

オープン環境でのマルチメディア・データベースの構築

Construction of Multimedia Database under Open Environment

中 川 伸 市

要 約 レポート、画像や表計算データといった多種多様な形式のデータ、つまり、マルチメディア・データを管理するデータベース・システムを開発した。クライアント(ユーザ)は、Macintosh (PowerPC と 68 K) と Windows (95 と NT) を対象とした。可能な限りデータの種類や機種に依存しない処理や操作ができる仕組みを実現した。

クライアント/サーバ、イントラネット、開発ツール、Java や周辺ソフトウェアといったオープン環境で、それぞれの長所を取り入れて開発した。検索はクライアントが多数になるため、導入や操作が容易なイントラネット方式にした。WWW ブラウザを使用した検索や表示は、レポート、画像、表計算データなどのデータを区別せずマルチメディア(ハイパーテキスト)形式のデータとして扱える。表示、ftp、ヘルパ・アプリケーションの起動など WWW ブラウザの機能を利用することで、効率的な開発を行った。また、情報が陳腐化しないように、登録はクライアント/サーバ方式のツールでデータの登録や更新が行えるようにした。画像を扱うため、登録時にサムネイル(実際の画像データを 100×100 画素程度に縮小したもの)を自動的に作成し、検索画面に高速に表示できるようにした。また、一括登録、データ容量が増大したときの対応策やセキュリティも考慮した。

Abstract We have developed a database system which manages data with diverse formats, such as reports, images, spreadsheets and so on as "multimedia" data. Having targeted Macintosh (PowerPC and 68 K) and Windows (95 and NT) users as potential buyers, we have developed a system that allows processing and operation which is relatively independent of a variety of data types of machine.

The system is developed with a capacity to take full advantage of the client/server system, Intranet, development tools, Java, and peripheral software under an open environment. Since a vast number of clients tend to search data, we have adopted the Intranet mode because of its easy installation and operational facilities. Data search and display using a WWW browser enables processing of data such as reports, images, and spreadsheets in a multimedia (hypertext) format without differentiating between them. We have been able to develop the system efficiently by using WWW browser functions such as display, ftp, and execution of the helper application. Moreover, we have applied the client/server tool to enable data registration and updating to prevent information from being obsolete. To handle image data, we have developed a function which automatically creates thumbnails (of 100×100 pixel reduced size images) when the images registered, and displays them on the search screen at high speed.

1. はじめに

多種多様な形式のデータ、つまり、マルチメディア・データを管理できるデータベース・システムを開発した。目的は、部署内に混沌と存在する情報・データを整理し、情報分析・解析やアイデア展開を支援することである。

対象とする情報は、レポート(画像を含むワープロ・データ)、画像(静止画像、動画像)、表計算データなどが混在する。要件として次の項目が挙げられた。

- 1) これら各種データをデータベース上に構築すること
- 2) 多角的かつ瞬時に検索できること
- 3) クライアントはネットワーク上の既存のパーソナル・コンピュータ(以降、PCと呼ぶ)で、Macintosh (PowerPC と 68 K)^{*1} と Windows (95 と NT)^{*2} から同様に操作できること
- 4) 機能の拡充が容易で、ユーザ・インタフェース部分は簡易言語が利用できること
- 5) システム導入が容易であること
- 6) データ量が増大したときのシステム拡張が容易であること
- 7) セキュリティを守ることができること

これらの要件を満たすため、クライアント/サーバ方式、イントラネット方式、開発ツールや Java^{*3} の調査を行った。既存の技術を利用し組み合わせるといったオープン環境での開発を進めた。

本稿では、まずクライアント/サーバ方式とイントラネット方式の比較、Macintosh 対応について調査した結果を報告する。次にシステムの機能や形態を紹介する。最後に Java の利用やセキュリティについて、工夫した点を記述する。

2. クライアント/サーバ方式とイントラネット方式

クライアント/サーバ方式かイントラネット方式かを定めるため、検索時の応答性と操作性を比較し、実現方法を考察した。

2.1 応答性の比較

次のような環境で評価を行った。

モデル・ケース : 10 件の 79 × 100 画素の縮小画像 (画像の検索では、高速化のため縮小した画像をまず表示し、選択のガイドとする。以降、サムネイルと呼ぶ) を検索し表示。テキスト・データは無し。サムネイルの原画はフルカラー画像 (1677 万色) で、Paint Shop Pro^{*4} でファイル変換や減色処理を行った。

クライアント機種 : Windows 95, 133 MHz, 32 MB メモリ

サーバ機種 : US 120^{*5}, 60 MHz, 128 MB メモリ

測定方法 : CGI^{*6} 実行時間は、HTML^{*7} ファイルを作成する単体プログラムのサーバでの実行実測値 (100 回実行し、1/100 にする) データベース開始実行時間は、Oracle Server^{*8} がセッションを開始終了する単体プログラムのサーバでの実行実測値 (100 回実行し、1/100 にする)

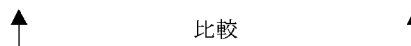
転送時間は、10 BaseT^{*9} を想定した理論値 (同一 Hub^{*10} の場合 0.5 MB/秒、別 Hub の場合 0.08 MB/秒とする)

表示時間は、クライアントでローカル・ファイルから表示するまでの実測値 (80 件表示し、1/8 にする)

結果は、表 1 のようになった。CGI を利用したイントラネット方式のデータベース検索は、都度サーバ側でセッション開始終了処理を行うため、応答性が悪い (キャッ

表 1 クライアント/サーバ方式とイントラネット方式の応答性の比較

	クライアント/サーバ方式				イントラネット方式		
ブラウザ*	日本ユニシス提案ボード				NetscapeNavigator3.0		
HTML サイズ*	-				1KB		
サムネイルの色数	1677 万色		256 色		1677 万色	256 色	
1 サムネイル・サイズ*	24KB		9KB		11KB	9KB	
10 サムネイル・サイズ*	0.24MB		0.09MB		0.11MB	0.09MB	
サムネイル・ファイル書式	BMP	TIFF	BMP	TIFF	JPEG	GIF	
CGI 実行	-	-	-	-	0.08 秒	0.08 秒	
データベース開始終了	-	-	-	-	0.38 秒	0.38 秒	
データベース検索	同等						
HTML の転送	同一 Hub	-	-	-	-	0.002 秒	0.002 秒
	別 Hub	-	-	-	-	0.013 秒	0.013 秒
サムネイルの転送	同一 Hub	0.48 秒	0.48 秒	0.18 秒	0.18 秒	0.22 秒	0.18 秒
	別 Hub	3 秒	3 秒	1.13 秒	1.13 秒	1.38 秒	1.13 秒
表示	0.25 秒	0.3 秒	0.25 秒	1.25 秒	0.38 秒	0.3 秒	
合計(データベース検索は除く)							
キャッシュに存在しない場合	同一 Hub	0.73 秒	0.78 秒	<u>0.43 秒</u>	1.43 秒	1.06 秒	<u>0.94 秒</u>
	別 Hub	3.25 秒	3.3 秒	<u>1.38 秒</u>	2.38 秒	2.23 秒	<u>1.90 秒</u>
キャッシュに存在する場合	-	-	-	-	0.38 秒	<u>0.3 秒</u>	



シュに検索データが存在する場合のみ、例外的に高速) ことがわかった。複数のクライアントが同時にアクセスした場合、サーバ処理が集中するため、イントラネット方式の不利が予想された。

ただし、改善した製品が主に 2 種類現れている。一つは、Oracle Web Server^{*11}、Illustra (Informix) Web Data Blade^{*12} や Net Dynamics^{*13} などの WWW^{*14} サーバ・データベース連携システムで、セッション管理や並列処理を行うことで高速化する^[1]。もう一つは、Oracle Power Objects^{*15}、Power Builder + Internet Developer Toolkit^{*16} や Java などのプラグイン方式で、WWW ブラウザ^{*17} にプラグインを提供し、直接リモートのデータベースにアクセスする^{[2][3][4][5]}。

後者はクライアント/サーバ方式並に高速化できるが、WWW ブラウザ(クライアント)側に SQL * Net^{*18} や JDBC^{*19} ドライバ (JDBC ODBC^{*20} ブリッジ + ODBC ドライバでも良い) の導入が必要となり、手間が発生する。筆者の知るところ、プラグイン方式の Macintosh 版は未だ無い。

まずはオープンな環境を構築するため、WWW サーバは CERN httpd^{*21} を使用し CGI を利用することにした。

前例はデータベース検索を比較したが、単に画面を変更する場合も、イントラネット方式は都度サーバにアクセスするため不利である。回避するには、Java や JavaScript^{*22} の使用が必要である。

また、サムネイル 10 件程度の転送データの場合、同一 Hub と別 Hub の比較のよ

うに、ネットワークの負荷が応答性に大きく影響することも判明した。回避するには、ネットワークの帯域を広げる、または、ビッグ・パイプ付きスイッチング Hub (100 BaseT^{*23}/10 BaseT) を導入し、高速化したいクライアントをサーバと同一 Hub に接続する。サーバは 100 BaseT、クライアントは 10 BaseT に接続するなどの考慮も必要となる。

2.2 操作性の比較

操作性を比較した場合、表 2 のようになる。クライアント/サーバ方式は、ダウンロードの操作が必須となり、操作性は良くない。イントラネット方式は、WWW ブラウザに直接表示できる場合があるが、レポート、画像、表計算データで操作性が異なる。EMBED タグを使用したプラグインもテストしたが、動作に不具合が発生しやすく、また都度データをロードしてしまうため応答性が悪くなってしまった。今後 ActiveX コントロール^{*24}などの WWW ブラウザの機能向上により、操作性の改善は見込めるが、セキュリティや Macintosh 対応に問題がある。よって、クライアントに存在するプログラムを自動起動するヘルパ・アプリケーション^{*25}の仕組みを利用することにした。

表 2 クライアント/サーバ方式とイントラネット方式の操作性の比較

	クライアント/サーバ方式	イントラネット方式	
WWW ブラウザ種類	—	NetscapeNavigator3.0	Active-X 対応ブラウザ
レポート文章(画像を含むワープロデータ)	▲:一旦ダウンロード後、アイコンをダブルクリックなどしてアプリケーション・プログラムを起動し表示	○:HTML に変換済みの場合、直接ブラウザに表示 (▲:登録時に必要な HTML 変換は、未だ不確かなものが多い) ▲:HTML 以外の場合、一旦ダウンロード後、アイコンをダブルクリックなどしてアプリケーション・プログラムを起動し表示 △:HTML 以外でもヘルパ・アプリケーションに設定しておけば、自動的にアプリケーション・プログラムを起動し表示	○:HTML に変換済みの場合、直接ブラウザに表示 (▲:登録時に必要な HTML 変換は、未だ不確かなものが多い) ▲:HTML 以外の場合、一旦ダウンロード後、アイコンをダブルクリックなどしてアプリケーション・プログラムを起動し表示 ○:HTML 以外でも Word など ActiveX 系の場合、直接ブラウザに表示
画像(TIFF などの画像データ)や表計算データ(Excel などの表データ)	▲:一旦ダウンロード後、アイコンをダブルクリックなどしてアプリケーション・プログラムを起動し表示	△:ヘルパ・アプリケーションに設定しておけば、自動的にアプリケーション・プログラムを起動し表示 △:EMBED タグにより、ページへの張り付け可能 (Windows 版のみ、また、動作が不安定)	△:ヘルパ・アプリケーションに設定しておけば、自動的にアプリケーション・プログラムを起動し表示 ○:Excel など ActiveX 系の場合、直接ブラウザに表示

○:良好に使用可能、△:使用可能、▲:不便

2.3 実現方法

前述の調査結果から、応答性はクライアント・サーバ方式が優れるが、操作性ではイントラネット方式が優れている。その他にシステム導入の手間については、サーバのみ対応すれば良いイントラネット方式が負荷が小さい。検索においては、多数のク

クライアントが存在することから、イントラネット方式を採用することにした。ミドル・ソフトウェアとして WWW ブラウザの Netscape Navigator (または Communicator) など(「表 3 実現方法の比較」の C 案)を使用することにした。

登録においては、WWW ブラウザからクライアント上のローカル・ファイルにアクセスできないため、イントラネット方式ではクライアントからレポート、画像、表計算データなどのデータ・ファイル(以降、実データと呼ぶ)をサーバにアップロードできない。よって少数を対象とする登録はクライアント/サーバ方式とし、Macintosh と Windows が同様に稼働する仕組みを調査した。

Macintosh 版の ODBC や JDBC が無いため、標準的なツールでは Macintosh からデータベースへのアクセスは実現できない。開発ツール - ネットワーク・ドライバ - データベースは、データベース・ベンダが提供するデータベース固有のミドル・ソフトウェアを採用した。特に Macintosh に対応するため、Oracle 関連ミドル・ソフトウェアの Oracle Power Objects/SQL * Net/Oracle Server(「表 3 実現方法の比較」の A 案)を使用した(1998 年 4 月現在、Oracle Power Objects や Oracle Developer/2000^{*26}のみ Macintosh と Windows に対応可能)。

検索ツールはスクリプトに似た言語の HTML、登録ツールは開発ツールを使用するため、機能拡充が容易となった。

3. システムの概要

3.1 データの扱い

登録や(表示、ダウンロードを含む)検索の対象となる情報は、次のとおりである。

- ① レポート・データ：Word^{*27}、HTML など
- ② 画像データ：Photoshop^{*28}、TIFF^{*29}、JPEG^{*30}、Postscript^{*31} や PhotoCD^{*32} などの静止画や Quick Time^{*33} などの動画
- ③ 表計算データ：Excel^{*34} など

属性データ(パラメタ=検索キー)をリレーショナル・データベースのテーブル上に、レポート、画像、表計算データなどの実データをディスク上に管理する(図 1)。属性データの中にユニークな名前に変換した実データのファイル名と書式に合った拡張子を持つ。拡張子によって MIME^{*35} タイプを設定し、検索ツールでは WWW ブラウザがデータをロードするとき MIME タイプ毎に設定したヘルパ・アプリケーションを起動し、表示や編集を行う。データの種類によらないマルチメディア(ハイパーテキスト)形式のデータとして扱うことが可能となった。

3.2 画像データベース

主な機能である画像データベースについて述べる。

3.2.1 サムネイル

サムネイルは、画像を 100×100 画素程度に縮小したものである。メニュー上に画像を逐一表示したのでは表示時間や表示領域の面で負荷が大きい。よって、画像の代替としてサムネイルを表示することで効率向上を図った。

WWW ブラウザには、実データへリンク付けしたサムネイルを表示する。サムネイルを指示することで、実データの ftp^{*36} によるダウンロードやヘルパ・アプリケー

表 3 実現方法の比較

案名	クライアント/サーバ方式			イントラネット方式
	A案	B案	C案	
導入ソフトウェア	クライアント	Oracle Power Objects アプリケーション + SQL*Net	Java アプリケーション + JDBC + JDBC- ODBC フリッジ + ODBC+SQL*Net など	WWWブラウザ (サーバからの HTML,Java ア プレット, Javascript)
	サーバ	SQL*Net + Oracle Server	SQL*Net + Oracle Server	WWW サーバ (+ CGI) + Oracle Server
マルチプラットフォーム対応	Windows95	○	○	○
	WindowsNT	○	○	○
	Windows3.1	○	×:JDBC-ODBC フリ ッジ無	○
	Macintosh (PowerPC)	○	×:JDBC-ODBC フリ ッジ無	○
	Macintosh (68K)	○	×:JDBC-ODBC フリ ッジ無	○
	互換性	△:ソースレベル よって、Oracle Power Objects 開発環境は、 各プラットフォームに必要	○:バイトコードレベル	○
ユーザ I/F カスタマイズ (開発ツール)	○:ただし使い易くす るために、Windows, Macintosh の固有コマ ンドを使用する可能性 あり	△:SQL 呼び出し部 分にプログラミング必要	△:SQL 呼び出し部分にプ ログラミング必要	
汎用性	△:Oracle 固有。使 い易くするために、 Windows, Macintosh の固有コマンドを使用 する可能性あり	○:Java 次第	○	
実データのデータベース 登録	○:ファイル転送の場合、実データのアップロード必 要。 ファイル共有の場合、登録ファイルをドラッグ・アンド ・ドロップ。→セキュリティ確保のため、ファイル転送を 採用		×	
実データのデータベース 検索 + 表示	△:ファイル転送の場合、実データのダウンロード必 要。ファイル共有の場合、検索ファイルをダブル・クリ ック。→セキュリティ確保のため、ファイル転送を採用		○:WWW ブラウザによる表 示の利用や実データのダウ ンロード。前項参照のこと	
導入、保守の容易さ	△:全クライアントへの対応		○:サーバのみへの対応	

○:対応可能、△:困難を伴うが対応可能、×:対応不可能

ションの起動による画像編集が可能である。

3.2.2 登録ツール

登録ツールは開発ツール Oracle Power Objects を使用し開発した(図2)。

GUI 経由でデータベースの属性データを追加,更新,削除できる。その他に,サムネイル自動作成のために画像処理ツールを起動したり,クライアント上の実データやサムネイルをユニークな名前に変換後サーバにアップロードするために ftp を起動させる。Windows の場合, ActiveX コントロールを使用することで,あたかも登録

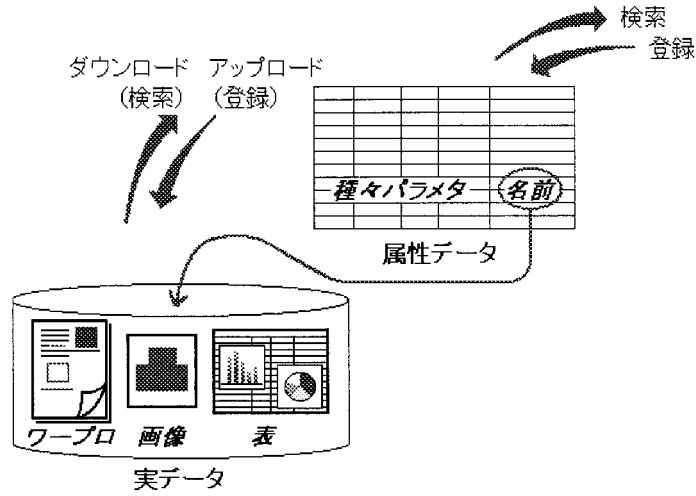


図 1 属性データとマルチメディア形式の実データ

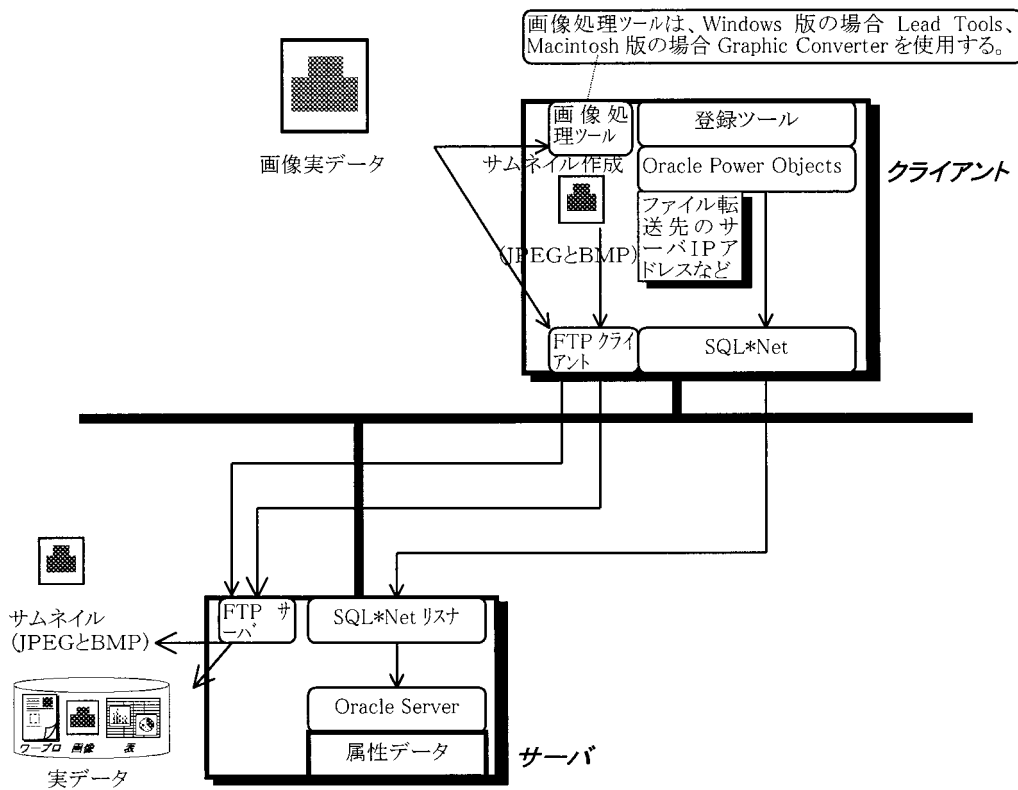


図 2 クライアント/サーバ方式による登録

ツールの 1 機能として、メニュー上に画像処理ツール Lead Tools^{*37}などの機能を組み込んだ。Macintosh の場合は組み込みが容易ではないため、Applescript^{*38}と CEQuicKeys^{*39}を起動してあらかじめ登録しておいたキー入力操作を自動再現し、画

像処理ツール Graphic Converter^{*40} や ftp ツール Anarchie^{*41} を自動実行するようにした。Photoshop, TIFF, JPEG, Postscript など様々な画像書式を入力し, WWW ブラウザ用に JPEG 書式のサムネイルを, Oracle Power Objects 用に BMP^{*42} 書式のサムネイルを作成するために, 市販の Lead Tools や Graphic Converter を利用することにした。

また多量のデータを登録するため, 名前やキーワードについては CSV^{*43} 書式の Excel ファイルからの一括入力を可能とした。

動画については, サムネイルの自動作成は現状不可能。クライアントでサムネイルを手動で作成し実データとともにアップロードするか, あるいは動画は大容量のデータになるため単にリンク付けするのみとし, 静止画と同様に検索ツールで扱えるようにする。

3.2.3 検索ツールと表示

検索はイントラネット方式の検索ツール (WWW ブラウザ) から行う。

属性表示, サムネイル一覧表示, 画像/表の実データの表示やダウンロードができる。画像/表の実データの表示は, WWW ブラウザのオプション機能のヘルパ・アプリケーションを利用する(図3)。ヘルパ・アプリケーションを自動起動するために, WWW サーバと WWW ブラウザに MIME タイプの設定を行うことが必要で, 表 4

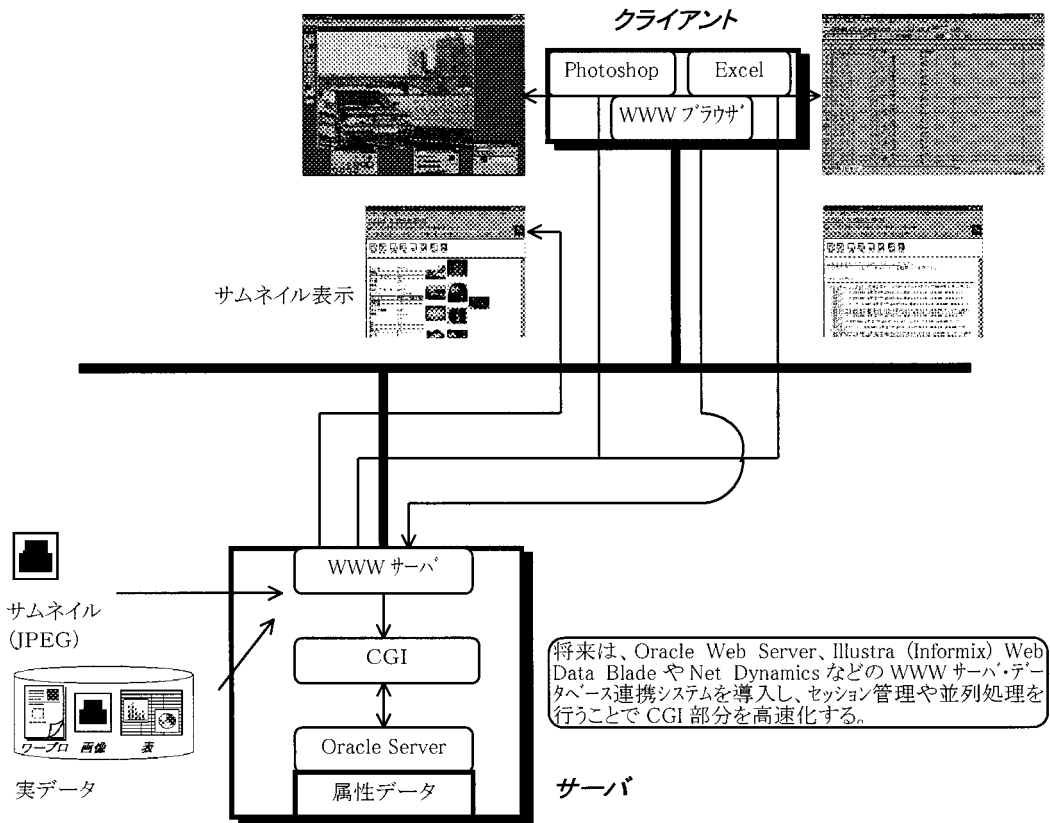


図 3 イン트라ネット方式による検索

のようにした。ただし MIME タイプは実データの拡張子によってデータの種類を判断するため、通常拡張子を付加しない（ファイルの属性を持っている）Macintosh では運用によって拡張子を付ける必要がある。検索結果の実データの種類によって、次のように表示が異なり、また、ダウンロードのみも可能である。

- 1) HTML 形式のレポートの場合、WWW ブラウザ内に表示する。
- 2) 画像の場合、通常はサムネイルの表示を行う。
さらに画像の実データを表示するとき、
- 3) Photoshop や TIFF 形式の場合、WWW ブラウザがヘルパ・アプリケーションとしてオプション登録した Photoshop などのペイント・ソフトウェアを自動起動する。
- 4) GIF や JPEG 形式の画像の場合、WWW ブラウザ内に表示する。
- 5) CSV や XLS^{*44} 形式の Excel 表データの場合、WWW ブラウザがヘルパ・アプリケーションとしてオプション登録した Excel などの表計算ソフトウェアを自動起動する。
- 6) Quick Time 形式の動画の場合、WWW ブラウザが備えた Quick Time プラグ・イン・アプリケーションが自動起動される。

表 4 WWW サーバと WWW ブラウザの MIME タイプの設定

ファイル形式	WWW サーバ		WWW ブラウザ
	MIME タイプ/MIME サブタイプ	拡張子	起動アプリケーション
Photoshop	application/psd	psd	Photoshop へのパス
TIFF	image/tiff	tif,tiff	Photoshop へのパス
BMP	image/x-MS-bmp	bmp	PaintShopPro へのパス
PhotoCD	image/x-photo-cd	pcd	未定義
Postscript	application/postscript	ps	Photoshop へのパス
EPS	application/eps	eps	Photoshop へのパス
CSV	application/csv	csv	Excel へのパス
Microsoft Excel ブック	application/xls	xls	Excel へのパス

3.2.4 システムの拡張

画像を扱うデータベースは大容量になる。将来のディスク増設、CD チェンジャ接続、新規サーバ導入、他サーバ利用などの容量の拡張に対応できるように次の考慮を行った。

画像データベースの登録のとき、サムネイル・ファイル名と画像実データ・ファイル名以外にファイル転送先（サーバの IP アドレス、ユーザ、パスワード、サムネイル・ディレクトリ、画像実データ・ディレクトリ、サムネイル URL、画像実データ URL）をデータベースに保存する（図 2）。ツールのファイル転送定義ファイル ftp.txt にファイル転送先を記述しておき、画像データベースの検索や削除のとき、データベースのファイル転送先（＝サムネイルや画像実データの保存場所）を参照する。

CD ROM へのデータ登録などによって CD チェンジャにファイルが移行した場合、データベースのファイル転送先を変更する。ただし、新規サーバの導入、他サーバの

利用を行う場合、それらのサーバには WWW サーバと ftp サーバが必要である。

サムネイル、実データを個々に URL で管理することで、周辺装置やサーバの追加やデータ移行に柔軟な対応が可能となった。

4. Java

4.1 実データのリンク付け機能

前項の登録機能の他に、実データの存在場所にリンク付けする機能も付加した。実データの登録や削除は、あらかじめサーバ上のホーム・ディレクトリに対し、ftp や telnet^{*45}などで各ユーザが行っておく。検索キーと実データへのリンク先を設定するために、WWW ブラウザから操作する。

サーバ上に存在する実データへのリンク先を登録するため、ディレクトリやファイル名を一覧表から選択できるようにした。サーバのディレクトリの検索とファイル名の表示を繰り返す部分には Java アプレット^{*46}を使用した。これを HTML で処理すると画面の書き直しが必要となるが、Java アプレットを使用すると、別ウィンドウの表示やディレクトリ名とファイル名だけの書き直しで済み、高速化が図れた。

4.2 Java の問題点

導入ユーザは Macintosh や Windows 上で、WWW ブラウザとして Netscape Navigator 3.01 [ja] を使用していた。Netscape Navigator 3.01 [ja] は、JDK^{*47}1.0.2 に対応しているため、Java コンパイラも JDK 1.0.2 対応の Microsoft 社 VJ⁺1.0 や Sun (Soft) 社 JDK 1.0.2 を使用した。発生した問題点と対策を次に記述する。

4.2.1 日本語のエンコードと日本語入力

Java アプレットで、サーバのマスタ・ファイルから読み込んだ値を入力画面の選択肢として表示したり、入力画面からの入力値によってサーバのマスタ・ファイルとデータベースを更新する場合があった。

サーバのマスタ・ファイルとデータベースは、文字コードに EUC^{*48}を使用する。Java アプレットでは文字コードを Unicode^{*49}にする必要がある。Unicode と EUC のコード変換をする Java 関数は、JDK 1.1.X から使用可能で、JDK 1.1.X に未対応の WWW ブラウザでは使用できない。結局、Java のメーリング・リストから入手した情報を元に、EUC と Unicode の変換のためのクラスを作成し、Java アプレットに組み込んだ⁷¹⁸¹⁹¹。

また、WWW ブラウザと Macintosh/Windows の機種の種類組み合わせによっては、Java アプレットの日本語入力の可/不可が発生した。

テキスト・エディタで日本語入力したものをコピー・ペーストする運用にした。

開発の時点で、Java コンパイラの JDK の最新仕様に対応する WWW ブラウザが無かったため、このような方法で問題を回避した。

4.2.2 機種依存のバグ

例えば、Macintosh 版の Netscape Navigator 3.01 [ja] では Java.util.Date 関数が使用できない。この関数は年月日の入力に対するエラー・チェックに使用しており、Macintosh 版の Netscape Navigator で実行すると、エラー・チェックのとき “Java.util.Date not found” というエラー・メッセージが表示され、以後処理が進まない。

よって、Macintosh では使用不可能となった。

Windows 版の Netscape Navigator では問題なく実行できるため、一部機能は Windows を使用して運用してもらうことになった。

4.2.3 その他

WWW ブラウザが Java アプレットをロード中は、停止操作が不可能となる。よってアプリケーションのサイズが大きいつき、途中で止めたくてもロードし終わるまで待たなくてはならない。Java アプレットは 1 実行単位を小さなサイズにすべきであることがわかった。

また、Macintosh 版の Internet Explorer を使用する場合、事前に Apple 社や Microsoft 社から Java 環境をダウンロードする必要があるため不便である。

その他に、Macintosh 版の Netscape Navigator では、プルダウンメニューの日本語が化ける（ただし、全体ウィンドウを再表示すると正しく表示する）。

4.3 Java の評価

マルチプラットフォームに有効と言われる Java だが、バージョン、機種特に Macintosh や WWW ブラウザの組み合わせによっては、不具合が発生する。それらの組み合わせ分だけ、テストも必要になる。現在も Java の仕様が流動的なため、バージョンが上がるとプログラムの修正も必要である。

今回の開発での Java の利用は、高速化が必要な部分「サーバのディレクトリの検索とファイル名の表示の繰り返し」に留めた。

5. セキュリティ

セキュリティを確保するため、次の工夫を行った。

5.1 ユーザ認証とアクセス制御

ユーザ認証とアクセス制御の種類は表 5 のとおりである。

表 5 ユーザ認証とアクセス制御の種類

機能	クライアント/サーバ方式	イントラネット方式	
	画像DBの登録	画像DBの検索	実データのリンク付け機能
①従業員コード	○	○	○
②端末の IP アドレス		○	○
③クライアント・プログラムの有無	○		
④WWW サーバ・CGI の POST 呼び出し		○	○
⑤WWW サーバ・ディレクトリ参照オプション		○	○
⑥セキュリティ・レベル		将来	将来
⑦セキュリティ・システム	将来	将来	将来
⑧UNIX のユーザ/パスワード			○

5.1.1 クライアント/サーバ方式

将来は、先頭でセキュリティ・システムのユーザ認証関数を呼び、ユーザ (= 従業員コード) とパスワード (= 従業員個人のパスワード) を入力 (= 認証) する。当面

は、従業員コードのみ入力し、本システムの利用ユーザ設定機能であらかじめ登録したユーザ以外はエラーとする仕組みで導入することにした。利用ユーザ設定は、従業員コード、名前、部署、ユーザの機密（セキュリティ）レベルを登録する。

利用端末設定機能もあるが、クライアント/サーバ形式での登録では端末の認証は行わない。

登録する情報には、都度セキュリティ・レベルを設定する。

5.1.2 イン트라ネット方式

将来は、先頭でセキュリティ・システムのユーザ認証画面を呼び、ユーザ（＝従業員コード）とパスワード（＝従業員個人のパスワード）を入力（＝認証）する。当面は、従業員コードのみ入力する。

その他に、クライアントの IP アドレスによって、WWW サーバがアクセス制御する。本システムの利用端末設定機能であらかじめ登録した IP アドレスの端末以外はエラーとした。利用端末設定は、IP アドレス、利用部署、管理者を登録する。PROXY サーバの IP アドレスを除外することで、部外からのアクセスを拒否した。もし、PROXY サーバの IP アドレスを含む設定の場合、PROXY サーバの設定によって部外からのアクセスを拒否する。

将来は、データへのアクセス単位にセキュリティ・レベルを認証する。ユーザのセキュリティ・レベルより高い場合、アクセス不可能となる。

特に、他ページへのリンクは、HTML の HREF 呼び出しではなく、CGI の POST 呼び出しにすることで、WWW ブラウザからの URL キー入力による直接アクセスを不可にする。開始ページのユーザ（＝従業員コード）の入力から開始しない限り、以降のページにアクセスできないようにした。

また、WWW サーバ（httpd）を起動する際、サーバのディレクトリ参照のオプションをオフ（CERN の場合、httpd dn で起動するなど）にすることで、サーバの機密の保護を行った。

5.1.3 セキュリティ・レベル

レベル分けは表 6 のようにし、アクセスの制限を設ける。

表 6 セキュリティ・レベル

セキュリティ・レベル	利用可能者	対象情報(データ)例
A	部長以上	人事、長期戦略情報
B	室長以上	長期計画
C	プロジェクト担当レベル	スケッチ、写真
D	部内一般	企画情報
E	社内開放	機密の低いレポート
F	社外開放	機密無し

6. おわりに

イントラネットという言葉ができる以前に、WWW サーバ、データベースや CD チェンジャを連携させたインターネットのシステムに携わった。そのとき、これは社内の情報システムに利用できると思った。今回その技術をドキュメントや画像の保存/

発信に適応した。

ユニークに変換したファイル名と実データを管理することで、データの種類によらないデータベースを構築できた。イントラネット方式を採用し、機種に依存しない検索システムを実現した。表示、ftp やヘルパ・アプリケーションの起動など WWW ブラウザの機能を利用することで、効率的な開発が行えた。特に画像データについては、サムネイルを自動作成し、実データの代替として使用することで、効率よい検索が行えるようになった。また、プログラム修正はサーバ上でのみ行えば良いため、システム導入が容易である。ユーザが多数になる検索システムには向いている。URL を管理することで、システム拡張にも柔軟に対応できる。

登録システムはクライアント上のファイルをアクセスするため、クライアント/サーバ方式にせざるをえない。Macintosh や Windows といった機種に依存しない開発を行うのは難しい。今回は開発ツールを利用することで回避したが、マルチプラットフォームに有効と言われる Java を利用したとしても、JDBC など Macintosh に未対応の場合が多いことがわかった。

開発期間の短縮のために、既存の技術をできるだけ組み合わせて利用したが、オープン環境ではバージョンの違いによる不具合も発生しやすいバージョンを固定する、最新版に追随する、肝心な部分は開発するといった見極めを行いながら、システムの充実を図っていきたい。

-
- * 1 Macintosh は、Apple 社の PC である。
CPU に PowerPC と 68000 系があり、それぞれに OS (オペレーティング・システム) が異なる。
 - * 2 Windows は、Microsoft 社の OS である。
Windows 95 や WindowsNT などの種類がある。
 - * 3 Java は、Sun Microsystems 社が提唱するプログラミング言語である。
 - * 4 Paint Shop Pro は、Windows 上で稼働する JASC 社のペイント・システムである。
 - * 5 US 120 は、日本ユニシス社のエンジニアリング・ワークステーションである。
 - * 6 CGI は、Common Gateway Interface の略で、WWW ブラウザからの要求に対し、サーバが別のプログラムを起動し、その結果を返すことを行うインタフェースである。
 - * 7 HTML は、Hyper Text Markup Language の略で、ハイパーテキスト形式のファイルを主に WWW ブラウザで表示するための言語である。
 - * 8 Oracle Server は、Oracle 社のデータベース・システムである。
 - * 9 10 BaseT は、10 Mbits/秒の転送速度を想定したツイスト・ケーブルを使用したネットワークである。
 - * 10 Hub は、10 BaseT などのネットワークを一旦集約するための機器である。
 - * 11 Oracle Web Server は、Oracle 社のデータベース連携 WWW サーバである。
 - * 12 Illustra (Informix) Web Data Blade は、Illustra (Informix) 社の WWW サーバ/データベース連携ツールである。
 - * 13 Net Dynamics は、Net Dynamics 社の WWW サーバ/データベース連携ツールである。
 - * 14 WWW は、World Wide Web の略で、インターネット上でハイパーテキストを使用して送受信する仕組みである。
 - * 15 Oracle Power Objects は、Oracle 社のクライアント側のデータベース開発環境である。
 - * 16 Power Builder + Internet Developer Toolkit は、Powersoft 社のクライアント側のデータベース開発環境である。
 - * 17 WWW ブラウザは、ハイパーテキストを表示するビューアである。
主に Netscape 社の Navigator と Microsoft 社の Internet Explorer がある。
 - * 18 SQL *Net は、Oracle 社のネットワーク・データベース・アクセス・ドライバである。
 - * 19 JDBC は、Java Database Connectivity の略で、Java プログラムからのデータベース・インタフェースである。
 - * 20 ODBC は、Open Database Connectivity の略で、Microsoft 社が提唱する標準データベース・インタフェースである。
 - * 21 CERN httpd は、CERN の WWW サーバ・デーモン・プログラムである。その他に、NCSA や Apache などが有名。
 - * 22 Javascript は、主に WWW ブラウザ上で稼働する Java に似たスクリプト言語である。

- * 23 100 BaseT は、100 Mbps/秒の転送速度を想定したツイスト・ケーブルを使用したネットワークである。
- * 24 ActiveX コントロールは、Windows 上で共通に使用できるカスタム・コントロールである。
- * 25 ヘルパ・アプリケーションは、WWW ブラウザが新たに起動するアプリケーションである。
- * 26 Oracle Developer/2000 は、Oracle 社のクライアント側のデータベース開発環境である。
- * 27 Word は、Microsoft 社のワープロ (ワードプロセッサ)・アプリケーションである。
- * 28 Photoshop は、Adobe 社のペイント・システムである。
- * 29 TIFF は、Tag Image File Format の略で、汎用的な画像書式である。
- * 30 JPEG は、Joint Photographic Experts Group の略で、ISO および ITUTS が標準化した静止画圧縮のためのアルゴリズムである。
- * 31 Postscript は、Adobe 社のページ記述言語である。
- * 32 PhotoCD は、Kodak 社が提唱する画像書式である。
- * 33 Quick Time は、Apple 社の動画アプリケーションである。
- * 34 Excel は、Microsoft 社の表計算アプリケーションである。
- * 35 MIME は、MultiPurpose Internet Mail Extensions の略で、データの種別を識別する。
- * 36 ftp は、File Transfer Protocol の略で、ファイル転送を意味する。
- * 37 Lead Tools は、Windows 上で稼働する Lead Technologies 社の画像処理ツールである。
- * 38 Applescript は、Macintosh に付属のスクリプト言語である。
- * 39 CEQuicKeys は、Macintosh 上で稼働するキー操作自動化ツールである。
- * 40 Graphic Converter は、Macintosh 上で稼働する Space Link 社の画像処理ツールである。
- * 41 Anarchie は、Macintosh 上で稼働する ftp クライアント・ツールである。
- * 42 BMP は、Windows 用の画像書式である。
- * 43 CSV は、Comma Separate Value の略で、カンマで区切った値を記述したテキスト・ファイルを意味する。
- * 44 XLS は、Excel 用の標準データ形式である。
- * 45 telnet は、Telecommunication Network の略で、米国国防総省が定義した仮想端末サービスである。
- * 46 Java アプレットは、主に WWW ブラウザ上で稼働する Java プログラムである。
- * 47 JDK は、Java Development Kit の略で、Java の開発環境である。
- * 48 EUC は、Extended Unix Code の略で、主に UNIX で使用する文字コード体系である。
- * 49 Unicode は、Java などで使用される新しい文字コード体系である。

- 参考文献** [1] 磯辺 裕一, “製品別アプリケーション・サーバー利用法”, UNIX USER, Vol. 5, No. 12, DEC. 1, pp. 54 ~ 61, 1996.
- [2] 高橋 信頼, “OPO 2.0 を出荷へ, Plug in 作成やブラウザ部品の標準装備で WWW アプリに対応”, 日経オープンシステム, no. 44, 11 月, pp. 206, 1996
- [3] 麻生 二郎, “WWW 対応は当たり前に, 3 階層への動きが目立つ”, 日経オープンシステム, no. 45, 12 月, pp. 251 ~ 261, 1996.
- [4] 斉藤 淳・森重 和春, “OPO アプリケーションがブラウザ上で快適に動作”, 日経オープンシステム, no. 45, 12 月, pp. 197 ~ 206, 1996.
- [5] 真島 馨, “開発の容易な IntraBuilder 実行速度重視なら OPO 2”, 日経バイト, 1 月, pp. 224 ~ 230, 1997.
- [6] JavaFrequentlyAskedQuestion (FAQ) HYPERLINK <http://www.webcity.co.jp/info/andoh/java/javafaq.html>
- [7] Java^(tm) House Mailing List Homepage, HYPERLINK <http://java.house.center.nitech.ac.jp/ml>
- [8] Java^(tm) house Brewers ML, java.house.brewers@center.nitech.ac.jp

執筆者紹介 中川 伸 市 (Shin'ichi Nakagawa)

1960 年生。1983 年早稲田大学理工学部建築学科卒業。
同年日本ユニシス(株)入社。グラフィック基本ソフトウェアや応用ソフトウェアの開発保守に従事。現在、エンジニアリングシステム開発室に所属。