



AWS X-Ray

AWS Black Belt Online Seminar

津郷 光明

Amazon Web Service Japan G.K.
Solutions Architect
2023/2

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
- 動画を一時停止・スキップすることで、興味がある分野・項目だけの聴講も可能、スキマ時間の学習にもお役立ていただけます
- 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=PLzWGOASvSx6FIwIC2X1nObr1KcMCBBIqY>

内容についての注意点

- 本資料では2023年2月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。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

自己紹介

名前：津郷 光明

所属：エンタープライズ技術本部
ソリューションアーキテクト

経歴：前職は国内SIerにてシステム開発に従事



好きなAWSサービス：AWS X-Ray / AWS Distro for OpenTelemetry

本セミナーの対象者

- マイクロサービスアプリケーションのパフォーマンス分析に課題を感じている方
- これからAWSを利用した運用設計を担当される方
- AWSでのObservabilityに興味のある方

アジェンダ

AWS X-Ray とは

AWS X-Ray の概念

AWS X-Ray によるアプリケーション分析

AWS X-Ray によるデータ収集の仕組み

AWS X-Ray の利用方法

AWS モニタリングサービスとの連携

料金・注意点

まとめ

アジェンダ

AWS X-Ray とは

AWS X-Ray の概念

AWS X-Ray によるアプリケーション分析

AWS X-Ray によるデータ収集の仕組み

AWS X-Ray の利用方法

AWS モニタリングサービスとの連携

料金・注意点

まとめ

マイクロサービスにおける障害対応・パフォーマンス分析

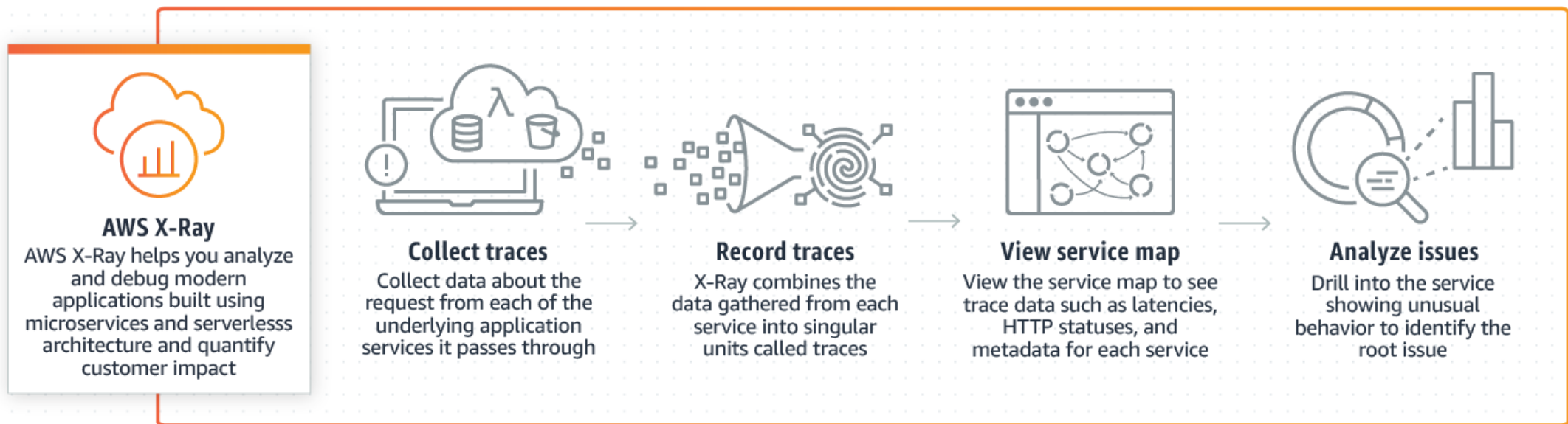
マイクロサービスでの障害原因の特定やパフォーマンスボトルネック特定は
モノリシックアプリケーションの場合よりも複雑になる

- **障害発生時の原因究明**
 - ログが分散・それぞれ独自のフォーマット
 - 依存しあう多数の多様なサービス群
- **パフォーマンス分析が難しい**
 - ボトルネックのサービスを特定する必要がある
 - サービス間の依存関係と全体像は日々変化



AWS X-Ray の概要

アプリケーションやその基盤となるサービスの実行状況を把握し
パフォーマンスの問題やエラーの根本原因を特定する



AWS X-Ray のメリット



リクエスト実行状況の確認

アプリケーションを構成する個々のサービスやリソースの実行結果ステータスを集計し、アプリケーションの実行状況を End-to-End で確認可能



アプリケーションの問題の検出

アプリケーションの実行状況についての関連する情報を収集し、問題の根本原因を調査可能



アプリケーションのパフォーマンス向上

サービスやリソースの関係をリアルタイムで表示し、レイテンシ増加やパフォーマンス低下などのボトルネックを特定可能



AWSとの連携

さまざまなAWSサービスと連携



さまざまなアプリケーション向けの設計

非同期のシンプルなイベント呼び出し、3層のウェブアプリケーション、数千のサービスから構成される複雑なマイクロサービスも分析可能

ObservabilityにおけるAWS X-Rayの位置付け

Observabilityとは

- システムの動作状況を把握できている状態
- システム運用において、判断に必要な情報がきちんと取得できている状態

3つの柱



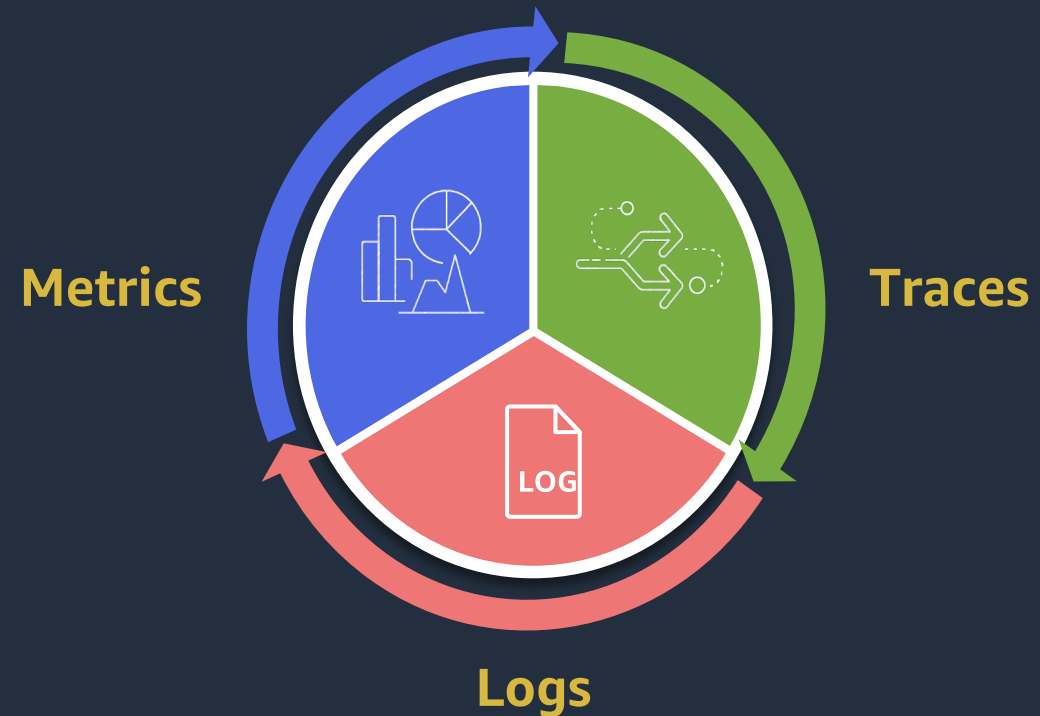
- ある時点のシステム状態を表現する **数値情報**
- 例) CPU 使用率、リクエストレート、など



- システム内で発生した **イベント情報**
- 例) アクセスログ、エラー情報、など



- 1つのトランザクションにおける **フロー情報**
- 例) HTTP リクエストの受け取りからレスポンスまで



ObservabilityにおけるAWS X-Rayの位置付け

Observabilityとは

- システムの動作状況を把握できている状態
- システム運用において、判断に必要な情報がきちんと取得できている状態

3つの柱



- ある時点のシステム状態を表現する **数値情報**
- 例) CPU 使用率、リクエストレート、など



- システム内で発生した **イベント情報**
- 例) アクセスログ、エラー情報、など



- 1つのトランザクションにおける **フロー情報**
- 例) HTTP リクエストの受け取りからレスポンスまで



AWS X-Ray

アジェンダ

AWS X-Ray とは

AWS X-Ray の概念

AWS X-Ray によるアプリケーション分析

AWS X-Ray によるデータ収集の仕組み

AWS X-Ray の利用方法

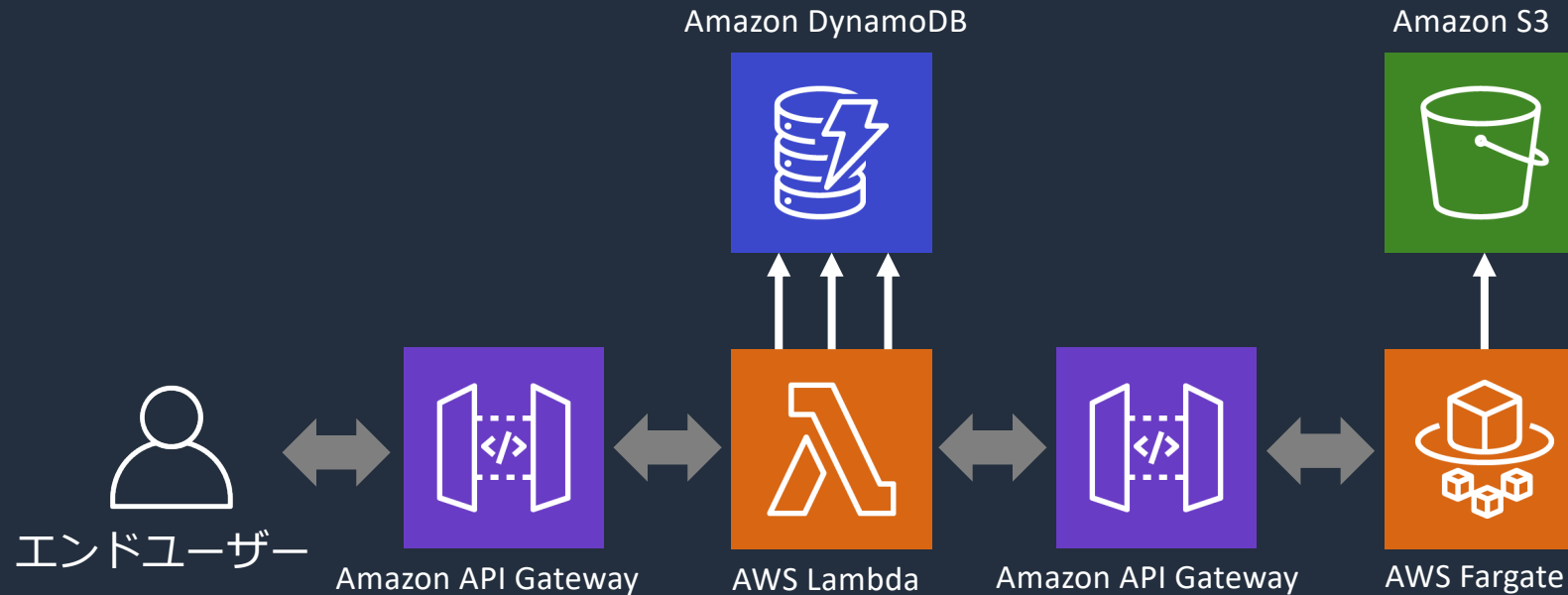
AWS モニタリングサービスとの連携

料金・注意点

まとめ

AWS X-Ray の概念

AWS X-Rayではセグメント形式でデータを管理します



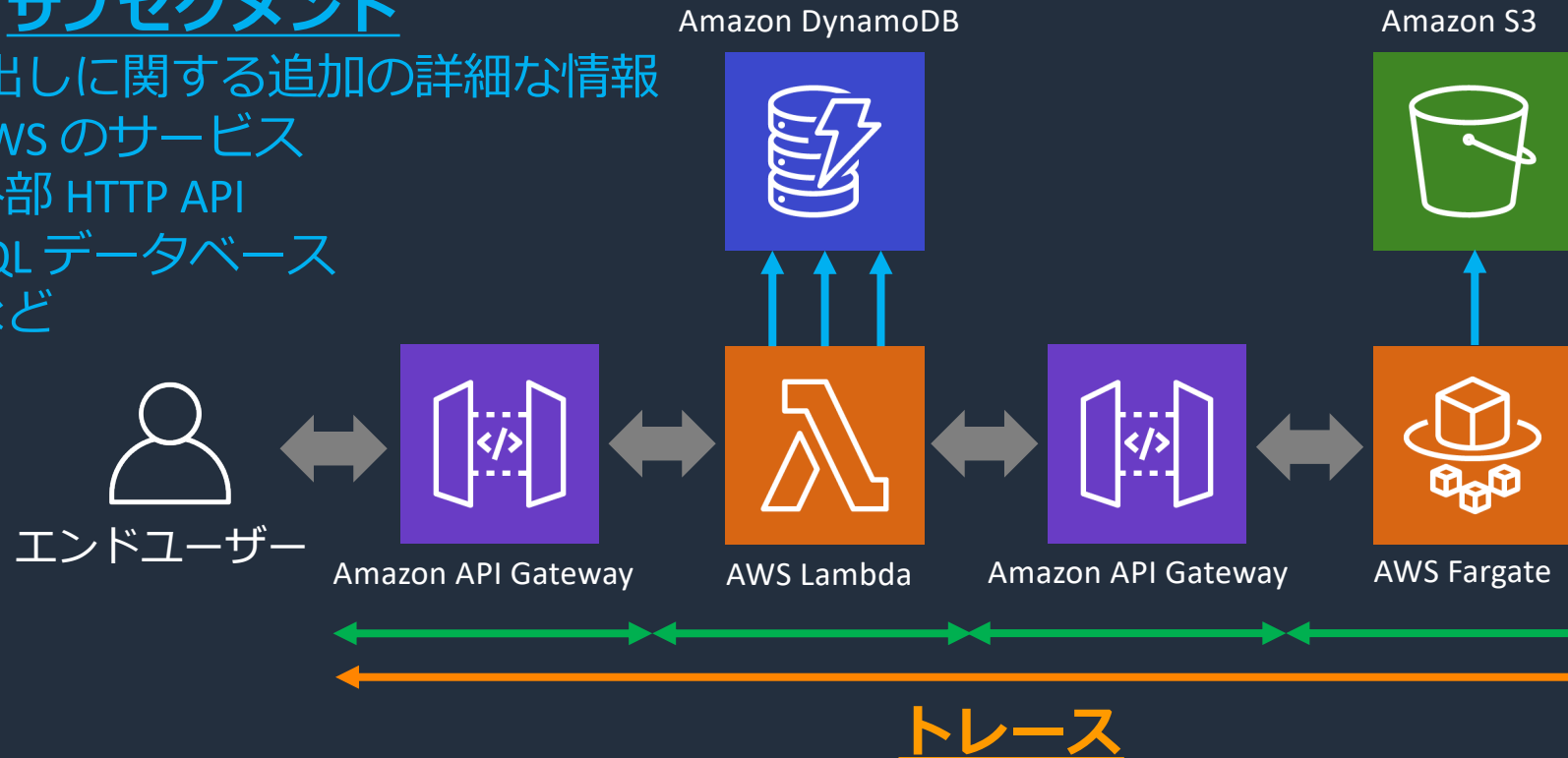
AWS X-Ray の概念

セグメント、サブセグメント、トレースについては以下のような単位となる

サブセグメント

呼び出しに関する追加の詳細な情報

- AWS のサービス
- 外部 HTTP API
- SQL データベース
- など



セグメント

動作に関するデータ

- リソース名
- リクエストの詳細
- 行った作業の詳細

1つのリクエストで生成されたセグメントの全てを収集したもの

AWS X-Ray の概念

セグメント、サブセグメント、トレースについては以下のような単位となる

サブセグメント

呼び出しに関する追加の詳細な情報

- AWS のサービス
- 外部 HTTP API

- どのようにトレースが含むセグメントを管理している？
- セグメントのつながり・順序はどう管理している？



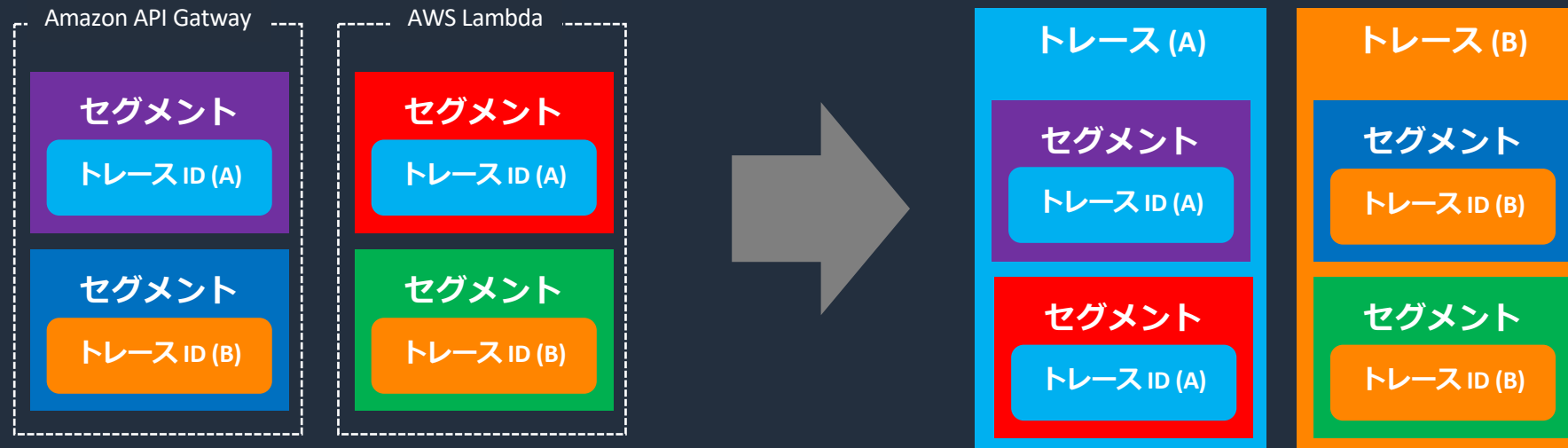
1つのリクエストで生成されたセグメントの全てを収集したもの

AWS X-Ray の概念

どのようにトレースが含むセグメントを管理している？

- **トレース ID**

- リクエストを追跡するための ID
- リクエストごとに一意
- 基本的に最初にセグメントを生成するタイミングで生成

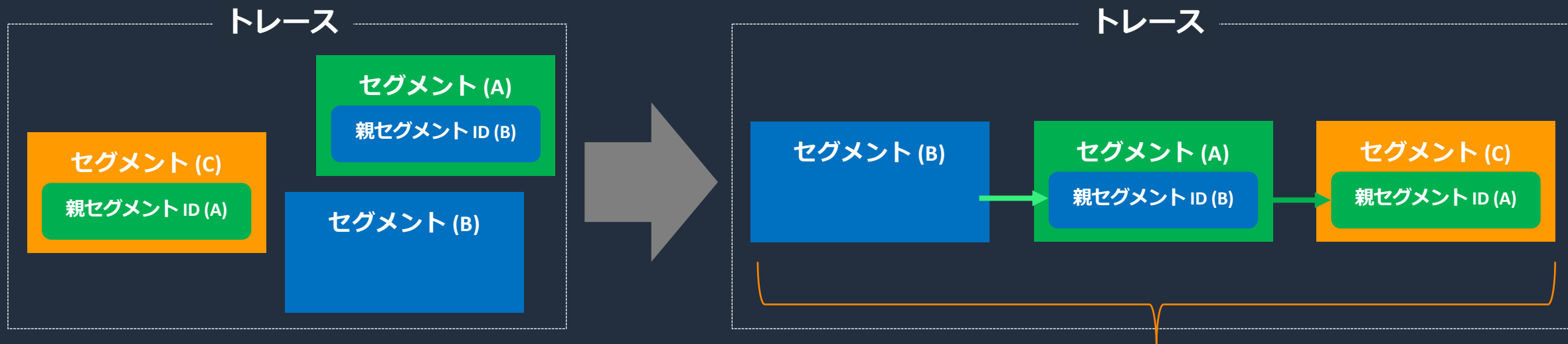


AWS X-Ray の概念

セグメントのつながり・順序はどう管理している？

• セグメント ID

- トレース内のセグメントに対して一意に決まる
- 単一のサービスのリクエストを追跡するための ID
- **親セグメント ID** によってセグメントのつながり・順序が決まる



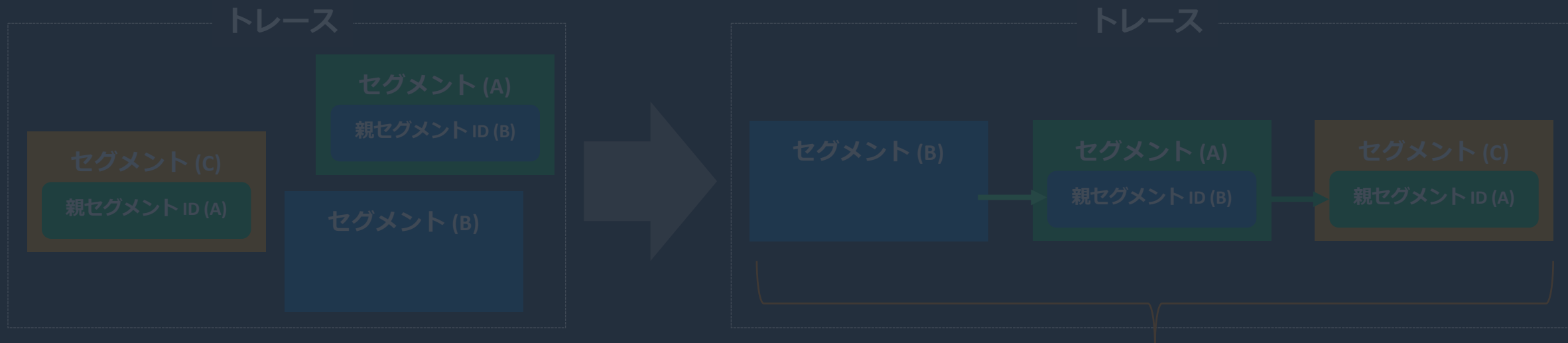
AWS X-Ray の概念

セグメントのつながり・順序はどう管理している？

- セグメント ID

- トレース内のセグメントに対して一意に決まる
- 単一のサービスのリクエストを追跡するための ID

• **親セグメント ID やセグメント ID をどう伝搬している？**



AWS X-Ray の概念

トレース ID や セグメント ID をどう伝搬している？

- **トレースヘッダー**

- トレース ID、親セグメント ID、サンプリングデシジョンが定義された HTTP ヘッダー

X-Amzn-Trace-Id: Root=1-57...93;Parent=53...d8;Sampled=1

トレース ID

親セグメント ID

サンプルデシジョン サンプリング対象か否か (1=true, 0=false)

AWS X-Ray の概念

サンプリング

- AWS X-Ray では**サンプリングルールに従い記録されるリクエストが決まる**
- デフォルトのサンプリングルール
 - 1秒ごとに最初のリクエストは必ずトレースされる
 - それ以降のリクエストについては全体の5%がトレースされる
- **なぜサンプリングが必要？**
 - 課金額のコントロール
 - パフォーマンスへの影響をコントロール
- **設定方法**
 - コンソール
 - ローカルルール (SDK から設定を読み込む)

AWS X-Ray の概念

セグメントに情報を追加したい

検索しやすくするため

- **注釈**

- フィルタ式 (後述) で使用するためインデックス化されたキーと値のペア
- 例 `"aws:api_stage": "prod"`

デバッグや分析のため

- **メタデータ**

- 任意のタイプの値を持つことができるキーと値のペア
- フィルタ式で使用するためのインデックスは作成されない
- 例 `"game": {"id": "abcdefg", "session": "xxxyyyzzz"}`

アジェンダ

AWS X-Ray とは

AWS X-Ray の概念

AWS X-Ray によるアプリケーション分析

AWS X-Ray によるデータ収集の仕組み

AWS X-Ray の利用方法

AWS モニタリングサービスとの連携

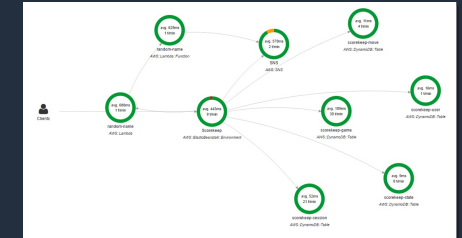
料金・注意点

まとめ

AWS X-Rayが提供するアプリケーション分析ツール

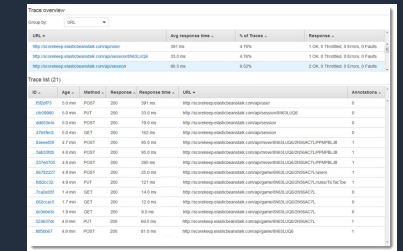
・サービスマップ：

マイクロサービスアプリケーションにおいて把握が難しいサービス間の依存関係をわかりやすく可視化



・トレースリスト：

詳細なトレースデータにアクセス可能セグメント、サブセグメントの実行状況を可視化



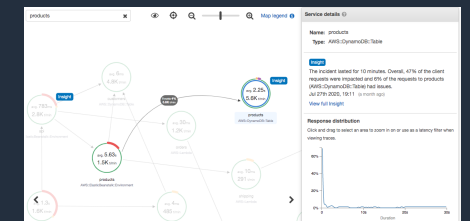
・アナリティクス：

インタラクティブな操作によって素早い分析が可能

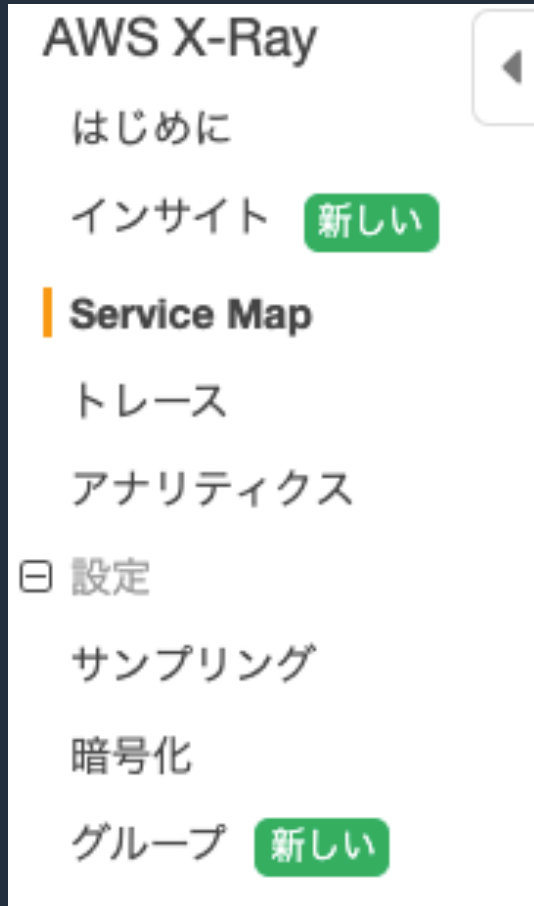


・インサイト：

パフォーマンスやアプリケーション上での問題を継続的に自動で検知できる



サービスマップ



- 読み取れる情報
- フィルタ式
- グループ



サービスマップ

アプリケーションによって生成されたトレースデータをビジュアルに表現

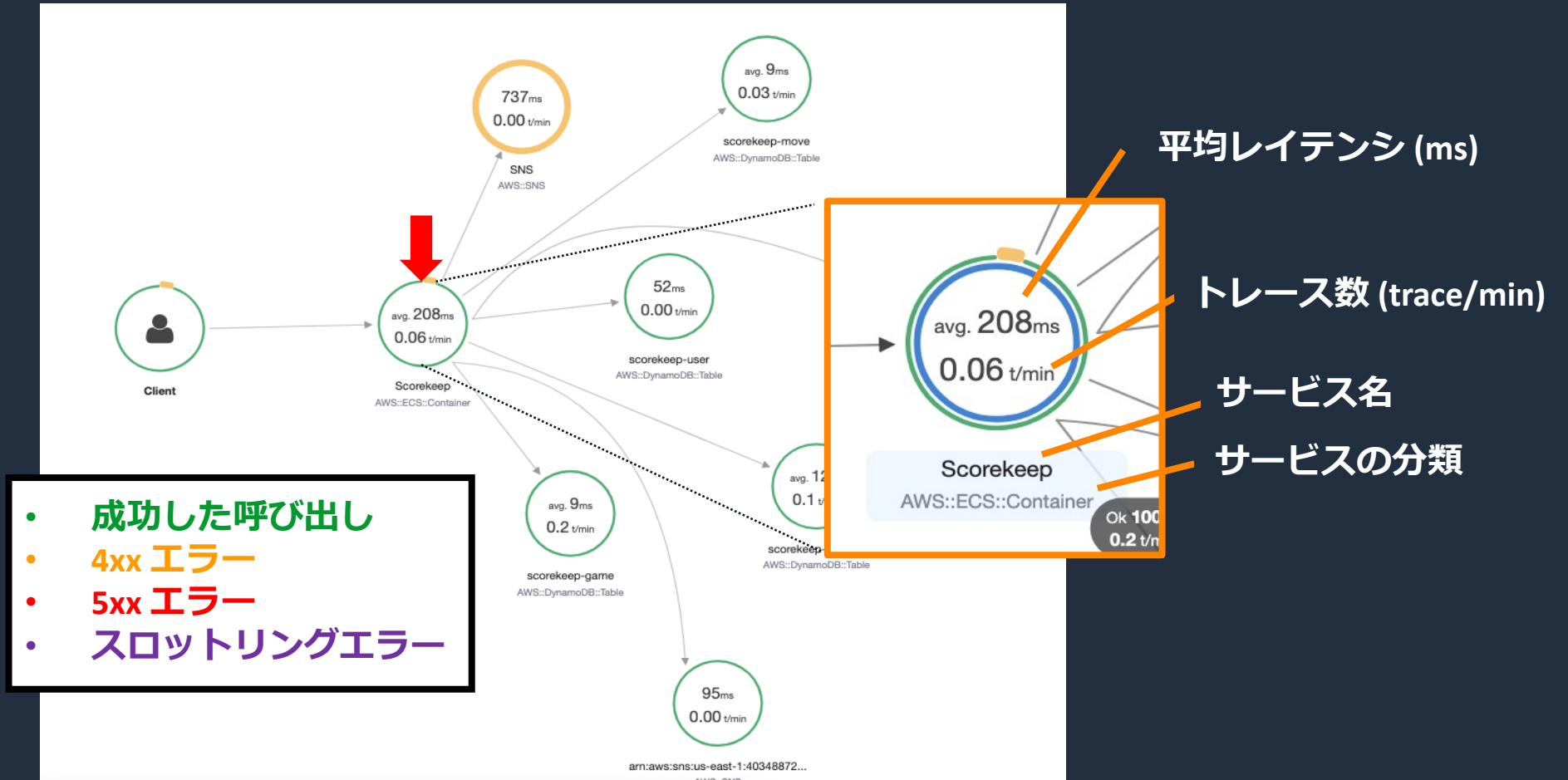
- リクエストを処理するサービスノード
- リクエストの発信元であるクライアントノード
- リクエストを処理するアプリケーションが使用するサービスノード

が表示される



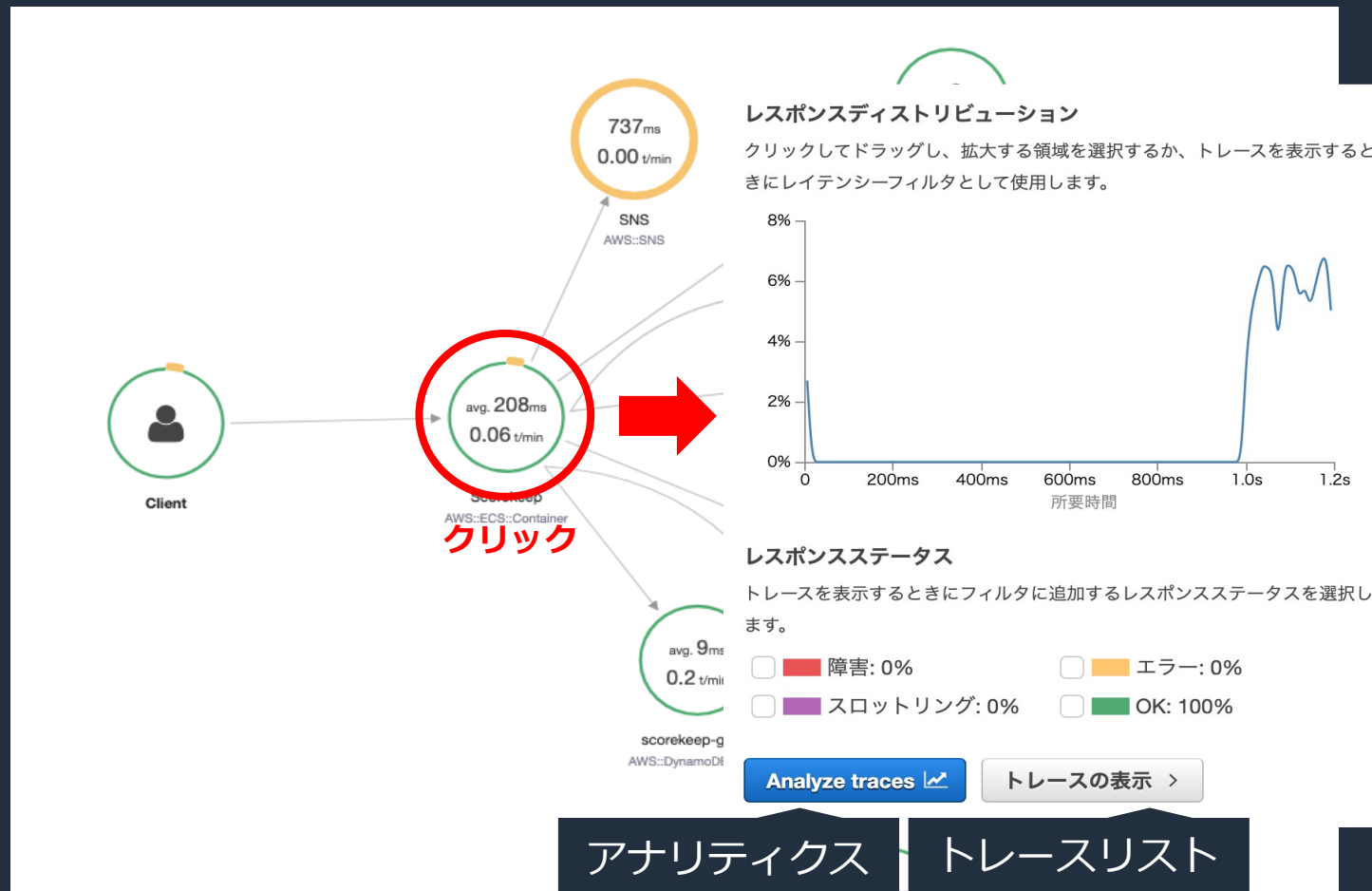
サービスマップ

各ノードには障害特定やパフォーマンス分析につながる情報が表示される



サービスマップ

調査したいノードを選択するとレスポンスに関する詳細情報が確認可能



フィルタ式

デフォルト ▾

🔍 サービス名、注釈、トレースID を入力します。または、[ヘルプ] アイコンをクリックして詳細を確認します。



- トレースを絞り込むための機能
- **トレースリストとアナリティクス**で利用可能
 - `duration >= 5 AND duration <= 8` 総所要時間が5秒～8秒
 - `http.status != 200` レスポンスステータスが 200 OK ではない
 - `service(name) { filter }` 特定のサービスを含んだリクエスト
 - `edge(source, destination) { filter }` 特定のサービス間の接続

https://docs.aws.amazon.com/ja_jp/xray/latest/devguide/xray-console-filters.html

グループ

デフォルト ▾

🔍 サービス名、注釈、トレースID を入力します。または、[ヘルプ] アイコンをクリックして詳細を確認します。



- グループ

- ・ フィルタ式で定義されるトレースのコレクション
- ・ サービスマップ、トレースリスト、アナリティクスで利用可能
- ・ Amazon CloudWatch にメトリクスを作成 (トレース数)

名前

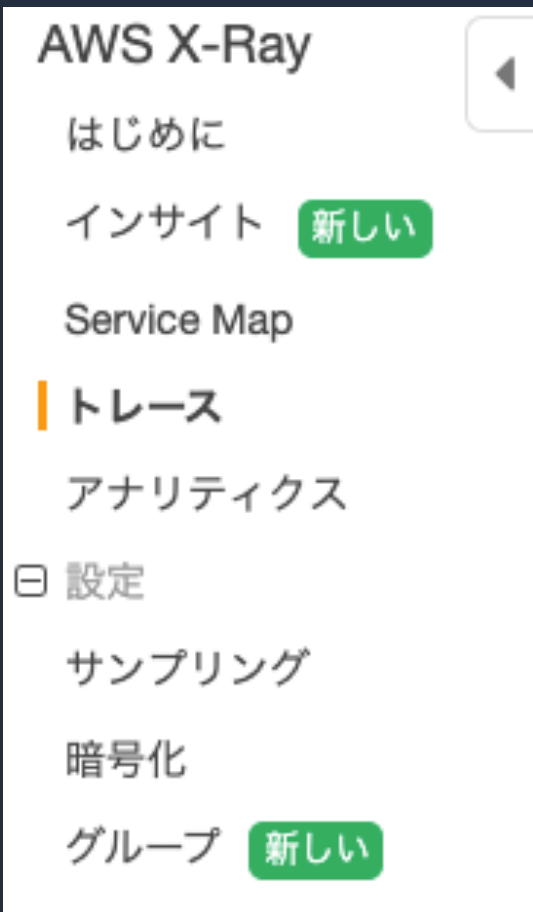
high-latency

フィルタ式

```
service("api.example.com") { fault = true OR responsetime > 5 }
```

https://docs.aws.amazon.com/ja_jp/xray/latest/devguide/xray-console-filters.html

トレースリスト



- トレースリスト
- セグメント・サブセグメント
- カスタムサブセグメント
- セグメント詳細

Method	Response	Response time	URL	Annotations
GET	200	391 ms	http://scorekeep.elasticbeanstalk.com/api/user	0
POST	200	33.0 ms	http://scorekeep.elasticbeanstalk.com/api/session/8N63LUQ6	0
POST	200	19.0 ms	http://scorekeep.elasticbeanstalk.com/api/session	0
POST	200	19.0 ms	http://scorekeep.elasticbeanstalk.com/api/session	0
POST	200	19.0 ms	http://scorekeep.elasticbeanstalk.com/api/session	0
POST	200	19.0 ms	http://scorekeep.elasticbeanstalk.com/api/move/8N63LUQ6/2N56AC7L/PPMPBLJB	1
POST	200	19.0 ms	http://scorekeep.elasticbeanstalk.com/api/move/8N63LUQ6/2N56AC7L/PPMPBLJB	1
POST	200	19.0 ms	http://scorekeep.elasticbeanstalk.com/api/move/8N63LUQ6/2N56AC7L/PPMPBLJB	1
POST	200	19.0 ms	http://scorekeep.elasticbeanstalk.com/api/game/8N63LUQ6/2N56AC7L/users	1
POST	200	19.0 ms	http://scorekeep.elasticbeanstalk.com/api/game/8N63LUQ6/2N56AC7L/rules/TicTacToe	1
POST	200	19.0 ms	http://scorekeep.elasticbeanstalk.com/api/game/8N63LUQ6/2N56AC7L	0
POST	200	19.0 ms	http://scorekeep.elasticbeanstalk.com/api/game/8N63LUQ6/2N56AC7L	0
POST	200	19.0 ms	http://scorekeep.elasticbeanstalk.com/api/game/8N63LUQ6/2N56AC7L	0
POST	200	19.0 ms	http://scorekeep.elasticbeanstalk.com/api/game/8N63LUQ6/2N56AC7L	1
POST	200	19.0 ms	http://scorekeep.elasticbeanstalk.com/api/game/8N63LUQ6/2N56AC7L	1

トレースリスト

トレースの要約の確認に加え、URL/レスポンスコード/またはその他のデータでトレースの検索が可能

トレース画面では

- URLに対するトレース分布やレスポンス平均時間
- トレースリストが標準で表示される

URL

トレースリスト

Trace overview

Group by: URL

URL	Avg response time	% of Traces	Response
http://scorekeep.elasticbeanstalk.com/api/user	391 ms	4.76%	1 OK, 0 Throttled, 0 Errors, 0 Faults
http://scorekeep.elasticbeanstalk.com/api/session/8N63LUQ6	33.0 ms	4.76%	1 OK, 0 Throttled, 0 Errors, 0 Faults
http://scorekeep.elasticbeanstalk.com/api/session	90.5 ms	9.52%	2 OK, 0 Throttled, 0 Errors, 0 Faults

Trace list (21)

ID	Age	Method	Response	Response time	URL	Annotations
...f5f2df73	5.0 min	POST	200	391 ms	http://scorekeep.elasticbeanstalk.com/api/user	0
...cfe39980	5.0 min	PUT	200	33.0 ms	http://scorekeep.elasticbeanstalk.com/api/session/8N63LUQ6	0
...dd653e4c	5.0 min	POST	200	19.0 ms	http://scorekeep.elasticbeanstalk.com/api/session	0
...4765fec8	5.0 min	GET	200	162 ms	http://scorekeep.elasticbeanstalk.com/api/session	0
...84eeef29	4.7 min	POST	200	95.0 ms	http://scorekeep.elasticbeanstalk.com/api/move/8N63LUQ6/2N56AC7L/PPMPBLJB	1
...3ab33fdb	4.8 min	POST	200	95.0 ms	http://scorekeep.elasticbeanstalk.com/api/move/8N63LUQ6/2N56AC7L/PPMPBLJB	1
...237e0705	4.8 min	POST	200	295 ms	http://scorekeep.elasticbeanstalk.com/api/move/8N63LUQ6/2N56AC7L/PPMPBLJB	1
...86782227	4.9 min	POST	200	25.0 ms	http://scorekeep.elasticbeanstalk.com/api/game/8N63LUQ6/2N56AC7L/users	1
...fd82cc32	4.9 min	PUT	200	121 ms	http://scorekeep.elasticbeanstalk.com/api/game/8N63LUQ6/2N56AC7L/rules/Tic Tac Toe	1
...7ca2e05f	1.4 min	GET	200	14.0 ms	http://scorekeep.elasticbeanstalk.com/api/game/8N63LUQ6/2N56AC7L	0
...062ccac5	1.7 min	GET	200	12.0 ms	http://scorekeep.elasticbeanstalk.com/api/game/8N63LUQ6/2N56AC7L	0
...dc0ebe3c	1.9 min	GET	200	9.0 ms	http://scorekeep.elasticbeanstalk.com/api/game/8N63LUQ6/2N56AC7L	0
...524637dc	4.9 min	PUT	200	69.0 ms	http://scorekeep.elasticbeanstalk.com/api/game/8N63LUQ6/2N56AC7L	1
...fdf5bb67	4.9 min	POST	200	81.0 ms	http://scorekeep.elasticbeanstalk.com/api/game/8N63LUQ6	1

トレースリスト

セグメントを選択することで、トレースに含まれるセグメント・サブセグメントの実行状況が可視化

Trace overview

Group by: URL

URL	Avg response time	% of Traces	Response
http://scorekeep.elasticbeanstalk.com/api/user	391 ms	4.76%	1 OK, 0 Throttled, 0 Errors, 0 Faults
http://scorekeep.elasticbeanstalk.com/api/session/8N63LUQ6	33.0 ms	4.76%	1 OK, 0 Throttled, 0 Errors, 0 Faults
http://scorekeep.elasticbeanstalk.com/api/session	90.5 ms	9.52%	2 OK, 0 Throttled, 0 Errors, 0 Faults

Trace list (21)

ID	Age	Method	Response	Response time	URL	Annotations
f52d7f3	0 min	POST	200	391 ms	http://scorekeep.elasticbeanstalk.com/api/user	0
1e639960	0 min	PUT	200	33.0 ms	http://scorekeep.elasticbeanstalk.com/api/session/8N63LUQ6	0
16163e3c	0 min	POST	200	19.0 ms	http://scorekeep.elasticbeanstalk.com/api/session	0
4765fec8	0 min	GET	200	0 ms	http://scorekeep.elasticbeanstalk.com/api/session	0
84eeef29	4.7 min	POST	200	0 ms	http://scorekeep.elasticbeanstalk.com/api/move/8N63LUQ6/2N56AC7L/PPMPBLJB	1
3ab33fdb	4.8 min	POST	200	0 ms	http://scorekeep.elasticbeanstalk.com/api/move/8N63LUQ6/2N56AC7L/PPMPBLJB	1
237e0705	4.8 min	POST	200	0 ms	http://scorekeep.elasticbeanstalk.com/api/move/8N63LUQ6/2N56AC7L/PPMPBLJB	1
86782227	4.9 min	POST	200	0 ms	http://scorekeep.elasticbeanstalk.com/api/game/8N63LUQ6/2N56AC7L/users	1
fd82cc32	4.9 min	PUT	200	121 ms	http://scorekeep.elasticbeanstalk.com/api/game/8N63LUQ6/2N56AC7L/multiplayer-top	1
7ca2e05f	1.4 min	GET	200	14.0 ms	http://scorekeep.elasticbeanstalk.com/api/game/8N63LUQ6/2N56AC7L/multiplayer-top	1
062ccac5	1.7 min	GET	200	12.0 ms	http://scorekeep.elasticbeanstalk.com/api/game/8N63LUQ6/2N56AC7L/multiplayer-top	1
dc0e3e3c	1.9 min	GET	200	9.0 ms	http://scorekeep.elasticbeanstalk.com/api/game/8N63LUQ6/2N56AC7L/multiplayer-top	1
524637dc	4.9 min	PUT	200	69.0 ms	http://scorekeep.elasticbeanstalk.com/api/game/8N63LUQ6/2N56AC7L/multiplayer-top	1
fd5bb67	4.9 min	POST	200	81.0 ms	http://scorekeep.elasticbeanstalk.com/api/game/8N63LUQ6/2N56AC7L/multiplayer-top	1

クリック

セグメント

Scorekeep AWS::ElasticBeanstalk::Environment

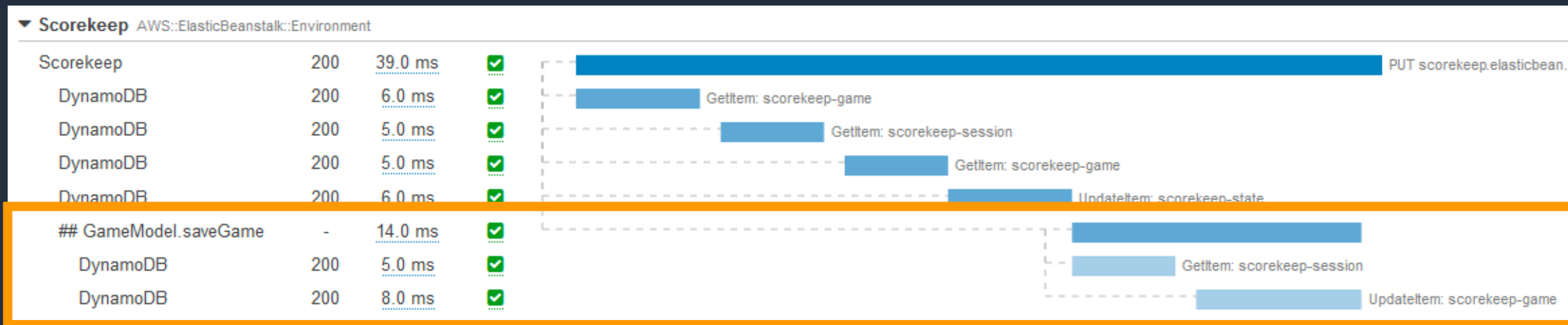
Segment	Count	Response time	Status
Scorekeep	200	39.0 ms	✓
DynamoDB	200	6.0 ms	✓
DynamoDB	200	5.0 ms	✓
DynamoDB	200	5.0 ms	✓
DynamoDB	200	6.0 ms	✓
## GameModel.saveGame	-	14.0 ms	✓
DynamoDB	200	5.0 ms	✓
DynamoDB	200	8.0 ms	✓

トレースリスト

サブセグメント

カスタムサブセグメント

監視・計測したい機能ごとにサブセグメントとして切り出すことが可能



カスタムサブセグメント

追加のサブセグメントを作成して、他のサブセグメントをグループ化やコードセクションのパフォーマンス測定、注釈とメタデータの記録が可能

セグメント詳細

トレースを選択することで、セグメント詳細の確認が可能

Trace overview

Group by: URL

URL	Avg response time	% of Traces	Response
http://scorekeep-elasticbeanstalk.com/api/user	391 ms	4.76%	1 OK, 0 Throttled, 0 Errors, 0 Faults
http://scorekeep-elasticbeanstalk.com/api/session/8N63LUQ6	33.0 ms	4.76%	1 OK, 0 Throttled, 0 Errors, 0 Faults
http://scorekeep-elasticbeanstalk.com/api/session	90.5 ms	9.52%	2 OK, 0 Throttled, 0 Errors, 0 Faults

Trace list (21)

ID	Age	Method	Response	Response time	URL	Annotations
f52df73	9 min	POST	200	391 ms	http://scorekeep-elasticbeanstalk.com/api/user	0
...
...
...
...
...
...
...
...
...
...
...
...
...
...
...
...
...
...
...
...

クリック



Scorekeep AWS::ElasticBeanstalk::Environment

Scorekeep	200	39.0 ms	✓
DynamoDB	200	6.0 ms	✓
DynamoDB	200	5.0 ms	✓
DynamoDB	200	5.0 ms	✓
DynamoDB	200	6.0 ms	✓
## GameModel.saveGame	-	14.0 ms	✓
DynamoDB	200	5.0 ms	✓
DynamoDB	200	8.0 ms	✓

クリック



セグメント詳細画面

Segment - Scorekeep

Overview Resources Annotations Metadata Exceptions

Segment ID: 69e25bb5bfe4c13f
Parent ID:
Name: Scorekeep
Origin: AWS::ElasticBeanstalk::Environment

Time

Start time: 2017-03-14 17:10:56.769 (UTC)
End time: 2017-03-14 17:10:56.814 (UTC)
Duration: 45.0 ms
In progress: False

Errors & Faults

Error: False
Fault: False

Request & Response

Request method: PUT
Request url: http://scorekeep-dev.pfj8pcpmww.us-west-2.elasticbeanstalk.com/api/game/1C6KM1DN/M0TTEU23/rules/TicTacToe
Request user_agent: Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/55.0.2883.87 Safari/537.36
Request x_forwarded_for:
Response status: 200

Close

トレースリスト



セグメント詳細

セグメント情報、HTTPリクエスト/レスポンスに関する情報、エラーや障害有無が表示される
カスタムサブセグメントの場合は注釈やメタデータの確認も可能

リソース	
Origin	AWS::ElasticBeanstalk::Environment

メタデータ	
Time	
Start time	2017-03-14 17:10:56.769 (UTC)
End time	2017-03-14 17:10:56.814 (UTC)
Duration	45.0 ms
In progress	False
Errors & Faults	
Error	False
Fault	False
Request & Response	
Request method	PUT
Request url	http://scorekeep-dev.pfj8pcpmww.us-west-2.elasticbeanstalk.com/api/game/1C6KM1DN/M0TTEU23/rules/Tic Tac Toe
Request user_agent	Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/55.0.2883.87 Safari/537.36
Request x_forwarded_for	
Response status	200

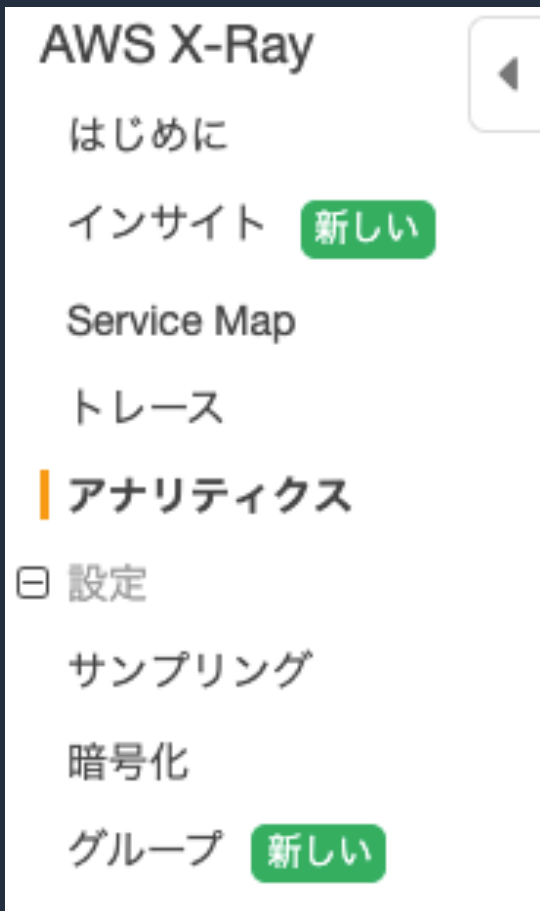
セグメント情報

時間情報

エラー情報

リクエスト
レスポンス詳細

アナリティクス

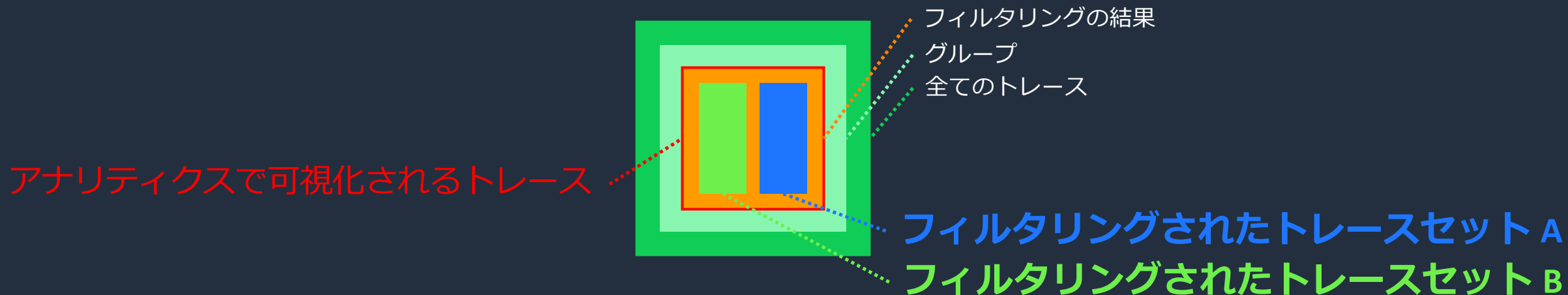


- フィルタリングされたトレースセット
- 応答時間の分布
- 時系列アクティビティ
- メトリクステーブル

フィルタリングされたトレースセット

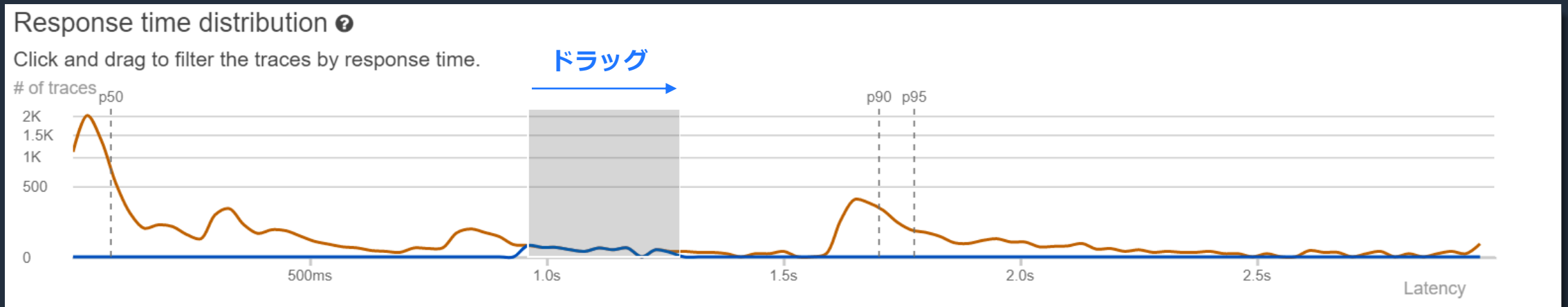
グラフに対してドラッグ テーブルに対して選択

- グラフやテーブルに対して操作を行いアナリティクス内に作成
- 最大で2つ (A と B) まで作成可能
- トレースセットを活用することで素早い分析が可能



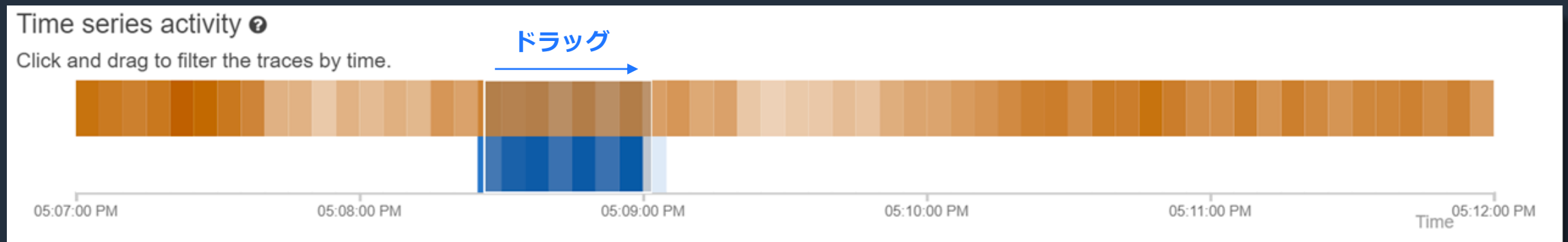
応答時間の分布

- 指定された応答時間におけるトレース数を示すグラフ
 - 横軸 応答時間
 - 縦軸 トレース数
- ドラッグすることで **フィルタリングされたトレースセット** を作成できる



時系列アクティビティ

- 指定された時間のトレース数
 - 横軸 時間
 - 濃さ トレース数 (濃い方が多い)
- ドラッグすることで **フィルタリングされたトレースセット** を作成できる



メトリクステーブル

HTTP METHOD	COUNT	%
GET	713	98.89%
POST	5	0.69%
-	3	0.42%

HTTP STATUS CODE	COUNT	%
200	719	99.72%
500	2	0.28%

- Http Status Code や Response Time Root Cause など事前に定義された属性に基づき分類されたトレース
- テーブルの行をクリックすることでフィルタリングされたトレースセットとして設定可能

インサイト

AWS X-Ray

はじめに

インサイト 新しい

Service Map

トレース

アナリティクス

設定

サンプリング

暗号化

グループ 新しい

- インサイトの有効化
- フィルタによる問題の絞り込み
- サービスマップからインサイト表示
- 問題の概要と調査

インサイトの有効化

アプリケーションの異常を自動で検出

- インサイトの有効化は、グループを選択しインサイトの有効化を選択する
- 初期状態ではグループにはDefaultが登録されているため、まずはDefaultグループのインサイト有効化がおすすめ
- Amazon EventBridgeを利用した通知の有効化も同時に選択が可能

インサイト

X-Ray Insights はアプリケーションパフォーマンスの異常を自動的に検出します。しきい値を手動で設定したり、X-Ray で計測するためにアプリケーションを変更したりする必要はありません。Insights の料金の詳細については、AWS X-Ray Insights の料金をご参照ください。 [詳細はこちら](#)

インサイトの有効化

通知の有効化 Amazon EventBridge を使用してインサイトイベントを配信します。EventBridge でのデータ保護の詳細をご覧ください。 [詳細はこちら](#)

フィルタによる問題の絞り込み

期間、グループ、状態(State)を指定することでインサイトをフィルタすることができる

インサイト フィードバックの提供

X-Ray Insights はアプリケーションパフォーマンスの異常を自動的に検出します。しきい値を手動で設定したり、X-Ray で計測するためにアプリケーションを変更したりする必要はありません。 [詳細はこちら](#)

開始 終了 グループ ▼ 状態 ▼

説明 ▼	所要時間 ▼	根本原因サービス ▼	異常なサービス ▼	グループ ▼	開始時間 ▲
<p>Overall, 1.7% of the client requests failed due to faults and 0.6% of the requests to PetSearch (AWS::ECS::Fargate) failed due to faults.</p> <p><input type="button" value="Closed"/> <input type="button" value="Fault"/></p>	17 分 52 秒	PetSearch (AWS::ECS::Fargate)	PetSearch (AWS::ECS::Fargate)	Default	12月 27日 2022, 02:13

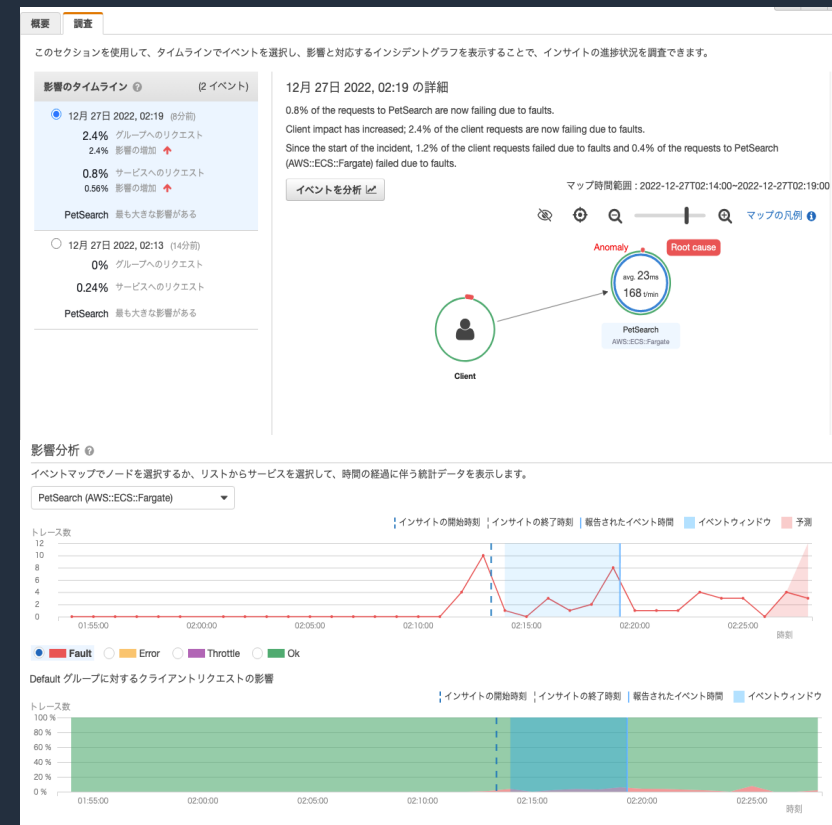
サービスマップからインサイト表示

- サービスマップ上のサービスからインサイト詳細情報へ遷移することが可能
- サービスマップ上でサービスの状態を確認し、インサイトの有無、インサイトの概要を確認することが可能

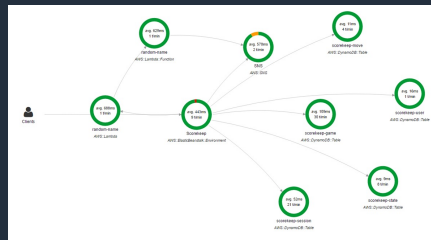
The screenshot displays the AWS Service Map interface for a service named 'PetSearch'. The main map shows a central 'PetSearch' node (AWS::ECS::Fargate) with a status of 'Ok 100%' and a traffic volume of '3 t/min'. It is connected to several other services: 'petlistadoptions' (AWS::ECS::Fargate, 'Ok 100%', '159 t/min'), 'Services-ddbpetaoption7B7Cf' (AWS::DynamoDB::Table, 'Ok 100%', '170 t/min'), and 'AWSSimpleSystemsM' (AWS::AWSSimpleSystemsM, 'Ok 100%', '0.03 t/min'). Two 'Client' nodes are also shown, one with 'Fault 2%' and '9 t/min' traffic, and another with 'Fault 9%' and '9 t/min' traffic. A red box highlights the 'Insight' button on the 'Services-ddbpetaoption7B7Cf' node. To the right, a 'サービスの詳細' (Service Details) panel is open, showing the name 'PetSearch' and type 'AWS::ECS::Fargate'. Below this, an 'インサイト' (Insight) section contains a summary of request failures: '0.7% requests to PetSearch and 1.9% requests to Default group are failing due to faults. Since the start of the incident, 1.8% of the client requests failed due to faults and 0.6% of the requests to PetSearch (AWS::ECS::Fargate) failed due to faults. 12月 27日 2022, 02:13 (27分前)'. A red box highlights the '詳細なインサイトを表示' (Show detailed insights) button. Below the insight text is a 'レスポンスディストリビューション' (Response Distribution) graph showing a sharp peak at 0 seconds and a very low tail extending to 3.0 seconds. At the bottom of the panel, there are 'レスポンスステータス' (Response Status) filters set to 'トラフィック' (Traffic) and a '閉じる' (Close) button.

問題の概要と調査

- インサイトの詳細は概要と調査で構成
- 概要タブでは、問題の概要、根本原因、影響の確認が可能
- 調査タブでは、より詳細な影響タイムライン、影響分析を確認可能



AWS X-Ray によるアプリケーション分析



特定のサービスに関連する
トレースリストの確認

サービスマップ
トレースデータの可視化

特定のサービスに関連する
トレースリストを分析

URL	Req. response time	% of Total	Response
http://localhost:8080/compensation	361 ms	2.1%	200, 200, 200, 200, 200
http://localhost:8080/compensation/employees	318 ms	1.9%	200, 200, 200, 200, 200
http://localhost:8080/compensation/employees/1	80.5 ms	0.5%	200, 200, 200, 200, 200

URL	Req. response time	% of Total	Response
http://localhost:8080/compensation	361 ms	2.1%	200
http://localhost:8080/compensation/employees	318 ms	1.9%	200
http://localhost:8080/compensation/employees/1	80.5 ms	0.5%	200

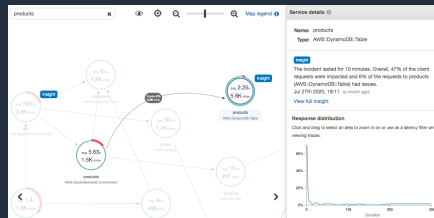
トレースリスト
トレースの詳細な情報

分析対象のトレースリストの
詳細を確認



アナリティクス
トレース分析用ツール

パフォーマンスの異常や
アプリケーションの問題を継続確認



インサイト
継続的な問題検知

アジェンダ

AWS X-Ray とは

AWS X-Ray の概念

AWS X-Ray によるアプリケーション分析

AWS X-Ray によるデータ収集の仕組み

AWS X-Ray の利用方法

AWS モニタリングサービスとの連携

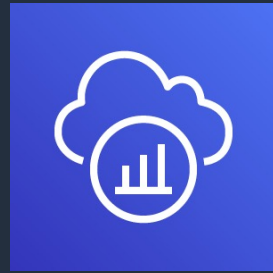
料金・注意点

まとめ

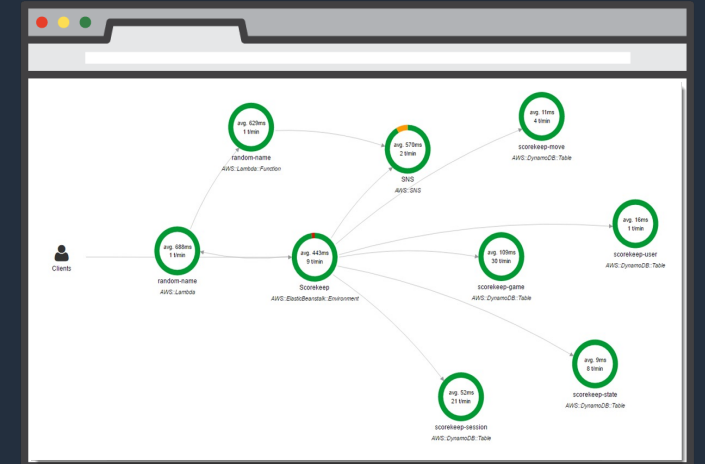
AWS X-Ray のデータ収集の仕組み



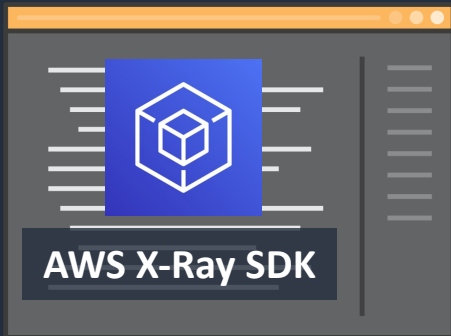
AWS X-Ray デーモン



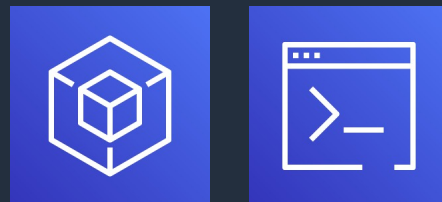
AWS X-Ray API



AWS X-Ray コンソール



アプリケーション



AWS SDK

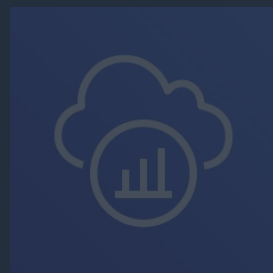
AWS CLI

スクリプトやツール

AWS X-Ray のデータ収集の仕組み



AWS X-Ray デーモン

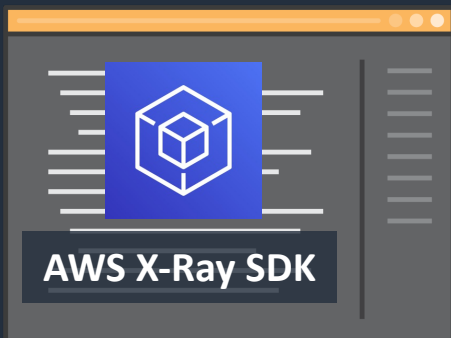


AWS X-Ray API

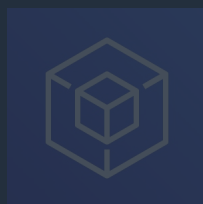


AWS X-Ray コンソール

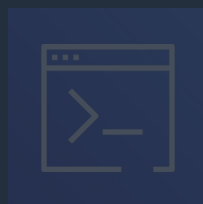
アプリケーションで作成したセグメントデータを X-Ray デーモンに送信



アプリケーション



AWS SDK



AWS CLI

スクリプトやツール

AWS X-Ray SDK



Java

- トレースデータを作成して X-Ray デーモンに送信するためのクラスとメソッドを提供



Node.js

- 受信リクエスト
- AWS SDK クライアント (Amazon DynamoDB, Amazon S3, ...)



Python

- 送信 HTTP 呼び出し



.Net

- SQL クエリ
- カスタムサブセグメント



Ruby

- ...



Go



Auto-Instrumentation Agent

- Auto-Instrumentation Agentを利用することで、既存のアプリケーションを修正することなくトレースデータの収集が可能に



Java

- **一般利用可能**
- Spring Boot で構築したウェブアプリケーションやサーバーベースのマイクロサービスなどに対応
- Apache HTTP クライアントや AWS SDK クライアント、および JDBC ベースの SQL クライアントが生成した、すべてのダウンストリームリクエストがキャプチャされる
- マルチスレッドアプリケーションでも自動的に実行



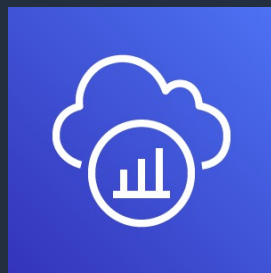
.NET

- **ベータ版**で利用可能
- インターネットインフォメーションサービス (IIS) をターゲットとする .NET アプリケーションや .NET Core アプリケーションに対応
- Entity Framework、Entity Framework Core、および .NET Core アプリケーション用 SqlConnection のトレースをサポート

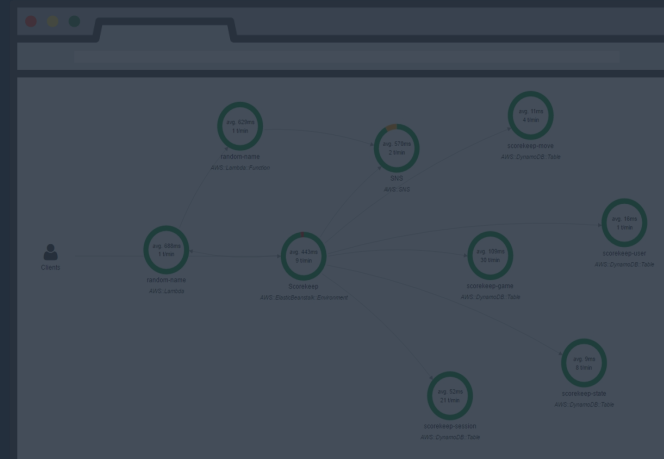
AWS X-Ray のデータ収集の仕組み



AWS X-Ray デーモン



AWS X-Ray API

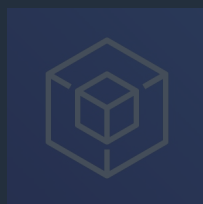


AWS X-Ray コンソール

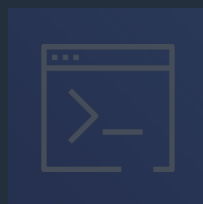
**X-Ray デーモンはセグメントデータを
バッファリングし X-Ray API に定期的に転送**



アプリケーション



AWS SDK



AWS CLI

スクリプトやツール

AWS X-Ray デーモン

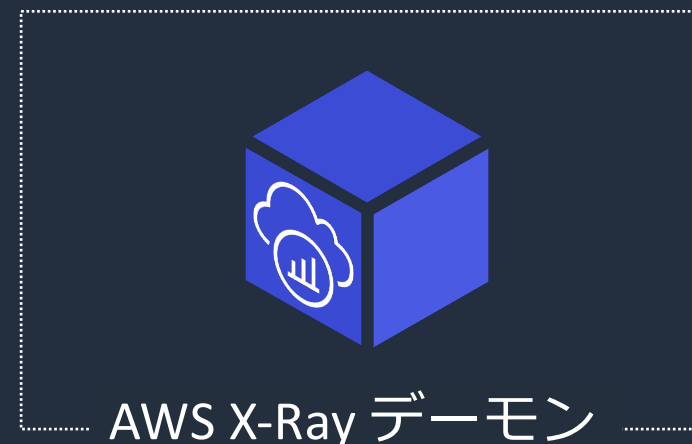
未加工のセグメントデータを収集して AWS X-Ray API に送信

UDP ポート 2000 (デフォルト) をリッスン

受信したデータをバッファリングして X-Ray API に送信する

実行方法

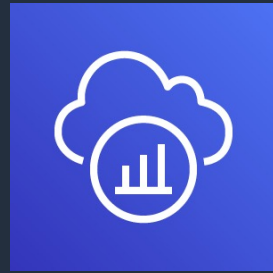
- | | |
|------------------------|------------------------------|
| •Linux | 実行可能ファイル, RPMパッケージ, DEBパッケージ |
| •OS X | 実行可能ファイル |
| •Windows | 実行可能ファイル, サービス |
| •AWS Lambda | コンソールから設定 |
| •AWS Elastic Beanstalk | コンソールから設定 |
| •Amazon ECS | Sidecar として実行 など |



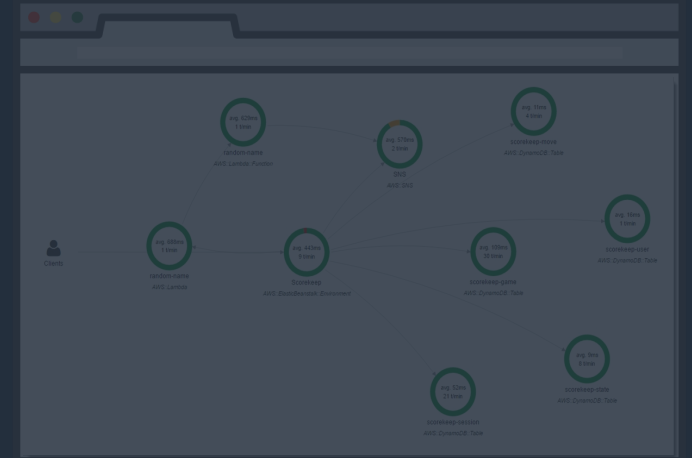
AWS X-Ray のデータ収集の仕組み



AWS X-Ray デーモン



AWS X-Ray API



AWS X-Ray コンソール

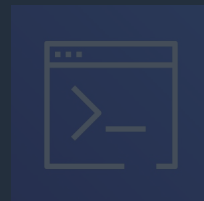
収集されたセグメントデータから
トレースを作成



アプリケーション



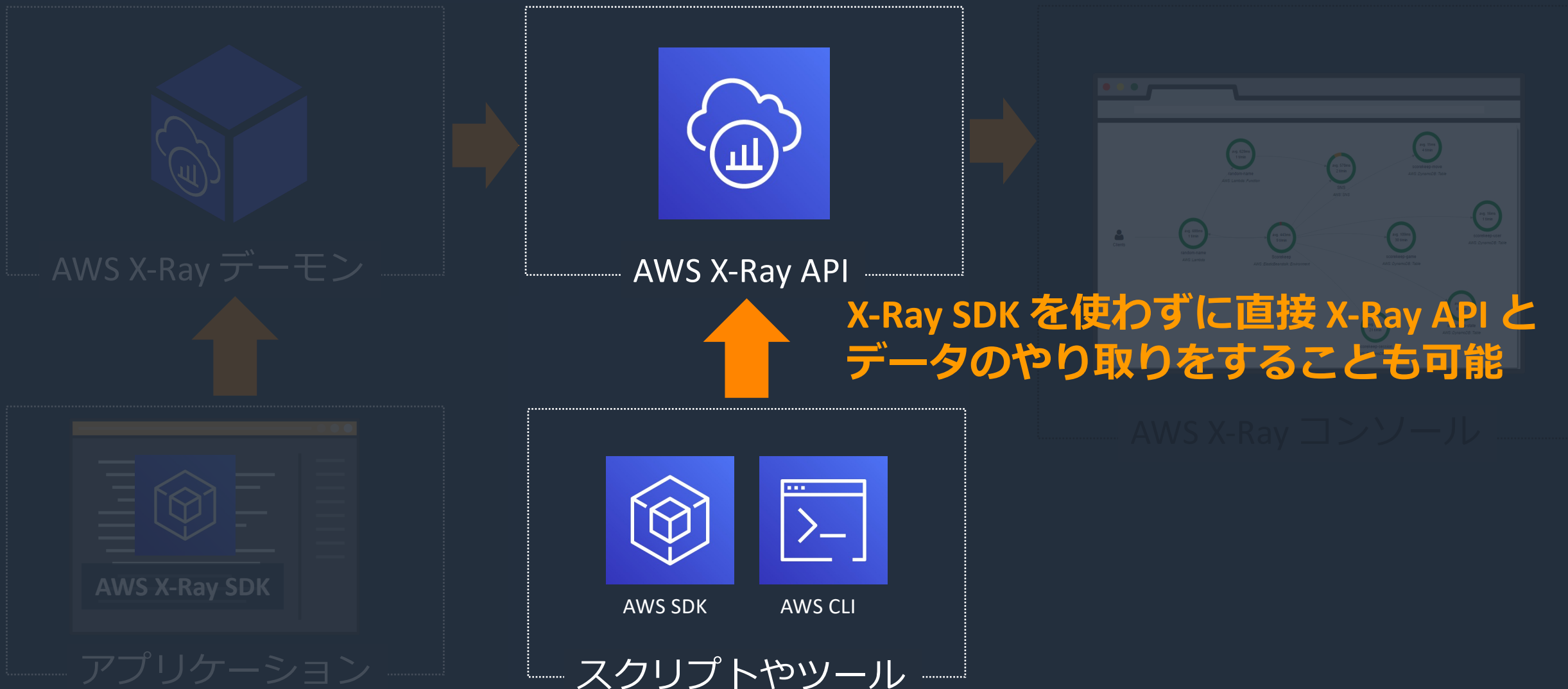
AWS SDK



AWS CLI

スクリプトやツール

AWS X-Ray のデータ収集の仕組み



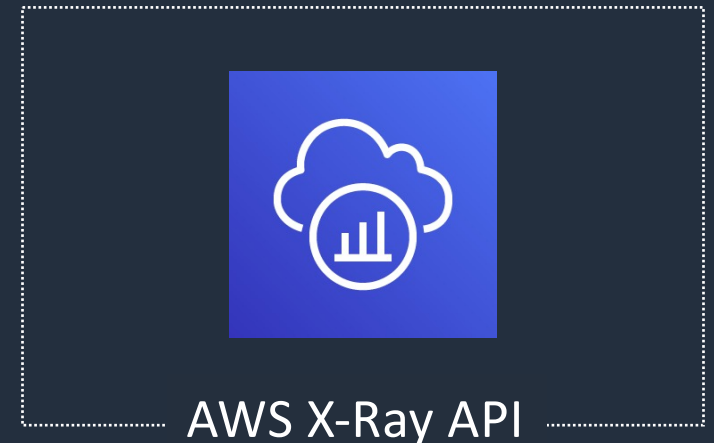
AWS X-Ray API

AWS X-Ray の全ての機能にアクセス可能

- トレースの追加・取得・更新
- グループの作成・取得・更新・削除
- サンプリングルールの作成・取得・更新・削除

利用方法

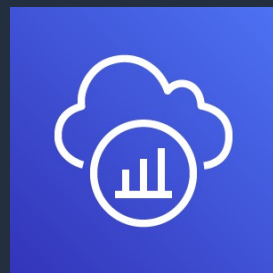
- HTTPS 経由
- AWS SDK 経由
- AWS Command Line Interface 経由



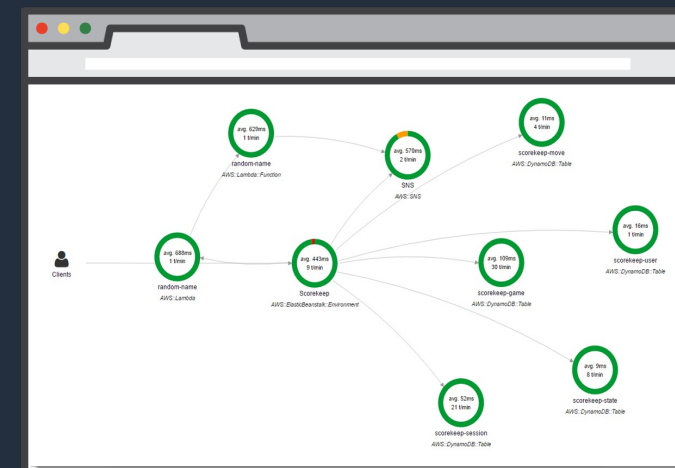
AWS X-Ray のデータ収集の仕組み



AWS X-Ray デーモン



AWS X-Ray API

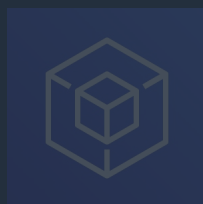


AWS X-Ray コンソール

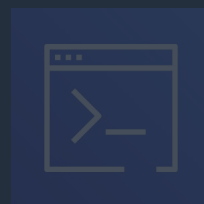
トレースをコンソールで可視化
分析・デバッグを行う



アプリケーション



AWS SDK



AWS CLI

スクリプトやツール

アジェンダ

AWS X-Ray とは

AWS X-Ray の概念

AWS X-Ray によるアプリケーション分析

AWS X-Ray によるデータ収集の仕組み

AWS X-Ray の利用方法

AWS モニタリングサービスとの連携

料金・注意点

まとめ

AWS X-Rayが対応しているサービス

多くのAWSサービスとの連携をサポート

- Amazon API Gateway
- Amazon EC2
- AWS App Mesh
- AWS App Runner
- AWS AppSync
- AWS CloudTrail
- AWS Config
- AWS Elastic Beanstalk
- Elastic Load Balancing
- Amazon EventBridge
- AWS Lambda
- Amazon SNS
- AWS Step Functions
- Amazon SQS
- Amazon S3

https://docs.aws.amazon.com/ja_jp/xray/latest/devguide/xray-services.html



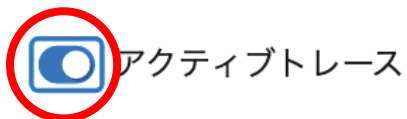
AWS Lambda

AWS Lambda 関数の設定から有効化が可能

- 多少のメモリ消費量の増加あり
- HTTP 呼び出し・注釈・メタデータの追加などにはAWS X-Ray SDK が必要
- 関数にはトレースデータを X-Ray にアップロードするためのアクセス許可が必要

AWS X-Ray 情報

有効にして [保存] を選択すると、Lambda コンソールは実行ロールのアクセス許可を確認します。実行ロールに必要なアクセス許可がない場合、Lambda コンソールはロールへの追加を試みます。



i 必要なアクセス許可が見つかりませんでした。Lambda コンソールは実行ロールに追加しようとします。実行ロールが必要なアクセス許可を拒否していないことを確認してください。



Amazon API Gateway

ステージの設定から X-Ray トレースの有効化が可能

- HTTP リクエストにトレースヘッダーを追加
- AWS X-Ray コンソールで定義したサンプリングルール

設定 **ログ/トレース** ステージ変数 SDK の生成 エクスポート デプロイ履歴 ドキュメント履歴 Canary

ステージのロギングおよびトレース設定を指定します。

CloudWatch 設定 [詳細はこちら](#)

CloudWatch ログ エラーと情報ログ ⓘ
すべてのイベントのロギングが有効

詳細 CloudWatch メトリクスを有効化 ⓘ

カスタムアクセスのログ記録

アクセスログの有効化

X-Ray トレース [詳細はこちら](#)

X-Ray トレースの有効化 ⓘ [X-Ray サンプリングルールの設定](#)

変更を保存

Amazon EC2



インスタンス起動時にユーザーデータのスクリプトを実行してデーモンを起動

- AWS X-Ray デーモンが実行可能な OS
 - Amazon Linux
 - Ubuntu
 - Windows Server (2012 R2 以降)

```
#!/bin/bash
```

Amazon Linux の例

```
curl https://s3.dualstack.us-east-2.amazonaws.com/aws-xray-assets.us-east-2/xray-daemon/aws-xray-daemon-3.x.rpm -o /home/ec2-user/xray.rpm
```

```
yum install -y /home/ec2-user/xray.rpm
```

※オンプレミス環境でも実行可能

<https://docs.aws.amazon.com/xray/latest/devguide/xray-daemon-local.html>



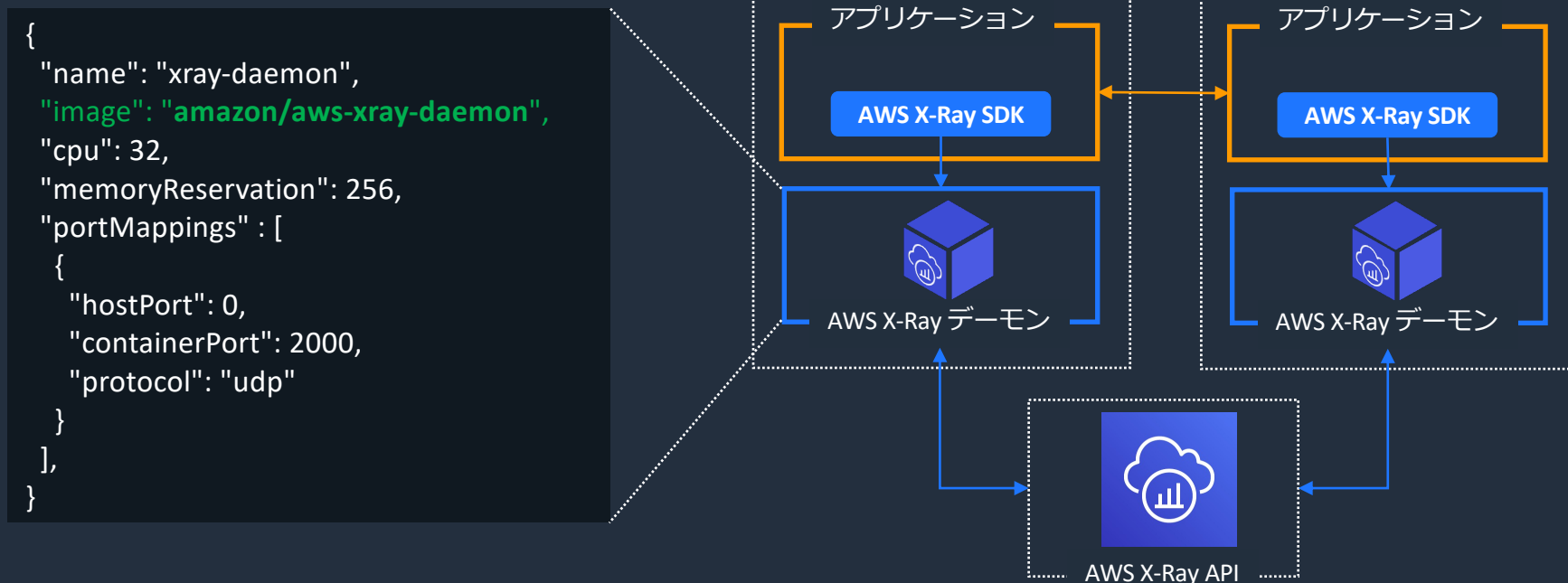


Amazon ECS

AWS X-Ray デーモンの Docker コンテナを Sidecar として設定

- ネットワークモードによってコンテナ周りのタスク定義の設定が異なる
 - ブリッジ (デフォルト) リンクを作成し 環境変数 `AWS_XRAY_DAEMON_ADDRESS` に 向き先を設定
 - `awsvpc`

特に設定は必要ない



AWS App Runner




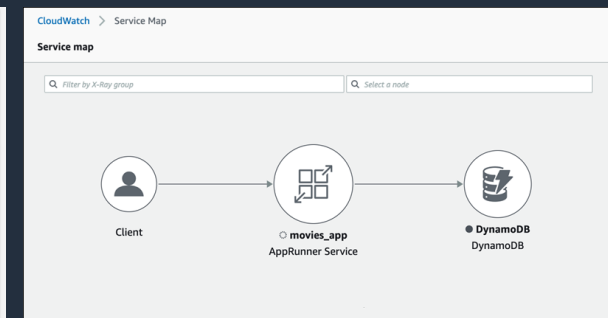
App Runnerサービス設定でトレースの有効化が可能

- アプリケーションが既に OpenTelemetry を利用してトレースを取得する実装となっている場合はアプリケーションに変更を加えることなく、App Runner サービスの X-Ray トレースを有効化することが可能
- X-RayではAWS Distro for OpenTelemetry (ADOT)との連携もサポート
- その後トレースの有効化をサービス設定で指定、サービス作成後も設定可能

▼ 可観測性
可観測性ツールを設定します。

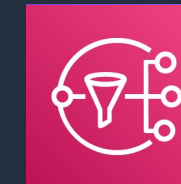
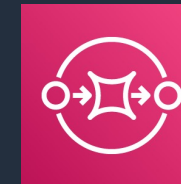
可観測性ツールを設定

 **Tracing with AWS X-Ray**
アプリケーションの X-Ray トレース。料金については、「AWS X-Ray の料金」を参照してください。



Amazon SQS, Amazon SNS

非同期アプリケーションのトレース



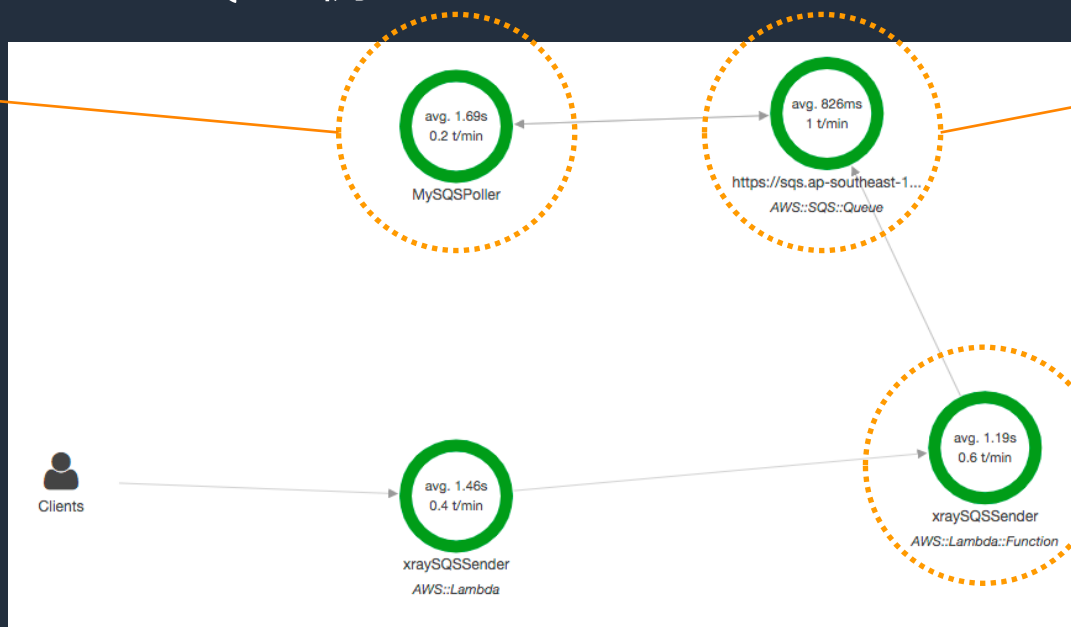
- Amazon SQS や Amazon SNS とも統合

アクティブ計測	受信リクエストをサンプリングして計測	Amazon SNS (スタンダードタイプのみ) , Amazon API Gateway, AWS Lambda
パッシブ計測	別のサービスですでにサンプリングされているリクエストを計測	Amazon SQS

* Amazon API Gateway と AWS Lambda Amazon SNSはパッシブ計測も可能

Amazon SQS の例

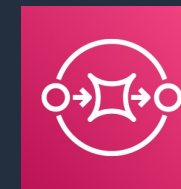
コンシューマー



Amazon SQS

AWS Lambda

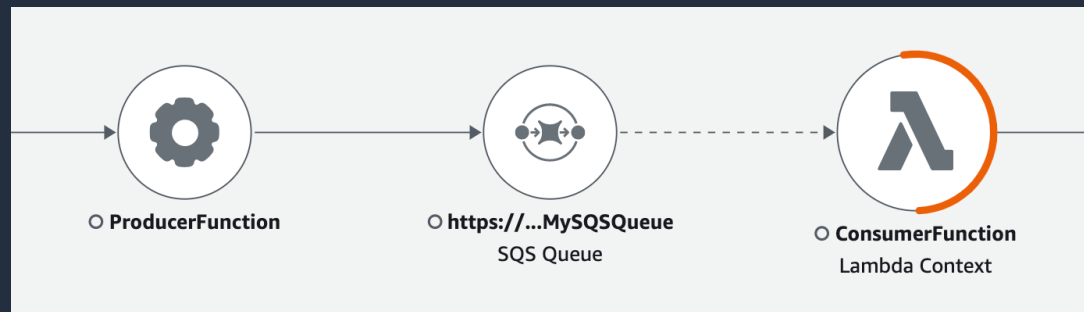
Amazon SQS と AWS Lambda による イベント駆動型アプリケーションのトレース



Amazon SQS と AWS Lambda を使用して構築されたイベント駆動型アプリケーションの トレースデータの取得が可能

- エンドツーエンドのフローを分析およびデバッグが可能
- Amazon SQS キューで待機していた時間など、パフォーマンス分析や異常原因の特定が可能
- AWS Lambda 関数に影響している Amazon SQS メッセージの特定が可能

AWS Lambda 関数で AWS X-Ray を有効にしている場合、アプリケーションコードやサービスの設定の変更は不要



アジェンダ

AWS X-Ray とは

AWS X-Ray の概念

AWS X-Ray によるアプリケーション分析

AWS X-Ray によるデータ収集の仕組み

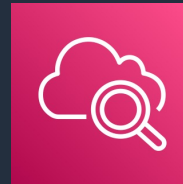
AWS X-Ray の利用方法

AWS モニタリングサービスとの連携

料金・注意点

まとめ

Amazon CloudWatchとの連携

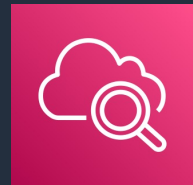


- AWS X-RayはAmazon CloudWatch と統合され、CloudWatch ServiceLens、CloudWatch RUM、および CloudWatch Synthetics をサポート
- Amazon CloudWatchコンソール上からサービスマップの確認や、トレースへのクエリ実行が可能

The screenshot shows the Amazon CloudWatch console interface for the 'PetSearch' application. The main view is the 'Service Map' (サービスマップ), which visualizes the application's architecture. A central node represents the 'PetSearch ECS Fargate' service. It is connected to several other services: 'SimpleSy...agement' (Simple Systems Management), 'StepFunct...13modiEmal' (Step Functions), 'petlistadoptions' (ECS Fargate), and 'S3'. The interface includes a search bar with the text 'X-Ray グループでフィルタリング' and a 'マップ表示' (Map View) button. The sidebar on the left contains navigation options such as 'ダッシュボード' (Dashboard), 'アラーム' (Alarms), 'ログ' (Logs), 'メトリクス' (Metrics), and 'X-Ray トレース' (X-Ray Traces).

The screenshot shows the Amazon CloudWatch console interface for the 'PetSearch' application, specifically the 'Traces' (トレース) view. The main view displays search filters and a 'クエリの実行' (Execute Query) button. Below the button, there is a section for 'クエリリファイナー' (Query Refiner) with a dropdown menu set to 'ノード' (Node) and a search input field containing 'ノードを検索'. The interface also shows a list of traces with a '314取得されたトレース' (314 traces retrieved) indicator. The sidebar on the left contains navigation options such as 'ダッシュボード' (Dashboard), 'アラーム' (Alarms), 'ログ' (Logs), 'メトリクス' (Metrics), and 'X-Ray トレース' (X-Ray Traces).

Amazon CloudWatch ServiceLens



メトリクス、ログ、トレースを相互に関連付け、サービスのエンドツーエンドビューを提供し、パフォーマンスのボトルネックを迅速に特定し、影響を受けるユーザーを特定するのに役立つ

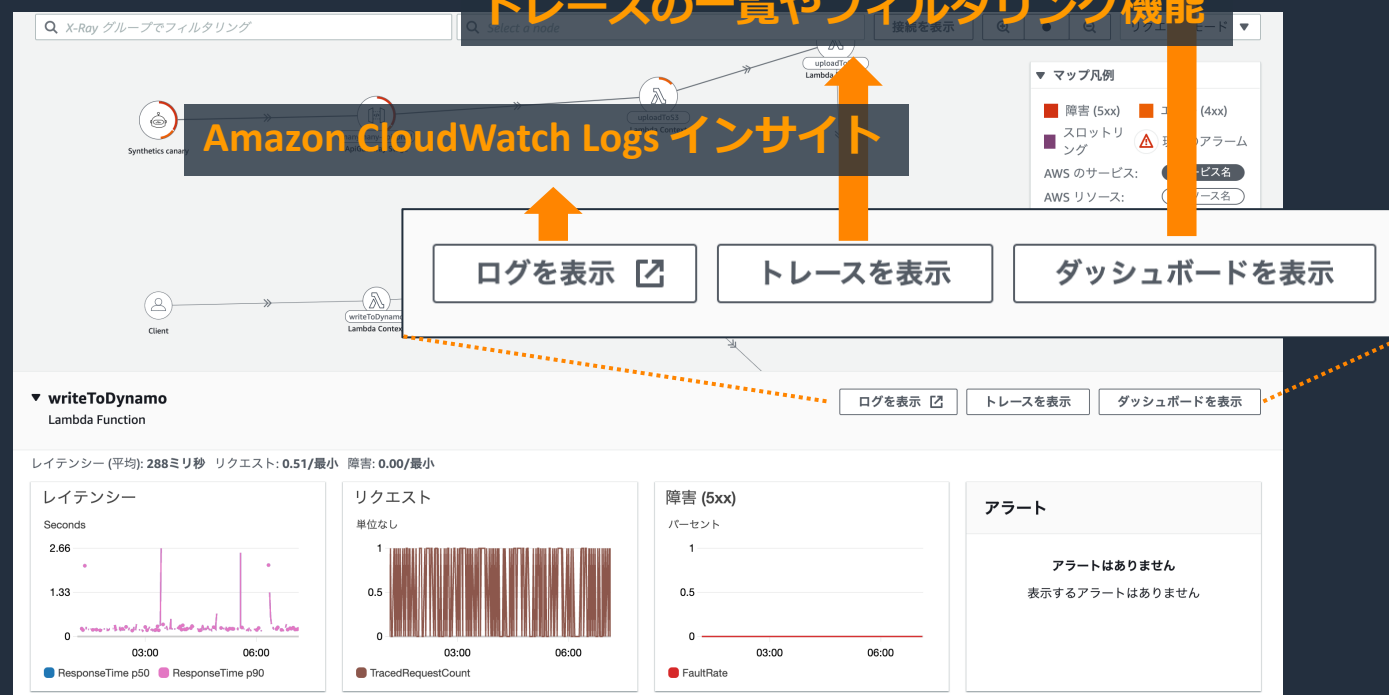
- トレース ← **AWS X-Ray のデータを利用**
- メトリクス
- ログ
- アラート

メトリクスの詳細やノードマップ

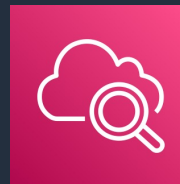
トレースの一覧やフィルタリング機能

サービスマップ

メトリクスグラフ、アラート



Amazon CloudWatch RUM



- AWS X-Rayとの統合を有効にすると、Amazon CloudWatch RUM がHTTP リクエストに X-Ray トレースヘッダーを追加し、HTTP リクエストの X-Ray セグメントを記録
- これによりAmazon CloudWatch コンソールでユーザーセッションのトレースとセグメントを確認できる
- アプリケーションモニターを追加する際にトレースの有効化が可能

▼ アクティブトレース - オプション

AWS X-Ray を使用してアプリケーションを計測し、トレース、セグメント、サービスマップを表示します。 [料金情報を表示します。](#)

AWS X-Ray でサービスを追跡する

AWS X-Ray と CloudWatch RUM を使用して、アプリケーションのエンドユーザーから開始するエンドツーエンドのリクエストパスを分析およびデバッグし、AWS のすべてのマネージドサービスを通じてエンドユーザーに影響を与えるレイテンシーの傾向やエラーを特定します。

Client: RUM

その他の利点

- ✓ AWS X-Ray および CloudWatch ServiceLens サービスマップでアプリケーションを表示します。
- ✓ エンドユーザーのリクエスト、セッション、エラーのトレースとセグメントを表示します。
- ✓ AWS X-Ray 分析を使用して傾向を表示します。

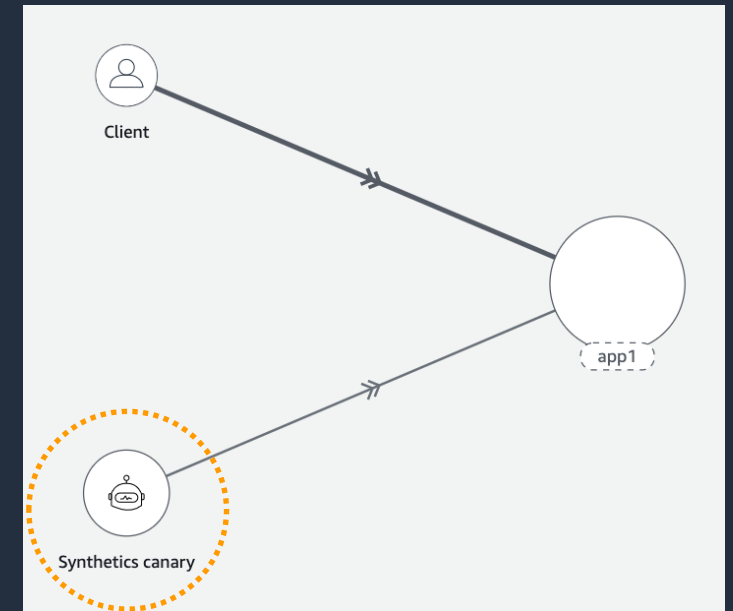
Amazon CloudWatch Synthetics



Canaryを作成してエンドポイントを監視

- Canaryとは
 - 特定のエンドポイントにアクセス
 - 定期的なスケジュールで実行
 - ロード時間データとUIのスクリーンショットを保存
 - 閾値によってアラームを作成
- Amazon CloudWatch ServiceLens、AWS X-Rayと統合

サービスマップ



Canary実行



AWS Distro for OpenTelemetry (ADOT)

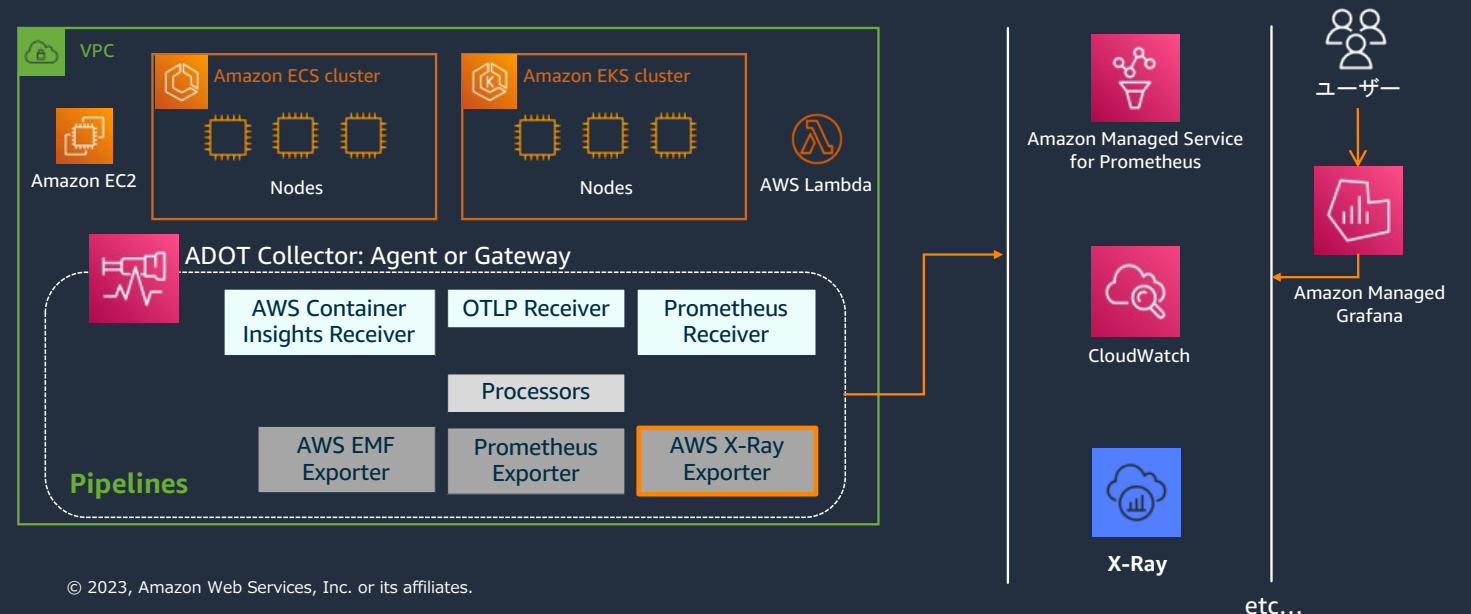
OpenTelemetryとAWS X-Rayを連携しトレースのモニタリングや可視化、分析が可能
AWS X-RayはAWSサービスのAWS Distro for OpenTelemetry (ADOT) との連携をサポートしている

AWS Distro for OpenTelemetry (ADOT) :

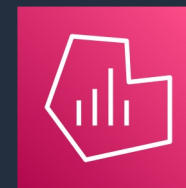


- 安全で本番環境に適したオープンソースディストリビューション
- OpenTelemetry へのアップストリームでの貢献
- Amazon CloudWatch、AWS X-Ray、パートナーソリューション等を含む Exporterを用意

AWS Distro for OpenTelemetry の Auto-Instrumentationで従来のX-Rayの実装が楽に
<https://aws-otel.github.io/docs/getting-started/python-sdk/trace-auto-instr>
<https://aws.amazon.com/jp/blogs/containers/tracing-an-aws-app-runner-service-using-aws-x-ray-with-opentelemetry/>



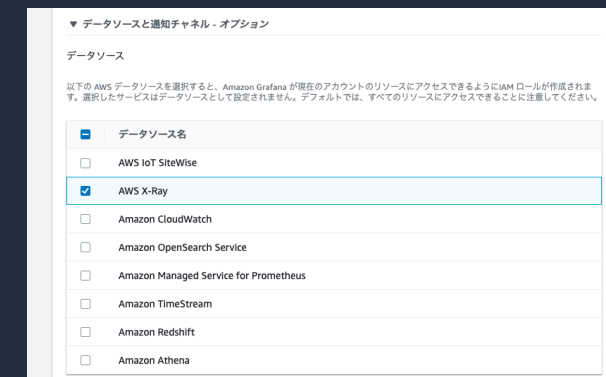
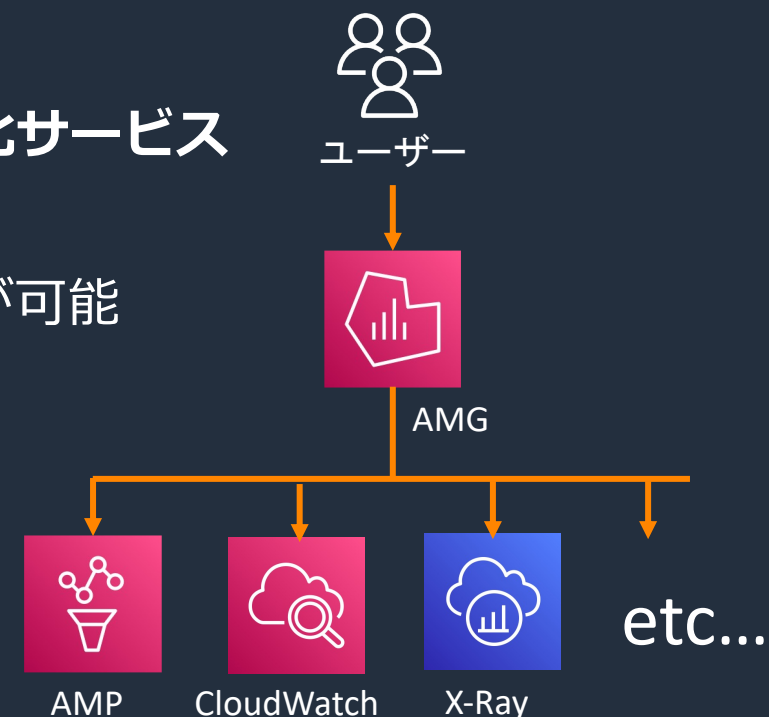
Amazon Managed Grafana (AMG)



Amazon Managed Grafana (AMG) :
Grafana Labs と連携して開発された、フルマネージドなデータ可視化サービス

データソースとして AWS X-Ray が標準でサポートされており、
Grafana ダッシュボード上でサービスマップやトレースの確認が可能

- 複数のデータソースからのメトリクスやログ、トレースを一元的に可視化
- Grafana の構築、スケーリング、パッチ適用などのメンテナンスは AWS が実施
- AWS Identity Center と統合されており、ユーザー毎の権限でダッシュボードとデータソースにシームレスなアクセスを実現
- AWS アカウントとリソースの検出
- API を利用して既存の Grafana 環境からクエリとダッシュボードをインポート可能



アジェンダ

AWS X-Ray と

AWS X-Ray の概念

AWS X-Ray によるアプリケーション分析

AWS X-Ray によるデータ収集の仕組み

AWS X-Ray の利用方法

AWS モニタリングサービスとの連携

料金・注意点

まとめ

料金 (2023/2 現在)

記録、取得、スキャンされたトレースの数に基づいて料金が発生
永続的な無料利用枠

- 毎月 トレースの記録は 10 万回まで無料
- 毎月 トレースの取得とスキャンは合わせて 100 万回まで無料

トレースの記録	100 万件あたり 5.00USD
トレースの取得	100 万件あたり 0.50USD
トレースのスキャン	100 万件あたり 0.50USD

* 無料利用枠を超えた分

* リージョン アジアパシフィック (東京)

<https://aws.amazon.com/xray/pricing/>

注意点

サンプリングについて

- デフォルトではサンプリングされる
- 100% トレースする設定も可能だが、料金とパフォーマンスへの影響を事前に確認する

データの完全性は保証されない

- 監査やコンプライアンスのツールとして使用することはできない

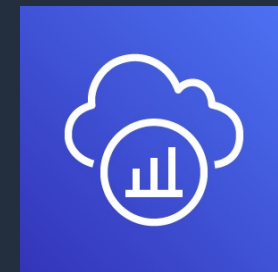
直近 30 日間の追跡データが保存

- 過去 30 日間の追跡データに対してクエリを実行可能

アジェンダ

- AWS X-Ray とは
- AWS X-Ray の概念
- AWS X-Ray によるアプリケーション分析
- AWS X-Ray によるデータ収集の仕組み
- AWS X-Ray の利用方法
- AWS モニタリングサービスとの連携
- 料金・注意点
- **まとめ**

まとめ



AWS X-Ray

有用なツール群

- ・ サービスマップ : サービス間の依存関係をわかりやすく可視化
- ・ トレースリスト : セグメントやサブセグメントの実行状況を可視化
- ・ アナリティクス : インタラクティブな操作によって素早い分析が可能
- ・ インサイト : パフォーマンスやアプリケーション上での問題を継続的に自動で検知

利用の容易さ

- ✓ AWS X-Ray SDK や AWS X-Ray デーモンによってアプリケーションに簡単に組み込める
- ✓ 多くの AWS サービスと統合されていて簡単にセットアップ可能

本資料に関するお問い合わせ・ご感想

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

<https://aws.amazon.com/jp/premiumsupport/>

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）

<https://console.aws.amazon.com/support/home#/case/create?issueType=customer-service>

具体的な案件に対する構成相談は、後述する個別相談会をご活用ください



ご感想はTwitterへ！ハッシュタグは以下をご利用ください
#awsblackbelt

その他コンテンツのご紹介

ウェビナーなど、AWSのイベントスケジュールをご参照いただけます

<https://aws.amazon.com/jp/events/>

ハンズオンコンテンツ

<https://aws.amazon.com/jp/aws-jp-introduction/aws-jp-webinar-hands-on/>

AWS 個別相談会

AWSのソリューションアーキテクトと直接会話いただけます

<https://pages.awscloud.com/JAPAN-event-SP-Weekly-Sales-Consulting-Seminar-2021-reg-event.html>



Thank you!