



Amazon EMR

基礎編

川村 誠

Solutions Architect
2023/09

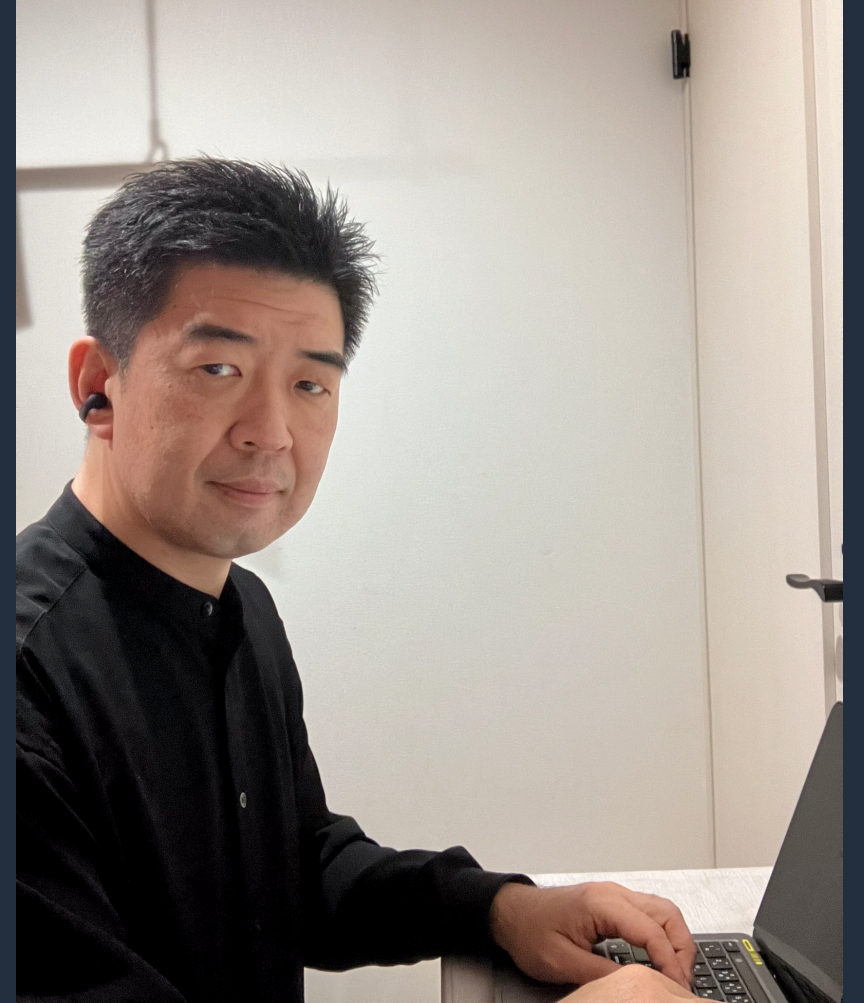
自己紹介

名前：川村 誠

所属：データ事業本部
ソリューションアーキテクト

好きなAWSサービス：

- Amazon EMR
- AWS Glue



アジェンダ

- はじめに
- Amazon EMR の基礎
- 進化するプラットフォーム
- クラスターのステートレス化
- Auto scaling
- スポットインスタンスの活用
- クラスター運用の自動化
- セキュリティ
- 料金
- 事例紹介
- まとめ

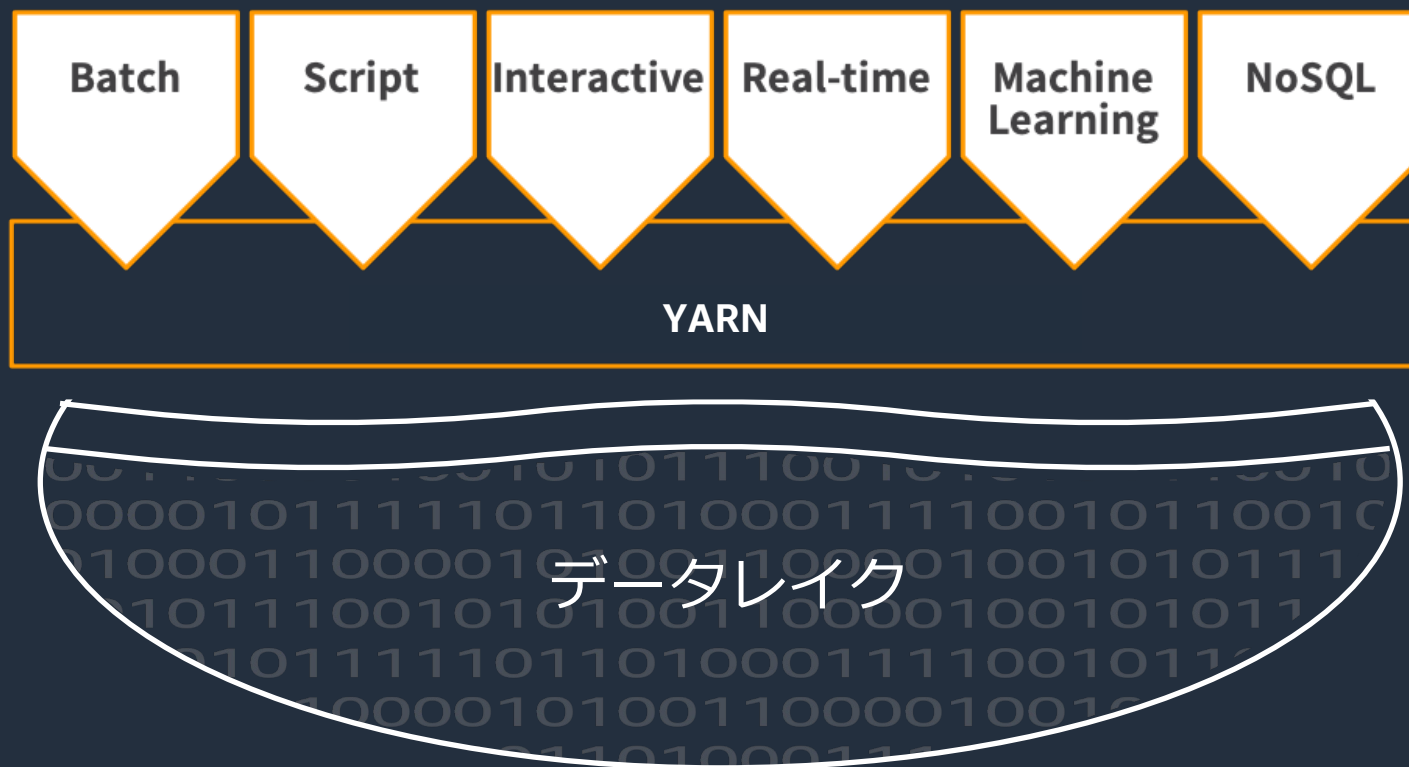
はじめに



Apache Hadoop とデータレイク



Apacheは、Apache Software Foundationの登録商標または商標です



- 分散処理
- 多様な分析
 - Batch/Script (Hive/Pig)
 - Interactive (Spark, Presto)
 - Real-time (Spark)
 - Machine Learning (Spark)
 - NoSQL (HBase)
- 幅広いユースケース
 - Log and clickstream analysis
 - Machine Learning
 - Real-time analytics
 - Large-scale analytics
 - Genomics
 - ETL

Amazon EMR の基礎



Amazon EMR

Spark, Presto, Trino, Hadoop, Hive, Hbase, Flink などの
オープンソースフレームワークを使用したビッグデータ分析が可能



パフォーマンスが最適化されたランタイムを利用可能

Spark、Hive、Presto、Flinkなどの一般的な
フレームワーク向けにパフォーマンスが最適化
されたランタイムで、100% オープンソース
API互換性を実現



セルフサービスデータサイエンス

EMR Studio を利用したデータサイエンス IDE と
Amazon SageMaker Studio との緊密な統合により、
オープンソース UX とフレームワークを使用した
アプリケーションの構築、視覚化、デバッグが可能に



最新の OSS 機能を利用可能

オープンソースでリリースされる新しい機能が
60日以内に利用可能になる



様々なデプロイメントオプション

EMR は、EC2・EKS・AWS Outposts 上で
ビッグデータワークロードを実行する
柔軟性を提供。新たなオプションとして、
EMR Serverless が利用可能に



ビッグデータ分析に最適なコストパフォーマンス

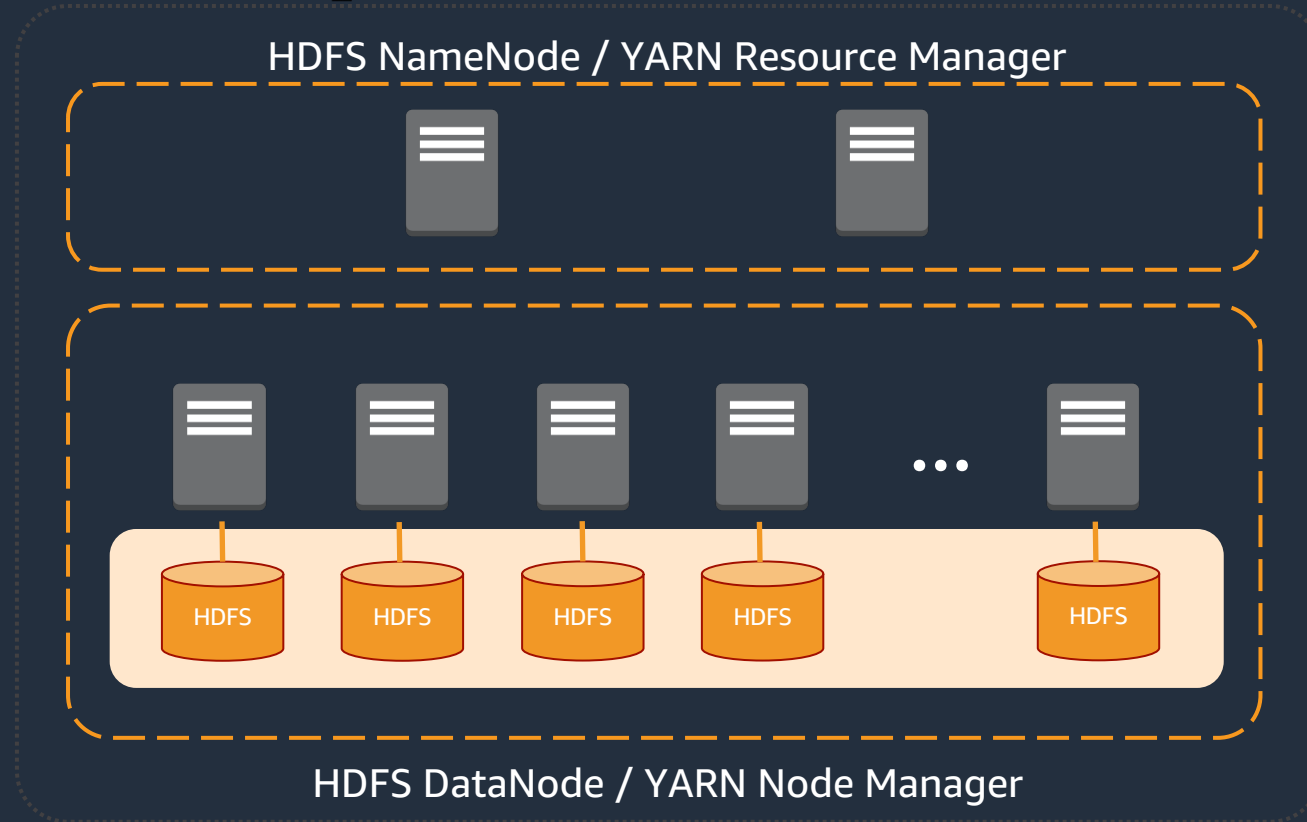
Amazon EC2 スポット、Amazon EMR マネー
ジドスケールリング、および 1 秒単位の請求を
使用してコストを削減



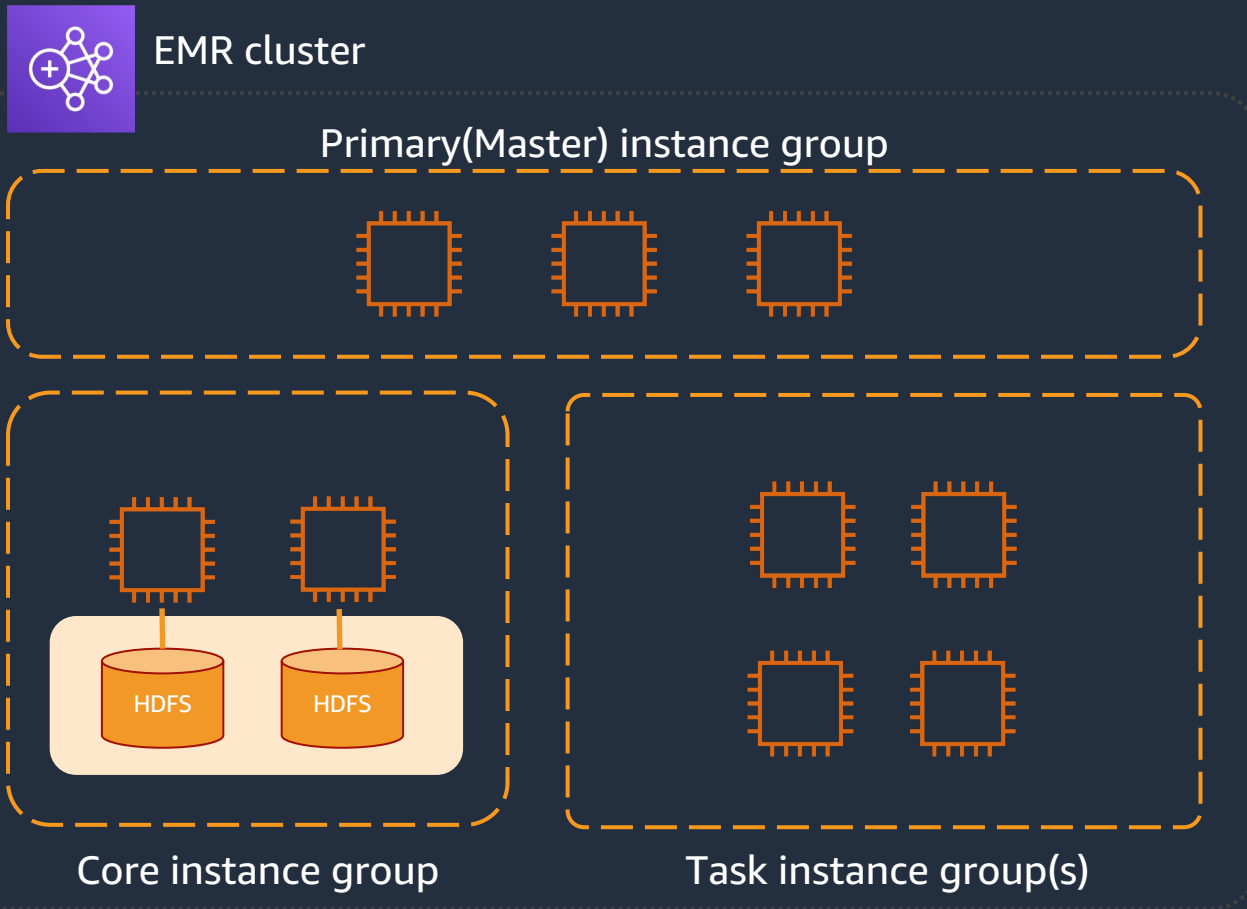
S3 データレイク統合

AWS Lake Formation と Apache Ranger による
きめ細かなアクセスコントロール、および
Apache HUDI との統合が様々な Amazon S3
データレイクのユースケースを実現

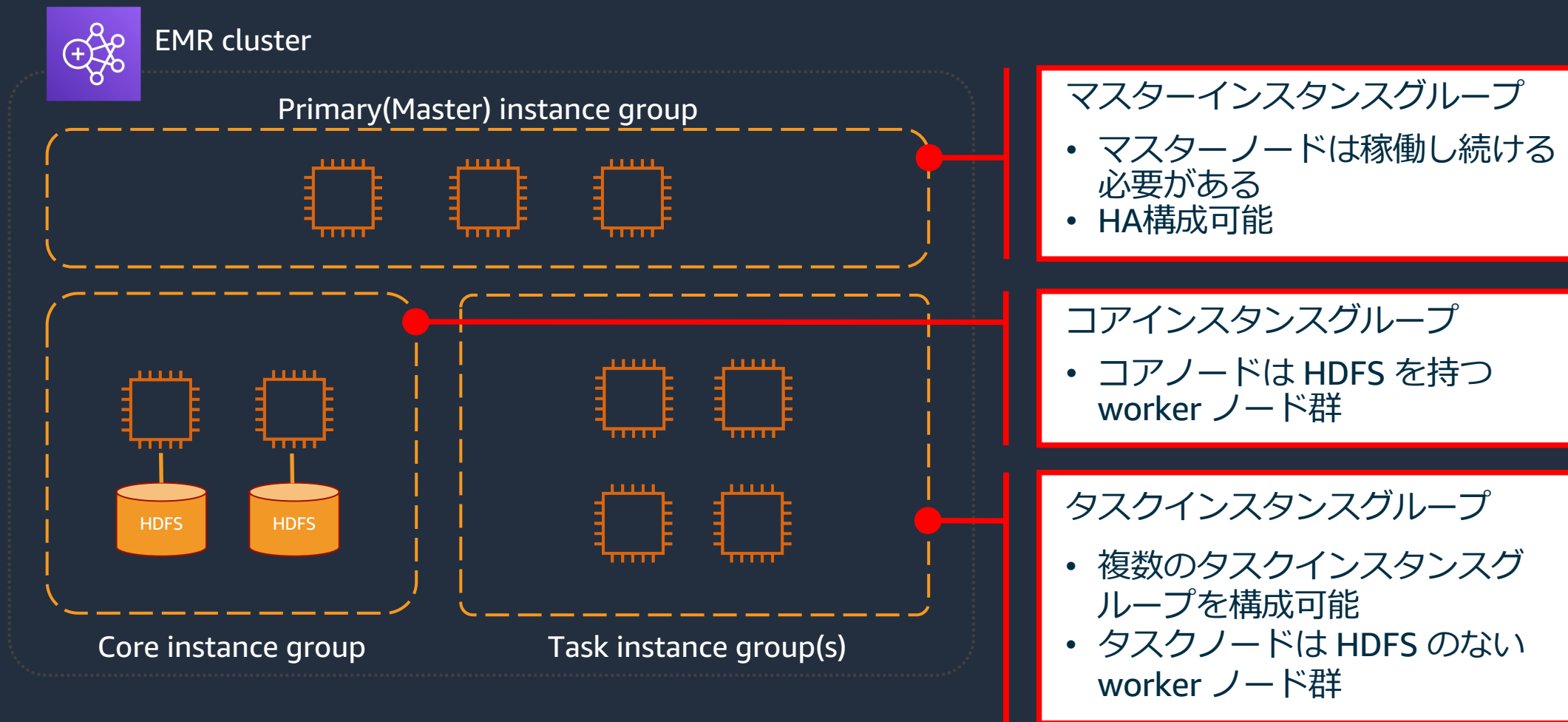
Hadoop クラスタ



EMR クラスタ



EMR クラスタ: 弾力性を実現するノード構成



マスターインスタンスグループ

- マスターノードは稼働し続ける必要がある
- HA構成可能

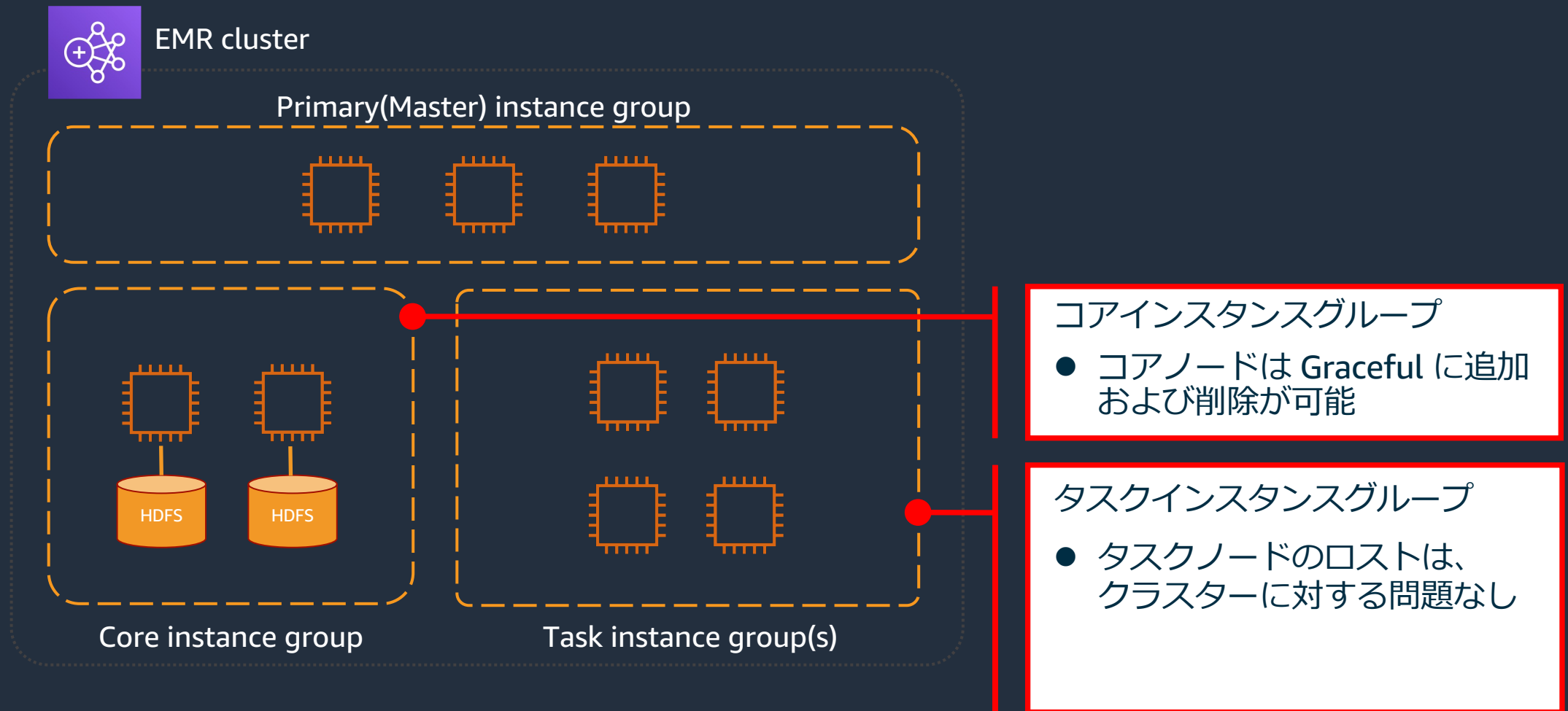
コアインスタンスグループ

- コアノードはHDFSを持つ worker ノード群

タスクインスタンスグループ

- 複数のタスクインスタンスグループを構成可能
- タスクノードはHDFSのない worker ノード群

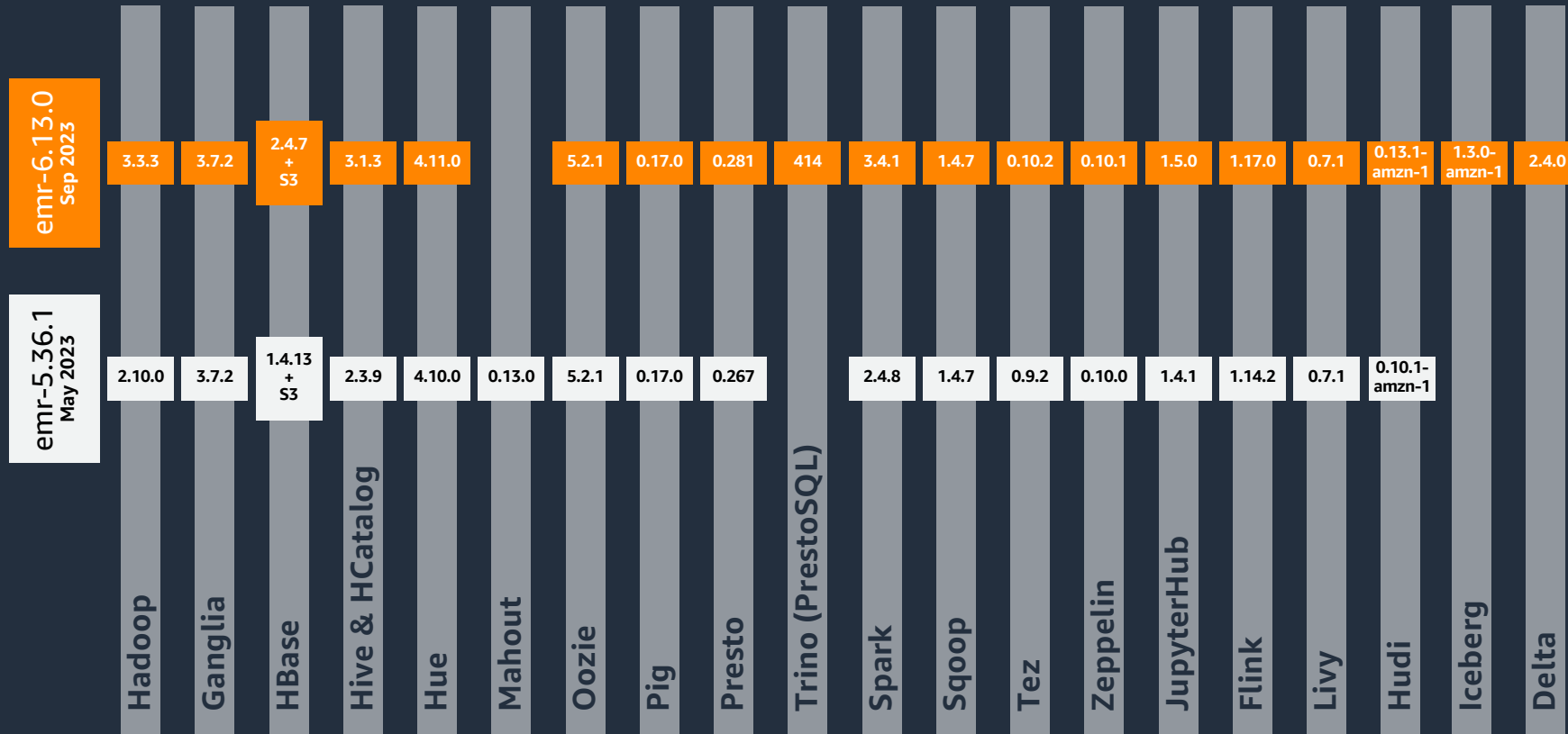
EMR クラスタ: 弾力性を実現するノード構成



EMR サポートアプリケーション

Hadoop と Spark のエコシステムに最新リリースをデプロイ

EMR Releases

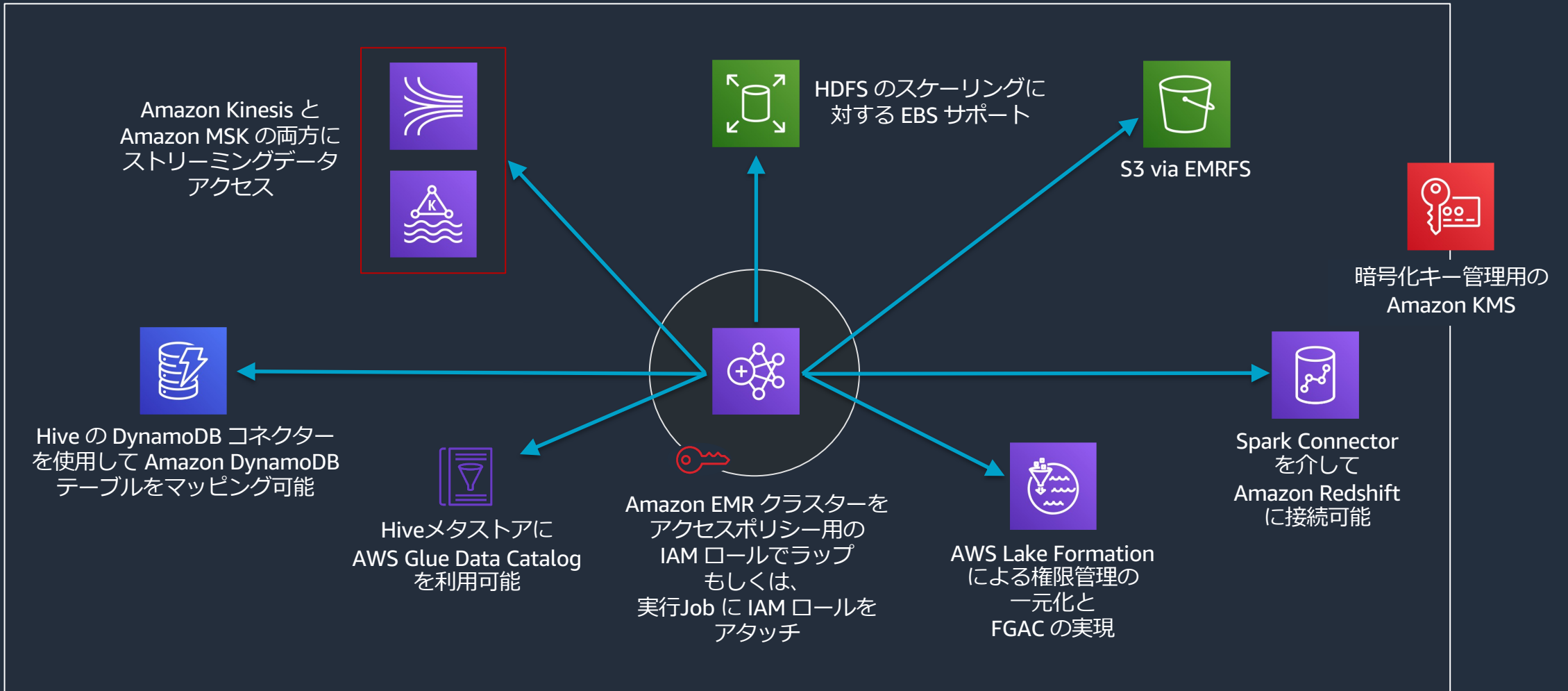


- 20 以上のオープンソースプロジェクトを利用可能: Apache Hadoop、Spark、HBase、Trino/Presto、Hudi、Iceberg など
- リリースから 30 日以内に最新のオープンソースフレームワークで更新

https://docs.aws.amazon.com/ja_jp/emr/latest/ReleaseGuide/emr-release-components.html



AWS サービスとのシームレスな統合

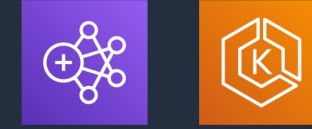


EMR Deployment Options



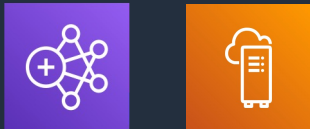
Amazon EMR on Amazon EC2

ワークロードに対して最高のコストパフォーマンスを発揮するインスタンスを選択可能



Amazon EMR on Amazon EKS

EKS での Apache Spark ジョブのプロビジョニング、管理、スケーリングを自動化



Amazon EMR on AWS Outposts

クラウドの場合と同様に、オンプレミス環境で EMR をセットアップ、管理、スケーリング可能



Amazon EMR Serverless

クラスターの管理や運用を行わずに、ペタバイト規模のデータ分析を実行可能

進化するプラットフォーム



最適化された Spark / Presto ランタイム

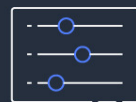
TPC-DS ベースの 3 TB ベンチマークの結果、標準の Apache Spark 3.0 よりも **3 倍** 以上高速

TPC-DS ベースの 3 TB ベンチマークの結果、標準の Presto 0.238 よりも **2.6 倍** 以上高速

オープンソース API に 100% 準拠しているため、アプリケーションを EMR に簡単に移行可能

パフォーマンス向上はデフォルトで有効

Dynamic-sized executors



Adaptive join selection



Dynamic pruning of data columns



Operator optimization



Early worker allocation



Intelligent filtering



Parallel/async initialization



Redundant scan elimination



Data pre-fetch



Broadcast join w/o statistics



Stats inference



Optimized metadata fetch



Amazon EMR runtime for Apache Hive

EMR 6.4 の Apache Hive 3.1.2 で 1.25 倍速いパフォーマンス

98 件クエリランタイムの幾何平均
(低い方がよりパフォーマンスが良い)



Apache Hive 3.1.2 on
EMR 6.4 vs. EMR 6.3*

**Based on TPC-DS 3 TB benchmarking running
16 node M5.8xlarge cluster*

パフォーマンスが最適化された Apache Hive ランタイム
を利用可能

ベストパフォーマンス

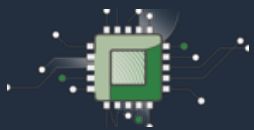
- 幾何平均で1.25倍高速
- 個々のクエリが最大2倍向上
- AWS Glue データカタログのクエリプランニング時間を短縮
- Amazon S3 からの ORC データのクエリ実行時間を改善

オープンソースの Apache Hive API に 100% 準拠

ワークロードに最適なインスタンスタイプの利用

柔軟なコンピューティング能力

汎用



M6 Family
M5 Family
M4 Family

バッチ処理

コンピューート



P3 Family
P2 Family
G4 Family
G3 Family
C6 Family
C5 Family
C4 Family

高速コンピューティング
/機械学習

メモリ



R6 Family
R5 Family
R4 Family

対話型の分析

ストレージ



D2 Family
I3 Family

大容量の HDFS

AWS Graviton インスタンスの活用: AWS Graviton2

インスタンスファミリーの中で最高のパフォーマンスを提供

M5(Intel アーキテクチャ汎用) と M6g(Graviton2: Arm アーキテクチャ汎用)、2つのインスタンスファミリーで構成した EMR (5.30.1) クラスタをTPC-DS ベースの 3 TB ベンチマークで比較



12%~16% の
パフォーマンス向上



20% コスト削減



最大 30% のコスト
パフォーマンス向上

Resource: <https://aws.amazon.com/jp/blogs/big-data/amazon-emr-now-provides-up-to-30-lower-cost-and-up-to-15-improved-performance-for-spark-workloads-on-graviton2-based-instances/>

AWS Graviton インスタンスの活用: AWS Graviton3

インスタンスファミリーの中で最高のパフォーマンスを提供

Graviton インスタンスファミリー C6g(Graviton2) と C7g(Graviton3)、それぞれで構成した EMR (6.9.0) クラスタをTPC-DS ベースの 3 TB ベンチマークで比較 (Spark/Trino)



13.65~18.73% の
パフォーマンス向上



7.93~13.35%
コスト削減



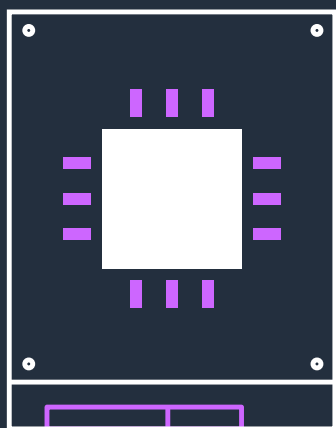
7-13% のコスト
パフォーマンス向上

Resource: <https://aws.amazon.com/jp/blogs/big-data/amazon-emr-launches-support-for-amazon-ec2-c7g-graviton3-instances-to-improve-cost-performance-for-spark-workloads-by-7-13/>



Amazon EBS GP3 の活用

コストパフォーマンスを最大化



gp3

General purpose SSD

リレーショナルおよび非リレーショナルデータベース、エンタープライズアプリケーション、コンテナ化されたワークロード、ビッグデータ、ファイルシステム、メディアワークフローに**最適なストレージ**

3,000 ベースライン IOPS (GP2 では 3 IOP/GiB)、および、**最大スループット 1,000 MiB/秒** (GP2 では 250 MiB/秒)で、容量とは別に IOPS とスループットをプロビジョニング可能

月額 **0.096 USD/GB***、以前の GP2 ボリュームよりも最大 **20% 低い**ストレージ価格

*東京リージョンでの価格

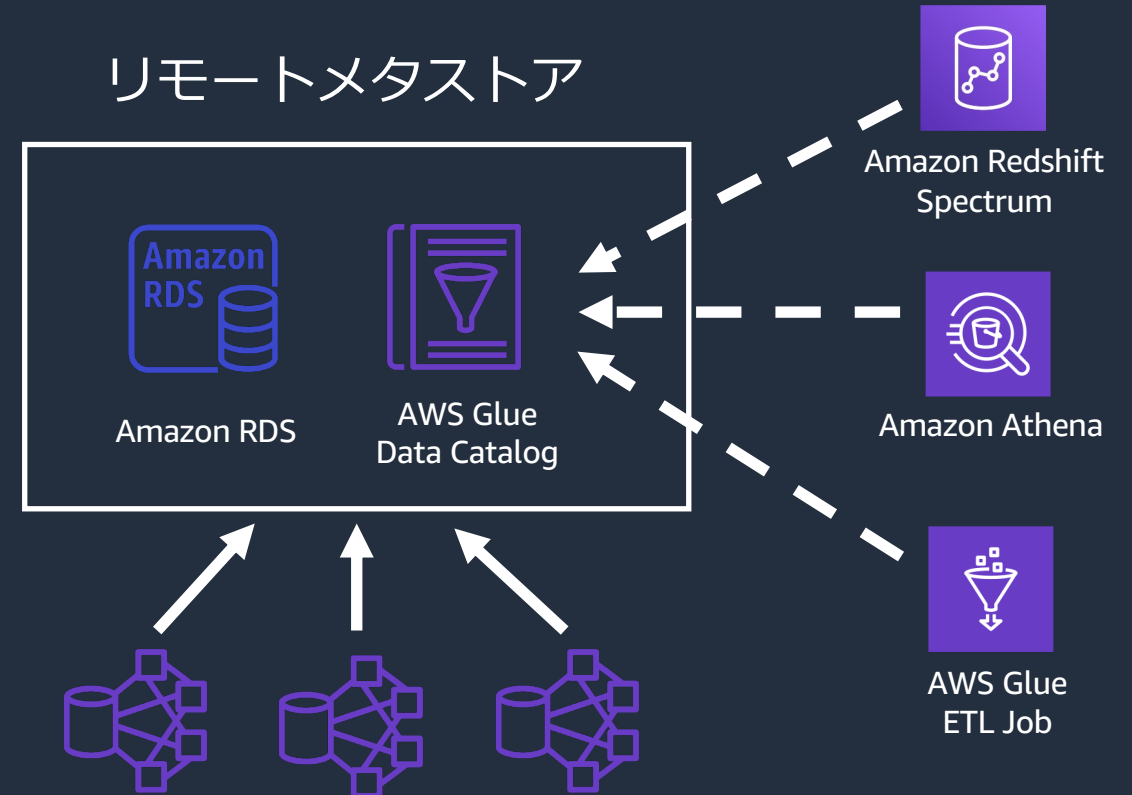
Resource: <https://aws.amazon.com/jp/about-aws/whats-new/2020/12/introducing-new-amazon-ebs-general-purpose-volumes-gp3/>

クラスターのステータス化

クラスタのステータス化：メタデータ

リモートメタストアの利用

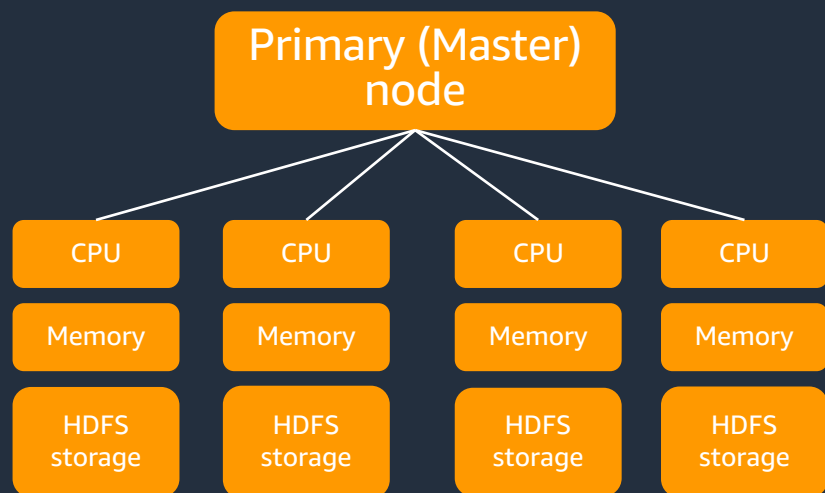
- メタストアをクラスタ外に保持
- 起動時間が短縮され、コストが削減される



クラスターのステータス化：ストレージ

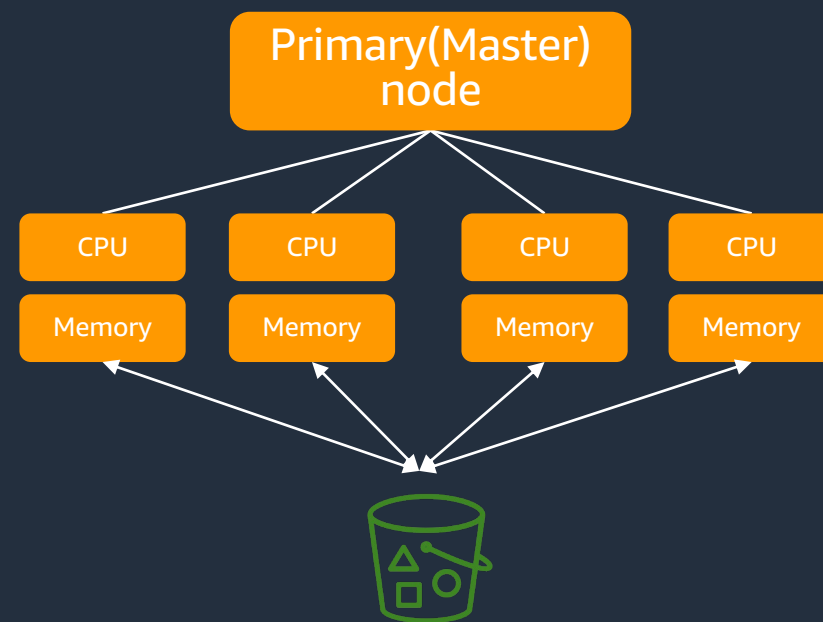
EMR File System (EMRFS) の利用

HDFS を利用する場合



- HDFS の基本戦略ではデータの冗長度は3
- ストレージコストがデータ量の3倍かかる

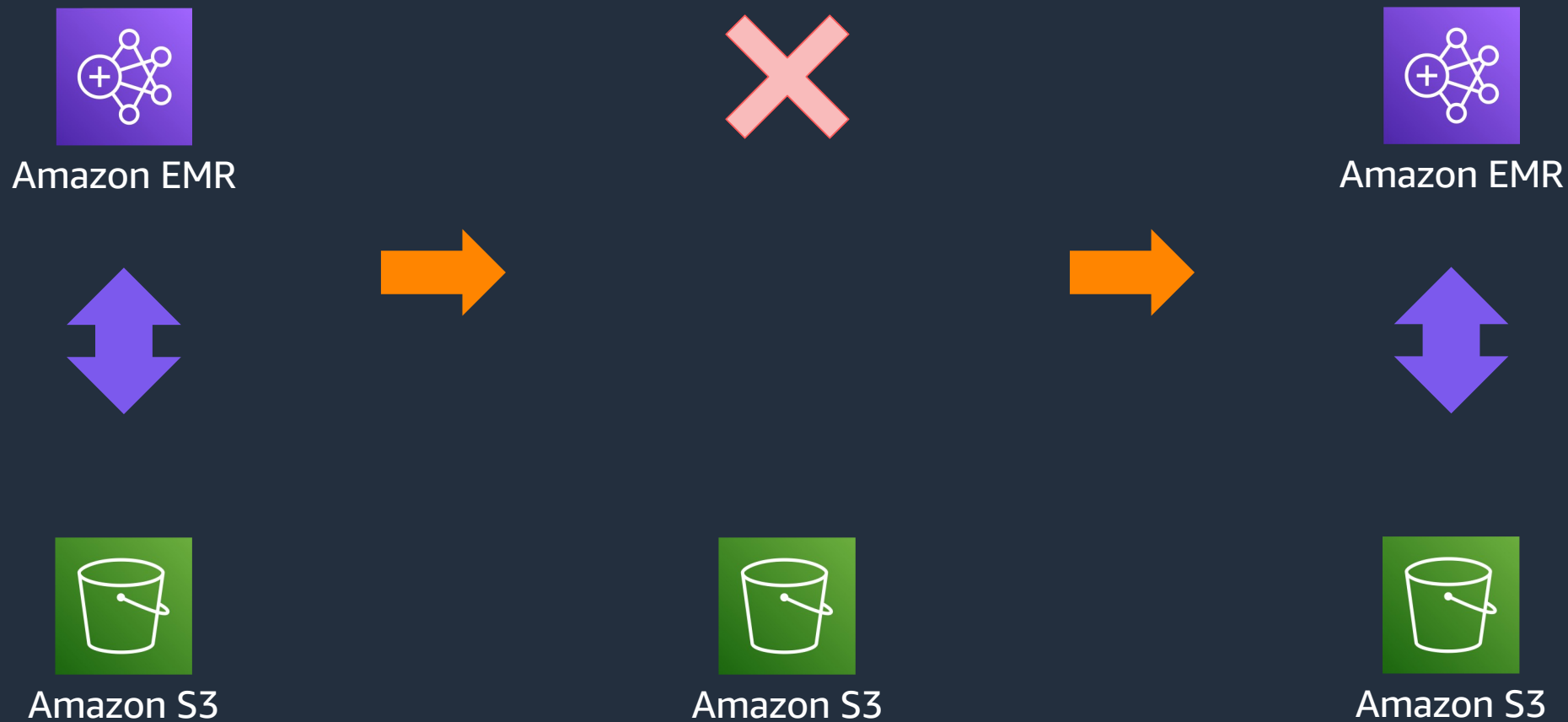
EMRFS を利用する場合



- EMRFS を利用すると HDFS のインターフェースから透過的に S3 上のデータを参照可能
- データの耐久性は S3 にオフロード

一時的なクラスター:

コンピューとストレージの分離で低コストを実現



Auto Scaling



マネージドスケールリング

クラスタサイズを調整してコストを自動的に削減



継続的に改善される
アルゴリズムで完全に
マネージド化された
エクスペリエンス
を提供



マネージドスケールリング
で高解像度メトリクス
を実現



最小/最大コスト
制約設定のみで
利用可能



Auto Scaling よりも
データポイントが多く、
反応時間が短い



コストを
20% ~ 60%
節約可能

マネージドスケーリング

クラスタサイズを調整してコストを自動的に削減

- 下記パラメータを設定するだけで利用可能

- 最小 - クラスタの最小ユニット数
- 最大 - クラスタの最大ユニット数
- オンデマンド制限 - オンデマンドユニット数の上限
- 最大コアノード数 - コアノードユニット数の上限

Core and task units	
Minimum:	<input type="text" value="5"/> <input type="button" value="↑"/> <input type="button" value="↓"/>
Maximum:	<input type="text" value="100"/> <input type="button" value="↑"/> <input type="button" value="↓"/>
On-demand limit :	<input type="text" value="10"/> <input type="button" value="↑"/> <input type="button" value="↓"/>
Maximum Core Node :	<input type="text" value="10"/> <input type="button" value="↑"/> <input type="button" value="↓"/>

- 制限事項

- Spark、Hadoop、Hive、Flinkなどの YARN アプリケーションでのみ機能
- EMR-6.0.0 を除く、EMR-5.30.0 以降の EMR で利用可能

<https://docs.aws.amazon.com/emr/latest/ManagementGuide/emr-managed-scaling.html>

強化されたマネージドスケールリング

コストとスポットの中断をさらに削減するための、デフォルトで有効な新しい機能強化



サポートされているすべての EMR バージョンで、インスタンスグループの容量認識(capacity awareness)がデフォルトで有効化

リアルタイムに EC2 スポットキャパシティメトリクスと統合され、インスタンスプールの深さに基づいて適切なタスクグループをスケールリングする



EMR 5.34 / EMR 6.4 から Apache Spark shuffle awareness がデフォルトで有効化

シャッフルデータがアクティブなノードをスケールダウンしない機能



EMR 6.12 から

Presto と Trino でマネージドスケールリングを高速に実現するための新しい再試行メカニズムを利用可能

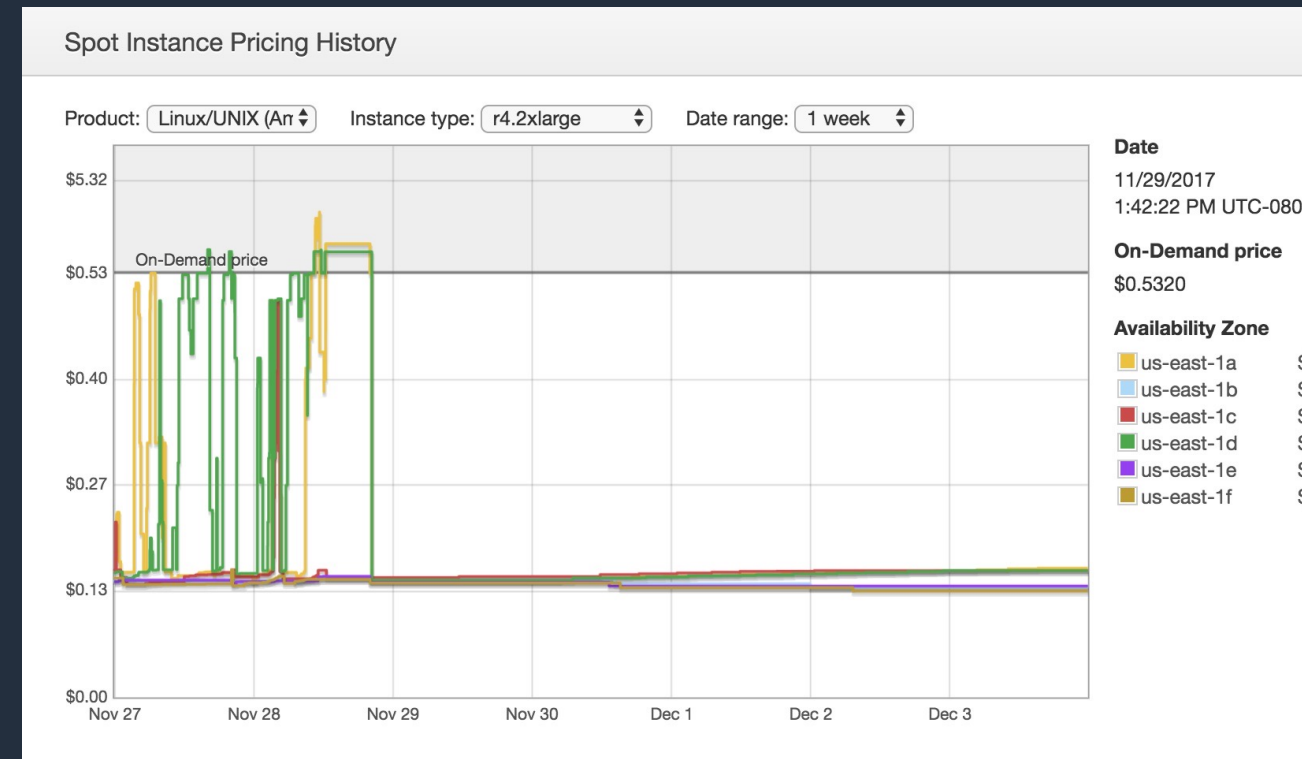
クラスターに設定された HDFS レプリケーション 係数を下回るコアノードがスケールダウンされないように

スポットインスタンスの活用

スポットインスタンス

より少ないコストでコンピューティングを加速

- Amazon EC2 の予備容量からオンデマンド(OD)価格の最大 90 %割引でインスタンスを利用可能
- 料金は長期的な需要と供給に基づいて決められ、スムーズに変化
- 中断は Amazon EC2 が OD 用のキャパシティを必要とする場合にのみ入札なしで発生



2017年11月
以前(入札有)

現在の
料金モデル

スポットインスタンスで得られる EMR のメリット

計算処理を加速



スポットインスタンスを利用するとオンデマンドインスタンスを利用する料金で、よりたくさんのインスタンスを実行可能

さらなるコスト削減



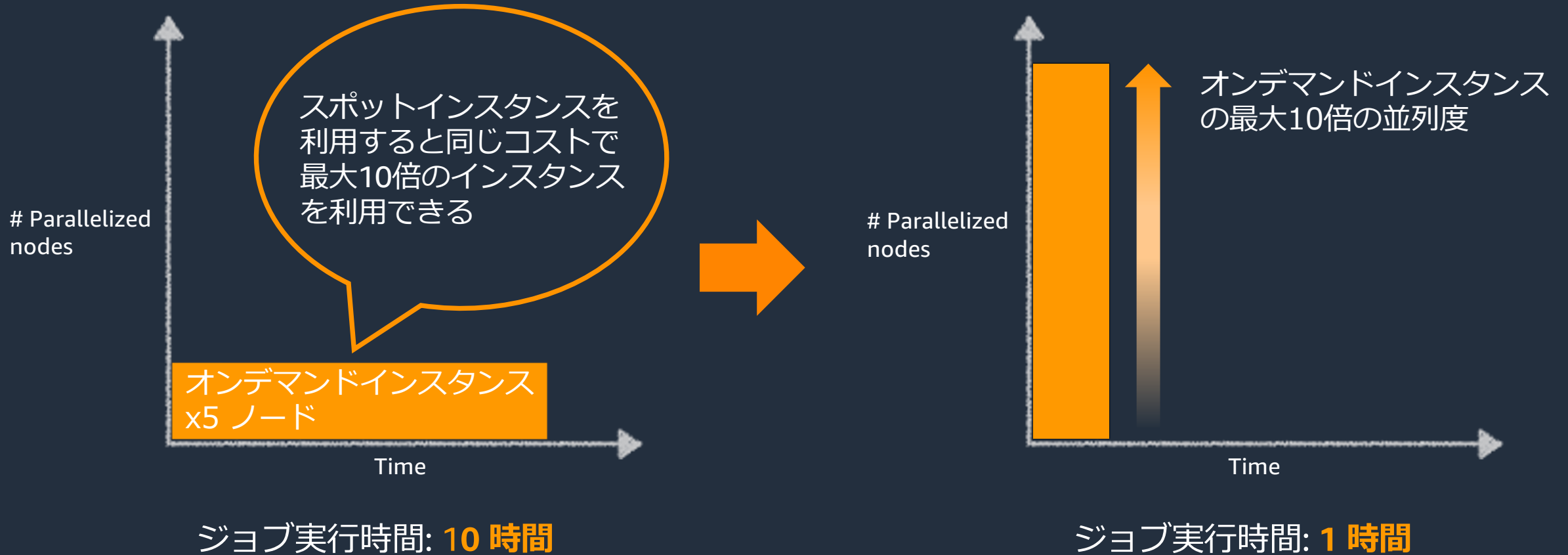
オンデマンド価格の最大 90% 割引でインスタンスを利用可能

スケールに合わせて構築



ワークロードの並列度を上げ、ジョブの実行時間を短縮できる

スポットインスタンス利用でワークロードの並列度を上げ、ジョブの実行時間を短縮できる



費用対効果の具体例



10 node cluster running for 14 hours
Cost = 1.0 * 10 * 14 = \$140

費用対効果の具体例



Add 10 more nodes on Spot

費用対効果の具体例



20 node cluster running for 7 hours

$\text{Cost1} = 1.0 * 10 * 7 = \70
(オンデマンド)

$\text{Cost2} = 0.5 * 10 * 7 = \35
(スポット)

$\text{Total} = \text{Cost1} + \text{Cost2} = \105

費用対効果の具体例



費用対効果:

50 % 処理時間削減 (14 → 7)

25% コスト削減 (140 → 105)

オンデマンドインスタンスとスポットインスタンスで柔軟なサービスレベルを定義できる

予測可能なコストでサービスレベルにあったクラスターを構築可能

コアノードにオンデマンドインスタンスを利用

EC2 の標準利用料でコストを計算可能



サービスレベルを超える予測不可能なワークロードに対するリソースコストをスポットインスタンスで極小化可能

タスクノードにスポットインスタンスを利用

オンデマンド価格の最大 90% 割引価格

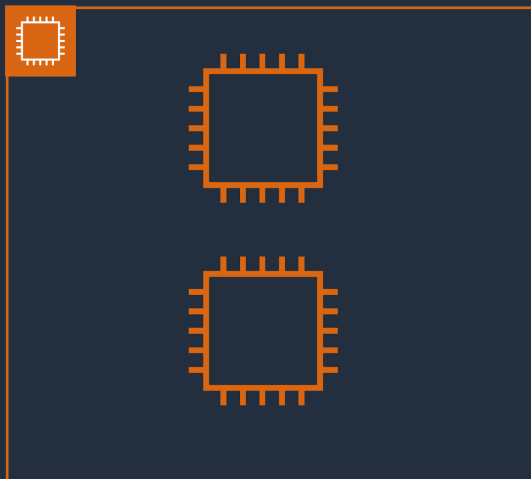
インスタンスフリート



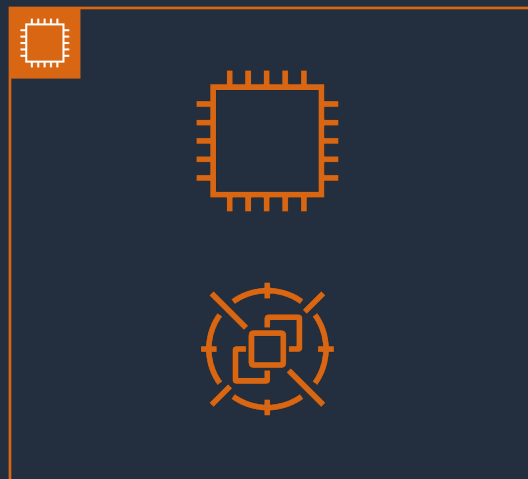
インスタンスフリート

EMR でスポットインスタンスを使いこなすための機能

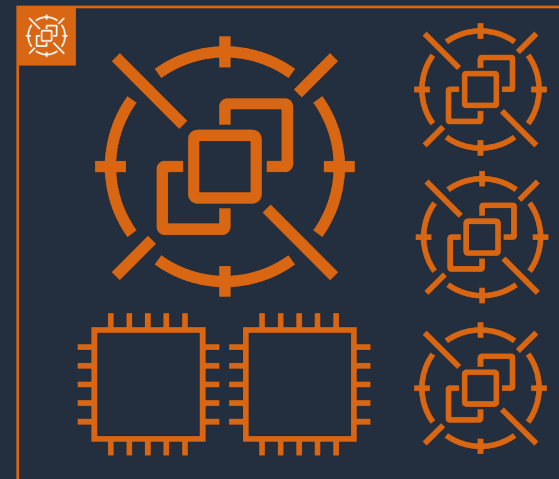
Primary(Master) nodes



Core instances

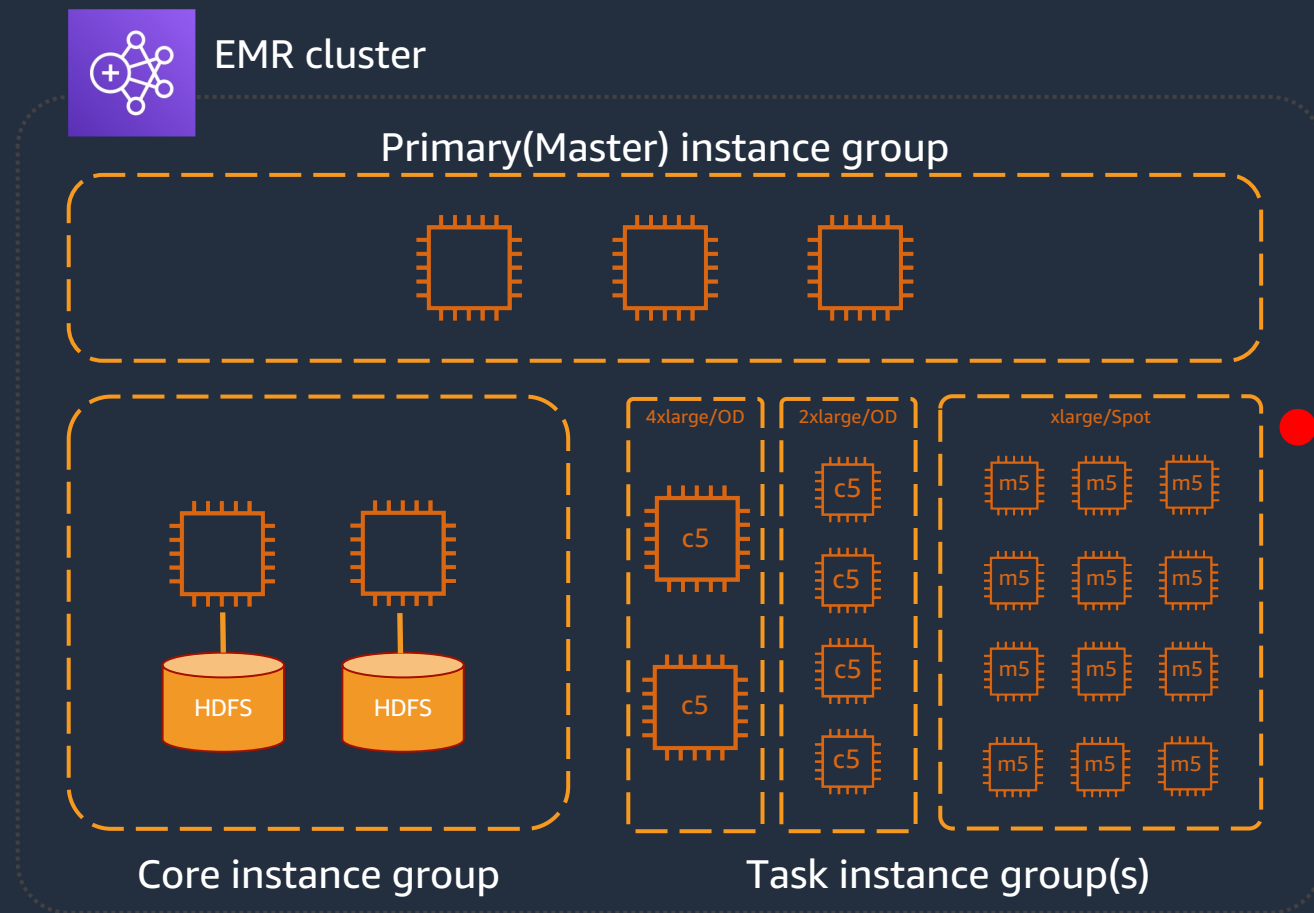


Task instances



- 指定した複数のインスタンスタイプリストから、スポット / オンデマンドでインスタンスをプロビジョニング可能とする機能
- 容量/価格に基づいて最適なアベイラビリティゾーンを EMR が選択

(参考) インスタンスグループ



グループ毎に、異なるインスタンス
タイプやオンデマンド/スポットを
指定することができる(最大48個の
Task インスタンスグループを作成
可能)

EMR のインスタンスフリートに最適な スポットインスタンス



- ✓ オンデマンドインスタンスとスポットインスタンスが混在するようにインスタンスフリートを設定可能
- ✓ 最も低い価格で最大容量のインスタンスに最適化されます
- ✓ スポットインスタンスが再利用(中断)されると、フリート内の別のインスタンスによって置き換えられます

スポットインスタンスの中断発生率を抑える機能 (Allocation Strategy) が利用可能に

- ✓ Allocation Strategy を利用すると、タスクインスタンスフリートに指定可能なインスタンスタイプ数が5種類から**15種類**に拡張され、インスタンスの確保がより容易に
- ✓ さらに、API/CLI からの実行で、各種フリートに最大**30種類**のインスタンスタイプ が設定可能
- ✓ 起動するインスタンスの数に対して**最適な容量を持つ (確保が容易/中断が発生しにくい) スポットインスタンス**を起動

<https://aws.amazon.com/jp/about-aws/whats-new/2021/06/amazon-emr-now-supports-up-to-thirty-instance-type-configurations-in-instance-fleets/>

Spot Instance Advisor の活用

Spot Instance Advisor

Region: OS:

Instance type filter:

vCPU (min): Memory GiB (min):

Instance types supported by EMR

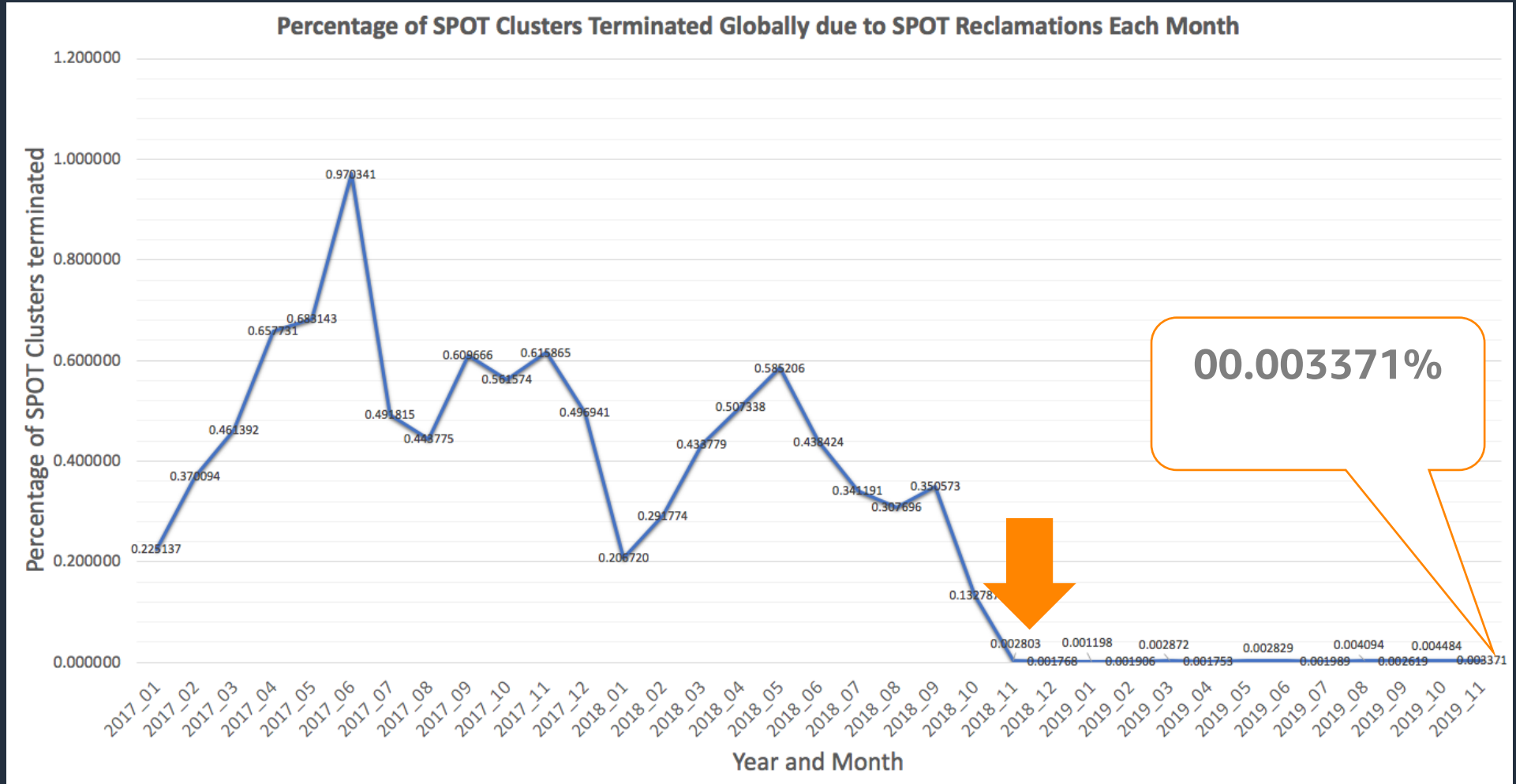
Instance Type	vCPU	Memory GiB	Savings over On-Demand*	Frequency of interruption ▾
i3.metal	72	512	70%	<5% □□□□□
r4.16xlarge	64	488	78%	<5% □□□□□
c5n.xlarge	4	10.5	76%	<5% □□□□□
m5d.24xlarge	96	384	76%	<5% □□□□□
m1.large	2	7.5	90%	<5% □□□□□

Amazon EC2 スポット配置スコアの活用

The screenshot shows the Amazon EC2 Spot Placement Score tool interface. On the left is a navigation sidebar with categories like EC2 Dashboard, Instances, Images, Elastic Block Store, and Network & Security. The main content area is titled "Spot placement score" and includes a breadcrumb "EC2 > Spot requests > Spot placement score". Below the title is a description: "Spot placement score helps you to select optimal Regions or Availability Zones to run workloads that can use multiple instance types." The interface is divided into three main sections: 1. "Target capacity and instance type requirements" with an "Enter requirements" button. 2. "Placement scores" with a "Calculate placement scores" button and explanatory text. 3. A configuration section for "Regions to evaluate" with a dropdown menu, a "Clear filters" button, and a checkbox for "Provide placement scores per Availability Zone". At the bottom, there is a table header with "Region" and "Placement score" columns, and a message stating that scores will appear after requirements are entered.



スポットインスタンス中断率の実際



さまざまなシナリオでオンデマンドインスタンスとスポットインスタンスを組み合わせる

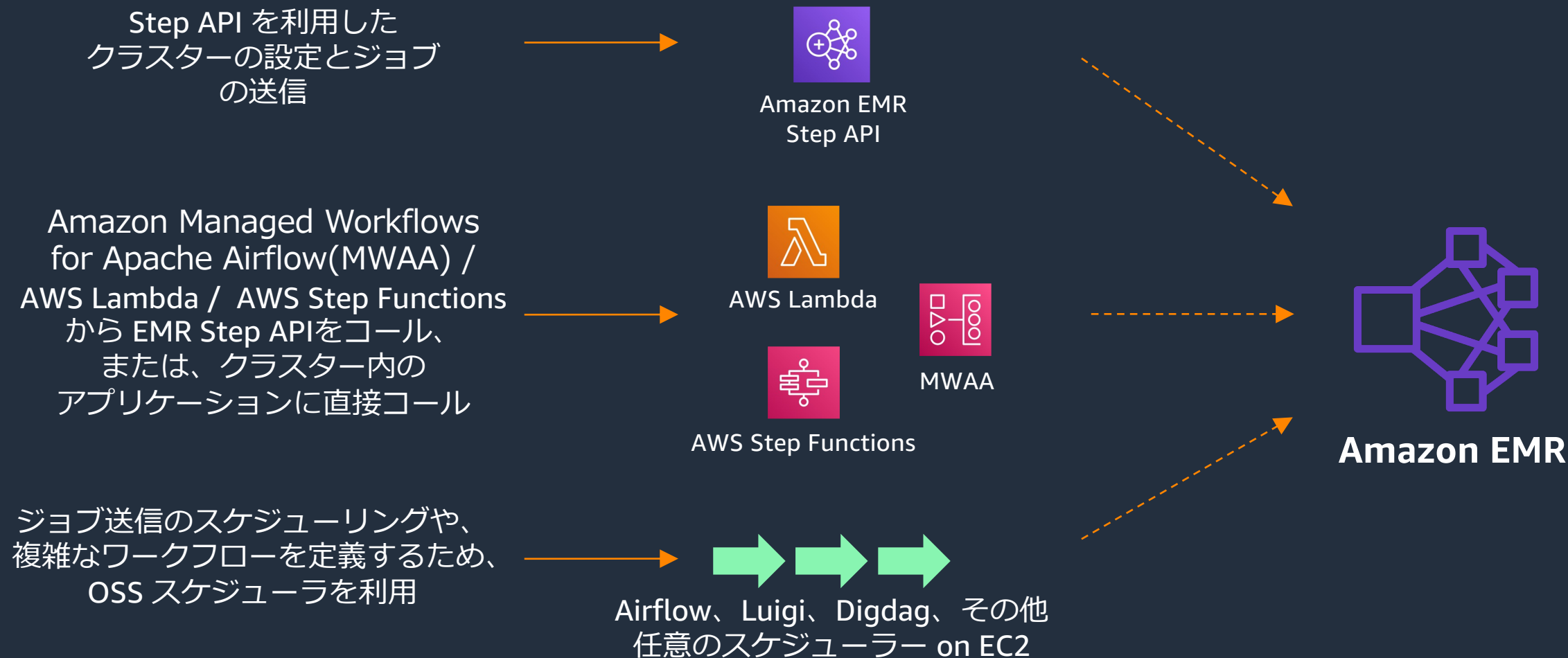
Scenario	Primary(Master) node	Core nodes	Task nodes
長時間稼働クラスターとデータウェアハウス	On-demand	On-demand or instance-fleet mix	Spot or instance-fleet mix
コスト重視ワークロード	Spot	Spot	Spot
データクリティカルなワークロード	On-demand	On-demand	Spot or instance-fleet mix
アプリケーションテスト	Spot	Spot	Spot

Ref: ガイドラインとベストプラクティス

<https://docs.aws.amazon.com/emr/latest/ManagementGuide/emr-plan-instances-guidelines.html>

クラスター運用の自動化

クラスター運用自動化の選択肢



セキュリティ



包括的なセキュリティ機能



アイソレーション



認証



認可



暗号化



監査

VPC

Private subnets

Security groups

LDAP

Kerberos

AWS SSO
(EMR Studio)

AWS IAM
(EMR Studio)

Cluster IAM role

User execution role

FGAC using Apache
Ranger

FGAC using
AWS Lake Formation
for Apache Spark
/Apache Hive

Encryption at rest

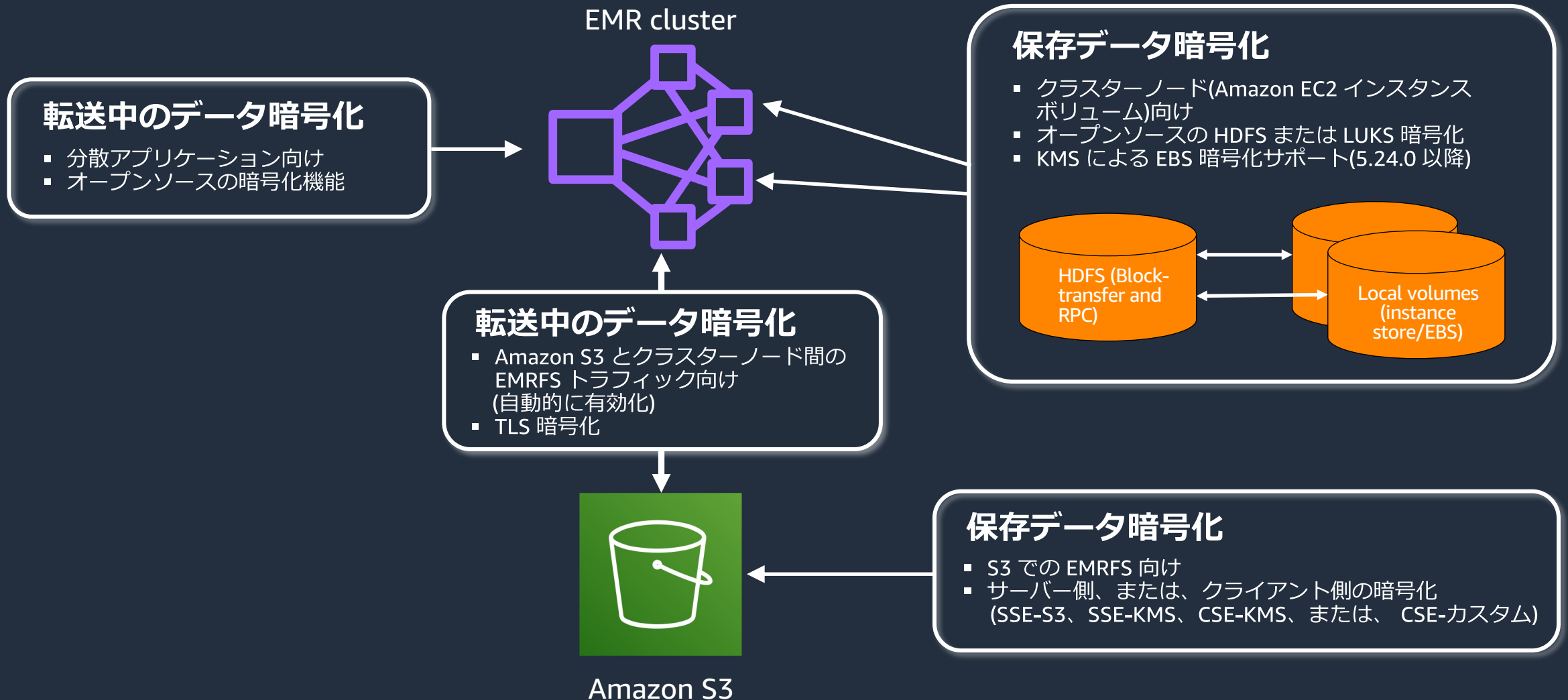
Encryption in
transit

Key management

Audit using Apache
Ranger via
Amazon CloudWatch
Logs

Audit using
AWS Lake Formation
via
AWS CloudTrail

EMR での転送中および保存中の暗号化



包括的なセキュリティ機能



アイソレーション



認証



認可



暗号化



監査

VPC

Private subnets

Security groups

LDAP

Kerberos

AWS SSO
(EMR Studio)

AWS IAM
(EMR Studio)

Cluster IAM role

User execution role

FGAC using Apache
Ranger

FGAC using
AWS Lake Formation
for Apache Spark
/Apache Hive

Encryption at rest

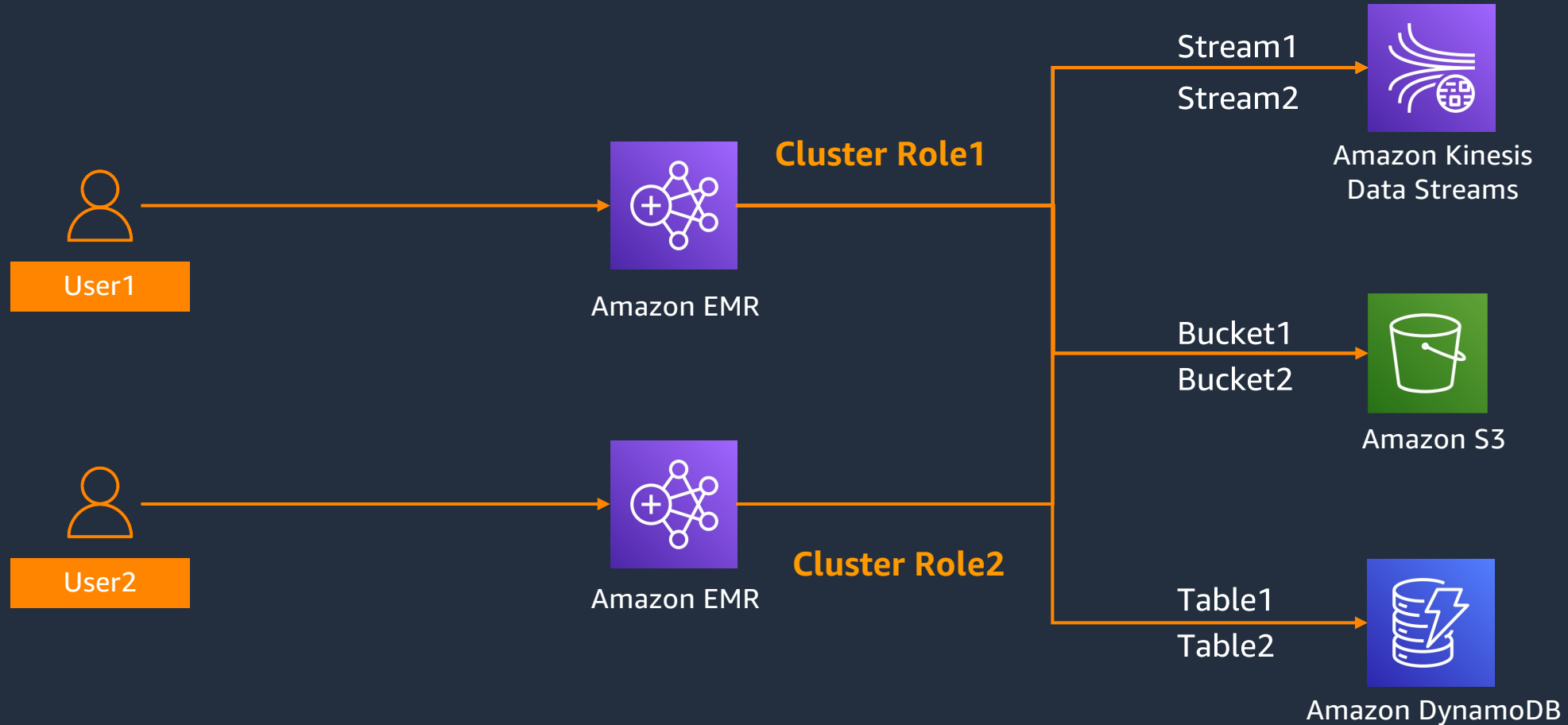
Encryption in
transit

Key management

Audit using Apache
Ranger via
Amazon CloudWatch
Logs

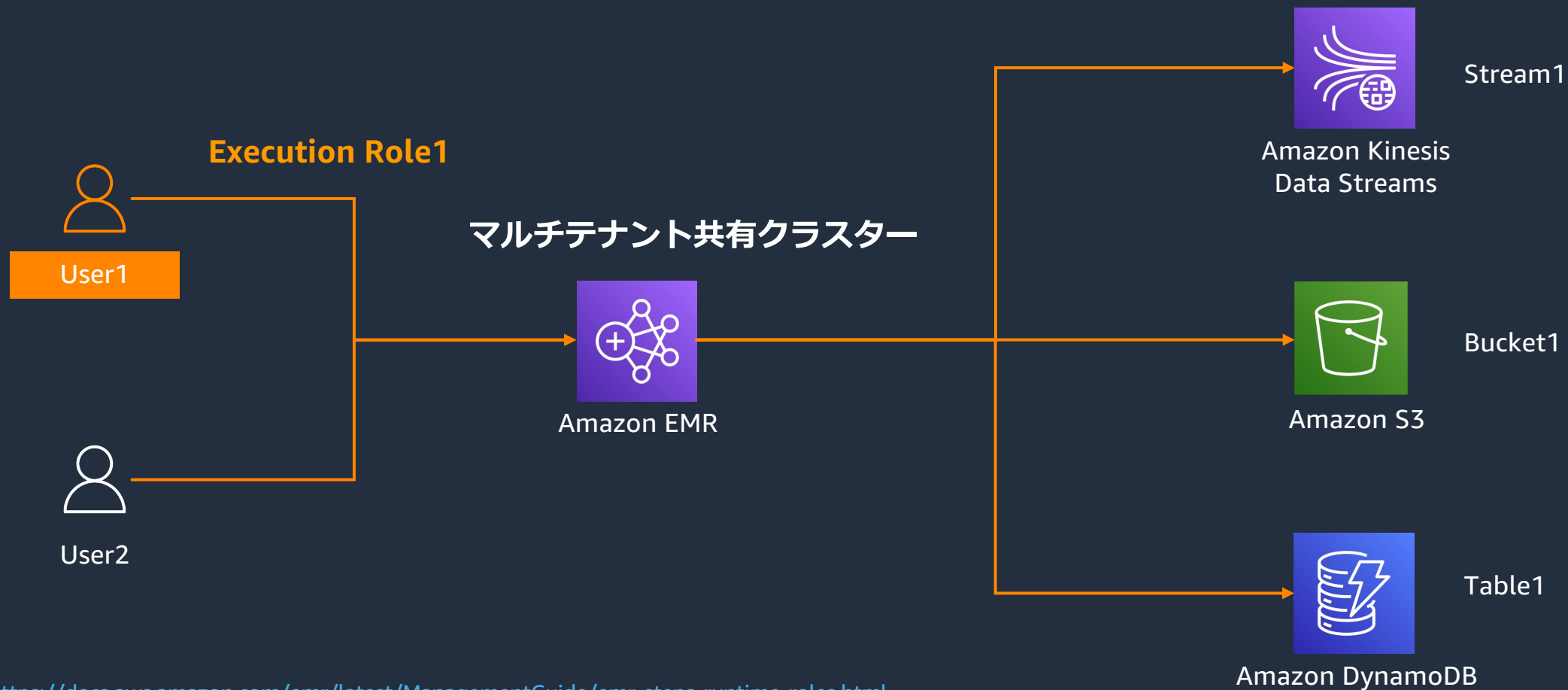
Audit using
AWS Lake Formation
via
AWS CloudTrail

既存の権限管理 (1 クラスター に 1 ロール)



マルチテナント共有クラスターの実現

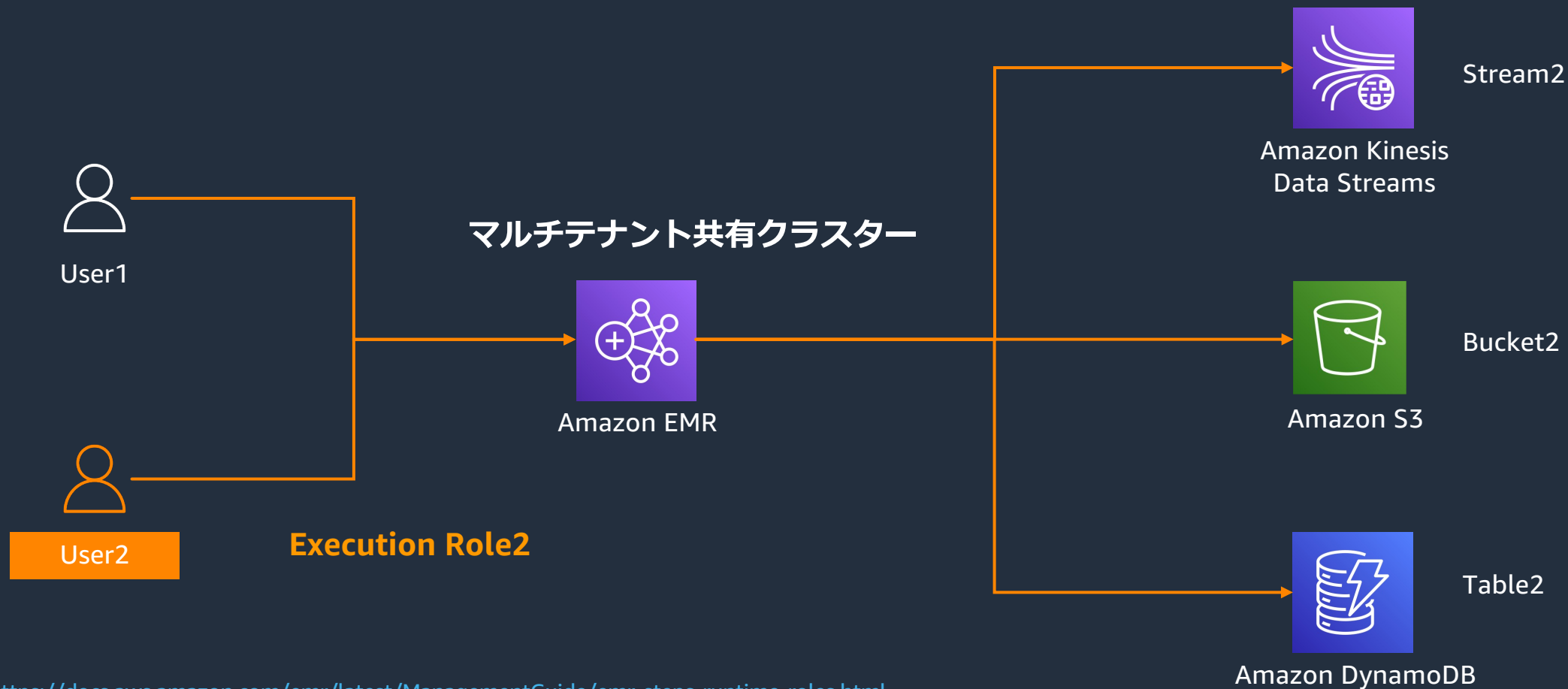
User Execution Role: User1 はリソース Stream1, Bucket1, and Table1 にアクセス可能



<https://docs.aws.amazon.com/emr/latest/ManagementGuide/emr-steps-runtime-roles.html>

マルチテナント共有クラスターの実現

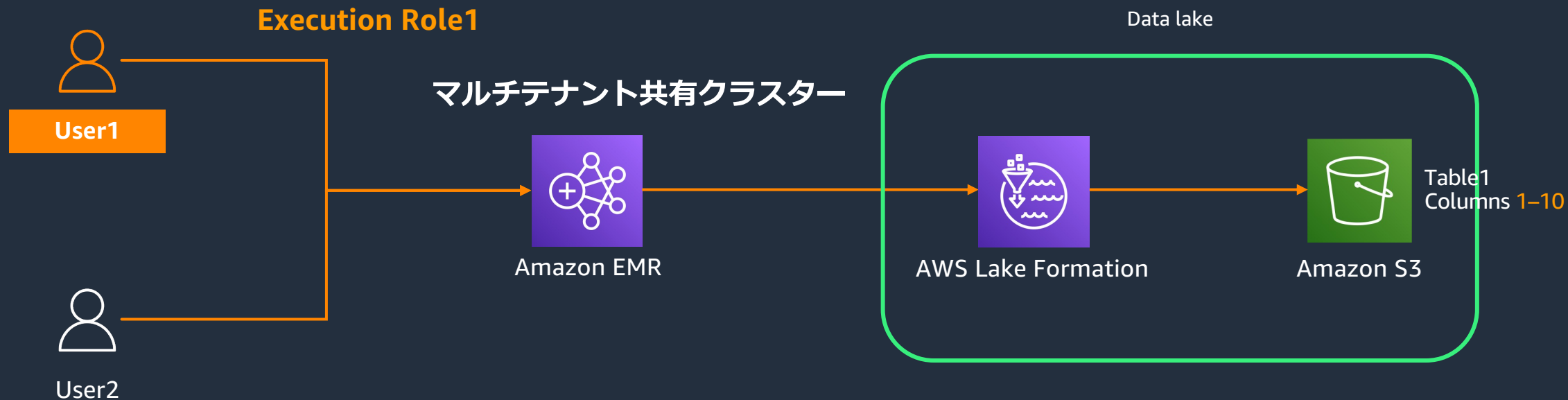
User Execution Role: User2 はリソース Stream2, Bucket2, and Table2 にアクセス可能



<https://docs.aws.amazon.com/emr/latest/ManagementGuide/emr-steps-runtime-roles.html>

きめ細かなアクセス制御(FGAC)の実現

FGAC using AWS Lake Formation: User1 は **Table1** の **Columns 1-10** にアクセス可能

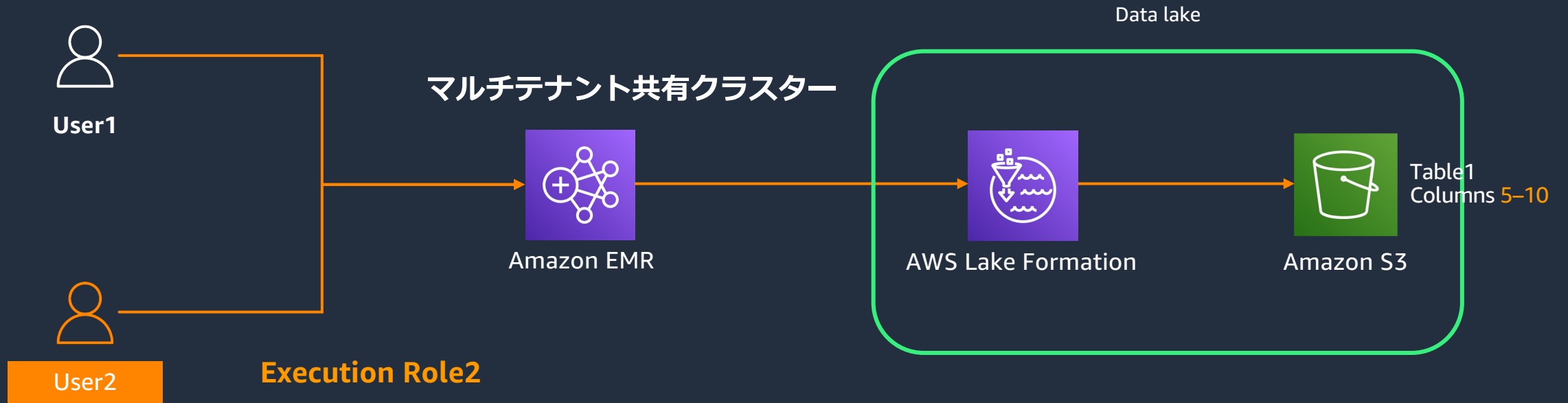


<https://docs.aws.amazon.com/emr/latest/ManagementGuide/emr-steps-runtime-roles.html>



きめ細かなアクセス制御(FGAC)の実現

FGAC using AWS Lake Formation: User2 は **Table1** の **Columns 5-10** にアクセス可能



<https://docs.aws.amazon.com/emr/latest/ManagementGuide/emr-steps-runtime-roles.html>

料金



EMR on EC2 の利用料金

- Amazon EC2(以下、EC2)インスタンスタイプ毎に決められた Amazon EMR と EC2 インスタンスの利用料金(いずれも 1 秒ごとに課金、最小課金時間は 1 分)
- アタッチされた Amazon Elastic Block Store (Amazon EBS) で発生する料金 (ストレージ、IOPS、スループット)
- Amazon S3 や AWS Glue Data Catalog、AWS Key Management Serviceなど、お使いのアプリケーションが AWS の他のサービスを使用する場合、状況により追加料金が発生

※ EC2 料金にはオンデマンド、1 年間および 3 年間のリザーブドインスタンス、Capacity Savings Plans、スポットインスタンスなど、さまざまなオプションが用意されています

<https://aws.amazon.com/jp/emr/pricing/>



事例紹介



多くの Hadoop / Spark プロジェクトを支えています



NETFLIX



REDFIN



HARRY'S



オンプレミス Hadoop から Amazon EMR への移行事例



1日あたり135億件のイベントを処理し、コストを 60% 削減 (約20万ドル)



5ヶ月未満でコストを600万ドル削減



運用コスト削減 + Sparkのパフォーマンスの向上 3倍

Aol.

75% の節約、60% を超えるの効率化



1つのモノリシックパイプラインを 3つの目的に構築されたクラスタに再設計

Amazon EMR の影響が大きい事例



1億4千万人のプレイヤーに対するニア・リアルタイム分析



日常的に3,000の一時クラスタを拡張



1日あたり1,000,000回のデータ実行を処理するPredixソリューションを強化



オンデマンド料金と比較して55%のコスト削減、リザーブドインスタンスと比較して40%のコスト削減を実現



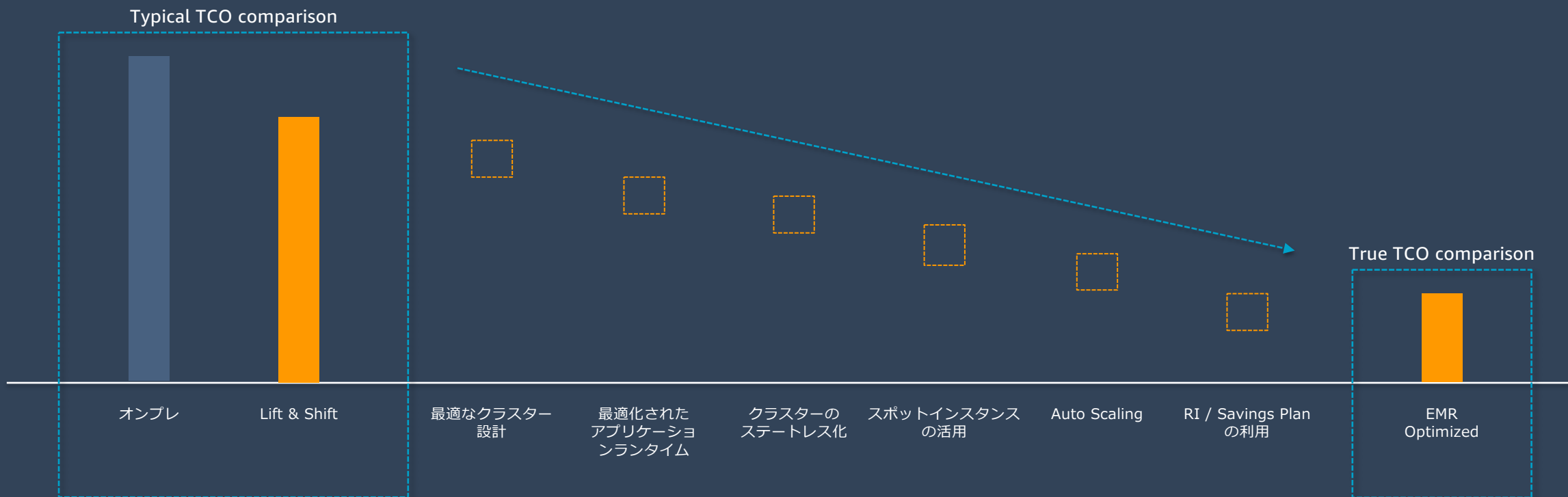
1億件を超える物件の査定に1日かかっていた計算処理が数時間で計算できるように

まとめ



まとめ

EMR はワークロードに対し、クラスターのコストパフォーマンス最適化を実現するための多くのオプションを提供し、進化を継続中

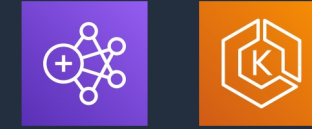


EMR Deployment Options



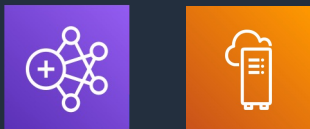
Amazon EMR on Amazon EC2

ワークロードに対して最高のコストパフォーマンスを発揮するインスタンスを選択可能



Amazon EMR on Amazon EKS

EKS での Apache Spark ジョブのプロビジョニング、管理、スケーリングを自動化



Amazon EMR on AWS Outposts

クラウドの場合と同様に、オンプレミス環境で EMR をセットアップ、管理、スケーリング可能



Amazon EMR Serverless

クラスターの管理や運用を行わずに、ペタバイト規模のデータ分析を実行可能

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おります。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)



Thank you!