マルチメディア通信と分散処理 43 － 4
（１９８９．９．２８）

ＯＺ：オブジェクト指向開放型分散システムアーキテクチャ

－ ＬＬＣタイプ３を活用する通信アーキテクチャ、実装、およびその評価 －

塚本享治　　吉江信夫　　近藤貴士
電子技術総合研究所　住友電気工業　シャープ

水谷　功　　田中伸明
住友電気工業　　松下電器産業

ＯＺはオブジェクト間の相互関係を分散システム全体で維持しながら複写・移動することを基本とする分散処理システムである。通信し合うオブジェクトの数が非常に多く、相互参照関係が逐次変化する上に、さらに通信相手が移動する。そのため、通信システムには、オブジェクト群を符号化した不定長パケットを、効率よく、しかも確実に宛先に送達する機能が要求される。そこで、まず、ＬＬＣタイプ３を活かすＯＳＩコネクションレス型基本標準を使った７層構成の『確認付きコネクションレス型プロフィル』を実現した。次に、これを活用して大量のデータの送達を保証する『高信頼バルクデータ転送プロトコル（ＲＢＴ３）』を実現し、さらに、宛先が移動した場合にも最終移動先に大量のデータを確実に送達することのできる『ヒンティング付き高信頼バルクデータ転送プロトコル（ＲＢＴＨ３）』を実現した。これらのアーキテクチャ、実装、実測性能、および評価について述べている。

Implementation and Evaluation of a Network System Based on LLC Type 3 for ＯＺ: Object-Oriented Open Distributed System

Michiharu	TSUKAMOTO	Electrotechnical Laboratory
Nobuo	YOSHIE	Sumitomo Electric Industries, Ltd.
Takashi	KONDO	Sharp Corporation
Isao	MIZUTANI	Sumitomo Electric Industries, Ltd.
Nobuaki	TANAKA	Matsushita Electric Industrial Co, Ltd.

In OZ system, distributed application programs consist of objects connected with each other. Objects are copied or moved from the caller to the callee keeping the relationsips among objects. Therefore, the communicating system must transfer bulk data reliably to the destination object even if the object is migrated to another host. This papar describes our solution to these problems. At first, we developed the reliable "Acknowledged Connectionless Profile"based on LLC type 3 and connectionless OSI standards. Then, we developed "RBT3: Reliable Bulk Data Transfer Protocol, Type 3" which transfer bulk data reliably using the acknowledged connectionless profile. Futher, RBT3 was extended to "RBTH3: Reliable Bulk Data Transfer Protocol with Hinting Facility,Type 3" in order to transfer bulk data reliably to the migrated destinations.

1．まえがき

アプリケーションが高度になるほど、複雑なデータ構造が必要になることが多い。しかし、通信によってしか情報交換できない分散システム上で、自由に高度なデータ構造を定義し操作することは簡単ではない。筆者らは、分散システム上で相互に関係を持ち通信しあうプログラム全体をデータ要素がさまざまな方法で連結された１つの分散型データ構造とみなし、分散処理とは隣接するデータ要素間においてデータ要素群を複写・移動することだと考えた。さらに、送信側で自由に定義・作成したデータ要素群を受信側が正しく解釈できるようにするには、データにそれを解釈する手続きを付随させたオブジェクトを基本のデータ要素とすべきであると考えた。このような発想のもとに、分散システム上のすべてのものをオブジェクトで表現し、オブジェクト群を複写・移動することにより処理を進める方式の『オブジェクト指向開放型分散システムＯＺ』の開発を進めてきた[1),2)]。

ＯＺでは、オブジェクトの数は非常に多く、しかも移動可能であるため、分散されたオブジェクト間で確実にオブジェクトを転送するには、コネクション型転送を採用するのは難しい。そこで、コネクションを設定しないで転送することのできるコネクションレス型転送方式を採用することにした。しかし、通常のコネクションレス型転送では、送達が保証されず、また一度に転送できるデータの長さにも制限がある。そこで、まず、ＯＳＩコネクションレス型基本標準だけを使って、パケット長には制限があるが送達が保証される確認つきコネクションレス型プロフィルを開発した。次に、これを活用して不定長パケットの送達を保証するコネクションレス型バルクデータ転送プロトコルＲＢＴ３(Reliable Bulk-data Transfer, Type 3) を開発した。さらに、宛先が移動した場合にも最終宛先への送達を保証するために、ＲＢＴ３にヒンティングと呼ぶ機能を付加したＲＢＴＨ３(Reliable Bulk-data Transfer with Hinting Facility, Type3)を開発した。

コネクションレス型プロトコルとしては、開発がほぼ終了したＯＳＩの基本標準群、ＬＡＮを対象としてアカデミックな分野で研究開発されてきたＢＬＡＳＴ[3)]やＮＥＴＢＬＴ[4)]、最近ＡＮＳＩで開発中のＸＴＰ[5)]などがある。ＲＢＴ３はＯＳＩの基本標準をベースにしてＬＡＮの特長を活かしたプラストプロトコルである。また、ＲＢＴＨ３はこれまでにない新しい機能を提供するプロトコルである。

本稿では、まずＯＺの全貌と通信システムに要求される条件について述べる。続いて、通信アーキテクチャ、実装、および実測性能について述べ、最後に考察する。

2．システム構成の概要

ＯＺ実験システムのネットワークは３セグメントで構成されるＬＡＮであり、各セグメントはブリッジで接続されている（図１）。幹線と支線の各セグメントにはＩＳＯ880[illegible]-4[6)]に準拠するブロードバンドトークンバス方式とキャリアバンドトークンバス方式の[illegible]のを採用した。ＩＳＯ8802-4を採用した理由として２点があげられる。第１は、ＯＺは通産省の大型プロジェクト『電子計算機相互運用データベースシステム』の一環として進[illegible]られており、同大型プロジェクトでは昨年度までの前期においては、ＬＡＮはＩＳＯ88[illegible]-4のブロードバンドトークンバス方式を使用するという合意があったためである。第[illegible]は、これらの２方式の組み合わせは、工場用ＬＡＮとして注目を集めているＭＡＰにも[illegible]用されており、適用分野がきわめて広いためである。

現在、このＬＡＮに接続されているマシンは、国産ワークステーション（ＵＮＩＸシステムＶ）２機種６台と電総研で開発したデュアルＣＰＵボード数枚からなるマルチプロセッサである。各エンドシステムはシステムバスで接続されたＬＡＮボードを介してメディアに接続されている。

エンドシステムのソフトウエアは３レベルからなる。最下位はネットワークソフトウエア、最上位は分散型アプリケーションプログラムを搭載するためのドメインと呼ぶ仮想計算機であり、その中間には仮想計算機が相互に通信しあう環境を実現する分散カーネルである。ドメインは、システムの種々の管理を行う管理系、システムとユーザに対して共通資源へのアクセスを提供するサーバ系、オブジェクトをロードして実行する実行系に分類される。

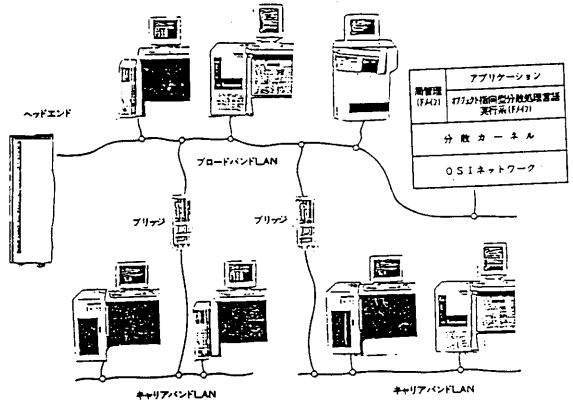

図１　ＯＺ実験システムの構成

3．通信システムが満たすべき条件

3．1　オブジェクトの表現

ドメインにおいては、すべての実体はオブジェクトという形式で表現される。オブジェクトがオブジェクトを引数としてオブジェクトに処理を依頼し、処理の結果をオブジェクトとして受け取る形で処理が進行する。処理効率を上げるために、オブジェクトはローカルオブジェクトとグローバルオブジェクトの２種類に分類されている。グローバルオブジェクトは分散される単位となるオブジェクトであり、分散システム全体で共有され、参照することができる。いっぽう、ローカルオブジェクトはいずれか１つのグローバルオブジェクトに所属し、所属するグローバルオブジェクトおよび同じグローバルオブジェクトに所属する他のローカルオブジェクト（仲間のオブジェクト）から参照されるだけである。

オブジェクトは、オブジェクトの本体であるセル、およびオブジェクトの識別情報と参照関係を記憶するプロクシ、の２つの部分に分けられている。プロクシは、オブジェクトの種類を記憶する属性、グローバルオブジェクトの場合に自身を識別する識別子ＵＩＤ(Unique ID)、ドメイン内にセル部があるときはそのアドレスを記憶するが他ドメインにセル部があるときはそのドメインの識別子ＤＩＤ(Domain ID) を記憶するアドレス、の３フ

ィールドで構成され、オブジェクトへの参照はすべてこのプロクシを介した間接参照となっている。ＵＩＤは、そのＩＤを付したドメインのＤＩＤとＤＩＤで付した通番ＬＩＤから構成され、ＵＩＤのドメインにＬＩＤを問い合わせると詳細が知れる。また、アドレスフィールドがＤＩＤのときは、そのＤＩＤに対応するドメインに目的のセルに至るプロクシが存在する。プロクシを集めた表をオブジェクト表と呼び、オブジェクト表だけで、そのドメインに存在するオブジェクト、およびそのドメインから参照しているオブジェクトの所在などが分かる（図２）。

３．２　オブジェクトの転送

仲間のオブジェクト間で処理を依頼し結果を受け取るさいには、オブジェクトへの参照関係（ポインタ）を渡す。これに対して、異なる仲間に属するオブジェクト間では、指定されたオブジェクトから参照関係を次々にたどり、ローカルオブジェクトは実体を複写し、グローバルオブジェクトは参照関係だけを複写する。異なるドメイン間でこのことが必要になると、オブジェクトの相互参照関係を保存するだけでなく、送信側で実行が中断されていたオブジェクトが受信側で再開できるようにレジスタやスタックの内容も修復できる形で転送しなければならない。このために、転送するオブジェクトの量は不定長となり、さらに、セル本体だけでなくプロクシも転送されるために、受信側ドメインのどのプロクシもそれまで参照していなかったドメインに対して新たな参照が生じることがある。参照されているプロクシに対して、新たに参照するプロクシが生れたことを通知するのは効率が悪いし、参照される側でその情報を維持するのも困難なため、プロクシには参照に必要な情報しか含めていない。

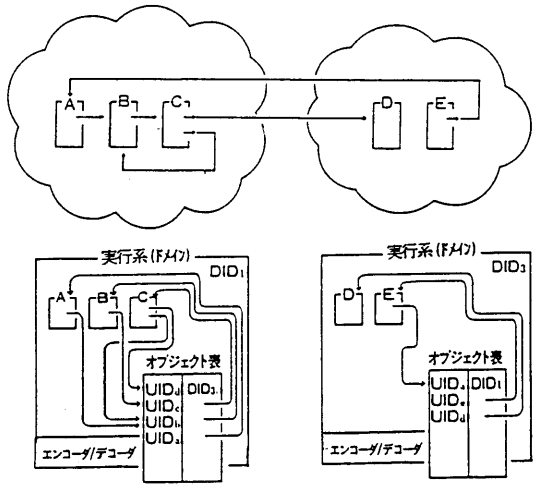

図２　オブジェクトの参照関係の表現

オブジェクトが移動するさいには、プロクシとセルを転送し、転送元ドメインにはアドレスフィールドを移動先ＤＩＤに変えたプロクシだけを残す。移動時の処理は簡単であるが、オブジェクトを受信してもプロクシが存在するにもかかわらず、目的のオブジェクトは移動してしまっているという状況がおきる。

通信システムに対しては、通信しあう対象の組み合わせの数が非常に多く、参照関係が逐次変化し、しかも対象が移動するという条件のもとで、不定長のデータを効率よく確実に宛先に送達する手段を提供することが要請される。

４．　通信アーキテクチャ

オブジェクト間における通信は、プロクシ間で行なわれるという見方とドメイン間で行なわれるという見方の２通りが考えられる。いずれの場合も、通信相手が非常に多く、しかも参照関係が逐次変化するため、通信対象間に固定的な通信路を設定する、いわゆるコネクション型モデルを採用することはできない。そこで、コネクションを設定することなく伝送できるコネクションレス型モデル[7]を採用することにした。

４．１　ネットワーク　－　確認つきコネクションレス型プロフィル

通信アーキテクチャの実現にあたり、次の３つの基本方針をたてた。まず、オブジェクトを転送するという考え方は密結合の考え方であるため、密結合が実際に実現できて有効性を発揮するＬＡＮを対象とする。次に、独自なプロトコルを開発することを避け、ＯＳＩのコネクションレス型基本標準だけを使ってプロフィルを開発する。さらに、このプロフィル単独でも一般的に使用できることを目指す。これらの方針にもとづき、コネクションレス型の欠点である送達を保証しないという問題を解決するＬＡＮ用のプロフィルを検討することにした。

ＬＡＮに関連するＯＳＩ下位２層のうち、物理層とＭＡＣ副層はＩＮＴＡＰ（情報処理相互運用技術協会）のＬＡＮ下位層実装規約書[8]に準拠し、モトローラ製ＥＴＢＣ(Enhanced Token Bus Controller) を使って実現された製品を使用することにした。そのため、検討する必要のあるものは、ＬＬＣ副層からＡＣＳＥまでである。設計したプロフィルを表１に示す。ＬＬＣ３の送達確認機能を利用したプロフィルであるところから、確認つきコネクションレス型プロフィルと名付けた。

表１　確認つきコネクションレス型プロフィル

層	基本標準	オプション選択
ＡＣＳＥ	ＣＬＡＣＳＥ(DIS10035)	バージョンあり、実装情報なし
プレゼンテーション	ＣＬＰＰ(DIS9576)	バージョンあり、セレクタ固定
セション	ＣＬＳＰ(DIS9548)	なし
トランスポート	ＣＬＴＰ(ISO6602)	チェックサムなし
ネットワーク	ＣＬＮＰ(ISO8472)	インアクティブ
ＬＬＣ	タイプ３(ISO8802-2/Pdad2)	送達確認（reply.exchange使用せず）
ＭＡＣ	トークンバス(ISO8802-4)	即時応答、非即時応答使い分け
但し、	(1)各層の最大ＰＤＵ長はＭＡＣの最大フレーム長で制限される。 (2)送信局が宛先局と同一セグメント上にあれば、ＭＡＣの即時応答機能の使用を推奨する。 (3)ブリッジはＭＡＣの優先度を保存する。	

ＬＬＣ副層の基本標準には３種のタイプが存在する。ＬＬＣ１はコネクションレス型で送達が保証されない。ＬＬＣ２はコネクション型であり、再送やフロー制御を行ない送達も保証されるが、コネクション数が小数に限られる。ＬＬＣ３[9]はコネクションレス型でありながら送達が保証される。ＬＬＣ１を使用するとＡＣＳＥまでは信頼性を保証する手立てがないため、ＯＺではＬＬＣ３を使用することにした。ＬＬＣ３には、送達確認機能（図３）以外に、相手ＬＳＡＰからのリードとＬＳＡＰ間でのデータ交換の機能があるが、リードと交換の機能はネットワーク層以上でこれを活用する基本標準がないので採用せず、送達確認機能だけを採用することとした。

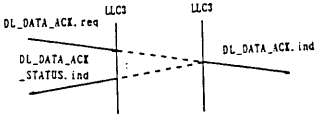

図３　ＬＬＣ３送達確認

ネットワーク層のコネクションレス型プロトコルＣＬＮＰは、分割・組立機能、経路選択・指定機能、などを持つが送達を保証しないので採用せず、標準で許されている空機能(inactive)で使用することにした。トランスポート層のコネクションレス型プロトコルＣＬＴＰとセション層のコネクションレス型プロトコルＣＬＳＰは、アドレスセレクタだけをもち、実質的には機能はないので、そのまま採用することにした。プレゼンテーション層のコネクションレス型プロトコルＣＬＰＰはアドレスセレクタと符号化・復号化機能をもつが、ＲＢＴで使用するさいには複数のパケットにわたって１つの符号化・復号化を行なうことになり、符号化と復号化が難しくなるので、ＯＺではオクテット列だけのＯＺコンテクストを設けることにした。ＡＣＳＥは送受信の相手を指定するための種々のパラメータをもっており、ＯＺでは応用エンティティ識別子でＲＢＴプロトコルマシンを識別し、応用プロセス識別子でドメインを識別することにした。

ＬＬＣ３をより効率よく活用するために、次のような工夫を行なった。ＬＬＣ３の下位に位置するＩＳＯ８８０２－４ＭＡＣ副層の基本標準には、受信側がトークンを保持したままで即時に応答する機能（即時応答機能）がある。これは、同一セグメント上の局間でしか使えないが、きわめて効率がよい。そこで、送信宛先局が同一セグメントにあれば即時応答機能を使用し、宛先局が同一セグメント上にないときは即時応答機能を使わないことにした。この判別を効率よく行なうために、ＭＡＣアドレスのＳＳＩ(Subnet Segment ID) をセグメントのＩＤに対応させることとした。ＬＳＡＰ以上のアドレスに関しては、Ｌセレクタは最大１６個、ＣＬＮＰは空なのでなし、Ｔセレクタは１つに固定、Ｓセレクタでアプリケーションを振り分け、Ｐセレクタは１つに固定した。１つのＬＳＡＰでは、応答が返ってくるまで、次のフレームが送信できないため、非即時応答では効率が悪くなる。そこで、複数のＬＳＡＰを使って同時に送信できるようにした。

ＣＬＮＰからＡＣＳＥまでの基本標準には送達確認を渡すサービスはないが、これはプロトコル上には表われないので、実装事項として導入した。

ＣＬＮＰを空機能で使用することにしたため、異なるセグメント間では通信できない。これを解決するために、ＭＡＣブリッジを導入することにした。

４．２　分散カーネル　－　ヒンティングつきバルクデータ転送

上記プロフィルを使って所望のサービスを提供するプロトコルを次の２段階に分けて検討する。まず、ＭＡＣフレーム長によって制限されない不定長パケットが効率よく確実に転送できる方式ＲＢＴ３を検討し、続いて、宛先が移動しても不定長パケットが移動先に確実に転送できるように拡張した方式ＲＢＴＨ３を検討する。

４．２．１　ＲＢＴ３

コネクションレス型サービスを使ったバルクデータ転送プロトコルの多くは、受信側からの確認をとることなく一方的に連続送信し、一連の送信終了後に受信側からの通知によって正しく受信されなかったパケットを再送する、いわゆるブラスト方式と呼ばれるものである。ＲＢＴ３の設計においては、ブラスト方式による再送機能と送信側複数ＬＳＡＰを用いたウインドウを使ったフロー制御機能を基本とし、次の４種類のサービスプリミティブを提供する。

(1) BT-DATA:　　ユーザデータを分割し、連続して転送するが、誤り回復やユーザへの結果の通知はしない。

(2) BT-DATA-ACK:　　ユーザデータを分割し、ウインドウが許すかぎり、連続して転送する。誤り回復やユーザへの結果通知機能を持つ。エンド間での送達確認が行なわれるが誤り回復はしない。

(3) BT-DATA-STATUS:　BT-DATA-ACK 要求の結果を通知する。このサービス発行の有無は、BT-DATA-ACK のパラメータによる。

(4) BT-ABORT:　　転送の中断・放棄を通知する。

BT-DATA とBT-ABORTは付加的なものなので説明は省略し、目的の高信頼バルクデータ転送に関連する BT-DATA-ACKとBT-DATA-STATUS（図４－１）について述べる。確認つきコネクションレス型プロフィルでは、受信側ＬＬＣ３からの送達確認を送信側が受信すれば、正しく転送されたものとみなすことができる。そこで、これをブラスト方式の送達確認として使用する。バルクデータ受信側ＬＬＣ３から応答されたパケットの送達確認がすべて成功であっても、エンド間の送達は保証されていない。このために、バルクデータ受信側が最終パケットを受信すると、バルクデータ受信の成否がバルクデータ送信側に通知される。バルクデータ送信側ＲＢＴ３は、これをユーザに指示する。いっぽう、バルクデータ受信側ＲＢＴ３は、その通知の送達確認を正しく受信したのちに、バルクデータ受信をユーザに指示する。

ＲＢＴ３の誤り回復は、送信側が主導権を持っており、受信側から誤り通知を積極的に行なうことはない。メディア上での伝送誤りにより宛先ＭＡＣに届かない時は送信側ＭＡＣ、宛先ＬＬＣ３から送達確認が届かない場合や失敗（宛先におけるバッファ不足など）の送達確認を受信したときは送信側ＬＬＣ３で再送する。それでも回復できないときは、バルクデータ送信側ＲＢＴ３ユーザにそのことを指示する（図４－２）。

ＲＢＴ３で分割したパケットは、空きのＬＳＡＰがあるかぎり、ＡＣＳＥを通じてＬＬＣ３に送信を要求できる。宛先ＬＬＣ３から成功の送達確認を受信するとＬＳＡＰが空くので、次の送信を要求できる（図４－３）。

４．２．２　ＲＢＴＨ３

ＲＢＴ３により、不定長のユーザデータを確実に転送することができる。しかし、ユーザデータを届ける宛先のオブジェクトが移動した場合には対応できない。移動先にフォワーディングする方法は、転送要求元と最終移動先間の送達を保証するのが難しく、オブジェクトが次々と移動した場合に間接参照の段数を減らすことができない。同報機能を使って移動先を放送する方法も、すべてのドメインに確実に通知するのが困難である。そこで、信頼性の高い対向通信を保証するＲＢＴ３の BT-DATA-ACKを次のように拡張する（図５）。ユーザデータを分割したパケットのうち、最初のものだけをまず宛先に送信し、その応答を待つ。宛先では、目的のオブジェクトが存在すれば成功と応答し、移動していれ

ば移動先情報をつけて失敗と応答する。失敗応答を受けるとユーザに移動先を指示するとともに、ＲＢＴＨ３が管理するプロクシを移動先に変更して移動先に再送する。通知成功応答を受けたさいの２番目以降のパケットの転送はBT-DATA-ACK と同様である。各ユーザは内部に所持するオブジェクトのプロクシをあらかじめＲＢＴＨ３に知らせておかねばならない。宛先から移動先のヒントをもらうことから、このメカニズムをヒンティングと名付け、この機能を付加したＲＢＴ３をＲＢＴＨ３と呼ぶ。

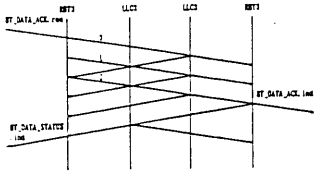

図4-1　ＲＢＴ３ﾌﾟﾛﾄｺﾙｼｰｹﾝｽ（正常）

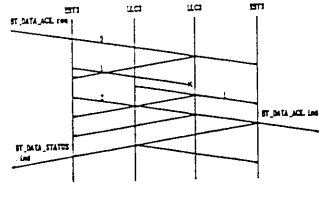

図4-2　ＲＢＴ３ﾌﾟﾛﾄｺﾙｼｰｹﾝｽ（誤り回復）

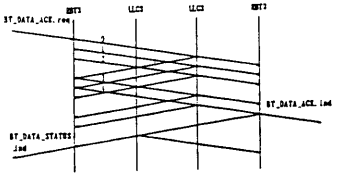

図4-3　ＲＢＴ３ﾌﾟﾛﾄｺﾙｼｰｹﾝｽ（ フロー制御）

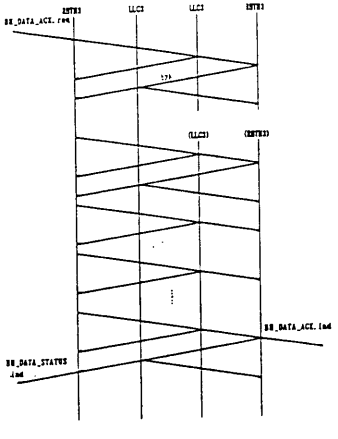

図5　ＲＢＴＨ３ﾌﾟﾛﾄｺﾙｼｰｹﾝｽ

5．実装

5．1　ＬＡＮボード

コネクション型ＯＳＩプロトコルのトランスポート層までを実装する目的で開発されたＬＡＮボードを利用し、そのプログラムを入れ替えることによって目的のものを実現した。その構成を図６に示す。ボード内には、ＲＯＭ、ローカルメモリ、デュアルポートメモリの３種類のメモリが設けられており、ＭＣ６８０００によって制御される。ＲＯＭには基本的なボードの制御プログラムが格納される。ローカルメモリはＥＴＢＣが送受信専用に使用するものであり、制御ＣＰＵがデュアルポートメモリとの間のコピーを行なう。デュアルポートメモリは、ホスト計算機のバスアドレスにマッピングされており、ホスト計算機と制御ＣＰＵ間でデータの授受に使用する。

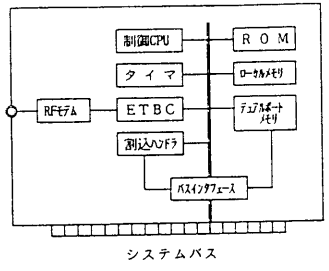

図６　ＬＡＮボードの構成

このボードでは、ＬＬＣ３までの処理を行なっている。ホストから制御レジスタを介して送信が指示されると制御ＣＰＵはデュアルポートメモリ上の２５６バイト単位で連結されたバッファからローカルメモリの連続領域にコピーしたのち、ＬＬＣ３送信コンポーネント（プロトコルマシン）処理を行なって、ＥＴＢＣに送信を指示する。制御ＣＰＵがローカルメモリに受信バッファを設けてＥＴＢＣに受信を指示すると受信可能になる。受信時にはＬＬＣ３受信コンポーネント処理がＥＴＢＣで行なわれたのち、制御ＣＰＵに受信が指示される。制御ＣＰＵは２５６バイト単位のバッファをデュアルポートメモリから獲得して分割コピーを行ない、そののち、ホストに受信割り込みを通知する。デュアルポートメモリに空きがないと、ローカルメモリからコピーできない。この場合には、ローカルメモリがいずれ枯渇するのでＥＴＢＣの受信エラーが発生し、送信元に受信エラー通知が返される。このようにして資源不足による背圧が伝播する。

5．2　ホスト

ＵＮＩＸマシンの場合には、ＣＬＮＰからＲＢＴＨ３までをＵＮＩＸドライバとして実現した。その構成を図７に示す。ドメインとＲＢＴＨ３間におけるプロクシの登録・変更用プリミティブは設けないで、共有メモリ上にオブジェクト表の一部を設け、ＵＮＩＸプロセスとして実現されるドメインと共有させた。

ユーザは、ユーザ空間内の送信バッファ列にデータを入れて、ioctl を用いてＲＢＴＨ３を呼ぶ。ＲＢＴＨ３では、送信プロトコルマシンを作ったのち、ユーザ空間内の送信バッファを共有メモリにコピーし、ＡＣＳＥが送信可能な最大パケット長単位に分割し、ヒンティング処理を行なったのち、残りのパケットの送信を要求する。ＡＣＳＥではデュアルポートメモリ内の２５６バイト単位のバッファ列を獲得してパケットを作り、空きのウインドウを探してＬＡＮボードに送信を要求する。現在の実装では簡略化のために、ウイ

ンドウに空きがない場合はスリープして、ウインドウが空くのを待つ。

ユーザは宛先の指示にＤＩＤとＵＩＤを用いる。指示されたＤＩＤがドメイン表に存在しない場合には、ＲＢＴＨ３からステーション管理にアドレス獲得を指示し、得られたアドレスをドメイン表に登録して送信処理を再開する。

ＬＡＮボードから受信割り込みが発生するとデモンが起動され、共有メモリ中に２５６バイト単位のバッファを獲得して、これにデュアルポートメモリからコピーののち、プロトコルを解析してＲＢＴＨ３に渡す。ＲＢＴＨ３では、それが最初のパケットならば受信側プロトコルマシンを作成したあと、オブジェクト表を参照しヒンティング要求に答える。宛先オブジェクトがドメインに存在すれば、後続のパケットを受信して連結する。一連のパケットを受信し、組み立てると宛先ドメインに受信を指示する。受信を指示されたドメインのプロセスはドライバを呼びだし、共有メモリをユーザメモリにコピーする。

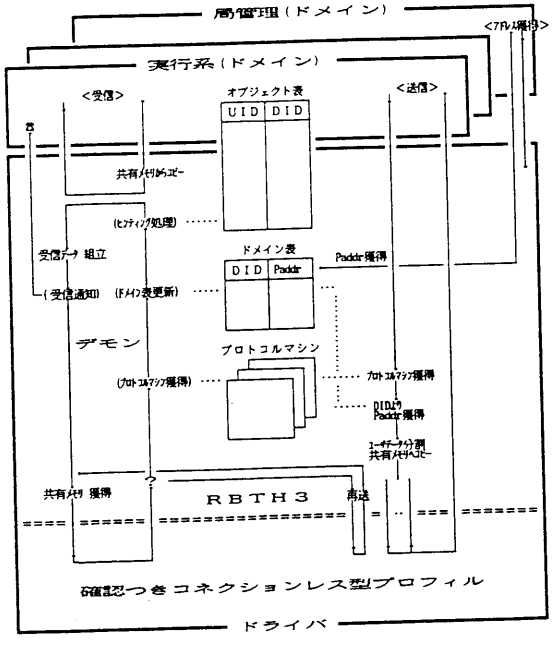

図７　ホスト計算機上のＲＢＴＨ３ソフトウエア構成

５．３　ブリッジ

ブリッジは、５．１で述べたＬＡＮボード複数枚をＵＮＩＸマシンにバスで接続し、ＵＮＩＸマシン上のソフトウエアによって実現されている。その構成を図８に示す。

ブリッジ用のＬＡＮボードでは、ＭＡＣ副層の送受信処理しか行なわない。中継するパケットはデュアルポートメモリに記憶される。ＵＮＩＸマシン上のソフトウエアは、ＬＡＮボードとインタフェースをとるためのドライバ、パケットの中継を行なう中継処理部、ブリッジを管理するブリッジ管理部、およびオペレータとインタフェースをとるブリッジモニタ部から構成されており、ブリッジの管理情報は共有メモリに記憶されている。

ブリッジでは、可能な限り無用なパケットを中継しないように配慮しなければならない。そこで、次の２種類の方法を採用した。

(1) ＥＴＢＣのフィルタリング：　ＯＺのアドレス体系はＳＳＩを利用している。そこで、各ＥＴＢＣのアドレスフィルタには、それが接続されているセグメントのＳＳＩを設定し、そのＳＳＩを持つパケットは受信しないようにしている。

(2) 中継処理部の学習・フィルタリング：　中継処理部の受信プロセスは、ドライバから、受信したＬＡＮボードの番号と受信したパケットのＰＣＩを受取ると、ＬＡＮボードの番号に対応するポート番号、発信ＳＳＩと宛先ＳＳＩを求める。次に、そのポート側に発信ＳＳＩのセグメントが存在することをフィルタリングテーブルに記憶し（学習）、さらに、宛先ＳＳＩがどのポートの先に存在するかをフィルタリングテーブルから求める（フィルタリング）。そののち、求めたポートに対応する送信プロセスに送信を指示する。対応するポートがなければすべての送信プロセスに送信を指示する。ＬＡＮの構成を変更したさいには、以前の学習結果を消さなければならない。そのため、フィルタリングテーブルの各エントリについてタイマ監視しており、一定時間以上アクセスされないとエントリは削除される。

このように、ＳＳＩをフィルタリングに用いることにより、必要なパケットだけをブリッジにとりこみ、さらにフィルタリングテーブルのエントリ数を少量にすることにより、無用な負荷がブリッジにかかるのを避けている。さらに、中継処理の効率を上げるために、受信したパケットはＬＡＮボードのデュアルポートにそのまま残し、送信プロセスからドライバに送信が指示されたときに、ドライバ内で受信側バッファから送信側バッファへコピーしている。

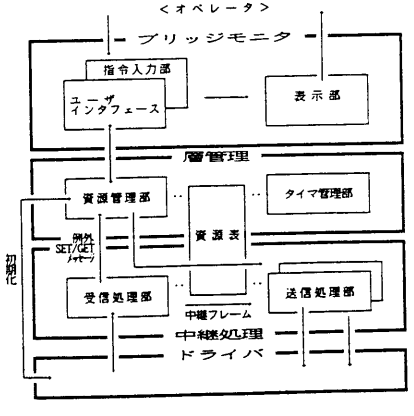

図８　ブリッジの構成

6. 性能測定実験

０Ｚの基本部分の開発が完了したので、それを改良した０Ｚ＋の開発に着手することにした。その設計に役立てるために、０Ｚの通信に関する部分の性能測定実験を行なった。その実測性能を図９にまとめて示す。

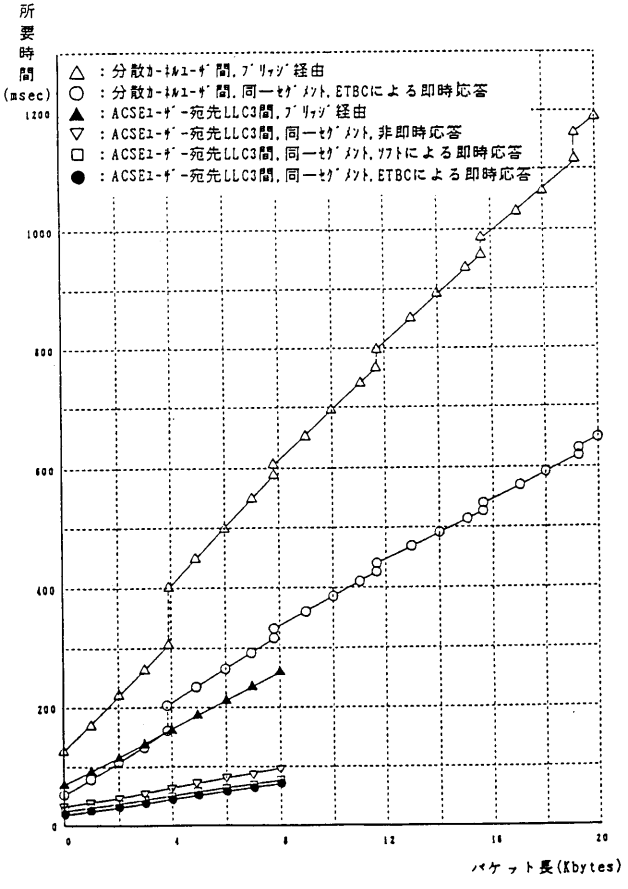

図９　性能実測値

6．1　確認つきコネクションレス型プロフィル

ＬＬＣ３送達確認の実現法には、即時応答を使用しない方法、ソフトウエアで即時応答を実現する方法、チップ（ＥＴＢＣ）によって即時応答を実現する方法、の３通りがある。図９における▽、□、および●は、ＵＮＩＸユーザプロセスからドライバ内に実装されたＡＣＳＥに対してデータを渡し、宛先ＬＬＣ３からの送達確認がユーザプロセスに戻されるまでの時間を測ったものである。即時応答の場合には、送信側がトークンを保持している時間内に送達確認を返す必要がある。ＥＴＢＣでは送達確認応答がハードウエアで行なわれるためにスロットタイムを短くすることができるが、ソフトウエアで実現する場合には短くできない。この実験では、ＥＴＢＣの場合は X'40'オクテットタイム（ X'40' オクテットの送信時間）、ソフトウエアの場合は限度に近い X'1200'オクテットタイムとした。

ＥＴＢＣ即時応答では、パケット受信から送達確認送信までの時間の実測値は数10μsec 以下であるため、０バイトのユーザデータ（ＰＣＩ長は１０８バイト）の場合の立ち上がり時間１６msec が、送信側におけるプロトコル処理とモジュール間インタラクションに要したオーバヘッドといえる。ソフトウエアで即時応答を実現する場合のオーバヘッドはさらに４msec 大きく、即時応答機能を使わない場合のオーバヘッドはまたさらに５msec 大きい。ＥＴＢＣ即時応答の場合の傾き８μsec/バイトから、メディア上の所要時間約１μsec/バイトを減じた７μsec/バイトが、送信側でユーザ空間からデュアルポートメモリを経てローカルメモリまでの間のバッファ確保とコピーに要する１バイトあたりの時間となる。即時応答機能を使わない場合の傾きがそれ以外に比べて大きいのは、受信側ＬＡＮボードにおいてローカルメモリからデュアルポートメモリにコピーしたのちに送達確認を戻しているためである。別の見方をすれば、この傾きの差２μsec/バイトがローカルメモリからデュアルポートメモリへのバッファ獲得・コピーに要する時間といえる。

6．2　ＲＢＴＨ３

図９における○は、分散カーネルユーザ（すなわちドメイン）からドライバ内のＲＢＴＨ３に渡されたデータが即時応答付きＬＬＣ３を使用したＡＣＳＥによって宛先のドメインに渡され、エコーバックされて送信側ドメインに戻されるまでの時間を半分にしたものである。

ＲＢＴＨ３の立ち上がりは５０msec である。ＭＡＣの最大フレーム長は約４Ｋバイトであり、４ＫＢまでの傾き３５μsec/バイトが、送受信両側におけるバッファ獲得とコピーの性能である。ＡＣＳＥで転送されたデータを宛先側でコピーする速度が送信側でのコピーと同じと考え、３５μsec/バイトから、両側におけるＡＣＳＥからＥＴＢＣまでの速度２×７μsec/バイトとメディア上の速度１μsec/バイトを引いた、２０μsec/バイトが両側ＲＢＴＨ３におけるバッファ獲得とコピーのおおよその性能と言えよう。

送信データが４Ｋバイトを越えると、ＲＢＴ３によって分割して、複数のＬＳＡＰをウインドウとして使って転送し、宛先ＲＢＴ３で組立が行なわれる。ちょうど４Ｋバイトごとに階段的に性能が低下するが、４パケット以上になるとウインドウ効果によって階段的な性能低下は認められなくなる。

6．3　ブリッジ

ブリッジを介して、６．１と６．２に対応する性能を測定したものが、図９における▲

と△である。

確認つきコネクションレス型プロフィルにおける立ち上がりは、即時応答つきＬＬＣ３を使った場合のほぼ４倍強の立ち上がりとなっている。４回の転送が行なわれることを考慮するとこの値は妥当なものといえる。４Ｋバイト以下において、ブリッジを介した▲の傾きから、それに対応する同一セグメント上における●の傾きをを引いたものがブリッジにおけるバッファ獲得とコピーの速度となる。この値は、２０μsec/バイトである。いっぽう、ＲＢＴＨ３に関する△と○の４Ｋバイトまでの傾きの差は１９μsec/バイトであり、上記の値とほぼ同じ値となる。

ウインドウ効果が有効に働いていないのは、連続してＰＤＵを送信するさいには、ＰＤＵ中継処理中に送信側から次々とＰＤＵを受信し、さらに送達確認ＰＤＵも受信するためにブリッジで輻輳が生じていることが原因だと思われる。

７．考察

７．１　性能

少量データ転送時に十分な応答速度が得られるように改善しなければならない。ＬＡＮボードとホストドライバ間のハンドシェイクの改善、ドメイン、ＵＮＩＸ共有メモリ、ＬＡＮボードデュアルポートメモリ、ＬＡＮボードローカルメモリというメモリの種類の減少、ドメイン（ＵＮＩＸプロセス）とドライバ間における記憶空間の問題とインタラクションの改善、などを行なう必要がある。

大量データ転送時に十分な性能が得られていない点も改善しなければならない。この解決方法は上記問題点の解決方法と関連するが、メモリの種類の減少やＤＭＡの採用などが必要である。現在の実装で容易に改善できる点としては、ドメインとドライバ間でデータ交換に使用するバッファの単位を大きくすることが上げられる（現在のところバッファ単位は２５６バイトであり、上記実験に使用したマシンでは、ドメインからドライバにコピーインし再びコピーアウトする性能は、２５μsec/バイトと３８μsec/バイトである）。

７．２　方式

確認つきコネクションレス型プロフィルは正しく動作することが実証された。ＭＡＰ（Manufacturing Automation Protocol)では即時応答機能しか採用しないために複数のセグメントにまたがった通信ができないが、宛先によって即時応答機能を使用するか否かを選択するというわれわれの方式はこの問題を解決している。ＲＢＴＨ３のヒンティング機能は宛先の存在を確認するためのものであり、コネクションレス型伝送方式を活用する上で有効であると確信している。

ＲＢＴＨ３は、ＬＬＣ３の即時応答機能を使えば、予想通りの振る舞い（性能は十分ではないが）をしていることが確かめられた。ＲＢＴ３はその一部であるため、ＲＢＴ３も正しく動作し、予想通りの振る舞いをするものと思われる。しかし、ブリッジを介したバルクデータ転送の場合には、輻輳が起きていると思われる。この原因としては、まずブリッジの性能が十分でないことがあげられるが、ＬＬＣ３送達確認がブリッジを輻輳させる要因となっている可能性がある。得られた特性を使って数理的な手法やシミュレーション手法を使って、ＲＢＴ３およびＲＢＴＨ３とブリッジとの関連について、究明する必要がある。

８．むすび

ＬＬＣ３をベースにした確認つきコネクションレス型プロフィル、これを有効に活用する高信頼性バルクデータ転送プロトコルＲＢＴ３、および宛先が移動した場合でも宛先に送達することを保証する機能をさらに付加したＲＢＴＨ３を設計して実現した。未だ十分な性能を達成してはいないが、いくつかの分野では方式的な正しさが実証できたといえる。ブリッジを介したバルクデータ転送時に見られた性能上の問題に関しては、早急に検討を行ないたい。ＯＺを改良したＯＺ＋では、その検討をもとに、よりよいプロトコルとするよう改良を加えたい。

なお、この研究は、通産省の大型プロジェクト『電子計算機相互運用データベースシステムの研究開発』の一環として進められているものである。その機会を与えられ、しかもいろいろ助言をいただいた電子技術総合研究所情報アーキテクチャ部長棟上昭男博士に感謝致します。また、住友電気工業、松下電器産業、シャープの関係各位のご助力にも感謝致します。さらに、筆者らの要望に応じて、ＬＡＮボードを再開発していただいた住友電気工業の方々、およびブリッジを開発していただいた松下電気産業の方々にも深謝致します。

参考文献

1) M.Tsukamoto etal.: The Architecture of Object-Oriented Open Distributed System: OZ. Interoperable Information Systems ISIIS '88. Ohmsha. pp153-166 (1988.11)
2) 塚本他：ＯＺ：ｵﾌﾞｼﾞｪｸﾄ指向開放型分散ｼｽﾃﾑｱｰｷﾃｸﾁｬ - ｵﾌﾞｼﾞｪｸﾄ指向型分散ﾌﾟﾛｸﾞﾗﾐﾝｸﾞ 言語とその実装、情処学会研究会報告、ﾌﾟﾛｸﾞﾗﾐﾝｸﾞ言語21-4 (1989.6)
3) D.Theriault: BLAST, an Experimental File Transfer Protocol. MIT-LCS Computer System Research Group RFC-217 (1982.3)
4) D.Clark etal.: NETBLT: A High Throughput Transport Protocol. Proc of ACM SIGCOMM '87. pp353-359 (1987.8)
5) M.Cohn: A Lightweight Transfer Protocol for the U.S. Navy SAFENET Local Area Network Standard. Proc of 13th Conf on Local Computer Networks. pp151-156 (1988.10)
6) ISO8802-4: Information Processing System - Local Area Networks - Part4: Token-passing Bus Access Method and Physical Layer Specification
7) ISO7498-1/Add.1: Information Processing System - Basic ReferenceModel - AD1 Addendum Covering Connectionless-mode Transmission
8) 情報処理相互運用技術協会: LAN下位層実装規約書 S012(V1.0) (1989.3)
9) ISO8802-2/Pdad2: Addendum to ISO DIS 8802-2 Logical Link Control - Acknowledged Connectionless-mode Service. Type 3 Operation