

6. 技術解説

T C P / I P ネットワーク

総合情報処理センター 鶴 正人

今日の情報化社会において、コンピュータネットワークの重要性はますます増大している。大学においても例外ではなく、従来はコンピュータやネットワークとは無縁とされていたような分野も含めて、研究／教育／事務の方法が見直されてきている。

そのようなネットワークにおいては、広範囲の相手と多様な情報通信ができる必要がある。すなわち、各種ネットワークサービスが実現できること（応用層プロトコルが普及している）、オープンで標準化されていること（コンピュータのメーカーや機種を選ばない）、多様な接続形態に対応できること（近距離から遠距離まで場合に応じた効率よい接続ができ、特にLANとパケット交換網に対応できる）が重要である。これらの条件を満たすコンピュータネットワーク技術として、TCP／IPとか、OSIといったような体系とが知られている。また、他に、メーカー系のSNA（Systems Network Architecture：IBM）とかXNS（Xerox Network Systems：Xerox）とかも広く使われている。

特に大学においては、TCP／IPという言葉は、UNIXの普及に伴って、かなりポピュラーになった感じがある。しかし、実は言葉しか知らないという人や、UNIX固有の技術だと誤解している人もいるのではないかと思う。

そこで、本稿では、代表的なコンピュータネットワーク技術体系であるTCP／IPについて、OSIの基本参照モデルをふまえながら、下記の順で概説する。

1. コンピュータネットワーク
2. TCP／IPとは
3. 基本的通信機能のプロトコル
4. 応用サービス機能のプロトコル
5. UNIXにおけるプログラミング・インタフェース
6. 長崎大学におけるTCP／IPネットワーク

ただし、特定のOS／マシン上での機能範囲、オペレーション等の説明を含まないため、「利用の手引き」としては使えないことを、前もってお断りしておく。

1. コンピュータネットワーク

「コンピュータネットワーク」という言葉はTPOに応じていろいろな意味に使われているが、ここでは、「異なるコンピュータ上で動作する主体（プロセス）間でデータを伝達する時の論理的媒介物」と考える。もっとも、「コンピュータ」の範囲自体も曖昧だし、プロセスには人間も含まれるので、厳密な定義ではない。

とにかく、これが存在してはじめて、離れた場所（隣の部屋かもしれないし、外国かもしれないが）のコンピュータ上で動作しているプログラム（あるいはそれを利用している人間）間のリアルタイムな情報のやりとりが可能になり、そして、単純には3つの面の利便がもたらされる。

1) 直接的には、コンピュータで処理すべきデータの伝達がフロッピーディスクやテープを用いた人手の運搬に比べて、非常に簡単かつ高速になる。これによって大幅な省力化ができる。

2) コンピュータによって実現されるサービス（資源）が、そこから離れた場所でも利用できる。これにより、データ、ソフト、ハードの一元管理や共同利用が計れる。

また、リモートxxx（例：リモートログイン）、ホームxxx（例：ホームバンキング）等と呼ばれる利用形態／サービスが可能になる。

さらに、逆に資源を分割・分散させて、全体としてのスループットの向上（負荷分散）、コストパフォーマンスの向上（機能分散）、信頼性の向上（危険分散）等を計る場合もある。

3) 新しいコミュニケーションの手段が誕生する。具体的には、電子メール、電子掲示板、電子ニュース等である。これらは、我々が日常の情報もコンピュータを使って処理するようになってきた（例えば、ワープロ、電子手帳、電子ファイリング等）ことで、急速に普及しつつある。

そこで、TCP/IPの説明の前に、まず、各種コンピュータネットワークを統一のとらえて標準化するためのモデルとして広く受け入れられている、OSI

(Open Systems Interconnection: 開放型システム間相互接続)のための基本参照モデルを簡単に説明する(→[JIS])。

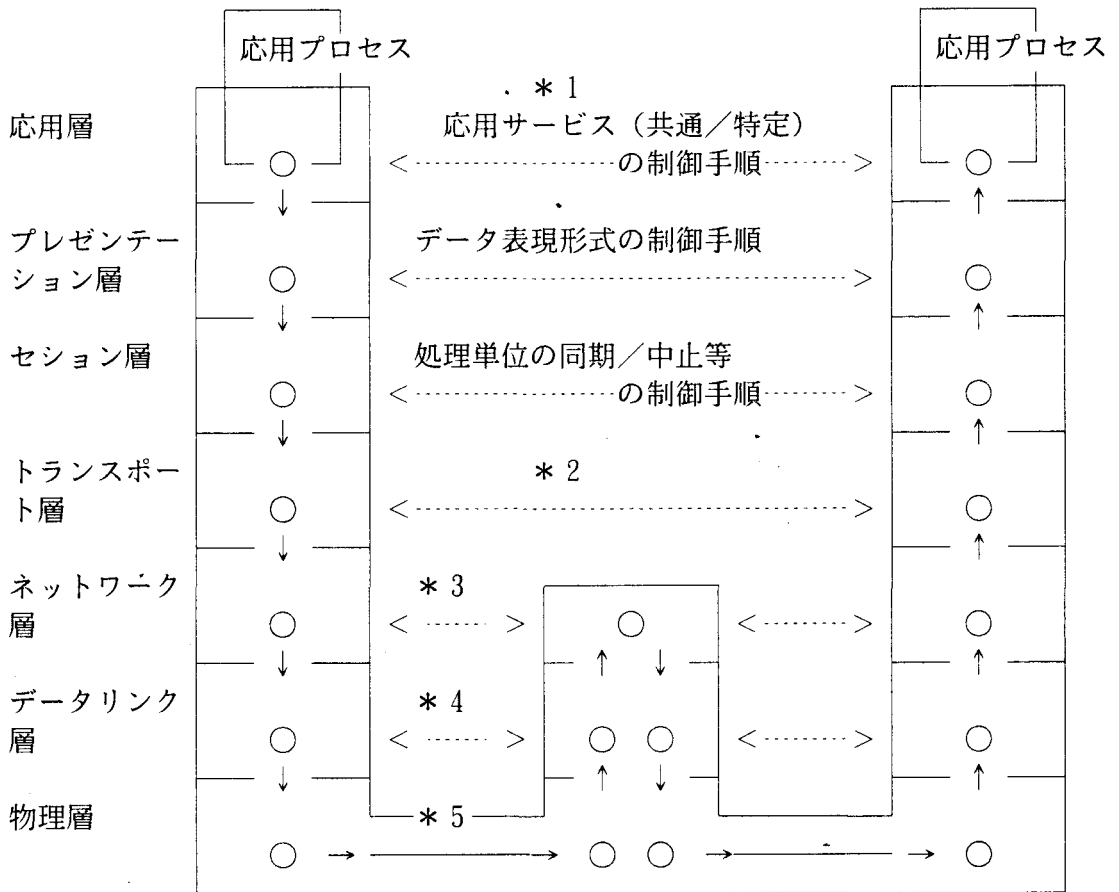
これは、ISO (International Organization for Standardization: 国際標準化機構)とCCITT (Consultative Committee for International Telegraph and Telephone: 国際電信電話諮問委員会)の合同作業で進められている、コンピュータネットワークの世界統一仕様であるOSI体系の基本となるもので、コンピ

ータネットワークの構造・機能（ネットワーク・アーキテクチャ）を，システムとシステムを結合するための7階層構造をもつプロトコル仕様とサービス定義の集まりとして規定しようとするモデルである（→図1. 1， 1. 2）。

ここで，OSI（体系）という言葉は，このモデルに準拠して開発・制定されつつある個々のプロトコル仕様+サービス定義（規格と呼ぶ）の集まりを指す。

基本参照モデルの「エンティティ」とは，通信処理を行う抽象的な動作単位であり，何らかのプロセスの一部と考えられる。また，ここでいう「システム」（オープンシステム）は，抽象的なエンティティの動作環境単位であり，通常は1つのコンピュータのハード/ソフトで実現される。

—図1. 1—（7階層基本参照モデル）



エンドシステムA

中継システムB

エンドシステムC

→ : 実データの流れ

< ... > : 各層（レイヤ）のプロトコルの流れ

○ : 各層（レイヤ）のエンティティ

* 1 :

応用層、プレゼンテーション層、セッション層は、以降のTCP/IPの説明においては明確な区切りをつけていないので、本稿内ではひとまとめに考えてもらってかまわない。なお、OSIの世界ではこの3層を上位層 (upper layers) と言い、下位の4層を下位層 (lower layers) と言う。

* 2 :

エンドプロセス間の高信頼・高能率データ転送の制御手順。ネットワーク層以下のサービス品質の差異を吸収する。

* 3 :

経路/中継の制御手順。エンドシステム間のデータ転送を提供する。多様な形態をモデル化するため、副層構造を定義する。

* 4 :

隣接システム間のデータ転送制御手順。物理媒体上に論理的伝送路 (データリンク) を作り、ある区切り (フレーム) 単位のデータ伝送/同期/エラー時の自動再送等を提供する。ただし、物理層でのバイト単位の同期に頼り、そのバイト列を流すだけの「無手順」 (フリーラン) もある。

* 5 :

機器/伝送路の物理的形狀、電気的特性、論理的機能等の接続のための規定。ビット列の伝送/同期を提供する (「非同期」とは、スタートビット、ストップビットによる調歩同期のことである)。モデム/DSU/音響カプラ等のDCE (データ回線終端装置) を用いて接続する場合は、DTE (データ端末装置=システム) - DCE間と、DCEより先に分けて考えられる。

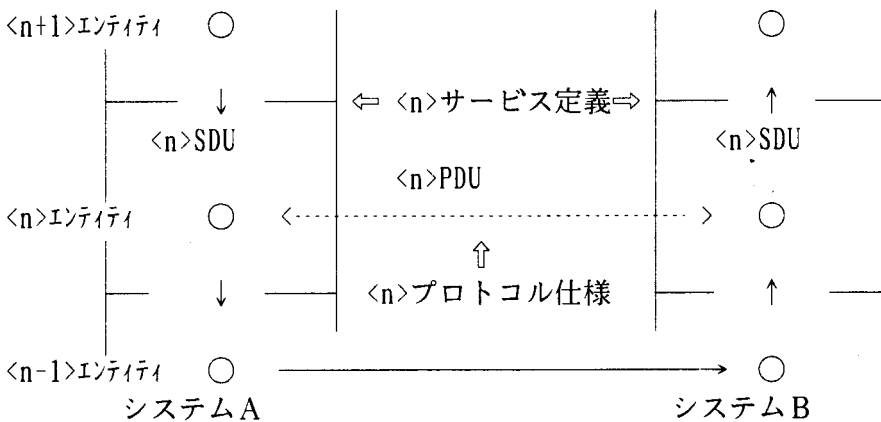
このモデルでは、第n層サービス定義とは、第n+1層と第n層の間の抽象的なインタフェースを決めるものであり、第n層エンティティが第n+1層エンティティに対して提供する論理的な機能の規定である。第n+1層と第n層の間で受け渡されるデータを<n>SDU (サービスデータ単位) と言う。

また、第n-1層の機能を用いて第n層エンティティ間 (同位エンティティ間) が交換するデータを<n>PDU (プロトコルデータ単位) と言う。第n層プロトコル仕様とは、基本的には<n>PDUの意味、順序、形式等、及び第n-1層サービスの使い方の規定である。<n>PDUは、一般に<n>制御情報 (<n>ヘッダとか<n>PCIとも言う) と<n>ユーザデータから成る (両方または片方のみ)。<n>制御情報は、第n層エンティティによって生成 (送信時) ・解釈

(受信時)され、第n層の動作を決めたり、第n+1層との間で意味の決まったパラメタとしてやりとりされたりする。一方、 $\langle n \rangle$ ユーザデータは、 $\langle n+1 \rangle$ PDUを転送し、第n層にとっては透過的である(→図1. 2)。このように、 $\langle n+1 \rangle$ PDUを $\langle n \rangle$ 制御情報でくるんだものが $\langle n \rangle$ PDUになることを、カプセル化と呼ぶことがある。 $\langle n \rangle$ PDUと $\langle n-1 \rangle$ SDUとの対応や、 $\langle n \rangle$ SDUと $\langle n \rangle$ PDUとの対応は、必ずしも1対1ではない。複数の $\langle n \rangle$ PDUを1つの $\langle n-1 \rangle$ SDUにまとめて下位に渡すことをコンカチネーション(連結)と言い、複数の $\langle n \rangle$ SDUを1つの $\langle n \rangle$ PDUにまとめて扱うことをブロッキング(結合)と言い、また、1つの $\langle n \rangle$ SDUを複数の $\langle n \rangle$ PDUに分けて扱うことをセグメンティング(分割)と言う。

これらの階層やサービス定義は、決して実製品のソフトウェア構造やインタフェースを規定するものではない。実装においては、複数の層が1つのプログラムだったり、中間層がAP(アプリケーション・プログラム=上位ソフト)でその上下の層が基本OS(下位ソフト)だったりする。

—図1. 2— (サービス定義とプロトコル仕様の関係)

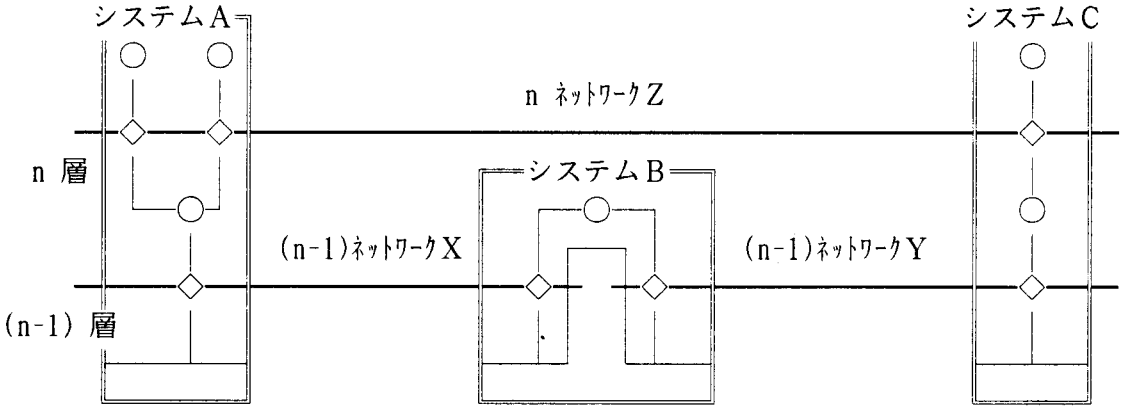


ここで、ネットワークという言葉を少し厳密に扱わないと、混乱を生じる。

上記の階層モデルにおいて、第n層は、第n+1層に対して、抽象的には $\langle n \rangle$ ネットワークを提供していると考えられる。第n+1層エンティティは、 $\langle n \rangle$ ネットワーク上の2つの点(サービスアクセスポイント=SAPと呼ぶ)間でデータの転送ができる。この点を指定するものが、 $\langle n \rangle$ ネットワーク上のアドレスであり、 $\langle n \rangle$ SAPアドレスと呼ぶ。1つの第n層エンティティは1つ以上の $\langle n \rangle$

SAPを提供し、1つの第 $n+1$ 層エンティティは1つ以上の $\langle n \rangle$ SAPと結びついて第 n 層サービスを利用する。つまり、各層毎にネットワークが存在し、複数の $\langle n-1 \rangle$ ネットワークが結合されてより広範囲な1つの $\langle n \rangle$ ネットワークが構築されたり(→図1.3), 1つの $\langle n-1 \rangle$ ネットワークの上に複数の $\langle n \rangle$ ネットワークが論理多重化されたりする(→図1.4)。

—図1.3— ($\langle n \rangle$ ネットワークと $\langle n-1 \rangle$ ネットワーク—中継—)

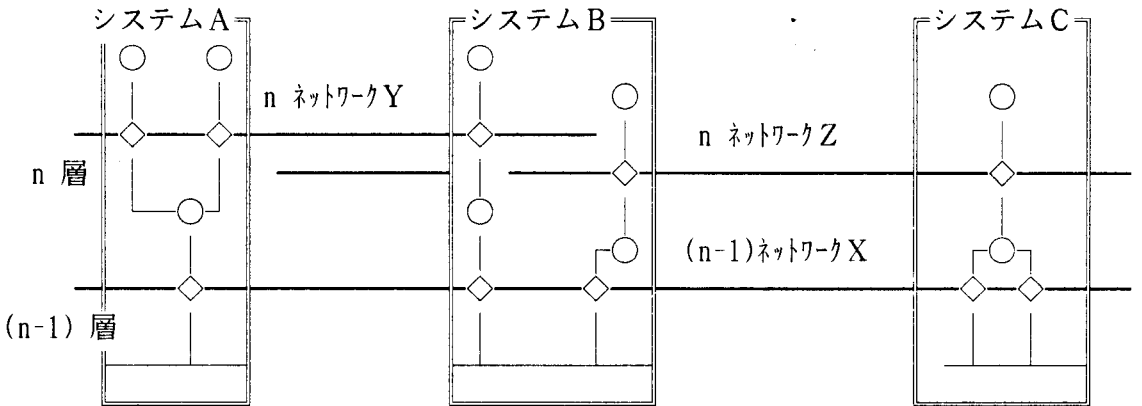


◇ : SAP ○ : エンティティ — : ネットワーク

例1 : X=EthernetLAN Y=TokenRingLAN Z=IP「ネットワーク」 B=ルータ

例2 : X=EthernetLAN(物理) Y=TokenRingLAN(物理) Z=LAN(LLC層) B=ブリッジ

—図1.4— ($\langle n \rangle$ ネットワークと $\langle n-1 \rangle$ ネットワーク—論理多重化—)



◇ : SAP ○ : エンティティ — : ネットワーク

例1 : X=EthernetLAN Y=OSI-コネクション型「ネットワーク」 Z=IP「ネットワーク」

例2 : X=TCP サービスネットワーク Y=RLOGINサービスネットワーク Z=TELNETサービスネットワーク

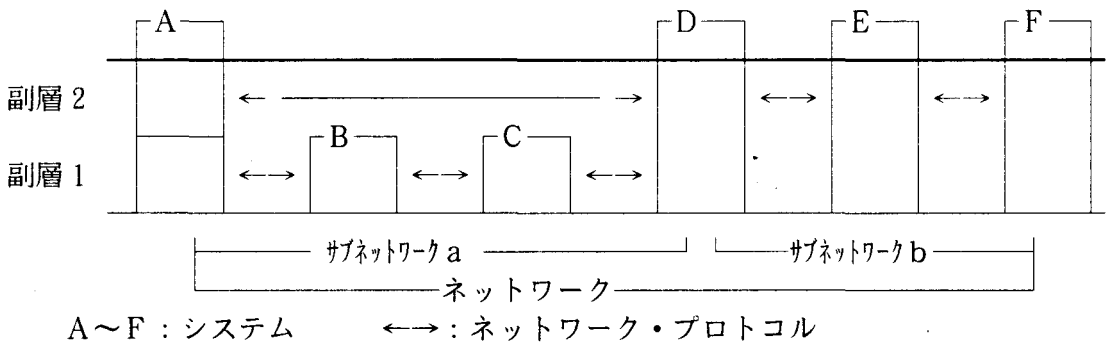
後者においては、異なる $\langle n \rangle$ ネットワークでは第 n 層エンティティが異なると

解釈すれば、それぞれが別の $\langle n-1 \rangle$ SAP を用いることで実現され、また、第 n 層エンティティが同一と解釈すれば、 $\langle n-1 \rangle$ ユーザデータ内の $\langle n \rangle$ プロトコル識別子を第 n 層エンティティが見て複数のプロトコルを判別することで実現される。これを TCP/IP ではマルチプレクシングと呼ぶが、この用語は OSI ではコネクションの多重化を意味するので避ける方がよい。

狭義のネットワークは、「ネットワーク層」によって提供されるものである。このネットワークによってはじめてエンドシステム間のデータ転送が可能になる。この層は、多様な通信媒体や既存の各種プロトコルを持つネットワークにまたがっての中継を実現するので、それをモデル化するための副層構造が必要になる（→図 1. 5）。

以降では、ネットワーク層によって提供されるものを「ネットワーク」、データリンク層が提供する直接伝送可能範囲の意味のものを単位ネットワーク、ある「ネットワーク」内の 1 つ以上の単位ネットワークの集まりをサブネットワーク、ネットワーク層より上位の層でプロトコル毎にその相互接続範囲を個別のネットワークと解釈する場合を xx サービスネットワークと呼んで、必要な場面では区別する。

— 図 1. 5 — (ネットワーク副層)



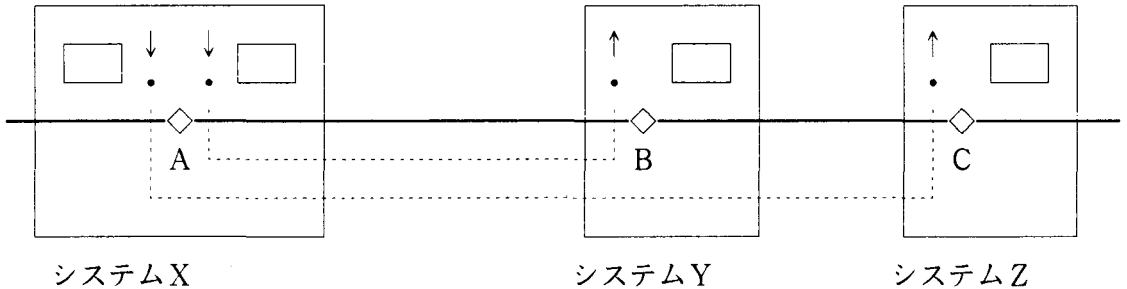
一般的には、 $\langle n \rangle$ ネットワークが第 $n+1$ 層に提供する基本機能はデータ転送である。データ転送のサービス方式は、コネクション型（仮想回線）サービスとコネクションレス型（データグラム）サービスとに大別される（→図 1. 6）。

コネクション型の場合、第 $n+1$ 層はまず $\langle n \rangle$ ネットワーク上の 2 点 ($\langle n \rangle$ SAP) 間に“コネクション”を設定する。これは、SAP 間を仮想的に直結するパスであり、コネクション設定後は、端点からデータを送信すれば、そのままの順で他方の端点において受信できる。もちろん、2 点間のパーマネントなコネクションを持つ場合もあり、その時は設定は不要である。

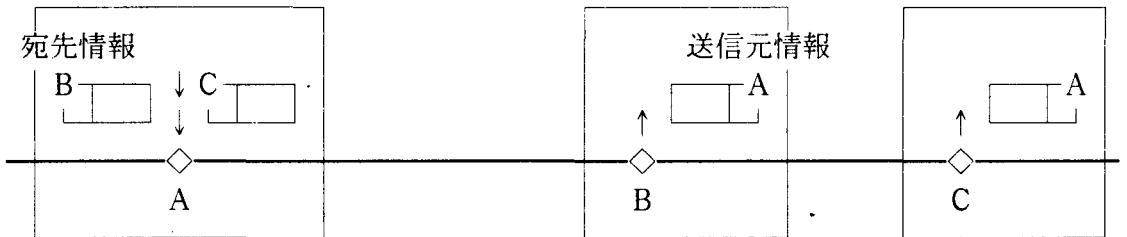
一方、コネクションレス型の場合、第 $n + 1$ 層はコネクション設定は不要だが、その代わり、1つの送信単位（データグラム）毎に送信先の $\langle n \rangle$ S A Pを指定する必要がある。各データグラムは独立に扱われるので、到着順序を保証しないようなプロトコルが多い。

—図1. 6—（アドレスAとB間、アドレスAとC間のデータ転送）

<コネクション型の場合>



—— は、ネットワーク、◇ は、ネットワーク上のS A P。
 ・は、コネクション端点、----- は、コネクション、↓↑ は、ユーザデータ送受信。
 <コネクションレス型の場合>



ただし、実存するネットワークにモデルを適用する時には、注意が必要である。
 1) モデルは非常に一般的なものであるが、実存するネットワークのいくつかの典型的なパターンに対するマッピングの方法（例えば、どこを「システム」と「システム」の境界とするか）を決めておかないと、実際のプロトコル規格の標準化ができない。

例えば、NTT等による広域網（電話、デジタル回線、パケット交換等）の場合は、網側は標準化の対象外である。物理層の規格は、各システム-DCE（モデム、DSU等）間の接続規定とDCEの機能とが中心であり、データリンク層及びネットワーク層の規格は、エンドシステム間か、システム-最寄りの交換機間かの規定である。よって、モデル上はDCEより先の網の実体は1つの”オープンシステム”か、または、単なる物理伝送路としてブラックボックス化して解釈する。

一方、LANの場合は、伝送路とその上の全ての装置を含んで標準化が進められている。

また、端末の扱いも自明ではない。「端末」は、ハードウェアの種類ではなく、広義のコンピュータのある使用形態（ネットワークを介した機能分散）と考えられ、PCでもエミュレータを乗せて端末として使用できる。これは一種のコンピュータネットワークであるが、モデル的には、データリンク層またはネットワーク層上に端末制御手順（一種の応用層プロトコルとみて）が乗った「実端末」接続と、応用層の仮想端末機能を経由する「仮想端末」接続とに分けられる（→図4.2）。この場合、端末制御手順は標準化の対象ではないようなので、その意味では実端末接続はオープンシステム間接続ではない。

なお、端末制御手順に関しては、本稿では、文字／行単位で送受信するものをTTY型と呼び、画面単位で送受信するものをブロック型と呼ぶことにする。

2) 実際のネットワークには管理組織があり、単一プロトコルでも管理境界をまたがる通信は何らかの制限が付き、単純に単一ネットワークとは言えない。それゆえ、管理組織の異なる「ネットワーク」を結合してできた、1つの「ネットワーク」を「インターネットワーク」と呼ぶことがある（→[村上]）。この関係は再帰的である。

3) 逆にプロトコル的に異なるネットワーク間が、プロトコル変換（ゲートウェイ）によって接続されると、単一のネットワークとして見える場合がある。プロトコルの異なる複数の n ネットワークを単一の第 $n+1$ 層プロトコルが結合するのは、ただの中継であり、ゲートウェイとは呼ばないことが多い。ゲートウェイで結ばれた仮想ネットワークにおいては、アドレス形式（通信相手を指定する方法）の統合／変換が大きな問題となる。

ゲートウェイは通常は、「ネットワーク」レベルか応用サービスレベルかのどちらかである。応用サービスでのゲートウェイの例として電子メールが重要で、UNIX間スプール転送(UUCP)、TCP/IP(SMTP)、IBM系(RSCS)、OSI(MOTIS)等のプロトコルをまたがったメールの相互転送が実現されている。

4) 1つのシステムが複数の異なる「ネットワーク」上に存在する（当然、複数の自SAPを使用できる）場合、上位の層は転送をはじめる前にどの「ネットワーク」（どのSAP）を使うかを選択する必要がある。ところが、どの「ネットワーク」を使うのが最適かの判断に「ネットワーク」層の持つ経路情報が必要になる。

これに対する解釈は、各システム内のローカルな方法でネットワーク層と上位の

層とが作用する，という立場と，ネットワーク層のS A Pはネットワークアドレス・リスト（またはリストに対応づけられた論理的なシステム名）で識別され，その中から最適なアドレス（「ネットワーク」）をネットワーク層が判断する，という立場がある。

5) 既存のプロトコル体系に適用すると，細かい点でモデルに違反しているか，または，概念・用語等の対応がきれいにとれないのが普通である（そのような差異があるからこそO S Iという標準化が必要になった）．T C P / I Pも例外ではない（→3章）．用語に関していえば，同じ用語が異なる意味を持つ場合もある．基本的には，O S I用語で考え，その上でT C P / I P用語との関係も理解するのがよいと思う．

また，O S I内のプロトコルでもC C I T T系の既存勧告は独自の複雑な規格体系があり，分かりにくい感じがする．例えば回線系の物理層勧告では，物理形状，回路規定，モデム機能等，個々の規格の規定範囲が複雑なので，普及している組み合わせを不正確な名前，例えば，V. 28/V. 24（25ピンRS232C低速用），V. 35（34ピン48K同期型用），V. 11/X. 21（15ピン高速同期型用），V. 11/V. 24（37-9ピン高速同期型用）等と呼び，それと，回線速度，アナログ網／デジタル網等の条件とで接続仕様を示すことが多い．

その他，I E E E（アメリカ電気電子学会），A N S I（アメリカ規格協会），E C M A（欧州電子計算機工業会）等の規格がO S I化された場合も元の名前や体系が残る場合がある．

6) さらに，実際のコンピュータ間のデータ通信と電話・F A Xや映像装置との共存に関しては，モデル的に不十分である．これらの背景には，実在するネットワークの運用や接続形態は，技術的にも，また，管理や経済上の理由によっても，日々変化しており，統一的・静的なモデル化に適しにくいという面がある．ただし，O S I基本参照モデルそのものも，技術変革に追従して拡張されていくはずである（多対多通信，マルチメディア通信等）．

2. T C P / I Pとは

T C P / I P（ティーシーピー・アイピーと呼ぶ）とは，D O D（Department of Defense：アメリカ国防総省）に属するD A R P A（Defense Advanced Research Projects Agency：アメリカ国防高等研究計画局）を中心に開発された，開発・研

究者向きコンピュータネットワーク技術の総称である。

ネットワーク技術とは、狭義に考えれば、通信プロトコル仕様の集まりであり、その意味では、TCP/IPとは、INTERNETプロトコル・スイートと呼ばれるプロトコル体系を指し、OSIと対比して使うことができる言葉である。そして、そこに含まれる重要なプロトコルとして、TCPとIPとがある（→3章）。

一般には、さらに実際のネットワーク実体とか実装上のノウハウとかを含めて考えるが、それらの多くの仕様や研究成果はRFC (Requests For Comments)と呼ばれる一連のドキュメントとして公開されており、ファイル転送や電子メール（→4章）を使って入手可能である。長崎大学では一部がUTS上で公開されており、それ以外も入手できる（→6章）。

TCP/IPの技術は、70年代末までに応用層を含むかなりの部分が研究されていたが、その後、ARPANET (DARPAによって作られた、全米を結ぶX.25パケット交換網) という広域ネットワーク上でIPプロトコルを運ぶことによって全米の大学・研究機関をTCP/IPプロトコルで接続できるようになり（その全体を1つの「ネットワーク」としてINTERNETと呼ぶ）、資源共有やコミュニケーションの手段として急速に使われ始めた。

さらに、80年代前半にUNIX4.2BSDに全面的に取り入れられ、また、その時により使いやすいサービスを提供する応用層プロトコルと利用インタフェースが開発されたため、TCP/IPとUNIXは密接に関連するようになり、ますます普及していった。そして、ワークステーションのための構内ネットワークの主流であるイーサネットLAN上でもいち早く実装された。また、多くの部分は、その後DODにより米軍標準(MIL-STD)として発行されている。

TCP/IPの普及の技術的理由は、一般に下記のように言われている。

1) 異機種間接続のために計画/開発され、仕様のみならず、開発上の意見交換等もオープンかつタイムリであり、実際、各種OS上で実装された。各メーカーが独自のネットワークアーキテクチャの開発を進めていた頃に、これは画期的なことであり、また、この成功が時代の流れをオープン化の方向へ変えた。

2) IP層が、広範囲なネットワーク(インターネットワーキング)に対応するよう、また、各種の物理媒体に対応するよう、設計された。それゆえ、手軽にIPネットワークの拡張/相互接続ができるので、INTERNETのような巨大なインターネットワークが出来上がった。

3) 応用層プロトコルの重要性が早くから認識され、エンドユーザが直接利用でき

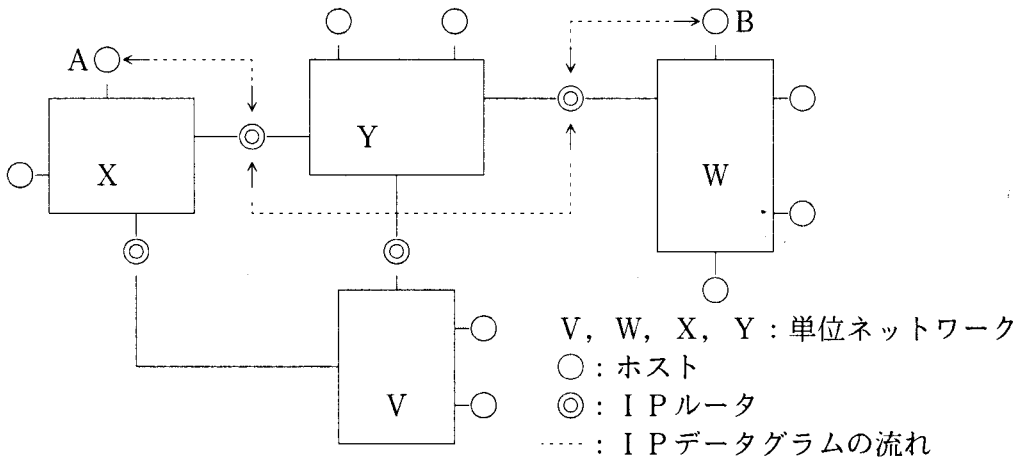
るネットワークサービスが標準化及び製品開発された。これにより、電子メール、ファイル転送、リモートログイン等の便利さが、ユーザ間に広まった。

また、ネットワーク上のコミュニケーションにより、ユーザ間の連絡／意見交換やドキュメント／ソフトウェアの配布等がタイムリーに行えるので、ネットワークの管理や環境整備の効率が向上した。

4) 3)を容易にするために、AP (アプリケーション・プログラム) を開発しやすい高信頼なトランスポート層相当のTCP層が定義され、かつ、それをAPから使えるインタフェースが各OS上に実装された。これにより、ユーザが容易に応用層プロトコルを開発でき、それがまた普及する、という好循環を生んだ。

TCP/IPでは、1章の基本参照モデル上のネットワーク層 (の最上位副層) として、IPという単一のプロトコルがある。そして、各種物理媒体上の単位ネットワークを、" ルータ" (IPルータ) が自由に結合してIP「ネットワーク」が形成される、というモデルである (→図2. 1)。ルータを" ゲートウェイ" と呼ぶ場合もあるが誤解しやすいので避けた方がよい。

—図2. 1— (ホストA, B間のIP接続)。

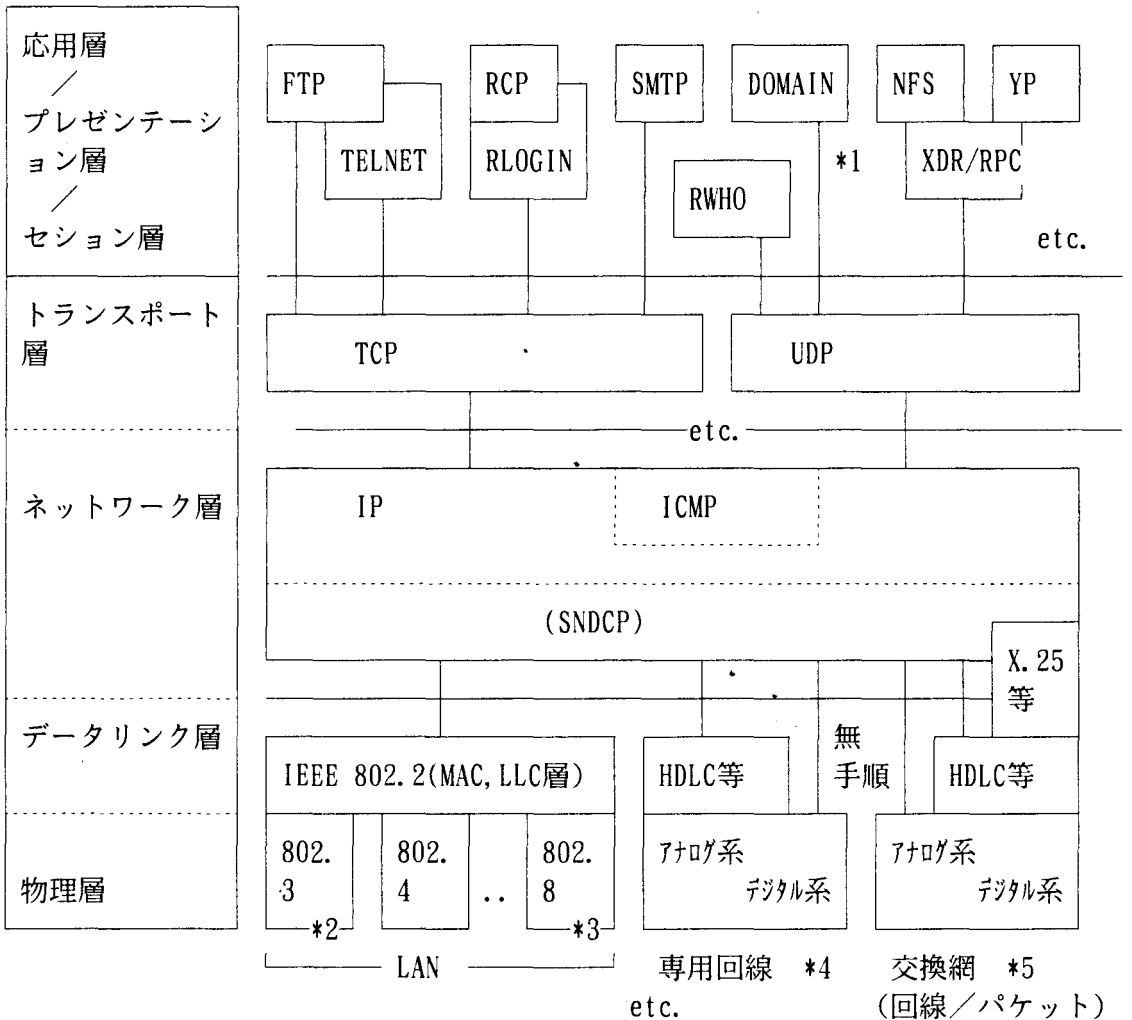


このIPプロトコルで接続されている各コンピュータ (ボード付きパソコンからメインフレームまで) を、" ホスト", " ノード", " サイト" 等と呼ぶ (ただし、これらの言葉は別の意味で使われることもあり、注意が必要である)。これは、基本参照モデルでの" オープンシステム" に当たる (" オープンシステム" の方が意味は広い)。本稿では、" ホスト" という呼び方で統一する。

それゆえ、IPアドレスさえ重ならないければ、個々の「ネットワーク」をルータでつなぐことで、原理的には全世界のホストが単一のIP「ネットワーク」で接続できることになる。実際、INTERNETでは直接・間接に海外ともつながっているため、SRI-NIC (Stanford Research Insutitute Internatinal: スタン

フォード国際研究所内のNetwork Information Center) という組織が全世界に対してIPアドレスが重ならないことを保証するためにネットワークアドレスの払出しを行っており、その意味では世界中を結ぶ巨大なIPネットワーク(またはインターネット)が存在するといえる。

最後に、TCP/IPの代表的プロトコルの名称とモデル上での位置を示す。
 -図2. 2- (TCP/IPの代表的プロトコルの体系)



* 1 :

DOMAINのようにUDP上とTCP上の両方に実装されているものがある。また、RPC, NFS等もTCP版が開発中らしい。

* 2 :

CSMA/CD方式LAN. イーサネット (Ethernet) とも言い、イーサネットのVersion2がIEEE 802.3との互換性を持つが、用語として区別せずに使われる場合

もある。物理的にはイエローケーブルとして知られる（もちろん別の色でもよいが）1/2 インチ同軸ケーブル（10Base5）が主流だが、他に1/4 インチ同軸ケーブル（10Base2）やツイストペア線（10Base-T）もあり、10Mbps（bps = 1秒間に伝送できるビット数）の伝送速度を持つ。

* 3 :

光ファイバーケーブル用トークンリング方式LAN、FDDIとか、ANSIのX3T9.5等とも言う。100～400Mbpsもの伝送速度を持つ。

* 4 :

構内モデム対向（モデム-モデム間はツイストペア線や同軸ケーブル等で直結）、NTT等の構外専用回線（特定通信回線と呼ぶ）や無線等で隣接ホストと直接に接続される場合を指す。

NTT等を利用する場合、物理層は、ホスト-モデム/DSU等間の規約だけが問題になり、これらは、TCP/IPとは無関係に一般的に標準化されている。

データリンク層は、ホスト-ホスト間を考えればよく、これもTCP/IPとは無関係に一般的に標準化されている。ただし、BSC（Binary Synchronous Communication）手順やそのJIS版であるベーシック手順は、元々、端末接続を想定しており、伝送制御文字（例えばASCII系の0x02はSTX：テキスト開始である）を使用するのでバイナリデータの転送にはよけいなエスケープが必要だったり、確認応答が多重化できなかつたりして、転送効率が悪い。よって、汎用ネットワークでは、高速に任意のバイナリデータを転送できるHDLC（High level Data Link Control）等が向いている。

伝送路全体をユーザが用意する場合は、例えば、モデム-モデム間の物理～データリンク層の規約、相性も問題になる。

* 5 :

NTT等の回線交換サービス（電話、DDX-C、INS回線等）やパケット交換サービス（DDX-P、DDX-TP、INS-P等）、または私設交換網を経由して相手ホストと接続される場合を指す。

回線交換は、物理～データリンク層とも、コネクションの設定/解放以外は専用回線と同じであるが、パケット交換では、ホストと隣接パケット交換機との間のX.25プロトコル（パケットの組立/分解を行う機能を持たない非パケット型ホストの場合はX.28）が必要になる。X.25の場合、ホストと最寄りの交換機間のデータリンク・プロトコルとしてHDLC-LAPBと呼ばれるものを用いる。

3. 基本的通信機能のプロトコル

下位層（物理～トランスポート層）について、図2. 2をもとに紹介する。

まず、物理～データリンク層は原則的には任意であり、個別対応のSNDCPによってその差異を吸収して上位にIP層を実現するモデルになっている。それゆえ、これらの層の規格は、OSI体系と同等と考えられる（→図2. 2の注）。TCP/IPでの物理～データリンク層として現在普及しているのは、下記のようなものである。

- ・イーサネットLAN接続。
- ・デジタル専用回線上のHDLC接続（ただし、SNDCP部分の標準プロトコルが普及しておらず、メーカー固有仕様が多い）。
- ・電話網を経由する無手順接続（ただし、モデム～モデム間でエラー回復等のプロトコルを使用する場合は普通である）。この場合、SNDCP部分にSLIPと呼ばれるプロトコルが使われることが多い。
- ・X. 25パケット交換網を経由する接続（→図3. 1）。この場合、X. 25やX. 28プロトコルがSNDCPと解釈できる。

基本的には、近距離のまとまったホスト群をLANで結合し、離れたLAN間を専用回線やパケット交換網で結ぶ（インターネットワーキング）（→図3. 1）。この時、物理的にLANを延長する接続装置がリピータ、データリンク層（MAC）レベルで接続する装置がブリッジ、IP層レベルで接続する装置（ホスト）がルータである。イーサネット（10Base5）の場合、同軸ケーブルをアダプタで直列につないで両端に吸収終端をつけた全体をセグメントと呼び、そのセグメント間をリピータ、ブリッジ、ルータ等をつなぐ。これらの装置では、隣接する2つのLANの間に置いて接続する場合を、「ローカル」、LANと中継通信路（専用回線、交換網等）との間に置いて接続する場合（よって対称的に2個必要）を「リモート」と呼ぶ。

最近では、無線の利用も注目されており、衛星回線を用いて2つのLANを結びリモート・ブリッジ接続も実用化が進められている。

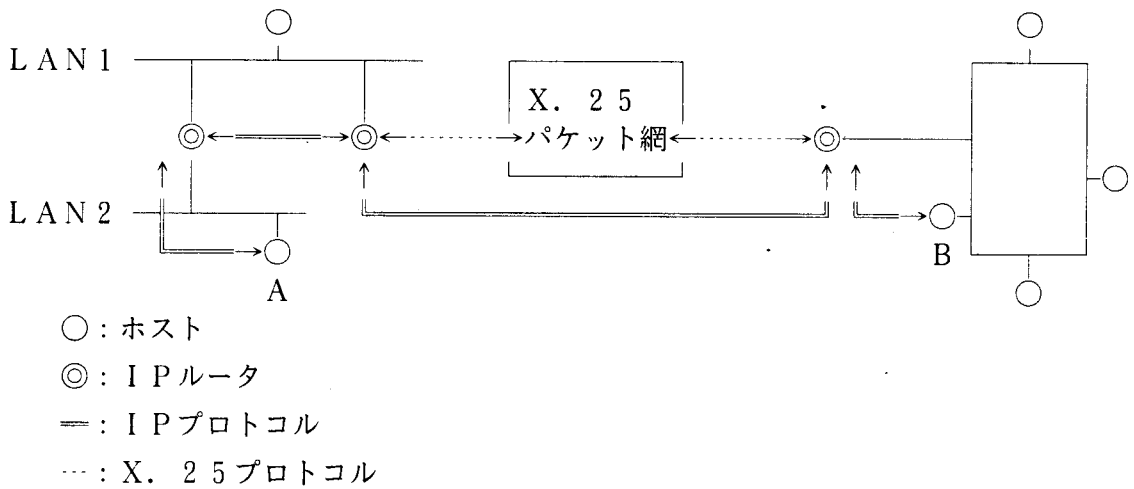
IP（Internet Protocol）は、2章でも触れたように、任意のエンドホスト間でのデータ転送を実現するための中継プロトコルで、上位の層に対してコネクショ

ンレス型データ転送（データグラム）を提供する。1つのユーザデータ（IP層SDU）は、1つまたは複数のIPデータグラムと呼ばれる単位（IP層PDU）に乗って転送される。複数に分割される場合（IP用語ではフラグメント化と呼ぶ）には、特にフラグメントデータグラムと呼ぶこともある。IP（フラグメント）データグラムは、制御情報とユーザデータからなり、その最大長はIPの下位の存在する単位ネットワーク毎に決まるMTU（Maximum Transfer Unit）に等しい。

IPデータグラムは、ルータによって中継されて、目的ホストへ届けられる。この転送では、確認応答や自動再送がなく、順序性も保証されないため、データグラムの欠落、重複、逆転等が起こる。そのため、“信頼性のない”データ転送と呼ばれており、上位の層で制御することを期待している。

他の特徴的機能としては、中継のループ等を防ぐための最大生存時間（通過するルータ数で示す）指定、正当性を確認するための制御情報に対するチェックサム、オプションとして、経路（通過ルータ）記録やタイムスタンプ等、がある。なお、一般にチェックサムとは、データを何らかの区切りで整数の並びとみなし、それらの整数値の総和を元のデータと一緒に送信し、それを受信側でデータの正当性チェックに用いる場合の、その総和値を言う。

—図3. 1—（ホストA-B間のIPネットワーク・ルーティング）



この経路制御の原理は、32ビット符号無し整数であるIPアドレスの内部形式の階層構造に基づいている（→図3.2）。標準的な方法は下記のようになる。

まず仮想的な世界で唯一のIP「ネットワーク」全体を分割して各々にネットワークアドレスを振り、必要なら各ネットワークをさらに分割して各々にサブネット

アドレスを振り、という風に階層化する。そして、始点ホストまたはルータは、
 1)宛先IPアドレスが自分の属する単位ネットワーク（複数あるかもしれない）内かどうかを調べる。そうならば、宛先IPアドレス内のホスト識別によって、その単位ネットワーク上で直接転送できる（SNDCCP/データリンク層の機能を用いて）。

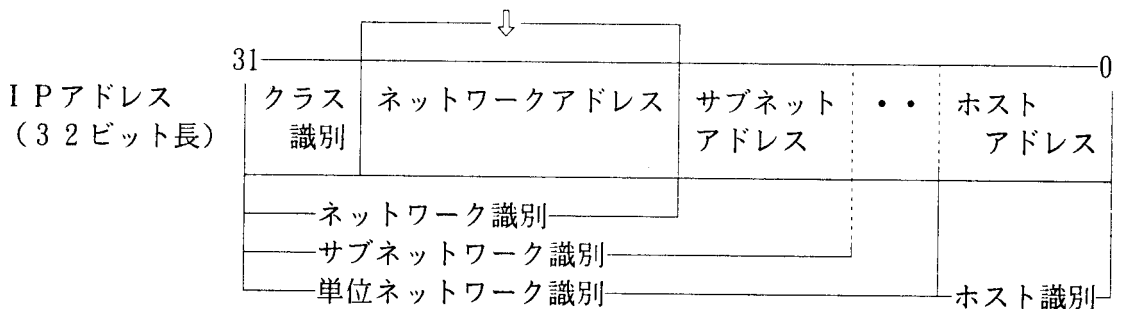
2)そうでないなら、ルーティングテーブルを順にサーチし、宛先IPアドレスに含まれるネットワーク識別（またはサブネットワーク識別、サブサブネットワーク識別等）が見つかったら、対応する中継先ルータ（自分の属する単位ネットワーク内に存在する）へ転送する。

3)そうでなくて、デフォルトのルータが定義されているなら、そこへ転送する。

ここで、（サブネット対応の）ルーティングテーブルは、仮想的な実装としては、〔ネットマスク、（サブ）ネットワーク識別、中継先ルータのIPアドレス、その中継ルートに関する評価情報〕の組の並びである。ネットマスクとは、宛先IPアドレスとビット毎の論理積を取ることでネットワーク識別（またはサブネットワーク識別、サブサブネットワーク識別、等）を取り出すためのビットマスクであり、評価情報は、動的なルート最適化において同一ネットワークへの複数のルートと比較するための情報である。

ただし、UNIXの4.2BSDのようにサブネット機能をサポートしていない製品もあり、そのようなホストでは自分と同じネットワーク識別を持つホストはすべて同じ単位ネットワーク内だと判断し、直接転送を試みる。よって、サブネットワーク内にサブネット対応でないホストが存在する場合は、擬似的にARP応答を返すような機能（proxy ARP）をあるホスト上に用意する必要がある（→〔加藤〕）。

—図3. 2—（IPアドレス形式） クラス毎にSRI-NICが管理（→2章）



—表 3. 1— (IPアドレスのクラス)

クラス	クラス 識別ビット	ネットワークアドレス長	ネットワークアドレス値	ネットワーク 内可能ホスト 台数*
A	0	7ビット	1 から 126	16, 777, 214
B	1 0	1 4ビット	1 から 16, 382	65, 534
C	1 1 0	2 1ビット	1 から 2, 097, 150	254

注* : サブネット分割しない場合の数.

このように、IPアドレスは最上位のビットパターン（クラス識別）でネットワークアドレスの長さが定められている。図 3. 2で、サブネット分割（ネットワーク内の階層化）をしない場合には、サブネットアドレスはなく、ネットワークアドレスの後ろがすべてホストアドレスになる。逆に、多段に分割する場合は、サブネットアドレスの後ろにサブサブネットアドレス等が続く。

IPアドレスの値の表記は、8ビット（1バイト）ずつを十進数表現し、小数点で区切る方法が一般的である。例えば、アドレス値が16進数表現の、0x8a0b0c0d、であれば、それはBクラスであり、「138.11.12.13」と表記する。

ここで、ホストアドレスが $\overline{0}$ （すべてのビットで1が立つ）ならば、その単位ネットワーク内でのブロードキャストを示し、ネットワークアドレスより後ろが $\overline{0}$ ならば、そのネットワーク内でのブロードキャストを示す。ところが、サブネット機能をサポートしていないホストがサブネット内に存在すると前者のブロードキャストを認識できない。また、古い仕様のホストではブロードキャストアドレスが $\overline{0}$ ではなくて0（すべてのビットが0）になっている。このような不統一が、実際のIPネットワークで大きな問題になっている。

各ホスト／ルータの持つルーティング情報の初期値はファイル等から読み込まれるが、その後、ネットワークの接続・離脱やルータの状態変更（アップ、ダウン）に応じてルーティング情報を動的に管理したい場合がある。そこで、そのための情報を定期的に交換するプロトコルがいくつかあり、中でもBSD系UNIXを中心に、単純なRIP（Routing Information Protocol）が広く普及している。RIPでは、ルータから他ルータやホストへ”自分から到達可能なネットワークとそのホップ数の一覧”を通知（advertise）する。通知を受け取った側は、それを元に自

分から他ネットワークへの最短ルートを計算し、必要なら自分の持つルーティング情報を変更する(→4章)。ホップ数とは、自分からあるネットワークへ最短経路で到達するまでに経由する他ホスト(ルータ)数である。ただし、ホップ数の最大値が14なので巨大なネットワークには向かない。

ここで、IPアドレスのようにホスト間で整数値(一般にはデータ構造)を転送する場合、ネットワーク上でのバイト列としての転送方法(バイト順)を統一しておく必要がある。なぜなら、ホスト内部(メモリ上)での表現はマシン・アーキテクチャによって異なるからである。TCP/IPでは、整数に関しては符号なしでMSB(Most Significant Bit:最上位ビット)側が先頭バイト側になるように転送する。これをネットワーク・バイト順と言う。

このようにIPは、単純な方法で柔軟なインターネットワーキングができるが、本格的な「ネットワーク」に使うには、問題もあるといわれている。

- ・アドレス長が32ビット固定。
- ・ルーティングが単純で動的最適化(代替え経路選択等)の機能が弱い。
- ・プロトコルの信頼性、性能が十分ではない。

これらは改良の研究も進められているようだが、改良版と普及版との相互接続性がなくなると結局は新しいプロトコルに置き換えることと同じなので、OSIへの移行等(→6章)との関係もあり、方向性がはっきりしていないように思える。

SNDCP(Subnetwork Dependency Convergence Protocol)は、IPの下位に存在する層対応のプロトコルの総称であり、LANの場合、ARP(Address Resolution Protocol)等が相当する。ARPは、同一単位ネットワーク内の相手ホストへ初めての転送を行なう時に、相手ホストのIPアドレスからLAN上のMACアドレスを得るためのプロトコルであり、全MACへのブロードキャストを利用した問い合わせ方式である。

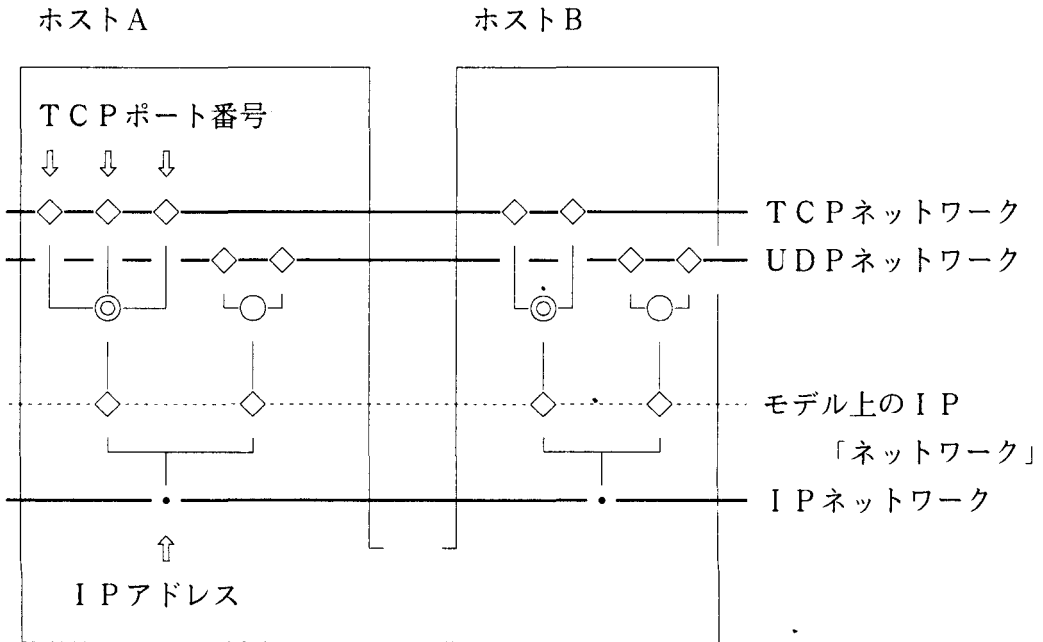
ICMP(Internet Control Message Protocol)は、IP層のプロトコルの一部であり、IPエンティティ間のエラー報告に使われるが、IPデータグラム内のユーザデータ域を使用するため、見かけ上はIP層の上位にある。

UDP(User Datagram Protocol)は、IPネットワーク上で、エンドプロセス間のコネクションレス型データ転送(データグラム)を提供するが、“信頼性”は

IPとあまり変わらない。IPとの主な違いは、2つある。

- UDPサービスネットワーク上のSAPはポートと呼ばれ、IPアドレス+UDPポート番号（1から始まる整数）で識別される。1つのIPアドレスに対して多数のポートが作れるので、プロセス毎にSAPを持つことができる（→図3.3）。
- オプションで、IPアドレスを含む全体チェックサムの検査を行うことができ、データ及び宛先の正当性確認が可能である。

—図3.3—（IPネットワークとTCP，UDPの関係）



— : ネットワーク， ◇ : SAP

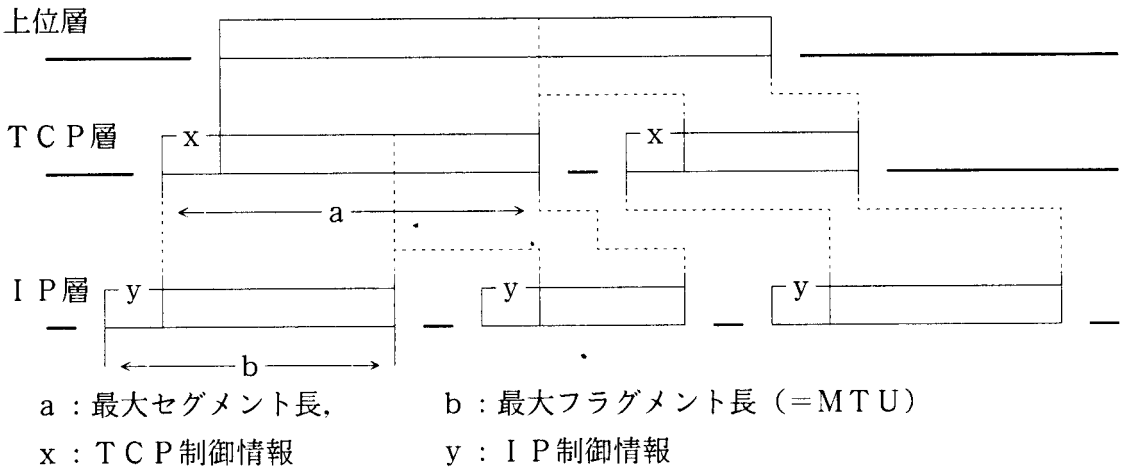
◎ : TCPエンティティ， ○ : UDPエンティティ

TCP (Transmission Control Protocol) は、IPネットワーク上で、エンドプロセス間の”信頼性のある”全二重コネクション型データ転送（仮想回線=virtual circuit）を提供する。TCPサービスネットワーク上のSAPも、UDPと同様にポートと呼ばれ、IPアドレス+TCPポート番号（1から始まる整数）で識別される。TCPコネクションは両端のポートによってのみ識別されるので、2つのポート間に1つのコネクションしか張れない（一方、OSIのコネクション型トランスポート層では2つのSAP間に複数のコネクションが張れる）。

TCP層PDUはセグメントと呼ばれ、コネクション制御、ユーザデータ転送、確認応答等の用途があり、最大セグメント長は受信側が指定できる。

ユーザデータはバイト列として扱われ、どこでセグメントに区切るかは基本的には送信側TCP層の自由である。よって、TCP層SDUは無制限長でそれが適当に分割されるとも、TCP層SDUは1バイト長でそれが適当に結合されるとも、解釈できる(→図3.4)。なお、必ずそこでセグメントを終わりにして強制送信することを上位層からTCP層へ指示(PUSH)することもできる(この情報は受信側にも伝えられる)。

—図3.4— (TCP層及びIP層と分割の関係)



TCPの技術的特長には、下記のようなものがある。

- バイト単位の可変長クレジットウィンドウを用いて、効率的な確認応答とエンド・ツー・エンドのフロー制御を実現する。一般にクレジットウィンドウとは、データ受信側が確認応答を送信側に返すようなプロトコルにおいて、送信側が確認応答を待たずに連続して送信してよいデータの範囲(量、個数等)を示し、これが大きいと、送信と確認応答のシーケンスが多重化され、ネットワーク使用効率上がる。また、動的に変更することでフロー調節ができる。
- 確認応答をタイマ監視して、タイムアウト時にデータを再送し、特にタイマ値の動的最適化を行う。
- TCP層SDUの種類として緊急メッセージをサポートしており、通常のデータが滞留していてもそれを飛び越せる。
- IPアドレスを含むチェックサムの検査によるデータと宛先の正当性確認を行う。

TCPを利用する上位層の立場から見ると、UDPと比較した場合、コネクショ

ン型でかつ信頼性が高いので、対話型応用や大量データ転送に向き、かなり使い易いが、その分だけオーバーヘッドは大きい。よって、オンライン業務でのトランザクション処理には向かないとされている。また、ブロードキャストが必要ならUDPが有効である。

同一のIPアドレス上にTCPとUDPが実現できるのは、IPの上位インタフェース及びIPデータグラム内に”プロトコルタイプ”があることによる。基本参照モデル上は、もしTCPとUDPを異なるエンティティと解釈すると、送信側IP層で上位プロトコルの違いを見てIPデータグラム内に設定し、受信側IP層では、これをみて上位プロトコルに振り分けることになる。よって、IPアドレス+プロトコルタイプが、IP「ネットワーク」のSAPアドレスと解釈できる(→図3.3)。

4. 応用サービス機能のプロトコル

応用サービス機能のプロトコルは、TCPやUDP上に実装されたものすべてが対象とも考えられ、範囲は明確でない。また、常に新しいサービスを実現するためのプロトコルの研究/開発が続けられている。開発経緯で分類すると、DARPA標準、BSD系UNIX、その他の大学系(MIT等)、その他のメーカ系(SUN等)がある。

最も標準的なもの(特に資源提供側:サーバ)には、TCP/UDPのポート番号が正式に割り当てられている(Well known port → [Comer])。それ以外でも広く普及しているものは、暗黙の割り当て番号がある。

BSD系UNIXであれば、/etc/servicesというファイルにサーバ側ポート番号の割当てが記述されている。また、netstatコマンドを使えば使用中のソケット(→5章)が表示され、実際の使用中ポート番号もわかる。

ここでは、一般的な各種応用(ネットワーク)サービスに対して、TCP/IPとしてのプロトコルの例を簡単に紹介する。

1) 電子メール、電子ニュース

自または他ホスト上のユーザのメールボックスへメール(文書)を送る機能であり、ネットワークサービス中で最も利用人口が多いと言われている。

DARPA標準のメール配送プロトコルにSMTP (Simple Message Transfer Protocol) があり、また、メール形式にはRFC 822という規格がある。

SMTPは、IPアドレスが解決しているホスト上のユーザへメールを送信する機能を持ち、送信ユーザ/受信ユーザの確認(許可)を行う。しかし、ユーザアドレスの表現形式や直接IPアドレスを知らない相手への中継方法は規定しておらず、また、障害時のメールの途中からの再送もない単純なプロトコルである。実際の中継の方法は、各電子メールネットワークの運用で決まり、多くの場合、階層的ドメイン名形式(ユーザ名@ホスト名.ドメイン名₀. . . .ドメイン名_i)のユーザアドレスを採用し、それに基づいたドメイン間の階層的な中継を行っている。

RFC 822は、1つのメールをエンベロップとヘッダ部とボディ部(後者2つをコンテンツと呼ぶ)とに分けて定義する。エンベロップはヘッダから生成される内部情報であり、ヘッダは、"フィールド名:フィールドボディ"の形をした、宛先(例えば、To:fxxxx@cc.nagasaki-u.ac.jp)のような、ユーザへの制御情報である。"Cc:"(カーボンコピー)フィールドは同報配信を定義している。ボディはユーザ間のメッセージ本体である。しかし、現状では想定がテキストデータのみであり、かつコード系については標準化されておらず、また漢字等の多バイト系コードの考慮もないので問題がある。今後、テキストデータ以外(グラフィックス、映像・音声、電子伝票等)に対応することも重要である。

電子ニュースには、NNTP等のプロトコルがある。各ユーザのメールボックスに配布されるのではなく、各システムに対して1部ずつ配布されるので、共通情報の配布・不特定多数との意見交換が効率的に行える。

2) 仮想端末

ホストA下の端末(ホストと端末間の接続は通常TCP/IPの範囲外である)がAを介して他ホストB上のAPとやりとりする(つまりホストB下の端末としてふるまう)機能である。単にホストAにネットワークを通して遠隔地の端末が実端末として接続されている場合とは違って、動的に目的ホストBを変更でき、それを可能にするために端末機能を仮想化している。ただし、端末にはピンからキリまであり、ホストとの機能分担も統一されていないので、端末制御手順を標準化して任意のホストと任意の端末の組み合わせを実現するのは、本質的に困難である(→1章)。

会話型端末の場合、ホストの利用にはログインが必要なため、「リモートログイン」と呼ぶことも多いが、一般にはユーザ認証やログインの手順の規定を含む必要はない(実際、TELNETプロトコルでは規定されていない)。一方、ただの

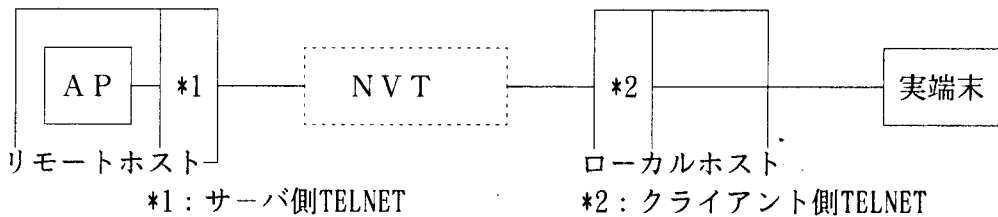
「遠距離端末」では「リモートログイン」という言葉は使わない。

DARPA標準にはTELNETがある(→[854])。TELNETではNVT(ネットワーク仮想端末、この用語は他のプロトコル系、例えばN-1でも使うが、本稿ではTELNETのものを指す)を定義し、原理的にはローカル側がNVTと実端末との変換をし、リモート側がNVTとAP/端末制御の期待している端末との変換をする。

NVTは、一種のTTY型手順による、入力(キーボード)と出力(タイプライタ)を持ったASCIIコード(もちろん伝送制御文字は含まず、また、書式制御文字の内、復帰改行、後退、ベル等の共通機能のみを規定している)の全/半二重端末であり、入力単位(文字/行)とエコー処理や、文字消去/出力中止/割込み/BREAK/制御権移動等の制御コード、それらへのエスケープ等を規定している。ASCIIコードは先頭ビットが0の8ビットとして送受信し、TELNET制御コードは0xF1~0xFF等の値を使う。

しかし、この範囲は一般の会話型AP利用においては狭いので、APや実端末に合わせて、実行時に追加規定(オプション)をネゴシエーションできる(もちろん、両方のTELNETがその追加規定をサポートしていないと意味はない)。

—図4.1 (TELNETのNVT) —

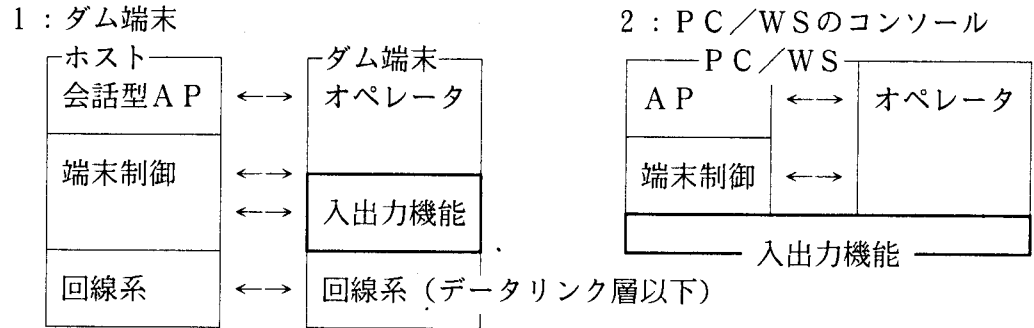


UNIXでは、元々ホストに多種類の端末をつなぐために、APから見える端末を汎用TTY端末(ASCIIコードの文字/行入力の全二重端末)としてモデル化・仮想化(上のNVTよりもう少し範囲が広い)し、モデル上の回線規約/機能と実端末の制御との可能な限りのマッピング(端末からの入力を解釈/加工する、端末への出力コードを決める、等)をホスト側の端末制御(ttyドライバ)と画面制御ライブラリ(curses)が行なう設計になっている。このマッピング情報は、gettytab(gettydefs)やtermcap(terminfo)のようなファイルで管理され、APからのioctlシステムコールやオペレータからのsttyコマンド等によって、動的に変更できる。それゆえ、TELNETでリモートログインした後、実端末に合わせてリモートホスト側の汎用TTY端末設定を変更すれば、かなりの範

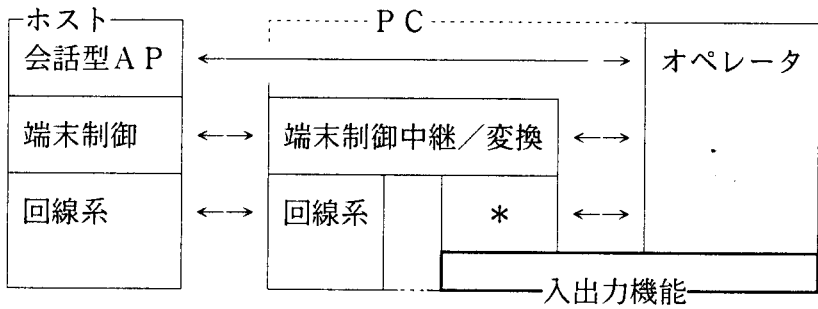
囲はうまく接続できる。t t yドライバは透過 (r a w) モードも持ち、大部分の回線規約を無効化してそのままのバイト列を通過させることもできる。

なお、UNIXのv iのような画面エディタ等では1文字入力必須/ローカルエコー不可/全二重であるが、すべてのA Pの使用時にTELNET間で入力及びエコーを1文字ずつ(1つのI Pデータグラムとして)転送するのは、効率が悪い。よって、実装によっては、TELNETがA P環境(汎用T T Y端末の設定条件)に連動して、転送単位を自動的に最適化しているかもしれない。

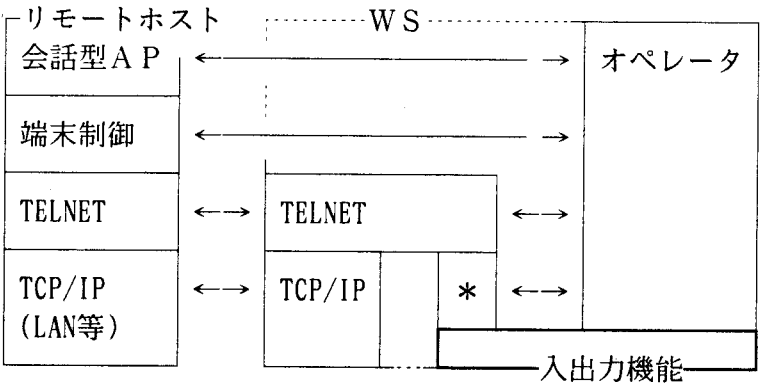
—図4. 2— (会話型端末のプロトコル階層のモデル化例)



3 : 回線接続の実端末としてのPC+端末エミュレータ



4 : TELNET接続の仮想端末としてのWS



注：

- ・入出力機能とは、ハード的組み込み機能を想定しており、オペレータとの境界はディスプレイの表示やキーの操作である。一方、端末制御との間にはコード系や制御シーケンスの規定を持つが、ダム端末の場合はそれを同位エンティティ間のプロトコル仕様と考え、PC/WSの場合は上下のエンティティ間のサービス定義と考えた。

- ・端末制御とAPとの境界はOSの作りに依存し、逆にオペレータ側は、AP、端末制御、TELNETのそれぞれとのプロトコルを持つ。なお、UNIXのシェルのようなコマンド・インタプリタは、APに含めて考える。

- ・*の部分はソフトウェア的にはrawモードの端末制御「モジュール」が存在する。原則的には透過なのでプロトコルを定義しないが、実際には何かの入力キーを食う場合もあるし、また、PC側のこの部分はフロントエンド型のかな漢字変換などを常時行う形態が多い。

- ・3の端末エミュレータには、ホスト上に対応するサーバAPを起動することでPCとホスト間のファイル転送ができるものがある。バイナリファイルの転送時には転送データが端末制御のコードとぶつからない工夫がされており、例えば、Kermitでは、アップロードの時はPC側で中身を可視文字列に変換し、ホスト側で逆変換したりする(→[藤井])。

また、会話型端末としての機能として、ホストに合わせた漢字コードの変換(例えば、シフトJISとEUC)を行うものが多い。

- ・4のWS(ワークステーション)からの仮想端末接続では、ローカルホストと実端末が一体化していると見ればよい。

- ・より複雑な実例を図6.3に示す。

BSD系UNIX間ではrloginがよく使われる。これは、UNIXのログイン規定やユーザ環境(標準入力/出力/エラー出力、端末属性-termcapのエントリー名-等)を意識した使いやすい機能を持つ。

3) ファイル転送

文字通り、異なるホスト上の2つのファイル間での全体データ転送の機能であり、TCP/IPではDARPA標準のFTPが代表的である(→[959])。

FTPでは、相手ホストに”ログイン”し、GET(相手ファイルから自ファイルへ)またはPUT(自ファイルから相手ファイルへ)する。相手ホスト上での”カレントディレクトリ”の概念をサポートしており、UNIX等のホストでは相対

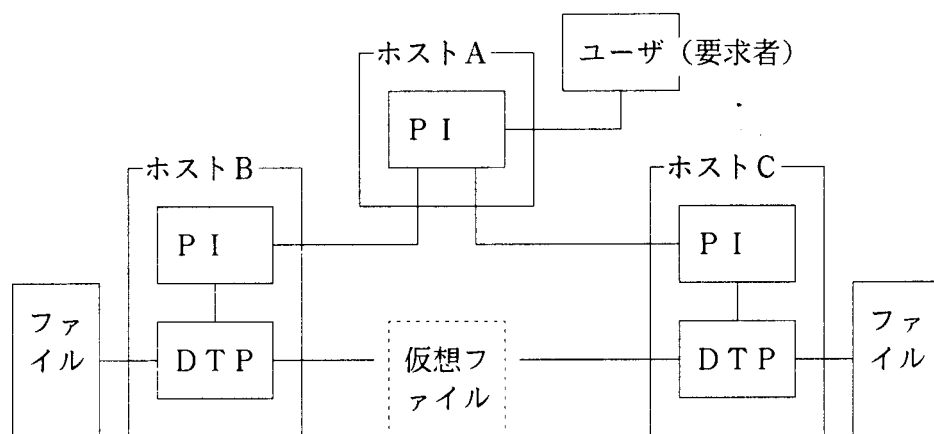
パス名指定ができる。また、異機種間転送のために、仮想的（共通的）なファイルモデルを定義し、双方のホストが転送時に実ファイルをそれにマッピング（変換）するようになっている。このモデルでは、構造=非構造/レコード列/インデックス付きページ等、型=ASCII/EBCDIC/イメージ（バイナリ）等、操作=GET/PUT/DELETE等、が規定されている。また転送方法としてストリーム（区切りなし）/ブロック/圧縮があり、ストリーム以外ではホストやネットワーク障害後の自動復旧（コネクションを再確立し最新チェックポイントから再送する）のプロトコルも定義されている。さらに、第三者転送（ホストAからの指示で、ホストB-ホストC間のファイル転送を行うこと）も可能である（→図4.3）。

内部的には、PI（プロトコル・インタプリタ）とDTP（データ転送プロトコル）とから構成され、別々のTCPコネクションを張る。前者では、TELNETプロトコルを用いて、文字列としてのFTPコマンドをやりとりする。

また、anonymous ftp という機能があり、anonymous というログイン名で自分がアカウントを持たないホストともファイル転送できる場合があり、公共的なファイルの入手等に利用される。

BSD系UNIX間ではrcpがよく使われる。内部的にrloginプロトコルを使うので、パスワードなしのバッチ的形態にも利用できる。

—図4.3—（FTPの第三者転送）



4) ジョブ転送

他ホスト上へジョブ（コマンドと入力データ）を転送して、処理を実行させ、結果（出力データ）を受け取る機能であり、ローカル/リモートホスト間の処理連携の最も単純な方法である。

DARPA標準にRJE（Remote Job Entry）があるが一般的ではないようであ

る。BSD系UNIXでは、`rlogin`プロトコルで実現される`rsh`コマンドがよく使われる。

5) リモートファイル・アクセス

他ホスト上のファイルに対してアクセスする機能であり、ネットワーク上の複数のホストのファイルシステムを結合して全体で共有すること（分散ファイルシステム）を目的としているが、いろんなレベルの分散及び分散不可視（ユーザ側がローカルファイル／リモートファイルの差異を意識しないでよいこと）がある。本当の分散を考えるなら、プロトコルだけでなく、OSとしてのファイルシステムの機能やインタフェースを共通化しないと有効性が低い。

TCP/IPでは、SUNが開発したNFS（Network File System）が広く普及している（UNIX以外にも）。これは、“リモートマウント”によってリモートファイルシステムの一部を自ファイルシステム（のファイルパス名体系）内に取り込み、ローカルファイルと同じようにAPからアクセスすることを可能にしている。リモートファイルシステム側ではアクセス状態を覚えない“ステートレス”制御なので単純で柔軟性があるが、排他制御やサーバダウン後の復旧等の信頼性には欠けている。

6) リモートデータベース・アクセス

汎用的に他ホスト上のデータベースに対してアクセスするプロトコル（特にリレーショナル型データベースへのアクセス）は、TCP/IPでの実用化はまだである。最終目的は、どの情報がどのホスト上に実在するかを知らずに検索できること（分散データベースシステム）である。

7) ネーム・サーバ

定義が曖昧であるが、たぶん「名前」と「情報」の変換を目的とする単純な分散データベースサービスといえる。

DARPA標準のDOMAIN（ドメイン体系ホスト／ネットワーク名とIPアドレス）、FINGER（ユーザ名とユーザ情報）等があり、また、SUNが開発したYP（Yellow Pages）も、ネットワーク上の各種情報（例えばパスワードファイル）の一元管理に利用されている。

8) ウィンドウ・サーバ

グラフィックス端末等のウィンドウを制御（入出力）する機能で、入出力を要求するAP側をクライアント、入出力装置（端末）側をサーバと呼ぶ。一般に複数のクライアントが同時に1つのサーバとやりとりするマルチウィンドウである。

MITで開発されたX-WINDOWが有名で、その場合、サーバ機能のみを持つグラ

フィックス端末をX端末と呼ぶ。サーバは、一般には複数のディスプレイ（端末）を持ちうるので、端末台数分のTCPポートをリザーブしておく必要がある（例えば、6000番以降を使用する）。

9) 分散トランザクション処理

複数のホスト上のプロセスが連携をとりながらデータベースの更新等を含む処理をリアルタイムに行うための、高速・高信頼プロセス間データ転送機能であり、処理の起動／終了や処理のコミット／ロールバック等も含んでいる。

商用ネットワークの利用形態（オンライン業務）では特に重要である。TCP/IPでの実用化はまだであるが、ベースとなる高信頼性データグラム通信として、UDPに代わるVMTTP（Versatile Message Transaction Protocol）が開発されている。

10) ネットワーク管理

ネットワーク全体を管理する機能で、障害監視と場所・原因解析、統計情報の収集、遠隔操作（リモートメンテナンス）等を含み、大規模なネットワークにおいては、運用（チューニング、障害防止／復旧等）に必須となっている。

最近、TCP/IPでもSNMP（Simple Network Management Protocol）が普及しはじめている。

11) セキュリティ管理

ネットワーク上での機密保護機能で、広義のネットワーク管理の一部であるが、分けて考える場合もある。ユーザの認証、共有資源へのアクセス権制御、通信データの暗号化等の話題を含んでおり、TCP/IPの適用分野の拡大による信頼性の要求やネットワークワームの問題等により重要視され始めている。

その他、いろんな応用層プロトコルが普及している。

例えば、BSD系UNIXでは、UDPのポート番号520上で、RIP（→3章）によって定期的にIPルーティング情報をroutedプロセスが交換している。さらに、UDPのポート番号111は、RPC（Remote Procedure Call）サーバが使用する。一般的に”RPC”は、他ホスト上に用意された処理（Procedure）を処理番号で呼び出す機能で、ネットワークを通してパラメタを渡し結果を受け取ることができるが、ここではSUNが開発した特定のプロトコルを指す。通常RPC上ではAP間でデータ構造をやりとりするので、マシン・アーキテクチャの違うホスト間でも各種データ構造（C言語のデータ型）の転送を可能にするための規定が必要になる。その例としてXDR（eXternal Data Representation）がある。

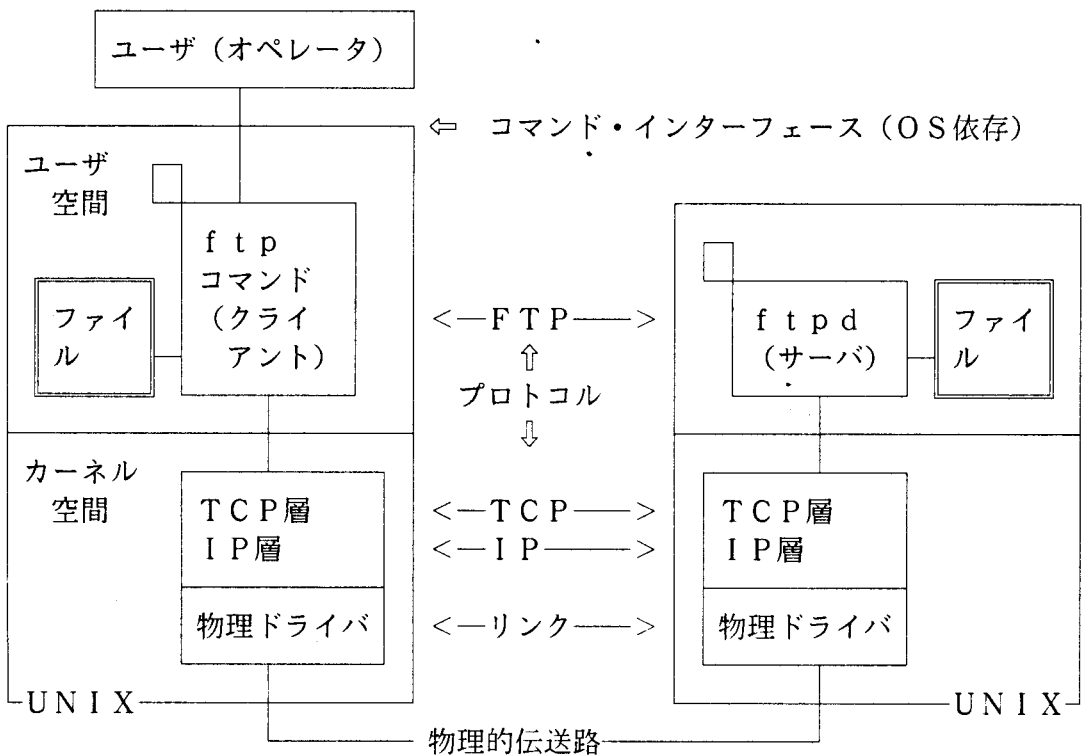
NFSやYPは自分のプロトコルを運ぶ手段として、これらを利用している。

また、日本の大学と企業の共同で作られたWnnという、かな漢字変換サーバも普及しつつある。Wnnでは、特定のホスト（サーバ）が大容量の辞書及び高度な変換機能を持ち、他のホスト（クライアント）はネットワークを介してそれを利用する。

なお、サービスだけでなく、ネットワークのテスト用のプロトコルもいくつか用意されている。

BSD系UNIXでのソフトウェア的構造を図4.4に例示する。FTP、TELNET等は、資源利用側（クライアント）からの要求に応じて、資源提供側（サーバ）の処理が起動される、いわゆるクライアント/サーバ型プロトコルである。

—図4.4— (UNIX 4.2 BSDでのFTPサービスのソフトウェア)



注：TCPコネクションは2本張られる。

これに対し、rwho等は、rwhod（デーモン）間で、定期的（例えば30秒毎に）にネットワーク上の各ホストの利用状況情報を交換し、蓄積する。ユーザからの情報参照要求時（rwho、ruptimeコマンド等）には、自ホスト内に蓄積されている最新情報を読むだけである。

なお、以前のUNIXでは、サービス毎のデーモンプロセスをブート時に1個ずつ立ち上げておき、クライアントからの要求時にforkしていたが、今のUNIXでは、ブート時には唯一のデーモン(inetd)を立ち上げ、クライアントからの要求時に種別(ポート番号)に応じて、fork&execするようである。ただし、NFSサーバのように常時アクセスされるものは、個別に常駐している。

今まで述べてきたものは、あくまでプロトコルの規定であり、一般的にはホストのOSの種類によってAPIインタフェースやマンマシンインタフェースは異なる。しかし、広く普及しているものに関しては大体は似たようなインタフェースを持ち、特にオペレータ・コマンドの形で提供される場合はそのままコマンド名になっているものが多い(telnet, ftp, rlogin等)。

ここで、注意すべきことは、あるプロトコルを”サポート”している実際の製品が規定範囲の全機能を実装しているとは限らないという点である。これは、各製品毎にマニュアルをよく見る必要がある(もちろん、自分側だけでなく、相手側も含めて)。ただし、機能範囲が異なる製品間での通信は、通常は機能の低い方に合わせる(ネゴシエーション・ダウン)ことで可能である。

TCP/IPの応用層は、OSI(の計画)に比べると、統一性、汎用性、網羅性等に欠ける面もあるが、逆に、必要性の高いものはほとんどが既に揃っており、実用性は高い。また、サービスのモデルや機能に関して、OSIの応用層に与えた影響はかなり重要だと推測される。OSIの規格は、まだ最終版が出ていないものもあり、製品としての普及ももう少し先であることを考えると、TCP/IPの応用層の意義は非常に大きいと思う。

5. UNIXにおけるプログラミングインタフェース

既存のネットワークサービスにない機能が必要になった場合や、研究や実験の目的で独自の応用プロトコルを開発する場合は、ユーザが、ネットワーク上のデータ通信機能を直接利用するプログラムを書くことになる。そのために、各言語（例えばC言語）を介してOSが提供するネットワーク・プログラミングインタフェースが存在する。UNIXにおいては、ユーザの書いたAP（アプリケーションプログラム）は通常1つのUNIXプロセスとして実行されるので、ネットワークを介したリモートプロセス間通信の機能が必要になる。

4. 2以降のBSD系UNIXには、ソケットと呼ばれるプロセス間通信機能がシステムコールとして提供されている。これは、自ホスト内のプロセス間通信だけでなく、TCPまたはUDPネットワーク上や、PUPネットワーク（XNS）上でのリモートプロセス間通信をサポートしている。

ソケットはプロセス間通信の入出力端点で、`socket`システムコールで生成され、ファイルの場合と似た記述子（descriptor）で識別されるが、その時に何らかのアドレス形式（ドメイン名）と対応付けられる。その1つがインタネットドメイン（=TCP on IPまたはUDP on IP）である。

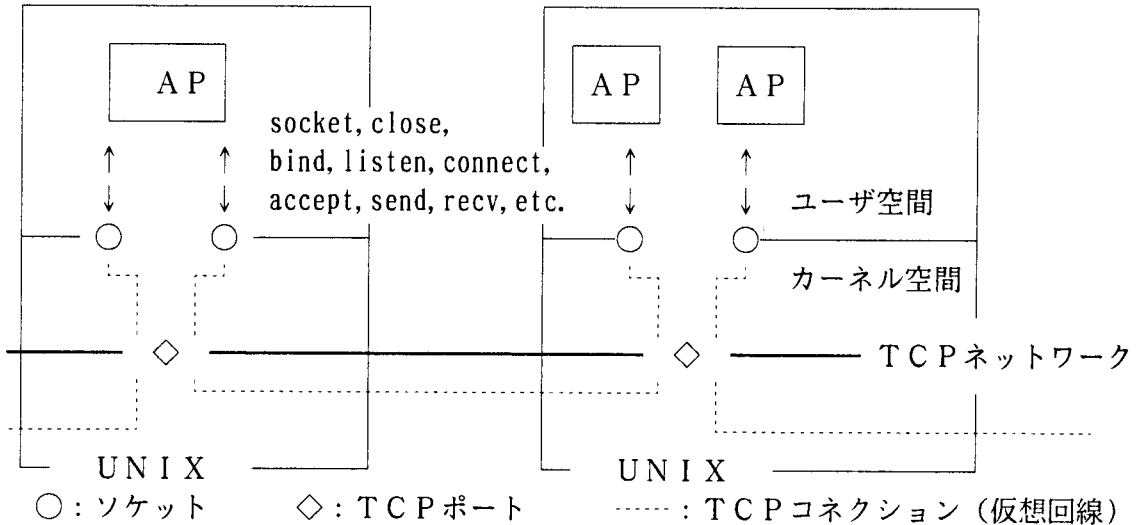
TCPの場合は、ソケットはまずTCPポートに対応づけられ、コネクション設定時には、さらにコネクション端点の意味を持つ（→図5.1）。この時、コネクション毎の多重処理を可能にするために、`fork`システムコールではソケットが子プロセスへ引き継がれる。そして、最後に`close`システムコールによってソケットが削除される。

システムV系では、今まではリモートプロセス間通信の使いやすい標準的なインタフェースはなかった（→[Bach]）。しかし、最近では、TCPやUDPのレベルのインタフェースをTLI（トランスポート・レベル・インタフェース）として標準提供している。これは、将来のOS I体系への移行（共存）を考慮している。

（→[日経1]，[日経3]）

また、カーネルの内部構造も、単なるデバイスドライバに代わってストリームという機構が開発され、`ioctl`システムコールを出せばAPが実行時にプロトコル・スタック（階層構造のモジュールの組み合わせ）を設定できる。

—図5. 1— (ソケットによるTCPネットワークアクセス)



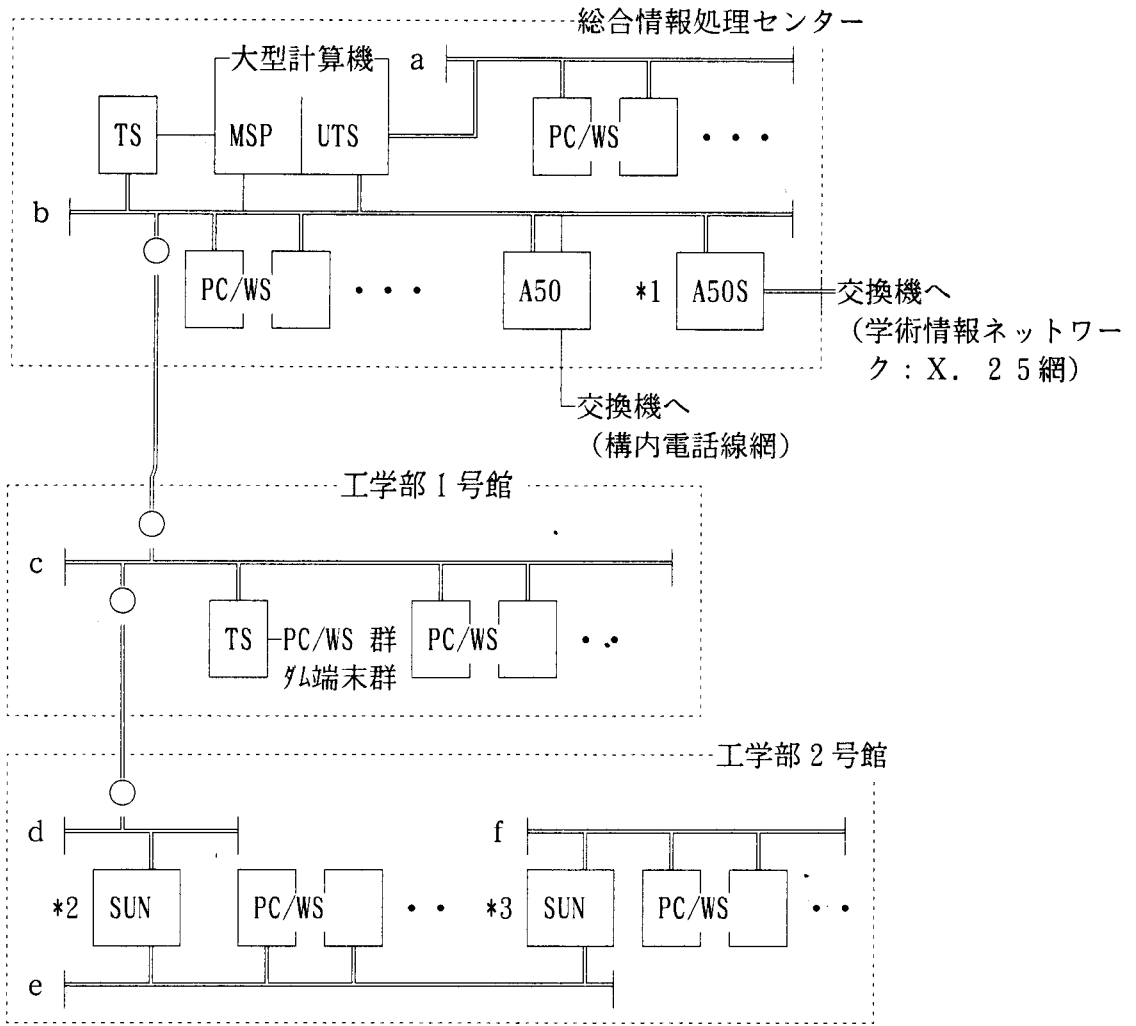
6. 長崎大学におけるTCP/IPネットワーク

長崎大学では、総合情報処理センター（以下ではセンターと呼ぶ）の大型計算機上にUNIX系OSのUTSを導入してイーサネットLANを張った時点から、いろいろな形でTCP/IPを使用している。そして、学外接続を考慮してSRINIC（→2章）のグローバルなIPネットワークアドレス（133.45.0.0）とドメイン名（nagasaki-u.ac.jp）を取得している。そして、まずJUNETに参加して電子メールレベルで学外と接続し、さらに学術情報ネットワーク上のIPネットワークであるJAIN（Japan Academic Inter-university Network）に参加してIPネットワークとして学外と接続した。この時点で他大学へのtelnetやftpも可能になった。

長崎大学IPネットワーク（対外的に認知されたネットワーク）は、今のところ文教地区の一部（センターと工学部1及び2号館）にあり、イーサネットLAN上に構築されている。現在、工学部2号館の電気情報工学では、IPルータ接続により別サブネットとしてイーサネットLANを拡張している所である（→図6.1）。

しかし、既に他学部でもイーサネット等のLANが引かれようとしているので、それらが長崎大学IPネットワークに参加してくると思われる。この時に、IPアドレスやブロードキャスト仕様等の整合の問題が発生するのを避けるためには、L

ANを引く時点で必ずセンターに相談されるようお願いしたい。
 -図6. 1- (長崎大学のIPネットワーク構成)



注:

a ~ f は、イーサネットLAN。○は、イーサネットLANを物理的に拡張するための光リピータ。サブネット構成は下記のようにになっている。

長崎大学—センターサブネット (b + c + d)
 —電気情報サブネット—専門教育用サブサブネット (e)
 —基礎教育用サブサブネット (f)

• LAN a 上には100台以上、LAN b 上には20台以上のLANボード付きFMRパソコンが接続されている。また、LAN e やLAN f には、それぞれ25台以上のSUNワークステーションが接続される。

• 「TS」はターミナル・サーバ (コミュニケーション・サーバとも言う) を指し、

通常2つの機能がある。1つは、2台のTSの端末側ポート間にTCPコネクションを張った後、その上で透過的にデータを転送する。よって、見かけ上MSPと端末とをRS232Cとモデムで直結した状態にできる。もう1つは、TSというホストに実端末がついている形態で、TS上でTELNETクライアントが動作し、他ホストへ仮想端末として接続できる。

• = がIPネットワーク接続を示す。なお、イーサネットLAN上ではIP以外に富士通独自プロトコル（FNA体系）が動作可能であり、FMR等から直接MSPへの接続及びA50からMSPへの接続は、そのプロトコルを使用している。

• *1~*3は、IPルータの役目を果たしている。*1は、dejimaというホスト名を持ち、長崎大学IPネットワークとJAINとを接続している。JAINを経由した他大学のIPネットワークとやりとりは、すべてここを通過する。*2は、センターサブネットと電気情報サブネットを結ぶ。*3は、電気情報サブネット内の2つの単位ネットワークを結ぶ。

学術情報ネットワーク（→ [川添]）は、文部省が主要国立大学（ノード校、長崎大学もノード校である）にパケット交換機を置き、それらをNTTの高速デジタル専用回線で結んで私設X.25パケット交換網を構築したもので、その上に複数種類の管理の異なるサービスネットワークやプロトコル体系が用意されている。管理上の”ネットワーク”を見ると、まず、”大学間コンピュータネットワーク”や”図書館ネットワーク”があり、N-1と呼ばれる独自プロトコル体系を用いている（前者によって長崎大のMSPから九大のMSPへログインして利用することができる）が、最近はOSIも導入され始めた。また、その他にも特定の目的の”ネットワーク”がいくつかある。そして、近年、”JAIN”という実験的なTCP/IPネットワークが新たに追加された。それゆえ、X.25パケット交換網としての学術情報ネットワーク、大学間コンピュータネットワーク、JAINへの参加はそれぞれ個別に申請する必要がある。

また、国内の他のIPネットワークとして、WIDE（Widely Integrated Distributed Environment）という大学（私立も含む）・民間を結ぶ実験ネットワークや、TISN（Today International Science Network）という東大理学部を中心とする学術ネットワークがあり、JAINともIPレベルで相互接続されている（→ [川添]）。

長崎大学でのTCP/IP利用形態の例をいくつか示す。

<<仮想端末及び遠距離端末>>

自室の端末（PC/WS含む）から，他の計算機へログインして利用する。

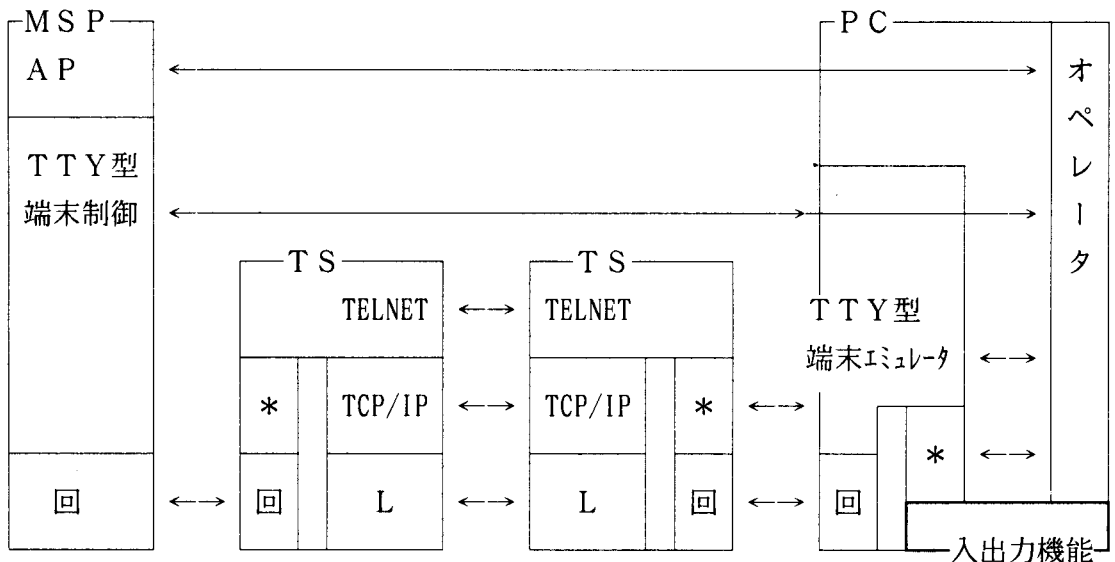
UNIX系ホストへのリモートログインには，telnetを直接利用できる。例えば，センターのFMRパソコン上ではFusionという製品名のTCP/IPソフトが動作しており，初期メニューでUTSを選択すると自動的にtelnetでログインする。また，LANボードのないPCでもTSに実端末としてぶら下がり，TSから他ホストへtelnetできる。さらに，デジタルホンからA50経由でUTSを使う場合も実はA50からUTSへtelnetしている。

MSP利用に対しては，下記のa～cのように，IPネットワークを遠距離端末の通信路として提供している。

a. MSP <-> TS <=> TS <-> PC, MSP <-> TS <=> WS

PC/WSを，TS及びLANを経由して，MSPに実端末接続する（APから見るとTTY型端末）。MSPの端末としてはフルスクリーン機能が限定される（ブロック型端末用のスクリーン制御を行うAPが利用できない，機能キーによるAPの起動ができない等）。ただし，PFDについてはTTY型をサポートしているので，この形態でも利用できる。漢字変換は，MSP側が行なう。

-図6. 2- (LANボードなしのPCからのMSP利用例1)



注：LはLAN接続。回は回線（RS232C無手順）接続。*は透過の端末制御。

b. MSP <-> A50 <=> PC/WS

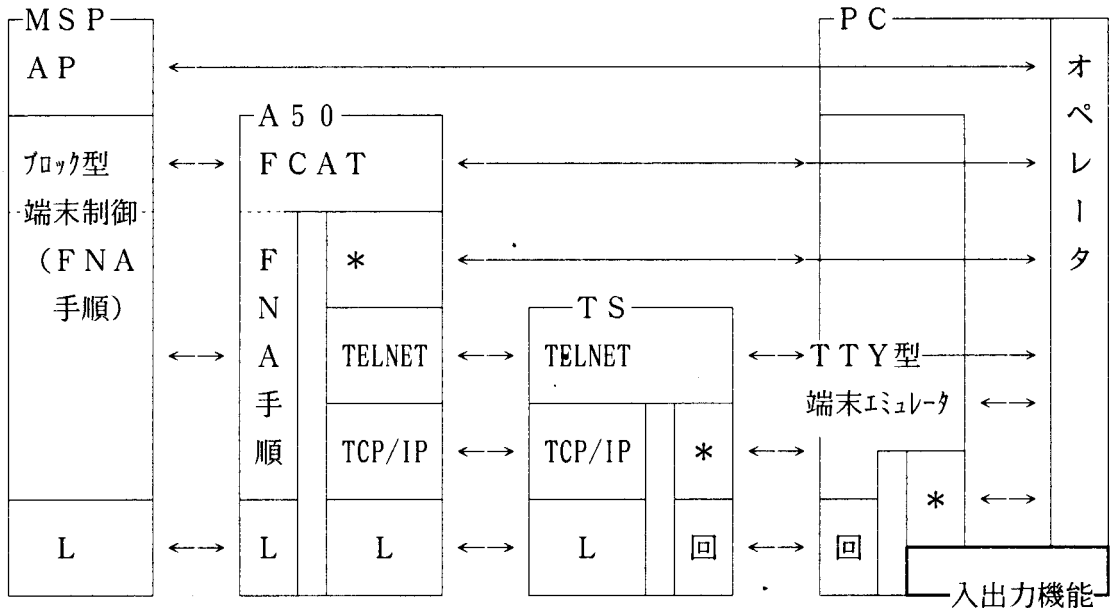
PC（LANボード付き）/WSを，LAN上のtelnetによってA50に

TTY型手順で仮想端末接続し、MSPブロック型端末エミュレータ（FCAT）を通してMSPを使う（APから見るとブロック型端末）。FCATは、マッピングができる範囲でブロック型フルスクリーン機能を実現し、漢字変換も行う。

c. MSP <-> A50 <=> TS <-> PC

PCを、TSに実端末接続し、telnetによって、さらにA50にTTY型手順で仮想端末接続し、A50上のFCATを通して、b.と同様にMSPを使う（APから見るとブロック型端末）。

図6. 3 - (LANボードなしのPCからのMSP利用例2)



注：LはLAN接続。回は回線（RS232C無手順）接続。*は透過の端末制御。

<<ホスト間のファイル転送やジョブ転送>>

例えば、UTS上のファイルを研究室のWSへコピーしたり、研究室のプリンタへ出力したり、研究室間でデータファイルを転送したり、他大学からPDS（パブリックドメイン・ソフト）を入手したり、ができる。

<<電子メール>>

センターがJUNETに参加しているので、UTSユーザ間、UTSユーザーWSユーザ間だけでなく、日本中（及び海外）のユーザとの電子メール交換が可能である。

また、BITNET（IBM系プロトコルを用いる世界的規模の電子メールネットワーク→ [川添]）やNINET（学術情報ネットワーク上で独自プロトコルを

用いてサービスされている電子メールネットワーク)とも交換可能である(例えば、九州大学がメールゲートウェイを実施しているので、そこを経由する)。

JUNET及びBITNETについては、本センターレポートの「JUNETの利用法」、「BITNETの利用について」に詳しい。

なお、センターによる電子ニュースサービスの導入(たぶんUTS上で行うことになる)は、まだである。

—表6. 1— (UTSの主なTCP/IPネットワークサービス)

サービス種別	応用プロトコル	ユーザコマンド	サーバ、デーモン (ポート)
仮想端末	TELNET	telnet	telnetd (23/tcp)
	rlogin	rlogin	rlogind (513/tcp)
ジョブ転送	rlogin	remsh	rshd (514/tcp)
ファイル転送	FTP; TELNET	ftp	ftpd (20 & 21/tcp)
	rcp; rlogin	rcp	rcp+rshd (514/tcp)
電子メール	SMTP; RFC822	mail mailx 等	sendmail (25/tcp)
他ホスト情報 *	rwho	rwho ruptime	rwhod (513/udp)

注: *は、将来サービスを中止する可能性もある。

また、UTS上ではrfcドキュメント(→2章)の一部を公開ドキュメントとし自由に参照できる。UTS上の/usr/pubdoc/rfcディレクトリ下にindex,txt0-5,txt6-8,txt9,..のようなディレクトリ(RFC番号の100番台の区切りで分類している)があり、そこにRFC959.TX1.Zのようなファイル名で置いてある。ここにはないrfcドキュメントを参照したい場合は、東大等からanonymous ftpで入手できる(→[Comer]の訳者付録)が、入手した場合は個人のWSやUTS上の個人ディレクトリに保存するのではなく後で他の人も参照できるようにUTSの上記のディレクトリの置くようにしたいので、センターに相談いただければ幸いである。

なお、これらのファイルはLempel-Ziv符号化で圧縮されており、通常のcatではなく、/usr/local/bin/zcatで表示する必要がある。

最後に、全く個人的な意見として、今後の長崎大学におけるTCP/IPネットワークの課題と思う事項を列挙する。

<<IPネットワーク環境の整備>>

今後3～5年ぐらいはTCP/IPの利用は増える一方と思われるので、IPネットワーク環境の整備が重要である。

1. 通信路としてのネットワーク設備の拡張。坂本/片淵を含めた学内各部門のホストをつなぐIPネットワークの実現。

- ・イーサネット支線LANとFDDI基幹LANに基づく、データ/音声/映像の統合通信路の全学的整備計画が検討されている（キャンパス情報ネットワーク構想）。しかし、進捗度合いによっては過渡的なつながりが必要になる可能性もある。

2. 利用拡大に備えたネットワーク管理の整備。

- ・前提条件として、センターを中心にした、長崎大学としてのネットワーク技術のレベル向上（技術面、管理面、学外動向等の知識/情報の取得）が重要である。特に今後、機種/版数の異なるホストが多数共存すると、プロトコルの詳細部分の解釈や版数の差異があり、いろいろな問題が発生する恐れもある。そのためには、ネットワークニュース等を利用した他大学関係者との情報交換も必要になる。

- ・当面は、センターがアドレス管理を行い、IPネットワークの保守の分割・障害の局所化を計るためのサブネットワーク分割を実施している。

- ・しかし、LANベースのIPネットワークは、通信路をホスト間で共有しており、元々ネットワークの運用とホストの運用が切り離せない面がある。

よって、個々のホストの運用管理者がネットワーク運用にも参加しているという意識が必要である。例えば、アドレス割当て（MACアドレス、IPアドレス）、ルーティング情報設定、ブロードキャストの実行、ネットワークへの負荷、セキュリティの問題（他ホストからの不当なアクセスに対する防御）等の管理について、各運用管理者が十分認識し、自己管理する必要がある。

- ・SNMP等のネットワーク管理プロトコルの運用（監視・障害解析・チューニング等）について、将来に備えてセンターで調査・検討を始める必要がある。

3. TCP/IPネットワークサービスの導入/サポート。

- ・センターのdejimaをメールドメインマスターとしてUTS等でJUNETの電子メールサービスを開始したが、個人WSへの配送やサブドメインの形成等が

残っている。また、ニュースサービスも導入検討中である。なお、ドメイン名サービスは当面不要と思われるが、将来、もし導入する時は電子メール側と連動させるべきである。

また、実際に有効利用するためには、ネットワーク関連情報（学内、学外）の広報や利用の手引き等の強化も必要であり、当面はセンターの活動の中でやっていくことになる。

・少なくとも現時点では日本のTCP/IPネットワークは有料の商用サービスではない。相互に接続されているホスト、ネットワーク（の中のボランティア）が協力して運用されている実験プロジェクトであり、最低限、自分のホストのネットワーク環境は自分で管理できなければ参加できない。よって、「サービスマンによる完全看護」や「手取り足取りの指導・マニュアル」が誰かから自動的にもらえる

（権利がある）という認識は正しくない。ただし、今までのボランティアの人達の遺産（ノウハウ、非公式ドキュメント、PDS等）は積極的に利用すべきである。

・将来的には、大学として、ネットワークサービスの立案／コンサル／インストール／保守等を専門に行うような会社と契約する方式が能率的かもしれない。

・学外接続「IPネットワーク」については、JAIN以外的高速ネットワークとしてWIDE計画が進められているので、センターとしてこの動向をフォローし、参加すべきかどうかを検討する必要がある。

4. 利用者の相互協力・情報交換の場のサポート。

・ネットワーク利用者の相互協力による運用を実現するためには、下記のような場が欲しい。そこでセンターによる場の提供が可能かどうかを検討するべきである。

－ネットワーク管理者／運用者と利用者との連絡

－技術的情報の交換、相談

－ボランティアグループ形成の母体

5. UTS上でのネットワーク・プログラム開発の環境。

・現在、残念ながらUTS上では、ソケット・システムコール・インタフェースはユーザ開放されていない。しかし、もし、TCP/IP上でのネットワークプログラムの開発／実験をUTS上で希望するユーザがあるなら、センターが富士通と相談することになる。その場合、TCP及びUDPのポート番号の管理をセンターが行う必要がある。

<<TCP/IPと他文化の共存>>

大学のようなマルチベンダ環境では、他文化との共存問題を解決していく必要がある。

6. 統合的な学内ネットワーク（キャンパス情報ネットワーク）.

- ・LANベースの伝送路によって統合的なネットワークを構築する場合、TCP/IPネットワークと非TCP/IP文化との共存が問題になる。
- ・例えば、センターの大型計算機の汎用OS（MSPまたは次期システムでのメーカ独自OS）に対するネットワークを介しての利用を、今後どう扱うべきかの検討が必要である。MSPがTCP/IP接続（その上でフルスクリーン端末手順が流れるように）をサポートし、また、高機能専用端末（VDS等）側もそれに対応して接続できるようになるのが、単純には理想的と思われるが、TCP/IPより先にOSI化する可能性もあり、7)とも絡む。
- ・当面は現状通り、FCATの利用及びイーサネット上の富士通プロトコルの利用で対応していこう。ただし、富士通プロトコルを通すためには、単純なルータではだめで、ブリッジ・ルータ（またはブルータ）と呼ばれる、IPパケット以外にはブリッジとして動作するような装置が必要である。
- ・いずれにせよ、多目的なネットワークにおいては、有限の経費の中で満たすべき条件を決めることは、各部局間の調整等も含み、大きな問題となろう。

7. OSIへの移行.

- ・長期的にみると、TCP/IP等のプロトコル体系は、OSIへ移行することが確実視されている。TCP/IPの生みの親であるDODも87年にGOSIP（Government OSI Profile：政府調達OSI標準仕様）の第1版を出し、また、長期的な移行準備を進めている。また、日本では新規のマルチベンダ・ネットワーク（特に官公庁関係）の多くがOSIを採用している。ただし、応用層については最終規格が出ていないものもあり、段階的導入になる。

そうはいつても、既存のTCP/IPネットワークが急になくなるわけではなく、重要なのは共存の形態である。共存のレベルとしては下記のようなものがある。

- －物理的／論理的ネットワーク設備を共有できること（混合）、
- －TCP/IPネットワーク上のホストとOSIネットワーク上のホストとの間で電子メールのような応用サービスレベルの連携ができること（応用層ゲートウェイ）、
- －実製品においてオペレータやAPに対するインタフェースが統一され、ユーザに違いを意識させないこと、

長崎大学においても、PC/WSや大型計算機でのOSIのサポート状況、及び

学術情報ネットワークでの移行状況に注目する必要がある。

これらの技術的課題に先立つ問題として、今後、全学的なネットワークに対して、どのような体制で責任を持ち、整備・運用していくかが重要である。このようなネットワークはセンターの計算機環境を利用するための単目的のものではなく、むしろそれ以外の利用形態が研究／教育／事務のそれぞれの目的で増加すると思われ、設備面でも運用面でも、全学的な合意・調整・協力の下でのネットワーク管理組織が必要である。少なくとも、現在のセンターだけでは人的・制度的にそのような機能はかなりむずかしいと思う。

欧米ではこのようなキャンパス・ネットワークはかなり進んでいて、組織的に運用されているそうであるが、日本ではどの大学でも全学的ネットワークの整備は最近のことであり、同様の問題を抱えている（ネットワークに対して、設備化だけでなくその後の運用にも十分お金をかける必要があり、またそれだけの価値がある、という認識が不足している）。例えば、東北大学では、89年10月からTAIN Sという統合的全学LANの運用が開始され、ここでは、全学的組織であるTAIN S運営委員会が管理分担、費用分担、アドレス管理等の決定を行い、日常の運用は学科、学部等の単位の「運用管理者」に任されている。しかし、実際には、「TCP/IP利用者の会」、「TAIN S研究会」、「UNIX研究会」等のボランティア組織が、技術普及、相談、調整等の活動を行い、利用の手引等の作成には大学生協も協力している。

今後、他大学とも協調しながら、長崎大学の方針を決め、体制づくりを行っていく必要があると思う。

7. 参考資料

- [石田] : コンピュータ・ネットワーク : 石田晴久ら編 : bit 増刊 : '86
- [福永] : コンピュータ通信とネットワーク : 福永邦雄 : 共立出版 : '87
- [Comer] : TCP/IPによるネットワーク構築 : D. Comer / 村井純ら訳 : bit 別冊 : '90
- [川添] : キャンパス・ネットワークング : 川添良幸ら訳編 : bit 別冊 : '90
- [TWG] : インタネットワーク 入門 : The Wollongong Group編 / 白鳥則郎ら訳 :
コンピュータ&ネットワーク LAN 別冊 : '89/11
- [村井1] : UNIXのネットワーク機能 : 村井純 : 情報処理, Vol. 7, No. 12 : '86

- [村井2] : UNIXワークステーション I : 村井純ら : アスキー出版 : '87
- [Bach] : UNIXカーネルの設計 : J. Bach / 村井純ら訳 : bit 別冊 : '90
- [JIS] : 開放型システム間相互接続の基本参照モデル : JIS X 5003(ISO7498) : '87
- [854] : Telnet Protocol Specification : J. Postel, J. Reynold : RFC854 : '83
- [959] : File Transfer Protocol : J. Postel, J. Reynold : RFC959 : '85
- [SUN1] : UNIX Interface Reference Manual : マニュアル : Sun Micro. : '86
- [SUN2] : Commands Reference Manual : マニュアル : Sun Micro. : '86
- [SUN3] : Networking on the Sun Workstation : マニュアル : Sun Micro. : '86
- [UTS] : UTS ネットワーク説明書(V21用) : マニュアル : 富士通 : '88/12
- [村上] : インタネットワーキング : 村上健一郎 : コンピュータ&ネットワーク LAN : '89/10
- [加藤] : サブネットについて : 加藤朗 : UNIXマガジン : '88/11
- [藤井] : Kermitをめぐる問題について : 藤井啓文 : bit : '91/2
- [日経1] : TCP/IPからOSI への移行方式 : 記事 : 日経コミュニケーションズ : '89/11
- [日経2] : インタネットワーキング の落とし穴 : 記事 : 日経コミュニケーションズ : '90/1
- [日経3] : ソケット方式からストリーム/TLI 方式へ : 記事 : 日経コンピュータ : '89/3
- [TAINS] : TAINS のやさしい使い方 : TCP/IP利用者の会 : 東北大生協 : '90

本稿でのTCP/IPの概説(2, 3, 4章)は、主に[Comer], [福永], [TWG], [村井1], [村井2], [854], [959]に依ったが、筆者の誤解があるかも知れないので、気付かれた方は御指摘いただければありがたい。個々のプロトコルの詳細に興味がある人は、[Comer]にRFCに関する詳しいリストが載っているので、それをインデックスにするとよいと思う。また、UNIXにおける使い方や実装方式については、各々のマニュアルを読むべきだが、[村井1], [村井2]に概説がある。

実際のネットワークやそれに参加しているホストの運用管理を担当する人には、各種ノウハウや技術情報が必要になる。[村井2], [日経2], [TAINS], [加藤], [川添]等が参考になりそうだが、最新情報をつかむには、各種雑誌やJUNET上のネットワークニュース等の活用が最も有効だと思う。

なお、各資料の間で言葉の定義、使い方が多少は異なる所もある。どれか1つが正しいというのではなく、その辺の多義性・可変性を理解して読む必要がある。

また、ネットワークの世界はまだまだ発展中であり、半年前の話の内容が既に古くなっていることもあるので注意すべきである。