AWSで実現する
研究者のための創薬研究基盤

CBI学会2021年大会 SS-13

Amazon Web Services Japan

Solutions Archtect

Yuhei Harada

自己紹介

原田 裕平 Yuhei Harada

アマゾン ウェブサービス ジャパン 株式会社
技術統括本部
ヘルスケア・ライフサイエンス部

Agenda

- 創薬研究におけるAWSの活用
    - サービスピックアップ紹介
- HPC on AWSの特徴・サービス
- サービスを組み合わせた解析ワークフロー自動化の方法

創薬研究における**AWS**の活用

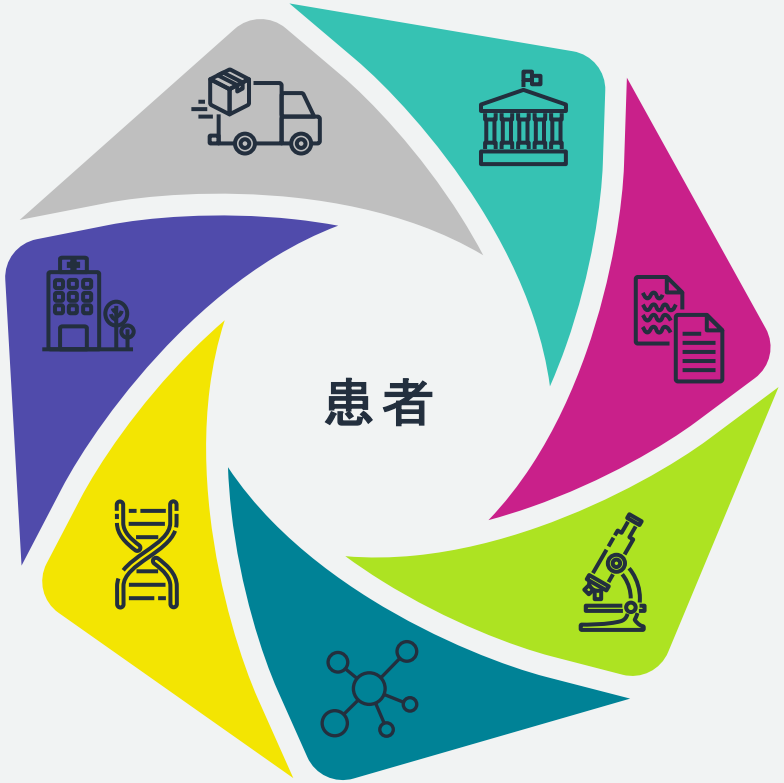
is the cloud computing arm of

AWS は生活者・患者をとりまくステークホルダーのITインフラをご支援

ベンダー
ヘルスケア IT ISV
診断
医療機器
グローバル SI

プロバイダー
病院のシステム
臨床検査医学
学術医療センター
薬局

ゲノミクス
研究
臨床
消費者に直結



政府
公衆衛生および規制当局
科学研究機関
保健省庁
現役および退役軍人の健康

医療費の支払者
健康保険
雇用主

NGOおよびNPO
保健協会
研究機関

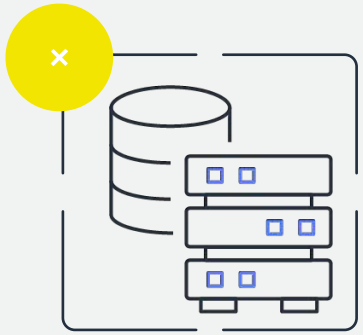
製薬と医療機器
研究・開発
臨床開発
製造およびサプライチェーン
コマーシャルとメディカルアフェアーズ

In-silico創薬研究における課題

データ活用の課題

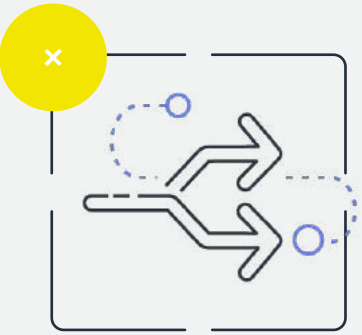
膨大なデータ量
共同研究者との安全なデータ共有

コンピューティングの課題



日々の増減する計算需要への対応
計算リソース保守管理

研究ワークフローの課題



反復的な手動タスクによる効率低下

In-silico創薬研究におけるAWSの提供する価値

データ活用の課題

膨大なデータ量

共同研究者との安全なデータ共有

容量無制限のスケーラブルなストレージ

きめ細やかなアクセス制御による

安全なデータ共有

コンピューティングの課題

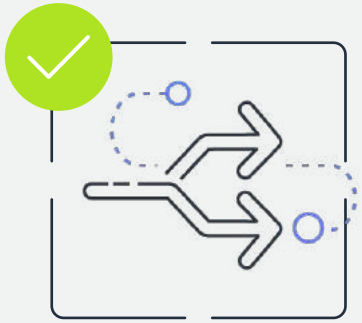
日々の増減する計算需要への対応

計算リソース保守管理

必要な時に必要なだけ計算リソースを確保

物理リソースの保守管理はAWSにお任せ

研究ワークフローの課題



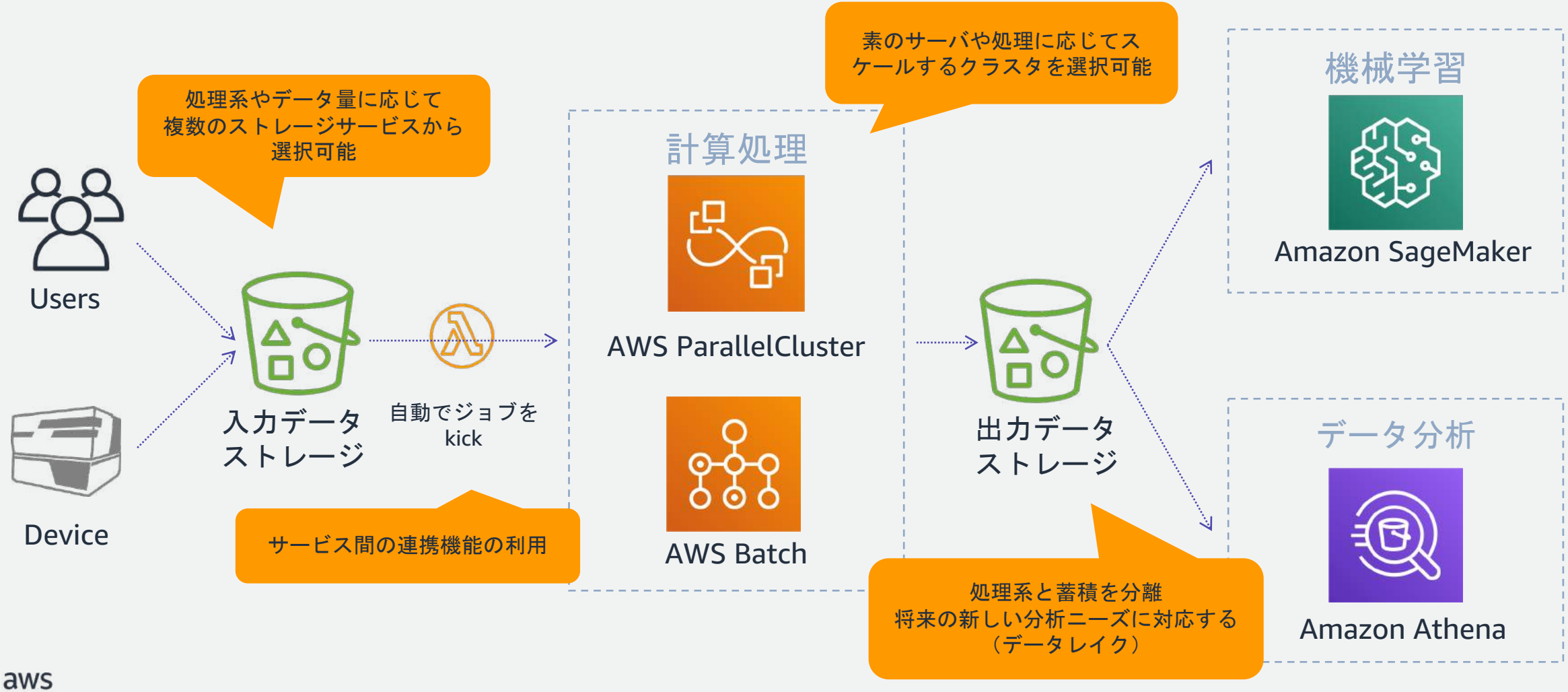
反復的な手動タスクによる効率低下

複数の処理を連携し自動化

データドリブンで

高スループットな解析を実現

AWSのサービスを活用したデータドリブンな解析例



AWSはやりたいことをご自身で実現する Self Service Platform

1つのサービスやツールでは自由度と実装コストの両立に限界

⇨**複数のサービスを適材適所で組み合わせ、やりたいことを最小の手間で実現する**

➡ Building Block

http://farm4.static.flickr.com/3514/3281353786_c1a130ff2e_b.jpg

- 価値創出にフォーカスできる
- 失敗や試行錯誤が容易
- リードタイムの短縮

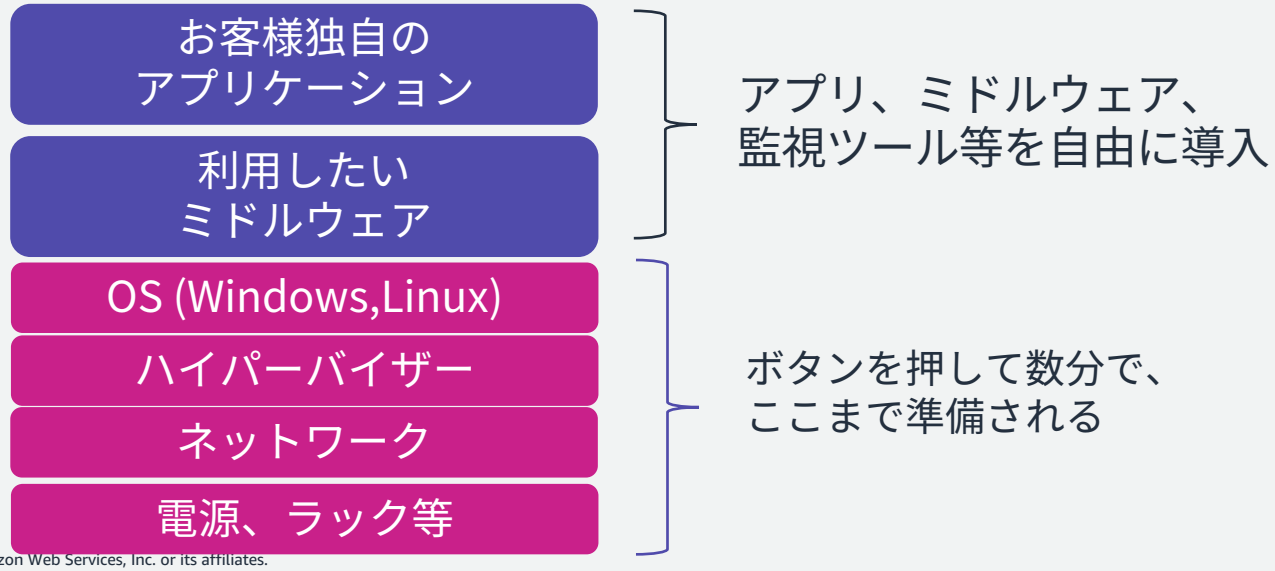
サービスピックアップ紹介

Amazon EC2(Elastic Compute Cloud)

- 数分で起動、1秒ごとの従量課金※で利用可能な仮想サーバ
- 多数のOSをサポート、ライセンス費用込みで従量課金
- 自由にソフトウェアのインストールが可能(ユーザがroot/Administrator権限)
- スケールアップ/ダウン、アウト/インが容易に可能

※ OSの種類により1時間ごととなるものもあります

・仮想CPU数追加、メモリ追加等、スペック変更が容易
過剰なリソースの用意ではなく、その時に最適なリソースでコスト最適化が可能

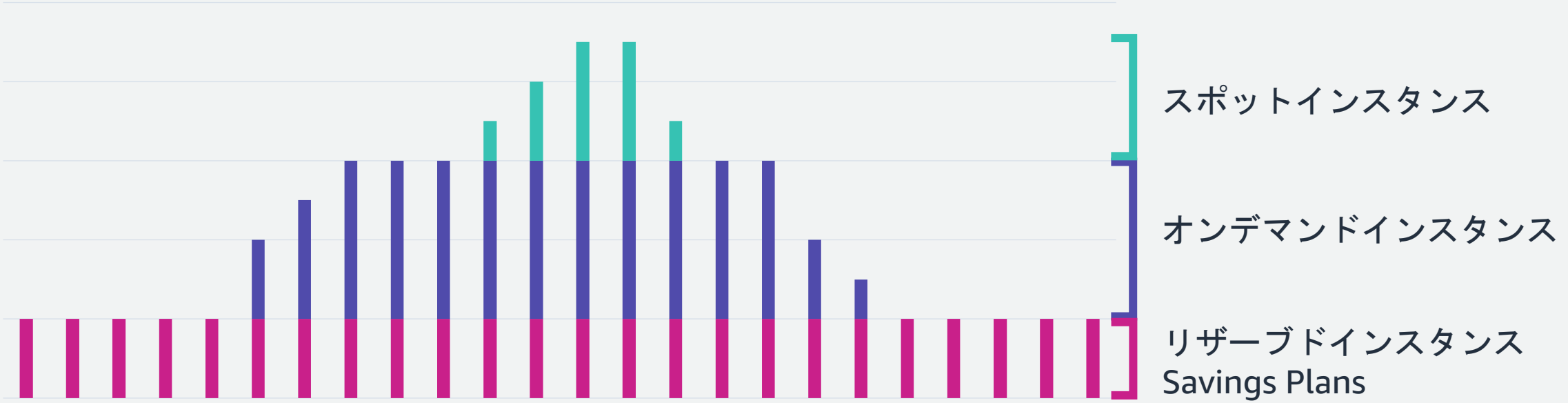


プロセッサとアーキテクチャの選択

Intel ® Xeon® Scalable (Skylake) プロセッサ	NVIDIA V100 Tensor Core GPU	AMD EPYC プロセッサ	Amazon ARM ベース クラウドプロセッサ	カスタムの FPGA ハードウェアアクセラレーション

アプリケーションとワークロードに応じて
最適なコンピューティング環境を選択

柔軟な購入オプションにより低コストでの研究利用が可能



研究用途での利用に向いており、低コストでの分析が可能

オンデマンドインスタンス

コンピューティング性能に対して
秒単位で 支払い
長期間のコミットメントは不要

スポットインスタンス

予備の EC2 キャパシティー
オンデマンド料金の**最大 90% を節約**

ステートレス、フォールトトレラント
ワークロード向け

リザーブドインスタンス

1 年または 3 年の契約を行うことで
オンデマンド **料金の割引** を受けられる

コンピューティングサービスの選択肢

EC2

サーバー

EKS

ECS

Fargate

コンテナ

Lambda

関数

自由度の高さ　HIgh → Low

管理の不要さ　Unmanaged → Managed

稼働に必要なもの

AMI (+ User data)
- サーバ/OSの設定
- ミドルウェア
- ランタイム

etc...

アプリケーションコード

Dockerイメージ
Dockerレジストリ

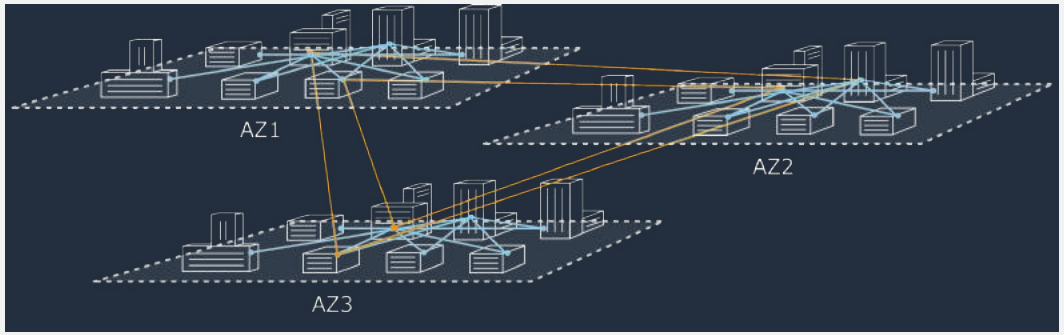
アプリケーションコード

アプリケーションコード

Amazon S3 (Simple Storage Service)

データ保存・バックアップ用途に向くオブジェクトストレージ

- 自動的に3箇所以上の AZ※へ隔地保管
- 設計上のデータ耐久性は、99.999999999%
- 容量無制限、サイジング不要
  - 1オブジェクトあたり5TBまで、
    オブジェクトの数は無制限
- データ容量に依存しない、**スケーラブルで安定した性能**
- 暗号化をサポート
- IAM、バケットポリシー、S3 アクセスポイントなどによる細かなアクセス制御が可能
- PrivateLinkを利用しDirectConnect 経由でアクセス可能



※ AZは物理的に距離の離れたデータセンター群
※ オブジェクトは自動的に3ケ所以上のAZへ隔地保存

S3ストレージクラス

S3 標準

- 頻繁にアクセスするデータ
- $0.023/GB～

S3 Intelligent-Tiering

- 変化するアクセスパターンのデータ
- $0.023～$0.002/GB

S3標準 - 低頻度アクセス

- 低頻度アクセスデータ
- $0.019/GB～

S3 1 ゾーン - 低頻度アクセス

- 再作成可能な低頻度アクセスデータ
- $0.0152/GB～

S3 Glacier

- アーカイブデータ
- $0.005/GB～

S3 Glacier Deep Archive

- アーカイブデータ
- $0.002/GB～

Frequent — *Access frequency* — *Archive*

様々な用途に応じて使い分けでコストパフォーマンスの向上

S3を中心とした様々なサービス連携が可能

その他のストレージの選択肢と比較

Amazon S3

スケーラブルなパフォーマンスを持ち、データ容量に対して低コストで利用できる

マウントして扱うことはできないため、ファイルを一度ローカルにコピーする必要がある。

Amazon Elastic File System (Amazon EFS)

フルマネージドでサーバの管理不要なファイルシステム
NFSによりマウント可能

パフォーマンスは利用容量に依存するため注意が必要。

Amazon FSx for Lustre

高いスループットを持つスケーラブルなファイルシステム
専用クライアントによってマウント可能
S3と統合されており、import/exportすることが可能

タカラバイオ株式会社 様

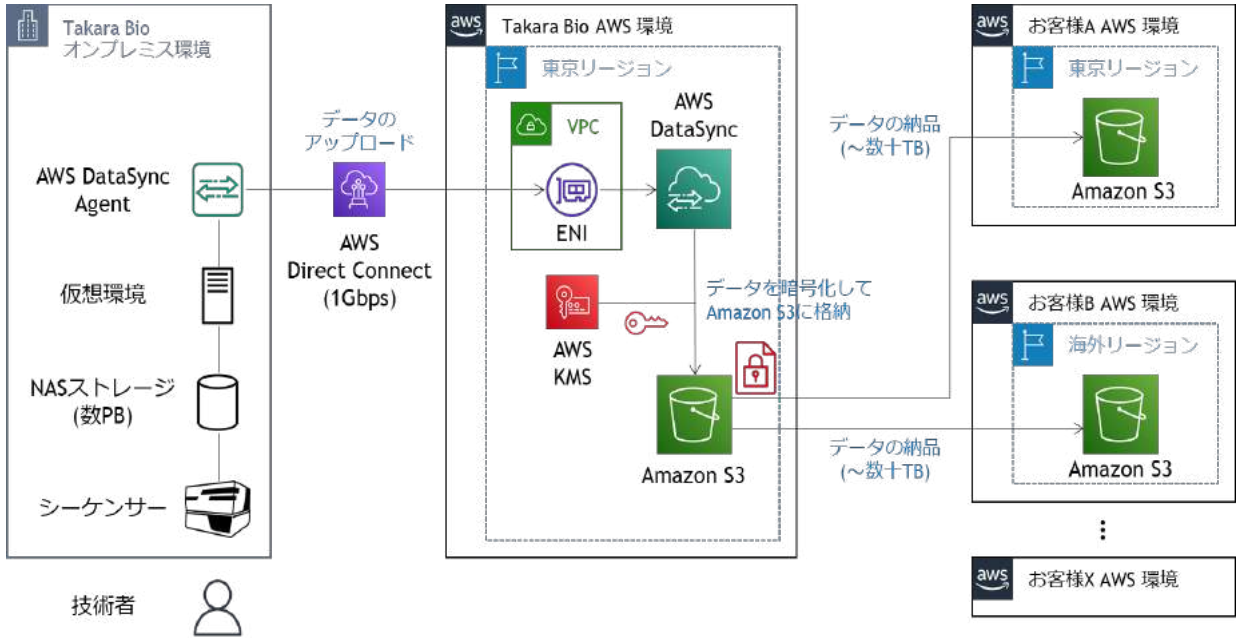
遺伝子解析・検査受託サービスにおける解析データの納品手段として、オンライン納品を確立
ゲノムデータ等高いセキュリティが要求される基盤にAWSを採用

データ提供元と提供先において利用するAWSアカウントIDが異なる（クロスアカウント）環境で、Amazon S3バケット間コピーでオンライン納品を実現

データ転送はAWSグローバルネットワーク内に閉じており、またS3バケットのデフォルト暗号化設定で、高セキュリティを確保

ゲノムデータの大容量化もAmazon S3の活用でスケーラビリティを確保

AWS DataSyncを活用したことで、データ転送効率が3倍アップ



In-silico創薬研究におけるAWSの提供する価値

データ活用の課題

膨大なデータ量

共同研究者との安全なデータ共有

容量無制限のスケーラブルなストレージ

きめ細やかなアクセス制御による

安全なデータ共有

コンピューティングの課題

日々の増減する計算需要への対応

計算リソース保守管理

必要な時に必要なだけ計算リソースを確保

物理リソースの保守管理はAWSにお任せ

研究ワークフローの課題

反復的な手動タスクによる効率低下

複数の処理を連携し自動化

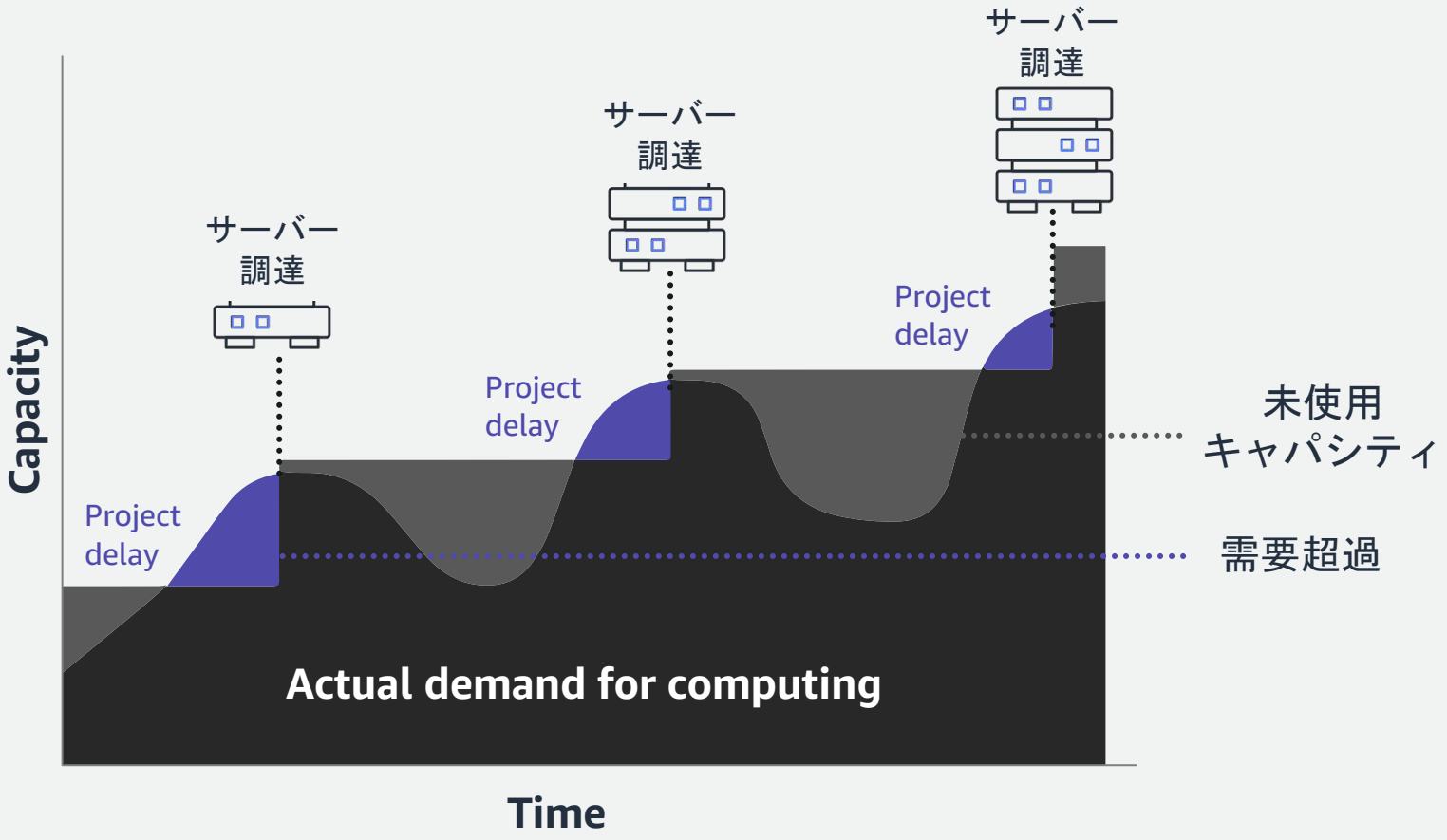
データドリブンで

高スループットな解析を実現

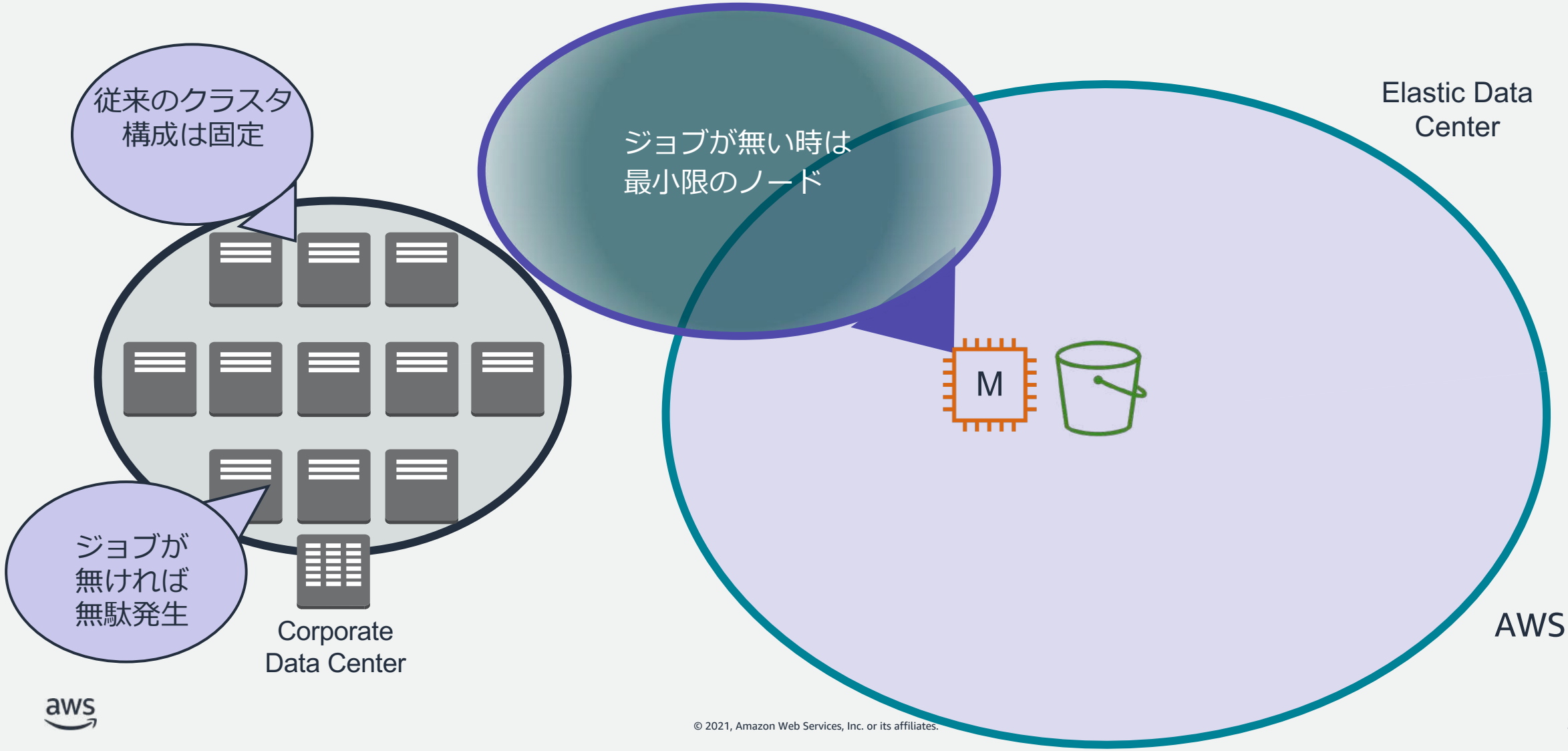
・HPC on AWSの特徴・サービス

従来のHPCクラスタの課題

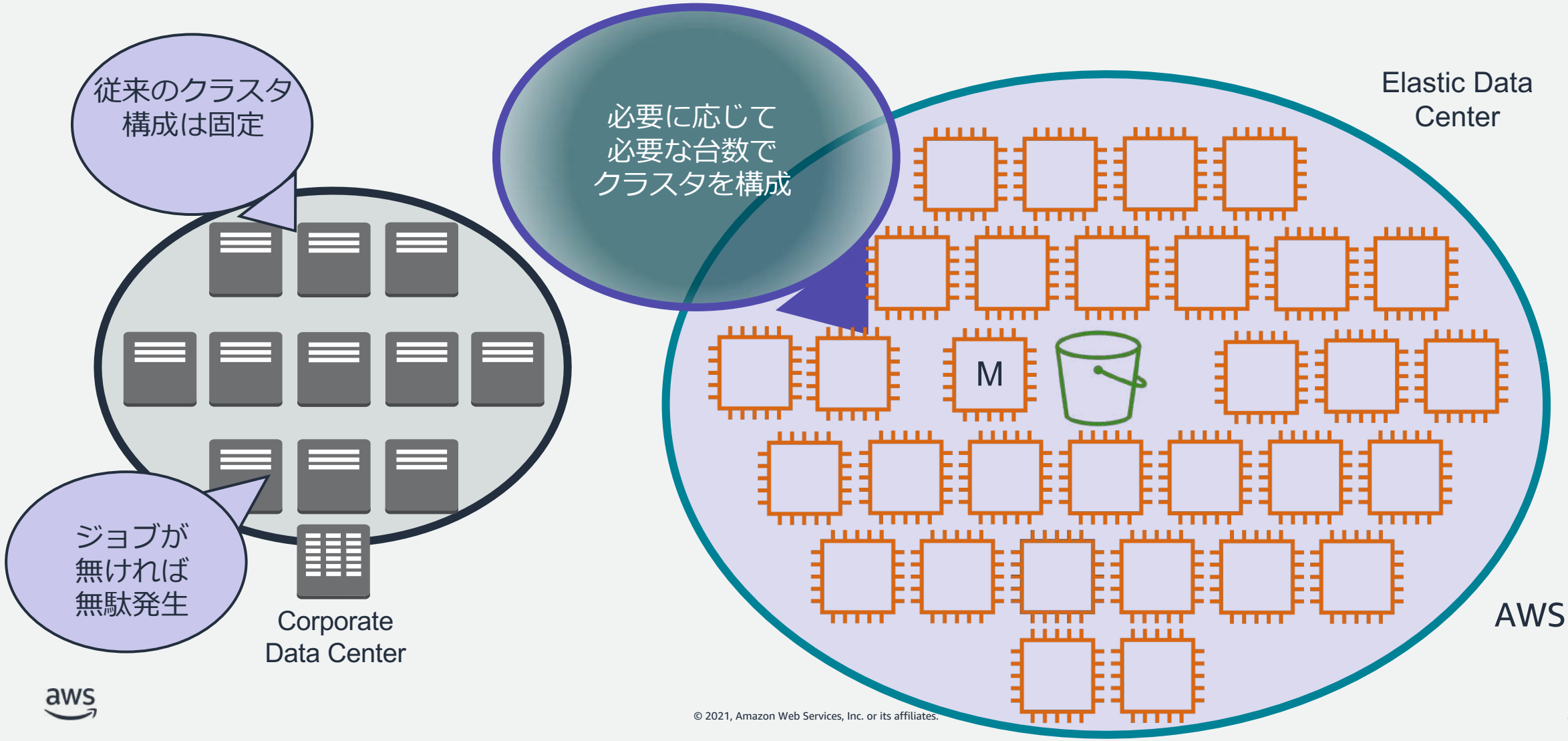
- オンプレミスでは変動する計算需要に対して、過不足の無いキャパシティを用意するのは非常に困難
- キャパシティの増加には時間がかかるため、研究ニーズへの対応が遅れる



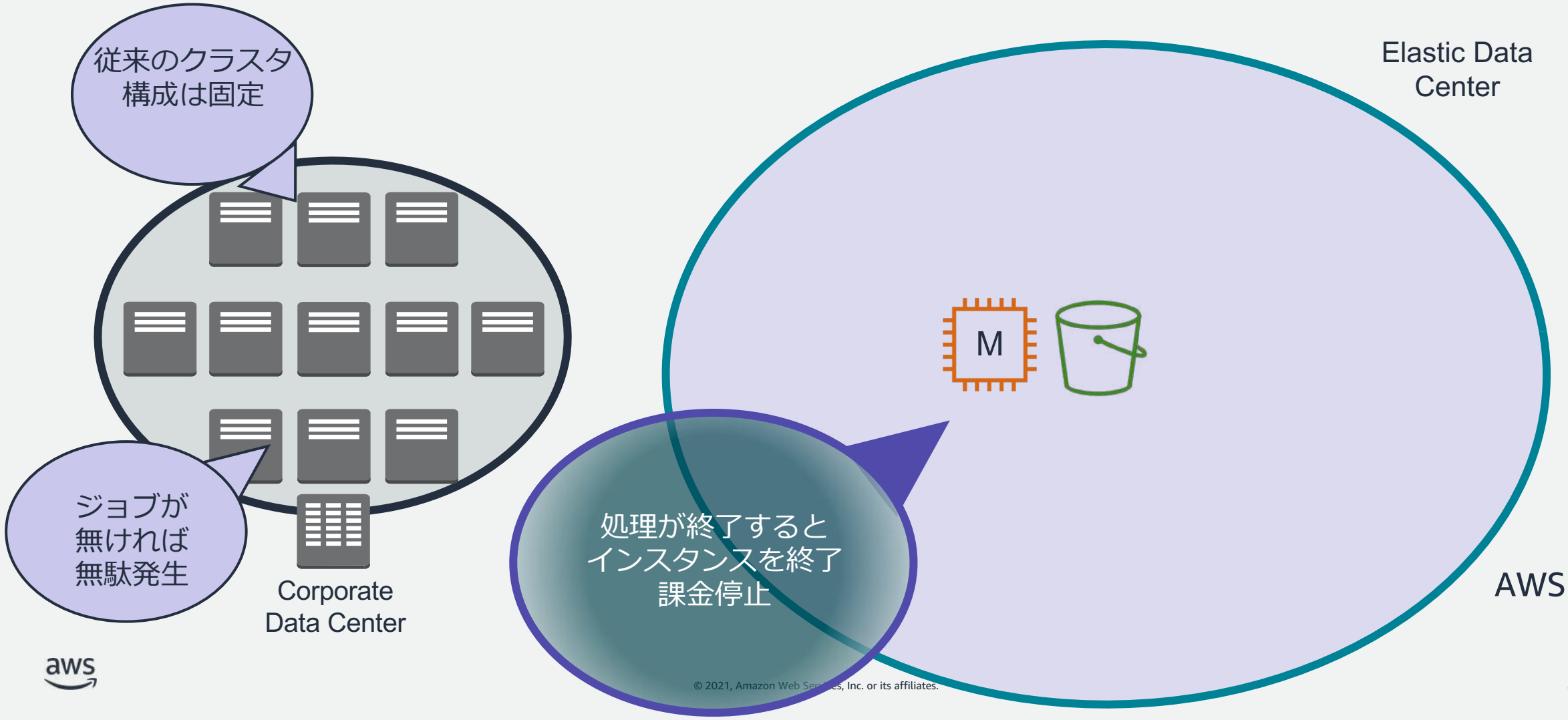
AWSではジョブ実行待ちの無いHPC環境を実現



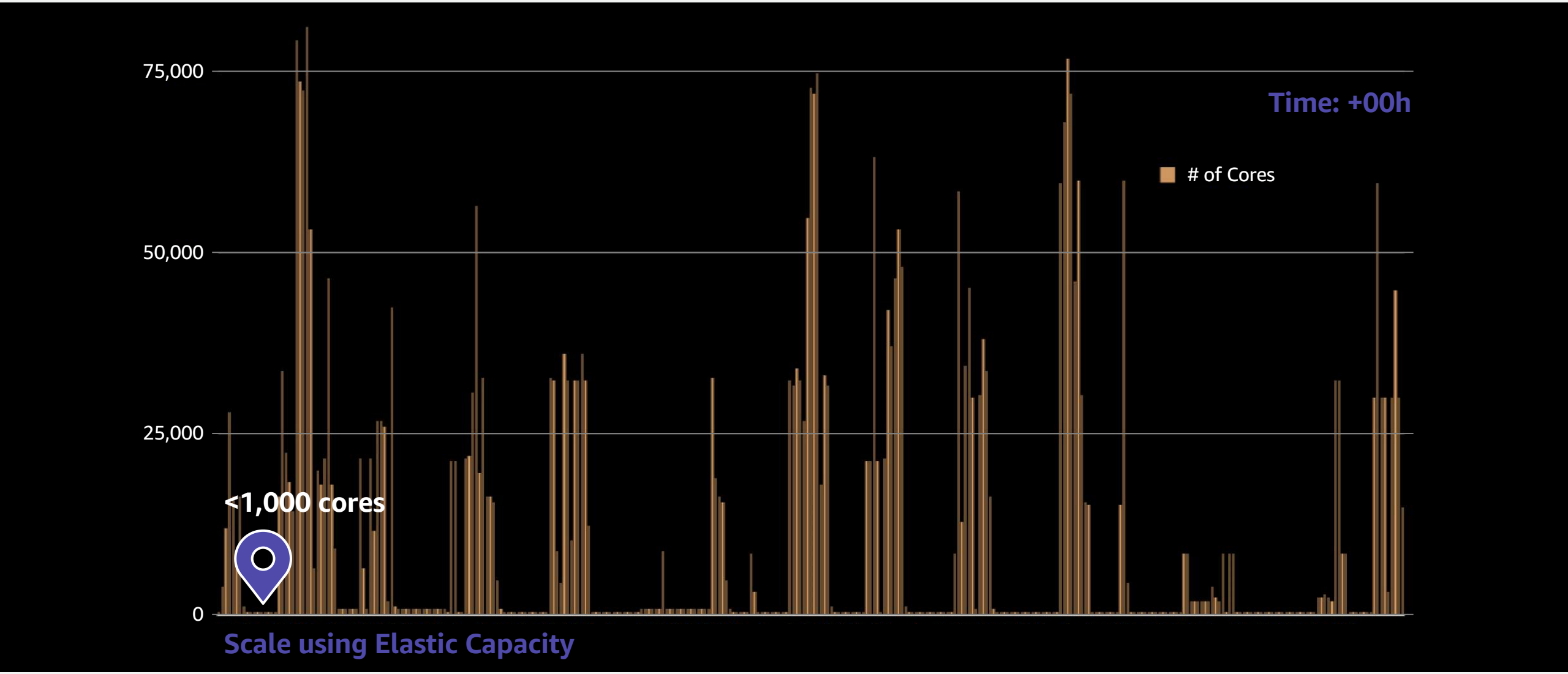
AWSではジョブ実行待ちの無いHPC環境を実現



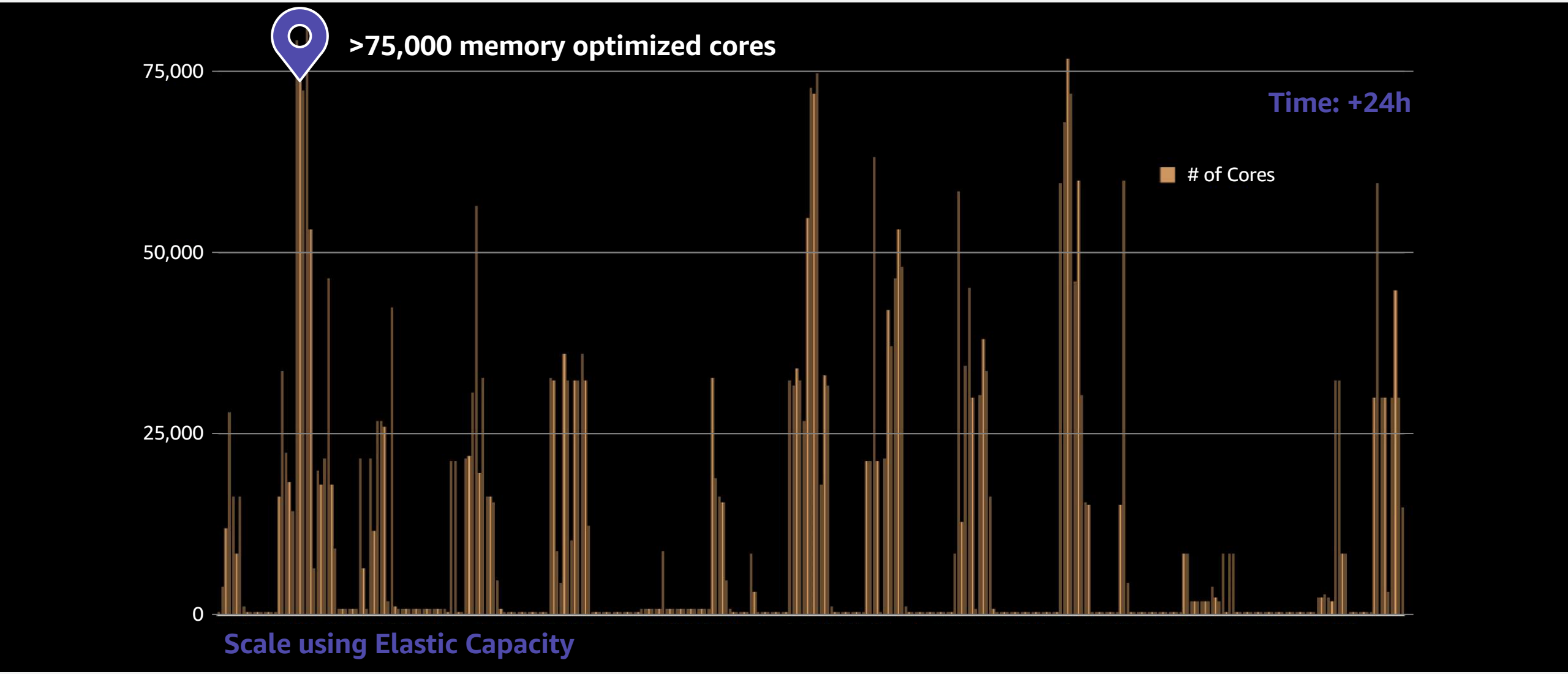
AWSではジョブ実行待ちの無いHPC環境を実現



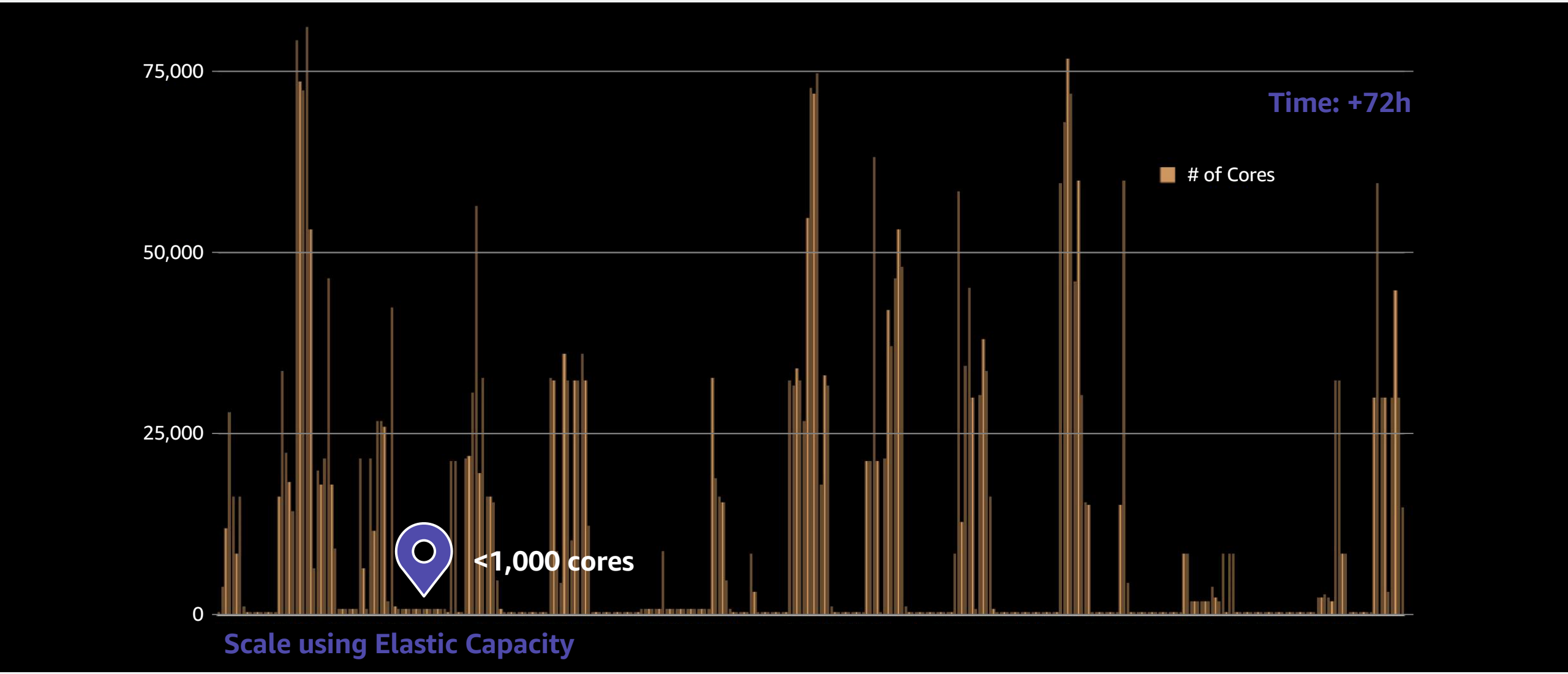
実際のスケーリングの様子



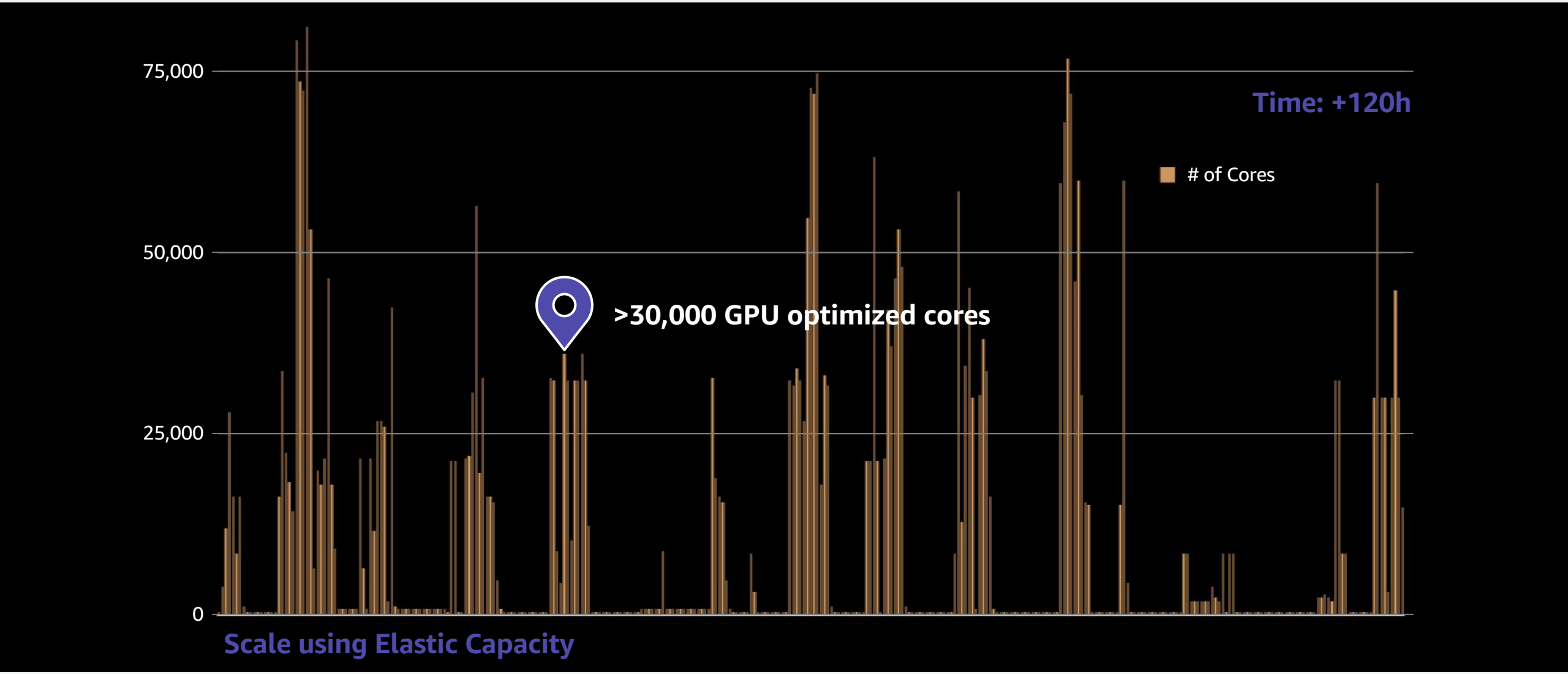
実際のスケーリングの様子



実際のスケーリングの様子

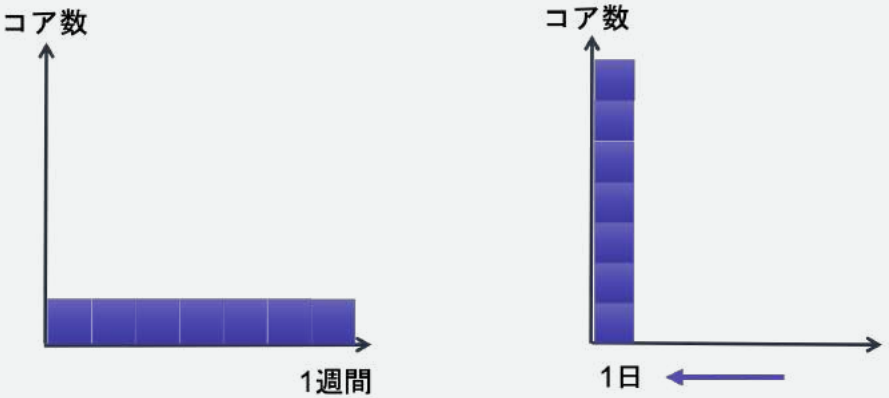


実際のスケーリングの様子



AWSで大規模コンピューティングを実行するには

クラウドのスケーラビリティの活用により
コスト効率よく大量の計算を行うことが可能



コンピューティング費用は「時間x台数」の積算なので
逐次処理をしても並列処理をしても費用は同じ

AWSのBuilding Blockで実現可能
しかし1からの構築は大変

実行基盤としてのEC2,ECS
ジョブのスケジューリングやインスタンスのスケールの仕組み
計算環境、ソフトウェアの管理
etc …

AWSのHPC向けサービス活用により、すぐに利用いただくことが可能

AWS Batch

コンテナベースの大規模バッチジョブ
コンピューティング環境を
フルマネージドで提供

AWS ParallelCluster

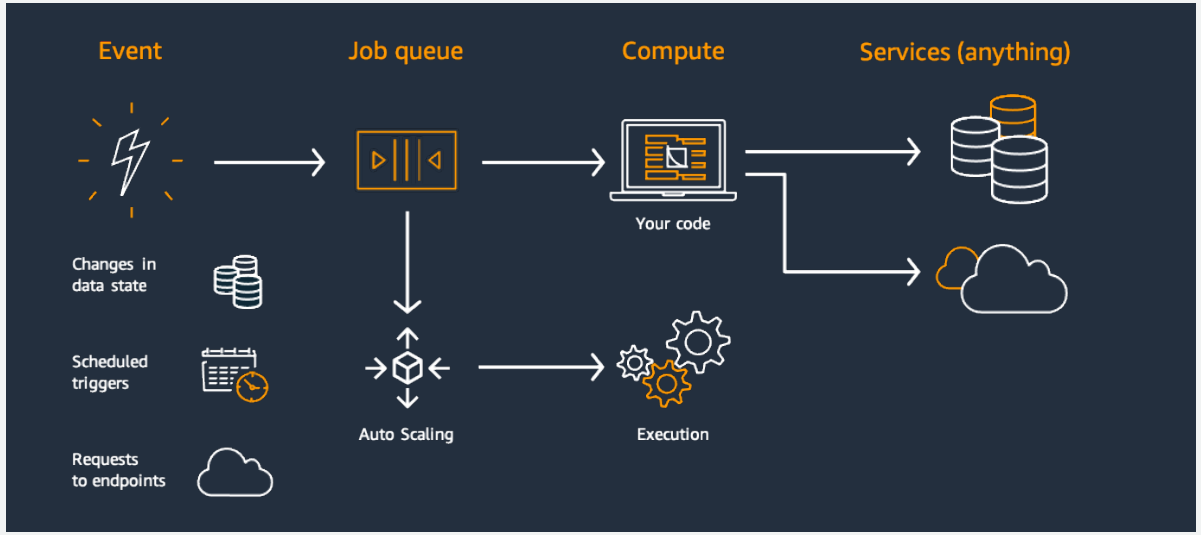
AWS上に HPC クラスタを自動で構築
Slurm / SGE / Torque といったジョブスケジューラに対応しており既存HPC環境からの移行が容易

本日はこちらをご紹介

AWS Batchとは

バッチコンピューティングのため環境をフルマネージドで提供

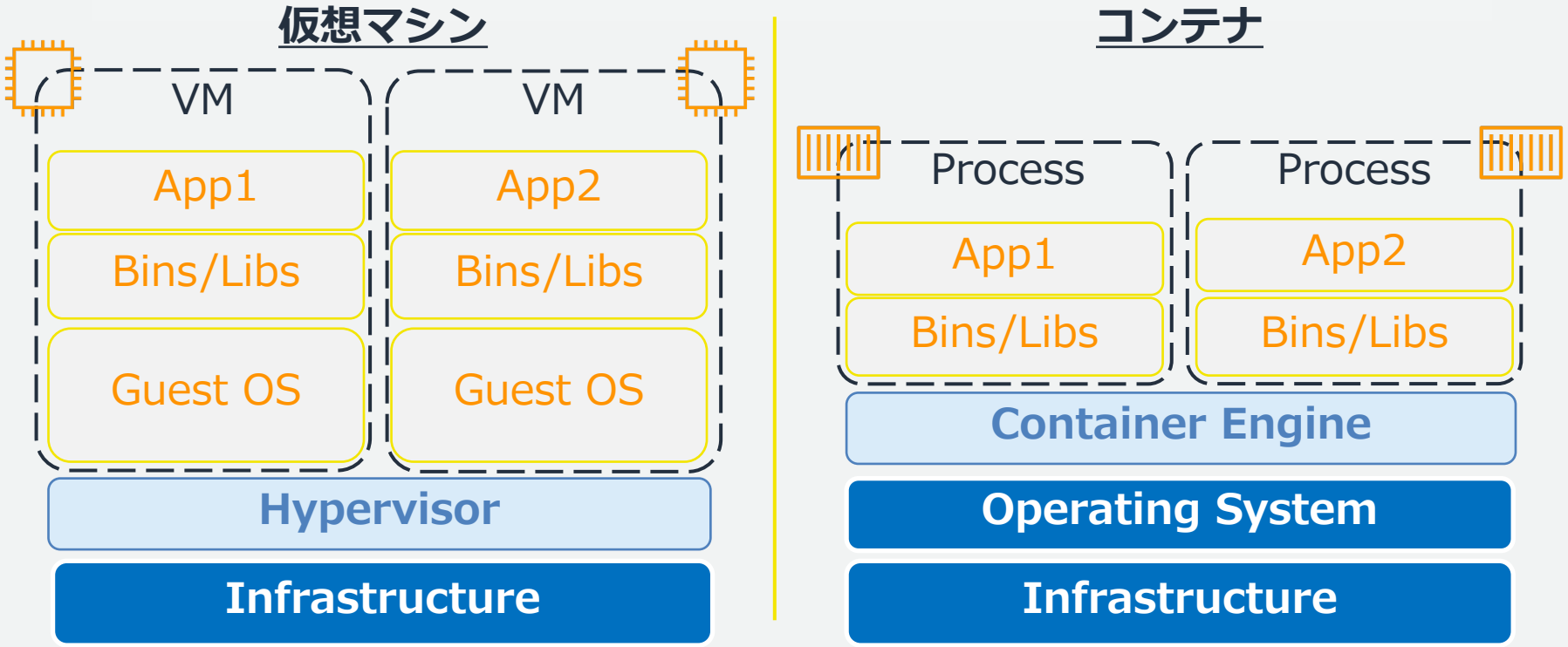
- AWS Batch がインスタンスの起動や停止を行うため、スケジューラや計算ノードなどの **管理が不要**
- ジョブは **Docker コンテナイメージ** を元に作成し、自動でスケールするコンピューティング環境で実行する
- コンピューティング環境ではインスタンスタイプや vCPU 数、スポットインスタンス利用有無などを任意に指定可能
- 100 万 vCPU クラスの大規模な計算にも対応

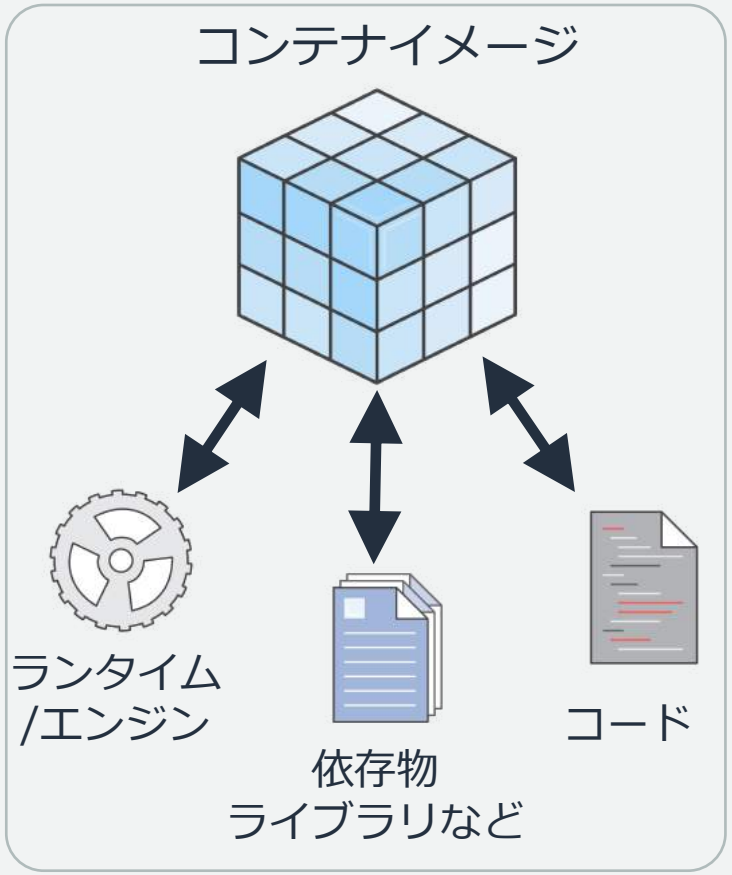


コンテナを用意するだけでスケーラブルなバッチコンピューティング環境が得られる

(補足)コンテナとは

リソースが隔離されたOS上のプロセス
(仮想マシンと同様に「起動・停止・削除」などのライフサイクルを持つ)





1つのOS上で、複数同時稼働実行環境を提供。
各々で独立したルートファイルシステム、CPU・メモリ、プロセス空間等を利用可能

(補足)コンテナの特徴とメリット

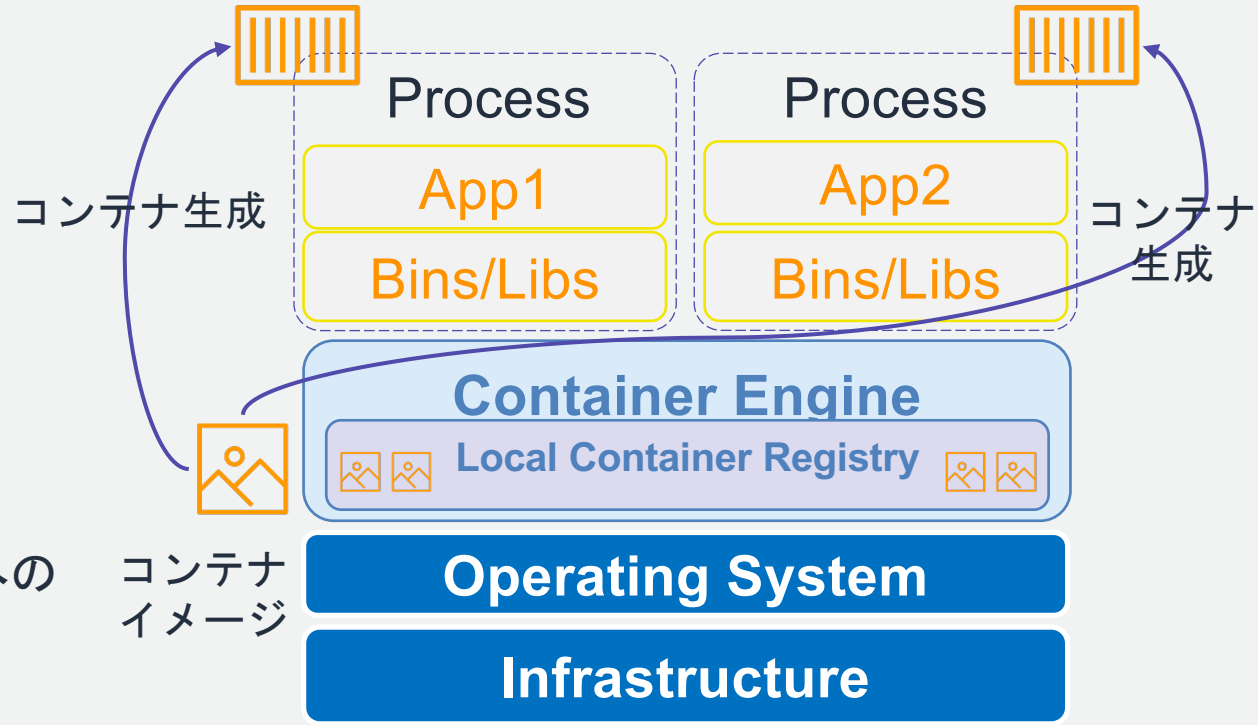
1つのイメージより複数のプロセスを起動できるため、High Throughputな計算で便利

ランタイムや依存物を含めてパッケージしているため、可搬性に優れ、処理に再現性が得られる

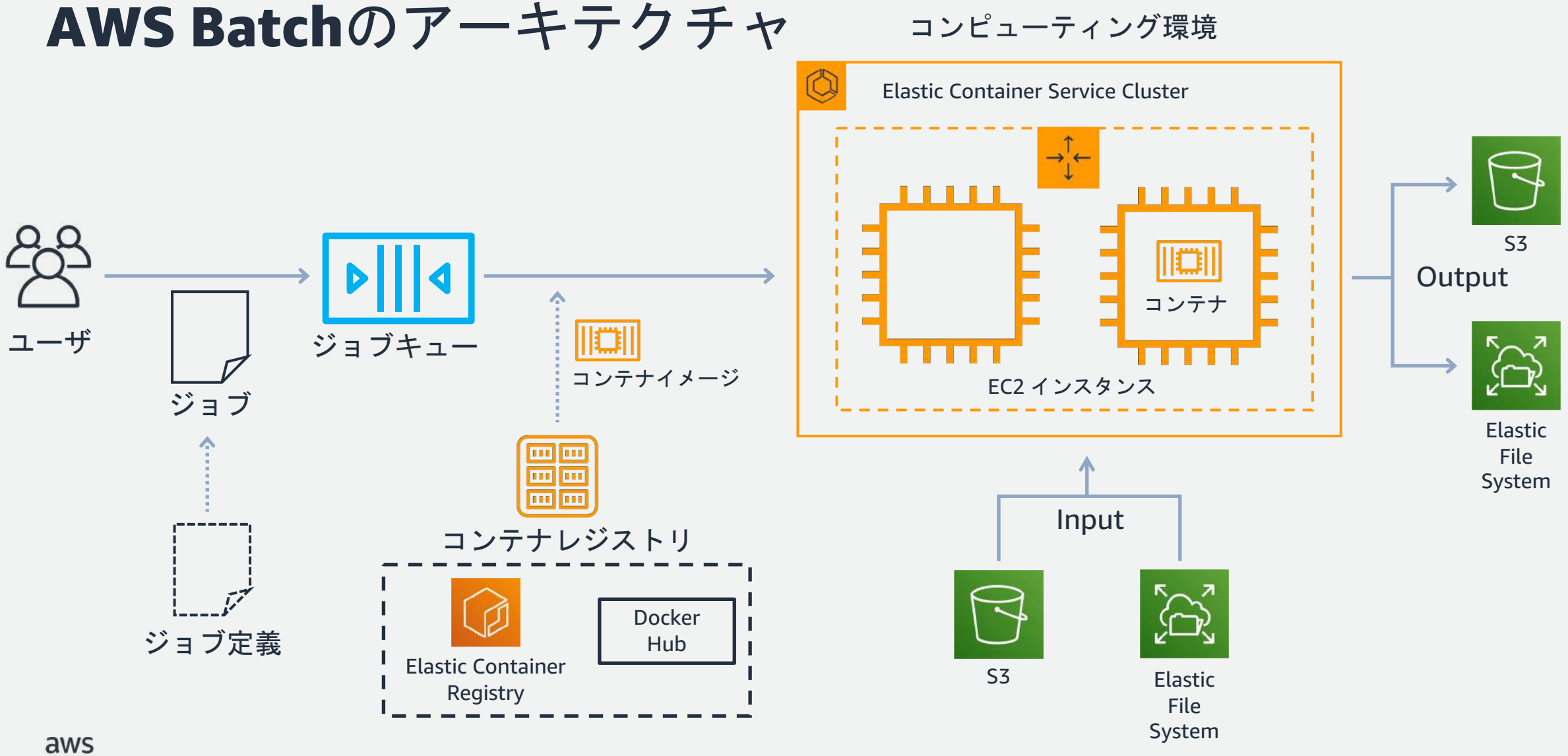
■ **スピード**
✓ 起動・停止が非常に高速

■ **柔軟性**
✓ 1つのイメージから複数の
コンテナを起動可 (スケール性)

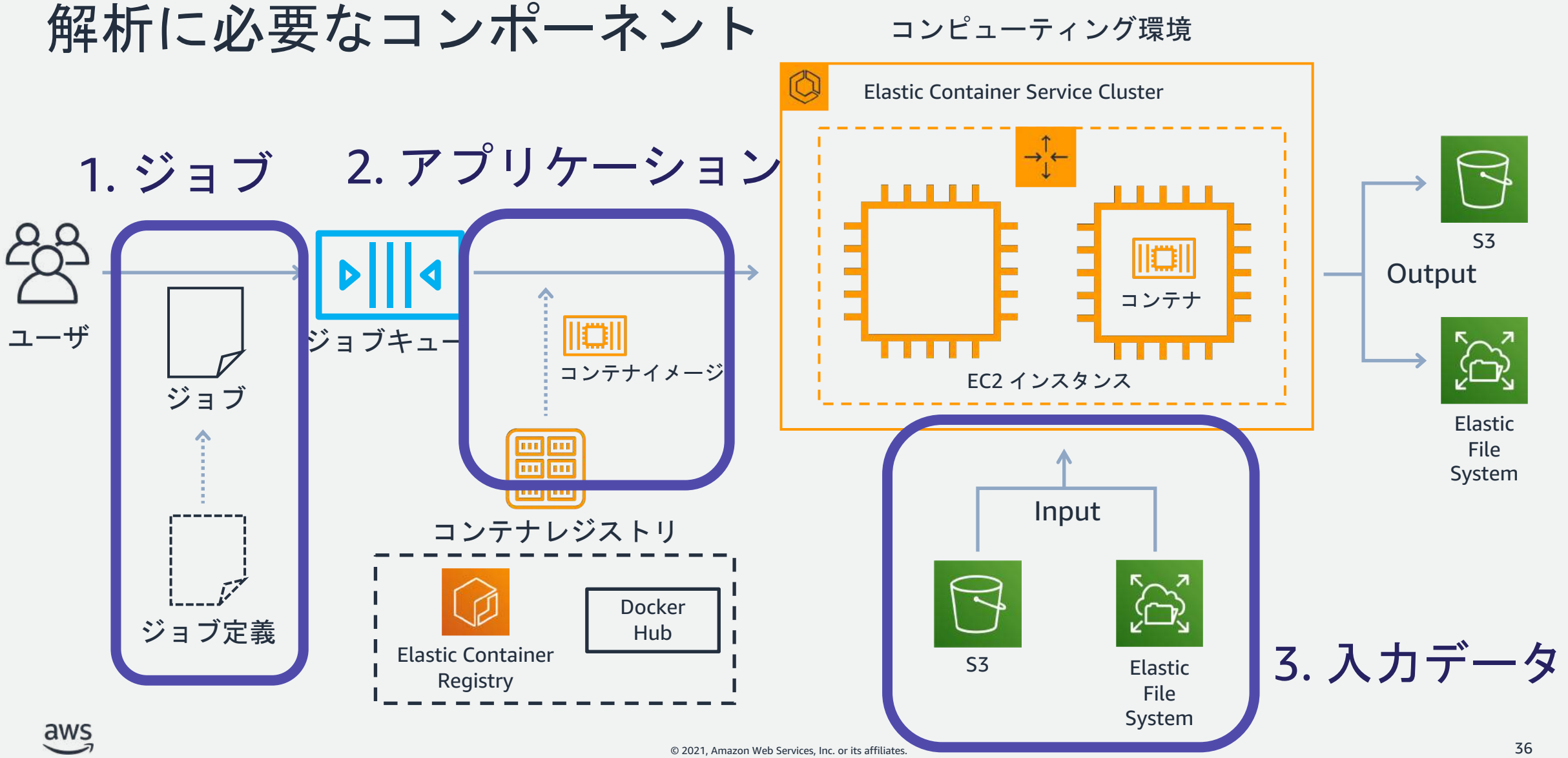
■ **可搬性**
✓ コンテナイメージは「不変」
✓ 「アプリケーションのビルドとデプロイ」への
組み込みが容易



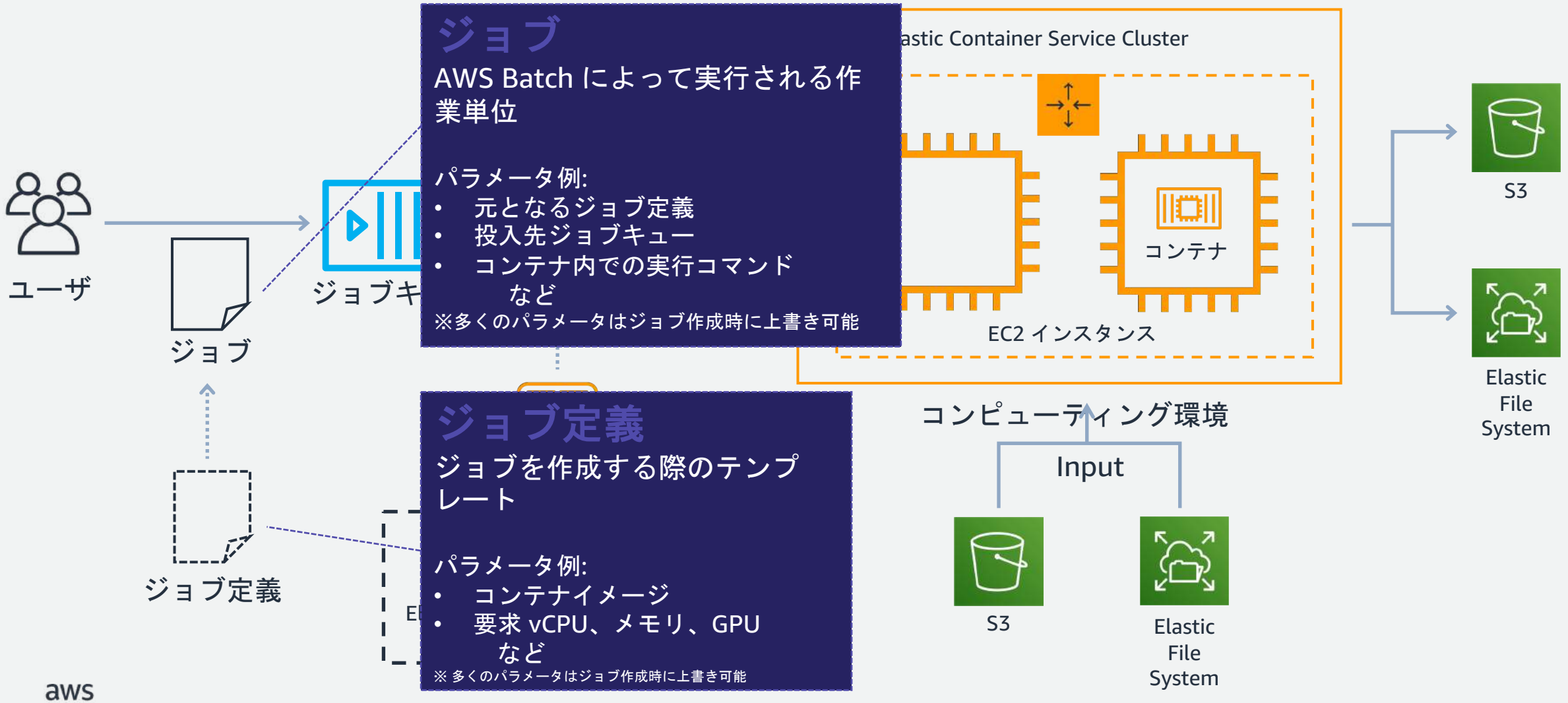
AWS Batchのアーキテクチャ



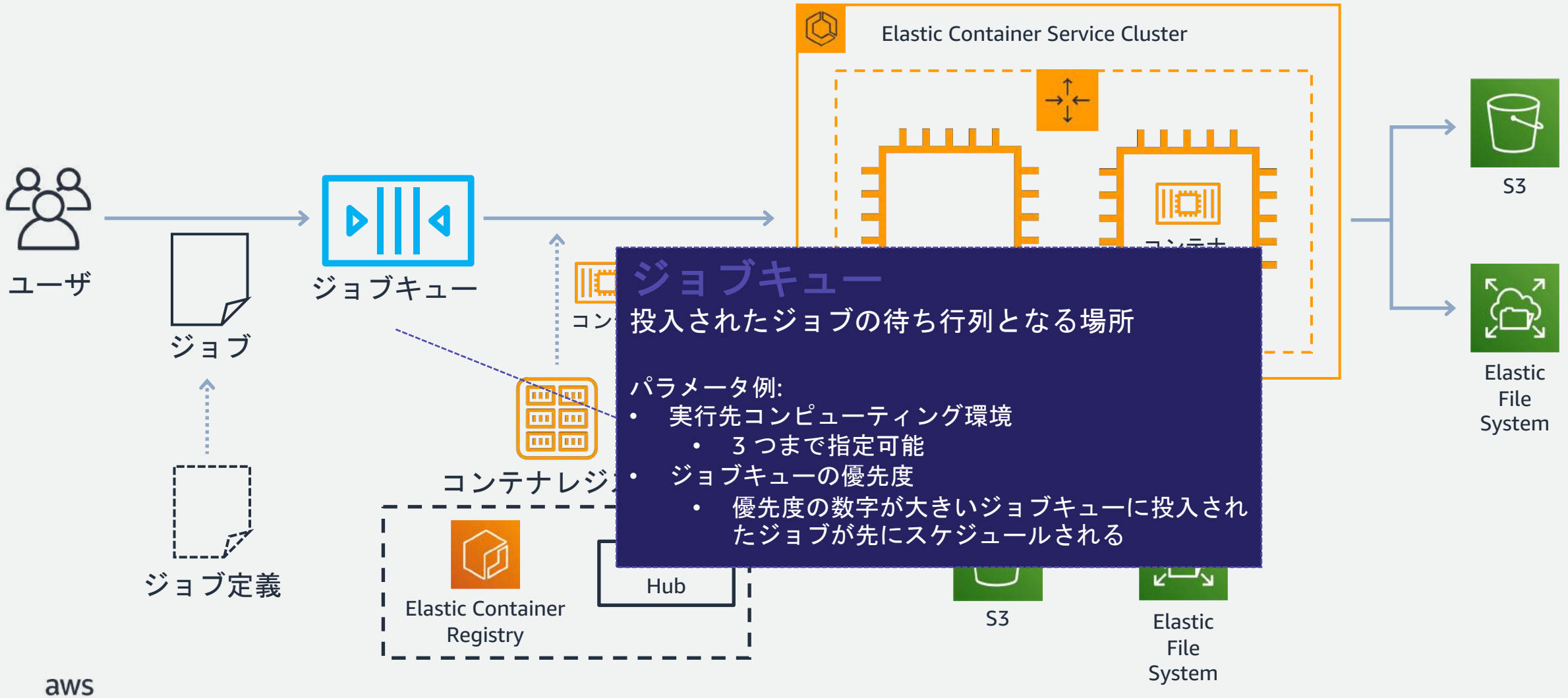
解析に必要なコンポーネント



AWS Batchのアーキテクチャ



AWS Batchのアーキテクチャ



コンピューティング環境

実際に計算を行う ECS クラスター

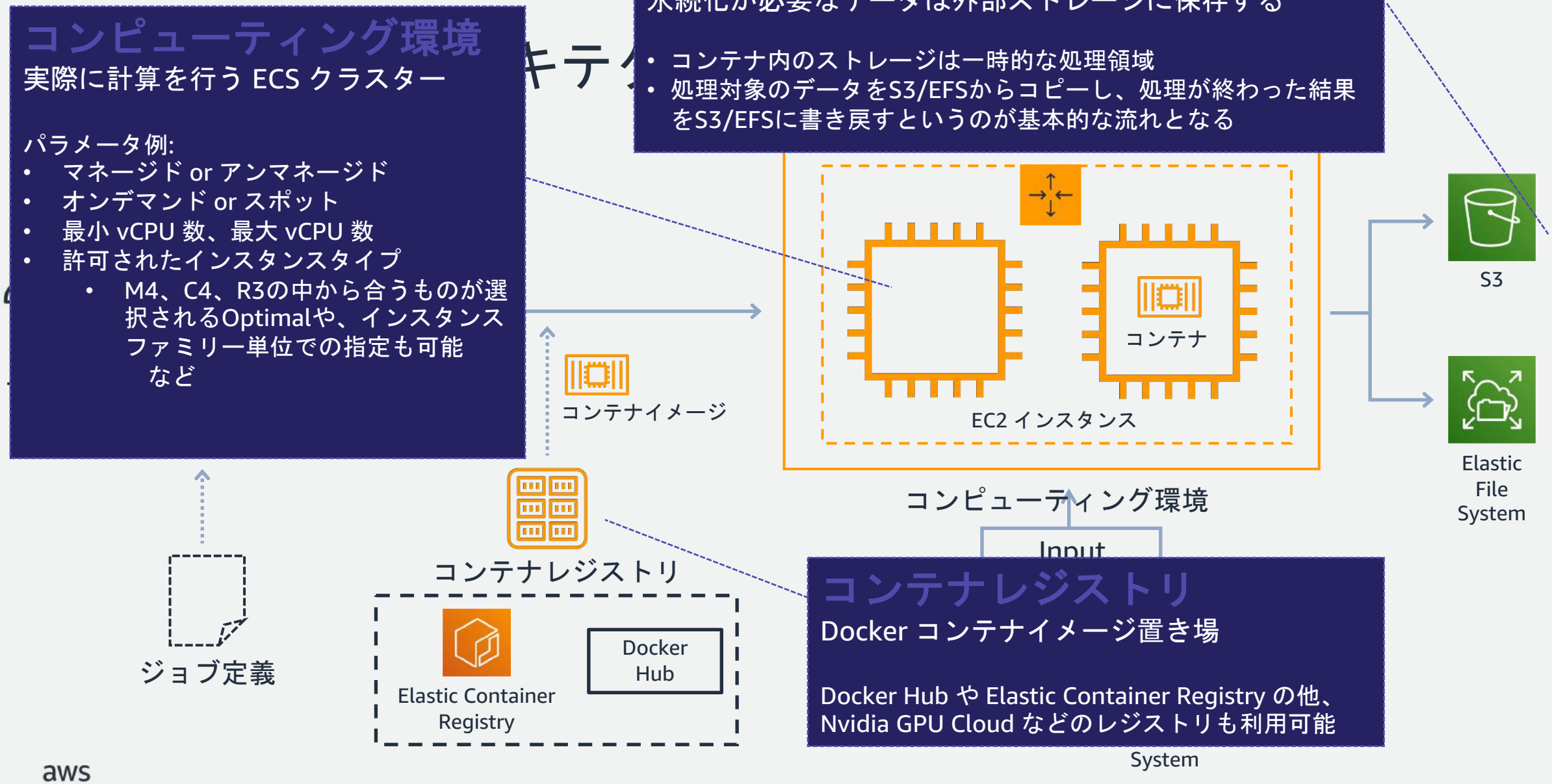
パラメータ例:

- マネージド or アンマネージド
- オンデマンド or スポット
- 最小 vCPU 数、最大 vCPU 数
- 許可されたインスタンスタイプ
  - M4、C4、R3の中から合うものが選択されるOptimalや、インスタンスファミリー単位での指定も可能
    など

ストレージ

永続化が必要なデータは外部ストレージに保存する

- コンテナ内のストレージは一時的な処理領域
- 処理対象のデータをS3/EFSからコピーし、処理が終わった結果をS3/EFSに書き戻すというのが基本的な流れとなる



コンテナレジストリ

Docker コンテナイメージ置き場

Docker Hub や Elastic Container Registry の他、Nvidia GPU Cloud などのレジストリも利用可能

HPC on AWSの特徴・サービスまとめ

コンピューティングの課題

日々の増減する計算需要への対応

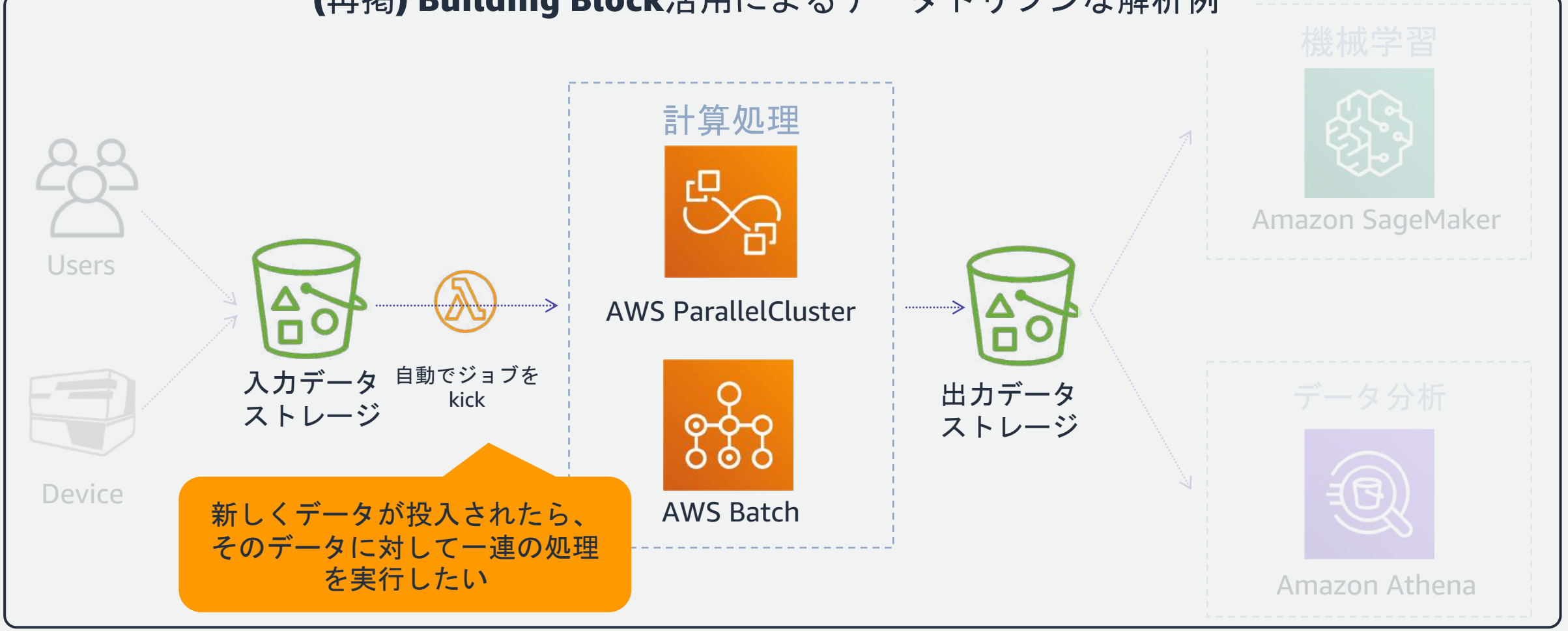
計算リソース保守管理

- クラウドのスケーラビリティにより、必要な時に必要な分だけ計算リソースを確保することで、コスト効率よく大量の計算を行うことが可能
- スケーラブルなバッチコンピューティング環境を提供するAWS Batchをご紹介
- AWS Batchを活用いただくことで、スケジューラや計算ノードを構築/管理することなく、自動でスケールする環境でのジョブ実行が可能

- サービスを組み合わせた解析ワークフロー自動化の方法

処理を自動化し研究効率を向上させる

(再掲) Building Block活用によるデータドリブンな解析例



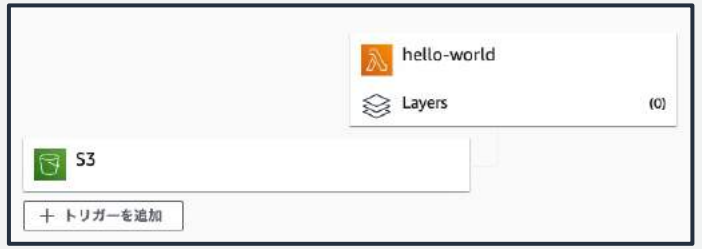
Amazon S3とAWS Lambdaによる自動処理の例

AWS Lambda

- 簡単なアプリケーションコードを実行できるサービス
- 実行環境の管理は不要
- 様々なイベントをトリガーに起動できる

Amazon S3

- データが投入されたタイミング（オブジェクトが作成された）際に通知を送るS3イベント通知機能を使ってLambdaとシームレスな連携が可能

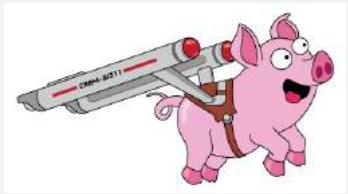


より複雑な処理の自動化

特にゲノミクス分析の分野では、解析ツール数が多くやパイプラインが複雑

AWS Step Functions

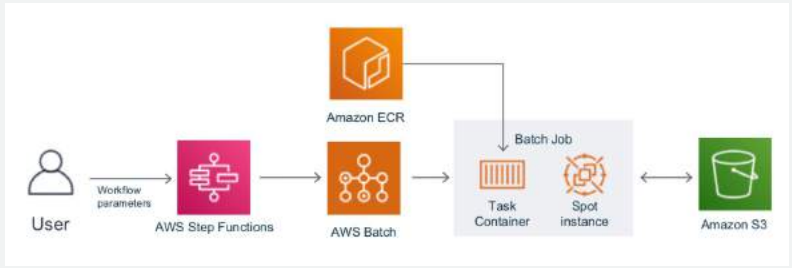
AWS サービスのオーケストレーション・ワークフローサービス

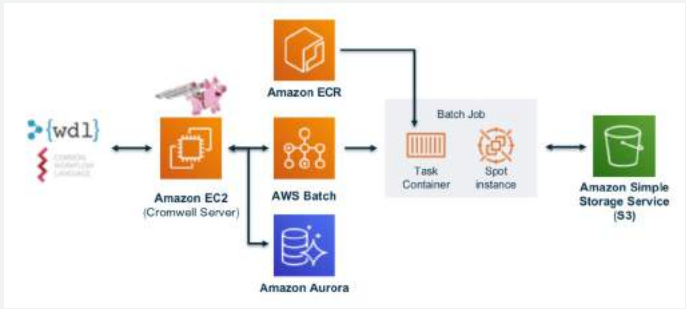


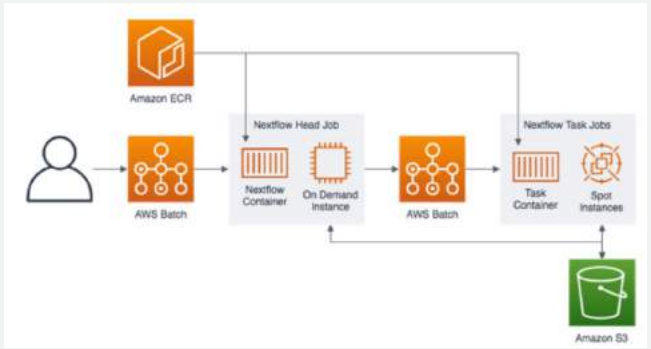
Cromwell

オープンソース オーケストレーションエンジン

どちらのパターンでもクイックに開始できるアーキテクチャ/テンプレートを提供







https://docs.opendata.aws/genomics-workflows/orchestration/orchestration-intro.html

ゲノミクス分析環境を素早く利用する方法

AWS CloudFormation

- AWSリソースと依存関係をテンプレートとして記述、それらのリソース群をスタックとして一括で起動をサービス
- スタック全体を容易に素早く作成・更新・削除が可能
- AWSが提供しているテンプレートを活用することで、リファレンス実装を素早くお試しいただくことが可能

Amazon Genomics CLI

- 2021年 9月27日に一般公開
- ゲノミクスワークフロー環境を素早くセットアップし実行するためのオープンソースのコマンドラインインターフェース（CLI）ツール
- 現在ワークフローエンジンとしてNextflow, Cromwell+WDLに対応

AWSソリューションズ実装の活用

https://aws.amazon.com/jp/solutions/implementations/

- AWSのBuilding Blockを使用し、実ユースケースにフォーカスしたソリューションをリファレンス実装として提供
- CloudFormationテンプレートやデプロイガイドが提供されており素早く検証、実装いただくことが可能

Biotech Blueprint on the AWS Cloud

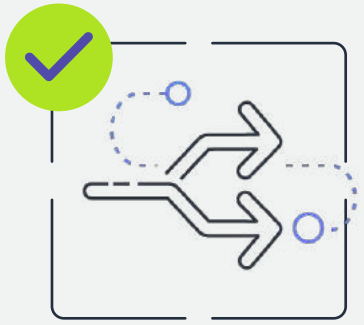
https://aws.amazon.com/quickstart/biotech-blueprint/biotech-blueprint/

Nextflow

https://aws.amazon.com/jp/quickstart/biotech-blueprint/nextflow/

解析ワークフロー自動化のまとめ

研究ワークフローの課題



反復的な手動タスクによる効率低下

- AWSのBuilding Blockを組み合わせることで、自由度高く、自動化処理を簡単に構築することが可能
- ゲノミクスなどのより複雑なワークフローに対しては、AWS Step Functionsやオープンソースのオーケストレーションエンジンをご利用いただくことが可能
- AWSが提供するテンプレートやツールにより、素早く環境を立ち上げることが可能

まとめ

- AWSの活用で様々な研究者のニーズを満たす基盤を構築できる
- コスト効率よく大規模計算を行うことができるAWS Batchをご紹介
- ゲノミクス分析の自動化や、構築済み環境を素早く起動する方法をご紹介

研究領域や業務でお困りごとありましたら、是非ご相談ください

AWSヘルスケア・ライフサイエンスのご紹介ページ

https://aws.amazon.com/jp/local/health/

お問い合わせ先

https://aws.amazon.com/jp/contact-us/

Thank you!