



Amazon EventBridge Scheduler

櫻谷 広人

Partner Solutions Architect
2023/8

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 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 2023 年 8 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。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)

自己紹介

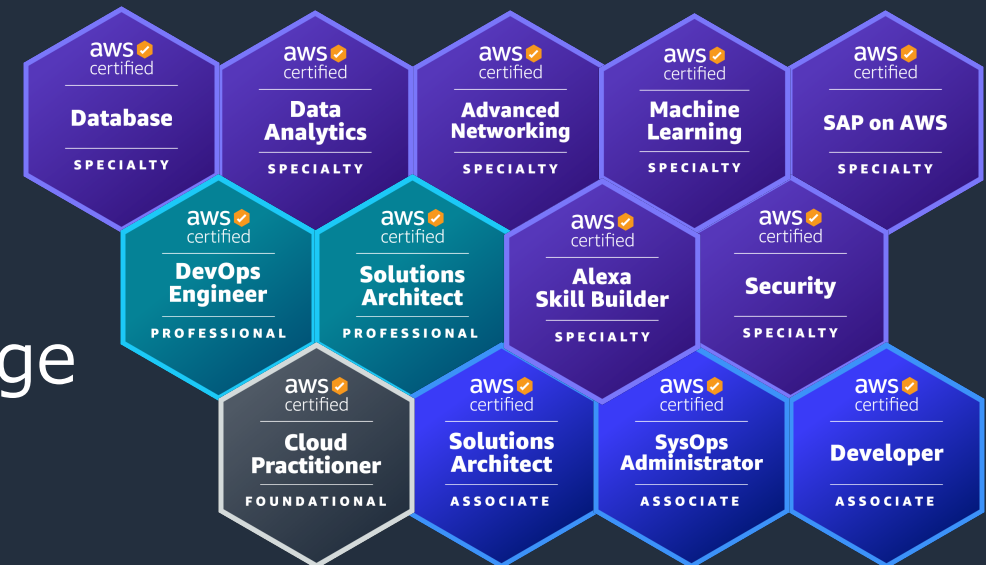
名前：櫻谷 広人 (Hiroto Sakuraya)

所属：AWS Technology Partnerships

SaaS, Partner Solutions Architect

経歴：主にバックエンドエンジニアとして Web サービスやネイティブアプリの開発に従事。前職では CtoC のスタートアップで 執行役員 CTO を務める。

好きな AWS サービス：Amazon EventBridge



本セミナーの対象者

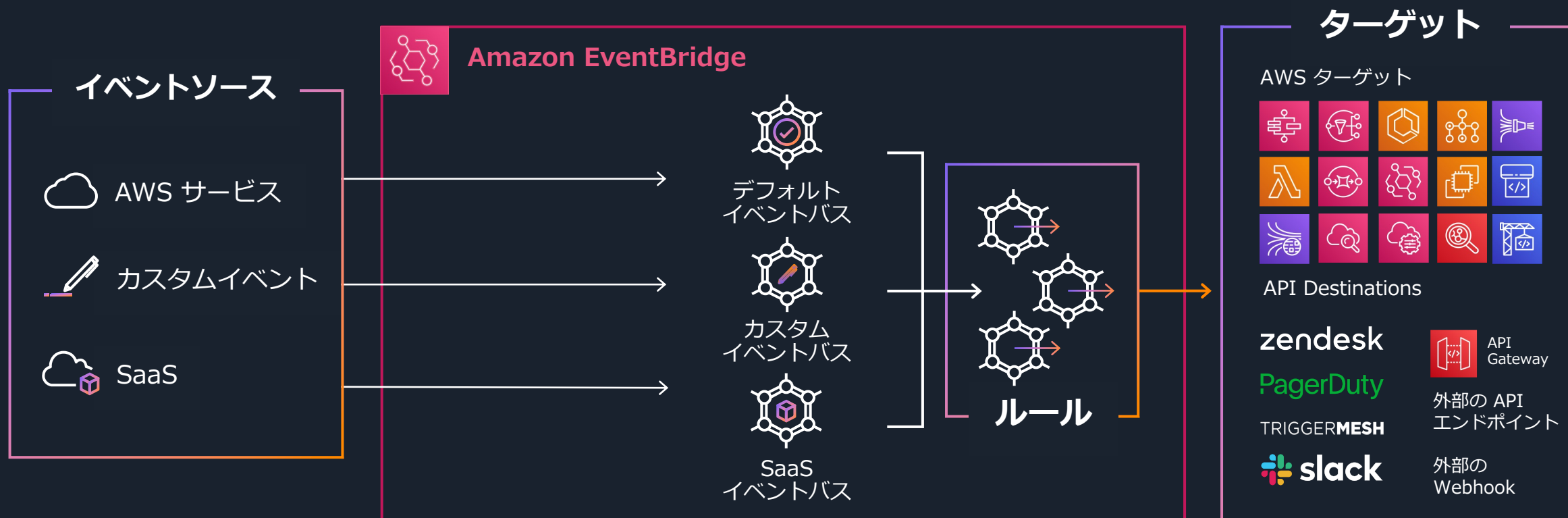
- Amazon EventBridge について深く学びたい方
- イベント駆動型アプリケーションの開発に興味をお持ちの方
- クラウドに最適なジョブ管理システム、スケジューラーをお探しの方

本セミナーの取り扱う範囲

- Amazon EventBridge Scheduler について

* Amazon EventBridge 全体または他の機能については別のセミナー動画をご覧ください

Amazon EventBridge とは



アプリケーション、統合された SaaS アプリケーション、および AWS のサービスから生成されたイベントを受信、フィルタリング、変換、ルーティング、および配信することができるサーバーレスイベントバス。大規模なイベント駆動型アプリケーションの開発を可能に。

Amazon EventBridge Scheduler とは

- マネージドなサーバーレススケジューラー
- タスクの作成、実行、管理を行うことができる
 - 例：毎日 0 時に特定の Lambda 関数を実行
 - 例：2023/12/31 15:00 に EC2 インスタンスを停止
- 270+ の AWS サービス、6000+ の API アクションを呼び出すことが可能
- 少なくとも 1 回 (at-least-once) の信頼性のある配信を保証
- リトライポリシー、デッドレターキューを設定可能
- マネジメントコンソール、AWS CLI、AWS SDK から設定可能
- ユースケース：運用の自動化、バッチ処理、ビジネスロジックの遅延実行
 - ex. ユーザーが解約を実行 → 残りの契約期間が過ぎた後にユーザー削除等を実施



EventBridge Scheduler のユースケース



EventBridge スケジュールルールとの違いは？

- スケジュールルールによって、これまでも似たようなことはできていた
- EventBridge Scheduler は、タイムゾーンのサポートやより高度なカスタマイズ性、スケーラビリティなどを備えた上位機能という位置付け
- 今後スケジュールベースでターゲットを呼び出す場合は、**EventBridge Scheduler** の方を**利用することが推奨されている**
- 既存の EventBridge スケジュールルールは今後も利用し続けることが可能
- サービスの API としては異なるので注意
 - EventBridge ⇨ **events.us-east-1.amazonaws.com**
 - EventBridge Scheduler ⇨ **scheduler.us-east-1.amazonaws.com**
- マネジメントコンソールでは、どちらの機能も EventBridge コンソールに集約されている

EventBridge Scheduler の強み



UTC 以外の様々な
タイムゾーンおよび
サマータイムをサポート



1回限りの実行
をサポート



数百万のスケジュール
を管理可能



270+のサービス、6000+の API
をターゲットに指定可能



EventBridge Scheduler で設定可能なスケジュールタイプ

1. Rate-based

指定の間隔で繰り返し定期実行

```
rate(5 minutes)
```

```
rate(12 hours)
```

```
rate(1 days)
```

- * 秒レベルの実行は未サポート
- * 分 / 時間 / 日 のみ選択可能

2. Cron-based

cron 式による複雑な指定

```
cron(0 8 * * * *)
```

```
cron(15 10 ? * 6L 2022-2023)
```

- * 各種ワイルドカードをサポート
- * UTC または特定のタイムゾーンを指定

3. One-time

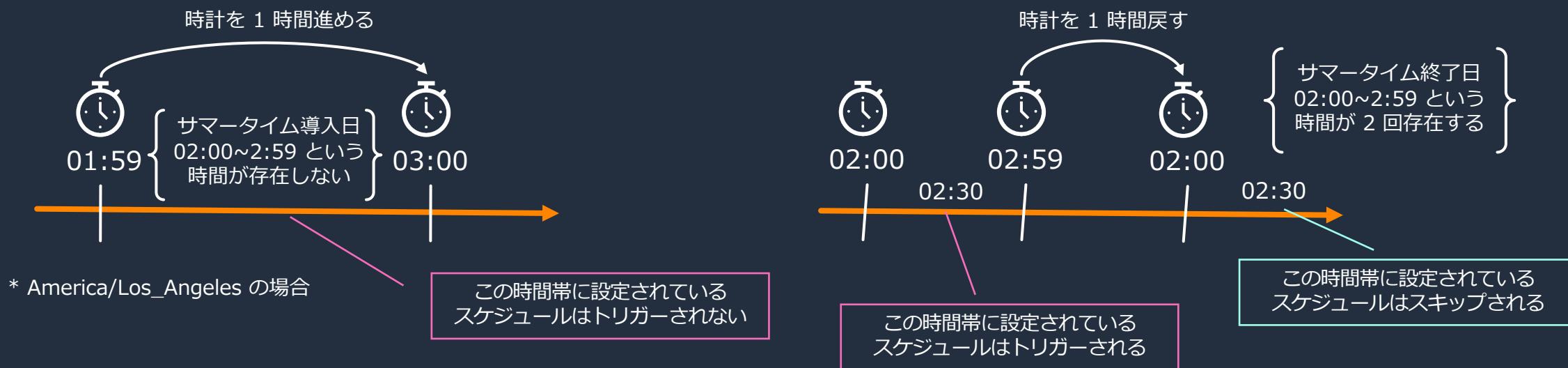
指定のタイミングで 1 回だけ実行

```
at(2022-11-20T13:00:00)
```

- * 秒は指定しても 0 に切り捨てられる
- * 有効な日付および時間を指定
- * UTC または特定のタイムゾーンを指定

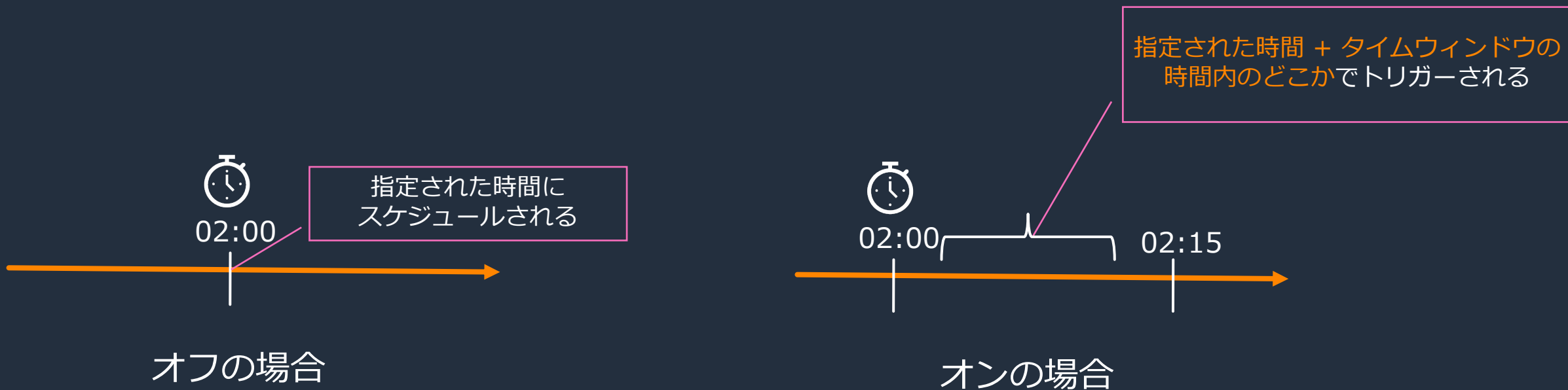
タイムゾーンについて

- IANA によってメンテナンスされている [タイムゾーンデータベース](#) を使用
ex. America/New_York, Asia/Tokyo, Japan
- サマータイムが導入されているタイムゾーンでは以下のルールが適用される
 - サマータイム開始時 ⇨ 存在しない時間に指定されているスケジュールはスキップ
 - サマータイム終了時 ⇨ 2 回存在する時間に指定されているスケジュールは 1 度だけ実行



フレックスタイムウィンドウ

- (指定した時間+タイムウィンドウ) 内でランダムに実行タイミングを分散させる仕組み
- ダウンストリームのサービスのスロットリング、過負荷を防ぐためなどの目的で利用される
- 時間ぴったりに起動する必要がなければ使っておくのがおすすめ



* 15分、30分、1時間、2時間、4時間から選択可能

Templated targets の設定例

Invoke
AWS Lambda

Lambda 関数

arn:aws:lambda:us-west-2: [redacted] function:event-scheduler-test

新しい Lambda 関数を作成

入力

API に渡すパラメータを含む JSON オブジェクト。サンプル値が含まれています。独自の値で JSON を更新してください。注意: パラメータ名は PascalCase である必要があります。 [詳細はこちら](#)

```
1 {  
2   "message": "Hello world!!",  
3   "scheduleArn": "<aws.scheduler.schedule-arn>",  
4   "scheduledTime": "<aws.scheduler.scheduled-time>",  
5   "executionId": "<aws.scheduler.execution-id>",  
6   "attemptNumber": "<aws.scheduler.attempt-number>"  
7 }
```

例えば Lambda Invoke の場合、呼び出す関数を指定し、渡す入力値を設定するだけで利用可能

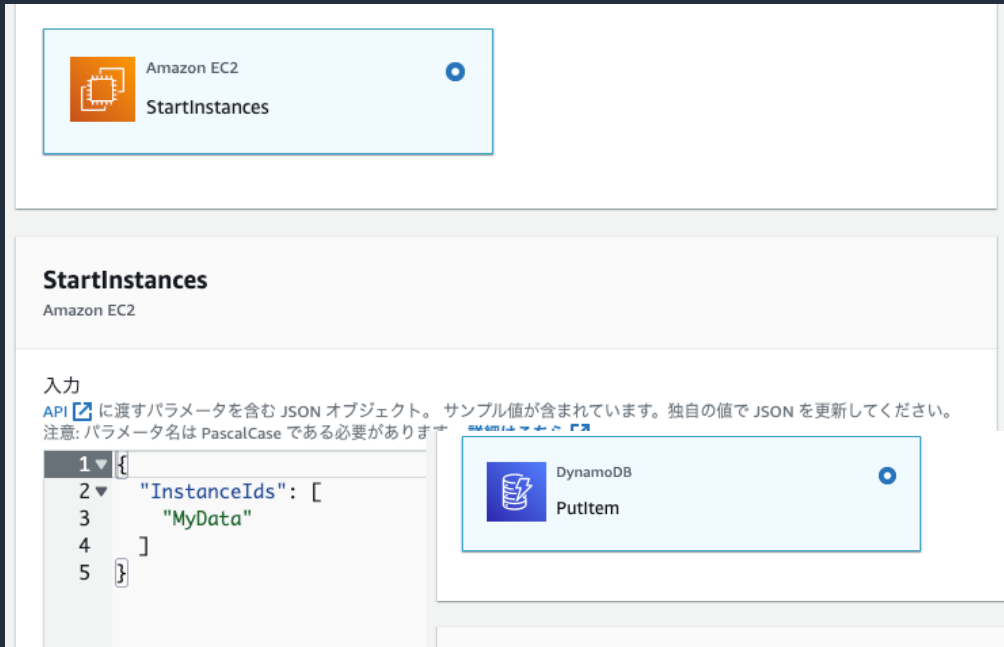
入力値では、事前定義されている **コンテキスト属性** も利用可能

- <aws.scheduler.schedule-arn> - スケジュールの ARN
- <aws.scheduler.scheduled-time> - スケジュールされている時間
- <aws.scheduler.execution-id> - 呼び出しごとに一意の ID
- <aws.scheduler.attempt-number> - 呼び出し試行回数のカウンター

Lambda 関数が受け取った event

```
{  
  message: "Hello world!!",  
  scheduleArn: 'arn:aws:scheduler:us-west-2:1234567890:schedule/default/test-schedule',  
  scheduledTime: '2023-08-01T12:00:00Z',  
  executionId: '9a64c9f4-a02a-42e2-9cfb-a6d88dda320c',  
  attemptNumber: '1'  
}
```

Universal targets の設定例



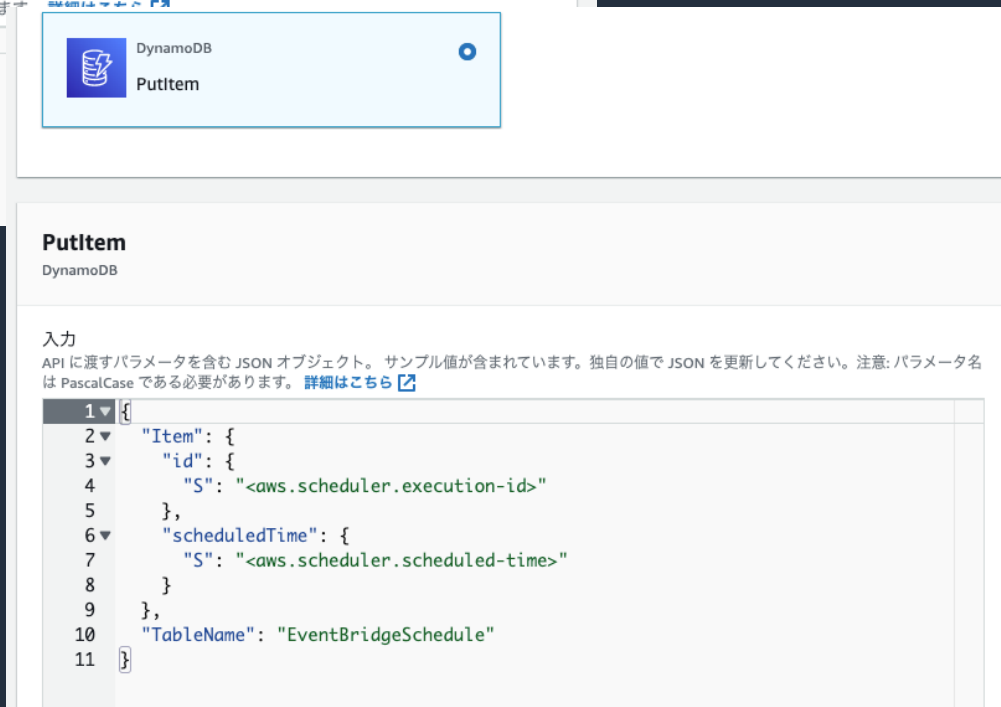
Amazon EC2
StartInstances

StartInstances
Amazon EC2

入力
API に渡すパラメータを含む JSON オブジェクト。サンプル値が含まれています。独自の値で JSON を更新してください。
注意: パラメータ名は PascalCase である必要があります。

```
1 {  
2   "InstanceIds": [  
3     "MyData"  
4   ]  
5 }
```

- 各 API のパラメータとして必要なものを JSON で入力
- Templated targets と同様にコンテキスト属性も利用可能



DynamoDB
PutItem

PutItem
DynamoDB

入力
API に渡すパラメータを含む JSON オブジェクト。サンプル値が含まれています。独自の値で JSON を更新してください。注意: パラメータ名は PascalCase である必要があります。詳細はこちら

```
1 {  
2   "Item": {  
3     "id": {  
4       "S": "<aws.scheduler.execution-id>"  
5     },  
6     "scheduledTime": {  
7       "S": "<aws.scheduler.scheduled-time>"  
8     }  
9   },  
10  "TableName": "EventBridgeSchedule"  
11 }
```



<input type="checkbox"/>	id	scheduledTime
<input type="checkbox"/>	482ac6d50f5e07b1	2022-11-24T12:37:00Z
<input type="checkbox"/>	482ac6d5ab3a4a8c	2022-11-24T12:38:00Z
<input type="checkbox"/>	482ac6d5833b06ae	2022-11-24T12:39:00Z
<input type="checkbox"/>	482ac6d596eda434	2022-11-24T12:40:00Z
<input type="checkbox"/>	482ac6d58ee8e675	2022-11-24T12:41:00Z
<input type="checkbox"/>	482ac6d5cd8a1f74	2022-11-24T12:42:00Z
<input type="checkbox"/>	482ac6d535454e9a	2022-11-24T12:43:00Z
<input type="checkbox"/>	482ac6d530718262	2022-11-24T12:44:00Z

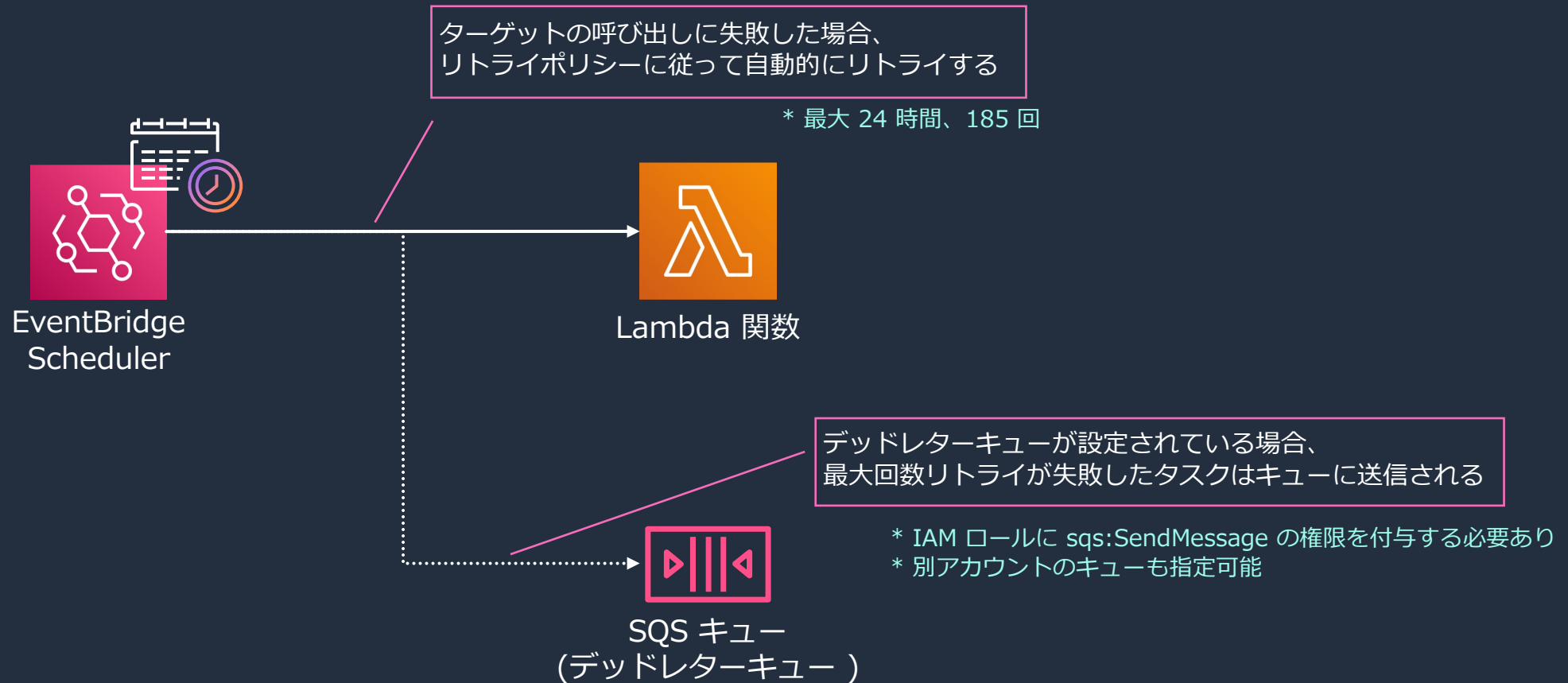
IAM ロールの設定

ターゲットとして指定したサービス API の実行権限を持った IAM ロールを指定する必要がある

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "lambda:InvokeFunction"
      ],
      "Resource": [
        "arn:aws:lambda:us-west-2:xxxxx:function:event-scheduler-test:*",
        "arn:aws:lambda:us-west-2:xxxxx:function:event-scheduler-test"
      ]
    }
  ]
}
```



エラーに対する回復性



EventBridge Scheduler の各種クォータ

- 作成可能なスケジュールグループ数：アカウントあたり最大 **500** まで
(*スケジュールのグループ化に使用)
- 作成可能なスケジュール数：アカウントあたり最大 **1,000,000** まで
(cf. EventBridge Rules: リージョンあたり 300 ルールまで登録可能)
- 指定可能なターゲット数：スケジュールあたり **1 つ**のみ
(cf. EventBridge Rules: 1 ルール内で最大 5 ターゲットまで指定可能)
- CreateSchedule API のリクエストレート：最大 **50** TPS
- 呼び出しスループット：最大 **500** TPS
(*以降はスロットリングされて遅延する)

など

- * 2023/08/01 時点
- * 上限緩和可能なものもあり
- * [Quotas for Amazon EventBridge Scheduler](#)

EventBridge Scheduler の料金

- スケジュールの呼び出し回数に応じて課金
- 1ヶ月あたり 14,000,000 回までは無料で利用可能
- その後 1,000,000 回呼び出しごと料金が発生
 - 例：オレゴンリージョンの場合：\$ 1.00
 - 例：東京リージョンの場合：\$ 1.25

* 2023/08/01 時点

モニタリング

取得可能なメトリクス

InvocationAttemptCount	EventBridge Scheduler がスケジュールを呼び出そうとした数
TargetErrorCount	ターゲットからエラーが返され、呼び出しに失敗した数
TargetErrorThrottledCount	ターゲット側でスロットリングされ、呼び出しに失敗した数
InvocationThrottleCount	EventBridge Scheduler 側でスロットリングされた数
InvocationDroppedCount	最大リトライ回数を超えて呼び出しに失敗し、呼び出しを中止した数

DLQ を設定している場合、以下のメトリクスも利用可能

- InvocationsSentToDeadLetterCount
- InvocationsFailedToBeSentToDeadLetterCount
- InvocationsFailedToBeSentToDeadLetterCount_<error_code>
- InvocationsSentToDeadLetterCount_Truncated_MessageSizeExceeded

* ログについては CloudTrail による API call の記録のみ
(+呼び出し先のサービスのログを活用)



Thank you!