

マイグレーションの現実解 リホストのポイント

レガシーシステムを最新プラットフォームで動かすには？ <リホストの進め方>



メインフレームなどレガシーシステムを活用している企業が、オープンシステムへ移行するために理想的な方法はハードウェア/ソフトウェアすべてを新しいプラットフォームに置き換えるリビルド(リエンジニアリング)です。しかし、現実的には時間もコストもかかるハードルの高い作業です。より現実的な選択肢としてソフトウェアの移植作業を中心としたリライト、ツールや仮想環境を利用したリホストに注目が集まっています。その中でも移行期間や予算面で選択する企業が多いリホストを中心にシステムマイグレーションのポイントを整理していきます。

マイグレーションとその進め方

前回は、レガシーシステムにおける課題と、その解決方法となる「マイグレーション」について探ってきましたが、実際にマイグレーションを進めるには、どのような作業が必要となるのでしょうか。

メインフレームを撤廃し、稼働中の業務システムをオープンシステムやオープンプラットフォーム上のサービスに移管する場合、まず考えたいのは、既存システムのうち「本当に必要な部分はどこか」という点です。

企業にとってコアビジネスが変わらないとしても、事業体制や市場、社会環境の変化によって、あまり使わない機能、稼働していないシステムがあるはずで、使わない機能の棚卸しを行うことで、システムマイグレーションの手法によっては、費やす時間とコストを圧縮することができます。

残すシステム、残さないシステムの整理ができれば、リビルド、リライト、リホストといったマイグレーション戦略を決定します。簡単におさらいすると、リビルドはシステムをほぼ新たに構築する作業となります。もし、要件定義や仕様など流用できる部分があるなら、ソフトウェアを新しい言語で移植するリライトが選択肢に挙がるでしょう。

しかし、いずれもコスト面が問題となります。コストや時間を節約しつつ、レガシーシステムを生かしたい場合はリホストが有力な選択肢となります。



マイグレーションの現実解 リホストのポイント

もちろん、どの戦略を採用するかは、マイグレーションするシステムの重要度（経営への貢献）、予算、時間などに依存します。最近では、移行ツール、仮想環境など、ソリューションも充実しつつあり、リホストを選択する企業が少なくありません。

リホストするサーバーについてはIaaS (Infrastructure as a Service) やPaaS (Platform as a Service) を利用すれば、さらに設計の自由度が増し、マイグレーションのためのハードウェアコストを削減することも期待できます。

マイグレーションのタイミングとスケジュール

マイグレーションを実施するタイミングですが、多くの場合、ハードウェアリースの更新時期が一つの目安になります。メインフレームや古い機器の場合は、メーカーが製造を終了したり、補修部品を含むサポートが終了したタイミングなどでも、マイグレーションを検討せざるを得なくなることがあります。

いずれの場合も、終了期限は事前に把握できるはずですので、マイグレーション作業、テスト作業を含め、余裕を持ってスケジュールを立てておくことが重要です。

リビルドやリライトは、一部の要件定義や設計までのプロセスを省略できる可能性はありますが、原則として新規開発を同じ作業工数や負荷を想定してスケジュールを考えておくべきでしょう。

一方、リホストは移行ツールがうまく適合すれば、かなり短期間でのマイグレーションが期待できます。ベンダーの中には、テスト期間も含めて数か月の移行メニューを提示している場合もあります。テストで問題が発生すればデバッグ作業は必須です。リホストとはいえ、すべて自動化できるわけではないので、とくにテストの手順、期間については注意する必要があります。

旧バージョンのWindowsやLinuxベースのシステムをマイグレーションしたい場合、古い環境をそっくり仮想化環境でラッピングすることも可能です。ラッピングで対応が可能であれば、ソフトウェアリソースはほぼそのまま仮想化マシンに移設するだけで済みます。サポートが終了したOSを使い続けたいなど、セキュリティ要件を満たさないシステムの温存に利用されるスキームですが、これも一種のリホストと言えます。

移行に必要な作業工数は、元のシステムに依存します。独自の業務プロセスに特化したもの、独自データベースに依存するものなど作り込みの度合いが高いほど、時間がかかりますので、スケジュールについては依頼するシステムベンダー等と綿密に検討・調整する必要があります。

マイグレーションの現実解 リホストのポイント

マイグレーションの注意点

マイグレーションの作業に際して注意すべきポイントとその対策を以下に整理してみましょう。

リビルド

スクラッチでの要件定義や設計は必要ありませんが、プラットフォームが変わるのでシステムの必要な部分、不要になる部分、設計変更が必要な部分を洗い出し、移行のための戦略を立てる必要があります。「とりあえずそのまま移行したい」「移行したほうが簡単だ」と考えがちですが、リビルドによる移行後も長期で運用する計画があるなら、業務プロセスも含めて見直しについても検討した方が良いでしょう。これを怠ると、テストや運用が始まってからのトラブル、改修につながりやすくなります。無駄なコストを抑える意味でも欠かせない作業です。

テスト

通常の開発案件と同様に、テストは網羅的に実施することが重要です。開発中、移行中のデバッグテストはもちろんですが、本番環境に移行するためのテスト運用期間も考慮しなければなりません。本番運用で初めて表面化する不具合もあるので、検収後のサポートやメンテナンスについても考えておきます。ツールを使ったリホスト作業でも、コードが意図したとおり動作をしない場合は手作業による修正が必要となります。古いデータベースシステムや特殊なデバイスの処理部分は、移行ツールでは対応しきれないことがあります。



パフォーマンス

マイグレーションしたシステムのパフォーマンスが十分でないこともあります。CPU性能など格段に進化しているはずなのに、処理が遅くなる問題もしばしば発生します。原因はケースバイケースのため、原因を特定し、解決するノウハウが必要となります。

セキュリティ

オープンシステムの場合、セキュリティ対策は最優先事項のひとつです。メインフレームやレガシーシステムの時代にはなかった概念なので、マイグレーションでは、移行したあとのシステムに必要なセキュリティ機能を実装するか、プラットフォームごとに外部との接点を厳重に管理しなければなりません。具体的には新しいプラットフォームをルーターやファイアウォールで切り離し、他のサーバーと同様なセキュリティ対策を実施します。セキュリティ面を考えると、新プラットフォームは、一般的にはオンプレミスサーバーよりIaaS、PaaSの利用が望ましいでしょう。

マイグレーションの現実解 リホストのポイント

成功に導くためのポイント

マイグレーションで失敗しないためには、まず事前の戦略をしっかりと立てることで、どのような改善が必要なのか、そしてどの部分が要らないのか。課題を明確化し、その上でマイグレーションのスキームを決定します。

そのためには、他社事例や使えるソリューションの精査が不可欠となります。SIerやシステムベンダーに依頼するなら、よく相談し、自社の要求を満たせるかどうか見極める必要があります。過去にどのような業界、どのようなシステムのマイグレーションを手掛けた実績があるのか。規模や移行後のサポート、メンテナンスの条件などもチェックしておく良いでしょう。

レガシーシステムとはいえ、一時しのぎのマイグレーションでなければ今後も長期的に使い続けることになります。きっかけはリース契約終了やハードウェア維持コストの削減かもしれません。しかしそのような場合でも、システムを戦略的に活用していくため、マイグレーションをアップグレードの一つのスキームと捉えておく姿勢が重要となります。

例えば、乱立したサイロシステムの統廃合、IaaSやPaaSなどクラウド活用によるリソースの最適化、ハードウェア資産の陳腐化リスクの軽減、システムの可用性やスケーラビリティの向上など、マイグレーションは、そんな改善策を実施するチャンスでもあります。

リホストは単なるハードウェアのリプレースではありません。仮想化やクラウドといった新しい技術を活用することで、「攻めの経営」につなげていくことができるのです。

