

解説

## コンピュータ・ネットワークにおける HOST-HOST プロトコル\*

伊藤 哲史\*\*

### 1. まえがき

コンピュータ・ネットワークとは、L. Roberts 他<sup>1)</sup>の論文においては“いかなるコンピュータ間においても相互に資源が共用できるように、自律かつ独立なコンピュータを相互に結合したシステムである”といっている。このように結合された自律かつ独立なコンピュータを Host と呼ぶことにする。

コンピュータ・ネットワークにおいて実現をめざす機能は、プログラムの共用、データの共用、ハードウェアの共用、ロードの均一化などである。

コンピュータ・ネットワークにおいては、Host 内のプロセスの協同作業によって一つの仕事が達成されるために、プロセス間に約束が必要になる。

プロトコルとは、複数のプロセス間で授受されるデータの形式、授受のタイミングなどに関して決められた約束ごとの集合である。この意味において、コンピュータ・ネットワークのプロトコルは、利用目的およびハードウェア構成によってかなり違ったものになる。

ここでは、特別なハードウェアを考慮しないで、利用目的が広い範囲におよぶ Host-Host プロトコルについて述べる。また、用語の統一がなされていないので、共通的に使用されているものはそれを利用したが、各ネットワークでばらばらな用語を使用しているものは、ARPA-net で使用している用語にしたがった。

コンピュータ・ネットワークに望まれる機能は多種

\* Host-Host protocol for a resource sharing computer networks, by Tetsufumi ITO (Japan Information Processing Development Center)

\*\* (財)日本情報処理開発センター

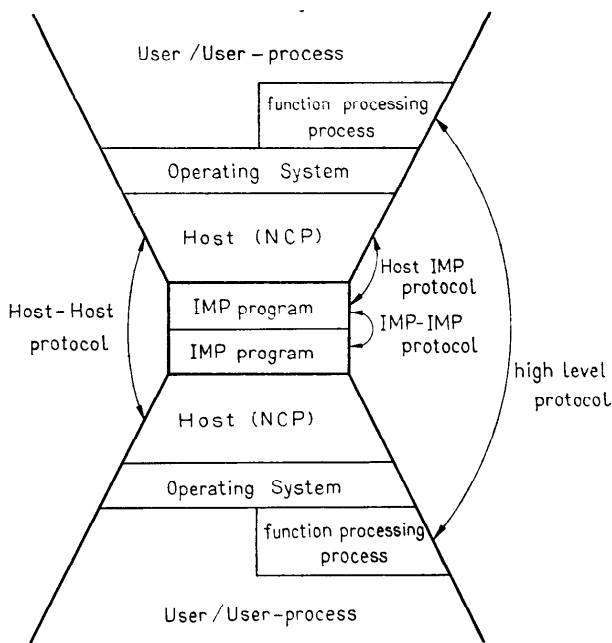


図-1 プロトコルの構造

にわたっているために、これらをすべて一つのレベルのプロトコルでサービスするのは困難である。このために、基本的機能をはたすものから、順次より高度な機能をはたすプロトコルへと積み上げる方式がとられる。

プロトコルにはプロセスが対応し、一般的には次のようなレベルがある。もちろんこれらのレベルのプロトコルの内部においても階層構造を持たせる場合がある。

- (1) IMP-IMP プロトコル  
メッセージ交換網を実現するもの。
- (2) Host-IMP プロトコル  
メッセージ交換網と Host の結合をする。
- (3) Host-Host プロトコル

NCP 間の会話，ユーザへのメッセージ交換などをする。

(4) 高位のプロトコル

機能ごとに決められたユーザ・レベルのもの。

現在，汎用目的で開発されたコンピュータ・ネットワークは，プロセス間通信を基本的な Host-Host プロトコルとしている。これは，プロセスを端末制御プロセスとすれば，TSSの共用も可能であり，ファイル操作プロセスとすれば，ファイルの共用も考えることができ，一般的な利用法への拡張が可能になるからである。

2. HOST-HOST プロトコルの機能

Host-Host プロトコルは，他の Host と協力して基本的なサービスを利用者にたいしてするためのものであり，各 Host の中に存在する\* network control program (NCP) がこの処理を担当する。

このプロトコルに必要な機能は，次のものである。

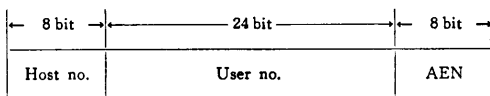
- (1) 利用者あるいはプロセスの識別方法。
- (2) NCP 間で授受されるメッセージ形式と意味づけ。
- (3) コントロール・コマンド(NCP 間会話コマンド)。
  - 利用者のメッセージ交換。
  - メッセージ・フローの制御。
  - エラー制御。
  - 割り込み信号の授受。

以下にこれらの機能の実現方法について述べる。

2.1 プロセスの識別方法

プロセス間通信においては，相手プロセスが通信回線のような物理的実体に直接結合されている保障がないので，プロセスを識別するための，ネットワーク内にユニークな名前をつける必要がある。

この名前を port-id と呼ぶが，分散形のコンピュータ・ネットワークにおいては，Host 番号とローカル番号という形で構成される。ローカル番号は，ユーザ番



AEN はユーザが種々のを作るためのものである。

図-2 ARPA-net の port-id

\* NCP は，Host 本体とは限らず，フロント・エンド・プロセッサの中で実現することもある。  
 \*\* ARPA-net, Cyclades, JIPNET がこの方法を利用。  
 \*\*\* NPL がこの方法を利用。

号，ユーザが任意につける番号より構成する方法と\*\*，NCP で一括管理する方法\*\*\*があるが，前者の方法の方が汎用性があり，ユーザが使い易い。

2.2 メッセージ形式と意味づけ

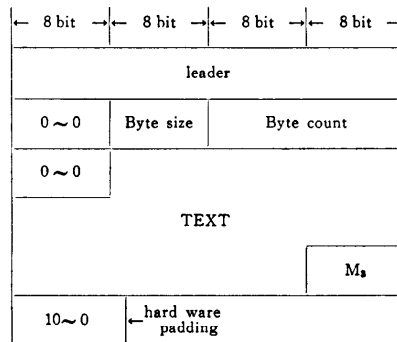
NCP 間で交換されるメッセージには，NCP の会話に利用されるコントロール・コマンドと，ユーザ・プロセスにわたすべきデータがある。

メッセージ形式は，そのネットワークがビット単位のデータをあつかうか，バイト単位のデータをあつかうか，データ交換網においてのデータの最大長，Host においてメッセージの分割，再組み立てをおこなうかどうかなどにより種々の形態がとられる。

また異機種のコンピュータを結合する場合には，各 Host の語長などを考え合せて，データ長，メッセージ・ヘッダーなどをうまく決める必要がある。

ARPA-net では，ビット単位のデータのとりあつかいができるように考慮されているが，Cyclades, NPL, JIPNET においてはバイト単位のデータをとりあつかうようになっている。ビット単位のデータは，同一機種間のバイナリ・データの伝送など特殊な用途にしか使用せず，バイト単位であっても，より高位のプロトコルにおいて注意すればビット・データも十分使用可能なので，バイト単位のメッセージで十分であろう。

Host-Host プロトコルでのメッセージの分割，再組み立ては，Cyclades においておこなわれているが，この機能も，送信側の出力した順序が保たれて，受信側でメッセージを受け取ることができるのであれば，より



leader : Host-IMP protocol による。  
 Byte size : テキスト長の最小単位をビットで示す。  
 Byte count: テキスト長を Byte size×Byte count ビットで示す。  
 M<sub>s</sub> : ソフトウェア・パディングで0ビット以上。内容は必ず0である。

図-3 ARPA-net の Host-Host プロトコルのメッセージ形式

高位のプロトコルに持ち込んでも十分なものである。

ユーザ・プロセスにわたすデータ、コントロール・コマンドの区別は、ARPA-net, JIPNET のようにメッセージ・リーダーでおこなうものと、NPL, Cyclades のように、コントロール・コマンドのコードとしてユーザ・データを指定するものがある。

2.3 コントロール・コマンド

コントロール・コマンドに持たせるべき機能を述べ、ARPA-net, Cyclades, NPL, JIPNET における概略の比較を表-1 に示す。

2.3.1 利用者のメッセージ交換

この手段として、バーチャル・コール方式、データ・グラム方式、Walden の方式<sup>1)</sup>がある。

バーチャル・コール方式は、プロセス間通信に先だって、2つのプロセス間に仮想的な通信路(コネクションという)を設定するものであり、Host-Host プロトコルにおいて、一般的に利用されている。

データ・グラム方式は、発信、受信 port-id をメッ

セージ中に持たせて、直接プロセスに送り込むものであり、Cyclades の基本機能としてサービスされている。

Walden の方式は、プロセス間通信をする2つのプロセスのメッセージ送、受信の要求の対応を、メッセージごとにとりながら伝送するものである。

バーチャル・コール方式の欠点は、多数のプロセスが互いにメッセージの交換をしながら処理を進めていくような場合には、コネクションの数が非常に多くなる点、一方のプロセスがコネクションを切断しないで消滅した場合には、コネクションが残ってしまうなどをあげることができる。

データ・グラム方式の欠点は、特定の Host にメッ

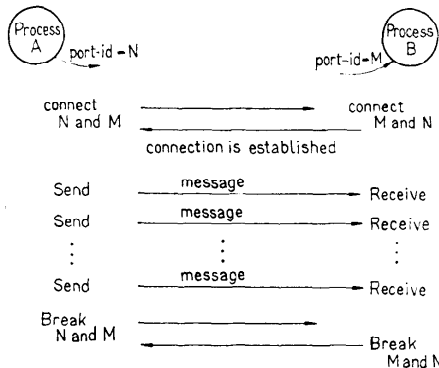


図-4 バーチャル・コール方式



図-5 データ・グラム方式

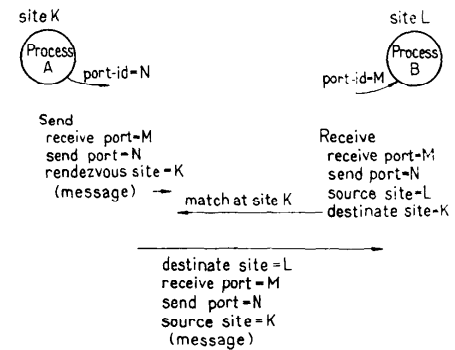


図-6 Walden の方式

表-1 Host-Host プロトコルの比較

	基本思想	プロセス間コミュニケーションの形態	メッセージの到着の確認	フロー制御		割り込み信	エラーの通知	メッセージ長	その他
				受信側準備	送信側準備				
ARPA-net	プロセス間コミュニケーション	・バーチャル・コール ・パスは一方向	RFNM (目的地IMPが発信)	受信側準備	コントロール・コマンドにより, allocate を通知	INS, INR, コマンドによる	ERR コマンドによる	8023 ビット (1パケットのとき936ビット)	NCP のリセット機能
IPNET	同上	同上	ACK (NCPが発信)	WABT	送信中止, 再開のコントロール・コマンド	POST コマンドによる	POST コマンドによる	9216 ビット (1パケットのとき1152ビット)	プロセスの発生機能
NPL	同上	・バーチャル・コール ・パスは両方向	なし メッセージ番号の不連続により, 紛失メッセージを検出し通知	受信側準備	コントロール・コマンドにより, allocate を通知	INTERR, UPT コマンドによる	ERROR, DROP コマンドによる	2008 ビット	ターミナルの制御機能
Cyclades	同上	同上 ただし, パスの設定を行わない方式もある	ACK (NCPが発信)	同上	反対方向のメッセージ, ACK に便乗	FL-TG コマンドによる	FL-ERR-OR コマンドによる	1976 ビット HOST のリアセンブル能力により 1976×127 ビットまで可	サービスに対して階層性がある

セージが集中しないように、フロー制御するのがきわめてむずかしい点をあげることができる。

Walden の方式の欠点は、メッセージごとに、プロセスの同期がとられるために、大量データを送る場合に、スループットが小さくなる可能性がある点をあげることができる。

### 2.3.2 メッセージ・フローの制御

フロー制御は、受信側 NCP において、メッセージを処理する能力以上のものが流入することから発生する問題を避けるために必要な機能である。

この手法としては、以下にあげるものがある。

- (1) バッファ確保方式
  - i) バッファ要求送信方式<sup>7)</sup>
  - ii) 受信側準備方式<sup>2)</sup>
- (2) クレート方式<sup>2)</sup>
- (3) 受信命令待ち方式<sup>2)</sup>
- (4) Window 方式<sup>2)</sup>
- (5) Wait before transmit (WABT) 方式

バッファ確保方式とは、メッセージ受信のためのバッファを受信側で確保し、そのバッファがある限り、メッセージを送ることができるようにするものである。確保する方法としては、送信側でメッセージが発生した時点で、バッファ予約メッセージを送り、その応答を待って、メッセージを送る方式と、コネクション確立時に、受信側でバッファを確保して、これを送信側に通知し、バッファがなくなった時点などで、新たに確保、通知する方式がある。

前者は、ARPA-net の IMP-IMP プロトコルにおいて採用されており、後者は、ARPA-net, Cyclades, NPL の Host-Host プロトコルにおいて採用されている。

クレート方式は、メッセージを送る場合には、クレートという送信権を持つものを確保したのちに、これに乗せて送るものである。メッセージを受け取ることにより、クレートも同時に確保できることから、1個のクレートをプロセス間に割り付ければ、交互通信となり、多重にすれば全二重に近くなる。

この方法のクレートが1個の場合を、初期の NPL の Host-Host プロトコルは採用していた。

受信命令待ち方式は、プロセスが受信命令を發したのち、データの伝送を開始するものである。この場合には、すでにプロセス内にバッファが用意されているので、NCP においてバッファを確保する必要がない。

Window 方式とは、送、受信の両側で、あらかじめ

一定の長さ（これを window という）のバッファを受信側で確保することによって制御するものである。送信側は、window の大きさまでのメッセージを送ることができ、受信の確認信号がくるまでは、メッセージ長だけ window の大きさが小さくなる。確認信号を受けると、それに対応したメッセージ長の大きさだけ、window が大きくなる。

この方式では、確認信号をだすものが、プロセスであれば、フロー制御は十分に動作するが、一般的には NCP がメッセージを受信したときに確認信号をだすため、受信側 NCP のバッファがあかないあいだに次のメッセージがくることになり、フロー制御が十分に動作せず、メッセージの再送手順を組み込んでおく必要がある。

WABT 方式は、受信側が送信中止をだすまで、メッセージを送るものである。これでは、送信中止を受け取るまでに発信したメッセージの再送手順を組み込んでおく必要がある。この方法は JIPNET において採用されている。

### 2.3.3 エラー制御

エラー制御には2つの要素がある。

第1番目は、コントロール・コマンドのパラメータなどのミスなどが発見されたときに相手側に通知する方法、相手側の NCP が正しく動作していないと考えられる場合のテスト機能および初期設定要求方法である。エラーの通知方法は、ARPA-net, Cyclades, NPL, JIPNET のいずれにもあるが、テスト機能などは ARPA-net においてのみある。

第2番目は、NCP 間で授受されるメッセージの紛失および重複のチェックを NCP レベルでおこなうかどうかということである。これは NCP 間のメッセージのチェックを NCP を末端としておこなうかどうかというコンピュータ・ネットワークにおける機能分担と密接にかかわりあうものである。

ARPA-net, JIPNET においては、発信地 IMP-目的地 IMP においておこなわれており、NCP にこの機能を持たせてはいない。Cyclades においては、パケット交換網のパケットあたりの誤り率（紛失および重複）が  $10^{-4}$  程度であり、NCP にこの機能を持たせている。

### 2.3.4 割り込み信号

割り込み信号は、プロセスにおいて特殊な状態が発生した場合に相手プロセスに対してそのむねを通知する手段をあたえたものである。

これには ARPA-net, NPL のように割り込みだけを通知するものと, Cyclades および JIPNET のように, 割り込みとともに 16~32 ビットのステータス情報を同時に送るものがある。

通知を受け取る側の手段に関しては, Host の OS に密接に関係があるためにプロトコルにおいて決めないのが普通である。

### 3. NCP

NCP は, Host-Host プロトコルで決められている機能を, Host-IMP プロトコルにしたがって, 他Host の NCP と会話し, その機能をユーザ・プロセスに対してサービスするプログラムである。

NCP の構造およびプロセスへのサービス形態は, Host-Host プロトコル, Host のオペレーティング・システム (OS) に密接に関係しているために, 一般的にはいえない。たとえば, JIPNET を作成するさいに, FACOM 230/75 では, NCP はサブ・モニタの形になり, HITAC 8450 では完全なユーザ・ジョブである。

このために, NCP を作成するために必要な OS の機能, プロセスへのサービスとして基本的にしなければならないものを, ARPA-net のようなプロトコルを前提として概説するとどめる。

NCP を作成するために必要な最小限の機能は, 次あげる5つである。

- (1) コミュニケーション装置に対するアクセス。
- (2) ジョブ間通信。
- (3) ジョブの起動および起動したジョブの監視。
- (4) ファイルの動的な割り付け, 消去。
- (5) ランゲージ・プロセッサなど既製プログラムをネットワークを通して利用可能にする。

コミュニケーション装置に対するアクセスは, 実際には IMP とソフトおよびハードウェア的のどのようにして結合するかという問題である。

JIPNET においては, 半二重の装置でチャネル間結合をおこなっているが, 半二重の装置では, 互いに非同期の転送をおこなおうとすると, コマンドのすれ違いがハードおよびソフトの両面において発生してかなりの問題点があることが報告されている。

ジョブ間通信はプロセス間通信をおこなうための必須条件である。既製の OS において, プログラム間通信の形でサービスされているものもあり, また OS レベルのマクロとしてならば利用可能なものもある。こ

のようなサービスがされていないと, 新たに OS に組み込むことになるが, この場合にはプログラムがロール・アウトされていることがあり, かなりやっかいな修正となることになる。

ジョブの起動および監視は, ユーザの依頼によってジョブの発生をする, あるいは高位のプロトコルにおいてリモート・バッチをサービスする場合などに必要となってくる。

ファイルの動的な割り付けおよび消去は, ファイル転送において必要になる。

既製プログラムの利用は, TSS を通じて利用する場合も含み, このためには既製の OS の一部を修正してプロセスが端末の代りに動くような機能を追加しなければならない。

以上に述べた以外に, NCP に必要な機能として追加するとすれば, タイマーによる状態監視であろう。

NCP は以上に述べた機能を利用して, プロトコルに対応したサービスをすればよいわけであるが, JIPNET においては基本的に次のようなサービス・コマンドを用意している。

- (1) Open : 自分および相手側の両 port-id を指定してコネクションの確立を要求。
- (2) Close : コネクションの切断。
- (3) Read : 確立したコネクションを通じてデータを読む。
- (4) Write : データを送る。
- (5) Listen : 自分側の port-id を宣言して, 相手からのコネクション確立要求および port-id を知る。
- (6) Post : ステータス情報を送る。高位のプロトコルにおいて割り込み信号を送るときなどに利用。
- (7) Wait : サービス・コマンドの終了などのイベントを待つ。
- (8) Set interrupt entry : 相手側から Post が発信されたときの割り込み番地を定義する。インプリメントがむずかしいので, ほとんどの場合は実現せず, Post は Wait によって受けている。

これらのサービス・コマンドを使用して, 高位プロトコルにおいてはどのようにしてコネクションを確立するかを, JIPNET で使用しているものを例として, 図-7 (次頁参照) に示す。

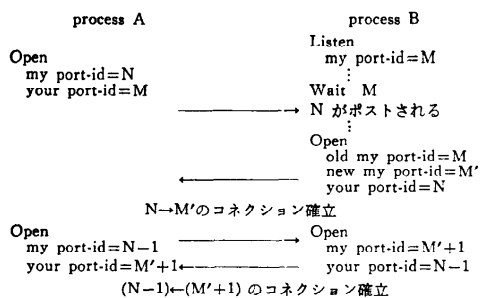


図-7 JIPNET の接続確立手順

#### 4. おわりに

紙面の関係で、各コンピュータ・ネットワークにおける Host-Host プロトコルの詳細および問題点に関する記述は省略したので、対応する文献を参照されたい。

ARPA-net は文献 9), Cyclades は 14), NPL は 16), JIPNET は 19) である。

19) には以上の 4 つのネットワークの Host-Host プロトコルの概説と検討も記述されている。

Kahn がプロトコルの重要性を強調し、Pouzin は“悪いプロトコルは頭痛の種である”といているように、コンピュータ・ネットワークにおいてプロトコルは非常に重要なものである。

コンピュータ・ネットワークの相互結合を容易にするために Host-Host プロトコルの標準化、コンピュータ・ネットワークの作成を容易にするためにユーザの機能追加が簡単にできる OS のサービスが望まれる。

#### 参考文献

- 1) D. C. Walden: A System for Interprocess Communication in a Resource Sharing Computer Network, CACM, Vol. 15, No. 4, pp. 221~230 (1972)
- 2) R. E. Kahn, W. R. Crouther: Flow Control in a Resource-Sharing Computer Network, Trans. IEEE Vol. COM-20, No. 3, pp. 539~546 (1972)
- 3) V. G. Cerf, R. E. Kahn: A Protocol for Packet Network Intercommunication, Trans. IEEE Vol. COM-22, No. 5, pp. 637~648 (1974)
- 4) E. Akkoyunlu, et al.: Interprocess Communication Facilities for Network Operating Systems, Computer 1974, June, pp. 46~55
- 5) 伊藤: Host level と protocol, コンピュータ・ネットワーク講習会テキスト (情報処理学会), pp. 18~23 (1975)
- 6) C. S. Carr, et al.: HOST-HOST communication protocol in the ARPA network, SJCC conf. proc. Vol. 36, pp. 586~597 (1970)
- 7) J. M. Mcquillan, et al.: Improvements in the design and performance of the ARPA Network, FJCC conf. proc. Vol. 41, pp. 741~754 (1972)
- 8) S. D. Crocker, et al.: Function Oriented protocols for the ARPA Computer Network, SJCC conf. proc. Vol. 40, pp. 271~280 (1972)
- 9) Host/Host protocol for the ARPA network. NIC # 8246, p. 37 (1972)
- 10) TELNET protocol specification, NIC # 15371~15373, 15389~15393, p. 42 (1972)
- 11) Official Initial Connection Protocol, NIC # 7101, 7155, 7103, p. 7 (1971)
- 12) L. Pouzin: Network protocols, Reseau Cyclades SCH 517, p. 25 (1973)
- 13) L. Pouzin: CIGALE, the Packet Switching machine of the Cyclades computer network, IFIPS conf. proc. 74, pp. 155~159 (1974)
- 14) H. Zimmermann, et al.: Transport Protocol. Standard host-host protocol for heterogeneous computer networks, Reseau Cyclades SCH 519.1, p. 31 (1974)
- 15) H. Zimmermann: Terminal access in the Cyclades computer network, Reseau Cyclades TER 510, p. 8 (1974)
- 16) R. A. Scantlebury, P. T. Wilkinson: The National Physical Laboratory Data Communication Network, '74 ICCS Stockholm, pp. 223~228 (1974)
- 17) P. A. Bailey, B. M. Wood: A Central File Store for the data communication network at the National Physical Laboratory, '74 ICCS Stockholm, pp. 229~238 (1974)
- 18) 日本情報処理開発センター: コンピュータ・ネットワークシステムの研究開発, 48-S 001, p. 253 (1974)
- 19) 日本情報処理開発センター: コンピュータ・ネットワークシステムの研究開発, 49-S 001, (1975)

(昭和 50 年 4 月 11 日受付)