

オープンソースによるメインフレーム資産の活用

Utilization of Legacy Mainframe Assets by Open Source Software

大塚 玲子

要約 米国ユニシスは近年 Java/J2EE 技術に注力しており、2200 メインフレームや MCP メインフレームにも最新 J2EE 技術を取り入れている。本稿では、米国ユニシスで開発されたオープンソース系ソフトウェアを用いた 2200 メインフレーム資産の活用例を提案する。2200 メインフレームを執筆対象とするが、MCP メインフレームでも同様の取り組みが行われている。2200 メインフレーム上の Java 実行環境で JBoss Application Server, JDBC ドライバ, Hibernate などのオープンソースソフトウェアを稼働させることにより、J2EE アプリケーションから 2200 メインフレーム資産の利用が可能となる。また、Eclipse での 2200 アプリケーション開発環境も提供され、2200 Java プログラムだけでなく、UCS COBOL や UCS C プログラムの開発にもオープンソースソフトウェアが利用できる。

本稿が、メインフレーム活用の再考のきっかけとなれば幸いである。

Abstract Unisys emphasizes Java/J2EE technologies and adopts these J2EE technologies on the legacy 2200 mainframe and MCP mainframe. This paper introduces how to utilize legacy mainframe assets by using open source software adopted by Unisys. The open source software, such as JBoss Application Server, JDBC Driver, Hibernate, running on the mainframe enables J2EE applications to access legacy mainframe assets. Unisys also provides 2200 plug-in for open development environment Eclipse IDE not only for the 2200 Java program but also UCS COBOL and UCS C applications. I'm pleased if this paper inspires the reader to re-consider the mainframe utilization.

1. はじめに

これまで様々な方法でメインフレームのオープン連携が行われてきたが、オープンソースの出現により、より容易により安価にオープンシステムとメインフレームとの連携が可能になった。

本稿では、米国ユニシスで開発されたオープンソース系ソフトウェアを用いた 2200 レガシー・リソースの活用例を提案する。2200 メインフレームを執筆対象とするが、MCP 系メインフレームでも同様の取り組みが行われている。また、本稿で取り扱う米国ユニシス製ソフトウェアは、日本ではまだ評価段階でありソフトウェアの仕様や詳細内容も今後変更される可能性があることに留意されたい。

2. オープンソースによるレガシー資産の活用

2200 メインフレームのオープン連携に関しては、これまで様々な製品やソリューションが提案されてきたが、結局はもっとも単純なファイル転送による連携が主流であった。米国ユニシスでは近年 J2EE (Java 2 Enterprise Edition) アーキテクチャに注力しており、メインフ

レームを含むすべてのプラットフォームで様々な J2EE ソリューションを展開している。

本章では、主に JBoss Application Server (以下 JBoss AS) を中心としたオープンソースソフトウェアを用いた 2200 メインフレーム既存資産の活用例を提案する。

2.1 JBOSS-2200 アーキテクチャ

JBoss AS はオープンソースの J2EE アプリケーションサーバであり、近年は BEA Web Logic Server や IBM WebSphere Application Server と並び、エンタープライズシステムでも使用例が増えてきている。J2EE 1.4 認定を獲得しており、商用製品にひけをとらない機能と品質を備えたオープンソースソフトウェアである。米国ユニシスは 2005 年 3 月に 2200 メインフレームの Java VM 実装である JVM2200 で稼働する JBoss AS を、JBOSS-2200 として製品化した。米国ユニシスは JBoss Inc. (2006 年 6 月に Red Hat に買収され現在は Red Hat の一部門となっている) とパートナー関係にあり、メインフレーム以外でも JBoss 関連製品の販売、サポートを積極的に行っている。

JBOSS-2200 を活用することにより、2200 メインフレームの RDMS データベースや TIP アプリケーションなど既存のリソースにアクセスすることが可能となる。JBOSS-2200 の全体概要を図 1 に示す。JBOSS-2200 では TIP/HVTIP や RDMS などメインフレームのさまざまなリソースにアクセスするためのリソースアダプタ (RA) が各種用意されており、ほぼすべてのメインフレームリソースにアクセスが可能である。

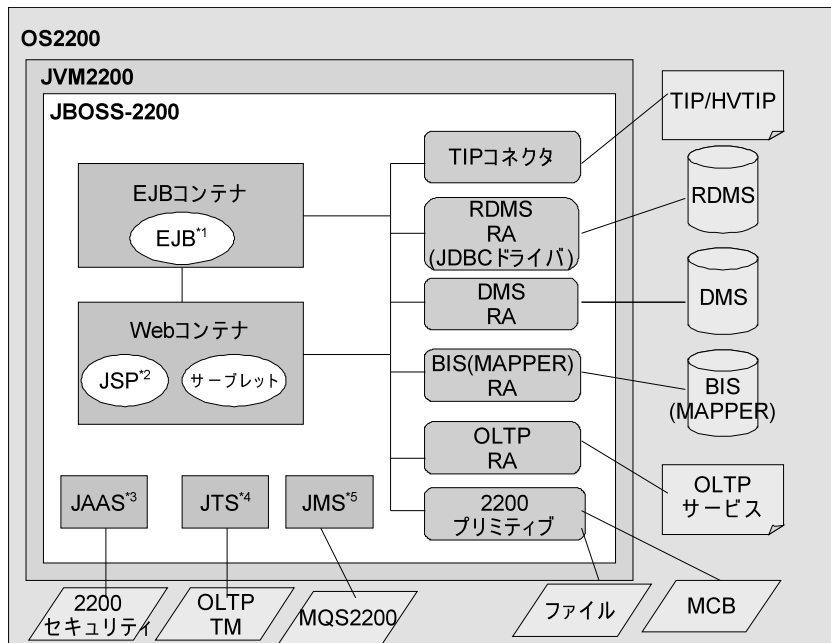


図1 JBOSS-2200 アーキテクチャ

以下に主要ソフトウェアの概要を述べる。

1) JVM2200

JVM2200 は、2200 メインフレームの EXEC OS 上でネイティブに稼働する Java Virtual Machine である。Sun Java に準拠した Java 実行環境を提供しており、他の Java 環境で作成した Java プログラムを、リコンパイルなしで稼働させることができる。JVM2200 の出現により、JBoss AS や JDBC ドライバなどの Pure Java プログラムが 2200 メインフレームでも稼働できるようになった。執筆時点 (2007 年 12 月) での最新レベルは 4R1 (Sun JRE 5.0 相当) である。

2) JBOSS-2200

JVM2200 で稼働する JBoss Application Server。オープンソースで提供されている JBoss Application Server をベースとしており、2200 メインフレーム用に以下の改良が行われている。

- ・ 2200 セキュリティとの連携
- ・ ログの 2200 コンソール出力
- ・ 管理コンソール (JBoss Administration Console) の提供
- ・ 効率改善

執筆時点 (2007 年 12 月) での最新レベルは 4.0 (JBoss AS 4.0.2 相当) である。

3) リソースアダプタ (RA) 群

JBOSS-2200 では、さまざまな 2200 リソースにアクセスするためのリソースアダプタが提供されており、ほぼすべての種類のリソースにアクセスすることができる。たとえば TIP コネクタを使用することにより、Java プログラムから TIP/HVTIP トランザクションにアクセスすることが可能である。これらのリソースアダプタ群は米国ユニシス製であるため日本固有のミドルウェアに対するリソースアダプタは提供されていないが、2200 プリミティブ等を用いて独自リソースアダプタを作成することも可能である。

4) 2200 プリミティブ

2200 ファイル I/O や MCB2200 などのプリミティブな API を呼ぶためのライブラリ群が用意されている。JVM2200 の JNI (Java Native Interface) を使用して独自のライブラリ群を作成することも可能である。

2.2 2200 J2EE アプリケーション

JBOSS-2200 を利用することにより、2200 メインフレーム上に J2EE アプリケーションを構築することが可能となる。新規に J2EE アプリケーションを構築することも可能であるが、既存のパッケージなど 2200 メインフレームでは稼働不可能と思われるシステムでも、JBoss AS で稼働する J2EE アプリケーションであれば、容易に移植することができる。また、J2EE アプリケーション同士であれば連携も容易で、外部の J2EE アプリケーションから 2200 メインフレームにアクセスすることも容易となる。

図 2 に 2200 での J2EE アプリケーション構築例を示す。たとえば、Struts などのオープンソースフレームワークで構築された既存の JBoss アプリケーションを 2200 メインフレームに

移植することができる。オープンソースの Web サーバである Apache Web サーバと、mod-jk と呼ばれるコネクタモジュールで連携することも可能である。Entity Bean^{*6} や Session Bean^{*7} などの EJB もそのまま移植可能で、データベースアクセス部分も標準的な SQL 範囲であれば RDMS 用の JDBC ドライバを利用するだけで、容易に 2200 対応が可能となる。2200 メインフレームの TIP/HVTIP トランザクションなどと連携したい場合は、TIP コネクタなどの連携用リソースアダプタを利用し、連携部分のみを新規に作成するだけでよい。

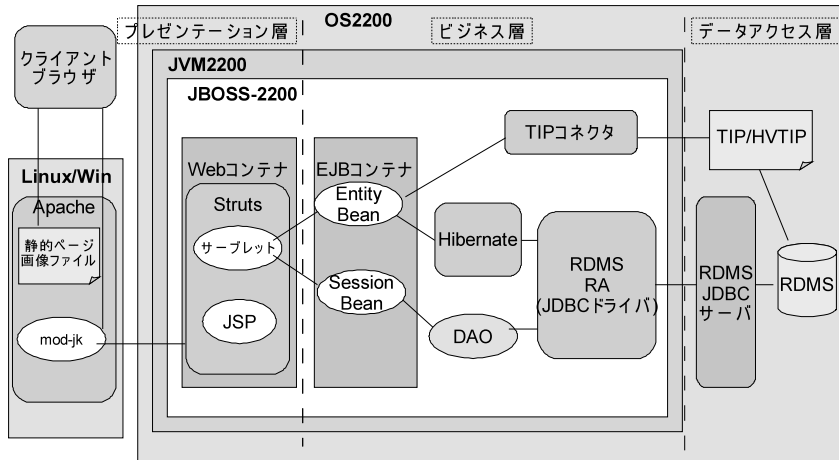


図2 2200 J2EE アプリケーションの例

2.2.1 JBOSS-2200 の配置に関する考察

J2EE システムを構築する際には、プレゼンテーション層、ビジネスロジック層、データアクセス層など、さまざまなレイヤーにアプリケーションを分けるのが一般的であるが、2200 メインフレームで J2EE システムを構築する場合に、どこまでを 2200 メインフレームに配置するかが重要となる。

セキュリティの観点からは、Web コンテナ以降のビジネスロジック層、データアクセス層までをメインフレームに配置するのが望ましい（図3の(1)参照）。Web コンテナ以降をメインフレーム側に置くことで、2200 メインフレームへのアクセスを AJP プロトコル（Tomcat コネクタ mod-jk が使用するプロトコルで通常はポート 8009 を使用する）のみに限定でき、用途の広い HTTP プロトコルアクセスを排除できる。さらに Web コンテナと EJB コンテナに強固な JBOSS-2200 セキュリティを適用することもできる。

また、図3の(2)に示すとおり、処理量の比較的多い Web コンテナ部分を外部に置き、EJB コンテナ以降をメインフレーム側に配置することも考えられるが、Web コンテナと EJB コンテナ間が RMI/IIOP プロトコルを使ったりリモート呼び出しとなり、オブジェクトのシリアライズが発生するため効率に問題が出る場合がある。Web コンテナと EJB コンテナを 2200 メインフレームに配置する場合は Web コンテナと EJB コンテナ間がローカル呼び出しとなり、オブジェクトのシリアライズは行われず効率にも問題はない。Web コンテナの処理ボリューム、EJB コンテナとの密接度、データ量などを考慮し決定することが望ましい。

図3の(3)のようにデータベースのみを 2200 メインフレームに配置し外部から JDBC 経由でアクセスすることも考えられるが、外部から直接データベースにアクセスできることになり

確実なデータ保護が必要である。

また、静的ページや画像ファイルなど Web コンテナや EJB コンテナを必要としないリソースは、Apache Web サーバなどのフロント・サーバに置き、コンテナの負担とネットワーク負荷を軽減するとよい。

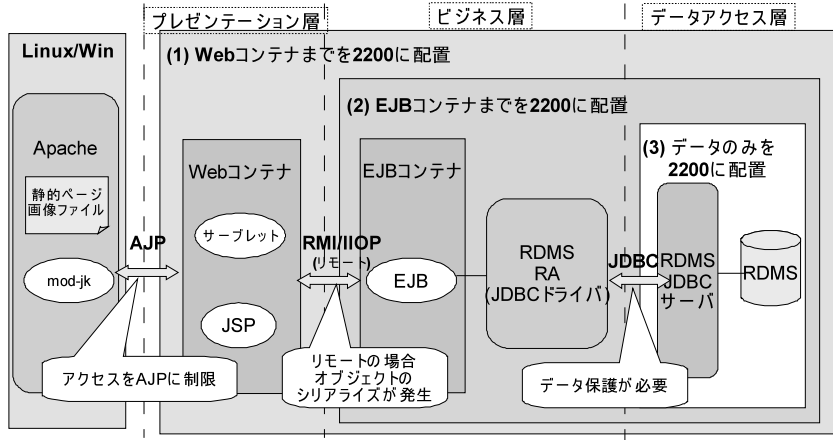


図3 JBOSS-2200の配置

2.3 レガシーデータの活用

2200 メインフレームのデータベースへのアクセス方法について考察する。

2.3.1 RDMS JDBC ドライバ

J2EE アプリケーションから 2200 メインフレーム上の RDMS データベースにアクセスするための JDBC (Java Database Connectivity) ドライバが提供される。RDMS JDBC ドライバは J2EE 環境でのグローバルトランザクション、ローカルトランザクションをサポートする。ただし、複数のリソースマネージャが参加するグローバルトランザクションを利用するには、2200 メインフレームの OLTP 実装である OLTP-TM2200 が必要である。OLTP-TM2200 は分散トランザクション環境におけるトランザクションマネージャの役割を果たし、複数のリソースマネージャを管理しグローバルトランザクションを実現する。

RDMS JDBC ドライバは通常 JBOSS-2200 上で稼働するが (図4の(1)参照)、JBoss-2200 を使用せずに RDMS JDBC ドライバを外部システムから直接使用して 2200 レガシーデータにアクセスすることも可能である (図4の(2)参照)。この場合の利点は、2200 メインフレーム側に Java 環境が不要なことであるが、データの保護に JBOSS-2200 のセキュリティ機能を使用できないため、注意が必要である。

RDMS JDBC ドライバの利用例として、たとえばオープン DB のデータをファイル転送やバッチプログラムで一括で RDMS データベースに反映しているようなシステムの場合、オープン DB へのアクセスと同時に RDMS データを更新することで、リアルタイムでのデータ同期が可能となる。

また、JBoss-2200 に Oracle などのオープン DB の JDBC ドライバを配置し、2200 メインフレームからオープン DB にアクセスすることも容易にできる (図4の(3)参照)。

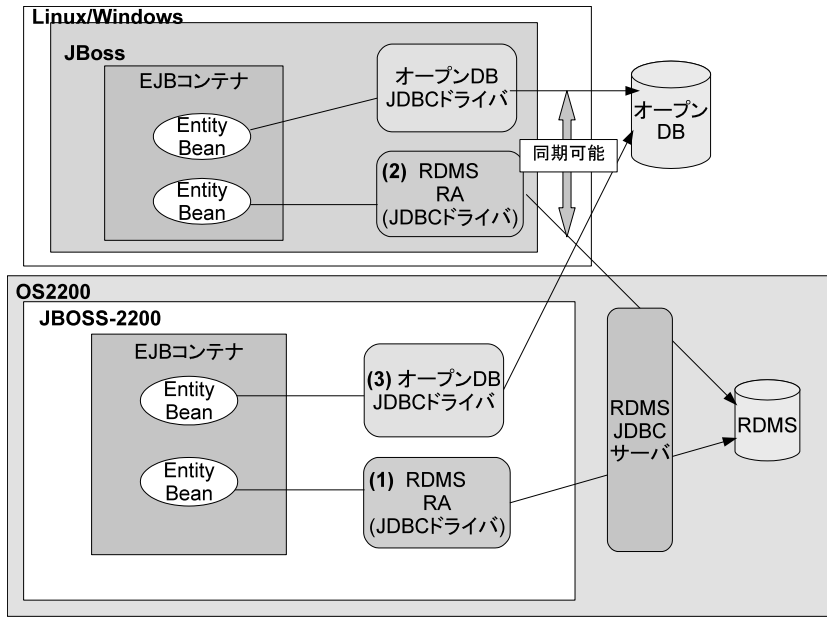


図4 RDMS JDBC ドライバ

2.3.2 Hibernate の活用

Hibernate は、永続データを扱うためのオープンソース Java フレームワークであり、データベースや SQL を意識することなく Java プログラムからデータを扱うことができる。Hibernate を使用することにより、アプリケーションロジックとデータ構造を分離することができ、プログラマはロジック記述に専念できる。

2200 メインフレームでも Hibernate から RDMS JDBC ドライバを介して RDMS データベースにアクセスすることが可能であり、メインフレームの既存データを Java の永続データとして利用できる。Hibernate ではデータベース管理システム (DBMS) 固有の操作や機能を取り扱うために、DBMS 独自の Dialect (方言) が DBMS ベンダーから提供されるが、米国ユニシスからも RDMS 用の Dialect が提供されている (図5 参照)。

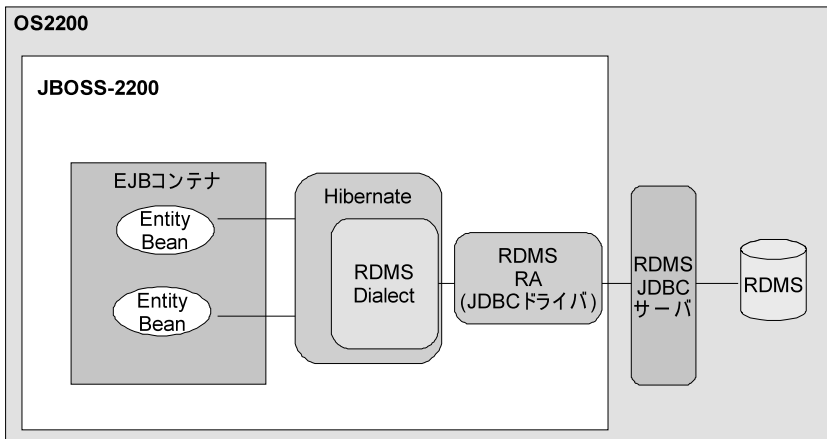


図5 Hibernate RDMS Dialect

2.4 2200 アプリケーションからのオープンシステム連携

外部システムからのレガシーアプリケーション，レガシーデータの利用だけではなく，逆方向つまりレガシーアプリケーションから外部システムへのアクセスも可能である。

外部システムへのアクセスは，2200 メインフレームの TIP アプリケーションやバッチプログラムから，UCS COBOL または UCS C の API で，JBOSS-2200 のソケットリスナー・サービスをコールすることで実現できる。ソケットリスナー・サービスは米国ユニシスから提供される JBOSS-2200 のサービスモジュールで，2200 プログラムから EJB を介して外部システムにアクセスすることができる。2200 プログラムからソケットリスナー・サービスへアクセスするには，2200 メインフレームのソケットインタフェースを提供する COMAPI を利用する。

図6にソケットリスナー・サービスを使った2200アプリケーションから外部J2EEシステムへのアクセス例を示す。

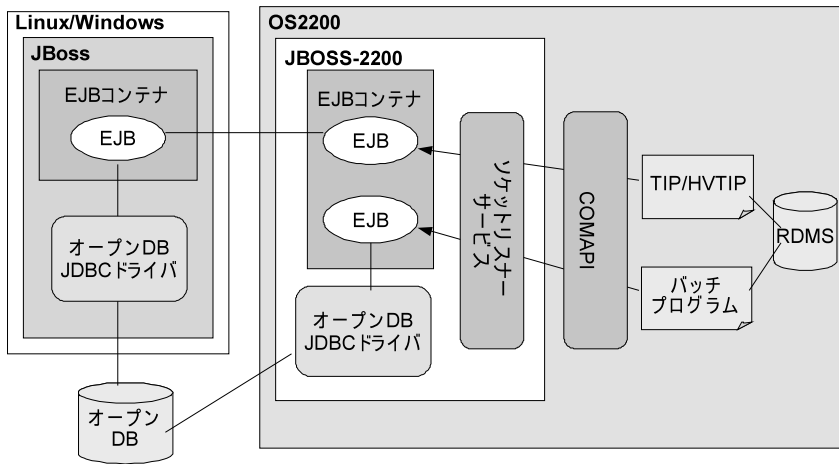


図6 2200 アプリケーションからの外部 J2EE アクセス

3. オープンソースによるメインフレーム開発環境の提案

オープンソースの活用により，2200 アプリケーションの開発がより容易に，より安価に行えるようになった。オープンソース開発環境のデファクトスタンダードである Eclipse には，様々なプラグインが様々な団体から提供されているが，米国ユニシスからも2200アプリケーション開発のためのプラグインが提供されている (Eclipse for OS2200)。2200 Java アプリケーションはもちろん，従来の UCS COBOL や UCS C 言語のアプリケーション開発も Eclipse で行うことができる。

Eclipse for OS2200 には，以下の2200独自機能が実装されている。

1) OS2200 プロジェクト

Eclipse は開発単位をプロジェクトとして管理するが，Eclipse for OS2200 では2200アプリケーション (UCS COBOL, UCS C, Java など) を構築するための OS2200 プロジェクトを提供する。プロジェクト・ウィザードで容易に OS2200 プロジェクトを作成できる。

2) OS2200 パースペクティブ

パースペクティブとは、開発対象とするターゲットやプロジェクトに合わせた画面構成のことで、OS2200 アプリケーションの開発を行うのに最適な画面構成（OS2200 パースペクティブ）が提供される。

3) @TIP アノテーション挿入機能

米国ユニシスが提供するリソースアダプタの一つである TIP コネクタを使用し、2200 TIP/HVTIP トランザクションを呼び出すには、TIP コネクタ用の Java コーディングが必要だが、Java5 のアノテーション機能を用いてプログラムに容易に TIP コネクタのコーディングを埋め込むことができる（@TIP アノテーション）。Eclipse for OS2200 では、ユーザ・プログラムに @TIP アノテーションを埋め込む機能が提供される。

図7に Eclipse for OS2200 の概観を示す。OS2200 パースペクティブで TIP コネクタ・プログラムを作成する例である。

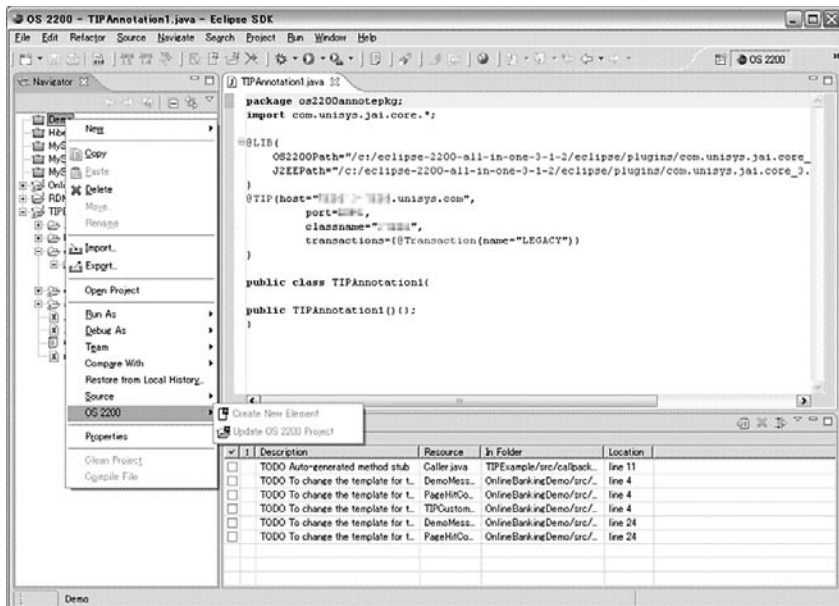


図7 Eclipse for OS2200

4. オープンソース活用の効用

オープンソースによる 2200 メインフレームの既存資産活用は、既存資産の有効利用という側面だけでなく、以下のようなさまざまなメリットがある。

1) 開発・テスト環境のオープン化

2200 アプリケーションを Java 化することにより、開発環境やテスト環境をオープン化することが可能となる。JBoss-2200 は、2200 メインフレーム上の JVM2200 以外の Java 環境でも稼働するため、JBoss-2200 をオープン環境での開発・テストに使用することもできる

し、オープンソースの JBoss AS で 2200 Java アプリケーションをテストすることも可能である。

2) コスト削減

開発環境・テスト環境にオープンソース・ソフトウェアを用いることにより、コスト削減がはかれる。

3) 要員確保が容易

2200 メインフレームの知識を持たない Java プログラマやオープン系データベースエンジニアでも、RDMS JDBC ドライバや Hibernate, TIP コネクタを使用することにより容易に 2200 リソース・アクセスのプログラミングができる。RDMS JDBC ドライバ, Hibernate に関しては 2200 メインフレーム知識はほぼ不要である。TIP コネクタについても、Eclipse for OS2200 を使用して Java プログラムに @TIP アノテーションを記述するだけで TIP/HVTIP 連携が可能となる。テスト作業にもオープンソースを利用すれば、2200 の操作知識も不要である。

4) ソースコードレベルでの障害解析が可能

JBOSS-2200 のベースである JBoss AS や Hibernate, Eclipse などのオープンソース・ソフトウェアはソースコードが公開されており、誰でも自由に閲覧することができるため、ソースコードレベルでの障害解析が可能となる。

5. おわりに

JBoss AS や Hibernate, JDBC ドライバなど、2200 メインフレームとは無縁と思われていたものが、いまや現実に 2200 メインフレームで稼働している。日本ではまだ評価段階であるが、ワールドワイドではすでに事例もいくつかあり、J2EE アプリケーションのプラットフォームとして 2200 メインフレームという選択肢もあることが証明されている。本稿で述べた利用例のほかにも、Web サービス連携, JMS (メッセージング) 連携も提供されており、今後も新技術への追従が行われる予定である。

次期メインフレームでは、Java 専用 IP (複数 CPU のうちのいくつかを Java 専用とし、使用制限や課金を優遇する機能) の構想もあり、効率が問題となる Java プログラムをメインフレームでも高速に実行することが可能となる。

本稿をきっかけにメインフレーム活用を再考していただけると幸いである。

-
- * 1 EJB (Enterprise JavaBeans) : J2EE アーキテクチャの一つで、ビジネスロジックを記述するための仕様。用途に応じて Entity Bean, Session Bean, Message Driven Bean などがある。
 - * 2 JSP (JavaServer Pages) : サーバサイドスクリプティングを実現するテクノロジーで、HTML ファイル中に Java プログラムを記述できる。
 - * 3 JAAS (Java Authentication and Authorization Service) : J2EE アーキテクチャのセキュリティ仕様。
 - * 4 JTS (Java Transaction Service) : J2EE アーキテクチャのトランザクション仕様。
 - * 5 JMS (Java Messaging Service) : J2EE アーキテクチャのメッセージング仕様。

- * 6 Entity Bean : EJB の仕様の一つでデータベースアクセスを実装する.
- * 7 Session Bean : EJB の仕様の一つでユーザセッションを実装する.

執筆者紹介 大塚 玲子 (Reiko Ohtsuka)

1986年日本ユニバック(株)入社. 東京理科大学理学部数学科卒.
2200/1100 メインフレームの基本ソフトウェア部門で米国ユニシ
ス製品の受入・保守・開発を行ってきた. 現在はユニアデックス
(株)ソフトウェアプロダクト統括部 OSS 推進部に所属し, JBoss
を始めとするオープンソースソフトウェアのサポートに従事して
いる.