「なんでAWS選んだんですか？」

野村総合研究所

新井　雅也

新井 雅也 – Masaya ARAI

野村総合研究所
ビジネスIT基盤推進部

★Certifications & Roles

aws partner network AMBASSADORS

AWS Well-Architected Lead

★Publications

＼New／

＼New／

★Communities & SNS

AWS community builder

JAWS-UG
コンテナ支部

@msy78

hatena blog
iselegant

本日のゴール

本日は、APN Ambassadorの一人として、

暗黙知として捉えられがちな”AWSを使う理由”について、

お話したいと思います。

皆さまが「AWSを」利用する理由について、

自答・再考いただくきっかけになれば嬉しいです。

「AWSを使っていますか？」

もしくは、使いたいと思っていますか？

使おうとしていますか？

「なんでAWS選んだんですか？」

上司から

後輩から

「なんでAWS選んだんですか？」

同僚から

お客様から

 「なんとなく。流行りで。」

安いから

やりたいことが
できそうだから

「なんとなく。流行りで。」

楽を
したいから

面白いから

便利だから

少しだけ質問を改めましょう。

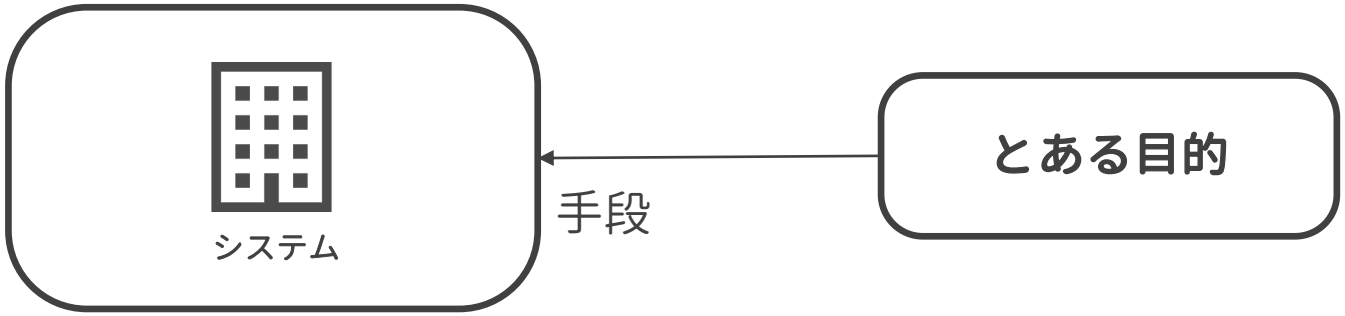
自分たちのビジネスを支えるプラットフォームとして、

「なんでAWS選んだんですか？」

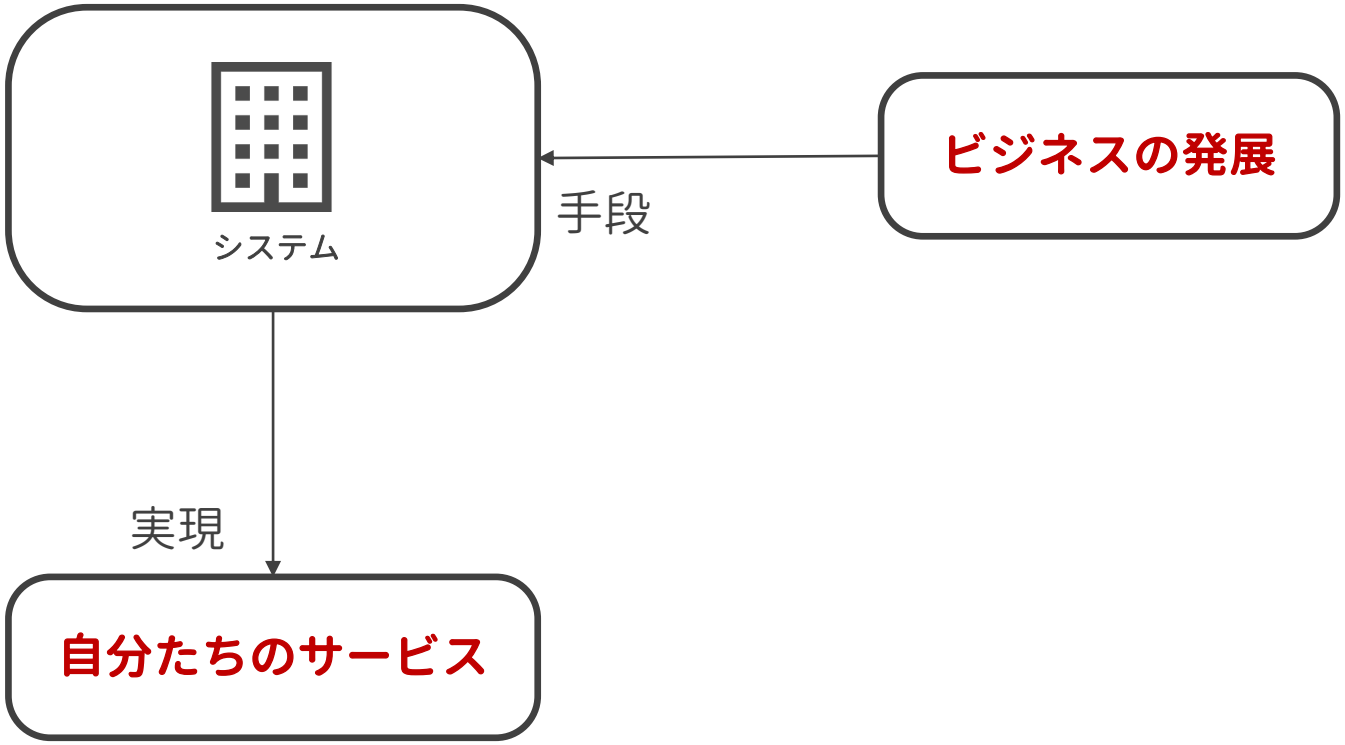
「。。。  」

とならないように、AWSを利用すべき理由を探っていきましょう。

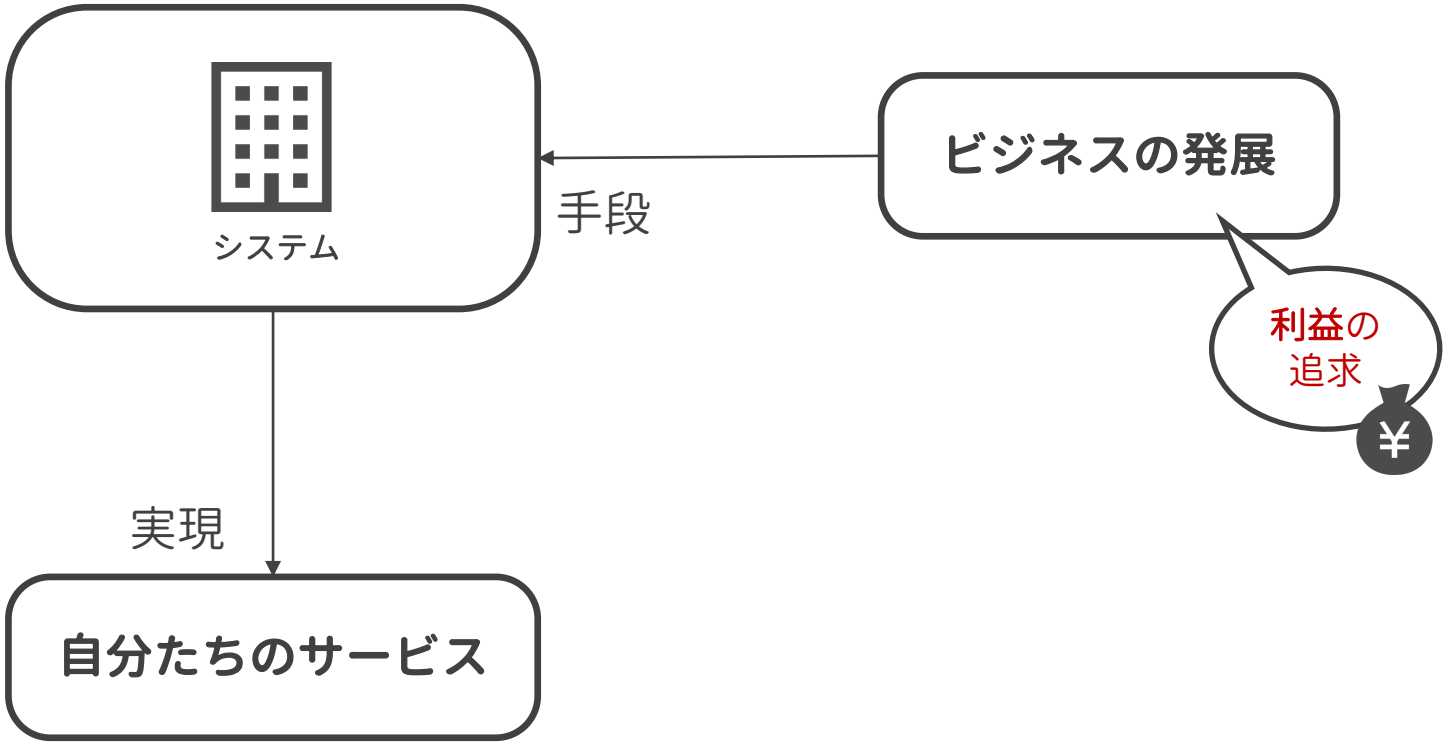
テクノロジー・システムは目的を達成するための手段の１つ。



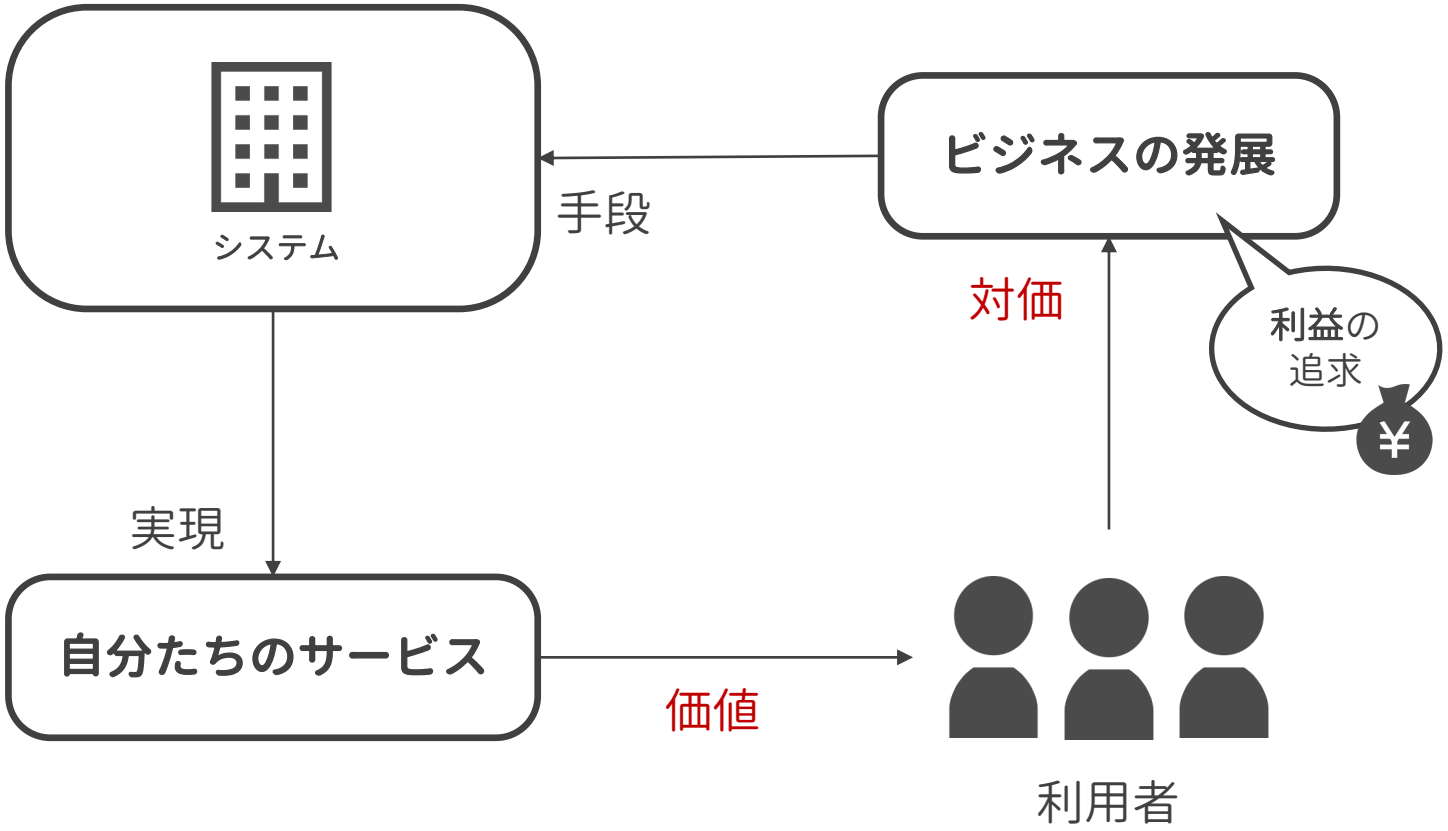
その目的とは、お客様もしくは自分たちのビジネスの成長と発展である。



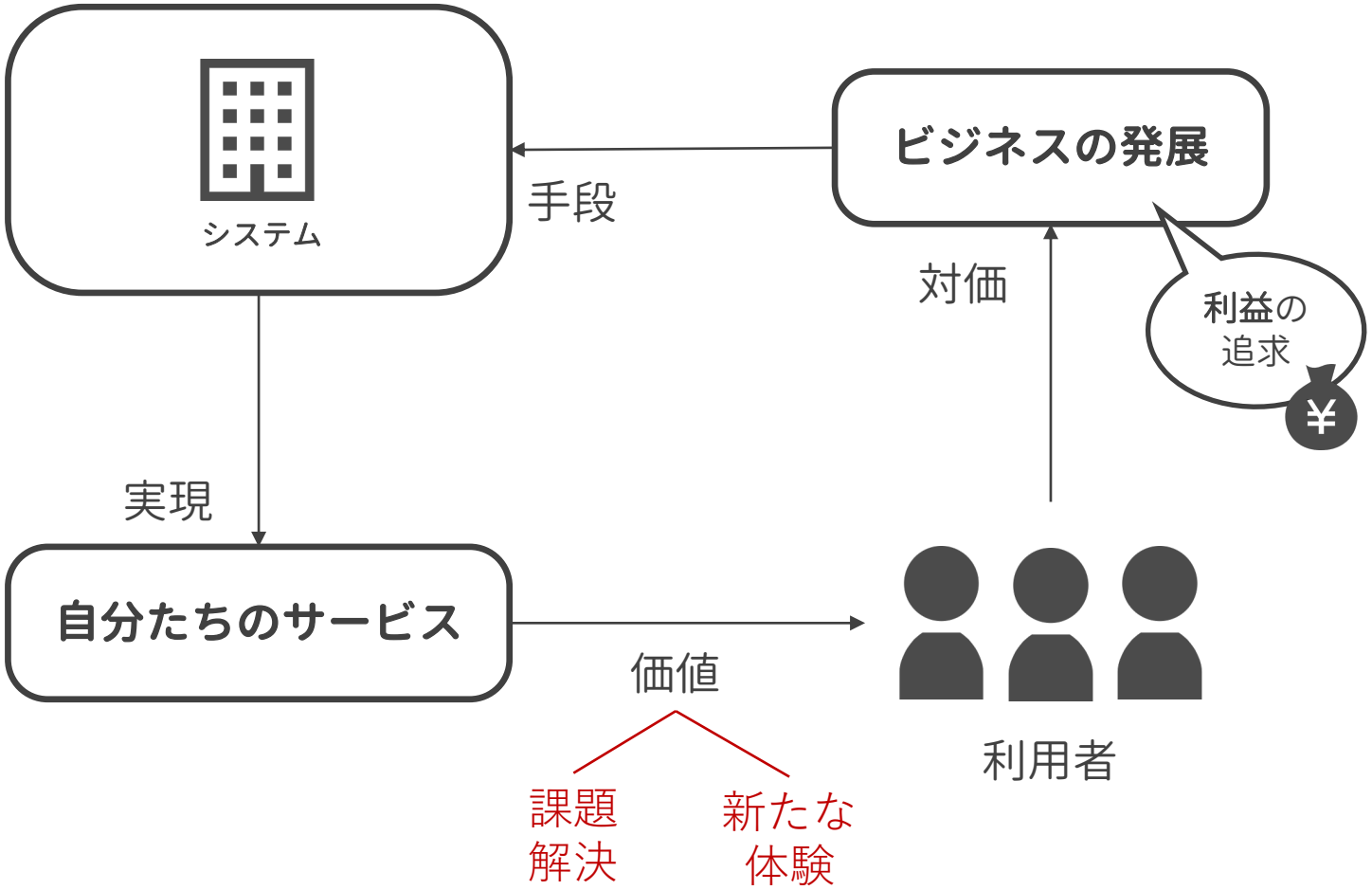
ビジネスの目的は利益の追求である。ただそれだけではない。



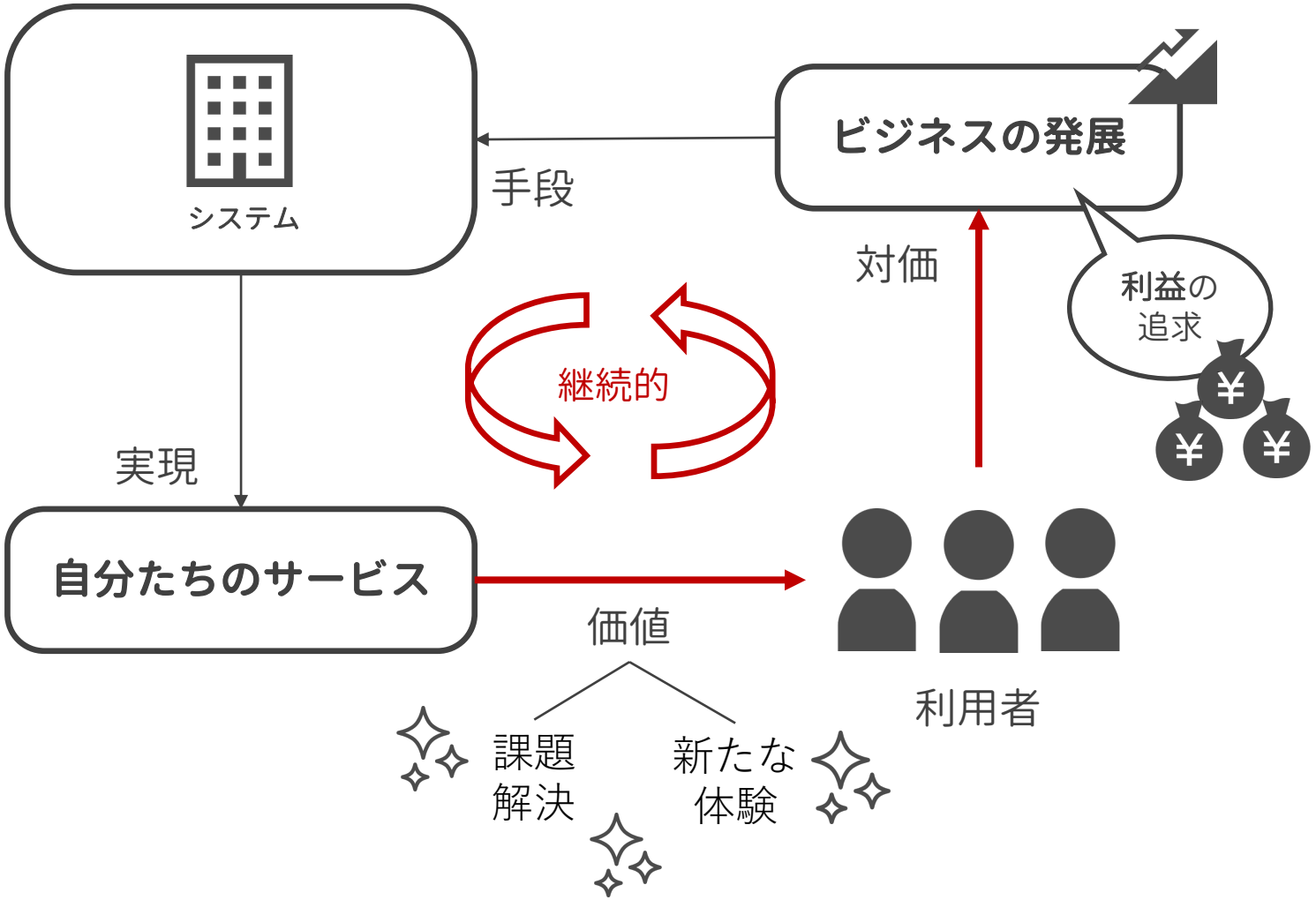
利益は自分たちのサービスが**利用者の価値**となって届いたことによる対価。



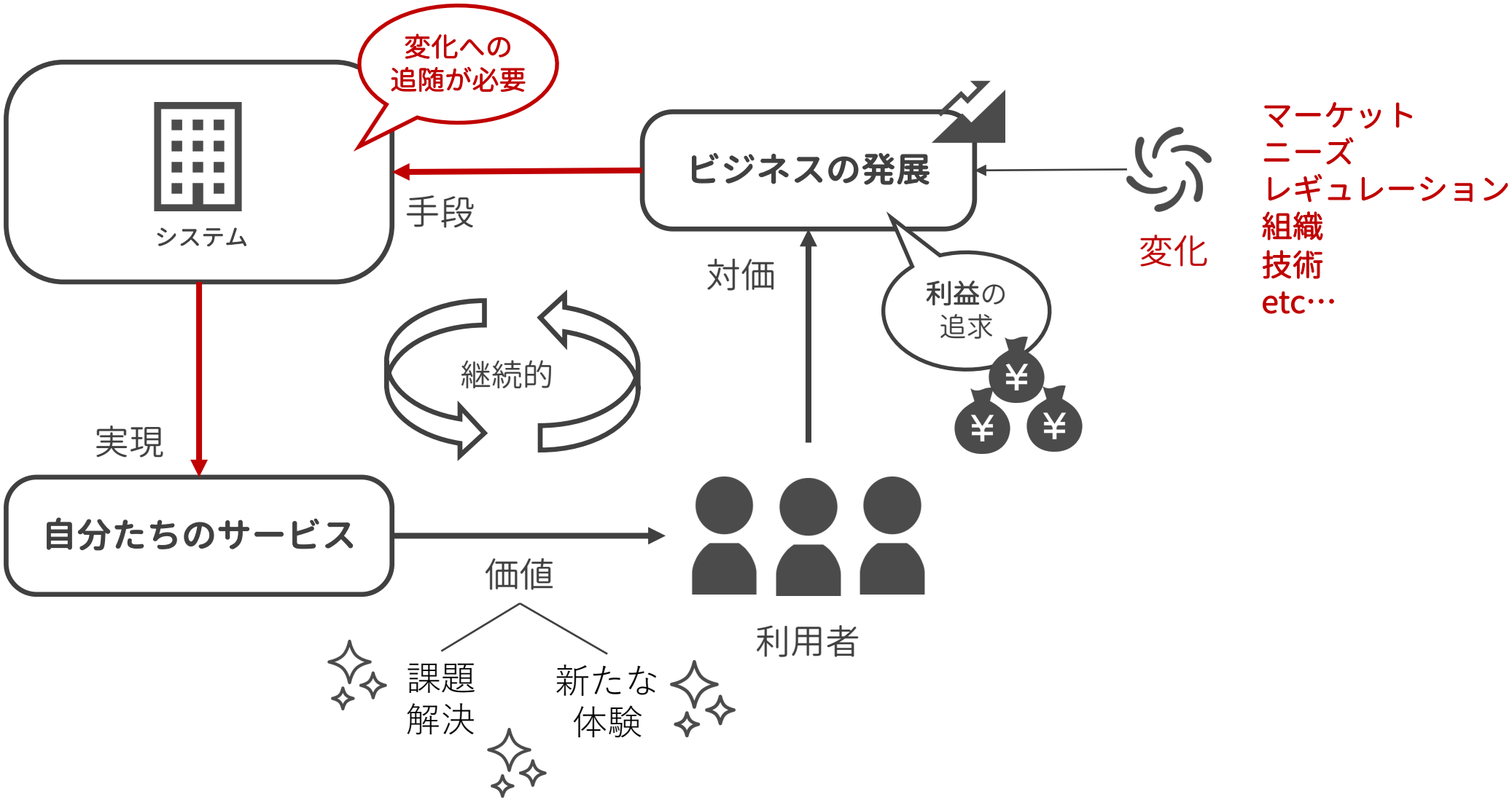
価値とは、利用者に対する課題解決・新しい体験と等しい。



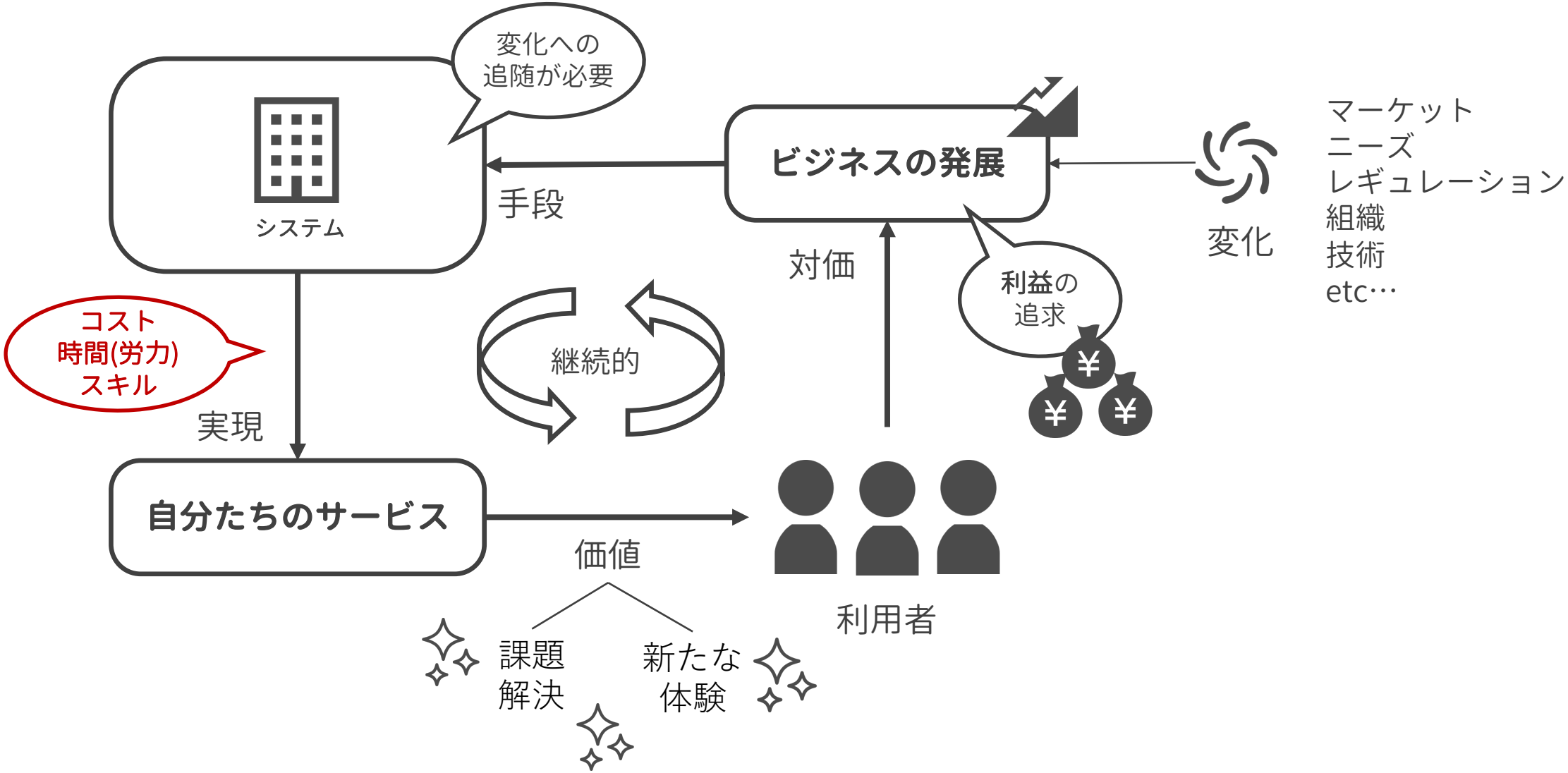
ビジネスをより発展させるためには利用者に対する価値を
継続的に高めていかなければならない。



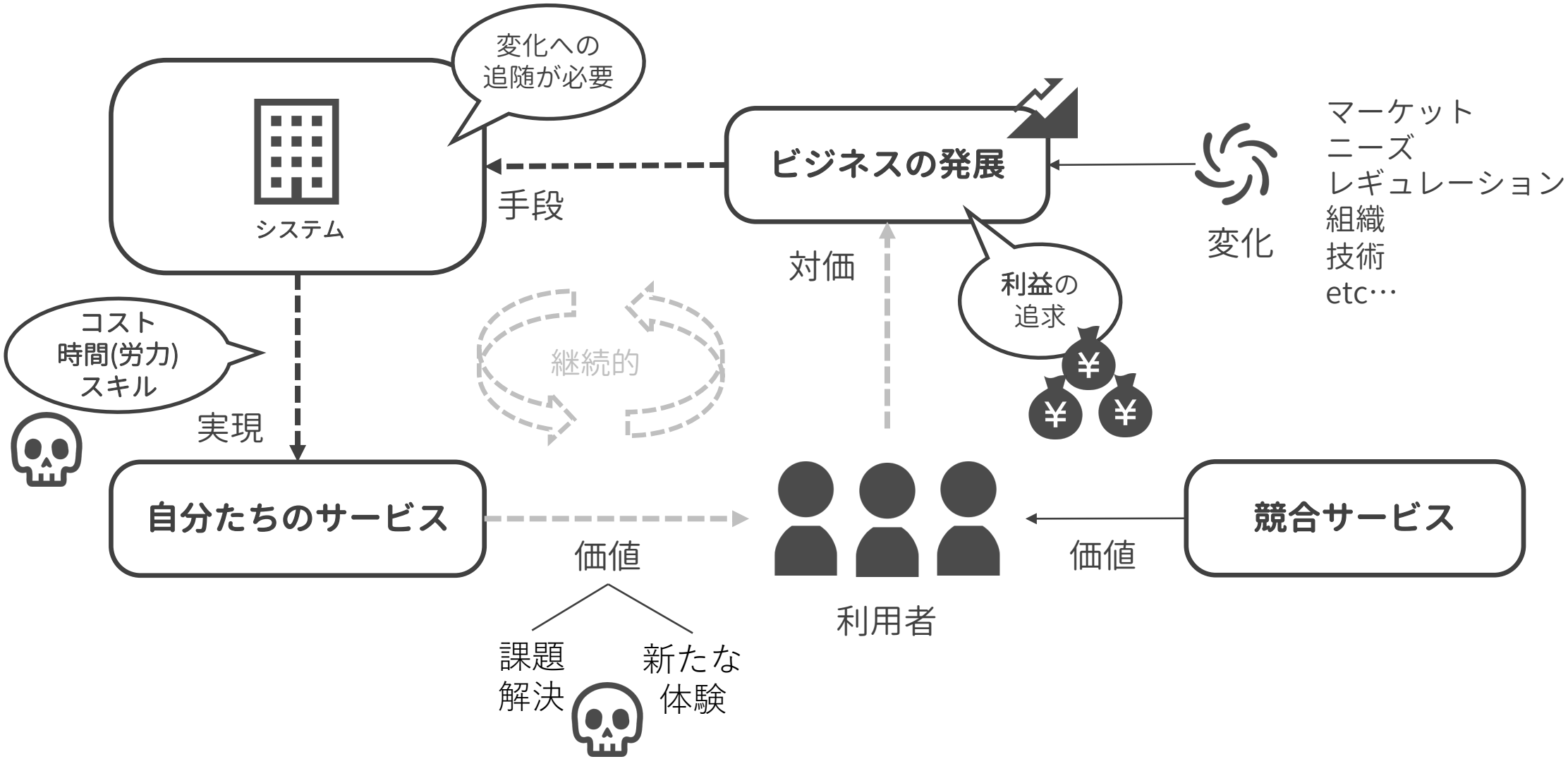
起こりうる様々な<u>変化</u>を捉え、サービスを改善していくことで、利用者への価値提供は維持される。



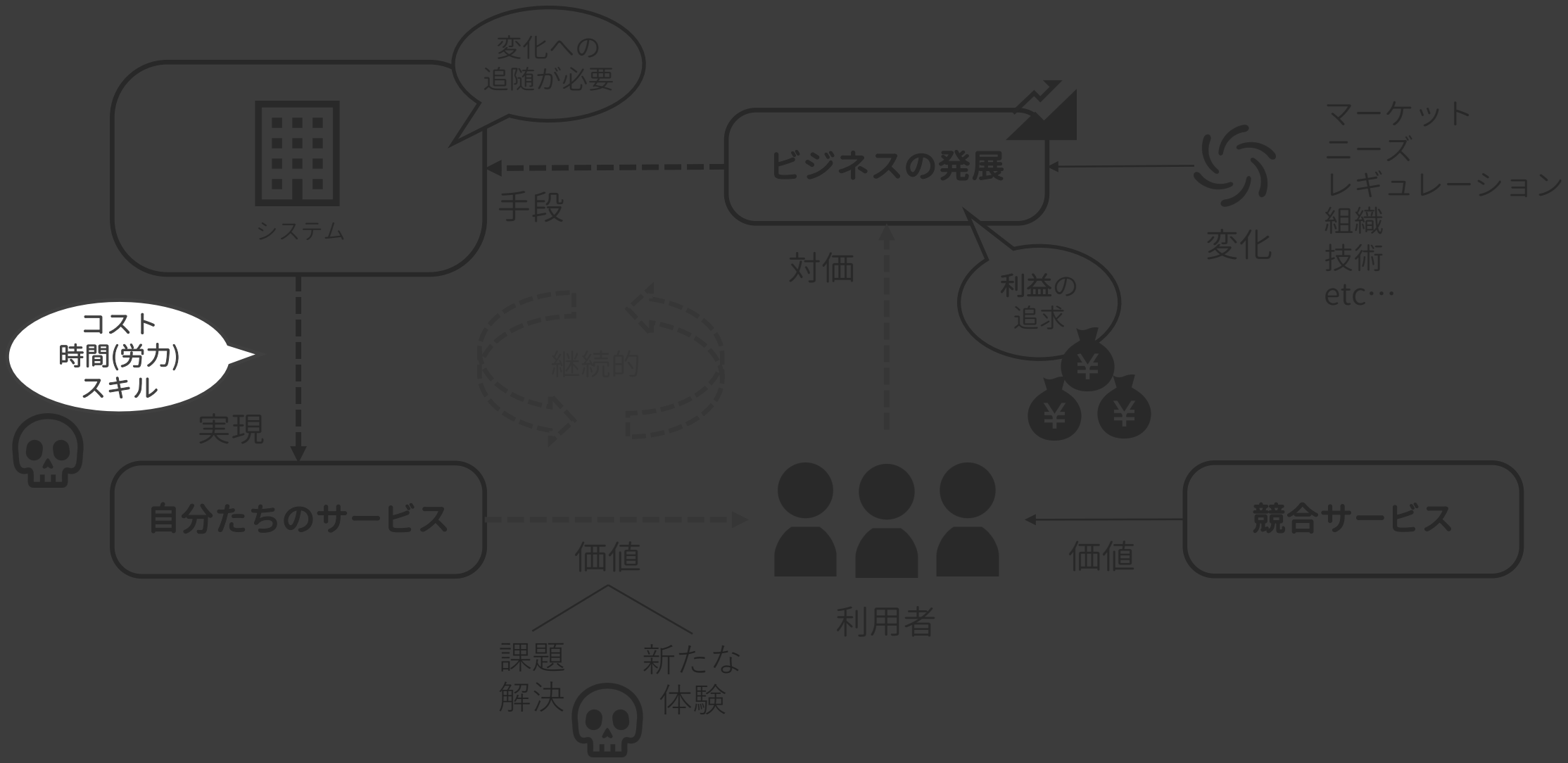
変化に追随し、常に見直すことは
当然ながらコスト・時間・スキル等が求められる。



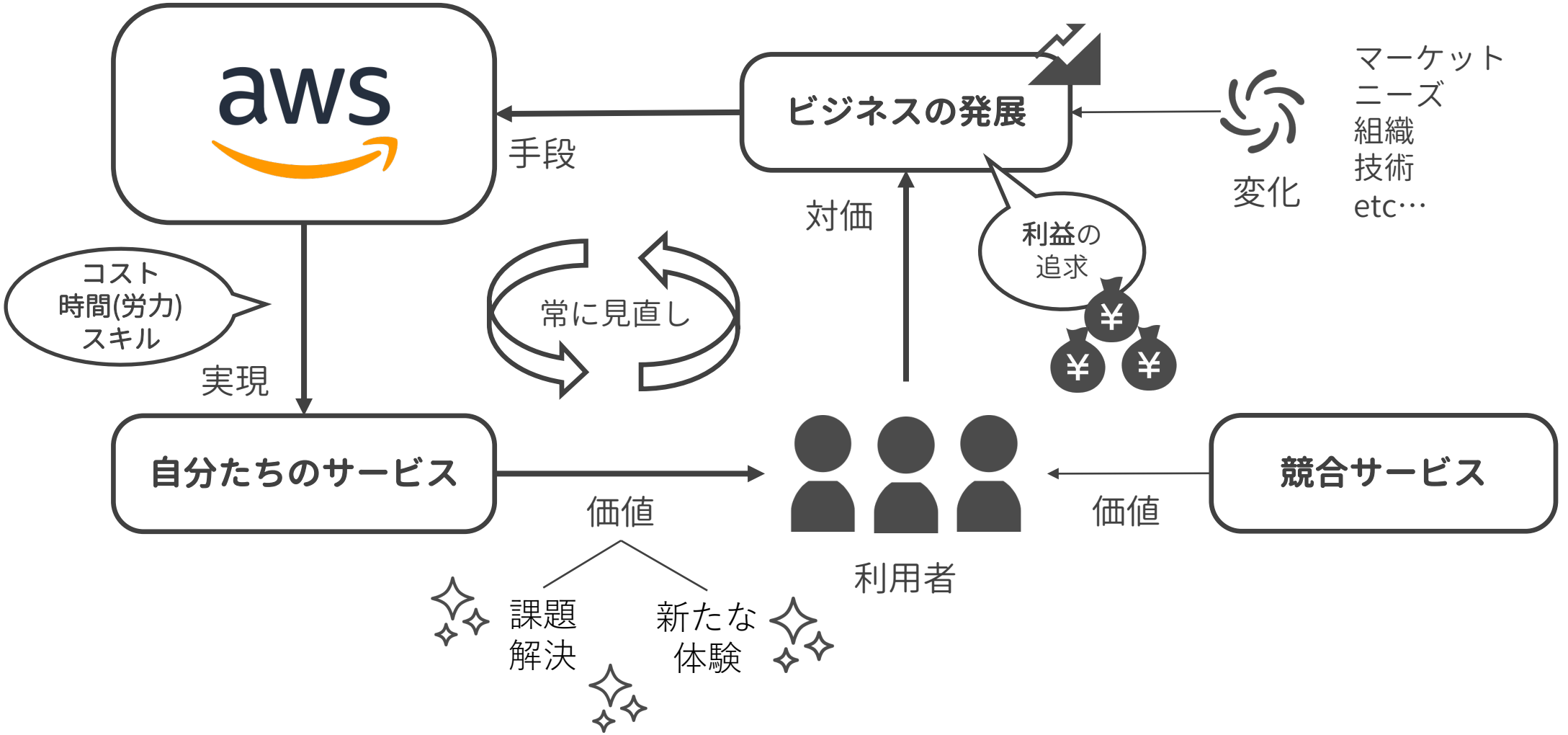
変化への適用を怠ると、当然ながらシステムは陳腐化する (=技術的負債)。
競合サービスと比較し、今までの価値が価値でなくなる。



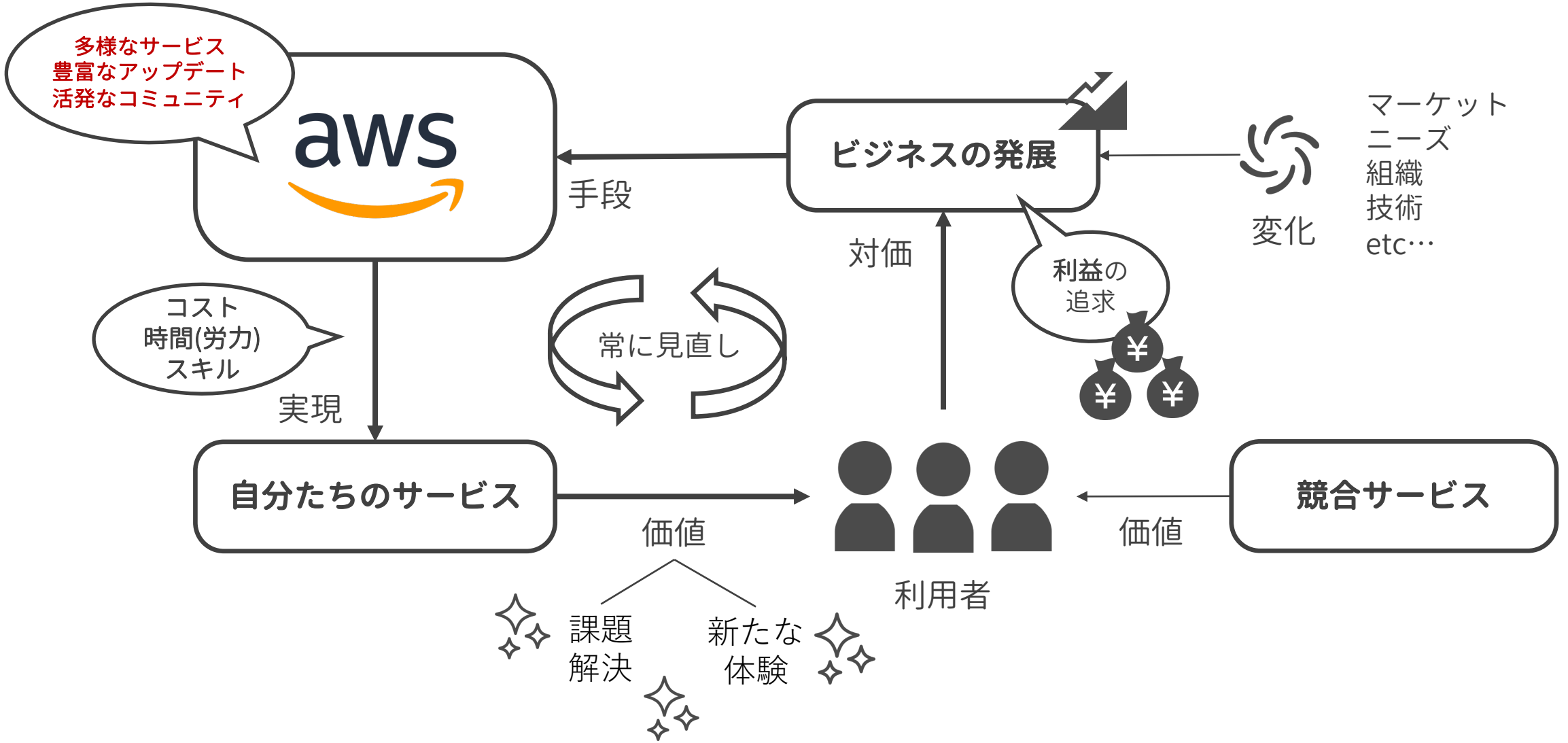
自分たちのビジネスにリソースを集中するために、
コストと時間（リードタイム）を最適化することが望ましい。



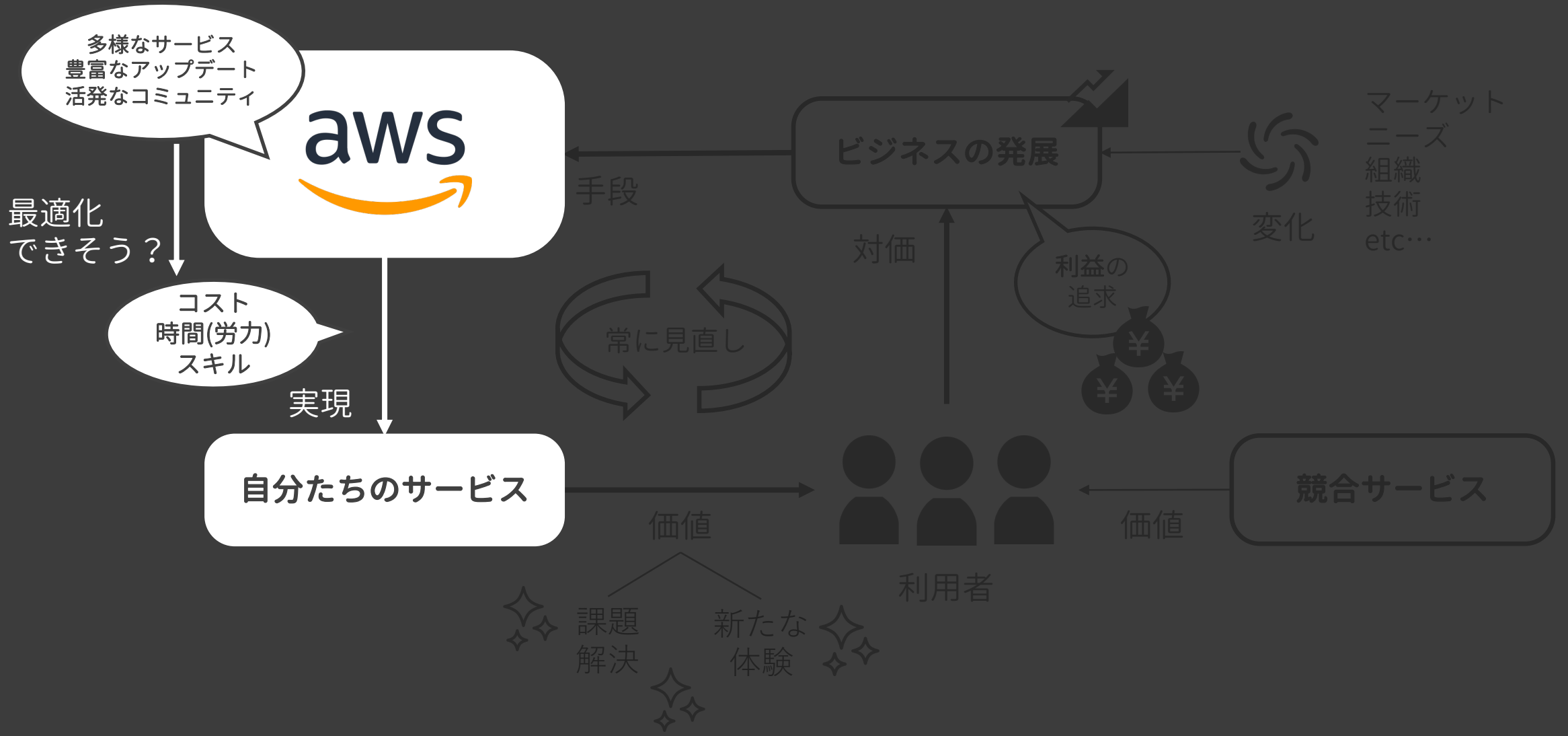
AWSで自分たちのユースケースに合わせた抽象度のサービスを活用することで、変化の追随を前提としたシステムにする。



AWSで自分たちのユースケースに合わせた抽象度のサービスを活用することで、変化の追随を前提としたシステムにする。



変化していくことを前提とするためには、
改めてAWSを利用する意義とスタンスを理解しておくことが重要。



AWSを採用する代表的な３つの理由

多様な
サービス

豊富な
アップデート

活発な
コミュニティ

AWSを採用する代表的な３つの理由

多様な
サービス

豊富な
アップデート

活発な
コミュニティ

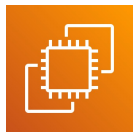
多様なサービスにより自分たちのビジネスにあった要件を追求できる

- AWSではBuilding Blockという考え方
  - 200以上の様々なサービス(Block)組み合わせるという思想
  - 多様なサービス ＝ 豊富な組み合わせ方 = あらゆるユースケースへの対応

- 抽象度が様々なサービスを組み合わせながら、
  自分たちのビジネス要件にカスタマイズ可能

代表的なAWSコンピューティングサービスを俯瞰する

代表的なAWSコンピューティングサービスを俯瞰する


EC2


Beanstalk


ECS

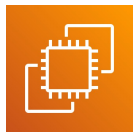

EKS


Fargate


App Runner


Lambda

代表的なAWSコンピューティングサービスを俯瞰する



EC2

- 自分たちで OS管理。
- 自由度の高い リソース設定。

Beanstalk

- ALBやDBを 一気に構築。
- OSレイヤは ある程度管理。

ECS

- コンテナ管理を マネージドに。
- 他AWSサービス と組み合わせ。

EKS

- Kubernetesを マネージドに。
- OSSエコシステム 中心な運用。

Fargate

- コンテナを サーバレスに。
- OS管理を したくない。

App Runner

- Webアプリ FW を使いたい。
- アプリ開発中心。

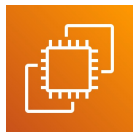
Lambda

- インフラを 意識したくない。
- アプリ開発中心。

代表的なAWSコンピューティングサービスを俯瞰する

オートスケール
容易なデプロイ

CI/CD
オートスケール
暗号化



EC2

- 自分たちで OS管理。
- 自由度の高い リソース設定。

Beanstalk

- ALBやDBを 一気に構築。
- OSレイヤは ある程度管理。

ECS

- コンテナ管理を マネージドに。
- 他AWSサービス と組み合わせ。

EKS

- Kubernetesを マネージドに。
- OSSエコシステム 中心な運用。

Fargate

- コンテナを サーバレスに。
- OS管理を したくない。

App Runner

- Webアプリ FW を使いたい。
- アプリ開発中心。

Lambda

- インフラを 意識したくない。
- アプリ開発中心。

インフラエンジニア向け

アプリエンジニア向け

コンテナワークロード

サーバレス

リフト寄り

クラウドネイティブ（シフト）寄り

事例：クレジットカード系システム構築における主要コンピューティングリソース検討

とあるお客様向け新規クレジットカード与信・決済管理のシステムの構築事例 (2021年10月～現在)。
カード発行業務（イシュイング業務）の主体として、
PCIDSS (Payment Card Industry Data Security Standard)の全331項目精査・準拠が必要であった。

※PCIDSS: クレジットカード情報の安全な取り扱いを目的に策定された国際セキュリティ基準

事例：クレジットカード系システム構築における主要コンピューティングリソース検討

★要件１
スタートアップ企業のサービスであり、アジリティの高い構成が前提。
→OS管理はしたくない。
→クラウドネイティブをベース。
→CI/CDと相性のよいコンテナ技術を採用。

★要件２
・PCIDSS準拠に伴い、ある程度自分たちでセキュリティ設計の柔軟さを残したい。
→コントロールプレーン＋データプレーンの採用。
→抽象度が高すぎるサービスは避ける。

★要件３
・PCIDSSのスコープをなるべく小さくしたい。
→コントロールプレーン側でPCIDSS対象となる範囲を減らしたい。

※PCIDSS: クレジットカード情報の安全な取り扱いを目的に策定された国際セキュリティ基準

事例：クレジットカード系システム構築における主要コンピューティングリソース検討

要件１の観点から見送り

	ECS	EKS	Fargate	App Runner	Lambda
	・コンテナ管理をマネージドに。	・Kubernetesをマネージドに。	・コンテナをサーバレスに。	・WebアプリFWを使いたい。	・インフラを意識したくない。
	・他AWSサービスと組み合わせ。	・OSSエコシステム中心な運用。	・OS管理をしたくない。	・アプリ開発中心。	・アプリ開発中心。

・自分たちでOS管理。
・自由度の高いリソース設定。
・ALBやDBを一気に構築。
・OSレイヤはある程度管理。

インフラエンジニア向け

アプリエンジニア向け

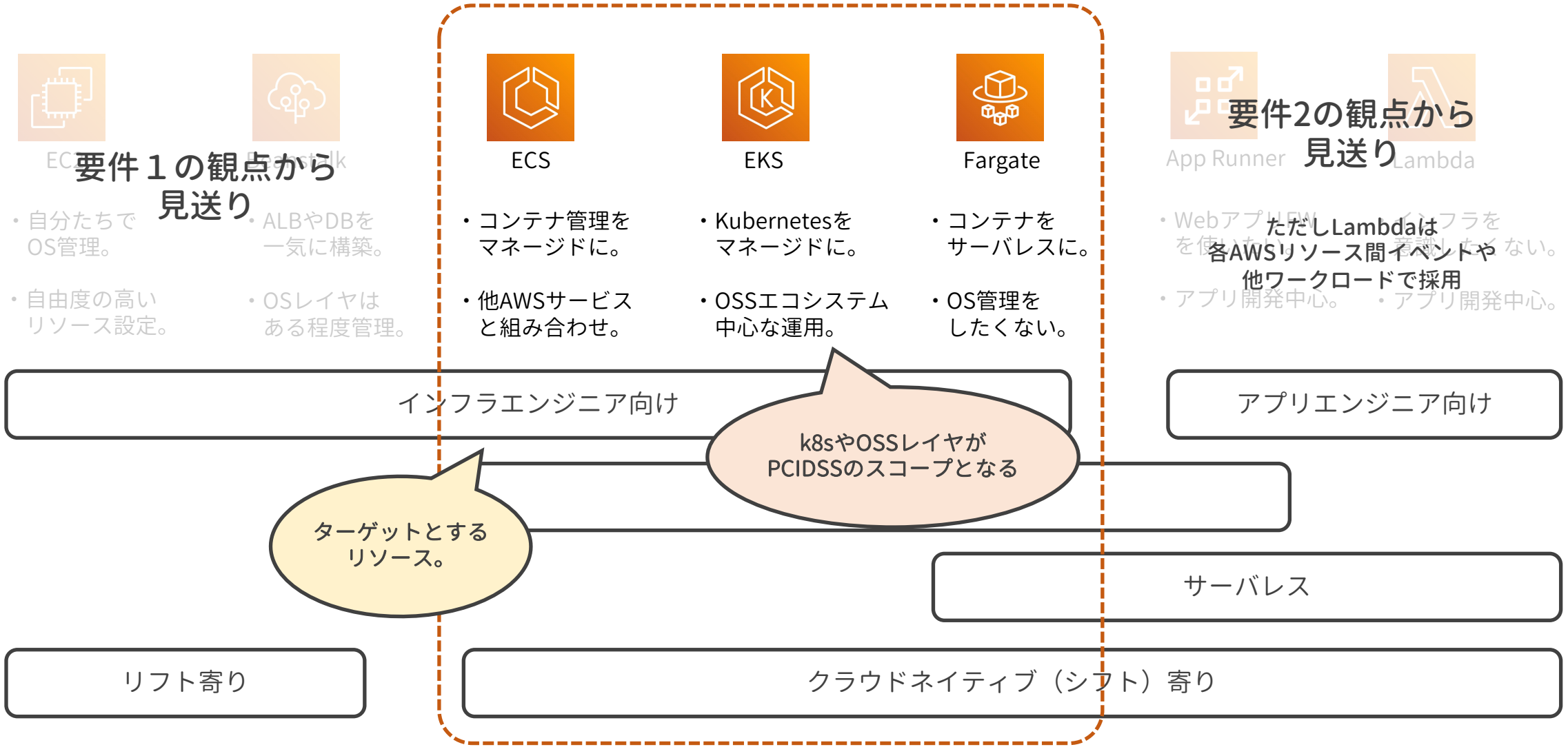
コンテナワークロード

サーバレス

リフト寄り

クラウドネイティブ（シフト）寄り

事例：クレジットカード系システム構築における主要コンピューティングリソース検討

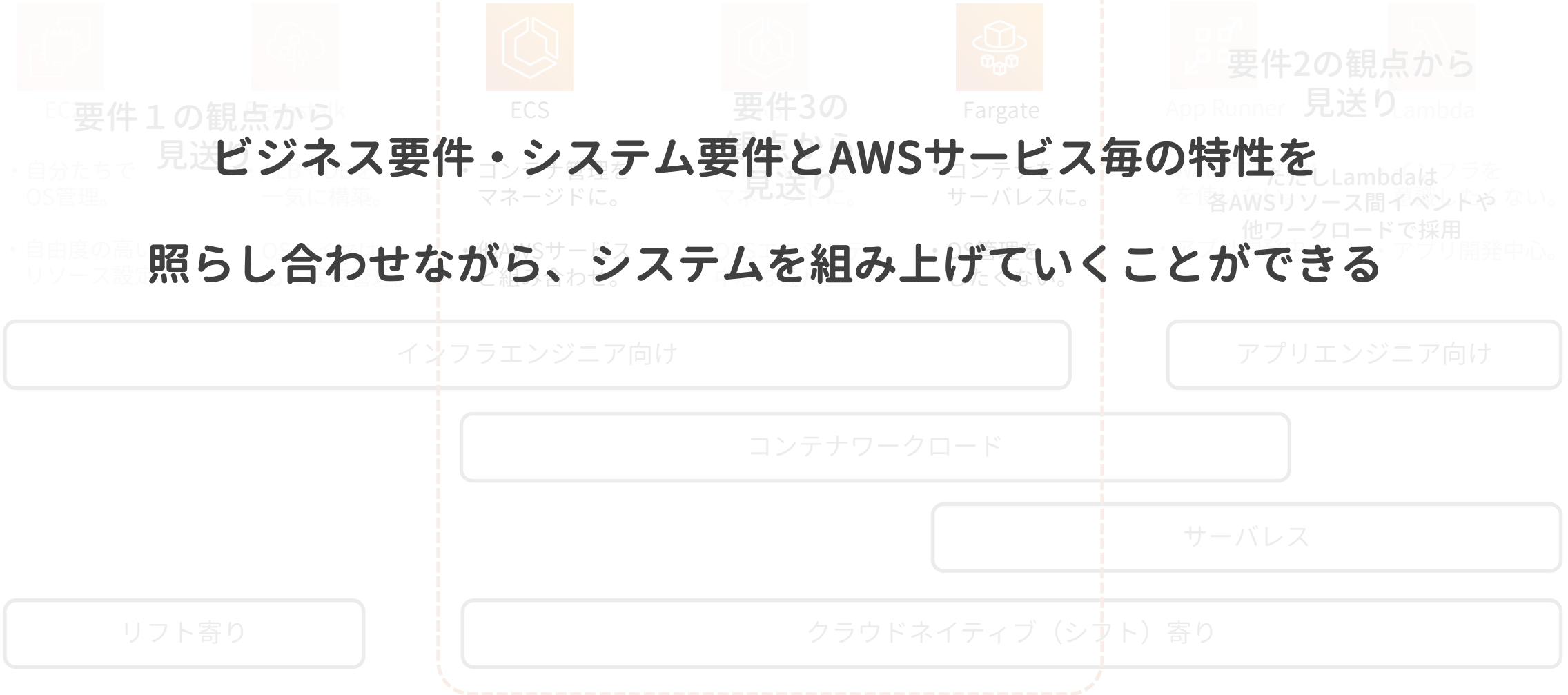


事例：クレジットカード系システム構築における主要コンピューティングリソース検討

事例：クレジットカード系システム構築における主要コンピューティングリソース検討

ビジネス要件・システム要件とAWSサービス毎の特性を

照らし合わせながら、システムを組み上げていくことができる



AWSを採用する代表的な３つの理由

多様な
サービス

多くのビジネス
要件に対して
柔軟に対応できる。

豊富な
アップデート

活発な
コミュニティ

AWSを採用する代表的な３つの理由

多様な
サービス

豊富な
アップデート

活発な
コミュニティ

多くのビジネス
要件に対して
柔軟に対応できる。

アップデートを追いながら、継続的なシステムの最適化につなげる

- AWSのアップデートは90%以上が利用者のニーズベース
  - ロードマップの変更が常になされている

- アップデートによる最適化を予め念頭に入れておく
  - 恩恵だけでなく、もちろんリスクも。

事例：クレジットカード系システム構築におけるAWSアップデートの恩恵

当初は以下の要件に関する設計考慮が発生していた。

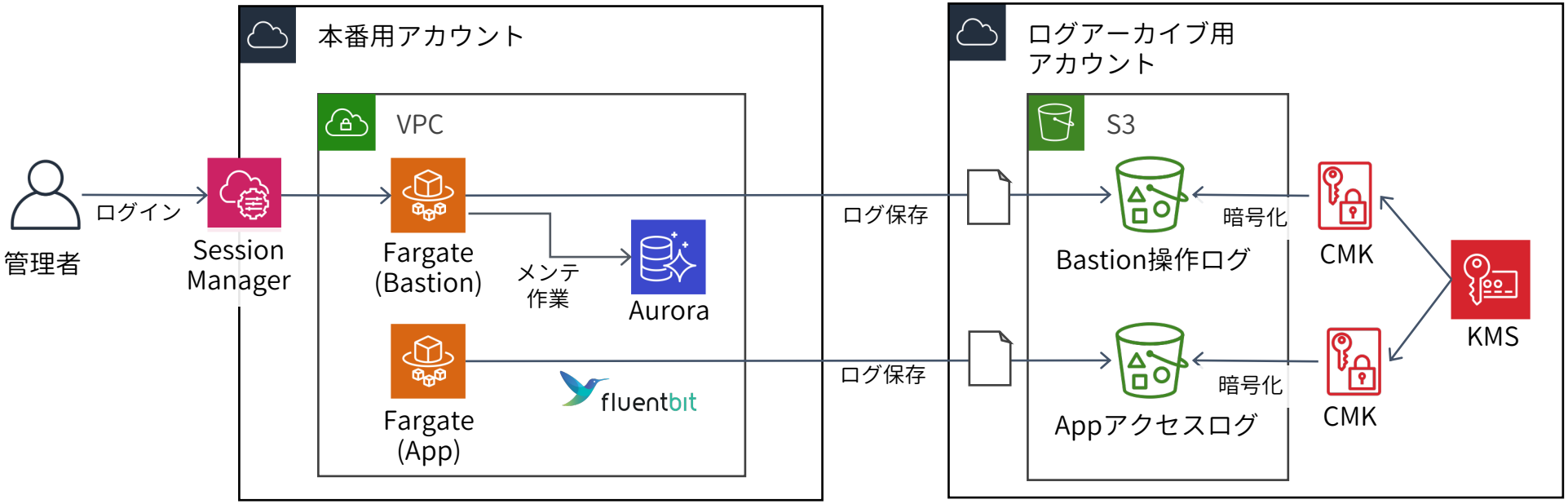
※8.1.8 セッションのアイドル状態が 15 分を超えた場合、ターミナルまたは セッションを再度アクティブにするため、
ユーザの再認証が必要となる。
10.1 システムコンポーネントへのすべてのアクセスを各ユーザにリンクする監査証跡を確立する
10.5.2 監査証跡ファイルを不正な変更から保護する
10.5.3 監査証跡ファイルは、変更が困難な一元管理ログサーバまたは媒体に即座にバックアップする

※引用：https://ja.pcisecuritystandards.org/_onelink_/pcisecurity/en2ja/minisite/en/docs/PCI_DSS_v3_2_1_JA-JP.pdf

事例：クレジットカード系システム構築におけるAWSアップデートの恩恵

当初は以下の要件に関する設計考慮が発生していた。

※8.1.8 セッションのアイドル状態が 15 分を超えた場合、ターミナルまたは セッションを再度アクティブにするため、
ユーザの再認証が必要となる。
10.1 システムコンポーネントへのすべてのアクセスを各ユーザにリンクする監査証跡を確立する
10.5.2 監査証跡ファイルを不正な変更から保護する
10.5.3 監査証跡ファイルは、変更が困難な一元管理ログサーバまたは媒体に即座にバックアップする

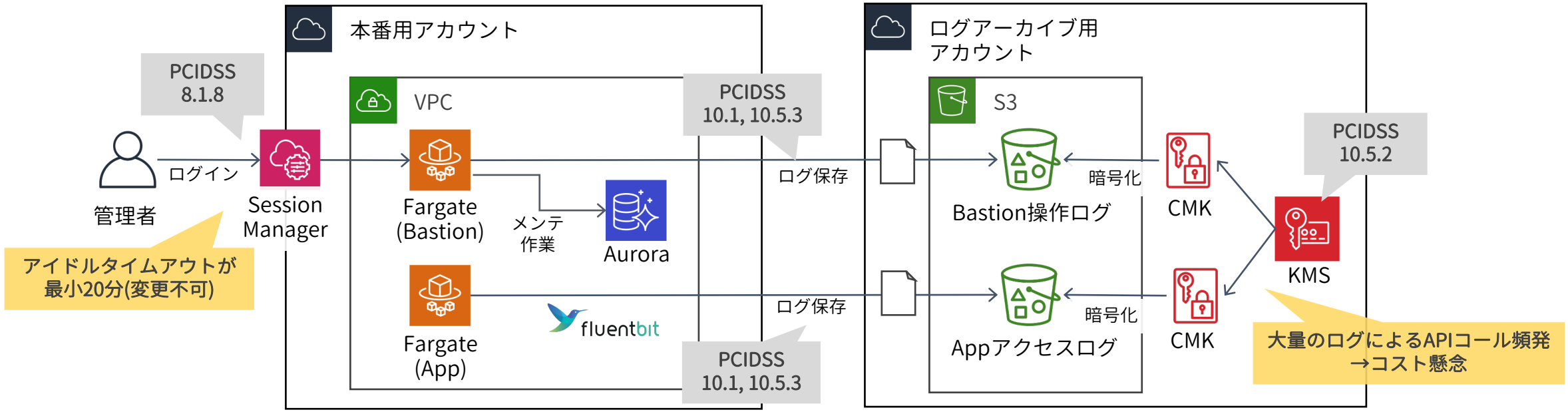


※引用：https://ja.pcisecuritystandards.org/_onelink_/pcisecurity/en2ja/minisite/en/docs/PCI_DSS_v3_2_1_JA-JP.pdf

事例：クレジットカード系システム構築におけるAWSアップデートの恩恵

当初は以下の要件に関する設計考慮が発生していた。

※8.1.8 セッションのアイドル状態が 15 分を超えた場合、ターミナルまたは セッションを再度アクティブにするため、ユーザの再認証が必要となる。
10.1 システムコンポーネントへのすべてのアクセスを各ユーザにリンクする監査証跡を確立する
10.5.2 監査証跡ファイルを不正な変更から保護する
10.5.3 監査証跡ファイルは、変更が困難な一元管理ログサーバまたは媒体に即座にバックアップする



※引用：https://ja.pcisecuritystandards.org/_onelink_/pcisecurity/en2ja/minisite/en/docs/PCI_DSS_v3_2_1_JA-JP.pdf

事例：クレジットカード系システム構築におけるAWSアップデートの恩恵

Now customize the idle session timeout value and stream session logs to Amazon CloudWatch Logs for Session Manager

Posted On: Nov 17, 2020

Session Manager, a capability of AWS Systems Manager, now offers customers greater control over how long sessions remain idle before being terminated automatically. This feature can help you meet compliance requirements, such as PCI Requirement 8.1.8, which requires that users reauthenticate if a session is idle for more than 15 minutes.

Additionally, customers can now stream session logs continuously to CloudWatch for the duration of a session, instead of waiting until the session is terminated. The logs are structured as JSON messages, and
identify the user initiating the session, the instance and session IDs, and the commands and output from the session. The ability to receive and process structured logs continuously throughout the duration of the session provides you with improved visibility into user activity. Using the structured logs, you can easily search for conditions such as session initiation or the use of a specific command, to help analyze and troubleshoot session activity.

To get started, in the navigation pane of the Session Manager console, in the navigation pane, choose Preferences. You can customize the idle session timeout value in the General Preferences. You can enable streaming logs by enabling logs in the CloudWatch logging section, and then choosing Stream session logs
as the logging option.

Session Manager is available in all AWS Regions where AWS Systems Manager is available. To learn more about Session Manager, see the Session Manager documentation. For information about AWS Systems Manager, see our product detail page.

Amazon S3 Bucket Keys reduce the costs of Server-Side Encryption with AWS Key Management Service (SSE-KMS)

Posted On: Dec 1, 2020

Amazon S3 Bucket Keys reduce the request costs of Amazon S3 server-side encryption (SSE) with AWS Key Management Service (KMS) by up to 99% by decreasing the request traffic from S3 to KMS. With a few clicks in AWS Management Console and no changes to your client applications, you can configure your buckets to use an S3 Bucket Key for KMS-based encryption on new objects.

Workloads that access millions or billions of objects encrypted with SSE-KMS can generate large request volumes to KMS. This is because KMS-encrypted objects in S3 use an individual KMS key and S3 makes a call to KMS for each read and write request to these objects. With S3 Bucket Keys, instead of an individual KMS key for each KMS encrypted object, a bucket-level key is generated by KMS. S3 uses this bucket key to create unique data keys for objects in a bucket, avoiding the need for additional KMS requests to complete encryption operations. This results in reduction of request traffic from S3 to KMS, allowing you to access encrypted objects in S3 at a fraction of the previous cost. S3 Bucket Keys can be configured through the S3 Management Console, SDK, or API. You will also have the option to override the S3 Bucket Key configuration for specific objects in a bucket with an individual per-object KMS key using the API and SDK.

Amazon S3 Bucket Keys are available at no additional cost in all commercial AWS Regions, including the AWS GovCloud, the AWS China (Beijing) Region, operated by Sinnet, and the AWS China (Ningxia) Region, operated by NWCD. To learn more about S3 Bucket Keys visit SSE-KMS documentation.

※引用：https://aws.amazon.com/about-aws/whats-new/2020/11/aws-customize-the-idle-session-timeout-and-stream-session-logs-t/
https://aws.amazon.com/jp/about-aws/whats-new/2020/12/amazon-s3-bucket-keys-reduce-the-costs-of-server-side-encryption-with-aws-key-management-service-sse-kms

事例：クレジットカード系システム構築におけるAWSアップデートの恩恵

Now customize the idle session timeout value and stream session logs to Amazon CloudWatch Logs for Session Manager

Posted On: Nov 17, 2020

Session Manager, a capability of AWS Systems Manager, now offers customers greater control over how long sessions remain idle before being terminated automatically. This feature can help you meet compliance requirements, such as PCI Requirement 8.1.8, which requires that users reauthenticate if a session is idle for more than 15 minutes.

Additionally, customers can now stream session logs continuously to CloudWatch for the duration of a session, instead of waiting until the session is terminated. The logs are structured as JSON messages, and
identify the user initiating the session, the instance and session IDs, and the commands and output from the session. The ability to receive and process structured logs continuously throughout the duration of the session provides you with improved visibility into user activity. Using the structured logs, you can easily search for conditions such as session initiation or the use of a specific command, to help analyze and troubleshoot session activity.

To get started, in the navigation pane of the Session Manager console, in the navigation pane, choose Preferences. You can customize the idle session timeout value in the General Preferences. You can enable streaming logs by enabling logs in the CloudWatch logging section, and then choosing Stream session logs
as the logging option.

Session Manager is available in all AWS Regions where AWS Systems Manager is available. To learn more about Session Manager, see the Session Manager documentation. For information about AWS Systems Manager, see our product detail page.

Session Managerのアイドルタイムアウトを
15分まで短くできるように。

Amazon S3 Bucket Keys reduce the costs of Server-Side Encryption with AWS Key Management Service (SSE-KMS)

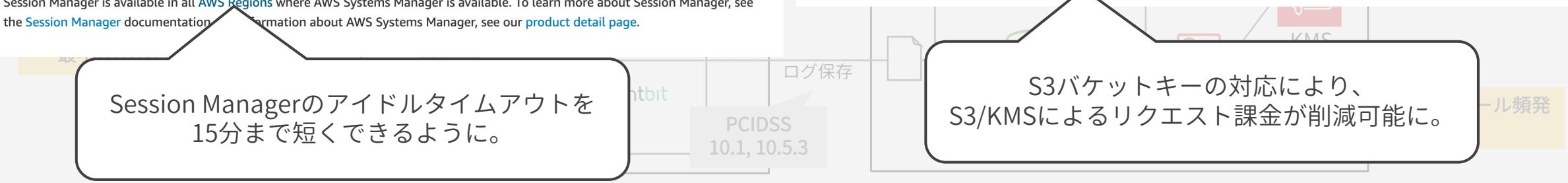
Posted On: Dec 1, 2020

Amazon S3 Bucket Keys reduce the request costs of Amazon S3 server-side encryption (SSE) with AWS Key Management Service (KMS) by up to 99% by decreasing the request traffic from S3 to KMS. With a few clicks in AWS Management Console and no changes to your client applications, you can configure your buckets to use an S3 Bucket Key for KMS-based encryption on new objects.

Workloads that access millions or billions of objects encrypted with SSE-KMS can generate large request volumes to KMS. This is because KMS-encrypted objects in S3 use an individual KMS key and S3 makes a call to KMS for each read and write request to these objects. With S3 Bucket Keys, instead of an individual KMS key for each KMS encrypted object, a bucket-level key is generated by KMS. S3 uses this bucket key to create unique data keys for objects in a bucket, avoiding the need for additional KMS requests to complete encryption operations. This results in reduction of request traffic from S3 to KMS, allowing you to access encrypted objects in S3 at a fraction of the previous cost. S3 Bucket Keys can be configured through the S3 Management Console, SDK, or API. You will also have the option to override the S3 Bucket Key configuration for specific objects in a bucket with an individual per-object KMS key using the API and SDK.

Amazon S3 Bucket Keys are available at no additional cost in all commercial AWS Regions, including the AWS GovCloud, the AWS China (Beijing) Region, operated by Sinnet, and the AWS China (Ningxia) Region, operated by NWCD. To learn more about S3 Bucket Keys visit SSE-KMS documentation.

S3バケットキーの対応により、
S3/KMSによるリクエスト課金が削減可能に。

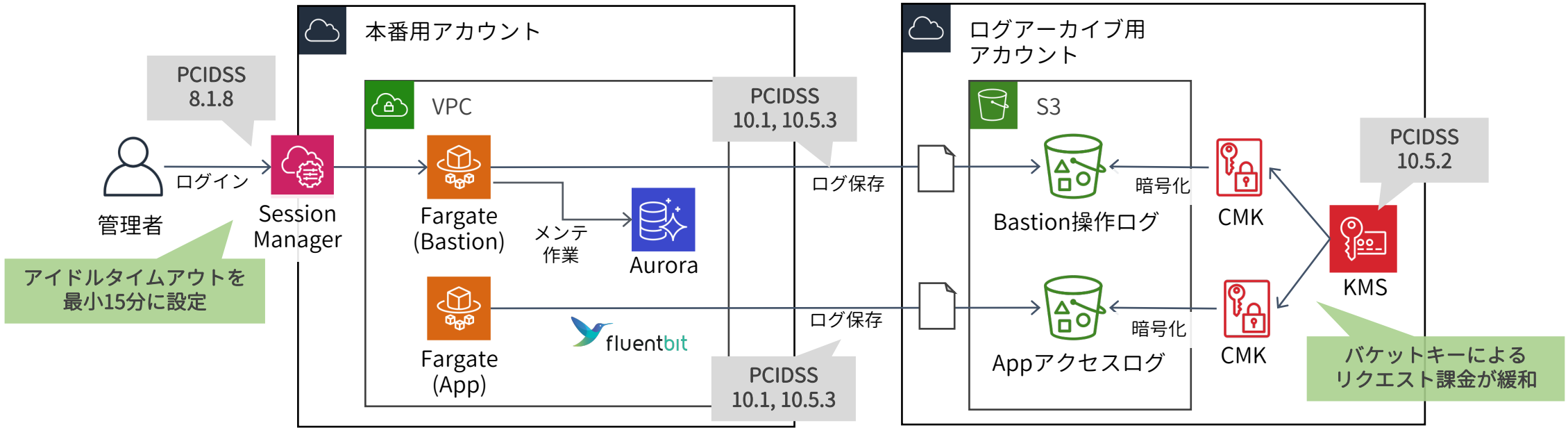


※引用：https://aws.amazon.com/about-aws/whats-new/2020/11/aws-customize-the-idle-session-timeout-and-stream-session-logs-t/
https://aws.amazon.com/jp/about-aws/whats-new/2020/12/amazon-s3-bucket-keys-reduce-the-costs-of-server-side-encryption-with-aws-key-management-service-sse-kms

事例：クレジットカード系システム構築におけるAWSアップデートの恩恵

当初は以下の要件に関する設計考慮が発生していた。

8.1.8 セッションのアイドル状態が 15 分を超えた場合、ターミナルまたは セッションを再度アクティブにするため、ユーザの再認証が必要となる。
10.1 システムコンポーネントへのすべてのアクセスを各ユーザにリンクする監査証跡を確立する
10.5.2 監査証跡ファイルを不正な変更から保護する
10.5.3 監査証跡ファイルは、変更が困難な一元管理ログサーバまたは媒体に即座にバックアップする



※引用：https://ja.pcisecuritystandards.org/_onelink_/pcisecurity/en2ja/minisite/en/docs/PCI_DSS_v3_2_1_JA-JP.pdf

事例：クレジットカード系システム構築におけるAWSアップデートの恩恵

当初は以下の要件に関する設計考慮が発生していた。

8.1.8 セッションのアイドル状態が 15 分を超えた場合、ターミナルまたは セッションを再度アクティブにするため、
ユーザの再認証が必要となる。
10.1 システムコンポーネントへのすべてのアクセスを各ユーザにリンクする監査証跡を確立する
10.5.2 監査証跡ファイルを不正な変更から保護する

AWSのアップデートはビジネスニーズと併走する形で実現されていく。

利用者側が変化に寛容になることで、恩恵を最大限受け入れられる。

※引用：https://ja.pcisecuritystandards.org/_onelink_/pcisecurity/en2ja/minisite/en/docs/PCI_DSS_v3_2_1_JA-JP.pdf

AWSを採用する代表的な３つの理由

多様な
サービス

多くのビジネス
要件に対して
柔軟に対応できる。

豊富な
アップデート

変化を受け入れることで
システムの最適化を
維持できる。

活発な
コミュニティ

AWSを採用する代表的な３つの理由

多様な
サービス

多くのビジネス
要件に対して
柔軟に対応できる。

豊富な
アップデート

変化を受け入れることで
システムの最適化を
維持できる。

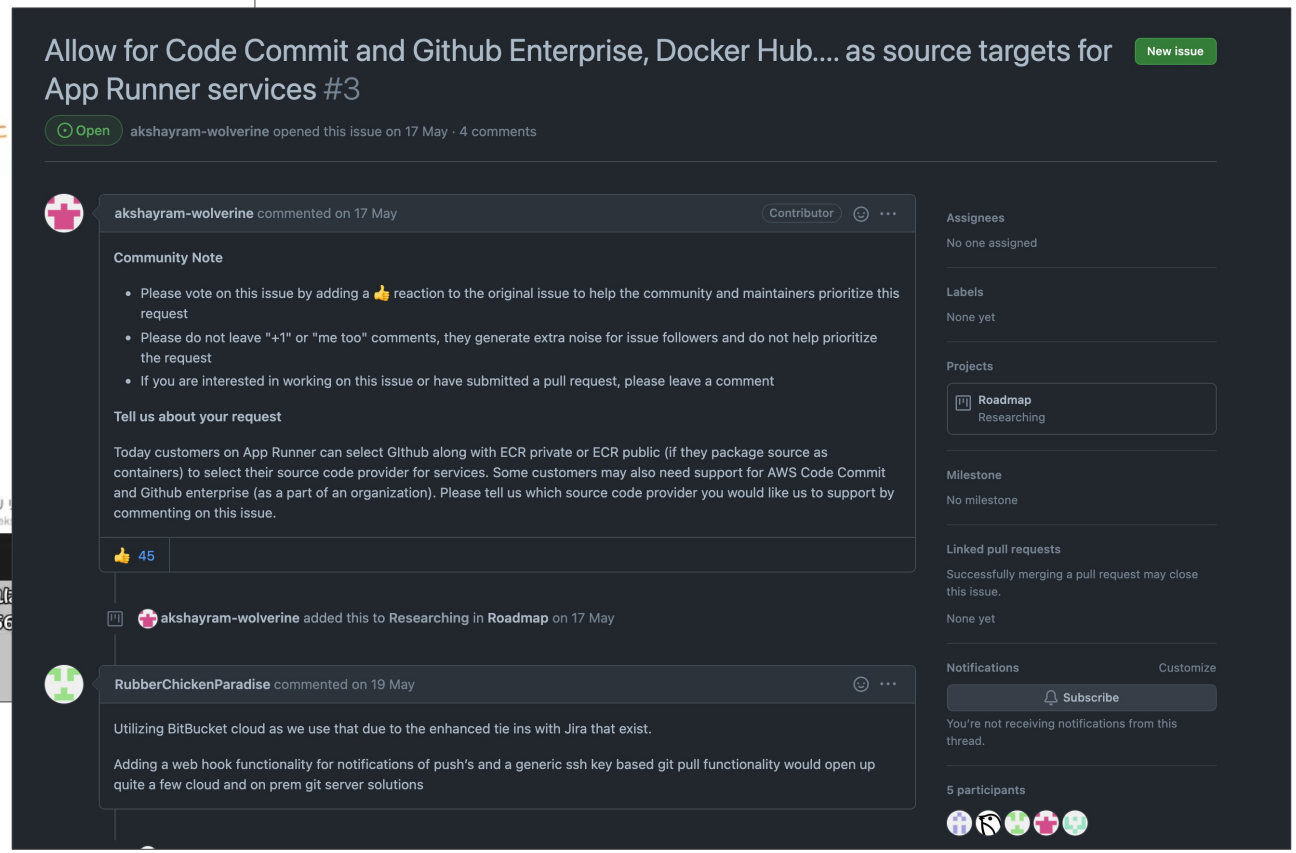
活発な
コミュニティ

コミュニティ・ミートアップは活きたユースケースを知る最高の場所

- AWSはグローバルレベルでコミュニティが非常に活発
- AWSで自分たちのビジネスにあったbuilding blockをするためには、

  多くのユースケースに触れることが鍵
  - block単体に詳しくても、組み上げ方が適切でなければシステムは実現できない。
  - ビジネスの前提や制約事項でアプローチが異なる。
  - 組み合わせの良し悪しに関する嗅覚を養える。
  - 多様なユースケースを知る ＝ 多様な価値の提供方法を獲得する

- キューレートされた情報を効率良くキャッチアップできる
  - 膨大な情報からアーキテクティングに必要なポイントを俯瞰できる
  - 多くのベストプラクティスが学べる



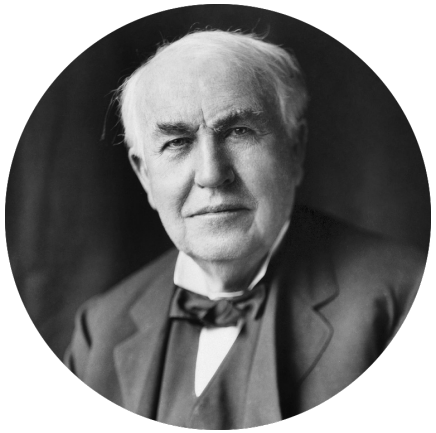
※引用：https://www.youtube.com/watch?v=sQWhn0S_qbY
https://github.com/aws/apprunner-roadmap/issues/3

ところで、それは誰にとってのベストプラクティス？

- ベストプラクティスとは、ある結果を得るために実践的な試行から
  一番よいと考えられるもの。
  - うまくいかないことには、理由がある。うまくいくことにも、条件がある。
    得られる結果を正しく捉え、自分たちのユースケースと照らし合わせて判断することが大切。
    - すなわち、それは自分たちにとってベストプラクティスではないことも往々にしてある。

- ベストプラクティスを知ること自体は大変有意義なこと。
  - 先人たちの苦労(うまくいかない理由)が詰まっている。

“うまくいかない”を知ることの重要さは今も昔も変わらない。
そしてそれは組織の財産となり、利用者への価値となる。



I have not failed.

I’ve just found 10,000 ways that won’t work.

- Thomas Edison -

※引用：https://ja.wikipedia.org/wiki/%E3%83%88%E3%83%BC%E3%83%9E%E3%82%B9%E3%83%BB%E3%82%A8%E3%82%B8%E3%82%BD%E3%83%B3

AWSを採用する代表的な３つの理由

多様な
サービス

多くのビジネス
要件に対して
柔軟に対応できる。

豊富な
アップデート

変化を受け入れることで
システムの最適化を
維持できる。

**活発な
コミュニティ**

多様なユースケースに触れ、
ビジネスにマッチした
組み合わせを知る場がある。

まとめ

「AWSを選んだ理由、それはね、、」

多様な
サービス

×

豊富な
アップデート

×

活発な
コミュニティ

多くのビジネス
要件に対して
柔軟に対応できる。

変化を受け入れることで
システムの最適化を
維持できる。

多様なユースケースに触れ、
ビジネスにマッチした
組み合わせを知る場がある。

「なんでAWS選んだんですか？」

「Thank you for your attention!」