

Java-CORBA 連携による基幹系システムにおける TCO の削減

6S-07

桐山 典之 齋藤 淳 落合 郁夫

NTT コミュニケーションウェア(株) 技術開発部¹

1. はじめに

顧客管理、料金、OSS、経理、人事、年金システムなどの高性能・高信頼性を必要とする基幹系 OLTP システムは、高速なレスポンスと高信頼性を確保するためにその多くが 2 層クライアント/サーバ方式の構成をとっている。このようなシステム構成では、①ライセンス取得/インストール作業などのクライアントへのインシャルコストの増大、②クライアントへの AP のデリバリ、トラブル対応といったメンテナンスコストの増大、③限られたクライアントによるサーバへの接続、といった問題が存在している。

これらの問題の解消を狙ったのが Web 技術の適用である。一般的に上記の問題 (TCO 削減、イントラネット化) の解決策として、Web サーバ上の CGI や NSAPI といったサーバサイドプログラムをクライアントと基幹系システム間のゲートウェイとすることで、既存の基幹系システムに手を入れることなくイントラネット化する方式がある。しかしこのような Web ゲートウェイ方式では、①HTML の利用により豊富な表現を持つ GUI が犠牲となる、②ゲートウェイが介在することでレスポンスが低下する、といった問題が代償となっている。(図 1)

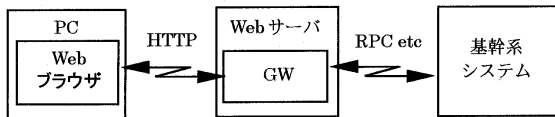


図1 Webゲートウェイ方式

本稿では基幹系システムにおける前述の 2 方式の問題点を同時に解決する一つのソリューションとして、Java と CORBA を用いた Web と基幹系の連携

実現方式 (以下 Web スルー方式) と実システムへの適用課題に対する解決策を報告する。

2. Web スルー方式の基盤技術

Web スルー方式の概念図を図 2 に示す。この方式では Web サーバからクライアント AP と通信モジュールを同時に Web ブラウザ上にダウンロードし、クライアント AP はダウンロードした通信モジュールを利用してクライアントと同一の通信プラットフォームでラッピングされた既存サーバにアクセスする。したがってクライアントは直接サーバとアクセスすることが可能なのでゲートウェイ方式のような処理のオーバーヘッドはほとんどない。またクライアントには Web ブラウザしか必要としないので完全にクライアント AP レスである。

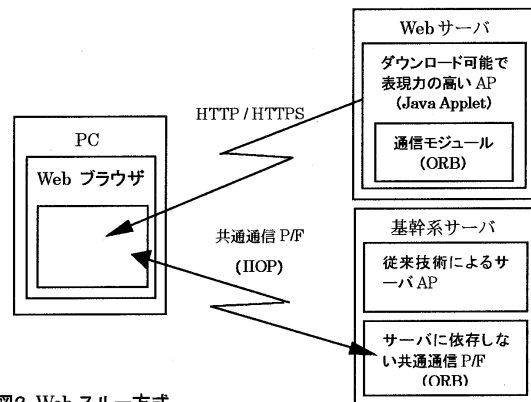


図2 Webスルー方式

クライアント AP に Java、共通通信プラットフォームに CORBA を選定した理由は、①Java Applet を利用することでクライアント AP レスと豊富な GUI の利用が可能、②共通通信プラットフォームとして CORBA の通信基盤 (ORB) を利用することで既存基幹系システムとクライアントの接続が可能 (CORBA の言語非依存性により既存基幹系システ

¹ NTT COMMUNICATIONWARE CORPORATION, 1-6 Nakase Mihama-ku Chiba-shi 261 Japan

ムがラッピング可能なため)、③JavaとCORBAの親和性の良さ、④Javaのクラスライブラリとして提供されているORB製品がありダウンロード可能、といった点が挙げられる。

3. 実運用時の課題と対策

次に基幹システムへ実際に適用する上で主に課題となった点を以下に上げる。

- ①Java Appletの実行性能とダウンロード性能
 - ②IIOPのファイアウォールの通過
 - ③クライアントと基幹システム間の通信暗号化
 - ④既存システムとCORBA間のエラー処理統一
- ①に関しては、JARによる圧縮、JREの事前インストール、Java-Plug-inのディスクキャッシング機能によるJava Appletのキャッシュ、実行速度の速いSwing GUIパッケージの利用等の対策を行った。この結果実行性能は従来の2層方式と同等の性能を維持し、ダウンロード性能も実用に耐えるものとなった。
- ②の課題に対してはIIOP packetsをWebサーバ上でHTTPトンネリングさせる製品を利用することで解決できた。また③に関しては市販のSSLパッケージを利用することでセキュアなIIOP通信を可能とした。最後の④の課題であるが、一つの解決策としては既存システム内部のエラーをCORBAのユーザー定義例外にマッピングしてやり、更に既存システムとObject Serverの間、Client APとスタブの間にCORBAシステム例外とユーザー定義例外を共通に扱えるようにするインターフェースを作成することが考えられる。こうすることによって業務AP開発者からは統一したエラーとして扱えるようになる。

実運用でのWebスルー方式のシステム構成イメージを図3に示す。クライアント(A)は既存2層方式のクライアントと同位置の構成である。この場合、Java Appletダウンロード後の業務サーバとの通信は全てIIOPになる。セキュリティが確保されたセグメント内での通信なので暗号化は必要なく、オーバーヘッドがないため高速な通信が可能である。クライアント(B)は、インターネット/イントラネット経由のアクセスを行う。業務サーバへの通信

はファイアウォールを超えるためにWebサーバ上のHTTPトンネリング機能を利用して行われる。全ての通信はSSLによって暗号化されセキュアな業務処理が可能である。

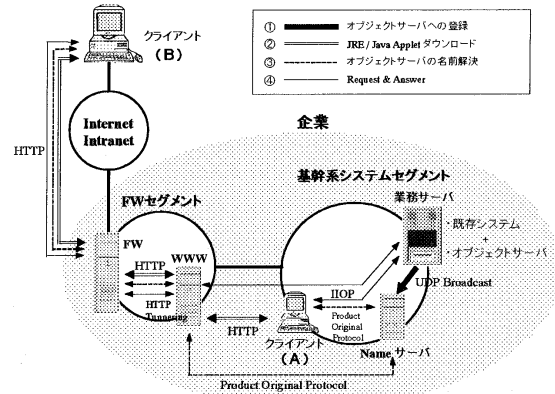


図3 運用時のWebスルー方式

4. まとめ

Webスルー方式は基幹システム2層システムのTCOを削減でき、かつ従来の性能を維持できるソリューションである。今後アプリケーションサーバを取り入れた構成が主流になると予想されるが、多くの既存システムはTCO削減を当面の課題としているため、新規にアプリケーションサーバを導入することは大きな追加投資となる。既存システム資源や開発方式をそのまま利用し、性能を落とすことなくTCOを削減したいと考えているシステム運用者の方にはメリットの大きい方式であると考えられる。

参考文献

- 1) 松山 貴之：“正しいJavaの使い方”，『日経オープンシステム』，Vol. 74，pp. 169-185
- 2) Lincoln C.Stein：“Webセキュリティガイド”，初版，1998，株式会社アスキー
- 3) 松野 良蔵：“Java+CORBA分散オブジェクトシステム構築”，初版，1999，翔泳社
- 4) Michael Morrison & Jerry Ablan：“続・Java言語入門 新しいフレームワークとAPI”，初版，1998，プレントイスホール出版