

基幹データベースを PostgreSQL に移行!?

その意外な理由とは?

－富士通の技術者に聞く！PostgreSQL の技術－

企業システムで運用する商用データベース製品をオープンソースソフトウェア（以降、oss）に置き換えることで、企業はさまざまなメリットを享受できる。コスト削減はもちろんのこと、コミュニティで実装される最新技術をいち早く活用して、新たなシステムを立ち上げることも可能だ。しかし既に稼働しているデータベースを別のデータベースにリプレースするためには、リスクも伴う。そのため、oss のメリットは十分認識しつつも、導入に躊躇して現状のデータベースをやむなく使い続ける企業も少なくない。そのような企業に対して、富士通は「富士通版 PostgreSQL」へのデータベース移行をお勧めしている。ベンダーが独自に提供するデータベースを oss ベースのデータベースに移行することで、何が起きるのか？その真相に迫る。

山本 明範 Akinori Yamamoto

富士通株式会社 ミドルウェア事業本部 データマネジメント・ミドルウェア事業部 マネージャー

専門分野：データベース

入社以来、データベース（Symfoware）の開発と、製品の技術支援に携わってきました。Symfoware が PostgreSQL を取り込んだことで、新たな商談の機会や、データベース関連製品を扱う皆様との接点が増え、忙しい毎日が続いています。オープンソースの PostgreSQL を中核としたことで、いろいろな協力者や、ツール、開発手法が使える点が従来の Symfoware にはない魅力だと感じています。Symfoware としてだけでなく、PostgreSQL エンタープライズ・コンソーシアムという日本企業の団体活動にも参加、ミッションクリティカル性の高いエンタープライズ領域のシステムへ、PostgreSQL の普及に尽力しています。

渡邊 博文 Hirofumi Watanabe

富士通株式会社 ミドルウェア事業本部 データマネジメント・ミドルウェア事業部

専門分野：データベース

入社以来、データベースの開発に携わり、アプリケーションインターフェイス機能を中心に担当。5 年ほど前から他社データベースからの移行支援に参画している。移行支援立ち上げ当初は、移行ツールなどが揃っておらず、アセスメントに 1 か月要する案件もあったが、現在では、最速で 1 日で完了するまでにスピードアップ。アセスメントに対するセキュリティ対策としては、資産があるところに向く、出張アセスメントで対応。本年度は、アセスメント依頼が好評で、昨年度の 5 倍の案件を対応中、週に 2 日は、出張アセスメントにも対応し、全国飛び回っている状態。今後は、アセスメントした案件を確実に、移行してもらえよう、製品へのフィードバック、移行ツールや移行ドキュメントの整備を実施していきたい。

データベース移行における最近の傾向

今回は移行について伺います。データベースというと、ベンダーが独自に提供する製品が根強いという印象がありますが、oss への移行の要件は多いのでしょうか

山本

そうですね。近年、増加しています。当社では特に PostgreSQL の採用が目立ち、昨年の同時期と比較すると 5 倍に伸びています。とはいえ、既に長年安定稼働しているシステムのデータベースを異なる製品にリプレースするとなると、これまでと同様の性能や運用を維持できるかというリスクが生じます。また、データベース製品ごとに SQL の仕様も若干異なりますから、それを利用するアプリケーションの改修やテストも必要になるでしょう。こうしたことから、なかなか PostgreSQL への移行に踏み切れない企業も多いようです。

移行をする、または移行に注目する企業に何か共通の傾向はありますか？

山本

以前はコスト削減のため、単体で稼働しているシステムやパッケージソフトウェアに組み込まれているデータベースを PostgreSQL に置き換えたいという要望が多かったのですが、最近は新しい業務に PostgreSQL を適用することが増えていて、その新規システムと既存システムの連携性をよくしてビジネスを活性化させたいという要望から、業務システム全体のデータベース基盤を見直

したいという相談も受けるようになりました。業務システムを構成する複数のデータベースを順次 PostgreSQL に置き換えていき、最終的にはデータベース基盤すべてを PostgreSQL で構成したいというニーズが増えてきているように感じます。

移行への注目は大きくなっているのですね。しかし、移行するということは簡単ではないとも思います。何かノウハウはあるのでしょうか。

渡邊

以前より、他のデータベース製品から当社の「Symfoware Server」への移行支援を実施してきていますが、そこで培った移行ノウハウで移行支援メニューを体系化して、アセスメント方法や移行ツール、移行ドキュメントを整備してきました。PostgreSQL への移行には、この Symfoware Server で蓄積してきた豊富な移行ノウハウを活用して、同様の支援メニューを用意しています。そして、そこで得たノウハウは PGECons（注 1）などの外部コミュニティに参画することで、PostgreSQL の技術情報蓄積にも貢献してきています。

注 1 「PostgreSQL Enterprise Consortium」の略で日本語表記は「PostgreSQL エンタープライズ・コンソーシアム」。PostgreSQL がエンタープライズの業務システムに適用できるようにするため、PostgreSQL 本体および各種ツールの情報収集と提供、整備などの活動を通じて、ミッションクリティカル性の高いエンタープライズ領域への PostgreSQL の普及を推進することを目的として設立された団体。

PostgreSQL のオープン性を重視した移行を推奨

ノウハウがツールやガイドの形になっているのですね。ところで移行というと、互換機能を提供するという考え方もあると思いますが、互換機能は提供していないのでしょうか。

渡邊

富士通版 PostgreSQL にも、他社データベースとの互換機能が備わっています。これを利用すれば、確かに商用データベースからの移行工数を最小限に抑えることができるでしょう。ただし私たちは、必ずしもこれが常にベストな方法だとは限らないと考えています。

と言えますと？

山本

確かに互換機能を使えば、データベース移行に伴うアプリケーションの修正作業は最小限で済みます。しかし結果的にそのアプリケーションは、依然として元々使用していた商用データベースの仕様に縛られ続けることになります。せっかく PostgreSQL を導入しても、アプリケーションが商用データベースの仕様に縛られたままでは、OSS のメリットも半減してしまいます。

「オープンソースならではの」「PostgreSQL ならではの」のメリットは、確かにこの方法では手に入れることができませんね。

山本

商用データベースとの互換性を保つためにベンダーが独自に実装した部分は、ブラックボックスになるので、オープン性は損なわれると考えるべきでしょう。その結果、商用データベース製品のベンダーロックインから脱出するために PostgreSQL へ移行したはずなのに、結局はまた別のベンダーにロックインされてしまうような羽目に陥りかねません。そうした事態を避けるためにも、私たちは可能な限りお客様の資産を PostgreSQL ベースに修正して移行することを提案しています。

修正箇所が多岐に渡る場合には、ある程度コストがかかるかもしれませんね。

山本

そうですね。互換機能をフル活用した場合と比べると、やはりコストは多くかかるかもしれません。しかし先ほどお話ししたとおり、たとえオープンソース製品に移行したとしても、オープン性が失われてしまっただけではそもそもの導入目的は達成できません。PostgreSQL のオープン性がもたらす将来のイノベーションを考えての移行であれば、多少のコストがかかったとしても、これを機に PostgreSQL への対応を一通り済ませておくことが望ましいと考えます。

その際のハードルを可能な限り低くすることが、データベース移行支援の目的なのですね。

山本

そのとおりです。ただ、お客様が抱える事情によっては、もともと使っていた商用データベースとの互換性を維持した方がいいケースもあります。従って「ここはオープンで」「ここは互換性維持で」といったように、ケースバイケースで柔軟に対応することも多いですね。

データベース移行の「何を、どのように」支援してくれるのか？

次に、移行アセスメントについて教えてください。

渡邊

移行アセスメントは、お客様が現在保有しているデータベース資産の情報をいただいた上でその内容を分析・評価して、富士通版 PostgreSQL への移行にどの程度の工数がかかるか、あるいはどんな技術的な課題が存在するかを洗い出すサービスです。この結果を基に、移行に踏み切るかどうかをお客様に判断していただきます。

「データベース資産の情報」とは、具体的にどのようなものを指すのでしょうか？

渡邊

現行のデータベースの DDL 定義や独自のプロシジャの SQL、あるいはアプリケーションのソースコードなどの情報です。お客様からこれら情報をいただき、それを富士通の技術者が分析し、富士通版 PostgreSQL への移行に要する作業工数を見積もります。この「アセスメント」が移行の第一歩としてとても重要な役割を果たします。アセスメントの結果をまとめた報告書には全体の工数だけでなく、修正作業が必要な箇所 1 つ 1 つにつき、その詳細な内容や難易度、そして「移行ツールを使った場合の工数」と「移行ツールを使わなかった場合の工数」をそれぞれ算出して提示しています。

〇〇株式会社 御中

2017年2月1日
富士通株式会社

移行に向けて(ご報告)

富士通版PostgreSQLへの移行に向けて、アプリケーション、DDL定義、PL/SQL定義をアセスメントした結果をご報告させていただきます。

1. アセスメント結果(概要)

対象資料	全体本数 (本数)	ステップ数 (Ks)	修正箇所 (箇所数)	修正工数1 (人日)	修正工数2 (人日)	備考
DDL	46	5.2	1,856	2.3	2.3	
PL/SQL	437	174.6	21,935	243.3	119.4	
SQL	9	1.4	14	0.1	0	
合計	492	181.2	23,805	245.7 12.3人月	121.7 6.1人月	

上述した移行工数は、移行対象資料の置き換え作業に掛かる工数です。テスト作業、性能評価などは含まれていません。
修正工数1: 移行ツールを使用しない場合の工数、修正工数2: 移行ツールを使用した場合の工数

2. アセスメント結果(詳細)

2. 1 DDL

修正分類	修正箇所 (箇所数)	修正工数1 (人時)	修正内容	修正工数2 (人時)	Level
CREATE TABLE文	46	0.4	「CREATE TABLE」は使用可能です。 文法については確認してください。	0.4	2
CREATE VIEW文	16	0.2	「CREATE VIEW」は使用可能です。 文法については確認してください。	0.2	2
合計	1,856	18.6		18.6	

2. 2 PL/SQL

修正分類	修正箇所 (箇所数)	修正工数1 (人時)	修正内容	修正工数2 (人時)	Level
ストアドファンクション	14	0.8	ファンクションのみ使用できます。 PL/SQLでは、ファンクションの宣言部をISまたはASで開始しますが、ASを指定します。	0.8	2
トリガー	1	0.0	トリガーは使用できません。 トリガーを明示してください。	0.0	1
MERGE文	15	15.0	UPSET文に書き換えてください。	15.0	2
UTL_FILE	550	315.0	対応する機能が無いため、別途対応が必要です。	315.0	3
UTL_MAIL	542	311.0	対応する機能が無いため、別途対応が必要です。	311.0	3
合計	21,935	1,946.3		955.1	

2. 3 SQL

修正分類	修正箇所 (箇所数)	修正工数1 (人時)	修正内容	修正工数2 (人時)	Level
SQL	9	0.1	SQLは使用可能です。	0.1	1

図：アセスメント結果のイメージ

工数やコストの概算から判断できるのですね。アセスメントが完了した後は、どのような作業を行うのでしょうか？

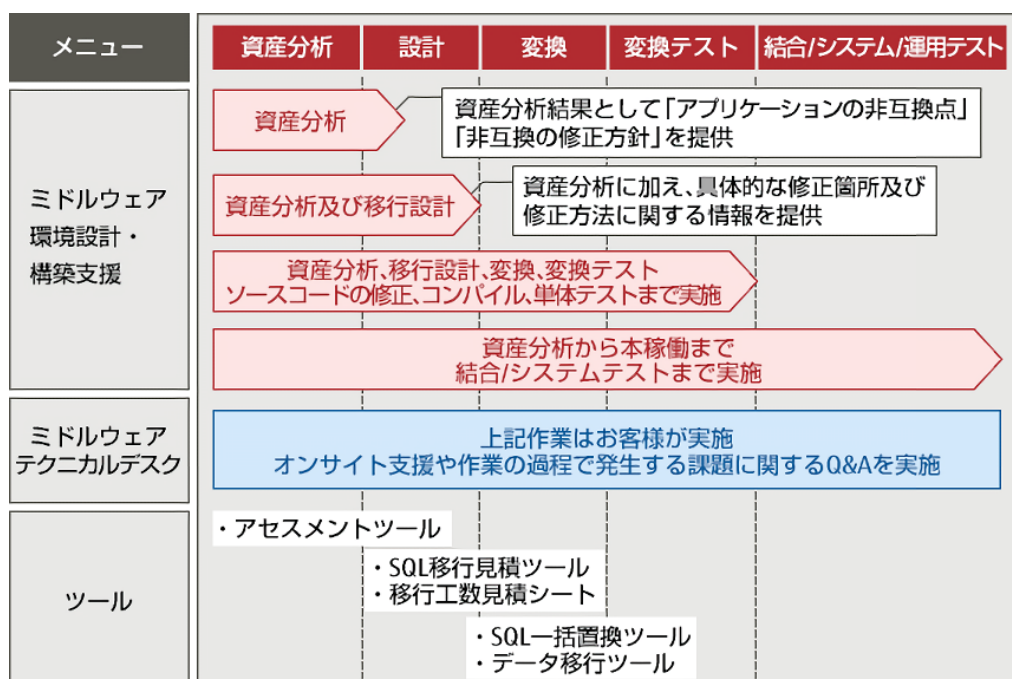
渡邊

アセスメント結果を基に、お客様がご自身で移行作業を行う場合もありますが、当社が移行作業を「移行サービス」として請け負う場合もあります。お客様の状況に応じたサービスを用意しています。

予算やリソース、時間的な制約などの都合に応じて、柔軟に支援の範囲を選択できるというわけですね。

渡邊

はい。お客様がご自身で行う場合は移行ツールや移行ガイドを提供いたします。さらにテクニカルデスクという形での支援もあります。



図：移行サービスの体系

移行ツールにはどのようなものがあるのでしょうか。

渡邊

例えば、オープンソースでは他社データベースからスキーマやデータを抽出し、PostgreSQL 用に変換するツールがあるので、データの移行はそのツールを使用してできます。当社からは、アプリケーションの移行ツールを提供しています。このツールを使用すると、ソースコードなどの記述を単純に置き換えるだけで対応できる作業は、自動的に処理できます。そのため、ほとんど移行工数が掛かりません。ただ、ツールによる自動置き換えだけでは対応しきれないものもあり、中には富士通版 PostgreSQL に対応する機能がないために、個別にプログラム開発などで対応しなくてはならないケースもあります。

移行ツールですべて置き換えるということは、やはり難しいのでしょうか。

渡邊

そうですね。アプリケーションは一つひとつ違いますから、それぞれの処理を考慮して移行しなければなりません。ですから、すべてを一括置換することは現実的に難しいです。この問題を少しでも解消するために、移行ツールにかけた際に自動置き換えできない部分には具体的なヒントを記載し、効率的に修正できるようにしています。

複数のデータベースを段階的に PostgreSQL へ置き換えていくには

移行する企業の傾向として、単体の移行から基盤全体を移行するという要件が出てきていると伺いましたが、このようなケースでは、移行の難易度は高くなるのでしょうか？

山本

確かに、複数のシステムで構成された大規模な業務システムの場合、やはりデータベース製品の置き換えは一筋縄ではいきません。すべてを一括して置き換えられればいいのですが、それではあまりにもリスクが高すぎます。かといって、1台ずつ PostgreSQL に入れ替えていけばいいかというと、それはそれでさまざまな課題に直面することになります。その最たるものが、データベース間の連携にまつわる問題です。

具体的には、どんなところが問題になるのでしょうか？

山本

複数のシステムから構成される業務システムでは、多くの場合、それぞれのデータベースが密接に連携しています。例えば、よく利用されているのが「マテリアライズド・ビュー・レプリケーション」という機能です。これはある商用データベースの機能なのですが、テーブルのようにクエリの結果をキャッシュしておける「マテリアライズドビュー」に、外部データベースから取得したデータを保持しておくことができます。外部データを参照する際に、毎回、外部データベースにアクセスしなくてよいため、効率よくデータを参照できます。このような場合、片方のデータベースを PostgreSQL に移行した際にも、移行前と同様にもう一方のデータベースのデータを参照できるようにする必要があります。

一部のデータベースを PostgreSQL に移行した場合も、業務の運用には影響を与えないようにするということですね。PostgreSQL に移行しても連携できるのでしょうか。

山本

PostgreSQL の場合、Foreign Data Wrapper（以降、FDW）を使用すると、外部の異種データベースと連携できます。ただし、FDW は外部データを参照するたびに、外部データベースにアクセスするため、それだけでは外部データの参照効率が低下してしまいます。PostgreSQL の場合もマテリアライズドビューと組み合わせることで、データを効率的に参照することができます。ただ、マテリアライズド・ビュー・レプリケーションが差分データのみを反映することでデータベースやネットワークに負荷をかけずに効率的に更新できる一方で、FDW とマテリアライズドビューの組合せの場合、すべてのデータを入れ換えることしかできません。

例えば、大量のデータを連携させる場合、データベースやネットワークに負荷がかかり性能に影響がでる、ということでしょうか。

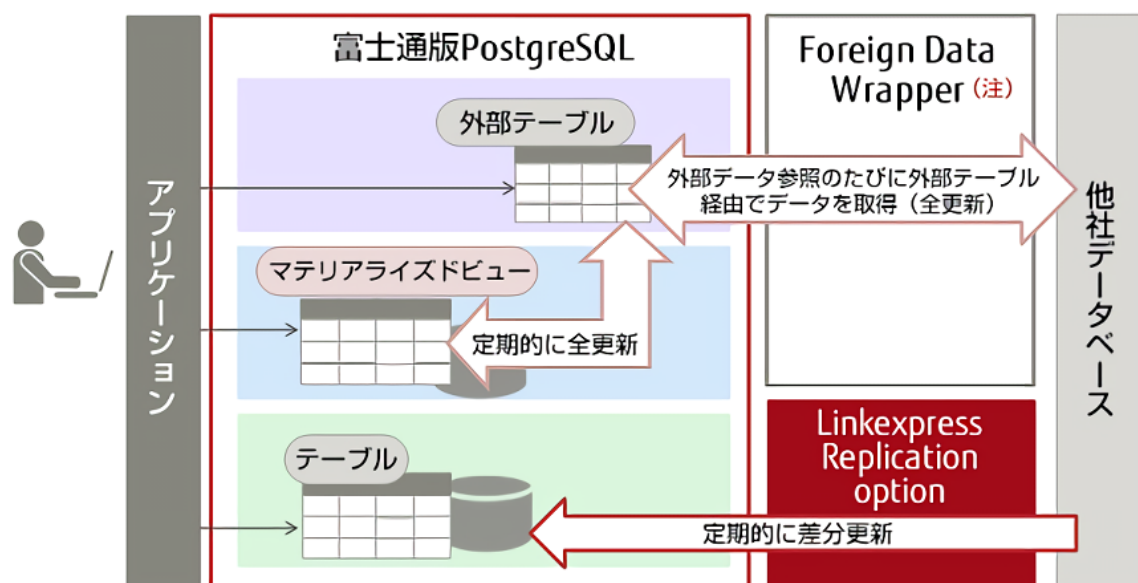
山本

そのリスクは否めません。FDW は OSS なので、差分反映できるように開発するという手段もありますが、富士通では「Linkexpress Replication option」というデータ連携製品を活用して、この問題を解消しています。これを使えば、マテリアライズド・ビュー・レプリケーションと同様に、データの差分反映が可能になります。データ量が多い場合や頻繁に更新をかける場合は、この方法が有効です。

状況に応じて手段を選択することで、PostgreSQL に移行した後も、移行前と同じ業務を実現できるのですね

山本

はい。特に移行の場合、移行前の性能や運用を維持することは、移行決定を左右する重要な要因になります。FDW やマテリアライズドビュー、Linkexpress Replication optionなどを業務に合わせて選択していくことで、段階的な移行を実現しています。



(注) Foreign Data Wrapperによる連携は一例です。開発により、更新方法を変更することができます。

図：データベース間の連携

データやアプリケーションが移行できるだけでなく、移行前の性能や運用などをどう維持できるかということも含めて「移行できる」ということが重要なんですね。今後、さらに移行支援策を拡充していく予定などはありますか？

渡邊

移行にかかる工数をさらに圧縮するためには、移行テストの作業をもっと効率化が必要があると考えています。そのために、移行テストを自動化できるツールや、効率的なテストを行うためのガイドなどを提供していく予定です。既にオープンソースでアプリケーションテストを自動化できるツールなどが出回っていますから、そういったものをうまく活用する方法なども提案できると思います。移行作業全体の中でテストが占める割合はかなり大きいので、これを大幅に効率化できれば移行のハードルもさらに低くできるはずです。

移行へのさまざまな不安が払拭されるお話でした。ありがとうございました。

2017年2月3日