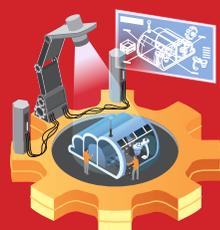


現場コンサルが実践で教える

# 業務システムを クラウド移行するために 押さえておくべき ポイント集

第3版



計画立案からRFP作成まで、現場コンサルが実践解説！

- 1章 計画立案 …… DX Readyとはなにか
- 2章 基盤選定 …… 自社にとって最適なクラウドとは
- 3章 移行方針策定 …… 安全確実な移行方針の作成
- 4章 運用要件 …… クラウドの真価を引き出す運用とは  
パブリッククラウド上で運用すること  
全ての運用作業が自動化できる時代到来
- 5章 組織構成 …… クラウドジャーニーを支える組織
- 6章 RFI/RFP作成 …… 最適な提案を受けるために アセスメント(PoC)フェーズ実施の検討

# 自社の業務システムを DX Readyにしましょう

## DX Readyとは何か

AI、IoT、ビッグデータをはじめとした最新テクノロジーは、かつてないほどのペースでビジネスや人々の生活に変化を起こしている中、企業はビジネス変革へのニーズに応えるべく、最新のテクノロジーを活用した「デジタルトランスフォーメーション(DX)」の実現が期待されています。どのように自社の業務システムをDX実現に取り組める段階(DX Ready)に進化させるべきなのかについて考えてみましょう。

## ✓ 2種類のDX

しばしば守りのIT、攻めのITという言葉が使われますが、DXも同様の守りのDX、攻めのDXという2つのタイプに分類して考えることができます。

### 守りのDX

業務の効率化を中心としたDXを指す。業務プロセスの改善、経営データの可視化、製品やサービスをジャストインタイムで提供するという既存ビジネスの高度化である。例えばテクノロジーを活用した働き方改革なども守りのDXといえる。



### 攻めのDX

今までにない新しい価値を持つ製品やサービス/ビジネスモデルを創出することを指す。新たな決済システム、Uber、Airbnbなどに代表される新しいビジネスモデルの創出が攻めのDXといえる。

どちらのDXを目指すにしても、まずは既存システムを「DX Ready」な状態にすることが重要です。そのために既存システムを「DX Ready」に進化させるためのステップとして、3つのアプローチが効果的です。

## ✓ DX Ready実現に向けて3つの効果的なアプローチ



### リフトアンドシフト手法によるパブリッククラウド移行

クラウドなくしてDXを実現するのは困難であり、まずはクラウドを使いこなす能力を身につける必要があります。そのためには既存システムをクラウド化し、クラウドの特性を学ぶとともに、クラウド特有の構築・運用スキルをユーザー自身も身につけることが重要です。



### SoRとSoEの連携

システムをその目的や特性(例:変更頻度が「数年単位」又は「数週間単位」など)に応じて分けるとともに、各システム間を相互に連携するシステムづくりが肝要です。典型的なパターンとしてSoRとSoEの分離がありますが、その際には各システムが柔軟に変更できるように、疎結合型連携にすることが重要です。



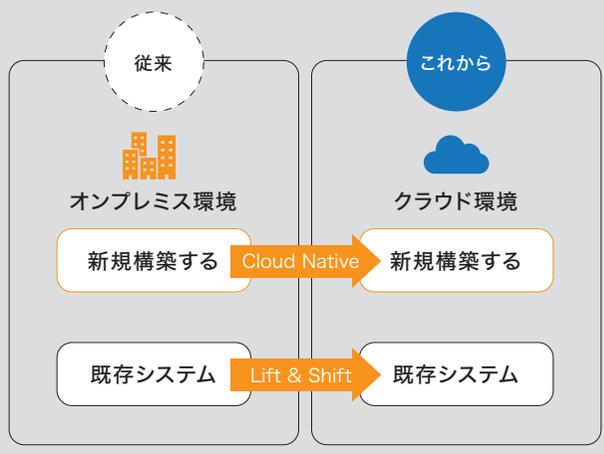
### データドリブンを実現するプラットフォームの構築

直感に頼らないデータを使った判断を実現するためのプラットフォーム作りが必要となり、データのリアルタイム化と、今までのデジタル化されていないアナログデータのデジタル化が必要です。前者を実現するのがERPシステムであり、後者を実現するのがIoT、AI、Data Lakeといったテクノロジーです。

# Tips

## リフトアンドシフトとは

用語の出自としては、2015年にGartner社が発表したクラウド神話Top10の中で、クラウドマイグレーション(クラウド移行)のトピックで使われたのが有名です。その定義については諸説あるものの、「既存システムをそのままクラウド上にあげて、クラウドに最適化させる」とあえて単純化して捉えるとわかりやすい。しばしば対比される手法として「クラウドネイティブ(Cloud Native)」がありますが、こちらはクラウドが保有する機能を最大限活用する前提で、システムやサービスを新たに構築することを指します。



## SoRとSoEとは

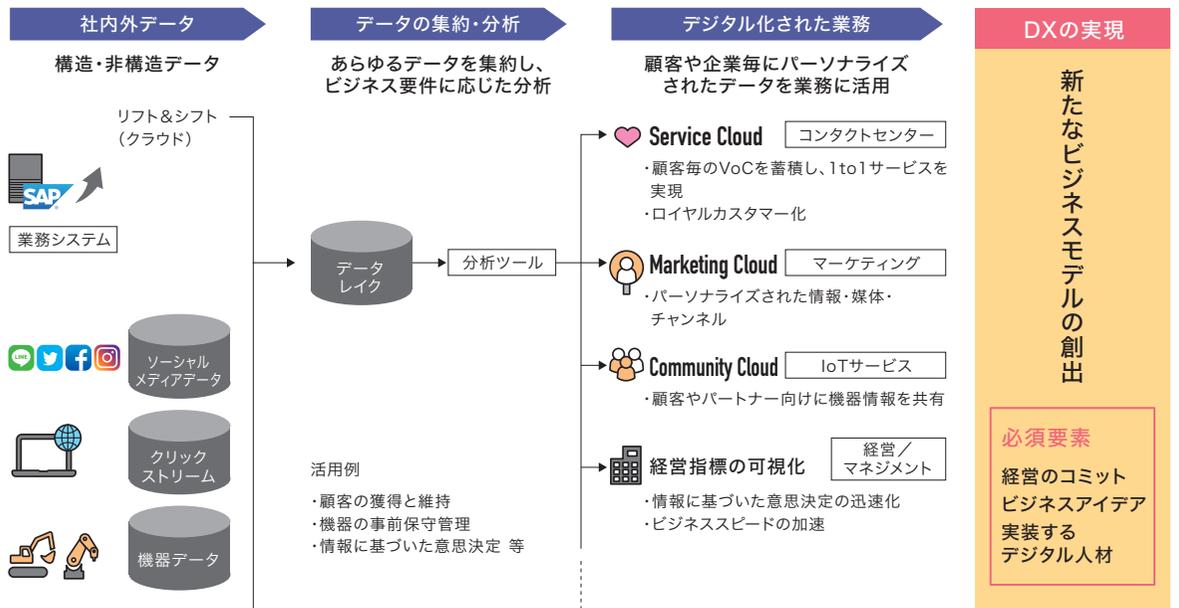
システムを目的と特性により分類した概念であり、SoR (Systems of Record) は情報を正しく「記録する」ためのシステムに対して、SoE (Systems of Engagement) は、顧客や取引先との「絆を作る」ためのシステムです。本概念が登場した背景としては、ヒト・モノ・コトの関係性がデジタル化した世界においては、社内に閉じてサイロ化したシステムと、そこに記録されたデータだけでは企業の競争力向上に陰りが見え始めていることが挙げられます。また最近では、SoRとSoE双方から得られたデータをもとに、新たな洞察や知見を獲得するためのシステムとしてSol (Systems of Insight) の具現化に向けたサービスが充実しつつあります。

企業ITにおけるSoR、SoEシステム比較

	SoR (System of Record)	SoE (System of Engagement)
システム領域	ERPを中心とする企業の基幹システム、決算システム	フロントエンド、スマホアプリ、CRM、UI、IoT
システム化重点項目	安定性重視、効率化	変化する世の中のニーズに対応するためにビジネス成果・スピード重視
開発導入手法	ウォーターフォール型 ビッグバン導入	アジャイル+ウォーターフォール型 スモールスタート
	開発前に要件定義を実施し、仕様を確定させる	要件定義は緩やか、正しい仕様は存在しない。トライ&エラーによるスパイラル開発
	品質確保が最優先	迅速なリリースを優先
導入スタイル	トップダウン型	ボトムアップ型
	IT部門主導	IT部門に限らない

## データドリブンを実現するプラットフォーム像

DXを進めるにあたって最も重要なテーマのひとつが、データドリブン(データ駆動)の実現です。データドリブンとは、マーケティングの世界ではしばしば使われていた言葉であるが、データに基づき次のアクションを起こしていくことを指します。



# 選定時の評価軸を 決めましょう

## 自社にとって最適なクラウドとは

先の目的を達成するには、クラウド自身がそのメリットを最大化するための機能を有している必要があります。従来のオンプレミス環境での基盤選定は、基盤一式が手元にあったためにテクニカル要素が選定基準の主でしたが、クラウドは目に見えないこともあり、それを安全に統治するための各種ガバナンス要素を加味した上で、評価することがポイントとなります。クラウドベンダーの評価にあたり、有効な評価軸とポイントを下記に列挙します。

### ✓ 必須要件

- 1 オンデマンドセルフサービス**  
柔軟なシステム基盤を実現できるように、必要に応じてセルフサービスで各種リソースの構築・操作・設定変更が実施できること。
- 2 柔軟なリソース提供・拡張**  
必要な時にすぐ(2時間以内など)リソースが提供でき、かつ拡張できること、自動拡張のためのオートスケール機能を有していること。また、オンライン構成変更機能(DISK容量変更・性能変更など)があるのが望ましい。
- 3 セキュア通信**  
自社ネットワークとの間で閉域網などを使い、セキュアな通信ができること。また、様々なクライアント(デバイス)や外部サービスから、セキュリティを考慮した通信ができること。
- 4 サービスレベル測定情報が収集可能**  
サービスが適切に実施できているか記録し、そのログが外部へ提供できること。具体的には、操作履歴、稼働履歴、障害履歴、性能情報など。
- 5 柔軟な契約**  
運用コスト最適化のため、秒/分/時間単位で課金ができ、かつ契約拘束期間などの制約がないこと。
- 6 APIによる外部制御**  
APIによってクラウドのコンポーネントを外部から自在に制御できること。
- 7 ユーザー権限**  
クラウド操作(ポータル上の操作及びAPIによる操作の両方)に関して、リソース単位で適切なユーザー権限が設定できること。
- 8 課金管理**  
従量課金のコスト配賦を実行できるようにするため、部門・プロジェクトごとなどユーザーのニーズに応じた柔軟な課金管理を実施できること。
- 9 SLA(Service Level Agreement)の定義**  
99.95%以上などSLAが明確に定義されていること。
- 10 強固なデータ保護・業務継続性**  
国内に物理的に分離された複数のデータセンターを持ち、バックアップデータなどが複数のデータセンターに同期保管されているなど、強固なデータ保護機能があること。
- 11 サービスレベル管理に関する外部監査を受けている**  
サービス業務が適切に行われていることの保証として外部の監査を受けており、かつその情報を開示できること。SOCレポートなどがこれに相当。
- 12 情報セキュリティに関する外部監査を受けている**  
企業の基幹システムに保持する情報の重要性から、情報セキュリティの外部監査をうけていること。ISMS、PCI-DDなどがこれに相当。

## 推奨要件

### ① 自動的なデプロイ/プロビジョニング機能を有していること

クラウド内のリソースを記述する言語と、記述内容を自動的にプロビジョニングできる機能が提供されていること。

また、アプリケーションの継続的デリバリーを支援するサービスが提供されていること。

### ② データドリブンを支えるための基盤

安全確実に構造化・半構造化・非構造化など様々なデータを蓄積するデータストア、データ変換機能、データカタログ機能、様々なBIツールを利用できるデータ分析基盤を有すること。

インテリジェンスな分析を可能とするML/AIなどサービスの提供。

IoTサービスなどクラウド上へデータ収集機能、更にエッジ/デバイスのモニタリング/コントロールを可能とするサービスの提供。

### ③ マイクロサービスを支援するためのサービス

コンテナベースのアプリの迅速な開発と展開が可能なオーケストレーション、セキュリティ保護、監視、および管理するためのさまざまなツールとサービスがマネージド・サービスとして提供されていること。

サーバレスコンピューティングサービスが提供されていること。

DBサービスなど各種マネージド・サービスが提供されていること(PaaS)。

### ④ グローバルネットワーク

海外展開などのために、グローバルで共通のオペレーションができること。また、グローバルに展開されたクラウド拠点間が、クラウドベンダーのネットワークセキュリティを利用して高速かつセキュアに通信ができ、かつネットワークセキュリティを強化する各種サービスが提供されていること。

### ⑤ マーケットプレイス

クラウド上に各種サードパーティ製品を導入、購入を可能とするマーケットプレイス機能が提供されること。更にデータプロバイダーが様々なデータを購入し、クラウド上へ展開できるマーケットプレイスなども今後重要となる。



## Tips

### クラウド時代の新しいアプリ開発における評価軸とは

クラウドの利点を最大限に生かしたアプリケーションを「クラウドネイティブアプリケーション」と呼びます。

従来とは異なるアプローチでアプリ実装する必要がありますが、その際に重要となるポイントをご紹介します。



設計時に  
自動化を組み込む

システムにおいて自動化は常にベストプラクティスの1つであり、自動化されたプロセスは、人間よりもはるかに速くシステムの修復やスケールアップ、デプロイできます。



状態を  
スマートに処理する

可能なかぎりコンポーネントをステートレスにするようにシステムを設計することで、スケールアップを容易に実現するとともに、問題発生時の修復作業を改善します。



マネージドサービスの  
選択

マネージドサービスは時間を大幅に節約し、運用上のさまざまなオーバーヘッドを取り除いてくれます。ビジネスにおける競争力向上のためのスピードを向上します。



多層防御を実践

インターネット接続を前提としているため、常に外部からの攻撃に対処するために各コンポーネントの間で認証を要求する多層防御のアプローチを採用します。



アーキテクチャを  
常に考える

組織のニーズが変わり、IT の環境が変わり、クラウドプロバイダー自体の能力が変わるのに応じてシステムのアーキテクチャを洗練させ、単純化し、改良し続けます。

## どのシステムから移行するのか、 どのようにして移行するのかを考えましょう

### 安全確実な移行方針の作成

クラウド基盤の選定が終わったら、次は移行方針を策定します。

自社システムが仮想マシン (VM) で稼働している場合は、比較的簡便にVMベースで移行することができますが、物理サーバで稼働している場合は、別途考慮が必要となることが多いです。クラウド移行の際には、プロジェクト計画フェーズの初期に、事前検証 (PoC: Proof of Concept) フェーズを設けて実環境で移行検証を実施することが推奨されます。これはクラウドの場合、システムリソースが共有型であるために、移行時間予測や必要なシステムリソースの算出に限界があり、かつそれらがクラウド利用料に直結するためであり、早期に実施することで実行フェーズ以降のプロジェクト計画リスクの低減を図れます。個々のシステム移行イメージが定まったら、移行時期の検討を行います。予め移行時間を予測できれば、すぐに具体的な検討ができます。なお、移行台数が多い場合には、多段階移行シナリオも視野に入れておくことをお勧めします。

### ☑ 「スモールスタート」で始めるのか、「全部のせ」がいいのか

#### スモールスタート型

- 関連子会社、海外拠点向けの比較的小規模な基幹システムをクラウドに移行します。
- 開発環境、検証環境、サンドボックス環境をクラウド上に構築します。
- 移行システム対象を限定することで、移行リスクを低減します。

#### 全部のせ型

- クラウド上に共通基盤を構築し、基幹システムを全て移行します。
- 構築当初から全システム稼働を前提とした基盤設計を行うことで、従来システム毎に実施していた設計コストや作業負荷を削減します。

### ☑ システム全体の共通ビジョンの必要性

まずは一部のシステムだけをクラウドに乗せることを検討するケースが多いのですが、検討を進めるにつれ必ず出てくるテーマが、システム全体に対するビジョンの必要性です。クラウドに対する理解が進み、その使いこなしや特性に慣れるに従い、パブリッククラウドが持つ最新機能やスピード

感を生かすことのメリットを実感するようになります。その結果、システムごとに個別最適化していた従来のやり方とのバランスを取りながら、自社にとって最適な新しい共通基盤のあり方を探ることになります。

### ☑ 最初に移行するのは基幹系? 情報系?

全てのクラウド移行を検討する企業が最初に考えるテーマですが、実際に基幹システムを移行した企業の声をまとめると下記の通りです。

- 先進的なIT技術はクラウドをベースに展開されており、最初に基幹システムを移行することで、最新の周辺ソリューションが適用しやすくなり、効果も出やすい
- 基幹システムは比較的規模が大きいこともあり、基盤コストの削減効果がやすい
- 基幹システムは稼働要件等が厳しく定義されており、最初にクラウドに持っていくことで、主要な技術要件が精査された結果、それ以外のシステムを移行する際に楽になる

## 代表的なサーバ移行方式を理解して、移行イメージを掴みましょう

### ケース1 ソースシステムが仮想サーバーの場合 (V2C)

#### VM Import/Export

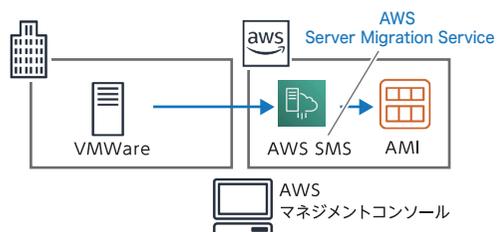
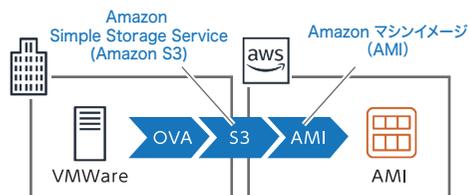
VM Import/Exportを使用して、仮想マシンイメージを既存環境から Amazon Elastic Compute Cloud (Amazon EC2) インスタンスにインポートできます。

#### AWS Server Migration Service (AWS SMS)

AWSマネジメントコンソールを介して数回クリックするだけで、サーバーの移行を開始および管理できます。AWS SMSを利用すると、ライブサーバーのボリュームの増分レプリケーションを自動化、スケジュール設定、追跡できるため、大規模なサーバーの移行を簡単に処理できます。

Microsoft Azure : Azure Migrate

Google Cloud : Migrate for Compute Engine



### ケース2 ソースシステムが物理サーバーの場合、または OS、DBMS等の基盤ソフトウェアを最新バージョンに変更して移行したい場合

#### OS、DB等の標準ツールを利用

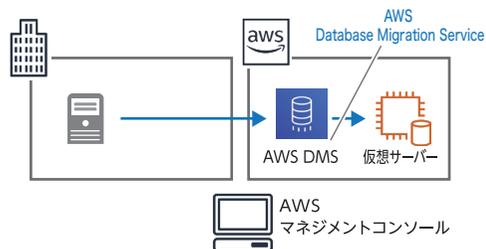
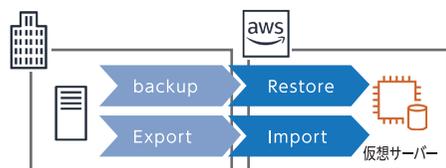
予め新しいOS、DBMSが導入済みのインスタンスに対してOS、データベースの標準バックアップツールやDB Export/Importコマンドを利用してデータを移行します。

#### AWS Database Migration Service (AWS DMS)

AWSマネジメントコンソールで数回クリックするだけで、広く普及しているあらゆる商用およびオープンソースデータベース間のデータの移行を始められます。移行開始後は、移行プロセス中にソースデータベースに発生した変更の自動的なレプリケーションを含む、移行プロセスの複雑な手順すべてがDMSによって管理されます。

Microsoft Azure : Azure Database Migration Service

Google Cloud : Database Migration Service



### ケース3 大量データの移行を伴う複数サーバ群を短期間で移行するために複数の手法を組み合わせる場合

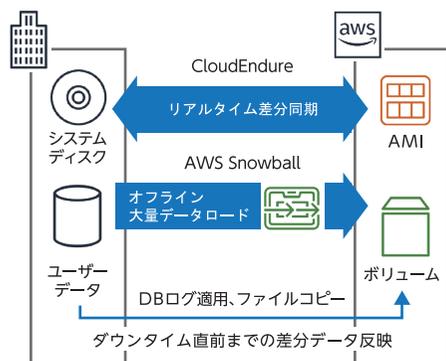
#### サードベンダー製ツール

AWS社が提供するライブマイグレーションツール「CloudEndure」を使って、クラウド上の仮想マシンに複製(リアルタイム差分同期)します。

#### Amazon Snowball

専用ストレージアプライアンスを用いることで、ペタバイト級の大規模データを高速にAWS上へ移行できるデータ転送ソリューションです。

#### OS、DBMS等の標準ツールを利用



# クラウド特有の運用方法を理解しましょう

## クラウドの真価を引き出す運用とは

米国国立標準技術研究所(NIST)によるクラウドコンピューティングの定義は次の通りです。「共用の構成可能なコンピューティングリソース(ネットワーク、サーバー、ストレージ、アプリケーション、サービス)の集積に、どこからでも、簡便に、必要に応じて、ネットワーク経由でアクセスすることを可能とするモデルであり、最小限の利用手続きまたはサービスプロバイダとのやりとりで速やかに割当てられ提供されるものである。」つまり、クラウド時代の運用とは、いつでも必要なときに、簡便な手続きで必要なコンピューティングリソースを使えるというクラウドの真価をうまく引き出す運用といえるでしょう。

## ☑ クラウドにおける運用最適化キーワード



### 自動化

クラウド自身が備えるAPIを活用して、インフラ構築や運用を自動化することで、省力化、効率化を図ると同時に運用品質を向上させる。また、従来人間が行っていた煩雑な定型運用作業をロボットで対応するという取り組みも、APIを活用することで容易に実現可能になる。



### 標準化

運用設計のノウハウを標準化(テンプレート化)し、APIを活用して、システム全体を自動的に構築したり、複数のクラウドサービスを組み合わせることで運用プロセスを標準化する。



### 可視化

自動化や標準化により、多くの運用業務を少ない人員で対応することになるが、運用プロセスを可視化する(例:サービス依頼件数、管理サーバーの台数、障害件数、ジョブ数など運用業務を数値で表す)ことで、運用効率、改善効果を正しく把握することができ、継続的な運用改善の取り組みを支える仕組みを作る。



## Tips

### クラウド時代の新しい運用「クラウド費用管理」とは？

使いたいときに使いたい分だけ使えるクラウドですが、費用をどのように管理していくかは、コスト最適化の観点で重要なポイントです。各クラウドではコスト管理機能は標準で提供されていますが、使い勝手はまだ進化中というのが実情です。また、コスト分析のためにはコスト情報に別途タグ付けをする必要がありますので、構築作業や日々の運用業務にも関連します。早期に対処することをお勧めします。

#### クラウド費用可視化・分析機能

- ・部門、システム、用途別など企業固有の切り口による集計
- ・費用実績と着地予想の可視化、計画及び特定月との比較分析

#### コスト最適化支援機能

- ・コスト削減のための購入プラン(AWS RI、セービングプラン)適用効果の可視化と最適化支援
- ・表示情報のCSV出力



**BeeX**  
Service Console

## パブリッククラウド上で運用するということ

自社データセンター等で行われてきた従来型オンプレミス運用と、AWS上の運用との違いについて、特徴的なポイントをご紹介します。

### ✓ セキュリティ運用

AWSでは「セキュリティは最優先事項である」と考えられています。積極的に外部認証を取得し、数多くの法令に遵守するなど、セキュリティに対しては継続的な投資を続けています。

自社内でシステムを管理するオンプレミスとAWSを比較してみると、「AWSはこんなにセキュリティ対策をとっているのか!」と安心することができるはずです。

AWSは、ユーザーとAWSの2者でセキュリティを確保する

「責任共有モデル」をとっています。データセンターの物理的なセキュリティ、内部のネットワークセキュリティなど、インフラに関わる領域は全てAWSが責任を持ち、担保します。一方お客様は、それ以外のOSやアプリケーション、自社のネットワークに注力するという責任分担になっています。お客様は施設やハードウェアの保護に関して全てAWSに任せられることが大きなメリットとなるでしょう。

#### AWSの責任共有モデル

お客様	クラウド”内の” セキュリティに 責任がある	お客様のデータ			
		プラットフォーム、アプリケーション、AWS Identity and Access Management			
		オペレーティングシステム、ネットワーク、ファイアウォール構成			
		クライアント側のデータ 暗号化とデータ整合性の認証	サーバー側の暗号化	ネットワークトラフィックの保護	
AWS	クラウド”の” セキュリティに 責任がある	コンピューティング	ストレージ	データベース	ネットワーキング
		AWSグローバル インフラストラクチャ		リージョン	エッジ ロケーション
				アベイラビリティゾーン	

### ✓ 可用性確保の考え方

AWSには、システムデザインの基本に「Design for Failure」つまり全てのシステムは故障する前提の下で、システムを設計するという考え方があります。

AWSの中でも代表的なサービスであるAmazon EC2を利用していたとしても、残念ながら絶対に落ちないということはありません。であれば発想を逆転し、1つのサーバーが落ちたとしても、継続して稼働できるシステム構成にしておくのが現

実的な考え方であり、そのような構成をAWSでも推奨しています。

たとえば、AWSは従量課金制なので、コールドスタンバイであれば冗長化しても微々たるコストしか発生しません。まさにクラウドならではの発想と言えるでしょう。このようなAWSのメリットを最大限に活かして、可用性を高める設計を検討してはいかがでしょうか。

### ✓ 24時間365日稼働監視、モニタリングの必要性

AWS上に自社のシステムを移行すると、仮想化OSより下の構成要素、例えば物理的なサーバー、ネットワーク、ストレージはお客様から見えなくなり、AWS社のサポートに全て委ねることになります。

その結果、ハード障害の発生で夜中に起こされたり、休日深夜のベンダー保守対応につきあうことがなくなるメリット

はあるものの、見えなくなった基盤部分は各社で共有されることもあり、障害発生の検知とその影響範囲は顧客自身が判断する必要があります。また、クラウドのリソース使用状況は課金に直結することもあり、リソース不足の事前回避といった短期の視点だけでなく、中長期の視点でリソース利用計画を講じることで、さらなるコストダウンが可能です。

## 全ての運用作業が自動化できる時代が到来

豊富なAPIが提供されるクラウドでは、運用作業のほぼすべてを自動化することが可能です。一方で、クラウド自体が激しく変化し、進化し続ける今、自社に最適な運用自動化実装アプローチの追及は大変重要となります。

### ✓ クラウド上での運用に求められるもの

#### 提供の頻度

クラウドの大きな特徴の一つにサーバー調達からOS等の基本設定までのスピードがあります。物理的環境では数週間かかっていたものがクラウドでは数分になります。ただし、アプリケーションの導入・設定作業は依然必要となりますので、インフラ提供の所要時間におけるボトルネックは人に移ることになります。

#### 作業の頻度

サーバの作成・破棄、起動・停止の回数が増えます。作業者のヒューマンエラーによる障害はゼロにはなりません。

#### 管理の頻度

確保したシステムリソース、起動時間が直接日々のインフラコストに反映されます。固定費用から変動費用になったインフラコストを適正に保つため、リソース状況やシステム構成を更に正確に把握する必要があります。

### ✓ 運用設計のポイント

#### 変化を前提とする

クラウドは日々進化を続けており、新機能を取り入れることでメリットを享受できます。よって、業務フロー、自動化等の運用設計は変化を前提とし、変更に強いものにする必要があります。

#### APIを利用した自動化を組み込む

自動化は、運用の速度・頻度を上げつつ、制度を向上させるために有効な手段です。APIを活用することでクラウドにおける多くの操作を自動化することができます。世界中のクラウドユーザが公の場で情報共有を行っているため、API活用のノウハウも得やすい状況です。

### ✓ 実運用のポイント

日々の小さい改善の活動が変化に強い運用を作ります。生み出された時間を次の改善に繋げるサイクルを回し続けることが、最終的に大きな成果に繋がります。

#### ドキュメントの鮮度を保つ

頻繁な変更に対応するドキュメント管理を実現するためには、変更の影響範囲を特定し、変更内容を適切に管理する必要があります。適切なドキュメントにアクセスできる検索機能と、変更履歴の自動取得機能を提供する「ドキュメント管理サービス」は、ドキュメントの鮮度を保ちつつ、高い品質を維持するのに効果的です。

#### 小さな自動化を継続的に組み込む

マスターとなる運用手順の中から効果の高い部分に対して自動化を適用していきます。その際に小さくシンプルな自動化に留めることで、場合によっては使い捨てることもできる、変化に強い運用となります。



## 「クラウド×基幹システム×運用」今すぐ使えるTips集

クラウド化後の運用に関するTipsはインターネット上でも沢山見つかります。しかし、その全てが基幹システムの運用に適用できるわけではありません。ここでは、現場を知るコンサルタントが基幹システム運用担当者に向けて厳選したクラウド運用Tipsをご紹介します。

### 手順は画面キャプチャよりコマンドで

クラウドサービスの管理画面は日々修正が入るので、手順書にキャプチャ画面を貼ってもすぐに陳腐化してしまいます。それと比較して、APIをコマンドから実行するCLIは変更が入ることが少ないので手順変更の回数を減らすことができます。

また、CLIのコマンドとオプションは、それ自体が何をするのかを表しているのので、日本語より意図が正確に伝わる場合があります。



インスタンスの作成



```
aws ec2 create-image \  
--instance-id 123 \  
--name "My Server"  
--reboot
```

### プログラムが苦手な方は、EXCEL+CLI活用で

プログラムが苦手な人でもAPIを活用する方法はあります。

#### CLI(Command Line Interface)

APIをコマンドから実行できるCLIは、単体でも高機能で多くのことが実現できます。

使い慣れたEXCELでオプションを組み合わせて、提携コマンドを作成可能

インスタンスID	名前	再起動
123	My Server	有

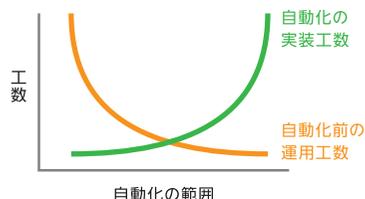
↓ CLI呼び出し

```
aws ec2 create-image --instance-id 123 ...
```

### 自動化の範囲

運用業務の自動化というと、特定の運用業務全体の自動化をイメージするかもしれませんが、完全な自動化は実装に大きな工数がかかり、変更に弱くなります。

適切な範囲の自動化を行うことで費用対効果の高い運用改善となります。



### Markdown記法

Markdown記法とは、基本的な文法を覚えるだけで、段落やテーブルを含む構造化された文書を手早く作成できる記法です。ブラウザで提供される画面は検索性、一覧性の高い運用ドキュメントとなります。

多くのサービスで提供されている変更履歴機能により、クラウド運用につきものの頻繁な変更にも耐えられるドキュメント運用を構築しやすくなります。

```
# 大項目  
## 中項目  
### 小項目
```

```
|カラム1|カラム2|カラム3|  
| --- | --- | --- |  
| 値1 | 値2 | 値3 |
```

大項目

中項目

小項目

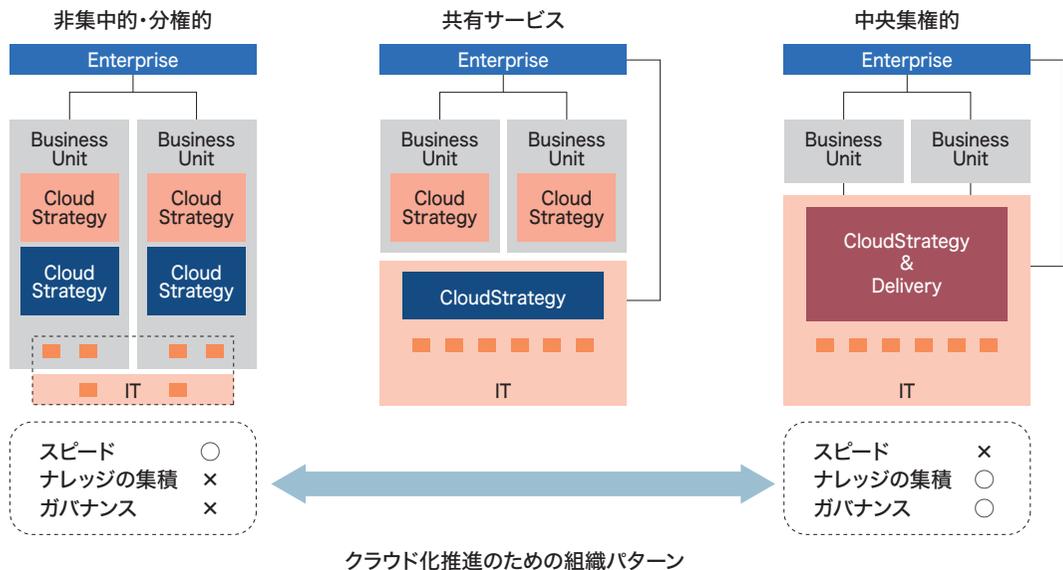
カラム1	カラム2	カラム3
値1	値2	値3

# クラウド化を推進するための 組織作りを考えましょう

## クラウドジャーニーを支える組織

オンプレミスをメインとした既存システムをクラウドに置き換え、ビジネスモデルや業務プロセスを「クラウドありき」の状態へ移行させるには、綿密な計画と複数のアプローチが必要です。これらを「旅路(ジャーニー)」に例え、クラウドジャーニーという言葉が生まれました。本章ではクラウドジャーニーを実現するための組織はどうあるべきなのかを掘りさげて考えてみましょう。

クラウド化推進のための組織構成例として、大きく以下の3つに分けることができます。



「中央集権的」は、戦略策定から環境構築までIT部門が全て行うというものです。一方、基本は現場任せで最低限のタスクをIT部門で行うのが、「非集中的・分権的」です。

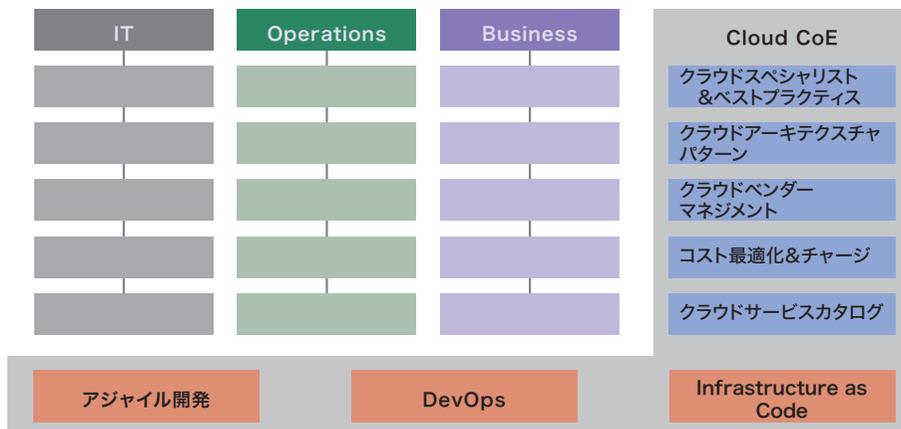
「中央集権的」の場合、ガバナンスを効かしやすいのが特徴で、特に基幹系システムなどのバックオフィスの業務システムに関しては、現場主導でバラバラと環境を作られると効率が良くないことが多いです。一方で外向きのWebシステムなどを2,3ヶ月で作成して、アジャイル的に随時機能を拡張していく場合、「中央集権的」に進めるとIT部門や既存ベンダーがボトルネックになることが少なくないため、こちらは「非集中的・分権的」で分離するといった使い分けのアプローチが重要です。

真ん中の「共有サービス」は戦略の策定は現場に任せるが、作る場所はIT部門に頼むというケースです。これはIT部門にとってはよくわからない中で、モノを作らないといけない状態になりえるという意味で、IT部門の方にとっては一番きついやり方ともいえ、お勧めはしません。

以上のことから、左右のどちらかに分けることになりませんが、そうはいつても全体としてガバナンスが効かないという側面は残ります。その場合にはCCoE(Cloud Center of Excellence)と呼ばれるクラウドの専門チームの組織を作って、会社全体を横串で見えるようにすることをお勧めします。

## Cloud Center of Excellence (CCoE)

規範的なガイダンスとサポートを提供



出典:クラウドCoE(Center of Excellence - Amazon AWS)

CCoEでは規範的なガイダンスと新機能を実装するためのサポートを集中的に提供します。これによって、業務システム側では、中央集権的なCCoEの支援をもとにガイダンスを作成しつつ、それに基づく共通基盤を作成することになります。

現場で新しく何かやりたい場合には、業務システム側のガバナンスを効かせ過ぎると上手くいかないのが、基本現場の方に任せながら、最低限のガバナンスを効かせるためのルール提供や、現場での構築を一緒に行います。なお、この仕組みを本格的に回すためには、大企業でないとい難しいという現実がありますが、規模が小さくても情報システム内で「是非、クラウドをやりたい」という担当者を、チームではなくてもひとりキーマンにして、ルール作りから初めて頂くと、柔軟性とガバナンスが両立した取り組みに着手できることがあります。また、いきなり全部をやる必要はありません。まずは中央集権的な取り組みでCCoEを確立した後に、現場主導の先進的な取り組みに着手することをお勧めします。



### クラウドジャーニーとデータジャーニー

クラウドの利点を最大限に生かしたアプリケーションを「クラウドネイティブアプリケーション」と呼びます。従来とは異なるアプローチでアプリ実装する必要がありますが、その際に重要となるポイントをご紹介します。

#### クラウドジャーニーとは

企業における「クラウド化(クラウドネイティブ化)への継続的な取り組み」を表した言葉です。オンプレミスを中心にしていた既存システムをクラウドに置き換え、ビジネスモデルや業務プロセスを「クラウドありき」の状態へ移行させるには、綿密な計画と複数のアプローチが必要になり、CoEの取り組みもその一つです。

#### データジャーニーとは

個人の主観にとらわれない、データに基づく意思決定の実現の取り組みを表した言葉です。経験と勘に頼って判断を下すのではなく、従来の業務システムからのレポートに加えて、現場の生のデータを機械学習やAI等の技術を活用することで効率よく分析を行い、より正確に最新状況を捉えた判断を下すことを目指します。

<u>Non-Tech</u>	<u>Tech</u>
BUSINESS	PLATFORM
PEOPLE	SECURITY
GOVERNANCE	OPERATIONS



# クラウド対応版RFI/RFPを 作成しましょう

## 最適な提案を受けるために

RFI(Request For Information、情報提供依頼書)、RFP(Request For Proposal、提案依頼書)を作成しましょう。「うちのシステムをクラウドに移行した場合の見積もり持ってきて。」だけでは、いくらクラウドインテグレータにとって腕の見せ所といっても限界があります。自社に最適なクラウドを各社から提案してもらうために、**自社の考えや正しい客観情報**をぜひ伝えましょう。

### RFPの章立て例

- 趣旨 <sup>▲</sup>
- 現行システム構成(As-Is)
- RFPに対するQ&A
- 背景・目的
- 新システムイメージ(To-Be) <sup>▲</sup>
- 提案書提出手順
- 依頼書の趣旨
- 提案依頼書内容
- 問い合わせ先
- 本資料の構成
- 提案内容(提案書、見積書) <sup>▲</sup>
- システム全体のビジョン <sup>▲</sup>
- 提案スケジュール

### 添付資料

- プロジェクトスケジュール
- 役割分担

- 成果物
- 自社紹介

- 現行システム構成一覧
- ネットワーク構成図 <sup>▲</sup>

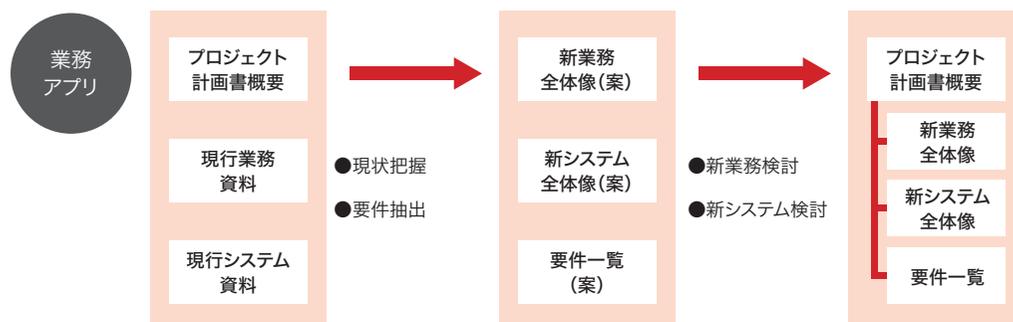
## ! 注意すべきポイント

- <sup>▲</sup> なぜ、クラウドを前提にしたのかについて、1章にある代表的な目的を参考に、明確に記述するとともに、狙いや効果などをかくことが内容の骨子がぶれないようにします。一言でクラウドといっても、様々な形態があります。特に指定せず幅広いクラウドの提案を求めることも可能ですが、パブリッククラウドの提案が欲しい場合は、明記しておきます。
- <sup>▲</sup> システム全体のビジョンがあると、RFP上で明確に書かれていない要件に対する提案事項に対して、自社のスタンスをある程度伝えることができます。
- <sup>▲</sup> サーバー一覧だけではなく、簡単に図示することで、より正確に要件を伝えることができます。(特にネットワーク、ストレージはSPEC一覧では読み取れないことがあります。)
- <sup>▲</sup> 同じシステム構成でも異なる価格体系を採用することで、金額は異なります。システム構成だけでなく、採用した価格体系とそれを選定した理由なども明記してもらいましょう。また、コスト算出期間については、導入後5年間等ではなく、年額、月額、初期コスト、ランニングコスト等で期間を区切った費用内訳を予め明記しておくといでしょう。
- <sup>▲</sup> ネットワーク構成については、ネットワーク接続構成図以外に、機材、利用機能等できる限り具体的に書くことをお勧めします。クラウド上では、全てソフトウェアで実装することになりますので、特殊なネットワーク構成や設定を行っている場合は特に注意が必要です。

## アセスメント (PoC) フェーズ実施の検討

クラウド移行においては、準備・実行フェーズに入る前に、アセスメント (PoC) フェーズを設定するのが一般的です。その背景として、単純な既存システムのクラウド移行要件以外に、様々な新しい技術を適用したり、プロジェクトの計画リスク低減を目的とした事前検証が必要になるためです。アセスメントフェーズはインプット (現行環境情報)、アウトプット (具体的なシステム全体像や移行方針) を明確にすることにより、その成果を確実に担保しましょう。

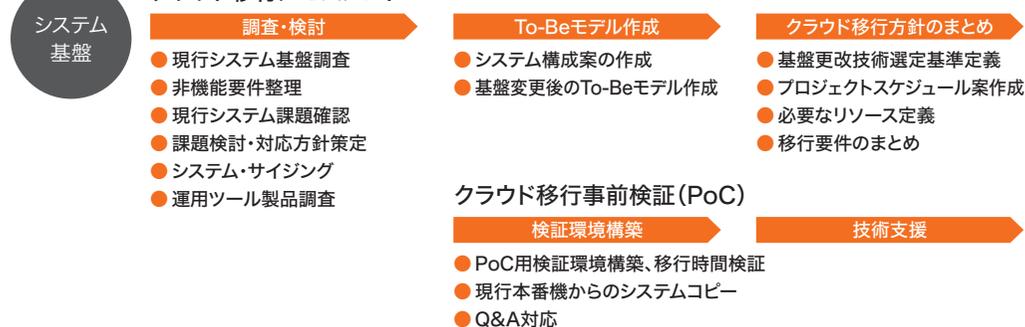
### アセスメントフェーズの実施イメージ



### 新しいアーキテクチャを検討する場合に必要なタスク



### クラウド移行アセスメント



クラウドは常に進化し続けるのが特徴であり、数ヶ月後には魅力的な新しいサービスがどんどん提供されます。一方で、オンプレミス環境で稼働していた基幹システムをクラウド上で稼働させるためのベストプラクティスは、多くの実績とともに既に確立されており、安心して取り組めることをお約束します。ご不安な点がございましたら、弊社まで、どうぞお気軽にご相談下さい。



お問合せ

## 株式会社BeeX

〒104-0061 東京都中央区銀座7-14-13 日土地銀座ビル10F

TEL : 03-6260-6240 E-mail : beex-marketing@beex-inc.com

※SAPは、ドイツおよびその他の国々におけるSAP SEの登録商標です。

※アマゾン、ウェブサービスおよびAWSは、米国その他の諸国における、Amazon,inc.またはその関連会社の商標です。

※Google Cloud は Google LLC の商標です。

※Microsoft Azure、その他のMicrosoftの製品は、米国Microsoft Corporationの米国およびその他の国における商標または登録商標です。

※その他記載されている、会社名、製品名、ロゴなどは、各社の登録商標または商標です。

※本リーフレットに掲載されているロゴ、文章、イラストを無断で転載、複製、再利用を禁止します。

2021年11月現在