

JAPAN | JUNE 20, 2024

aws SUMMIT



DOL-3

# Amazon CodeCatalyst カスタムブループリントを用いた ゴールデンパスの構築と管理

**Ryuji Hori 堀 竜慈**

技術統括本部 エンタープライズ技術本部  
サービスグループ トラベル・交通・物流ソリューション部  
ソリューションアーキテクト





# 対象者、本セッションで学べること

- 対象者

- アプリケーションやサービスの開発、運用に関わる方
- Amazon CodeCatalyst の基礎をご存じの方

- 本セッションで話すこと

- ゴールデンパスとして Amazon CodeCatalyst カスタムブループリントを活用するメリットと方法

- 話さないこと

- ゴールデンパス、プラットフォームエンジニアリングについての詳細
- Amazon CodeCatalyst の基礎知識

# 参考情報

- プラットフォームエンジニアリングについて  
学べるセミナー
  - <https://aws.amazon.com/jp/blogs/news/20240229-platform-engineering-event/>
- AWS Black Belt Seminar  
Amazon CodeCatalyst Overview 編
  - [https://pages.awscloud.com/rs/112-TZM-766/images/AWS-Black-Belt\\_2023\\_Amazon-CodeCatalyst-Overview\\_1031\\_v1.pdf](https://pages.awscloud.com/rs/112-TZM-766/images/AWS-Black-Belt_2023_Amazon-CodeCatalyst-Overview_1031_v1.pdf)



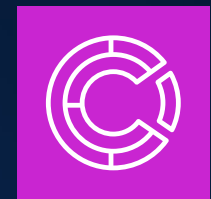
# 自己紹介

名前：堀 竜慈 (Hori Ryuji)

所属：アマゾンウェブサービスジャパン合同会社  
技術統括本部 エンタープライズ技術本部  
サービスグループ  
トラベル・交通・物流ソリューション部  
ソリューションアーキテクト

経歴：国内電機メーカーでシステムインテグレータ、  
ベンチャー企業でアプリケーション開発

好きな AWS サービス：Amazon CodeCatalyst



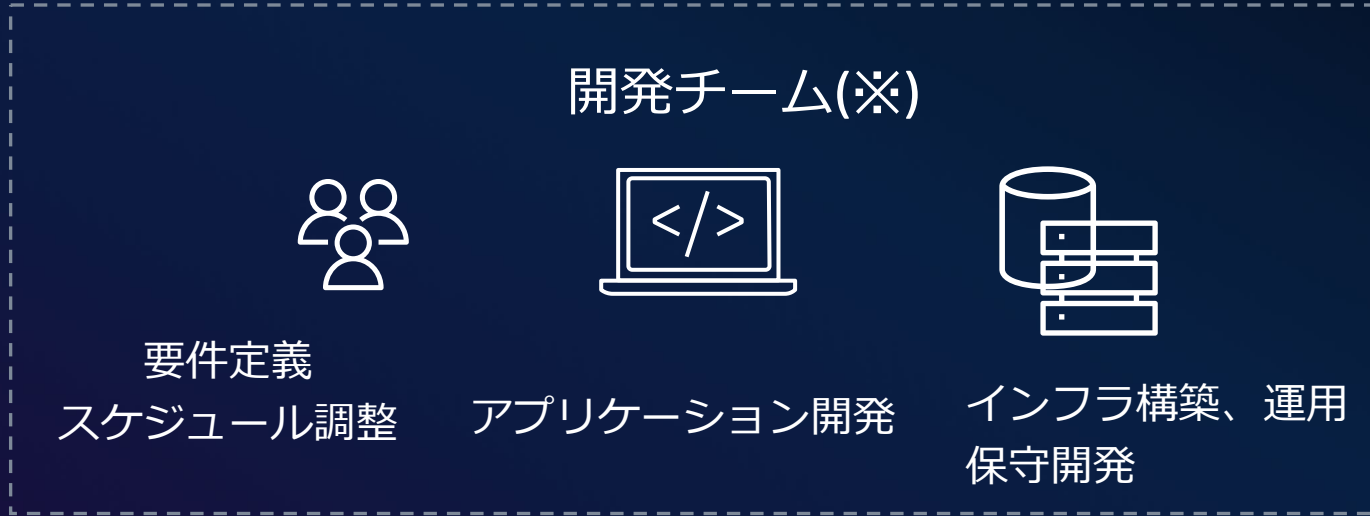
**Amazon CodeCatalyst**

# Agenda

1. 背景：プラットフォームエンジニアリングとゴールデンパスについて
2. ゴールデンパスとしての Amazon CodeCatalyst カスタムブループリントとは？
3. Amazon CodeCatalyst カスタムブループリントの構築
4. Amazon CodeCatalyst カスタムブループリントの更新と管理 **(with DEMO)**

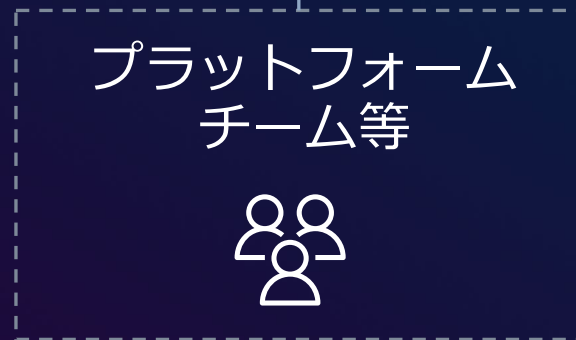
# プラットフォームエンジニアリングと ゴールデンパスについて

# プラットフォームエンジニアリングとは



開発チーム(※)の開発者体験と生産性を向上させることを目的として、開発チームにゴールデンパスを効率的に提供するアプローチ

ゴールデンパスの提供



負荷の吸収、軽減

## ゴールデンパスとは??

※ 価値のある単一の仕事の流れに沿って働くフルスタックな開発チーム

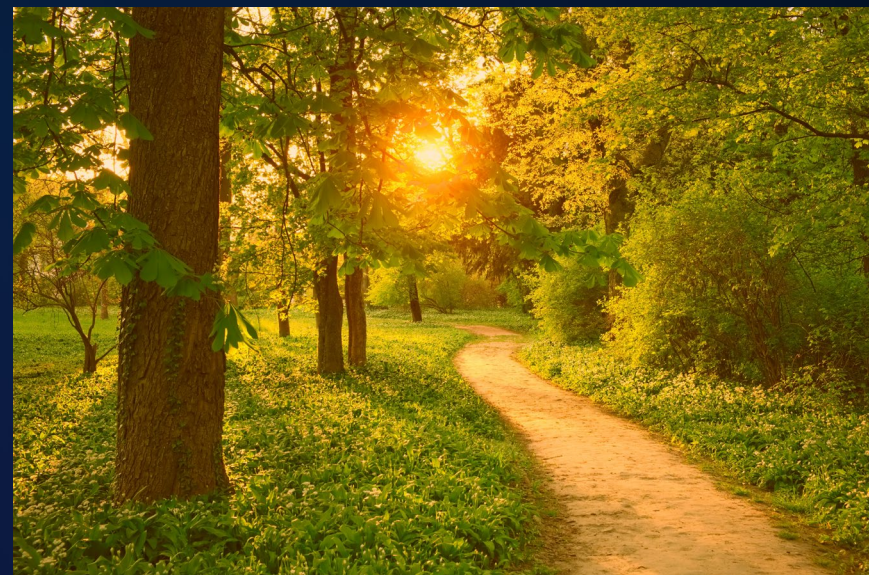
# ゴールデンパスとは

開発チームの負荷(※)を軽減するための、標準化/統合された  
ツール、テンプレートなどのセットでセルフサービスで活用できるもの

- ライブラリ、IaCコード、CI/CD ワークフロー、プロジェクトテンプレート etc...
- 提供方法は、社内リポジトリ、自社構築プラットフォームを用いた配布 etc...

ゴールデンパスを用いると、  
開発チームはソフトウェア開発の作業を  
自分達で全て一から始めなくて良い

セルフサービスであるため、  
開発チームが効率的に活用を開始可能





# ゴールデンパス提供側の辛さの例

- 提供したゴールデンパスの更新を開発チームに伝える際の辛さ
  - 開発チームは開発作業に忙しいので更新を放置されがち
  - 取り込んでもらうためには効率的な仕組みが必要だけど、大変
  - 各開発チームの更新の取り込み状況の把握が困難



- Amazon CodeCatalyst カスタムブループリントを活用してみよう！

開発プロジェクトテンプレートとしてのゴールデンパスの  
構築、更新の伝達、管理をマネージドに実現



Amazon CodeCatalyst



# ゴールドデンパスとしての Amazon CodeCatalyst カスタムブループリントとは？

# Amazon CodeCatalyst カスタムブループリント

組織に合わせて独自に定義された、統合された開発プロジェクトテンプレート

▶ マネージドな環境を用いて、構築、更新の伝達、管理を低負荷に実現



# カスタムブループリントを用いるメリット



プロジェクトテンプレートとしての  
ゴールデンパスを迅速に構築、提供

カスタムブループリントの更新を  
効率的に伝達

各プロジェクトの更新適用状況を  
管理する機能を実現

# カスタムブループリントを用いるメリット



プロジェクトテンプレートとしての  
ゴールデンパスを迅速に構築、提供

カスタムブループリントの更新を  
効率的に伝達

各プロジェクトの更新適用状況を  
管理する機能を実現

- 開発に必要なツール群が統合された、プロジェクトテンプレートとしてのゴールデンパスを構築、提供
  - AWS とシームレスに統合された開発プロジェクトテンプレート
  - カスタムブループリントを構築、公開するためのテンプレートを用いて、迅速に開発を開始可能

# カスタムブループリントを用いるメリット



プロジェクトテンプレートとしての  
ゴールデンパスを迅速に構築、提供

カスタムブループリントの更新を  
効率的に伝達

各プロジェクトの更新適用状況を  
管理する機能を実現

- プルリクエスト形式の伝達により、  
各プロジェクトへの更新を効率化
  - カスタムブループリントは全てコードにより定義されているため、プルリクエスト形式で伝達可能
  - 開発チームは自分で更新差分を確認した上で取り込みの是非を判断可能



projen

# カスタムブループリントを用いるメリット



プロジェクトテンプレートとしての  
ゴールデンパスを迅速に構築、提供

カスタムブループリントの更新を  
効率的に伝達

各プロジェクトの更新適用状況を  
管理する機能を実現

- 各プロジェクトのブループリントの適用状況をまとめて評価できる
  - 各プロジェクトがいつのバージョンのカスタムブループリントを適用しているか一覧で確認できる
  - 組織の規模が拡大しても、組織全体の更新状況の把握が容易



projen

# Amazon CodeCatalyst カスタムブループリントの構築

# カスタムブループリントの構築

開発チーム



プラットフォームチーム

プロジェクトの作成



Amazon CodeCatalyst space

ブループリント構築  
プロジェクト

Source repository

Issue

Workflow

Dev Environment

② プロジェクトの開発環境を用いて、  
カスタムブループリントを開発

① ブループリントビルダーを用いて、  
ブループリント構築用のプロジェクトを作成

ブループリントビルダー

(ブループリント構築のためのテンプレート)



© 2024, Amazon Web Services, Inc. or its affiliates. All rights reserved.



# カスタムブループリントの開発

## プロジェクトの全てをコーディングする

- **blueprint.ts**

- デプロイ先のAWSアカウント、リージョン等の環境情報
- CI/CD ワークフロー、開発環境
- プロジェクト作成時のウィザード、etc...

- **static-assets フォルダ**

- アプリのサンプルコード
- IaC サンプルコード

```
124 substitutionAsset.findAll('cdk-ts-main/**', forEach((asset) => {
125 //console.log(asset.path().slice(asset.path().indexOf('/') + 1))
126 new SourceFile(
127 repository,
128 asset.path().slice(asset.path().indexOf('/') + 1),
129 asset.substitute({
130 tableName: options.infrastructureConfig.tableName,
131 readCapacity: options.infrastructureConfig.readCapacity,
132 yourName: options.applicationConfig.title,
133 stackName: options.infrastructureConfig.stackName,
134 })),
135 );
136 });
137
138 // create an environment, if I have one
139 let environment: Environment | undefined = undefined;
140 if (options.environment) {
141 environment = new Environment(this, options.environment);
142 }
143
144
145
146
147
148 const workflowBuilder = new WorkflowBuilder(this, {
149 Name: 'cdk-workflow',
150 Triggers: [
151 {
152 Type: TriggerType.PUSH,
153 Branches: ['main'],
154 },
155 ],
156 Actions: {
157 BuildAndTest: {
158 Identifier: 'aws/managed-test@v1',
159 Inputs: {
160 Sources: ['WorkflowSource'],
161 },
162 Outputs: {
163 AutoDiscoverReports: {
164 IncludePaths: [
165 '**/*',
166 ],
167 ExcludePaths: [
```

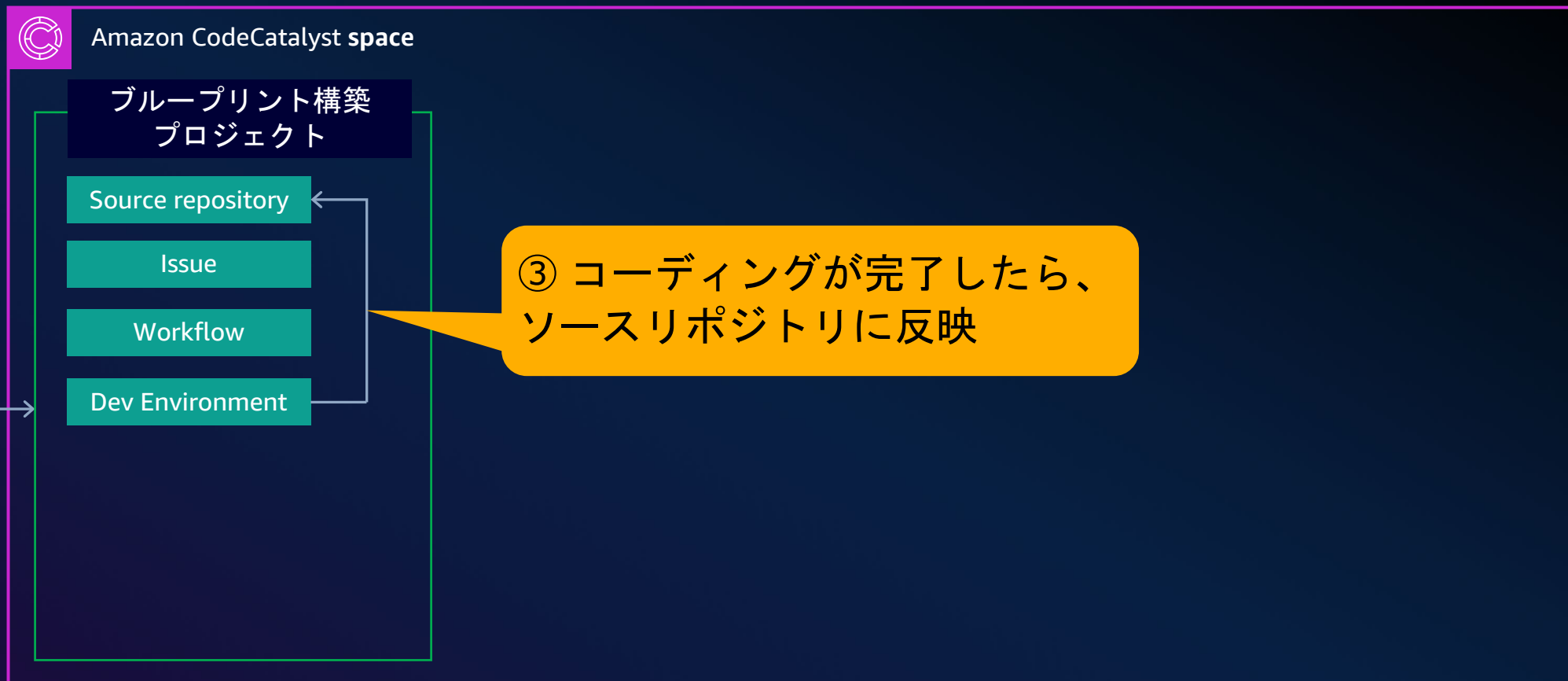
# カスタムブループリントの構築

開発チーム



プラットフォームチーム

プロジェクトの作成



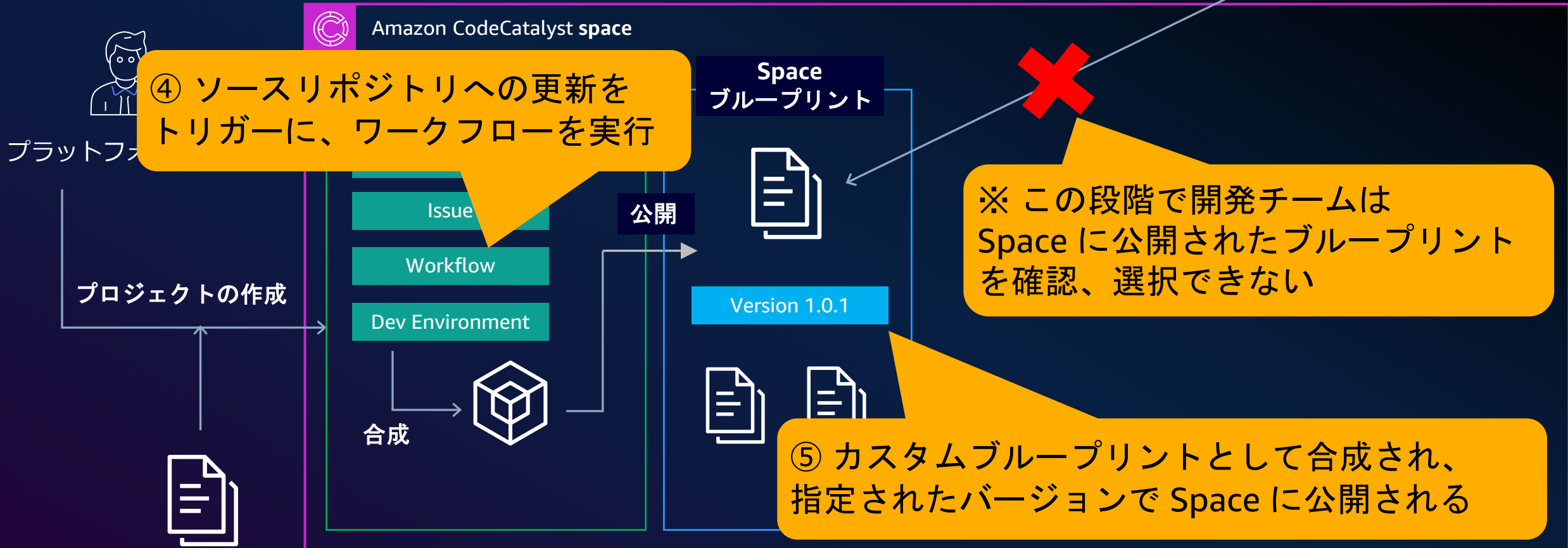
**ブループリントビルダー**

(ブループリント構築のためのテンプレート)



# カスタムブループリントの構築

開発チーム

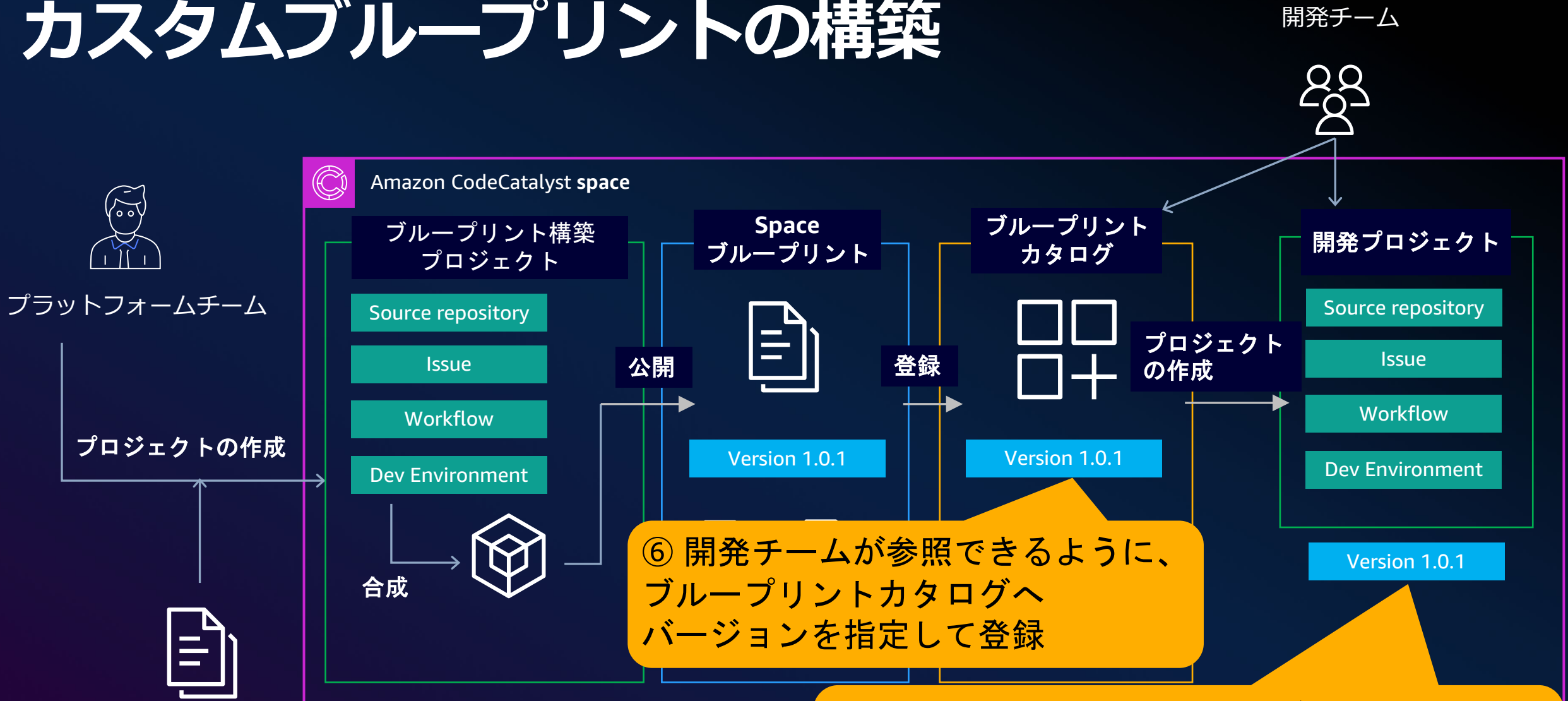


## ブループリントビルダー

(ブループリント構築のためのテンプレート)



# カスタムブループリントの構築



⑥ 開発チームが参照できるように、ブループリントカタログへバージョンを指定して登録

⑦ 開発チームは、セルフサービスでカタログのカスタムブループリントを選択し、プロジェクトを構築できる

## ブループリントビルダー

(ブループリント構築のためのテンプレート)



# Amazon CodeCatalyst カスタムブループリントの更新と管理 (with DEMO)



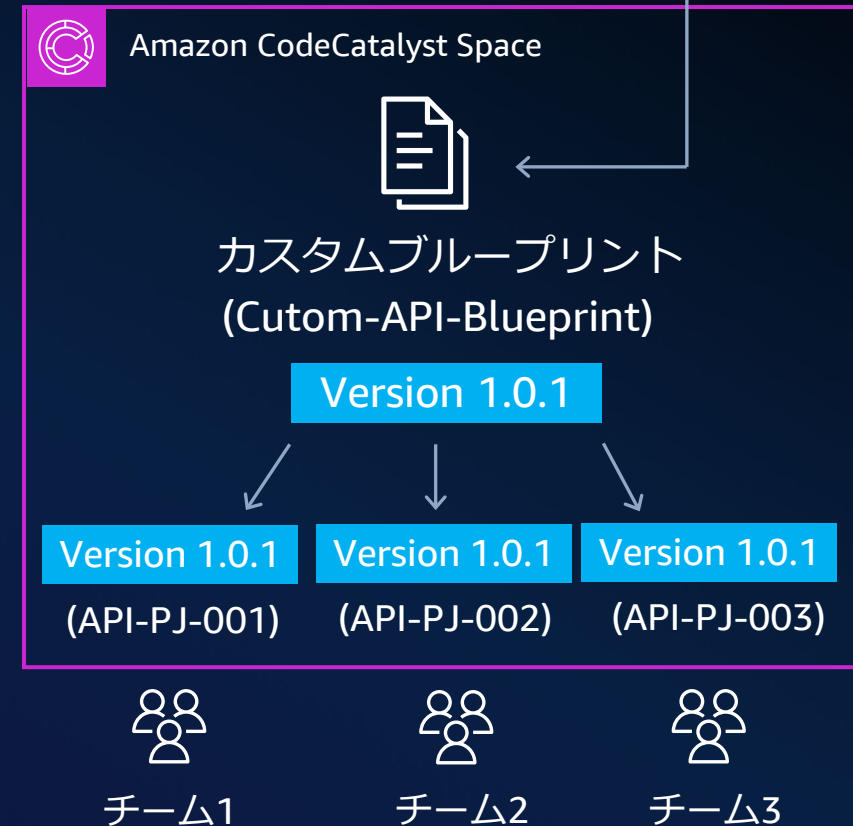
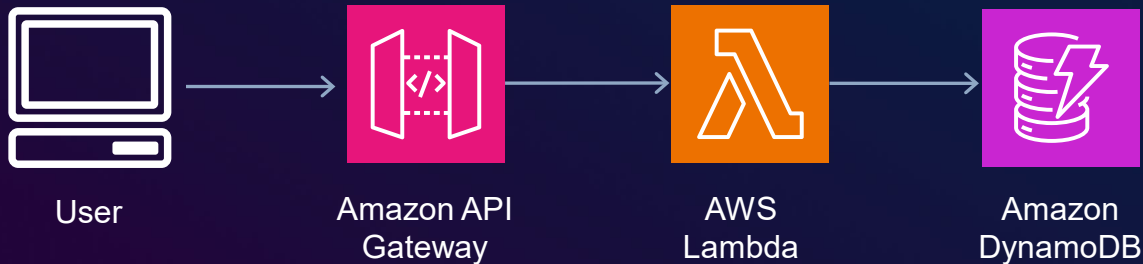
# DEMO シナリオ ～前提部分～

- プラットフォームチームは、WEB APIの開発プロジェクトテンプレートとして、カスタムブループリントを提供しており、開発チームが活用している
- プラットフォームチームは、開発組織全体の**セキュリティ向上のために**、何か取り組めないかと考え始めていた

チームに負担をかけず、より**セキュア**に開発してもらうには・・・



プラットフォームチーム



Version 1.0.1

Version 1.0.1



# カスタムブループリントの更新

～DEMO～

開発チーム



プラットフォームチーム

ブループリントの更新



Amazon CodeCatalyst space

ブループリント構築  
プロジェクト  
(API-Blueprint-Builder)

Space  
ブループリント

ブループリント  
カタログ

Source repository

Issue

Workflow

Dev Environment

Version 1.0.1

Version 1.0.1

Source repository

Issue

Workflow

Dev Environment

Version 1.0.1

プロジェクト  
(API-PJ-001)

② 更新が完了したら、  
ソースリポジトリにコードを反映

① 開発環境を開き、ワークフローに  
関係するコードを編集  
(SAST/SCA 工程追加)





# カスタムブループリントの更新

～DEMO～

開発チーム



開発プロジェクト  
(API-PJ-001)

Source repository

Issue

Workflow

Dev Environment

Version 1.0.1

ブループリント  
カタログ



(Custom-API-Blueprint)

Version 1.0.1

Space  
ブループリント



(Custom-API-Blueprint)

Version 1.0.1

Version 1.0.2



更新版  
公開

Amazon CodeCatalyst space

ブループリント構築

Source repository

Issue

Workflow

Dev Environment

合成



③ ソースリポジトリへの更新を  
トリガーに、ワークフローを実行

④ 新しいバージョンの  
カスタムブループリントが、  
Spaceブループリントに公開される

ブループリント  
の更新

プラットフォーム



# カスタムブループリントの更新

～DEMO～

開発チーム



プラットフォームチーム

ブループリントの更新



Amazon CodeCatalyst space

ブループリント構築プロジェクト  
(API-Blueprint-Builder)

Source repository

Issue

Workflow

Dev Environment

合成



Space  
ブループリント



(Custom-API-Blueprint)

Version 1.0.1

Version 1.0.2



更新版  
公開

登録

ブループリント  
カタログ



(Custom-API-Blueprint)

Version 1.0.2

開発プロジェクト  
(API-PJ-001)

Source repository

Issue

Workflow

Dev Environment

Version 1.0.1

⑤ 新しいバージョンの  
カスタムブループリントを  
カタログに登録



# カスタムブループリントの更新

~DEMO~

ブループリントが更新されたことを通知 (メール、Slack等)

開発チーム

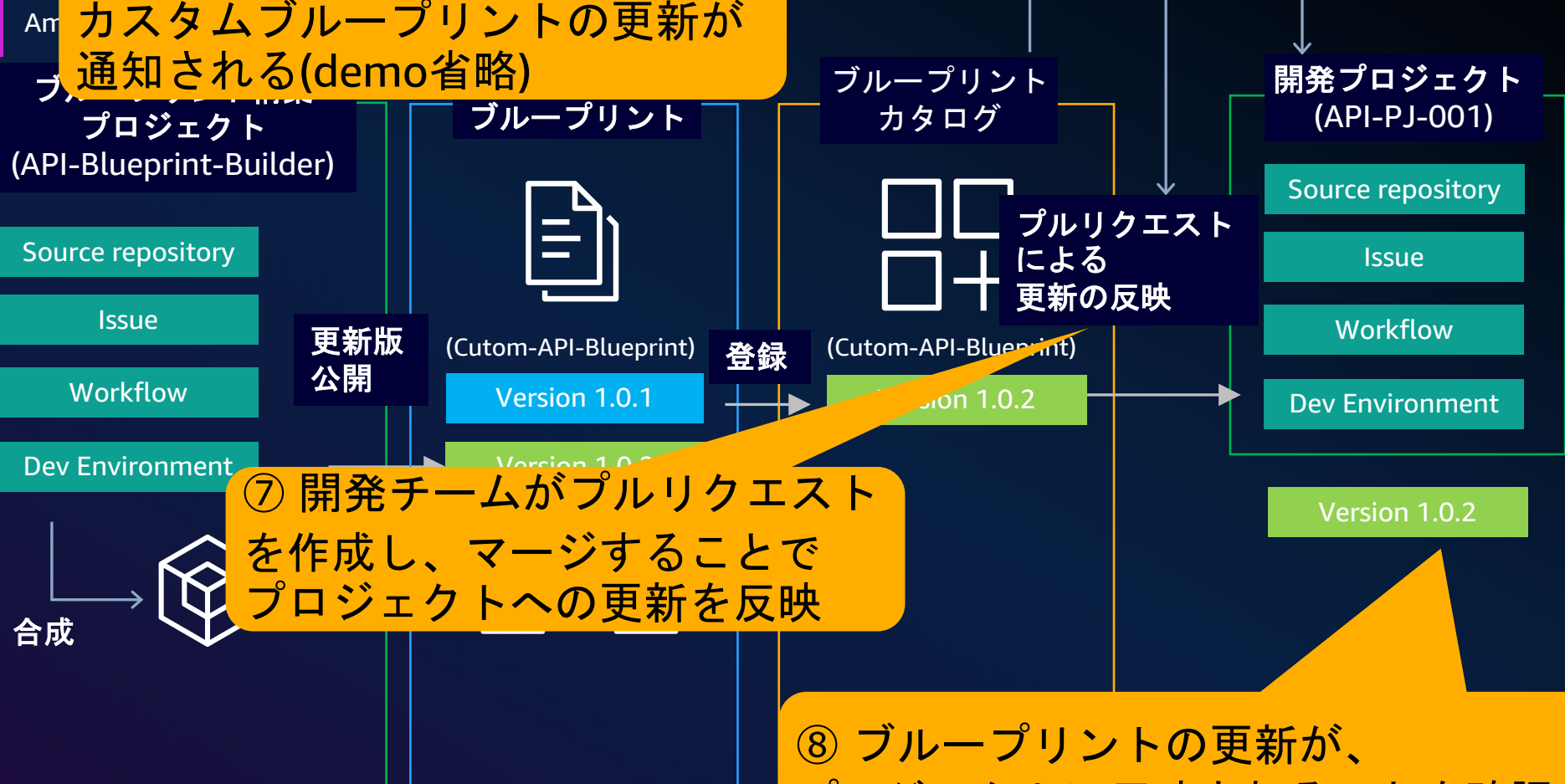


⑥ 開発チームに、  
カスタムブループリントの更新が  
通知される(demo省略)



プラットフォームチーム

ブループリント  
の更新



⑦ 開発チームがプルリクエスト  
を作成し、マージすることで  
プロジェクトへの更新を反映

⑧ ブループリントの更新が、  
プロジェクトに反映されることを確認

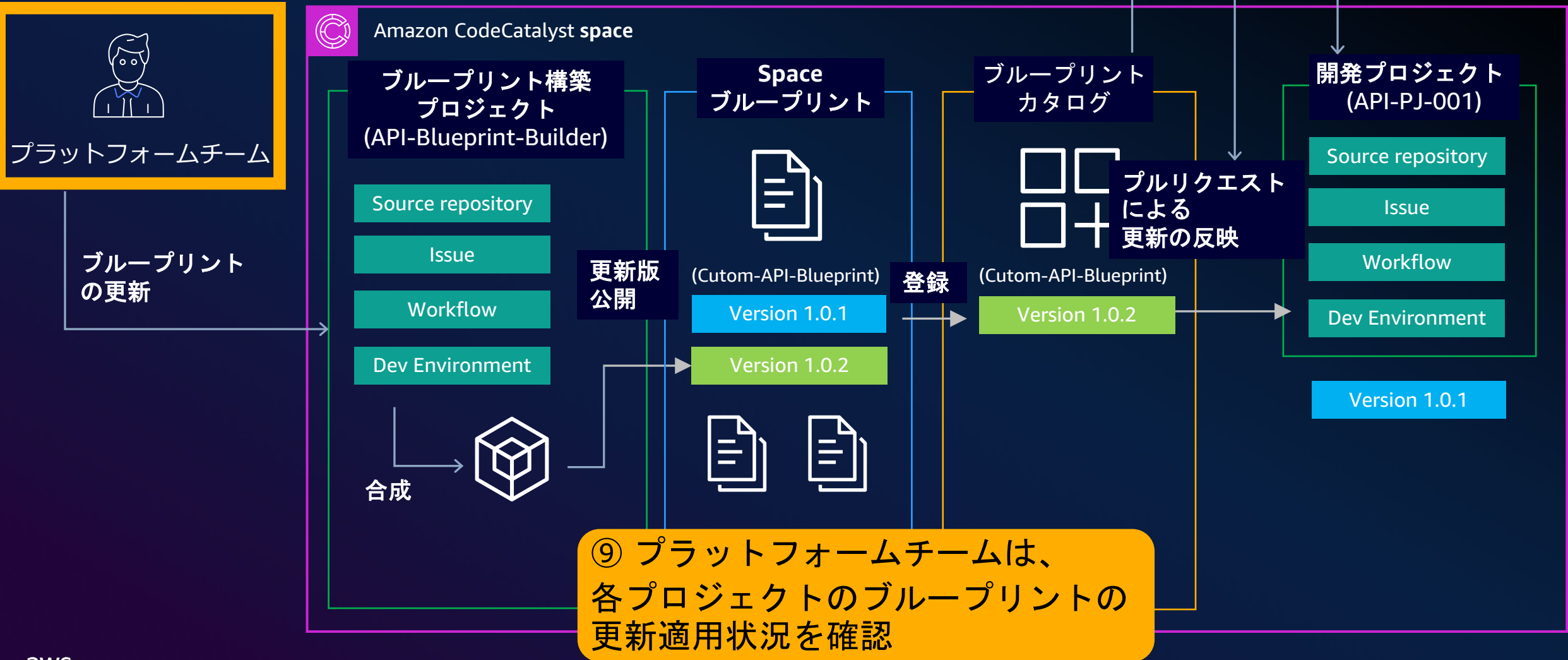


# カスタムブループリントの更新

～DEMO～

ブループリントが更新  
されたことを通知  
(メール、Slack等)

開発チーム



⑨ プラットフォームチームは、各プロジェクトのブループリントの更新適用状況を確認







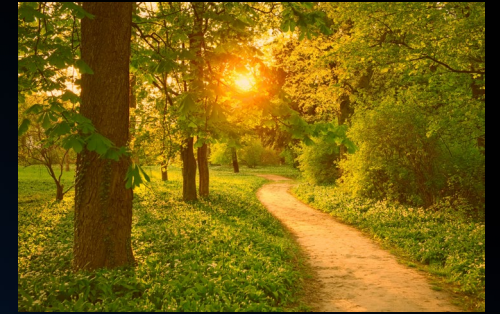
# まとめ



# まとめ

- Amazon CodeCatalyst カスタムブループリントをゴールデンパスとして用いる際のメリットについてお伝えしました
  - 開発プロジェクトテンプレートとしてのゴールデンパスの迅速な構築と提供
  - 開発チームへの効率的なカスタムブループリントの更新の伝達
  - 各プロジェクトにおける更新の反映状況の管理機能の実現
- Amazon CodeCatalyst カスタムブループリントの構築、更新の伝達、管理 を実現する手順についてお伝えしました

# 今後にむけて



- Amazon CodeCatalyst カスタムブループリント  
ハンズオンで WEB API を作ってみよう！(※)
  - <https://catalog.workshops.aws/custom-blueprint/ja-JP>
- ユーザガイドを読んでより深い知識を得よう！
  - <https://docs.aws.amazon.com/codecatalyst/latest/user-guide/custom-blueprints.html>



※ Amazon CodeCatalyst Enterprise Tier (20\$ per User) が必要

※ Amazon CodeCatalyst 利用可能リージョンはオレゴン、アイルランドのみ

(デプロイ先は全リージョン選択可能)

# Thank you!

Ryuji Hori 堀 竜慈

horiry@amazon.co.jp

