

J-Smile を支える IT イノベーション (テクノロジー) —大規模基幹システムへの Web 技術適用—

Web Technologies Applied for Large-scale Information System

原田 敬太 HARADA Keita JFE スチール IT 改革推進部 主任部員(部長)
森本 哲也 MORIMOTO Tetsuya JFE スチール IT 改革推進部 主任部員(副部長)
大川 浩志 OHKAWA Hiroshi エクサ 第一事業部 鉄鋼システム第一開発部基盤チーム チームマネージャー

要旨

新統合システムの開発・稼働基盤構築の過程では、開発生産性、システム品質、パフォーマンス、稼働後の環境変化に対する柔軟性など、多くの解決すべき課題があった。これらに対し、グローバルスタンダードの積極的採用、最適な稼働プラットフォームの選定、全面的な Web システム化、java フレームワークの整備、開発フェーズ全般の標準化などにより、当初の課題を解決した。

Abstract:

In the process of platform construction of a new integrated system, many tasks such as the productivity of development, system quality and its flexibility were tackled. These were solved by adopting the global standard, improving java framework, standardizing developmental phases, and selecting the optimum platform.

1. はじめに

JFE スチールの本社業務のほぼ全領域にわたる大規模基幹システムの再構築を短期間に高品質で実現し、かつ今後の環境変化に迅速に対応し得る柔軟性を確保するための、開発・稼働基盤整備に注力した。JFE グループで過去に例がない大規模なシステム再構築に対し、多少チャレンジングな要素もあったが積極的に最新の技術や製品を採用し、プロジェクト全体の目標達成に基盤面からも寄与することができた。

2. 開発・稼働基盤の構築方針

基盤領域ではコンピュータの能力向上を十分活用しつつ、グローバルスタンダードを積極的に採用することで、所期の目標達成を図った。Fig. 1 に基盤領域構築の考え方を示す。

システム全体のリアル処理ニーズ、利用者の利便性向上、および最新技術適用容易性や将来性を考慮し、新統合システムは Web ベースの java アプリケーションとすることを基本とし、3 階層構造のシステムとした。java アプリケーションは従来の COBOL アプリケーションの数倍のコンピュータ資源を必要とするが、必要十分な能力と拡張性を

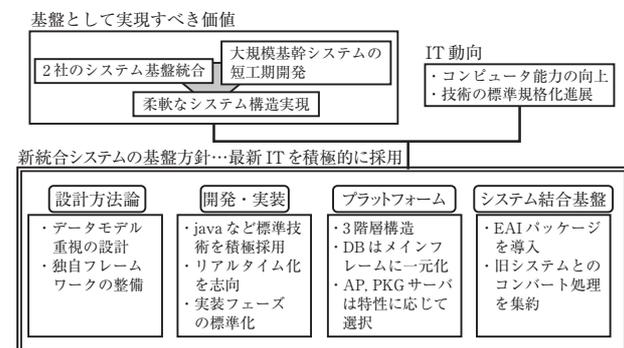


Fig. 1 Revolution of IT infrastructure

を持ったコンピュータを新たに導入して対応した。

設計、開発・実装のほぼ全フェーズの標準化を行うとともに、java 言語での開発生産性、品質向上を目的としたフレームワーク整備を行った。

既存周辺システムとのデータ授受では現行インタフェース仕様を保証する必要があり、新旧コード間の変換処理が

IBM, AIX, WebSphere, MQSeries, AIX, DB2, z/OS, および, zSeries は、IBM Corporation の商標である。
Microsoft, Windows は、Microsoft Corporation の米国およびその他の国における商標である。
UNIX は、The Open Group がライセンスしている米国およびその他の国における登録商標である。
Solaris, Java, および、すべての Java 関連の商標およびロゴは、Sun Microsystems, Inc. の米国およびその他の国における商標または登録商標である。
Curl は住商情報システム株式会社の登録商標または商標である。

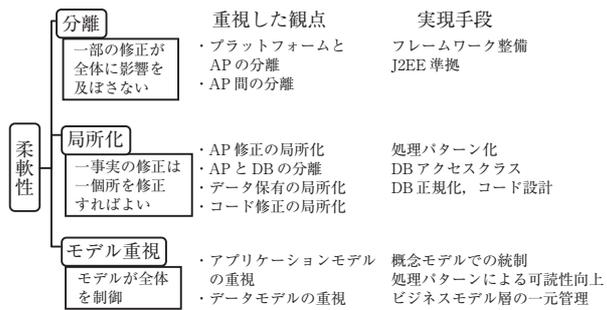


Fig. 2 Policy for flexible structure

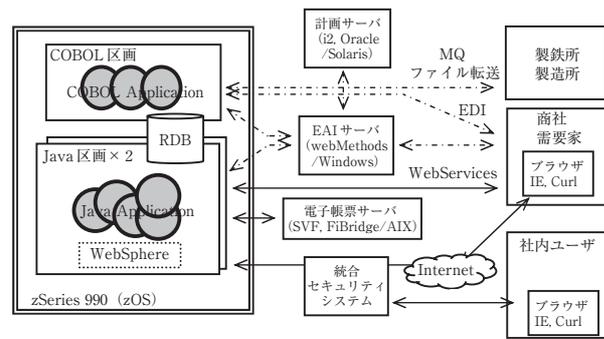


Fig. 3 System structure

必要である。また、システム間接続の切り替えにも柔軟に対応可能とする必要があった。これに対しては、独自開発とせず、enterprise application integration（以下、EAI）パッケージを導入することで対応した。

Fig. 2に示すとおり、システムの柔軟性実現に対しては、特に「適切なモデリングとその具現化」が重要なポイントと認識し、初期の分析・モデリングフェーズの作業に重点を置くとともに、モデルを実装段階で崩さないために、データベース（DB）設計～実装を専任で担当する体制の整備、開発工程の各種標準化などの施策を講じた。また、「適切な分離と局所化」もキーポイントと考え、独自フレームワークの整備に加えて、処理のパターン化、DBアクセスルールの徹底などを行った。

3. 採用技術の紹介

3.1 ハードウェア、基本ソフトウェアアーキテクチャ

新統合システムの大半はWebベースであり、java言語で開発した。稼働プラットフォームは、プレゼンテーション層、アプリケーション層、データベース層からなる3階層構造を採用している。

データベースは、パフォーマンス面、安定性、将来的な継続性の点で優位であることから、メインフレームコンピュータ上に一元的に保持している。javaアプリケーションを稼働させるアプリケーションサーバ（WebSphere）は、規模やシステム特性を考慮し、販流系システムはメインフレームで、経管系システムはUNIXサーバで運用している。システム構成をFig. 3に示す。

販流系システムの主要機能を稼働させているメインフレームコンピュータ（zSeries 990）は、javaアプリケーション用2区画、COBOLアプリケーション用1区画の計3区画からなるクラスタ構成とし、可用性向上を図った。COBOLアプリケーション用区画は、対製鉄所、対商社など他システムとのデータ授受機能も担っている。

後述のアプリケーションフレームワークをプラットフォームへの依存性が極小となるよう整備し、開発・テストはパソコン（Windows）、サーバ（AIX）で行い、稼働は

メインフレーム（z/OS）およびサーバ（AIX）とするなど、javaアプリケーションのハードウェアプラットフォームからの独立性を実現できている。

計画系、経理系システムはパッケージ製品を導入しており、これらはそれぞれ独立したUNIXサーバにて運用している。また、電子帳票サーバ、EAIサーバなどの周辺系もUNIX、Windowsサーバとするなど、それぞれの特性に応じて適材適所の稼働プラットフォームを選定した。

多データ項目を扱うことに起因するJVM（Java仮想マシン）のメモリ不足の対応には苦慮したが、必要以上にメモリを消費しているアプリケーションをチューニングするとともに、各機能の特性や同時利用者数などを考慮してJVMの分割、アプリケーション配置を調整した。この結果、販売・生産・物流系のシステムは、最終的にはリアル処理用JVM31×2個（クラスタ構成）、バッチ処理用JVM18個の実行環境としている。

3.2 Webシステムのクライアント

JFEグループ内外合わせた利用者は4000名以上の規模であることから、新統合システムは基本的にはWebブラウザのみで利用でき、クライアントパソコンへのプログラム配信は不要な方式とした。ただし、詳細な鋼材仕様を出力する複雑なユーザインタフェース（仕様設定業務）は、HTMLやjavascriptでの実現は困難であったため、リッチインターネットクライアントの実現手段としてCurlを採用した。Fig. 4はCurlで開発した画面例である。

Curl採用は複数のリッチクライアント実現手段候補から、要件の実現性、開発・保守生産性、稼働環境からの独立性



Fig. 4 Example of the screen developed by Curl

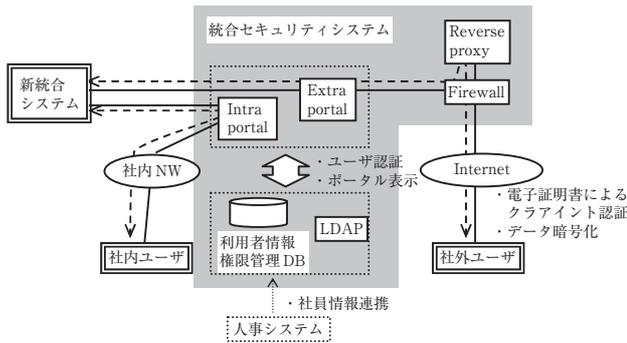


Fig. 5 Outline of integrated security system

(アプリケーションサーバに非依存であること)、アプリケーションフレームワークとの親和性などの観点から評価し総合的に判断した。クライアントPCにはCurl実行環境をインストールする必要があるが、Curlアプレット（プログラム）は自動的にダウンロード、実行されるので配布の仕組みは不要である。

ユーザ管理については、JFEグループ共通の基盤として構築済みであった、統合セキュリティシステムを利用することで、ユーザID/パスワードの一元管理、共通ポータルメニューからのシステム利用、シングルサインオン、JFEスチール外から接続時の電子証明書によるクライアント認証、データ暗号化を実現している。Fig. 5は統合セキュリティシステムの概要である。

3.3 独自フレームワークの構築

javaアプリケーションフレームワークは、高品質なシステムを高生産性で開発できるよう、struts（オープンソースのWebアプリケーションフレームワーク）をベースに独自開発にて構築した。複数の市販フレームワーク製品の比較評価も行ったが、データモデルの忠実な実装、鉄鋼業務への親和性、開発・保守生産性などの観点から独自開発が適切と判断した。

Fig. 6にフレームワークの基本構造を示す。既存システムや業務内容の分析などを通じ、新統合システム構築のベースとなる基本パターン（30の処理パターン、12の画面パターン）と49個の共通機能部品を事前に準備した。アプ

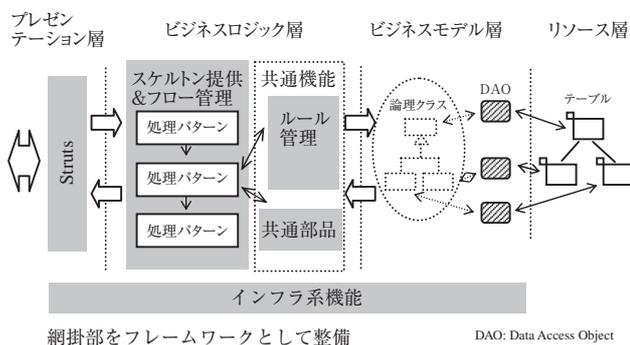


Fig. 6 Structure of application framework

リケーションの設計・開発者は、パターンカタログから適切なものを選択し、この枠組みの中でプログラムを組み立て、個別/固有処理部分を穴埋め式で記述することで、アプリケーションを開発できる。これにより統一性を持ったアプリケーション構造になることから、稼働後においても高い保守性、柔軟性を実現できている。

画面パターンはアプリケーション全体で共通のと想定される画面要素（業務メニュー画面、検索&一覧画面、単票画面、帳票出力画面など）を抽出し、雛型のプログラムソースコードを準備したものである。処理パターンは、ビジネスロジックのうち共通的に扱えると想定される処理（チェック処理、項目編集処理など）を抽出しパターン化したものである。パターン一覧は処理パターンカタログとして整備し、設計書雛型、ソースコードテンプレートも合わせて提供することで、開発生産性向上を図った。

その他、処理パターンの中から呼び出されて使用される、カレンダー機能、採番機能などの共通部品系、ルール管理などのテンプレート系の共通機能部品を整備している。また、ログ、ダンプ、トレース出力機能など、アプリケーションが直接意識しないインフラ系共通機能も組み込み、テスト・チューニング作業の効率化を図った。

データベースアクセスはdata access object（以下、DAO）と呼ぶ共通部品のみを介して行うことを大原則として徹底し、個々のアプリケーションが直接SQLを発行することは認めていない。DAOはデータベースの実装を専任で担当するメンバが管理しており、全体整合性や品質を保証している。

今回構築したフレームワークは、バッチ処理をjavaで実現するための仕組みも実現した。これにより、オンライン/バッチ処理間で処理ロジックが共用できるため、開発量抑止、稼働後のメンテナンス負荷軽減に寄与している。

3.4 システム間接続へのEAI導入

本社側システムは新システムの稼働にともない、各種コードを新体系で運用するが、製鉄所をはじめとする既存他システムには旧コードで運用しているものが多い。このため、システム間のデータ授受に際し、新旧コード間の変換/逆変換が必須であった。また、品種・機能ごとに段階的に実施するシステム移行時や今後の既存システム刷新時の切り替えを円滑に行えるようにする必要もあった。新統合システムでは、Fig. 7に示すように、EAIツールwebMethodsを導入し、変換ロジックをアプリケーションから分離するとともに、システム間の接続箇所を集中させることで、この課題に対応した。システム切り替え時には、一時的に大量のデータコンバート、DBセットアップ作業が発生するが、このコンバート処理もEAIサーバにて実行した。

コンバート処理ロジックの部品化と再利用により、高い

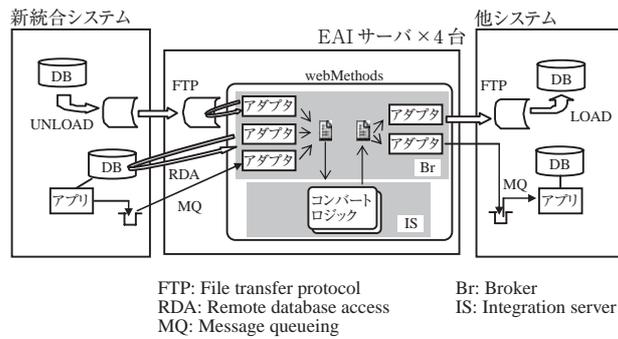


Fig.7 EAI System

開発生産性と品質の向上,均一化を実現するとともに,部品単位,部品間のテストも効率的に実施することができた。複数の開発単位(たとえば,品種ごと)の段階的本番化や,結合テスト,運用テスト,本番の複数環境にまたがる検証,切り替え,データ移行に対しては,複数のPCサーバにEAIを配置し,アダプタ(接続部品)を用いた接続環境を整備することで,同時複数接続,短時間での切り替えを実現した。

4. 評価

新統合システムは従来のシステム構築手法と異なる部分が多く,さまざまな局面で新たな標準化とその徹底が必要であった。このような背景から基盤分野の検討,環境整備は業務設計やアプリケーション設計に先行して実施し,開発チームをリードする形となった。初期フェーズで実施したデータモデルの構造が崩れないように維持しつつ,柔軟構造を実現するため,基盤チームによる全体統制を徹底し,標準から外れた構造のアプリケーションの排除に努めたことで,プロジェクト全体の目標達成に大きく貢献できた。

フレームワーク,共通部品を整備したことにより,アプリケーション構造の均一化,処理ロジックの粒度統一,プログラムの可読性向上,品質の均一化・向上に貢献できたと評価している。このアプリケーションのコンポーネント化をベースとして一部機能のWebサービス化を実現するなど,今後のservice oriented architecture (SOA)への発展につながる試みも一定の成果を得た。また,基礎となるデータ構造,データベースの構築を専任チームが担う体制としたことで,一貫性を保ったシステム構造を実現することができた。

今回,メインフレーム上で販流系javaアプリケーションを稼働させたことは本システムの大きな特徴である。種々のソフトウェアは最新バージョンを採用したこともあり,テスト当初はDB,アプリケーションサーバなどのベンダ提供製品のバグに遭遇することも少なくなかったが,従

来のメインフレーム製品同様,比較的早期に原因究明,バグフィックスが提供された。また絶対的な能力の高さ,堅牢性はメインフレームならではの安定感があるなど,基幹システムのプラットフォームとして妥当な選択であったと考えている。

新統合システム全体を見ると,メインフレーム,分散系サーバ,複数ベンダのソフト製品,ネットワーク,proxyサーバなどの経路上の構成要素,PC環境のバリエーションなど,動作環境の組み合わせは膨大である。このような複合環境で発生する障害について,問題箇所の切り分け,原因究明,対策を行うには,各分野に精通した多くの専門システムエンジニアが必要であったが,それを機能させるための体制,指揮命令系統も有効に働いた。

javaアプリケーションの稼働に必要なコンピュータ資源のキャパシティプランニングも大きな課題であった。早期にプロトタイプを開発し,COBOLで開発した従来型アプリケーションとjavaアプリケーションでCPU資源の使用量比較を行い,最終的な必要資源量を約3000MIPSと見積もった。これに対し実績は10%増程度の範囲に収まっており,問題ない精度であったと評価できる。

5. おわりに

開発生産性向上,高品質なシステム構築,稼働後の環境変化への柔軟な対応,さらには継続的な業務改革という命題に対し,最新のITを適切に活用した新統合システムの開発・稼働基盤構築の取り組みについて述べた。

JFEへの経営統合前の準備段階から,約3年間の基盤整備活動は一旦完了したが,今後のビジネス環境およびIT動向の変化に対し,システム稼働基盤も適切な対応を継続的に行っていくことが重要と認識している。また,今回得られた成果物やノウハウをJFEグループ各社のシステム化に展開し,グループ全体のIT水準の向上を図ってきたい。



原田 敬太



森本 哲也



大川 浩志