



レガシーをサーバレス化で大勝利したかった話

クラウドLT大会 vol.9

TOPPANデジタル株式会社

TOPPAN



INDEX

1. 自己紹介
2. 発表概要
3. レガシーシステムについて
4. リアーキテクチャ
5. 落とし穴
6. まとめ / 今後の展望
7. 最後に



自己紹介

名前： 加納 広太（かのう こうた）

所属： TOPPANデジタル ICT開発センター DXソリューション開発部

役割：

1 自社IoTソリューションの開発

組み込みデバイス / Webアプリ(backend) / インフラ

2 webアプリ/インフラのアーキテクチャ検討・実装など

資格：



加納 広太



【公式】 TOPPANデジタル | ICT開発センター (@toppan_dxd_ict) on X



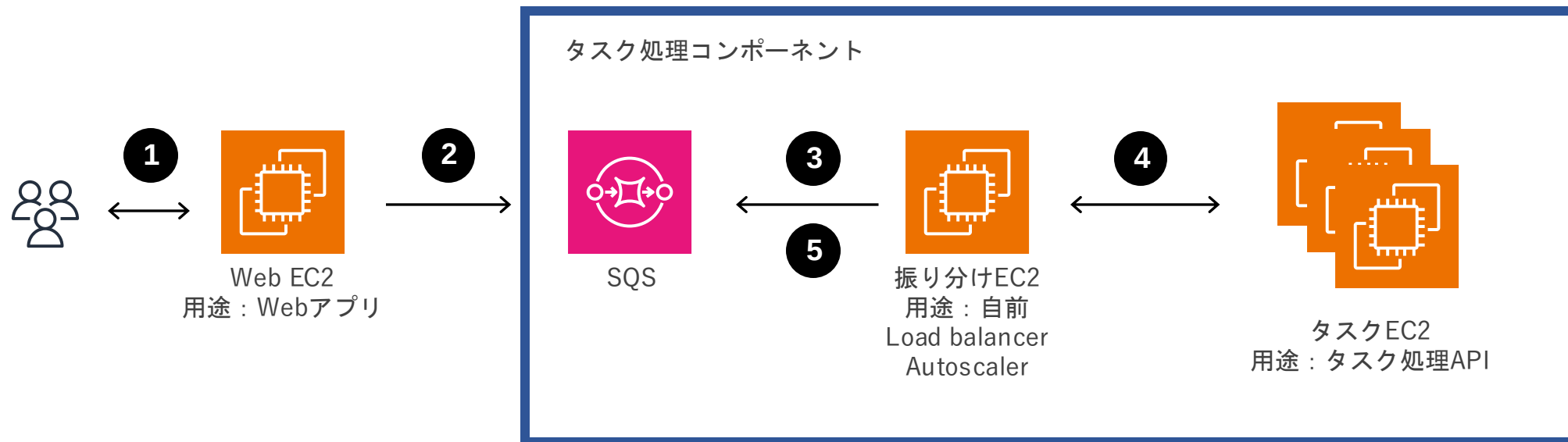
発表概要

○ サーバレス化奮闘記

× AWSサービスの紹介

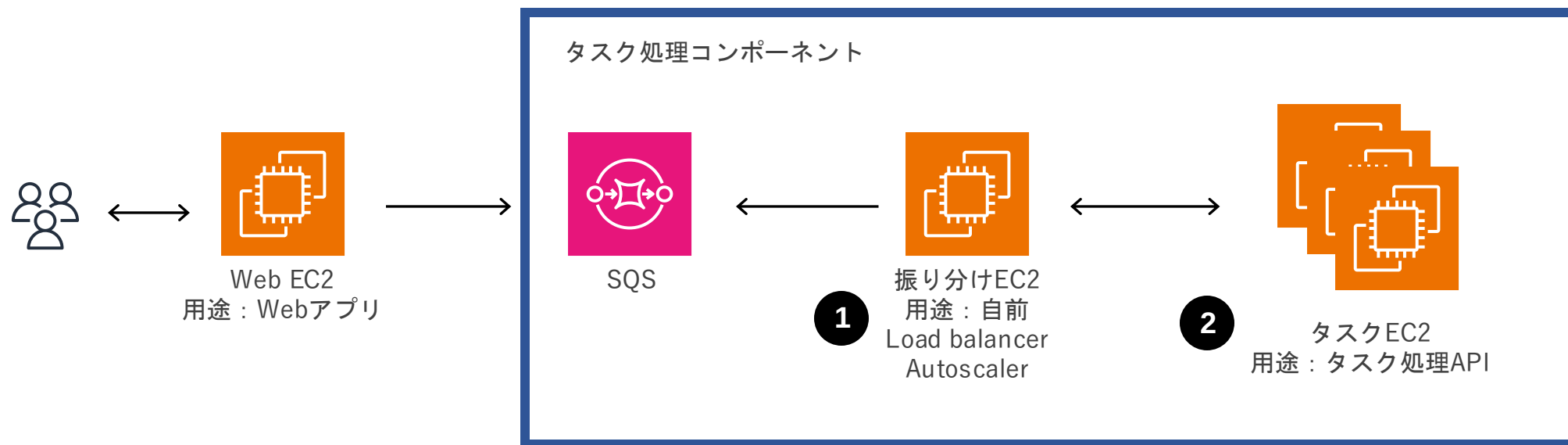
× アーキテクチャの詳細な説明

レガシーシステムについて



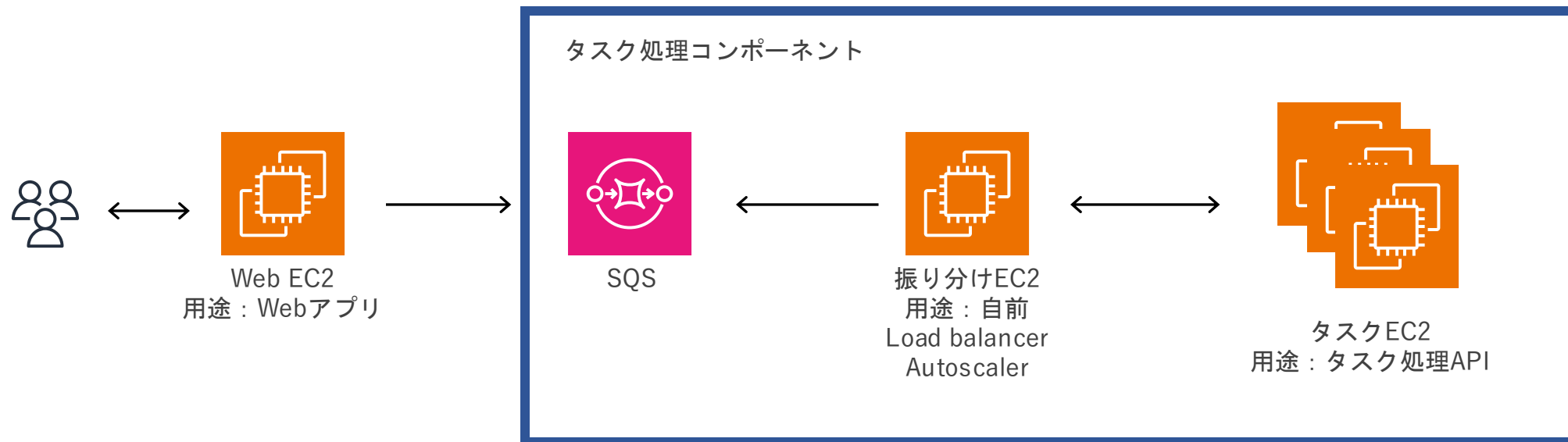
1. ユーザーがwebアプリからタスク処理を依頼
2. WebEC2がタスクの内容をSQSにpush
3. 振り分けEC2がタスク内容をSQSからpop
4. 振り分けEC2がタスクEC2にタスク処理を依頼し、結果を取得
5. 振り分けEC2がSQSに処理結果をpush

要求：タスク処理コンポーネントの費用が高すぎ、とにかく安くしたい！



問題点

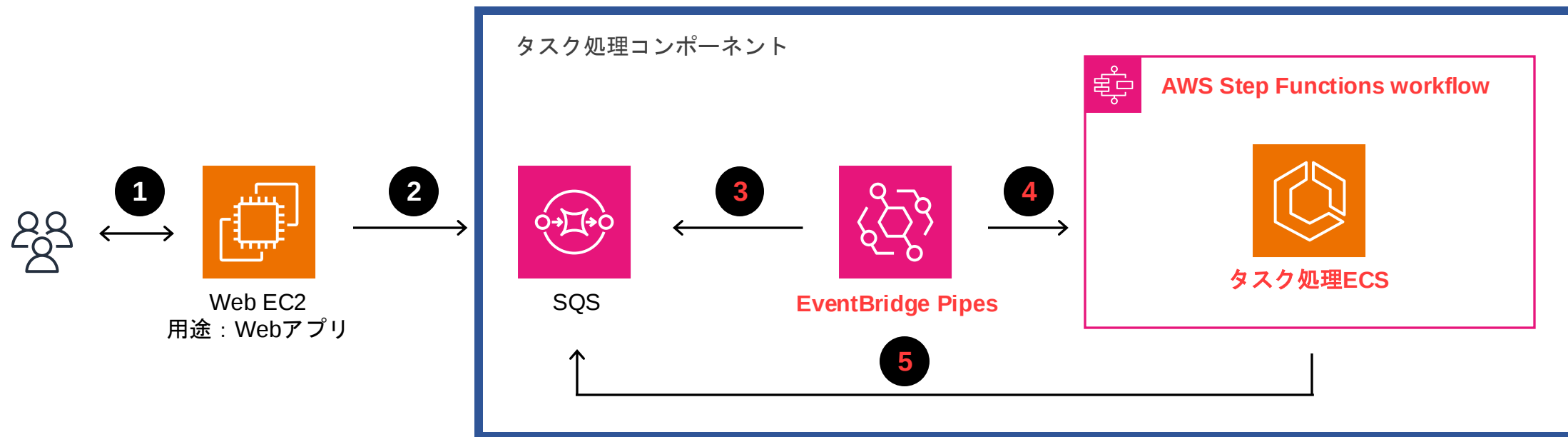
1. 振り分けのロジックを自前実装しているため、実行環境の更新などが定期的に必要なになる
振り分けEC2自体にもコストが掛かる
2. 振り分けEC2のAutoscalingロジックの都合で、使用されていなくても一定台数は起動してしまいコストが高い



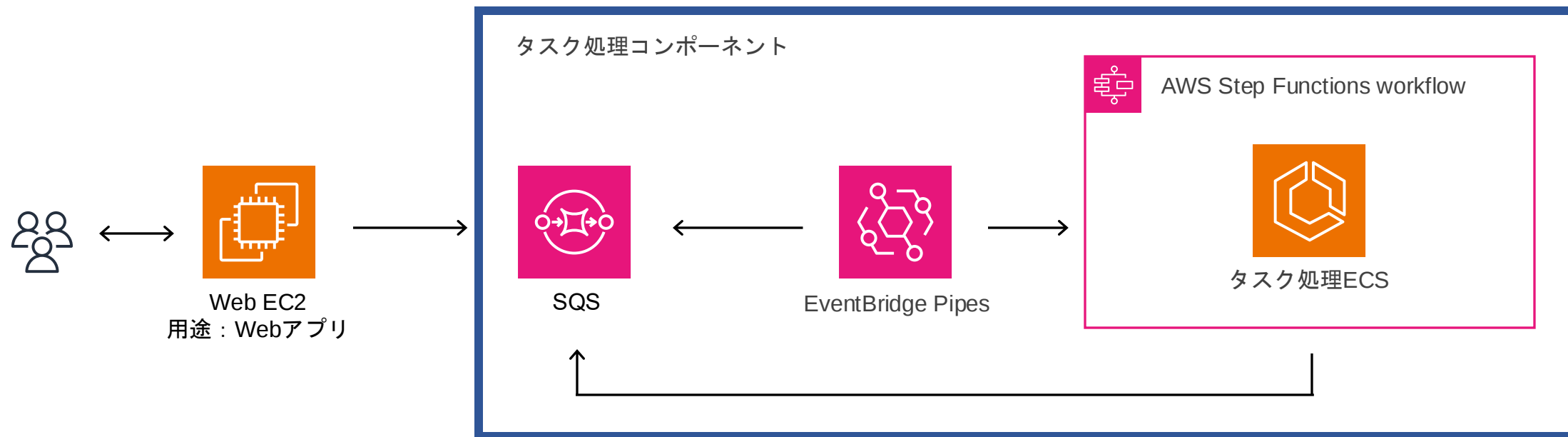
基本的にはEC2が常時起動しているのがコスト増の原因
未使用時のコスト削減のためサーバレス化を検討！



リアーキテクチャ



1. ユーザーがwebアプリからタスク処理を依頼
2. WebEC2がタスクの内容をSQSにpush
3. EventBridge Pipesがタスク内容をSQSからpop
4. EventBridge PipesがStepFunctionsを起動し、ECS（タスク処理API）を実行
5. ECSがSQSに処理結果をpush



■各サービスを選定した理由

- SQS:** 既存のWebの構成をなるべく変更したくなかったため、SQSを軸に検討
- EventBridge Pipes:** SQSとStep Functionsの統合をサーバレスで実現するのに最適なため
- Step Functions:** ECSタスクのエラーやスロットリング対応のため
- ECS:** EC2で稼働していたタスク処理APIを置き換えるため
※15分以上の稼働必要なためLambda不可

実装したフローが導通したときの気持ち



と思ったのも束の間。。。。



落とし穴

突然ですが、みなさんはAWS SQSの「一時キュー（仮想キュー）」はご存知でしょうか。

[Amazon SQS一時キュー - Amazon Simple Queue Service](#)によれば、

仮想キューは、**Temporary Queue Client**が作成するローカルデータ構造です。

仮想キューを使用すると、トラフィックの少ない複数の宛先を単一のAmazon SQS キューに結合できます。（中略）

一時キューの最も一般的な使用例は、リクエスト-レスポンスメッセージングパターンです。

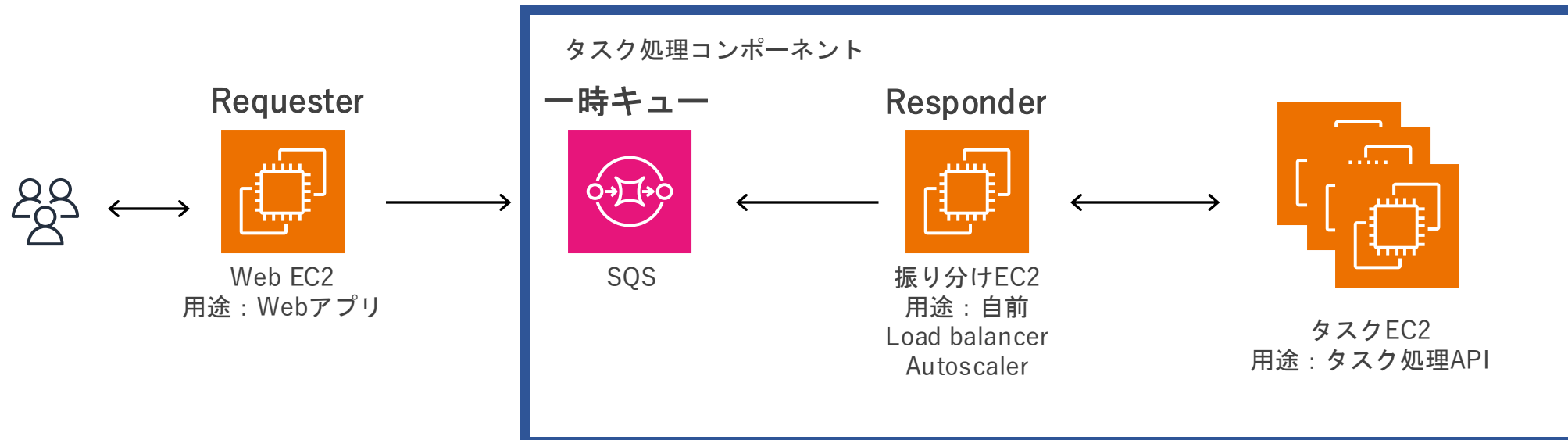
このパターンでは、要求者が各レスポンスメッセージを受信するための一時キューを作成します。



SQSを同期的（リクエスト-レスポンスパターン）に利用することができるようになる仕組み

一時キューを利用するためには、リクエスト側とレスポンス側の両方でロジックの実装が必要

実は元実装のSQSは一時キューなるものとして利用されていた



ロジックが必要なため

単純に振り分けEC2部分をEventBridge Pipesなどに置き換えることは不可能…

とほぼ…

まとめ / 今後の展望

Good

- ・サーバレス実装の知見を得ることができた
- ・タスクEC2 → タスクECS での稼働を確認することができた
- ・~~一時キュー！そういうものもあるのか~~

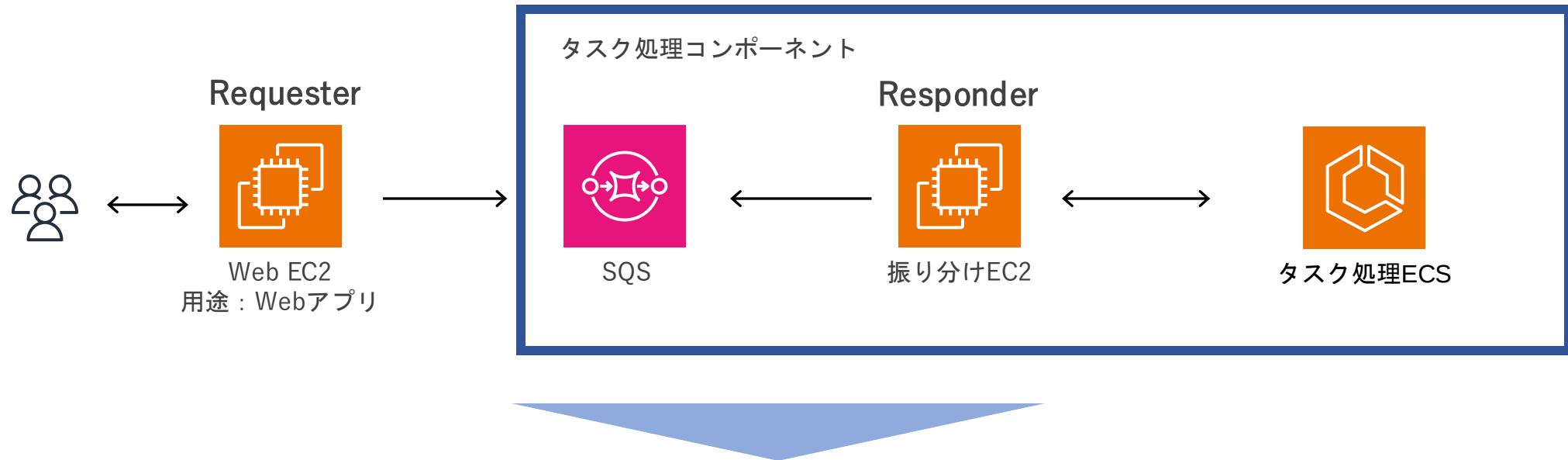
Motto

- ・そもそも最初から一時キューが使用されていると知っておくべきだった
- ・動作確認は実施したが、各リソースの作り込み（エラー処理、二重実行の抑制など）についてはまだまだこれから

まとめ / 今後の展望

パターン1 サーバーレス化の検討を継続する

パターン2 振り分けEC2の機能を一部オミットして利用する



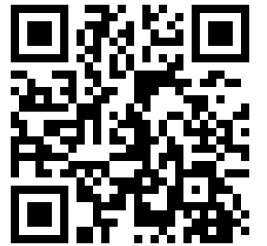
基本的にはパターン1で進めつつ、難しければ
どこかのタイミングでパターン2への切り替えを検討



最後に

5. 最後に

メンバーを積極的に募集中です！
気軽にドアノックしてみてください！



ホーム

ミッション

価値観

メンバー

働き方やメンバーに
興味がある方へ

仕事に興味がある方へ

会社情報

ポートフォリオ



SaaS開発チーム

吉田 学さん
他10人



開発戦略・デベロッパーサク セスチーム

原井 隆浩さん
他2人



ZETA・メタバース開発チー ム

篠田 健一さん
他9人



物流DX開発チーム

関根 良太さん



顧客DX支援チーム

生駒 健司さん
他1人



可能性を高めるTOPPANのデジタル変革

Erhoeht-X

ご清聴ありがとうございました

