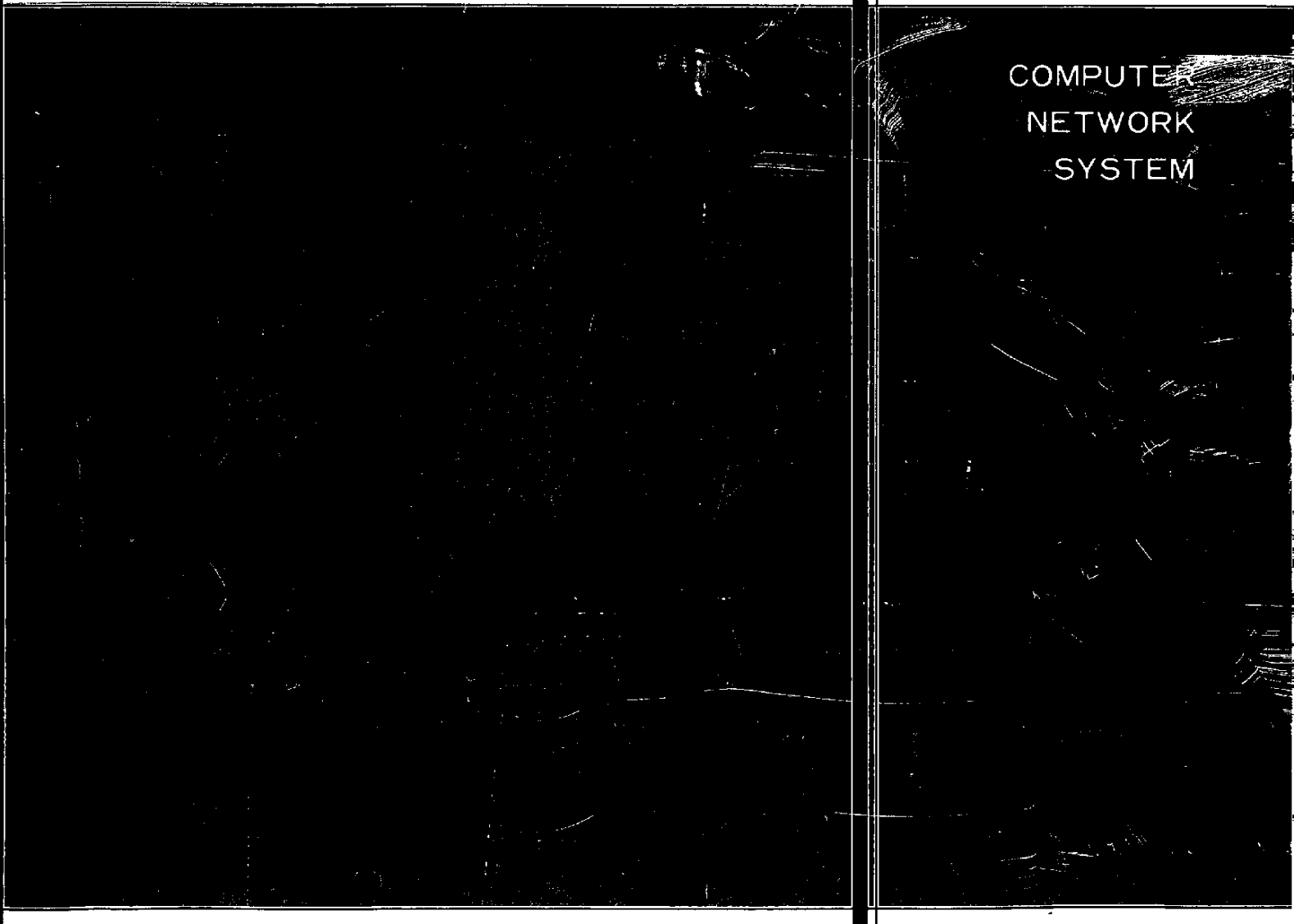
JIPDEC
ジプデック

コンピュータ・ネットワーク・システム

解　説　書



財団法人　日本情報処理開発協会

この解説書は，日本自転車振興会から競輪収益の一部である機械工業振興資金の補助を受けて，昭和５２年度情報処理普及促進のソフトウェア指導マニュアルの作成と普及の一環として作成したものであります。

はじめに

当財団は情報処理技術の研究開発の一環として，昭和４８年より「コンピュータ・ネットワーク・システム」の研究開発に着手いたしました。

複数個のコンピュータを相互に連結し，ハードウェア，ソフトウェアおよびデータ・ベースの共用を目的とするリソース・シエアリング・コンピュータ・ネットワークの開発は，米国におけるＡＲＰＡＮＥＴの成功をきつかけに，現在世界的に急速に進展しつつあります。

我が国においても，電々公社をはじめ，官庁，大学，各種企業等の多くの機関でそれぞれの立場に応じたシステム開発が実施されつつありますが，真の意味での実用的なネットワーク・システムを構築するには，まだ相当の技術蓄積と経験が必要であると考えられます。

当財団は異ったメーカの複数台のコンピュータを有しており，それらの相互間リソースを共有するための技術的問題点を具体的なインプリメントを通して解明することを目的として昭和48年度より４年間にわたりＪＩＰＮＥＴ（ JIPDEC Integrated project Network ）の研究開発を進めてまいりました。

本解説書は，この開発技術と経験を基として，コンピュータ・ネットワーク・システムについての内外における一般動向ならびにそれを構築するに当っての諸問題等を総体的に解説したものであります。

本解説書が，これからコンピュータ・ネットワーク・システムを構築しようとしている方々およびこの方面に興味をお持ちの方々に広く活用され，我が国情報処理産業発展の一助として寄与できれば幸甚と存じます。

昭和５３年３月

本解説書についてのお問合せ先

当財団　開発部ソフトウェア普及担当　　電話　（０３）４３４－８２１１

内線２０５

目　　次

1. コンピュータ・ネットワークとその実例

1．1　コンピュータ・ネットワークの定義

コンピュータ・ネットワークについてLawrence G・Robertsは次のように云っている。

「コンピュータ・ネットワークとは，お互いにその資源（ハードウエア・ソフトウエアおよびデータ）を共用し合うことができるように結合され，それぞれ独立した機能を持つコンピュータ・システムの集まりである。」

1．2　コンピュータ・ネットワークの効用

コンピュータどうしを結合し，あるいはリソースを共用することにより，コンピュータ・ネットワークは，次のような効用があるとされている。

① ハードウエア・リソースの共用

超高速ＣＰＵ，特殊プロセッサ（例えば，行列演算がパラレルに実行できる，array processor），大容量記憶装置，漢字入出力や高性能プロッタなどの特殊入出力周辺装置など高価なハードウエア・リソースを共用し，経済性を向上させる。

② ソフトウエア・リソースの共用

言語プロセッサ，ユーティリティ・プログラム，各種アプリケーション・プログラムなどの特徴あるソフトウエアを相互に利用しあうことにより，ソフトウエア開発の二重投資を避けうる。

③ データ・ベースの共用

科学技術文献をはじめとする各種のデータ・ベースの共用で，今後もっとも特色あるリソースは，データ・ベースであろうといわれる。

④ コンピュータ・センターの専門化が可能

各コンピュータ・センタが各々のユーザのすべての要求を満足させようとすると，いずれも各種の機能をもった大型センタにせねばならぬが，それぞれのセンタがその得意とする分野で，より高度のアプリケーションを充実し，それらの専門化したセンタを相互に連結することにより，そのネットワークは非常に質の高いリソースを供給しうることとなる。

⑤ 複数システムへのアクセス

例えば，１つの端末から複数のコンピュータのリモート・バッチやＴＳＳ処理（タイム・シェアリング処理）を利用しうる。

⑥ 信頼性の向上

あるコンピュータ・システムがダウンしても，ネットワーク内の他のシステムの代替が可

能となるので，総合的な信頼性が向上する。ファイルに関しても，とくに重要なファイルはバック・アップを他のシステムに置くことにより，破壊を防ぐことができる。

⑦ 負荷の分担

非常に忙しいシステムと比較的ヒマなシステムとのあいだで，負荷を分担し合い平均化することにより，全体的な稼働率を高める。

⑧ システム間の互換性の向上

ハード，ソフト，データ・ベースなど，各種リソースの異機種間の互換性を内部処理的に実現しうる。

⑨ 共同作業の道具

大学，研究機関などのあいだで，各自のコンピュータあるいは端末を通して交信を行ないながら，共同研究を行なうことができる。

⑩ 時差の利用

広域のネットワークでは，時差を利用して総合的なシステムの稼働率を上げることが可能となる。

⑪ 安全性の確保

ハードウェア，データ・ベースなどの貴重なリソースを，何らかの災害から守るために，地理的に分散しておく方が危険率が少ない。企業内ネットワークにおいても最近はこの必要性が強調されるようになった。

⑫ 仮想コンピュータ・システムの実現

コンピュータ・ネットワークのより高度なコントロール技術によれば，ネットワーク全体を1つの大きな仮想コンピュータ・システムとみなし，入力したジョブは，その総合機能の中で，自動的に最適な処理が行なわれるという使用法も可能となろう。

また，データに関しても，分散あるいは集中されたその物理的所在や構造を意識せずにアクセスしうる，仮想データ・ベースの実現も考えられる。

1．3 コンピュータ・ネットワークの要素と形態

コンピュータ・ネットワークはそれぞれ独立したコンピュータ・システムであるホスト（HOST）と，ホスト間の通信をつかさどるデータ交換網とからなる，結合に際し，図1－1に示すようにホスト（図のH）や端末（図のT）をデータ交換網に結合する節となるところをノード（NODE）（図のN）と呼び，このノード自身が1種のコンピュタ機能（ノード・プロセッサまたはCCPと呼ぶ）を持つことが多い。ノード・プロセッサを介することにより，次のような利点があるとされている。

① 異機種ホスト間のインタフェースの調整のしやすさ

② ホストの負荷の軽減

③ ネットワーク拡張の容易さ

④ ネットワークの信頼性の向上

ＡＲＰＡＮＥＴやＣＹＣＬＡＤＥＳ等ではノードを含むこのデータ交換網をサブネット（Ｓubnet）と呼んでいるが，このサブネットの形態により，ネットワークは幾つかの型に分類される。図１－２に代表的な型を示す。

集中型はセントラル・ノードがデータ交換の制御を集中管理する。すべてのトラフィックがセントラル・ノードを通るため，コントロールはやりやすく処理効率は一般によいが，逆に信頼性の点からはセントラル・ノードのダウンが全ネットワークのダウンにつながる。

分散型は図に示すように中心となるノードがなく，すべてのノードは同じ資格で制御を平等に分散して行なう。分散型の第１の特長は信頼性にあり，ノードはそれにいたる経路を少なくとも２つはもつため，一部のノードや回線が故障してもネットワーク全体をダウンさせることはない。また，拡張性も比較的高い。

ループ型（またはリング型）では，各ノードは環状の回線上を回つてくるデータの宛名を調べ，自分宛のものがくればそれを取り込む。この型は回線長が最短ですみ，コントロールも簡単でオーバヘッドも少ないが，トラフィック量が多い場合は回線容量の不足をきたすことがある。

以上のような各種の基本型のほかにこれらの混合型も存在する。例えば，図１－３

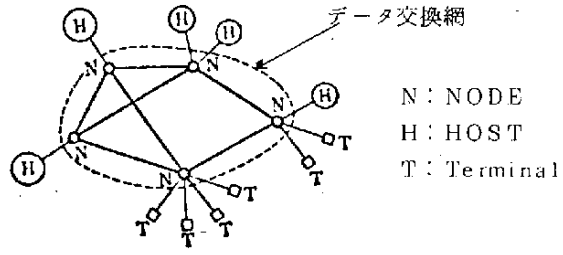
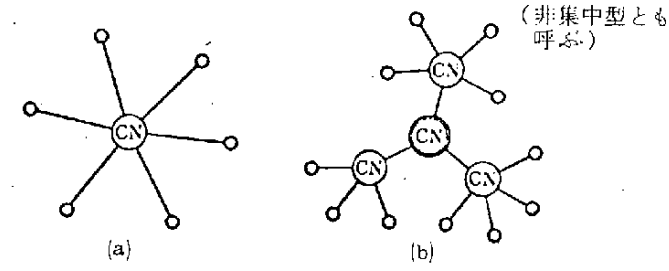
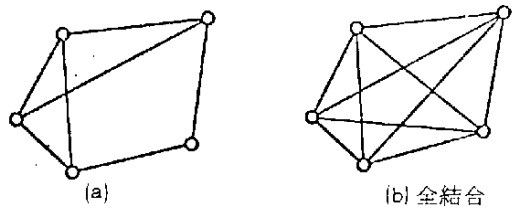


図１－１　コンピュータ・ネットワークの構成要素

A 集中型(Star or Centralized)



B 分散型（Distributed）



C ループ／リング型(Loop or Ring)

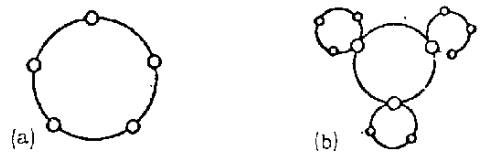


図1－２　ネットワークの型

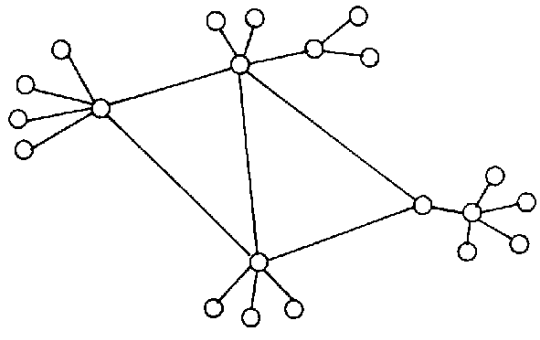


図1－３　混　合　型

に示すように幹線は分散型であるが，各ノードはローカルなネットワークのセントラル・ノードの役割をもつたものも考えられ，幹線間は信頼性を，ローカルには効率を重んじた構造といえよう。

図１－４にネットワークの構成例を示す。

ホストや端末を交換網に結合する際に，ＦＥＰ（Ｆront Ｅnd Ｐrocessor）やＴＣＰ（Ｔerminal Control Processor）を介す形，既存のＴＳＳネットワーク自身がノードを介して交換網に接続される形など，多彩な構成が考えられる。

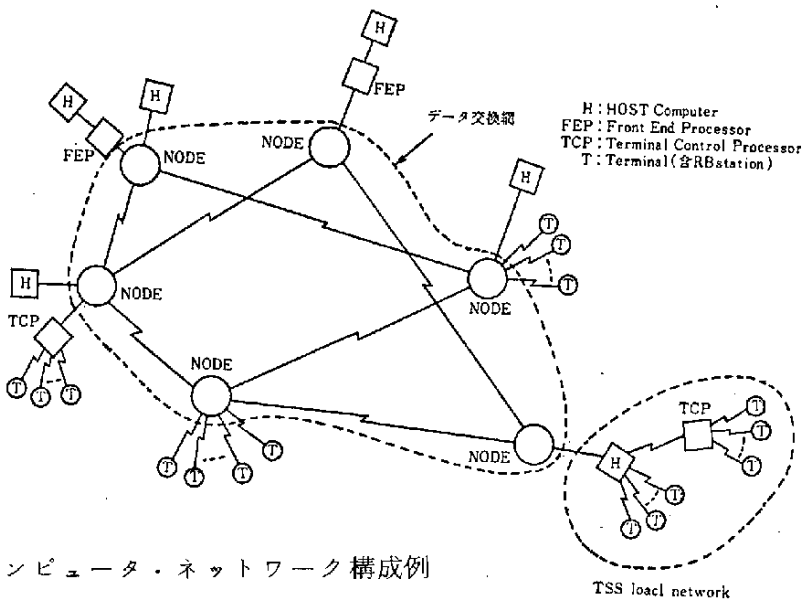


図１－４　コンピュータ・ネットワーク構成例

なお，ネットワークに参加するホストが同一機種のみのものを均質（homogeneous）ネットワークと呼び，異なつた機種どおしを結合したものを不均質または異機種（heterogeneous）ネットワークと呼ぶ。今後のネットワークは，後者がより一般的となるであろう。

１．４　コンピュータ・ネットワークにおける処理パターンの例

表１－１に，コンピュータ・ネットワークにおける処理パターンの例を示す。

①から④までは２つのホストを対象としたもので，①はＨＯＳＴ－ＡからＨＯＳＴ－Ｂに，ある実行形式プログラム（ＯＰ）とデータ・ファイル（Ｄ）を使用して処理（Ｅ）を行なうことを依頼し，結果をＡに返送する形である。②はその場合データ・ファイルがＡにあり，あらかじめ，あるいは処理中にデータをＡからＢに送る形である。③はデータ・ファイルだけでなく，ソース・プログラム・ファイル（ＳＰ）もＡにあり，これをＢに送ってコンパイル(Ｃ)も依頼する。また④は，Ｂにあるソース・プログラムとデータとをＡに取り寄せてコンパイルおよび実行を行なう。

次の⑤から⑨は，Ａ，Ｂ，Ｃの３つのホストにわたって処理を行なう場合の例で，例えば，

⑤は，Cにあるデータ・ファイルを使用して，Bにプログラムで処理をすることをAから依頼する形である。また，⑨は，Cにある特殊出力装置（例えば，漢字プリンタ）を共用するような場合の例で，Aにあるデータを Bに送って処理を行ない，結果をCに出力する。

⑩から⑭は，リモート・バッチや会話型端末からネットワークにアクセスする形で，端末（T）はここでは制御情報の入力（S）と結果の出力（O）の両機能を含むものとする。

このうち⑩から⑫は，あるホスト（この場合A）につながる端末から他ホスト（この場合B）の会話型あるいはリモート・バッチ処理の使用であり，⑬，⑭は特定のホストを経由せず，交換網に直接結合された端末からのネットワーク使用である。

⑮，⑯，⑰は，ネットワークのやや進んだ使用法であり，⑮，⑯は2つのホストにデータ・ファイルが分散しており（D1とD2），⑰は処理自体が2つのホストで分担（E1とE2）されている。

表1－1　処理パターンの例

	HOST-A	HOST-B	HOST-C	HOST-D	端末
①	S,O	E,OP,D			
②	S,O,D	E,OP			
③	S,O,D,SP	E,C			
④	S,O,C,E	SP,D			
⑤	S,O	E,OP	D		
⑥	S,O,D	E,C	SP		
⑦	S,O	SP,D	E,C		
⑧	S,O,SP	D	E,C		
⑨	S,D	E,OP	O		
⑩	T,D	E,OP			
⑪	T	E,OP,D			
⑫	T	E,OP	D		
⑬	E,OP,D				T
⑭	E,OP	D			T
⑮	E,OP	D1	D2		T
⑯	T	E,OP	D1	D2	
⑰	S,O	E1,OP1	E2,OP2	D	

S：制御情報の入力
O：結果の出力
E（またはE1，E2）：実　行
C：コンパイルまたはアセンブル
SP：ソース・プログラム・ファイル
OP（またはOP1，OP2）：実行形式プログラム・ファイル
D（またはD1，D2）：データ・ファイル
T：端末（SとOを含む）

1．5　コンピュータ・ネットワークの処理機能

コンピュータ・ネットワークにおいて前節で述べたような各種の処理を行なうためには，データ交換網とホスト側とで機能を分担して処理を行なうこととなる。

ネットワーク内のあるホストから他のホストへ，あるいはある端末とホスト間のデータやメッセージの転送にかゝわる処理は当然データ交換網の分担となり，ホスト相互間コミュニケーション，TSS処理，リモート・バッチ処理等のプログラム間での交信機能はホスト側の分担となる。以下にデータ交換網側とホスト側のそれぞれの処理機能の概要を示す。

1. 5. 1　データ交換網とその機能

A．データ交換網

(1)　回線交換とパケット交換

交換方式の分類は以下のとおりである。

- 交換方式
  - 回線交換
    - 空間分割
    - 時分割
  - 蓄積交換
    - メッセージ交換
    - パケット交換

このうちコンピュータ・ネットワークでは回線交換とパケット交換という対照でとらえることが多い。

通常の電話交換のように，具体的な通話要求が生じた場合，発信地と受信地の間に物理的な伝送路が設定され，それを通してメッセージが送信される方式を回線交換と呼ぶ。

これに対しパケット交換とは，発信地から出されるメッセージをパケット（packet；小包の意）と呼ぶ行先アドレスをつけた幾つかの伝送単位に分割し，それぞれのパケットをその時点のトラフィックや障害状況等に応じて任意のルートを通して目的地に転送し，目的地で再びそれを元のメッセージに組立てる方式である。

この両者の特性は表１－２のように比較することができる。

表１－２　回線交換とパケット交換の比較

項　目	回線交換	パケット交換
伝送遅延時間	バラツキが少なく平均的に小	小であるが，ルーティング・メカニズム，トラフィック状況等によりバラツキがある。
接続時間	大	小
データの信頼性	中程度，データの蓄積を行なわないので，交換機障害の影響は少ない。	大，ノード間でのチェックが可能である。
回線の利用度	データの送信時に通話路を占有する。	平均的な利用度大
交換施設の費用	小	大
利用面の特性	両端が物理的に直結されているため，リアルタイムの即時通信が可能。多量のデータの連続転送，アナログデータの転送等に向く。	蓄積交換機能による符号変換，速度変換，同報通信，メイルボックス機能等も可能。異なる速度や手順を持つ端末やコンピュータ間通信に適する。

(2) 公共データ交換網

個々のネットワークが専用のサブネットを持たず，汎用かつ公共的なデータ交換網を多目的に共用することにより，経済性を図ろうという計画が各国で盛んに行なわれている。主なものを表１－３に示す。

表１－３　各国のデータ交換網

国　名	名　　称	交換方式	開発主体	稼働開始
英　国	EPSS※(Experimental packet Switched Service)	パケット交換	BPO(British Post Office)	1976
フランス	RCP※	パケット交換	フランスPTT	1974
	TRANSPAC	パケット交換	フランスPTT	1978
西ドイツ	EDS(Electronic Data Switching System)	回線交換	西ドイツDBP	1977
カナダ	DATAPAC	パケット交換	TCTS(Trans CANADA Telephone System)	1976
	INFOSWICH	回線+パケット	CNCP(Canadian National Railways & Canadian Paciffic)	1976
スペイン	CTNE	パケット交換	CTNE(Compañia Telefonica Nacional de España	1972
北　欧	NPDN(Nordic Pubric Data Network)	回線交換	北欧４ケ国のPTT	1978
日　本	DDX	パケット交換 回線交換	NTT	1979
	VENUS	パケット交換	KDD	1979

※　実験網

B．データ交換網の機能

公共網であれ，専用網であれ，データ交換網に対する基本的要求はその信頼性，効率性および拡張性にある。

まず，信頼性の観点からは，なんらかのノードまたはホスト，あるいは回線の一部などの

障害をできるだけすみやかに検出し，それら障害部分を論理的に切り離し，ネットワーク全体に与える影響を最少にするよう，すぐれたフェイルソフト機能を備えることが望ましい。

次に効率の面からは，会話型処理の応答速度は充分速く，かつ大量データ転送においては充分なスループット（ throuput ）が得られることが望ましい。ネットワークを通して会話型処理を行なっている端末ユーザが，例えば，２秒以内のレスポンスを要求した場合，２秒のうち１秒がその会話型サービスを行なっているホストのローカル・ネットワークで消費されるとすると，サブネット内は少なくとも１秒以内で通過せねばならない。

一方，大量のデータ転送を行なう場合には，全体としてのスループット，即ち，ノードおよび回線を含め，単位時間当たりにどれだけの伝送処理が行なえるかが問題となる。

両者を満足させる手段としては，例えば，会話型用には短いメッセージ，データ転送用には長いメッセージというように，使用目的に応じてメッセージ長を可変とし，それぞれにふさわしいコントロールを行なう，あるいはより積極的には，会話型にはパケット交換方式を，データ転送には回線交換方式をという切り換えを行なうことも考えられる。

また，拡張性の面からは，ネットワークの拡張やトポロジの変更に対し，ノード・プロセッサのソフトウェアが多くの影響を受けぬことが要求される。

(1) ノード・プロセッサ

ノード・プロセッサがどの程度のハードウエア機能をもつべきかは，ホストとデータ交換網の負荷分担の問題とも関連するが，一応の目安として，データの高速転送および伝送の信頼性の確保にかかわるいっさいの処理は，データ交換網の分担とするとすれば，少なくともノード・プロセッサはまずプログラマブル（Programmable）であると同時に，蓄積交換の目的を果たすためのバッファ・エリアとして何がしかのメモリ容量をもつことが必要となる。これらの要求から，従来は汎用小型コンピュータをノード・プロセッサとして利用することが多かつた。

ノード・プロセッサの性能は，直接ネットワーク全体の効率に大きな影響を与える。したがって最近は，通信制御を目的とした，専用のプロセッサが出現し，例えば，データ交換の基本機能のハードウェア化，割込み処理機能の高度化，ＲＯＭ（Read Only Memory）の利用など，低価格でかつ高性能の通信制御プロセッサ（ＣＣＰ＝Communication Control Processor）が使用されるようになつた。

この例として，１９７２年に米国でLockheed SUEという新しいプロセッサが発表された。これは，従来ソフトで行なっていた割込み処理をハード化したともいえる画期的なもので，PLURIBUS IMPとしてARPANETの一部で使用されている。ところで，異機種ネットワークにおけるノード・プロセッサは，各ホストに固有のものとするか，ホストには無関係な共通のものとするかが問題となる。ホストとのインタフェース整合の容易さ，あ

るいはホストの負荷分担をかねてノード・プロセッサにＦＥＰ（Ｆront　End Processor)としての機能をもたせ得る可能性，といった点からすると，ホスト固有のノード・プロセッサの採用も考えられぬことではない。一方，ネットワークの拡張性，データ交換網の早期の安定性および信頼性というような点からみると，すべてのノード・プロセッサは同一機種であり，しかも内部のソフトウェアも同一であることが望ましい。現在では後者の考え方が一般的である。そして前述のホスト負荷の軽減およびインタフェース整合の容易化というような目的のためにはノード・プロセッサとホストとの間にさらにもう1つホスト固有のＦＥＰを置くという形で解決する例が多い。

勿論データ交換網が前節で述べた公共網である場合はユーザにとつてノード・プロセッサに対する選択の余地はなく，ノード・プロセッサはコンピュータというよりは，交換網の一部に組込まれた特殊ハードウェアとして扱われることとなろう。

(2)　パケット交換網の機能

以下分散型パケット交換網の場合を中心に述べる。

①　パケット分割と再組立

ホスト間で転送されるメッセージは任意の長さを持つが，交換網内で定めた一定長に分割し，行先アドレスを付したパケットとして発送する。このメッセージのパケット化と，目的地における再組立の機能である。既存のネットワークにおいては，これをホスト側の分担としているものもある。

②　ルーティング（routing）

発信地ノードから目的地ノードへメッセージを伝送するルートを定めることで，次のような要求がある。

(a)　最短時間で目的地に達すること。

(b)　ネットワークの拡張やトポロジの変化に影響せぬアルゴリズムであること。これは，実際の拡張や変更だけでなくネットワークの一部，例えば，あるノードやある回線の障害により一時的にネットワーク・トポロジが変化することも含む。

(c)　トラフィックの変動によく追従しうること。

(d)　ノードの負担が軽いこと。これには，アルゴリズムの複雑さによる処理負荷と，パケット・コピーの保存のためのスペース上の負荷とがある。

(e)　ルーティング情報伝達のためのトラフィック・オーバヘッドを過度に増加させぬこと。

(f)　ルートを平均的に使用すること。

分散型ネットワークでは，発信地ノードから目的地ノードへ到着する経路は最低2通りはあるというのが特徴である。これが集中型より信頼性が高いといわれるゆえん

でもある。しかし，実はそのためにどこを通ってゆけば一番よいかというルートを選択する仕事が必要となる。ルーティングの研究というのは，コンピュータ・ネットワーク以前からも種々行なわれてきたが，現在のネットワークで採用されている代表的な手法は固定法（ fixed routing ）と適応性（ adaptive routing ）である。

固定法は２つのホスト間のメッセージの転送ルートが予め定められているもので，適応法はその時点のトラフィックや網内の障害状況により適宜ルートを変更する方法である。

③ フロー・コントロール（ flow control ）

データ交換網へのメッセージの流入を無制限に許すと，データ交換網内のコンジェスション（ congestion）や各種のロック・アップ（ lock up ）が発生し，ネットワークが硬直状態におちいる可能性がある。これを回避するため，データ交換網へ流入するトラフィックを規制するのがフロー・コントロールである。

フロー・コントロールは全体としてのトラフィックをある程度規制した上で，更に各ノード内のバッファ管理のいきとどいたメカニズムによる立体的な両面作戦が必要である。

④ シークェンシング（ sequencing ）

データ交換網内のトラフィックや障害の状況によって先に出発したメッセージが先に目的地に到着するとは限らない。その順序に意味がある転送では，目的地において，到着メッセージの順序制御を行なう必要が生ずる。

これがシークェンシングであり，既存のネットワークにおいてはホスト自体でこれを行なうものもある。いづれにしろ各メッセージに番号をつけ順序を制御し，それによって伝送エラー制御もついでに行なわれることとなる。

⑤ 障害対策

データ交換網の障害対策に対する基本的態度は，次のようなものである。

(a) フェイルソフト（ fail soft ）機能が万全であること。即ち，前述のように，障害部分を取り除いた環境下で可能なかぎり運転が続行されること。

(b) ネットワークを構成する，それぞれ異なった要素間，例えば，ホストとデータ交換網，ノードと回線など，障害時の切り分けが容易に行なわれること。

(c) 障害発生やその回復手続きの過程において，他の部分の運転に影響を与えぬこと。

(d) 障害の検出，その回復のテストなどが，可能なかぎり自動的に行なわれること。

また，障害対策の対象は次のようなものである。

ａ．回線障害

ｂ．ノード障害

ｃ．ホスト障害

d．メッセージの紛失・重複

e．ノード内バッファの充満

f．デイスコネクション・ノード（経路上のノードや回線の障害のため，如何なるルートを通つても到達し得ないノード）の検出

⑥ その他

ネットワーク内のトラフィックのモニタリング機能，障害記録管理，交換網テストのための擬似トラフィック発生機能，ノード障害時のプログラム再ロード機能等がある。

1.5.2 ホスト側のネットワーク・コントロール機能

ここではＦＥＰもホスト側の一部として考える。

A．ホストとデータ交換網の機能分担

コンピュータ・ネットワークは，それぞれ異なったリソースをもつ独立したホスト・コンピュータ群と，前号で述べたデータ交換網とから成り立つ。そこで，ホストとデータ交換網の機能分担はどうあるべきかという問題が発生する。ホストとデータ交換網の機能分担の基本線は，ホストが各種リソースの供給源であるのに対して，データ交換網はホスト間の交信や相互のデータ転送のための通信網の役割をになう。

両者の分担を定めるにあたって，まず留意すべき点は

① ホストの負荷を減じ，結合のためにホスト本来のリソースを過度に浪費しないこと。

② ホストの障害がネットワーク全体に影響を与えないのと同時に，ネットワークの障害が独立システムとしてのホストの稼働になんら影響を与えないこと。

③ 結合にともなうノード・プロセッサ，ＦＥＰ（Ｆront End Ｐrocessor）などの新たな機器設備のコストが適度であること。

④ 利用目的に適したネットワーク処理効率が得られること。

などがある。

現在のネットワークで問題となっている具体的な負荷分担の項目には，次のようなものがある。

① メッセージのパケット化と，その再組立

② メッセージのエラー伝送の処理

誤った目的地への伝送や，メッセージの重複などの検出，処理

③ 到着メッセージの順序の制御

④ ネットワーク・アクセスに対するユーザの資格チエック

⑤ アカウンティング

このうち④⑤は，既存のネットワークでは，ほとんど特殊ホスト（ＮＣＣ）が受持って

いる。ただし，専用網では交換網自身の分担としているものもある。

また①②③は，ホストの負荷の軽減とデータ交換網の効率と，どちらを重視するかによって定まり，ARPANETではデータ交換網で,CYCLADESではホストで分担している。ARPANETでは，ネットワーク全体の障害状況としてはデータ交換網に比べホストの障害がかなり多いという実積経験から，①②③のような処理をデータ交換網の責任とすることにより，ネットワーク全体の信頼性を向上させる目的も果たしていると思われる。一方，CYCLADESは，前述の公共網や，広域の国際ネットワークを指向しているため，データ交換網になるべく多くの負担をかけない主義のあらわれであろうと推察される。

B．ホスト側の機能

ホスト側に存在するネットワークのコントロール・プログラム（ＮＣＰ）は各ホストのＯＳ（オペレーティング・システム）の一部，あるいはそれに準ずる形で存在することとなる。

異機種結合によるネットワークでは，それぞれまったく独自な思想によって作られている異なったＯＳとのインタフェースを充分考慮したうえで，ＮＣＰをどのように実現すべきかを検討しなくてはならない。

以下にホスト側に持つべき主な機能を示す。

(1) プロセス間のコミュニケーション

各ホストのジョブの処理単位は，ジョブ，ジョブ・ステップ，タスクなどと，一般にそれぞれ異なる可能性があるが，ネットワーク上ではいずれもこれをプロセス（process）という概念で統一する。そしてホスト間コミュニケーションは，それぞれのホスト内のプロセスどうしがＮＣＰを介してコミュニケーションを行なう形で実現される。即ち，単一コンピュータ内では図１－５①に示すように，いくつかのプロセスがＯＳを経由して相互に連結をとりながら，一般にはマルチ処理が行なわれている。②は２つのコンピュータ間における通信で，各ホスト内のプロセス自体は，相手プロセスが他のコンピュータに存在しているのか自分のコンピュータ内にいるのかを区別する必要はなく，ＮＣＰがそれぞれのＯＳとのインタフェースをとり，データ交換網を通して相手ホストのプロセスと連絡をとればよい。結局，ホスト間コミュニケーションに対するＮＣＰの任務は

① 各ホストのバッチ，リモート・バッチ，ＴＳＳなどの各処理単位を一元的に扱うことを可能とするとともに

② 他ホストのプロセスも自ホストのプロセスに準じた形で扱い得る機能をサービスすることである。

具体的には各ホストのプロセスの識別法，２つのプロセスの間での交信の開始，終了，両者間のデータの授受の方法，プロセス間のフロー制御等が細かくとりきめられている。

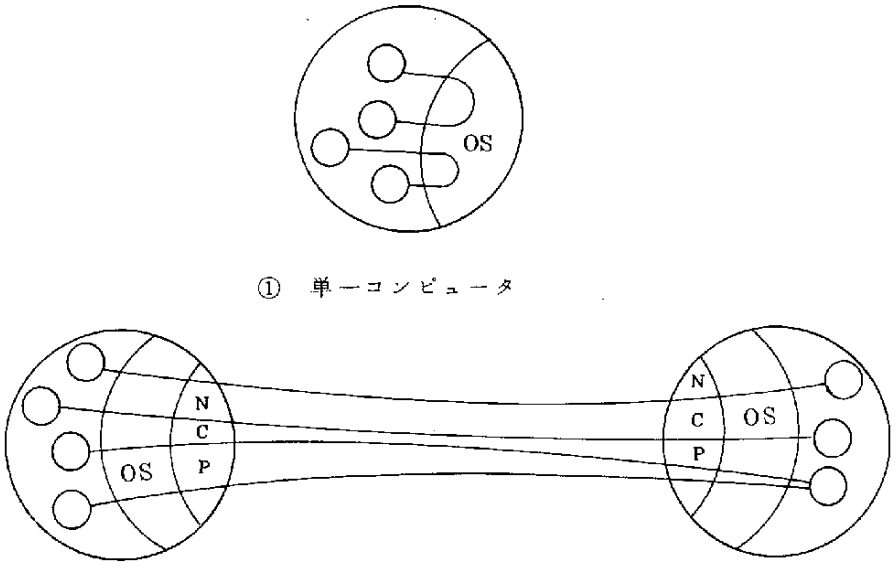


図1—5 プロセス間通信

(2) 仮想端末処理

コンピュータ・ネットワークの代表的機能の1つは，ある1つの端末から，種々のＴＳＳサービスや，データ・ベース・サービスに自由にアクセスし得ることである。しかしながら，端末機の種類や機能は多様であり，いかなる端末にも対応し得る手段として現在最も広く採用されている方法は，標準的な仮想端末機能を設定し，図1－6に示すようにネットワーク内では各種ホストは相手をひとまずその標準仮想端末とみなして処理を行ない，現実の端末えのマッピングを行なうモジュールを別途設けて対応させるという方法をとっている。また，各種のリモート・バッチ・ステーションに対してもほぼ同様な考え方で処理が行なわれている。

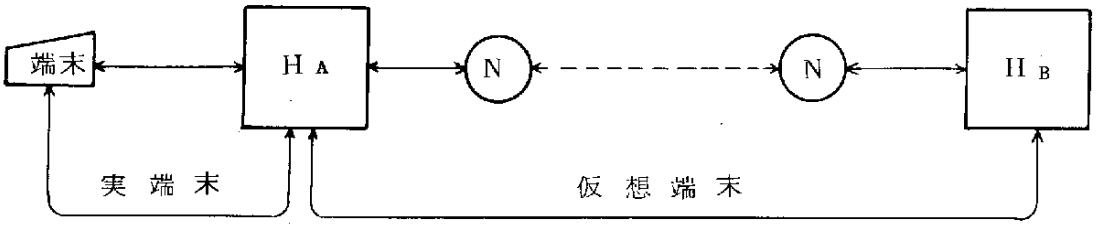


図1—6 仮想端末と実端末の領域

(3) ファイルの互換性の確保

2つの異なるホスト間でファイルを転送し合い，相互に利用し合うことを考えると，先ず両者の間でファイルの互換性がなければむづかしい。ファイルの仕様は各ホストの

データ管理機能によって左右されているため，ここでも仮想ファイルの考え方が必要となる。また，基本的要素としてコード系，バイト長，ワード長等種々の変換が行なわれる。

(4) 会話型／リモート・バッチ処理の実現

異機種ホスト間で，会話型処理（ＴＳＳ処理）あるいはリモート・バッチ処理を可能とする。

(5) ユーザ・インタフェース

ネットワーク・ユーザがそれぞれの目的でネットワークにアクセスする場合の各種コマンドをサポートせねばならない。図１－７はその例である。

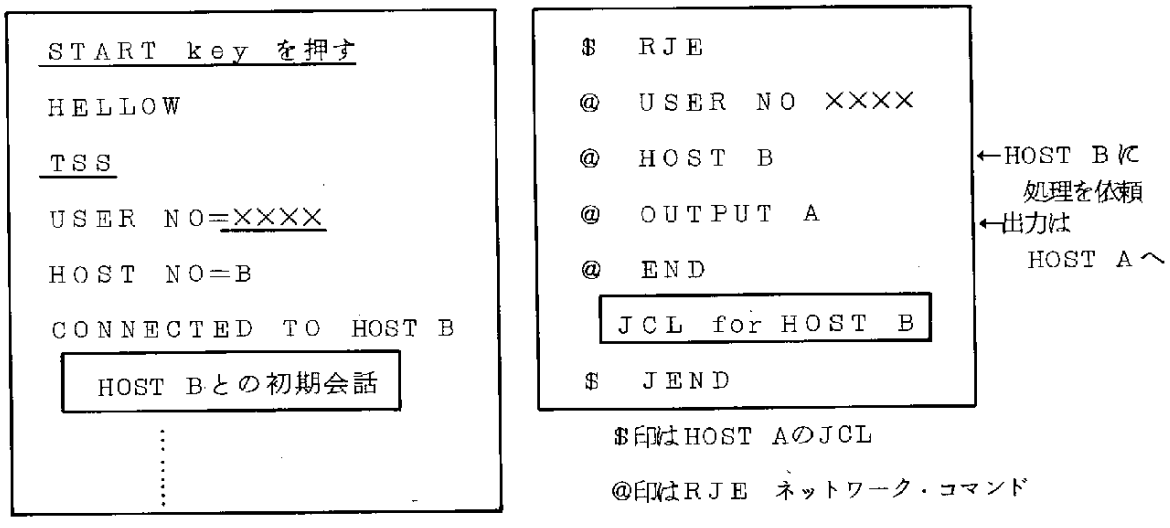


図１－７ ユーザ・コマンドの例

1.5.3　プロトコル

以上のようなネットワークの諸機能はプロトコル（Protocol）と呼ぶ規約によりその論理機能が厳密に定められる。

プロトコルは図1－8に示すように隣り合ったノード間の伝送レベルのものからユーザ・プロセス間の高位のものまで，いくつかの階層に分けられる。

多くの要素からなるコンピュータ・ネットワークのような複雑なシステムにおいては，このような論理的階層化によって，あるレベルのプロトコルの変更が他のレベルに影響を与えぬと同時に，より下位のレベルの機能をブラック・ボックス化（Transperent）し得るメリットがある。

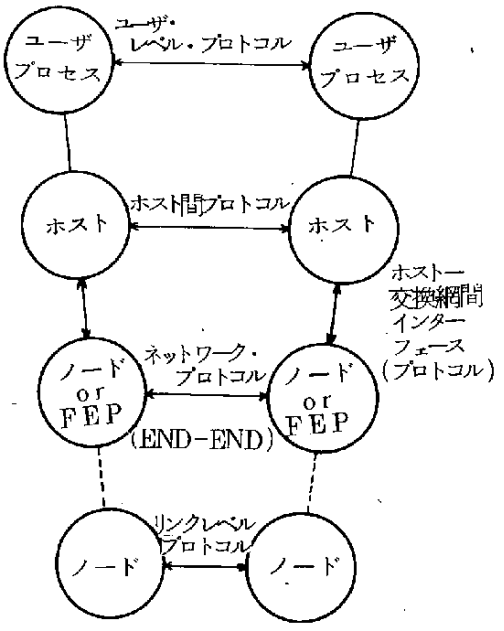


図1－8　プロトコルの階層の例

表1－4にプロトコルの機能を示す。

なお，現在これらのプロトコルの国際標準化活動が盛んに行なわれており，リンク・レベルのHDLC（High Level Data Link Control）がISOで，ホストや端末とパケット網とのインタフェースX25がCCITT（International Telegraph and Telephone Consultative Committee）でそれぞれ国際標準案としてほぼ固まってきている。

表１－４　プロトコルの機能

(1)　リンク・レベル・プロトコル

相隣る２つのノード間のプロトコル

①　パケットのフォーマットと制御情報の意味付け

②　パケット伝送の確認（ＡＣＫ）

③　伝送エラーの検出と再送　　等

(2)　ネットワーク・レベル・プロトコル

発信地と目的地のプロトコル

①　メッセージのパケット分割と再組立

②　発信地，目的地間の伝送確認と再送

③　メッセージの順序制御

④　交換網内のフロー制御

⑤　ルーティング

⑥　優先制御　　等

(3)　ホスト・レベル・プロトコル

ホスト間の交信にともなうプロトコル

①　プロセス間のコネクション確立と閉鎖

②　プロセス間のデータ転送

③　ホスト間のフロー制御

④　割込制御（優先制御）

⑤　相手ホストのテスト機能　　等

(4)　ユーザ・レベル・プロトコル

一般ユーザの直接使用するプロトコル

代表的なもの

①　会話型処理（ＴＳＳ処理）プロトコル

②　リモート・バッチ処理プロトコル

③　ファイル転送プロトコル　　等

しかしながら一方，これらのプロトコル階層と，ホストや端末をデータ交換網に接続する場合のインタフェース，各プロトコル処理プログラムの各種ハードウェア装置えのマッピング等を含め，最近はネットワーク・アーキテクチャとして表１－５に示すように，各コンピュータ・メーカが独自の構想を打ち出し，標準化とは矛盾した傾向も強くなってきている。

表１－５　各社のネットワーク・アーキテクチャ

メーカ名	ネットワーク・アーキテクチャの名称	発表時期
IBM	SNA (Systems Network Architecture)	1974. 9 1976. 10 (拡張)
DEC	DNA (Digital Network Architecture)	1975. 4
バロース	DNS (Decentralized data processing Network System)	1976. 6
ユニバック	DCA (Distributed Communication Architecture)	1976. 11
レイセオン	PNA (Processing terminal Network Architecture)	1976. 11
東芝	ANSA (Advanced Network System Architecture)	1976. 12
日電	DINA (Distributed Information processing N. A.)	1976. 12
電電公社	DCNA (Data Communication Network Architecture)	1977. 3
富士通/日立	MSNA (M-Series Network Architecture)	1977. 3
沖	DONA (Decentralized Open Network Architecture)	1977. 3
ニクスドルフ	NCN (Nixdorf Communication Network)	1977. 3
富士通	FNA (Fujitsu Network Architecture)	1977. 5
三菱	MNA (Multi-share Network Architecture)	1977. 6
日立	HNA (Hitachi Network Architecture)	1977. 9

1．6　コンピュータ・ネットワークの例

1.6.1　海外における例

表1－6に海外の主なコンピュータ・ネットワークの例を示す。ここではとくに広域，公共のものを選び，銀行間，航空会社間など，専用のものは除いた。米国では国防省ARPAが推進役となり，１９６０年代後半から開発に着手したARPANETに先導され，その商用版ともいえるTELENETと，従来からのＴＳＳネットワークの発展形としてのTYMNETやCYBERNET 等が代表的なものとしてあげられる。商用のものはいづれもヨーロッパえの拡張を果たし，除々に他の国々をも含む国際ネットワークえの道をたどっている。

一方ヨーロッパでは英国の国立研究機関であるＮＰＬ（National Physical Laboratory）が技術的に先導役となり，フランスのＩＲＩＡがＣＹＣＬＡＤＥＳの開発で広域的実験研究の成果をあげ，これらが中心となって当初から欧州諸国内の国際情報ネットワーク建設に対する関心が強い。

表１－６　コンピュータ・ネットワークの例

名　称	開発主体	地　域	目　的	稼　動
ARPANET	米国防省ARPA	米　欧	研　究　用	１９７０
CYBERNET	Ｃ　Ｄ　Ｃ	米　欧	商用 TSS	１９７４
TYMNET	TYMNET　社	米　欧	商用 TSS	１９７６
TELENET	TELENET 社	米英加メキシコ	商　用　VAN	１９７５
CYCLADES	フランスIRIA	フ　ラ　ン　ス	研　究　用	１９７４
Ｅ　Ｉ　Ｎ	ＥＣ 研究機関	欧　１４カ国	リソース・シェアリング	１９７６
EURONET	ＥＣ　ＰＴＴ	欧　９カ国	情報ネットワーク	１９７８

(1)　ARPANET：現在の世界的なコンピュータ・ネットワークブームの元祖ともいうべきもので，米国防省が数百万ドルをかけ１９６０年代後半に開発した画期的なシステム。６０以上のノードと百数十台のホストが接続され，米国内の主要大学，研究機関の殆んどがこれに参加していた。ARPANETはほぼその目的を達成し，ユーザの多くはTELENETに移っているといわれる。図１－９，１０にその地理的およびロジカル・マップを示す。

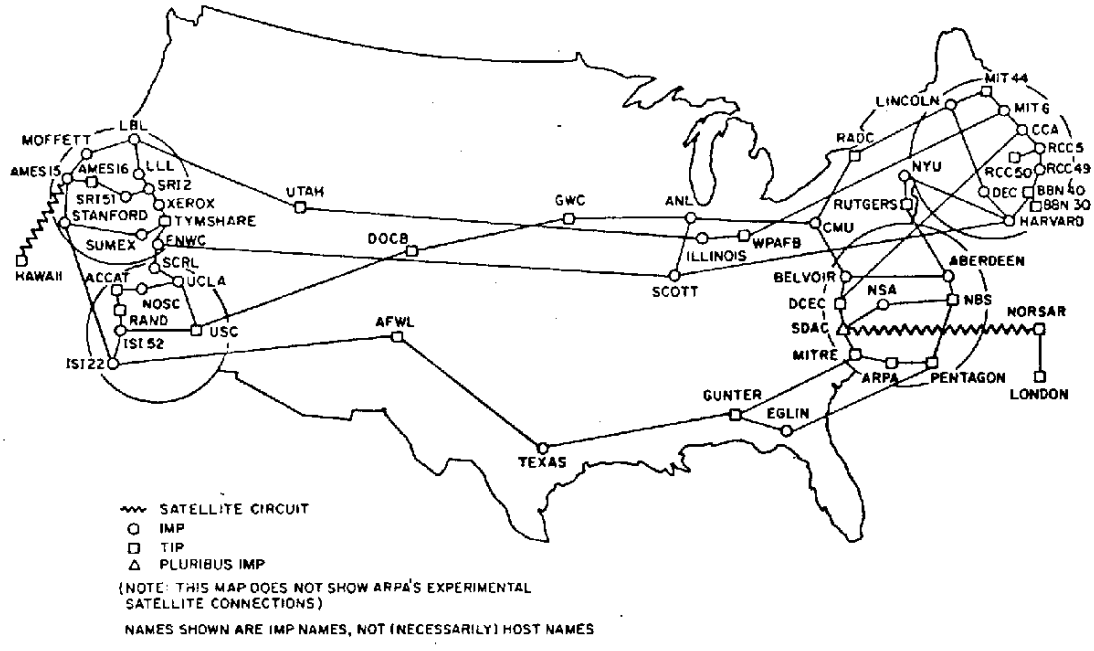


図1—9 ARPANET geographic map, April 1977

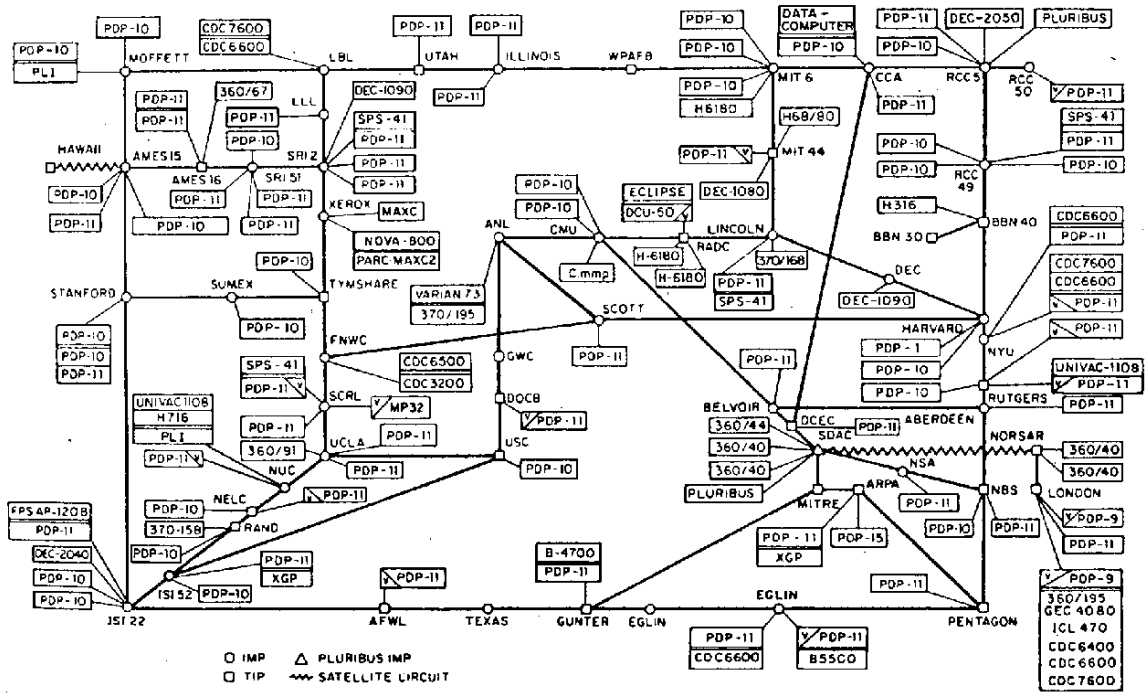


図1—10 ARPANET logical map, March 1977

ＡＲＰＡＮＥＴ の功積は色々あるが，とくに

① 異機種間のリソースシェアの実施

② パケット交換方式の実用化

③ プロトルコ体系の確立

④ ノード・プロセッサ方式（ＩＭＰ）の実現

等は特筆すべきであろう。

(2) ＣＹＢＥＲＮＥＴ：米国西部（Palalto, Losangels），中部（Mineapolis Detroit）東部（Boston, Newyork, Washington）南部（Heuston）等全米の１０数個の都市にＣＤＣ７６００，６６００や３３００のホストを持つＣＤＣの商用ＴＳＳネットワークで，同一メーカ機種という特長を生かし，時差を利用して相互のロード・レベリング，即ち，各ホストの負荷の均一化をはかり効率を上げている点が注目される。また，あるホストのダウン時に他のホストの代替が容易であるのも大きな利点である。１９７５年時点でホスト数１３，ノード数３６と云われ，回線交換方式で最高４０.８ＫＢＰＳの速度を持つ。また，最近ＩＢＭのマシンも幾つか接続されている。

(3) ＴＹＭＮＥＴ：これも米国籍の商用ＴＳＳの発展形であり，すでにヨーロッパにも進出し，現在大きな新展をみせているといわれるが，ＴＹＭＮＥＴ社はＴＹＭＳＨＡＲＥ 社の子会社で１９６６年からＴＹＭＳＨＡＲＥ 社が行なってきたＴＳＳネットワーク・サービスを引継ぎ，データ・ベース・サービス，データ収集，情報検索，メッセージ伝送等のサービス・メニューを充実した。

ＴＹＭＮＥＴは現在米国内６１都市で営業を行なっており，１９７９年には１００都市 以上でのサービスを計画している。１９７５年時点で接続ホスト２６台，（ｘＤＳ ９４０ ２３台，ＰＤＰ－１０ ３台）ノード数８０といわれ，フランス・英国等への拡張も行なわれている。ＡＴ＆Ｔその他から賃借した２４００～９６００ＢＰＳの専用吹による蓄積交換機能を持つ。

このシステムはＮＣＣと称する４つのNetwork Supervisor station（特定ホストCalifornia (2), Paris，Newjergy）を持ち，これがユーザ・アクセスの審査，稼働統計，課金記録，障害検出等ネットワーク運用上の管理を行なっている。また，回線障害時のメッセージ伝送ルートの変更，決定の機能も持つ。４つのＮＣＣのうち常時１ケがアクティブで他が予備として異常時に切換える方式をとっている。

このネットワークの親会社にあたるＴＹＭＳＨＡＲＥ 社のシステムはすでに衛星回線を通し仏，英，ベルギー，スイス等ヨーロッパ諸国にサービスを提供していた実積があり，また，ＴＹＭＮＥＴとカナダのパケット交換網ＤＡＴＡＰＡＣ との相互接続も７６年に行なわれ

た。これら国際サービスの主力はＮＬＭ（National Library of Medicin）のMEDLARSなどを始めとする各種のデータ・ベース・サービスといわれている。

(4) ＴＥＬＥＮＥＴ：ＴＥＬＥＮＥＴ社はＡＲＰＡＮＥＴのパケット網の開発の主体となったＢＢＮの子会社であり，社長のLawrence G. Robertsは，元国防省ＡＲＰＡのDirectorであるとともに，ＡＲＰＡＮＥＴの開発責任者でもあった。この環境から容易に類推し得る様にＴＥＬＥＮＥＴはＡＲＰＡＮＥＴの技術を充分に活用した商用ｖＡＮシステムである。

１９７５年８月に営業を開始し，現在全米約５０以上の都市においてサービスを行なっており，ＴＥＬＥＮＥＴに接続されているホスト・コンピュータはすでに１００台に近く，端末は約1.０００台にのぼる。ユーザは各種企業，大学，研究機関等広範囲にわたり，それらのＴＳＳやコンピュータ・センタあるいはそれぞれのローカル・ネットワークが，ＴＥＬＥＮＥＴの交換網をそのまゝ利用しているケースが多い。その意味ではＴＥＬＥＮＥＴはコンピュータ・ネットワークというよりは商用パケット交換網というべきかも知れないが，例えば，現在1つの端末からＴＥＬＥＮＥＴに接続された各種ホストの４００以上のアプリケーション・プログラム，１００種以上のデータ・ベースに自由にアクセスでき，約５０種類ものプログラム言語を使用できるという状態になっており，立派にリソース・シェアリグ・ネットワークの様相を整えている。

また，ＴＥＬＥＮＥＴの特長について述べると次のとおりである。

① ＡＲＲＡＮＥＴの技術と経験を参考にし，目的に合ったかなりの改良を加えている。

ＩＭＰを各ホスト・サイトにおいたのは信頼性上問題があり，各都市にTelenet Central Office(地方局）をおき，そこにＩＭＰに相当するものを２重化しておき，ユーザは各自のホスト・コンピュータや端末をこれに接続する形にして信頼性をあげている。

② 新しい設備を構築せずに，ＡＴ＆Ｔ，ＷＵＩ等既存の通信業者から回線をリースし，それに付加機能を加えている。

５０ＢＰＳから５６ＫＢＰＳの巾広い速度に対応しうる。

③ 網えの接続方法を多様化

ユーザの既存システムにできるだけ手を入れずに接続し得る配慮を行なっている。（図１－１１参照）

④ ホスト接続のためのソフトウェア（ＮＣＰ）がＩＢＭ，Honeywell DEC，Amdahlのマシン用に原則としてＦＥＰ（Front End processor)内におさめられる形で提供される。

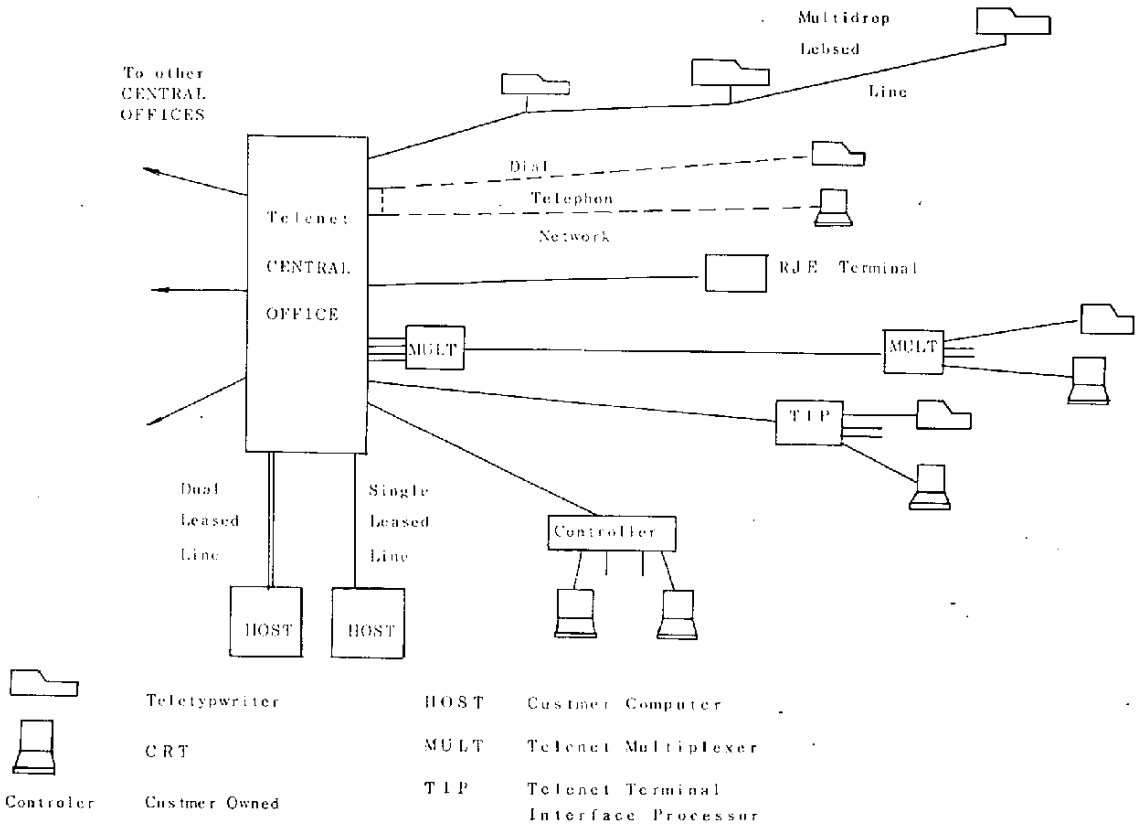


図1－11　Telenetへの接続

⑤　TELENETの網とホスト間のインターフェイス．プロトコルがCCITTによる国際標準X25の原形であった。

⑥　衛星通信の併用も計画している。

Satellite－IMPを設置し，1.544MBPSの高速通信が可能，国内のみでなくヨーロッパえも拡張する企画がある。

⑦　カナダ，メキシコ，フランス，イギリス等えの国際サービスが行なわれている。

(5)　CYCLADES：フランス政府の情報代表部，軍，PTT，研究機関，大学等の協力によつて開発されたフランス全土にわたるネットワークであり，データ・ベース・シェアリングを主目的とした広域実験研究ネットワークである。

1972年から75年にわたる開発費2100万フラン（126億円）の80％を政府の情報代表部が，20％を軍が負担したといわれる。また，PTTは電話回線およびモデムを無料提供し，政府直属の研究機関であるIRIAがこのプロジェクトの推進母体となった。図1－12にそのマップを示す。現在19のホストと，6つのノード，6つのTerminal Concentrator を持つ。19のホストのうちCIIのマシンが15台，

ノード・プロセッサはＣＩＩのＭＩＴＲＡ１５，サブネットはＣＩＧＡＬＥと呼ばれる専用のパケット交換網である。

このネットワークの主なねらいは

① 広域にわたる実験網

② プロトルコの研究とネットワーク間の相互接続。１９７４年にＮＰＬのネットワークやＥＳＡ（European Space Agency―Frascati）のネットワークと結合している。

③ 分散型データ・ベースの実現

このネットワークは１９７５年から稼働を初め13の大学，研究機関，７つの政府のデータ・プロセシング・センタが主なユーザである。

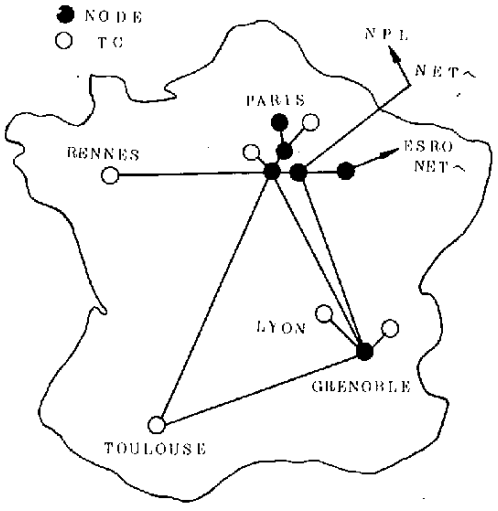


図1－12 ＣＹＣＬＡＤＥＳ

ＣＹＣＬＡＤＥＳ の成果はフランスＰＴＴの商用パケット網ＴＲＡＮＳＰＡＣ，ヨーロッパの国際ネットワークＥＩＮ，ＥＵＲＯＮＥＴ等に活かされているという。

(6) ＥＩＮ：European Informatics Network

１９７６年ＥＥＣは科学技術の研究調査グループを設立し，翌１９７０年に他のヨーロッパ諸国も含めてＣＯＳＴグループ（Cooperation Europeenne dans le domaine de la recherchc Scientifique et Technique）が結成された。このＣＯＳＴグループは数多くのプロジェクトを持ち，その１１番目がコンピュータ間通信の問題であったためＣＯＳＴ１１と呼ばれることもある。ＥＩＮの参加国はフランス，イタリー，ノルウェー，ポルトガル，スペイン，スイス，英国，ユーゴスラビア，オランダ，西ドイツ，ベルギー，スェーデンの１２カ国といわれ，現在図1－13に示すようにＬＯＮＤＯＮ（ＮＰＬ），Paris（ＩＲＩＡ），Zurich（The Federal Institute of Technologly），Ispra（Italy, EURATOM：Joint Research Center）Milano（Politecnica di Milano）の５個所にノードがあり，これにそれぞれ

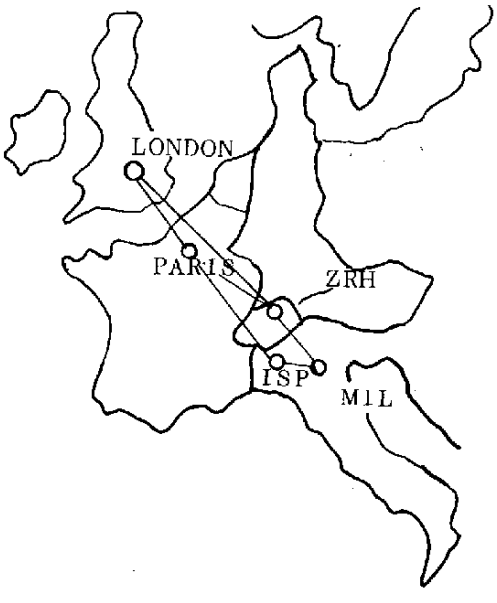


図1－13 ＥＩＮ

ＩＣＬ　ＫＤＦ９　　（ＮＰＬ）
ＩＲＩＳ　　８０　　（ＩＲＩＡ）
ＣＤＣ　６４００　　（Ｚｕｒｉｃｈ）
ＩＢＭ３７０／１６５　　（Ｉｓｐｒａ）
ＵＮＩＶＡＣ１１０８　　（Ｍｉｌａｎｏ）

等のホストが結合されている。

ＥＩＮの目的は主として参加国内の大学，研究機関を結ぶ国際実験網であり，

① この共同研究を通して各国のアイデアおよび情報交換を行なう。

② 各機関内のコンピュータ・センタ間で各種リソースの共用を行なう。

ことを期待している。具体的な開発はフランスのＳＥＳＡと英国のLogicaが行なった。Paris のＩＲＩＡでＣＹＣＬＡＤＥＳに接続しているが，ＣＹＣＬＡＤＥＳ はＥＩＮのローカル・ネットワークの位置付となる。

(7) ＥＵＲＯＮＥＴ：European On−line Information Network

ＥＣ諸国内の科学技術，経済，社会等のドキュメント情報の共用のための国際情報ネットワークで，１９７５年，ＥＣ加盟の９カ国，英国，西ドイツ，フランス，イタリー，オランダ，ベルギー，デンマーク，アイルランド，ルクセンブルグの各ＰＴＴ主管局によりインプリメントすることが決定された。

第１期計画を図１−１４に示す。

このネットワークはベーシック・レベルではＥＩＮのプロトコルを参考にし，また，インプリメントもＥＩＮ同様ＳＥＳＡおよびLogica が担当している。１９８０年の完成時には，約７００台の端末から，物理，化学，宇宙，ｴﾈﾙｷﾞｰ，原子核物理，治金，医学，農学，経済，統計，法律などの各種情報のデータ・ベースにアクセスしうるようになることが期待されている。現在まず約３０のインフォメーション・サービス機関の加盟が検討されており，データ・ベース数は約１００にのぼる。

なお，開発コストは１９７７年末までにＥＣが約３６０万ＵＳ＄（≒１０億）を出資し，更に，方法論やデータ・ベースの設置，諸技術の開発などに６００万ＵＳ＄(≒１８億)

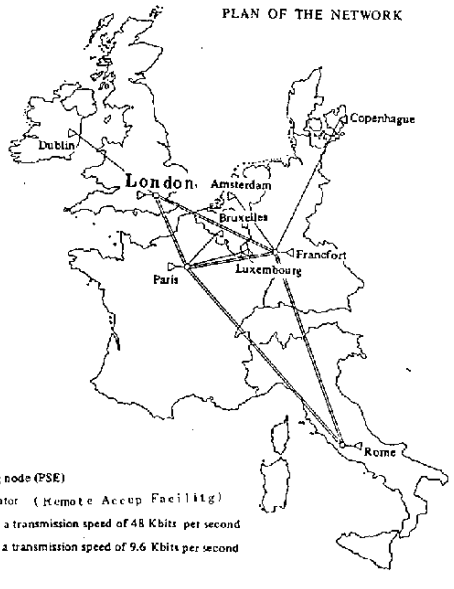


図1−14　EURONET

の資金を必要とするといわれる。

また，１９７８年以降の第２次計画には更に追加の投資が行なわれるという。

この主たる目的は

① 各国が独立にシステムを保有する重複投資をさける。

各国別の通信機構を作りあげるのにくらべ１／８から１／１０の経費ですむといわれる。

② この種のシステムは，より広域に且つより多数のユーザを対象とすることによりスケール・メリットが生ずる。

③ ＥＣ加盟国内で大型プロジェクトの協同開発の実積を作る。

④ このネットワークの建設，運用を通しＥＣ加盟国内の技術的交流の際の標準化を促進する。

ＥＵＲＯＮＥＴの利用に際し，現在下記のような各種の検討および作業が行なわれている。

① 農業，生医学，工業，環境，エネルギー，統計，数学等を初め３０の専門分野においてそれぞれのデータ・ベースの構築と利用に関する研究プロジェクトができている。

② 著作権，あるいはロイヤルティ，個人プライバシーに対する検討。

③ 各国の電信電話料金を考慮してのネットワークの使用料金の検討。

距離に無関係にネットワーク使用料金をきめる。

④ 各データの使用言語，各用語のデイレクトリ，専門用語の採用，あるいは資料テキストの自動翻訳の問題（英語が８０％といわれる）

⑤ ネットワーク上での標準化ソフトウェア

⑥ オペレータやユーザの教育

なお，ＥＵＲＯＮＥＴは将来他の公共ネットワーク，例えば，仏のＴＲＡＮＳＰＡＣＫ，英のＥＰＳＳ，独やデンマークの交換網との接続も考えており，また，既存の情報サービスや端末を最少の労力とコストで吸収しうることを目標としている。また，ＥＣ加盟国以外にも門戸を開いてゆく。

1.6.2 国内における例

我が国におけるコンピュータ・ネットワークは，米国には勿論，ヨーロッパ主要国に比べても遅れをとっているといわざるを得ない。とくに本格的なリソース・シェアリングを目指すものの実用化はまだ今後の問題である。

しかしながら最近に至り，企業内あるいは公共サービス関係のシステムとしてコンピュータ・ネットワーク的要素を持つものが幾つか出現し始めた。こゝではそれらの中から代表的と思われるものをいくつかとりあげる。

(1) 国鉄－旅行業者間ネットワーク

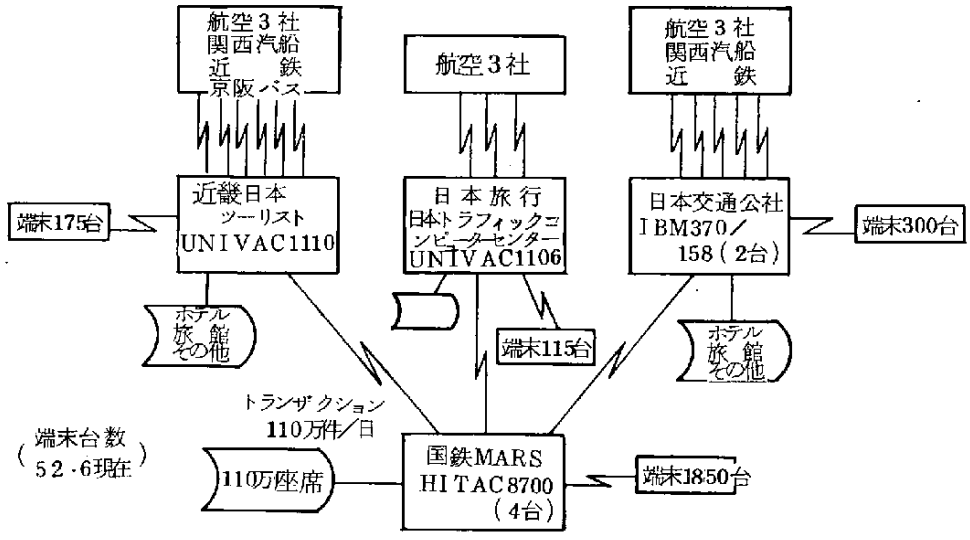


図1－15　旅行予約コンピュータ・ネットワーク

国鉄，日本交通公社，日本旅行，近畿日本ツーリストの4社間で，各種予約業務用の大型コンピュータ・システムを直結し，大規模なネットワーク・システムを昭和54年4月から稼働させる予定となっている。

図1－15に構成図を示す。

このシステムの目的は次のようなものである。

① 現在，国鉄の座席予約在庫110万座席のリアルタイム・ベースのサービスが一挙に拡大する。

② 即ち，現在，国鉄MARSによる「みどりの窓口」サービスの端末機は1.850台であるが，これが，各旅行業者システムの端末からも直接国鉄の座席予約が可能となり，トータルの端末機が3.400台となる。

③ 各旅行業者の端末は，国鉄の座席予約のみならず，航空券，ホテルの予約等の業務も同時に行なうことができる。

④ 各旅行業者の予約業務にも自動発券が可能となり，また，売上げ集計などのバック・グランド処理も，オンライン・ベースで相互に連動される。旅行業者におけるこれによる省力効果は大きい。

⑤ 各端末は漢字プリント処理を可能とする。

コンピュータ・ネットワークの立場からみると，このシステムは，ネットワーク間接続の一例と考えることができる。即ち，国鉄を始め4社はいずれも，すでに独自のネットワークを持っており，それぞれのネットワークどうしの接続によってその機能を倍増するケ

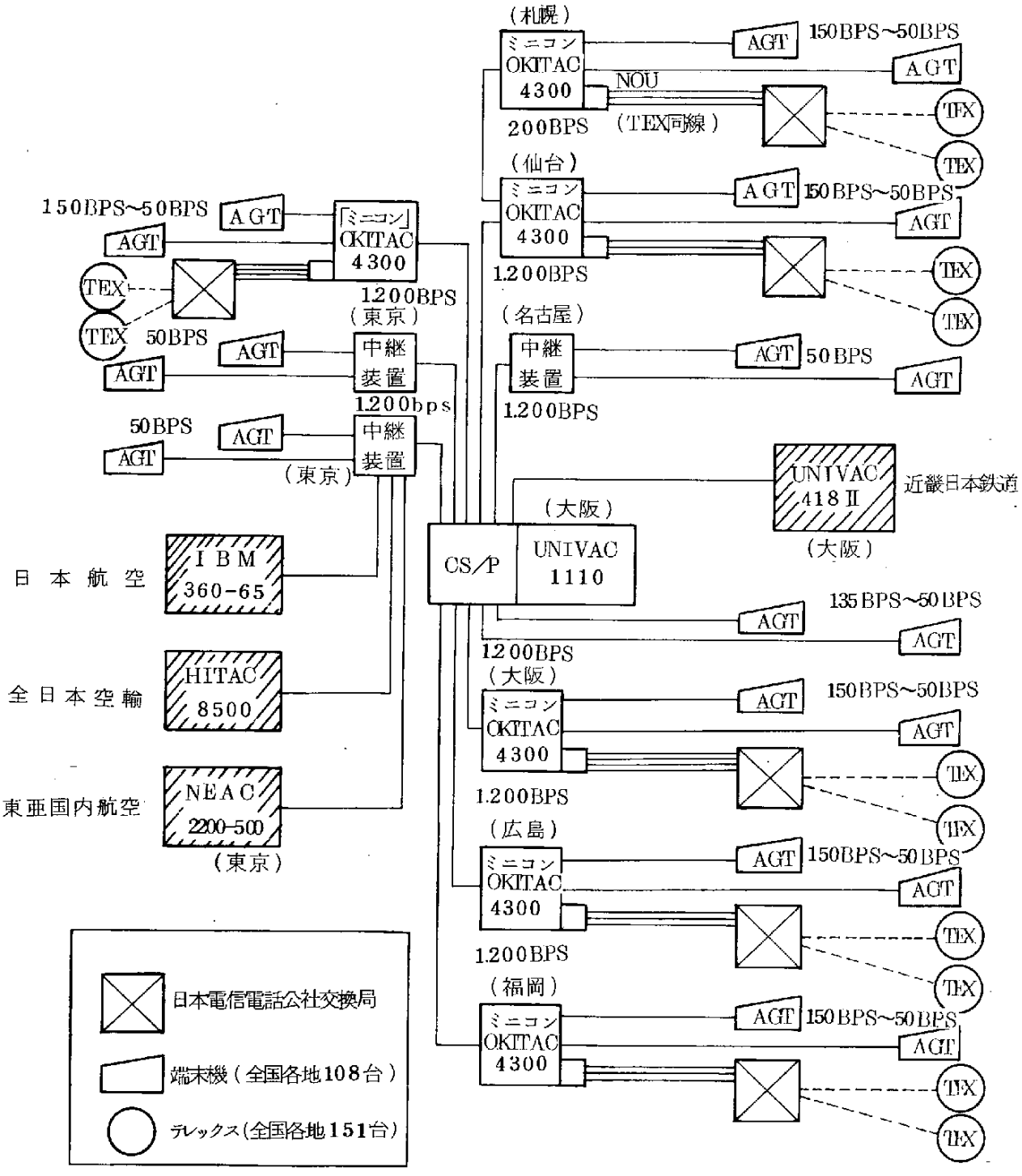


図1－16　近畿日本ツーリスト・オンライン・ネットワーク
（航空機座席予約システム・ネットワーク）

ースである。今後ともこのような発展形態がコンピュータ・ネットワークの１つの典型とならう。

また，リソース・シェアリングの観点からは，端末機の共用の１例といえる。なほ，図１－１６に既存の近畿日本ツーリストのネットワークの構成図を示す。このシステム自身も，すでに３航空会社，近鉄およびツーリストの本社のマシン(ＵＮＩＶＡＣ)との相互接続を行なっている。

(2) キャッシュ・ディスペンサ・システム（ＣＤＳ）

大都市の駅，デパートなどに設置されているＣＤ（現金自動支払機）を銀行間で共同利用するネットワークである。

昭和５０年１１月からサービスを行なっており，設計，建設およびセンタの保守は電電公社が行なっている。

図１－１７にその構成を示す。

システムの特長は以下のようなものである。

① 現在6大都市とその周辺に２２０台のＣＤが設置されており，参加５４銀行がこれを共用し，ＣＤ設置の二重投資を避けることが目的である。

② 各銀行の種々コンピュータが，サブ・センタを介してＣＤセンタに接続されており，これらは網側で新たに定めた統一インタフェースに従っている。

③ 東京，名古屋，京阪神に地区毎の監視センタがある。

④ 顧客がＣＤで引出し操作を行なうと，ＣＤセンタおよびサブ・センタを経由して各自の取引銀行のコンピュータに情報が送られ，顧客の預金元帳から支払額を差引きＣＤに返信する。これを受けたＣＤは現金を顧客に支払う。

コンピュータ・ネットワークの立場からみると，このシステムのリソース共用のタイプはＣＤ端末の共用である。

(3) 旭化成のＡＣＴシステム

旭化成工業（株）において全国の１００を越える事業所を結ぶ企業内，分散処理ネットワークで昭和４８年から一部稼働を開始している。

図１－１８にその構成を示す。

本システムの特長は以下のようなものである。

① 東京－大阪－福岡を結ぶサブネットはパケット交換を行なっている。

② サブネットの各ノード（ＡＣＴ－Ⅱ）に現在，ＩＢＭの３台の３７０ホストと，ミニホストとして集中型のメッセージ交換，メッセージ集配信ノード(ＡＣＴ－Ⅰ／ＡＣＴ－Ⅲ)を接続し，階層化を行なっている。即ち，分散型と集中型の両要素を持つ。

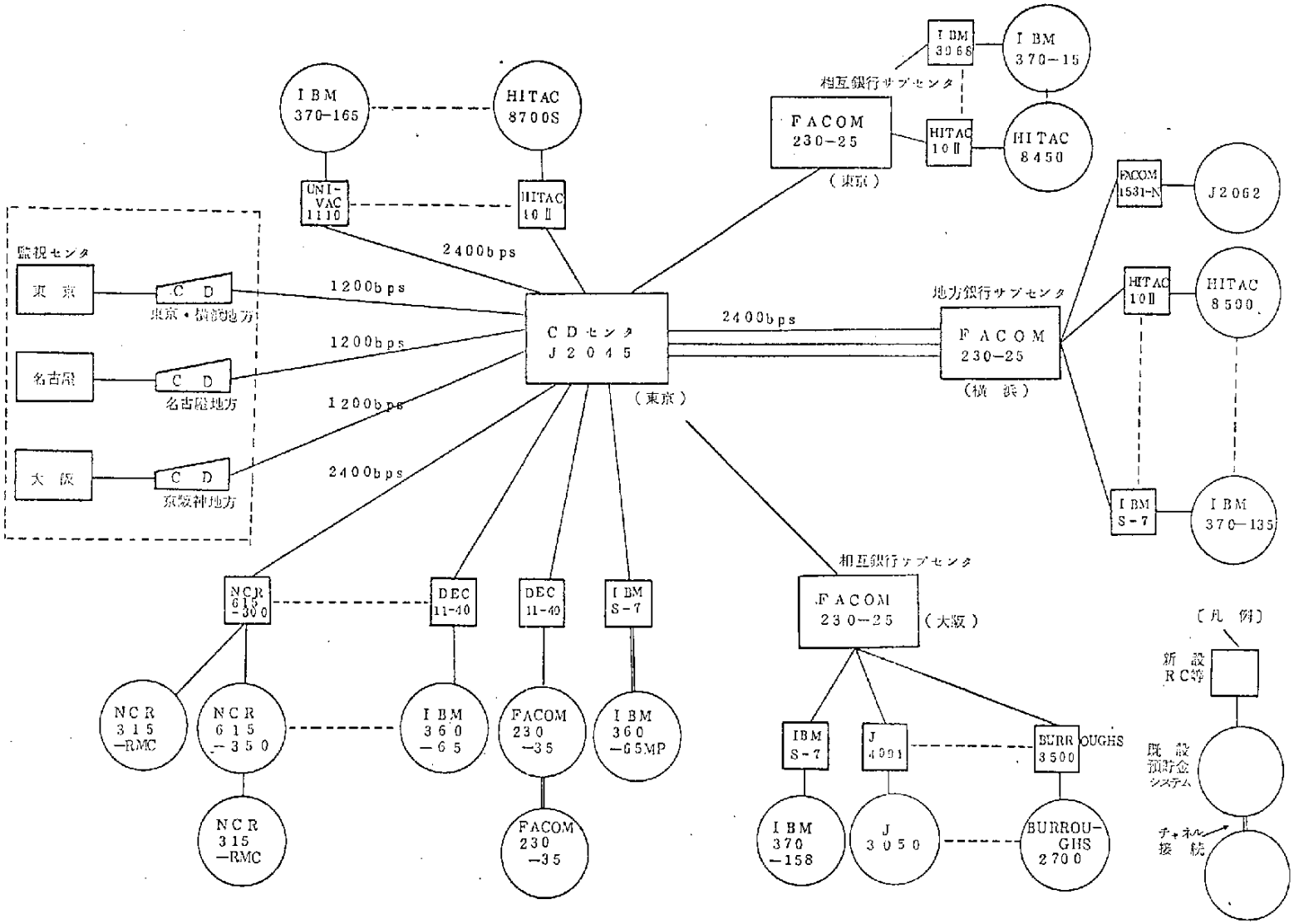


図1—17 CDシステム・ネットワーク

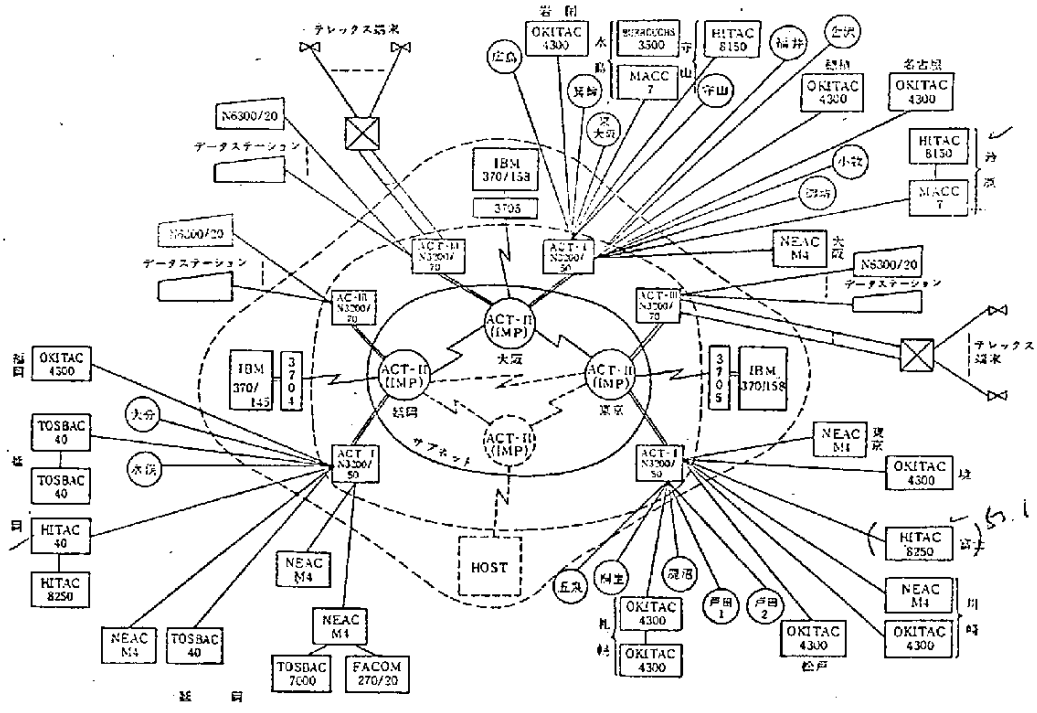


図1－18　ＡＣＴ　ネットワーク構成図

③　３台のＩＢＭホストは地域毎の製品別生産管理を水平型分散処理の形で行なうと同時に，これらのホストはまた，ローカルな端末からのリモート・バッチ処理および会話型処理にも対処している。

④　階層型のプロトコル体系が確立している。

本システムは企業内システムではあるが，ホスト・リソースのシェアリング要素を持ち，分散型システムであると同時に企業内コンピュータ・ネットワーク・システムの代表的なものの１つである。

(4)　石川島播磨重工（株）の分散処理システム

このシステムは当初から業務により使用コンピュータを使いわける分散処理を指向している。現在，図１－１９に示すように多彩なコンピュータが結合されている。

本システムの特長は次のようなものである。

①　当初から生産管理業務はＩＢＭ，設計・技術計算はＵＮＩＶＡＣ，経営事務処理はＡＣＯＳとそれぞれのマシンでの分業体制が確立していた。

②　これらのマシンが，ＩＢＭ　System 7，ＯＵＫ　９４００　ＴＯＳＢＡＣ　ＤＮ３４０を介して，それぞれ結合され，これらの小型機はＦＥＰ（Front End Processor）としての機能と通信制御機能を果たしており，とくにＯＵＫ９４００はネットワーク管理の機能をもつ。

③　造船部門の生産管理は従来，東京，横浜，名古屋，呉，相生のそれぞれの地区により，生産船種が異なっていたため，各地のＩＢＭ３７０により，それぞれの船種毎の生産管理を行なっていたが，船種の需要の変化により，他地区と共通の処理も発生し，相互のリソース共用の要求もでてきた。

⑥　近くＵＮＩＶＡＣ１１１０を１１００－８２にＩＢＭ３７０－１５８を３０３３にリプレースする予定もあるといわれるが，同時にＳＮＡ（ＩＢＭ）とＤＣＡ（ＵＮＩＶＡＣ）の両方を利用する可能性もあるという。

⑤　リソース共用の形としては，横浜，名古屋等の３７０－１１５から東京の３７０－１５８をリモート・バッチ方式で利用したり，ＩＢＭ機種の各端末からＵＮＩＶＡＣ１１１０にアクセスし科学技術計算を行なうケースもある。

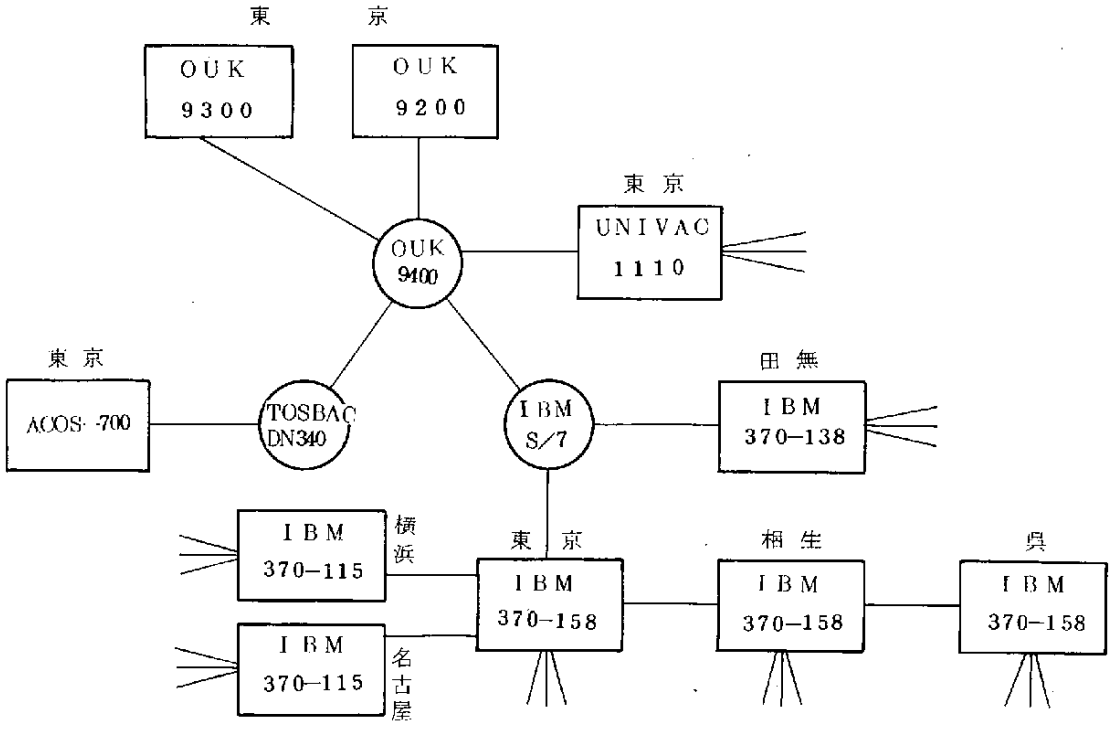


図１－１９　石川島播磨重工のシステム構成

(5) 大学間ネットワーク

リソース・シェアリングを目的とする数少ない典型的コンピュータ・ネットワークとしては先づ大学間のシステムがあげられる。

文部省の大型プロジェクトとして昭和４９年から開発を開始し，現在，東大・京大の両コンピュータ・センタ間でＴＥＬＮＥＴ（会話型処理プロトコル），ＲＪＥＰ等のユーザ・レベル・プロトコルを含む，第一バージョンのシステムが稼働中である。

今後，漸次全国の数大学間のネットワークに拡張されるものと考えられる。

なお，このシステムでは電電公社が５４年度からサービスを開始するデジタル・データ網（回線交換およびパケット交換）をデータ交換網として使用するところに大きな特長がある。

(6) ＪＩＰＮＥＴ

（財）日本情報処理開発協会が昭和４８年度から５１年度にかけ開発したリソース・シェアリング・ネットワークであり，これについては4.2に詳述する。

2. ネットワーク・プロトコル

2．1　プロトコルの概要

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。上記のような広義の定義を行なうと，従来使用されてきた用語であるインタフエース（例えば，2つ以上の構成要素の境界でそれらに共用される部分，2つ以上の装置を連結するハードウェア構成要素である場合，2つ以上のプログラムによって使用される記憶装置の一部またはレジスタの場合などがある。）との関係が明確ではない。

A．プロトコルの構造

上記の定義関係を明確にするために，図2－1に示すような構造のモデルが考えられる。

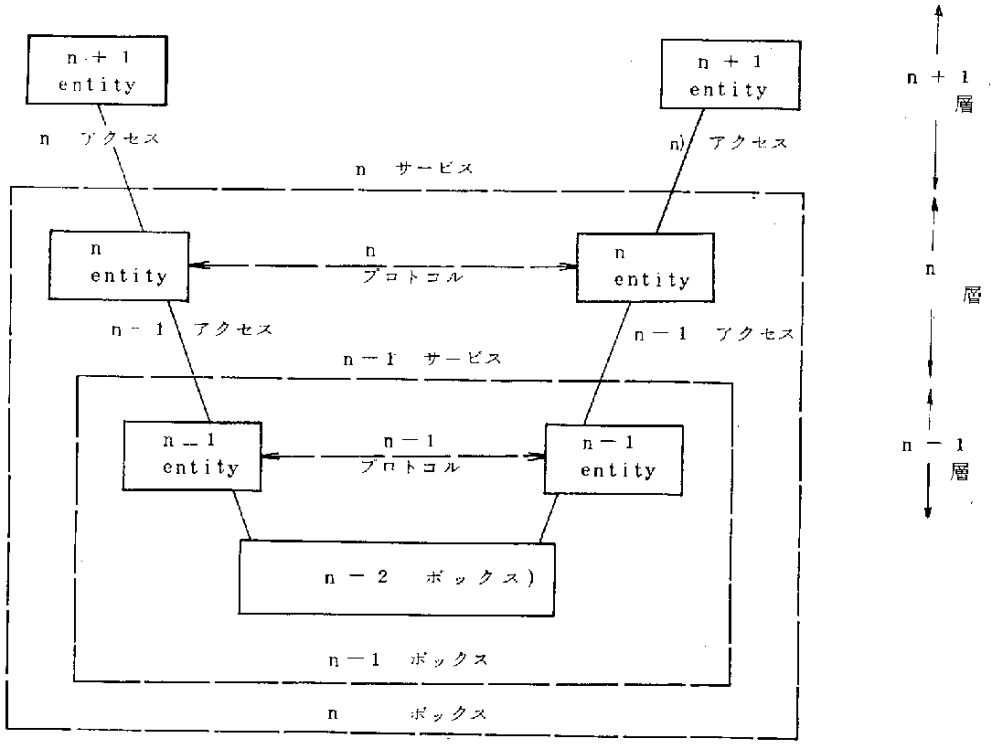


図2－1　プロトコルのモデル

次にこのモデルを簡単に説明する。

① 全体の処理は多層構造よりなるエンティティによって行なわれる。（層）

② 各エンティティは，他の系において，自分と同じレベルのエンティティと交信する。（プロトコル）

③ ある層は，それより1つ下の層が果す機能（サービス）を，定められた約束により（ア

クセス）利用する。

④　nエンティティはnの機能n－1サービス。n－1アクセスを利用して果たす。

⑤　nエンティティはnサービスをn＋1エンティティにあたえる。

⑥　同一層のエンティティ間（プロトコル）および直接隣り合うエンティティ間（アクセス）以外に，エンティティ間での交信を禁止する。

このようなモデルを利用することによってプロトコルとインタフェース（モデルではアクセスと称している）の違いが明らかになろう。

ただし，完全に独立な装置間でのインタフェースまで含めたプロトコルと呼ぶ場合がある。（ＡＲＰＡネットワークのHost-IMPプロトコル）

B．プロトコルの機能

プロトコル／エンティティを多層化する要因となったのは次の点である。

①　汎用性

②　拡張性

③　ハードウェアのコスト・パフォーマンスの向上

コンピュータ・ネットワークは通信と情報処理とが結びついた糸であり，それぞれ独立に，サービス層，機能層を構成している。

①　交換網

- データ・リンク制御
- スイッチング制御
- End—to—Endのエラーなどの制御
- その他

②　ホスト

- End—to—Endのエラーなどの制御
- プロセス間通信

  レターグラム・モード

  リエゾン・モード
- アプリケーションに依存した種々の機能——ファイル転送，ＴＳＳ共有など，これらの関係をＡＲＰＡＮＥＴ，ＤＮＡを例として図示したのが図２－２，２－３である。

コンピュータ・ネットワークにおける機能を一般的に示すと図2－4のようになる。

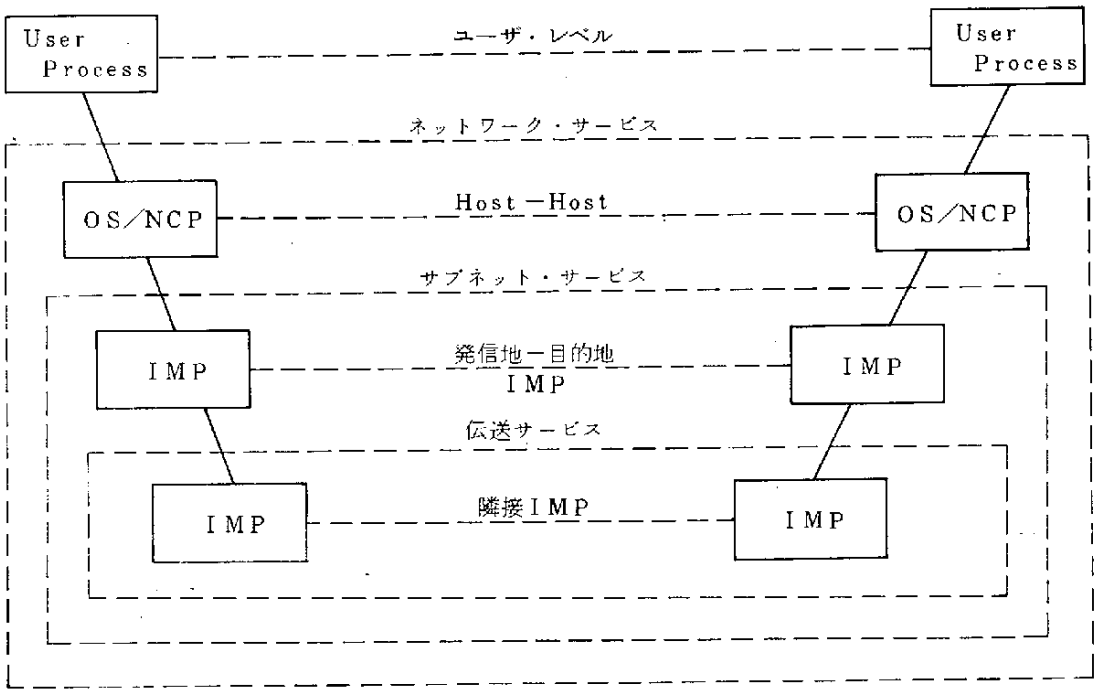


図２－２　ＡＲＰＡＮＥＴの階層

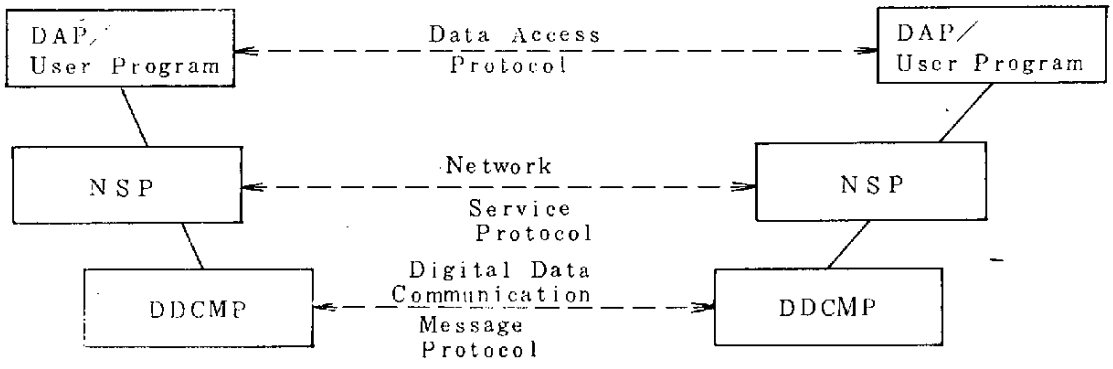


図２－３　ＤＥＣのＤＮＡの構造

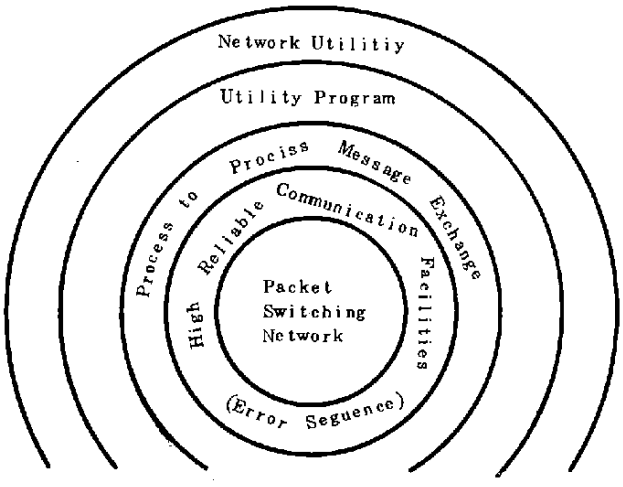


図２－４　機　能　分　割

C．プロトコルの物理的な位置づけ

前節においては，プロトコルの論理的な構造を述べた。これらのプロトコルがどのような形でハードウェア装置上にインプリメントされるかというのも重要な問題である。図２－５はＡＲＰＡＮＥＴを例とした図である。

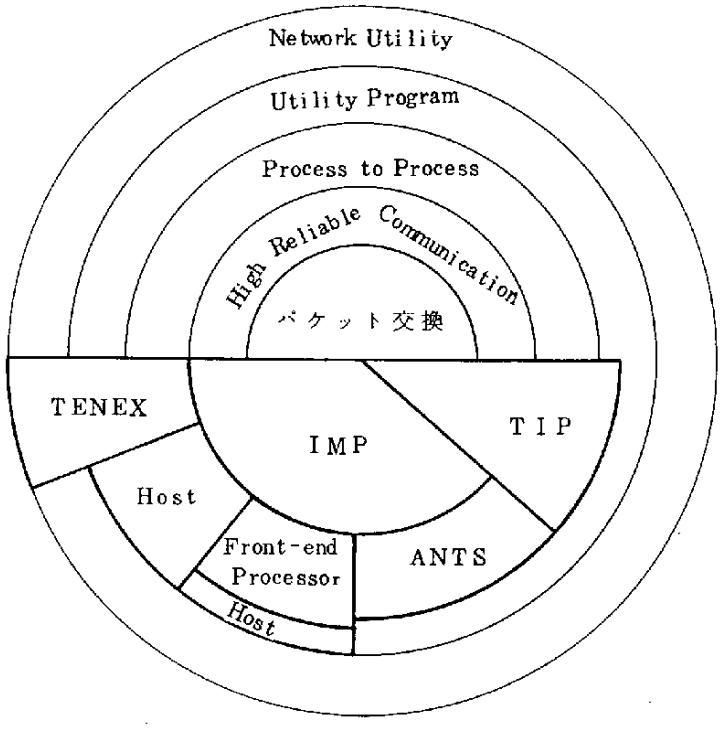


図2—5　プロトコルのハードウェア上での表現ＡＲＰＡＮＥＴを例として

①　ＩＭＰパケット交換，信頼性の高い通信の実現。

②　ＭＩＰ　ＩＭＰの機能，プロセス間通信，他ホストのＴＳＳの利用。

③　ＡＮＴＳプロセス間通信，他ホストＴＳＳ，ＦＴＰ，ＲＴＥを利用。

④　ホスト・プロセス間通信他ホストを高位プロトコルのサービスを共有。

⑤　Ｈｏｓｔ+ＦＦＰ　④のネットワークに関する処理の大部分をＦＥＰで行なう。

⑥　ＴＥＮＥＸ処理コンピュータを意識しないサービスを行なう。

2．2　各レベルのプロトコルの機能

2.2.1　ＨＯＳＴ—ＨＯＳＴ　プロトコル

Ｈｏｓｔ—Ｈｏｓｔ プロトコルは，他のホスト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，各ホストの中に存在するＮＣＰ（network control program）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。

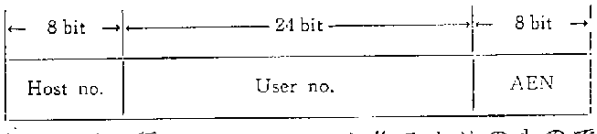
◎ 割り込み信号の授受。

以下にこれらの機能の実現方法について述べる。

Ａ．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，ホスト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。



AENはユーザが種々のport-idを作るためのものである。

図２－６　ＡＲＰＡＮＥＴのport-id

Ｂ．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，ホストにおいてメッセージの分割，再組み立てを行なうかどうかなどにより種々の形態がとられる。

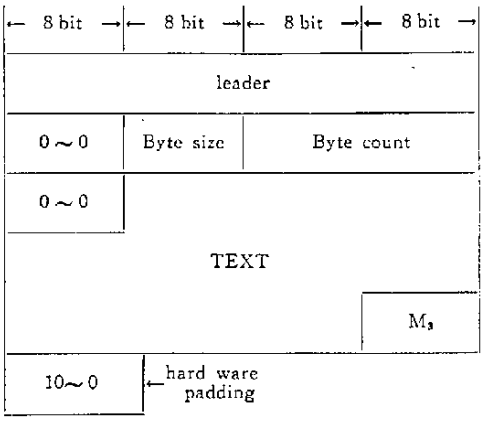
また，異機種のコンピュータを結合する場合には，各ホストの語長などを考え合せて，データ長，メッセージ・ヘッダーなどをうまく決める必要がある。

ＡＲＰＡＮＥＴ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 において行なわれているが，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ることができるとすれば，より高位のプロトコルに持ち込んでも十分なものであろう。

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，ARPANET，JIPNETのようにメッセージ・リーダで行なうものと，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。



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_3　　: 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2－7　ARPANETのHost-Host プロトコルのメッセージ形式

C．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，ARPANET，CYCLADES，NPL，JIPNETにおける概略の比較を表2－1に示す。

表2－1　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		割り込み信号	エラーの通知	メッセージ長	その他
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，パスの設定を行なわない方式もある	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

(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，特定のホストにメッ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。

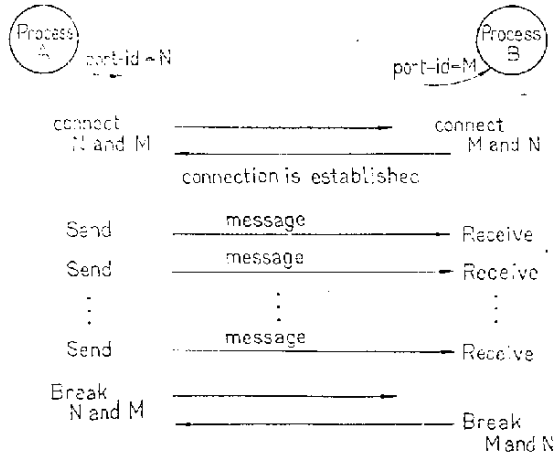


図2－8　バチャール・コール方式

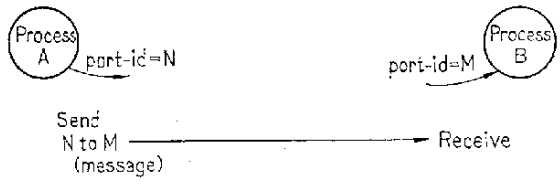


図2－9　データ・グラム方式

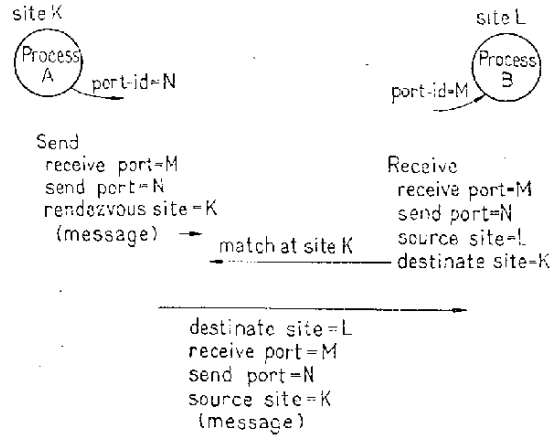


図2－10　Waldenの方式

(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

ii）受信側準備方式

② クレート方式

③ 受信命令待ち方式

④ Window 方式

⑤ Wait before transmit（WABT） 方式

(a) 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，その応答を持って，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，新らたに確保通知する方式がある。

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

(d) 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。

(C) 受信命令待ち方式は，プロセスが受信命令を発したのち，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。

(d) 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。

(e) 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。

(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，ＡＲＰＡＮＥＴ，ＣＹＣＬＡＤＥＳ，ＮＰＬ，ＪＩＰＮＥＴのいずれにもあるが，テスト機能などはＡＲＰＡＮＥＴにおいてのみである。

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レベルで行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を末端として行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。

ＡＲＰＡＮＥＴ，ＪＩＰＮＥＴにおいては，発信地ＩＭＰ－目的地ＩＭＰにおいて行なわれており，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。

(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ＡＲＰＡＮＥＴ，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，ホストの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。

2.2.2 高位プロトコル

高位プロトコルは，現在の最高位の層にあたるプロセスにおいて処理されるプロトコルである。これは，それぞれの機能ごとに利用者，システムのサービス側を規定しインプリメントする。本来のプロトコルの概念からいえば，高位プロトコル自体も多層化していてよいわけであるが，現在のところでは，はっきり多層化しているものは見あたらない。

高位プロトコルは利用者が勝手に作られるわけであるから，種類は無限にあるといってよい。しかし，重要なものは次にあげたもので，特に上の３つであろう。

① 会話形端末アクセス・プロトコル

② ファイル転送プロトコル

③　リモート・バッチ

④　グラフィック・プロトコル

⑤　メール・プロトコル

⑥　データ・ベース共有プロトコル

A．会話形端末アクセス・プロトコル（ＩＴＰ）

このプロトコルは，遠隔端末からＴＳＳ，オンライン・システムなどをアクセスするためのものであり，ARPANETの実積，Euronetの利用者予測を見れば明らかなように，遠隔端末からの資源の共有がコンピュータ・ネットワークにおける主たる手段である。この意味においてＩＴＰは最も重要なプロトコルであり，各コンピュータ・ネットワークにおいて最も整った形をしている。

ＩＴＰは２つの基本機能より構成される。

まず第１は，初期会話および終了会話手順にあたるもので，サービスを要求する側（ユーザ）とサービスをする側（サーバ）との間でのサービス開始の確認を行なう手順，終了をする手順を決めたものである。

第２番目は，仮想端末（Network Virtual Terminal：NVT）といわれるものでネットワークにおける仮想的な標準の端末機能を定義したものである。

ＩＴＰはこの２つの機能を利用することによって，仮想端末から特定のサービスを受ける，あるいは仮想端末間での交信を可能にしている。このプロトコルは端末レベルの標準化を行なっており，ＴＳＳなどの利用に当たってのコマンド体系の標準化をめざしたものではない。このために，利用者はサービスを受けたいすべてのホストの機能についての知識が必要である。

(1)　初期会話，終了会話手順

これはInitial Connection Protocol（ICP）とも呼ばれているものである。

一般的にいって終了会話手順は，使用したデータ・リンクの切断順序がサーバとユーザの間で異なっていると，デット・ロックが発生する可能性があるので切断順序を決めるだけで十分であろう。

初期会話手順では，リンクの設定順序ばかりでなく，どのようにしてユーザの要求するサービスにつながるかという要素を持っているために，大きくわけて次の２とおりの方法のいずれかがとられている。

第１番目は，それぞれのサービスに対して特別なポートを割り当てておき，ユーザからサーバのポートに対してコネクションの確立要求を行なう。サーバはこれによって相手側の要求を知ってコネクション確立の応答をし，サービスを開始する。これはＪＩＰＮＥＴで利用しているものであり，ARPANETのそれも手順はかなり複雑になっているが基本的

には同様の方式である。

第2番目は，ユーザ側からのサービス要求を受け付けるポートを1個だけあるいはサービス・グループごとに1個ずつ用意しておく。ユーザはこのポートに対してコネクションを確立するなどして，サービスの要求を行ない，その後，受けたいサービスの内容についての指示を行なう。サーバ側は，相手の要求するサービス内容を知った後に，それのサービスが可能なプロセスとのコネクションを確立させるものである。これはCYCLADES，NPLで利用されている。

この2つの方法を検討する上で考慮しなければならないのは，加入者が使用できるポートの数，プロトコルの種類，1つのプロトコルでサービスする種類（例えば，ITPでTSS，リモート・バッチの2つのサービス），INWGのEnd-to-Endプロトコルにおけるようなレター・グラム・サービスの有無などである。

第1番目の方法は非常に単純であるのでプロトコルの種類が少ないときには有利である。しかしながらプロトコルの種類が多い場合あるいは1つのプロトコルで多くの種類をサービスする場合などになると第2番目の方法が有利になってくる。

(2) 仮想端末

オンラインあるいはTSSにおいて，各ホスト・コンピュータでは手順の異なったクラスの端末について，それぞれの手順ごとにターミナル制御プログラムを用意している。ローカルなTSSにおいてはこの手順の種類は数種であるが，コンピュータ・ネットワーク全体で考えると非常に多くの種類の端末になるので，すべての端末の制御をホストで行なうのは不可能になってくる。このような背景で，サーバとユーザの間でネットワーク標準端末という既念が導入された。これが仮想端末であり，図2－11のようにサーバはユーザを仮想端末とみなして制御し，実際の端末への対応づけはユーザにおいて行なわれる。

仮想端末は，入力用としてキーボード，出力用としてプリンタ機能を持つタイプライタ的な装置を想定している。これを設定するときに考慮するのは次の点である。

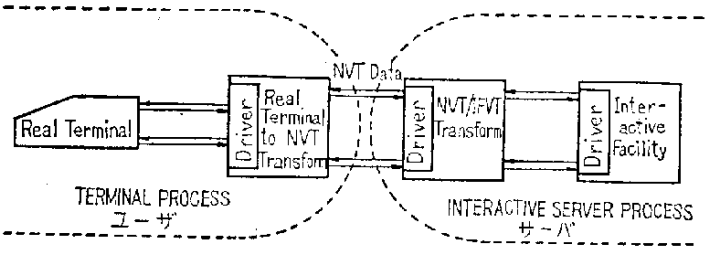


図2—11 サーバとユーザITPの関係

(a) 基本機能と拡張機能

仮想端末はタイプライタの如きものが想定されているが，実際の端末は，タイプライタ，キャラクタ・ディスプレイ，グラフィック・ディスプレイなど非常に広い範囲にわたっており一意的に機能設定するのが困難である。このために，最小機能としてすべてのホストにおいてサービスすることを原則とする基本機能と，より良いサービスをするために，ユーザとサーバ間での合意ができれば一部の機能を追加，変更できるオプションより構成さ

れる。

オプションの選択に関しては，サーバとユーザ間での会話が必要になってくる。この会話方式としては，初期設定手順が終了した段階でのみ行なうものと，任意の時点で必要な会話を行なうものがある。後者の場合には，非同期的に行なわれるので，要求と拒否，承諾のコマンドの機能設定に対する考慮，同じコマンドを2度以上だしてはいけないなどの制限をつけないとデッド・ロック，要求のループにはいる可能性がある。

(b) コード系

仮想端末としては，国際的な標準コードに準拠するコードを使用することによって標準化している。これはIA5，USASCII，JISなどのコードを利用することを意味している。

このような標準コードがあるとはいっても，次のような点についての問題点が残されている。

まず第1番目は，CR（復帰），LF（改行），NL（復帰改行）の問題である。即ちNL機能しか持たない端末に対して，CR，LFを独立した機能としてサービスするように要求するかどうかということである。

一般的な利用方法の場合には，CR，LFは全く独立に動作させることはあまりないが，パスワードを黒くぬりつぶしたり，FORTRANの1H＋の機能をサービスするときに利用される。

この問題に関して，ARPANETにおいては，NL動作を行なわせる場合にはCR，LFの順で送り，それぞれを独立に使用するときには，CRの後にNULLコードを入れることを要求している。しかしながら，これを実際にしようとするとサーバ側に非常に大きな負担がかかることがあるために，あまり良い解決方法とはいえない。

JIPNETのように，NL端末のときには，原則としてLFをNLとみなすようにして，それ以上に関してはプロトコルでは決めていない例もある。

(c) 割り込み信号と同期の問題

無限ループに入ってしまったプログラムの強制終了，大量出力を始めたプログラムの終了などのために，端末装置から使用されるのが割り込み信号である。

割り込み信号は，緊急性を要するものであるから，入力されてはいるがプロセスに渡されていないデータを追い越すことがある。ほとんどの場合にプロセスは割り込み信号の受信によって入力命令を出すので，このようなときには，割り込み信号を入力する前に送られたデータは捨てられないと，処理結果がおかしくなってしまう。

このような同期の問題に対しては，次のような対策がとられる。

第１番目は，Host-Host プロトコルで，サービスされる割り込み信号の送信において，先行するデータを捨てるものである。

第２番目は割り込み信号を送ると同時に，データが流れるリンクに対して同期をとるための文字を挿入する方法である。

第３番目は，同期の問題は端末に利用者がいるとすれば，多くの場合に対応した処置がとれるので，利用者の端末操作手順で解決してもらう方法である。

同期をとる問題に関しては，第２番目の方法が最良であると考えられる。

B．ファイル転送プロトコル

ファイル転送プロトコル（ＦＴＰ）は，大容量データを効率よく正確に送信することを目的としたものであり，動作モデルは基本的には図２－１２の形態となる。

ＦＴＰの基本的な機能としては次のものがあげられる。

① 初期会話，終了会話手順

② ネットワーク仮想ファイル（ＮＶＦ）

③ データ転送方式

④ ＦＴＰのコマンド

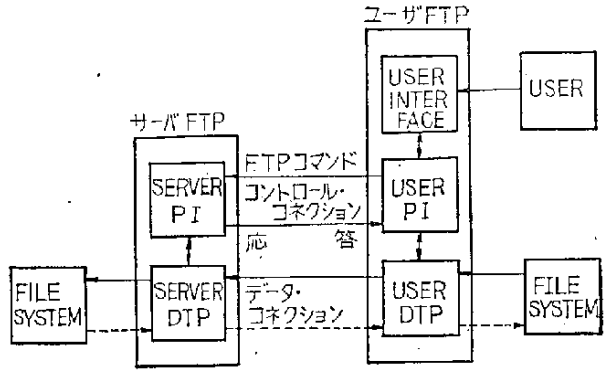


図２－１２　ＦＴＰの動作モデル

ＦＴＰとＩＴＰが非常に異なる点は，ＩＴＰが通信路の両端におけるコード系，制御手順だけを共通化したのに対して，ＦＴＰはユーザとサーバ間でのコマンドの共通化をはかっていることである。コマンドの共通化は，必然的に両ホストにおけるファイル管理などの違いを利用者に意識させないようにすることになる。

上記のうち①に関してはＩＴＰにおいてすでに述べたので②～④についてのみ述べる。

(1) ネットワーク仮想ファイル

ＮＶＦは，ＦＴＰにおいて取り扱うファイルをＩＴＰにおけるＮＶＴと同じように標準化しようとするものである。

いままでのところでは，ＮＶＦの設定には成功していないといってよい。この理由としては，ファイルの管理，アクセスがそれぞれのホストで異なっていることがあげられる。ホストにおいては，順編成，直接編成などのファイルのアクセスをサービスしているが，ＦＴＰとしては比較的標準化しやすい順編成ファイルだけを扱っている。

転送されるファイルの内容としては，プログラム，データ，ジョブ出力結果などがありホストごとにそれぞれの表現が異なっている。ＦＴＰにおいては，基本的にはファイルの内容をそのまま他のホストにコピーすることを前提としている。もちろん，ＥＢＣＤＩＣと

ＩＡ５間のコード変換などはサービスしている例も多いが，固定小数点および浮動小数点数の表現形式を相手に知らせて変換をするというサービスまでをオプションとして含むのははＥＩＮのＦＴＰだけである。

(2) データ転送方式

ファイルの送信側と受信側の間にあって，実際データがどのような形で転送されるかを規定するものである。データの転送に関しては原則としてトランスペアレンシーを保証しながら効率よく転送することである。データ転送方式において利用されているのは次の方式である。

① ストリーム転送

② レコード化転送

③ コンプレス転送

(a) ストリーム転送は，ファイルの最初から最後までをデータ・ストリームとして扱って転送するものであり，送られた側でのファイルの記憶形態は，そのホストにおける標準的なファイルのそれと一致することになる。

(b) レコード化転送は，一定のレコード長あるいはブロック長を持ったファイルを送るときに使用されるもので，この方式で送られた場合には，新たに作られたファイルは，もとのファイルと論理的には同じである。

(c) コンプレス転送は，プリンタ出力，カード入力などのようにブランクなどの特別なコードが多いデータを圧縮して効率よく転送できるようにするものである。

その他の機能として，受信側のファイルがラインプリンタなどの場合には，紙が切れるなどの理由によって再送を要求しなければならないことがある。この機能をサービスするために，リスタート・マーカーを転送データに挿入することが行なわれる。

(3) ＦＴＰのコマンド

ＦＴＰのコマンドは，ファイルの指定，転送方式，ファイルの送受信のいずれであるかなどを指示する機能を持たせている。

ファイル転送以外にも，ある利用者のファイル名のリスト，ファイルのダンプ，ファイル名の変更，ファイルの消去などのサービスをするコマンドも持つ。

ＦＴＰコマンドは，ファイル転送においてネットワーク標準という形で統一されているが，ファイル名やファイルの存在場所を示すようなコマンドにおいては，パラメータとしてローカル・ホスト特有の表現を許さなければならない場合がでてくる。このようなローカル・ホスト特有な表現を標準化するのが今後の大きな課題であり，この結果がＦＴＰの有効性を決めるカギとなる。

2．3　プロトコルの国際標準化動向

情報処理技術や部品技術の進展と，ビジネスの高度化，多様化と相挨って，データ通信システムは，大規模化，オンライン化，広域化の傾向にある。これらの動向に対応すべく，データ通信システムは，よりネットワーク的な様相を帯びてきており，ネットワーク・アーキテクチャの研究や，トータル・システムとしてのプロトコルの体系的な整理が活発に行なわれている。コンピュータ・メーカもＩＢＭがＳＮＡ，ＵＮＩＶＡＣがＤＣＡ等が，各々の特長を持ったネットワーク・アーキテクチャを相次いで発表している。

一方，データ通信用の通信回線サイドの動きも目覚しいものがある。従来広く利用されているアナログ専用線，公衆交換網（電話網，加入電話網）に加えて，データ通信向きの新しい通信向きの新しい通信メディアとして，ディジタル専用線，公衆データ交換網の開発，導入が具体化してきている。これらの動向に対応し，ＩＳＯ，ＣＣＩＴＴおよびＩＮＷＧ等の国際機関において，国際的な仕様の統一や標準化が進展している。

ＩＳＯでは，主としてユーザ・システムの面からの各種プロトコルの検討が行なわれ，一方ＣＣＩＴＴでは，主として通信メディアの面から公衆データ通信網の検討が行なわれている。また，ＩＮＷＧは，パケット交換網からハイレベル，プロトコルにいたるまでの広い範囲にわたって標準化案などの作成を行なっており，ＩＢＭをはじめとしたコンピュータ・メーカも各種のプロトコルの標準化を進めている。

ここでは，ＩＳＯ，ＣＣＩＴＴ，INWG，IBM，DDXの５者を取り上げ基本プロトコルの現在の技術動向に関し，比較，評価を試みる。

2.3.1　ＩＳＯ（International Standards Organization）

ＩＳＯでは，リンク・レベル・プロトコルを３つに分けて標準化を進めている。

(1)　フレーム構成は，ＩＳ３３０９ハイレベル・データ・リンク制御手順として承認され資料化されている。今後，手順クラスにおいて，バランス型が導入されると，これに伴う若干の変更が生ずると考えられる。

(2)　手順要素

手順要素は，ＤＩＳ４３３５で定義されている。これは，ＨＤＬＣ（ハイレベル・データ・リンク制御手順）で用いられる動作モード，コマンドおよびレスポンスの様式と機能が定義してある。データ・リンクの設定には数種の「モード設定コマンド」が定められ，このコマンドを受信したステーションが確認応答を返送することによって，データ・リンクが確立される。情報メッセージは，１コマンドまたは１レスポンスによって送信される。情報メッセージは，コマンド／レスポンスと共に送受信される送信番号，受信番号によって確認が行なわれ，確認がとれない場合には，所定の手順によって再送などの回復手順が

実行される。手順要素については，次のような審議状況にある。

(a) DRAFT IS4335は，N1251の修正を含め，TC97による郵便投票にかけられている。

(b) 追加コマンド／レスポンスについては，NI300がまとまり，SC6にて，76年10月1日を期限として郵便投票にかけられていた。

(c) 手順クラスのバランス形導入に伴ないDIS4335にかなりの変更が生ずるはずであり，NI301としてまとめられている。

(3) 手順クラス

上記2つの標準を用いて実際のデータ通信システムを形成する際の標準実施方法をクラスに分けて定めるものである。不平衡型手順クラスと平衡型手順クラスの2つに分けられている。手順クラスの進捗状況をまとめると次のとおりである。

(a) 76年5月のストックホルム会議の結果，今後のベース・ドキュメントとして，NI299(PROPOSED CLASSES OF PROCEDURE)NI301がまとめている。

(b) 今後まだまだ結論が出るには，時間がかかるものと思われる。77年3月のシドニー会議までには，はっきりしてくるものと期待している。

(c) 当初クラスとして，不平衡型，平衡型，対称型の3種が考えられていたが，対称型は，不平衡型の一変型と考えて，現在では，2つのクラスが設定されている。

ISO／TC97／SC6では，以上述べたリンク・レベル・プロトコルの標準化に加え，ネットワーク・レベル等，高位のプロトコルの標準化も取組みはじめた。しかし，現時点では，関係国から約15通の文書が提出され，研究すべき事項のリスト・アップが行なわれた段階で，国際標準案の作成はこれからという段階である。ISOが目的とする高位プロトコルの範囲と，公衆パケット交換網用のCCITT勧告，X25のパケット・レベル・プロトコルとの関係についても，現在のところ明確でない。現在提出されている検討報告書からその動向を推定すると次のとおりである。

(1) ネットワーク・レベルの機能

ネットワーク・レベルが遂行すべき機能について，「発信元アドレス」，「受信先アドレス」，「シーケンス番号」など，どの提案にも共通的に含まれる項目のほか，メッセージの緊急度の表示，メッセージのセキュリティに関する表示等の提案がでている。提案の中には，ネットワーク・レベルの機能とそれより高位のレベルの機能を完全に独立させたいとするものもあり，一部に重複させているものもある。

(2) 通信用ヘッダの形式

何らかの形で通信ヘッダ・アイデンティフィケーションが必要という点で意見の一致がみられる。また，固定長のいくつかのサブヘッダの次に，更に，いくつかの可変長ヘッ

ダを置く形式も提案されている。要するに，適用範囲について融通性のある形式とするか，固定されたヘッディングとするかという点に問題がありそうである。

2.3.2 ＣＣＩＴＴ（Consultative Committee on International Telegraph and Telephone ）の動向

昨年のＣＣＩＴＴ第６回総会において，公衆パケット交換網用の最も重要な勧告であるＸ２５が採択された。この勧告Ｘ２５は，ユーザのメッセージとパケット間の変換を行なうパケット分解・組立機能・（ＰＡＤ）を内蔵した端末（PACKET MODE TERMINAL）の網とのインターフェース（ＤＣＥ／ＤＴＥインタフェース）を規定したもので，内容としては，次の３つのレベルについて基本仕様を定めたものである。

① フィジカル・レベル

② リンク・レベル

③ パケット・レベル

(1) フィジカル・レベルは，物理，電気条件を定めたものである。パケット端末は，２４Ｋｂ以上の高速端末を対象とすることから，同期形のインターフェースとしており，専用線サービスや，回線交換サービス用とコンパティビィリィティのあるＸ２１およびＸ２１ ｂｉｓを適用することとしている。

(2) リンク・レベルは，パケット端末と網との間で情報を送受するためのデータ・リンク上の伝送制御用プロトコルを決めたもので，ＣＣＩＴＴでは，ＬＡＰ（LINK ACCESS PROTOCOL)と呼んでいる。Ｘ２５のＬＡＰはＩＳＯのＨＤＬＣと細部については，相違したものである。例えば，手順クラスは，Ｘ２５は，対称型としているが，ＩＳＯには，このクラスがなくなっている。しかしながら基本部分，即ち，フレーム構成，要素，手順は，ほとんどはＩＳＯ標準の一部とみなされている。

(3) パケット・レベルは，パケット交換網を経由して，ユーザ端末間で通信する場合のプロトコルを示すもので，次の２つがある。

(a) バーチャル・コール

(b) パーマネント・バーチャル・サーキット

(a)バーチャル・コールは，現在一般的に利用されている回線交換原理に基く網での制御方式の考え方の延長上にあるものであり，電話網等と同様に，呼の概念を導入し，発端亦一着端末間に論理的なリンク（バーチャル・リンク）を設定して通信を行なう方式である。バーチャル・コール方式の場合には，通信の開始に先立って，論理リンクの設定があり，このための呼制御用パケットが往復し，通信可能な状態を確認した後，データ・パケットが相互に転送される。通信終了後，論理リンクの解放のための呼制御用パケットが往復す

ることとなる。呼制御の段階で，網内異常，相手話中，相手空番等，論理リンクの設定ができない場合は，その旨発信側へ網より通知される。パケット網内に論理リンクを設定する場合，網の両端のノード間のみでリンクを意識し，中間のノードでは関与しない方式と中間のノードでも意識して管理する方式とが考えられる。前者は，制御が容易になると共に，網内のリソースの有効利用が計れるが，同一呼に属する各データ・パケットの網内転送ルートが異る場合もでてくる。後者の場合は逆になり，制御が複雑になるとともに，リソースが増えるが，網内でのパケットの転送の信頼性が高くなる。このどちらをとるかは各国のインプリメンテーションに委ねられている。

通信状態に入ってからは，パケット順序制御とパケット流量制御の機能がある。順序制御は，加入者線上でのパケット多重リンクでのリンク単位の順序を保証するとともに，パケット交換網は蓄積交換網であり，網内のルーテングによっては，網の内部で順序が乱れる場合があるために，データ・パケットにインターフェースの上下両方向で順序番号を付与することとしている。この番号情報は，パケット・リンク・レベルでの情報送受の確認も兼ねており，インターフェース上の番号を網内でも保存すれば，エンドとエンドでのパケット送達確認機能としても利用できる。

一方，流量制御は，端末と網との間のトラヒック量を制御する手順である。これは基本的には，発端末から網に流入するトラヒック量と網から着端末へ流出するトラヒック量との間の整合をとる機能であり，端末速度の差異やトラヒック量の差異を吸収するとともに，ユーザから見て可能な限り実時間に近いパケット転送ができるようにしたものである。他方，網から端末に情報を送る場合に，端末側の都合に合わせてトラヒック量を制御し，端末をガードしている。流量制御はあくまで論理的なパケット・リンク単位に，いわばCALL BYUCALLに行なうトラヒック制御であるが，このメカニズムとしては，各種の方式が考えられ，ウインドウ方式が最も一般的である。Ｘ２５でも採用されており，これは通信に先立って予め連続受信可能なパケット数（ウインドウ・サイズ）を決め，そのサイズ以内に収まるように，上述の順序制御用の確認番号により制御するものである。

(b) パーマネント・バーチャル・サーキットは，任意の端末間の通信を前提にしているが，データ通信サービスでは，専用線的な利用も多いことから，端末間に常時論理リンクを設定しておき，バーチャル・コールの呼設定手順を省略して通信できるものである。即ち，通信段階では，バーチャル・コールと基本的に同一となり，従量制の専用線サービスと見ることができる。一般的には，バーチャル・コール方式に対し，データ・グラム方式と呼ばれるものがある。これは，呼という概念がなく，従って論理的なリンクもない。即ち，各パケットにアドレスを付与し，端末は単に，網に投入すればよい。換言すれば，呼制御段階と通信段階の区別がない。また，バーチャル・コールでの呼制御段階でのオーバ・ヘッ

ドを少くする方式として，呼制御パケットにデータも相乗りさせる方式（ファストセレクト方式）も検討が進められている。

A．一般端末の収容

７６年１１月に開かれたスペシャルラポータ会議で，ＰＡＤ機能を有していない端末（ＮＯＮ　ＰＡＣＫＥＴ　ＭＯＤＥ　ＴＥＲＭＩＮＡＬ）をパケット交換網に接続する方式が論議された。基本的には，パケット交換網内にＰＡＤ機能を持たせ，一般端末を電話網を介して，ＰＡＤに接続するかＰＡＤへ直接接続する。また，パケット端末とＰＡＤも接続する２つに分離して標準化する方向となっている。（図２－１３参照）

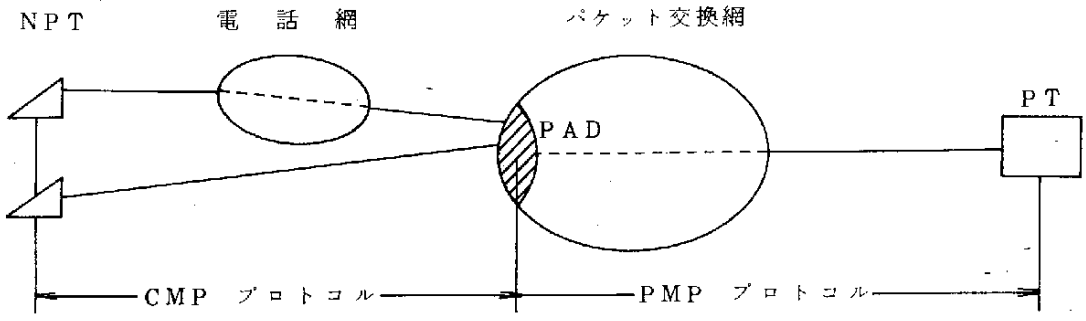


図2—13　NPTの収容とプロトコル

- 端末（ＮＰＴ）—ＰＡＤプロトコル

  Ｘ．２Ｘ

  ＣＭＰ　：　ＣＨＡＲＡＣＴＥＲ　ＭＯＤＥ　ＴＯ　ＰＡＤ
- 端末（ＰＴ）—ＰＡＤプロトコル

  Ｘ．２ｙ

  ＰＭＰ　：　ＰＡＣＫＥＴ　ＭＯＤＥ　ＰＲＯＴＯＣＯＬ

ＣＭＰプロトコルは，多様な属性を持つ端末に対して，どのように標準化をはかるかが大きな問題である。端末インターフェースとしてＸ２５のようなユニークなものを標準として決めることは，大半の端末（特に既存機種）を網から締出すこととなるので現実的でない。現在の方向としては，端末属性の多様性を容認したうえで，これらの属性をパラメータ化し，そのパラメータを網で記憶する形式やユーザが指定する形式等を標準化することにより，多様性を吸収していこうという方向にある。このことは，網の反対側にあるホスト（ＰＴ）からみた場合，全パラメータを属性として持つ仮想の標準端末と通信するように見えることとなる。即ち，ＰＭＰプロトコルは，ホスト（ＰＴ）が，網内のＰＡＤ（即ち，仮想標準端末）と通信するためのプロトコルであり，ＰＴから見れば，このプロトコルは，網の入口とのインターフェース（Ｘ２５）より，１つ高いレベルのプロトコルとなる。従って，このＰＭＰ

は，Ｘ２５に一部機能追加した形となろう。ＰＴから見た場合，通信相手がＰＴの場合と，網内のＰＡＤの場合とでは，プロトコルが若干異ることとなるがこれら両者の関係をどのように体系化していくかが今後の課題であろう。

Ｂ．その他の話題

(1) 勧告Ｘ．２５の補充

次のような項目の補充が行なわれている。

① 物理的インターフェース

回線交換網と専用線の２種があることを明記した。

② フレーム・レベル・インターフェース

ＬＡＰの改良，ＩＳＯとの一致。

③ パケット・レベル・インターフェース

通番モジュロ８から１２８拡張板のパケットの形式およびロジカル・チャネルの割合当の不明確事項を修正。

(2) パケット網間インターフェース

早急な標準化の必要性が合意されており，Ｘ２５に基づく，Ｘ．７Ｘが準備されている。Ｘ２５を適用するとして，何が問題となるかが検討されている。問題点としては，ＲＲレスポンスの返し方に２とおりの解釈がある等である。

2.3.3 ＩＮＷＧの動向（End-to-End プロトコル）

End-to-End プロトコル（しばしば，HOST-HOST プロトコルと呼ばれる）は，利用者に対してプロセス間通信機能をあたえるためにパケット交換サービスの上におかれるものである。

あるネットワークの利用者達は，１つのコンピュータ・メーカに結びつくことを欲しないで，いくつかのタイプのコンピュータで使用できるEnd-to-End プロトコルを必要とする。ネットワークは，間もなくあるいは後になって相互結合（inter-connect）されるようになるので，プロセス間通信に使用できる標準的なEnd-to-End プロトコルに対する強い要求がある。

ネットワーク相互結合の問題は，国際情報処理学会（ＩＦＩＰ）の作業グループ（Working Group）6.1において特にとり上げられてきた。このグループ内で，いくつかの提案がなされ，議論されてきた「ＩＮＷＧ６１，ＩＮＷＧ７２，ＣＥＫＡ７４，ＩＮＷＰ２１」。ここでの議論は，徐々にある概念（標準End-to-Endプロトコル「Standard End-to-End protocol」）へ収束するようになり，この提案となった。

A．ネットワーク・アーキテクチャと用語

ネットワークにおいては，ターミナルと同様にコンピュータも種々の方法によってグループ化され構成され得るので（例えば，１つのコンピュータが数個のシステムを含むとか，１つのシステムがいくつかのコンピュータからなっている）仮想ホスト（Virtual Host）の概念を導入する。仮想ホストとは，発進地仮想ホスト（source virtual Host）から目的地仮想ホスト（destination virtual Host）へパケットを送るサブネットにとって，１つの実体として認識できるリソースの集合である。

通常，ジョブ，プロセス，ターミナルなどのようないくつかの実体（プロセスと略す）が，仮想ホストのリソースを共用するであろう（即ち，ほとんどのシステムは，多重プログラミング・システムである。）

これらのプロセスは，サブネットに対してインタフェースを共用しなければならない。更に，これらのプロセスのあるものにとって，パケット交換は機能が少なすぎると考えられる。よって，各々の仮想ホストは，トランスポート・ステーション（Transport Station）をもっている。トランスポート・ステーションは，サブネットに対するインタフェースの多重化とパケット交換サービスに対する価値付加の役目をもっている。トランスポート・ステーションは，トランスポート・プロトコルによって動作する。

（図２－１４参照）

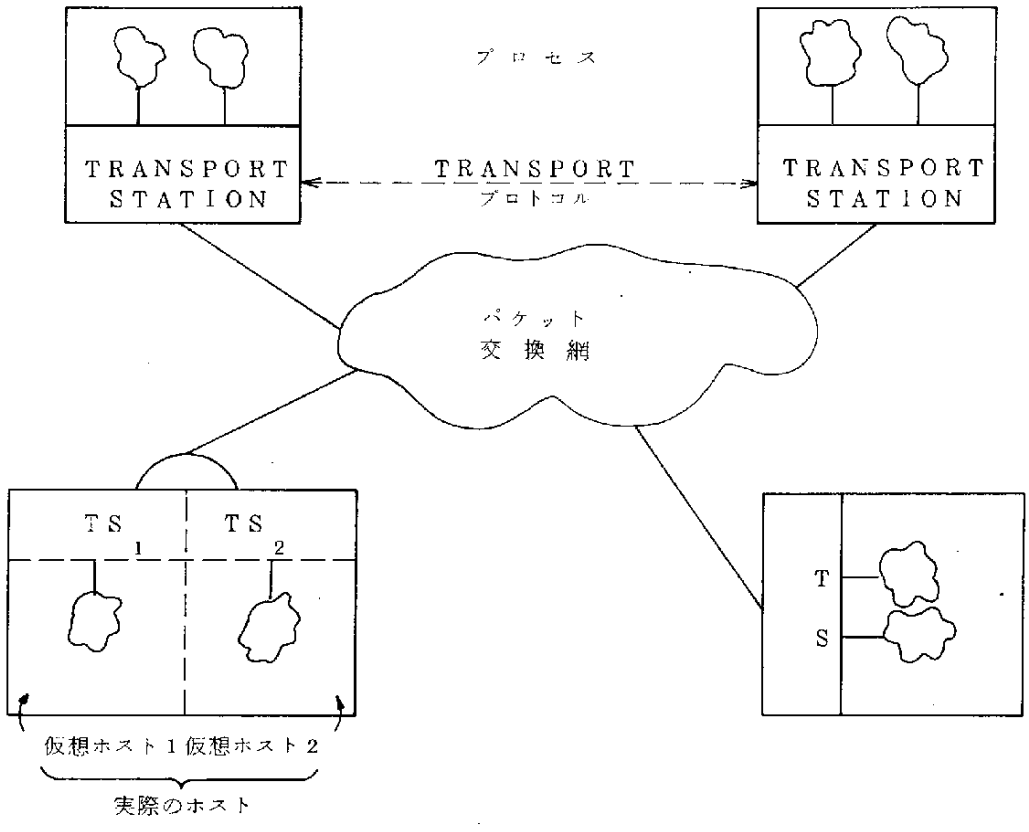


図２－１４　TRANSPORT アーキテクチャ

B．ポートと関連づけ（Ports and Associations）

パケット交換網は，発進地トランスポート・ステーションから目的地トランスポート・ステーションへパケットを運ぶ。発進地と目的地の両方のトランスポート・ステーションは，パケット・ヘッダにあるアドレスによって識別される。（図２－１５参照）

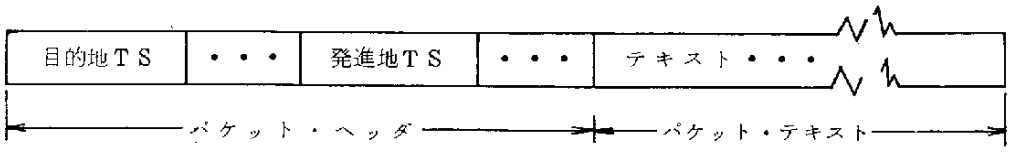


図２－１５　単純化されたパケット・フォーマット

パケットの中で運ばれる情報の最終利用者は，加入者（subscriber）である。この加入者の識別アドレスは，転送中の情報の各々と関連していなければならない（加入者という言葉は，パケット交換網の直接利用者というよりも，むしろトランスポート・ステーションの利用者に対して使用している）。各々のシステムは，加入者のプロセス，リソース，Ｉ／Ｏストリームを識別するそれら自身の方法を持っているので，ポート（ports：PT's）よりなるプロセス間通信の名前空間（name space）を導入する。通信側からみると，トランスポート・ステーションはちょうどポートの集合となっている。（図２－１６参照）

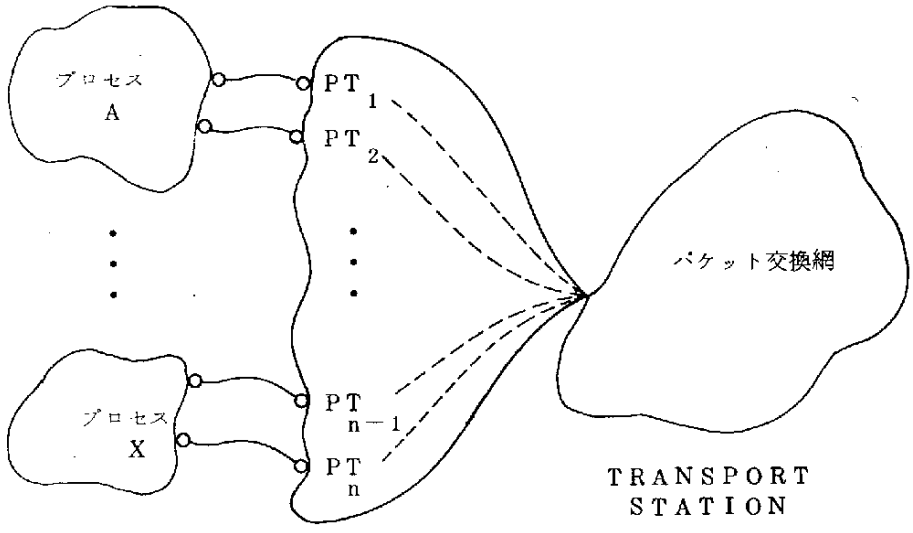


図２－１６　トランスポート・ステーションは，ポートの集合である。

ポート１６ビットの名前によって識別される。トランスポート・ステーションにおいて，ポート名（port indentifier）は，通信のために，内部名（internal name）と局所的に関連づけられている。そのマッピングは，ある場合には動的に行なわれるべきである。しかし，分散管理において動的な名前のアロケーションをするときの同期をとる問題を避けるために，可能ならば静的に決定することをすすめる。しかし，このことはこれ以上の大き

なサイズを否定するものでない。（図２－１７参照）。更に，加入者の識別子からプロセスあるいはジョブへのマッピングは，すべてのトランスポート・ステーション・インプリメンテーションにおいて均一であることを含まない。また，高位プロトコルが，加入者内のポート番号について提案した隣接性を利用することを強制しない。しかし，この隣接性は，より高位のあるプロトコルにとっては便利である。

ポート名のうちのあるもの（例えば，0～255）は，"よく知られたポート（well-known port）"（主として，ロガー，ファイル転送サービス，ホスト・ステータス報告，リモート・ジョブ・エントリ等のような標準的なサービスにあたるものである）としてとっておくべきである。これらの勧告を除いて，加入者Ｉ／Ｏチャンネルとポート名の関係は，プロトコル作製者の判断にまかされている。

簡単さのために，各々のパケットは，１つの発進地ポートから１つの目的地ポートへ行く情報だけを含んでいる。ＩＦＩＰ ＷＧ 6.1がＣＣＩＴＴ へ提案したデータ・グラム（datagram）のフォーマット（ＩＮＷＧ８３）によれば，ポート番号は発信地と目的地アドレス・フィールドの低位のビットのところにおかれてもる。（図２－１８参照）

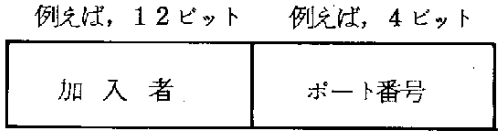


図2－17　ポート名

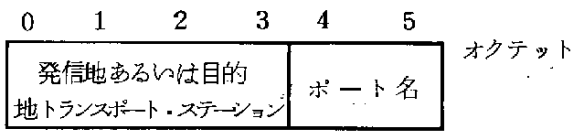


図2－18　ポート・アドレスの完全なフォーマット

トランスポート・ステーションは，パケット交換網のポート・レベルまでの拡張として，付加価値サービスを提供するポート間機構の集合として考えられる。お互いに交信しようとするポートは，何らかの形で"関連づけ（associate）" られなければならない。一対のポートの関連づけ（association）は，ポート間プロトコル（port-to-port protocol）によって制御される。この関連づけは，全二重（full-duplex）であり，発進地，目的地アドレスの組合せによって識別される。一対のポート間では，一組の関連づけだが存在し得る。ポート・アドレス（トランスポート・ステーション名も含む）は，ネットワーク内で通用する名前の最小スペースより構成される。このために都合のよいことに，機能，サービスリソースに対するネットワーク内で通用する名前としてポート・アドレスを使用できる（例えば，ＴＳＳは，よく知られたポートとして識別され，端末は１つのポートとして識別される。）。これは，電話システムにおいて，個人や会社が電話番号によって識別されるのと全く同じである。

外からみたとき，会社の電話番号は共用されているので，同時に，いくつかの通話ができる。そして呼び出した方はすでに誰かが通話をしているかどうかを考える必要はない。逆に

個人の電話番号は共用できないので，一時点に１つの通話しかできない。

同様の思想がポートにも適用できる。もし，ポートが共用可能なリソース（例えば，ＴＳＳ）をアクセスする手段ならば，そのポートは共用されるべきである。つまり，同時にいくつかの関連づけを扱うことができる。関連づけのための名前づけの規則は，図２－１９に示されるように，どのポートもいくつかの関連づけによって共用されることを可能としている。

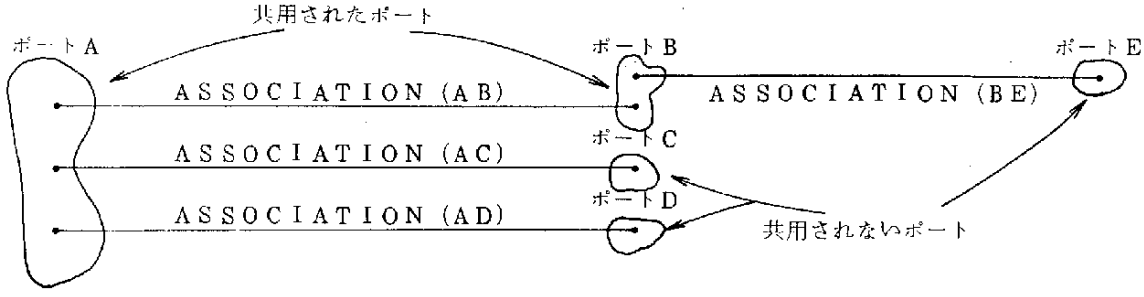


図２－１９　共用された／共用されないポート

C．レター（Ｌｅｔｔｅｒｓ）

トランスポート・プロトコルは，関連づけのできたものの１つのポートからもう１つのポートにレター（ＬＴ）を転送するものである。

レターは，最大長が２５０００オクテットを越える可変長の情報である（フラグメント長が２１６オクテットの場合）。ほとんどすべての物理レコードがレターの中に入れ得るというのがこの考え方である。

このようにして，加入者レベルにおけるデータの分割をさけている。

このようにレター全体を送信プロセスからあたえられ，必要ならばトランスポート・ステーションによっていくつかのパケットに分割される。そして到着するとリアセンブルされて，全体として受信プロセスに渡たされる。

このようにトランスポート・ステーションにおいては，バッファ管理はレターのレベルで行なわれる。

エラー制御とフロー制御は，バッファ管理と結びついているので，それらもまた，レターのレベルで導入される。

必要なときには，レターは最後のものを除いて２１６オクテットの固定長のフラグメント（fragment：ＦＲ）へ分割される。各々のフラグメントは，適当な制御情報をつけて，１つのインター・ネットワークパケット（inter network packet）として送られる。各々のレターは，その関連づけ内では一意的なもので，ある１６ビットの参照番号（reference number:ＭＹ－ＲＥＦ）を持っている。これによって同じ関連づけを通じて送られる異なったレターのフラグメントは，レター内で順序づけられた番号（ＦＲ－ＮＢ）を持つ。ＥＮＤ－

OF-LETTER(EOL) フラグは，レターの最後のフラグメントを示している。

FR-NB（7ビット）とEOL（1ビット）は1つのオクテットにはいる。フラグメントは到着するとリアセンブルされてレターのコピーとなる。（図2－20参照）

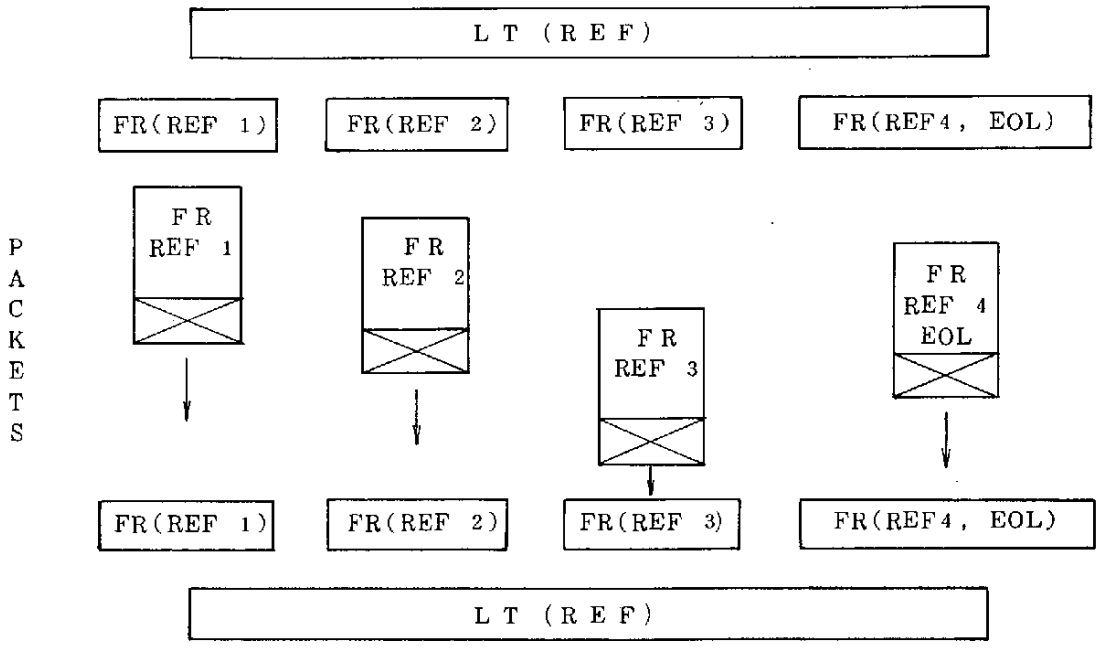


図2－20　フラグメンテーションとリアセンブリ

前処理の機能は，FEP(front-end processor)において処理され得る。そしてレターー交換サービス(letter switching service)は，メイン・プロセッサにおいて行なわれる。

D．関連づけのオペレーション(Operation of association)

関連づけは，結合(liaison)とレター・グラム(lettergram)の2つのモードで同時に動作することができる。レター・グラム・モードでは，レターは，送信側が確認信号(acknowledgment)を受けることを想定してお互いに独立に送られる。

利用者は，レターの交換に先だって，それらのポートをアクティブにする。レター・グラム・モードは，郵便システムにたとえられるレターの交換サービスを行なう。結合モード(liaisonmode)では，レターの伝送に先だって初期設定コマンド(initialization command)が交換される。

初期設定では，次のことが行なわれる。

① 結合(liaison)の両端がアクティブあることの確認。

② オペレーションにはいるサービスの同意。

③ 付随したパラメータの初期化

特にエラーやフローの制御を含んだ種々のオプションがある。結合モードのセッションは，終了コマンド（termination command）の交換によって終了する。

E．トランスポート・コマンド（Transport command）

データ・グラムはトランスポート・ステーションとパケット交換網（ＰＳＮ）との間での情報交換の単位である。

ここではＩＦＩＰ WG6.1のCCITTに対する初期の提案（ＩＮＷＧ83，ＩＮＷＧ84）である標準Ｄ1データ・グラムのフォーマット（standard D1 datagram format）を仮定した。しかし，これは単に具体的にするだけである。データ・グラムは，トランスポート・ステーション間でトランスポート・コマンドを運ぶために使用される。

データ・グラム・ヘッダは関連づけられた発言地と目的地の完全なポート・アドレスを含む。ＰＳＮは，一般的に上位部分（トランスポート・ステーション・アドレス）のみを目的地トランスポート・ステーションへのルーティングのために利用する。

初期の提案においては，トランスポート・ステーション・レベルのヘッダはデータ・グラム・ヘッダとオーバラップ（共用フィールドを持つ）することができることを示していた。

図２－２１において，トランスポート・ステーション End-to -End プロトコル・ヘッダを持ったＤ1データ・グラム・ヘッダのオーバラップを説明している。

提案されたトランスポート・ステーション・ヘッダは，少し長いフィールド幅を持つこと以外は，（ＩＮＷＧ83）において提案されているＥ1フォーマットとほとんど同じである。図２－２３は，データ・グラムとトランスポート・ステーションのヘッダ・フォーマットを詳細に説明している。

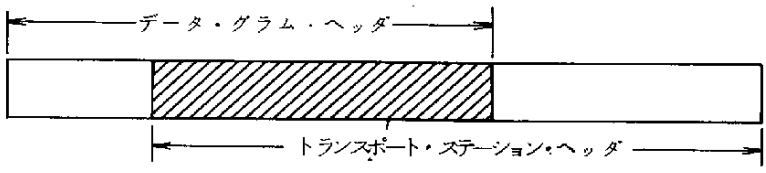


図２－２１　データ・グラムとトランスポート・ステーション・ヘッダのオーバラップ

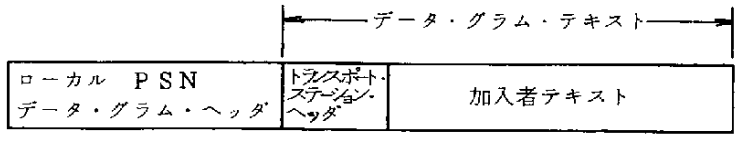


図2－22　ローカルＰＳＮデータ・グラム中のトランスポート・ステーション・ヘッダ
加入者テキスト　（subscriber text）

標準データ・グラム・フォーマットは，ＳＰＮ間でのネットワークの相互結合が行なわれるまで確立されないということが起こるかも知れない。

もしこれが本当ならば，少し異なった形が持ち上がる。このときには，オーバラップは存在しない。図２－２２はこの場合を説明している。

図２－２３において，種々のヘッダ・フィールドの意味は次のとおりである。

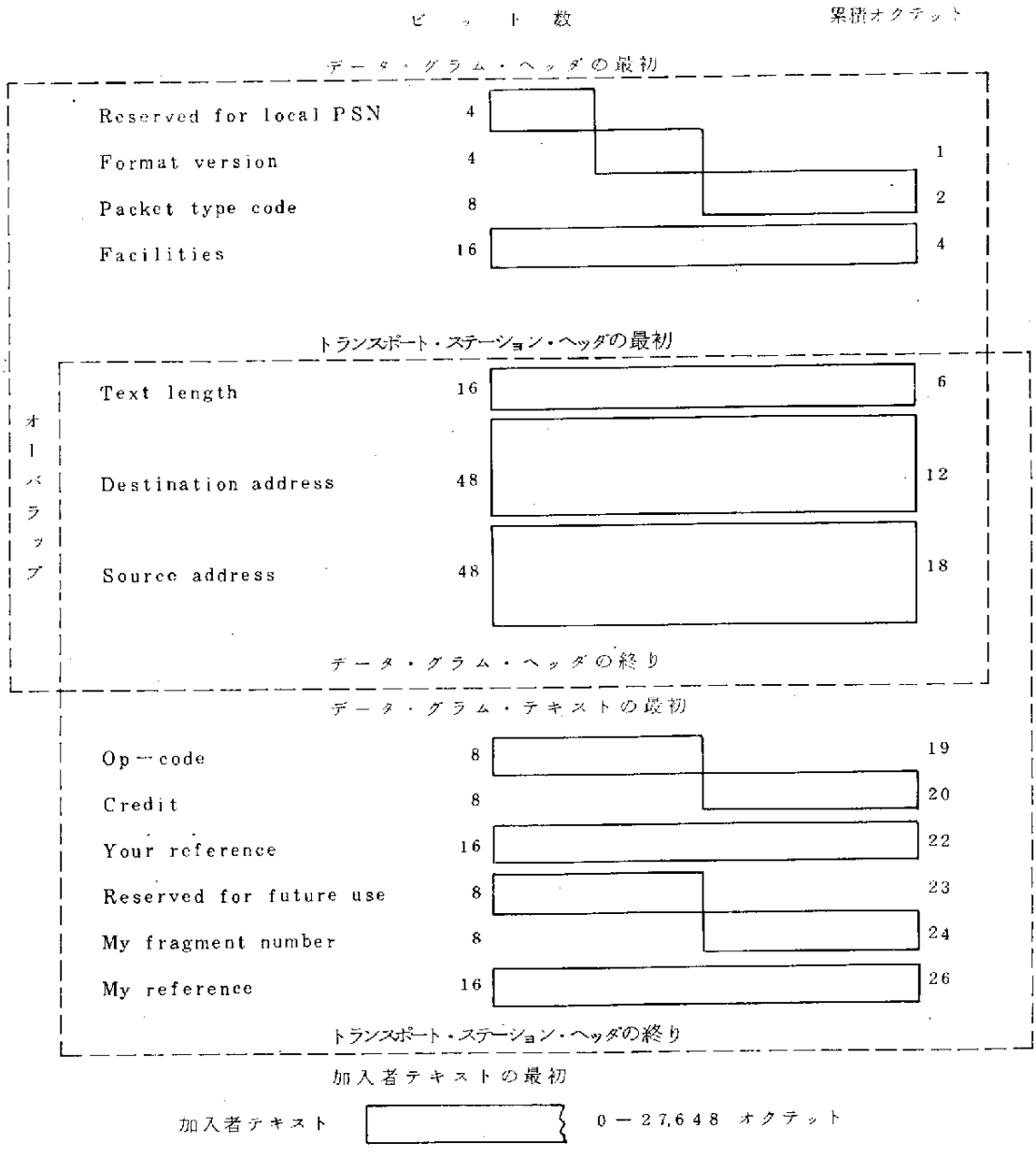


図2－23　トランスポート・ステーション・ヘッダをもったデータ・グラムのフォーマット

① Local PSN－4ビット

このフィールドは，ＰＳＮがすでに持っているローカル・フォーマットにおいても，ネットワーク結合のフォーマットにおいても，動作できるようにする。

② Format Version－4ビット

このフィールドは，ＰＳＮによって認識される他の種々のフォーマットを許す。例えば，将来の拡張，あるいは，データ・グラム以外のサービスのためである。

③ Packet type code－8ビット

このフィールドは，データ・グラムの中で選ばれる種々の特殊な情報パケットの識別を可能にする。例えば，呼（call），診断（diangnostic），ネットワーク・サービス（network service）である。

④ Facilities－16ビット

このフィールドは，ローカルＰＳＮおよび標準インター・ネットワーク・サービスの両方のために多くのサブ・フィールドに分けられる。これは，測定，診断の助け（diagnostic aids），特殊なルーティング（special routing），アカウンティング等である。

⑤ Text lenght－16ビット

このフィールドは，パケット中にあるテキストの長さを示す。データ・グラム(ＣＣＩＴＴ標準）において使われたとき，最大テキスト長が２５５オクテットであるので8ビットだけが意味を持つ。

しかしながら，あるＰＳＮはもっと長いサイズを受け付ける。そして将来の技術では，パケット・サイズの実質的な拡大ができるであろうと考えられる。ここで述べられたEnd-to-Endプロトコルでは，加入者のテキストは，最大長２７６４８オクテットを持つレター に限られているので，加入者テキストの長さとして，１４ビットだけが使用され得る。（即ち，それぞれ２１６オクテットのフラグメントが１２８コである。）

⑥ Address－96ビット

発信地，目的地アドレスは，各々４８ビットの２つのグループに分けられる。各々のアドレスの高位３２ビットは標準インター・ネットワークのトランスポート・ステーション・アドレス（ＰＳＮ名も含む）のために予約されている。各々のアドレスの下位の１６ビットは，ポート名のために予約されている。

⑦ ＯＰ－code－8ビット

このフィールドは，コマンドの受信側で遂行されるべき動作を一意的にする(図2—24参照）。ビット0は，Ｒビットと呼ばれ，コマンドの受信側に対して，確認信号（ＡＣＫ：acknowledgment）を返送するように依頼する（Ｒ＝１）のに使用される。レター・グラム・モードにおいては，レターのフラグメントのどれかのＲビットが1にセットされていれば，レター全体を受信したのちにＡＣＫを送り返さなければならない。（レター・グラム・モードにおける関連づけの操作を参照）。組合モードにおいては，1にセットされたＲビットは，直ちにステータスを報告するという要求と解釈されねばならない。誤り制

御が指定されたときには，ステータスとはＹＲ－ＲＥＦとして送られた最後の値と解釈されなければならない。フロー制御が指定されたときには，ＣＲＯ－ＮＢも含んでいなければならない。ステータスは結合におけるその時点の状態として意味のあるコマンドによって送らなければならない。

ビット１，２は将来の利用のために予約されている。

ビット３は，関連づけのモードを示す（１：レター・グラム，０：結合）

ビット４－７はコマンドを識別する。

ニーモニック＼ビット	0	1	2	3	4	5	6	7	意味
LG－LT	R	0	0	1	0	0	0	0	レター・グラム・モードにおいて，レターのフラグメントを送るために使用される。
LG－ACK	0	0	0	1	0	0	0	1	レター・グラム・モードにおいて，レターのAckを送るために使用される。
LG－NACK	0	0	0	1	0	0	1	0	レター・グラム・モードにおいて，否定のAckを送るために使用される。
LI－LT	R	0	0	0	0	0	0	0	結合モードにおいて，レターのフラグメントおよび逆方向のAckを送るために使用される。
LI－ACK	R	0	0	0	0	0	0	1	結合モードにおいて，Ackを送るために使用される。
LI－INIT	0	0	0	0	0	0	1	1	結合モードの初期設定に使用される。
LI－TERM	0	0	0	0	0	1	0	0	結合モードを終了するために使用される。
LI－PURG	R	0	0	0	0	1	0	1	結合モードにおいて割り込みを送り，残っているレターの消去のために使用される。
LI－ERR	0	0	0	0	0	1	1	0	結合モードにおいて，エラー信号を送るために使用される。
LI－NACK	R	0	0	0	0	0	1	0	結合モードにおいて，否定のAckを送るために使用される。

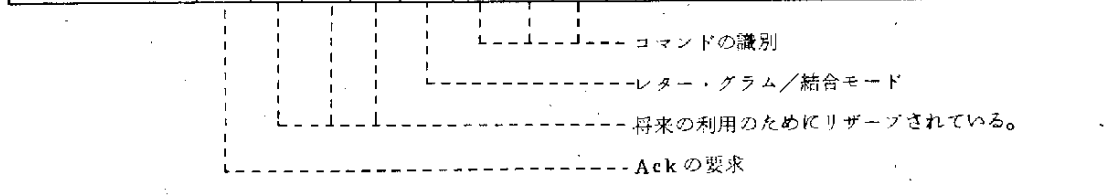


図２－２４　トランスポート・コマンドのＯｐ-code

⑧ Credit— 8ビット

図2－27に示されているのを除いて，このフィールドは，オペレーションが結合モードのとき，発信地，目的地ポートの間のレターのトラフィック・フローの制御のために使われる。

Creditは，最後にＡＣＫを返されたレターの次にレターを送る許可のためのものであり，結合ごとに受信側トランスポート・ステーションから送信側トランスポート・ステーションから送信側トランスポート・ステーションに与えられる。

受信側トランスポート・ステーションは，より小さいcredit番号を送ることによってcreditを自由にとり返すことができる。というのは，最後にＡＣＫを受けたレターの参照番号と最新のcreditの合計が，送信側トランスポート・ステーションによって発信できる最大の参照番号（reference number ）を決定する。

⑨ Fragment number（ＦＲ—ＮＢ）— 8ビット

ＬＩ—ＬＴとＬＧ—ＬＴコマンドにおいて，ＦＲ—ＮＢフィールドの下位の7ビットは，パケットの加入者テキスト・フィールドの中に現われるレター・テキストの第1フラグメントに関係する番号である。

ＦＲ—ＮＢの高位のビットは，End-of-Letter（ＥＯＬ）ビットと呼ばれる。パケットに入っているレター・テキストの最後のフラグメントは，これが，レターの最後のフラグメントであることを示すために使用される。

他のコマンドにおいては，ＦＲ—ＮＢフィールドは，ＯＰ— code フィールドの拡張として使われる。（図2－25参照）

コ マ ン ド	フラグメント番号フィールドの使用
LG — LT	ＥＯＬ，ＦＲ—ＮＢ
LI — LT	ＥＯＬ，ＥＲ—ＮＢ
LI — INIT	要求されるサービス
all other commands	未　使　用

図2－25　フラグメント番号フィールドの利用

⑩ Your Reference（YR—REF）— 16ビット

このフィールドは，図2－26に示されたもの以外では，ＡＣＫを受けとったレターのreferenceを示す。

コ マ ン ド	Your Reference フィールドの使用
LG －LT	未　使　用
LI －INIT	送信するレターの最大長
LI －ERR	エラーのあったレターのReference
LG －NACK	未だ決められていない
LI －NACK	同　上

図２－２６　Your Reference フィールドの利用

⑪　My Reference（MY－REF）－１６ビット

このフィールドは送られた最後のレター（LI－PURGあるいはLI－TERMのコマンドにおいて）あるいは送られているレター（加入者テキストがある場合には，例えば，LI－LTあるいはLG－LTコマンド）のリファレンスを示す。LI－INTコマンドにおいては，それは結合を通じて送られるレターに対する番号付けの初期値を示す。

種々のトランスポート・コマンドは，図２－２７に要約されている。

	CRD－NB	YE－REF		FR－NB	MY－REF
OP－COD	CREDIT NUMBER	YOUR REFERENCE	RESERVED FOR FUTURE USE		MY REFERENCE
LG－LT				FR－NB	MY－REF
LG－ACK		YR－REF			
LG－NACK		YR－REF			
LI－LT	CRD－NB	YR－REF		FR－NB	MY－REF
LI－ACK	CRD－NB	YR－REF			
LI－INIT	CRD－NB	MAX－LT－LG (Octets)		SERVICES	MY－REF
LI－TERM	TERM CODE	YR－REF			MY－REF
LI－PURG	CRD－NB	YR－REF			MY－RFF
LI－ERR	ERR－CODE	YR－REF			Last value sent as MY－REF
LI－NACK	CRD－NB	YR－REF			

図２－２７　Transport コマンドのフォーマット

F．レター・グラム・モードにおける関連づけ（Association operation in lettergram mode）

レター・グラム・モード（lettergram mode）において，コマンドのセットとしてＬＧ－ＬＴとＬＧ－ＡＣＫ（そして多分ＬＧ－ＮＡＣＫ）に制限されている。そして，Ａｃｋを用いて，レターの交換を行なわせている。

送信側は，その関連づけの内部においてユニークなＭＹ－ＲＥＦを各々のレターにつける役目を持っている（そして，結合モードで動作している同一の関連づけによって使われるreferenceとは独立である。）。これらのユニークなＭＹ－ＲＥＦを発生することについては，次の理由によって何の問題もない。

① referenceフィールドは１６ビットの長さである。

② データ・グラムの最大の存在期間は短かい時間に，制限される（例えば，３０秒「ＩＮＷＧ８２」）というのがＩＦＩＰのＷＧ 6.1によって勧告されている。

③ レター・グラム・モードは広い帯域を指向するものでない。

全体のトランスポート・ステーションに対して，１つのカウンターで十分であろう。

もしＬＧ－ＬＴコマンドのＲビットが１にセットされているならば，受信側は，レター全体を受け取った後，直ちに受け取ったレターのＭＹ－ＲＥＦと同じものをＹＲ－ＲＥＦに入れたＬＧ－ＡＣＫを送り返すことが要請されている。

タイム・アウトは，送信側がレター（あるいはＡｃＫ）の紛失を検出し，これを加入者に通知することを可能にする。

もし目的地ポートが存在しない時，アクティブではない時，あるいはバッファがない時には，Ａｃｋは返送されず，送信側ではレターの紛失と同等となる。

可能な回復（recovery）手段は，加入者がレター・グラムを再送することである。

パケットは紛失するかも知れないので，受信側でのリアセンブルは，それぞれのレターに関してタイム・アウトによって保護されている。

このタイム・アウトは，（時間的に）最初のフラグメントを受け取ったときにセットされ，それぞれのフラグメントを受け取るとリセットされる。最終的にはすべてのフラグメントを受け取ったときに消去される。

もしタイム・アウトが発生したら，リアセンブルは放棄され，レターはエラーがあったものと考えられる。（もしエラー制御が利用されているならエラーはたぶん回復されるであろう。）

G．結合モードにおける関連づけ操作（Association operation in liaison mode）

結合モードでは，レターが送られる以前に，初期設定が行なわれなければならない。そして，終了処理（termination）は，結合モードのセッションを終了させる。

初期設定と終了処理は，単純で対象的なランデブー手法を使うことによって行なわれる。（図２－２８参照）

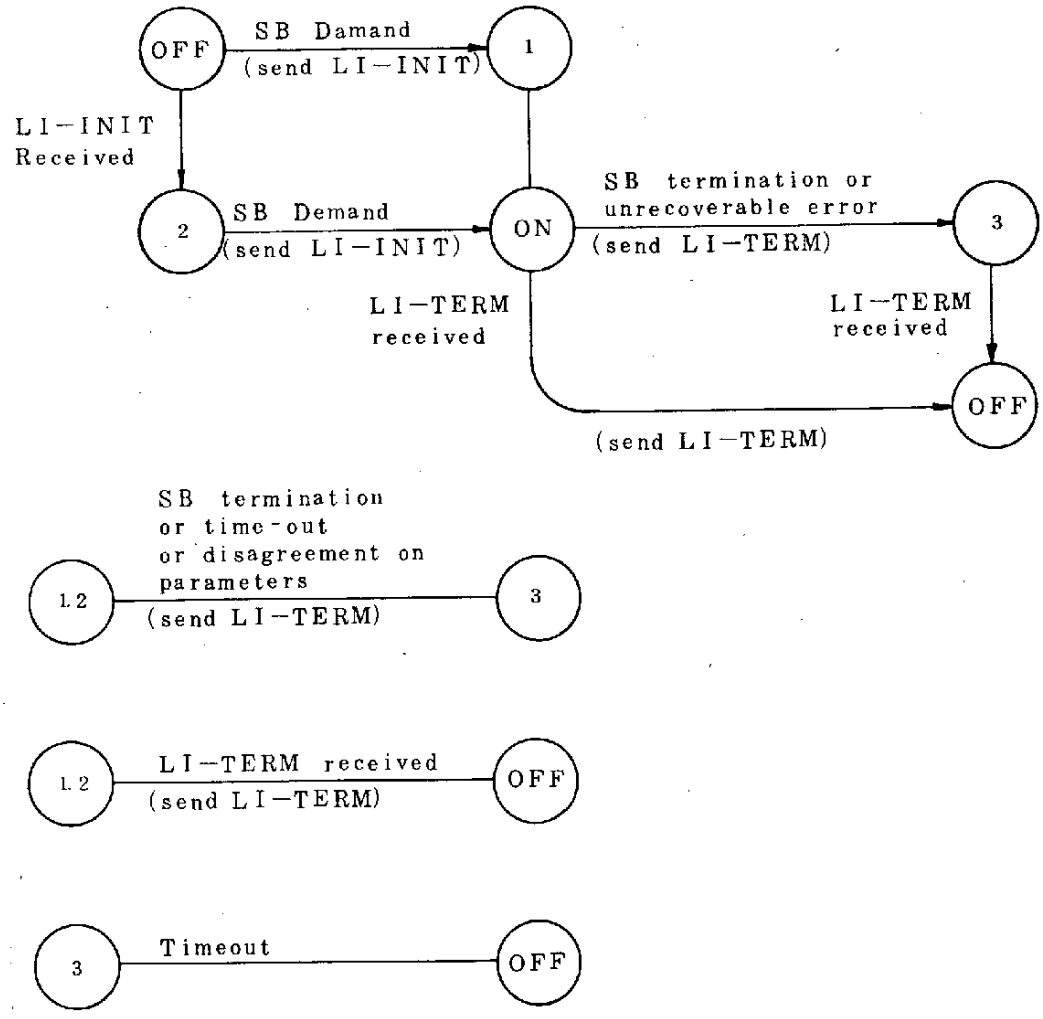


図２－２８　結合モードの初期化と終了

初期設定は，もし交換されたＬＩ－ＩＮＩＴコマンドが一致したときに成功する。即ち，（もしあれば）同じパラメータを持った同じセットのオプションを持つことを要求する。途中の状態は，タイム・アウトにより保護されている。予期しないコマンドは，何もしないで無視される。

種々のオプションが結合モードにおいて動作するよう設定され得る。オプションの要求は，図２－２９に示されるようにＬＩ－ＩＮＩＴコマンドのサービス・フィールド中の適当なビットのセット（０：ｏｆｆ，１：ｏｎ）によって指示される。

結合モードにおいては，レターは順番に番号づけされる。そして，それはＬＩ－ＩＮＩＴにおいて送られたＭＹ-ＲＥＦの次の番号から始められる。

reference フィールドは大きいので（１６ビット），reference はサイクリックに再使

用される。結合モードはレターの順序づけを可能にする。

LI－PURGE コマンドは，結合を通じて，一方の加入者からもう一方に"割り込み"信号を送るときに利用される。

ビット番号	選択されるサービス
0	レターにするエラー制御
1	レターに対するフロー制御
2	多重フラグメント・パケット
3	レターに対するチェック・サム
4－7	将来の利用のためリザーブされている。

図２－２９　ＬＩ－ＩＮＩＴコマンドのサービス・フィールド

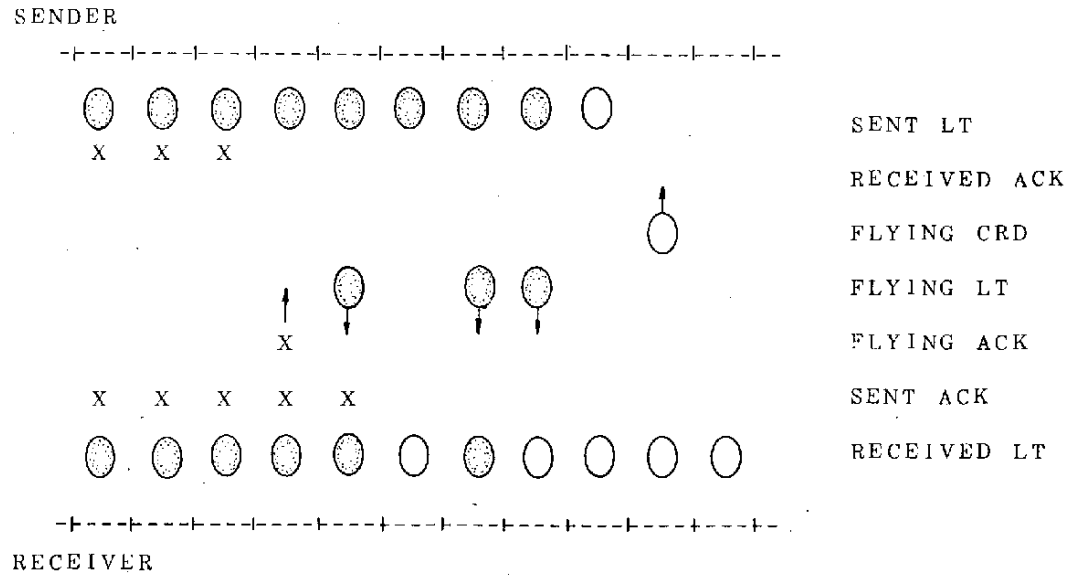


図２－３０　レターのエラー制御

"割り込み"信号の正確な翻訳は受信側加入者に従う。しかし，現在の処理を中断あるいは終了して，割り込みのあとに送られたレターをスキャンし始めるような方法を一般的な形のＯＳはとるべきである。ＬＩ－PURGE は，テキストを持たないで，ＦＲ－ＮＢフィールドは使わないけれども，ＬＩ－ＬＴに割り当てられたＭＹ－ＲＥＦをフィールドに含んでいる。送信側トランスポート・ステーションは，フロー制御によって，この reference 番号でＬＩ－ＬＴの伝送が禁止されていたとしても，割り込み情報を送らなければならない。同様に，もし直前のまだ受け取ってないレターがありＬＩ－PURGEのＡｃｋされることにより，以前のレターが紛失する可能性があったとしても（もしエラー制御が利用されていれ

ば），受信側トランスポートはただちにＬＩ－ＰＵＲＧＥ を受け取らなければならない（そしてＡｃｋ を返す）。

① 受信プロセスに"割り込み"信号を渡す。

② ＬＩ－ＰＵＲＧＥ コマンドまでの未処理レターを消す。

Ｈ。基本サービス（Ｂａｓｉｃ ｓｅｒｖｉｃｅ）

もし，オプショナル・サービスが指定されないならば，レターは基本サービス・モードで交換される。基本サービスは，次のような違いはあるが，レター・グラムと似かよっている。

① 初期設定は，両端がアクティブである（あった）ことと，コミュニケーションをしようとしていることを確かめる。

② 消去と割り込み機能が使用できる。

Ｉ．エラー制御（Ｅｒｒｏｒ ｃｏｎｔｒｏｌ）

もし，エラー制御が指定されたならば，すべてのレターは，最終的にＡｃｋ されなければならない。即ち，

① もし，レターが紛失したならば，ＹＲ－ＲＥＦ は進まないであろう。

② １つのＬＩ－ＡＣＫは数個のレターのＡＣＫをすることができる。

ということである。

送信側は，レターの最後のフラグメントを送ってから，最大の遅れ時間以内にＡＣＫを受け取るものと思っている。

もしＡＣＫを受信しないならば，すべてのＡＣＫを受け取っていないレターは紛失したものと考えられ，最初のＡＣＫを受け取っていないレターから再送する。ＡＣＫの受信が再び期待される。もしＡＣＫが受信されなかったならば，この処理が再びくり返される。

もし，レターが"Ｎ"回送られてそれでも成功しなかったならば，送信側のトランスポート・ステーションは，この状態を加入者に報告する。

この方法は，単純であるが，ある場合には最適なものではないかもしれない。

別の再送方式，多分，否定のＡＣＫ（ｎｅｇａｔｉｖｅ ａｃｋｎｏｗｌｅｄｇｍｅｎｔ）を含むものが将来の研究として考慮することができる。

Ｊ。フロー制御

このオプションは，フラグメント化－リアセンブリの頭のところで，レターのレベルでのフロー制御をさせる（バッファの管理のために）。

結合が使われているとき，それは，エラー制御と関連を持ち，両方向に対し，次のようになされる。

- フロー制御の初期設定において，結合の両端は，それぞれの方向に送られるレターの最大長に関して同意をする（サイズは，それぞれの方向において違っていてもよい）。これはＬＩ－INIT コマンドのＭＡＸ－ＬＴ－ＬＧフィールドで示される。
- 受信側は，Credit（ＣＲＤ）を，送信側に対して確保する。１つのレターを送ることの許可を意味する。

それぞれの確保はＬＩ－ＬＴ，ＬＩ－ＡＣＫコマンド中のＡＣＫ（ＹＲ－ＲＥＦ）に関係している。８ビットのCredit番号パラメータ（ＣＲＤ－ＮＢ）は，以下のことを意味している。つまり，（ＣＲＤ－ＮＢ）＝０は，レターを送ってはいけない。この他の時は，（ＹＲ－ＲＥＦ）＋１から（ＹＲ－ＲＥＦ）＋（ＣＲＤ－ＮＢ）までのreference 番号をもったレターを送ってもよい。

- ＣＲＤ－ＮＢ＝２５５は，好きなだけレターを送ってよいことを意味する。
- 送信側トランスポート・ステーションは，受信側の制御の下での処理によって，送信量の上限を制限される。この制限は，LI-ACKまたはＬＩ-ＬＴが受信されたときアップ・デートされる。

Ｋ．多重フラグメント・パケット（Multi-fragment packet）

このオプションは異なったパケット・サイズをもったネットワーク間での通信を可能にするものである。（ＣＣＩＴＴによって提案されているように）データ・グラムの最小長は２５５オクテットである。

フラグメント・サイズ（２１６オクテット）は，ＣＣＩＴＴデータ・グラム中のトランスポート・ヘッダとうまくあうように選ばれた。

しかしながら，新しい技術がとり入れられたとき，ネットワークはデータ・グラムよりもっと長いパケットを運ぶであろう。

多重フラグメント・パケット・オプションは，それが指定されれば，ＬＩ－ＬＴコマンドの加入者テキスト・フィールドに同じレターの数個の連続的なフラグメントを両端で入れることを可能にする。唯一の制限はパケット・サイズである。

ＦＲ－ＮＢフィールドは，加入者テキストの最初のフラグメントの番号を示し，ＥＯＬビットは，加入者テキストの最後のフラグメントがレターの最後であるかどうかを示す。

パケットは，異なったパケット長を使っている２つのネットワーク間のゲートウェイを通じて手渡されるので，ゲートウェイが適当なトランスポート・ヘッダを持ついくつかの小さいパケットに分割する必要があるであろう。

L．レターのチェックサム（Checksum on letters）

チェックサムはそれぞれのレターに関連している。それは，レターの最後のオクテットにおかれ，レターの内容のエラーの検出に使用される。このオプションは，のちほど正確に決められるであろう。

M．ユーザ・インタフェース（User interface）

ユーザ・インタフェースを与えることができ，ネットワーク・アクセス法（Network Access Method ）として与えられることができる基本的な部分を示すだけである。

OPEN－PT：ポートをアクティブにする。

RECV－LG：レダー・グラム・モードで遠隔地ポートからレターを受け取る。

SEND－LG：レター・グラム・モードでレターを送る。

CLSE－PT：ポートをアクティブでなくする。

OPEN－LI：結合を初期設定する。

RECV－LI：結合モードでレターを受け取る。

SEND－LI：結合モードでレターを送る。

CLSE－LI：結合を終了処理する。

2.3.4 ＩＢＭの動向

昭和４９年９月に，昭和５０年１０月を最初のリリースとするＳＮＡ（SYSTEM NETWORK ARCHITECTURE）が，発表された。このシステムは，１ホスト・システムによる階層構成によって，ネットワークを構築し，ホストが集中管理するものであったが，昭和５１年１１月には，機能拡張が発表され，複数ホストによるネットワーク化が可能となった。このシステムの最初のリリース予定は，昭和５３年３月となっている。

A．ＳＮＡの概要

ＳＮＡは，機能の分散，接続の独立性，装置の独立性，構成の柔軟性を特徴としている。

(1) 機能の分離

ＳＮＡの中心となる概念は，通信システムの機能を，はっきりと定義された１つの論理的階層に分割したことにある。即ち，主な機能層として，適用業務層，機能管理層，伝送サブシステム層に分割した。ここでは，基本プロトコルに対応する伝送サブシステム層のみを取上げる。

(2) 伝送サブシステム

伝送サブシステムは，ＮＡＵ（ネットワーク・アドレス・ユニット）の間のデータ交換を行なう。３種類の要素が伝送サブシステムを構成する。それらはデータ・リンク制御

（ＤＬＣ），経路制御（ＰＣ），伝送制御（ＴＣ）である。ＤＬＣ要素は，ノード間のリンクを管理する。ＰＣ要素は，データ単位に対して２つのネットワーク・アドレス間の経路指定を行なう。ＤＬＣとＰＣの２つは，全体として，共有される共通ネットワークを構成する。共通ネットワークは，伝送制御要素間でデータ単位を伝送する。TCは，セッション制御と，共通ネットワークへのデータの流入流出を管理する。

① ネットワーク・アドレス可能単位（ＮＡＵ）とセッションの種類

ＳＮＡは，３種類のＮＡＵを定義している。それらは，システム・サービス制御点（ＳＳＣＰ），物理装置（ＰＵ），論理装置（ＬＵ）である。セッションは，これらＮＡＵ間に張られる経路であり，ＬＵ－ＬＵ，ＬＵ－ＳＳＣＰ，ＰＵ－ＳＳＣＰの３種類がある。

② ＴＣ要素

これは，ＮＡＵに伝送サブシステムへの直接のアクセスを可能とする。これは，結合点管理，セッション制御（ＳＣ），ネットワーク制御（ＮＣ）からなる。

結合点管理は，ＮＡＵ，ＳＣ，ＮＣから受ける情報（要求応答単位ＲＵという）および制御情報を受け，この制御情報から，要求応答ヘッダ（ＲＨ）を作り，ＲＨ－ＲＵ（ＢＩＵという）をＰＣに渡す。また，ＢＩＵを受取った結合点管理は，ＲＨを解読する。結合点管理が行なう機能には，要求あるいは応答を宛先に送ること，送受されるＢＩＵに対して順序番号を割り当てたり検査したりすること，応答を要求に対応させ，且つそれを適切な順序に保つこと，データの流れのペーシング（整調）および例外状況とセンス情報を作り出したり，それを処理したりすることがある。

ＳＣは，セッションを設定し，セッションに必要な資源を獲得するために使用される。

ＮＣは，ＮＡＵ間通信のために即に設定されているセッションを使って，結合点管理とＰＣが共通ネットワークを通して，通信する手段を提供する。

③ ＰＣ要素

これは，共通ネットワークのデータ・リンク資源を管理し，ＢＩＵをその共通ネットワークを通じて伝送する。ＰＣは，ＮＡＵの場所とそれを結ぶ経路を知っており，BIUを実際に伝送されるデータ単位（ＢＴＵ：伝送ヘッダとしてのアドレッシング情報，マッピング情報，順序標識等とＢＩＵの組合せたものをＰＩＵというが，これをブロック化したもの）である。

④ ＤＬＣ要素

これは，個々のデータ・リンクを管理する。各ノードにあるＤＬＣが，そのノードにつながるリンクを管理し，物理的な構成によって，一次局または二次局としての機能を果す。このプルトコルは，ＳＤＬＣもその一形式であり，また，システム／３７０のデータ・チャネルとそれに付随するプロトコルも１つのデータ・リンク制御の形式である。

ノード間を伝送されるデータ単位を基本リンク単位（ＢＬＵ）と呼ぶ。ＳＴＵは，データ・リンク制御により，リンク制御情報（リンク・ヘッダとトレーラ）が加えられ，ＢＬＵとされる。

⑤　ネットワーク・ノードの種類

ネットワークの物理構成は，ホスト，通信側制御装置，集合制御装置および端末装置の4種類のノードによって定義される。ここで注意すべきことは，物理構成には，入出力装置も物理的実体として存在する。しかし，入出力装置は，ネットワーク・アドレスを持たず，ＳＳＯＰによって使われる構成情報の中には存在しない。即ち，入出力制御装置は，集合制御装置または端末装置または端末装置によって制御され，割り振られる物理的資源である。ホスト・ノードと通信制御装置は，サブエリヤを制御できる。通信制御装置ノードは，中継機能と境界機能の2種類の機能を提供できる。中継機能を提供する通信制御装置ノードは，完全ネットワーク・アドレスの処理により，メッセージを他のサブエリヤへ送る。境界機能を提供する通信制御ノードは，完全ネットワーク・アドレスを，隣接する集合制御装置あるいは端末装置ノードのために局所アドレス形式に変換する。集合制御装置と端末装置ノードは，セッション内のデータ流れのスケジューリングのサポートを接続している通信制御装置ノードから受ける。通信制御装置ノード，また，それに接続された端末ノードに対して，基本的な物理装置サービスを提供する。

⑥　データ・リンクの構成

ネットワークのノードを結合するデータ・リンクには，システム／３７０データ・チャネルとＳＤＬＣは，２地点間，分岐，ループなど多様な通信構成を選択できる。

以上の動向からネットワーク，特に通信サブネットワークへ接続条件の標準化については，通信サブネットワークの提供者が公衆法によってその一元的運営を保証されている公社であるため，比較的容易に実施されうる。もちろん，監督官庁の指導のもとに，国民のコンセンサスを得てから実現されうるものである。

表２－２は新データ網サービスの内のパケット交換網サービスを利用する加入者が守るべき標準のプロトコル，インタフェースである。すでに各メインフレーム・メーカが発表しているネットワーク・アーキテクチャがパケット交換網を利用する場合には，これらのプロトコル条件等を守る必要がある。

なお，表２－２は，文献（情報処理，１９７７，ｖＯＬ１８，ＮＯ１０，「公共情報システムにおける広域ネットワークの構成と標準化」）より掲載させていただいた。

表２－２　プロトコルモジュール一覧

規定	No.	モジュール名 略称	モジュール名 名称	概要
速度クラス	11	48K b/s	同左	端末の通信速度について規定.
	12	9.6K b/s	〃	〃
	13	4.8K b/s	〃	〃
	14	2.4K b/s	〃	〃
	15	1,200b/s	〃	端末の通信速度，調歩ひずみについて規定.
	16	300b/s	〃	〃
	17	200b/s	〃	〃
物理条件	21	DIS4903	15ピンコネクタ	ISO 標準DIS4903に準拠.
	22	IS 2110	25ピンコネクタ	ISO 標準IS2110に準拠.
	23	IS 2593	34ピンコネクタ	ISO 標準IS2593に準拠.
電気条件	31	V. 10	IC 用不平衡複流回路	CCITT 勧告V.10に準拠.
	32	V. 11	IC 用平衡複流回路	CCITT 勧告V.11に準拠.
	33	V. 35	48K b/s専用平衡複流回路	CCITT 勧告V.35の電気条件を規定する部分に準拠.
	34	V. 28	不平衡複流回路	CCITT 勧告V.28に準拠.
接続回路とその動作	41	X. 21	同期伝送サービス	CCITT 勧告X.21のうち，専用線に関する規定の部分を適用.
	42	X.21bis (V.35)	同期48K b/s伝送Vサービス	CCITT 勧告V.35に準拠.
	43	X. 21bis (V.24)	同期伝送Vサービス	CCITT 勧告V.24同期伝送の規定*に準拠. (* 速度ごと規定).
	44	X. 20	調歩伝送サービス	CCITT 勧告X.20に準拠.
	45	X. 20bis (V.24)	調歩伝送Vサービス	CCITT 勧告V.24調歩伝送の規定*に準拠. (* 速度ごと規定).
伝送制御	51	HDLC-BA	バランス形HDLC手順	ISO勧告 IS3309, DIS4335に準拠.
	52	HDLC-UA1	アンバランス形HDLC一次局手順	ISO 勧告IS3309, DIS4335, DIS6159に準拠.
接続制御	61	X.25 call	X.25 接続制御手順	CCITT 勧告X.25のうち接続制御部分の規定に準拠.
	62	X. 20	調歩伝送サービス	CCITT 勧告X.20に準拠.
	63	NCU	網制御装置操作手順	CCITT 勧告X.20に準拠し接続制御を端末に代って行う装置.
	64	UI	UI 接続制御手順	HDLC手順の UIフレームを使用して接続制御を行う手順.
データ転送	71	X.25data	X.25データ転送手順	CCITT 勧告X.25のうちデータ転送の規定部分に準拠
	72	Basic-H	会話形ベーシック手順	ISO標準, IS1745, IS2110, IS 2628, IS 2629 (JIS C 6362) に準拠した手順.
	73	Basic-F	全二重形ベーシック手順	
	74	DEL #1	セット #1デリミタ手順	ベーシック系の端末を対象としたデリミタの手順.
	75	DEL #2	セット #2デリミタ手順	無手順系の端末を対象としたデリミタの手順.
	76	DEL #3	セット #3デリミタ手順	〃
	77	HDLC-UA2	アンバランス形HDLC二次局手順	ISO標準 IS3309, DIS4335, DIS6159 に準拠.
端末制御	81	X.25 TC	TCC 端末制御	標準伝送制御コードを使用した端末制御の規定.
各種サービス	91	B. CUG	ベア形閉域接続	サービスの仕様と利用のための規定.
	92	CUG	グループ形閉域接続	〃
	93	DC	ダイレクトコール	〃
	94	AD	短縮ダイヤル	〃
	95	ID	相手通知	〃
	96	CC	料金のセンター一括払い	〃

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 ICCC 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 ICCC 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)
(20) ISO/TC 97/SC 6 N 1167: Agenda item 24-Future program of work, United Kingdom contribution on high level protocols, p. 5 (1975)
(21) ISO/TC 97/SC6 N 1173: Working Document from ECMA TC 9 to ISO/TC 97/SC 6, p. 5 (1975)
(22) ISO/TC 97/SC 6 N 1249: Minutes of meeting of ad hoc WG 1, p. 22 (1975)
(23) Cerf, McKenzie, Scantlebury, Zimmermann: A proposal for an Internetwork End-to-End Protocol, INWG General Note # 96, p. 29 (1975)
(24) File Transfer Protocols, NIC # 17759, p. 50 (1973)
(25) Remorte Job Entry Protocols, NIC # 12112, p. 24 (1974)
(26) H. Zimmermann: Virtual Terminal Protocol-Prosed Specifications, Reseau Cyclades Ter # 503.1, p. 14 (1974)
(27) H. Zimmermann: Protocole Client-Serveur de Terminaux, Reseau Cyclades Ter # 502.2, p. 8 (1974)
(28) EPSS Liaison Group: An Interactive Terminal Protocol, INWG General Note # 94, p. 61 (1975)
(29) ESPP Liaison Group: A Basic File Transfer Protocol, INWG General Note # 93, p. 46 (1975)
(30) P. Schicker: Virtual Terminal Protocol, INWG Protocol Note # 32, p. 25 (1976)
(31) P. Schicker, et al.: Bulk Transfer Function, INWG Protocol Note # 31, p. 24 (1976)
(32) R. Scantlebury, P. Wilkinson: The National Physical Laboratory Data Communication Network, Proc. of 2nd ICCC, Stockholm, pp. 223～228 (1974)
(33) 日本情報処理開発協会: コンピュータ・ネットワーク JIPNET の研究開発, 50-S 003, p. 361 (1976)

(34) 山本，伊藤，小川，鍛冶他：情報処理学会第16回大会論文集，論文番号 # 108～# 114，pp. 215～228 (1975)
(35) 電子計算機利用に関する技術研術会：リソースシェアリングシステム，p. 197 (1976)
(36) H. Zimmenmann : High Level Protocol Standardization : Technical and Political Issues, Proc. of 3rd ICCC, pp. 373～376 (1976)
(37) 北川敏男，島内武彦編：広域大量情報の高次処理，東京大学出版会，p. 1300 (1976)
(38) 吉田，松下：パケット交換網の現状と国際標準化について，情報処理，Vol. 16, No. 3, pp.220～229 (Mar. 1975)
(39) CEKA 74 – V. Cerf, R. Kahn, "A Protocol for Packef Network Interconnection," IEEE Transactions on Communications, May 1974.
(40) INWG 60 – L. Pouzin, "A Proposal for Interconnecting Packets Switching Networks," March 1974. IFIP W. G. 61 General Note 60.
(41) INWG 61 – H. Zimmermann, M. Elie, "Transport Protocol. Standard Host-Host Protocol for Heterogeneous Computer Networks," IFIP W. G. General Nots 61.
(42) INWG 72 – V. Cerf, Y. Dalal, C. Sunshine, "Specification of Internetwork Transmission Control Program (revised)," IFIP W. G. 61 General Note 72, December 1974.
(43) INWG 73 – V. Cerf. "Minutes of the Stockholm meeting of IFIP W. G. 61," IFIP W. G. General Note 73, August 1974.
(44) INWG 76 – A. Mckenzie, "Comments on the document "Contribution on packt formats in public data networks" contained in INWG 55," IFIP W. G. 61 General Note 76, February 1975.
(45) INWG 79 – L. Pouzin, "Standards in Data Communication and Computer Networks," IFIP W. G. General Note 79, March 1975.
(46) INWG 82 – IFIP W. G. 61, "Timing Parameters of Packet Switched Data Networks-User Implications," IFIP W. G. 61 General Note 82, May 1975 (Submitted to CCITT as Delayed Contribution CCMVII D68).
(47) INWG 83 – IFIP W. G. 61, "Basic Message Format for Inter-Network Communication," IFIP W. G. 61 General Note 83, May 1975 (Submitted to CCITT as Delayed Contribution COM VII D69).

(48) INWG 84 — IFIP W. G. 61, "Data Commuications Standards," IFIP W. G. 61 General Nots 84, May 1975 (Submitted to CCITT as Delayed Contribution COM VII D70).

(49) INWP 10 — A. Belloni, M. Bozetti, G. LeMoli, "Routing and Internetworking," IFIP W. G. 61 Protocol Note 10, August 1974.

(50) INWP 11 — A. Belloni, M. Bozetti, G. LeMoli, "Remarks on the D—Format," IFIP W. G. 61 Protocol Note 11, November 1974.

(51) INWP 16 — A. McKenzie, "Proposed Revision of D—Format," IFIP W. G. 61 Protocol Note 16, May 1975.

(52) INWP 19 — Rapporteur on Packet Switching, "Proposals for Recommendations," CCITT Rapporteur Group on Question 1/VII—Point C, March 1975.

(53) INWP 21 — G. LeMoli, "A Proposal for Fragmenting Packets in Internetworking," IFIP W. G. 61 Protocol Note 21, April 1975.

(54) INWX 3 — D. Lloyd, M. Galland, P. Kirstein, "Aim and Objectives of Internetwork Experiments," IFIP W. G. 61 Experiment Note 3, January 1975.

3. ネットワーク・アーキテクチャ

3.1　コンピュータ・アーキテクチャからネットワーク・アーキテクチャ

オンライン分野における通信システムの"アーキテクチャ"がコンピュータ・メーカ各社より相次いで発表されている。各社独自のものであるが，ようやく通信分野にも"アーキテクチャ"が確立されようとしている。１９６０年代のコンピュータ・アーキテクチャ確立時に匹適する動きである。

3.1.1　コンピュータ・アーキテクチャ

１９６４年にＩＢＭ社の発表したシステム／３６０以来，コンピュータは第３世代に入ったといわれている。第３世代になってから，コンピュータ・アーキテイクチャ（computer architecture）という言葉がしばしば使われるようになった。

アーキテクチャという言葉は「構造を持ったものを設計し組み立てるための技術または科学」という意味であり，「方式設計思想」としてとらえられる。コンピュータ・アーキテクチャは「コンピュータ・システムを構築し，そのシステムを動かすためのハードウェアおよびソフトウェアを設計する際の拠所となる論理仕様」である。

第３世代の最も大きな特徴は，システム的な考え方を徹底化したことであるといえよう。その基本思想は，ユーザの立場になってハードウェア，ソフトウェア，そして運用法などを総合的にもう一度みなおし，過去にとらわれずに，新しい設計思想で構成し直したことである。

第３世代のコンピュータの特徴は，パフォーマンスの広い範囲を同一アーキテクチャのいくつかの処理装置でカバーし，プログラムとデータの互換性を持たせている。これは，小規模処理装置から大規模処理装置に移行するとき，プログラムの変更を必要としないようにとのユーザへの配慮として生まれたことが，コンピュータ・メーカにとっても提供するソフトウェアや教育サービスなどが一元化できるので多くの利益をもたらしている。

また，第３世代以降のコンピュータでは，適用業務に対する汎用性がいろいろの点で配慮されている。以前のコンピュータでは，技術計算用と事務計算用の機種は明確に分かれていたが，今日ではそのようなことは少なく，１つの機種でその双方が効率よく処理できるように工夫されている。

3.1.2　ネットワーク・アーキテクチャ

１９６０年にコンピュータと遠隔地にある端末装置を結ぶ通信システムの概念が出現して以

来，数多くのオンライン・システムやＴＳＳなどのデータ通信システムが開発された。コンピュータを中心とするオンライン・システムは，いまだ十数年の歴史にもかかわらず，座席予約，銀行業務，行政関係，交通管制，販売管理，医療関係といった広範囲な分野で目覚ましい発展を遂げ，社会経済活動に大きな比重を占めてきている。

一方，オンライン・システムを設計という観点から見ると，第３世代のコンピュータ・アーキテクチャ確立時には，このような発展を予期してなかったことが実情であり，オンライン分野における統一的なアーキテクチャの確立はなおざりにしていたといえる。このため，これらのシステムは，個々のアプリケーションを満す専用システムとして設計されており，必ずしも統一的なアーキテクチャに沿って設計されたものではなかった。その結果，数多くの多種多様なテレプロセシング・アクセス・メソッド，データ伝送制御手順，そして端末装置が存在する状態となり，各メーカともになんらかの標準化や体系化を必要とされるに至った。このことが，業務拡大に対応した拡張に対して融通性を失い，かつユーザ側の大きな負荷を招いている。例えば，大規模なオンライン・システムの構築には新システムの導入に最低３年，新しい機能の追加でも１～２年の歳月を費やしているのが現状である。

このことは，オンライン，通信システムの分野において基本的な思想の転換と，新しい通信システムのアーキテクチャの確立をうながす動機となった。

一方，通信システムのアーキテクチャの確立の原動力（技術的背景）となっているのは，ＬＳＩ技術に支えられたマイクロ・コンピュータの発展と普及，そしてＩＳＯによるビット・オリエンテッドなハイレベル伝送制御手順ＨＤＬＣ（High Level Data Link Control）の標準化作業動向である。オンライン・システムを構成するのにマイクロ・プログラム制御プロセッサやミニコンピュータを通信制御プロセッサやフロント・エンド・プロセッサとして使用することがコスト的に現実的になり，通信制御機能をホスト・コンピュータ以外に垂直（縦）方向に分散することが可能となり，更に，マイクロ・コンピュータによるインテリジェント端末が出現し，ホスト・コンピュータの負荷の一部を端末側に分担させることが可能となった。

これとは別に，１９６８年に開発を開始したＡＲＰＡネットワークが成功したのをきっかけに各国の研究機関によるコンピュータ・ネットワークが多数出現した。コンピュータ・ネットワークは各々独立したホスト・コンピュータを相互に接続し，お互いにそのリソース（ハードウェア，ソフトウェア，そしてデータ・ベース）を共用し合う。即ち，同レベルのリソース（アプリケーション・プログラム－アプリケーション・プログラム，ＣＰＵ－ＣＰＵなど）間で水平（横）方向の機能分散を行なう，広域リソース・シェアリングを目指す動向である。

コンピュータ・ネットワークで採用しているパケット交換はデータ通信に適しており，各国のコモン・キャリヤ（公衆通信業者）が続々とパケット交換網を構築している。商用のパケッ

ト交換網としては，１９７１年にスペインの電電公社ＣＴＮＥ（Compania Telefonica National de Espana）が最初のサービスを開始した。アメリカのTelenet 社は全米６２の都市でサービスを提供する計画を立て，最初はボストン，シカゴ，ニューヨークなど７都市で実施することとし，１９７５年８月に営業を開始し，現在（１９７７年）は６０都市でサービスを行なっている。また，日本の電電公社は，１９７８年上期にデイジタル・データ交換網ＤＤＸ（Digital Data Exchange）サービスの開始を予定している。基本的な通信サービスとしては，回線交換サービス，パケット交換サービスのほかに，従来の特定通信回線サービスに相当する専用回線サービスも予定されている。加入者インタフェースとしては，回線交換方式にはＣＣＩＴＴ勧告X.21，また，パケット交換方式に対しては勧告X.25が採用されることになっている。

以上の動向に対処し，多様化するユーザのニーズに対応するため，各メーカ共に柔軟性のあるネットワーク・アーキテクチャ（network architecture）を発表している。ネットワーク・アーキテクチャはコンピュータ・アーキテクチャに対応させて「データ通信システムを構築するため，データ通信の機能を明確に分離し，階層構造にし，各階層間の通信規約（プロトコル）を定め，ハードウェアおよびソフトウェアを設計する際の拠所となる論理仕様」といえる。これは，過去において互換性の問題を単一アーキテクチャに基づいてファミリ体系により解決したことに比適する。

ネットワーク・アーキテクチャの効用は，単にコンピュータ・ネットワークの実現やオンライン・システムの通信系の設計を容易にするだけでなく，単一アーキテクチャに基づく製品群の相互接続を容易にすることにより，装置の方式寿命を大きくするとか，同一ハードウェアやソフトウェアを多くの分野で共用することにより経済性の達成などにも効果がある。

ネットワーク・アーキテクチャは，現在および将来に渡るテクノロジのコスト／パフォーマンスを考慮して，ソフトウェア，ハードウェア，あるいは両者でインプリメントされることになる。

3.2　集中処理ネットワークから分散処理ネットワークへ

データ通信システムのネットワーク形態は，図３－１に示すように大きく３つに分けることができる。これは，データ通信システムの発展形態でもある。集中処理ネットワークは，主として通信処理機能のインテリジェンスがホストのみに集中しているか，各所に分散しているかによって更に２つに分けられ，それぞれ別型の通信形態を使用している。しかし，分散型データ・ベース・ネットワークと分散型アプリケーション処理ネットワークは，２つの通信形態のどちらでも実現可能である。

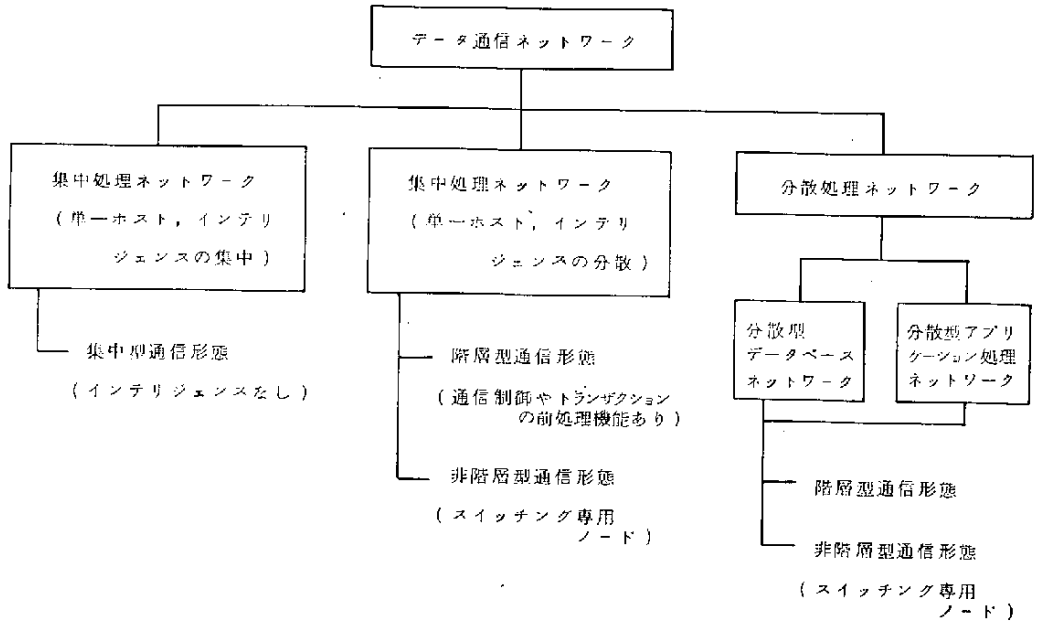


図３－１　データ通信システムのネットワーク形態

インテリジェンスがホストのみに集中した集中処理ネットワークは，１９６０年代末から１９７０年代初めに広く使用され，１９７０年代中頃には，主として通信処理機能のインテリジェンスを分散した集中処理ネットワークが出現した。１９７０年代末から１９８０年代初めには，真の分散処理ネットワーク － 分散型データ・ベース・ネットワークと分散型アプリケーション処理ネットワーク － へと移行するであろう。

集中処理とは，「すべてのデータ処理機能が，ネットワークに接続されている唯一のホスト・コンピュータ（大型機でも小型機でもよい）で実行される」ことを意味する。一方，分散処理とは，「ネットワークに接続されている２台以上の（必要なデータ・ベースを記憶するための大容量記憶装置を具備している）ホスト・コンピュータでデータ処理機能が分担される」ことを意味する。ネットワークに接続されているホスト・コンピュータは，同一ビル内にあってもいいし，地理的に離れていてもよい。

3.2.1　集中処理ネットワーク

集中型通信形態はネットワークの型であり，各々の端末は単一ホスト・コンピュータにポイント・ツー・ポイントあるいはマルチポイント回線で，あるいはフロント・エンド・プロセッサ（ＦＥＰ：Front－End Processor）を通して直結されている（図３－２参照）。

階層（ツリー状）型通信形態では，各々の端末はマルチプレキサに接続し，マルチプレキサはコンセントレータに接続し，そしてコンセントレータはＦＥＰに接続し，ＦＥＰはホスト・

コンピュータに接続している（図３－３参照）。この形態では，端末からのメッセージはマルチプレキサやコンセントレータのいくつかの層を通ってホスト・コンピュータへ送られる。

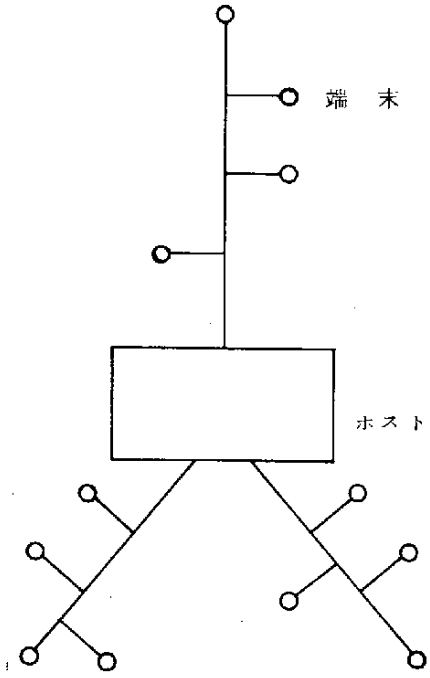


図３－２　集中型通信形態

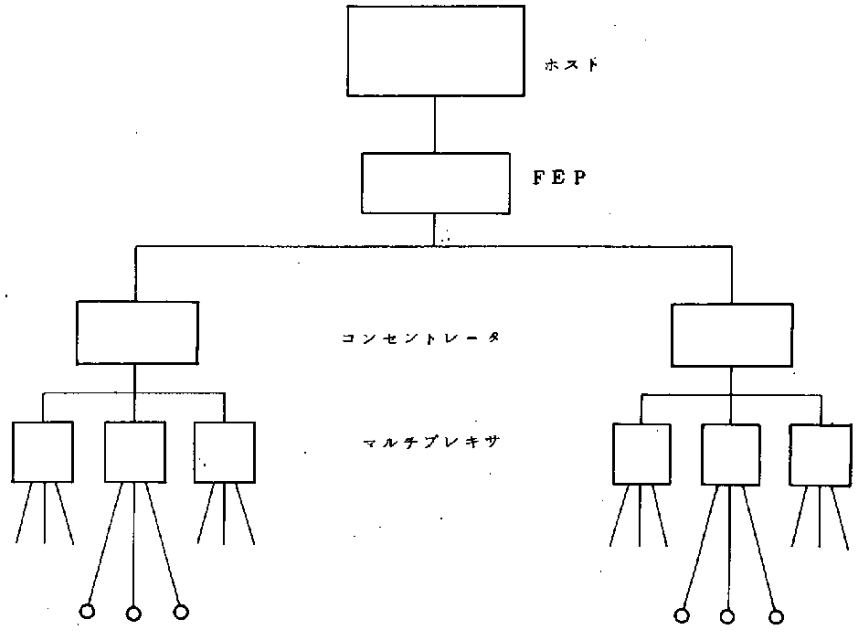


図３－３　階層型通信形態

非階層（網状）型通信形態では，スイッチング・ノードの各々は，ほぼ同じ能力をもっている。例えば，私設あるいは公衆パケット交換網はこの非階層構造である（図３－４参照）。こ

の形態では，スイッチング・ノードは少なくとも２つの他のスイッチング・ノードと接続されるように構成されるのが普通であり，通信網の信頼性が一段と向上する。

インテリジェンスが通信制御処理装置や端末にも分散されるようになると，例えば，インテリジェント即ち，プログラマブル・マルチプレキサを使用すると，集線効果により通信回線コストの削減が可能である。端末側にデータ処理機能のインテリジェンスを持たせると，入力データに対して範囲チェック，数字チェック，チェック・ディジットの正当チェックなどと，基本的なデータ・エディティング機能と，端末オペレータのために端末に促進メッセージを表示したり，端末のファンクション・セレクション・キー（Function selection key）をオペレータが押した場合，例えば，それに対応する伝票（invoice）をディスプレイ画面に表示することが可能となる。また，インテリジェンスを分散することにより，端末側ではトランザクションの前処理が行なわれるためレスポンス・タイムが短縮され，同時にホスト側の負荷が軽減されることになる。

3.2.2　分散処理ネットワーク

分散処理ネットワークでは，ネットワーク中に２台以上のホスト・コンピュータがあり，そこではアプリケーション処理と更新をともなうデータ・ベース・アクセシングが行なわれる。ここでは，ある特定の機能のみが分散された端末（クラスタ）もホスト・コンピュータとみなすことにする。分散型データ・ベース・ネットワークでは，更に，ホスト・コンピュータはオンライン用データ・ファイルを記憶するため補助記憶装置（ディスクなど）を持っている。

分散処理ネットワークは，数台の大規模コンピュータとたくさんのミニ・コンピュータから構成される。ホスト・コンピュータと補助記憶装置の規模は，ネットワーク中の個々の設置場所，即ち，ノードに必要なデータ処理能力とデータ・ベース能力に依存する。

大規模な通信を主体にしたネットワークのユーザは，今日では，大規模な集中型コンピュータ・コンプレックスでは充分なデータ処理能力を供給することができなくなっており，必要十分な速いレスポンスや大量データへの容易なアクセスが実現できなくなってきていると考えている。１台の大規模なホスト・コンピュータで充分なデータ処理能力がある場合でも，個々のデータ処理能力が少いミニ・コンピュータを処理ノードとして多数接続するほうが経済的になってきている。ミニ・コンピュータはパフォーマンスが向上し，より安価になってきており，ネットワーク中の通信ノードとして，あるいはアプリケーションやデータ・ベースに対するデータ処理構成要素として採用するようになってきている。これにより，ネットワーク全体のそれぞれのレベルにデータ処理能力とデータ・ベース能力を分散することが可能になる。

分散処理への移行は，エンド・ユーザの要求を迅速に満たし，ミニ・コンピュータを主体と

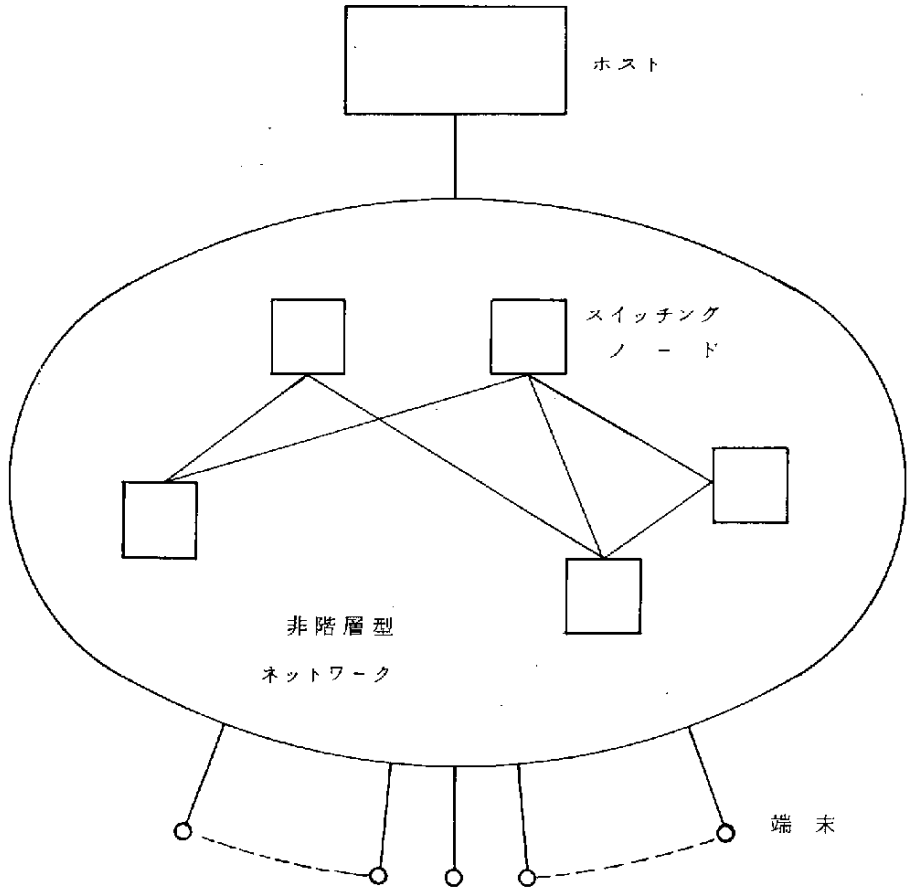


図３－４　非階層型通信形態

したシステムと分散処理を可能とする端末を採用することによるコストの削減やパフォーマンスの改善と共にエンド・ユーザに対するデータ・ベースへのアクセスを容易にすることを可能にする。

分散処理は，従来のインテリジェンスを集中した通信形態のネットワークでは１台のホスト・コンピュータ以外にはどこにも機能が分散された場所がないので実現できない。

なお，分散処理は，階層型でも非階層型通信形態でも実現できる。階層型分散処理ネットワークは垂直分散処理ネットワーク，非階層型分散処理ネットワークは水平分散処理ネットワークとも呼ばれる。

図３－５は３台の大規模なホスト・コンピュータを高速データ・リンクで接続した複合階層型ネットワークの例である。この種のネットワークはいくつかのノードのレベルで構成される。階層の最上位にある処理ノードは強力なデータ処理能力を持つ大規模なホスト・コンピュータである。最下位（即ち端末）は通常，限定されたデータ処理能力を持つか，全然持たないかである。中位ノードはＲＪＥ（Remote Job Entry）ワーク・ステーションであり，かなりのデー

タ処理能力とデータ・ベース能力を持ち，ローカルやリモートのオペレータ端末をサポートする。

分散処理ネットワークでは，簡単なトランザクションは端末あるいは端末コントローラで処理される。更に処理を必要とするトランザクションはより上位のノードに渡される。これにより，通信は，端末と実際のデータ処理やデータ・ベース・アクセシングを行なう地点の間のみに限定される。集中処理と比べ，通信回線のコストを削減し，オーバ・ヘッドを軽減し，レスポンスを短縮する。更に，ホスト・コンピュータの通信オーバ・ヘッドが軽減し，且つネットワークのより低いレベルで負荷が分担，処理されることによりホスト・コンピュータの処理負荷が軽減する。

階層型ネットワーク構成は，ユーザの会社組織の形態と非常によく一致する。それ故に，階

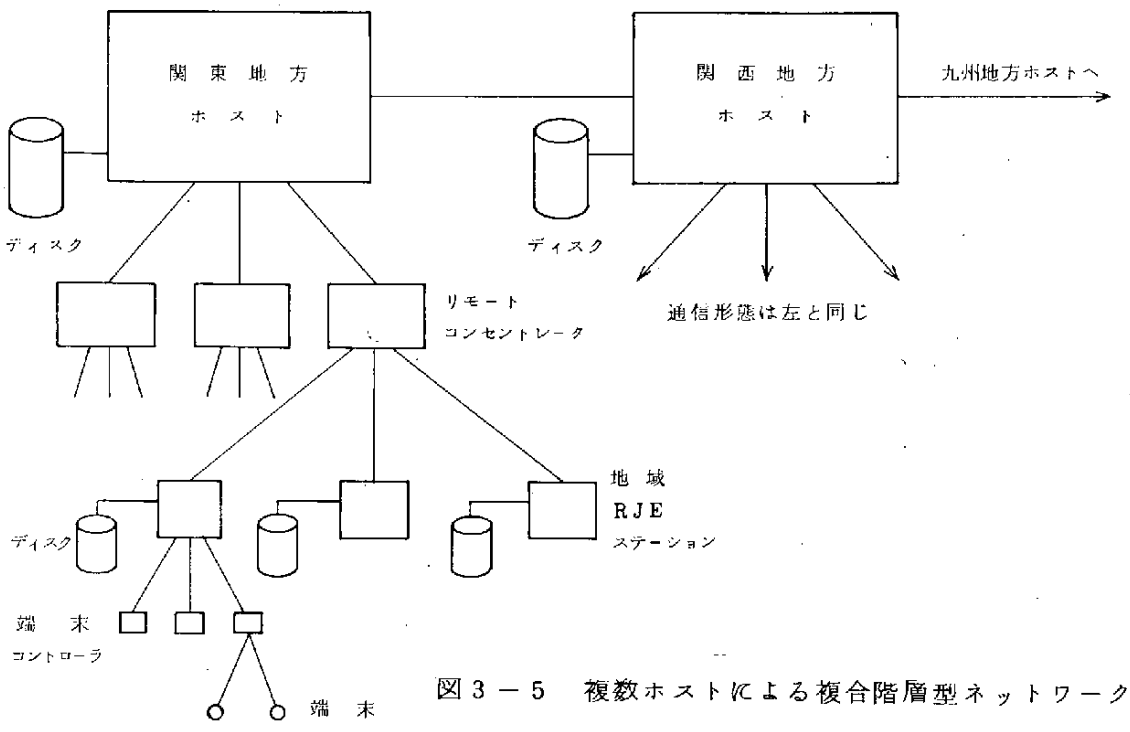


図３－５　複数ホストによる複合階層型ネットワーク

層型分散処理ネットワークは，企業内オペレーションに適している。

階層型ネットワークの１つの欠点は，回線や装置が故障した場合，オペレーションが続行できなくなることである。例えば，ＲＪＥワーク・ステーションがサービスを中止した場合，端末はそれより上位のノードと交信できなくなる。しかし，その端末が接続している端末コントローラのみで実行されるある特定の作業は続行することが可能である。ネットワークの利用可能性（availability）を向上させるためには，冗長な回線，バック・アップ用のダイヤル施設，そしてある特定のノードには冗長な装置などを用意する。しかし，余分なコストがかかる。

ネットワーク利用可能性を増加させる1つの方法は，非階層型あるいは多重接続型の通信ネットワーク形態を使用する。例えば，公衆パケット交換を使用する。この型のネットワークを図3－6に示す。ホスト・コンピュータと大容量メモリが各々のノードに設置される。非階層型ネットワークでは，各々のノードは少なくとも2つの他のノードと接続している。ノードの多重接続により，通常のオペレーション時に，メッセージの各種ルーティングだけでなく，回線やノードの故障に備えて代替経路を組み込んでおくことも可能である。

分散型データ・ベースや分散型アプリケーション処理にこの多重接続型ネットワーク形態を使用すると，各々のノードに同等のデータ処理能力や記憶容量を持たせれば，ほぼ同じような機能を分担させることが可能となる。そのような形態は，企業間ネットワークの構築に適している。この型ネットワークはデータ・ベースの共用に特に適しており，今後十年間に急速に発展することが期待される。

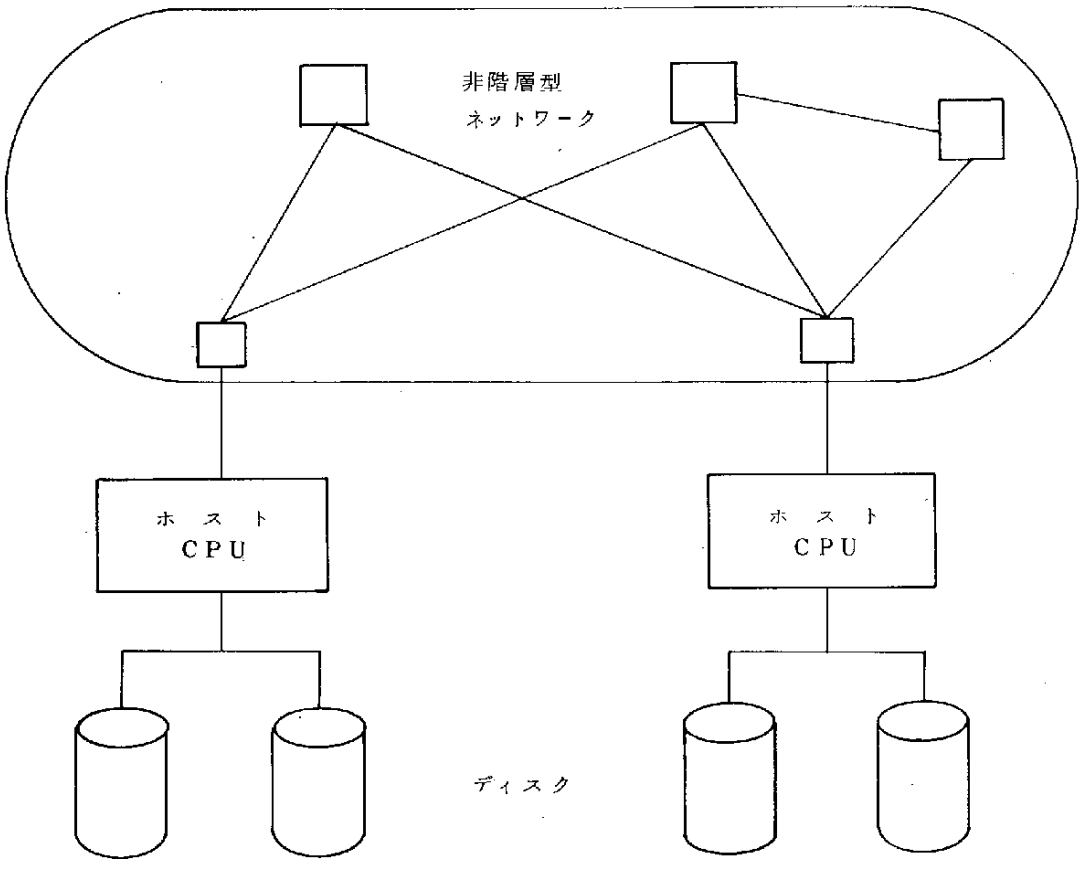


図3－6　非階層型分散処理ネットワーク

3.3 既存のデータ通信システムの問題点

既存のデータ通信システムの問題点を以下に箇条書で示す。

- 端末の種類，伝送制御手順に依存して伝送制御プログラムを作成しなければならない。
- アプリケーション・プログラムを作成するプログラマは，端末や回線の特性や制御方法を詳しく知らないとプログラムを作成することができない。
- アプリケーション・プログラムは通信回線網の変更により影響を受ける。
- 個々のアプリケーション・プログラムで回線や端末装置などの障害に対して対策（再送，迂回，回復など）を講じなければならない。
- アプリケーション処理機能，メッセージ処理機能，回線制御機能などが明確に分離されていない（図３－７参照）。
- システムの柔軟性と拡張性を欠く。
- センター側に機能および負荷，制御および管理が集中している。
- システムの信頼性が低い。
- コンピュータ間の接続を行なう場合，一方をコンピュータとし，他方は端末として扱わなければならず（主従関係を強いる），対等の通信が実現できない。
- ネットワーク資源の共用が不可可能である。
  －異種端末のマルチ・ポイントができない。
  －アプリケーション・プログラムおよび端末の共用ができない。

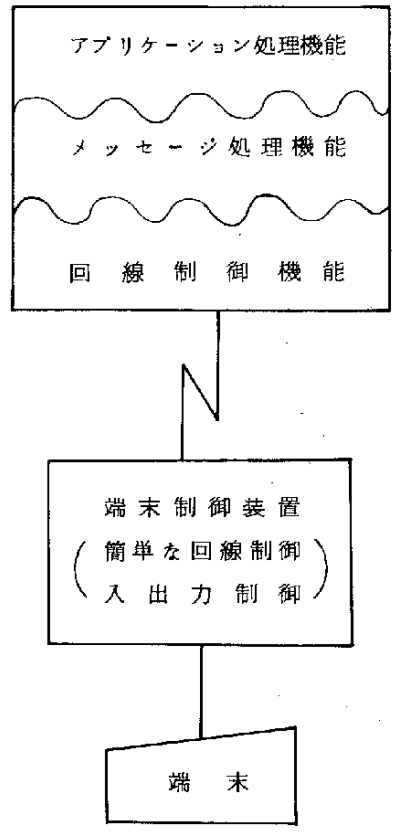


図３－７　機能の未分化

- 互換性のないデータ伝送制御手順が多数存在する。
- 全二重通信を実現しにくい。
- セレクト・ホールド（select hold）方式であり，マルチ・ポイントの能率低下が著しい。
- 伝送制御機能の中に機器制御機能の一部が同居している。
- バイナリ・データの伝送には特別な配慮が必要である。

・　伝送制御の信頼性が低い。

以上の問題点を解決するために，データ通信システムの標準化と統一化を目指すのがネットワーク・アーキテクチャである。

3.4　ネットワーク・アーキテクチャの概要

3.4.1　ネットワーク・アーキテクチャの目標

ネットワーク・アーキテクチャは，基本的には以下に示す目標で体系化される。これは，ネットワーク・アーキテクチャの利点でもある。

(1)　ホスト・コンピュータの負荷の軽減

ホスト・コンピュータの負荷の軽減は，専用のコミュニケーション・プロセッサによる通信処理機能の負荷分散，インテリジェント端末などによるアプリケーション処理をローカルで行なう（データ処理の負荷分散）ことによって実現する。

(2)　端末装置の使用可能度の向上

端末装置側にインテリジェンスを持たすことにより，データ処理機能の分散が可能となり，センタ側がダウンしても，ローカルに独自のオペレーションの続行を可能とする。

(3)　回線コストの削減（トラフィック量の軽減）

インテリジェンス端末装置は，オペレータが入力したデータを要約（summarize）や圧縮（Compression）をしてセンタ側に送ることが可能である。従って，1本の通信回線により多くの端末装置をマルチ・ポイント接続したり，今まで必要とされた回線速度よりも低速の通信回線を選択することが可能になる。

(4)　応答時間の短縮，システム・スループットの向上

インテリジェント端末装置は，オペレータが入力したデータを端末側のみで処理し，結果をディスプレイ装置などに出力することができる。即ち，通信回線を介してセンタ側へデータを送り，センタ側で処理し，その結果を受けとるに要する時間が不要となる。従って，応答時間が短縮され，同時にセンタ側の負荷が軽減されることによりシステム全体としてのスループットが向上する。

(5)　システムの設計・導入・保守の容易性の向上

ネットワータ・アーキテクチャでは，機能を明確に分離し，アプリケーション機能以外はメーカの標準プロダクトとして提供される。従って，ユーザはこれまでのように装置制御／回線制御プログラムの開発に煩わされることがなくなり，アプリケーション・プログラムのみの開発に専念することができる。また，端末装置が異なっても，同一のアクセス・マクロによりアクセスでき，回線網構成も意識する必要がない。機能の共通化および機能

の境界の明確な分離が行なわれており，端末装置が異なっても，同一の導入手法および保守手順の適用範囲が拡大される。

(6) システムの柔軟性・拡張性の向上

各機能の境界を明確に分離し，同一機能相互間を共通化して開発することにより，ネットワークの変更（回線網構成の変更や端末装置の追加・変更など）がアプリケーション・プログラムにほとんど影響を与えない。また，ある機能層の変更が他の機能層に影響を与えないため，システムの柔軟性および拡張性が実現される。

(7) システム内の諸資源の有効利用

ネットワーク・アーキテクチャでは，システム内の諸資源が有効に利用・共用される。例えば，端末装置からは任意のアプリケーションをアクセスできる。アプリケーション・プログラムは，複数の端末装置を同時に使用可能である。伝送速度さえ同じであれば，どんな端末装置でも1つの回線にマルチ・ポイント接続が可能である。マルチ・ポイントにおいて，端末Aに情報を送信しながら，端末Bから情報を受信（全二重のオペレーション）することが可能である。

3.4.2 ネットワーク・アーキテクチャの具体化

ネットワーク・アーキテクチャは，前項の目標を次に示す方法で具体化する。

① ネットワークに対する論理的モデルを設定する（論理構造によるネットワークの表現）。

② 通信機能とデータ処理機能を階層化する（階層構造）。

③ 同一レベルの機能層相互間のプロトコル（protocol）と上下の隣接層相互間のインタフェースを規定する（プロトコルとインタフェース）。

④ 多種・多様な端末装置の差異を吸収し，端末の仮想化を行なう（仮想端末）。

⑤ ユーザ・インタフェースを標準化する。

⑥ 標準化されたハイレベル伝送制御手順を採用する。

ネットワーク・アーキテクチャは，本来，物理的な実現手段とは切離された論理的な概念であり，規定である。しかし，実際には物理的な実現のし易さがネットワーク・アーキテクチャを定める際の重要な要素になる。従って，既に発表されている各社のネットワーク・アーキテクチャは，比較的限定されたシステムの用途や構成，あるいはそれを実現する手段との対応を意識したものが多いように見うけられる。

A. 機能の階層構造

ネットワーク・アーキテクチャでは，データ通信に必要な通信機能を明確化し，機能の

重複を避け，適切に分割し，機能を階層化する。各階層のプロトコルを規定する場合，各階層のプロトコルの相互の関連を考慮して，情報の重複を避けること，共通に使用する情報の引継ぎを容易にすること，矛盾を生じないことなどの配慮が大切である。

階層構造は，複雑・大規模なシステムを設計する場合の常套手段である。階層構造で作成されたシステムの特徴は，

① 階層レベルごとに閉じているので，作成しやすく，信頼性の向上が期待できる。

② 階層構成が良く考え決められたアーキテクチャであれば，個々のレベルの作り直しや改良が簡単であり，柔軟性・拡張性の向上が期待できる。

③ あるレベルの変更が他のレベルに与える影響を最少にし，自レベルより下位レベルにある機能層をブラック・ボックスとして扱える。

④ 今後のシステムに要求される高性能化に対して，ますます低価格化するハードウェアの使用を（各々のレベルに除々に採用できるので）容易にできる。

などである。一方，これに対して，

① きれいに階層化するのである程度の分割損，つまり分割に伴うオーバヘッドが生じて，特に性能の追求という面からは問題がある。

② 現在すでにあるソフトウェアは，アプリケーションにディペンドして機能をごちゃごちゃに作ってあることが多いので，新しい階層構造で作成されたシステムとの共存が困難である。

などの欠点がある。

現在，分割損は，素子技術，回路技術の進歩によるハードウェア・コストの低減あるいはマイクロ・プロセッサ，ＬＳＩ技術によって打破できる状況になってきている。

機能層の機能分担および負荷分担と，それらに対するプロトコルおよびインタフェースの決定は，ネットワーク全体の処理効率，信頼性，拡張性などに著しく影響し，ネットワーク・アーキテクチャ設計上の重要なポイントとなる。

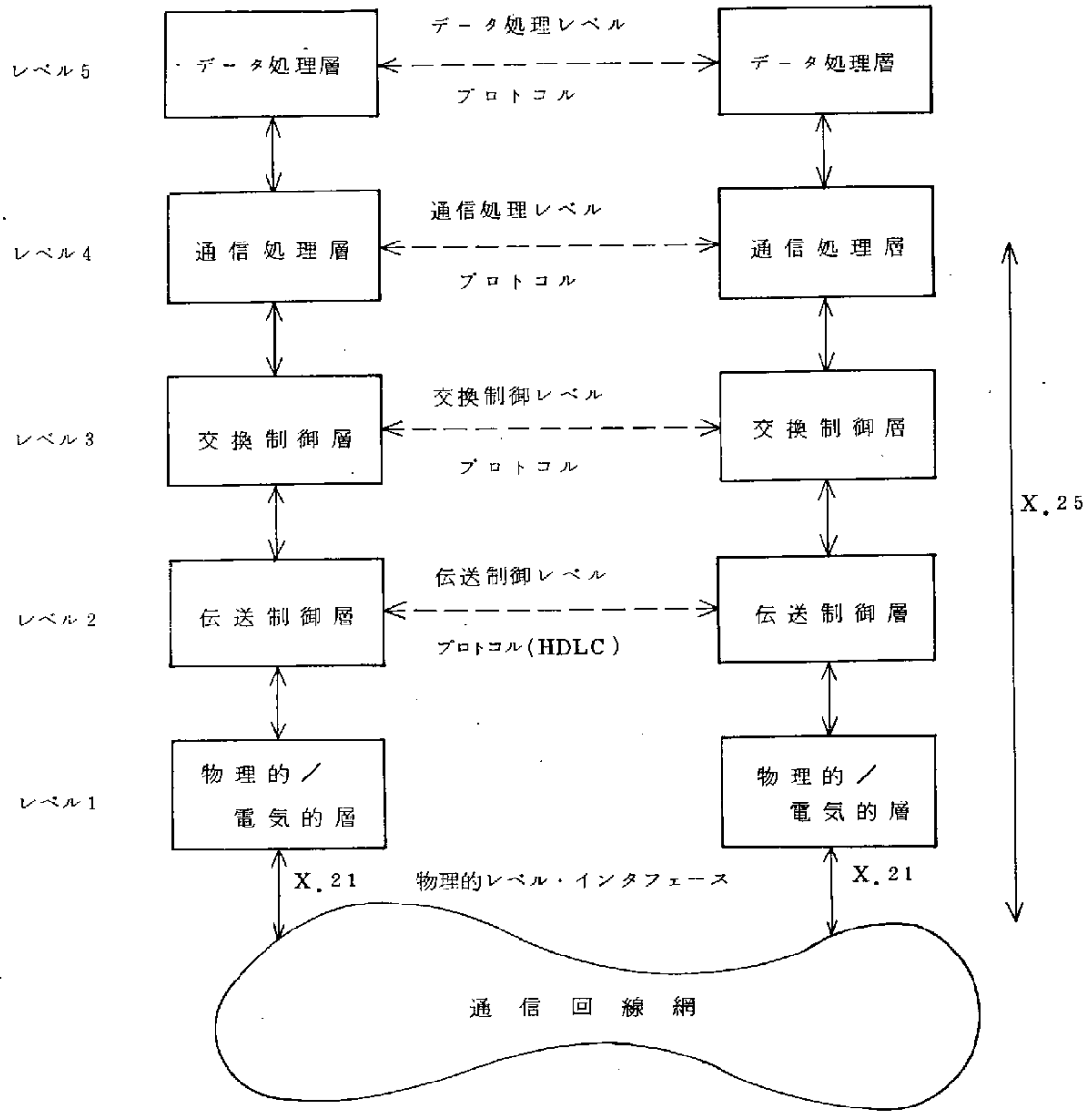
データ通信システムは，図３－８に示すように５レベルの機能層に分けることができる。

(1) 物理的／電気的層

物理的／電気的インタフェースは，同期方式（同期式／調歩式），速度クラス，コネクタのピン数と形状，信号線の種類，電気的特性を規定するものである。通常，ＣＣＩＴＴの勧告Ｘ.２１がディジタル網の，Ｖ.２４がアナログ網（Ｖシリーズ・モデム）（ＥＩＡ Electonic Industries Association（United States Standard）ＲＳ－２３２－Ｃに相当）のインタフェースとして使用される。

(2) 伝送制御層

伝送制御層（TC：Transmission Control layer）は，フィジカル・リンク（通信回線など）で結ばれた隣接ノード間のデータ転送を保障する機能（データ・リンクの確立／切断，送信権の制御，



←——→ ： 論理パス（プロトコル）

←————→ ： 物理パス（インタフェース）

図3－8 ネットワーク・アーキテクチャの階層構造
（プロトコル階層）

データ伝送制御，伝送誤り制御（順序制御を含む）など）を持ち，物理的なノード間の接続方式を吸収する。伝送制御レベルのプロトコルとしては，今後はＩＳＯのＨＤＬＣ（High Level Data Link Control)が一般に使用されるようになるが，既存の端末をサポートする場合は従来のベーシック手順（ＢＳＣ（Binary Synchronous Communications）など）も併用される。ＨＤＬＣでは，コンピュータ（コミュニケーション・ノード）とコンピュータ（コミュニケーション・ノード）間はバランス型手順クラスが，コンピュータ（コミュニケーション・ノード）と端末間はアンバランス型手順クラスがそれぞれ適用される。ＣＣＩＴＴのこのレベルのインターフェースとしては，ＬＡＰ（Link Access Procedure；ＨＤＬＣのシンメトリカル型手順クラスを使用）とＬＡＰＢ（Link Access Procedure Balance；ＨＤＬＣのバランス型手順クラスを使用）が併用される。電電公社の新データ網（ＤＤＸ）のパケット交換サービスでは，パケット形態端末の場合，ＣＣＩＴＴＸ．25ＬＡＰ Ｂ に準拠したバランス型手順クラス（ＨＤＬＣＢＡ）と，ＩＳＯ国際勧告に準拠した端末側（ＤＴＥ）を一次局（網側を二次局）とするアンバランス型手順クラスのＡＲＭ（Asynchronous Response Mode;HDLC－UAI）を，そして一般端末の場合，ＩＳＯに準拠した端末側（ＤＴＥ）を二次局（網側を一次局）とするアンバランス型手順クラスのＡＲＭ（ＨＤＬＣ－ＵＡ２）を採用する予定である。

(3) 交換制御層

交換制御層（ＳＣ：Switching Control layer）は，発信地ノードと目的地ノード間のデータ転送を保障する機能（エンド・ツー・エンド制御，ノード・ツー・ノード制御，データ伝送制御，伝送誤り制御（順序制御を含む），フロー制御，エンド・ツー・エンド・ルーティング制御，ノード・ツー・ノード・ルーティング制御）を持ち，物理的なネットワーク構成を吸収する。このレベルでは，更に，データ・リンクの物理特性（伝送速度，伝送誤り率，伝播遅延）とノードのバッファ容量を考慮してメッセージを最適な伝送長にするためセグメント化やブロック化が行なわれる。

(4) 通信処理層

通信処理層（ＣＰ：Communications Processing layer）は，アプリケーション・プログラムと端末，アプリケーション・プログラムとアプリケーション・プログラムなどの間でのデータ転送を保障する機能（プロセス間のロジカル・リンクの確立／切断，メッセージ・マルチプレキシング，送信権の制御，データ伝送制御，フロー制御，割込み制御など）を持ち，それらの間が全二重のポイント・ツー・ポイント仮想回線で接続された状態を実現する。付加価値サービスとしては，端末機器やアプリケーション・プログラムの

固有の特性に応じて，端末制御，コード変換，フォーマット変換，データ圧縮などが行なわれる。ネットワーク全体のリソースの監視および管理がこのレベルで行なわれる場合がある。このレベルのプロトコルは，End－to－End プロトコルとも呼ばれる。このレベルにほぼ相当するインタフェースとしては，ＣＣＩＴＴ の勧告X.25がある。ＩＳＯでは，このレベルのプロトコルを現在検討中である。

(5) データ処理層

データ処理層（ＤＰ：Data Processing layer）は，アプリケーション・オリエンテッドな機能を実行する部分であり，通常，ユーザ自身で作成する。タイム・シェアリング，リモード・ジョブ・エントリ，ファイル転送，データ・ベース・アクセスなどの汎用性のあるプロトコルは，メーカより提供される。データ処理層は更に階層化（機能レベルとユーザ・レベル）される場合がある。このレベルのプロトコルは，ユーザ・レベルのプロトコルとも呼ばれる。ＣＣＩＴＴでは，このレベルのプロトコルを現在検討中である。

図３－９には，ネットワークの論理構成を示す。図３－１０には，メッセージが発信地ノードから目的地ノードへどのようなルートおよび機能層を通って運ばれるかを示している。

図３－１１に示すように，伝送制御層の集合体により伝送サービス・ネットワーク（Transmission Service Network）を，交換制御層と下位層の集合体により交換サービス・ネットワーク（Switching Service Network)を，通信処理層と下位層の集合体

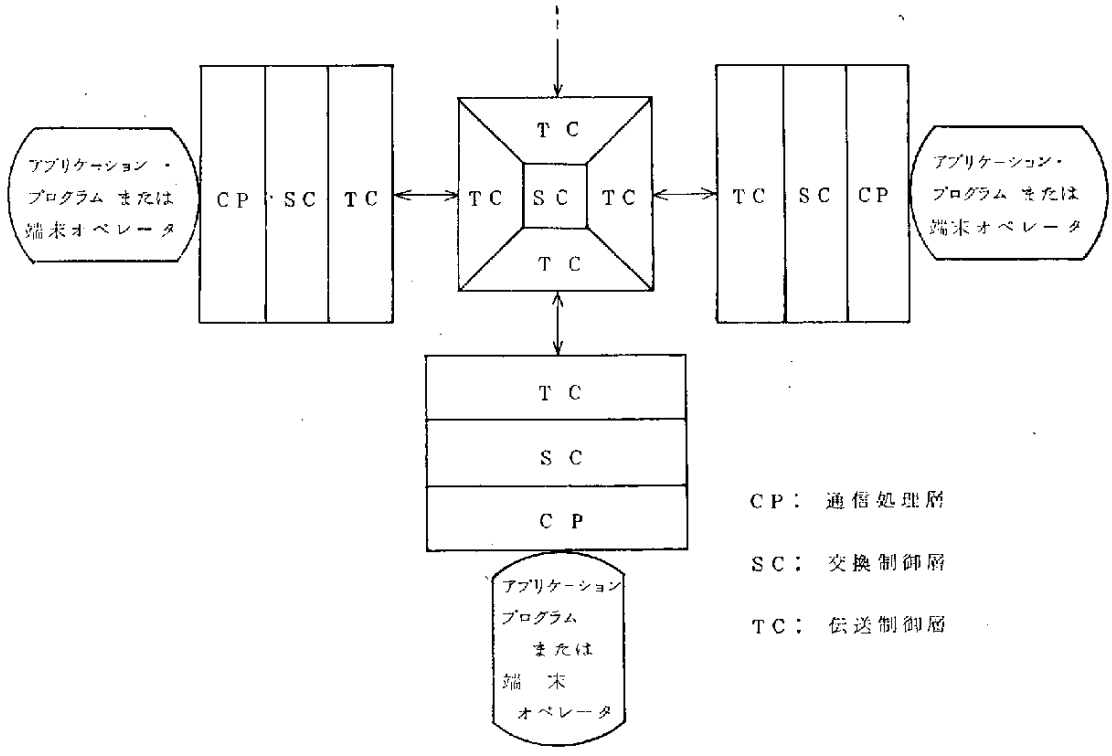


図３－９　ネットワークの論理構成

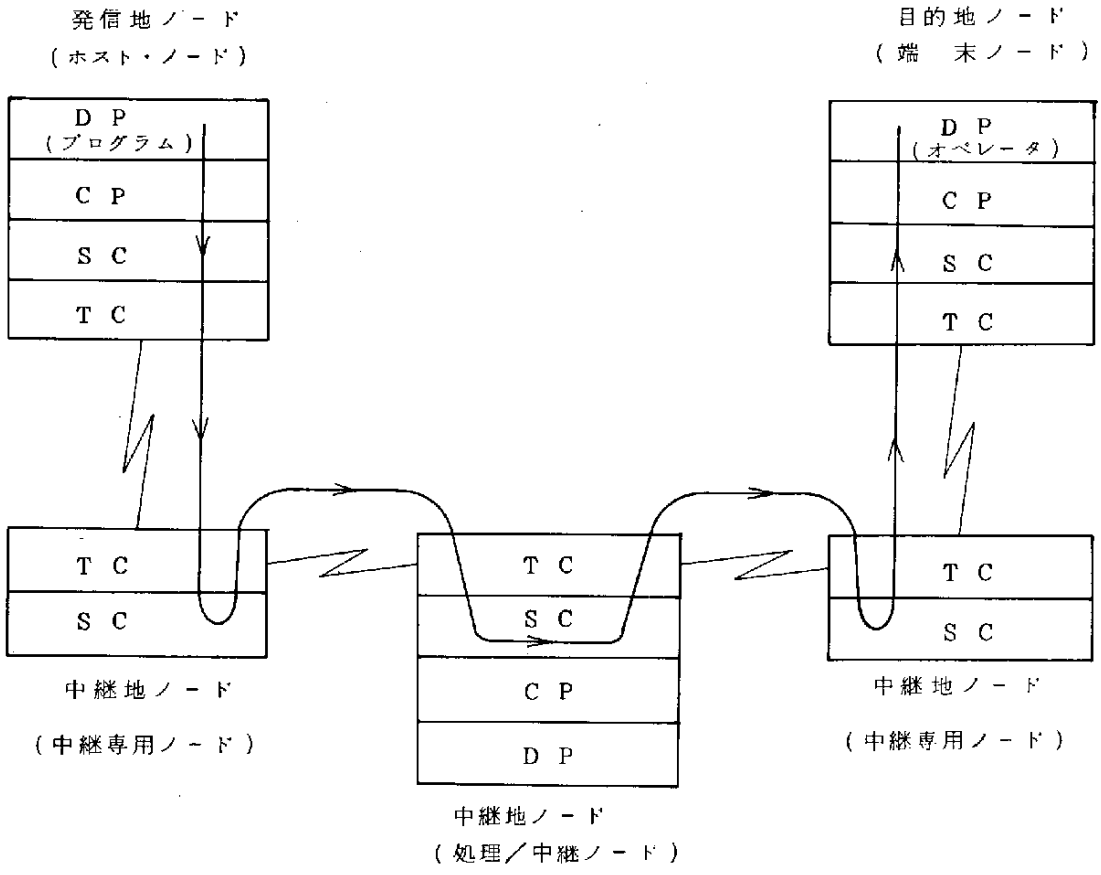


図3－10　　メッセージのルーティング

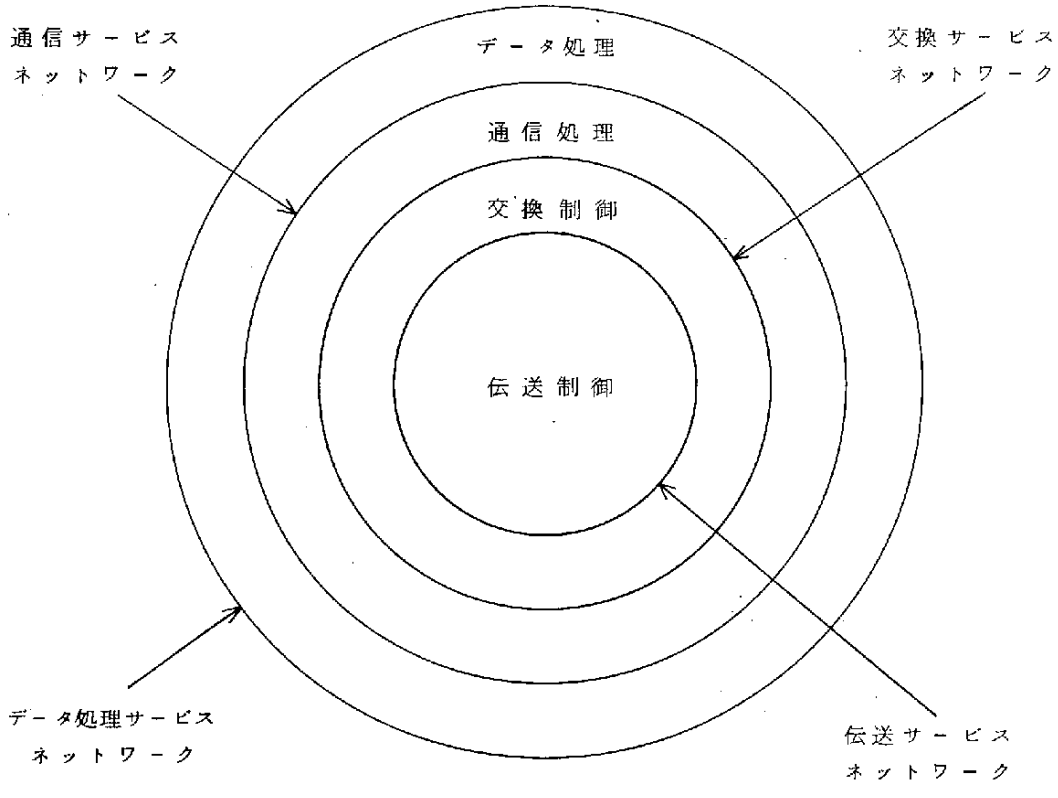


図3－11　ネットワーク・アーキテクチャによるサブネットワーク

により通信サービス・ネットワーク（Communication Service Network）を，そしてデータ処理層と下位層の集合体によりデータ処理サービス・ネットワーク（Data Processing Service Network）のサブネットワークをそれぞれ形成する。

B. プロトコルとインタフェース

プロトコルは同等レベルの機能層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。一方，インタフェースは，上下の隣接機能層間でのプロトコルと同様な約束ごとの集合である。

図３－１２に示すように，同等レベルの機能層は各々の別のプロセッサ上に存在するので，プロトコル情報は各々のレベルのヘッダに入れて運ばれる。一方，隣接機能層は同一のプロセッサ内に存在するので，インタフェース情報はレジスタやメモリを介してパラメータとして運ばれる。

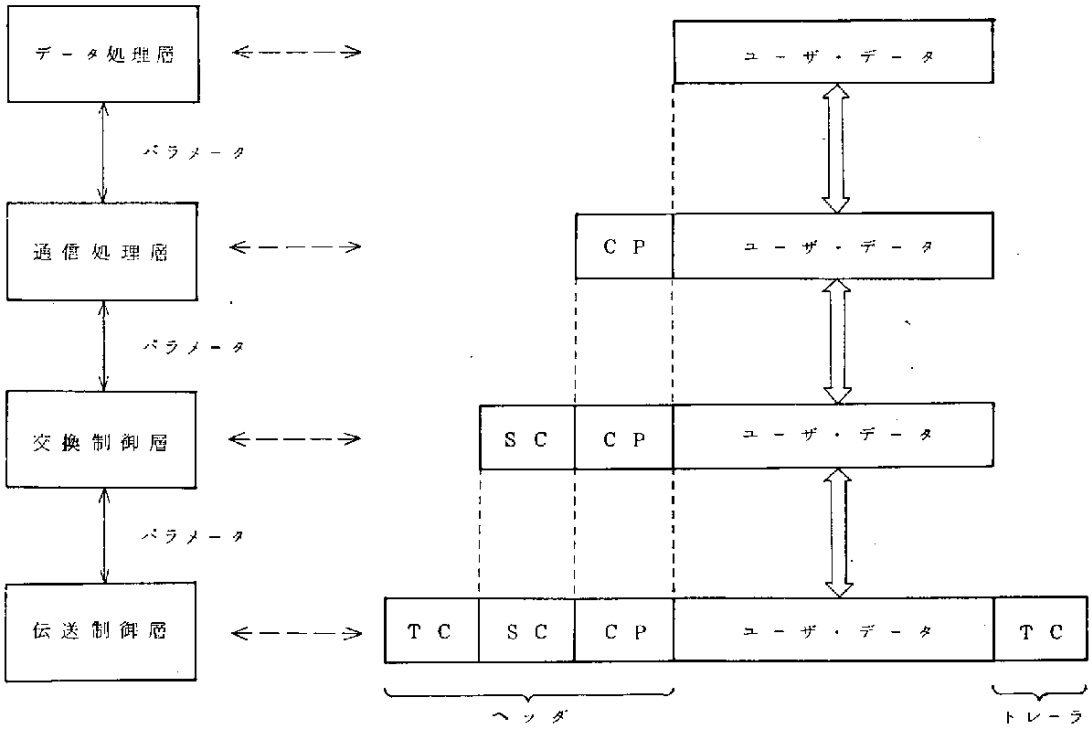


図３－１２　機能層間の制御情報の授受

C. 端末の仮想化（仮想端末）

端末の仮想化は，ネットワーク・アーキテクチャにおける重要な概念の１つである。既存の端末との差異をＦＥＰなどで吸収して，ＯＳ（またはアプリケーション・プログラム）まで影響を与えないようにすることは，ＯＳから見れば一種の仮想化を行なっていることになる。

コンピュータと端末とが通信をする場合に，コンピュータは端末の通信速度，通信形式（全二重／半二重），同期方式，通信回線種別（専用回線／交換回線），伝送制御手順，装置制御手順，文字コード，行幅，ページ長などの全ての特性を知っていて，その特性に合った制御をしなければならない。このことは，通信相手が端末でなく，コンピュータあるいはプロセッサの場合でも同様である。

多種・多様な端末の特性をいちいち意識せず画一的に扱うことを可能とするために，標準化（統一化）された仮空の特性を持つ端末，即ち，仮想端末（ＶＴ，Virtual Terminal）を規定し，仮想化を図る。コンピュータはこの仮想端末を対象にして通信を行ない，ネットワークの中のどこかで仮想端末と実端末（ＲＴ，Real Terminal）との変換を行なう（図３－１３参照）。この実／仮想変換は，①ホスト・コンピュータ，②ＦＥＰ，③端末制御機能を持つノード・プロセッサに機能分担させることができるが，端末に高度のインテリジェント機能をもたせ得る場合には，実／仮想変換を端末側にて行なわせることが望ましい。

コンピュータ間通信でも，お互いに相手を仮想端末と見なすことにより，仮想端末のままで通信を行なうことが可能である。

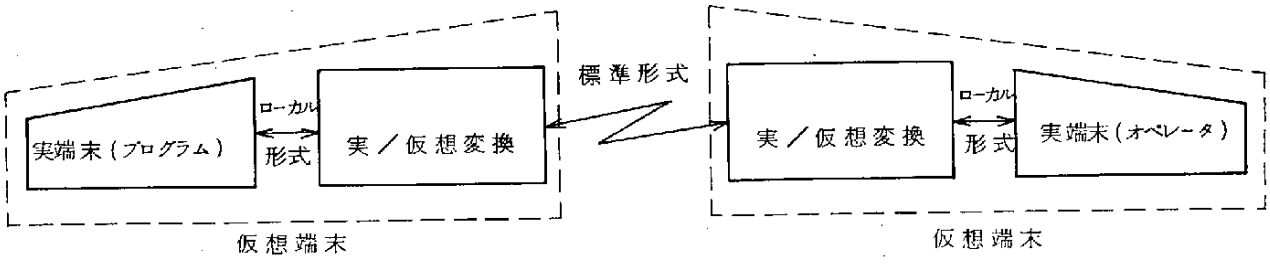


図３－１３　仮想端末の実現方法

3.5　ネットワーク・アーキテクチャの現状と動向

１９７４年９月にＩＢＭ社がシステム・ネットワーク体系（ＳＮＡ：Systems Network Architecture）を発表した。ＩＢＭ社のＳＮＡ発表に刺激されて，内外のメイン・フレーム・メーカだけでなく端末メーカ各社とも続々とネットワーク・アーキテクチャを発表している（表３－１参照）。

各社の発表内容は，ＩＢＭ社など数社を除いて，今後のデータ通信システムの開発構想であ

表3－1　各社のネットワーク・アーキテクチャの比較表

メーカー名	ネットワーク アーキテクチャの名称	発表時期	ホスト	ローカル CCP	リモート CCP	ディストリビューテッド・プロセッサ	DLC プロトコル	特　　徴　（セールス・ポイント）
IBM	SNA (Systems Network Architecture)	1974.9 1976.10 （拡　張）	S/370 シリーズ	3704 3705	3704 3705	3790	SDLC	・機能の分散　・通信制御およびネットワーク管理機能の独立 ・端末装置制御の標準化　・システム構成の融通性 ・処理の分散（複数システム，ネットワーキング機能）　・SDLCの採用
DEC	DNA (Digital Network Architecture)	1975. 4	PDP 11，15 シリーズ	PDP 11，15 シリーズ	PDP 11，15 シリーズ	—	DDCMP	・ファイル・シェアリング　・プログラム・シェアリング　・プログラム・データ・シェアリング　・回線特性からの独立　・ダウン・ライン・システム・ローディング ・ダウン・ライン・エグゼクティブ・ディレクティブ
バロース	DNS (Decentralized Data Processing Network System)	1976. 6	B700 B800 シリーズ	DCP (B774 B776)	B775 B776	B776 B80	BDLC ベーシック	・コミュニケーションの分散　・処理の分散　・データベースの分散 ・システム全体の監査，機密保持対策　・ジェネレータ形式のソフトウェアによる柔軟性，拡張性　・使用する言語は全てハイレベル言語
ユニバック	DCA (Distributed Communication Architecture)	1976.11	ユニバック 1100 シリーズ	DCP	DCP	—	UDLC	・コミュニケーション・システムのブラック・ボックス化　・あらゆるネットワーク形態が可能　・処理の分散　・ネットワークの独立　・システムの信頼性（ARMIS） ・接続に対する柔軟性（異機種もサポート）
レイセオン	PNA (Processing Terminal Network Architecture)	1976.11	PTS 1200 シリーズ	—	—	—	RDLC BSC S/S	・分散処理ネットワーク　・ホスト・コンピュータ負荷の軽減 ・システム開発が容易　・キー・ステーションの分散配置　・コストの節減
東芝	ANSA (Advanced Network System Architecture)	1976.12	ACOS77 600～ 900	DN-380	RT145 RT165	RT145 RT165	HDLC ベーシック	・機能の分散　・処理の分散　・管理の分散　・HDLCの採用 ・CCITT X.25に準拠した方法を採用　・パケット交換技術の採用 ・データベースの分散
日電	DINA (Distributed Information Processing N.A.)	1976.12	ACOS77 200～ 900	N7291 N7294	N5740	N200 N100 N8200	HDLC	・処理の分散　・コミュニケーションの分散　・データベースの分散 ・HDLCの採用　・新データ網への接続（X.25インタフェース採用） ・バーチャル・ターミナル思想　・パケット交換技術の採用
電電公社	DCNA (Data Communication Network Architecture)	1977. 3	DIPS その他	（CCP）	（RCCP）	?	HDLC	・CCITT X.25に準拠した方式を採用　・ディジタル・データ交換網（DDX）を有効に利用するプロトコル　・ネットワーク・ユーティリティの実現 ・端末の仮想化機能　・日電，日立，富士通，沖のメーカー4社との共同研究
富士通／日立	MSNA (M-Series Network Architecture)	1977. 3	Mシリーズ	CCP	CCP	?	HDLC	・通信ネットワーク資源の共用　・変更の柔軟性，拡張性 ・通信制御およびネットワーク管理機能の独立　・機能の分散　・処理の分散 ・データベースの分散　・HDLC手順の採用
沖	DONA (Decentralized Open Network Architecture)	1977. 8	OKITAC-50シリーズ	OKITAC-50シリーズ	OKITAC-1090シリーズ	OKITAC-50シリーズ	HDLC	・柔軟性のあるネットワーク構築（DCNA，他社ネットワークとの接続） ・リソース・シェアリング・ネットワーク　・分散処理ネットワーク ・メッセージ交換ネットワーク　・ミニコン・ネットワークによる経済性の追求
ニクスドルフ	NCN (Nixdorf Communication Network)	1977. 8	8800 シリーズ	8800 シリーズ	—	—	HDLC SDLC BSC S/S	・機能の分離独立化　・資源の共用　・機能相互間の共通化，標準化 ・データベースの分散　・モジュール構造による変更の柔軟性，拡張性 ・共通の言語（ビジネスBASIC）
富士通	FNA (Fujitsu Network Architecture)	1977. 5	Mシリーズ F"5"/"8" シリーズ	CCP UCP	CCP UCP	F-V F-VoⅢ F-VoSⅢ	HDLC ベーシック	・リソース・シェアリング　・ネットワークのバーチャル化 ・高効率／高信頼性システム　・コンポーネントの充実　・ネットワーク資源の活用 ・回線網設計の柔軟性　・国際互換の推進　・セキュリティの配慮
三菱	MNA (Multi-share Network Architecture)	1977. 6	COSMO シリーズ	M6680	M6680	M80 M70	HDLC ベーシック	・マルチシェア　・分散処理，分散制御　・HDLC手順の採用 ・新データ網への接続（X.25インタフェースの採用） ・異種計算機，従来端末との接続
日立	HNA (Hitachi Network Architecture)	1977. 9	Mシリーズ	CCP	CCP	H-20 H-80	HDLC	・通信ネットワークのブラック・ボックス化　・新データ網への接続 ・ネットワーク資源の有効利用　・通信ネットワークの変更の柔軟性 ・負荷の分散　・リスクの分散　・運用操作性，保守性の向上

り，論理構造ならびに機能層ごとのプロトコルを詳細に規定した技術資料は皆無である。確固としたネットワーク・アーキテクチャが定まっていない状態である。一般的には，特に国産各社の場合，輪郭はともかくとして詳細かつ具体的なネットワーク・アーキテクチャは，ＩＳＯあるいはＣＣＩＴＴなどの標準化動向，ＩＢＭ社（ＳＮＡ）や電電公社（ＤＤＸ，ＤＣＮＡ）の動向，および，ここ１，２年の製品開発と並行して確立されていくと見るのが妥当であろう。

各社とも，既存のデータ通信システムの有する問題点の解消，今後実現してゆかなければならない分散処理ネットワークへの志向という観点でほぼ同一の方向を目指すものと考えてよい。しかし，各社がそれぞれネットワーク・アーキテクチャを発表しているが，その内容を比較すると，オンライン・システムの強化に重点を置くもの（ターミナル・ネットワーク），コンピュータ・ネットワークの構築に重点を置くもの，ネットワークの集中制御あるいは分散制御に重点を置くもの，通信機能の分散，負荷（処理）の分散，あるいはデータ・ベースの分散に重点を置くもの，公衆パケット交換網とのインタフェースに重点を置くもの，（他メーカの）異機種ホスト・コンピュータ間結合に重点を置くもの（オープン・ネットワーク），ユーザ・レベルのプロトコルをも規定しているもの等，それぞれ独自のセールス・ポイント（狙い）を打ち出している。

特にＩＢＭ社の狙いは，データ通信システムにおけるメイン・フレーム・メーカの分担を増大させることを意識して，極力ホスト・コンピュータおよび端末サイドの付加価値を大きくし，回線コストを削減することを配慮していることが注目される。ユーザにとっては，システムにかかる総合的なコストを節減する，いわゆる「トータル・コスト・セービング」の実現であり，運用コストの比率が，コンピュータ・メーカ側か，あるいは公衆通信業者側に多くなるかは，あまり重要ではない。

ユニバック社，東芝，日電などの各社は，ＣＣＩＴＴの勧告Ｘ.２５をふまえ，公衆パケット交換網とのインタフェースをサポートするなど，通信回線網の高度化を意識する傾向にある。公衆パケット交換網との接続において，通信網のインテリジェンスをどの程度利用しているか，単に料金の安い針金とみなして，機能が重複してるかなど，インテリジェント網との調和（整合）がどう図られているかは，評価する価値がある。

ユニバック社は，いかなる他社のネットワーク（異機種コンピュータ）とも接続可能であることを第一セールス・ポイントにしている。

日電，日立，富士通，沖の４社は，電電公社と共同でデータ通信網アーキテクチャ（DCNA：Data Communication Network Architecture）を開発すると発表した。

電電公社では，ディジタル・データ交換網（ＤＤＸ：Digital Data Exchange）サービスを本格的に利用してもらうには，これまでメーカでまちまちだったコンピュータ，端末と回線

網を接続する際のインタフェースやプロトコルを統一することが要請されてくるわけで，メーカが公社のＤＣＮＡに準拠したものを採用することにより，開発の二重投資が避けられるなど国家的・経済的観点からも効率的であるという考えである。ＤＣＮＡが実用化されれば，異機種コンピュータおよび端末間の通信が容易となるので，ユーザが情報やコンピュータを広域的かつ効率的に利用できることとなり，将来のデータ通信に関する利用の高度化とその普及に対する社会的要請に応えるものとしている。電電公社が投じた標準化への波紋がどう広がって行くか，今後の動向が注目される。

ネットワーク・アーキテクチャは，単に現実の問題点を解消するためのみではなく，今後出現するであろう（あるいは出現しつつある）複数のコンピュータを有機的に結合するコンピュータ・ネットワーク，さらにはリソースの共用を理想的なかたちで行ないうるコンピュータ・ユーティリティへのアプローチとしての役割りをはたすものである。

従って，ネットワーク・アーキテクチャは，通信処理と情報処理との接点にあたる技術であり，情報処理産業と公衆通信産業との接点として捕える必要がある。現在のようにコンピュータ・メーカ各社が，自社のシェア拡大のみを目的として，独自のネットワーク・アーキテクチャを競作する状態は決して好ましいことではなく，将来必ず問題となる全国的なコンピュータ・ネットワーク，あるいは国際ネットワークなどを構築する際に，大きな障壁となることも考えられる。各社がネットワーク・アーキテクチャを個々に作成する限り，他社のコンピュータ（異機種ホスト）の接続をも考慮しているといっても，いずれにせよ満足のいく相互接続（相互乗入れ）はできないものと考えざるを得ない。

最近にいたり，このような問題に関する国際的な論議が盛んに行なわれてきており，ＩＳＯあるいはＣＣＩＴＴなどの場を中心に国際的な標準化の方向に動いている。しかしながら，リング・レベルのデータ伝送制御手順を除いて，技術的に広範囲な内容を含んでおり，コンピュータ・メーカとコモン・キャリア双方の利害に関する問題も多く，完全な標準化には長い年月が必要であると考えられる。

今後，ネットワーク・アーキテクチャは，このような国際的な標準化および各国の国情に応じたコモン・キャリアとの整合を考慮しながら発展し，具体的なシステムの構築がなされていくであろう。

参 考 文 献

(1) 出羽、下田
コンピュータ・ネットワーク入門、
別冊コンピュートピア、１９７７年６月、pp.１３４

(2) 特集：分散処理指向、
データ通信、Vol.9、No.３、１９７７年３月号、pp.２８－７８

(3) 特集：分散処理指向Ⅱ
データ通信、Vol.9、No.６、１９７７年６月号、pp.２８－７３

(4) 特集：最近のネットワーク体系について私はこう考える、
情報科学、通巻１３３号、１９７７年３月号、pp.９－２４

(5) 特集：ネットワーク体系を考える、
情報科学、通巻１３４号、１９７７年５月号、pp.１４－７６

(6) 特集：ネットワーク・アーキテクチャの活用とその期待、
ビジネス・コミュニケーション、Vol.14、No.７、１９７７年７月号、pp.３１－７５

(7) 電子協国際動向専門委員会
分散システムに関する調査、
日本電子工業振興協会、５２－C－３２４、昭和５２年３月、pp.２５６

(8) 電子協技術動向専門委員会
分散処理計算機システムの動向、
日本電子工業振興協会、５２－C－３３４、昭和５２年３月、pp.２７８

(9) 情開国際情報ネットワーク委員会
国際情報ネットワークに関する調査研究報告書、
日本情報処理開発協会、５１－R－００８、昭和５２年５月、pp.４１０

(10) 小特集：コンピュータ・ネットワーク、
情報処理、Vol.16、No.７、July １９７５、pp.５９１－６５３

(11) 伊藤哲史
HOST－HOSTプロトコルおよびハイレベル・プロトコルの動向、
情報処理、Vol.18、No.９、Sept.１９７７、pp.９２５－９３２

(12) 鍛治勝三
ＪＩＰＮＥＴのＮＣＰとロジカル・リンク、
日経エレクトロニクス、No.１３６、１９７６年６月１４日号、pp.７６－９５

(13) 松崎稔
出揃い始めたコンピュータ各社の通信ネットワーク構想を探る、
日経エレクトロニクス、No.１５７、１９７７年４月４日号、pp.３８－５１

(14) ネットワーク・アーキテクチャの現状と問題点、

ビジネス・コミュニケーション、Vol．１４、№５、１９７７年５月号、pp.７１－７８

(15) フジ・テクノ・システム主催

コンピュータ・ネットワーク・アーキテクチャーとネットワークにかかる諸問題、

昭和５２年３月２９～３０日

(16) 日本経営科学協会主催

急転開しつつある分散処理システムの最新情報 ── コンピュータ・ネットワークからコンピュータ複合体まで ──、

昭和５２年４月１８～２１日、pp.５６

(17) 日本工業技術センター主催

コンピュータ・ネットワーク・システムに於けるアーキテクチャの動向とその実際、

昭和５２年５月１７～１８日、pp.８９

(18) 日本データ通信協会主催

近くサービスが予定されている新データ網サービス（デジタル交換網サービス）のすべて ── サービス機能とその使い方 ──、

昭和５２年５月２６日、pp.６６

(19) 日刊工業新聞社主催

ネットワーク・アーキテクチャーとプロトコル、

昭和５２年６月６～７日

(20) 日本経営科学協会主催

急転開しつつあるコンピュータ・ネットワークの最新情報、

昭和５２年９月１３～１４日、pp.４１

(21) Auerbach

What is Network Architecture ?、

Computer Decisions, Vol. 8 №６, June １９７６, pp. ２４－３３

(22) Booth, G.M.

Distributed Information Systems,

Proc. AFIPS NCC, June 1976, pp. 789—794

(23) Doll, D.R.

Relating Networks to Three Kinds of Distributed Function,

Data Communications, Vol.6, №３, Mar. 1977, pp. 37—42

(24) Ziegler, K. JR.

Distributed Data Base Where are You ? ──(A Tutorial),

IFIP Information Processing, Aug. 1977, pp. 113—115

(25) Taylor, F.E.

The Relative Merits of Distributed Computing Systems,

ACM International Computing Symposium, Apr. 1977, pp. 357—365

(26) Palmer, I. and Davenport
Distributed Data Bases,
Infotech Sate of the Art Report : On — line Data Bases 2 : Invited Papers, 1977, pp. 203—226

(27) Kimbleton, S.R. and Schneider, G. M.
Computer Communication Networks : Approaches, Objectives, and Performance Considerations,
ACM Computing Serveys, Vol. 7, No. 3, Sept. 1975, pp. 129—173

(28) Barber, D. L. A.
The Role and Nature of a Virtual Terminal,
ACM Computer Communication Review, Vol. 7, No. 3, July 1977, pp. 5—22

(29) Day, J. and Grossman, G. R.
An RJE Protocol for a Resource Sharing Network,
ACM Computer Communication Review, Vol. 7, No. 3, July 1977, pp. 77—88

(30) Martin, F. J. JR.
The Federal Communications Commission and Major Policy Matters Affecting Computer Communication,
Proc. AFIPS NCC, June 1976, pp. 467—476

(31) Rybczynski, A., Wessler, B., Despres, R., and Wedlake, J.
A New Communication Protocol for Accessing Data Networks — The International Packet — Mode Interface,
Proc. AFIPS NCC, June 1976, pp. 477—482

(32) EPSS Liaison Group
An Interactive Terminal Protocol,
INWG General Note #94, June 1975, pp. 61

(33) Dijkstra, E. W.
The Structure of the "THE" Multiprogramming System,
Communications of the ACM, Vol. 11, No. 5, May 1968, pp. 341—346

(34) Cerf, V., McKenzie, A., Scantlebury, R., and Zimmermann, H.
Proposal for an International End — to — End Protocol,
ACM Computer Communication Review, Vol. 6, No. 1, Jan. 1976, pp. 63—89
; also INWG General Note #96, Sept. 1975, pp. 29

(35) Pouzin, L.
The Communications Network Snarl,

Datamation, Dec. 1975, pp.70—72

(36) Sanders, R.W. and Cerf. V. G.
Compatibility on Chaos in Communications,
Datamation, Mar. 1976, pp. 50—55
(訳：データ通信ネットワーク・プロトコルの標準化とその問題点、コンピュートピア、1976年6月号、pp. 36—47)

(37) Folts, H. C. and Cotton, I. W.
Interfaces : New Standards Catch up with Technology,
Data Communications, Vol. 6, No.6, June 1977, pp. 31—40

(38) 小野、浦野、鈴木
標準パケット網プロトコルを用いたプロセス間通信、
情報処理学会コンピュータ・ネットワーク研究会資料12、1977年9月21日、pp. 10

(39) 石野福弥
パケット交換網の通信規約、
情報処理学会コンピュータ・ネットワーク講習会テキスト、昭和51年9月28日、pp. 25—33

(40) Twyver, D. A. and Rybczynski, A. M.
Datapac Subscriber Interfaces,
Proceedings of the Third International Conference on Computer Communication, Aug. 1976, pp. 143—149

4. ネットワーク・システムのインプリメンテーション

4.1 CYCLADESの例

4.1.1 CYCLADESの概要

A．歴史と目的

フランスにおいては，１９７０年以来ＣＲＩ（Comite consultatif pour la Recherche en Informatique ：情報処理の研究のための諮問委員会）で行なわれてきた研究によって，コンピュータ・ネットワークの重要性が立証された。

これをうけて，１９７２年の初頭より，ＤＩ（Delegation a Informatioque ：情報代表部）の指導のもとに開始されたのがCYCLADES網（Reseau CYCLADES）である。

CYCLADES プロジェクトは，軍，大蔵省，文部省，情報代表部，ＩＲＩＡ，Ｐ＆Ｔ（郵政省）より構成される理事会により監督されている。このメンバーの中で，情報代表部がプロジェクトの予算（１９７２年－１９７５年）である２０.８百万フランの８０％を用意し，軍が２０％を負担する。Ｐ＆Ｔは，電話回線およびモデムを無料で提供する。ＩＲＩＡが，このプロジェクトの実行機関としての役割をはたす。

CYCLADES の目的は，種々の新らしい分野での実験の道具となるプロトタイプのコンピュータ・ネットワークになることである。特に，つぎにあげる６つの分野が考えられている。

① データ・コミュニケーション
② コンピュータの相互会話
③ 協同ジョブの研究
④ 分散形データ・ベース
⑤ インタフェースおよび共通言語
⑥ リソースの共有

これらの目的が，次のような段階をへて実現されていった。

７３.９：３台のホストと１台のノードよりなるネットワークで３つの機能をデモンストレーション。

- オペレータ間通信
- ファイル転送
- リモート・ジョブ・エントリ

7 4.2 : 4台のホストと3ノードからなるパケット交換網（CIGALE）ができ上った。
CIGALEは，1日3時間ずつ稼動した。

7 4.6 : CIGALEが7ノードとなった。

7 4.8 : European Informatics Network（EIN）の一環として，CYCLADESとNPLネットワークが結合された。
この段階では，CYCLADES側からNPL の機能の利用は可能になったが，この逆はまだできていない。

7 4.11 カナダのトロントでTSS, リモートバッチ・ファイル転送，ローカル・ターミナル・アクセスのデモンストレーションを行なった。

7 5.1 : CIGALEが，1日6時間稼動になった。

7 5.2 : 予算の関係で，ノードのうち4台をターミナル・コンセントレータとして使用できるようにして，コンセントレータを新規に設置しないことにした。
IRIAにネットワーク・コントロール・センタ設置（Mitra－15使用）。
P&Tリサーチセンタ（Rennes）にネットワーク・メジャメント・センタを設置（Mitra－15使用）。

7 5.7 : この時点で使用可能な機能は次のとおりとなった。
CII，IRIS－80およびIBM360／67のTSS，リモート・バッチ，ファイル転送，ローカル・ターミナル・アクセスが可能。
CII 10070 のリモート・バッチ，ファイル転送，ローカル・ターミナル・アクセスが可能。

B．参加団体および規模

CYCLADESは，20の研究機関の参加を予定して作られた。このうち 13機関は，ネットワーク利用の研究を主眼とする研究機関および大学であり，残りの7機関は，ネットワーク利用が軌道に乗ってから以後開発を進めて行く行政局の管理機関である。

参加団体およびホスト・コンピュータの現状については，新らたな参加団体，参加をとりやめた団体，ホストのリプレースを行なったセンタがあり，正確にはつかめない。主なる参加団体は，表4－1に示すとうりである。

C．CYCLADESのもたらすもの

CYCLADESの開発にあたっては，官公庁，郵政省，軍，情報処理産業など，各種の分野にわたっての利益および期待がまとめられたので，これについて述べてみる。

(1) 行政機構

国の経済活動において，行政機構が負わされているサービスは数多く，このことから

由来する負担は，はなはだしいものがあり，増加の一途をたどっている。行政機構は，大量に情報を収集し，処理し，普及せしめるという重要な任務を負っている。行政機構は，それの所有している情報を共有化することによって生ずる利益について認識を新たにしてきた。

いくつかの計画がすでに試みられている。人間の身体（SAFARI）または道徳（SIRENE）に関する共通目録の作成，諸々の活動や生産物のためのカタログ，中央

表4－1　参加センタ

Institut de Recherche d Informatique et d'Automatique (IRIA) 2 center
Compagnie Internationale Pour I'Informatique (CII)
Meteorologie Nationale (METEO)
Institut de Recherche des Transports (IRT)
Compagnie Internationale pour I'Informatique Centre Scientifique de Grenoble (CII)
Universite de Grenoble (IMAG)
Centre Universitaire de Calcul de Lyon (CCILS)
Ecole des Mines de St—Etienne (MINES)
Centre d'Etudes et Recherches de Toulouse (CERT)
Universite de Toulouse (TOU)
Universite de Rennes (REN)
Centre National d'Etudes des Telecommunications (CNET)
Centre Commun d'Etudes de Te'le'communications et Television (CCETT)
Centre Electronique de I'Armement (CELAR)
Ecole Superieure d'Electricite'(ESE)

官公庁（税関，国税局，厚生省等）により計画された情報電送システムの設置などである。

CYCLADESは，これら一連の活動のうちの1つであり，次にあげるようなサービスの提供を行なっている。

(a) データの交換

いくつかの方式が研究され，次いで提案されることになっている。例えば，報告書，コピー等の転送，文献検索の質疑および文献抽出，他のネットワークに蓄えられた索引プログラムへの直接のアクセスなどである。

(b) プログラムの交換

行政サービスに関するいくつかのプログラムは共有することができる。それは，次の様な場合である。

- ネットワークを通じて，プログラムの普及と保守が必要な場合。その各サービスは適切な計算機の下で行なわれることになる。
- 遠隔地から，ネットワークを通してデータを送り処理結果を受けとるような処理が必要な時。

(c) リソースの共有

この領域こそ，CYCLADES が，行政の要請にもっともよく答え得る方式の選択ができる場合であり，もっともよく柔軟性を示す場でもある。これらリソースのうちわけは次のとおりである。

- コンピュータ
- 集中管理装置と回線
- 回　　線

結論として，CYCLADES プロジェクトは，行政に関して次のような利点を持っている。

- 短期的に見れば，自由な使用に供することによってデータ転送の道具となり得る。
- より長期的には，特定の外形の研究に関した行政に関する，一個または数個の情報ネットワークの概念から1つの方法論を引き出すことができる。これは，コンピュータ・ネットワーク（ことに，データまたはプログラムの信頼度に関する問題点を整理した上で標準化されたプロセデュアの創造のことであるが）の共同開発によって生じた問題点の研究と，一連のヴァリアントを持った実験によって可能になる。

CYCLADES の設置と並行して，共同作業グループである，システム・ダンフォルマシオン（Systemes d'Information）が，異なった研究センタ間のデータ・ベースへのアクセス，および情報の流動性の持つ問題点の研究を進めて行く。設置チームと一致協力しながら，その作業によって，ネットワークに関連した適応

の諸問題を予測し，解決して行くことができる。

(2) PetT

PetTの行政機構は，CYCLADES 網の受益者であるといえる。それは，PetTが，連結されているデータ・バンクへのアクセスの可能性の恩恵を蒙っているからであり，また，PetTはデータ伝送とデータ交信に関しては，パイロット的役割りを果しているからである。

(3) 軍　　隊

CYCLADES の実施によって得られる手腕や技術は必ずや軍関係のネットワークの概念に貢献することであろう。軍本部は，それ故大いにプロジェクトに興味を示すであろう。

(4) 情報処理産業

情報処理産業にとって，CYCLADES は，将来のシステムのハードウェアとソフトウェアの特性を決定するためには最良の場であるといえる。

CYCLADES は，コンピュータ・ネットワークの1つの系統の原型であるといえる。その系統の特徴は，並列処理，コミュニケーション，専用化，信頼度などによって示されることになるであろう。それ故，新しいアーキテクチャ，新しい使用モード，効率の高い伝送用の機材および技術が考案されねばならぬであろう。同時に，現在既に不足しており，その需要が将来急増すると考えられる規格品の開発もされねばならぬであろう。

コンピュータの異種性のために，CYCLADES は，互換性の無さという問題点の分析，または，それを是正する手段を必要とするようになるであろう。これらの作業は，同一でなければ少なくとも相互に，機械的に翻訳可能なコンピュータの構造の問題に行きつくであろう。

ネットワークの設置に積極的に参加しているC．I．IにとってCYCLADES は，長期的な目で見れば十分な利益となるものであろう。

1972年から1975年に，C．I．Iの勢力の及ぶ市場で，コンピュータ Mitral 15 の引渡し台数が制限されるのが事実だとしても，このようなシステムの概念の出現によって，Mitra 15 のカタログを改良し，補充する（ことに結合装置の性能の改良であるが）こともできるであろうし，また，機材の特権的状態によって，買い入れ側にしてみれば，開発を強く要求されている同種的下位ネットワークを構成することができるようになるであろう。

≪ 分散されたインテリジェンス ≫ の概念は，新しいシステムのアーキテクチャの基礎がためにもなる。このようにして，ソクラテス・プロジェクト（分散データ・バ

ンク）は，CYCLADES に関して自由使用可能な最初のソフトウェアの１つを生みだすことになるだろう。

建造者たちがコンピュータ・ネットワークの開発に専心している時に，CYCLADES は，パイロット的な実験を行ない，その成功は，その産物とＣ．Ｉ．Ｉ．の商標のイメージ，この両者の頭上に輝しい光を投げかけるであろう。

(5) サービス企業と情報処理会議

これらの企業は，将来における，コンピュータ・ネットワークの存在と直接のかかわりがある。しかも，CYCLADES の実現に一役買っているのであり，これらの企業は CYCLADES の実験と能力を大いに評価することになるであろう。というのも，これら企業は，オンライン処理システムを持つことを，コミュニケーションに関するソフトウェアの実現を，汎用あるいはアプリケーション・ソフトウェアの普及等を要求されているからである。

フランスの主要な企業の大部分は，CYCLADES の実現に参加するのに必要なあらゆる能力を持ったハイレベルの人材を備えている。フランスの市場に触手を延ばそうと持ちかまえているアメリカの企業によって，これら企業が脅かされないためには，ぜひともこの能力を開発する方向に向わなければならない。

(6) 大　学

短期的観点よりすれば，CYCLADES 網の実現に伴う主要な利益の１つは，大学のパイロット的研究センタの参加になるといえる。数多くの学生達が，ネットワーク建設に関する諸問題について，また，機材や実際に働いがいるソフトウェアの特性について，そこから様々な研究テーマを見い出すことができるからである。また，同時に，彼らは直接役に立つ専門的な経験をも重ねて行くのである。実験段階に入ると同時に，CYCLADES は，大学の研究センタあるいは研究所のコンピュータを結合することができるであろう。各研究センタは，このようにして，システムの非常に拡張された全範囲を，また，ランゲージや，データ・ベースを利用できることになる。研究者作業，学生の養成は，彼らが適当な道具を自由に使えるだけに益々，効果的なものになるであろう。

(7) 研　究

前述したごとく，コンピュータ・ネットワークの主要な刷新面は，その利用，ことに研究チーム自身のための利用にあるといえる。

ところが，研究をして行く上で，様々な障害につきあたることがしばしばである。例えば，地理的条件，学閥等々の障害である。これに反して，CYCLADES は，共通の

目的にそって作業を行ないながら，様々なチーム間で，情報を交換したり，対話をかわしたりすることができるのである。ネットワークが機能を開始すれば，ネットワークを通して，これらチームの構成員達は共に作業をしつづけることができるのである。

さらにいえば，情報処理関係の人々のみが関係者ではないのであって，コンピュータの使用者でありさえすれば，すべての学問の研究者達も，他の研究センタの同僚と情報を容易に交換できるという恩恵に浴することができるのである。

D．開発内容

CYCLADES において当初企画された作業のリストを以下に示す。

これらの作業は，７３／７４年に計画されていたものであり，それぞれ７３年初頭より開始された。

① Abonne Transiris－CYCLADES（CII－Grenoble）

期限 ： １９７３年１２月

CYCLADES ネットワークの利用者に対して，Transiris サービス（タイム・シェアリング，リモート・バッチ，……）へのアクセスを可能にするソフトウェア。次のプロトコルに関係する。

- ＳＴ／ＳＴ
- ユーザ／サーバ・コネクション
- バーチャル・デバイス

② Siris ７／８の下でのターミナル・コンセントレータ（CEA）

期限 ： １９７４年３月

ローカル端末が，ネットワークによって他のサービスを受けることを可能にするソフトウェアＣＴの実現。

③ Ｒ－ファイル Siris －７（CII－Grenoble）

期限 ： １９７３年１２月

リモート・ファイルに対するＶＤＡＭ アクセス法（ダイレクト・アクセス）の拡張。ネットワーク上に，ファイルが分散され得るＳＯＣＲＡＴＥ システムの直接の応用。

④ ＩＭＡＧのＩＢＭ ３６０／６７のＣＰサービスへのアクセス（IMAG－Grenoble）

期限 ： １９７４年４月

リモート端末からＣＰ／６７の利用を可能にするようなインタフェース・ソフトウェアの実現。次のプロトコルに関係する。

- ＳＴ／ＳＴ
- ユーザ／サーバ・コネクション

・ バーチャル・デバイス

⑤ IBM 360／67のOS／MVTのリモート・バッチ・サービスに対するアクセス（IMAG）

期限 ： 1974年5月

リモート端末からIBM 360／67においてジョブを実行できるようにするインタフェース・ソフトウェアの実現。

⑥ CPの下でのターミナル・コンセントレータ（IMAG－Grenoble）

期限 ： 1974年2月

ローカル・CP端末がネットワークを通じて他のサービスを受けることができる。

⑦ OS／MVTの下でのターミナル・コンセントレータ（IMAG）

期限 ： 1974年3月

ローカル端末が，バッチ・セッションの間に，ネットワークを通じて他のサービスを受けることができる。

⑧ タイム・シェアリングSMAへのアクセス（CERT－Toulouse）

期限 ： 1973年12月

CERTのCII 10070で実現されるソフトウェアで；リモート・ユーザがSAMシステムへのアクセスを可能にする。次のプロトコルに関係する。

・ ST＼ST

・ ユーザ／サーバ・コネクション

・ バーチャル・デバイス

⑨ STに対するリスタートの問題の研究（IMAG）

期限 ： 1974年10月

いろいろな場合の故障を調べ，STがリスタートできる水準を明確にすることを可能にする。

⑩ CYCLADESのドキュメンテーション・システムの設置（CCILS－Lyon）

期限 ： 1974年10月あるいは1975年

完全なドキュメンテーションに対して関係者が参照できるように，ネットワーク上に4ケ所，論文に関する4つのストックを設置する。

⑪ ネットワーク・インタープリターの研究および開発（FACULTE－Toulouse）

期限 ： 1974年9月

インタープリターの形で定義された1つの"ネットワーク・マシン"が実際のマシン上にすべて実現される。そして，実際の範囲内で利用するもの，オペレーティング・シ

ステムおよび利用手続きの異質性による諸問題を克服するためのものである。

⑫ ネットワーク上で分散されたプログラムの実行（CERS）

期限 ： １９７４年１月

いろいろなサイトに分散されたプログラムのアクティベイト，同期の問題の研究。

⑬ ＳＴの測定（FACULTE—Rennes）

期限 ： １９７４年６月

ＳＴの機能の評価を可能にする"スパイ・マシン"の設計と実現。"スパイ・マシン"による測定。測定結果の利用。

⑭ シミュレーション（FACULTE—Rennes）

期限 ： １９７４年６月

ST／10070 のシミュレーション。"動作中のネットワーク"レベルでのセンタのシミュレーション。CIGALEのシミュレーション。（最初のアプローチ）

⑮ ファイル転送のプロトコル（ESE—Rennes）

期限 ： １９７４年１２月

いろいろなサイトに共通を，ファイルの転送，蓄積および交換手続きの設計。

⑯ CYCLADESのターミナル・コンセントレータ

期限 ： １９７４年６月

Mitra—１５の下で，ＳＴ／ＳＴおよびPAV プロトコルを利用するターミナル・コンセントレータの仕様決定と実現。これらのコンセントレータは，Siris ７／８のRB，TSサービスならびにCP，OS／MVT，SAMのサービスにアクセスすることができる。

⑰ ネットワーク言語

ネットワーク言語の研究のために，SOCネットワークと協力する。

⑱ SOCRATE（CII—Grenoble）

異機種のSOCRATEに関する問題の一般的解決の研究。

4.1.2 CYCLADESの開発コスト分析

A．プロジェクトの見積り

研究プロジェクトにおいて，精度の高い費用の予測は一般に困難である。この理由は，研究の課程において新らたに発生する出費があるからである。しかしながら，コンピュータ・ネットワークにおいては，すべての技術が全く新らしいものではない。それらのうちの多くは，すでに試みられ，かなり成功したものである。

CYCLADES が，プロジェクトの開発コストのチェックのために利用したのはARPANETである。これは，ARPANETが，CYCLADES より以前に実施された唯一の同様なシステムであったからである。もちろん，ARPANET の数字を生のまま利用したわけではなく，目的，作業環境，制約などの違いが考慮された。

ネットワークの構成のために必要なコストについての考察は，1971年の初頭に，一般的な仮定をおいて実行可能かどうかということで行なわれた。これは，費用見積りのための確かな根拠となった。

プロジェクトの主要な仕事が再評価され計画されたのち，よりつっこんだ分析が1972年の初頭に行なわれた。この中で実行すべきものは，大まかにいえば，次のように分割された。

ⓐ 調　整

ⓑ ホスト・インターフェースと交換網の一般的な仕様

ⓒ プロジェクト・マネージメントのためのソフトウェア・トゥール

ⓓ 理論的な側面の一般的な研究

ⓔ パケット交換網のソフトウェア

ⓕ ターミナル・コンセントレータのソフトウェア

ⓖ パケット交換網とターミナル・コンセントレータのためのハードウェア

ⓗ ホスト・インターフェースのソフトウェア

ⓘ リモート・ジョブ・エントリ，ファイル転送，コマンド言語，会話形アクセス，データ・ベース・ハンドリングのようなソフトウェア・サポート

1つのものを，4つの機種において作成するものとされた。

ⓚ データ・ベースを含むアプリケーションの研究と委託。1つにつき，6つの機種において作成するものとされた。

① ネットワークに関する研究

全体のプロジェクトは，1972年から75年までの4年間の計画であった。それぞれの年度には，それぞれの主題があった。

- 72年 ： 設計
- 73年 ： インプリメンテーション
- 74年 ： テスト
- 75年 ： アプリケーション

仮定したネットワークの規模は，20台のホスト，6台のパケット交換ノード，7台のターミナル・コンセントレータであった。

表4－2　プロジェクト別コスト分析表

	工数（人月）					機械時間（時間）					賃貸／購入ハードウェア（千フラン）					ハードウェアの研究（千フラン）				
	72	73	74	75	合計	72	73	74	75	合計	72	73	74	75	合計	72	73	74	75	合計
a　調整	12	12	12	12	48	24	24	24	24	96										
b　一般的な仕様	36	12	0	0	48	36	12	0	0	48										
c　ソフトウェア・トゥール	9	12	12	0	33	45	60	60	0	165										
d　一般的な研究	16	48	48	48	160	80	240	240	240	800										
合　計	73	84	72	60	289	185	336	324	264	1,109										
e　パケット交換網ソフトウェア	25	33	12	12	82	125	165	60	60	410	100	100	100	100	400	0	0	0	0	0
f　ターミナル・コンセントレータ・ソフトウェア	10	46	14	12	82	50	230	70	60	410	0	0	0	0	0	0	0	0	0	0
g　交換網＋コンセントレータ	18	22	13	12	65	0	0	0	0	0	0	800	1,600	2,600	5,000	800	0	0	0	800
合　計	53	101	39	36	229	175	395	130	120	820	100	900	1,700	2,700	5,400	800	0	0	0	800
h　ホスト・インタフェース・ソフトウェア	10	160	100	100	370					1,850	200	200	200	200	800	100	100	0	0	200
j　ソフトウェアサポート	48	144	64	48	304	240	720	320	240	1,520	0	0	0	0	0	0	0	0	0	0
k　アプリケーション	0	144	193	156	493	0	720	1,025	780	2,525	0	0	0	0	0	0	0	0	0	0
合　計	58	443	357	304	1,167	240	1,440	1,345	1,020	5,895	200	200	200	200	800	100	100	0	0	200
総合計	184	633	468	400	1,685					7,824	300	1,100	1,900	2,900	6,200	900	100	0	0	1,000

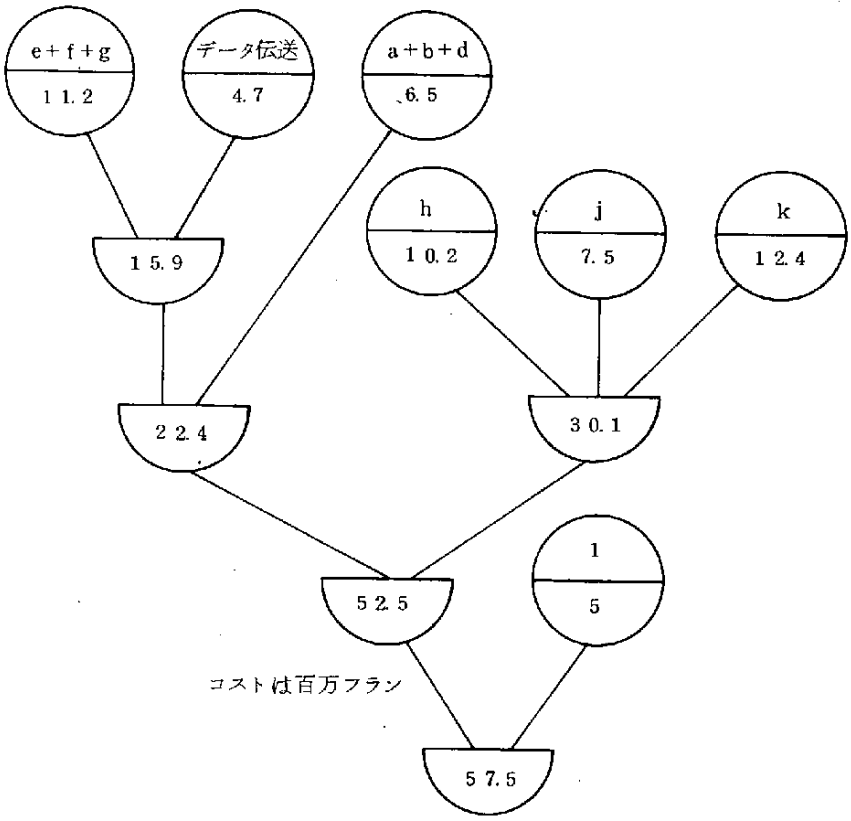
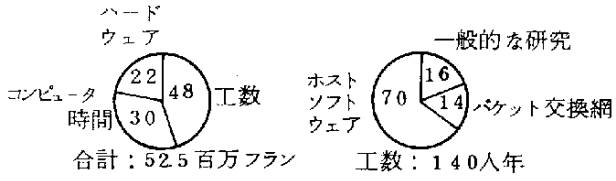
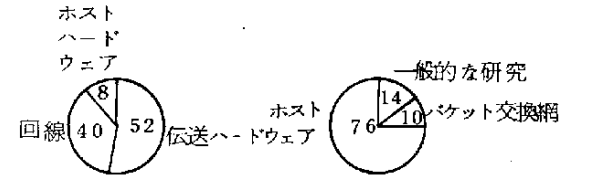


図４－１　コスト構成





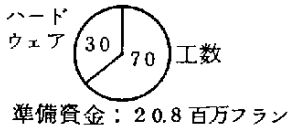


図４－２　相対コスト

人件費や機械使用時間の単位時間当りのコストは，正確に決めることができない。この理由は，プロジェクトの種類，使用コンピュータ，期間がそれぞれ異なっているからである。平均のレートは次のように決められた。

1人月（man·mouth）	：	15KF（約102万円）
1機械時間（Computer hour）	：	2KF（約13.6万円）

（KF＝1,000 Francs）

それぞれの年のプロジェクトごとの分析は，表4－2に数字で示されている。回線とモデムの評価は，はずされている。この理由は，フランスP＆Tとの間で75年末までは無料であるということになっていたからである。

種々の費用が，平均のレートによってフランに換算された。そしてデータ通信装置と関係する研究を加えるとネットワークを建設するのに必要な全費用となる。

この金額は57.5百万フラン（約39.1億円）となるが，この過程は，図4－1に示す。

最初から，参加センタは，工数，機械時間，ホストのハードウェア的な結合装置に関して何らかの費用を分担することが仮定されていた。

実際には，結合ハードウェアは小さなものであった。この理由は，CYCLADESでは広く使用されているメーカの伝送手順と，CCITTのインターフェースを採用したからである。

参加センタでは，ホストあたり次の費用を用意することが決められた。

工数	4人月
機械時間	300時間
ハードウェア	40千フラン（約272万円）

この結果として，残り部分は20.8MF（約14,144億円）となった。そしてこれは，情報代表部と軍の研究部によって用意された。コスト要素に関する用途ごとのグラフは図4－2に示されている。

B．プロジェクトの運用状況

前項で述べた最初の計画は，実行においてずれが生じた。新会員の募集，組織化に関して，いくつかの活動がスケジュールよりも遅れてスタートした。そして72年と73年の出費が予定よりも少なかった。この傾向は，74年には逆になり，75年にも持続されるであろう。

最初のつまづきにもかかわらず，主な目標はスケジュールどうりであった。1974年7月の時点で，予測された予算の約半分が投ぜられたが，全体的な予測が合わないような

徴候はない。

いくつかの分野において，このプロジェクトは異なった成長をした。実験であるとは逆に，いくつかの開発は，しだいに実用化の方向に向いていった。

最初は計画されなかったが，パケット・ターミナル，無線リンクなどの新しい分野が研究された。CYCLADES は，研究，オペレーション，実用化において，一時的な実験というよりはむしろ異なった方向で継続したと考えられており，新しいプランと適当な資金が用意されるのが期待されている。

4.1.3 CYCLADESの成果と今後

A．CYCLADESの利用

CYCLADES の利用に関して，基本的には2つの側面をみることができる。1つはコンピュータ・ネットワークの実験的な役割りであり，もう1つはそれの商用化である。前者をCYCLADES－0，後者をCYCLADES bis と呼んでいる。

(1) CYCLADES－0

初期の計画から75年末までのシステムをこう呼んでいる。

ネットワークの利用における主たる目的は，大学あるいはいくつかの研究所にコンピュータを持っている文部省のために，大学間ネットワークを構築することであった。

それ故に要求されたのは，大型コンピュータでサービスされているメーカが作り上げたソフトウェアを，ネットワークを通してターミナル・コンセントレータからのアクセスも含めて多くのユーザ間で共用できるようにすることである。この種のサービスは75年1月からグルノーブルのコンピュータで開始され，満足のいくものになっている。

実際に，グルノーブルの2台とIRIA の1台のコンピュータがサービス体制にある。他のものは技術的には動いているが実際，デモンストレーション，研究などに使用されている。IRIA のコンピュータは，過負荷状態にあるので，ジョブのうちのあるものは，グルノーブルのホストで実行している。

これに加えて，CYCLADES－0は，ネットワーク言語，新型の端末，新しいオペレーティング・システムなど，ネットワークの研究のための道具として使われている。

このような分野で，75年7月現在のところ進められているプロジェクトは次のとうりである。

・ TSSあるいはリモート・バッチで使用できるプロセッサ：APL，SOCRATE（汎用データ・ベース・マネージメント・システム），MISTRAL（ドキュメンテーションとインフォメーションのリトリーバル），テキスト・エディタやコンパイラ

などの通常のソフトウェア開発の道具。

- これらを使った，化学関係のデータ・バンク（THERMODATA），回線シミュレータ，コンピュータ・シミュレータ，経済モデル・システムなどのプログラム。
- 社会科学，言語学，教官管理，CAI，血友病に関するデータ・バンクなどのアプリケーションを開発中である。

軍事関係の実験や，関税や財務に関するパイロット・アプリケーションが準備されつつある。

(2) CYCLADES bis

CYCLADES bis は商用版である。フランスの官庁のうち現在2つの部門が，プライベートCYCLADES を採用した。これは，ソフトウェア会社によって設置されるだろう。

いくつかの会社では，自社デザインの自社システムを作ることを望む。この傾向は，新しい技術が広まったときに起る。技術スタッフは，新らたに経験することを望み，自社デザインのシステムだけが要求を満たすことをマネージャに説得さる。ある場合には，そのような投資は正しいであろう。しかし，一般的にいえば，経験あるマネージャは，開発したところが保証するシステムを好む。

コンピュータ・ネットワークにおいても同様である。

CYCLADES は，異機種ネットワークに対して，よく決められたプロトコルを持った唯一の商用化システムである。現在のシステムは殆んどCIIの製品でサポートされているが，CYCLADES のプロトコルを種々のオペレーティング・システムにおいて商用化して使うことができるようにする研究が進められている。

B．開発状況

CYCLADES プロジェクトは，種々の分野におよんでいる。次にあげるのは，そのうちの主なものである。

(1) 分散形データ・ベース

長期的な目標は，いくつかの異なったデータ・ベースを並行的にアクセスできるようにすることである。

第一のステップはSOCRATE を利用することによってインプリメントされた。このデータ・ベース・システムは，CII，SIEMENS，IBMのコンピュータで使うことができる。

いくつかのコンピュータに分散された同機種のデータ・ベースにアクセスできるネッ

トワーク・アクセス法がインプリメントされた。第二のステップは，２つのデータ・ベース・システムを結合することである。

科学技術情報の共有や配布のために相当の努力がされ，この１０年間に特別な形のデータ・バンクを作るために相当の投資がされた。これは全世界的なものである。フランスにおける調整機関はＢＮＩＳＴ(Bureau National d'Information Scientifique et Technique)とＢＡＢ(Bureau d'Automatisation des Bibliotheques)である。これらの機関は，全ヨーロッパの科学情報をプールしようとしているＥＵＲＯＮＥＴプロジェクトによってヨーロッパ共同体に含まれている。ＣＹＣＬＡＤＥＳ と ＳＤＳ－ＥＳＲＯ ネットワークの相互結合は，この一部である。この分野におけるフランスとカナダの間で行なわれた協力は，７５年に共同プロジェクトとなる。

全世界レベルにおいて，すべての国の科学情報の配布を計画し組織化する目標を持った長期の研究がＵＮＥＳＣＯで行なわれている。

(2) ネットワーク言語

ポータビリティのあるネットワークに共通な言語を作ることの可能性に関する研究プロジェクトがある。この作業は，ＥＩＮ(Europian Informatic Network)と共同で行なわれている。

(3) シミュレーションとモデル化

ホスト・プロトコルは，論理性と性能の評価のためにインプリメンテーションに先だってシミュレーションが行なわれた。また，ルーティングとフロー制御手法をテストするためにＣＩＧＡＬＥのモデルが開発された。後者の場合，カナダのワーテルロー大学，その他のヨーロッパの研究グループとの協力で行なった。

(4) 端末からのアクセス

コンセントレータは，種々の構成を許したり，ＣＩＧＡＬＥによる管理ができるようにするために修正中である。標準ネットワーク・パケットを直接処理できるプロトタイプの端末とミニ・コンセントレータの研究を行なっている。無線ターミナルも研究されている。

(5) ネットワークの拡張

何台かのターミナル・コンセントレータが７５年と７６年に増設される。２ないし３台のノードが必要になってくるであろう。

ＣＹＣＬＡＤＥＳの開発は，公衆パケット交換網のＴＲＡＮＳＰＡＣ と緊密に協力している。ＣＩＧＡＬＥとＴＲＡＮＳＰＡＣのデータ・グラムのコンパティビリティは保証されている。

C．動　　作

コンピュータ・ネットワークは相互に通信を行なわなければならないので，標準化が重要である。

プロジェクトの最初の頃から，標準化は考えられてきた。そして，他のネットワーク作成者とあるレベルのコンパティビリティを可能とするための意見の一致を見い出す努力をしてきた。ＮＰＬとの相互結合はこの例である。

他の要素は，公衆網とプライベート網の問題である。ＣＩＧＡＬＥは，ＣＣＩＴＴＴ用語でいえば"データ・グラム"と呼ばれるサービスを行なっている。それ故に，プライベートなパケット交換網を作っているユーザは，将来の拡張計画に応じて，公衆パケット交換網と入れ替えることができる。

ＣＹＣＬＡＤＥＳ の成果は，しだいに個々の企業に侵透していっている。ソフトウェアとハードウェアの開発と技術は，今や商用化できるところにきた。そのレベルの開発はもはやＩＲＩＡ の使命ではない。現在ＩＲＩＡ においては典型的なネットワーク技術を必要とする軍，財務，あるいは公共のドキュメンテーション・システムのようなプロトタイプのアプリケーションに力が入れられている。

4.2　ＪＩＰＮＥＴの例

4.2.1　ＪＩＰＮＥＴの概要

Ａ．目　　的

ＪＩＰＮＥＴ（ＪＩＰＤＥＣ　Integrated　Project Network）は実用と実験の両目的を持つ分散型のリソース・シェアリング・ネットワークである。ネットワークの規模は当初当協会内の国産３機種ＦＡＣＯＭ，ＨＩＴＡＣ，ＮＥＡＣを結合したローカルなものではあるが，将来は端末やＮＡＰ（Network Access Processor）による外部からのアクセス，また，外部システムとの結合等の拡張も考慮している。ＪＩＰＮＥＴにおける実用上の要求は，地理的に離れている部門からの３機種の総合リソースの利用，３機種間のハード，ソフト，データ・ファイル等の可能な範囲での互換性の確立，２機種以上の協業による特殊ジョブの処理，漢字入出力，マーク・シート・リーダ等のペリフェラルの各ＣＰＵ からの共用等であり，実験的な目的としては，必要とするユーザ・レベルのプロトコルの種類，ホスト内のネットワーク・コントロール機能の効率やオーバヘッドの測定；コンピュータ・ネットワークのコスト効率やパフォーマンスの評価，異機種間の分散型データ・ベース・マネージメントあるいは仮想マシンとしてのネットワークの有効性の検討等を具体的なインプリメントを通して行なうものである。

コンピュータ・ネットワークの本質は，それぞれ特色ある各種リソースを相互に有効に利用することにある。真に効果のあるリソースの共用を実現するには，それぞれのホスト側でのリソース提供のサポート機能が，経済性，効率性等の諸点からも充分納得し得る質を保つことが要求される。しかしながら，従来のＯＳとの関連もあり，現在のネットワークにおいてはこの点に関しては必ずしもまだ充分満足し得る状態にあるとはいい難い。ＪＩＰＮＥＴにおける主たる目標は，この事実に焦点を当て，特に国産コンピュータを対象とし，今後我国におけるコンピュータ・ネットワーク利用の実現可納性と実用性を検証し，その促進のための考察を具体的なレベルで行なうことにある。

B．システム構成

ＪＩＰＮＥＴは分散型ネットワークのモデルとして現在は図４－３に示すように，３つのホスト・コンピュータとＮＥＡＣ　３２００／５０をノード・プロセッサとする３ノードのサブネットおよび各種の最新端末群をノードに接続した形で形成されている。ノード間は４８Ｋbps の高速通信回線で接続し，ＨＯＳＴ—ＮＯＤＥ 間は８ビット・パラレル転送のインタフェース・アダプタを製作し，セレクタ・チャネルにより結合している。また，ノード２に結合されているＵ２００はＮＡＰ（ Ｎetwork　Ａccess　Ｐrocessor ）と呼びＡＲＰＡＮＥＴのＡＮＴＳ（ＡＲＰＡ　Ｎetwork Ｔerminal Ｓystem）に相当する。

システムの構成にあたって，以下のような諸点に留意した。

① コンピュータ・ネットワークとしての基本的な機能が全て実験可能な構成であること。

② コンピュータ・コンプレックス・システムとして，当センタ特有の利用法の実験にも適した構成であること。

③ 我が国での実用性が比較的高いと思われるＡＲＰＡＮＥＴでいうＴＩＰ（Ｔerminal ＩＭＰ）あるいはＡＮＴＳ等を重視した形をとった。

④ 将来，他のシステムあるいはネットワークとの結合などの拡張が容易であること。

⑤ 従来，各ホスト・コンピュータで行なってきた作業が，何等変更なしに引き続き稼動可能のこと。なお，現在のソフトウェアの変更なしに１６ノードまでは拡張可能で，各ノードには３つまでのホストが結合し得る。

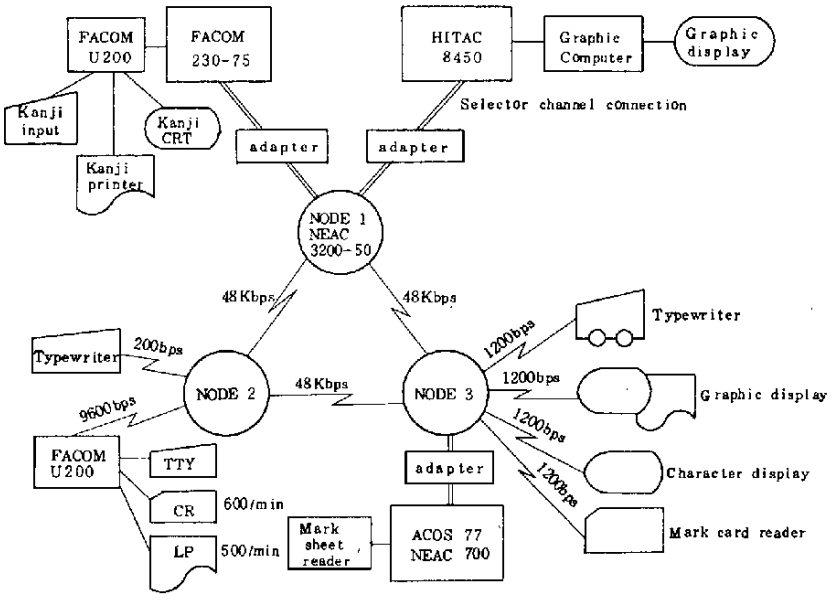


図４－３　ＪＩＰＮＥＴ構成図

C. 機　　能

ＪＩＰＮＥＴにおける主な処理パターンの例を図４－４に示す。①～④は２つのホストを対象としたもので，例えば

	HOST A	HOST B	HOST C	HOST D	端 末
①	S, O	E, P, D			
②	S, O, D	E, P			
③	S, O, D, SP	E, C			
④	S, O, C, E	SP, D			
⑤	S, O	E, P	D		
⑥	S, O, D	E, C	SP		
⑦	S, O	SP, D	E, C		
⑧	S, O, SP	D	E, C		
⑨	S, D	E, P	O		
⑩	T, D	E, P			
⑪	T	E, P, D			
⑫	T	E, P	D		
⑬	E, P, D				T
⑭	E, P	D			T
⑮	E, P	D1	D2		T
⑯	T	E, P	D1	D2	
⑰	S, O	E1, P1	E2, P2	D	

S：制御情報の入力
O：結果の出力
E（またはE1, E2）：実行
C：コンパイルまたはアセンブル
SP：ソース プログラム ファイル
P（またはP1, P2）：実行形式プログラム ファイル
D（またはD1, D2）：データ ファイル
T：端末（SとOを含む）

図４－４　処理パターンの例

①はＨＯＳＴ Ａから制御情報をＨＯＳＴ Ｂに送り(S)，ＨＯＳＴ Ｂにあるプログラム(P)およびデータ・ファイル(D)を使用して処理(E)を行なうことを依頼し，結果をＨＯＳＴ Ａに戻し出力(O)するケースである。また②はＡからＢに同様に処理を依頼するが，その際，データ・ファイルがＡにあり，これを予め，あるいは処理中にＢに送りながら処理を実行する。また③はその際ソース・プログラム・ファイル（ＳＰ）もＡにあり，これをＢに送りコンパ

イル(O)も依頼する。

次の⑤から⑨は，A，B，Cの3つのホストにわたって処理を行なう場合の例で，例えば⑤は，Cにあるデータ・ファイルを使用してBにあるプログラムで処理をすることをAから依頼する形である。また⑨は，Cにある特殊出力装置（例えば，漢字プリンタ）を共用するような場合の例で，Aにあるデータを Bに送って処理を行ない結果をCに出力する。

⑩から⑭は，リモート・バッチや会話型端末からネットワークにアクセスする形で，端末(T)はここでは制御情報の入力(S)と結果の出力(O)の両機能を含むものとする。

このうち⑩から⑫は，あるホスト（この場合A）につながる端末から他ホスト（この場合B）の会話型あるいはリモート・バッチ処理の使用であり，⑬，⑭は特定のホストを経由せず，ノードに直接結合された端末からのネットワーク使用である。

⑮，⑯，⑰は，ネットワークのやや進んだ使用法であり⑮，⑯は2つのホストにデータ・ファイルが分散しており（D1とD2），⑰は処理自体が2つのホストで分担（E1とE2）されている。これらはそれぞれ，分散型データ・ベースと複数ホストの協業により，1つのジョブを実行する形の例であり，より多くのホストを対象とすることももちろん考えられる。

D. プロトコル

図4－5にプロトコルの階層を示す。

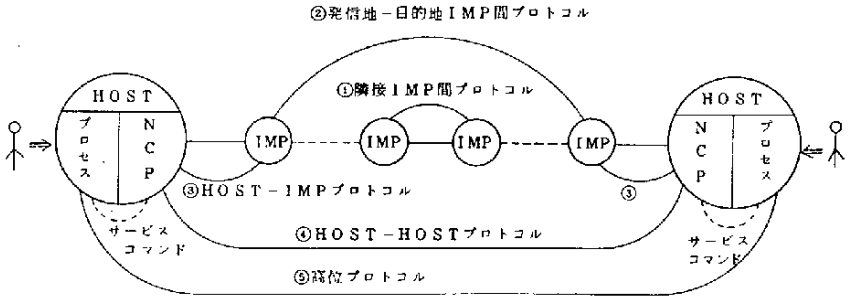


図4－5　JIPNETのプロトコル

① 隣接ノード間プロトコル

隣接したノード間での伝送データのフォーマット，受信確認方式，伝送エラーの検出と処置，再送の手続，パケットのルート選択および相互の障害と回復の検出と処置等に関する規約がある。

② 発信地ノード — 目的地ノード間プロトコル

発信地と目的地間のEnd — to — End 処理にともなう規約であり，メッセージのパケット分割やリアセンブル（reassemble），メッセージの粉失，重複等に関する伝送エラーの検出と処置，サブネット内のフロー・コントロール，伝送されたメッセージの順序付けなどに関する規約がある。

③ NODE — HOSTプロトコル

このプロトコルはサブネットとホストの間のデータの授受に関する規約であり，データ・フォーマット，送受信の手順および確認の方法，相互の障害状態の検出などに関するものがある。

④ HOST — HOSTプロトコル

このプロトコルは，ホストどうしが交信し，相互にコントロール情報およびデータを転送し合い，ネットワークを通して各種の処理を行なうための基本となる規約である。例えば，ホスト間のデータ・フォーマット，相互交信のためのホスト間のコネクションの確立，コネクションを通したデータの転送，データ転送に伴うホスト間のフロー・コントロール，相手側の障害の検出などに関するものである。

⑤ 高位プロトコル

ネットワーク・ユーザが直接使用するコマンドの形で規定されるハイレベルのプロトコルで，現在DSP（Demand Service Protocol），RBP（Remote Batch Protocol）およびFTP（File Transfer Protocol）をサポートしている。

これらのうち①～④はシステム・レベルのプロトコルである。これらのプロトコルはノード・プロセッサ内のSCP（Subnet Control Program）と各ホスト内のNCP（Network Control Program）により分担して処理されるが，プロトコルの設定におけるホストとサブネットの負荷分担は，ホストの負荷軽減を考慮し，メッセージのパケット化，リアセンブリ，伝送エラーのリカバリ，シーケンシング等，End-to-Endの殆んどをサブネットの分担としている。

E. 高位プロトコルの処理

現在は，DSP（Demand Service Protocol），RBP（Remote Batch Protocol）およびFTP（File Transfer Protocol）がサポートされている。

各ホスト内のプロセスにはユーザ・プロセスとシステム・レベルのユーティリティ・プロセスとを設け，高位プロトコルの処理は拡張性を重視し，それぞれユーティリティ・プロセスとしてサポートしている。従ってホストはそれぞれのプロトコル毎にユーザ・プロ

セスとサーバ・プロセスとを持っている。図４－８に各プロトコル処理の概念図を示し，図４－６，４－７はそれに対応するジョブ・デックおよび会話の例である。

```
¥ ID   ××××
# USER NO.      ××××
# HOST B
# OUTPUT S
# END
JCL for HOST B
¥ JEND
```

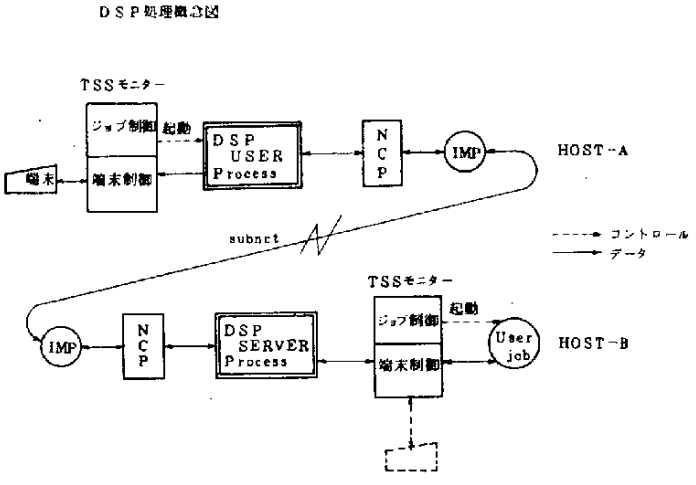
¥ HOST AのJCL
RBPコマンド

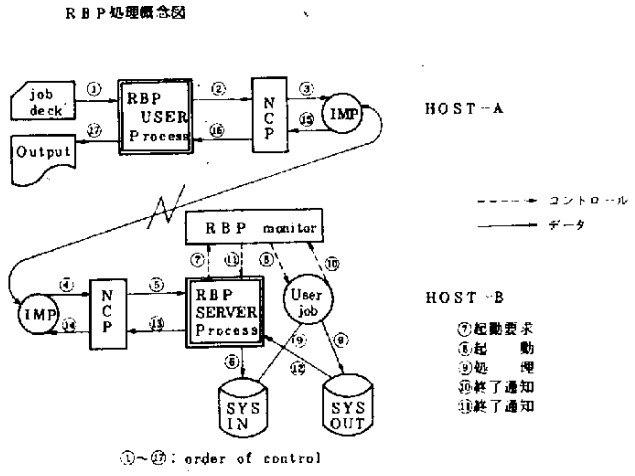
図４－６　ＲＢＰのジョブ・デック

```
BK ?      @OPEN     DSP     B
          USER NO =      ×××××
          DSP SERVICE START
          HOST－Bとの会話
BK        @CLOSE
          DSP SERVICE END
```

under lineはユーザの入力
BKはBreak KeyのKey in

図４－７　ＤＳＰの会話の例





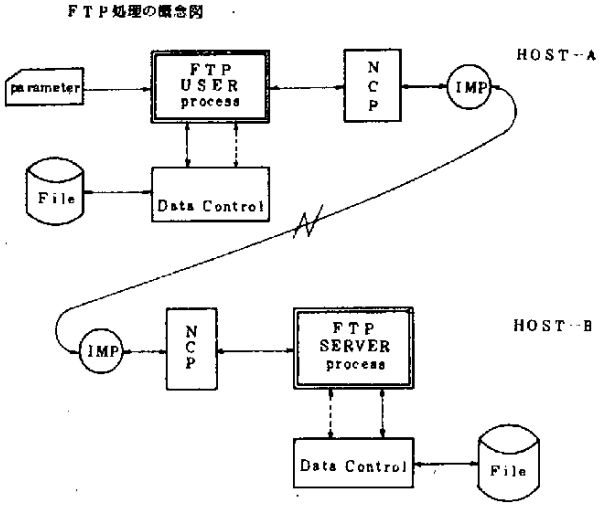


図4－8　高位プロトコル処理の概念図

F. サブネット（Subnet）

サブネットはパケット交換方式を採用しており，ネットワーク全体の信頼性，会話型処理および大量データ転送における効率性，ホストやノード数の増加，任意の回線速度等に応じられる拡張性および可変性を重視し設計した。全体的にARPANETの実績をかなり利用したが，各ノードにサブネットのトポロジを意識させている点，長短メッセージのパイプ管理，ルーティングのアルゴリズム，発信地，目的地間のメッセージのACKをホス

ト間で行なっている点等に関しては，かなり異なっている。Artificial Traffic Generaterを用いて測定した結果，最大スループットは単一パケット・メッセージ（144byte以下）の場合，58K bit／sec，多重パケット・メッセージ（1152byte以下）の場合，64K bit／secであった。また，round trip timeは100byte メッセージの場合1Hopで27ms，2Hopで52msであり，20Hopでは約0.5秒と推定される。

以下に各機能の概略を述べる。

(1) ノード（IMP ： Interface Message Processor）の機能

ノードは発信地ホストから目的地ホストに信頼度よくビットを伝送するコミュニケーション・システムとして動作する。この様な高信頼度，高能率なデータ伝送機構を実現するために，JIPNETではサブネットワークのソフトウェアに関して，その設計目標として次の様な基本的なねらいを設定した。

① サブネットの動作はホスト・コンピュータの動作と独立であること。

サブネットとホスト・コンピュータとの間に論理的な支援関係があってはならない，即ち，ホスト・コンピュータがサブネットの論理的動作を変える様なことがあってはならない。このようにすれば，ホスト・コンピュータに発生した障害が原因でサブネット機能が停止するような事態が避けられることになる。

② IMPは相互に依存関係を持たないこと。

1つのIMP のダウンがサブネット全体の機能を停止させることがないようにするためには，IMPは各々自律な系として動作することが望ましい。

③ 会話型処理を可能にするためには，短いメッセージに対して十分な応答特性が得られること。会話型処理においては，人間の我慢できるレスポンスの限度は約2秒であるといわれている。ARPANET では，この内1秒が低レベル・ネットで費やされるとして，サブネットの平均応答特性は0.5秒を目標値としている。現在達成されている値は約0.2秒であるという。

④ ファイル・トランスファー等の利用形態に対しても十分耐え得るように，大量なデータ伝送に対しては十分なスループット・レイトを保証すること。

⑤ ネットワークの拡張性を考慮したソフトウェアであること。

サブネットの拡張時に，新たなソフトウェアを作成したり，従来のソフトウェアを大々的に修正したりすることがないようにするために，IMP のソフトウェアは全て同一にする。また，動作アルゴリズムがネットワーク・トポロジーに依存するようなことがなるべくないようにする。現在JIPNETがIMP として採用している機種は

N－３２００／５０であるが，ノード・プロセッサを同一機種で統一することはソフトウェアの統一化と云う観点からも必要である。

以上述べたような目標を達成するために，ＩＭＰは本来の蓄積交換機能に加えて，効率，信頼性を支えるための各種の機能を遂行しなければならない。これには，パケットを，最適径路を選んで送り出すためのルーティング機能，ネットワークの混雑（Congestion）を緩和するためのフロー・コントロール機能，メッセージのシーケンスを保証するシーケンス・コントロール機能，サブネットの障害に対処するための障害対策機能，ＩＭＰの自己保存機能，パフォーマンスに関する有用なデータを収集するモニタリング機能等の各種の機能が含まれている。以下ではこれらの機能の各々に関して，その動作機構を説明するが，これはネットワーク・ソフトウェアという観点からは，ＨＯＳＴ－ＩＭＰプロトコル，ＩＭＰ－ＩＭＰプロトコル，発信地ＩＭＰ－目的地ＩＭＰプロトコルに関する記述にあたるものである。

(a) 蓄積交換機能

サブネットはホストから受取ったメッセージをパケット化し，蓄積交換を行なって目的地ノードに届ける。目的地ノードではこれらのパケットをリアセンブルし，もとのメッセージに再組立し，目的地ホストに渡す。隣接ノード間の伝送確認は基本的にはパケット単位のＡＣＫ信号によって行なわれ，両ホスト間のメッセージの伝送確認はＲＥＣＥＩＶＥ信号によって行なわれる。隣接ノード間は回線を論理的な4.チャネルに分割し，回線の利用率を上げている。なお，ＡＣＫ，ＲＥＣＥＩＶＥの返送は原則として逆方向のメッセージに便乗させている。

(b) ルーティング

ルーティングの手法には種々のものがあるが，パケットの高速伝送，トラフィック変動に対する適応性，トポロジー変化への追従性，ノードの計算負荷の軽減等を考慮して，「Ｓｈｏｒｔｅｓｔ Ｑ＋Ｂｉａｓ」の手法を採用した。この手法は定常状態では外部からは情報をとり入れずに，ノード内部における動的な送信待ちキューの長さと，ネットワークの静的構造を示すバイアス値との和を各出力回線ごとに比較し，その最小のものを選びパケットを送り出すというものである。バイアス値は，ノードおよび回線のアップ・ダウンにともなうサブネットの重大な変化の際のみ動的に計算し更新している。

(c) フロー・コントロール

フロー・コントロールはＡＲＰＡＮＥＴのパイプの概念を採用し，発信地 － 目的地ノード間にパイプという論理的なパスを設定し，このパイプの容量を制限することによりフローを制御している。ＪＩＰＮＥＴでは単一パケット・メッセージ（１４４バイト以下）

用のＳ－Pipeと多重パケット・メッセージ（１１５２バイト以下）用のＬ－Pipeとの２種類のパイプを設け，それぞれ別個に管理している。現在Ｓ－Pipeには４個までの単一パケット・メッセージを同時に流すことができ，多重パケット・メッセージは目的地においてバッファがアロケートされたことを確認した後Ｌ－Pipeを通し，一時に１個だけ流すことができる。これにより，短かいメッセージの応答性が長大メッセージのディレィにより受ける影響を少なくしている。

(d) シーケンシングとエラー検出

サブネットに入る各メッセージには，各パイプごとに０～２５５までの番号がラウンディング（rounding）につけられて送り出される。この番号により目的地ノードではメッセージのシーケンシング（sequencing）を行なっており，発信地－目的地ノード間でのメッセージの紛失，重複あるいはRECEIVEの紛失等のエラー検出にもこのシーケンシング機構を利用している。

(e) ロック・アップ

サブネットでは次の４種のロック・アップ（lock up）を想定し対策を考慮した。

- store & forward lock up
- sequencing lock up
- reassemble lock up
- HOST／NODE lock up

これらのlock upに対してはノード内におけるバッファの管理アルゴリズムを十分に検討して防御手段を講じた。なお，sequencing lock upは１つの目的地にメッセージが集中する場合に発生するものであり，HOST／NODE lock upはHOST－NODE間のアダプタが半二重のために発生するものである。

(f) 障害対策

サブネットの障害対策は次のような点を基本方針としている。

- できる限り早期に発見する。
- 障害の発生およびその回復過程で，サブネット全体の運転は中断させない。
- 一部の構成要素が障害で失われても，それを除いた環境下での運転が可能なこと。
- 障害の回復は自動的に行なわれる。
- 異なった要素間における障害の切りわけが容易であること。
- ホスト障害の影響をネットワーク全体に及ぼさぬとともにネットワークの障害を各ホストのローカル処理に影響させない。

表４－３に障害の検出と処置の一覧表を示す。

表４－３　障害の検出と処置

障害の種類	検　出　法	処　置
回線障害	ハードウェア割込みによる	他ノードに通知，診断パケットの応答が８０回あれば回復とみなす。
隣接ノード障害	引き続く２０パケットの伝送に対し応答のない場合	同上，障害隣接ノード宛のパケットは棄却し，他のパケットはreroutingを行なう。
ディスコネクション・ノード	コネクション・マトリックスの演算による	自ホストに通知し，そのノード宛のメッセージ送出を一時中止
ホスト障害	ノードからホストへのWRITEコマンドが１０秒以内に終了せぬ時	そのホストへ送出するメッセージを棄却し，発信地へDOWN RECEIVEを返す。
メッセージの紛失	4／16秒以内にRECEIVEが返送されぬ時	紛失メッセージ番号を付して目的地にInquiry packetを送る。
メッセージの重複	メッセージ番号(0～255)により目的地で検出	先の到着したものを生かし棄却
RECEIVEの紛失	発信地からのInquiry packetにより知る	既に該当メッセージを受取っていればRECEIVEを再送し，まだならばメッセージの再送要求を出す。
バッファの充満	バッファ管理メカニズムによる	ロック・アップ防止のため一定量のバッファを保留しそれ以上のパケットは棄却，その旨発信地に通知

(2)　TIP(Terminal Interface Message Processor)

ＪＩＰＮＥＴのＴＩＰはＡＲＰＡＮＥＴのそれとほぼ同じような機能をはたすものである。即ち，メッセージ交換をするＩＭＰ機能に端末装置の制御機能とホスト機能の一部を付加したものである。

ＴＩＰがユーザ端末に対してサービスするのは，ＤＳＰ(Demand Service Protocol)，ＲＢＰ(Remote Batch service Protocol)，およびＴＳＰ(inter Terminal communication Protocol)である。

ＴＩＰにおいては，ステーションがそれぞれのサービスを受ける単位となる。

ステーションとはタイプライター，ディスプレイのように会話型で使用できる端末（Control端末と呼ぶ）が1台と必要に応じてカード・リーダのような入力専用端末が1台，ラインプリンタのような出力専用端末が1台より構成されている。

ユーザは任意にステーションを構成し，このステーションからDSP，RBP，TSPのサービスを受けることができる。

(a) 機　　能

ＴＩＰの処理機能としては，①端末制御，②ＴＩＰコマンドの解析，③リモート・ホストとのコネクション管理などがあり，これらを処理するために必要なタスク，キューの関係を図４－９に示す。

(i) ＴＣＰ（Terminal Control Program）タスク

端末制御手順ごとに端末を制御するタスクで新しい手順の端末が付けられる毎に作成し追加しなければならない。ＴＣＰの機能は①端末の制御，②コードの変換（実端末コード←→ＮＶＴコード）である。

(ii) コマンド・インタプリタ・タスク

ステーションを単位として管理するタスクで，端末からbreak　キー打鍵後入力されるＴＩＰサービス・コマンドを解析する。

(iii) ＴＩＰ　ＮＣＰタスク

ＴＩＰを１つの独立したホストとみなし，リモート・ホストとのメッセージのやりとりを行なうタスクである。機能としては①コネクションの管理，②リモート・ホストとのデータ・フロー管理，③ＮＣＰコントロール・コマンドの解析等がある。

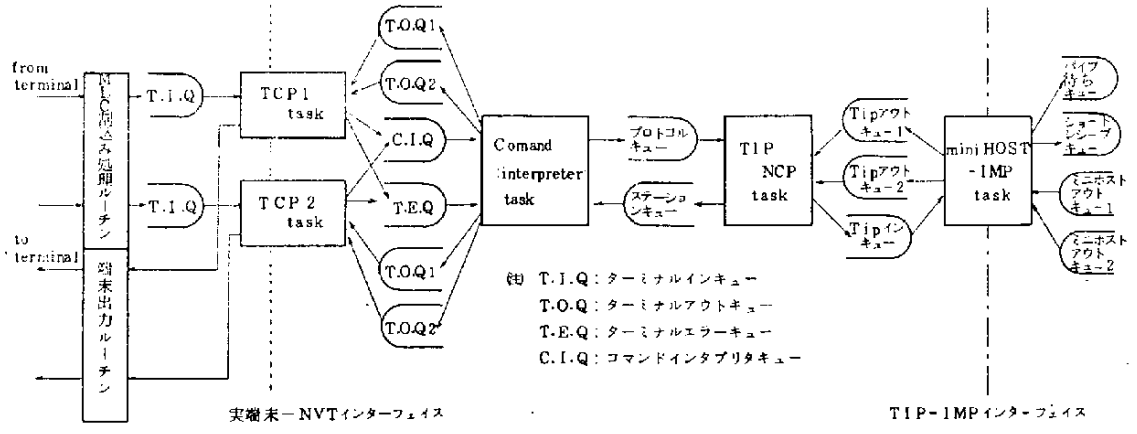


図４－９　ＴＩＰにおけるタスクとキューの関係

(iv) mini HOST－IMPタスク

ＴＩＰを１つの独立したホストとして管理するタスクである。このタスクの機能は①ＴＩＰ－ＩＭＰ間でのデータの受渡し，②目的地ホストの妥当性のチェック等である。

(b) ＴＩＰの使用法

(i) ＴＩＰと端末の接続

ユーザはＴＩＰからサービスを受けたい場合には，まず第１にコントロール端末からbreakキーを打鍵しなければならない。この動作によりＴＩＰは？を端末に出力して，表４－４に示すＴＩＰコマンドの入力待ちになる。

(ii) サービスの開始，終了要求

サービスの開始要求には@ＯＰＥＮ，サービスの終了要求には@ＣＬＯＳＥコマンドを用いる。例えば，ＦＡＣＯＭ２３０／７５のＴＳＳを使用したい場合には

```
ＢＫ ？ @ＯＰＥＮ ＤＳＰ 6
        ＵＳＥＲ  ＮＯ＝０６０００１
          ⋮   } ＦＡＣＯＭのＴＳＳに従う
ＢＫ ？ @ＣＬＯＳＥ
```

のように入力する。

表４－４ ＴＩＰサービス・コマンド

@OPEN	ＤＳＰ、ＲＢＰおよび端末間通信サービスを要求する。
@CLOSE	ＤＳＰ，ＲＢＰおよび端末間通信サービスの終了を要求する。
@WAIT	不特定の端末間通信ユーザに対して自端末を呼び出し待ちの状態にすることを要求する。
@CWAIT	ＷＡＩＴの状態を止めることを要求する。
@SEND	端末間通信において相手端末にメッセージを送信することを要求する。
@POST	ＤＳＰの使用中にリモートホストのサーバーに割込み信号を送ることを要求する。
@ASSIGN	入力専用端末，出力専用端末をステーションとして定義する。
@FREE	入力専用端末，出力専用端末をステーションから切り放す。
@TRANSFER	入力あるいは出力専用端末に対して入出力動作の開始を指定する。
@GIVEBACK	入力あるいは出力専用端末で行なっていた入出力動作を止める。
@RESTART	入力専用端末からの入力中に受信エラーを起こしたときその再入力の準備が終ったことを指示する。
@INSERT	端末に出力するメッセージの最後がlinefeedでないとき，linefeedの挿入を指示する。省略時は自動的に挿入される。
@RESET	newlineにより復帰改行が行なわれる端末装置のためlinefeedの自動挿入を止めることを指示する。
@STATUS	ステーションを定義するための入力あるいは出力専用端末の現在の状態の表示を指示する。

G. N C P

ＮＣＰ（Network Control Program）は各ホストに存在するネットワーク用のコントロール・プログラムである。

ＪＩＰＮＥＴは汎用のネットワークであり，ホスト間の通信は各処理プロセス間通信の形で実現している。即ち，図4－１０に示す様に各プロセスはそれぞれ，ホストのＮＣＰを通して交信を行なう。

各ＮＣＰは既存ＯＳのもとに付け加えられ，ＦＡＣＯＭではサブモニタとして，他の２機種ではユーザ・プログラム，即ち，ＨＩＴＡＣではリアルタイム・ジョブとして，ＡＣＯＳ では特権スレーブとして作成している。

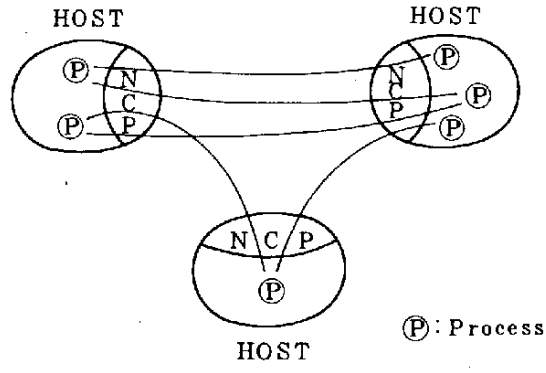


図４－１０ プロセス間コミュニケーション

ＮＣＰはＨＯＳＴ－ＨＯＳＴおよびＨＯＳＴ－ＮＯＤＥ プロトコルの規定にもとづく処理を行なうものであり，次のような機能を持っている。

① プロセス間のコミュニケーション

バーチャル・コール（Virtual Call）方式にもとづき，プロセス間のコネクションの確立とその閉鎖，コネクションを通してのデータの授受，プロセス間の同期などの処理を行なう。コネクションの確立とは，２つのホストのそれぞれのプロセス間で，データ流のためのロジカルなパスを作ることである。各プロセスにはホスト番号とユーザ番号とからなるネットワークに対してグローバルなｉｄがつけられ，ＪＩＰＮＥＴ ではこれをｐｏｒｔ－ｉｄと呼ぶ。１つのコネクションは２つのプロセスのそれぞれのｐｏｒｄ－ｉｄのペアに対して確立される。

また，１つのコネクションで両方向のデータ流を許す。

② ホスト間のフロー・コントロール

バッファ確保方式を採用している。ホスト間で１つのコネクションが確立すると，受信側ホストはメッセージを受信するためのバッファを１個以上（各ホストのその時点のバッファ事情により２～１０個）確保してその個数を送信側ホストに通知する。送信側は受信側において準備されたバッファ個数分のメッセージ送信を完了すると，新たにバッファ確保の通知がくるまでそのコネクションに対するメッセージ送信を一時中断する。

受信側はできるだけ速かにバッファを確保し，送信側に通知する。

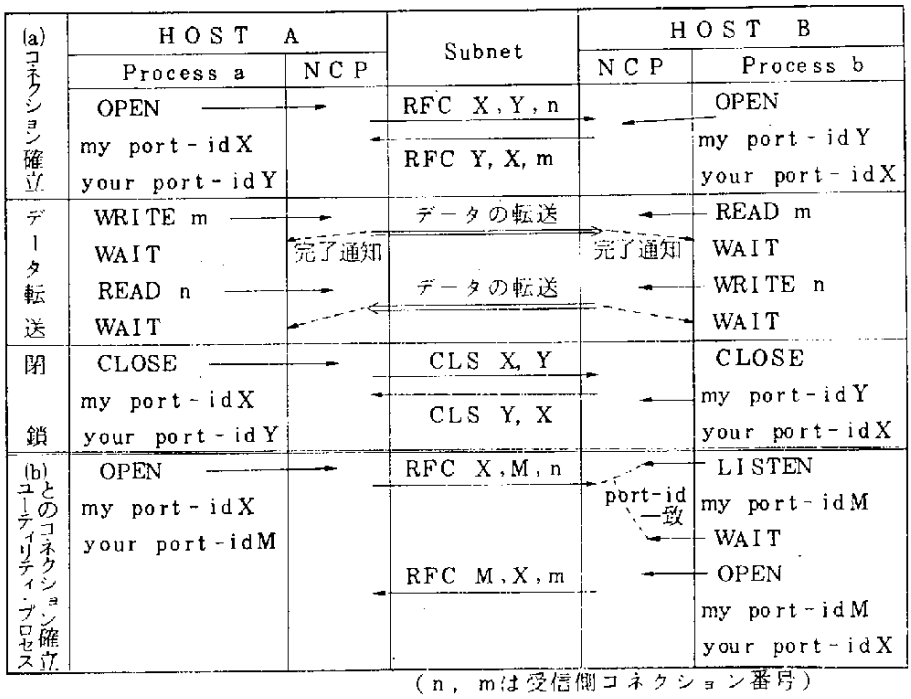
１つのホストで一時に確立し得るコネクション数は３０前後であり，ホストあたり１００ないし１５０パケット分のバッファを持っている。

③　コントロール・コマンドとサービス・コマンド

２つのホストのＮＣＰ間は，コントロール・コマンド，即ち，ＨＯＳＴ－ＨＯＳＴ プロトコル・コマンドにより交信される。一方ユーザ・プロセスとＮＣＰ間にはサービス・コマンドが設けられており，サービス・コマンドには多重コネクション管理，フロー・コントロールなどのマクロ機能を含んでいる。表４－５に両コマンドの種類を，図４－１１に両者の関連の例を示す。(b)はＭがユーティリティ・プロセスの場合である。

表４－５　コントロール・コマンドとサービス・コマンド

サービス・コマンド名	機　　能
ＮＣＢ	コネクション確立に必要な情報（ポート名など）を定義する
ＮＯＰＥＮ	コネクションの確立を指令する
ＮＣＬＯＳＥ	コネクションの閉鎖を指令する
ＮＲＥＡＤ	コネクションを通じてメッセージを受信する
ＮＷＲＩＴＥ	コネクションを通じてメッセージを送信する
ＮＬＩＳＴＥＮ	他系からのコネクション確立要求（ＲＦＣ）を待つ
ＮＳＥＴＥＣＢ	他系からのイベントを受信する領域（ＥＣＢ）を宣言する
ＮＰＯＳＴ	他系へイベントの発生を通知する
ＷＡＩＴ ＳＷＡＩＴ	} これらはＭ－Ⅶのシステム・マクロ命令であり，サービス終了などの同期を取る



（n，mは受信側コネクション番号）

図４－１１　両コマンドの関連

H. ＮＡＰ（Network Access Processor）

独自のホスト・コンピュータは持たぬが，端末を通してネットワークを使用したいユーザのために既ＴＩＰ（Terminal Interface Message Processor）を開発した。ＴＩＰはＩＭＰ（Interface Message Processor）と同一プロセッサ内にインプリメントされ，サブネットの一部であるため，個人向けの修正や改良を行なうことは不可能である。

ＴＩＰはＤＳＰ（Demand Service Protocol）などのハイレベル・プロトコルをサービスするために大量のＣＰＵを消費するので，ＩＭＰが大量のトラフィックを処理している場合，ＴＩＰのサービス率が低下する。また，ＴＩＰは多種多様な端末をサポートしており，多種類のプログラムが存在することによる安定度の低下により，サブネット自体の信頼性を低下させる原因となる。

ＴＩＰには，ディスク装置などの補助記憶装置がないので，ログオンしたユーザに対して正当性や予算などのチェックを行なうことができない。また，主記憶容量の点から，多様な端末や周辺装置を多数接続するのが困難であり，ＲＢＰやＦＴＰなどのハイレベル・プロトコルの全機能のサポートすることがむつかしい。

また，ユーザがネットワーク中のホスト・コンピュータ・システムをアクセスする場合，ユーザはそれに対する多様なアクセス・プロトコルや手順を熟知しなければ，自由自在に使いこなすことができない。なぜならば各ホスト・システムは，イニシャル・コネクションの確立／切断，ログオン／ログオフ，アプリケーション・システム（サブシステム）の呼出しなどに対し異なった手順を採用している。そのため，特定のホスト・システムに依存しない仕事をやろうとしても，ユーザは特定のホストを選択されたホストに対する個別のアクセス手順やそのレスポンス・メッセージの意味などに精通しなければならない。

したがって，ネットワーク全体で共通の標準コマンドを設定し，すべてのホスト・コンピュータに対してこの標準コマンドで使用できることが望ましい。しかし，これから新規に作成されるアプリケーション用のコマンドを除き，現実には，多種多様なホストのオペレーティング・システムやアプリケーション・システムが既に存在しており，これらに対して標準コマンドをサポートするように変更や追加を行なうことは，現実には困難であろう。

これに対処する例としてはNetwork Access Machineがある。これはユーザとホスト・システムの間にミニ・コンピュータを設置し，このミニ・コンピュータが，ユーザが入力した標準コマンドを受け取り，ターゲット・ホスト（Target Host）のローカル・コマンドに変換し，そのローカル・コマンドをターゲット・ホストに送り，それに対するローカル形式のレスポンス・メッセージをターゲット・ホストから受け取り，標準形式の

レスポンス・メッセージに変換して，ユーザに渡すものである（図４－１２参照）。

ＪＩＰＮＥＴにおいて開発したＮＡＰ（Network Access Processor）は，前述のようなＴＩＰの欠点を改善し，ユーザがネットワーク使用を安易にするためのインテリジェンス機能を実現する手段を与えるとともにＴＩＰ以上の多種多様な端末や周辺装置の制御を可能とし，且つ異なったハイレベル・プロトコルのサポートも含めシステムの変更を容易にすることを目的としたネットワーク・アクセス用プロセッサである。

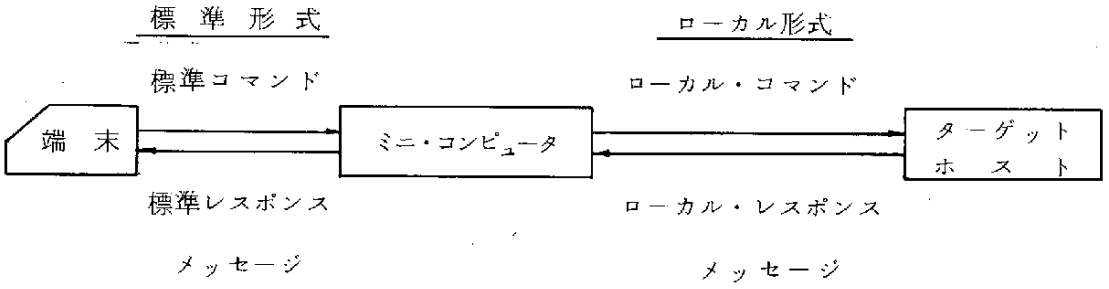


図４－１２　標準コマンド／レスポンスの使用

4.2.2　ＪＩＰＮＥＴの効率評価

ＪＩＰＮＥＴは過去４年にわたり開発を進め，実用レベルでの利用が可能になったコンピュータ・ネットワークである。

このネットワークを対象として次の３つの形態からの効率評価を行なった。

①　ホスト間通信の効率評価

②　ファイル転送における効率評価

③　ＴＳＳ利用プロトコルのインプリメントと効率評価

A.　ホスト間通信の効率評価

コンピュータ・ネットワークにおいて，あるホストを使用したとき，ローカルな処理と比べての効率評価を行なった。

(1)　測　定　項　目

次の３項目について，ホスト間の測定を行なった。

①　メッセージのスループット

②　メッセージの伝送時間

③　メッセージ処理に必要なＣＰＵ時間

(2)　測定方法とパターン

(a)　測　定　方　法

今回行なった測定では，サブネットのプログラムを改良し，ＩＭＰが持っている１ミリ秒のタイマを利用してメッセージの伝送時間，個数などを計測するとともに，ホ

ストのジョブ・アカウント・リストも併用した。

指定した長さのメッセージを1,000 個出力あるいは入力するプログラムをユーザジョブとして実行し，ジョブが終了したのちアカウント・リスト，ＩＭＰプログラムの統計データによって解析を行なった。ＦＡＣＯＭのＮＣＰの場合にはアカウントがとれないので（サブモニタであるため）ＣＰＵだけを使用するプライオリティの最も低いジョブ（idle）を同時に実行し，そのアカウントと，idleだけを実行したときのアカウントを比較することによってＮＣＰのＣＰＵ使用時間を推定した。

(b) 測定パターン

ＪＩＰＮＥＴでは，３台のホストは図４－１３のように結合されている。

ＦとＨは高速伝送が可能なように同一のＩＭＰに結合してあるが，比較の目的でＦをＡのところにつけ替えた場合の測定も行なった。

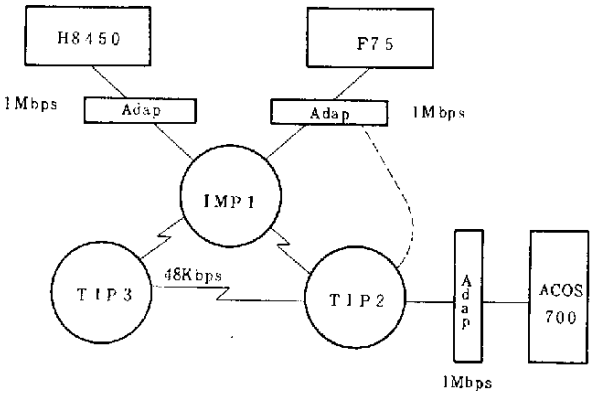


図４－１３　ＪＩＰＮＥＴの構成

(3) 測　定　結　果

ＪＩＰＮＥＴは，会話形指向のＳパイプ・メッセージという１４４バイト以下のものと，高スループットを得るためのＬパイプ・メッセージという１１５２バイト以下のものがあり，それぞれ伝送手順が異なるので別々に述べる。

(a) Ｓパイプ・メッセージ

３台のホストが，メッセージを送受信するのに使用するＣＰＵ時間は図４－１４のとおりである。

Ｈの場合は，送信に１７ミリ秒，受信に２３ミリ秒のＣＰＵ時間を使用する。受信の方がCPUを多く使用するのは，ＮＣＰとユーザ・プログラムの間での会話手順が，受信の方が１回多いことによる。

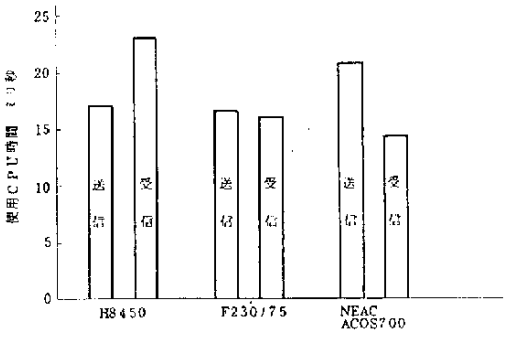


図４－１４　１４４バイトＳパイプ・メッセージの使用CPU時間

Aの場合その逆になるのは，ＮＣＰがＩＭＰからのメッセージ送信要求割込みを受け付けるために，割り込みが入ると実行される入力コマンドを発信しているが，送信のときには，このコマンドをキャンセルしたのち出力コマンドを発信しなければならないためである。

(b) Ｌパイプ・メッセージ

Ｆ←→Ｈ間での測定結果を，表４－６および図４－１５に示す。

表４－６より明らかなように，高速伝送を行なったときには，ホストのＣＰＵ使用時間はかなり大きくなる。

ホストのＣＰＵ使用時間が大きいこと，ＮＣＰにおいてサブネットのＬパイプ・メッセージの特性を生かしきれないなどの理由により，図４－１５に示すようなスループットの減少が起こる。

また，図４－１６にみられるように，ホスト－ＩＭＰ間の手順が同じならば，ＮＣＰのＣＰＵ使用時間は，Ｆのようにパケット個数に対して一定であるか，Ａのようにパケット個数に比例するかのいずれかである。Ａの比例現象は，コマンド・チェーンが使用できないため，ｎパケット・メッセージに対してｎ回コマンドを発信することにより起ったものである。

以上の効率評価結果を検討する

表４－６　Ｌパイプ・メッセージ1,000個の測定結果

項目		動作時間 sec	スループット Kbit/sec	FのCPU時間 sec %	HのCPU時間 sec %
F→H	同一IMP	48.9	188.	23s 47%	24s 49%
1152バイト	回線経由	216.	42.6	23s 11%	24s 11%
H→F	同一IMP	52.1	177.	16.5s 32%	26s 46%
1152バイト	回線経由	208.	44.3	16.5s 8%	24s 12%

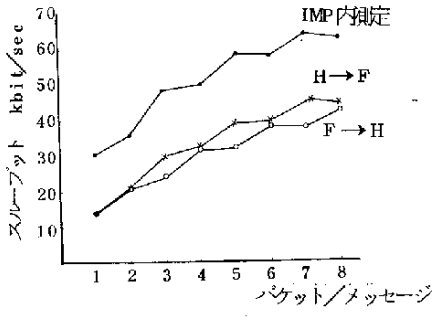


図４－１５　スループットＶＳ．メッセージ長

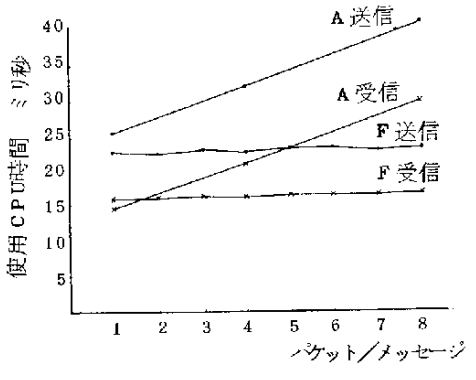


図４－１６　Ｌパイプ・メッセージの使用ＣＰＵ時間

と次のようなことがいえる。

コンピュータ・ネットワークにおいて，ＮＣＰがメッセージ処理に使用するＣＰＵ時間は，今まであまり注目されてはいなかったが，実際には予想外に大きく，これを何らかの方法によって小さくする必要がある。ＪＩＰＮＥＴにおける一連の測定は，ＮＣＰとＩＭＰおよびユーザ・プログラムとの相互交信の回数がＣＰＵ使用時間に大きな影響をあたえることを示している。このことは，送受信するメッセージ長あるいはパケット長を大きくすることによってスループットの向上，ＣＰＵ時間を減少させることが可能なことを意味する。

しかしながら，根本的には，ＮＣＰのハードウェア化，あるいはホスト・コンピュータ外にＮＣＰを置くなどの方法を検討していなければならない。

B. ファイル転送における効率評価

ＦＴＰ（File Transfer Protocol）はコンピュータ・ネットワーク内のホスト間でファイルを効率よく転送するための高位プロトコルである。

ＦＴＰの最も重要な点は大量のデータを効率よく転送することにある。

ここではＦＴＰを通したファイル転送の効率，ホスト間通信のメッセージ転送効率との比較検討，ソース・プログラムの転送，Ｉ／Ｏのオーバラップ機能，およびコード変換等の影響を評価した。

(1) 測　定　項　目

ＦＴＰプログラムのモデルを図４－１７に示す。

今回の測定では，ユーザＦＴＰ・サーバＦＴＰ間でファイルの転送を行ない

① スループット

② ファイル転送に使用したＣＰＵ時間に重点を置き測定した。

この測定データとホスト間通信の測定データからファイルの転送効率に影響を与えるとみなされる

① プロトコルの諸機能とその処理

② Ｉ／Ｏのオーバラップ機能

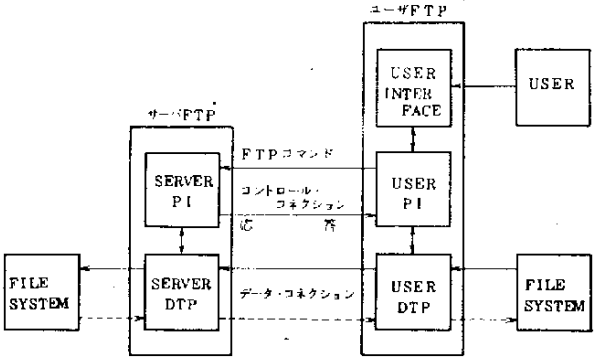


図４－１７　ＦＴＰプログラムのモデル

の２点について検討し，ＦＴＰプログラム作成上の留意点を述べる。

(2) 測定方法・測定パターン

ＦＴＰプログラムのスループットと使用ＣＰＵ時間の測定は，ホスト間通信の効率評価に用いた測定方法・測定パターンと同一のものである。ただしホスト間通信で用いた処理プログラムは単にメッセージ（１～８パケット）を送信あるいは受信するだけのテスト・プログラムであるが，ＦＴＰプログラムはディスク上のファイルをレコード単位で読み込み，それをホスト・ホスト・プロトコルに従ってメッセージに分割し送信する，あるいは受信したメッセージをレコードに組み立て，ディスク上にデータ型あるいはソース型ファイルとして作成する実用的なプログラムである。

C. 測　定　結　果

以下に測定と検討の結果を示す。

(a) プロトコルの諸機能と効率

プロトコルがファイル転送の効率に影響を与える要因は以下の３点である。

① 最大レコード長：ＦＴＰで扱うレコードの最大長は２０４８バイト（ＪＩＳ規定）である。

ホスト・ホスト・プロトコルで規定される最大メッセージ長は１１５２バイトであるため，これを越えるレコードを送信する場合には必ずレコードをメッセージに分割して送信する。また，ＦＴＰ特有のオーバヘッド（ヘッダー）がある。しかしながら表４－７のデータ型ファイルの転送効率は，ホスト・ホスト・レベルのスループットと同程度のものであり，上述の要因はスループットには影響を与えないことがわかった。

最大レコード長の設定はＦＴＰの利用形態とバッファのコア・オーバヘッドを考え合わせて決めてゆく問題である。

表４－７　データ型ファイルの転送効率（2048バイト×1000）

転送方向	FのCPU時間（秒）	HのCPU時間（秒）	メッセージ伝送時間の平均（m秒）	実効スループット（KBIT／秒）	オーバ・ヘッドを含むスループット（KBIT／秒）
F→H	47.6	102	426.4	38.4	39.7
H→F	36.9	75	439.0	37.3	38.6

② ライブラリ・ファイルの作成：ソース型ファイルは，各ホストの言語処理プログラムの入力となりうる形式で蓄積・管理している。

表４－８にデータ型とソース型ファイルの転送効率を示す。ソース型ではスループットの減少と使用ＣＰＵ時間の増大をまねく。これにはライブラリ・ファイルを作成するホストのデータ管理機能にディペンドするものであるが，ソース型ファイルのサ

ポートは付加機能であり，このスループットの低下は一般利用に際してさほど影響を与えないと思われる。

③ コード交換：文字型データはサーバＦＴＰの存在するホスト系における標準の文字コード（ＥＢＣＤＩＣあるいはＪＩＳ）で蓄積・管理される。

コード変換が加わる場合には，スループットが4.3％減少し，使用ＣＰＵ時間が１６％増加する。

(b) Ｉ／Ｏのオーバラップ機能

表４－９より送信側のディスクＩ／ＯとＮＣＰへのＲ／Ｗをオーバラップすると，メッセージ間隔が減少し，スループットが増加する。同一ＩＭＰ結合の時この影響はより顕著になる。

表４－８　データ型・ソース型ファイルの転送効率の比較

ファイル	転送方向	FのCPU時間（秒）	HのCPU時間（秒）	ALLOC数	メッセージ間隔の平均（m秒）	スループット（KBIT/秒）	備考
データ型	F→H	47.7	6.8	0	25.3	32.1	レコード長：1200 転送回数：1000
	H→F	36.7	6.5	0	72.6	2.9	
ソース型	F→H	68.7	10.9	1,751	364.9	14.9	
	H→F	25.7	5.5	2	107.2	29.5	

表４－９　オーバラップ機能の有無による転送効率の比較

送信側のオーバラップの有無		伝送時間の平均（m秒）	メッセージ間隔の平均（m秒）	ALLOC数	スループット（KBIT/秒）	備考
有	同一IMP	51.4	4.97	1,927	90.3	方向：F→H レコード長：1200 転送回数：1000
	回線経由	268.0	25.3	0	32.1	
無	同一IMP	55.3	8.4.4	1,012	66.3	
	回線経由	268.1	39.9	0	30.7	

以上の効率評価結果を検討すると次のようなことがいえる。

当初プロトコルで設定した諸機能がファイル転送の効率に影響を与えると推測されていたが，ソース・ライブラリ作成を除いて，他の機能はさほど効率に影響を与えていないことがわかった。このことはＩ／Ｏのオーバラップやファイル・アクセス等に留意してＦＴＰプログラムを作成すれば，ホスト・ホスト・レベルのスループットは期待できることを示している。

また，表４－９に示される同一ＩＭＰ結合と回線経由のデータを比較検討した結果，次のことがわかった。通常のコンピュータ・ネットワークにおいてはホストの処理能力がサブネットの伝送能力を上回っているが，同一ＩＭＰに結合したホスト間ではサブネットの伝送能力がホストの処理能力を上回るので，ＡＬＬＯＣ（バッファ予約済）の送出が頻繁に起こりスループットの低下をまねく。コンピュータ・ネットワークを指向する回線経由とコンピュータ・コンプレックスを指向する同一ＩＭＰ結合という異なった形態をもつネットワークにおいて，高スループットを得るためには下位のレベルでそれ

それ異なったプロトコルをサービスする必要がある。

C. ＴＳＳ利用プロトコルのインプリメントと効率評価

ＪＩＰＮＥＴではＤＳＰプロトコルを実現するために，実端末の存在するＴＩＰおよびＦＡＣＯＭ　２３０－７５にユーザＤＳＰを，また，ＴＳＳ機能をもつＦＡＣＯＭ　２３０－７５およびＡＣＯＳ　７７／Ｎ７００にサーバＤＳＰをインプリメントし，現在実用レベルで使用している。

ここでは，インプリメント上の問題点とネットワークを経由してＴＳＳを利用する場合，通常のＴＳＳの利用形態に比べてどれ程余分にホスト・コンピュータを使用するか，また，応答時間がどれ程遅れるかの２点について測定および評価を行なった。

(1) ＤＳＰのインプリメント上の問題点

既存のＴＳＳをネットワークを経由して使用するためにＴＳＳモニタは，サーバＤＳＰプロセスを仮想端末として制御する。従って，ＴＳＳとサーバＤＳＰのインタフェースをとるため，ＦＡＣＯＭでは端末制御プログラムを改造した。また，ＡＣＯＳではＴＳＳインタフェース・モジュールをＯＳに組込んだ。

インプリメント上，プロトコルとして問題となった点は，

① カナ文字や英小文字を扱えるＴＳＳとそれを印字できない端末での変換

② 端末のnewline 動作には，復帰および改行の２動作で行なうものと，どちらかのコードにより１動作で行なうものがあり，newline と復帰や改行機能をそれぞれ単独に実現するためのコードの扱い

③ 割込み信号（ＮＣＰのＰＯＳＴコマンドを使用している）の受信とその時点で送受信しているデータとの同期

などである。

(2) 測定項目および方法

ＤＳＰの実現モデルを図４－１８に示す。

ＤＳＰの端末ユーザにとっては，ネットワークを経由してＴＳＳを使用しても，通常のＴＳＳの利用形態に比べてほぼ同等の応答速度と利用料金でサービスを提供できるこ

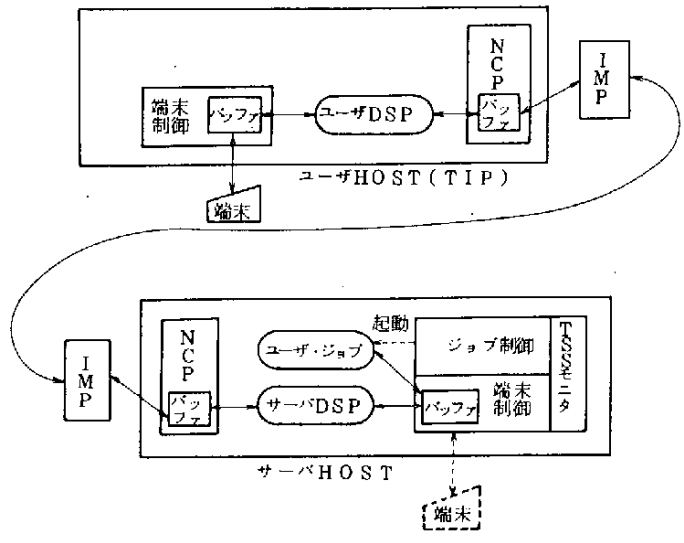


図４－１８　ＤＳＰの実現モデル

とが望ましい。

今回は特に既存のＯＳやＴＳＳそのものの評価には立ち入らず，ネットワークに関係する部分のみの評価に止めた。また，これを評価するためにパラメータにより出力メッセージ数を変えられるユーザ・プログラムを作成し，このプログラムを実行することにより サーバＤＳＰ～ユーザＤＳＰ 間でメッセージの送受信を行ない，

① メッセージの送受信に使用したホスト側のＣＰＵ時間

② 応答時間

について測定した。

応答時間はユーザＤＳＰ～サーバＤＳＰ間で送受信したメッセージ数とユーザＤＳＰの経過時間の関係から推定した。なお，測定には２００bps 端末を使用し，ＮＣＰの測定データはホスト間通信でのデータを使用した。

(3) 測 定 結 果

(a) ネットワークを経由してＴＳＳを使用する場合のオーバヘッド

ＴＩＰからネットワーク経由でFACOM 230−75のＴＳＳを利用した場合，使用するＮＣＰ，サーバＤＳＰのＣＰＵ時間は表４−１０のとおりである。これをグラフにあらわすと図４−１９のようになる。

図４−１９から，ホスト側のＣＰＵオーバヘッドはネットワークを経由しない場合に比べて，１メッセージ当り約１７ミリ秒増加することがわかる。

表４−１０ ネットワーク経由により使用するホスト側のＣＰＵ時間

送受信メッセージ数（144バイト/メッセージ）	測定用プログラムCPU時間（CPUTEST）(秒)	サーバDSP CPU時間（CPUDSP）(秒)	NCP CPU時間（CPUNCP）(秒)	ネットワーク→TSS CPU使用時間＝CPUNCP＋CPUDSP＋CPUTEST (秒)
147	0.3	0.2	2.026	2.926
547	0.5	0.2	9.026	9.726
1,047	0.7	0.2	17.276	18.176

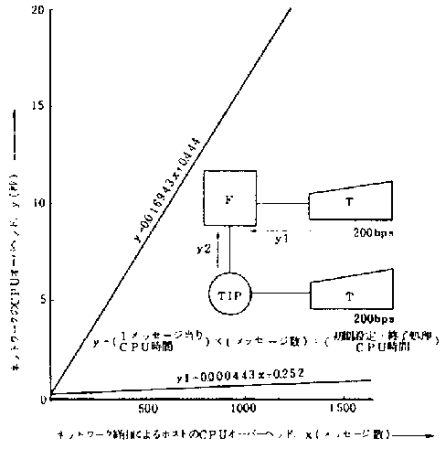


図４−１９

(b) 応 答 時 間

図４−２０はＪＩＰＮＥＴで利用可能ないくつかのＴＳＳ利用経路に基づき，ユーザＤＳＰの経過時間と送受信メッセージ数の関係をグラフに示したものである。

以上のような効率評価結果をまとめると次のとおりである。

サーバＤＳＰのオーバヘッドはＮＣＰに比べてかなり小さいことがわかった。しかし，ネットワークを通してサービスする端末数が増えることによってＮＣＰ＋サーバＤＳＰのオーバヘッドとして，ホストのローカル・ジョブに与える影響は大きいと思われる。

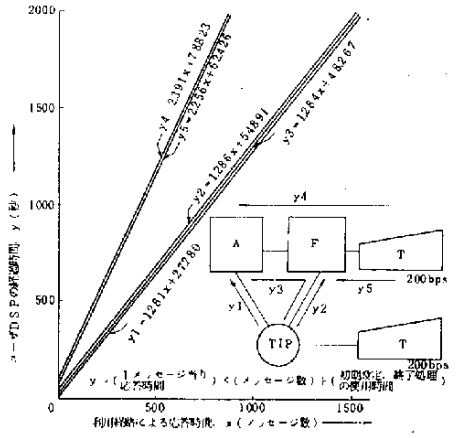


図４－２０

ＴＳＳの応答時間は一般に３秒以内が理想とされる。図４－２０からもわかるようにネットワークを経由することによる遅れは特に問題はなく，利用経路による遅れも余り大差のないことがわかった。図４－２０のｙ４の差はローカルな端末制御の相違にディペンドしている。また，ｙ１とｙ３で大差がないのはローカルな端末Ｉ／Ｏ処理とホスト間のメッセージ送受信処理がオーバラップされるからである。サービスする端末数が増えることによる応答時間の遅れは，ホストの既存のＯＳあるいはＴＳＳの性能にディペンドし，ネットワーク固有の要素はない。

4.2.3　コンピュータ・ネットワークへのホストの参加方式に関する一考慮

A.　ＪＩＰＮＥＴの実績

コンピュータ・ネットワークの価値は，それに参加し，有用なリソースを創造している人々の数に比例するといわれている。この意味で，ホストがネットワークに容易に参加できること，運用コストが低い形でのホストとサブネットの結合方法が必要となる。

表４－１１　ＪＩＰＮＥＴにおけるハード，ソフトの調査検討期間

作業項目	期間
インタフェースアダプタの機能検討	S.48.7～S.48.9
〃　の結合テスト	S.49.6～S.49.7
H8450のOSインタフェースの検討	S.48.7～S.48.7
F230-75　〃	S.48.10～S.49.5
ACOS700　〃	S.50.1～S.50.4

昭和４８年度から４年間にわたり，全体として４０人年の工数をかけて当財団で実施したコンピュータ・ネットワークＪＩＰＮＥＴでの開発を通じて得られた実績データおよび成果をもとにして，ホストの参加方式の一提案を行なう。

ＪＩＰＮＥＴにおける実績は，次のことを示している。

① 特注のハードウェアの機能設定を行ない，検収を完了するまでに１年間以上かかる。また，検収後もハードウェア障害の発見と修復に３日～１ケ月かかった。〔表４－１１参照〕

② ソフトウェアの開発およびテストに，ホスト１台あたりそれぞれ４人年程度の工数がかかった。〔表４－１２参照〕

③ ネットワークのサービスにあたってのホストの負荷は予想以上に大きい。〔表４－１３参照〕

④ ＯＳの機能追加などを行なった場合には，障害切り分けの必要性がある。またバージョンアップに追従するために要員の確保が必要となる。この結果，保守費用がかさむばかりではなく，メーカから新

表４－１２　ソフトウェア開発工数

（単位時間）

ソフト＼項目	調査	設計	コーディング デバック	計	備考
ＡＣＯＳ－ＮＣＰ	1108.5	530.5	1827	3466	3人 期間1年
ＦＴＰ ユーザ	0	99.5	233.5	333	経験４年
ＦＴＰ サーバ	0	155	476	631	
ＤＳＰ ユーザ	42.5	143	308	493.5	経験２年
ＤＳＰ サーバ	19.5	56.5	181	257 ※(1457)	経験７年

※ ＤＳＰサーバの工数には，既存ＯＳへの機能追加として，ＴＳＳインタフェースモジュールの作成工数，約８人月を必要とするのでそれを加算した値

表４－１３　ホストの負荷

項目＼機種	FACOM 230-75	HITAC 8450	ACOS77 NEAC700
ＯＳのサイズ	160 kw	130 kb	146 kw (常駐38kw)
ＮＣＰの形態	サブモニタ	ユーザジョブ	特権スレーブ
ＮＣＰのサイズ	20.7 kw	51.3 kb	24 kw
FTPのサイズ ユーザ	19 kw	65 kb	17 kw
FTPのサイズ サーバ	18 kw	60 kb	
DSPのサイズ ユーザ	11 kw		5 kw
DSPのサイズ サーバ	10 kw		
RBPのサイズ ユーザ	17 kw		
RBPのサイズ サーバ		38 kb	
１４４バイトメッセージのＣＰＵ使用時間			
送信	16 ms	17 ms	20.8 ms
受信	16.6 ms	23 ms	14.3 ms

表４－１４　ＯＳのリリース状況

（コンピュータＢ）

ＯＳ バージョン	一般ユーザへのリリース時期	当財団へのリリース時期	備考
V1.1	50.2	50.7	チャネル・アダプタ・モジュール追加
V2.1	50.8	51.2	TSSインタフェース追加
V3.0	51.3	なし	リリース要求せず
V3.1	51.9	52.3	DBMSの利用のため
V3.2	52.3	未定	

しく提供されたＯＳの利用できる時期が遅れる。〔表４－１４，４－１５参照〕

⑤　５２年度内に予定されているホストのリプレースにあたって，工数のかからないホストの参加方式を検討する必要がでてきた。

表４－１５　システム障害状況（コンピュータＡ）

期　間	ＯＳ障害によるシステムダウン回数	同左中ネットワークに起因する回数	ハードウェア障害によるシステム・ダウン	備考
S.51. 3 ～ 4	10		2	
5 ～6	6		0	
7 ～8	2		4	主に電源異常
9 ～10	7	4	5	9月よりネットワーク・モニタをクローズ運用に採用
11 ～12	5	2	2	
S52.11 ～12	3	0	0	

B.　新しい参加方式

4.2.2，4.2.3までで述べた状況をふまえて，新しい参加方式を検討した。この検討の結果，ホストとサブネットのソフトウェア・インタフェースを効果的にとるために，価格低下の著しいミニコンを仲介として利用することになり，その開発を始めた。このミニコンをＮＦＥＰ（Network Front End Processor）と呼び，図４－２１に示すように，コンピュータ・ネットワークのＯＳとしての機能を原則としてすべて実行する。現在のところＮＦＥＰは，主記憶装置６４ＫＢのＵ２００を使用して開発を進めている。

ＮＦＥＰ の特徴は，第１に，ホスト従属の部分とネットワークにおける共通部分を明確に分離してミニコン上にソフトをインプリメントすることである。第２に，ホストとの結合にあたっては，ホスト－ＩＭＰプロトコルのように標準化せず，ホスト従属のインタフェースを利用することである。

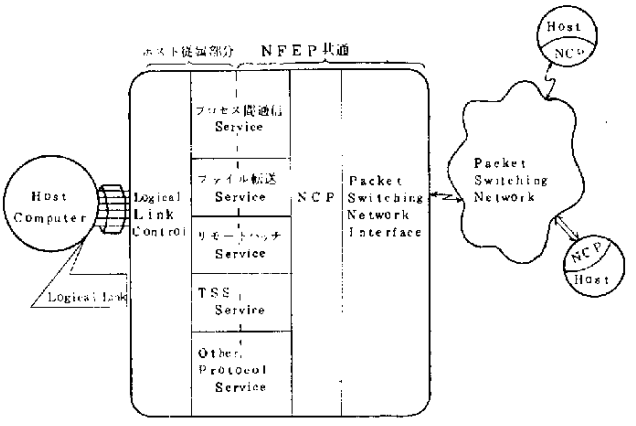


図４－２１　ＮＦＥＰの構造

この結果，次のあげる効果が期待できる。

①　ホストのＯＳとの関係をできるだけ少なくすることにより，開発時の調査および作成コストの削減ができる。ハイレベル・プロトコルとしてＤＳＰだけを考えたときのネットワーク基本ファシリティ開発は，約２／３の工数（１１人月削減）で作成が可能の見通しである。〔表４－１６参照〕

② NFEP は，ホストごとの異なる部分だけを作り変えればよい。このために変換作業が容易になる。同様のことが新しくネットワークに参加するホストに対してもいうことができる。表4－16の総計3663時間のうち少くとも，ホスト・ホスト・プロトコル以下の処理の1638時間，およびハイレベル・プロトコルの825時間のうち半分に当る410時間は共通に利用できるので，14人月程度の工数削減ができる。

表4－16 NFEPによるホストの結合の場合のデータ

（単位時間）

項目 / ソフト	調査	設計	コーディング デバック	計
フィージビリティ調査※	450			450
ホスト・ホスト・プロトコル以下の処理※	213	405	1065	1683
ホスト・NFEPのインタフェースプログラム※※	225	180	300	705
ハイレベル・プロトコル※※				
DSPユーザ	75	105	120	300
DSPサーバ	150	150	225	525
計	1113	840	1710	3663

※ 実績値を示す

※※ 見積値を示す

③ ネットワークにかかわるOS部分をホストから追放できるために，ホストの負荷（CPU，記憶装置）が軽減する。原理的にはほとんど0にすることができる。

④ ホストの標準インタフェースを利用するために，障害の切り分けが容易になる。また原則的には（ホストのOSの信頼性が十分に高ければ）ネットワーク関係の障害ではシステム・ダウンは起らない。

以上のようにNFEP による結合は，多くの長所を持っているが，今後の課題もいくつかある。その主なものは次のとうりである。

① データ・ベース処理など高位のユーザ・レベル・プロトコルの処理をさせる場合の，NFEPとホストの機能分担

② 新しいネットワーク・アーキテクチャにおいても有効であるかどうか

これらの点については，流動的な外部環境に追従しつつ，今後とも検討を要する。

4.2.4 JIPNETの開発経過

JIPNETの開発経過を表4－17に示す。

開発初年度である昭和48年度は，システムの基本構成を検討し，ネットワーク機能の検討およびハードウェア機器の仕様設定および導入を行ない，昭和49年度にJIPNET第1バージョンのシステムをインプリメントした。第1バージョンは，基本的なものではあるが，コンピュータ・ネットワークとして最低限必要な，サブネット・コントロール，ホスト間通信，

表４－１７　ＪＩＰＮＥＴの開発経過

時期（年・月）	内　　容
昭和４８年度	（調査・検討段階）
48. 4	ＪＩＰＮＥＴの開発開始
48. 4～ 6	ネットワークの構成，結合方式および異機種結合上の問題点検討
48. 4～ 8	結合用ハードウェア機器選定
	ＩＭＰ用に，Ｎ３２００／５０を採用　最新ターミナルを数台選定
48. 4～ 9	既存コンピュータ・ネットワークの技術調査
48. 6～ 9	ＪＩＰＮＥＴの基本機能設定
48. 7～ 9	結合用インタフェース・アダプタ機能検討
48. 7～11	ハードウェア・モニタにより既存ホストの効率測定
48. 9～49. 3	基本プロトコル（ホスト－ホスト，ホスト－ＩＭＰ，ＩＭＰ間）の検討
	ＮＣＰ（ＦＡＣＯＭ，ＨＩＴＡＣ）の実現手段検討
	サブネット機能の論理設計
49. 1	ＡＲＰＡのDr.R.E.Kahnによるレクチャー開催
49. 1	プログラミング・シンポジウムにて論文発表
49. 1～ 3	ＩＭＰおよびターミナルの設置，結合テスト
昭和４９年度	（第１バージョンのインプリメント）
49. 4～ 8	高位プロトコルとしてＤＳＰ，ＦＴＰ，ＲＢＰを規定
49. 4～10	サブネットのソフトウェア作成
49. 4～11	ＮＣＰ（Ｆ，Ｈ）の作成
49. 6～ 7	アダプタの結合テスト
49. 8	ホスト・コンピュータＦ／６０をＦ／７５にリプレース
49. 8～11	Ｆ／７５用サーバＤＳＰ，サーバＦＴＰ，ユーザＲＢＰ作成
	Ｈ－８４５０用サーバＲＢＰ，ユーザＲＢＰ作成
	ＴＩＰ用ユーザＤＳＰ作成
49. 12	ＪＩＰＮＥＴ第１バージョン完成
	工技院利用研に対しデモンストレーション
50. 1～ 2	サブネットの効率測定
50. 2	Ｎ／５００をＡＣＯＳ／７００にリプレース

時期（年・月）	内　　容
昭和４９年度	（第１バージョンのインプリメント）
５０．３	ＮＣＰの効率測定 ＡＣＯＳ／７００の結合テスト
昭和５０年度	（　評　価・改　良　）
５０．４～　６	ホスト－ホスト・プロトコルの改良 ＤＳＰの改良
５０．４～１２	海外の技術資料の調査 （仮想マシン，分散型データ・ベース等）
５０．６～１１	改良版ＮＣＰ（Ｆ／７５，Ｈ－８４５０）の作成 ＡＣＯＳ　ＮＣＰの作成 改良版ＴＩＰの作成（ユーザＤＳＰも含む） 改良版ＤＳＰの作成 Ｆ／７５用サーバＤＳＰ
５０．８	第２回ＵＪＣＣにて論文発表
５０．９	海外調査（米国，欧州）
５０．１１	情報処理学会　全国大会に７論文発表
５０．１２～５１．１	改良システムの効率評価（ＮＣＰ，ＤＳＰ）
５０．１２～５１．３	ＡＣＯＳ用サーバＤＳＰの作成
５１．１～５１．５	改良版ＦＴＰの作成 Ｆ／７５用　　ユーザＦＴＰ，サーバＦＴＰ Ｈ－８４５０用　　ユーザＦＴＰ，サーバＦＴＰ
５１．１～５１．５	改良版ＤＳＰの作成 Ｆ／７５用　　ユーザＤＳＰ 改良版ＮＴＣＰの作成 Ｅ／７５用　　モニター組込
昭和５１年度	（総合評価，高度機能の検討）
５１．４～５２．１	高度機能（ＤＤＢＡ，ＮＪＣＬ）の技術調査および検討
５１．４～５２．３	ＮＡＰ／ＮＦＥＰの作成
５１．７～　９	改良版システムの総合評価 ＮＣＰ，ＤＳＰ，ＦＴＰ

時期（年・月）	内　　容
昭和５１年度	（総合評価，高度機能の検討）
51. 7～10	改良版ＲＢＰの作成 Ｆ／７５用　　ユーザＲＢＰ Ｈ－８４５０用　　サーバＲＢＰ
51. 7～12.	ＮＪＣＬの言語仕様および実現性の検討
51. 9～52. 3	ＡＣＯＳ用ユーザＦＴＰの作成
51. 11	情報処理学会全国大会にて４論文発表
51. 11～52. 2	ＣＪＰによる実験

ＴＳＳ利用，ファイル転送，リモート・ジョブ・エントリの諸機能を有しており，このシステムを用いて実際のアプリケーション・プログラムによる利用実験を行なった。

昭和５０，５１年度は，リプレースされた新ホストのＮＣＰを作成するとともに，第１バージョン・システムの評価により，改良版プロトコルを規定し，各ホストのＮＣＰおよび高位プロトコル処理プログラムの改良版である第２バージョン・システムをインプリメントし，総合的な評価を行なった。とくに５１年度は，ＮＡＰ（Network Access Processor）のインプリメント，ＣＪＰ（Cooperated Job Processing）の実験等を行なうとともに，ネットワーク利用の高度化を図るための分散型データ・ベース（ＤＤＢＡ）やネットワーク・ジョブ制御言語（ＮＪＣＬ）の検討を行なった。

第 4 章　参 考 文 献

・Hawaii 大学

(1) W.H.Bortels,"Simulation of Interference of Packets in The ALOHA Time-Sharing System," March 1970,B70-2

(2) Alan C.H.Kam,"UHTSS Library Management Yesterday,Today and Tommorrow," Nov.1970, B70-3

(3) John M. Davidson, "UHTSS Console Control Program and OS Interface." Aug. 1971, B71-6

(4) Joho Davidson, "The ALOHA System Interface to TSO," May 1972, B72-3

(5) Franklin F.Kuo, "Selected Papers on Computer - Communication Nets." May 1974, B74-4

(6) Richard Binder,"Variable-Length Packets In an Unslotted ALOHA Channel," Jan. 1975,B75-2

(7) Michael J·Ferguson, "An Analysis of Variable Length Packets in Unsloted ALOHA," Feb.1975,B75-7

(8) David Wax,"Design Considerations For The ALOHA Radio Communication Sub-System," Feb. 1975,B75-11

(9) David Wax, "A Description of The ALOHA Radio Communication Sub-System," Feb. 1975.B75-12

(10) Franklin F. Kuo "User Stadards for Computer Networks," June 1975,B75-19

(11) Franklin F.Kuo, "Public Policy Issues Concerning ARPANET," June 1975,B75-20

(12) Richard Binder , "A Possible Scheme for Repeater Addressing," CCG/W-41

(13) 電子計算機利用に関する技術研究会：リソースシェアリングシステム P. 197,1976

(14) 日本情報処理開発協会：コンピュータ・ネットワークＪＩＰＮＥＴの研究開発，49-S001 50-S001，50-S003

(15) 第１７回情報処理学会全国大会論文集（229）：ＪＩＰＮＥＴにおけるホスト間通信の効率評価

(16) Wood, D.C. and Trehan, R.K.,"An Assessment of the Performance of Packet Switching Networks",EURCOMP '75 PP.1-11

(17) 日本情報処理開発協会：コンピュータ・ネットワークＪＩＰＮＥＴの研究開発，51-S003 P. 367,1977

(18) 電子計算機利用に関する技術研究会：リソースシェアリングシステム（NEEPのフィージビリティ調査）P. 138,1977

(19) JIPDEC, コンピュータ・ネットワークJIPNETの研究開発，50－S003,P. 361, 昭和51年3月

(20) JIPDEC, コンピュータ・ネットワークJIPNETの研究開発, 49－S001,P. 366, 昭和50年3月

(21) 山本他，JIPNETに関する論文，情報処理学会第16回大会，#108～#114, 昭和50年

(22) 伊藤他，JIPNETの効率評価に関する論文，情報処理学会第17回全国大会，#262～#264，昭和51年

(23) 伊藤他，コンピュータ・ネットワークへのホストの参加方式に関する一考察，情報処理学会第18回全国大会，#299 昭和52年

5. ネットワークの運用

ネットワーク・システムの規模が拡大し，多様なユーザがネットワークを利用するのに伴い，ネットワークの運用にかかわる課題が数多く発生する。

現在は，まだ本格的な商用コンピュータ・ネットワークの歴史が浅く，運用の問題がすべて解決されているわけではないので，本稿では以下のような課題をとりあげ，それぞれに関する事例を紹介することとする。

① ネットワークの信頼性と管理センタ

② ネットワークにおけるセキュリティ

③ ネットワークの評価とその活用

④ アカウンティング

⑤ ネットワークの運用の業務体制

これらの課題の他に，ユーザがネットワークの各ホストに存在する豊富なリソースを使いこなすために，ネットワーク・リソースの案内情報サービスが必要不可決となるが，現時点では，TELENET 社におけるサブスクライバのサービス・ディレクトリーができている程度である。

5．1 ネットワークの信頼性と管理センタ （文献１，６より引用）

ネットワークの保守，診断，障害対策等に関するネットワーク・コントロール・センタの役割について述べる。

ネットワーク機能の追加，変更に伴うメンテナンスをユーザ・サービスに影響を与えず，且つネットワークの信頼性低下を引き起さぬよう実施することが必要である。

事例としてARPANETのNCC（Network Control Center）をとりあげ，その概要，開発経緯を紹介する。

ARPANET は，分散型の蓄積パケット交換ネットワークである。インターフェイス・メッセージ・プロセッサ（IMP)と呼ばれるスイッチング・ノードが，高速（主に50Kbps）専用回線を介して相互に接続されており，付属のホスト・コンピュータのために通信機能を果たしている。特殊なスイッチング・ノード，ターミナルIMP（TIP）が多くのタイプのターミナルの直接接続を可能にしている。ARPANET ・コントロール・センタ（NCC）は，IMP，TIP,そして通信会社の回線から成る通信サブネットワークの操作・保守・変更の責任を負っている。

１９７０年初期に，NCC の機能はBolt Beranek and Newman Inc.(BBN)にあるIMP に接続されたテレタイプだけを使用して，システム・プログラマやハードウェア・デ

ザイナによってむしろ自由な形で実行された。１９７５年の春，ＮＣＣ には，コンピュータ・オペレータ，ハードウェア技術者や専門家，プログラマ，ＮＣＣ に専門的な責任を持つ全ての人々により，２４時間ぶっ通しで人間が配置されている。また，必要な時にハードウェアの開発スタッフの能力を集めている。

現在，ＮＣＣ によって使用されているハードウェアは，専用のＨ－３１６コンピュータ（ＴＩＰ のような形態），ネットワークに接続されたＰＤＰ－1の１部と何台かのＰＤＰ－１０，多数のテレタイプ，２台の１２００ボー・プリンタ，そして様々の周辺装置である。この拡張のいたる所に，操作，保守，変更に対するベーシックな責任が絶えず残されていたが，その活動の幅は，ＡＲＰＡとユーザ共同体の要求に反応して連続的に成長して来た。

Ａ．ＩＭＰ の保守機能

① ループ・バック・テストモード

プログラム制御下でのホスト・インタフェース，モデム・インタフェースの折返しテスト機能

② 電源障害の自動検知と電源復旧時のプログラム自動再スタートのための機構

③ 監視タイマ（watch dog timer）

プログラムの正常動作を監視し，プログラムが障害と認識されると，自動的に隣接ＩＭＰよりＩＭＰ プログラムを再ロードし再スタートする。

④ ＩＭＰ 統計パッケージ

・パケット・トレース

・メッセージ長，パケット長のヒストグラム蓄積

・ＩＭＰ 内部状態のスナップ・ショット

・アーティフィッシャル・トラフィック・ジェネレータ

上記の情報をＮＭＣ （ネットワーク測定センタ）に送出する。

⑤ ＤＤＴ （デバック用パッケージ）

・コマンド・インタプリタ

・メモリ・ワードの検査と変更

・メモリ・ブロックのクリア

・特定値がストアされたメモリの探索

ＤＤＴ はネットワークを介してリモートから操作でき，ネットワークを介して返答を返すことができる。ＤＤＴ の制御はＮＣＣ からのみ操作されうる。

Ｂ．ＮＣＣ 以前のオペレーション

最初のＡＲＰＡＮＥＴ のノードは，１９６９年末，ロスアンジェルスのカリフォルニア大学

に設置され，続いて３つのネットワーク・ノードがカリフォルニアとユタに設置された。こうした初期の時代におけるネットワークのオペレーションは，多少連続して各サイトにいたＢＢＮ スタッフのメンバーにはもちろんのこと，それらのサイトに雇われていた職員の援助に大いに頼っていた。ＩＭＰ 障害からの復旧時には，ＢＢＮ のプログラマと共に働いていた（そのサイトでいっしょに，あるいは電話を介して）サイト職員が，ＩＭＰ のコンソール・テレタイプを介してＤＤＴ をローカルに使用した。各ＩＭＰ は高速紙テープ・リーダを装備しており，各サイトはＩＭＰ ソフトウェアの最新版紙テープを保持していた。復旧にＤＤＴを使用できない場合には，サイト職員がテープ・リーダを通してＩＭＰ の再ローディングを行なった。もし何度かローディングしてもＩＭＰ 障害が続くようだと，問題はハードウェアにあるとされ，全てのマシンに保守契約を結んでいるハネウェル・フィールド・サービスが適当な診断・修理を行なうために呼び出された。

各ＩＭＰ は，自身に接続された通信会社の回線の状態を監視し，ＩＭＰ のフロント・パネル上の一連のライトにその状態を表示した。サイト職員は定期的にそのライトを調べ，もし回線がダウン状態にあると思われる時は，インターフェイスをループにしテスト・トラフィックを発生させるため，ＩＭＰ のコンソール・テレタイプを通してＤＤＴ を使用した。そして，その結果はフロント・パネルのライトに，ＵＰあるいはＤＯＷＮ の状態として再び表示された。サイト職員はＢＢＮ と連絡をとり，テストの結果いかんでは，ハネウェルや通信会社に装置の障害を通知した。

C. 初期のＮＣＣ 機能

ネットワークの５番目のＩＭＰ は，１９７０年の始めに，ＢＢＮ に設置され，その後しばらくして，テストと診断がかなり集中化されるようになった。

定期的な報告をＢＢＮ のＩＭＰ につながれたコンソール・テレタイプに送ってくるように，"トラブル・レポート"統計プログラムが各ＩＭＰ 内に取り付けられた。ＩＭＰ の状態と回線の状態のデータは，テレタイプにタイプ・アウトできるようにＡＳＣＩＩテキストとして送出された。

ＩＭＰ のハードウェアとソフトウェアの開発チームのメンバーは，時折このタイプ・アウトされたものを入念に調べ，必要な時には適当な行動を起こすべき責任を負っていた。もっぱら個人がテレタイプの監視に割り当てられることはなく，職員は通常正規の勤務時間の間だけしか役に立たなかったので，回線やＩＭＰ が，障害通知や修正作業の開始なしに長い間放置されることも起こりえた。しかし，ネットワークは非常に小型であり，そのオペレーションが大部分のホストの活動にとって決定的なものではなかったため，このことはたいした問題にならなかった。

D. IMP ソフトウェアの分散

IMPは平均月1台の割り合いで追加され続け，１９７０年末にはネットワークに１３台のIMP が取り付けられていた。IMP はハネウエル５１６をベースに，全くコンパチブルで，高速紙テープ・リーダを装備していた。新IMP ソフトウェアの譲渡は，最初紙テープの分配とほとんど同じ様式でIMP に再ローディングしたため，サイト職員の信頼にその基礎を置いていた。ソフトウェアは，他のプロジェクト用に数年前BBN で開発されたマクロ・アッセンブリ言語で書かれ，この言語用のクロス・アッセンブラを使ってPDP －１でアッセンブルされた。

１台のプロトタイプIMP がBBN でソフトウェア検査に役立っていた。このマシンで，ソフトウェアがかなり確実に作動していることが明らかになった後，プログラマは全ネットワークにソフトウェア用テープを分配した。このソフトウェア設置の後，必要なデバッキングが，サイト職員の多大な協力と，BBN からのIMP DDT の広範な利用によって，オンラインで実行された。他の協力を必要とするような計算（例えば，ルーティングの計算）を伴ったIMP プログラムを，現場に設置する前に，BBN にある１台のオンライン用マシンだけで充分に検査することは困難であった。従って，新IMP ソフトウェアの設置後，ネットワークの信頼性が全く乏しいという期間が存在した。

最初から，各IMP は，各マシンやそのローダ自身に固有の定数を含む小領域を除いて，DDT やコンソール・スイッチによるコマンドで，そのコア・メモリの再ロードを隣接するIMP から獲得するという機能を持っていた。また，監視タイマもこのメカニズムに使用された。

以後の新ソフトウェア譲渡の手順は，各サイトにそのIMP の定数やローダ領域だけのテープを郵送しておき，前もって決めておいた時間に各サイトはそのIMP にこのテープをロードすることであった。IMP プログラムの新版全部が，高速紙テープ・リーダによってBBNのIMPに読み込まれ，BBN に隣接するサイトは，BBNのIMPから再ローディングを行なうようサイトのIMP に指示する。これがうまく行くと，その再ローディングされたサイトから隣接したサイトへと，次々に新ソフトウェアがネットワーク全体に行き渡る迄再ローディングが行なわれる。

E. ＮＣＣホストコンピュータ

ネットワークがその大きさにおいて拡大し続けるに従がい，人間の介在を必要とするネットワーク事象についてBBN のテレタプに印字されたものを調査することは，次第に困難となってきた。加えて，IMP や回線の性能，アプローチの飽和の進捗状況を得るための回線利用の統計，ネットワーク利用についてのアカウンティング・アルゴリズムを調査するため

のホスト・トラフィックといったものに関する定期的な報告に重要な興味が存在した。これらの要因によって，人間の介在を必要とする事象を調べるのはもちろんのこと，ネットワークの性能を監視し，報告に必要な多くの記録を行なうことに専念しているＢＢＮ のＩＭＰに，１台のホストを付けることになった。ＮＣＣ ホストとして知られるこのホストのために選択されたコンピュータは，ネットワークＩＭＰ と同一のハネウェル３１６コンピュータでホストとモデムのインターフェイス，１２Ｋの１６ビット，メモリ，そしてリアル・タイム・クロックを持っていた。この種のハードウェアを選んだということは，ＮＣＣホストの障害時には，プロトタイプＩＭＰが一時的に代用されうるということを意味していた。ＮＣＣホスト自身とは別にパッケージ化されている一連の周辺装置インターフェースが，テレタイプ，ディスプレイ・ライト，そしてオペレータ・スイッチなどをドライブするために提供された。そしてこの装置は，必要な方のＣＰＵのＩ／Ｏバスに容易に接続されることができた。

１９７１年中頃迄に，ＮＣＣホストはオペレーション可能となっていた。従がって，各ＩＭＰ によって発生される"トラブル・レポート"のフォーマットに幾らか変更を加えることができた。まず最初に，バイナリ・エンコーディングのために，ＡＳＣＩＩテキスト・フォーマットを取り止めた。次に，ホスト・トラフィックや回線利用にについての統計はもちろん，更に多くの内部ステータス情報を含むようにレポートを拡張した。また，１分間に１度のチェックを行ない，変化が検知されるとすぐに変化の通知を送出するよう，レポーティングの頻度を増やすことができた。

ＮＣＣホストは，これらのレポートを調べ，ネットワーク状態の変化を認めて，この時までに週５日，１日１２時間，オペレータ作業に専念するまでに成長していたオペレーティング・スタッフに警告を発する仕事を与えられた。ＮＣＣホストは，そのホストにつながれたテレタイプへのプリント出力や，ＩＭＰあるいは回線の障害ではライト・ボックス上のライト（各ＩＭＰや回線用に別のライトがある）を点滅させ警報を鳴らすことによって，ネットワーク状態の変化をオペレータに通知した。また，ＮＣＣ ホストは，回線利用やホスト・トラフィックの統計を蓄積し，別のタイプライタにそれらの要約を印字した。

Ｆ．保守活動の集中化

１９７２年の中頃までには，ＮＣＣ は，週７日，１日２４時間のシングル・オペレータを用意するまでに拡張されていた。この頃，作業時間内にはＮＣＣ におり，就業時間後には家にいて容易に手に入れうるハードウェアとソフトウェアの開発スタッフのメンバーによるバック・アップを伴って，最初の責任体制がオペレータに割り当てられた。ＮＣＣ の専用電話が取り付けられ，全サイトの代表，ネットワーク回線に責任を持つキャリア・オフィスの代表，そしてハネウェル・メンテナンス・オフイス （ＩＭＰハードウェアの修理に

第１の責任を持っていた）の代表が，ＮＣＣのオペレータを介してＢＢＮと接触をとるよう指令を受けた。また，オペレーティング・スタフは，新サイトや新回線の設置をスケジューリングしたり，ネットワークの結合性をそこなうことのないようにルーチンの保守活動を調整するといったような多くの周辺的な問題についての責任を引き受けた。通信会社やハネウェルの修理オフィスの現地保守員は，個々のサイトからではなくＮＣＣ からだけ保守依頼を受けるように指令を受けた。というのは，ＮＣＣだけがネットワークのグローバルな構成を知っており，１つの領域における問題が別の領域の問題にどのように関係するかを理解することができたからである。

G. ＩＭＰハードウェアの診断

一般に，特定のＩＭＰ におけるコードの化けはそのＩＭＰ にだけ影響を及ぼすであろう。しかしながら，ネットワークのルーティング情報の計算は全てのＩＭＰ が等しく分かち合う分散した作業であり，ルーティング・コードの計算における障害はネットワーク規模の大きな損害をもたらす可能性がある。例えば，１９７３年に，ハーバードのＩＭＰのルーティング情報を計算したコードがメモリ障害によって破壊されるという事件が起った。この障害は，ハーバードが全隣接ノードに対し，ハーバードへ向うのが全着信地へのベスト・ルートであると通知するという性質のものであった。これにより，ハーバードの近所にいた全トラフィックがハーバードに流れ込み，ついには捨てられてしまうといった事態になった。

上述のエラー（Ｉ／Ｏチャネル・エラーとメモリ障害）を検知するために，ＩＭＰ プログラムは，各パケットと全ての重要な領域にあるＩＭＰコードに対するソフトウェアによるチェックを組み込むように変更された。もしソフトウェア・チェック・エラーが見つけられると，そのＩＭＰ のソフトウェアかハードウェアに障害があると推定され，エラーを起こした情報のコピーがＮＣＣ（ＰＤＰ－１）に送られた。更に，自身のコードにソフトウェア・チェック・エラーを検知したＩＭＰ や，稼動している回線上で２００回以上の再送を必要としたＩＭＰは，ダメージがその１台のＩＭＰ でくい止められるように，直ちに再ロードが行なわれた。

また，ソフトウェアによるチェックではカバーしきれない問題から，障害をマシンの特定の部分に隔離しようとする別の方法を捜さねばならなかった。このようなツールの１つに，ＩＭＰ 検査プログラムがある。（これもＰＤＰ－１上を走っている。）これは，稼動しているＩＮＰのコピーを取り，それとＰＤＰ－１の大容量記憶装置にストアされているメモリ・イメージとを１ワードづつ比較するものである。このツールは様々の目的に使用されるが，特に与えられたＩＭＰ メモリの一部のビットが化けている疑いのある時に使用され，そうした障害発生の診断と完全な修復に役立った。

H．ＴＩＰサービス

ネットワークに約２０のＴＩＰが設置されていた１９７３年頃の終りに，ＴＩＰとネットワークの信頼性に対するＴＩＰユーザ共同体の認識が，ＴＩＰやネットワークについての問題によって説明され非常に貧しいものであることが，ＡＲＰＡ とＮＣＣにとって明白となった。ＴＩＰが物理的に設置されているサイトにいないユーザの間では，特に問題が激しいように感じられた。

ユーザは，問題の原因が彼のターミナルとＴＩＰ間の回線，ＴＩＰのモデム，モデムへのＴＩＰのインターフェイス装置，ＴＩＰ自身，ネットワーク・ユーザがアクセスしようとしているホストのいずれにあるのかといったことには，特に興味がなかった。つまる所，ユーザはこうしたパス上のどのような要素に対しても制御権を持っていなかったのである。必要なのは，ユーザがトラブルを報告し，援助を得ることができるようなコンタクト・ポイントであった。障害の原因となり得る非常に多くの要素に対して制御権を持っており，ネットワーク中で唯一つ便利で連続してスタッフを配しているポイントは，ＮＣＣであった。ＮＣＣの責任はターミナルＩＭＰのインターフェイス装置までであったが，一部のユーザ共同体の不満がかさむことにより，ＮＣＣはユーザとユーザがサービスを受けようとしているコンピュータ間のパスにおける，付加的な要素についても責任を持つことに同意した。

- コンサルティング業務
- ＴＩＰ接続モデムの定期的な遠隔診断

I．品質保証プログラム

１９７３年末までには，非常に多くのグループが規則正しい方法で，生産施設としてネットワークを利用していた。それに先立つ２年程前に，ＩＭＰは，ハードウェアやソフトウェアの障害のため，平均して１，1.5，３％といった平均ダウン時間を持っていた。しかしながら，１９７３年末にこの充分低いダウン時間の平均レベルをさらに減少させようという要求が増大した。そこで１９７４年初めに，信頼性のレベルを９９.５％以上にしようという試みについて幾つかのステップを持った。

最初の重要な変化は，ＢＢＮの技術ステッフの＂工場＂としての責務から，保守およびハードウェアの品質管理の責任を切り離したことであった。

第２のタイプの分離は，プログラマではなくＮＣＣのオペレータによって実行されるソフトウェア管理手順の設置であった。

表５－１　ＩＭＰダウンタイムの減少

Quarter	Average Percent Down
3rd, 1972	1.21
4th, 1972	1.80
1st, 1973	1.95
2nd, 1973	1.07
3rd, 1973	2.09
4th, 1973	1.51
1st, 1974	.80
2nd, 1974	.85
3rd, 1974	.57
4th, 1974	.26
1st, 1975	.32
2nd, 1975	.31

Ｊ．新ＮＣＣホストの機能

前述のＮＣＣトラブルレポートに以下の報告事項が追加された。

- ネットワーク中の通信回線の状態とエラー頻度
- ネットワーク中の各ホストのｕｐ／ｄｏｗｎの変化
- ＩＭＰ内の各キューの長さ
- ＩＭＰの種々のコンソール・スイッチの位置
- 各種トラップ

トラップ・メカニズムは，特定のブロックのコードが実行される時にいつでも，プログラマや保守員がＮＣＣにＩＭＰの状態を通知することを可能にしている。１つのトラップ・サブルーチンが各トラップ状態についての，ＩＭＰのプログラム・カウンタ，アキュムレータの内容，そしてインデックス・レジスタの内容を報告する。このようなメカニズムは，ＩＭＰ内の様々なブロックのコードの実行頻度を測定するのに便利である。トラップ・メカニズムは，ソフトウェア・チェックの障害やハードウェアの障害に関連した他の故障を報告するために利用される。この件に関しては，１９７４年の終りに，全時間保守作業に専念するようなシステム・プログラマがＮＣＣスタッフに追加された。他のツールとして，このプログラマは，特定の障害を起こしているコンポーネントを切り離す際にハードウェア上の困難に遭

遇するであろうと思われるＩＭＰにトラップをかけたり，疑わしいソフトウェア・バグを調べるため全ＩＭＰにトラップをかけるような時に，ＤＤＴを利用することができる。

- 診断情報の収集

  コア・ダンプをあるホストに送出

K。将来の問題

１９７５年の春に，ネットワークは約６０のノードを持つまでに成長した。ＮＣＣの歴史の全時点におけると同じように，一連の新しい問題が先に見えている。これらは，ＮＣＣによるある種の"ネットワーク"ホスト（"プライベート・ライン・インターフェイス"ホストのような）のオペレーションについての責務，ハネウェルをベースとしたＩＭＰとプログラム的にコンパチブルでないＰluribus：ＩＭＰ用の新回線の設置，そして，ＩＭＰがこのチャネルを介してのみアクセスできるようなマルチ・アクセス・ブロードキャスト・サテライト・チャネルの診断用に新しい手順を必要とするような"サテライトＩＭＰ"の導入などを含んでいる。

5．2　ネットワークにおけるセキュリティ（文献２より引用）

不特定多数のユーザが多目的に利用する可能性の強いコンピュータ・ネットワークでは，単一システム以上のセキュリティに対する配慮が必要となる。

ここではセキュリティを機密保護という観点でとらえ，侵害行為を列挙した後，それらに対する対策を紹介する。

5.2.1　セキュリティ

情報化社会の進展に伴い情報処理の量的拡大，多目的化，広域化，高度利用等は益々発展的である。それと共に情報処理システムの効率向上もさることながら，コンピュータ・システムおよびコンピュータ・ネットワークの安全性や情報の機密保護等セキュリティの問題も非常に重要なものとして認識されるようになってきた。

セキュリティという言葉は，さまざまな定義で使われている。プライバシーの法的保護の問題まで対称とした広い意味で使われる場合もあるが，一般的にプライバシーの問題はセキュリティと区別している。

セキュリティは，更に広義と狭義に分けて使われている。

①　セキュリティ（広義）

セキュリティとは，ハードウェア，ソフトウェアおよびデータを適当な手段を講じて，破壊や侵害から守ることである。

② セキュリティ（狭義）

知る権利，使う権利の無い者からのアクセス侵害や破壊から保護することである（機密保護）。

③ インテグリティ

本来あるべき完全な状態を保持することである。偶発的なハードウェア障害，ソフトウェア障害，ソフトウェア障害からの破壊防止，破壊された場合のロギング，障害時の回復手段等を含む（安全性維持）。

広義のセキュリティは，狭義のセキュリティとインテグリティに分けて考えられている。

A．侵害行為

コンピュータ・システムやコンピュータ・ネットワークの安全性を脅す要因をセキュリティに関する故意によるものとインテグリティに関する偶発的なものに分類してみる。

(1) セキュリテイ関連

(i) システム担当者の不正侵入

システムに精通した技術者がソフトウェアを故意に改ざんし侵入を図る。

(ii) 使用済用紙等からの情報漏洩

廃棄してある出力用紙やカーボン紙等からの情報採取

(iii) 記録媒体の盗難

磁気テープ，磁気ディスク・パック，カード，紙テープ等運搬可能な記録媒体を盗む。

(iv) 回線からの盗聴

通信回線から信号を傍受する。

(v) システム障害箇所からの侵入

ソフトウェアのバグ，ハードウェアの障害箇所等を利用して侵入する。

(vi) 破壊活動

第三者による外部からの破壊活動

(vii) 第三者の不正侵入

他人のパスワードやユーザ番号を使用し，不正に情報を入手する。

(2) インテグリティ関連

(i) システムの障害

ハードウェア障害，ソフトウェアのバグ，通信回線障害，端末装置障害など。

(ii) 使用者の不注意

オペレータの操作ミス，アプリケーション・プログラム，コマンド指定のミス等の不注意な行為。

(iii) コンピュータ関連設備の故障

電源設備，空調設備等の機械の故障など。

(iv) 天災

地震，火災，風雨等の天災地変。

また，侵害行為の対象から分類すると次のようなものが考えられる。

(i) 端末装置，通信回線，コンピュータ等のハードウェアに関するもの。

(ii) オペレーティング・システム，ユーザ・プログラム等のソフトウェアに関するもの。

(iii) プログラム，オペレータ，管理者，ユーザ等の人に関するもの。

(iv) ネットワーク・システム，データ・ベース・システム等ハードウェア，ソフトウェア，人の有機的組合わせによるもシステム全体に関するもの。

B．セキュリティ対策

システムに対する暴露，改ざん，破壊，妨害等の侵害行為におけるセキュリティ対策は一般に次のことを可能にする必要がある。

① これらの行為が公認された方法以外でのアクセスを禁止する。

② これらの行為の記録がとれる。

③ これらの行為の源泉が追求できる。

④ これらの行為を行なった者を排除できるシステムにする。

このようなセキュリティ対策について，段階的に大きく２つに分類して考察する。

(1) コンピュータ・システムのセキュリティ

バッチ処理の形態にみられる機能中心のシステムにおけるデータ・ファイルやコンピュータの運用の安全性対策。更にデータ・ベース・システム等のファイル中心のシステムの安全性対策も含まれる。

(2) コンピュータ・ネットワークのセキュリティ

コンピュータに通信回線を通して遠隔地に端末を持つオンライン・システムやＴＳＳの形態等のシステムにみられるリモート・ターミナル・システムでの安全性対策。更に，それらの発展形態として，ファイル，処理プロセッサ，通信回線，ターミナル等の有機的複合体としてのコンピュータ・ネットワークでの安全性対策。

5.2.2 コンピュータ・ネットワークのセキュリティ

コンピュータ・ネットワークのセキュリティ対策を考察するにあたり，従来のオンライン・システムやＴＳＳで代表されるマルチ・ターミナル・システムにおいては，コンピュータ・システムから通信回線を通して目の届かない遠隔地に端末が置かれている。このような遠隔端末に

対するセキュリティ対策を考える必要がある。（図５－１参照）

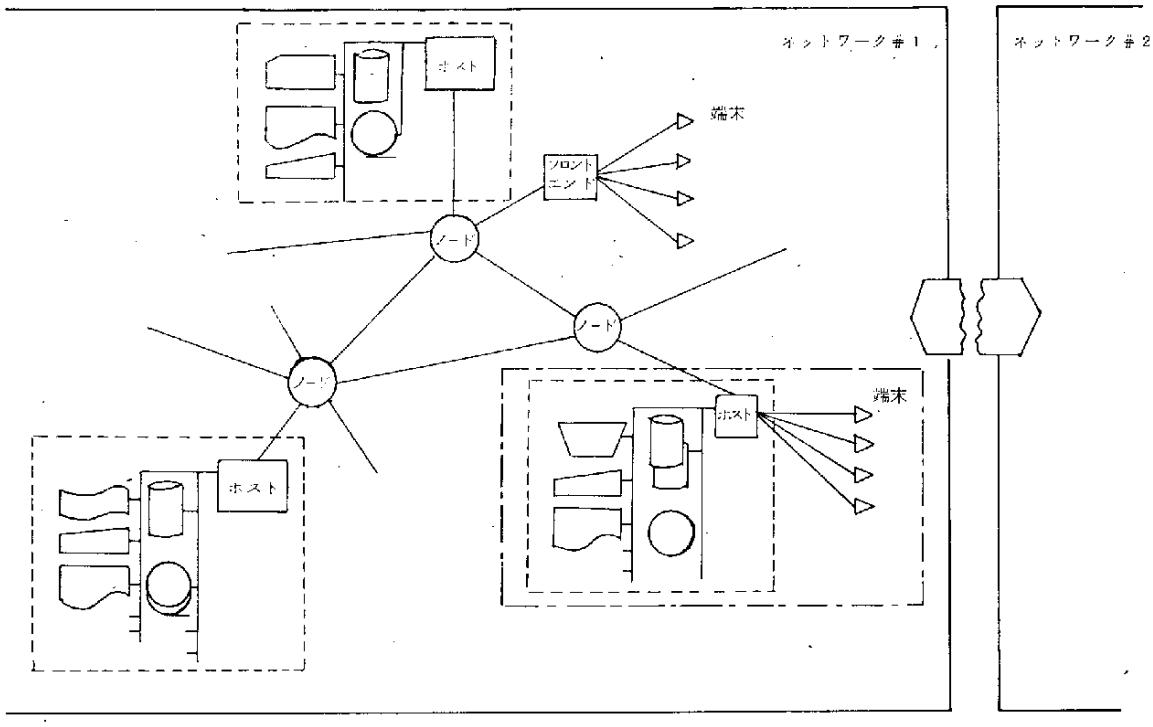


図５－１　コンピュータ・ネットワークとセキュリティ

Ａ．リモート・ターミナルのセキュリティ

リモート・ターミナルにおける管理的対策はコンピュータ・システムの場合と合せ，更に次のようなことが考えられる。

(1)　監　視

警備員による監視，ロギングの分析など。

(2)　端末利用者の身許調査，信用調査など。

次に技術的対策としては，次のようなことが考えられる。

(1)　通信回線からの漏洩防止

電気的，磁気的な結合による盗聴を防ぐのは困難である。しかし，暗号化／符号化したデータを伝送することにより漏洩したデータを無意味化し防止を可能にする。

通信の安全性を保つために暗号化（encryption）の方法として次のものがある。

(i)　情報の列の文字位置を一定の規則で入れかえる方法（transposition）

(ii)　各文字を一定の規則または表で別の文字にさしかえる方法（substitution）

(iii)　文字列と任意のビット列との論理演算を施し暗号化する方法(additive encoding)

(2)　ユーザＩＤの保護

ユーザのＩＤを頻繁に変更することにより他人に対し，ユーザ固有のＩＤ（パスワード，キー等）を判りにくくする。

B。ネットワークのセキュリティ

ネットワーク上のセキュリティの問題は，コンピュータ・システムとリモート・ターミナルのセキュリティ対策の拡張と考えられる。

遠隔端末は，インテリジェント・ターミナルあるいは他のホスト・システムに置き換えられると，他のホスト・システムの端末からのアクセスも管理する必要が起きてくる。このように，マルチ・ターミナル・システムからインテリジェント・ターミナル・システムへ，インテリジェント・ターミナルからコンピュータ・ネットワーク・システムへと至ると，データ・セキュリティの問題は，その機能上益々複雑化してくる。

しかし，コンピュータ・ネットワークにおいては，セキュリティ機能をネットワーク内の何故に，どのように持たせることが，機能的，経済的に有効かということがまず重要な問題としてとらえられる。

例えば，次のようなことがあげられる。

(1) ネットワークを通る利用者の資格は，ネットワーク制御コンピュータで集中してチェックする方法。

(2) ネットワークの利用者の資格は，ホストとコミュニケーション・サブネットワーク間にチェック機構を持つ処理装置を持ち，この装置間で分散管理し，チェックする方法。

(3) ファイルのアクセス・チェックは，各ホストに分散して持たせる方法。

(4) ネットワークが異種のコンピュータで構成されている場合，ホストと通信ネットワークとのインターフェイスをとるためにコミュニケーション・プロセッサを導入し，通信レベルのプロトコルの同期を計り，同時に通信上のセキュリティ対策も合せて行なう方法。

次に，重要な問題の１つである通信上の安全性に注目してみる。

通信上の安全性の確保（即ち，情報の漏洩に対する対策）としては，暗号化による方法や伝送方法がある。その具体的対策には，次のようなことが考えられる。

(1) 通信回線からの漏洩防止

(i) 暗号化による方法

リモート，ターミナルの場合の暗号化による方法と同じである。

(ii) パケット交換による方法

メッセージをパケット化することによりメッセージ全体の把握を困難にする。

(a) メッセージの単純なパケット化による方法。

(b) １つのメッセージを構成するパケットの順序を入れ替える方法。

(c) 1つのメッセージ内のパケットのルートを変更する方法。

(iii) メッセージの宛先名を頻繁に変更する方法。

メッセージ宛先名を頻繁に変更することにより，第三者がネットワーク内を伝送中のメッセージを見ても，誰と誰が交信しているか判らなくする。

ネットワーク・セキュリティの問題は，主としてホスト・システムと通信回線におけるセキュリティの問題に帰着されるであろう。また，ネットワーク・システムも実用化段階に入ったばかりであり，セキュリティ対策の重要性が深く認識されながらも，現存のシステムでは充分に配慮されていない状態にある。従がって，現段階ではどのシステムにどの程度のセキュリティ対策が必要か，処理効率の低減と費用の負担によるコスト効率を勘案し模索しているとみてよい。

また，ネットワークの汎用化，広域化の志向，ネットワーク間の結合，無線・衛星通信方式の導入，データ・ベースの分散化等コンピュータリゼーションの発展と共に，経済上，法律上，社会上等技術的でない問題も絡み，セキュリティ問題は益々重要性を増し，その対策は複雑且つ困難になって行くであろう。

C．安全性対策へのアプローチ

(1) 設計レベルでの安全性対策の考察。Frank R. Heinrich は，システム設計レベルにおけるネットワーク・セキュリティのアプローチについて考察している。安全なネットワークの設計は，集中的キー管理をベースとした制御原理と集中管理を基礎に置いている。これは，ネットワーク構造にネットワーク暗号化装置（Network Cryptographic Device）と呼ぶ保護装置とネットワーク・セキュリティ・センター（Network Security Center；NSC）と呼ぶ専用の処理装置を導入することで目的を達成している。

(A) ネットワークの概念

コンピュータ・ネットワークでは，多数のホストと端末が連結され，ネットワーク・リソースと呼ばれている。これらのリソースの相互連絡は，ハードウェアとソフトウェアの機能から成立っている。ネットワークの機能は論理的な面から3つの層に分割して考えられる。（図5－2参照）

(i) 層1 ネットワーク・リソース

(ii) 層2 連結指向の機能

(iii) 層3 コミュニケーション・サブネットワーク

このようなネットワークの機能において，ネットワーク・セキュリティの脅威を回避するために三層間の論理的関係を変更するのでなく，新しい機能層を追加することによ

り解決を図ろうとするものである。

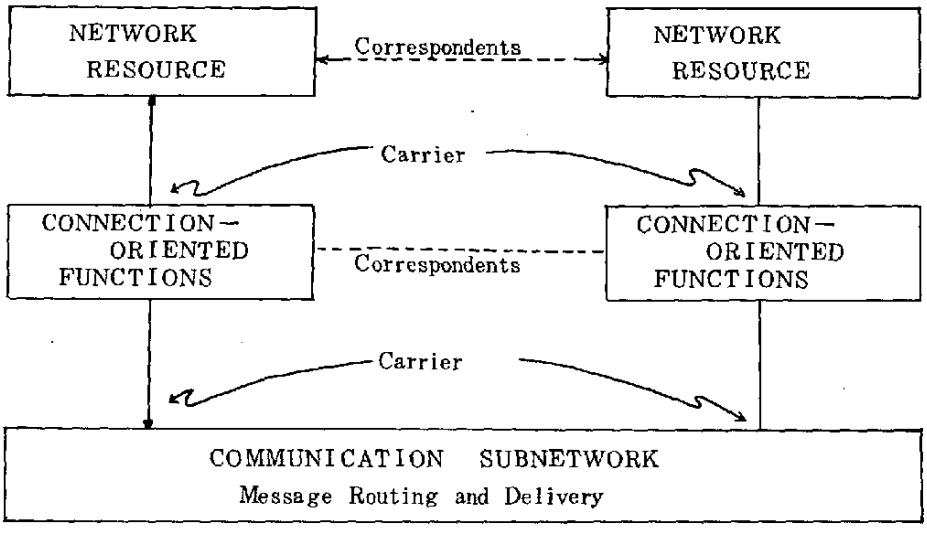


図５－２　Layers of network functionality

(E)　システム構成

完全なインタコンピュータ・ネットワークのシステムの設計は図５－３に示される。

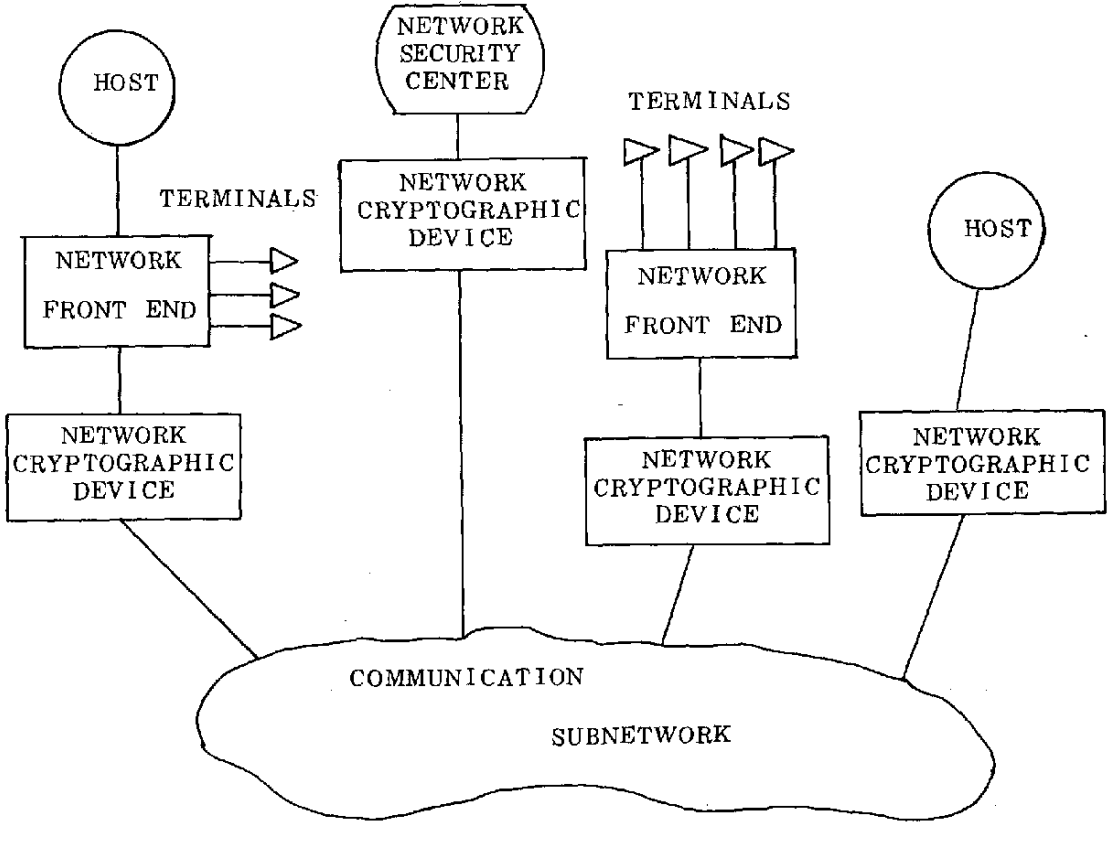


図５－３　System level design

暗号装置（cryptographic devices）は，データの暗号化と解読を行ない，この変換は秘密のパラメータ・キーで実行される。これが，ネットワークの論理的構造に附加される層である。

ネットワーク・セキュリティ・センタは，新しいネットワーク・リソースである。リソース間の相互連絡は，アクセス制御情報によってＮＳＣで公認された時だけ許される。

ネットワーク・フロント・エンド（ＮＦＥ）は，ホストや端末に対し連結指向の機能を遂行するための処理装置である。

(C) 暗号装置

暗号装置には２つのタイプがある。

(i) マスタ暗号装置（ＭＣＤ）

ＮＳＣにおける暗号装置である。ＭＣＤは各々のＳＣＤ間のメッセージの暗号化と解読が可能であり，ＳＣＤに対し新しいキーの設定を管理する。

(ii) スレーブ暗号装置（ＳＣＤ）

ＮＳＣ以外のネットワーク・リソースにそれぞれ結合される。ＳＣＤは遠隔地より新じいキーを受け入れる。１つの端末に１つの新しいキーを保持し，ホストやＮＦＥの場合は，重複した論理的結合をサポートするため，複数の新しいキーを保持する必要がある。

(D) ネットワーク・セキュリティ・センター（ＮＳＣ）

ＮＳＣは，ネットワーク・ユーザの識別を有効にし，ネットワーク・リソース間の連結を公認する。そして，アクセス要求の公認後，サブジェクトに対し，別個のランダムな暗号キーを生成する。また，全てのアクセス要求の情報を記録し，不正な侵入者の企てを検知すると警報を発する。

従がって，ＮＳＣはユーザＩＤを立証する充分な情報を含むデータ・ベースと，アクセス要求の正当性を決定できる充分なアクセス制御情報を維持する必要がある。

データ・ベースは，折よく動的に更新され，この更新は，ＮＳＣのセキュリティ担当者かＮＳＣとネットワーク・ホスト間のプロトコルで行なわれる。

ＮＳＣアクセス制御情報は，３次元のアクセス制御マトリックスの概念的なモデルで実現され，その要素は次のように定義される。

(i) サブジェクト（Subject）

アクセス要求を始めることのできるユーザやプロセスのような実体（Entity）。

(ii) オブジェクト（Object）

アクセス要求の目標となるデータ・ファイル，プロセス，ホスト・コンピュータ・システム，他のネットワーク・リソースのような実体。

(iii) ケィパビリティ（Capabilities）

サブジェクトがオブジェクト上で遂行するアクション。

(E) ネットワーク・フロント・エンド

ＮＦＥはネットワーク・リソースとコミュニケーション・サブネットワークとのインターフェイスであり，ホストや端末に代ってコネクションを指向した機能を実行する。また，端末に対しユーザ・レベルのコマンド・インターフェイスでもある。ＮＦＥはソフトウェア費用とシステム負荷を軽減し，安全なフロント・エンドはネットワーク・セキュリティを高める。

(F) 考　察

このネットワーク構造は，ネットワーク・セキュリティの脆弱性を非常に縮少させている。

ＮＳＣは，合法的なユーザだけがネットワーク・リソースにアクセス可能であり，公認されたアクセス要求を許可されるよう保証するための安全なネットワーク・ファシリティを用意している。

ＭＣＤは，事実上ネットワーク通信に対するセキュリティの脅威を排除し，ネットワーク・リソースの公認を助成している。しかしながら，現在利用可能な暗号装置は，これに適合した特性を持っていないが，現在の技術では充分製作可能である。

そこで，コスト効率の高いセキュリティの保証が，現在の技術によってネットワークにも実現可能になる。

一方，分散型キー管理に基いたネットワーク・セキュリティへの効果的アプローチがある。これは，集中制御方式が法的，政策，司法，信頼性，実際的な制約で不可能なとき有効である。この方式では全ての暗号装置が同じであるが，他の暗号装置に対しキーの生成や中継をする機能が一層複雑になる。ＭＣＤは排除され，ＮＳＣはオプションになる。

(2) 通信上の安全性対策の一手法

通信上の侵入行為（情報の漏洩や偽造など）に対する対策の一案として，動的なプロセスの名前付け（Dynamic Process Renaming）により通信上の安全性を達成しようとする方法が提案されている。

ネットワークの安全性の確保は，ホストとネットワークのインターフェイスを司どるコミュニケーション・プロセッサ（ＣＰ）に集中して行なわれるべきであると指摘している。

カリフォルニア大学で開発された分散型コンピュータ，システム（Distributed Computer System；DCS）では，ソフトウェア・システムの一部であるメッセージ交換モジュール（これはメッセージを各々のホスト内で内部的に径路選択する）が考えられ

ている。

これは，自分が結合されているホスト内に存在する全プロセスのリストを内蔵しており，メッセージが巡回してくるとそのリストと照合される。メッセージの宛先が，そのリストの要素と一致すると，そのメッセージはコピーされ，ホストに渡される。不一致のときはそのメッセージを全ネットワークのどこかに存在する宛先プロセスに向けて送出する。

このようにして，安全かつ高度に一様化されたメッセージの分配機構を実現している。

そして，このような機構をホストとは独立した自律的な要素（ＣＰ）に持たせることが，より効果的としている。それは，異なる目的に使用されるホストに対し，ＣＰの機能はほぼ一様であると考えられるためである。

このような分配機構を安全性を考慮し発展させた安全性機構は，基本的には次のようなものである。

交互に通信を行なっているプロセスの名前を動的に変化させることができ，しかも名前の変更が充分頻繁に行なわれることにより，

(i) 観察者が，伝送中のメッセージを見ても，誰と誰が交信しているか分らず，継続した意味のある知覚が得られない（情報漏洩防止）。

(ii) 侵入者に対して，今送信しているプロセスが次にどの様な宛先で返事を貰うのか分らず，侵入者が偽の情報を流すことが，極めて困難である（情報偽造防止）。

次に，動的なプロセスの名前付けにより必要な安全性を達成するプロトコルについて説明する。

このプロトコルは，適当に安全なＣＰ，非常に脆弱な通信路から成るネットワーク内で動作することを目的として作成されている。

安全性を考慮したプロトコル（Security Protocol）の基本は，"名前"を順番に生成していくことである。

通信路が２つのサイト間で確立されるとき，各サイトのＣＰは"名前"列生成に使用されるアルゴリズムの合意のもとに1つ定められる。各メッセージにはメッセージ同期フィールド（Message Synchronous Field ；ＭＳＦ，"名前"列生成のためのアルゴリズムに適用される）が含まれている。

プロセスＡが，プロセスＢと通信したい場合，Ａは自分のＣＰ１にその旨を知らせる。そのＣＰ１はＡＢ間に通信路を確立する。そして，Ａより指定されたサイトのＣＰ２に向けて，パラメータ，ＭＳＦ，プロセス名Ｂを送り出す。これを受けたＣＰ２は，パラメータによりアルゴリズムを決定する。Ａにメッセージが返送されるとき，最初のメッセージ中のＭＳＦがアルゴリズムによって変換され宛先名として使われる。更に，このメッセージでは，ＡサイトのＣＰ１がＢへの返送メッセージに使用するＭＳＦが同様に

挿入されるというものである。（図５－４）

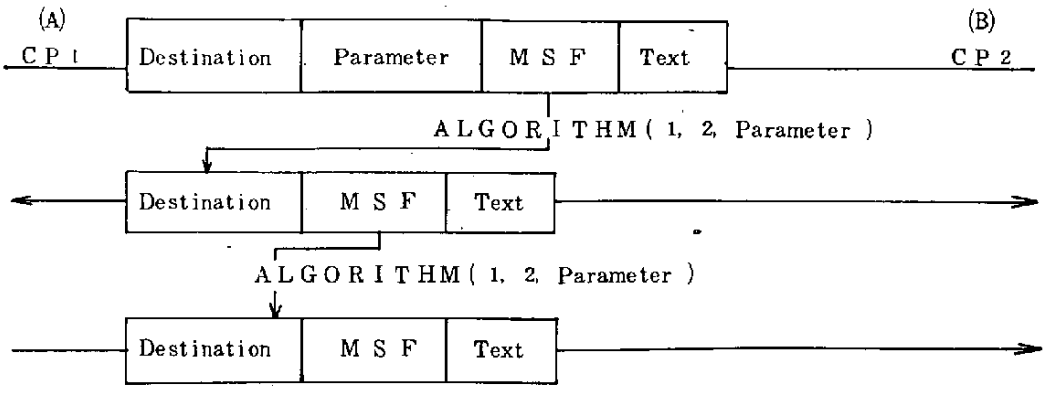


図５－４　プロセスＡとＢの通信

(3)　ＯＣＴＰＵＳのセキュリティ対策

ＯＣＴＰＵＳは，カリフォルニアの原子力研究所ＬＬＬ（Lawrence Livermore Lab）にある信頼性を極度に追求したコンピュータ・コンプレックス型のネットワークである。

ＯＣＴＰＵＳの特徴は，次の諸点に要約される。

(i)　高度な信頼性の追求

(ii)　ホストとターミナル間の自由なアクセス可能性

(iii)　特殊端末の利用可能性

(iv)　単一データ・ベースの実現性

(v)　サブシステムの集合体

ＯＣＴＰＵＳの設計目標を達成するため，セキュリティ対策として，その設計の根底をネットワークの安全なソフトウェアをベースに次のように考えている。

(i)　特別な公認によるシステムの変更。

(ii)　自己および他人と共有の情報以外でのアクセス禁止。

(iii)　情報を分類する階層構造の規則性保持

また，設計上具体的対策として次の機能を導入し解決を図っている。

(i)　ハードウェアの保護機能。

(ii)　ユーザ識別のパスワード。

(iii)　個人，共有，公用等を区別するファイル構造。

(4)　ＡＲＰＡＮＥＴの安全性問題

ＡＲＰＡＮＥＴが今後一般に公開する可能性を踏まえ，公共政策に関する問題，即ち，所

有権の管轄，ネットワーク間結合，プライバシーとセキュリティ，課金，税金，関税の問題が論じられている。

その可能性には，次のものが前提条件として考えられている。

(i) 全米政府に対してコンピュータ・ネットワークを拡大する。

(ii) ＤＯＤ（U. S. Department of Defense）による現状の運用を継続する。

(iii) 商用の付加価値キャリアに払い下げる。

(iv) 運用を中止する。

プライバシーとセキュリティ関連の問題では，最近までデータ・セキュリティはＡＲＰＡＮＥＴでは問題にならなかった。しかし，国防関係の利用者がＡＲＰＡＮＥＴやその関連技術に興味をもつようになり，データやＯＳの安全性が最優先になった。ＡＲＰＡＮＥＴが保護の標準を設定する場合，ネットワーク間結合が普及する前に標準を設定する必要がある。その場合，次の問題が当面関係してくる。

(i) セキュリティとプライバシーに対する責任と実行の権限。

(ii) ユーザのメッセージ・ルーティングに対する制御の可否。

(iii) 全てのネットワークがデータ・ベースと同様の保護標準を持つ必要性。

(iv) セキュリティ・ガードはゲートウェイの機能か，ホスト自身の問題か。

(v) ネットワーク間結合のセキュリティ維持のため我慢できるオーバヘッドの種類とその支払方法。

(vi) 異種のネットワークは異ったレベルのセキュリティを提供し且つ相互結合の可能性。

このような問題は一般に汎用ネットワークの構築および国内外のネットワーク間結合において考えられる重要なセキュリティ問題である。

(5) Ｓ．Ｗ．Ｉ．Ｆ．Ｔ．のセキュリティ

Ｓ．Ｗ．Ｉ．Ｆ．Ｔ．において，セキュリティは１９７５年４月１日の委員会の議事項目の重要なテーマであり，安全保護の原則は次の事項で同意されている。

① ログイン手順はＳ．Ｗ．Ｉ．Ｆ．Ｔ．システムの実行に委される。

② 入出力連続番号の使用はシステム運用に委される。

③ Ｓ．Ｗ．Ｉ．Ｆ．Ｔ．は国家のコンセントレータとユーザの銀行間の通信上の保護に責任を持たない。

④ 暗号装置は全ての主要な国際通信回線に設置される。

⑤ コンピュータ・ベースの端末を使用する銀行はＳ．Ｗ．Ｉ．Ｆ．Ｔ．で準備した標準認証を利用する。

⑥ コンピュータ・ベースの端末を使用しない銀行は，現在の実施に即して双方で同意したテスト・キーを使用し続けてもよい。

このようなセキュリティ政策を反映して，次の3項目の要求に関するセキュリティ手順が考えられている。

① 伝送エラーの検定と訂正。

② S．W．I．F．T．の運用における第三者の監視や干渉の防止を含むネットワークのプライバシー。

③ 送受信間での識別と認証。

①の手順は主としてライン・プロトコルで達成され，次のチェック・コードの使用も含まれている。

(i) VRC（Vertical Redundancy Check）

(ii) LRC（Longitudinal Redundancy Check）

(iii) CRC（Cyclic Redundaney Check）

他の手順はS．W．I．F．T．で検証するため入力連続番号の割付けによっている。

図5－5はプライバシーの方策を示す。メッセージはコンセントレータとスイッチング・センタ間でS．W．I．F．T．により暗号化される。また，センタではフォーマット・チェックのためメッセージの解読が必要であるが，メッセージは暗号形式で格納されている。

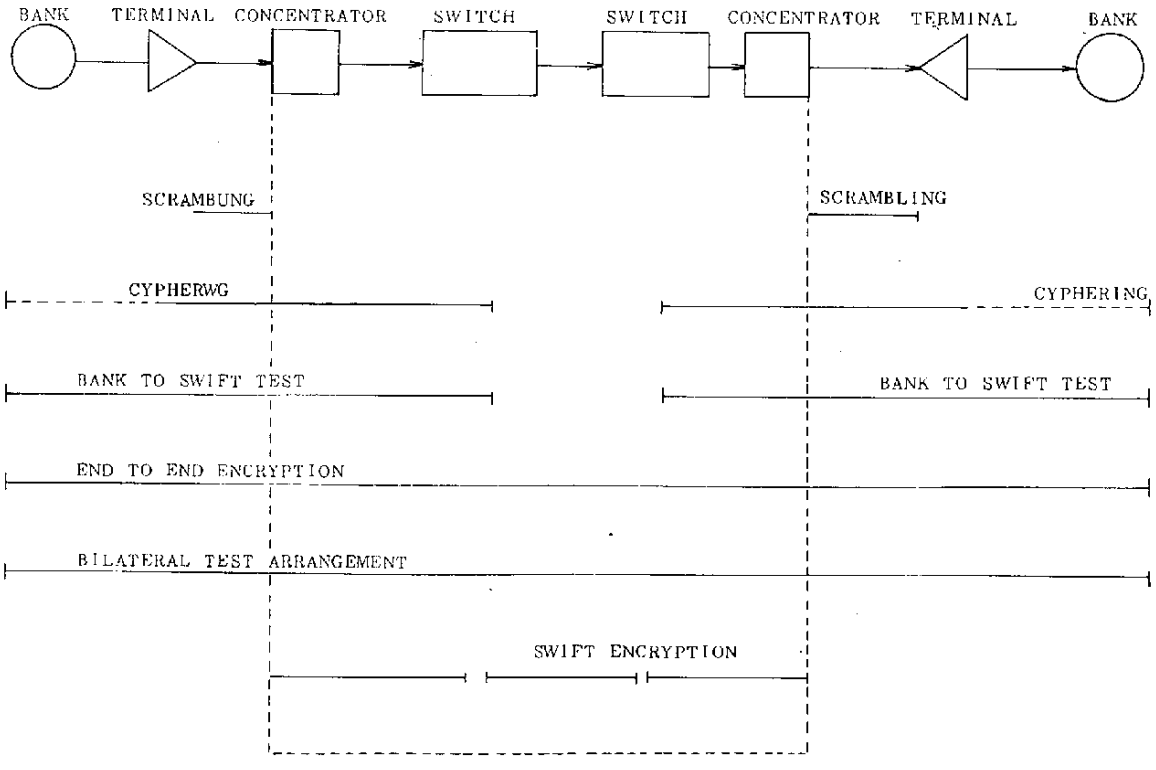


図5－5 SECURITY PROCEDURES

暗号化はトランクラインで監視を困難にし，干渉の防止を図っている。コンピュータ・ベースの端末利用者は端末とＳ．Ｗ．Ｉ．Ｆ．Ｔ．間の暗号を使用できる。テレプリンタやテレックス利用者は端末とコンセントレータ間の保護のため周波数帯変換装置が使用可能である。ユーザ間の暗号化は双方合意の取決めで可能であり，そのためメッセージはフリー・フォーマットとして処理されている。Ｓ．Ｗ．Ｉ．Ｆ．Ｔ．は純粋にキャリアの役を勤める。

送信者や受信者の識別や認証は，ログイン手順の使用と検証のための入力連続番号の割付けによっている。端末は各ユーザ個有のコードで識別された後各種の機能に対しシステムと結合される。

5．3　ネットワークの評価とその活用（文献３．４．６より引用）

ネットワークの規模が拡大するのに伴い，ネットワークが所定の性能を発揮しているかどうかを定期的に測定，評価し，ネットワーク性能の維持・改良に努めることが必要である。この事例としてＡＲＰＡＮＥＴの性能評価の例を紹介する。

また，ＮＢＳで開発されているネットワーク測定マシンと，その利用法等についても概要を紹介する。

コンピュータ・ネットワークは，諸々の分野の複雑で広範囲な技術を必要としており，まさに膨大な量の技術文献が発表されている。ネットワークの評価・技術に関する文献も例外でなく，いろいろな観点からいろいろな技術で評価が行なわれている。

例えば，

（経済性の観点）　ネットワーク・システムの顧客に対する適切なサービスの提示や経費・利益の把握技術

（開発上の観点）　ネットワークを開発する場合のサービス目標や設定や目標達成を把握する技術

（改善上の観点）　システム構成要素の有効利用を把握する技術

（運用上の観点）　システムの能力や機能を正しく把握する技術

などである。

文献３は，代表的なネットワーク・システムＡＲＰＡＮＥＴの性能評価の論文である。この論文では，シミュレーション・マシンＰＤＰ１１／４５を用い多重パケット・トラヒックをネットワーク（ＡＲＰＡＮＥＴ）に許される最大のスピードで１０分間送り込む方法で，達成可能な持続するスループットと遅延を定期的に測定している。

この手法で得られた結果として

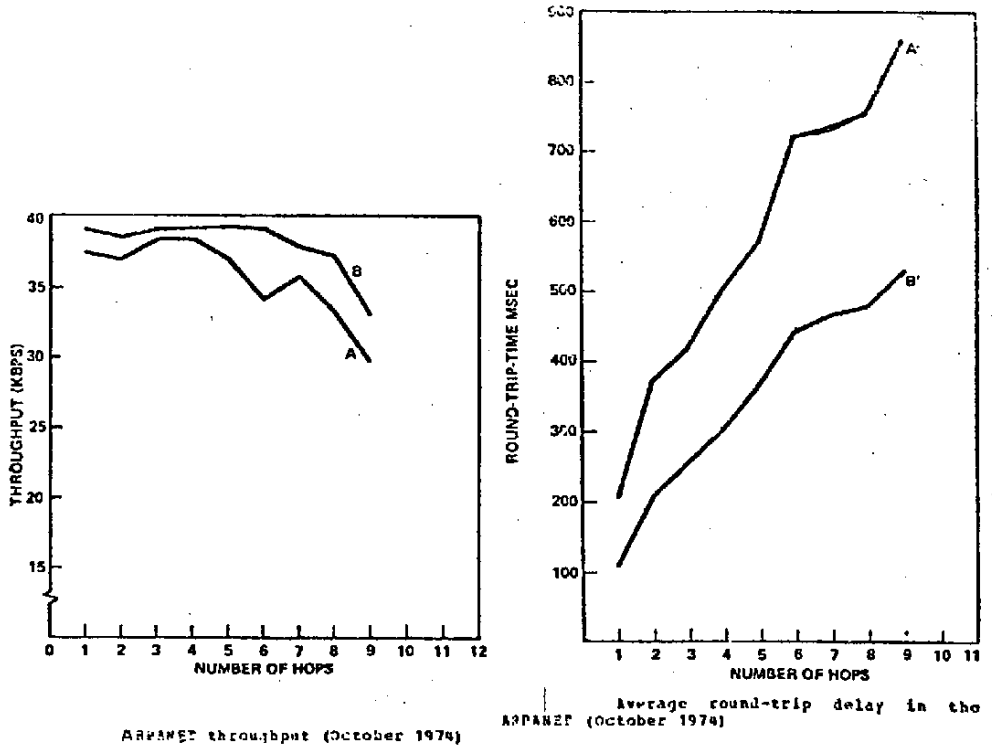
- cpuの速度によりかなり激しくスループットが低下する事実があるが，低下の多くは

高いレベルの効果のないプロトコルの採用による。

- 代替経路がスループットに与える効果は少ないが，ネットワークの信頼性の面で代替経路は非常に重要である。
- パケットが転々とネットワーク内を往き戻りして（ルーピング），時折非常に長い遅延が認められた。この防止のために回線ホールド・ダウン機構を設けた。
- ルーピングの問題も含め変更された新メッセージ処理によるスループットを測定した結果，以前に比べはるかに悪かった（５０％減少）。この原因は，処理遅延の増大と不必要な制御メッセージの送信によるものである。
- さらに改善を加え測定した結果，以前観測したものより劣るが，改善前よりはるかによい結果を得た。

などの報告がされている。

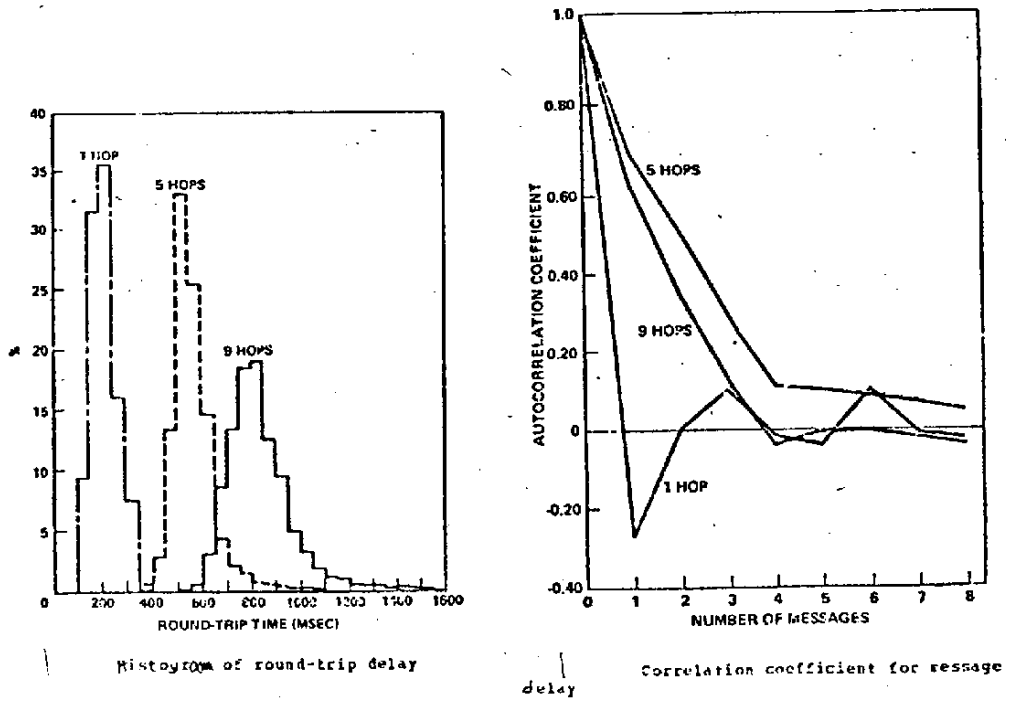
この論文では「一端露出した障害の解決をみつけることはむしろ容易である。重要なことは，設計段階でそれをいかに取り除くかであり，解析および測定によってシステム全体を定期的に評価することが，ネットワークの急激な発達とソフト／ハードの変更には必要である」と強調している。図５－６に測定結果を示す。



ARPANET throughput (October 1974)

Average round-trip delay in the ARPANET (October 1974)

図５－６の(1)　スループット測定結果



Histogram of round-trip delay

Correlation coefficient for message delay

図５－６の(2)　スループット測定結果

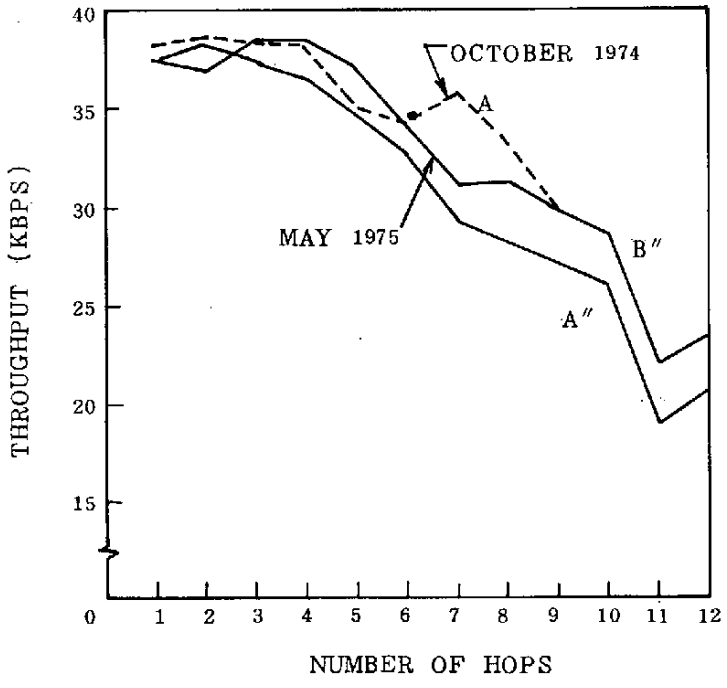


図５－７　ARPANET throughput (May 1975)

図５－７に，１９７５年に測定したスループットを示している。図５－６で示したように，ホップの距離の関数としてスループットを示している。線－A″は１０

分間の測定の平均スループット，線－B″ は，最良の１５０の連続したメッセージのスループットをそれぞれ示している。

図５－６（１９７４年１０年）の線－Ａを，比較のために，図の中に含めた。新しいメッセージ処理プロセジュアを用いたスループットが，１９７４年１０月時点に測定したスループットに劣るが，しかし，１９７５年２月に観測したものよりはるかに良いということに注目している。

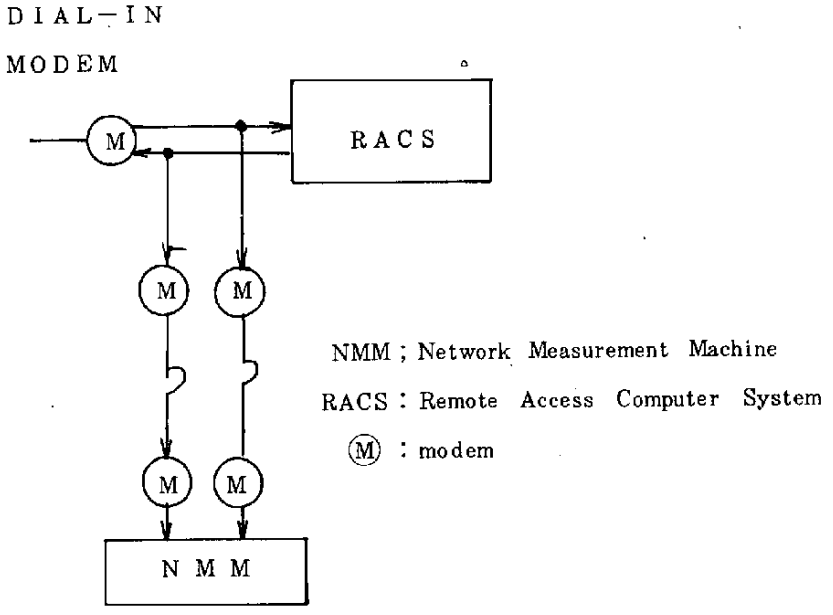


図５－８　ネットワーク測定マシンの応用

文献４では，マン・コンピュータ会話型の評価モデルを置き，コンピュータ・サービスの測定をするネットワーク測定マシン（ＮＭＭ）と称するデータ収集マシンを開発し，このデータから負荷および応答を解析するルーチンをモデルに組み込んでいる。

測定装置は高精度の実時間計，ネットワーク・インタフェース，およびデータ記録装置からなる。現在の測定は，ミニコンピュータに基づくシステムのようなものを提案し，それを実施した。このようなハードウェアとソフトウェアを結合したシステムはネットワーク測定マシン（Ｎetwork　Ｍeasurement　Ｍachine：ＮＭＭ）として知られている。（図５－８参照）ＮＭＭはネットワークを介して転送された全てのキャラクタを認識している。各キャラクタと暗黙のうちに関係しているものはＮＭＭにより記録された２つの記述子である。最初のものはキャラクタの出所がユーザあるいはコンピュータのいずれからであるかを識別する。そして他の１つはキャラクタ発生の時間を記録する。キャラクタの出所（Ｓource）はコミュニケーション制御から得られ，時間は外部の時計から得られる。得たデータには，応

答遅延への要求，そして応答経過時間終了への要求を含んでおり，また，要求，認知そして応答でのキャラクタ数がカウントされている。ネットワーク利用時に得られるデータはユーザおよびシステムの有効転送率，ユーザおよびシステムの印字されないキャラクタの割合，ユーザおよびシステム利用の間での時間比較，そしてエコー・キャラクタの数から構成されている。

この測定技術はコンピュータ・ネットワークのサービス目標の設定や目標達成の判断に活用され，また顧客に提供するネットワークの水準を明らかにし，売買契約の基礎になるであろう。

5．4　アカウンティング

コンピュータ・ネットワークにおけるアカウンティングは，通常のＴＳＳの場合と同様に，ホスト・リソースの使用料と通信コストの２つに分けて考えることができる。

コンピュータ・ネットワーク特有の問題としては，各ホストにおけるアカウンティングの標準化，複数ホストに渡るジョブに対する課金があげられよう。

ここでは，ＡＲＰＡＮＥＴにおけるアカウンティング上の問題，ユーザからみたネットワークアカウンティングの問題，網間接続におけるアカウンティング等について解説し，最後に商用サービスであるＴＥＬＥＮＥＴ，ＴＹＭＮＥＴ の料金例を紹介する。

5.4.1　ＡＲＰＡＮＥＴにおけるアカウンティング問題（文献5.6より引用）

Ａ．ＡＲＰＡＮＥＴの課金問題

ＡＲＰＡＮＥＴの初期においてＡＲＰＡ は全ホスト，全ユーザの通信，計算の費用を払った。ＡＲＰＡＮＥＴが稼動し，ＡＲＰＡ 関係以外の人がネットワークに入ってくるようになったので，ネットワークの課金を合理的に行なうシステムを作る必要がでてきた。現在の所このようなシステムはないが，疑いなくごく近いうちにできるであろう。ＵＳＣ／ＩＳＩのようなホストはすでにホスト・コンピュータ利用での課金計画を持つている。

２つのコスト，つまりリソース・コストと通信コストの２つがある。リソースにはハードウェア，ソフトウェア，データ・ベースがある。この多くはホストの制御下にある。

通信料については簡単なやり方はパケットの数で計算し，距離では計算しないことである。これはパケット・スイッチのもっとも重要な原則であり，衛星通信が実稼動したときにもこの原則が貫ぬかれよう。この原則はいわゆる〝近親〟交換，同一場所にあるホスト間での交換，（ＡＲＰＡＮＥＴ ではこういうことがある），の場合はよいものといえない。距離に関する料金法を局所的なものについて考えて見ないといけないだろう。

Ｂ．国際料金体系と税金

１９７５年初現在でＡＲＰＡＮＥＴはノルウェーと英国にホストを持っている。数年内に他の国のホストがＡＲＰＡＮＥＴに加わることになろう。国境を越えてネットワークがあるということは他国のホストを利用することで物理的にコンピュータを移すことなくコンピュータ資源を輸入していることになる。ネットワーク装置に関する限りでは他国のホストを使うことは問題なくてもより大きな問題にぶつかる。即ちハードウェア，ソフトウェア，データ・ベース，技術的ノウハウ輸出入，流通貨幣の輸出入制限，国境を越えての，はっきりした形になっていないもの製品の受渡しに関する関税等である。ＡＲＰＡＮＥＴに多くの他国が加わると，ＡＲＰＡＮＥＴはコンピュータにおけるＩＢＭが当面していると同じ問題をかゝえ，全ての政府の法的制限を合わせないといけない。

5.4.2　ユーザからみたネットワーク・アカウンティングの問題（文献２５，より引用）

(i)　アカウンティングの内訳を明確にし，できるだけ詳細に表示する。また，ジョブ・コネクションの終了時にその都度必ず出力する。

(ii)　ネットワーク内の処理機能，データ保管，伝送などのそれぞれのアカウント単位をユーザにわかり易く設定する。

(iii)　料金支払い事務を一元化する。例えば，ＩＳの場合，提供者側はＩＳの料金を請求し，一方ネットワーク運用側はネットワーク・サービス料金を請求する立場にある。ネットワークではユーザと提供者の両方の立場をとる参画者が多く，アカウント体系が複雑化する可能性がある。少なくとも純粋のユーザはネットワーク料金とＩＳ料金等を事務処理上は一元化して欲しい。

(iv)　ダウン対策の場合誰が必要経費をみてくれるか。

(v)　どのくらいのダウンが通常と考えられるか？

(vi)　ネットワーク内のあるシステムの提供者がユーザから処理依頼を受け，ネットワーク側のトラブルによりその処理を適切に処理し得なかった場合，ユーザがネットワーク運用者側の料金の返済を要求することが妥当であるか？また，その方法はどうすればよいのか？

(vii)　ネットワーク・プロセシングで複数ホストにわたるジョブの場合のアカウンティングはどうなるのか？　例えば，あるホストでジョブが途中でアボートし中断したが，他のホストではすでに分担されたジョブは終っているのでアカウントが発生する。

ユーザは目的のジョブが完成していないので支払いの義務はないと当然考えるが一部のホストでは課金され支払いが発生する可能性がある。

(viii)　ＡＲＰＡＮＥＴのユーザはネットワーク中のどのホストでも自由にリソースとして活用できるかのように考えられているが，かなりの制約がある。例えば，アカウントのとり方が標準化されていないなど真の意味のリソース・シェアが実現されているとはいえない。従

がってアカウント方法については，少なくともネットワーク内だけでも標準化すべきである。

5.4.3　網間接続（Gateway）のアカウンティング（文献2より引用）

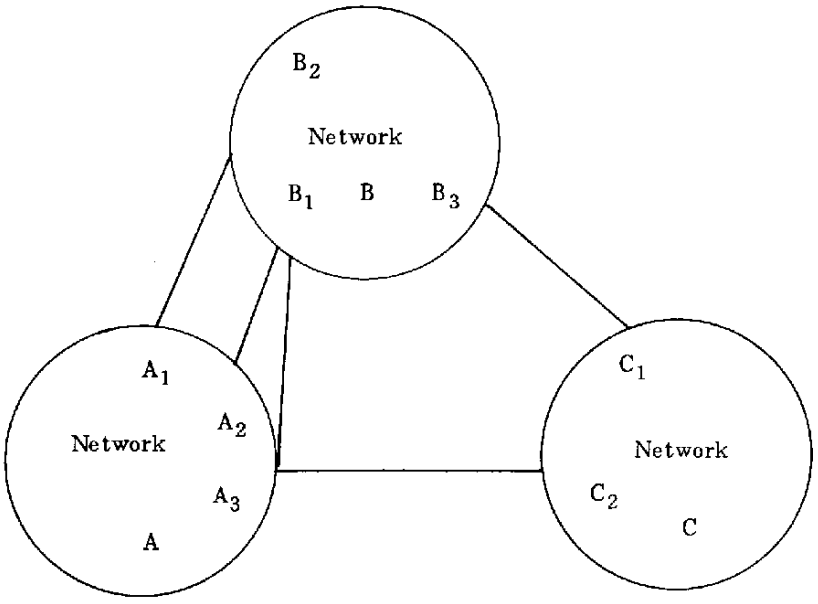


図5－9　網間接続のアカウンティング

Network A，B，Cが異ったアカウンティング体系をそれぞれに持っていると考えられいろいろな可能性が考えられる。

(i)　網外に出る料金が設定され，それぞれの間で料金が交換される。

(ii)　対網外料金一率にしてどのネットワークまたは複数のネットワーク間でも同じ料金にする。しかし，網内はそれぞれの料金にする。

(iii)　誰れが網間のアカウンティングを行なうか，方向を考えた料金なのか，AからCに行くのに，たまたまA－B－Cを通って高く料金を取られたらかなわない。

5. 4. 4 使用料金の例(文献２より引用)

(1) Telenet Monthly Charges :

• Host-Telenet access channel (AT&T DDS line)	$ 400
• Host access port at TCO	200
• Multiple connection charge	200
• Use of public dial TCO ports (5800 hours at $ 1.40 per hour)	8,120
• Approximately 4 million packets (at $0.60 per kilopaket)	2,400
TOTAL	$11,320

Total Telenet charges per terminal-hour : $1.95

Present communications cost per terminal-hour : $3.90 (Using multiplexers in 12 remote cities)

(2) Telenet Monthly Charges :

• Host-Telenet access channel	$ 300
• Host access port at TCO	200
• Multiple connection charge	200
• 24 private dial ports in TCOs @ $200/month	4,800
• AT&T foreign exchange lines to 3 remote cities	1,100
• 1.54 million packets (at $0.60 per kilopaket)	924
TOTAL	$7,524

Total Telenet charges per terminal-hour $3.42

Present communications cost per terminal hour : $8.64 (Using WATS lines)

(3) Tymnet の料金

表５－２ Tymnet の料金

STORE － AND － FORWARD COSTS	
	0 － 300B/S
	CONNECTION COST PER HOUR
HIGH-DENSITY AREAS	$ 2.20(1) (2)
LOW-DENSITY AREAS	5.20(1) (3)
WATS	13.20(1) (4)
(1) PLUS $0.10PER 1,000INPUT／OUTPUT CHARACTERS (2) $0.10 MINIMUM CHARGE PER CONNECTION (3) $0.15 MINIMUM CHARGE PER CONNECTION (4) $0.20 MINIMUM CHARGE PER CONNECTION	
	600 － 1,200B/S
	CONNECTION COST PER HOUR
HIGH-DENSITY AREAS	$ 3.20(1) (2)
LOW-DENSITY AREAS	6.20(1) (3)
WATS	13.20(1) (4)
(1) PLUS $0.03PER 1,000INPUT／OUTPUT CHARACTERS (2) $0.10 MINIMUM CHARGE PER CONNECTION (3) $0.15 MINIMUM CHARGE PER CONNECTION (4) $0.20 MINIMUM CHARGE PER CONNECTION	

５．５ ネットワーク運用の業務体制（文献２より引用）

ネットワーク運用の業務体制については，ユーザからみた業務体制に対するコメントという形でいくつかの意見を紹介する。

5.5.1　業務体制

ネットワークを運用する場合に業務的な体制が整っていなければならない。業務的に分業され責任関係が明確になっていないとユーザからの問い合せ，非常時においてその機能を遂行できなくなる。

A．業務体制

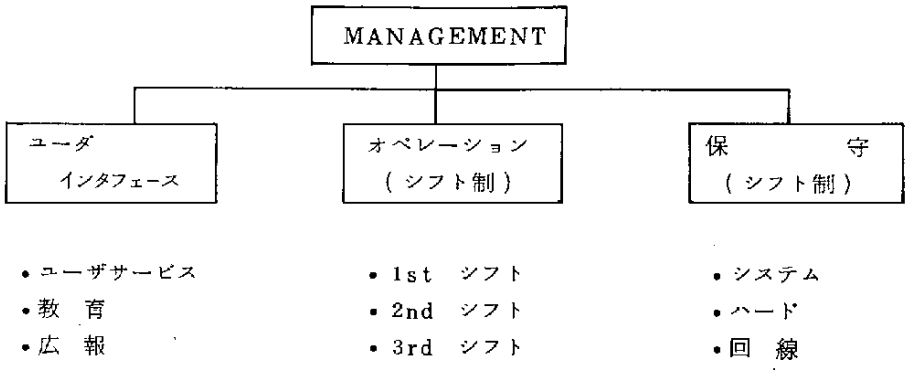


図５－１０　業務体制

(a)　各セクションの最低数の要員は常に待機しておかねばならない。

(b)　並行して教育，非常時の訓練，運用の合理化などに取り組む必要がある。

(c)　ソフト（新しいバージョン）は，検査使用期間を隔て実動にもっていく。

(d)　ハード保守は定期的なスケジュールに基いて実行される。

(e)　チェック・リスト等を作りマンネリ化を防ぐ。

(f)　シフト制を完備し人権問題に注意する。

B．保　守（メインテナンス）

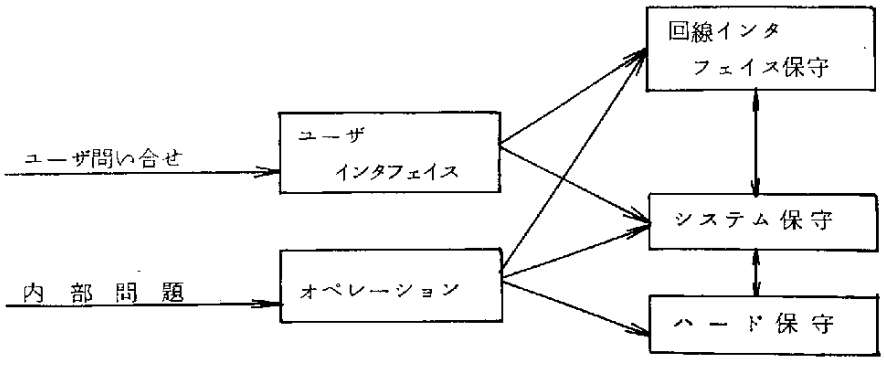


図５－１１　通常の保守の例

(a) ユーザとのインタフェイスは１人の方が望ましい。

(b) １問題単位で処理できない場合のみ複合して処理する。

(c) ユーザ問題は必ずフォローして相手に何らかの形で満足を与えるようにする。

(d) 問題を回線保守，システム保守，ハード保守等とたらい回ししないように気をつける。

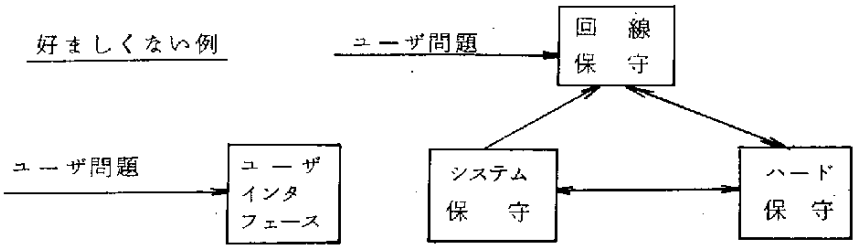


図５－１２　保守の好ましくない例

(e) ローカルな保守については，保守ステーションを設け，そこから保守を定期時，非常時に分けて行なう。

5.5.2　運用に対する要望と問題点

A．ユーザ側から運用側に対する要望

- ユーザは誰に問題を話せばよいのか。
- ユーザの依頼を受ける人がまちまちでは困る。
- ユーザの依頼を受けた人が，他人に依頼し，責任が薄れ，最終的なフォローがない。
- ユーザはネットワーク運営の体制を見えるようにしてほしい。
- 問題が起った場合，速く処理してほしい。
- 安くしてほしい。
- むだなことはいらない。
- ネットワーク内またはサービスに関する資料が欲しい。
- 将来使いたいが，見積りをすぐ欲しい。
- 予算で働いているので料金のリミットがオーバーする以前に知らせて欲しい。
- アカウント番号を変えたい。
- ソフトウェアの使用料を取りたい。
- ソフトウェアの宣伝をネットワーク内にやりたい。

B．運用側が指適する問題点

- ネットワーク用ソフトウェアの信頼性
- ネットワーク用ハードウェアの信頼性
- 運用時間中にはハードウェアの切替えや電源の変更などは行なわないこと。
- 標準保守スケジュール
- 緊急保守
- 要員教育
- シフト制の要員体制
- ユーザ加入数の月間予測
- ソフト，ハードのアップ・グレード計画
- 施設のセキュリティ問題
- 非常時の電源装置，冷却装置のバック・アップ
- 共通網の場合，新しく網に参画するシステムの質の検査機構の確立，標準的検査項目および検査技術の整備
- ユーザ・システムの変更，拡大に対する上記検査機能のフォロー・アップ

第 5 章 の 引 用 文 献

文献 1. : THE ARPA NETWORK CONTROL CENTER

: Alexander A. Mckenzie

Bolt Beranek and Newman Inc.

Cambridge, Massachusetts

: Fourth Data Communications Symposium 75

文献 2. 国際情報ネットワークに関する調査研究報告書

(財) 日本情報処理開発協会

51－R008 昭和52年5月

文献 3. : Throughput in ARPANET

-Protocols and measurement

: by Leonard Kleinrock and Holger Opderbeck

Computer Science Department

University of California, Los Angeles

: Fourth Data Communications Symposium '75

文献 4. : A new approach to performance evalution of computer networks

: by Mayshall D. Abrams

: Institute for Computer Science and Technology

National Bureau of Standards

Washington, D. C. 20234

文献 5. : 1974 Symposium Computer Networks

: Public Policy Issues Concerning ARPANET

: by F. F. Kuo

: The ALOHA System, Univ. of Hawaii

: Fourth Data Communications Symposium '75

文献 6. : コンピュータ・ネットワークに関する調査

～最新の実用化技術と将来の展望～

: (社) 日本電子工業振興協会

51－C－305 昭和51年3月

文献 7. : Management strategy for controlling network operations

: Howard L. Giles I. B. M. Raleigh, N. C.

: Data Communications／August 1977

ご利用者各位

〒105
東京都港区芝公園3－5－8
機械振興会館内
財団法人　日本情報処理開発協会
開発部ソフトウェア普及担当
電話　東京（03）434－8211
内線　205　（大代表）

お　願　い

本解説書をご利用頂きまして，ありがとうございました。

さて，本解説書についてご気付の点がありましたなら，下欄にご記入のうえ，当財団担当までお知らせ下さいますよう，お願い申し上げます。

なお，他の開発ソフトウェア，その他についてもご意見を併記して戴ければ幸甚に存じます。

ご住所 貴会社 （団体）名 ご芳名	解説書名
ご意見	

昭 和 53 年 3 月 発 行

発行所　財団法人　日本情報処理開発協会
東京都港区芝公園３－５－８
機 械 振 興 会 館 内
ＴＥＬ (434) ８２１１ (代表)

印刷所　ヨ シ ダ 印 刷 株 式 会 社
東京都葛飾区奥戸４－21－４
ＴＥＬ(692) ８６８６(代表)