

## 協同同期型会議通信のための開放型遠隔文書操作機能 (ORDE)の提案および実装評価

中尾 康二      田中 俊昭      鈴木 健二

国際電信電話株式会社    研究所

あらまし    分散会議システムの環境においては、会議の題材となる「文書」を単なる静止画や動画としてではなく、ワープロで作成された「電子的文書」として、その段落や図表に至るまで操作/編集を同期的かつリアルタイムに行うことができる協同同期型会議通信が必要となる。本論では、協同同期型会議通信を実現する開放型文書操作方式を提案し、その機能の実証を行なう。具体的には、開放型文書操作方式の通信機能としてORDE(Open Remote Document Editing)をOSI(開放型システム間相互接続)環境において提案/設計し、さらに、パソコン上にORDEを用いた会議システムを実装し、ORDEの実現性、有用性を実証する。

和文キーワード    ORDE, 開放型文書操作方式, OSI, 協同同期型会議通信

## Proposal and Evaluation of Open Remote Document Editing (ORDE) for Cooperative Synchronous Conference Communication

Kouji NAKAO, Toshiaki TANAKA and Kenji SUZUKI

KDD Research and Development Laboratories

2-1-15, Ohara, Kamifukuoka-shi, Saitama, 356

**ABSTRACT**    In the distributed conferencing applications, it is required to study on Cooperative Synchronous Conference Communication (CSCC) which is able to provide synchronous real time manipulation / editing of documents up to the content level (paragraph, figure). In this paper, the concept of Open Document Manipulation for CSCC is firstly clarified. Then, based on the concept, the new communication function so-called Open Remote Document Editing (ORDE) is proposed and evaluated by implementing an experimental system. Consequently, it is basically clarified that ORDE will provide sufficient functions to realize the Open Document Manipulation for the CSCC environment.

英文 key words    ORDE, Open Document Manipulation, OSI, Cooperative Synchronous Conference

## 1. はじめに

近年の情報処理技術および通信処理技術の発展に伴い、各種情報通信システムの開発が進められており、そのひとつに「電子会議システム」がある。電子会議システムにおいては、会議資料発表、協同文書執筆、電子契約など多様な応用があり1,2,3)、これまで主に静止画情報、ビデオ情報や手書き情報を用いたシステムの構築がなされてきた4,5,6,7)。しかしながら、分散会議システム的环境においては、会議の題材となる「文書」を単なる静止画や動画としてではなく、ワープロで作成された「電子的文書」として、その段落や図表に至るまで操作/編集を同期的かつリアルタイムに行うこと(協同期型会議通信と呼ぶ)が必要となる8)。

これまでの協同期型会議通信のための文書操作機能の研究には、大別して2つのアプローチが存在する。一方は、それぞれの分散システムにおいて全く同種のテキスト/グラフィックエディタを同一環境で動作させ、マウスやキーボードによるイベント情報(カーソル位置、キー入力等)を直接交換することにより、分散システム間において全く同じ文書処理環境を提供する方式「イベント方式」と呼ぶ)である8,9,10)。

他方は、会議システムで利用するワープロ、およびウィンドウシステムなどに依存せずに、会議の対象となる「文書」に対して構造化処理(ODA: 開放型文書体系11)などを施し、その構造化要素(例えば、段落や図)を遠隔より開放型プロトコルで一元的に検索、更新および編集する方式(「開放型文書操作方式」と呼ぶ)である12)。

イベント方式は、同一の文書処理システム(ワープロなど)を用いて簡易に実現できるといった特徴がある。しかしながら、文書処理システムは、エディタ機能、表示スペース、表示タイミングなどが機種毎に異なっているため、異種システム間の相互接続は難しく、たとえ同種のシステムを用いた場合でも、そのシステムのためのイベント処理機能の開発が必要となる。

一方、開放型文書操作方式では、利用する文書フォーマットがそれぞれ異なっている、異種文書間の橋渡しとして中間的な文書構造(ODAなど)を用いることにより、通信のために互いに共通な「文書構造」を保有し、その文書構造を単一な通信機能により操作するといった「文書操作に関わる開放型システム」を実現することができ、異種システム間の相互接続の問題を解決できる。

開放型文書操作方式を実現するためには、文書構造化、および通信機能(プロトコル)の研究が重要であり、文書構造化に関しては、近年、各方面で研究開発が進められている13,14)。しかしながら、開放型文書操作方式のためのプロトコルについては、文書操作方式の概念、および詳細な通信機能が明確になっていないため、文書構造化と組み合わせた総合的な開放型文書操作方式の実現性、有用性が明らかになっておらず、今後の研究が必要とされてきた。

そこで、本論では、電子会議システムの各種応用で重要となる協同期型会議通信のための開放型文書操作方式を提案し、評価システムを作成することによりその機能の実証を行なう。具体的には、開放型文書操作

方式の通信機能としてORDE(Open Remote Document Editing)をOSI(開放型システム間相互接続)環境において提案/設計し、さらに、パソコン上にORDEを用いた会議システムを実装し、ORDEの実現性、有用性を実証する。まず、2.において開放型文書操作方式の概念、および実現のための要求条件を述べ、3.において開放型文書操作方式を実現するORDEを設計する。さらに4.では、ORDE実装方針、具体的なORDEの実装手法および各種ORDE機能の評価について述べる。最後に、5.でこれらを考察する。

## 2. 協同期型会議通信のための開放型文書操作方式

### 2.1 協同期型会議通信

協同期型会議通信とは、双方の会議システムで同一の文書群を物理的にそれぞれ保有し、それら文書(またはその一部)を互いに同期的かつリアルタイムに協同で操作(検索、更新、編集)する通信形態と定義する。本通信では、会議文書の一貫性を保つために、双方の会議システムが文書の操作を同時に実行する非同期型導入を禁止している。しかしながら、操作権の概念を導入することにより、各会議システムが平等に文書操作に参加できる形態をとる。但し、ある一時点においては、文書操作を行なっている会議システムは一つとする。本形態の例としては、複数のクライアントにより同一の内容を審議折衝する協同文書作成や電子契約などが挙げられる。

### 2.2 開放型文書操作方式の概念の明確化

開放型文書操作方式とは、遠隔に存在する文書サーバをクライアント(操作者)がアクセスする通信形態において、サーバに存在する開放型文書の構造要素(例えば、段落や図)をクライアントが開放型通信機能により部分検索、更新、編集する通信方式であると定義する。ここでサーバ上の開放型文書は、クライアントとの間で文書構造化に関して相互理解の可能な文書構造を指

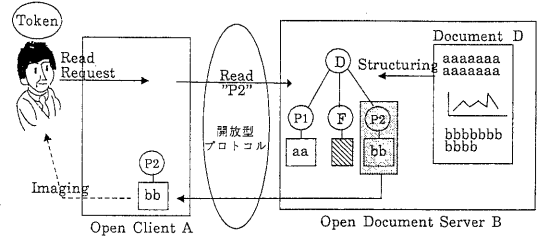


図1(a) 文書部分検索の例

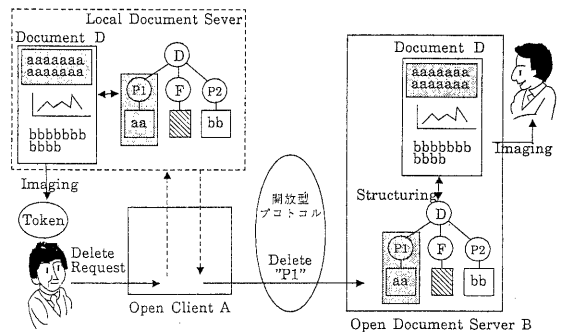


図1(b) 文書部分編集の例

図1 文書操作方式の概念図

し、開放型通信機能は、開放型文書の構造の操作を司る単一な通信プロトコルで、開放型文書の種別に依存しないものとする。尚、ここで「操作」とは、段落や図などの文書の内容に関わる総合的な検索、更新、編集機能を目指す。

具体的には、クライアントは文書の一部分の検索や編集を行なうために、文書の構造要素の識別子(例えば、段落の番号)を用いて操作の対象となる部分の指定を行い、図1(a)/(b)の例で示すように、それぞれ文書の部分検索(段落P2部分の読み出し)、および文書内容の部分編集(段落P1部分の削除)を実行する。クライアントとサーバの間で交換されるRead "P2", およびDelete "P1"などの手順要素は、開放型プロトコルにより構築する。尚、本方式は、「文書」を編集操作の対象としているため、テキスト情報のみならず、図形、イメージなど文書として包含される情報メディアをすべて操作の対象とすることが可能とする。

協同期型会議通信においては、図2で示す通り、各々のシステム(Sub-System)はクライアント機能とサーバ機能を保有し、ある一時点をみると、一方のクライアントAが他方のサーバBにアクセスする形態をとる(図2上)。この際、クライアントA側のシステムでは、文書操作(例えば、文書D部分Cの編集)をシステムB(サーバB)に要求すると同時に、要求した文書操作と全く同一の処理を自分のシステム内でローカルに実行する。また、操作権(トークン)の移動を行うことにより、クライアントBも同一の文書及び同一の文書内容の操作が可能となる(図2下)。結果的に本通信では、双方のシステムが協同で同一の文書内容の操作を会議形式で実行できる。

## 2.2 開放型文書操作方式の実現のための要求条件

開放型文書操作方式を実現するには、異種システム間の通信を意識させずに行なう必要があり、既存の文書処理システムを生かしたシステム設計を行う必要がある。従って、既存の各種文書処理システムがもつ共通性を抽出しそれらを活用するとともに、開放型文書

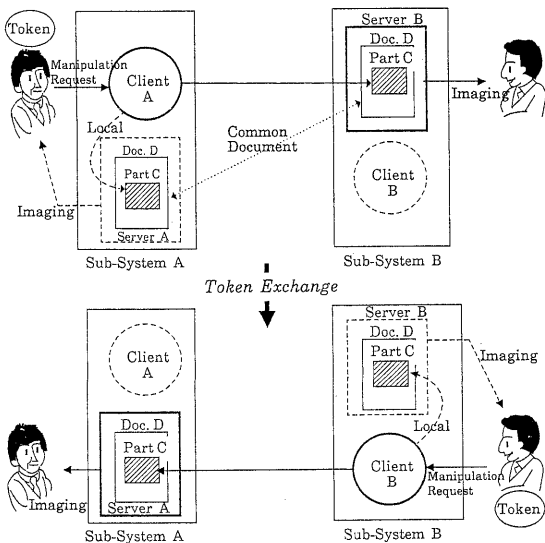


図2 協同期型会議通信における開放型文書操作方式

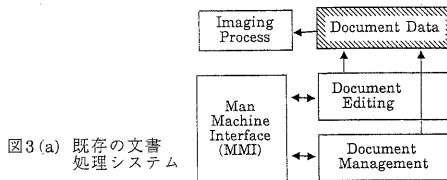


図3(a) 既存の文書処理システム

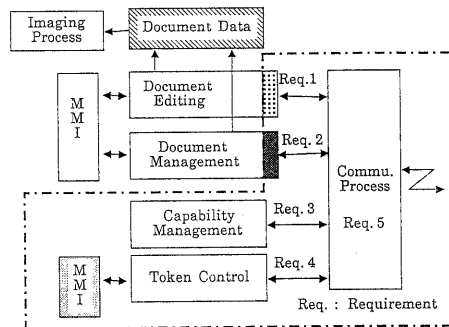


図3(b) 文書操作方式のための文書処理システム

図3 文書処理システムの機能構成

操作方式のために要求される新たな機能について以下に整理する。

既存の文書処理システムは、図3(a)で示す通り、文書操作機能(文書の作成、修正、削除など)、文書管理機能(文書オープン/クローズ、文書の作成/削除など)、文書処理/文書操作の結果を表示する可視化処理機能、および利用者とのインタフェースを提供するMMI(マンマシンインタフェース)機能により主に構成される。

既存の文書処理システムを利活用する形態で開放型文書操作方式を実現するためには、図3(b)の一点鎖線で囲まれた領域の機能を拡張、新規追加する必要がある。拡張および新規追加する機能を以下の要求条件(要件: Req)において明確にする。

(要件1): 既存の文書処理システムが扱う文書は、文書構造、およびその構造要素の管理体系(識別番号の付与方法など)がそれぞれ異なっているため、開放型文書操作方式を実現するには、全く同種の文書を用いるか、あるいはODA等の中間的な文書構造を介して通信をするかのいずれかとなる。従って、すべての文書種別の利用を実現するために、文書種別に依存しない文書操作機能の構築が必要である。

(要件2): 文書のオープン/クローズなどを行なう文書管理機能は、各種文書処理システムにおいて用いられており、機能的には共通部分が多い。本文書操作方式に基づく通信においては、遠隔にある文書のオープン/クローズ、および文書登録/削除/複写に関する操作オペレーションが必要となり、これらの機能を提供する一元的なプロトコルが必要である。

(要件3): 遠隔に存在する文書の一部分を識別するために、文書を構造化し、その構成要素(段落や図)を識別の単位とする。クライアントはサーバの保有する文書構造に関する知識が必要であり、実際の文書操作を開始するまでに、クライアントとサーバの間で使用する文書種別を決定する。従って、使用する文書種別を決定

するための文書操作能力の指示/発動機能を保有する必要がある。

(要件4)：各会議システムが平等に文書操作に参加できるように、クライアントが保有する文書操作権を相手システムに移動する機能が必要となる。すなわち、クライアントとサーバの関係を交代する操作権管理機能を提供する必要がある。

(要件5)：相互接続性を考慮し、開放型文書操作方式を実現する文書操作、文書管理、および操作権管理などの通信機能を、OSI通信環境における通信機能(サービス/プロトコル)として構築することが必要である。

(要件6)：人間が文書操作(検索, 更新, 編集)をリアルタイムに行なうためには、ひとつの操作から次の操作までの待ち時間を極力少なくし、スムーズな文書操作を実現することが必要である。

### 3. 開放型遠隔文書操作機能(ORDE)の設計

2.で述べた開放型文書操作方式の概念、および要件を満足する通信機能として、以下に開放型文書操作方式をOSI環境において提供する「遠隔文書操作機能(ORDE:Open Remote Document Editing)」を設計する。

#### 3.1 ORDEの提供する通信機能

ORDEは、以下の通信機能を提供する。

##### (1) 文書操作機能 (要件1)

###### ● 文書操作オペレーション

既存の各種文書処理システム(ワープロなど)に搭載されている文書操作オペレーションは、部分検索(探索)、マーカーセット(操作対象の位置表示)、保護/非保護(アクセス禁止/解除)、作成、削除、属性修正、複写/移動/置換にほぼ集約でき、これらを解析すると、すべてのオペレーションは、

```
[コマンド名][何を][どこへ*][どのように*]
```

```
[どこから*][何と*] (*はオプション)
```

のパラメータの組み合わせによって表現されている。

ここで[コマンド名]とは、作成、削除、修正、部分検索などのオペレーション種別を指し、[何を]とは、コマンドの対象となる操作対象(DMO)でいずれも必須パラメータである。また、[どこへ]は移動/複写先を、[どのように]は属性の変更内容を、[どこから]は検索/探索などの範囲指定を、更に[何と]は置換の代替物をそれぞれ示す。従って、これらのパラメータを用いて汎用的に操作オペレーションを規定することにより、文書種別に依存しない機能の設計が可能である。

###### ● 操作対象の識別手法

上記操作オペレーションの引数である[何を][どこへ\*][どこから\*]は、一つの文書内の場所を示す識別情報であり、ここでは文書構造のオブジェクト(DMO:段落や図など)を用いてその場所の識別を行う。しかしながら、既存の文書処理システムの保有する文書構造や各種国際標準で規定される文書構造(ODA, MHEG, SGML)はそれぞれ異なっているため、一元的な記法に従ったDMOの表現は困難である。従って、ORDEにおいては、図4で示すように、対象となる文書操作オブジェクト(DMO)の記述に汎用性を持たせ、文書種別に依存しない形で文書構造のオブジェクトを運べるような記述を採用した。

##### (2) 文書操作管理オペレーション機能 (要件2)

```
ManipulationObject ::= SEQUENCE OF SEQUENCE
{
  documentIdentifier [0] Document-Id,
  start-object [1] Manipulation-Object-Id,
  end-object [2] Manipulation-Object-Id OPTIONAL,
  subordinate-flag [3] IMPLICIT BOOLEAN OPTIONAL
}
-- T:: = including subordinate objects,
-- F:: = not including subordinate objects.

Document-Id ::= OCTET STRING
Manipulation-Object-Id ::= SEQUENCE OF ObjectValue
ObjectValue ::= CHOICE {
  single-ASN.1-type [0] ANY,
  octet-aligned [1] IMPLICIT OCTET STRING,
  arbitrary [2] IMPLICIT BIT STRING
}
```

図4 文書操作オブジェクトの汎用的な記述例

既存の文書処理システムにおいて利用されている文書のオープン/クローズ(文書選択/解放も含む)、文書の削除、複写、および文書登録の機能を、遠隔に存在する文書サーバに対して実行できるようにORDEを規定する。このためには、文書操作管理のためのオペレーション種別と文書識別子をORDEのパラメータとして保有する必要がある。また、本オペレーションを確実に実行するために、クライアント側のオペレーション処理は、サーバ側の処理が終了し、確認応答を受けとった後に実行する。

##### (3) 操作能力の指示/発動機能 (要件3)

操作能力の指示/発動の手段としては、通信開始時に、クライアントが保有する操作能力をサーバに対して指示し、サーバは自分が可能とする文書操作能力をその応答として返答することにより、クライアントは文書操作を開始する前にサーバの文書操作能力を理解する。クライアントは、この能力結果を用いてサーバに要求する文書操作能力を選定し、具体的な文書操作を開始する時点で、選定した文書操作能力の発動を行う。

##### (4) 操作権管理オペレーション機能 (要件4)

同期制御を行うために、操作権の概念を導入する必要がある。操作権は、実際に文書操作を実行するクライアントに対して与えられるもので、複数利用者による文書の同一箇所での操作(文書操作の衝突)を排除し、交代制で文書操作を実現する手段である。ここで想定する通信形態においては、会議システム間で共有する文書を互いのシステムで物理的に保有し、文書操作の結果を互いのシステムの文書に反映させていく形態をとるため、文書操作の衝突は極力避けることが要求される。従って、操作権を保有するクライアント側の人間が直接操作権の移動を起動する方式が最も信頼性が高く、本通信形態に適している。尚、本方式では、図3(b)で示すように操作権を起動するためのMMIが必要であり、操作権を移動する場合は、操作権を保有しているクライアントが基本的に移動先を決定し、その移動を実行する。さらに、操作権を保有していないサーバ側の利用者が文書操作を要求する場合は、操作権譲渡の要求をクライアントに送信することも可能とする。

#### 3.2 OSI環境におけるORDE (要件5)

##### (1) ORDE通信モデル

開放型文書操作方式の概念を考慮すると、ORDEの通信モデルとしては、図5に示すように、単一のクライ

アントがひとつのサーバに接続するポイントツーポイントの通信形態において、サーバ側の文書の内容をクライアントが操作する通信機能をORDEサービスとして提供する通信モデルが抽出できる。

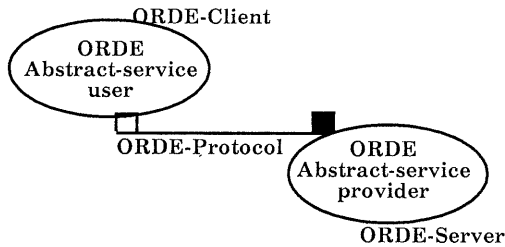


図5 ORDE通信モデル

(2) OSI環境におけるORDEの位置づけ

多様なOSI通信应用からORDEで規定する通信機能を汎用的に利用できるようにするため、ORDEが提供する通信機能をひとつのOSI应用層ASE(ORDE-ASE)として位置づけ、単一应用コンテキストとして文書操作機能のモジュール化を行うこととする。また、ORDEにおいては、文書操作の確認応答(操作の成功/失敗)が必要であるため、セッション全2重機能を用いる。さらに、できるだけORDE-ASEの機能を軽減し、既存の通信機能を活用することとし、操作オペレーションの要求とその確認応答といった手順要素については、DOAM(Distributed Office Application Model)で採用されているRO(遠隔オペレーション)サービスを利用することとする。従って、基本となる应用コンテキストは、ORDE-ASE/ROSE/ACSEとなる。

ORDEは、OSI参照モデルに従い、ORDE利用者、およびORDEサービス提供者に区分される。ORDE利用者は、ORDE通信モデルのクライアントとサーバにより構成され、それぞれのORDE利用者に対してORDEサービス提供者により以下に示すORDEサービスが提供される。

(3) ORDE抽象サービス定義

表1 ORDE 抽象サービス

サービスタイプ	抽象サービス名	
アソシエーション管理	Initiate service (操作能力指示を含む) Terminate service	
文書操作管理 オペレーション	Doc-Delete / Doc-Copy / Doc-Save services	
	Doc-Open / Doc-Close services	
操作 オペレーション	読み出し型	Search service Point(マーカ-セット)
	変型	Create / Delete / Modify / Replace Copy / Move / Lock(保護 / 非保護)
操作権管理	Token-Give / Token-Please services	

ORDEが提供する文書操作に関わる抽象サービスを表1に示す通り規定し、それぞれをROSEマクロに従って記述する。一例として、図6においてODA文書を用いた「削除オペレーション(Delete)」の実現例を示す。本例においては、クライアントはサーバ上のODA文書に対し、連続した3つのパラグラフ(Para.)を削除する要求を行っている。一点鎖線で囲まれた部分がODA文書の構成要素を識別する記述規定(Identification of Document Fragments)である15)。また、削除領域としては開始段落(30)と終了段落(31)を指示しており、これらの情報をORDEで規定する削除オペレーション(Delete Operation)を用いて伝達している。サーバは、操作の結果を削除結果(Delete Result)としてクライアントに返答する。

(4) ORDEプロトコル仕様

ORDE-ASEは、下位通信機能としてROサービスを利用するため、ORDE抽象サービス定義はROSEマクロにより記述される。従って、ORDEプロトコル仕様としては、各抽象オペレーションのプロトコル識別子を設定するのみであり、ここではDOAM環境で利用されているROSE上の各種通信と重複しないプロトコル識別子を付与した。

4. ORDEの実装および機能評価

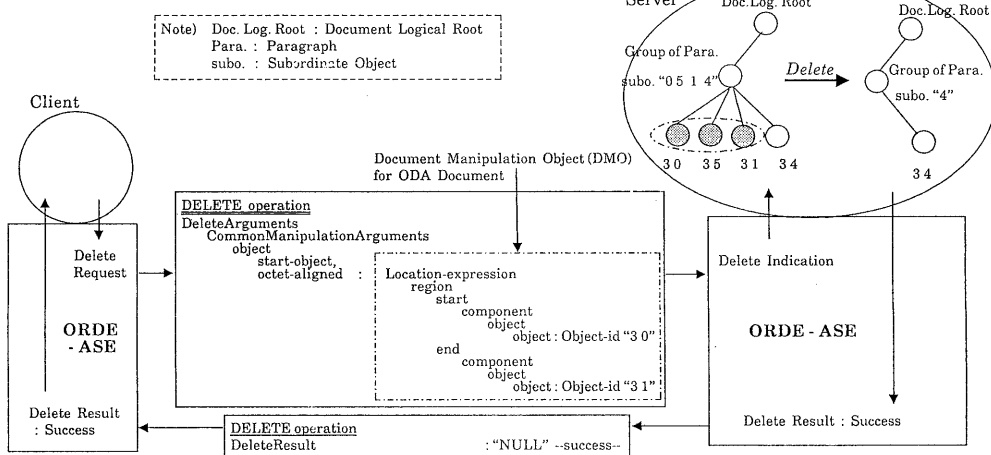


図6 ODA文書に対する「削除オペレーション」の実現例

#### 4.1 実装方針

(1) ORDEの各種機能を最大限に利用する会議用協同期型形態(図2)を本実装の通信形態とする。基本機能の検証を目的とするため、2台のパソコンでポイントツーポイント通信を実現し、音声を用いた会話手段も用意する。

(2) 文書処理システム(ワープロ等)の多様性を考慮して、中間的な文書構造を用いたORDEを実現し、異機種間のワープロソフト間でのORDEの可能性を検証する。中間的な文書構造としては、ODAを採用する。

(3) ORDEサービスが適用される通信ネットワークの多様性を考慮し、文書操作を実行する部分(ORDEの主機能)とROSE/ACSE以下の下位通信機能部を独立させる。

(4) 実装するORDEを、通信機能モジュール毎に独立に作成し、通信機能の拡充や変更への対処を容易とし、それぞれのモジュールの高速処理化を目指す。

#### 4.2 ORDEのモジュール構成

4.1で述べた設計方針に従い、3.で設計したORDE機能を、ORDE利用者部、ORDE-ASE部、および下位通信機能部の機能モジュールを用いて実装した(図7)。また、すべての機能モジュールをソフトウェア(C言語)で実現した。

##### (1) ORDE利用者部

ORDE利用者部は、文書サーバとクライアントの機能の双方を保有し、システムが操作権を持っている場合はクライアントとなり、持っていない場合はサーバとなる。尚、クライアントとサーバにおける操作権の有無を示すマークを、図8で示すように「鉛筆とマイク」を用いて表現した。本機能部は、以下の①利用者ワープロ処理部、および②ORDEインタフェース部の2つのモジュールにより構成される。

利用者ワープロ処理部は、既存のワープロのインタフェースをそのままの形で利用者に提供し、ワープロとODAの文書構造との対応付けの処理を実行する(13)。また、操作権の移動に関する利用者インタフェースを既存のワープロに対して拡張し、利用者が直接操作権の移動、要求ができるようにした。

ORDEインタフェース部は、利用者ワープロ処理部とORDE-ASE部との橋渡しを行う部分で、API部、

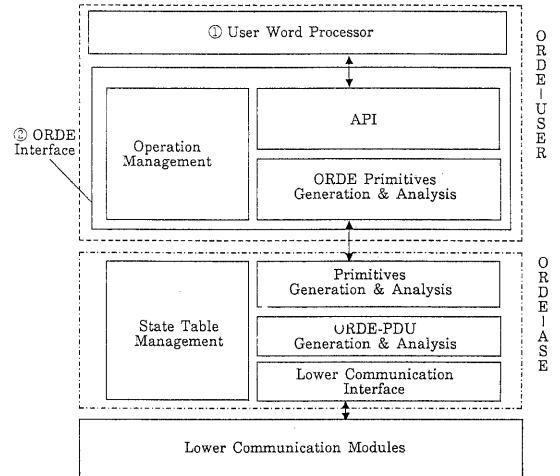


図7 ORDEの機能モジュール構成

ORDEプリミティブ生成/解析部、およびオペレーション管理部により構成した。

API部は、操作者(人間)により操作されたオペレーション(移動、削除など)をORDEによって伝達する情報に翻訳したり、逆にORDEによって伝達された情報を利用者ワープロ処理部へマッピングするための応用プログラミング・インタフェースをC言語の構造体を用いて提供する。

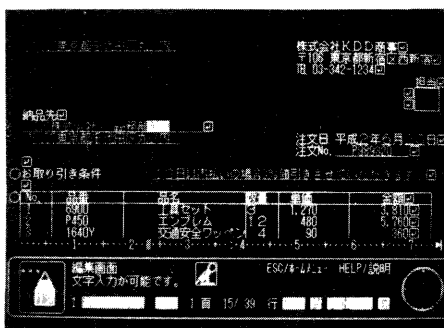
ORDEプリミティブ生成/解析部は、ORDE-ASE部と送受信するサービス・プリミティブの生成と受信したプリミティブの解析を行う。

オペレーション管理部は、操作オペレーション、文書操作管理オペレーションの管理、操作能力の指示・発動、および操作権の管理を実行する。

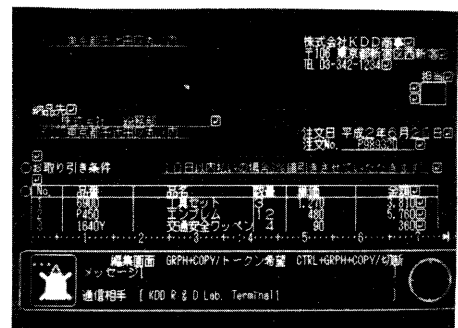
##### (2) ORDE-ASE部

ORDE-ASE部は(1)プリミティブ生成/解析部、(2)ORDE-PDU生成/解析部、(3)状態遷移管理部、(4)下位通信機能対応部というモジュール構成をとった。

プリミティブ生成/解析部は、ORDEインタフェース部より発行されるサービス・プリミティブの解析や、受信したORDE-PDU(プロトコル・データ単位)の情報からサービス・プリミティブの生成を行う。



Client



Server

図8 ORDEシステムにおける文書操作例

ORDE-PDU生成/解析部は、プリミティブ生成/解析部との対応をとり、PDU生成/解析のための処理を行う。

状態遷移管理部は、ORDEプロトコルマシンの状態遷移を管理し、サービス・プリミティブの順序制御、不正PDU受信などのチェックを行う。

下位通信機能対応部としては、どのような下位通信インタフェースにおいても動作できるように、ROSEおよびACSEのサービス・インタフェースを用いることとした。このことにより、ネットワーク(ISDN, LAN, X.25, 公衆電話網など)に依存しないORDEサービスが実現できる。

### (3) 下位通信機能部

下位通信機能部としては、ROSE/ACSE/プレゼンテーション機能を実装し、セッション機能以下は、ISDNボードおよびLANボード(CSMA/CD型)を用いた。ISDNボードは音声/データからなる2つのBチャネルを提供するため、音声による会話もボードを用いて実現した。一方、LANボードにおいては、音声をインバンドで利用できないため、音声用の回線をORDEとは別に用いた。

## 4.3 ORDE機能の評価

### 4.3.1 評価手法

実装したシステムをINTEL 80386(CPU)、16MHz(クロック)、MS-DOS(OS)において動作させ、3.で述べた操作オペレーション(作成、削除、修正)、文書操作管理オペレーション(文書オープン)、および操作権管理オペレーションをそれぞれ実行させ、各機能モジュールにおける処理時間を測定した。また、操作能力指示/発動についてはその基本動作の確認を行った。なお、下位通信機能部の通信ボードとしては、ISDNボードおよびLANボードの2種類を用いた。

### 4.3.2 評価結果

#### (1) 文書操作オペレーションの評価

ORDE利用者部(利用者ワープロ処理部)、ORDE利用者部(ORDEインタフェース部)、ORDE-ASE部、および下位通信機能部において、作成、削除、修正オペレーションに関する各所要時間(各機能部で処理に費やした時間)を測定した。なお、処理時間の最小測定単位を0.025秒としたため、すべて0.025秒単位の切り捨てとなる。測定結果から、文書操作オペレーションに関して以下の評価結果が得られる。

(ア) 操作オペレーションの種類に依存せず、ORDEインタフェース部、およびORDE-ASE部の処理時間は、平均して0.1秒~0.15秒であり、他機能モジュールと比較して処理負荷が少なく、高速処理が実現できた。

(イ) 下位通信機能部としてISDNボードとLANボードを利用した場合を比較したが、全体的な処理効率においてほとんど差が見られなかった。

(ウ) 作成、削除、修正を実行する情報量(例えば、文字数が少ない場合(10文字前後)は、全体の処理時間は1.2秒前後となり、操作者による連続的な文書操作において支障がないことが分かり、(要件6)を基本的に満足することが言える。一方、400文字の作成オペレーション(他文書からのコピー)を実行すると、全体の処理時間が3.5秒程度になり、これは主に利用者ワープロ部に全体の70%以上の処理負荷が集中することが分かった。

#### (2) 文書操作管理オペレーションの評価

クライアントから文書オープンオペレーションを実行させ、相手遠隔サーバ上の文書を開く操作を行った結果、ローカルで文書オープンを実行する時間(x秒)と比較して、遠隔の文書のオープンが終了するまでにかかる時間は、平均して(2x+0.6)秒かかることが分かった。尚、xはオープンする文書サイズに依存する。

ここでもORDEインタフェース部、およびORDE-ASE部の処理時間は、平均して0.05秒と短い。また、全体として要するオープン処理時間はローカル処理のほぼ2倍であるため、実用上問題がないと考える。

#### (3) 操作権管理の評価

クライアントから操作権の移動を施行し、サーバに操作権が移動するまでの時間を測定すると、利用者への表示処理を含めて約1秒であった。従って、操作権移動により連続的な文書編集が妨げられることは極めて少ないと言える。

本評価システムにおいて、操作権を移動する手段を整理すると、①クライアントの意志で移動する場合、②サーバ側の利用者がORDE機能により要求する場合、③音声によりサーバ側の利用者が要求する場合の3通りの利用が可能である。それぞれの評価を行ったところ、手段①および③ではクライアントが決断してから約1秒で操作権が移動されるため、実用上問題がない。手段②については、音声回線がない場合の救済手段としては価値が見いだせるが、音声回線がある場合は、操作権の要求(Token-Please)後にその移動(Token-Give)がなされるため、手間がかかり有用でないことが分かった。

#### (4) 操作能力指示/発動の評価

通信接続時の操作能力指示、および操作能力発動機能は正常に動作していることを確認した。また、本指示/発動のためのORDEプロトコルパラメータは数バイトであり、通信負荷はほとんどない。

## 5. 考察

開放型文書操作方式を用いた分散型リアルタイム系協同文書処理機能を実現するために、文書構造の操作を行う「通信機能:ORDE」の設計、およびORDE機能の評価するために、ORDEの実装/機能評価を行った。以下に、設計手法、実装手法、および処理効率(応答性)の視点からORDEの有効性、実用性を考察する。

### 5.1 ORDEの設計手法

(1) 本論で設計したORDEは、既存の文書処理システムに対して機能拡張、および機能追加を行なうことにより、文書操作オペレーション、および文書操作管理オペレーションを一元的に規定でき、操作能力の指示/発動機能、および操作権管理機能をOSIの通信環境上に体系的に構築することができたため、(要件1)から(要件5)を基本的に満足できたとと言える。

(2) 各種ワープロが規定する文書情報のフォーマットに介入せずに、文書操作の対象となる文書操作オブジェクト(DMO)をオクテット列として処理しているため、同種のワープロであれば直接的な文書操作サービスを実現することができる。さらに、実文書情報をODA文書やその転送構文に対応付けることができた。

ば、異機種のワープロ間での文書操作サービスが実現できる。

## 5.2 実装手法

本実装においては、通信機能をORDE利用者部、ORDE-ASE部、および下位通信機能部といった機能モジュールに明確に分け、互いの機能モジュールが独立した機能を提供できるようにした。このような実装手法においては、以下のような利点が生まれる。

(1) ORDE利用者部のワープロ機能が提供するローカルな操作オペレーションを一元的に規定したORDE操作オペレーションに対応付け、ORDE通信プロトコルとして一貫性のある通信インタフェースをORDE利用者部に提供できる。

(2) ORDE-ASE部と下位通信機能部とのサービスインタフェースをROSE、およびACSEとしたことにより、本評価システムで検証したようにISDN、LANなどの幅広い下位通信機能の上に本操作機能を構築することができる。

(3) ORDE-ASE部は、C言語で記述されており、ORDE利用者部に対してC言語の構造体ライブラリの形で各操作インタフェースが提供されているため、アーキテクチャの異なるシステムへの移植性が高いと言える。具体的には、PC-9801シリーズ、VAX (VMS)などで基本動作の確認を終えている。また、プログラム・サイズもORDE-ASE部で20K(行)と小型システムに移植可能なサイズで構築することができた。

## 5.3 処理効率

(1) (要件6)に関連して、各種トランザクション処理における人間の許容待ち時間は、通信を介した処理を伴う場合は2~3秒であるとの結果が報告されている<sup>16)</sup>。従って、4.における測定結果、および機能評価から判断すると、本稿で設計したORDEインタフェース部、およびORDE-ASE部は、非常に高い処理効率を提供でき、(要件6)を満足できると考えられる。

(2) 通常の文書操作処理では、操作する文書情報の量はあまり多くない(数文字、数行程度)と考えられる。しかしながら、扱う情報量が増加した場合、例えば400文字の作成操作においては、他の操作に比べてORDE利用者部の利用者ワープロ処理部の処理負荷が増加しており、全体的な応答性は低下せざるを得ない。このことは、要求する文書操作の情報量が増加した場合、2~3秒といった応答性は確保できないことを示唆している。従って、全体の応答性の向上は、利用者ワープロ処理部の処理効率に大きく依存しており、ローカルなワープロ機能の処理の応答性を超えることはできないため、達成できる処理効率には限界がある。

## 6. 結び

本論では、電子会議システムの各種応用で重要となる協同期型会議通信のための開放型文書操作方式の概念を明確化し、その実現に必要とされる要求条件を抽出した。さらに、遠隔の文書を開放型文書操作方式をベースに相互に操作するための基本プロトコルとなるORDE(Remote Open Document Editing)を提案/設計した。また、ORDEの実用性や有用性について評価するために、ORDEの実装、および機能評価を実施した。この結果、開放型文書操作方式の実現のための要

件1から要件6を基本的に満足でき、開放型文書操作方式の実用性、および有用性のあることを報告した。

今後の課題としては、3人以上の複数参加者による会議システムの構築、文書操作結果一致保証機能、協同で非同期に編集作業を行う形態への適用を考慮した文書の部分アクセス制御、およびSGML(Standard Generalized Markup Language)文書やアスキー文書を対象とした構造化処理メカニズムなどの検討を進めるとともに、ここで設計した通信機能を多様な通信応用へ適用し、利用の拡大に努めることとしたい。

最後に、日頃御指導頂くKDD研究所浦野所長、眞家次長、並びに御討論頂いた羽鳥画像情報処理グループリーダー、若原交換グループリーダーに感謝します。

## 参考文献

- 1) 石井 "グループウェア技術の研究動向", 情報処理学会誌 Vol.30 No.12 (1989).
- 2) 松下 "グループウェアの展望と動向", 電気情報関連学会連合大会 S32-1 (1990).
- 3) 阪田 "グループウェアの実現技術" SRCシリーズ (1992)
- 4) Ishii H., Kobayashi M. and Grudin J. : "Integration of Inter-personal Space and Shared Workspace : ClearBoard Design and Experiments", Proc. CSCW'92, ACM, pp33-43, (1992).
- 5) Mary D. P. Leland, Robert S.Fish and Robert E. Kraut "Collaborative Document Production Using Quilt", CSCW'88 pp. 206-215 (1988).
- 6) 阪田, 上田 "構内マルチメディア在席会議システムの実現とその評価", 情報処理学会論文誌 Vol.31-No.2 249-259 (1990).
- 7) 遠藤, 加藤, 山崎 "対話型画像通信システムQuick-CUICK "C"の試作(1) システム構成と基本機能", 電子情報通信学会春期全国大会 D-382 (1990).
- 8) Ellis C. A., Gibbs S. J. and Rein G.L. "Groupware Some Issues and Experiences", COMMUNICATION OF THE ACM, Vol. 34, No. 1 (1991).
- 9) Stefik M., Bobrow D., Foster G. Lanning S. and Tatar D. "WYSIWIS Revised : Early Experiments with Multiuser Interfaces", ACM Transactions on Office Information Systems, Vol 5, No. 2, pp. 147-167 (1987).
- 10) Gregg Foster and Mark Stefik "Cognoter, Theory and Practice of a Collaborative Tool", CSCW'86 pp. 7-15 (1986).
- 11) CCITT 勧告 T.410 シリーズ "Open document architecture (ODA) and interchange format" (1988).
- 12) 中尾, 遠藤 "オーディオグラフィック会議の基本要件条件とそのプロトコルの提案" 信学技報 OS87-20 (1987).
- 13) 柴田, 久保, 秦 "マルチメディア文書交換形式(ODA)を既存DTPシステムに適用するための論理構造変換方式" 情報処理学会第44回, 全国大会 (1992).
- 14) 田中, 中尾 "文書会議システムにおけるODAコンバータ方式に基づく文書処理機能の設計および実装/操作" テレビジョン学会誌 (1993/10 掲載予定).
- 15) CCITT 勧告草案 T.422 "Abstract Interface for Document Manipulation" (1993).
- 16) R. B. MILLER "Response time in man-computer conversational transactions", Spring Joint Computer Conference, Vol.33 (1968).

盛光印刷所