

移行ツールを用いた高速道路料金収受システムへのOSS導入手法

Method Using Migration Tools to Implement Migration from Commercial Software to Open-Source Software for Expressway Toll Collection Systems

神崎 大智 KANZAKI Daichi 弓倉 陽介 YUMIKURA Yosuke

高速道路料金収受システムをはじめとするインフラシステムでは、安定稼働や長期保守の観点から、システム構築に必要な基本ソフトウェア(OS)やミドルウェア(MW)は、従来、商用ソフトウェアが用いられてきた。しかし近年、新サービスへの対応や、行政や事業者の動向から、オープンソースソフトウェア(OSS)を活用する必要性が高まってきており、移行コストや、品質保証、長期運用・保守など様々な課題を解決する必要がある。

そこで東芝インフラシステムズ(株)は、高速道路料金収受システムにおいて、既存資産を生かしながら、効率的かつ迅速にOSSを導入するため、移行ツールを活用した手法の開発を進めている。今回、従来のデータベース(DB)ソフトウェアからOSSのPostgreSQLへの移行を試行し、この手法の有効性を確認した。

Commercial software has conventionally been used for the operating systems and middleware of many infrastructure systems, including expressway toll collection systems, from the viewpoint of ensuring the stable operation and long-term maintenance of such systems. In recent years, however, there has been a growing need for the replacement of commercial software with open-source software (OSS) in response to the introduction of various new services as well as the changes that taking place in the public administration and business environments. It is necessary to address a variety of issues in this regard, including migration costs, quality assurance, and long-term operation and maintenance.

To resolve these issues, Toshiba Infrastructure Systems & Solutions Corporation has been engaged in the development of a method using migration tools to swiftly and efficiently implement migration from commercial software to OSS for expressway toll collection systems while making use of a system's existing assets. We have confirmed the effectiveness of this method through trial migrations of applications and data from conventional commercial software to PostgreSQL, an open-source relational database management system, using three types of migration tools.

1. まえがき

CASE (Connected, Automated, Shared & Service, Electric) は、自動車産業に大きな変革をもたらすといわれている。そして、その進展は、車両にとどまらず、道路インフラ、及び高速道路料金収受システムをはじめとする情報インフラの高度化を促していくと予測される。これは、情報インフラが高度な無線通信技術を用いて、自動運転車に対してドライバーを介さずに、直接やり取りできるようにするためである。

特に、高速道路の料金収受分野では、“Connected”によって車両から取得できるデータが多量・多様・多頻度になり、このデータを生かした料金施策の転換への要望が高まる可能性がある。また、“Shared & Service”の発展のためには、MaaS (Mobility as a Service)と高速道路料金収受システムとの連携が考えられ、ほかのサービスと道路利用料の決済一元化や、予約システムなどとの連携による道路利用の需給調整などが想定される。これらを実現するため、高速道路料金収受システムは今後、ほかの様々なシステムとの接

続や、多種多様なデータ交換に応じる必要がある。

一方で、高速道路料金収受システムは、主要道路事業者(東日本高速道路など合計6社)の平均交通量では令和元年度累計で約1,000万台/日⁽¹⁾、平均料金収入金額では約100億円/日⁽²⁾を扱うシステムであり、現代社会に欠くことができない重要なインフラシステムとなっている。24時間365日無停止で稼働し、車両メーカーに関係なく公平で、正確な料金を収受できることが、何よりも求められる。このため、品質保証や長期の保守対応が必要であり、これまでは通常、商用ソフトウェアが採用されてきた。しかし、CASEによって日々新しいサービスが生まれ、変容していくことを想定すると、旧来のシステム構築の方法では、構築に時間を要するため、その変化に追従できないことが危惧される。それに対し、近年、新しい技術をいち早く取り入れていくOSSの発展は目覚ましく、社会的な要求も相まってOSSの導入検討が必要になってきている。

ここでは、CASEの進展に伴う社会的な要求の変化に対応するために、高速道路料金収受システムにOSSを導入する際の課題を整理し、実際に移行ツールを用いて試行した

事例と評価について述べる。

2. 高速道路料金収受システムへの社会的要求

2.1 インフラシステムの構築時間の短縮

CASEの急速な進展に伴い、例えば自動運転車やコネクテッドカーなどは、通信を活用してソフトウェアをアップデートできるようにする計画³⁾が見られる。近い将来、車両のアプリケーションが、スマートフォンのように簡単に追加・機能向上できるようになる可能性が高い。

一方、従来のインフラシステムは、仕様の検討から実際のサービスリリースまでの構築に多くの時間を要している。施策の検討開始からサービスリリースまで、長いものでは数年掛かるケースも見られる。それに対してOSSは、極めて短いサイクルでバージョンアップを繰り返し、新しい技術を取り込んで進化し続けている。そのため、OSSを適宜、高速道路料金収受システムに適用することができれば、構築に掛かる時間を削減して、CASEによる急激な変化に追従した様々な新しいサービスに対応することが可能になる。

2.2 OSSの導入の機運

近年、政府は、官公庁における情報システムの調達について、ベンダーロックイン(特定のベンダー技術に依存したシステムや製品を採用したために、ほかのベンダーに乗り換えることが困難な状況)を回避するために、OSSの活用を検討事項として掲げている⁴⁾。高速道路料金収受システムも公共性の高い情報システムであり、OSSを活用する方針が求められている。

また、インフラ事業者も、システム導入・運用コスト削減のために、従来の商用ソフトウェアではなくライセンス費が不要となるOSSの活用を検討している。その結果、特定のOSSを使用することやライセンス料が必要な製品を使用しないことが要求仕様に盛り込まれるなど、具体的な動きも見られ始め、OSS活用の機運が高まっている。

3. OSS導入の課題と解決策

3.1 課題

既存システムへのOSSの適用には、幾つかの課題がある。例えば、OSSにより従来機能を実現するための移行コスト低減や、システム構築及び運用時における保守サポート体制の確保、OSSのソースコードに含まれる脆弱性への対応と悪意あるコードの検閲、利用するOSSのライセンス管理と当該OSSコミュニティへの貢献、などである。

既存システムやデータといった、今ある資産を活用しながら、どのようにして安全かつ迅速にOSS化へ対応していくかが課題となる。

3.2 移行ツールの活用

既存資産を生かしながら、効率的かつ迅速にOSSへ移行する一つの方法として、移行ツールの活用がある。

4章以降で、移行コスト低減及び頻繁に更新されるOSSに対応する、東芝インフラシステムズ(株)の取り組みについて述べる。具体的には、当社の高速道路料金収受システムで主に使用されている商用DBソフトウェアから、OSSのDBソフトウェアであるPostgreSQL⁵⁾への移行を、効率化と低コスト化を目的に、移行ツールを活用して試行した事例について述べる。

4. 移行手法の検討と適用

DBソフトウェアの移行には主に、移行アセスメント(評価)実施や、スキーマ(テーブルや、ビュー、インデックスなど)の移行、アプリケーション(プログラムや、SQL (Structured Query Language) など)の移行、データ(既存システムのDBに格納されているデータ)の移行などが必要となる。今回は、PostgreSQLの普及を推進することを目的に設立されたPostgreSQLエンタープライズ・コンソーシアム⁶⁾が発行している移行ガイドブック⁷⁾の移行フローを参考に、次の手順で移行を実施した(図1)。

①事前チェック チェックツールであるdb_syntax_diff
 を利用し、PostgreSQLで実行できないSQL中のキーワードのチェックを行い、ソリューション管理シート(ノ

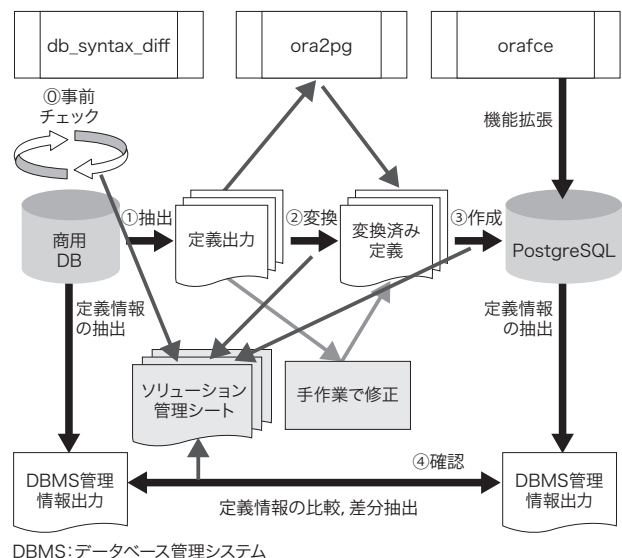


図1. 商用DBソフトウェアからPostgreSQLへのDB移行フロー
 db_syntax_diff, ora2pg, 及びorafceの三つの移行ツールを活用し、商用DBソフトウェアからPostgreSQLへの移行を実施した。
 Flow of processes for database migration from commercial software to PostgreSQL

ウハウ集)へ反映する。

- ①抽出 定義情報(DDL (Data Definition Language), DCL (Data Control Language))を商用DBから抽出する。
- ②変換 移行ツールora2pgを利用して、自動変換可能なものを交換する。自動変換できないものは、DB間の互換性をサポートするツールであるorafceを利用し、更に手作業でPostgreSQL対応のSQLに修正する。想定外の事象が発生した場合は、ソリューション管理シートへ追加する。
- ③作成 ora2pgで変換した定義ファイル及び手作業で修正したファイルを実行し、オブジェクトを作成する。エラーなどで、想定外の対応が必要な場合は、ソリューション管理シートに追加する。
- ④確認 オブジェクトの件数、種類、名称などを出力し、移行前の結果と比較することで移行結果を確認する。差異がある場合は、ソリューション管理シートに追加する。

今回、移行に活用したツールは、db_syntax_diff, ora2pg及びorafceである。これらのツールの概要を、表1に示す。

ora2pgは、アプリケーションの移行が可能であるが、SQLだけを対象としており、高速道路料金収受システムで主に使用しているC言語やJavaには活用できない。そのため、SQL以外のプログラムに関しては、db_syntax_diffを用いて修正が必要な箇所を確認し、手動修正した。

5. 結果と評価

今回は、4章で述べた三つのツールを活用したが、その中でも重要なora2pgについて、アプリケーションとデータを移行した結果と評価を、以下に示す。

5.1 アプリケーションの移行

ora2pgによるPostgreSQLへの移行では、システム全体

のうち修正が必要だったのは10%であり、残り90%は修正不要であった(図2(a))。ora2pgの活用で、修正が必要だったもののうちコーディングの20%を自動化できた(図2(b))。また、SQLに限定すれば、コーディングの42%を自動化できた(図2(c))。最後に、PostgreSQL移行で修正したコードに使われているプログラム言語は、SQLが48%、C言語が42%であり、これら二つの言語が大きな割合を占めていることが分かった(図2(d))。

これらの結果から、ora2pgはSQLの移行に有効なツールであり、十分活用できるといえる。しかし、修正したコードの中で占める割合が大きいC言語には適用できず、改善の余地がある。

更に、今回の試行で、自動変換は実行できたが、実際にプログラムを実行しないと判明しないエラー事象(例えば、商用DBソフトウェアでは問題なく実行されていた暗黙の型変換がPostgreSQLでは実行されずエラーとなる)が発生したため、コーディング後のテストは不可欠であることが判明した。

5.2 データの移行

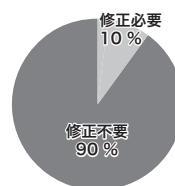
過去の運用で得られたデータを基に、種類・サイズ・件数が現実に近いデータを用意し、ora2pgを活用してデータを移行した。

データの移行には、INSERT文の実行によりデータ移行するDATAモードと、CSV (Comma-Separated Values) ファイル形式で出力したものをCOPYコマンドで登録するCOPYモードの二つのモードがある。具体的な移行データ

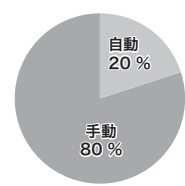
表1. DB移行ツール概要

Outline of database migration tools

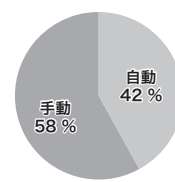
ツール名	利用工程	主な機能
db_syntax_diff	移行アセスメント アプリケーション移行	• Java, C言語などのソースコードに記載されているSQL文の修正が必要な箇所を抽出し、レポート出力する
ora2pg	移行アセスメント スキーマ移行 データ移行 アプリケーション移行	• 商用DBソフトウェアのオブジェクトをPostgreSQLのDDLの書式に変換する • 商用DBソフトウェアのDBデータをPostgreSQLのDBに移行する • SQL手続き言語(ファンクション、プロシージャ)をPostgreSQLで実行可能な形式に変換する
orafce	移行アセスメント アプリケーション移行	• 一部の商用DBソフトウェア関数をPostgreSQL上でも利用できる互換機能をプロシージャ形式で提供する



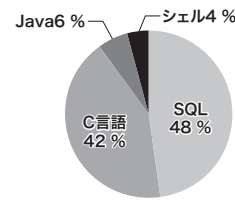
(a) システム全体に対する修正の要/不要



(b) 修正したコードの修正手法



(c) 修正したコードの修正手法 (SQLだけ)

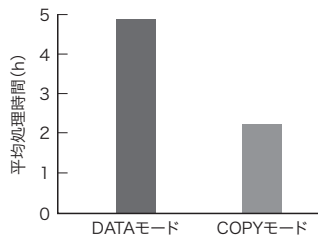


(d) 修正したコードのプログラム言語

図2. アプリケーション移行時の修正割合の結果

移行ツールの活用により、修正が必要なコードのうち20%(SQLに限定すると42%)を自動化できた。

Results of breakdown of error correction rate at time of application migration



モード	移行データ件数	平均処理時間	平均処理性能
DATAモード	274,869,047件	4 h 52 min 10 s	15,680件/s
COPYモード		2 h 14 min 6 s	34,162件/s

図3. データ件数と処理時間の結果

今回の例では、COPYモードは、DATAモードの約2倍の速度でデータを移行できた。

Results of measurement of data volume and processing time at time of data migration

件数、平均処理時間、及び平均処理性能を図3に示す。今回の例では、COPYモードはDATAモードに比べて、約2倍の速度で移行でき、COPYモードが有効であることが分かった。

6. 考察

今回使用した三つのツールはいずれも有効であり、特に、ora2pgはアプリケーションの移行、データの移行ともに、効率化に寄与することが分かった。しかし、ora2pgは、SQLの移行にしか使えず、そのほかのC言語などのアプリケーションプログラムには使用できない。SQL以外で作成されたアプリケーションプログラムの移行をいかに効率化するか、今後の課題である。

このように、プログラム言語の構成によって移行ツールの変換効率が変わるため、事前に移行対象システムのプログラム言語構成を確認した上で、開発計画(工程やコストの検討)を実施する必要がある。例えば、今回利用したツールの場合は、SQLとそれ以外の言語の割合を考慮して計画することが重要である。

また、今回の試行で、コーディングは自動化が可能であったがその後の確認テストは不可欠であることが判明した。開発工程のうちテスト工程が占める割合は少なくないため、今後はテストの自動化についても検討し、更なる効率化を目指していく。

7. あとがき

今回、当社がシステム構築して提供している高速道路料金収受システムを、移行ツールを活用して、OSSであるPostgreSQLへ移行した場合、アプリケーションの移行の

うちコーディングの約20%が自動化できることが分かった。移行するシステムのプログラム構成により移行ツールの変換効率が変わるが、移行ツールの活用は有効であることが分かった。

今後は、CASEをはじめとする社会的な情勢変化に対応するため、OSSの活用が更に拡大することが予想され、そのほかの課題であるOSSの脆弱性対応や、セキュリティ対策、長期サポートなどについても検討していく。この論文が、OSS活用の効果的な例となり、OSS普及に貢献できれば幸いである。

文献

- (1) 日本高速道路保有・債務返済機構. “各高速道路会社のR元年度最新四半期及び年度累計の交通量(速報値)”. <<https://www.jehdra.go.jp/pdf/kotu/0201-0203.pdf>>, (参照 2022-05-23).
- (2) 高速道路調査会. 高速道路統計月報 令和元年5月分. 高速道路と自動車, 2019, 62, 8, p.87-98.
- (3) 国土交通省. “自動車の特定改造等の許可制度を本年 11月より開始します —適切なソフトウェアアップデートを確保するための環境整備について—”. プレスリリース. <<https://www.mlit.go.jp/report/press/content/001358066.pdf>>, (参照 2022-06-02).
- (4) 公正取引委員会. “(令和4年2月8日)官公庁における情報システム調達に関する実態調査について”. 報道発表資料. <https://www.jftc.go.jp/houdou/pressrelease/2022/feb/220208_system.html>, (参照 2022-05-23).
- (5) The PostgreSQL Global Development Group. "PostgreSQL: The World's Most Advanced Open Source Relational Database". PostgreSQL homepage. <<https://www.postgresql.org/>>, (accessed 2022-05-23).
- (6) PostgreSQLエンタープライズ・コンソーシアム. “PostgreSQLエンタープライズ・コンソーシアム” PostgreSQLエンタープライズ・コンソーシアムホームページ. <<https://www.pgecons.org/>>, (参照 2022-05-23).
- (7) PostgreSQLエンタープライズ・コンソーシアム. “WG2活動報告書 移行ガイドブック”. <https://pgecons-sec-tech.github.io/tech-report/html_wg2_MigrationGuidebook/wg2_MigrationGuidebook.html#document-wg2_MigrationGuidebook>, (参照 2022-05-23).



神崎 大智 KANZAKI Daichi
東芝インフラシステムズ(株)
社会システム事業部 道路ソリューション技術第二部
Toshiba Infrastructure Systems & Solutions Corp.



弓倉 陽介 YUMIKURA Yosuke
東芝インフラシステムズ(株) 社会システム事業部
道路ソリューション技術第二部 技術士(情報工学部門,
総合技術監理部門) 日本技術士会・情報処理学会会員
Toshiba Infrastructure Systems & Solutions Corp.