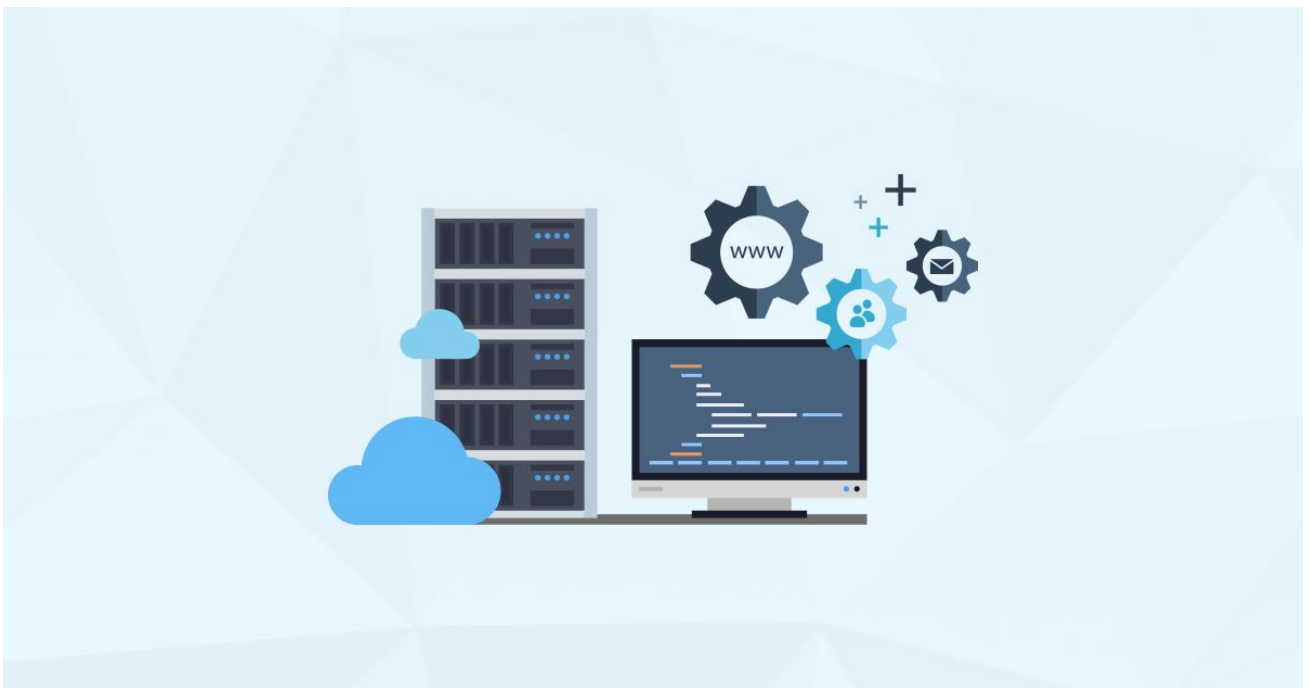


AWS への移行を徹底解説！ オンプレミスからクラウド環境への移行手順



内容

移行先としての AWS.....	3
グローバルインフラストラクチャとセキュリティ.....	3
継続的な値下げ.....	3
AWS サービスの拡張と機能改善.....	3
AWS への移行のための事前調査.....	4
AWS への移行のステップ.....	4
AWS への移行の計画と実行.....	7
AWS への移行の方針を決める.....	7
AWS へのデータ移行を考える.....	8
社内関係者・外部連携先との調整は綿密に.....	9
AWS への移行は作業セットの抽出でまとめる.....	10
AWS への移行作業の管理を楽にする、マイルストーン設定のコツ.....	10
移行対照表を活用してタスクの洗い出しとチェックリストをつくる.....	10
想定外を防ぐ、リハーサルを使った AWS への移行作業の準備.....	11
AWS への移行本番の切り替え.....	13
担当者の連絡体制とフローをきめておく.....	13
移行時体制の構築の作業ステップをまとめる.....	13
AWS への移行作業 直前・直後の 3 ポイント.....	14
AWS への移行後のポイント.....	15
旧環境の廃棄.....	15
AWS のクラウド環境の利活用.....	15
構成の適正化.....	15
運用性の改善.....	15
移行で活用したい AWS サービス.....	16
仮想マシンの移行.....	16
大容量データの転送.....	16
データベースデータの移行.....	17
おわりに.....	17

クラウドファーストへのシフトが加速していく中、オンプレミスからの移行を検討している企業も増加傾向にあります。では実際にオンプレミスからアマゾン ウェブ サービス (AWS)のようなクラウド環境に移行をする場合には、どうやって移行をすすめればいいのでしょうか？

本記事では、AWSのクラウド環境への移行に必要な考え方、事前準備や移行時の注意点などをお伝えします。

移行先としての AWS

ここではオンプレミス環境から AWS へシステムを移行するとき、特にオンプレミス環境と比較した場合に AWS が優位になる点をいくつか挙げてみることにします。

グローバルインフラストラクチャとセキュリティ

AWS は全世界に 20 個の地理的リージョンにある 61 個のアベイラビリティゾーンを有しています。(2019 年 4 月時点) AWS のリージョンは低レイテンシー、高いスループット、そして高度の冗長ネットワークで接続されている複数の物理的に独立・隔離されたアベイラビリティゾーンが用意されていますので、基盤全体で見たときには、高い可用性と耐障害性を実現しています。また、セキュリティ面でも、ISO や SOC を初めとした数多くの第三者監査によるセキュリティやコンプライアンスについての検証が実施されています。

AWS を利用するという事は、高い可用性と耐障害性、そしてセキュリティ対策が施された環境を莫大な初期投資をしなくても使えるということでもあります。

継続的な値下げ

AWS はサービスを開始してから過去 10 年間で 60 回以上の値下げを実施した実績があります。AWS サービスの利用拡大に伴い、AWS 自体に規模のメリットによるコスト優位性が生じることが一つの理由として挙げられますが、AWS はこの利益を値下げという形で顧客に還元します。サービスを利用する側としては利用を継続するだけで適正なランニングコストが維持されるともいえます。加えて使っているリソースが値下げされることもあるでしょうから、長期的に見ればランニングコストが低減していく恩恵を受けられるかもしれません。

AWS サービスの拡張と機能改善

AWS はこれまで 100 を超えるサービスを提供しています。仮想サーバーなどのコンピューティング領域にとどまらず、今では IoT やデータ解析などの領域も含めた IT インフラの幅広い分野でサービスを提供しています。例えば 2017 年に行った機能改善は 1,430 回にも及びます。また、そのうち 90% 以上は利用者からのフィードバックに基づく実装ということです。サービスの豊富さと拡充されるスピードの速さは他社クラウドと比較しても全く引けを取りません。

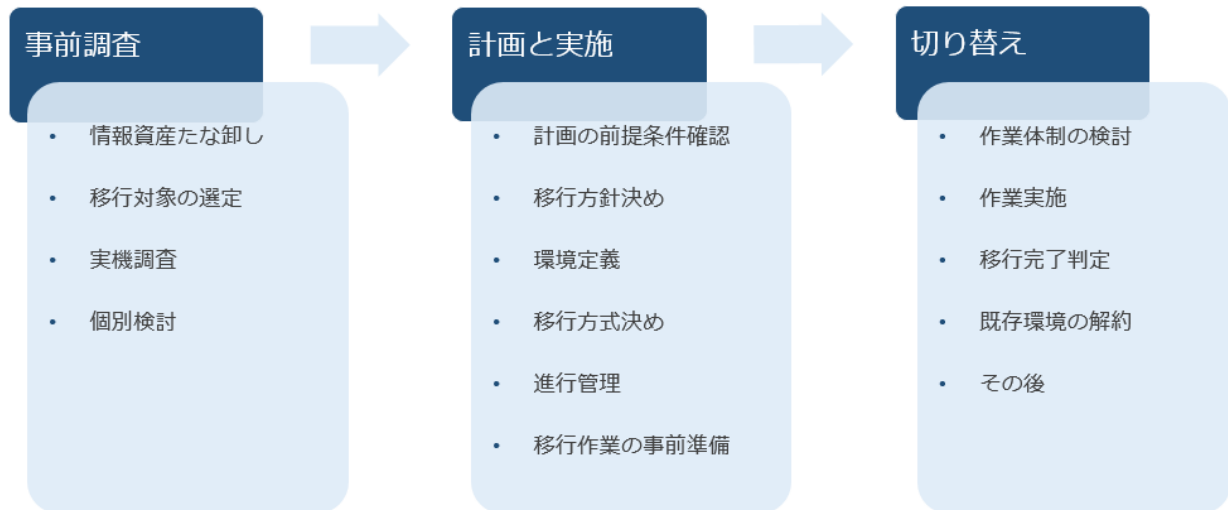
ここまで、オンプレミス環境からの移行先として AWS の優位な点をいくつか挙げさせていただきました。今回は 3 点ほど挙げる形を取りましたが、以下もわせて参考にさせていただきたい情報です。

[AWS のクラウドが選ばれる 10 の理由](#)

AWS への移行のための事前調査

AWS への移行のステップ

事前調査、計画と実施、当日の切り替えの3つの作業ステップを順次進めていながら、移行時の作業を明確にしていく必要があります。AWS へ移行するために必要な事前調査について5つの手順などを紹介します。



1.一番初めに決めること ゴール設定

まず移行のプロジェクトを立ち上げるときに重要となるのが「ゴール設定」です。

- なぜ移行を行うのか
- 何をどこまで移行するのか
- いつまでに移行を行うのか

それぞれについて考え、移行の完了条件を設定することが大切です。

2.情報資産の棚卸し

事前調査ではステークホルダー、情報資産、月額費用の切り口でリストアップすると移行対象を把握しやすいでしょう。

- ステークホルダー
システムに関わる人との密な連携が移行には必須です。社内の業務やシステムの担当者だけでなく、外部委託先やAPIなどで連携している外部サービスがある場合にはその担当者の連絡先もはじめに整理しておきましょう。移行に関係する各種業務調整や、移行作業でトラブルや障害が発生した場合の対処がスムーズです。

- 情報資産

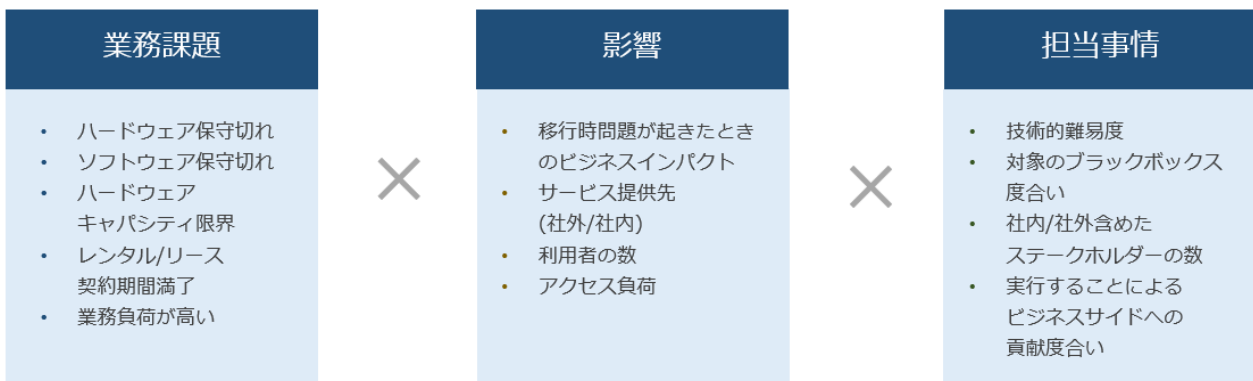
実際に移行を行うシステムのドキュメント、ハードウェア資産、ソフトウェア資産などを把握します。見落としがちなのが、サーバーやシステムにアクセスするための各種ログイン情報や秘密鍵などです。移行作業に直接影響するため、アクセス手段の確保は重要です。

- 月額費用

外部でデータセンターやレンタルサーバーを借りている場合は、解約の時期を決める必要があります。解約条件を確認しましょう、一般的に移行が完了するまでは新旧両システムを並行で稼働させる事が多いため、移行期間中のコストも予算確保しておく必要があります。

3. 移行対象の選定

AWS への移行対象を選定します。例えば対象がシステムの一部なのか、全部なのかといった範囲や、複数ある場合にはどれから移行するかという順序も検討します。移行の優先順位付けは「業務課題」、「影響度合い」、「担当事情」等の切り口で比較すると、判断しやすくなるでしょう。比較的ビジネスインパクトが小さく、利用者の少ないシステムから PoC(Proof of Concept (概念実証)) を兼ねて移行を進めてみるのは多くの場合良い選択です。小規模なシステムからの移行を早い段階で一通り行ってみることで、移行時の注意点が把握しやすい、早い段階から AWS 上でシステムを運用する経験を蓄積できる、等のメリットが考えられます。



4. 実機調査

設計書、設定シート、構成図、手順書などの各種ドキュメントが揃っていたとしても、実機調査をせずに移行計画を立てることはトラブルの元です。管理ドキュメントの内容があっているのかの確認を含め、必ず実機調査を行いましょう。過去、急ぎで CPU やメモリなどのリソースを追加したが、管理ドキュメントは更新されていなかった、というケースはよく聞く話です。実機の OS やミドルウェアのバージョンが管理ドキュメントより新しい、ということも考えられます。現環境の概要を把握するためにドキュメントを使いつつ、実機の状態を構成上の「正」として押さえるのが調査精度の面ではより安全でしょう。

逆に不要な設定が残ったままだった、ということもあるでしょう。実機調査と合わせてここで管理ドキュメントの更新と各種設定の棚卸しも行っておくと、その後の検討がやりやすくなるでしょう。

5.移行に検討が必要なもの

- バージョンが古いOS、ミドルウェア
 移行対象とするシステム、特にアプリケーションの動作要件は早めに把握しておきたいものの一つです。AWS側で移行前と同じ環境を用意できない場合が考えられるからです。例えばOSならRedHat4以前、Windows2000以前、あるいはミドルウェアのバージョンがあまりにも古いケースなどが挙げられます。この場合は、OSやミドルウェアは新しいバージョンの環境を用意することになるため、アプリケーションの動作テストが重要なタスクになるでしょう。場合によってはアプリケーション側の改修が必要なケースも考えられます。
- アプリケーションにIPアドレスやIDがハードコーディングされている
 アプリケーションそのものにIPアドレスやハードウェアなどのIDがハードコーディングされているような場合、AWSに移行を行うタイミングでやはりアプリケーション改修と動作テストが必要になるでしょう。
- クラスタや負荷分散設定
 オンプレミス環境でクラスタ構成を組んでいる場合、構成によっては実現方式を見直すことになるかもしれません。同様にロードバランサーを利用して、アプリケーション要件で特殊な負荷分散設定を行っているケースなども注意が必要です。

AWS への移行の計画と実行

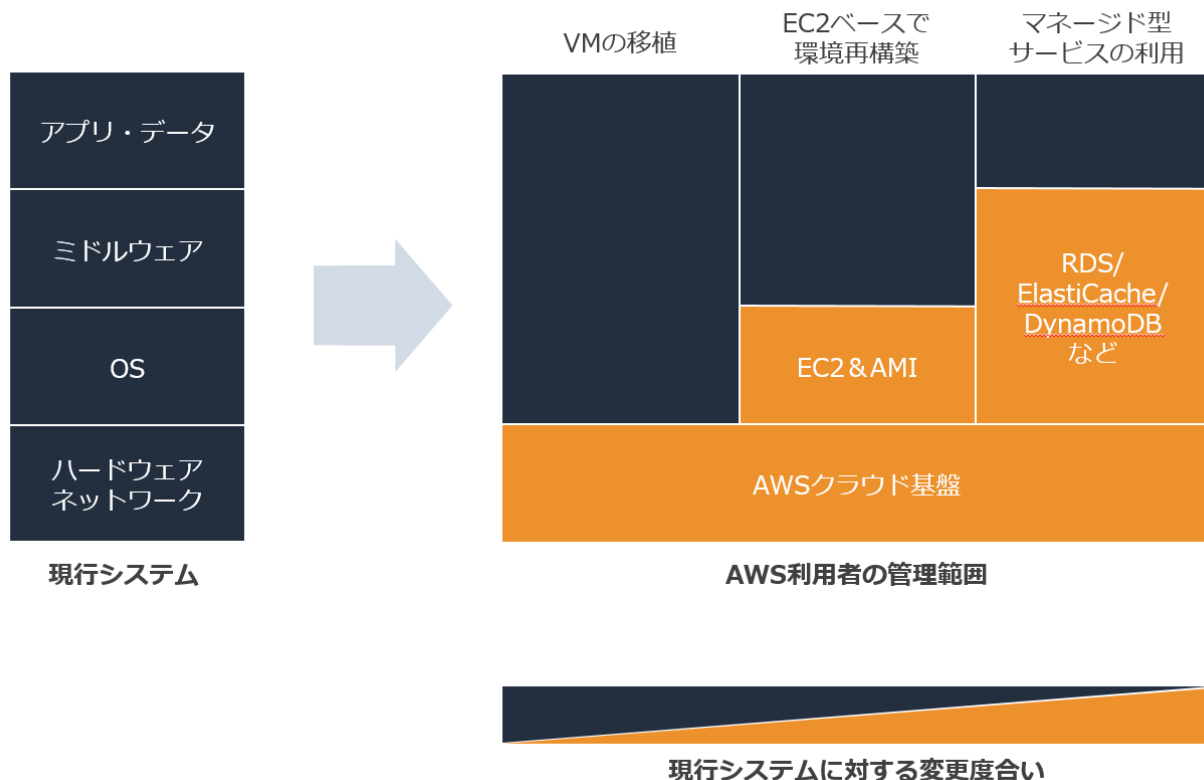
AWS への移行の方針を決める

事前調査を行い、課題の洗い出しを行ったあとは、各システムで AWS への移行をどう行うかの方針・計画を決めていきます。AWS へ移行する場合には主に VPC(Virtual Private Network)環境のネットワーク構成と、OS より上の層について、移行対象のシステム・データをどのように配置するかを検討していきます。

OS やミドルウェアのバージョンの問題がない限りは、EC2 をベースにミドルウェアを含めたアプリケーション・データを移行する方法が分かりやすいでしょう。

RDS や ElastiCache に代表されるマネージド型のサービスを使うと、データのバックアップ、エンジンのバージョンアップ、冗長化など、EC2 ベースでシステムを構築した場合には個別に検討しないといけない機能が予め用意されている、という点から移行後の運用負荷軽減が期待できます。できればマネージド型サービスを活用する方向で検討してみると良いでしょう。

サーバー移行方針の例



以下の項目は設計を進めていく中で注意しておくの良いケースを TIPS としてまとめています。特に AWS への事前申請が必要なケースは設計のタイミングで必要なリソースを洗い出し、上限の緩和申請をタスクに落とせると、その後の進行がスムーズです。また、移行作業で一時的に必要な環境やリソースの手配は見落としがちのため、注意が必要です。

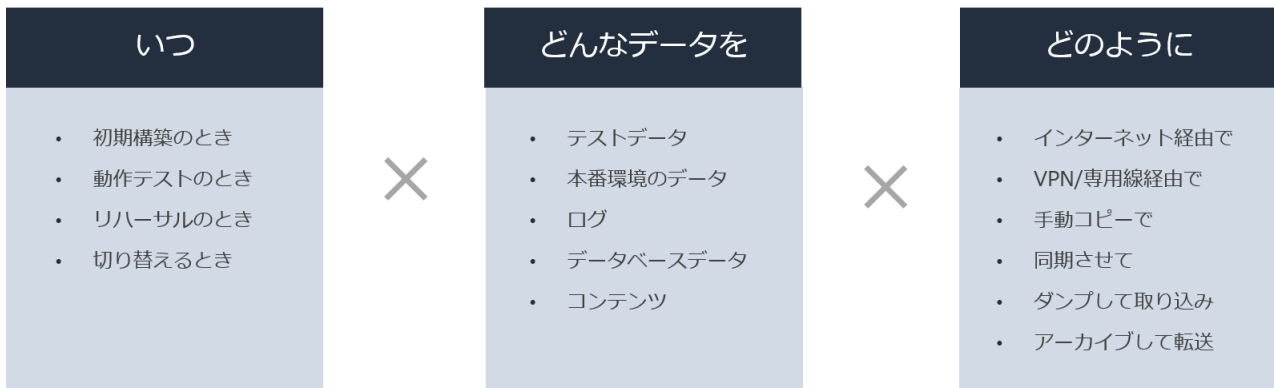
項目	注意点	理由
プライベート IP	プライベート IP アドレス帯は既存の環境と別にしておく	専用線や VPN の通信が必要になることを考慮して予め別にしておくと楽
固定 IP(EIP)	外部との連携に固定 IP アドレスを必要とするかを事前に洗い出す	外部連携の関係で固定 IP が必要な場合があるため(アクセス制御)
メール送信	メールの送信が必要な場合は方式設計が必要	特に SES を使わない場合は、AWS に対して上限緩和申請が必要
インスタンスタイプ	可能なら EC2/RDS/ElastiCache のインスタンスタイプはサイズを揃える	リザーブド購入を検討するなら、インスタンスタイプをそろえると管理しやすい
上限の緩和	AWS の各サービスには上限設定がされており、上限を緩和する申請が必要な場合がある	設計時にできるだけ必要なリソースの数をピックアップし、上限緩和申請もタスクに含める
テクニカルサポート	AWS から十分なサポートを受けるためにはサポート契約が必要	プロダクション用途なら、ビジネスサポート以上の契約を結ぶことを推奨

AWS へのデータ移行を考える

データの移行は移行計画をたてるうえで大切なポイントです。データベースのデータや Web のコンテンツ、各種ログなどオンプレミスの環境に分散配置されているこれらデータを、いつ、どのように移行するか、を検討します。

移行しないといけないデータを最新の状態で全て AWS に移行するというのがデータ移行のゴールになりますが、システムを切り替えるとき以外にも本番と同等のデータを移行しなければいけないタイミングがおそらく出てくるでしょう。例えば、初期構築完了後の動作テスト、あるいはシステム切り替え作業のリハーサルでは、動作確認や作業シナリオの確認のため、本番運用中の実データとほぼ近いものが必要になるでしょう。

データの移行を考える際には、いつ、どんなデータを、どのように移行するかという切り口で整理してみると、移行計画や移行方式を検討しやすくなるでしょう。



移行対象のデータ量も考慮事項です。例えば大量のデータを転送する必要がある場合には

- AWS Direct Connect などの専用線を敷設
- AWS Snowball のようなデータ転送サービスを利用

など、データの配送手段も検討事項になってきます。特に回線の手配は時間がかかるため、プロジェクトの進行管理上、早めの対応が必要です。データ量によっては転送時間も少なくはないため、ここにかかる時間も考慮が必要です。

社内関係者・外部連携先との調整は綿密に

AWS へ移行する対象のシステムやサービスに外部との連携がある場合、自社から外部のシステムを参照している場合と、自社に外部からデータを参照している場合の 2 つのパターンがありますが、どちらも切り替えるタイミングの調整が必要です。

例えば外部からのデータ連携処理があるため特定の時間帯は止められない、メンテナンスするときは最低 1 ヶ月前の告知が必要など、AWS への移行に際して時間、作業時間や日付の制限がある場合にはあらかじめ把握し、計画したうえで移行を進める必要があります。

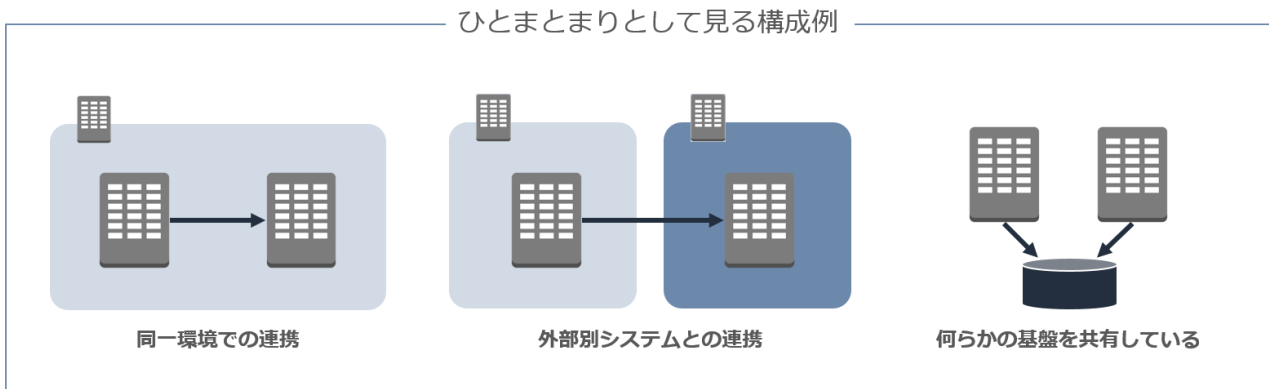
また自社環境から外部連携先にアクセスする場合に、先方がアクセス制限をかけているケースがあります。連携テストを行いたいときに、開発環境と新しい本番環境がつながるように許可をもらう必要があり、先方都合でリードタイムがかかることもしばしばです。タスクとして洗い出しを行い、遅延なく移行が進められるよう調整を行いましょう。連携テストの場合は、先方で一時的に試せる検証環境を用意している場合もあるため確認しておくことをおすすめします。

その他、人間系の業務調整もこのタイミングで合わせて調整します。社内の別部署でもコンテンツ更新を行っている場合は更新計画の確認が必要です。あるいは DNS レコードの管理を外部に委託している場合には設定変更の依頼方法やリードタイムを踏まえたうえで計画に盛り込む必要があるかもしれません。

AWS への移行は作業セットの抽出でまとめる

全てのシステムを AWS に移行する場合でも、一斉に移行を行うことは難しいため、いくつかのセットにわけて移行します。AWS への移行の方法を考える際に、「一連の移行作業」を「ひとまとまりで行う必要がある」単位でセットにすると計画が立てやすくなります。

一般的にはサービスもしくは業務システムが 1 セット単位になりますが、運用や機能拡張のタイミングでサービス同士がつながっていることも多いため、システム連携や基盤の結合度合いなどの視点で分解していく作業も必要です。



AWS への移行作業の管理を楽にする、マイルストーン設定のコツ

複数のシステムやサービスを AWS に移行する際には、移行するひとまとまりの作業セットにわけ、この作業セットごとに同じ基準でマイルストーンを決めると全体としての見通しが立てやすくなります。

移行はイレギュラーな事態が起きやすいため、「予定通り進んでいるか」よりも「作業がどこまで終わっているのか」を把握する必要があります。同じ粒度でマイルストーンを引いておくと、作業がどこまで完了しているかが可視化しやすいため、トラブルが起きた際に作業の順番の入れ替えが検討しやすくなるためおすすめです。

もう一つのポイントは移行前と移行後の判定をマイルストーンとして定義することです。移行前の判定では、移行対象のシステムが AWS 上で動作させたときに致命的な問題がないことを確認します。移行後の判定は移行したデータの内容も含め、システムの切り替えが完了したとみなしてよいかどうかを判定します。移行前・移行後の判定をマイルストーンに組み込むことで、完了条件がより明確になるでしょう。

移行対照表を活用してタスクの洗い出しとチェックリストをつくる

現状と移行完了時に何がどう変わるのかを把握するために、設計や移行作業を行う前に移行対照表を作成し、まとめておくと便利です。

例えば IP アドレスや OS やミドルウェアなどの移行対照表を統合や破棄の記載も含めて用意します。Web 系のシステムであれば DNS の切り替えを行うことが多いので、いまのレコードと新しい環境のレコードを整理しておくと、移行対照表自体が当日の作業内容となります。

移行対照表をつくることで、必要な作業の洗い出しとチェックリストの役割も果たしてくれるため、管理が楽になります。

IP アドレスの対照表の例

現行		移行後	
Host A	xxx.xxx.xxx.xxx	Instance A	xxx.xxx.xxx.xxx
Host B	xxx.xxx.xxx.xxx	Instance B	xxx.xxx.xxx.xxx
Host C	xxx.xxx.xxx.xxx	統合・破棄	

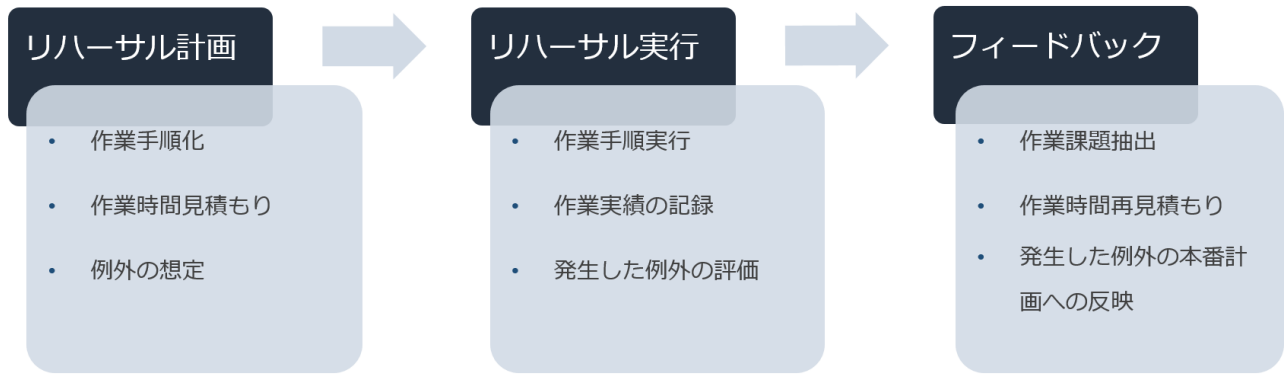
OS/ミドルウェアの対照表の例

現行		移行後	
Host A	Apache 2.2 PHP 5.3 CentOS 5	Instance A	Apache 2.4 PHP 7 Amazon Linux 18.03
Host B	IIS 6.0 Windows Server 2003	Instance B	IIS 8.5 Windows Server 2012R2
Host C	Apache 2.2 PHP 5.3 CentOS 5	統合・破棄	

想定外を防ぐ、リハーサルを使った AWS への移行作業の準備

移行手順が一旦確立したら、移行リハーサルを検討します。移行リハーサルの目的は、段取り上のヌケモレを発見する、データ移行などのオペレーションについて、見積もりの範囲内で収まるのか、など、計画しただけでは分からない、想定外の事象を洗い出すことです。

移行リハーサルを実施時に発生した問題に対策を打つことで、移行作業の本番当日の作業精度を向上させ、安心して当日を迎えることができるでしょう。



AWS への移行本番の切り替え

担当者の連絡体制とフローをきめておく

システム切り替えの当日は、連絡体制の構築が非常に重要です。システム切り替え作業に関わる担当者の連絡先や連絡する手段などを事前にまとめておきましょう。連絡を取りにくい時間帯があるようなら、それも制約として計画に含めておくことより確実です。また、トラブルがない場合にも作業の進捗状況は原則として同報通知を行うのが良いでしょう。特に作業する場所が物理的に離れている場合にあるケースですが、誰がどこまで作業が進めているのかが共有できてないと、後続タスクの作業者の作業進捗に影響が出てくる可能性もあるためです。

移行時体制の構築の作業ステップをまとめる

作業ステップをまとめておきます。ポイントは連絡を取り合うタイミングと判断ポイントを先にきめておくことです。とくに切り戻しの判断タイミングと、基準の設定が重要です。

万が一移行作業に問題が起きたとき、切り戻しの判断が遅れてしまうと、結果的にサービスの停止時間が伸びて業務影響が大きくなってしまうこともあるため、注意が必要です。

作業ステップ例

項番	作業内容	担当	想定時間 (単位：分)		連絡	備考
0	システム連携先への停止通知	〇〇	-	-	〇〇→関係各社	事前作業
1	システムの閉塞処理	△△	30	0:30	△△→□□	
2	コンテンツ同期	□□	60	1:30		
3	データベースデータダンプ	□□	60	2:30		
4	データベースデータ転送	□□	180	5:30		想定より時間がかかる場合、別途連絡・対応協議
5	データベースデータインポート	□□	60	6:30		
6	アプリケーション動作確認	〇〇/△△	120	8:30	△△→〇〇	
7	移行判定	〇〇	30	9:00		NG時は以降の作業をスキップ、予備日に対応
8	DNS切り替え	□□	30	9:30		
9	アプリケーション動作正常性確認	〇〇/△△	120	11:30	△△→〇〇	
10	最終判定	〇〇	30	12:00		NG時はメンテナンスモードに切り替え、問題解決を図る
11	システム連携先への切り替え完了通知	〇〇	-	-	〇〇→関係各社	

AWS への移行作業 直前・直後の 3 ポイント

AWS への移行作業の計画を立てる上での TIPS をいくつか挙げてみます。

イレギュラー判断について

予想される不測の事態をできるだけ事前に洗い出します。イレギュラーな事象が起きた時の対応シナリオを用意しておくことで、切り戻しが必要な場合の判断ポイントの定義がしやすくなるでしょう。また、不測の事態を予想して、シナリオにまで落としておくことが、システム切り替え作業を安心して確実に行うことにもつながります。

連絡体制について

先述しましたが、システム切り替え作業は、移行作業は複数チームで行う場合が少なくありません。円滑な移行作業のためには、連絡体制と定期的な連絡ポイントを適切に設定することが重要です。業務上利用してかまわない場合は Slack や Chatwork といったコミュニケーションツールの利用も検討すると良いでしょう。

要員計画について

移行作業は夜間や休日などに行う場合も考えられます。移行完了後、利用者からの問い合わせや、イレギュラーな事象に対する個別対応が発生することも考えられるため、移行作業が完了した後のバックアップ体制も踏まえて要員計画を立てておくプロジェクト関係者の負荷軽減につながるでしょう。

また、要員の役割はできるだけ分業します。特に作業状況の連絡者と移行作業の作業者は、特定の担当者が同時にこなしていくのは中々難しいため、原則として分けることが望ましいでしょう。

AWS への移行後のポイント

AWS への移行作業が完了した後のタスクなどいくつか挙げてみます。

旧環境の廃棄

無事に移行が完了した後に行うタスクとして、忘れがちなのが現環境の契約解除や撤去のタスクです。無駄なコストを発生させないためにも移行計画のタスクとして必ず含めるようにします。

一点注意として、現環境の契約を解除してしまうと、もう元には戻れないため、不測の事態にそなえ、しばらくは新旧のシステムの並行稼働期間を設定するのが良いでしょう。

AWS のクラウド環境の利活用

AWS のクラウド環境への移行が一段落したら、移行した環境をよりクラウドにフィットさせるような改善活動を行いたいところです。ここでは「構成の適正化」「運用性の改善」「管理負荷軽減」の観点から改善ポイントとなる項目を挙げています。

構成の適正化

- シングル構成部分(Single Point Of Failure)を見つけて排除
- インスタンスサイズの適正化
- スケールアウト構成

運用性の改善

- CLI や API の活用
- デプロイの見直し
- ログの集約管理

管理負荷軽減

- Amazon EC2 の利用を減らす
- サーバレスアーキテクチャの採用
- オペレーションの自動化対応

移行で活用したい AWS サービス

仮想マシンの移行

VM Import/Export

VM Import/Export を使うとオンプレミスの仮想マシンイメージを EC2 に移行することができます。移行に際して大きな変更を加える必要が無い点は魅力的ですが、利用できる仮想化ソフトウェアや OS は限定されていることに注意が必要です。移行元のオンプレミスで稼働しているシステムと要件が合致しているか確認する必要があります。

AWS Server Migration Service

AWS Server Migration Service を利用すると VMware vSphere または Microsoft Hyper-V/SCVMM 上で動いている仮想マシンを AWS に移行することができます。オンプレミスの仮想化環境に AWS Server Migration Service Connector を導入して、AWS Server Migration Service Connector 経由で AWS 環境に仮想マシンのデータを転送します。こちらも VM Import/Export 同様、移行要件が合致しているかどうかはしっかりと確認する必要があります。自社で対応が難しい場合には同ツールを使ったソリューションを提供している AWS のパートナーに依頼して進めることを考えてみるのも良いかもしれません。

大容量データの転送

ネットワーク経由のデータ転送

移行作業時は、データの転送を考えるとセキュリティが担保されたネットワークが必要です。AWS ではインターネット経由の通信はもちろん、インターネット VPN、専用線(AWS Direct Connect)を利用することが可能です。インターネット経由でのデータ転送にセキュリティ面などから制約がある場合、リードタイムとコストを理由として、インターネット VPN 経由でのデータ転送が採用されることが比較的多いですが、回線のセキュリティと通信品質を重視する場合は AWS Direct Connect を利用することになるでしょう。

AWS Snowball

移行作業では、大容量のデータを AWS に転送させないといけないケースもあるでしょう。例えば、ネットワーク越しに数十 TB のデータを移行する必要があったとしたら、データ転送だけで相当の時間がかかります。AWS Snowball はそんなときに利用を考えたいサービスです。AWS Snowball は大容量の物理デバイスを利用した、データの配送サービスです。AWS からデータを移行するための Snowball デバイスが送られてくるので、この Snowball デバイスにオンプレミス環境のデータを転送したのち、AWS に Snowball デバイスを配送します。AWS 側でのデータインポートが完了すると、Snowball デバイスに保存したデータは、オブジェクトストレージである S3 に格納されます。Snowball デバイスには 10GBaseT ネットワーク接続が同梱されているため、大量のデータもすばやく転送することができます。また、Snowball デバイスは耐障害性やセキュリティ面でも考慮されており、データ配送上のリスクも考慮された設計になっています。

データベースデータの移行

AWS Database Migration Service

データベースのデータを移行する場合に活用したいのが AWS Database Migration Service です。これはデータベース間のインポート・エクスポート、同期を実現するものです。これは異なる DB エンジン間のデータ移行にも対応しています。例えば Microsoft SQL Server から MySQL のような移行にも活用可能です。多彩なソース、ターゲットエンドポイント指定が可能のため、活用の仕方次第では柔軟な移行計画が立てられそうです。

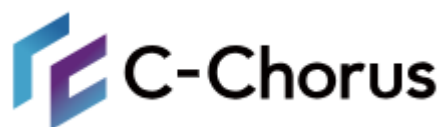
異なる DB エンジン間のデータ移行は AWS Schema Conversion Tool(AWS SCT)を合わせて使っていくことになるでしょう。ただ、全てが完全に移行できるとは限りませんので、基本的には事前に検証の上で作業計画を立てる必要があるでしょう。

以上が、オンプレミス環境から AWS への移行を行う際に活用したいサービスです。

おわりに

オンプレミス環境から AWS への移行を考えたとき、全てを AWS に移行してしまう、というのが理想ではあります。しかし、環境によっては、運用レベルから、移行作業に必要な停止時間を確保できない、などの理由で、AWS に全てを移行するのが難しいケースもあります。この場合はオンプレミス環境に一部を残し、AWS とのハイブリッド環境を志向するのが解決策になるかもしれません。

AWS への移行は運用や管理の負担が大きいものから行い、手間を減らすことから始めることがおすすめです。運用の時間を最小化し、新しい技術の検証や開発への注力など本来の事業に貢献できる部分に時間をあてることができるようになるでしょう。本記事の内容がみなさまの AWS 移行への一助となれば幸いです。



NHN テコラスの AWS 活用支援サービス C-Chorus では、物理やマルチクラウドに精通したエンジニアチームが導入・移行を徹底サポートします。オンプレミス環境に構築したシステムの AWS への移行も安心です。

AWS 導入・移行支援サービス