

CALS対応文書管理システムの拡張と将来性

佐藤昌志 神山典子

株式会社 東芝

4M-4

1. はじめに

近年、インターネットを中心とする情報技術の急激な進歩により、製品のライフサイクル全体をシステム化するというCALSのコンセプトが徐々に実装可能になってきている。前回[1]は、段階的なCALS実装の第1歩として、既存の文書管理システムにWWWブラウザからの文書閲覧機能を、将来の拡張に柔軟に対応できる3層アーキテクチャにより実現した。また、閲覧するデータのフォーマットとしてCALS標準のTiled-G4形式を扱えるようにするため、JavaのアプレットとしてTiled-G4ビューアを開発を行った。

CALS実装の第2段階としては、文書管理システムをベースとして、既存システムとの連携によるシステム統合と将来の拡張に向けてのインフラ技術の評価・検証を行った。

本稿では、実装第2段階として、文書のライフサイクル全体をシステム化するために、既存の文書承認ワークフローシステムとの連携による承認文書の自動登録を実現したことと、複数システム統合によるインフラの複雑化解消と将来の拡張性への対応としてJavaのRMIを利用したオブジェクト通信基盤を構築したことについて述べる。

2. システム概要

本システムは、前回[1]構築したシステムに既存ワークフローシステムを統合し、既存構成管理システムのインフラとしてJavaのRMIを利用したものである。システム構成を図1に示す。図は3段に分かれており、下から、プレゼンテーション層、ファンクション層、データ層の3層アーキテクチャとなっている。

Document Management System for CALS,  
its extension and capability  
Masashi Sato, Noriko Kamiyama  
TOSHIBA CORPORATION

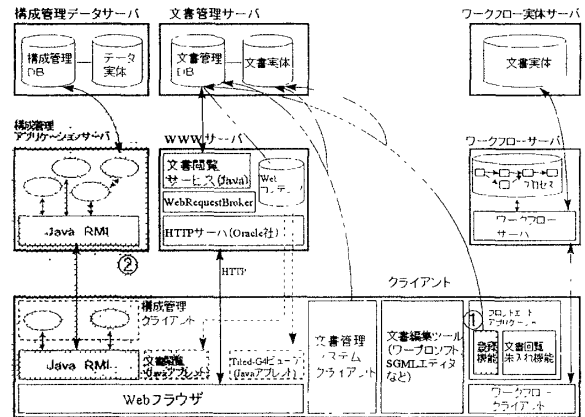


図1 システム概要

今回開発した部分 (図1の網掛け部分) について以下に説明する。

①ワークフローシステムとの連携

CALSシステムは、決してゼロから構築しなければならないものではなく、既存のハードウェア、ソフトウェアを有効利用することにより実装されるものである。ここでは、従来別のシステムであった文書承認ワークフローシステムをCALS対応文書管理システムに取り込んだ。取り込む方式としては、ワークフロークライアントを介して、ワークフローサーバでの管理情報を文書管理システムの管理情報に変換し、実体とともに文書管理システムへ登録する機能を追加することによるクライアント側での容易な連携とした。

②Javaテクノロジーの利用

CALSシステムでは外部との連携が非常に重要となる。製品のライフサイクルは1つの企業の中で閉じているとは限らず、今後はEDIを利用した調達業務など外部との連携が必要な場合が多くなっていくと思われる。その場合、データフォーマットの標準化はもちろん、アプリケーションを含めたデータの共有が必要となる。

本システムでは、このような将来の外部接続性を考慮して、基盤技術としてCORBAを想定したアプリケーション設計を始めた。具体的には、

既存の構成管理アプリケーションをオブジェクト指向技術を利用して再設計し、Javaによる実装を行った。今回は、市場のORB製品がCORBAに完全準拠している段階ではないことを考慮し、Javaテクノロジーの追求を優先しJavaのRMIを利用した。

また、前回[1]Oracle7サーバ上のストアプロセスとして開発した文書閲覧サービスを、Oracle WebServerのユーティリティを利用することによりJavaでラッピングし、クライアント側のJavaからの利用を可能にした。

### 3. CALS実装の課題とポイント

今回のCALS対応文書管理システムを実装するに当たっては、既存資産の利用、オブジェクト指向標準技術の利用といった部分に重点をおいた。

まず、既存資産の利用における課題としては、それぞれ独立して構築されたシステム間のデータや機能の重複の回避が挙げられる。図1のワークフローシステムとの連携においてもシステムを統合した場合、管理情報や実体が重複する。そこで、データを静的、動的の2種類に分類し、ワークフローで扱う実体データは一時的、動的なデータとし、文書管理システムで扱うデータは静的・永続的管理対象のデータとして扱うことにより、機能とデータの役割を切り分けた。

次に、オブジェクト指向標準技術の利用は段階的CALSシステムの実装において、非常に重要な位置を占める。この技術により、イントラネット技術を利用して社内を中心にシステム化を進めると同時に外部との連携の受け口を作ることが可能になる。今回はJavaのRMIを利用したが、将来的にはCORBAを中心としたOMGの標準に従い、外部との接続性を強化していく。

インフラとしてCORBAを利用する場合、既存のインフラ上のアプリケーションの移行が必要になるが、その場合以下の3つの方法が考えられる

- ①オブジェクト指向分析・設計による再設計
- ②ORBへのゲートウェイの構築
- ③ORBに対応したラッピング

また、外部との接続を考えた場合、ファイアウォールや暗号技術を用いたセキュリティ技術の実装も必須となる。

以上、セキュリティ技術の実装も必須となる。

以上の検討事項を踏まえて、今後CALSシステムの構想を図2のように作成した。

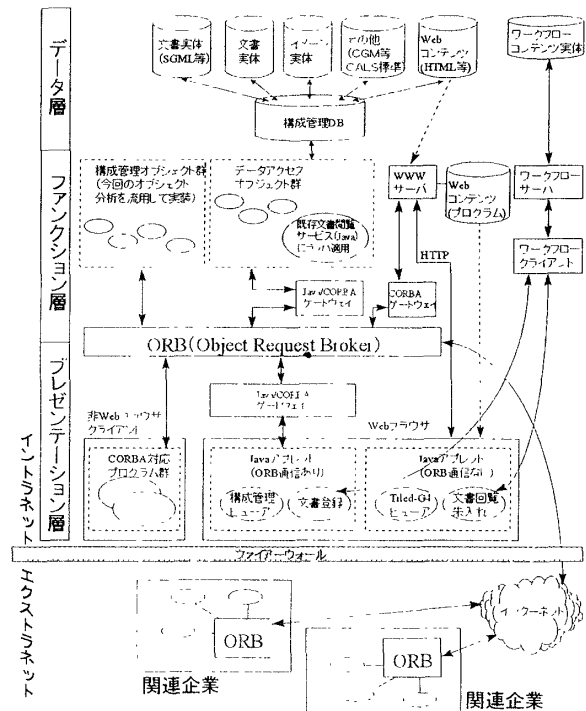


図2 CALSシステム将来構想

### 4. まとめ

CALSという大規模なシステムリエンジニアリングを行うには、既存資産を利用すること、組み合わせた資産の整合性を保つこと、また生み出す資産を将来に活かすことが重要なポイントとなる。今後も標準技術を取り入れ、CALSシステムの段階的、外向的な実装を行っていく。

### 参考文献

- [1] 神山・佐藤、「CALS対応文書管理システム」、情報処理学会第54回全国大会 p.4-371
- [2] 成田・保西ほか、「CORBAとJava分散オブジェクト技術 p.55-126、ソフト・リサーチ・センター

※Javaおよび他のJavaを含む標準は、米国Sun Microsystems, Inc.の商標であり、同社のJavaブランドの技術を使用した製品を指します。  
 ※ORACLEはOracle Corporationの登録商標です。  
 ※本論文に掲載の商品の名称は、それぞれ各社が商標として使用している場合があります。