



Amazon OpenSearch Serverless のご紹介

Shuhei Fukami

Solutions Architect

Amazon Web Service Japan

自己紹介

深見修平(Shuheï Fukami)

アマゾン ウェブ サービス ジャパン 合同会社

Solutions Architect

ISV/SaaS テリトリのお客様を中心に技術支援を担当

好きなAWSサービス :

Amazon OpenSearch Service

Amazon Redshift Serverless

AWS Step Functions



アジェンダ

- OpenSearch とは
- Amazon OpenSearch Serverless の概要
- デモ
- まとめ

OpenSearch とは



コミュニティ主導・Apache 2.0 ライセンスのオープンソース検索・分析スイート

データストア、検索エンジンの **OpenSearch**、可視化、UI ツールの **OpenSearch Dashboards** から構成されている

セキュリティ、パフォーマンス分析、機械学習など様々なプラグインによる機能拡張が可能

ユースケース1: 全文検索



半構造化データ、非構造化データの様々な側面や属性から、最適な製品、サービス、ドキュメント、回答を素早く発見する



コスト、セキュリティ、規模の要件に応じて、関連する検索結果をリアルタイムで取得する

INDUSTRY USE CASES



e コマースプラットフォーム:
適切な製品をすばやく見つける



**ドキュメントポータル (科学
研究記事、投資分析、または
診療録(カルテ)):** スピーディー
かつ関連性の高い
ドキュメント検索体験



**レコメンデーションエンジン
(ウィークリープレイリスト、
レシピ):** パーソナライズ
されたレコメンデーションを
提供することで、ユーザー
エンゲージメントを高める



**プラットフォーム検索
サービス:** 機械学習機能を
備えた、使いやすく
スピーディな検索体験

ユースケース2: ログ分析



インフラストラクチャと AWS サービスの問題を特定、診断、修正。製品の遅延と安定性を改善



大量のストリーミング データから、セキュアかつ費用対効果の高い方法で洞察を得る

INDUSTRY USE CASES



アプリケーション監視:
インフラストラクチャは機能しているか？
レイテンシとエラー率は？
アプリケーションの問題の原因は？



セキュリティ監視: 疑わしい認証アクティビティはないか？
この IP アドレスによってどのデータにアクセスされたか？
侵害の事実はあるか？



ビジネスインサイト:
ユーザーが興味を持っているコンテンツ/製品は何か？
最も使用されている機能と使用されていない機能はどれか？
最もアクティブなユーザーとその理由は？



オブザーバビリティ:
どのサービスで問題が発生しているか？
リクエスト処理の遅延はどこで発生しているか？

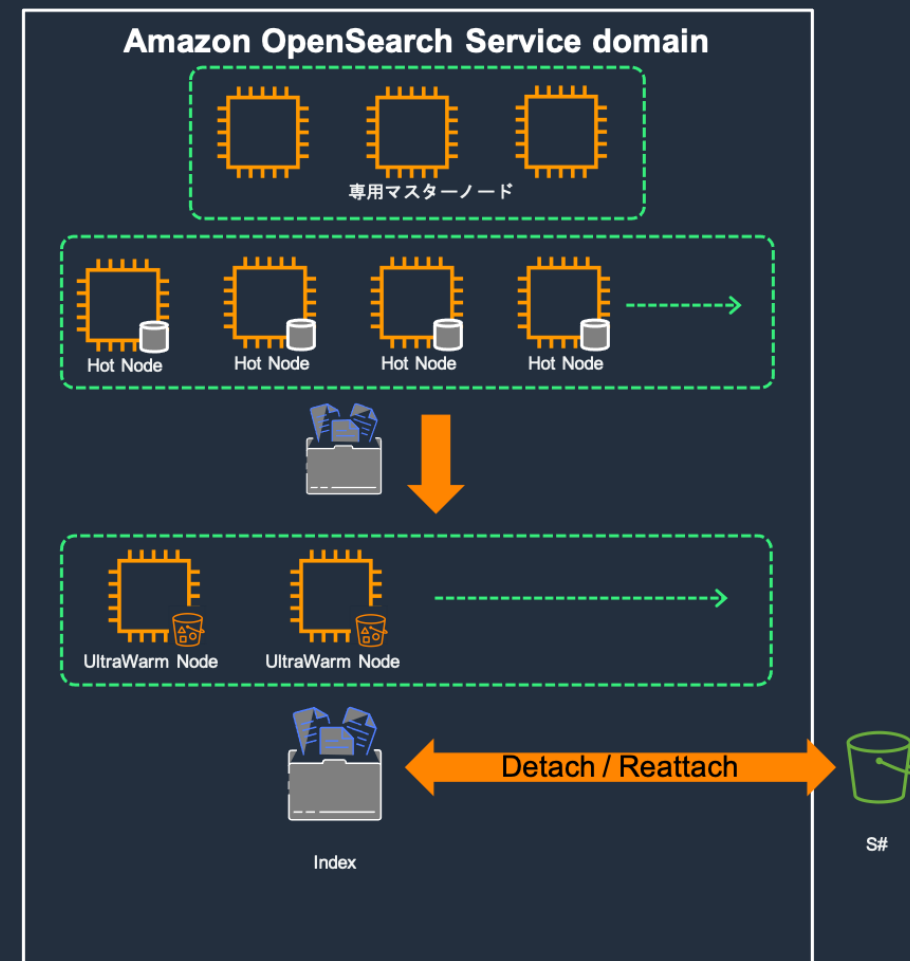
従来型アーキテクチャの課題

設計時の課題

- データサイズ、シャード数、リクエストレートなどをベースにクラスターのサイジングやインデックスの設計が必要

運用中の課題

- ピークトラフィックを無理なく捌くためには事前のスケールが必要
- スケールは Blue/Green デプロイメントを伴うため実行コストが高い
- セキュリティ対応やバグフィックスのために、不定期に配布されるソフトウェアアップデートの適用が必要
- ユースケースによってはインデックスのライフサイクルの管理が必要



Amazon OpenSearch Serverless





Amazon OpenSearch Serverless

クラスターの管理なしに検索と分析ワークロードを実行



管理が容易

クラスターのサイジング、スケーリング、チューニング、シャードとインデックスのライフサイクル管理が不要に



速度

リソースを自動的にスケールし、高速なデータ取り込みレートとクエリ応答時間を一貫して維持



エコシステム

既存の OpenSearch クライアント、パイプライン、API を使用して数秒で利用を開始できる

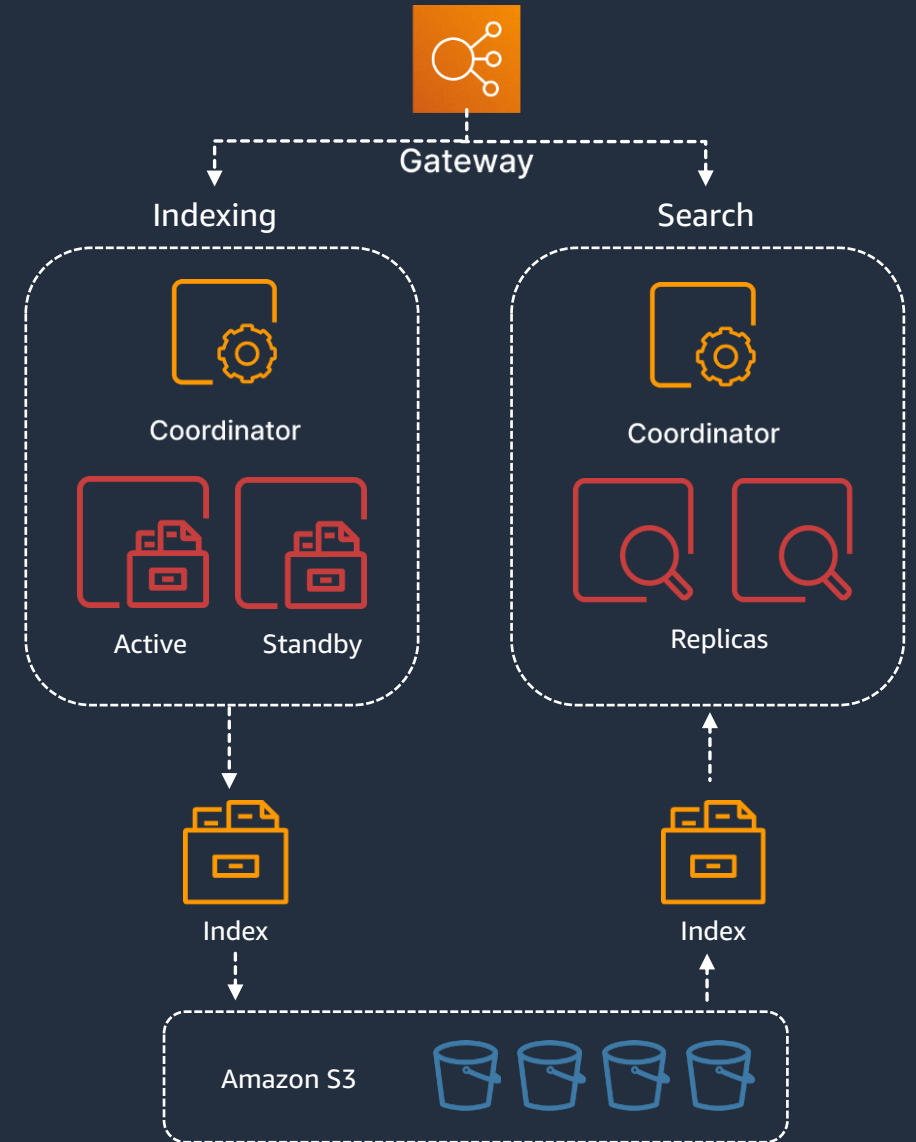


費用対効果

事前のリソースプロビジョニングは不要

Serverless アーキテクチャによる運用の改善

- インデキシング、検索、ストレージワークロードを分離。インデキシングと検索ノードは複数の Availability Zone に分散配置され、トラフィックに応じて個別にスケールする。
- 実データは S3 に格納されるが、検索パフォーマンスを高めるためにノードのボリュームにもキャッシュされる
- インデックス内のデータのライフサイクルは、サービス側で自動的に管理されている



Amazon OpenSearch Service と同様の方法で利用可能

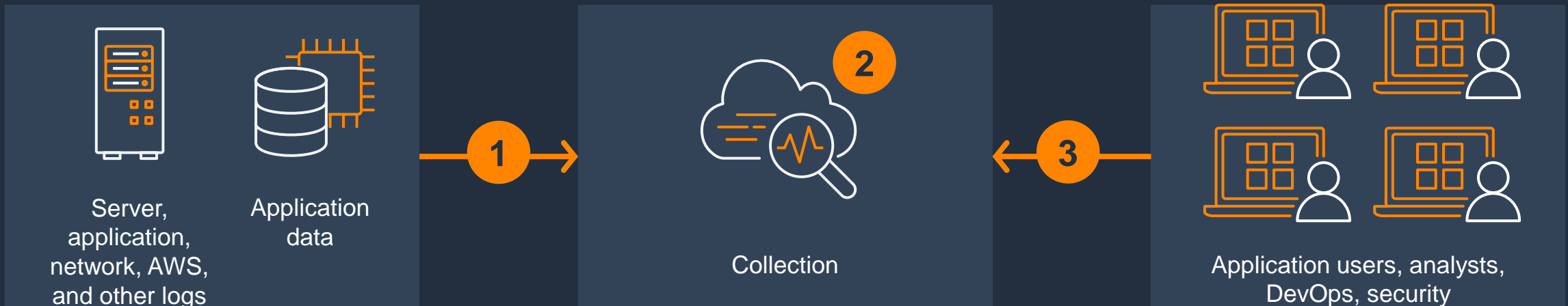
使い慣れたクライアントをそのまま利用可能

1 JSON 形式の **ドキュメント** を REST API 経由で **コレクション** に送信

2 ドキュメントがコレクション上の **インデックス** に格納され、検索可能になる。

3

- クラスターの REST API エンドポイントにクエリを発行し、JSON 形式のドキュメントを取得
- **ダッシュボード** に Web ブラウザからアクセス、可視化や分析作業を実施



OpenSearch Serverless によって削減可能な運用タスク

データライフサイクルの管理

- インデックス内のデータライフサイクルはサービスによって自動的に管理される
- ISM (Index State Management) や Curator によるインデックスの明示的な世代管理は不要
- コンプライアンスやコスト上の理由で時系列に沿ってインデックスを削除したい場合のみ、時系列のインデックス作成をユーザー側で実施する

バージョンアップグレード、ソフトウェアアップデート

- OpenSearch Serverless では、アップグレード・アップデートはサービスに影響のない形で、自動的に行われる。ユーザーによるアクションは不要

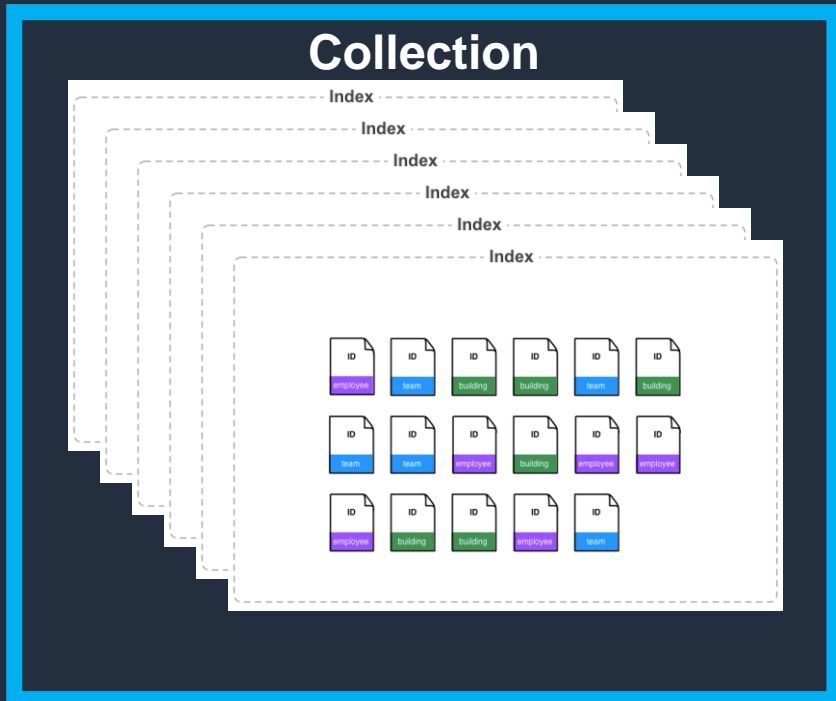
クラスターリソースのモニタリング

- CPU 使用率など各ノードのリソースの個別モニタリングは不要となる
- エラーレートやスループットといった全体的なモニタリングは別途必要



Collection (コレクション)

- Amazon OpenSearch Service におけるドメイン(クラスター)に相当する、論理的なインデックスの集合
- API、ダッシュボードエンドポイントはコレクションごとに発行される
- 2023 年 5月時点では OpenSearch 2.0 に準拠



```
{
  "collectionDetails":[
    {
      "arn":"arn:aws:aoss:us-east-1:123456789012:collection/1iu5usc4bz2f",
      "collectionEndpoint":"https://1iu5usc4bz2f.us-east-1.aoss.amazonaws.com",
      "createdDate":1637086138524,
      "dashboardEndpoint":"https://1iu5usc4bz2f.us-east-1.aoss.amazonaws.com/_dashboards",
      "description":"Some description",
      "id":"1iu5usc4bz2f",
      "type": "SEARCH",
      "name":"my-collection",
      "status":"ACTIVE"
    }
  ]
  "collectionErrorDetails":[]
}
```

コレクションの種類

- 検索用のコレクションと時系列データ用のコレクションが存在する
- コレクションの種類に応じて利用可能な API や、内部的な動作が異なる

SEARCH collection

- コンテンツ検索や商品検索などでの利用を想定
- ほぼ全てのデータはノードにアタッチされたホットストレージに格納される。
- ホットストレージに収容できないデータは S3 に格納され、検索時に改めてホットストレージに取得される
- ドキュメント ID を指定してドキュメントを作成することが可能

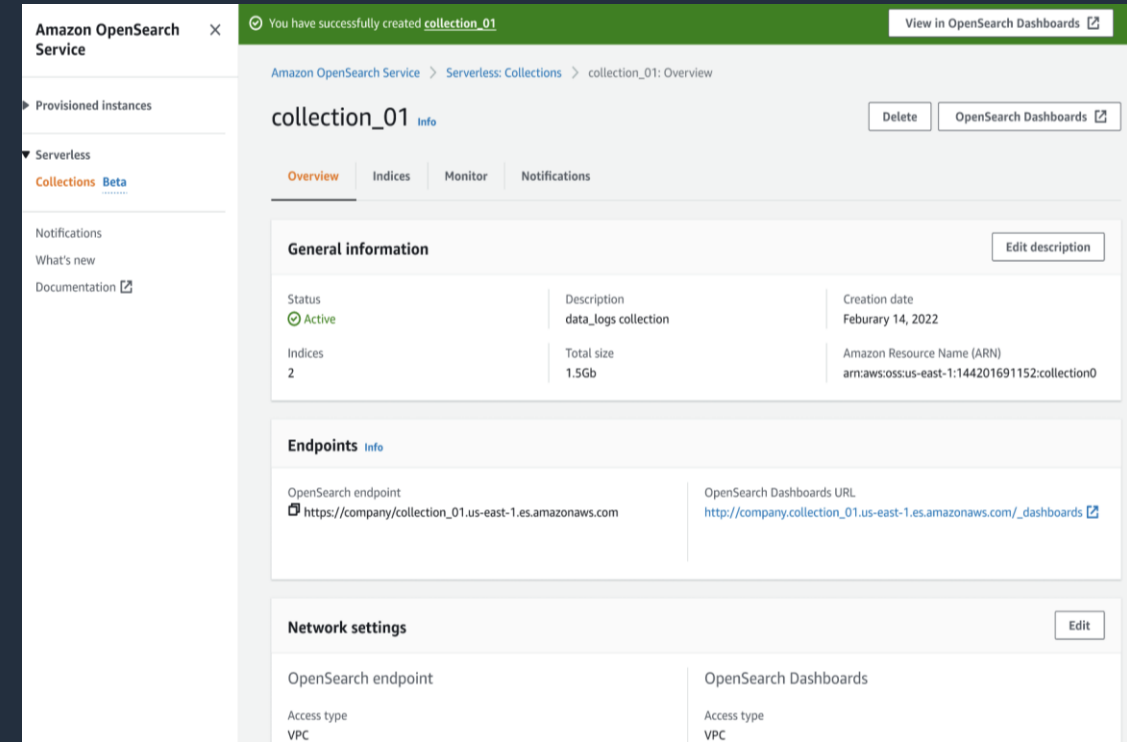
TIME-SERIES collection

- モニタリングやリアルタイム分析等で大規模なデータを扱うケースを想定
- ノードにアタッチされたホットストレージとウォームキャッシュを組み合わせる。
- 直近 24 時間のデータはホットストレージに、それ以降のデータはウォームキャッシュに格納される
- ドキュメント ID を指定してドキュメントを作成することは不可



API、コンソール、CloudFormation により数分で環境を構築

1. コレクション (インデックスのグループ) を作成。VPC とパブリックエンドポイントをサポート。コレクションごとに固有の暗号化キーを利用可
2. 従来の OpenSearch クライアントやパイプラインを使用してデータを投入
3. 既存の OpenSearch API を利用してインデックスに対してクエリを実行
4. サーバレス環境に展開された OpenSearch Dashboards でデータを分析



利用可能なクライアント

OpenSearch Serverless は API エンドポイントへアクセスする際、署名付きリクエストの利用が必須となる。IAM ユーザーもしくは IAM ロールのクレデンシャルを利用可能なクライアントを使用すること

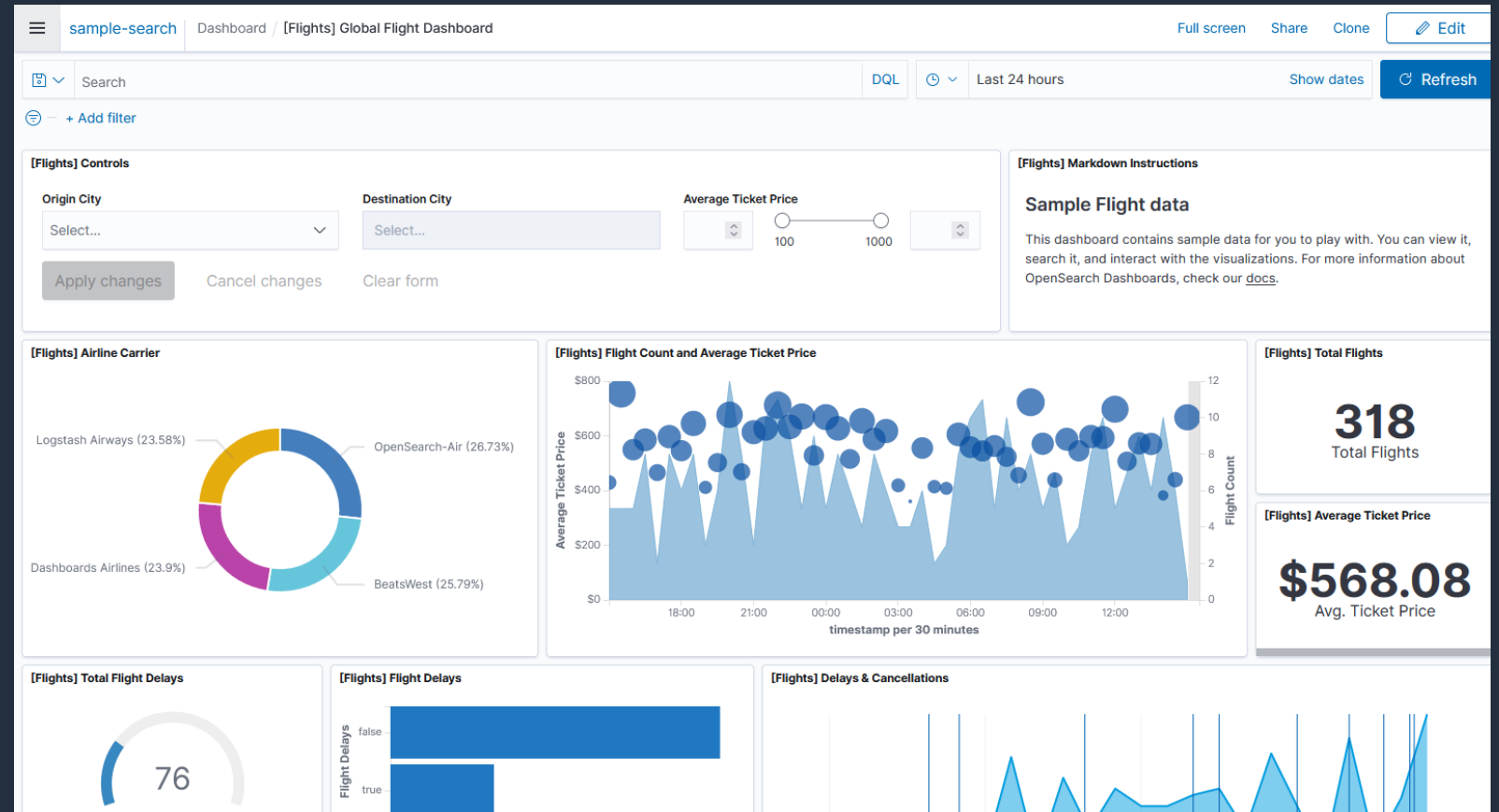
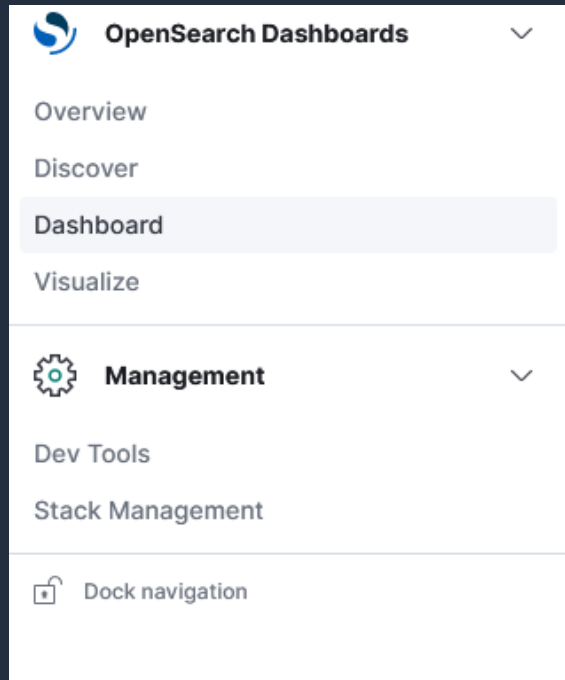
書き込み

- OpenSearch Client Library (バージョン 2.0 以降)
- Logstash ([logstash-output-opensearch](#) プラグインを利用)
- Fluentd([Fluentd OpenSearch plugin](#) プラグインを利用)
- Fluentbit ([OpenSearch output](#) を利用)
- Amazon Kinesis Data Firehose

検索: OpenSearch Client Library (バージョン 2.0 以降)

OpenSearch Dashboards

- Amazon OpenSearch Service 同様可視化、ログ検索用途で利用可能
- 現状は Discover, Dashboard, Visualize, Dev Tools の基本機能のみ提供



OCU (OpenSearch Capacity Unit) とは

検索とインデキシングそれぞれの処理に対して、個別に割り当てられるキャパシティユニット

1 OCU

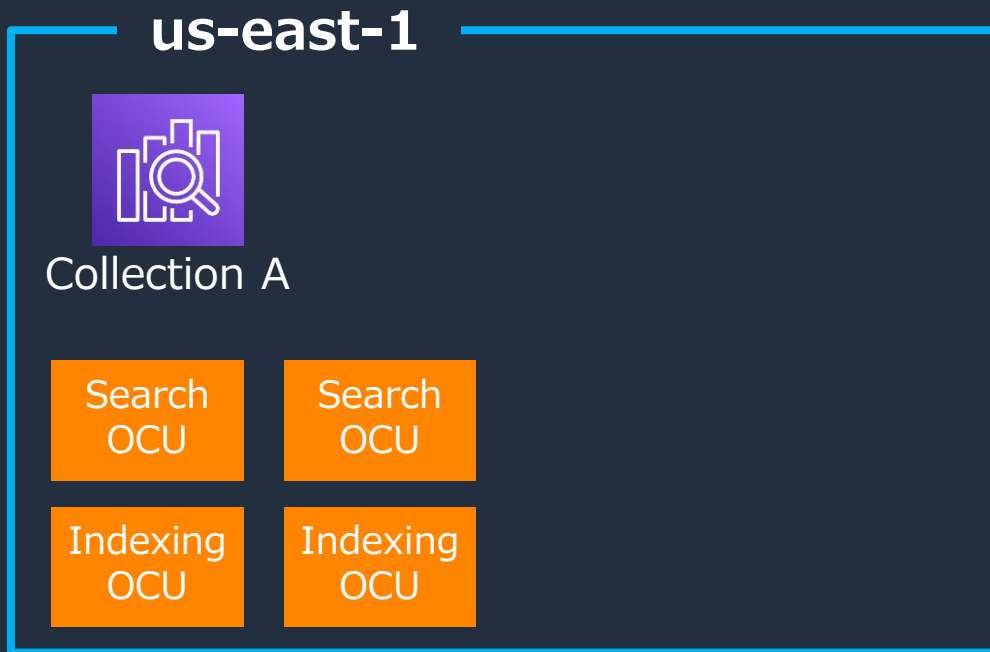
1 vCPU

6 GiB RAM

120 GiB Disk

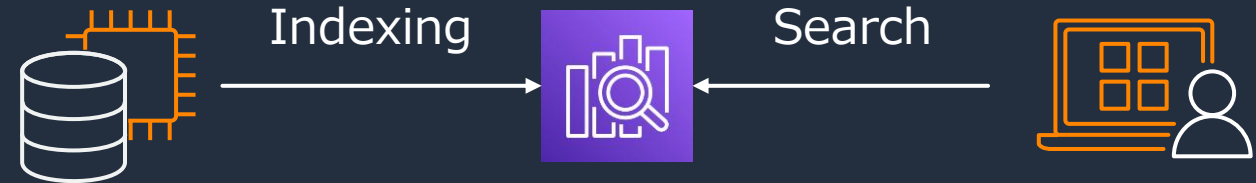
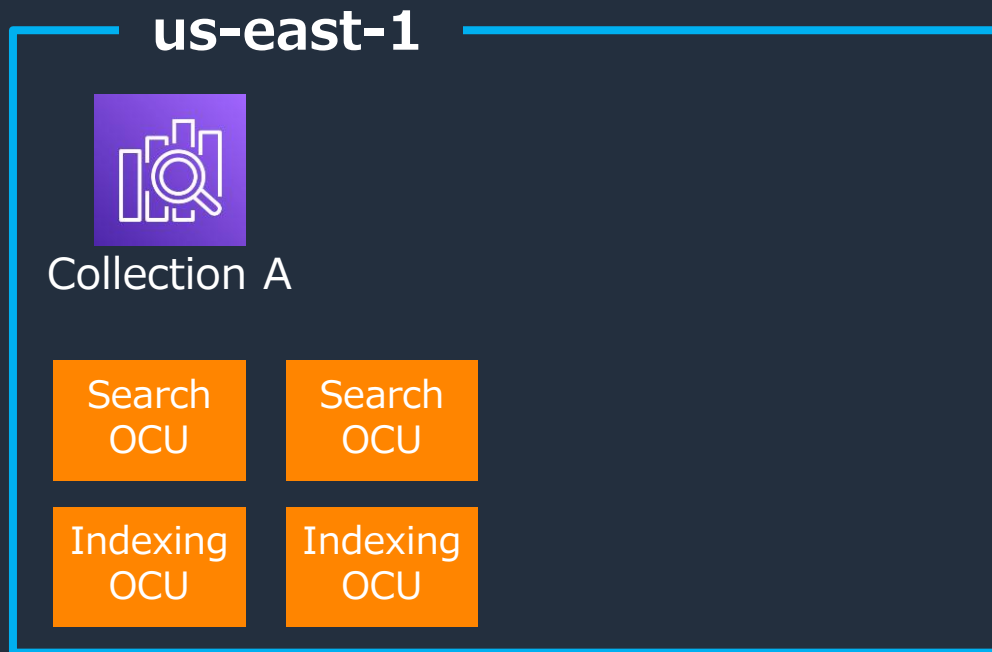
OCU (OpenSearch Capacity Unit) 割り当て

- 最初のコレクションを作成すると、アカウントごと(厳密にはリージョンごと)に、検索とインデキシングに 2 OCU ずつ、計 4 OCU が割り当てられる



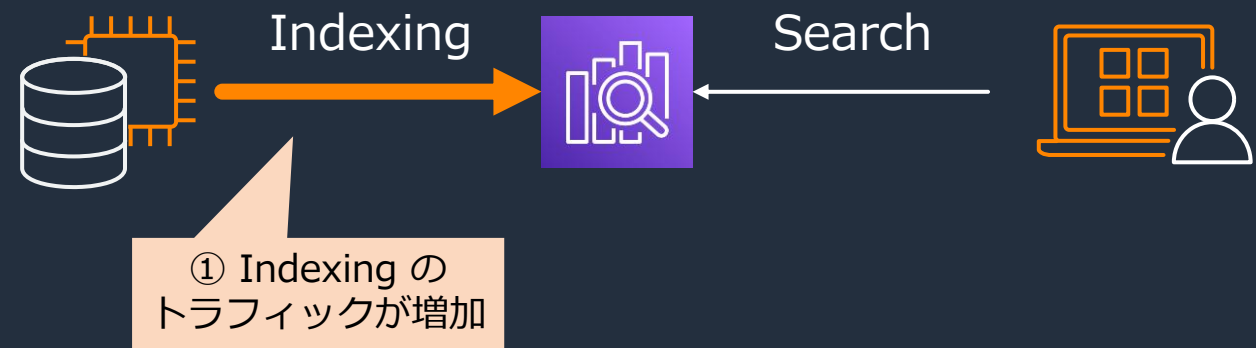
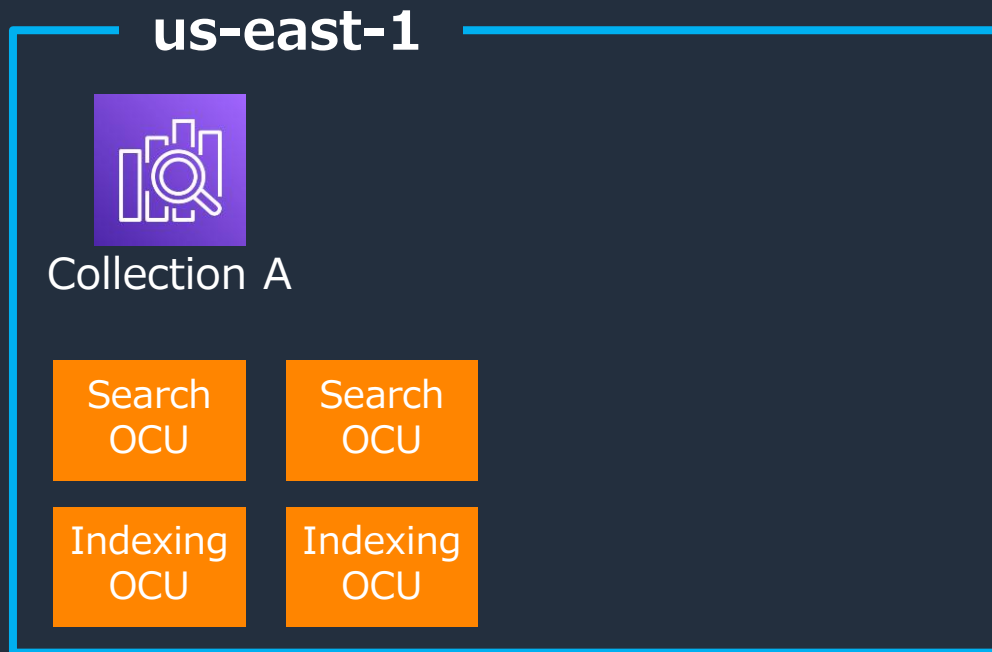
OCU (OpenSearch Capacity Unit) 割り当て

- 最初のコレクションを作成すると、アカウントごと(厳密にはリージョンごと)に、検索とインデキシングに 2 OCU ずつ、計 4 OCU が割り当てられる
- 処理要求や負荷の増加に応じて自動的に OCU が増加する (スケールアウト)



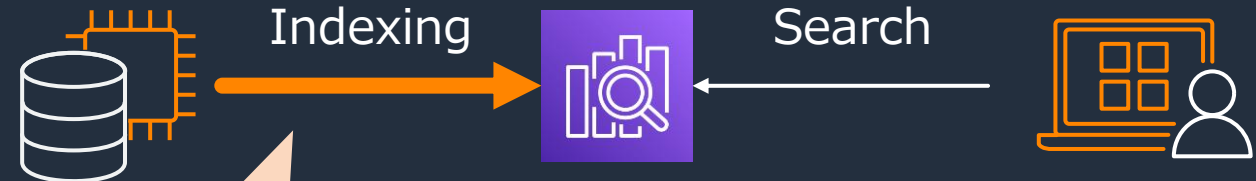
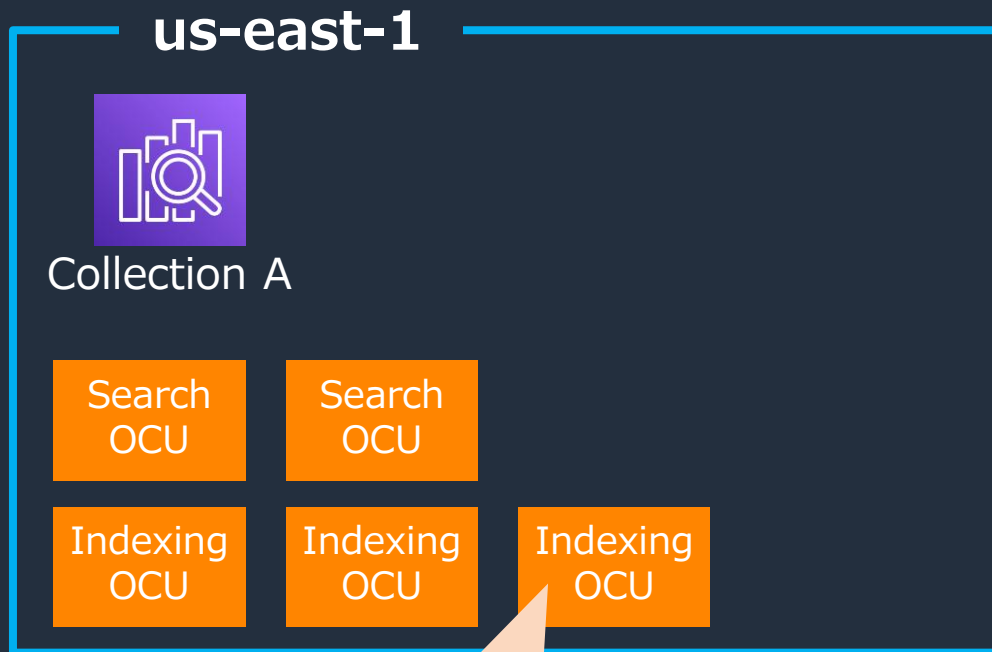
OCU (OpenSearch Capacity Unit) 割り当て

- 最初のコレクションを作成すると、アカウントごと(厳密にはリージョンごと)に、検索とインデキシングに 2 OCU ずつ、計 4 OCU が割り当てられる
- 処理要求や負荷の増加に応じて自動的に OCU が増加する (スケールアウト)



OCU (OpenSearch Capacity Unit) 割り当て

- 最初のコレクションを作成すると、アカウントごと(厳密にはリージョンごと)に、検索とインデキシングに 2 OCU ずつ、計 4 OCU が割り当てられる
- 処理要求や負荷の増加に応じて自動的に OCU が増加する (スケールアウト)

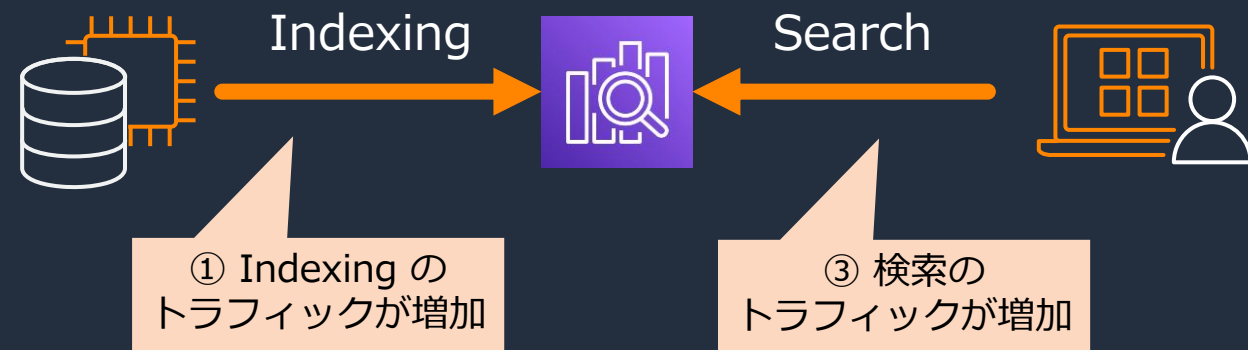
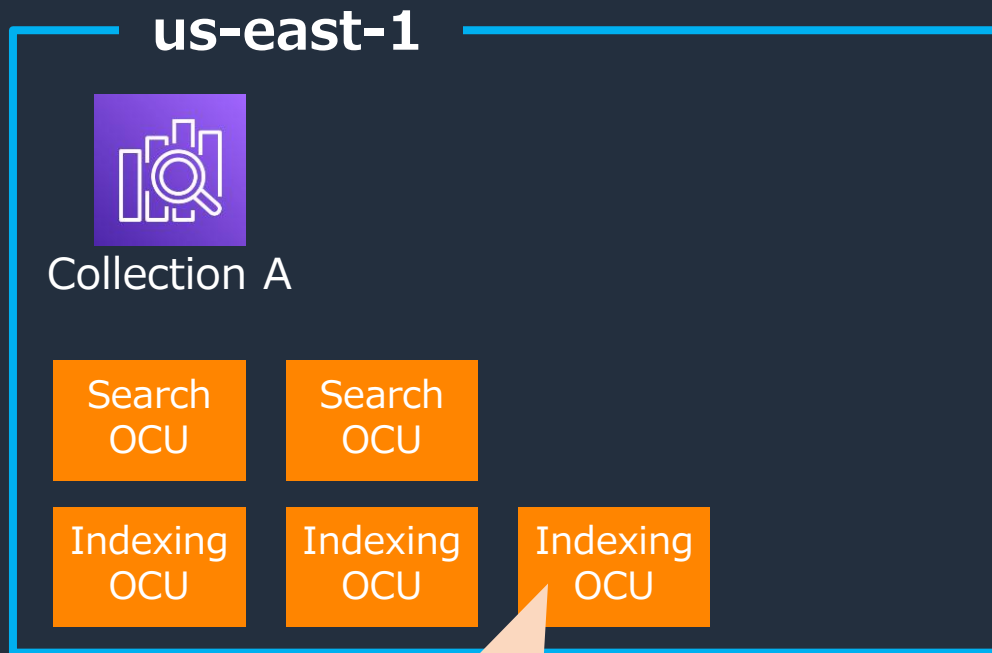


① Indexing の
トラフィックが増加

② Indexing OCU が
増加

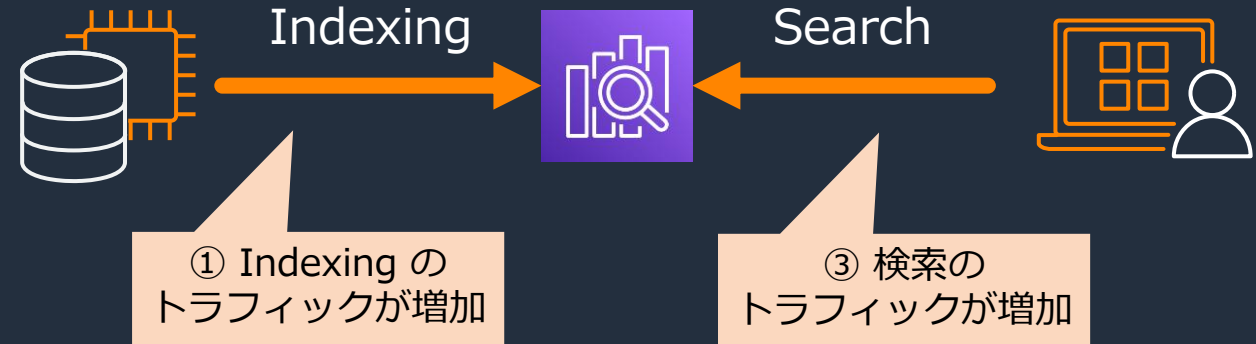
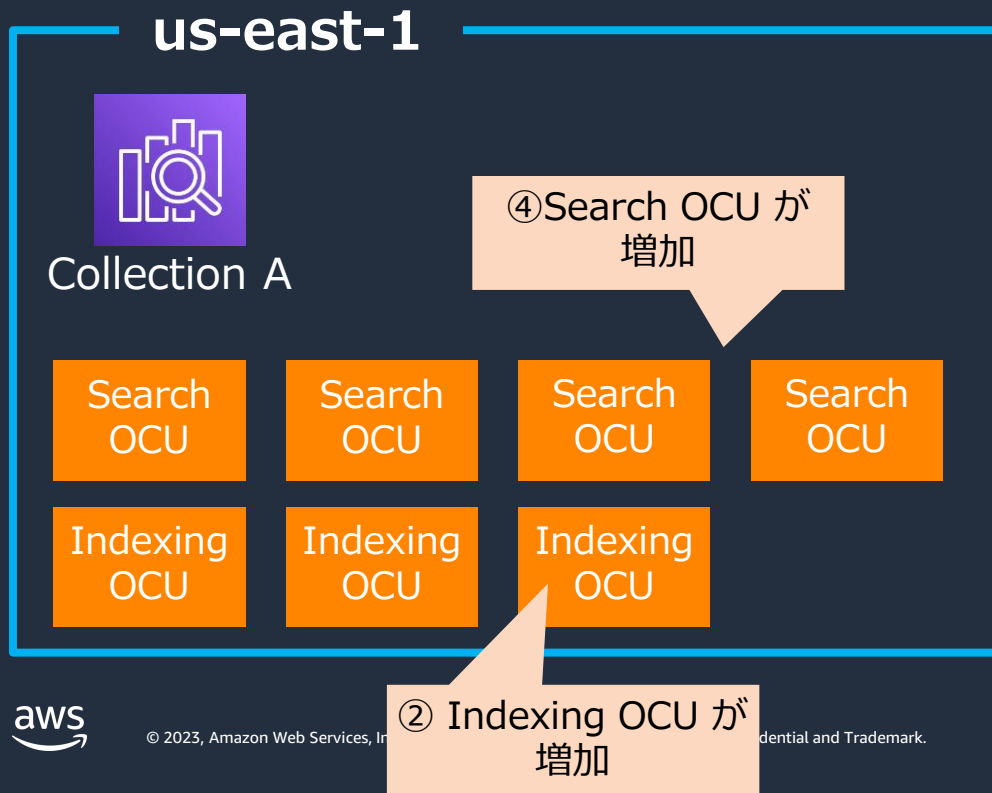
OCU (OpenSearch Capacity Unit) 割り当て

- 最初のコレクションを作成すると、アカウントごと(厳密にはリージョンごと)に、検索とインデキシングに 2 OCU ずつ、計 4 OCU が割り当てられる
- 処理要求や負荷の増加に応じて自動的に OCU が増加する (スケールアウト)



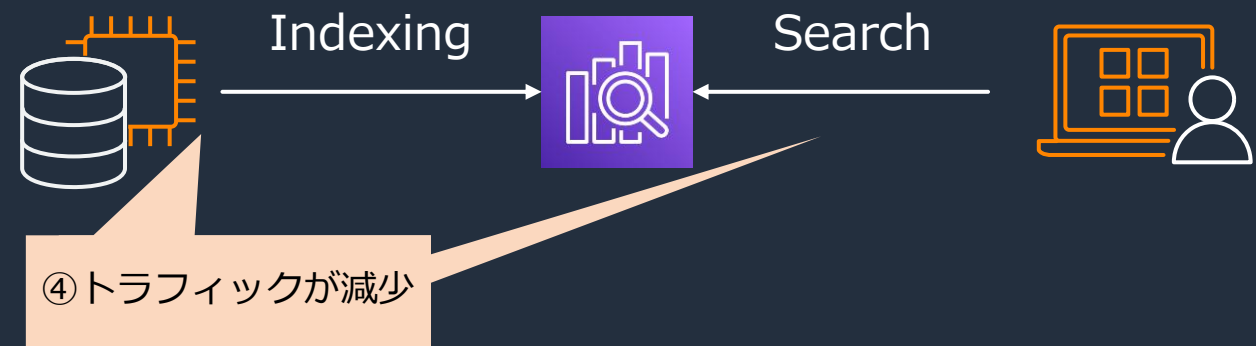
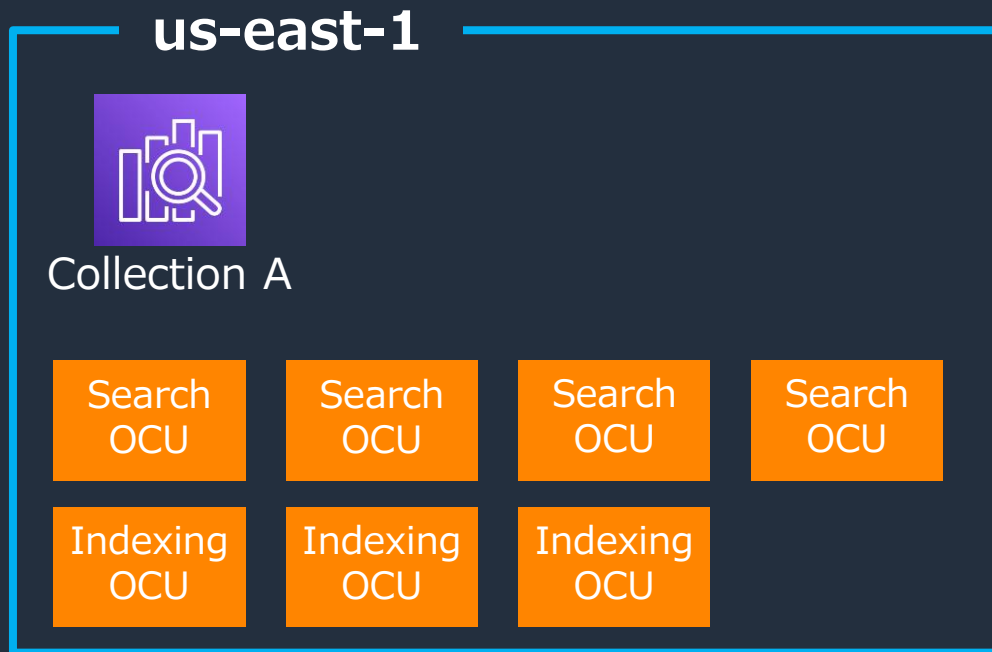
OCU (OpenSearch Capacity Unit) 割り当て

- 最初のコレクションを作成すると、アカウントごと(厳密にはリージョンごと)に、検索とインデキシングに 2 OCU ずつ、計 4 OCU が割り当てられる
- 処理要求や負荷の増加に応じて自動的に OCU が増加する (スケールアウト)



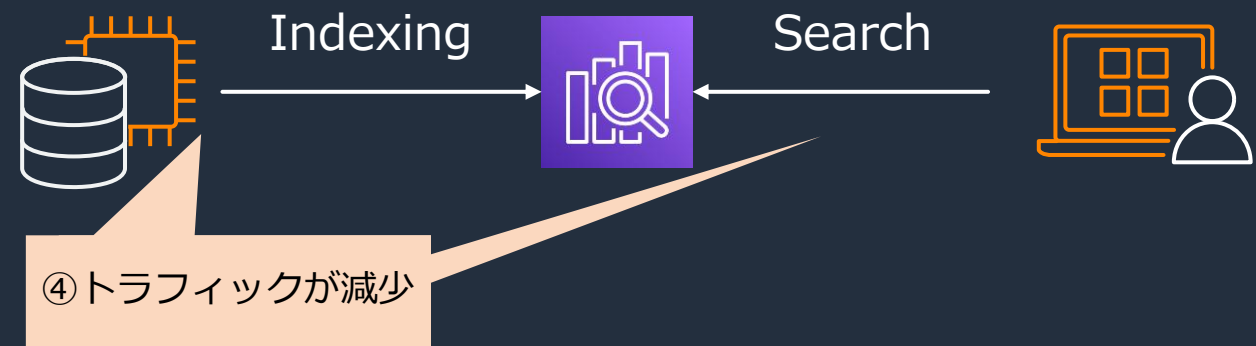
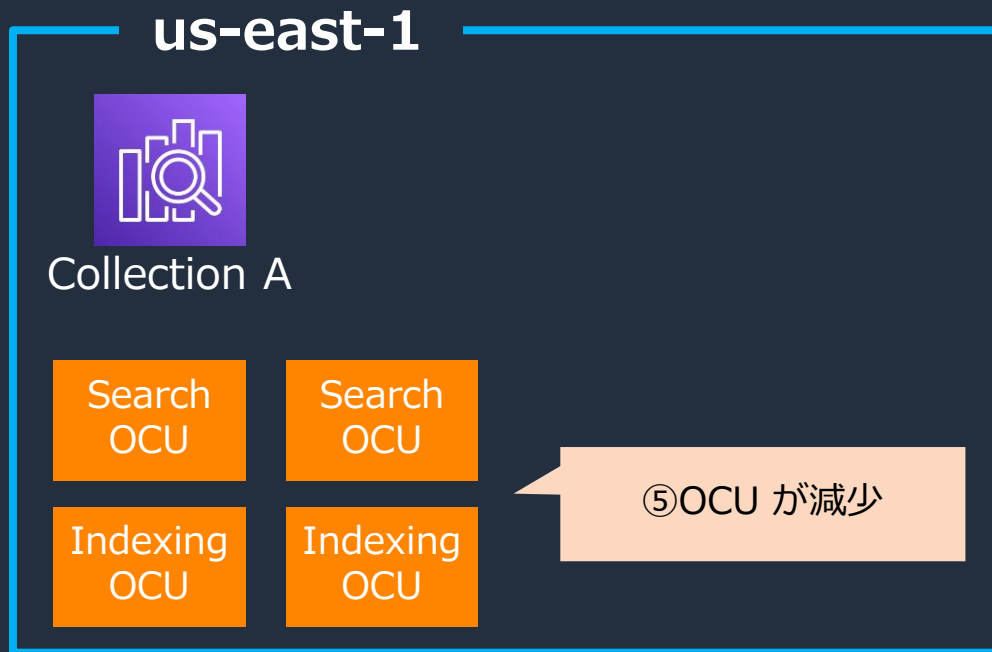
OCU (OpenSearch Capacity Unit) 割り当て

- 最初のコレクションを作成すると、アカウントごと(厳密にはリージョンごと)に、検索とインデキシングに 2 OCU ずつ、計 4 OCU が割り当てられる
- 処理要求や負荷の増加に応じて自動的に OCU が増加する (スケールアウト)
- 処理要求や負荷の減少に応じて自動的に OCU が減少する(スケールイン)



OCU (OpenSearch Capacity Unit) 割り当て

- 最初のコレクションを作成すると、アカウントごと(厳密にはリージョンごと)に、検索とインデキシングに 2 OCU ずつ、計 4 OCU が割り当てられる
- 処理要求や負荷の増加に応じて自動的に OCU が増加する (スケールアウト)
- 処理要求や負荷の減少に応じて自動的に OCU が減少する(スケールイン)



スケールアウト / スケールインの判断基準

- Ingest ノード、Search ノードの CPU、メモリ、ディスクリソース、ホットシャードに対する書き込み、読み取り要求の増加によって、閾値を超えた場合にスケールする
- 反対に、閾値を下回った場合は、徐々にスケールインを行う

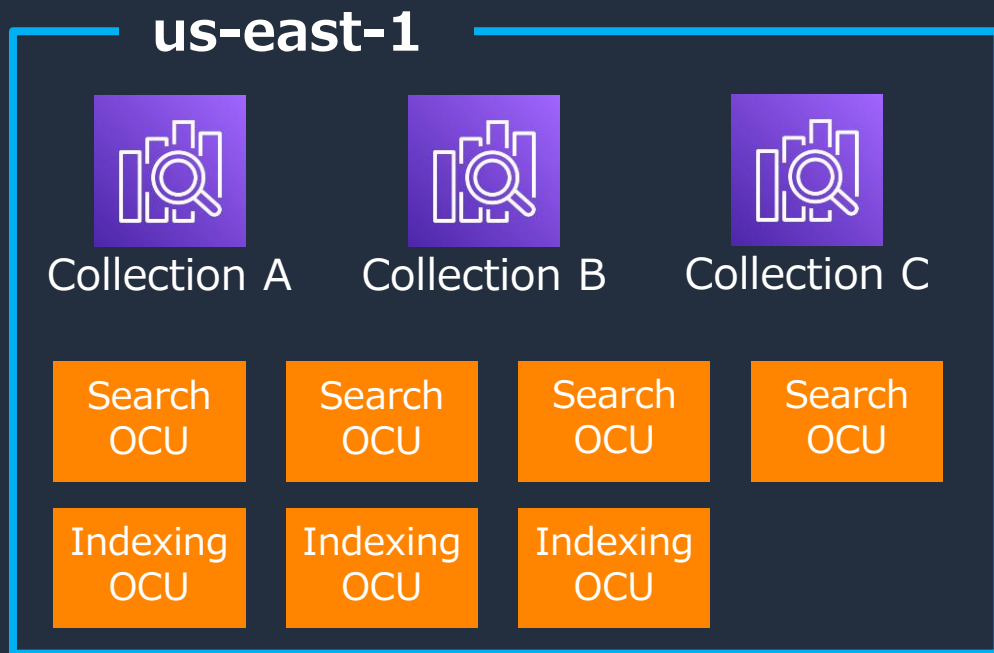
<https://aws.amazon.com/opensearch-service/faqs/#Serverless>

Q: When does OpenSearch Serverless scale up, and when does it scale down?

When the system resources such as CPU, memory, and disk limits in the ingestion or search nodes are breached or it notices hot shards processing large amounts of read or write requests, OpenSearch Serverless horizontally scales out nodes in response to increased workload demand. Similarly, when the resource utilization falls below a certain threshold, OpenSearch Serverless will automatically and gradually scale in the resources without impacting the performance.

OCU (OpenSearch Capacity Unit) の特徴

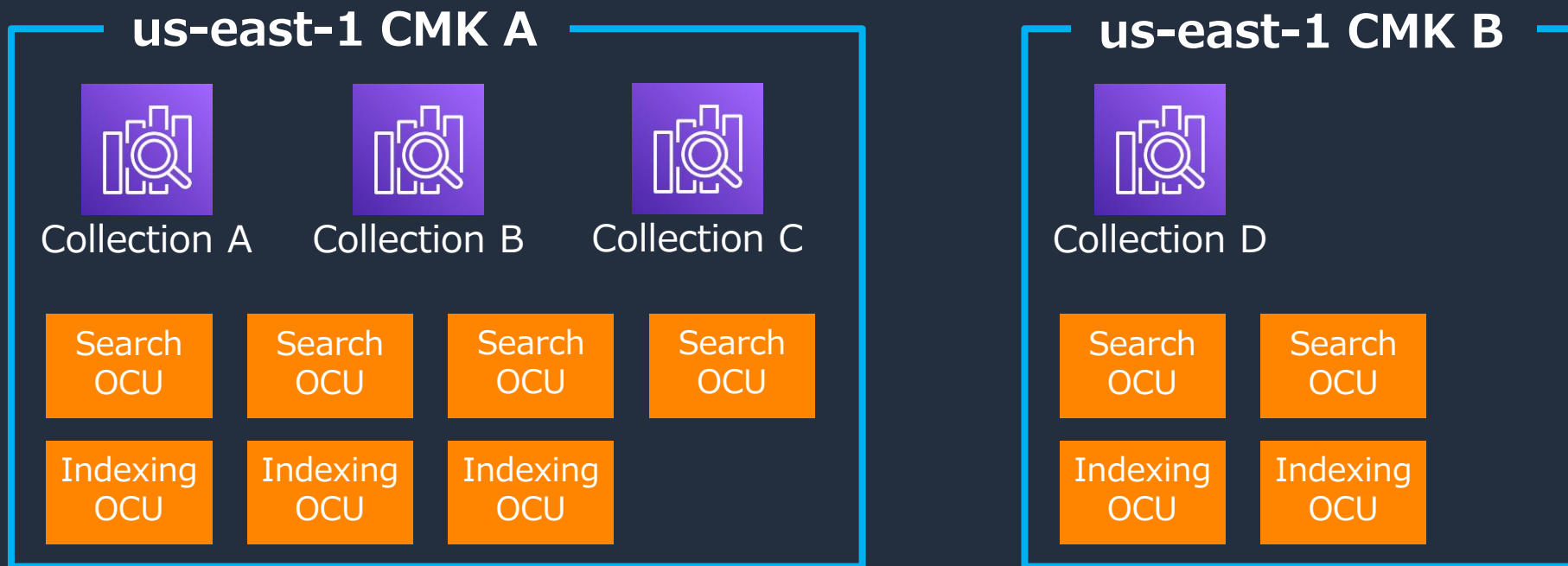
- ワークロードの有無にかかわらず、割り当て OCU に応じた時間課金となる
- OCU は同一リージョンであれば複数コレクション間で共有される



7 OCU

OCU (OpenSearch Capacity Unit) の特徴

- ワークロードの有無にかかわらず、割り当て OCU に応じた時間課金となる
- OCU は同一リージョンであれば複数コレクション間で共有される
- 異なるキーで暗号化を行った場合、OCU 共有の対象外となる



$$7 \text{ OCU} + 4 \text{ OCU} = 11 \text{ OCU}$$

料金 ポイント

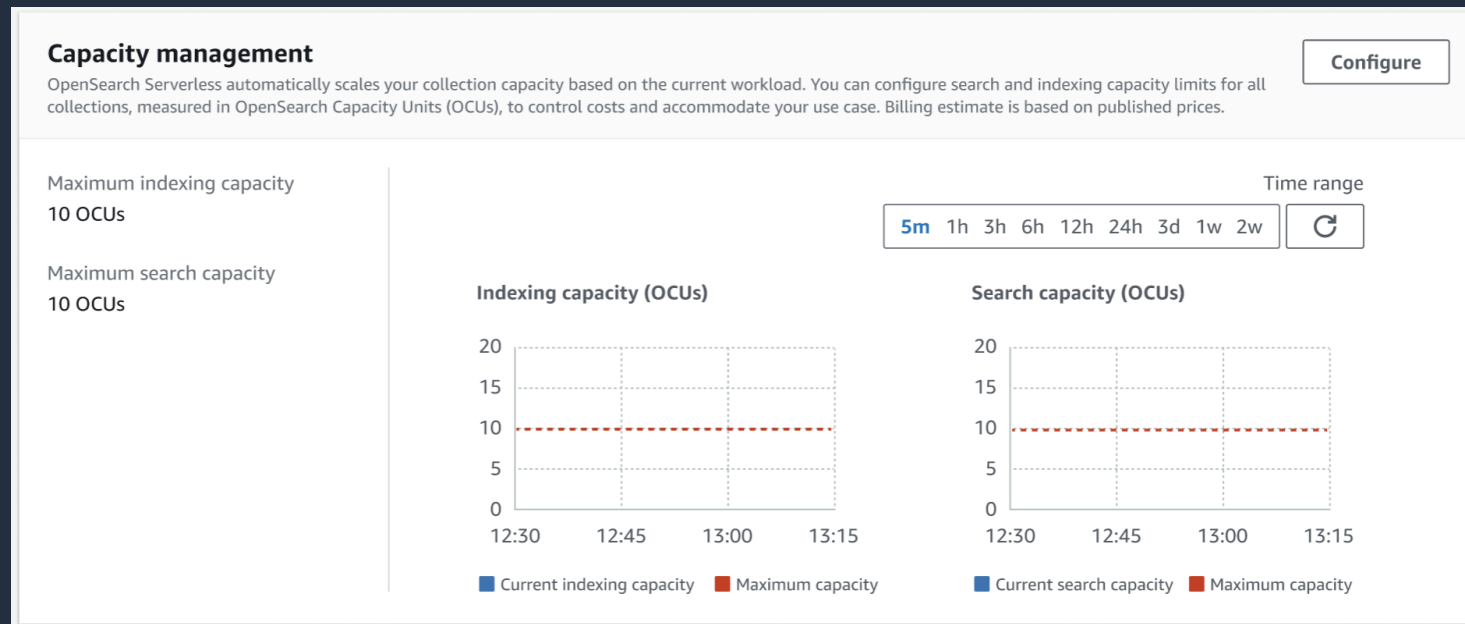
- OpenSearch Serverless の料金モデルは、割り当てられたキャパシティユニット(OCU)に応じた時間課金

料金概要(東京リージョン)

- OCU – インデキシング : \$0.334 per OCU per hour
- OCU – 検索: \$0.334 per OCU per hour
- マネージドストレージ: \$0.026 per GB / month
- OpenSearch Dashboards は無料で利用可能

キャパシティリミット

- インデキシング、検索 それぞれのキャパシティに上限を設けることが可能
- 2 から 50 まで自由に変更可能。デフォルト値は 10
- 現在割り当てられている OCU より低い Limit を設定することはできない
 - 割り当てられている OCU は CloudWatch メトリクスから確認可能



Amazon OpenSearch Serverless Demo



まとめ

- OpenSearch Serverless は OpenSearch 互換のサーバレスサービス
- 負荷に応じて OCU が動的に増減する、自動スケールアウト、スケールインをサポート
- 従来アーキテクチャで行っていたアップデートなどの運用タスクを削減
- コレクションの種類によって一部の仕組みや利用できるAPIが異なる
- 従来の Amazon OpenSearch Service からの移行に際しては、既存のワークロードや要件を確認し、移行の可能性を検討してから行うこと



Thank you!