

XMLを用いた 簡易ビジネス情報ライブラリの検討

数野 顕 高林 貴美 石井 宏次 合田 信久 大野 邦夫

INSエンジニアリング株式会社

本システムは、ウェブブラウザをクライアントとし、RDBをコアとした3層のクライアント・サーバシステムで、個人管理、ドキュメント管理、会議議事録管理など、中小の企業や組織における汎用的かつ拡張可能なオフィス・システムへの適用を狙いとしている。そのために、XMLを高度な機能を実現するためではなく、既存のシステムとの相互運用と、利用者に対するシステム導入、利用・運用などの容易化という方向に絞って適用した。また、その実現手法としては、極力、既存のカタログ製品の組み合わせにより実現可能とすることを試みたものである。

A Study on The XML Based Easy Business Information Libraries

Akira KAZUNO, Takami TAKABAYASHI, Kouji ISHII, Nobuhisa GOUDA,
and Kunio OHNO

INS Engineering Corporation

4-31-18, Nishi-Gotanda, Shinagawa-ku, Tokyo, 141-0031, Japan

This paper describes an XML based extensible, customizable, and simplified office system named "Easy Business Information Libraries" which supports personal information management, document management, and meeting report management for small businesses. The architecture of the system is based on three-tier client/server system with web-browser, web-server, and RDB. XML was not applied for the advanced technology to realize high level services, but focused on the realization of inter-operability to the legacy systems, and easiness for users to introduce, operate, and maintain the system. The system implementation is also simplified by customizing the existing component products organization.

1. まえがき

多くの企業では、社内システム（イントラネット）の構築に際し、以前の紙による稟議や承認・決済の代わりにグループウェアシステムや専用のワークフローシステムを導入し、文書の決済や承認を行いつつあるが、必ずしもスムーズに運用されていない。日本の企業組織のビジネスプロセスは必ずしも明確なルールに基づいて管理運営されているわけではなく、厳格な管理を要求するワークフロー管理システムは、日本の組織に適合しない場合が多いためである[1]。そのため、ワークフロー管理システムは導入したものの、実際の運用に当たっては当初想定した運用からは考えられなかった例外事項が大量に発生し、その対処のためのプログラムの修正、運用ルールの変更などに煩わされる場合が少なくない。すなわち、本格的なグループウェアシステムやデータベースシステムに依存した社内OAシステムは、文書の一元管理・再利用、情報の共有などを要求する。そのため画面や帳票、データベース項目の変更が頻繁に生じると、その変更作業は容易ではなく、一般にはシステム構築者の手を煩わせる事態となるのである。

今後日本の企業もグローバル化に対処するためには、上記のような欧米流の厳格なワークフロー管理を取り入れてゆく必要があるであろうが、中小規模の企業や組織にとっては、日本の実情に適した簡易なソリューションが必要ではないかとも思われる。例えば、従来の紙による稟議や文書の承認・決済をワークフロー管理システムを用いずに単純なRDBを用いるクライアント・サーバ的なアーキテクチャで支援するような考え方である。特に専任のシステム管理者を置くことが困難な中小企業などにおいては、Webサーバを立ち上げてイントラネット化は計りたいものの、ワークフロー管理までは実施できない企業が大半であろうと思われる。本システムはそのような企業や組織に適合する情報管理システムの試みである。

2. 本システムのねらい

企業組織における情報管理は、部門によって異なるが、それでも多くの場合は、

- ①社員、顧客、関連企業や組織などの人々の名簿の管理、
- ②社員の勤務表やスケジュール表の管理、
- ③営業折衝記録、会議の議事録などの企業文書

④一般的な各種ドキュメント管理

などであろう。本システムは、以上のような一般的なオフィス業務を支援する文書および情報を作成、管理、配信する業務を幅広く支援することを目的とする。

一昔前であれば、このようなニーズは、PCにおけるオフィス・スイート製品の典型的な市場であった。文字図形画像編集機能を包含したワープロ、スプレッドシート、ローカルな簡易RDBなどがあれば、名簿の管理、スケジュール管理、ドキュメント管理はPCのローカル機能のみで可能であった。なお、これはローカルなPCによるシステムであったため、ワークフローは従来の紙によらざるを得なかった。そのためPCのプリンタと複写機から大量の紙の資料が生産され、オフィスは膨大な紙の資料に支配されるに至ったのである。

その使われ方に変化が現われたのは、インターネットと、その企業内バージョンであるイントラネットが出現、普及した後である。インターネット、イントラネットの普及により、本格的なネットワーク活用が実現された。すなわち、Eメールにより、個人レベルの低価格なメッセージ通信が実現された。さらにウェブブラウザにより、情報の配信・公開が極めて容易になった。この2種類の道具立てで、オフィス業務は大きなインパクトを受けるに至った。

イントラネットが普及した後、名簿の管理、スケジュール管理、ドキュメント管理のような用途の製品としては、クライアント・サーバ化されたオフィス・スイート製品のウェブ対応版が挙げられるであろう。しかし、オフィス・スイート製品の場合は、特定のベンダーの製品に縛られてしまうため、幅広いベンダー製品の相互運用には適合できないという欠点があった。また、技術進歩の早いパーソナルコンピュータ分野においては、オープンでニュートラルなシステム構成が本質的に望まれるようになりつつある。本システムは、以上の問題に対しXMLをニュートラルなデータ形式として活用することにより解決し、さらにオープンな拡張性の実現を狙うものである。

3. システム構成

3.1 基本構成と処理の流れ

本システムの基本構成を図1に示す。本システムは、ウェブ・サーバをフロントエンドとし、バック

ヤードにRDBと文書ファイルシステムを擁する、3層のクライアント・サーバ・システムである。クライアントシステムとしては、ウェブブラウザを適用す

るが、Eメールによるモバイル端末からのアクセスも可能としている。ミドルウェア層の基本的役割は、データ変換である。

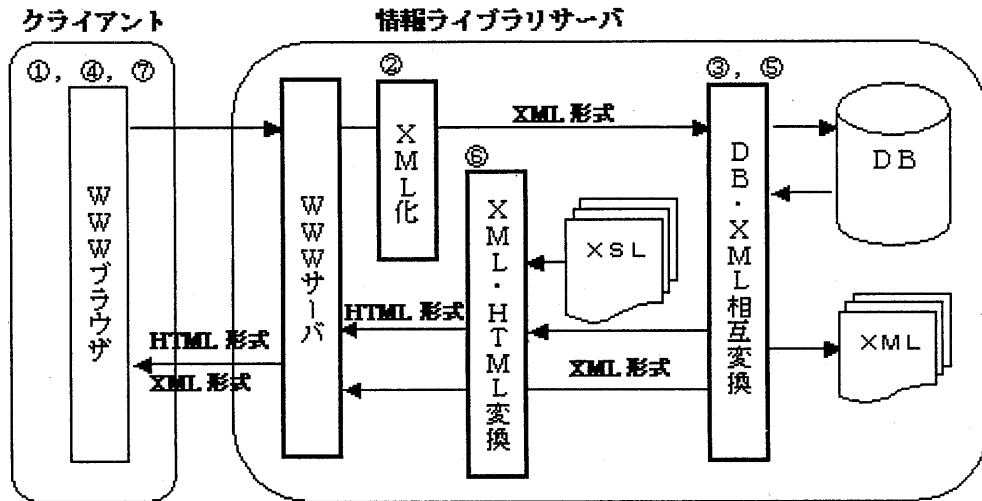


図1 本システムの基本構成

クライアントからの入力データは、ウェブ・サーバ（CGI）経由で入力され、ミドルウェアでXML化される。XMLインスタンスがデータの場合は、iConnectorを通じてOracle RDBに登録される。文書の場合はそのメタデータのみがXML化され、データと同一のプロセスを経てRDBに登録され、文書実体は、当該文書アプリケーションのファイルとして管理される。

データ参照の場合は、以上の流れとは逆にデータはデータベースまたはXML文書ファイルから、XML/HTML変換を通じてブラウザ側へ渡される。モバイル端末系とのインターフェースには、インフォテリア社のiMessengerを用い、ウェブブラウザの代わりにEメールによるサービスも可能としている。

3.2 ユーザインタフェース

ウェブブラウザを用いる一般ユーザへのインタフェースは、下記のとおりである。一般のオフィス作業者が特別な訓練などをせずに利用できるように極力単純な構成にしている。

3.2.1 データ入力

ウェブブラウザを利用するユーザからの入力データは、下記の①から③のようなステップでブラウザからデータベースへ格納される。

- ①入力テンプレートを用い、ウェブブラウザよりHTTP経由で、データをウェブサーバへ送信する。
- ②送信されたデータを”XML化モジュール”（Perlによるミドルウェア）を用いXMLデータに変換する。
- ③変換されたXMLデータを、”DB・XML相互変換モジュール”（iConnector for Oracle）経由でデータベースに格納する。

3.2.2 データ更新、削除

データ入力の場合と基本的には同一である。処理要求メニュー（追加・更新・削除）に対応した形式で、XMLデータをデータベースへ変換し書き込んで格納する。

3.2.3 データ参照

この場合は、④から⑥のステップを経てデータが、データベース側からウェブブラウザへと移動する。

- ④ウェブブラウザよりHTTP経由で、検索条件をウェブサーバへ送信する。
- ⑤送信された検索条件に基づき、データベースから抽出したデータをXMLデータに変換する。
- ⑥ウェブブラウザの種類・バージョンと処理要求に合わせてXSLを選択し、XMLデータをHTMLデータに変換する。
- ⑦XMLデータまたは、HTMLデータをウェブサーバを介してウェブブラウザへ送信し、送信されたデータをウェブブラウザ上に必要な形式で表示する。

なお、クライアント端末のウェブブラウザの種類により表示処理に若干相違がある。すなわち、マイクロソフトのIE5 (Internet Explorer - 5) の場合はXMLデータとXSL両者がウェブサーバへと渡されるのに対し、他のブラウザではXMLから変換されたHTMLデータのみが渡される。

3.3 動作環境とハードウェア・ソフトウェア構成

本システムの動作環境とハードウェア・ソフトウェア構成は以下のとおりである。

- (1)プラットフォーム：PC/AT互換機
- (2)OS：WindowsNT4.0ServerServicePack3以降
- (3)データベース：Oracle R8.0.5
- (4)ウェブサーバ：IIS3.0以降
- (5)ウェブブラウザ：Internet Explorer 5
- (6)開発言語：Perl for Win32、JavaScript、HTML
- (7)ミドルウェア：iConnector for Oracle、iMessenger

本システムは、通常、情報ライブラリサーバと複数のクライアント（ウェブブラウザ）により運用される。サーバの負荷分散を計りたいければ、ウェブサーバとデータベース・サーバを異なるハードウェアで構成することも可能である。

4. 情報ライブラリサーバの機能

現在、本システムは、顧客管理機能、営業情報管理機能、ドキュメント情報管理機能、システム利用者管理機能を提供している。今後の利用状況により、勤務管理、スケジュール管理などの機能を実装することも可能である。

4.1 顧客管理機能

顧客管理機能に関しては、社員名簿や、知人名簿としても使えるように配慮している。従って、通常名刺に記載されている、“氏名”、“所属”、“役職”、

“企業名”、“企業住所”、“電話”、“FAX”、“Email”といった一般的項目のほか、社員用の“採用年次”、“専門分野”、“業務経歴”、“性別”、“年齢”、“自宅住所”、“自宅電話”などの項目を設けている。さらに、顧客用の項目として、年賀状や暑中お見舞いなどの送付に関する項目が定義されている。さらに、名簿管理の一般的な項目として、“個人ID”、“氏名カナ”、“登録日”、“旧ID”、“変更理由”、“備考”といった項目が用意されている。このサービスの画面例を図2に示す。このように、ウェブブラウザから容易に入力することが可能である。

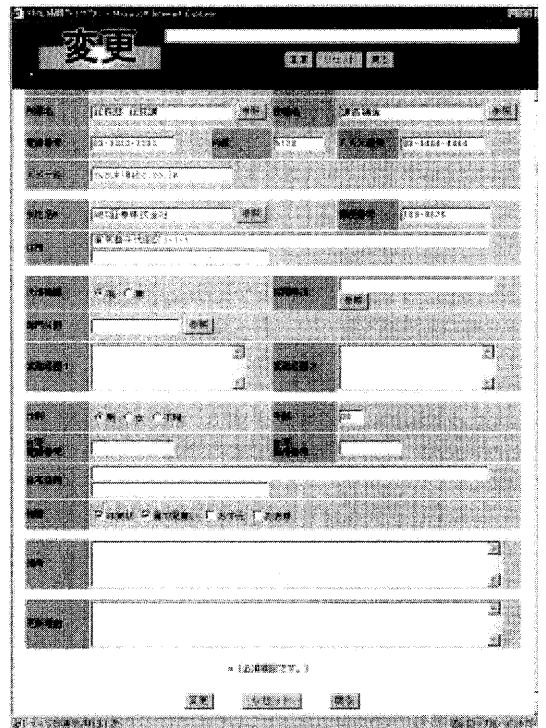


図2 顧客情報管理画面

以上の情報は、RDBへの格納に便利のようにフラットな構造で定義されている。フラットであれば何もXMLで定義しなくても良いのではないかといった議論もあるが、XMLの採否、情報構造の平滑化と階層化に関しては、システムのポリシーに依存する。今回のシステムが、中小の組織における簡易なシステムをねらい、かつ各種システムとの相互運用を可能にすることをポリシーとしたことから導かれ

た結果である。この件に関しては、さらに後の”考察”の項で検討する。

4.2 ドキュメント情報管理機能

ドキュメント情報管理機能では、ドキュメントのメタ情報をRDBで管理する。従って、ドキュメント実体はファイル名として管理されるだけである。

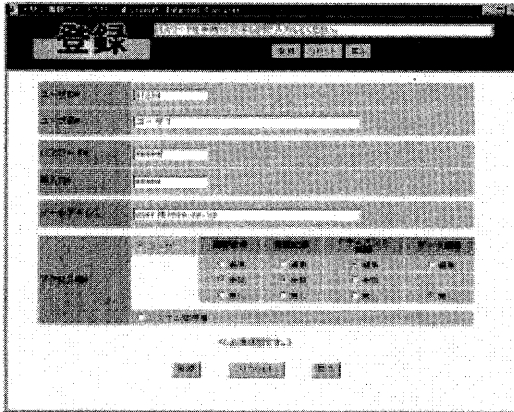


図3 ドキュメント管理機能画面の例

メタ情報としては、“ドキュメントID”、“作成者”、“作成日”、“ドキュメント名”、“概要”、“ファイル名”、“アクセスパス名”、“旧ドキュメントID”といった情報が項目として定義されている。

以上の項目に基づき、各種ドキュメント情報の登録、変更、削除、参照の他、履歴管理機能を実現している。登録画面の例を図3に示す。

登録されたドキュメントの一括管理が可能である。さらにウェブブラウザによりダウンロードしたドキュメントを新たに登録する機能や、その再利用、登録・変更されたドキュメント情報の世代管理（複数世代に渡る履歴管理）も実現されている。そのため、“改訂情報”、“改訂理由”、“改訂日時”、“改訂者名”、“改訂者ID”といった項目が追加されている。

文書をファイル名で管理するのは、極めて容易であるが、それらのドキュメントを参照、更新するためには、対応するアプリケーションが必要となる。この問題についても後に考察する。

4.3 営業記録管理機能

企業文書の代表的な存在である営業記録や会議議事録の管理機能を提供する。これらの文書は、紙の場合は稟議書のような形式で、関連部門の責任者に回覧・承認されるものである。従ってこの機能は、本来であればグループウェアやワークフロー管理システムを用いて実現されるべき機能である。

従ってここでは、営業記録や会議録の登録、変更、削除、参照機能の他、関係者への回覧機能を実現していることに特徴がある。実行画面例を図4に示す。

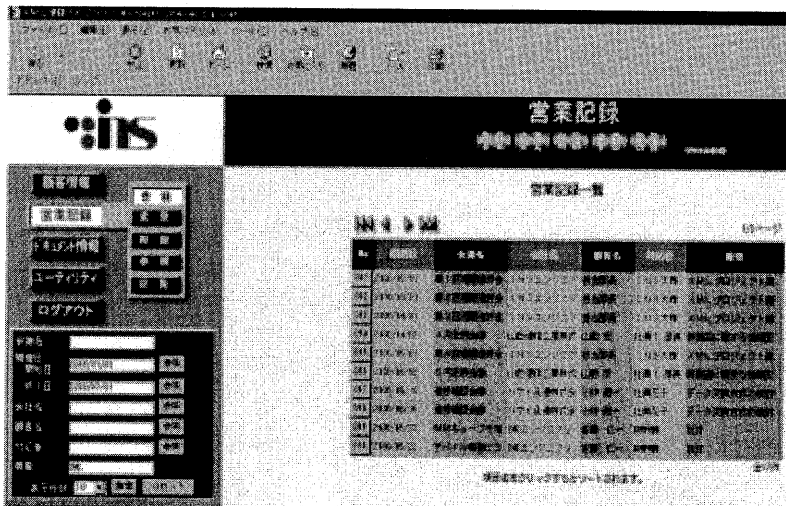


図4 営業記録管理画面

回覧機能は、既存のEメールシステムの送信機能を利用する。このメール通知機能を用いて、登録された営業記録や会議録の承認を案内する。通知を受けたユーザは、その後、ウェブブラウザ上で営業記録の閲覧・承認を行うことになる。

以上の機能を実現するために、“記録ID”、“記入日”、“記入者”、“会議名”、“会議場所”、“開催日”、“開始時刻”、“終了時刻”、“相手企業”、“所属”、“氏名”、“電話番号”、“自社対応者”、“資料”、“目的”、“概要”、“経緯”、“内容”、“特記事項”、“承認者1”、“承認日1”、“メールアドレス1”、“承認者2”、“承認日2”、“メールアドレス2”……といった項目が設定されている。以上は、承認者関係を除き、通常の議事録や折衝記録の項目に対応している。

以上の項目を用いて、営業折衝記録や会議議事録が、ミドルウェアのサーバアプリケーションにより処理され、逐次、承認者に配信・回覧される。

承認には、必ず情報ライブラリへのログインが必要であり、ログインされたユーザ名とその権限によって承認チェックされる。システムユーザアカウントを保持していないクライアント及び、アカウントを保持していても承認者でない場合は閲覧しかできないため、不正な承認が行われることはない。

4.4 システム管理機能

この機能は、システム管理者の機能である。システム管理者とは言っても、Unixシステムのスーパーユーザのような専門家ではなく、若干の専門知識を習得した一般ユーザでサポート可能な機能を提供するものである。

具体的には、ユーザ情報の登録・変更・削除・パスワード変更の機能を提供する。設定画面を図5に示す。ユーザ情報を参照し、当該ユーザの各種機能へのアクセス権の設定、システムへのログイン管理を行う。システム管理者の場合は、全ての機能を使用する権限を保有するが、システム管理者以外のユーザは、当該ユーザ情報以外の登録・変更・削除へのアクセスは許されない。

4.5 その他のユーザ支援機能

イントラネットでのウェブブラウザからのクライアント操作以外に、“XMLメールデータ登録”のサービスを実現している。XMLメールデータ登録機能とは、上記各サービス毎に用意されたテンプレートに必要な項目を入力し、サーバへメール転送を行うことでデータの登録、変更を行うことが可能である。

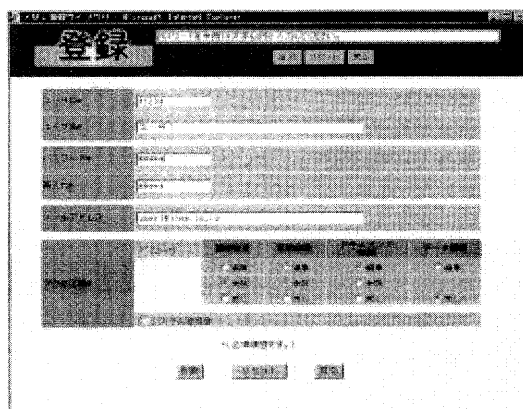


図5 システム管理機能の画面例

その他にも、以前登録したデータを参照し入力データとして使用する参照機能、一覧表示で表示する件数の設定機能、データソート機能、画面表示データの外部ファイルへの出力機能も備えている。

各サービスでは、ウェブブラウザの印刷機能を利用し、スタイルシートで定義された帳票形式での印刷が可能である。

5. 本システムの特徴

ここでは、本システムの特徴を示す。

5.1 容易なシステム導入

本システムは、データの入力、文書の登録の手段としてウェブブラウザを採用している。その結果、クライアントは、プラットフォームのOSに依存せずに実装することが可能である。また、クライアントアプリケーションのインストールも不要である。

今回のプロトタイプにおいては、DBとしてOracleを用いたが、このシステムは特定のDBに制約されるものではない。Oracle以外でも、Access、Notes、SQL Serverなどを用いることができる。従って、これらのRDBを用いるウェブブラウザを用いるクライアント・サーバシステムを運用している場合には、iConnectorと関連のモジュールで構成される最小限のサブシステムの追加・変更で、XMLデータを利用した本システムへの移行が可能である。

5.2 容易な機能追加とシステム拡張

XMLを標準データ形式としているため、他システムとの連携、相互運用を図ることが可能である。また、既存のシステム全体を変更することなく、サブシステム単位で拡張しながら、XMLデータとの連携を図った機能を追加・導入することが可能であり、拡張性に優れている。

5.2.1 各種データ表示・印刷への対応

XMLデータを必要な各種形式（一覧表示、詳細表示、印刷用フォーマット）でウェブブラウザ上に表示するために、その実現方法として、XMLスタイルシート言語（XSL: eXtensible Style Language）を使用した。その結果、データを閲覧するアプリケーションを簡単に構築することが可能であった。そのため、ウェブブラウザ上での表示様式や印刷様式に変更が生じたり、新規の様式を追加する場合もスタイルシートの変更のみで容易に対応が可能である。

5.2.2 データベース項目の追加・変更

データベースの項目に変更が生じた場合でも、図1で示した”DB・XML相互変換”処理で対処可能である。XMLデータのタグ情報とデータベース・スキーマ項目は、ルール定義により実現されている。そのため、ルール定義の変更でタグ情報とデータベース・スキーマ双方を容易に変更することができる。

5.2.3 遠隔端末、モバイル端末によるアクセス

図1で示した”XML化処理”では、データのXML変換と”DB・XML相互変換”処理と連携してメール機能を利用したアクセスを実現する。従って、遠隔のメール端末や、モバイル端末（Windows95/98/CE搭載機、およびZaurus）から”XML化処理”モジュールへデータを送信することにより、データベースへのアクセスとデータ登録・変更が可能である。

5.2.4 容易なメンテナンス

本システムは、既存のカタログ製品（”Oracleデータベース”、”ウェブ・サーバ”、”iMessenger”、”iConnector”、”Internet Explorer5”）を基本モジュールとして構築され、本システム個別の開発部分は極小化されている。そのため、以上の製品に関する知識と、それに追加された本システムの機能について習得すれば、本システムのメンテナンスは可能である。

6. まとめ

本システムは、中小の企業や組織に適合する、拡張可能なオフィスシステムをねらっている。具体的には、基本的な用途を以下の3項目に絞り、幅広い利用者層と維持管理の容易さを狙ったものである。

①社員、顧客、関連企業や組織などの人々の名簿の管理、

②営業折衝記録、会議の議事録などの企業文書

③一般的な各種ドキュメント管理

また、その実現手法としては、極力、既存のカタログ製品の組み合わせにより実現可能とした。

本システムにおいて、以上の狙いはほぼ達成されたと考えているが、検討を要する課題も抽出された。以下にそれらについて述べて考察を加える。

7. 考察

7.1 フラットな情報構造

情報構造から見たXMLの特徴は、DOMに象徴される階層的な構造を構築可能な点にある。そうであれば、XMLを使わずに、一般のプログラム言語によるSQL照会機能と、CSVのようなファイル形式で実現可能ではないかという意見である。

確かにそのとおりであり、Netwareなどの一昔前のクライアント・サーバシステムは、そのような作りであった。では、逆にNetwareのようなシステムが、衰退してしまった理由を考えてみよう。それはインターネットとの相互運用性の欠如にある。

それでは、インターネット対応の上記のようなクライアント・サーバシステムであるマイクロソフトのASP（Active Server Page）のようなウェブブラウザによるクライアントのものとは、どう違うのであろうか？ASPはウェブブラウザによるRDB（SQL Server）へのアクセスは可能にしたが、基本的にはマイクロソフト製品に閉じた世界でしか運用できなかった。それに対し、本製品は、ベンダーを選ばないシステム構成が可能である。

以上から分かるとおり、情報構造としてはフラットな構成をとったが、XMLを使うメリットは確かに存在すると言えるであろう。

7.2 iConnectorによる制約

XMLを適用したことにより、従来のRDBへの照会システムではサポートされていた機能で実現できな

くなったものも存在する。例えば、RDBは、選択 (selection)、射影 (projection)、直積 (product)、集合和 (union)、集合差 (set-difference) といった基本演算と、結合 (join) といった複合演算を包含する関係代数演算をサポートしている。ところがRDBインタフェースのiConnector for Oracleの仕様において、これらの演算の全てをサポートしているわけではない。そのためにカスタマイズの際にはその制約に注意する必要がある。

昨年開発した情報統合サーバ (KMサーバ) [2]において、レガシシステムにおける情報を意味レベルでフィルタリングする機構を開発したが、上記制約のためにiConnectorを適用できなかった経緯がある。

7.3 ドキュメント管理機能

ドキュメントの参照に際し、当該ドキュメントのアプリケーションをクライアントマシンが実装しておく必要がある。この問題は、実は根が深く、本システムのような単純なシステムだけでなく、大規模なイントラネットシステムでも同様な問題になる。

例えばドキュメントの形式をHTMLに限定してしまえば、最も単純なソリューションとなる。しかし、HTML自体はページや段構成といった機能が無く、レイアウトを重視するドキュメントには向かない。

レイアウトを重視するドキュメントの場合には、MS WordやPDFを用いる場合が多い。MS Wordは、オフィスにおける最もポピュラーな文書作成のツールであり、PDFは清書フォーマットの標準的な形式である。従って、この3種類のフォーマットが現状のデファクト・ドキュメントファイル形式と言えるであろう。

7.4 既存のシステムとの相互運用

既存のシステムとの相互運用とオープンな拡張性を適用するシステムとしては、本来OMG (Object Management Group) のCORBA (Common Object Request Broker Architecture) が狙っていたものであるが、通信プロトコルとしてのIIOP (Internet InterORB Protocol) が、基幹系のシステム以外では殆ど普及していないことを考えると、IIOPなどの相互運用は考慮する必要は無さそうである。但し、何らかの理由で、本システムを基幹系と通信さ

せる必要が生じた場合[3]や、特定のデータを複数のプログラム言語で処理せねばならないような場合には、CORBAとの相互運用が現実の解となり得る可能性もある。

モバイル系システムとの相互運用については、WAP (Wireless Application Protocol) のWML (Wireless Markup Language) が国際的には注目されている[4]が、日本ではNTTドコモのi-Mode方式が主流となっているためWMLの展開は未知数である。しかし今後はモバイル系が主流になるという予測もある[5]ので、その時代への対処 (特にデータやコンテンツの移行) も考慮すべきであろう。

8. おわりに

本システムは、中小の企業や組織における汎用的かつ拡張可能なオフィス・システムを狙いとしている。そのために、XMLを高度な機能を実現するためにはなく、既存のシステムとの相互運用と、利用者に対するシステム導入、利用・運用などの容易化という方向に絞って適用した。また、その実現手法としては、極力、既存のカタログ製品の組み合わせにより実現可能とすることを試みたものである。

文献

- [1] 大野邦夫; “いま求められる企業文書管理：米国事情から日本での課題まで”, Open Middleware Report HITACHI, Vol.8, Spring, (1999)
- [2] 吉田正人ほか; “オブジェクト指向スクリプト言語RubyによるXML応用システムの検討”, 情報処理学会デジタルドキュメント研究会研究報告, DD21-3, (1999.11)
- [3] T.J.Mowbray, et. al.; “The Essential CORBA - System Integration Using Distributed Objectd”, John Wiley & Sons, Inc. (1995) 邦訳 “CORBAアーキテクチャ入門 - 分散オブジェクト構築成功の実践テクニック”, トッパン, (1999)
- [4] Jim Stock, “Data gathering using wireless technologies”, Proc. XML Europe 2000 at Paris, June 12 - 16, p.753, (2000)
- [5] Kimmo Ryttonen, “Mobile commerce and WML”, Proc. XML Europe 2000 at Paris, June 12 - 16, p.741, (2000)