

[Sreakeサービス紹介] SRE(Site Reliability Engineering)支援

はじめに

1. SREの定義

SRE = Google発のIT信頼性を向上させる技術的アプローチ

サービス (= サイト) が顧客から信頼を勝ち得るには、機能そのものが優れているだけでなく、

- ・ リリース頻度の高さ (頻度高く機能がUpdateされているか)
- ・ スケーラビリティの高さ (ユーザーが増えても耐えられるか)
- ・ 復旧の素早さ (障害からリカバリーまでの時間)

など、顧客が利用したいタイミングで意図した通り快適に機能が提供できているかどうか、といった「サイト信頼性」が重要。

その「サイト信頼性」に向けてソフトウェアエンジニアが設計やアプローチ、チームビルディングを行うことがSRE

*DevOpsを踏まえた具体的な実装方法がSRE



SRE

開発者とインフラで同じツールや技術を使い、オーナーシップを共有する
(= 顧客への提供価値最大化を目指す)

失敗/トラブルとリリースのバランスを取る
失敗のコストを減らし、迅速な行動を促す
(= 柔軟性を保ちつつ、機会損失を最小限にし、顧客の信頼を得る)

システムに長期的な価値をもたらす仕事に集中するために、手作業を最小限にする
(= トラブルの火種、手作業を減らし、自動化により、顧客の信頼を得る)

システム運用はソフトウェアで解決でき、を前提に全ての運用を測定/可視化して判断、行動する
(= データドリブンな行動で継続的且つ客観的な運用になり顧客の信頼を得られる)

2. SREが求められる背景 (1)

- **ビジネスイノベーションの加速**

- 顧客ニーズの変化・多様化を受けて、多くの企業は顧客体験の向上に向けた新しいサービスの提供や、機能拡張を柔軟にタイムリーに行う必要性に迫られています。
一方で、ITシステムの変更はITシステムの停止リスクを高め、企業にとって大きな経済損失や顧客満足度の低下に繋がりがねません。ITシステムのイノベーション・機能開発と、安定稼働・信頼性の確保といった、トレードオフの関係にある両者のバランスを取ることが企業には求められます。
- しかしながら、従前のITシステムや運用では以下のような課題から、イノベーションと信頼性といった相反する両者のバランスを取ることが困難です。
 - 開発と運用との間で意思疎通が行われていない
 - リリース作業に伴う影響が多岐にわたる
 - リリース作業が手作業で行われている
- SREの考え方は、こういったイノベーション向上や、信頼性確保を阻害する要因を、SLI/SLOの導入や、マイクロサービス化、リリース作業の自動化等によって解消し、信頼性とイノベーションのバランスを取りながら顧客提供価値を最大化を目指すものです。

⇒信頼性を保ちながら、サービスのイノベーションスピードを向上させることで顧客提供価値を最大化する

2. SREが求められる背景 (2)

- ITシステム運用の高度化・効率化

- 現在のITシステムは、従来のオンプレ環境下でのITシステム管理から、クラウドサービスの利用、マルチクラウド化等、運用管理対象の環境が多岐に及ぶとともに、マイクロサービスアーキテクチャの活用等により、複数のサービスを組合せた複雑なシステムが構築されるようになっていきます。ITシステムが多様化する一方で、運用については従前の運用が踏襲され、以下のような状況が常態化している組織も多いのでは無いでしょうか。
 - 手順書ベースのオペレーションの繰り返しが発生している
 - 属人化に伴い、特定担当者への過負荷が生じている
 - リリース作業の都度、作業手順・スケジュールの整備が必要となっている
 - 対処要否の分からないアラートが都度発生している 等
- 従来の運用方法では、多様化・複雑化するITシステムを効率的に管理することは困難になっている中、SREの考え方は、前述のような運用課題をソフトウェア開発と運用の両方の知識を持つ人材によって解消し、効率的なシステム運用の実現を目指すものです。

⇒ソフトウェアエンジニアリングによる効率的な運用を介したサービスの信頼性を向上させる

3. SREの構成要素概要

- **リスクの受容/SLOの定義**
 - サービスに対して求められる信頼性や、許容されるリスクの程度を定量的に定め（SLI/SLO）、定量化された値に基づいて、開発⇄運用間が同じ基準（共通言語）で、リスクを管理することで信頼性とイノベーションスピードのバランスをとるもの。
- **トイルの撲滅/自動化**
 - 以下に該当するような、手作業（トイル）を自動化により削減するもの。
（トイルの定義）手作業である/繰り返される/非戦術的である/自動化できる/長期的な価値を持たない/サービスの成長に併せて増加する
- **リリースエンジニアリング**
 - 再現可能で自動化されたリリースプロセスを整備するもの。
- **モニタリング**
 - 長期的なトレンドの把握や適切なアラートによる問題解決を行うため、モニタリング環境の整備・高度化を行っていくもの。システムの状態のみならず、コストや、セキュリティ・品質といったシステム運用開発に関わるものを可視化し、データドリブンな世界を志向する。
- **単純さ**
 - マイクロサービス化や、標準アーキテクチャの整備・採用等により、シンプルなシステムを構築するもの。

4. SRE導入によるシステム開発・運用における様々なメリット

SREの原則	開発上のメリット	運用上のメリット
リスクの受容 SLOの定義	<ul style="list-style-type: none">・ イノベーションの促進: 許容されるリスク範囲を明確にし、SLOを定量的に定めることで、開発チームはシステムの信頼性を明確に意識しながら、新しい機能開発や技術革新を推進することが可能になる。・ 迅速な意思決定: リスクとベネフィットを天秤にかけ、迅速かつ確かな意思決定を行うことができる。・ 開発効率の向上: 開発チームに適切な権限を与えることで、自律性と責任感が高まり、開発効率を向上させることにつながる。・ 開発と運用の連携強化: SLOの共有を通じて、開発Tと運用Tが共通目標を理解し、連携してシステムの信頼性向上に取り組むことができる。	<ul style="list-style-type: none">・ システム状態の可視化: SLOにより、システムの状態を可視化し、問題発生を早期に検知することができる。・ 問題解決の迅速化: SLOに基づいて問題の優先順位を判断し、迅速に問題解決に取り組むことができる。・ コスト削減: 過剰な運用コストを抑え、効率的な運用を実現することができる。・ ユーザー満足度の向上: SLO達成により、ユーザー満足度の向上につなげることができる。
トイルの撲滅 自動化	<ul style="list-style-type: none">・ 開発者の創造性向上: トイルを撲滅することで、開発者は創造的な作業に集中できるようになり、より革新的な機能開発が可能になる。・ 開発効率の向上: トイルを自動化することで、開発作業を効率化し、開発スピードを向上させることができる。(テスト自動化・インフラ自動化)・ 開発者満足度の向上: トイルを撲滅することで、開発者満足度を向上させることができる。	<ul style="list-style-type: none">・ 運用者のモチベーション向上: トイルを撲滅し、自動化することで、運用者の負荷を軽減し、モチベーションを向上させることができる。・ 人材不足の解消: 運用にかかる人員を削減し、人材不足の解消に貢献することができる。・ 人為的ミスの防止: 自動化により、人為的ミスを減らし、システムの安定性を向上させることができる。

4. SRE導入によるシステム開発・運用における様々なメリット

SREの原則	開発上のメリット	運用上のメリット
リリースエンジニアリング	<ul style="list-style-type: none">・ 開発と運用の連携強化と安定したリリース: リリースエンジニアリングを通じて、開発・運用間が連携し、スムーズで安定したリリースを実現することができる。・ 迅速なリリース: CI/CD環境の整備により、開発サイクルを短縮し、迅速なリリースを実現できる。	<ul style="list-style-type: none">・ システム障害の減少:・ リリースエンジニアリングにより、システム障害を減少させることができる。・ ユーザー満足度の向上: 安定したリリースと迅速なリリースにより、ユーザー満足度を向上させることができる。
分散システムのモニタリング	<ul style="list-style-type: none">・ システム動作の可視化: 分散システムのモニタリングにより、システム全体の動作状況を可視化し、問題を早期に発見することができる。・ パフォーマンスの向上: モニタリング結果に基づいて、システムのパフォーマンスを分析し、改善することができる。	<ul style="list-style-type: none">・ 問題解決の迅速化: 問題発生時に迅速に原因を特定し、解決することができる。・ システム全体の安定性の向上: モニタリング結果に基づいてシステムを改善することで、システム全体の安定性を向上させることができる。・ 迅速な意思決定: 定量的に状態を捉えることで、改善箇所の特定や、迅速かつ確かな意思決定を行うことができる。
7.単純さ	<ul style="list-style-type: none">・ 開発の容易化・迅速化: シンプルなシステムは開発が容易で、変更やメンテナンスも容易となり、開発サイクルの短縮、迅速なリリースに寄与する。・ デバッグの容易化: シンプルなシステムはデバッグが容易で、問題を迅速に解決することができる。	<ul style="list-style-type: none">・ 運用の容易化: シンプルなシステムは運用が容易で、運用負荷を軽減することに繋がる。・ 問題解決の容易化: シンプルなシステムは問題解決が容易で、システム障害発生時の復旧を迅速化することができる。・ コスト削減: インフラや人員のコストを削減することができる。

1. SRE導入時の課題

- SRE導入時の課題

- 既存システムに対して適用する際に、どこから初めて良いか分からない。
- SREに精通した人材が自社にいない。リソースが不足自社内での推進が難しい。
- SRE導入による定量的な効果（ROI）について自社では十分な説明が行えない。
- SRE導入に対応した運用の仕組み・体制が必要となる。
- 技術面での課題が大きく、簡単に導入することが困難。



**SRE導入を希望する
お客様から頂くニーズ**

- 外部有識者として、伴走型でSREの導入を支援して欲しい
- 将来的には、お客様エンジニアへの内製化支援をして欲しい
- SRE、CCoE、Platform Engineerなどのチーム作りから一緒に進めて欲しい
- インフラ・アプリケーション・セキュリティなど含めて全方位で対応して欲しい

1. SREを導入する際のポイント

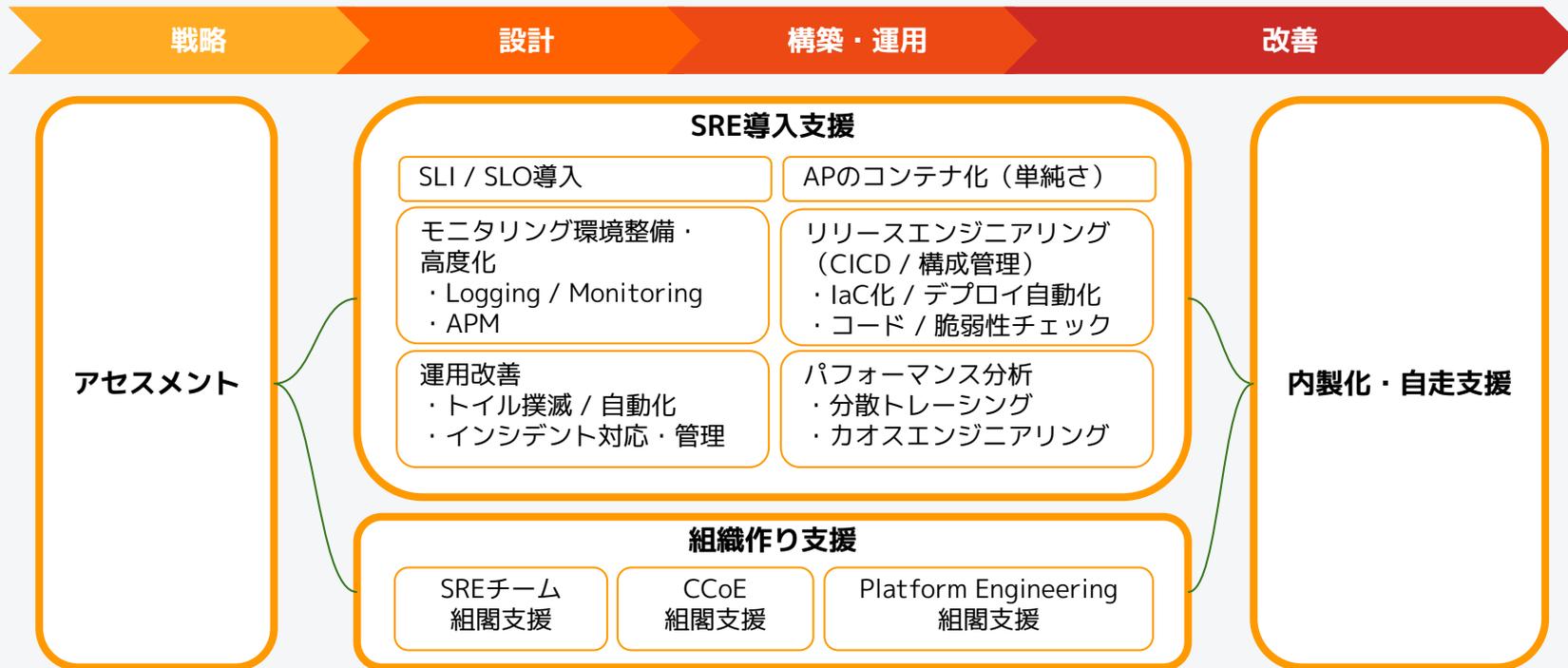
- **段階的な導入**
 - まずは小規模なプロジェクトや特定のサービスから始め、徐々に範囲を拡大していくことが重要。
 - 段階的なアプローチにより、組織が新しい技術やプラクティスに適応し、リスクを最小限に抑えながら、SRE導入を進めることができる。
- **組織文化の変革**
 - SREの導入は、技術的な変化だけでなく、組織文化の変革も伴う。
 - エンジニアやスタッフが新しい技術やプラクティスを受け入れ、活用するためには、専門的な技術サポートや教育が不可欠。
 - イノベーションを促進するため、SRE導入の意義を組織全体で共有する必要がある。



これらの要素を意識し、適切に組み合わせることで、SRE導入による効果を得ることができる。
また、組織文化の変革によって、迅速なシステム開発への意識が高まり、ビジネス競争力の向上にも繋がる。

SRE支援サービスの内容

SREの概念に基づき運用改善、プラットフォーム整備、エンジニアリング内製化を支援



システムが目指すべき定量的（現実的）な目標を定めることで、正しいエンジニアリングができる

リスクと効率のバランス問題

- ・ システム100%の信頼はありえない
（システムは必ず障害を起こす）
- ・ 過度の信頼（安全性）を作るには莫大なコストがかかる
- ・ 安全性を追い求めると機能リリースの頻度が減る（顧客体験が向上しない）
- ・ 顧客はそもそも高い信頼（安全性）と「極端に」高い信頼性の違いはわからない、気づかない

システムが目指す指標をたてる

- SLI(Service Level Indicator)
 - ・ サービスレベル目標
 - ・ 何をもとにシステム良し悪しを判断するか指標となるもの
- SLO(Service Level Objective)
 - ・ サービスレベルに対する社内の目標値
 - ・ 目標を超えるまで不具合解消を優先
- SLA(Service Level Agreement)
 - ・ サービスレベルに対する対外的な保証値

エンジニアリングとして、新規機能を優先すべきか不具合解消を優先すべきか目標が明確になることで、ビジネス側との連携もしやすくなる（共通認識を持つことができる）

トイルを測定し、自動化、振り返りを繰り返すことで顧客信頼向上に向けたエンジニアリング最大化に繋がる

トイル

手作業のシステム業務

長期的な価値を持たない
システム業務

繰り返されるシステム業務

サービスの成長に比例して
増加するシステム業務

自動化可能なシステム業務

戦術的なシステム業務

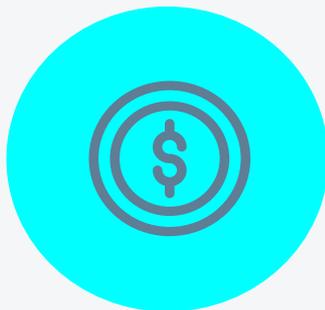
- ・ エンジニアの工数削減
 - ・ エンジニアの士気向上
 - ・ エンジニアチームの生産性向上
 - ・ エンジニアリング（オペレーション）のプロセス明確化
 - ・ ヒューマンエラーの削減
 - ・ 顧客要望のレスポンス改善
- （顧客要望に注力できる時間を確保できる結果、信頼性向上に繋がる）

システム（アプリケーション）の動きを監視し正確に対処できる状態をつくり予測不可能な状態を防ぐ



システムの状態：

外形（ユーザー視点の）監視、メトリクス（システム内部状態の）監視、ログ監視、APM実装（アプリケーションのパフォーマンス監視）などを可能な限り行い、システムの状態を客観的に把握できるようにする



コストの状態：

システムは正常だが、クラウド利用料などのコストが急増することがあるためコストの可視化を行う



セキュリティ・品質管理の状態：

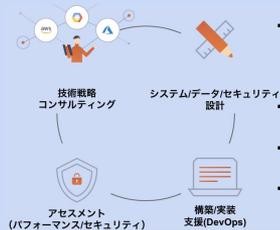
セキュリティの状態可視化、警告通知に対してのエンジニアの対応時間、CICD（継続的なソフトウェアのデリバリー）の実行時間などの開発生産性の可視化を行う

SRE支援サービスの特徴

1. SREの考え方を軸にした全方位型のご支援

- 高度な技術力と幅広い領域の経験を持つエンジニアが多数在籍しており、伴走型でお客様に最適対応のご支援を行います。
- インフラ・アプリケーション・セキュリティなど含め全方位型で対応可能です。
SREの考え方を元に運用まで見据えたご支援を行います。

クラウドネイティブインフラ・セキュリティ支援



- ・ コンテナ化、コンテナ運用ご支援 (Kubernetesに強み)
- ・ CI/CD環境構築支援
- ・ クラウドセキュリティ設計・運用支援
- ・ Observability設計・運用支援



クラウドネイティブアプリケーション開発支援

- ・ コンテナ化対応などクラウドネイティブ対応に関するアプリケーション開発のご支援
- ・ アプリケーションモダナイゼーション支援
- ・ バックエンド開発支援 (Go, Python, TypeScript)
- ・ フロントエンド開発支援 (Vue, React)



GenAI基盤構築支援



- ・ VertexAIシリーズ構築運用内製化支援
- ・ 外部SaaS連携支援
- ・ Gemini API導入支援
- ・ AI CCoE立ち上げ支援
- ・ SRE LLMの提供

データモダナイゼーション支援

- ・ BigQuery/Dataplex データ基盤構築支援
- ・ BI (Looker) 構築をフルスタックで支援
- ・ Snowflakeメインのデータ基盤構築支援
- ・ DBRE支援 (Spanner/AlloyDB)
- ・ NewSQL (TiDB, YugabyteDB) 支援



2. SRE支援領域における豊富な実績



トヨタ自動車様

製造現場におけるAIプラットフォーム内製化を
インフラレイヤーから支援

- ・ Google Cloud×ハイブリッドクラウド
×Kubernetesという難易度の高いPJTを1チーム
で支援
- ・ 取組が評価されGoogle Cloud Next'22に登
壇

インテック様

金融機関向けシステムのインフラ刷新を支援

- ・ AWS×クラウドネイティブ×インフラのコー
ド化を実現
- ・ リリースするだけでなく、その後の保守運用
や組織体制まで見据えた提案と実現し、運用コ
ストが1/3に削減
- ・ アジリティの高い開発運用組織に向けたSRE
のノウハウを提供



NTTデータ様

日本最大クレジットカード決済ゲートウェイの
デジタル化をインフラレイヤーから支援

- ・ マルチクラウド×kubernetes×SREを実践す
る内製チームを1から支援
- ・ 巨大システムでありながら、モダンスタック
且つアジリティの高いスケーラブルな組織運
営、内製化ノウハウも提供

3. 高度な技術力と自走できるメンバー

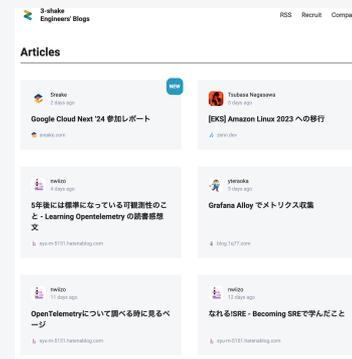
- クラウドネイティブ技術は変化が激しく常に最新の技術領域に対して習得が必要です。Sreakeのエンジニアはこれら最新の技術領域に対するインプットとアウトプットを高速で行いながら、お客様のニーズに対して最適な提案ができるように取り組んでいます。



<弊社で監訳した本>



<最新技術の調査・検証結果のアウトプット>



<各分野の有識者によるアウトプット>

- クラウドネイティブ技術領域のご支援は、お客様にて詳細な要件を出すことが難しいケースが多いです。そのため、弊社メンバーが自走しながらお客様に対して提案型のご支援をさせて頂いております。
- 技術のみであれば業務委託などのご支援で良いケースもありますが、提案型のご支援という点もお客様の評価を頂いております。

4. 組織作り、内製化のご支援

- **CCoE組織作りのご支援**

- CCoEは、クラウド戦略の策定、ベストプラクティスの共有、および組織全体のクラウドへの取り組みを支援するための専門チームであり、クラウドネイティブへの取り組みに対する技術や知見が組織全体で共有・統一され、各開発に対して横断的に展開する際に有効な組織です。
- CCoEは、クラウドガバナンスの確立、コスト管理、セキュリティポリシーの策定など、クラウド導入に関わるあらゆる側面をサポートする役割を持つため、クラウドネイティブの導入・運用と合わせてCCoEの組織作りを行うことが効果的です。
- 弊社では様々な企業のCCoEに対する支援を踏まえ、CCoE組織作りのご支援を行っております。

- **SRE組織作り、採用・育成支援**

- SREは、システムの信頼性と効率を維持するためのエンジニアリングアプローチです。運用にソフトウェアエンジニアリングの手法を取り入れ、様々な開発や運用の自動化、効果的なデータ収集や可視化を行います。SREチームは、開発プロセスの自動化、モニタリング、インシデント管理などを通じて、クラウドネイティブなシステムの安定性と信頼性を維持する役割を持つため、クラウドネイティブの導入・運用と合わせてSREの組織作りを行うことが効果的です。
- 弊社では様々な企業のSRE支援を踏まえ、SRE組織作りの支援、SRE採用・育成の支援を行っております。

- **内製化の支援**

- クラウドネイティブ技術に対するワークショップ、TAP（Tech Accelerarionプログラム）などのご支援が可能です。
- クラウドネイティブ技術を使ったアプリケーションのプロトタイプ開発、アーキテクチャー設計などを短期間でお客様と一緒に進めることで内製化のご支援を行います。

SRE支援のご相談がございましたら
次のお問い合わせ先にご連絡ください。

3>SHAKE

お問い合わせ先：

株式会社スリーシェイク

住所： 東京都新宿区大京町22-1

URL: <https://sreake.com/contact/>

Email: business@3-shake.com

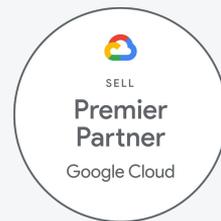
会社概要

会社名 株式会社スリーシェイク
設立日 2015/1/15
代表者 代表取締役社長 吉田 拓真
所在地 東京都新宿区大京町22-1
グランファースト新宿御苑3F・4F

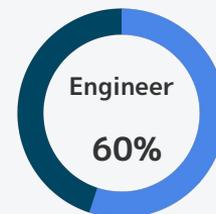
Mission: インフラをシンプルにして
イノベーションが起きやすい世界を作る

Vision: 労苦 (Toil) を無くすサービスを適正な価格で提供し続ける

Value: エンジニアリングレイヤーに横たわる人、手法、ツールが
サイロ化されて労苦が発生しているプロセスをシンプルにし
サービス機能開発に集中できるソリューション
(SRE、DevSecOps、DataOps、HROps) を提供する



Google Cloud、AWSの両方に強みを持ち
SREを軸にご支援





SRE/DevOps



- ・ SRE総合支援からセキュリティ対策を全方位支援
- ・ Geminiを用いた生成AIの活用支援

BizOps



- ・ クラウド型ETL/データパイプラインSaaSの決定版
- ・ あらゆるSaaSをノーコードで連携

SecOps



- ・ ワンストップで脆弱性診断を行うセキュリティ対策SaaS

HR



- ・ ハイスキルフリーランスエンジニア紹介エージェント



IT内製化 / 高度化

クラウドネイティブ化

モダナイゼーション

ITアジリティ向上

Thank You