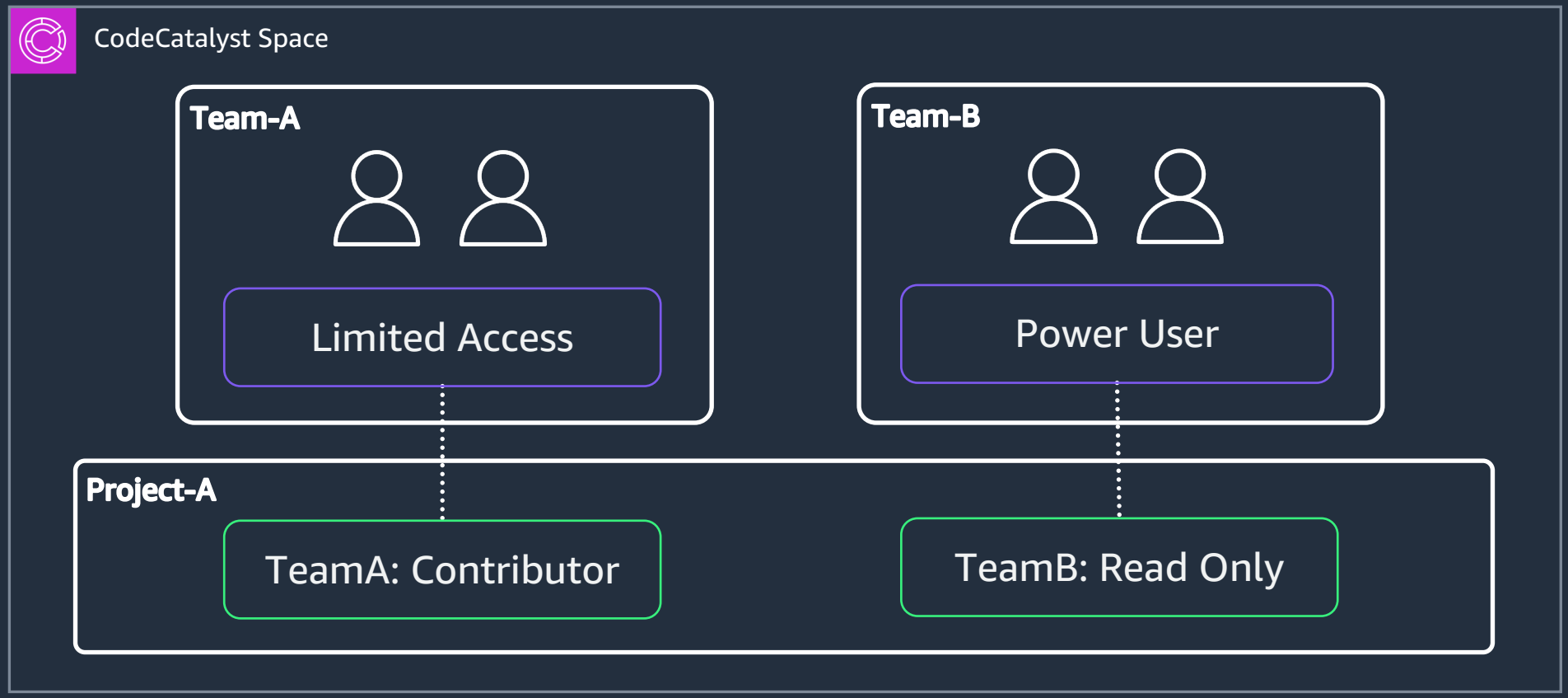
- Space Role: Limited Access
- Project-A Role: Project Administrator
- Project-B Role: Read Only



Team

- Team という単位で Space の User をグループ化できる
- Team に対して Role を割り当てることで、同じ役割の User の権限をまとめて管理できる
- Team の管理をするには Space Administrator の Role が必要

Team



- Project ごとに Team に対して Project Role を付与する
- Team に付与する Project Role を管理するには Project Administrator もしくは Space Administrator の Role が必要

Team のメンバー管理

Team のメンバーを管理する方法は 2 種類

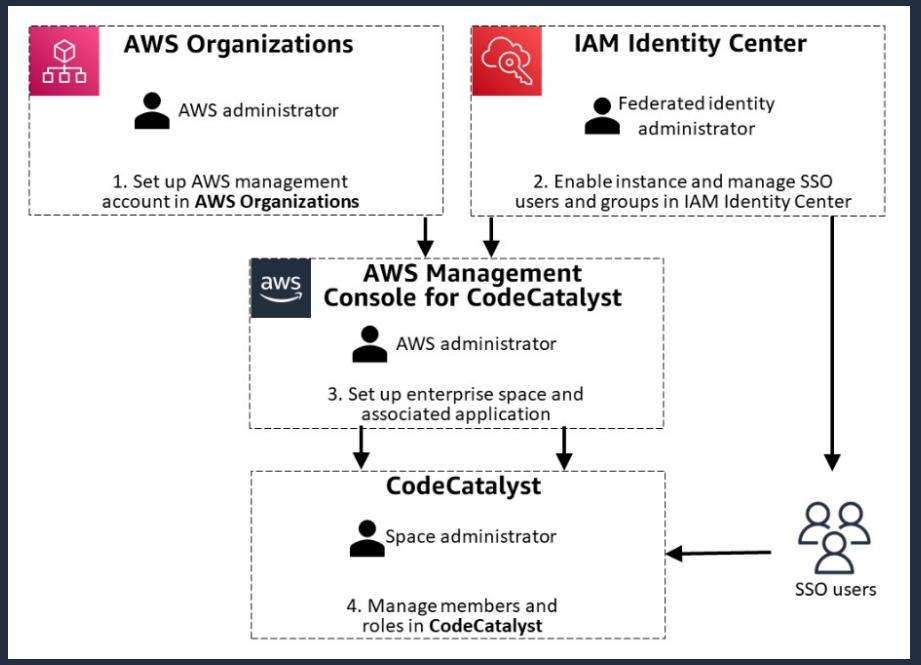
Team membership	Description
直接管理	• Team に直接メンバーを追加・削除
SSO Group による管理	• IAM Identity Center を利用したフェデレーションをサポートする Space 用 • IAM Identity Center で管理している Group を関連付け、その Group に参加する User を Team のメンバーとする

IAM Identity Center を利用したフェデレーション

- IAM Identity Center を利用したフェデレーションにより、企業や組織で管理された ID を利用して CodeCatalyst の Space にアクセスできる
  - IAM Identity Center の ID ソースとして外部の ID プロバイダーを構成すれば、外部の ID プロバイダー (AzureAD, Google Workspaces, Okta, etc...)で管理する ユーザーにも IAM Identity Center 経由で CodeCatalyst へのアクセスを提供できる

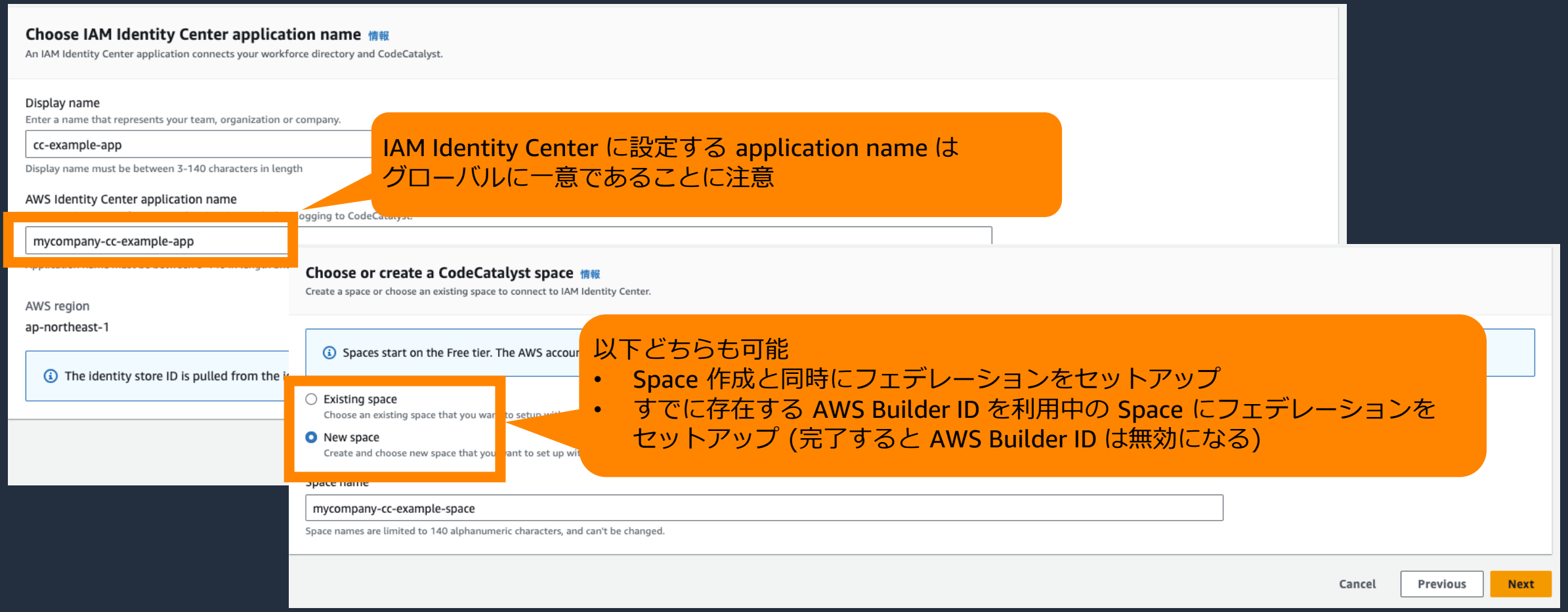
- IAM Identity Center は Organization インスタンスと Account インスタンス のどちらも選択可能

- 同一 Space 上での AWS Builder ID との併用は不可



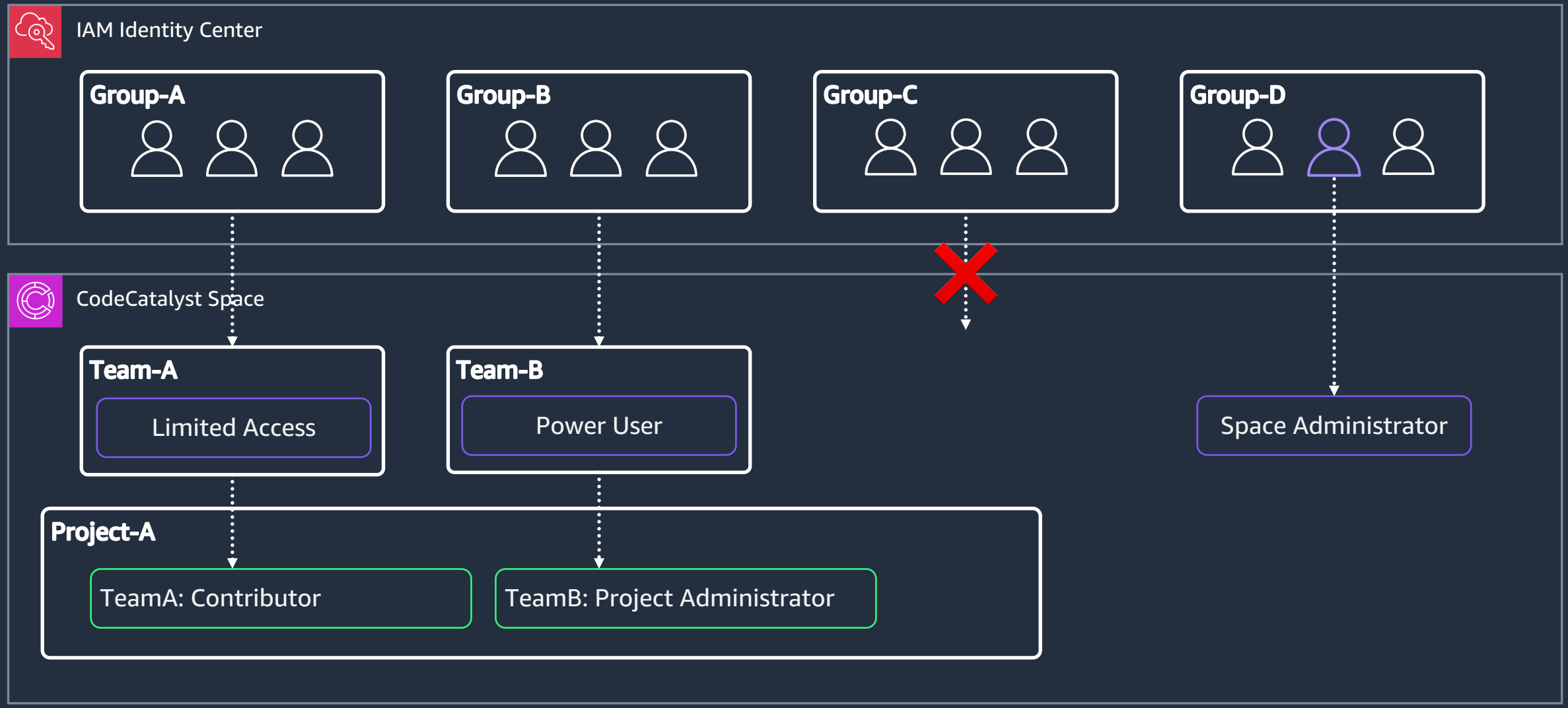
https://docs.aws.amazon.com/codecatalyst/latest/adminguide/concepts.html
https://docs.aws.amazon.com/codecatalyst/latest/adminguide/setting-up-federation.html
https://docs.aws.amazon.com/ja_jp/singlesignon/latest/userguide/supported-idps.html
https://docs.aws.amazon.com/ja_jp/singlesignon/latest/userguide/organization-instances-identity-center.html
https://docs.aws.amazon.com/ja_jp/singlesignon/latest/userguide/account-instances-identity-center.html

IAM Identity Center を利用したフェデレーション

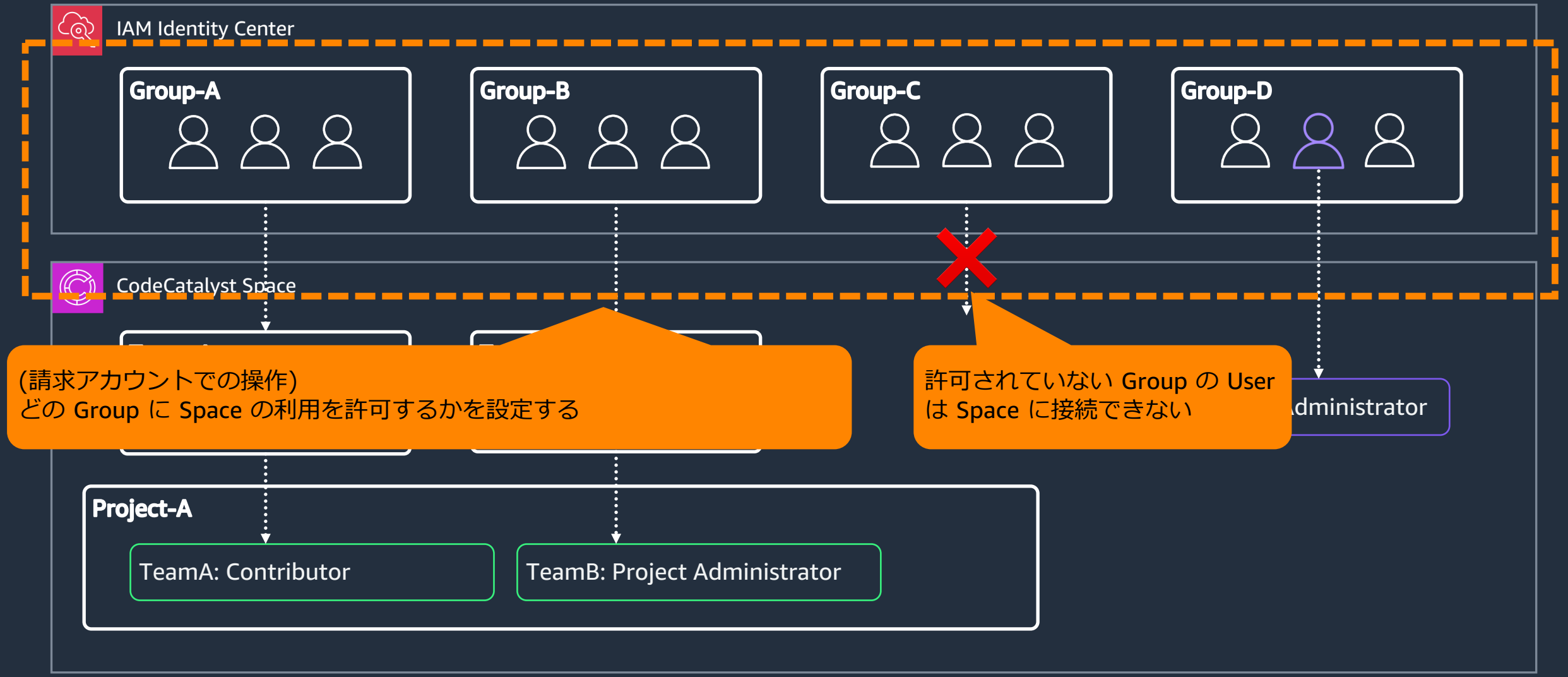


https://docs.aws.amazon.com/codecatalyst/latest/adminguide/concepts.html
https://docs.aws.amazon.com/codecatalyst/latest/adminguide/setting-up-federation.html

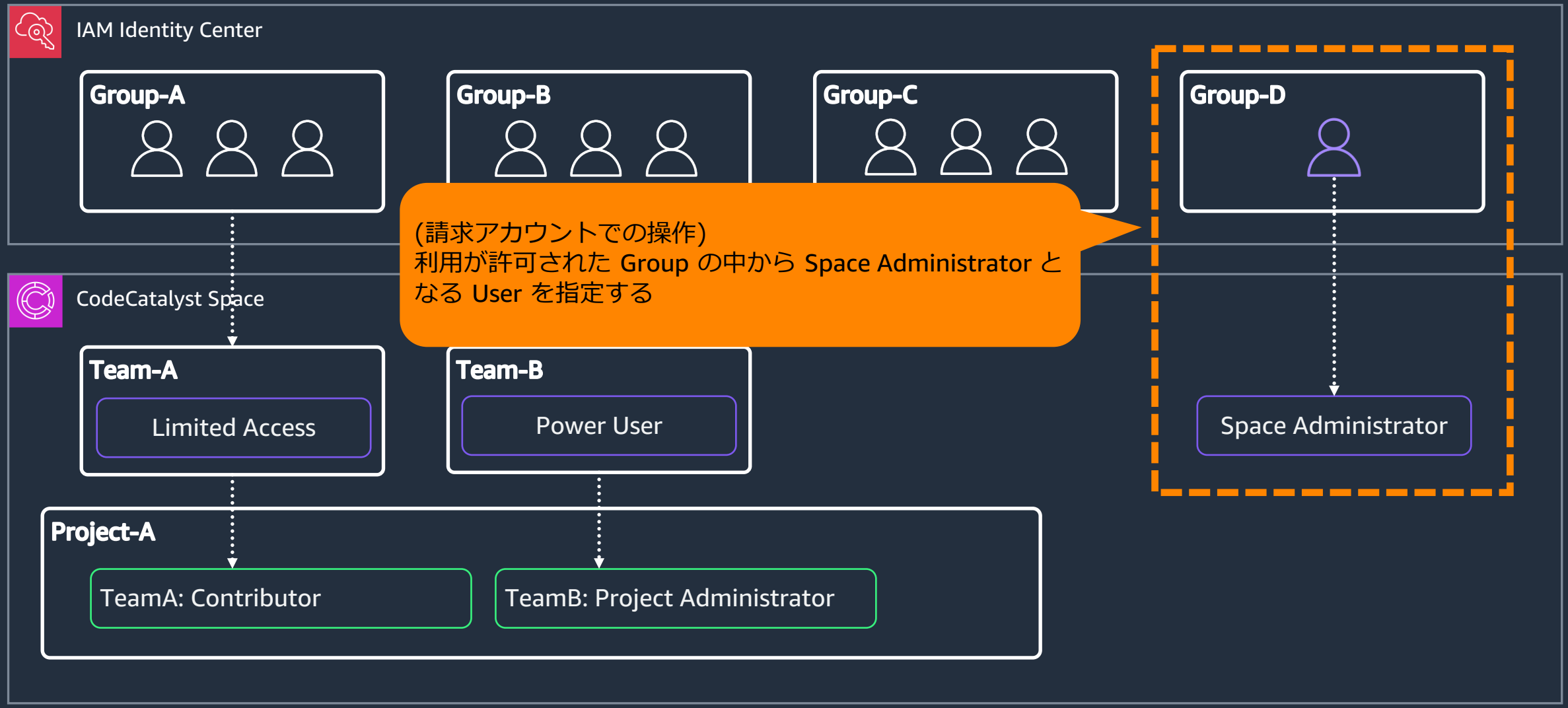
IAM Identity Center を利用したフェデレーションの例



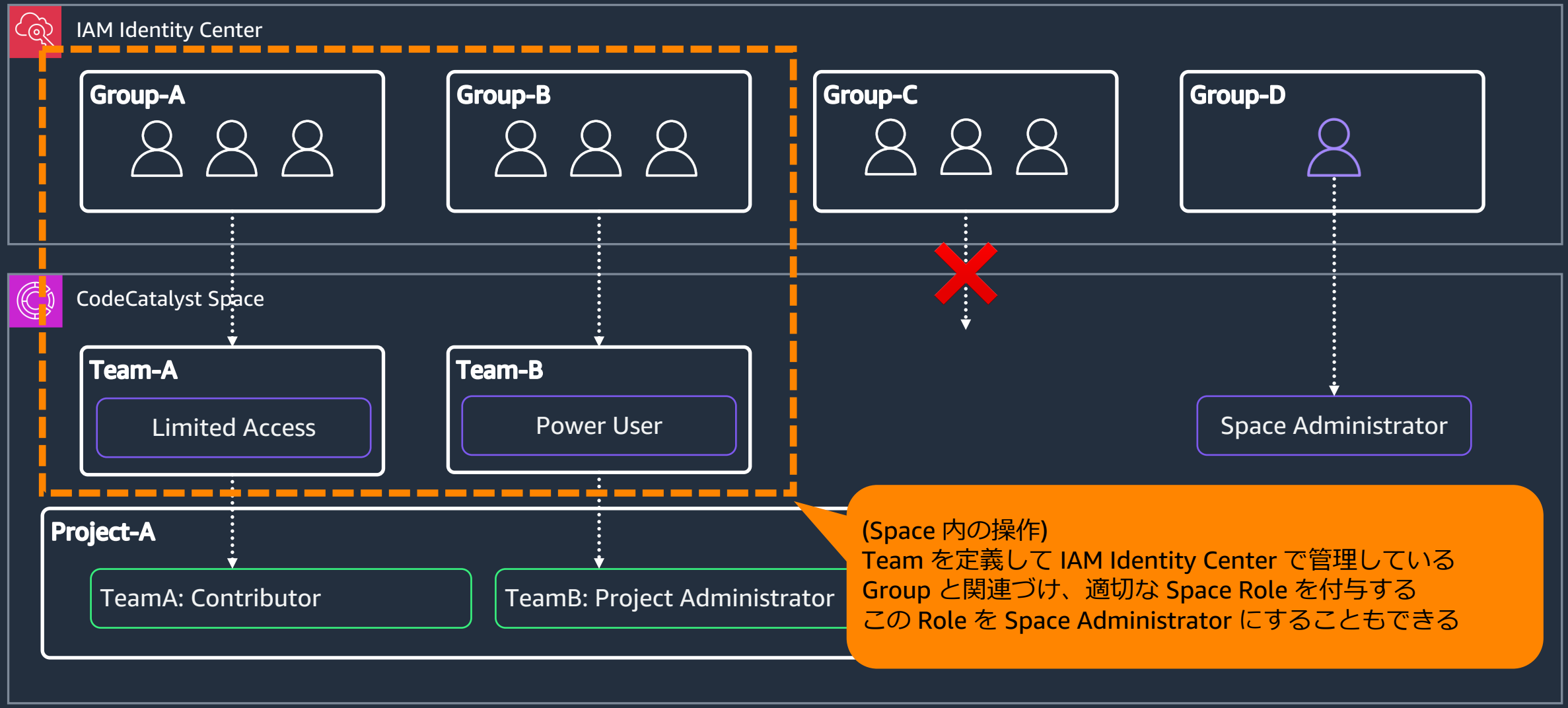
IAM Identity Center を利用したフェデレーションの例



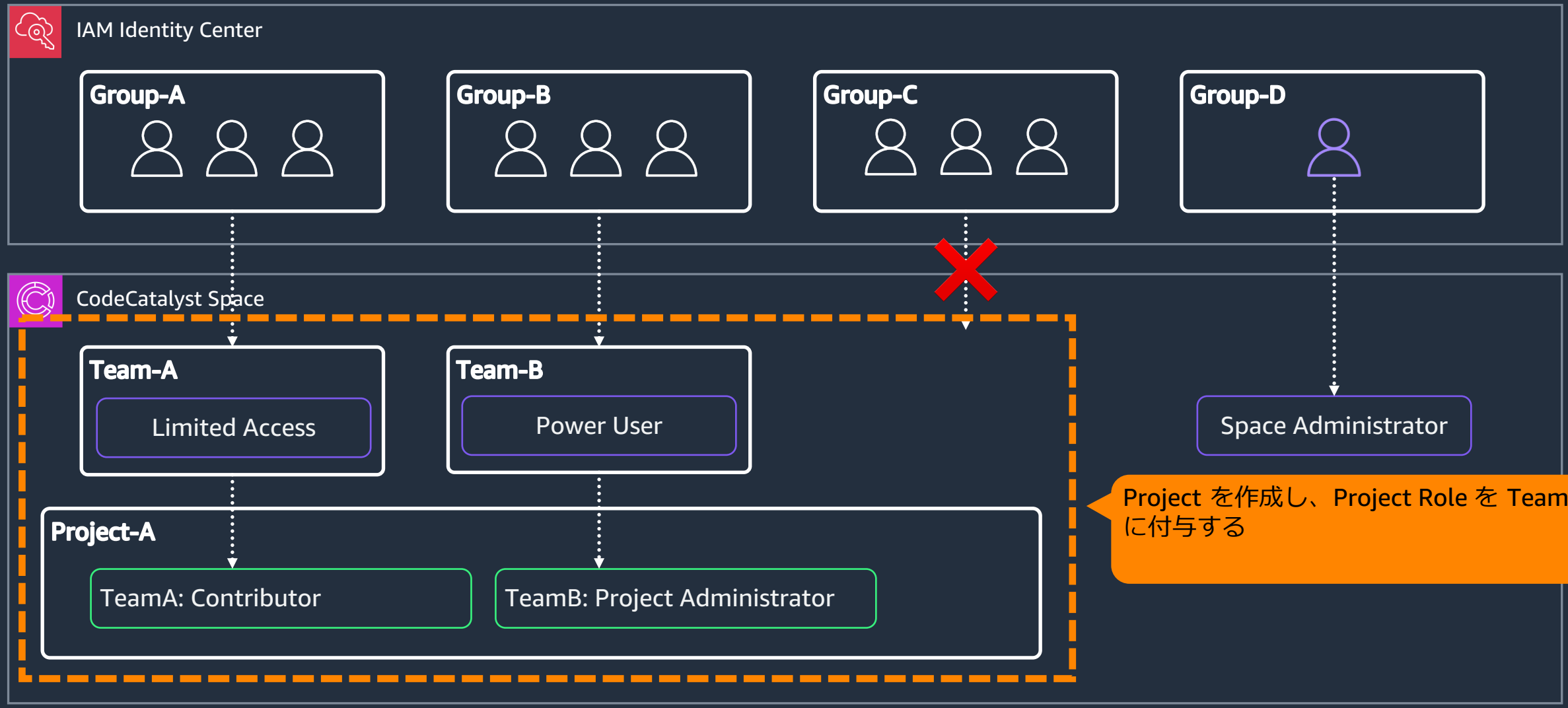
IAM Identity Center を利用したフェデレーションの例



IAM Identity Center を利用したフェデレーションの例

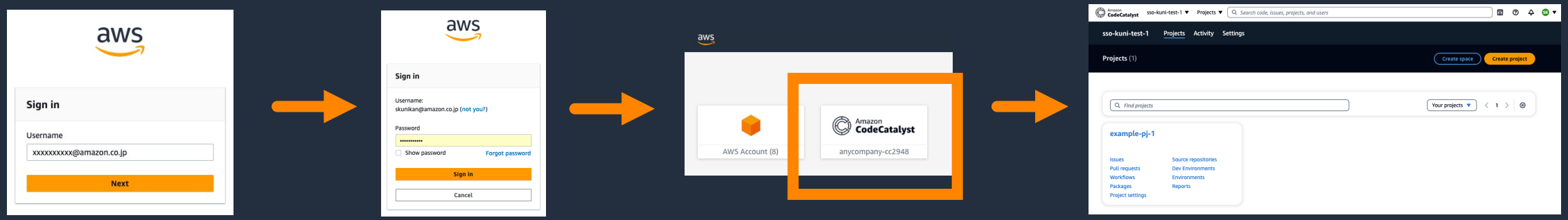


IAM Identity Center を利用したフェデレーションの例

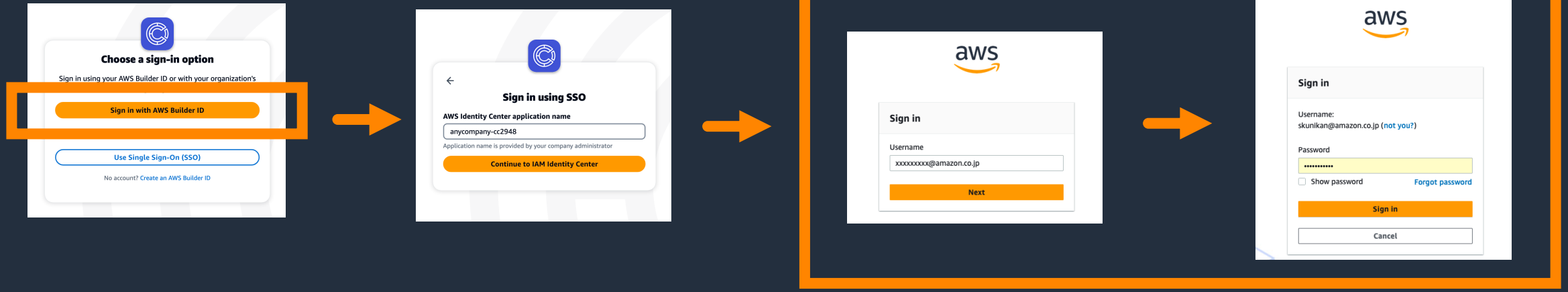


IAM Identity Center を利用したフェデレーション

AWS アクセスポータルからアクセス



CodeCatalyst 共通 Login フォームからアクセス



IAM Identity Center の Sign In フォーム

CLI/SDK での CodeCatalyst 操作

- CodeCatalyst のリソースには Space の User でアクセスする必要がある (IAM のクレデンシャルではアクセスできない)
- AWS Builder ID もしくは IAM Identity Center を利用する SSO の Profile を事前に作成しておく
- CLI/SDK では作成した Profile を利用してリソースにアクセスする

AWS Builder ID の場合の Profile 例

```
[profile codecatalyst]
region = us-west-2
sso_session = codecatalyst

[sso-session codecatalyst]
sso_region = us-east-1
sso_start_url = https://view.awsapps.com/start
sso_registration_scopes = codecatalyst:read_write
```

IAM Identity Center の場合の Profile 例

```
[profile codecatalyst]
region = us-west-2
sso_session = codecatalyst

[sso-session codecatalyst]
sso_region = ap-northeast-1
sso_start_url = https://d-0000000000.awsapps.com/start
sso_registration_scopes = codecatalyst:read_write
```

認証に使っている IAM Identity Center インスタンスのリージョンと Start URL

https://docs.aws.amazon.com/ja_jp/codecatalyst/latest/userguide/set-up-cli.html
https://docs.aws.amazon.com/cli/latest/userguide/sso-using-profile.html

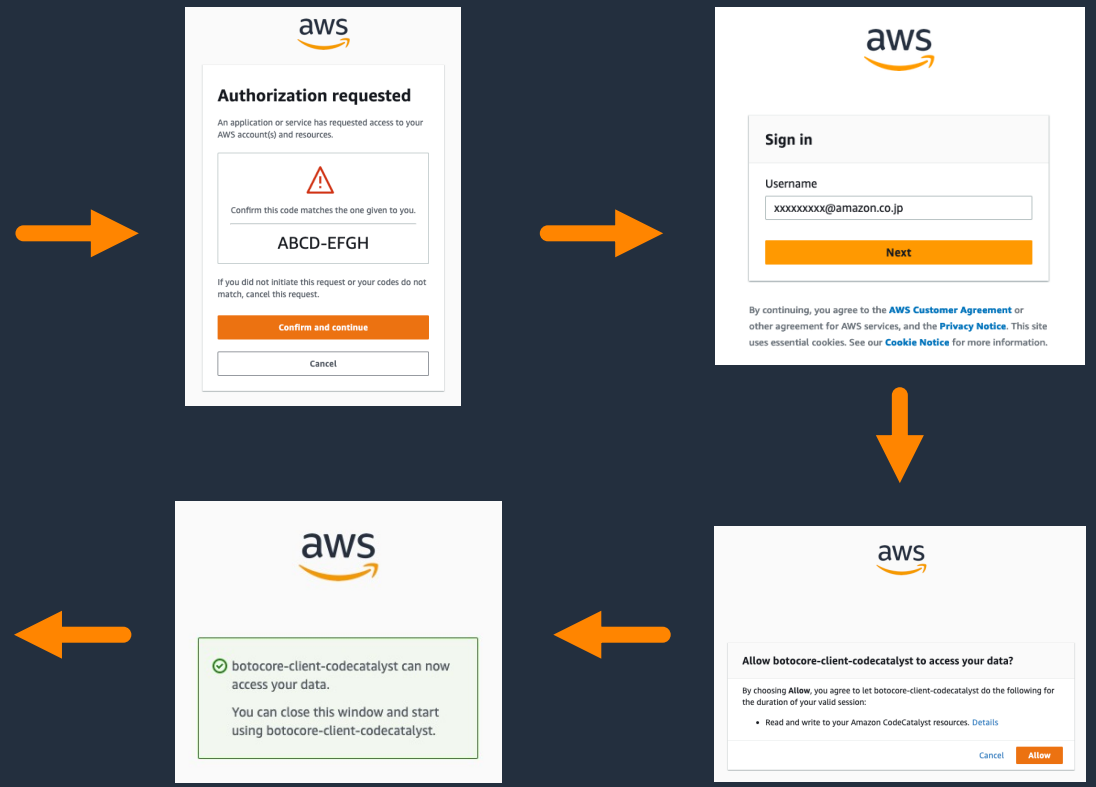
CLI/SDK での CodeCatalyst 操作

AWS Builder ID の場合はここで
AWS Builder ID の Sign In フォーム
が表示される

IAM Identity Center を利用したフェデレーションの場合の AWS CLI の例

```
$ aws sso login --profile codecatalyst
Attempting to automatically open the SSO authorization page in
your default browser.
If the browser does not open or you wish to use a different
device to authorize this request, open the following URL:

https://device.sso.ap-northeast-1.amazonaws.com/

Then enter the code:

ABCD-EFGH
Successfully logged into Start URL: https://d-
0000000000.awsapps.com/start

$ aws codecatalyst list-projects --profile codecatalyst \
>   --space-name [YOUR SPACE NAME]
```

(CodeCatalyst リソースへのアクセスが可能)

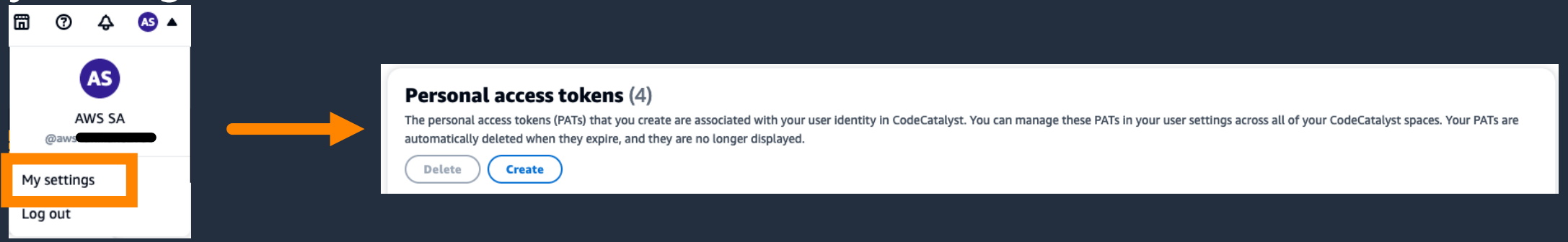


https://docs.aws.amazon.com/ja_jp/codecatalyst/latest/userguide/set-up-cli.html
https://docs.aws.amazon.com/cli/latest/userguide/sso-using-profile.html

PAT（Personal Access Token）

- ローカル Git ツールから CodeCatalyst のリポジトリへのアクセスには PAT (Personal Access Token) を利用する
- PATは User に関連付けられる
  - User が参加しているすべての Space/Project で共通して利用できる
- PAT 発行時は画面上に一度だけ表示される
  - ローカルコンピューターに安全に保存して利用する
  - 紛失または流出した場合は新しいものを作成し、既存のものを削除する

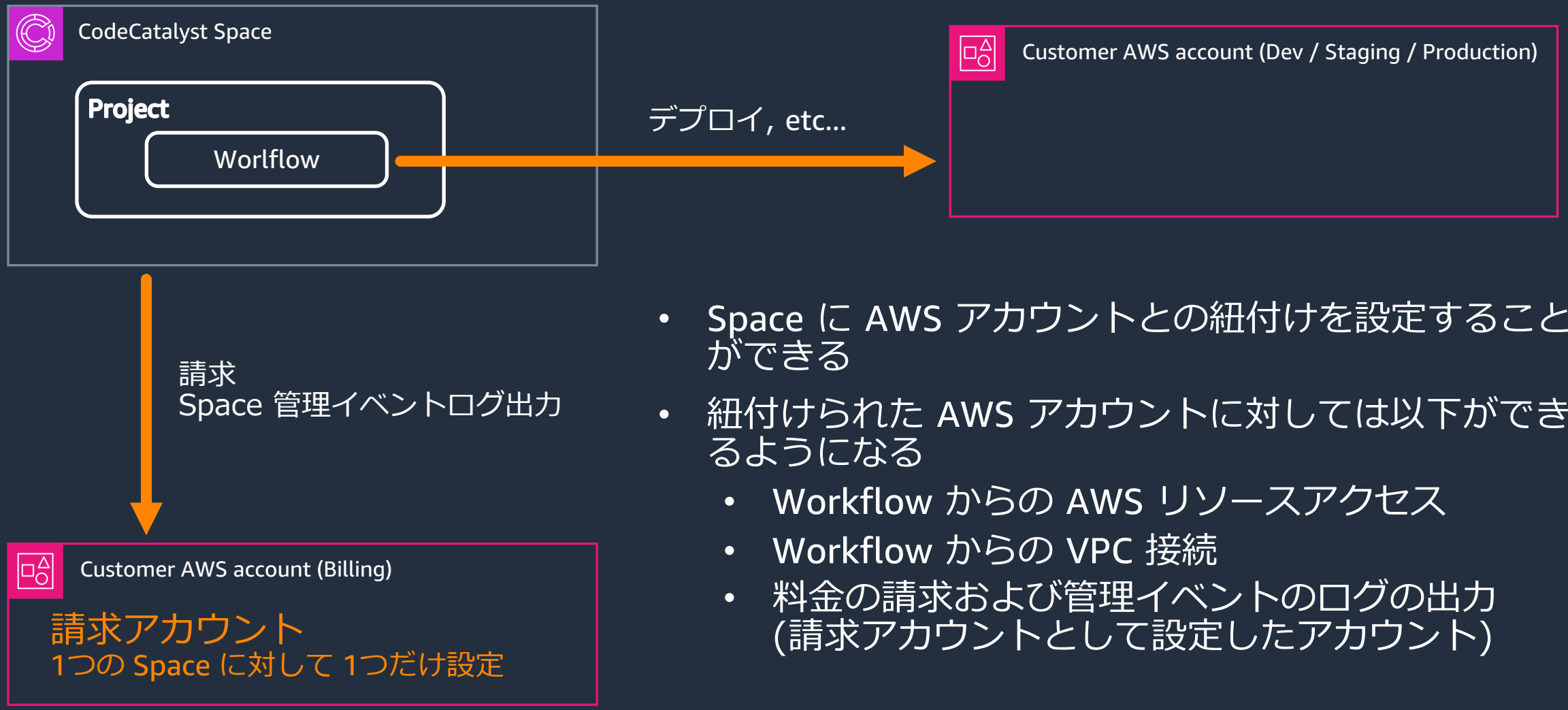
MySetting から発行



AWS アカウントへの接続

お客様の AWS アカウントとの紐付け

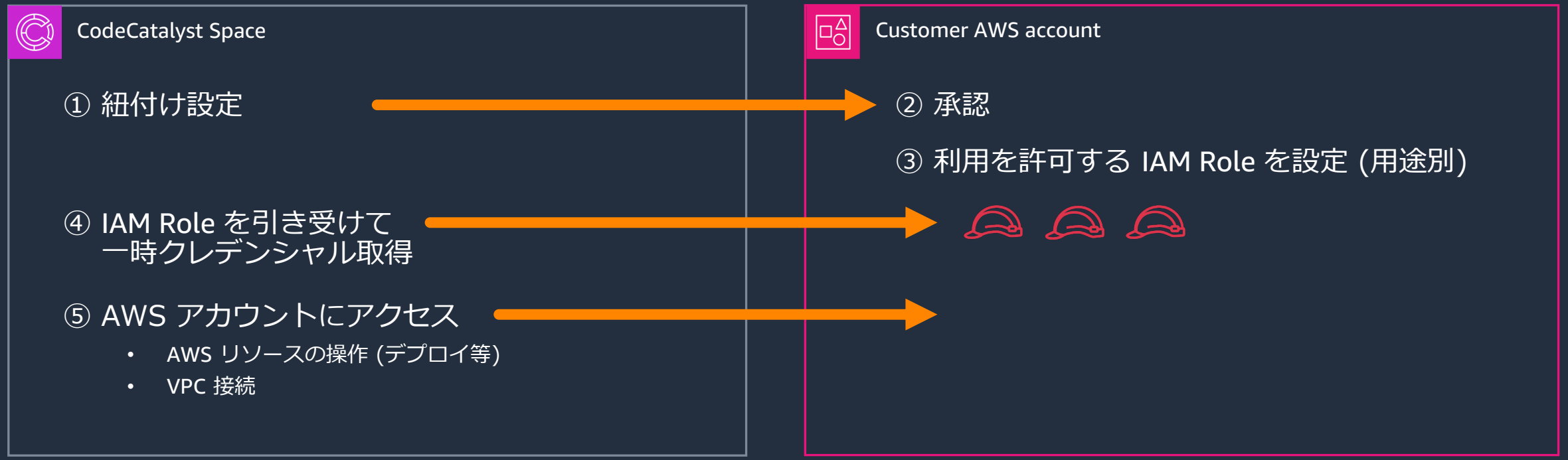
再掲



- Space に AWS アカウントとの紐付けを設定することができる
- 紐付けられた AWS アカウントに対しては以下ができるようになる
  - Workflow からの AWS リソースアクセス
  - Workflow からの VPC 接続
  - 料金の請求および管理イベントのログの出力
    (請求アカウントとして設定したアカウント)

お客様の AWS アカウントとの紐付け

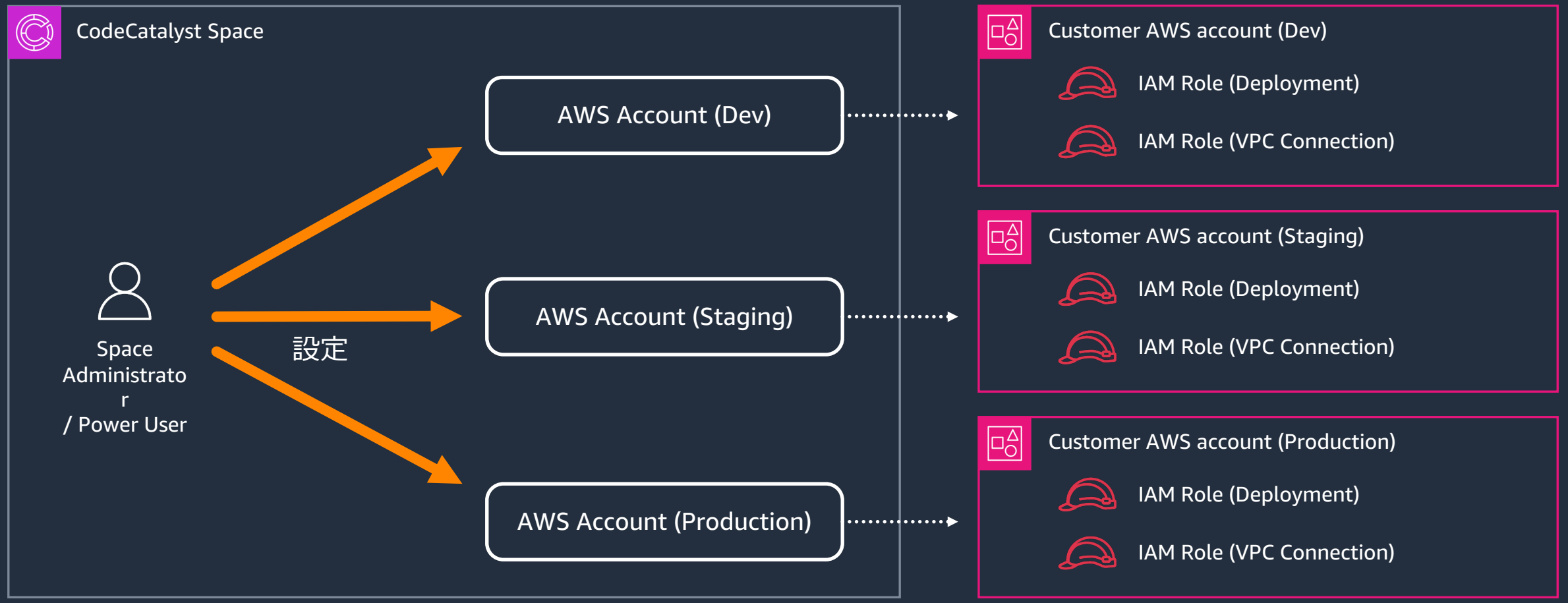
再掲



- 紐付け設定の完了には AWS アカウント側の承認が必要
- Space では AWS アカウントに許可された IAM Role の中から必要なものを選択して引き受け、AWS リソースにアクセスする

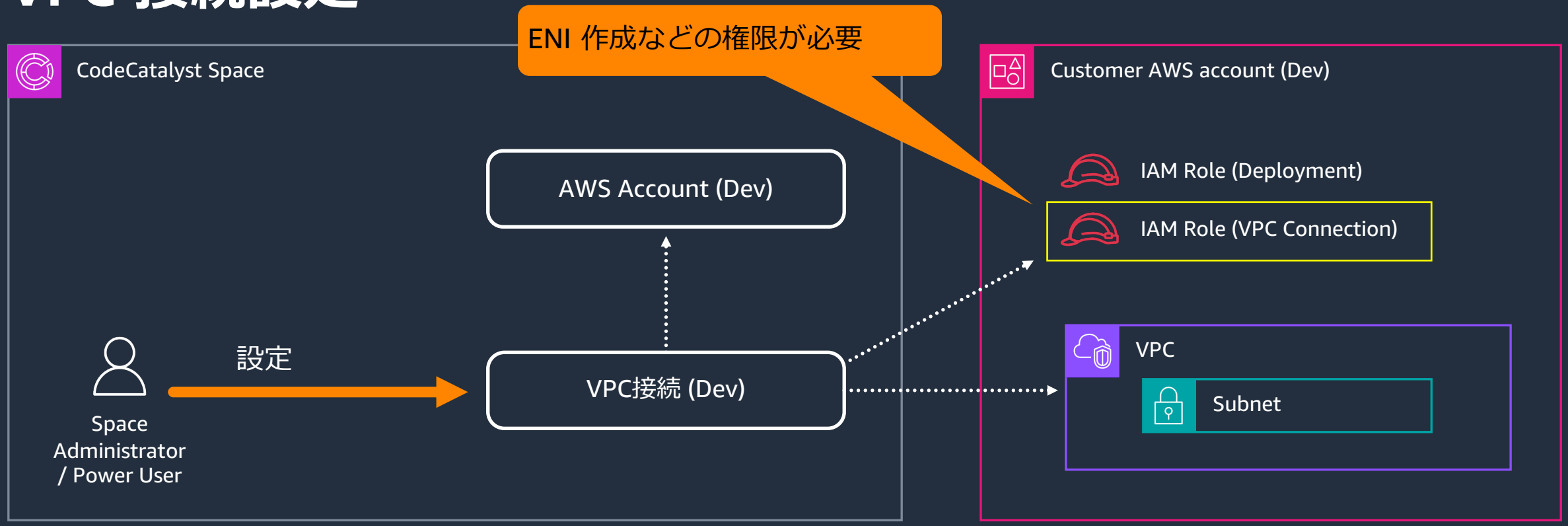
お客様の AWS アカウントとの紐付け

再掲



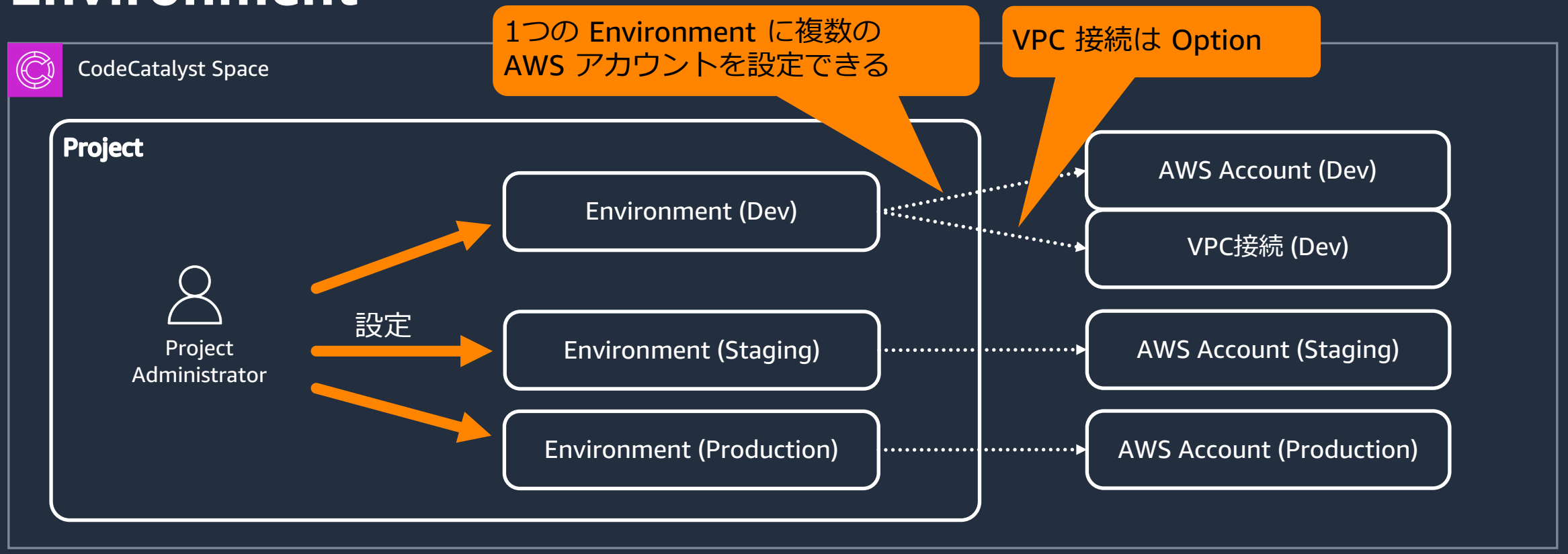
- Space と AWS アカウントの紐付けは Space レベルで管理する
- 設定には Power User 以上の権限が必要

VPC 接続設定



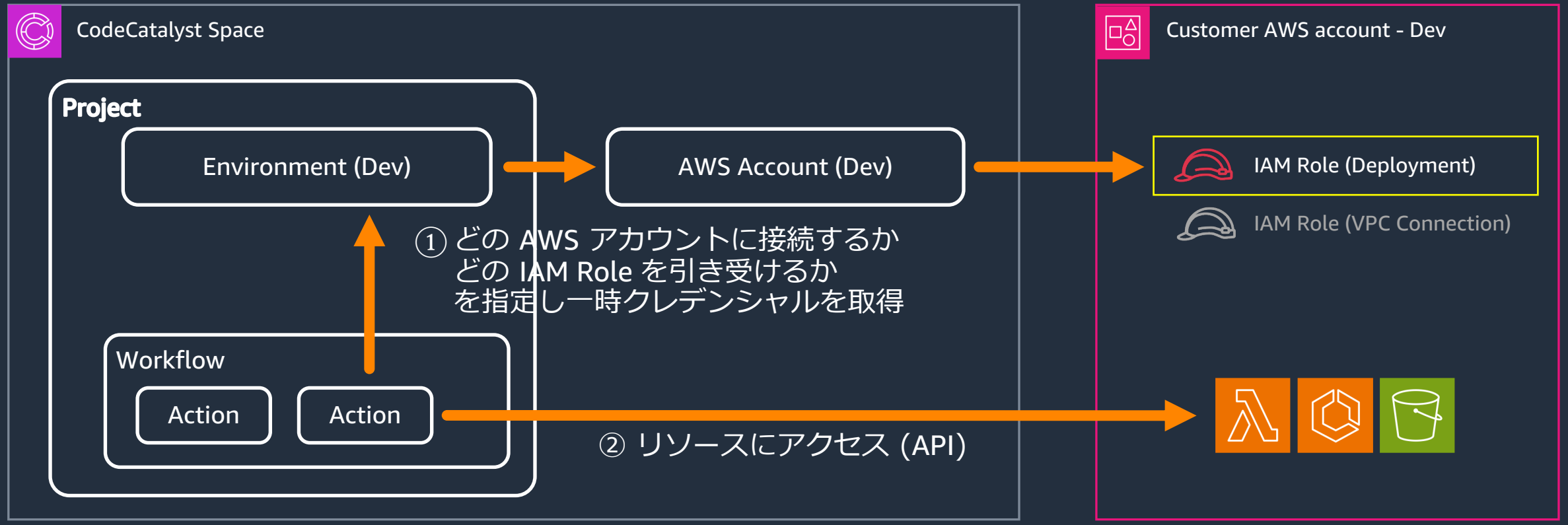
- VPC 接続設定の管理は Space レベルで行なう
- Space に紐付けられている AWS アカウントから 接続先 VPC/Subnet や 接続に利用する IAM Role を選択する

Environment



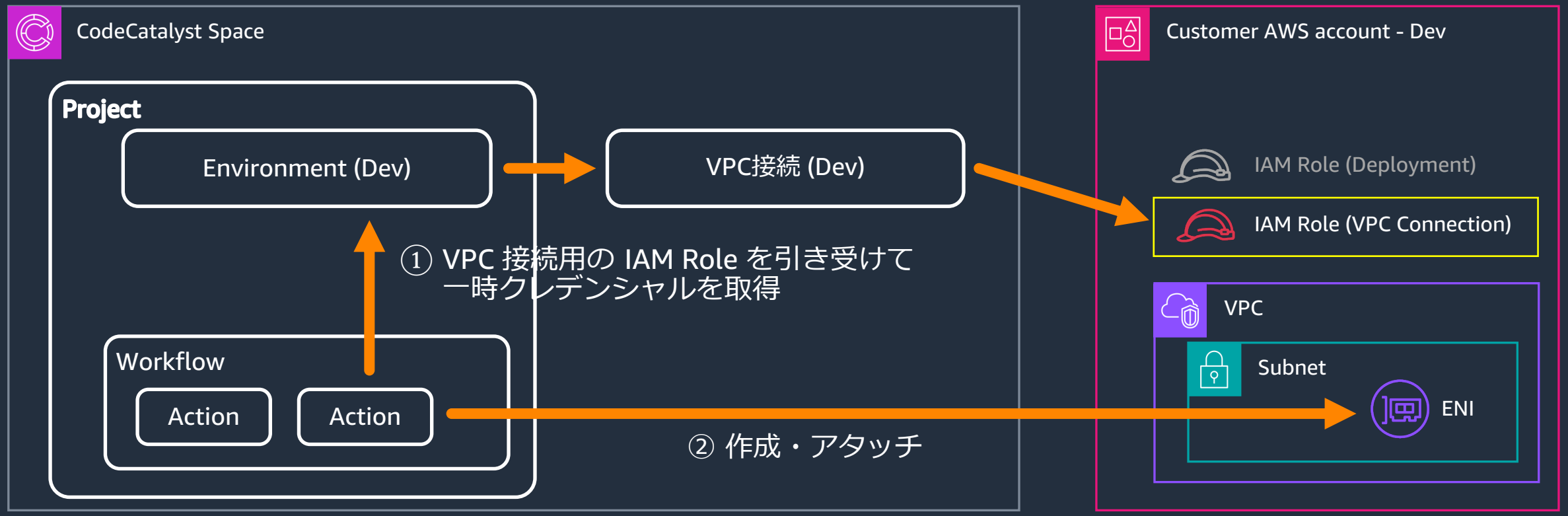
- Project 内では、接続先 AWS アカウントや VPC をまとめて「Environment」という単位で管理する
- 設定には Project Administrator の権限が必要
- Space で紐付けがされていても、Project 内で Environment として設定されていない AWS アカウントには Workflow からアクセスできない

Workflow から AWS リソースへのアクセス



- Workflow の中から Action ごとに Environment を指定し、その Environment を使ってアクセス可能な AWS アカウントの中からアクセス先アカウントを選択
- アクセス先アカウントで利用可能な IAM Role の中から用途に応じた適切なものを選択して引き受け、AWS リソースにアクセス

Workflow からの VPC 接続



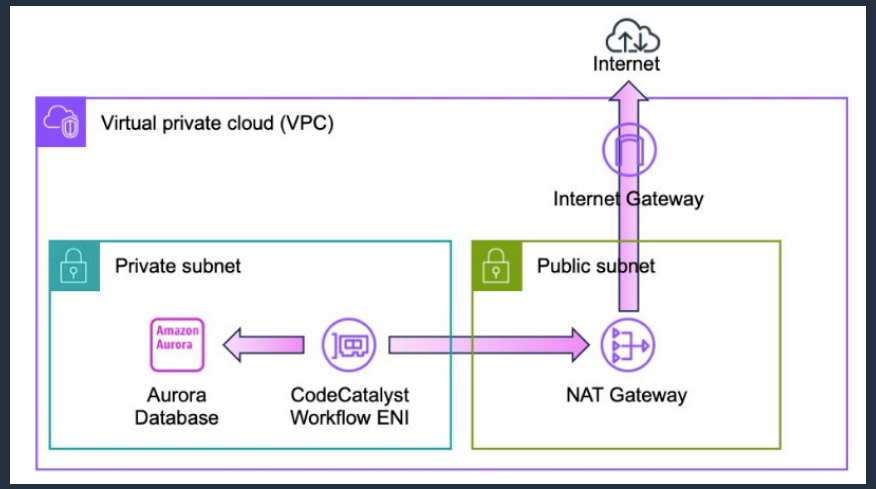
- Environment に VPC 接続が設定されている場合は、予め設定された IAM Role を引き受けて VPC に接続 (接続先 VPC にENI を作成し、Action 実行環境にアタッチ)
- Space で デフォルト VPC 接続が設定されている場合は、Space 内のすべての Workflow Action が明示的な設定なしで自動的にその VPC に接続する
  (Environment の指定で上書き可能)

Workflow からの VPC 接続

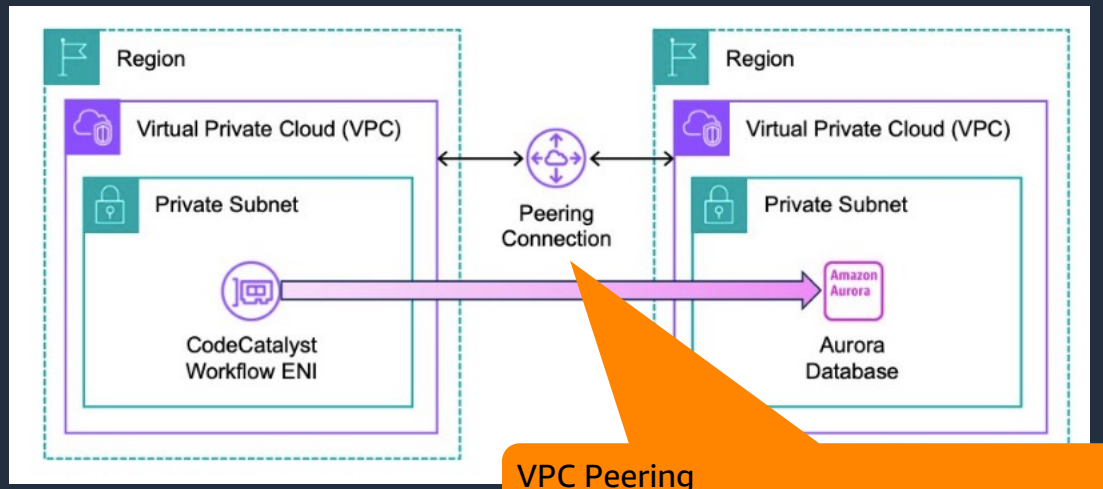
制限事項

- Space と同じリージョンの VPC にのみ接続できる
- Compute Type: Lambda の Workflow Action からは接続できない
- Operating System: Windows の Compute を使う Workflow Action からは接続できない
- CodeCatalyst が VPC 内に作成する ENI に Public IP を付与することはできない

Internet への経路を用意する例



別のリージョンへのアクセスを実現する例



VPC Peering
Or TransitGateway + TransitGateway Peering

https://docs.aws.amazon.com/codecatalyst/latest/adminguide/managing-vpcs.html
https://docs.aws.amazon.com/codecatalyst/latest/adminguide/managing-vpcs.set-up.html
https://aws.amazon.com/jp/blogs/devops/using-amazon-codecatalyst-with-amazon-virtual-private-cloud/

IAM Role – 信頼ポリシー

CodeCatalyst の Space からの利用を許可する IAM Role 信頼ポリシーの例

```
"Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
             "Principal": {
                "Service": [
                    "codecatalyst-runner.amazonaws.com",
                    "codecatalyst.amazonaws.com"
                ]
            },
            "Action": "sts:AssumeRole",
            "Condition": {
                "ArnLike": {
                    "aws:SourceArn": "arn:aws:codecatalyst:::space/<spaceId>/project/*"
                }
            }
        }
    ] >
```

CodeCatalyst の Service Principal を指定

特定の Space のすべての Project からの利用を許可

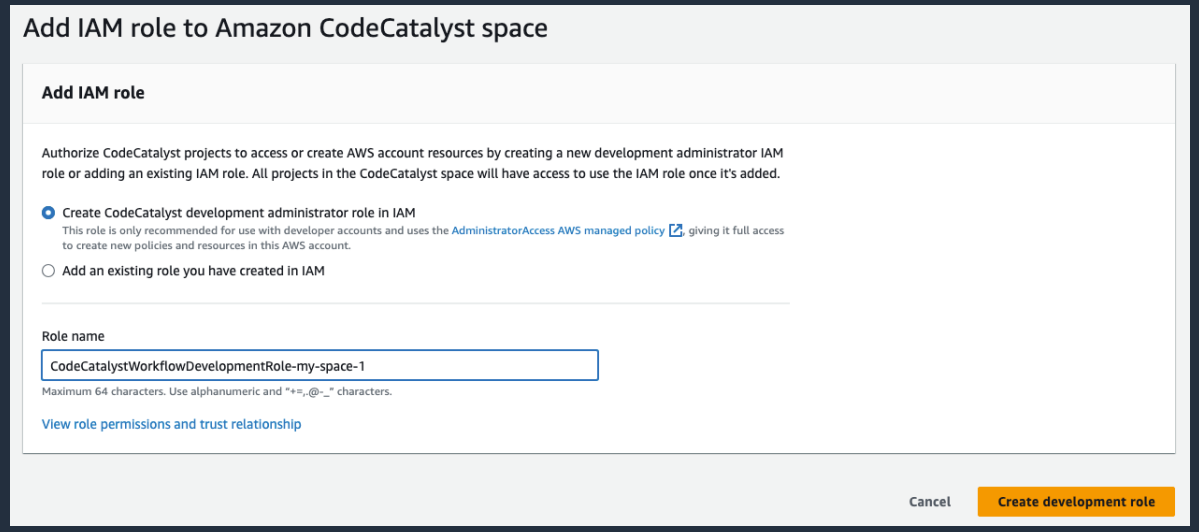
```
"Condition": {
                "StringEquals": {
                    "aws:SourceArn": "arn:aws:codecatalyst:::space/<spaceId>/project/<projectId>"
                }
            }
```

特定の Project に制限する場合の例

https://docs.aws.amazon.com/codecatalyst/latest/userguide/trust-model.html
https://docs.aws.amazon.com/codecatalyst/latest/userguide/security-iam.html

CodeCatalyst development administrator role

- アカウントに紐付けられた Space から利用できる IAM Role をマネジメントコンソールから数クリックで作成
- Space 内のすべての Project から利用可能
- AdministratorAccess の AWS Managed Policy がアタッチされる
- 開発アカウント以外での利用は非推奨



https://docs.aws.amazon.com/ja_jp/codecatalyst/latest/userguide/ipa-iam-roles.html

クォータ

クォータ

項目	上限
User の alias	3-100文字の間の長さ 半角英数字のみ (大文字/小文字 両方可)
1 User あたりの PAT 発行可能数	100
1 User が 1 日に招待できるメンバー数 (※)	500
1 つの E-Mail アドレスに対して 1 日に招待を送信できる数(※)	25
メンバー招待の有効期間(※)	24時間
メンバー招待時の E-Mail 検証の有効期間(※)	10分

※ AWS Builder ID で User を管理する Space の場合に適用

詳細は公式ドキュメントを参照してください
https://docs.aws.amazon.com/codecatalyst/latest/userguide/ipa-quotas.html

Thank you!