

分散対話型マルチメディア・システムの実装

松崎 正敏

通信・放送機構 川崎リサーチセンター

〒210 川崎市幸区堀川町 66-20 川崎市産業振興会館3階

CORBAに準拠したオブジェクト指向技術を採用したSOMとその分散化を分散オブジェクト・モデル(DSOM)により実現し、その上に分散対応されたマルチメディア部品の複合文書パーツを配置して複合文書の流通を行う分散対話型マルチメディア・システムを実装したので報告する。

キーワード CORBA SOM DSOM 分散システム マルチメディア 部品

Interactive Distributed Multimedia System

Masatoshi Matsuzaki

Telecommunications Advancement Organization of Japan

KAWASAKI RESEARCH CENTER

66-20, Horikawa-cho, Saiwai-ku, Kawasaki-city, 210

We developed Distributed Multimedia System by SOM(System Object Model) and DSOM(Distributed SOM) on PC and Workstation.

SOM is Object Oriented Technology and DSOM(Distributed SOM) is it's distribution framework as OMG CORBA.

Multimedia Object Parts compose the OpenDoc Compound Document and can be located on any distributed server and synchronous update between server and clients.

Key words CORBA SOM DSOM Distributed System Multimedia Parts

1. はじめに

パーソナル・コンピュータとイントラネットの普及によりオフィスでのシステム化のパーソナライズが進む中で、従来の文字中心の文書から表や図・静止画をはじめとしたマルチメディア・データにより構成された複合文書(Compound Document)による情報の伝達

が一般的になってきた。

これに伴って効率的なオフィス・システムのために、その複合文書を構成する各々のマルチメディア・データの部品化と共有、有効利用が使い易いシステムとして不可欠な要素となり重要度が高まってきている。

一方、分散対話型のプラットフォームとし

て OMG^[2.1]の CORBA2.0 に準拠したシステムも表-1のごとく対応がすすみつつある。

今回の分散対話型マルチメディア・システムは、LAN/WANにより構成されているイントラネット上に分散対話型のプラットフォームとして SOM/DSOM、マルチメディア・パーツ操作のために OpenDoc^[2.10]を採用し構築してある。

このイントラネットにより構築されているオフィス・ネットワークの中でマルチメディア・パーツがどこに有るのかを意識することなく位置透過的に使用できると共に、常に最新のデータを更新に合わせリアルタイムに提供される様にする事で従来のクライアント/サーバ・システムを発展させた、クライアント・サーバを意識すること無くマルチメディア・データの部品を対話的に操作する事ができるシステム^[1.2]である。

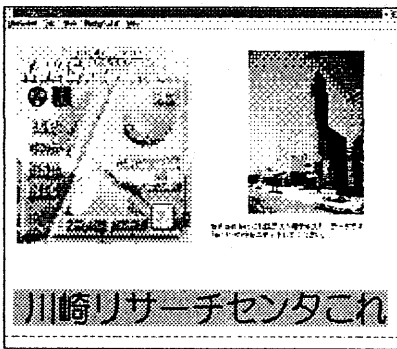


図1 複合文書 (例)

2. 分散対話型マルチメディア・システム

今回の分散対話型マルチメディア・システムはオブジェクト指向技術を採用して設計・開発をすると共に、更に開発言語等におけるオブジェクトの定義や動作に依存しないよう

Features	ORBs								
	Digital Object-Broker	Viaigenic ViaBroker	IBM SOM	Sun Joe/NEO	HP Orb Plus	IONA Orbis	Expersoft Power-Broker	ICL DAIS	
IOP	○	○	○	○	○	○	○	○	○
IR	○	○	○	○	○	○	○	○	○
Static	○	○	○	○	○	○	○	○	○
Dynamic	○	○	○	○	○	○	○	○	○
Language Bindings									
C	○		○	○	'97	'97			○
C++	○	○	○	○	○	○	○	○	○
Java	'97	○	'97	○	'97	○	○	○	○
Smalltalk	'97	○ (via DNS)	○		○	○ (via ParcPlace)			
COBOL		'96	○				○		'97
Ada							○		
CORBA services									
Naming	○	○	○	○	○	○	○	○	'97
Events	'97	○	'97	○	○	○	'97	○	○
Life Cycle	'97	○	'97			'97	'97	'97	○
Trader	'97	'97	'97			'97	'97	'97	○
Transactions	'97	○ (via thirdParty)	'97	'97	'96	○ (via thirdParty)	'97	'97	'97
Concurrency	'96	'97	'97	'97	'96	'97	'96		
Security	'97	'97	○	'97	'97	'97	'97	'97	○
Persistence	'96		'97	○	'96	○	○	'97	'97
Externalization	'96		○		'96	'96	'97		
Query	'96	'96	'96	○		'97	'96	'96	'96
Collections	'96	'96	'96			'97	'96		
Relationships	'96		'97	○		'97	'97		
Time	'96	'97	'96			'97	'96	'96	'96
Licensing	'96		'96			'97	'96	'96	'96
Properties	'96		'96	○		'97	'97	'96	'96

表1 TheCommercialORB

出典:InstantCORBA^[1.1]

にするためにシステム・オブジェクト・モデル(SOM)の上に構築した。

OpenDoc もこの SOM の上に構築されたアーキテクチャであり、またフレームワークである。

分散対話型マルチメディア・システムの稼働環境としては OS/2、AIX のプラットフォーム上に SOMObjects、OpenDoc を導入し開発言語として VisualAgeC++ と REXX を使用した。

また、イントラネットはネーム・サーバを保有し構築されていることを前提としている。

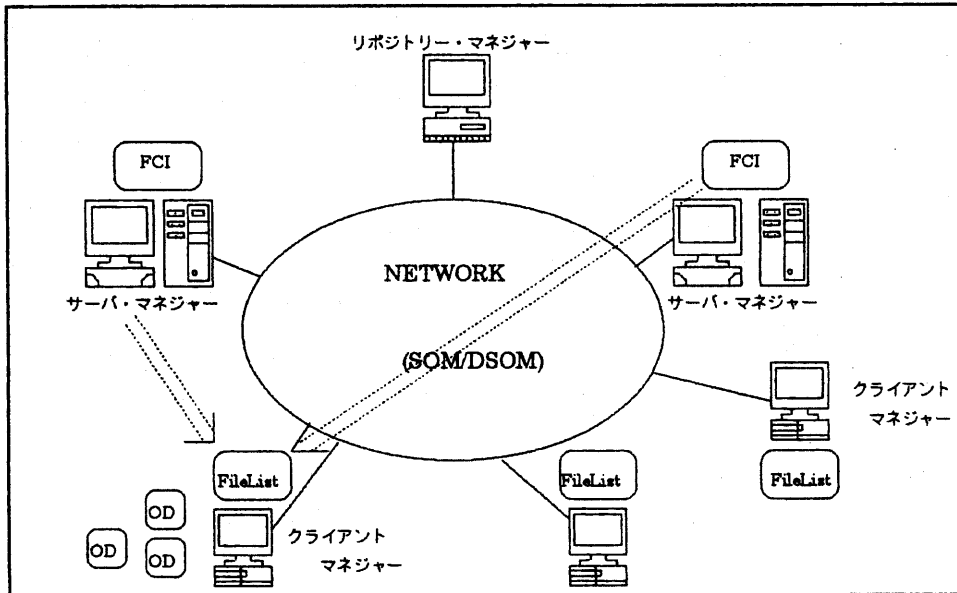


図2 分散対話型マルチメディア・システムの構成

2. 1 サーバ・システム

サーバ・システムは OS/2 または AIX 上で稼働し、ディレクトリー・サービスを行うシステムに1つのリポジトリ・マネジャーおよびマルチメディア・パーツのファイル・コンテンツ・インフォメーションを保有する複数のサーバ・マネジャーの2種により構築する。

サーバ・マネジャーは OS/2, AIX いずれの上でも稼働が可能であるが、リポジトリ・マネジャーは AIX 上でのみ稼働する。

この AIX 上のリポジトリ・マネジャー導入環境では、同時にサーバ・マネジャーを稼働させる事も設定により可能である。

2. 2 クライアント・システム

OS/2 上に OpenDoc を使用した分散対応マルチメディア・パーツの開発環境とクライアント・マネジャーが稼働して分散対応マルチメディア・パーツを用いた OpenDoc 複合文

書を作成する実行環境の2種類の環境を構築した。

3. 分散対話型マルチメディア・システムの構成要素

3. 1 リポジトリ・マネジャー

分散対話型マルチメディア・システムの中で唯一稼働し、分散オブジェクトである全てのサーバ・マネジャーと全てのクライアント・マネジャーのマシン・エイリアスあるいはホスト名とそれに該当するオブジェクト ID の情報を管理し、サーバ・マネジャーとクライアント・マネジャー間の通信を実現するためにサーバやクライアントの稼働場所情報、すなわちネームサービスを行い、この情報を提供する。

3. 2 サーバ・マネジャー

分散対応されたマルチメディア・コンテンツの保有・管理をする。

サーバ・システムの中で唯一稼働するが、複数のマシンで同時に稼働させることが出来る。

サーバ・マネジャー上には保有するマルチメディア・コンテンツの参照情報を管理するための FCI(FileContentInformation)と呼ぶ制御情報がある。

3. 3 FCI(ファイル・コンテンツ・インフォメーション)

各サーバ・マネジャー上にファイル形式として存在するマルチメディアのコンテンツが登録されたもので、物理的に1つの GIF ファイルや1つのテキスト・ファイル等の物理ファイルと一対一に対応している。

また、FCIはそのコンテンツをリンクにより参照しているクライアントに関する情報を管理する。

サーバ・マネジャー上では SMBATCH というコマンドを通じて、この FCI を登録・削除あるいは更新とそれに伴う更新通知の処理を行います。更新通知とは上記の物理ファイルが入れ替えられたり内容が替わったとき、これを参照中の分散対応 OpenDoc パーツに変更があった旨の通知を出すことを指す。

3. 4 クライアント・マネジャー

クライアント・マネジャーは1つのマシンで唯一稼働し、クライアント上の分散対応 OpenDoc パーツとサーバ上にあるマルチメディア・コンテンツをサーバ・マネジャーとの間で仲介を行う。そして分散対応 OpenDoc パーツと同様にサーバ自身もパーシステントなオブジェクトとして他のマシンへの移動も可能となっている。

クライアント・マネジャーは、システムの中のサーバにある FCI が登録・削除されても

常に最新のリストを保持し、更新通知によりそれを分散対応 OpenDoc パーツに渡す役目を果たしていると共に、分散対応 OpenDoc パーツからのリンク要求やリンク解除を、指定されたサーバ・マネジャーに転送する役目を果たしている。

また、クライアント・マネジャーの提供する API を直接使用すれば OpenDoc 複合文書の中だけでなく OS/2 の PM(Presentation Manager)のプログラムでも分散コンテンツが使用可能になっている。

3. 5 分散対応 OpenDoc パーツ

分散対応された OpenDoc パーツは、通常の OpenDoc のクラスにミックスイン・クラスの処理を付加することで、どの OpenDoc パーツからでも分散対応 OpenDoc パーツを作成できる。

分散対応された OpenDoc パーツは分散オブジェクトであるクライアント・マネジャーに対して、リンク要求を出したり更新情報を問い合わせたりしてネットワーク上の分散コンテンツ・ファイルを取捨選択したりロードできる一方、それらのコンテンツ・ファイルに変更があった場合には更新通知を受け取る事によりパーツの表示内容を変更する。

分散対応 OpenDoc としてリンクが張られているコンテンツは、クライアント・マネジャーによりサーバ上の実ファイルが取得されて一時的にクライアント上のローカル・ファイルとして作成される事により、クライアント上の分散対応 OpenDoc パーツはこの一時的に作成されたファイル名を得て読み込んで処理をする。

分散対応 OpenDoc パーツはリンク情報を含め、パーシステントなオブジェクトとして扱う事ができる。すなわち、他のマシンへ持

っていつでも更新通知を受け続ける。

分散対応 OpenDoc パーツのクライアント・マネジャーとの通信は、すべてミックスイン・クラス(MDistConnect)で実装されているのでサブクラスでは、ミックスイン・クラスで実装されているメソッドをコールすることで、クライアント・マネジャーとの通信が可能となる。

4 運用試行

4. 1 起動

全てのサーバ・マネジャーとクライアント・マネジャーは、リポジトリ・マネジャーと通信するためにリポジトリ・マネジャーのオブジェクト ID を取得する必要があるために 1)リポジトリ・マネジャー2)サーバ・マネジャー3)クライアント・マネジャーの順序で起動する必要がある。

ただし、サーバ・マネジャーがリポジトリ・マネジャーと同一マシンで稼働しているときには、特に立ち上げの順序やコマンドは必要無い。

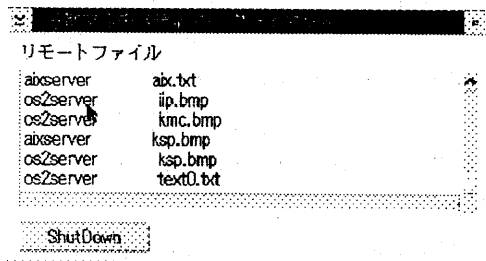


図3 分散対応された OpenDoc パーツの表示

クライアント・マネジャーは、分散対応 OpenDoc パーツが分散コンテンツを従来通りのファイル・アクセスと同様に扱えるようにするために一時ファイルを稼働しているマシン内に作成する。リポジトリ・マネジャ

ーのオブジェクト ID の取得後、クライアント・マネジャーを稼働することにより、各サーバー上に保有保有されている分散対応マルチメディア・コンテンツ情報がウインドウにより表示される。

4. 2 分散対応したマルチメディア・コンテンツの操作

4. 2. 1 登録・更新・削除

コンテンツを新規に登録したり、更新・削除するときにはサーバ・マネジャーの稼働しているマシンで下記にしめす様なリストを指定して、追加・更新・削除にあわせたコマンドを発行する。

リクエスト機能	ファイル名	種類	場所
ADD	File1.bmp	BIN	d:¥dir1
UPDATE	File2.txt	STR	d:¥dir2
REMOVE	File3.bmp	DUMMY	DUMMY

表2 コンテンツ操作要求リスト

4. 2. 2 OpenDoc パーツ

分散対応 OpenDoc パーツも通常の OpenDoc パーツと同じ操作が可能である。

ワークスペース上の"OpenDocTemplate"フォルダーから、OpenDoc パーツをドラッグして、ワークスペース上にドロップすると、ドキュメント(アイコン)が作成される。

このアイコンをダブル・クリックするとドキュメントがオープンされて、パーツが起動される。または、コンテナ・パーツである他のパーツの上にドロップすると自動的にパーツが起動される。

4. 2. 3 終了

起動する順序とは反対に 1)クライアント

ト・マネジャー2)サーバ・マネジャー3)リポ
ジトリー・マネジャーの順序で終了する。

クライアント・マネジャーの終了前にはリ
ンクしている全ての OpenDoc パーツを含む
複合文書をクローズしておく必要がある。

5 おわりに

今回研究室内の環境に CORBA に準拠し
た SOM/DSOM を使用してマルチメディア・
コンテンツを位置透過的に扱う機構を実装し、
各種のパーツを扱えると共に更新の同時反映
等の機能を確認すると共に、具体的な情報発
信に注目して適用分野の検討を実施した。

この結果容易に、分散環境の構築ができ
るとともにミックスイン・クラスを用いて分散
対応のパーツの作成が出来ることが実証でき
た。また作成したパーツを用いて、ローカル
に保有していない情報であっても、情報の階
層を意識せずに利用が可能であり、人手を介
さずに常時表示しておくようなウエルカム画
面に対する情報の即時反映のような、ブロード
キャスト処理等に応用が可能であることを
確認した。

また今回の実装による試行からでは
CORBA に準拠した開発環境・実行環境を前
提としたとしてもコンテンツの分散による、
ローカル開発テストと分散環境下の実行環境
の存在がマルチメディア・コンテンツの流通
のためのコラボレーションに対し有効で有る
ことを十分に実証することは出来なかった。

今後は、WAN により構成されている規模
の大きなイントラネットにおける更新処理の
反映等に関する定量的な評価や、CORBA に
おける LanguageBinding の Java への拡張を
ふまえ異なるメーカー間のプラットフォーム
への拡張への対応をふまえた試行による検証
が考えられる。

謝辞

今回の報告にあたりご指導いただいた法政
大学廣瀬克也教授、ご支援いただいた日本ア
イ・ビー・エム (株) 鈴木俊宏氏、北条眞氏、
古澤修氏に感謝いたします。

1. 参考文献一覧

- [1.1]RobertOrfali 他,313P,Instant CORBA,
WileyComputerPublishing,1997
- [1.2]分散対話型マルチメディアシステム解説書,119P,
日本アイ・ビー・エム,東京(1996)
- [1.3]小野沢 博文,250P,分散オブジェクト指向技
術,ソフトリサーチセンター,東京(1996)
- [1.4]酒井皇治,250P,OpenDocプログラミング入
門,技術評論社,東京(1996)

2. 参考サイト

- [2.1]<http://www.omg.org> OMG
- [2.2]<http://www.digital.com/info/objectbroker>
Digital Equipment Corporation
- [2.3]<http://www.visigenic.com>
VISIGENIC software
- [2.4]<http://www.software.ibm.com/ad/somobjects> IBM
- [2.5]<http://www.sun.com> Sun microsystems
- [2.6]<http://hpcc920.external.hp.com>
HEWLET PAKARD
- [2.7]<http://www.iona.ie> IONA Technologies
- [2.8]<http://www.expersoft.com> Expersoft
- [2.9]<http://www.icl.com> ICL
- [2.10]<http://opendoc.macintosh.net>
Macintosh