



AMAZON QUICKSIGHT ROADSHOW | TOKYO

Dive Deep Session 2

# フォルダとAPIを活用した シングルアカウントでの複数環境運用

2023年8月4日

宮崎 太郎 ( Taro MIYAZAKI )

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 インダストリソリューション第四部  
シニアソリューションアーキテクト

# 自己紹介

## 宮崎 太郎 (Taro Miyazaki)

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 インダストリソリューション第四部

シニアソリューションアーキテクト



データレイク・分析基盤構築 / クラウド組織立ち上げ / アーキテクティングを中心に  
AWS のご利用を検討されている幅広い業種のお客様をご支援

# Agenda

1. **アセット管理の課題**
2. **環境分離の選択肢**
3. **フォルダ分離で使用する権限管理機能**
4. **ソリューション - フォルダによる環境分離**
5. **まとめ**

# QuickSight における アセット管理の課題

# QuickSight の環境管理にお困りではありませんか？

QuickSight を本番利用していくと、開発のステージに対応した複数環境（開発・検証・本番など）を管理したい場面が出てくるかと思います。

## ダッシュボード開発における環境分離ニーズ

- QuickSight アセットの開発者と利用者が分かれており、開発環境で開発、検証ユーザがテストの後にビジネスユーザー向けに展開するというステップを踏みたい。
- 複数の部門・ユーザ・顧客など利用者属性が異なる環境を維持しており、各利用者属性ごとに QuickSight アセットは分離したい。
- 環境は分離したいが複数アカウントの管理はコストが多重になる、アセット管理に関する運用の手間が大きいため簡易的な管理方法を取りたい。

# QuickSight における 環境分離の選択肢



# QuickSight での環境分離の選択肢

パターン	概要	イメージ		
1.アカウント分離	<ul style="list-style-type: none"> <li>複数の QuickSight アカウントで運用</li> <li>各アカウントを各環境に見立てる</li> </ul>	QuickSight アカウントA 開発	QuickSight アカウントB 検証	QuickSight アカウントC 本番
2.名前空間分離	<ul style="list-style-type: none"> <li>一つのQuickSight アカウントで運用</li> <li>名前空間機能を使用し環境を分割</li> </ul>	<div style="border: 1px solid black; padding: 5px; text-align: center;">QuickSight アカウント</div> <div style="display: flex; justify-content: space-around; margin-top: 10px;"> <div style="border: 1px solid black; padding: 5px; text-align: center;">名前空間A 開発</div> <div style="border: 1px solid black; padding: 5px; text-align: center;">名前空間B 検証</div> <div style="border: 1px solid black; padding: 5px; text-align: center;">名前空間C 本番</div> </div>		
3.フォルダ分離	<ul style="list-style-type: none"> <li>一つのQuickSight アカウントで運用</li> <li>フォルダ機能を使用し環境を分割</li> </ul>	<div style="border: 1px solid black; padding: 5px; text-align: center;">QuickSight アカウント</div> <div style="display: flex; justify-content: space-around; margin-top: 10px;"> <div style="border: 1px solid black; padding: 5px; text-align: center;">フォルダ 開発</div> <div style="border: 1px solid black; padding: 5px; text-align: center;">フォルダ 検証</div> <div style="border: 1px solid black; padding: 5px; text-align: center;">フォルダ 本番</div> </div>		

# QuickSight での環境分離の選択肢

パターン	Pros.	Cons.	ユースケース例
1.アカウント分離	<ul style="list-style-type: none"> <li>マルチアカウントによる完全な環境分離</li> </ul>	<ul style="list-style-type: none"> <li>複数アカウント必要</li> <li>ユーザ・アセット管理が煩雑</li> <li>ユーザ重複によるコスト増</li> </ul>	大規模利用・規制業界・顧客専用環境などアセットの完全分離による高セキュリティが必要なケース
2.名前空間分離	<ul style="list-style-type: none"> <li>比較的高レベルな環境分離</li> <li>シングルアカウントで運用可能</li> </ul>	<ul style="list-style-type: none"> <li>APIベースのみでの管理</li> <li>ユーザ・アセット管理が煩雑</li> </ul>	顧客別環境分離が必要なマルチテナント SaaSなどで API を使用した作業の自動化が必要なケース
3.フォルダ分離	<ul style="list-style-type: none"> <li>シングルアカウントで運用可能</li> <li>シンプルな運用</li> </ul>	<ul style="list-style-type: none"> <li>簡易的な環境分離</li> </ul>	社内利用や比較的小規模でシンプルだが実効性のある環境分離を行いたいケース

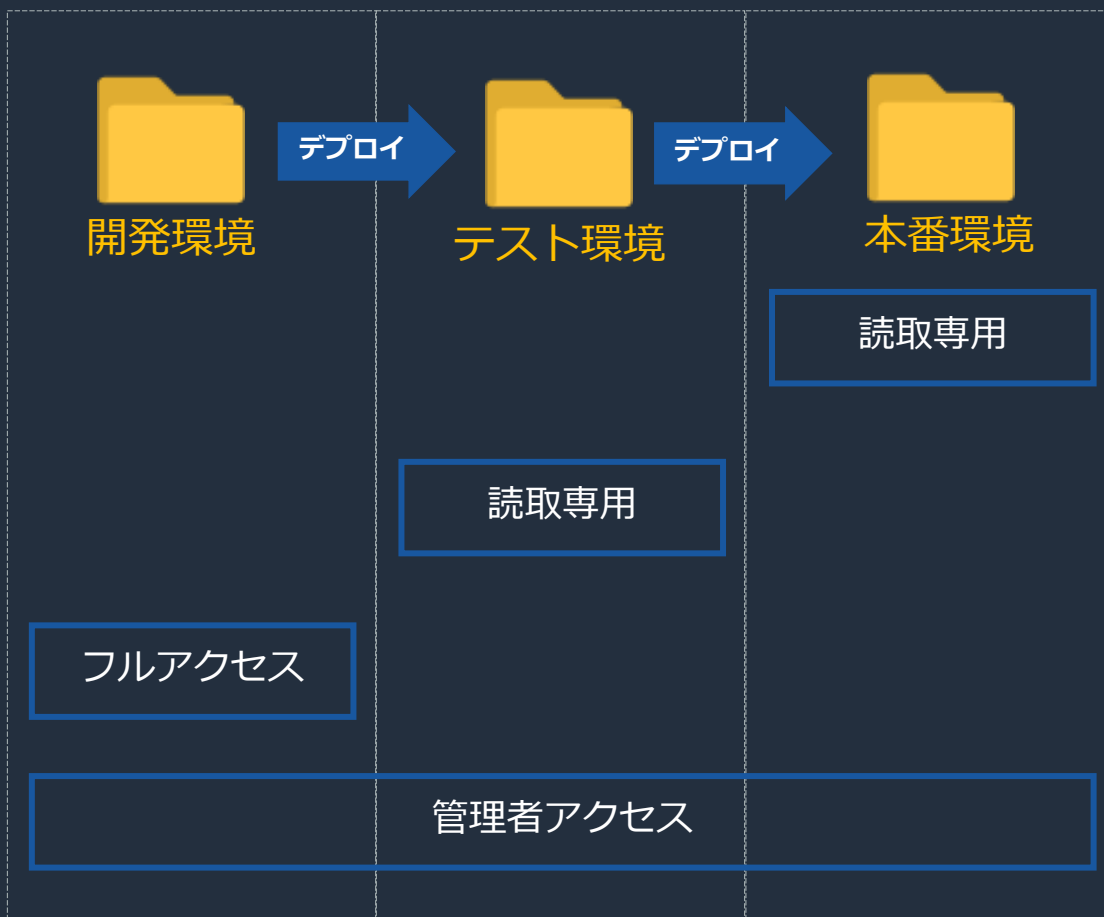


# QuickSight での環境分離の選択肢

パターン	Pros.	Cons.	ユースケース例
1.アカウント分離	<ul style="list-style-type: none"> <li>マルチアカウントによる完全な環境分離</li> </ul>	<ul style="list-style-type: none"> <li>複数アカウント必要</li> <li>ユーザ・アセット管理が煩雑</li> <li>ユーザ重複によるコスト増</li> </ul>	大規模利用・規制業界・顧客専用環境などアセットの完全分離による高セキュリティが必要なケース
2.名前空間分離	<ul style="list-style-type: none"> <li>比較的高レベルな環境分離</li> <li>シングルアカウントで運用可能</li> </ul>	<ul style="list-style-type: none"> <li>APIベースのみでの管理</li> <li>ユーザ・アセット管理が煩雑</li> </ul>	顧客別環境分離が必要なマルチテナント SaaSなどで API を使用した作業の自動化が必要なケース
3.フォルダ分離	<ul style="list-style-type: none"> <li>シングルアカウントで運用可能</li> <li>シンプルな運用</li> </ul>	<ul style="list-style-type: none"> <li>簡易的な環境分離</li> </ul>	社内利用や比較的小規模でシンプルだが実効性のある環境分離を行いたいケース

# ソリューション概要：共有フォルダによる環境管理

## 概要

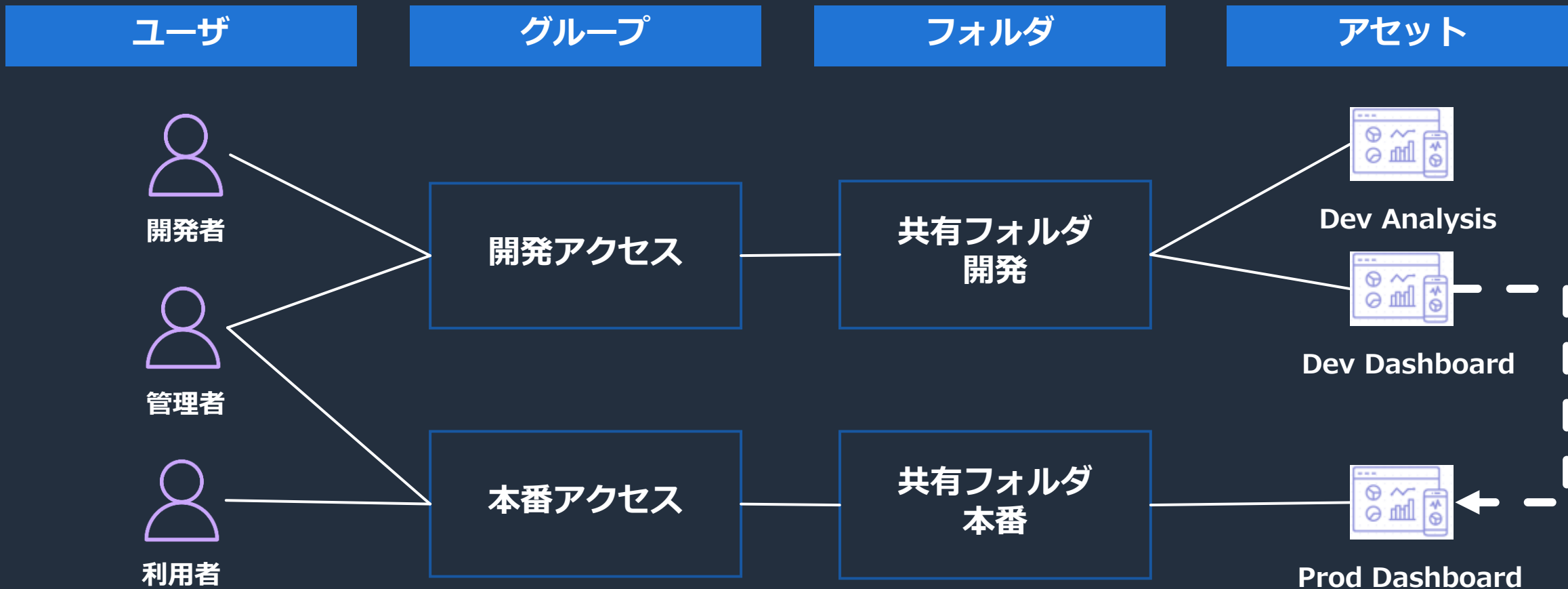


## ポイント

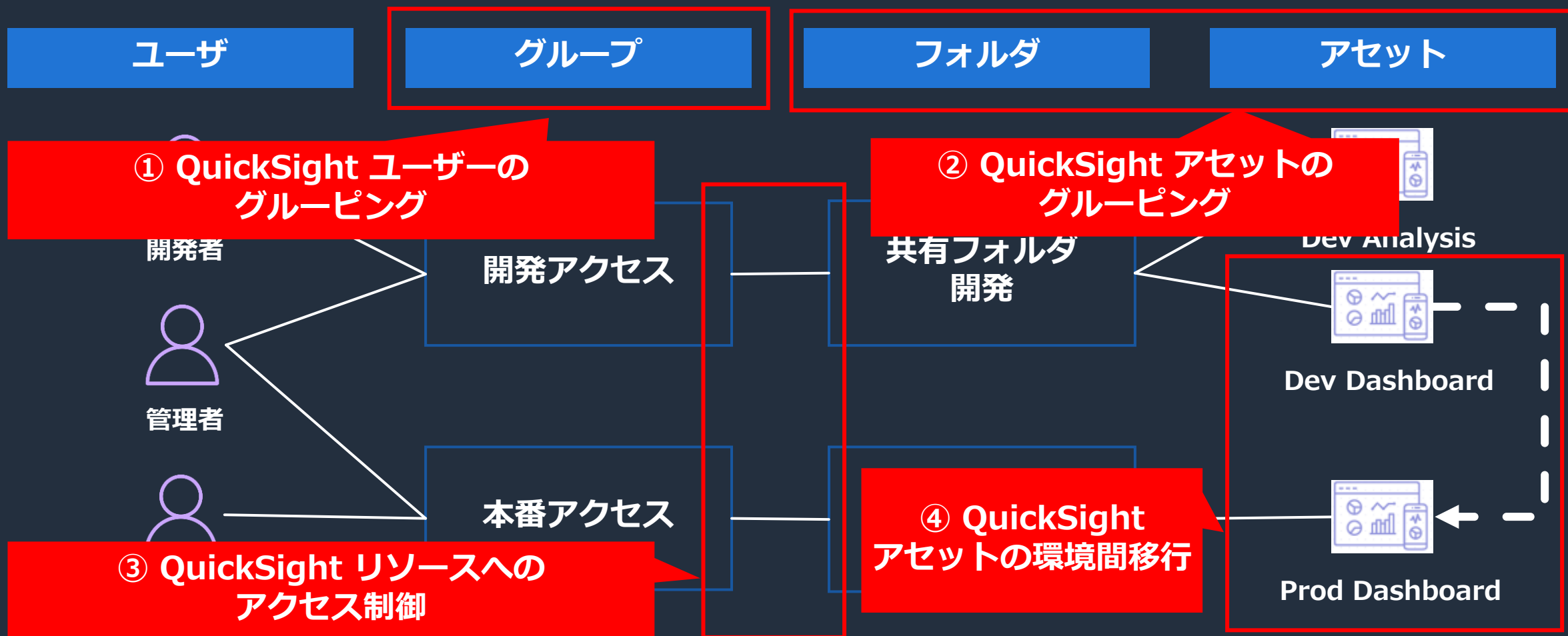
- ✓ 1 アカウントでフォルダを利用して複数環境を実現
- ✓ 各環境へのアクセスはグループにて管理
- ✓ 各環境へのアセットデプロイはAPI/手動で実施
- ✓ 各環境で同じセットのリソース保持（名前やIDで制御）

# QuickSight フォルダ分離で使用する 権限管理機能のおさらい

# QuickSight における権限管理の要素



# QuickSight における権限管理の要素

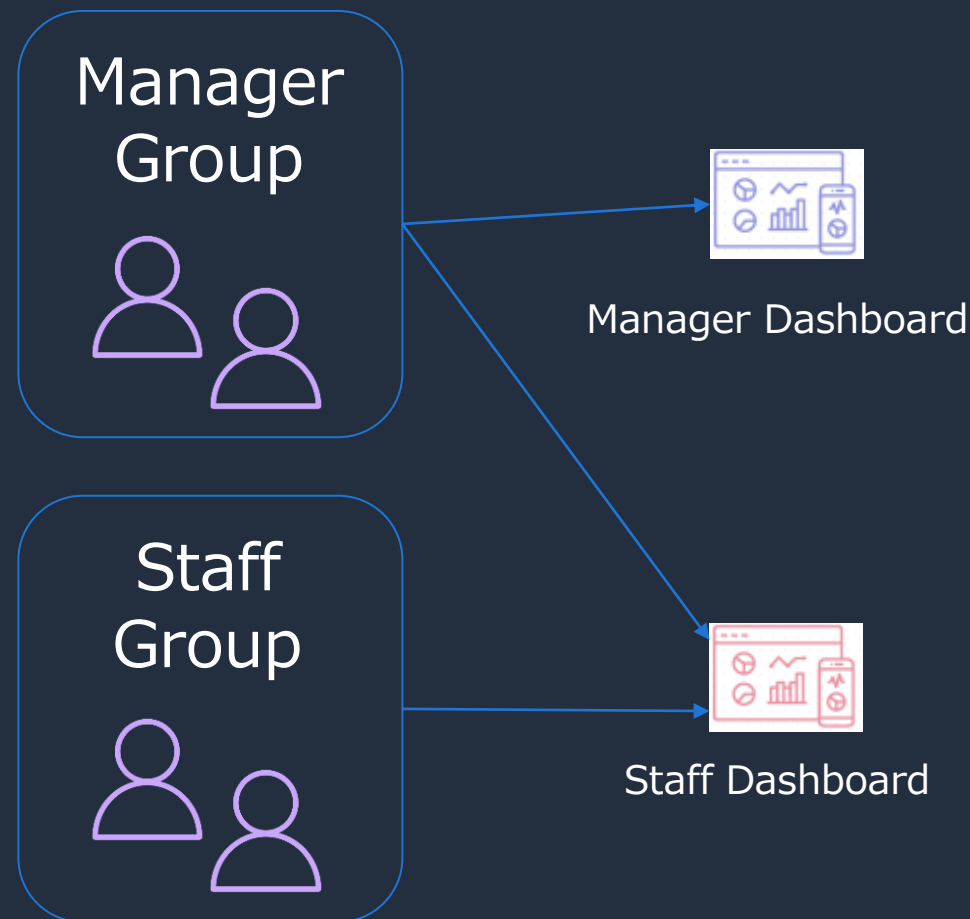


# QuickSight における権限管理の要素

- ① QuickSight ユーザーのグループ핑 → グループ
- ② QuickSight アセットグループ핑 → 共有フォルダ
- ③ QuickSight リソースへのアクセス制御 → リソースパーミッション
- ④ QuickSight アセットの環境間移行 → アセットコピー (API or 手動)

# ① Group (グループ)

- QuickSight内にグループを作成し, ユーザを所属させることが可能
- グループに対してダッシュボードや分析など, 各種リソースへのアクセス権限を付与することが可能. グループに対してアクセスを許可することで, グループに所属するユーザ全員がリソースにアクセス可能になる
- グループは認証の仕組み(ID/パスワード, IAM, Active Directory) とは独立して登録可能
- ただし, Active Directory 連携を行っている場合, Active Directory のグループと QuickSight のグループは自動で同期される
- 利用方法 : コンソール, AWS CLI もしくは API



# ① Group (グループ) 管理画面からグループ作成

## グループの作成 グループの説明を編集

## ユーザーの追加・削除

ユーザーを管理

グループの管理

お客様のサブスクリプション

SPICE 容量

アカウント設定

セキュリティとアクセス権限

VPC 接続の管理

モバイル設定

ドメインと埋め込み

アカウントのカスタマイズ

SP が SSO を開始

QuickSight の管理 / グループの管理

### グループの管理

新しいグループ

グループの検索



グループ	説明
grp003	経理部門グループ
grp002	人事部門グループ

QuickSight の管理 / グループの管理 / test

test

説明を追加

ユーザーの追加

フルユーザーネームで検索



ユーザー名	Eメール	アクション
itohdaik@amazon.co.jp	itohdaik@amazon.co.jp	削除

参考リンク : [https://docs.aws.amazon.com/ja\\_jp/quicksight/latest/user/creating-quicksight-groups.html](https://docs.aws.amazon.com/ja_jp/quicksight/latest/user/creating-quicksight-groups.html)

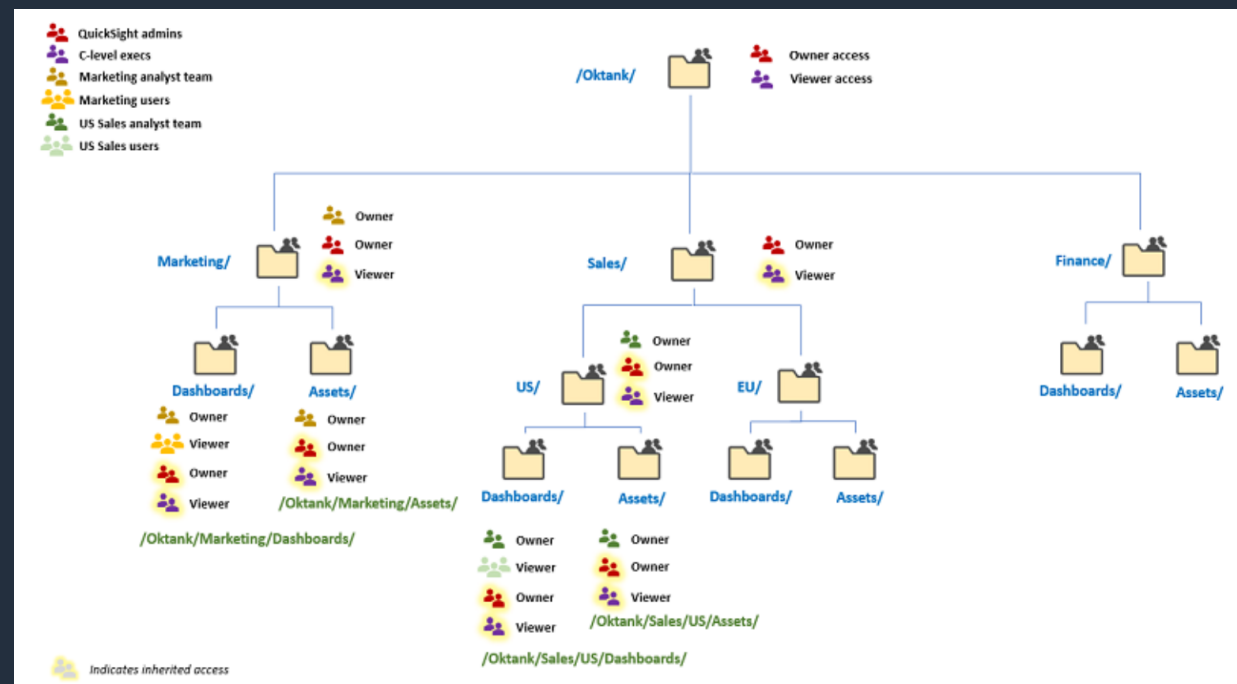
注 : グループの削除はCLI・APIで実行





## ② 共有フォルダ

- **共有フォルダ**：フォルダ単位でアセット（ダッシュボードやデータセット等）の共有/権限管理をする
- 共有フォルダでは Owner あるいは Viewer のアクセス権限を付与して管理
- 親フォルダの権限は子フォルダに引き継がれる
- **Owner**：Admin, Author に対してフォルダへのアクセス権限を付与し, アセット(サブフォルダ含む)の追加, 削除, フォルダ自体を他のユーザー, グループへ再共有する権限を持たせる
- **Viewer**：Admin, Author, に加えて, Reader に対して付与できる権限. フォルダ内のアセット閲覧に限定される

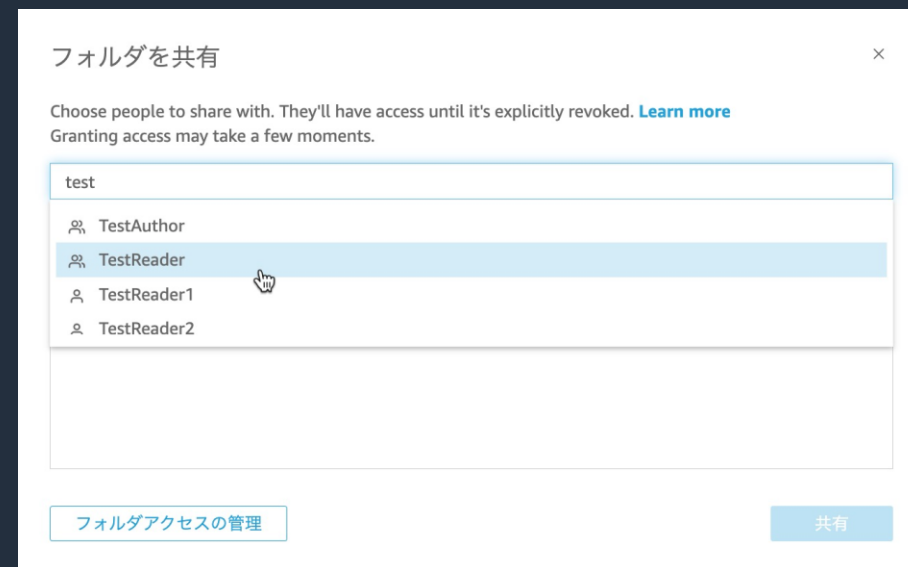


Organize and share your content with folders in Amazon QuickSight

<https://aws.amazon.com/jp/blogs/big-data/organize-and-share-your-content-with-folders-in-amazon-quicksight/>

## ② 共有フォルダ 利用方法

- 最上位フォルダの作成はAdminのみ。
- 作成したフォルダはグループ、ユーザーに対して共有することで他グループ、ユーザーからの参照、管理が可能となる
- 作成したフォルダに対してデータセット、分析、ダッシュボードを登録可能
- フォルダ作成や名前変更権限はカスタムパーミッションで制御可

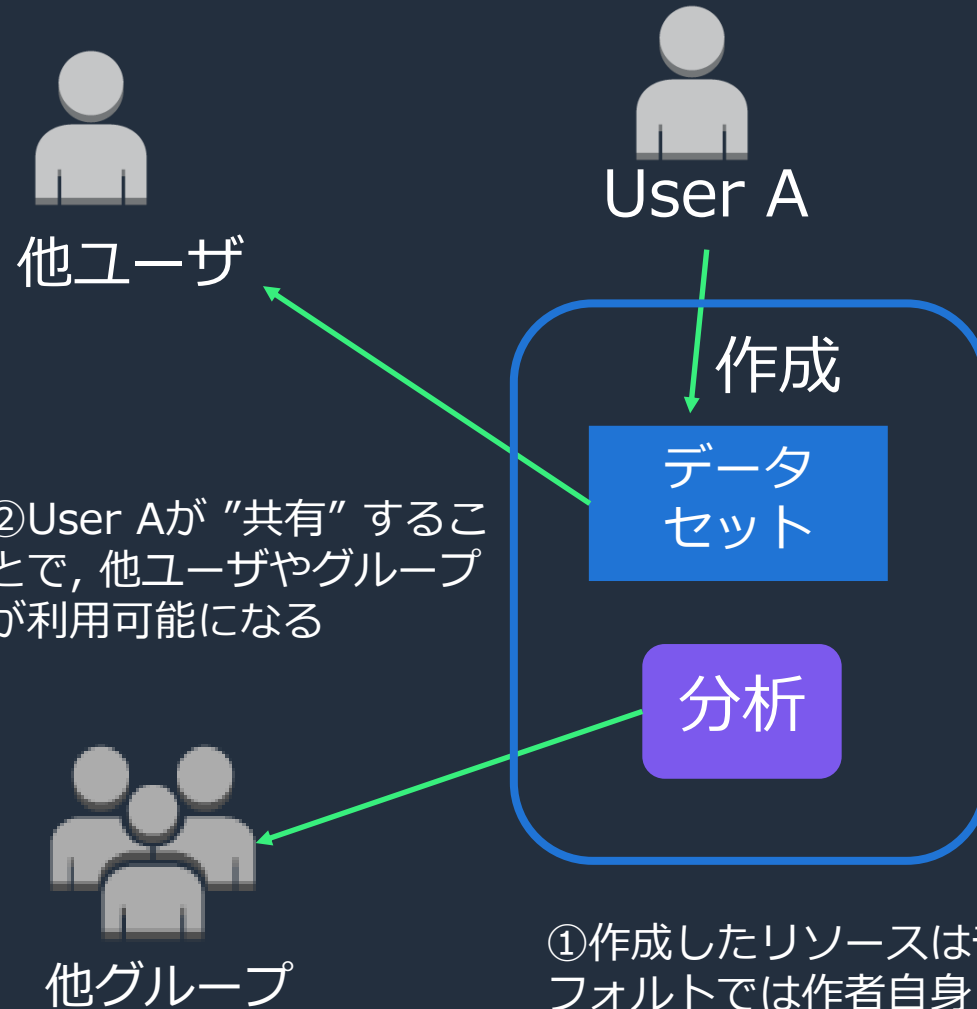


### ③ リソースパーミッション

QuickSight では、アセット側で個別にアクセス制御を行っている。アセットは、作成直後は作成者からのみ閲覧、編集が可能

- データソース
- データセット
- 分析
- ダッシュボード
- テンプレート

作成者によってグループ、ユーザーに対してアクセス権限が付与されるか、または共有フォルダに登録されることで他のユーザー、グループから閲覧が可能となる



② User Aが "共有" することで、他ユーザやグループが利用可能になる

① 作成したリソースはデフォルトでは作者自身しかアクセスできない

## ④ API/CLI による操作

CLI/API により QuickSight の様々なリソース操作が可能

- ユーザ・グループ操作
- データソース、データセット操作
- SPICE操作
- 埋め込み
- 分析、ダッシュボード操作
- テンプレート
- JSON形式のビジュアル情報出力とその利用  
※分析、テンプレートとダッシュボードに対応

amazon web services [aws]

### quicksight

#### Description

Amazon QuickSight is a fully managed, serverless business intelligence service for the AWS Cloud that makes it easy to extend data and insights to every user in your organization. This API reference contains documentation for a programming interface that you can use to manage Amazon QuickSight.

#### Available Commands

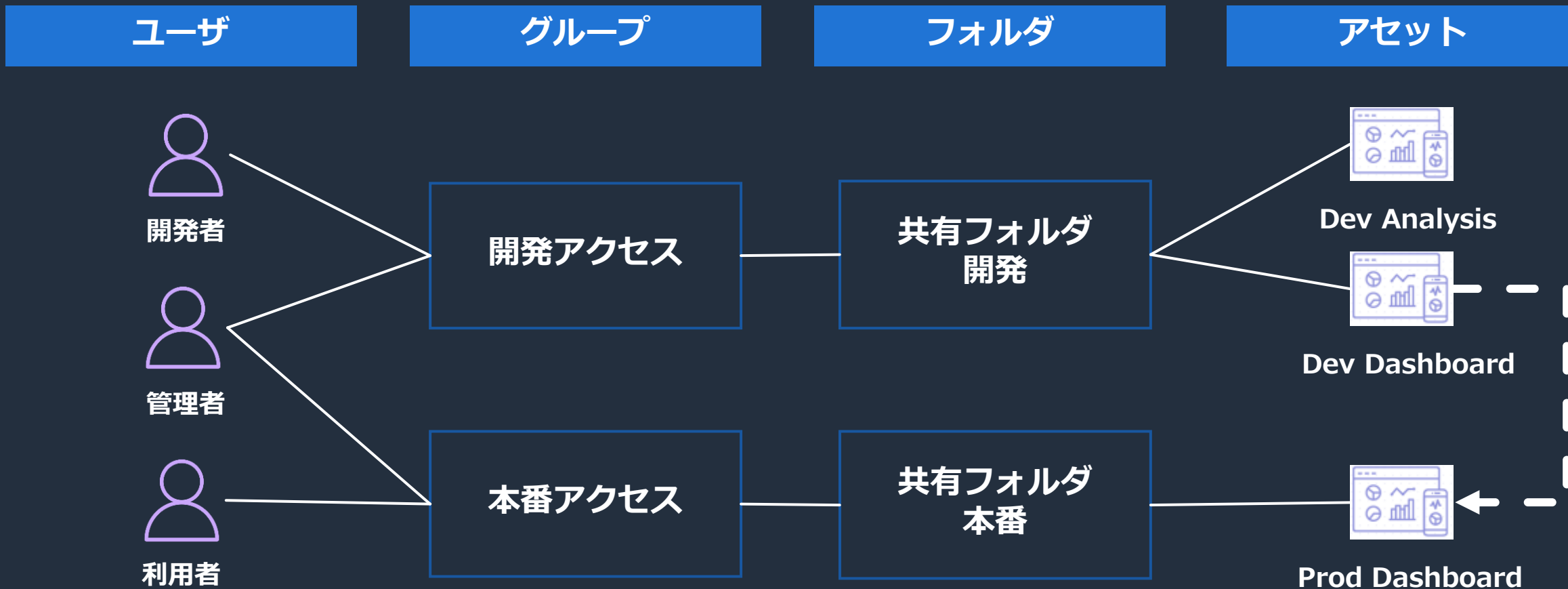
- cancel-ingestion
- create-dashboard
- create-data-set
- create-data-source
- create-group
- create-group-membership
- create-iam-policy-assignment
- create-ingestion
- create-template
- create-template-alias
- delete-dashboard
- delete-data-set
- delete-data-source
- delete-group
- delete-group-membership
- delete-iam-policy-assignment
- delete-template
- delete-template-alias
- delete-user
- delete-user-by-principal-id
- describe-dashboard
- describe-dashboard-permissions
- describe-data-set
- describe-data-set-permissions
- describe-data-source
- describe-data-source-permissions
- describe-group
- describe-iam-policy-assignment
- describe-ingestion
- describe-template
- describe-template-alias
- describe-template-permissions
- describe-user



# (再掲) QuickSight における権限管理の要素

- ① QuickSight ユーザーのグルーピング → グループ
- ② QuickSight アセットグルーピング → 共有フォルダ
- ③ QuickSight リソースへのアクセス制御 → リソースパーミッション
- ④ QuickSight アセットの環境間移行 → アセットコピー (API or 手動)

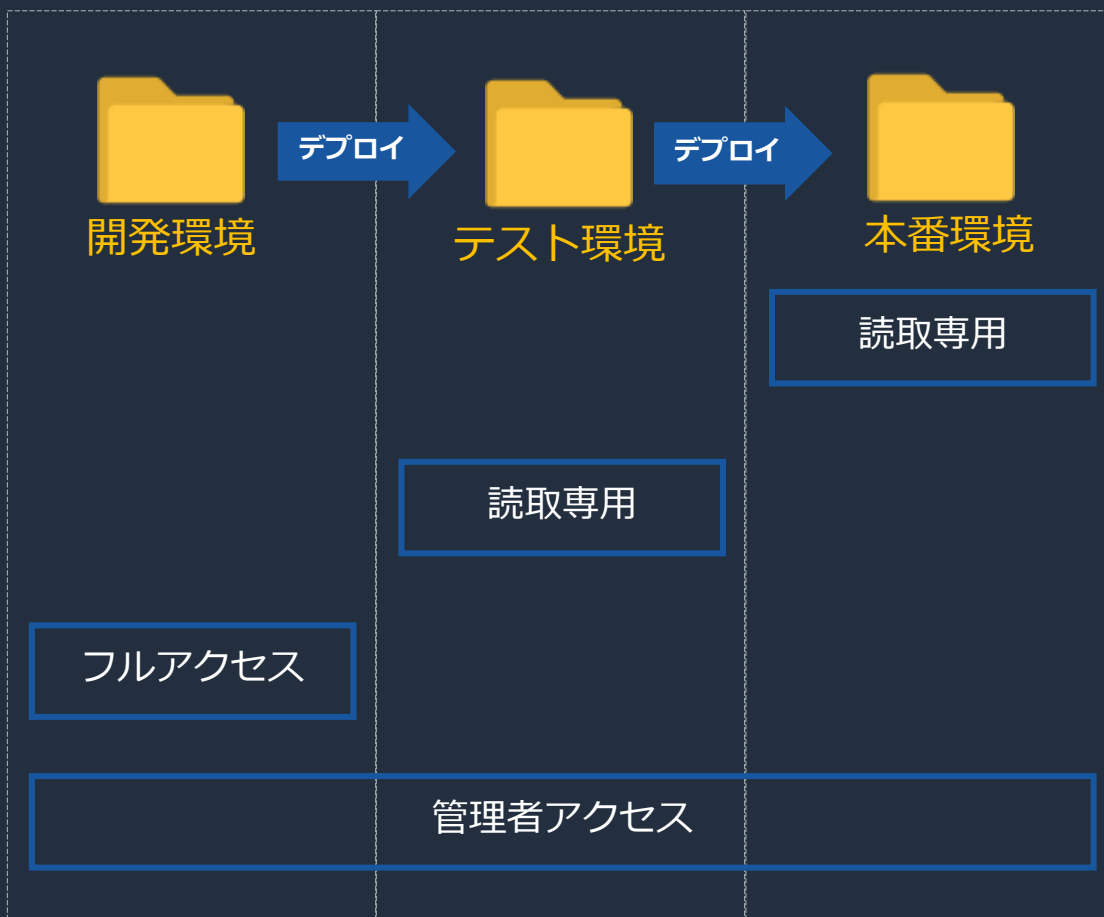
# (再掲) QuickSight における権限管理の要素



# ソリューション： QuickSight 共有フォルダによる環境管理

# ソリューション概要：共有フォルダによる環境管理

## 概要



## ポイント

- ✓ 1 アカウントでフォルダを利用することで複数環境を実現
- ✓ 各環境へのアクセスはグループにて管理
- ✓ 各環境へのアセットデプロイはAPI/手動で実施
- ✓ 各環境で同じセットのリソース保持 (名前やIDで制御)

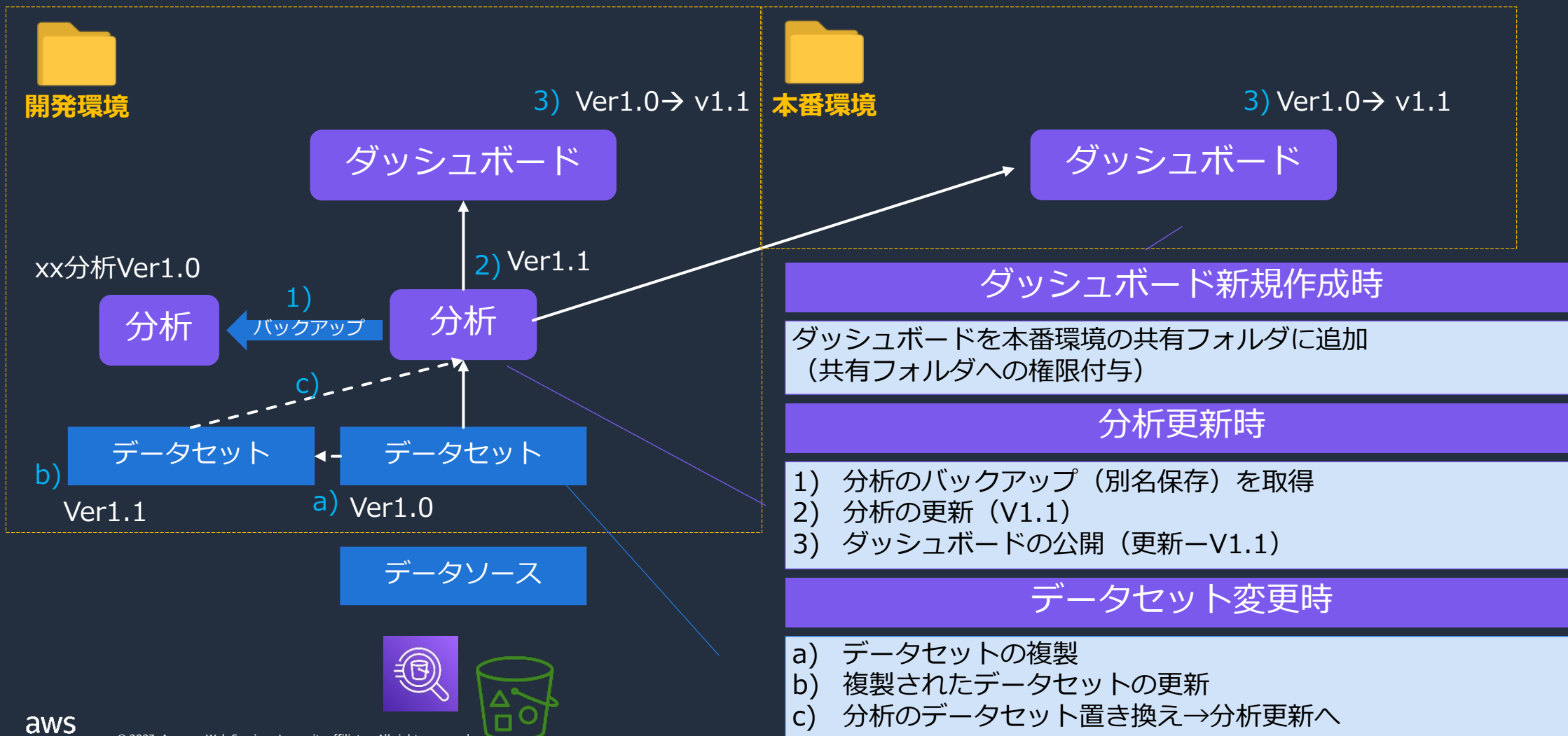


# 共有フォルダ環境分離：実装パターン

パターン	概要	分離レベル		操作
		ダッシュボード	分析・データセット・データソース	
①-1. リソース共有(GUI)	データは環境間で共有し、ダッシュボードの権限を各環境(フォルダ)に付与していくパターン	個別	共有	GUI/API
①-2. リソース共有(API)	(1-1.を一部 API で実施するパターン)	個別	共有	API
②-1. リソース分離(GUI)	データソースを含むすべてのアセットを環境別に保持し、GUI でデータを環境間コピーするパターン	個別	個別	GUI/API
②-2. リソース分離(API)	(2-1.を一部 API で実施するパターン)	個別	個別	API

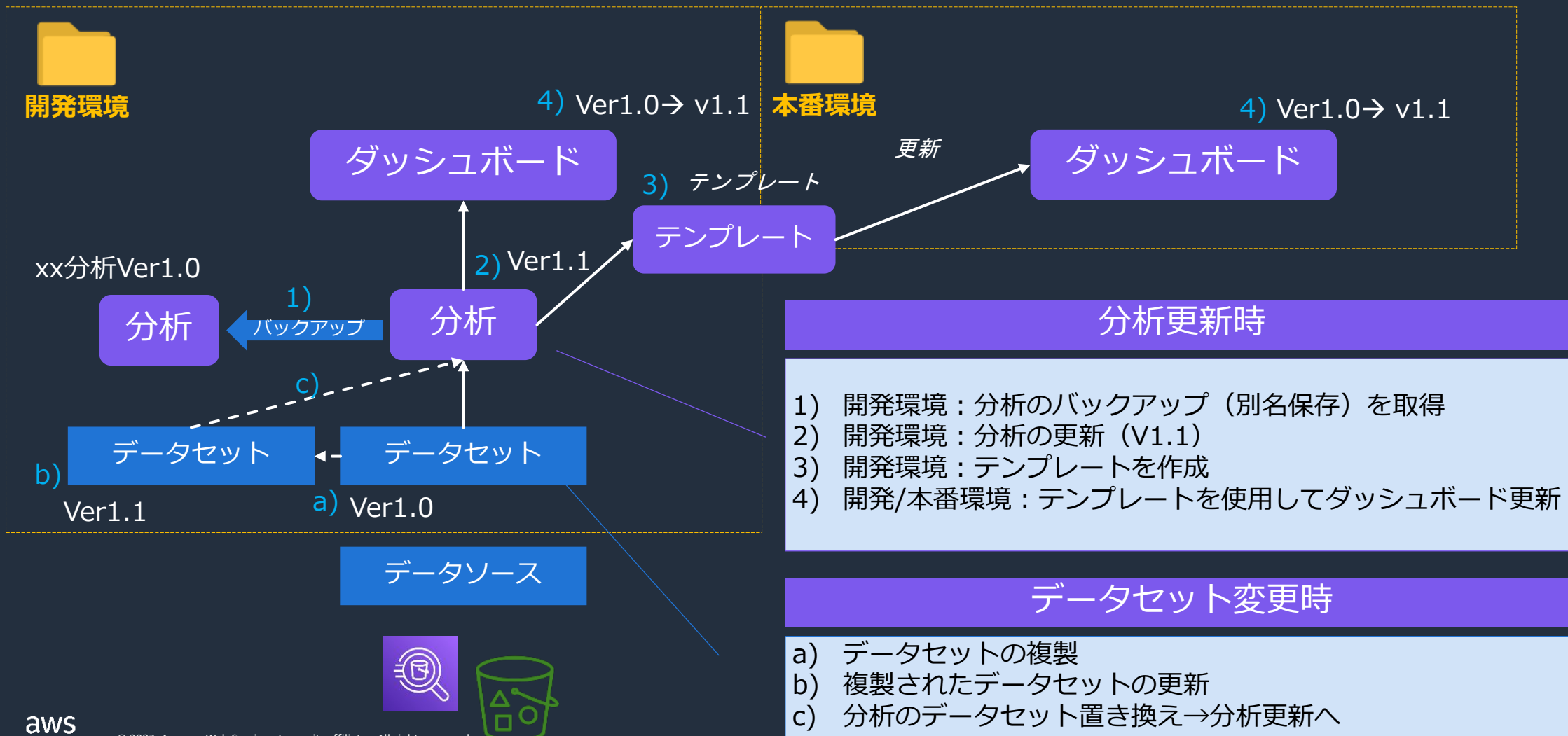
# ①-1 リソース共有パターン (GUI)

データは環境間で共有し、ダッシュボードの権限を各環境(フォルダ)に付与していく  
 ※分析バージョンを戻す際にダッシュボードURL変更



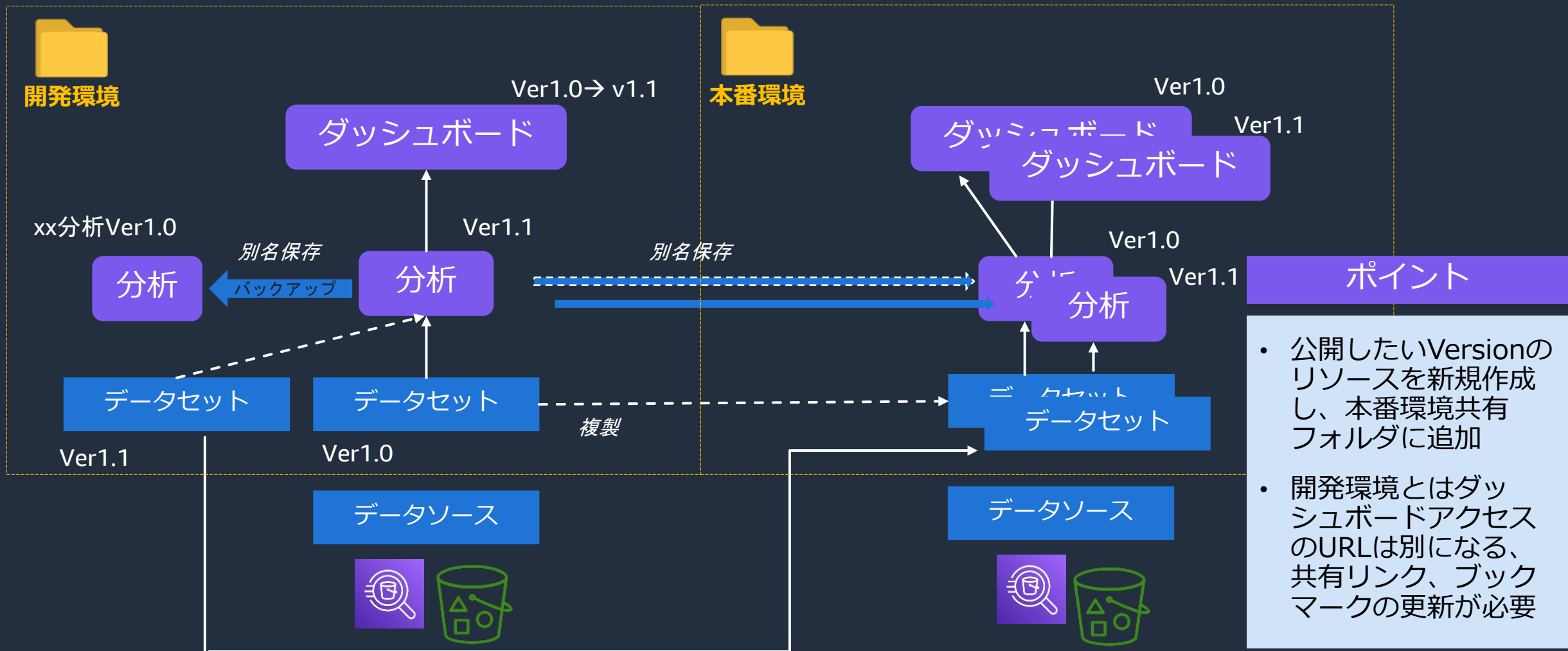
# ①-2 リソース共有パターン (API)

データは環境間で共有し、ダッシュボードの権限を各環境(フォルダ)に付与していく  
 ※ API を使用し、ダッシュボードアクセスURLを保持



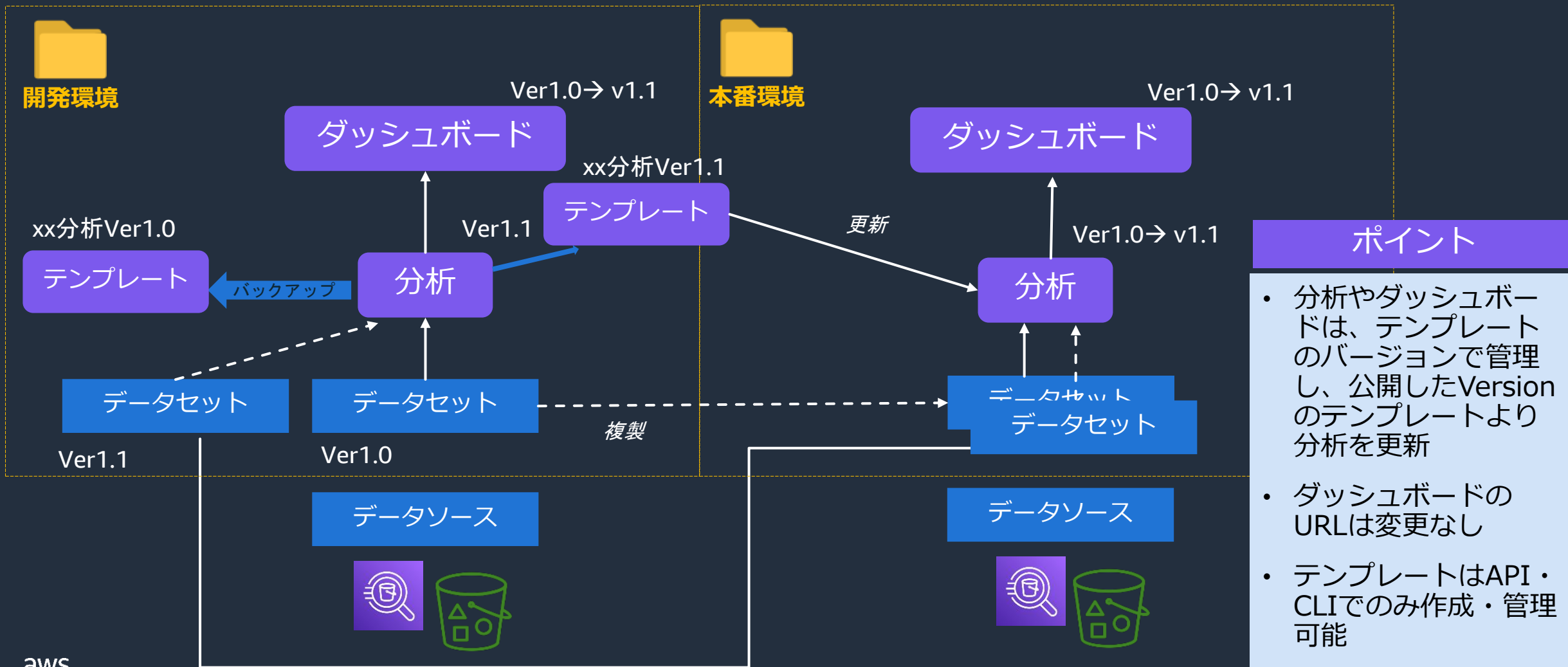
# ②-1 リソース分離パターン (GUI)

データセット・分析を含む全ての  
リソースを環境間で別管理していく  
※開発環境のダッシュボード更新都度アクセスURL変更



## ②-2 リソース分離パターン (API)

データセット・分析を含む全ての  
リソースを環境間で別管理していく  
※分析をテンプレートで作成することでURI変更なし



# (再掲) 共有フォルダ環境分離：実装パターン

パターン	概要	分離レベル		操作
		ダッシュボード	分析・データセット・データソース	
①-1. リソース共有(GUI)	データは環境間で共有し、ダッシュボードの権限を各環境(フォルダ)に付与していくパターン	個別	共有	GUI/API
①-2. リソース共有(API)	(1-1.を一部 API で実施するパターン)	個別	共有	API
②-1. リソース分離(GUI)	データソースを含むすべてのアセットを環境別に保持し、GUI でデータを環境間コピーするパターン	個別	個別	GUI/API
②-2. リソース分離(API)	(2-1.を一部 API で実施するパターン)	個別	個別	API

# (再掲) 共有フォルダ環境分離：実装パターン

パターン	概要	分離レベル		操作
		ダッシュボード	分析・データセット・データソース	
<b>環境間でリソースを共有してよい場合</b>				
①-1. リソース共有(GUI)	データは環境間で共有し、ダッシュボードの権限を各環境(フォルダ)に付与して共有パターン	個別	共有	GUI/API
①-2. リソース共有(API)	(1-1.を一部 API で実施するパターン)	個別	共有	API
<b>環境間でリソース分離が必要な場合</b>				
②-1. リソース分離(GUI)	データソースを含むすべてのアセットを環境別に保持し、GUI でデータを環境間コピーするパターン	個別	個別	GUI/API
②-2. リソース分離(API)	(2-1.を一部 API で実施するパターン)	個別	個別	API

# まとめ





# Summary

- ✓ QuickSight における環境分離には複数の方法がある  
→アカウントレベル、名前空間レベル、フォルダレベル
- ✓ シングルアカウントで手軽に環境分離を行う場合、フォルダ分離を検討
- ✓ 検討時、環境間でどこまでデータを共有して良いかを考慮しパターン検討
- ✓ GUI に加え CLI/API を使用することで運用の自動化・効率化が可能

# Call to Action

- ✓ QuickSight のセルフサービスハンズオンをご覧ください
- ✓ [Amazon QuickSight - Visualization Basics \(Japanese\)](#)



**Thank you!**