ネットワーク・プロトコルとは何か（TCP/IP を例として）

富士通株式会社　関西営業本部
寺島　兼司

1　はじめに

最近のコンピュータネットワークの普及にともない新聞、雑誌などでよく「プロトコル」や「 TCP/IP 」と言う言葉が見られるようになりました。また、神戸大学においても 94 年 3 月末の神戸大学情報ネットワークシステム（KHAN）の完成にともない、主として「TCP/IP というネットワークプロトコル」を使用したコンピュータネットワークが完成するに至りました。ところが、「プロトコル」と言う言葉も「TCP/IP」と言う言葉もコンピュータの専門家でないと理解できていないのが現状です。

ここでは、なるべく分かり易い形でプロトコルや TCP/IP について説明するとともに、パソコンなどを KHAN に接続する時に必要となる最小限の知識を紹介することとします。

2　プロトコルとは

プロトコルとは何でしょう。手元の英和辞典で調べてみますと「プロトコル（Protocol）、議定書（を作る）、条約などの原案（を作る）、外交上の儀礼」と言う訳がついています。また、最近の新しい辞書ではコンピュータ用語として「通信の手順」などと言う訳も出ています。プロトコルの元々の語源は外交上の国と国の間の「約束ごと」、「外交手続き」というような意味であったように思えます。

2.1　生活の中でのプロトコル

プロトコルを「約束ごと」、「手順」と考えた時、生活の中にもたくさんのプロトコルらしき物が存在します。ここでは、富士通の京橋の事務所近くにあるラーメン屋さんに行った場合を考えて見ます。

1　標準的なプロトコルの場合

客、ラーメン屋さんの暖簾をくぐる。
店員：いらっしゃいませ。ご注文をどうぞ。
客　：サービス定食。
店員：サービスはお昼しかやっていませんが？
客　：それじゃ、カラアゲ定食。
店員：ラーメンは何にしましょう。
客　：醤油ラーメンでお願いします。
店員：以上でしょうか。
客　：ビールをお願いします。
店員：生とビンがありますが。
客　：生をお願いします。
店員：以上でしょうか？
客　：はい。
店員：暫くお待ちください。

2　ハイレベルプロトコルの場合（その１）

客、ラーメン屋さんの暖簾をくぐる。
店員：いらっしゃいませ。ご注文をどうぞ。
客　：カラショー、生一丁。
店員：以上でしょうか？
客　：はい。
店員：暫くお待ちください。

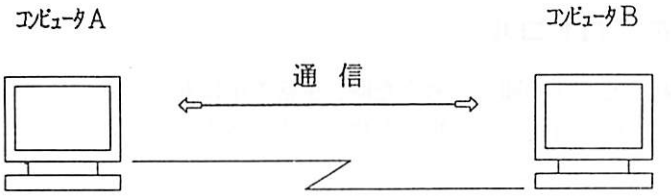
3　ハイレベルプロトコルの場合（その２）

（いわゆる、常連さんの場合）
客、ラーメン屋さんの暖簾をくぐる。
店員：いらっしゃい。
客　：いつもの。
店員：…

このように、上記の例で上げた会話がうまく成り立つためには客と店員の間でいくつかの「約束ごと」や「手順」が必要となります。たとえば、「会話は声で行う」、「会話は日本語で行う」、またハイレベルプロトコルの例ではこの店独自の言い方「カラショー」、「生」、常連さんの「いつもの」が通じることが必要です。また、日常会話の中でこのようなプロトコルの例を知りたければ、某ハンバーガー屋さんや某ファミリーレストランに行き、「いらっしゃいませ。… へようこそ」から始まる会話を体験すればよいでしょう。

2.2　コンピュータ間通信でのプロトコル

コンピュータの世界でプロトコルと言った時、プロトコルは通常「通信手順」、「通信規約」と訳されます。それでは下記のようにコンピュータＡとコンピュータＢの間でコンピュータ間通信が正常に行われるためには何を予め決めて置く必要があるでしょうか。以下に例を上げます。



通信が成り立つために決めて置く必要のある事柄（例）

1. どのような回線を使用して通信を行うか？

   イーサネット、NTT の専用線、RS-232C etc.

2. どのくらいの速さで通信を行うか？
3. 受信したデータの内容に誤りがないか判定するには？
   また、誤っていた時の対処方法は？

   誤り：　ノイズ、データ抜け、順番間違い etc.

   対処：　再送信、受信データから復元 etc.

4. どちらからデータを送るのか？

   同時に送れるのか、交代交代なのか？

   一方通行なのか？

   相手に送信権を譲る時はどうするのか？

   緊急データが出た時の割り込み方法は？

5. どのような道順でデータを送るのか？

   神戸 ⟶ 大阪 ⟶ 東京 ⟶ ニューヨーク

6. どのような言葉でデータを送るのか？

   ASCII コード、JIS コード、シフト JIS コード、EUC コード etc.

7. 会話の構成はどのようにするのか？

   会話形式、バッチ（一括）形式

2.3　プロトコルの種類

コンピュータ間通信で使用されるプロトコルは今までの説明で出てきた TCP /IP 以外にあるのでしょうか。実は世の中にはたくさんのプロトコルが存在します（表 1）。プロトコルには IBM や富士通、日本電気のようなコンピュータメーカが独自に定めた「メーカ独自規格」や ISO や JIS のような団体が定めたいわゆる「標準規格」、「コンピュータの利用者団体が定めた規格」などのたくさんの種類があります。ただ、残念なことに、日本人と英語を喋るアメリカ人が通訳を通してでないと会話ができないように、お互いのプロトコルを理解できないコンピュータ間では通信ができません。そこで最近では色々なメーカのコンピュータを接続する時はなるべく「OSI プロトコル」のような「国際標準規格」や「TCP/IP プロトコル」のような「事実上の標準規格 （Defact Standard） 」を使用するようになりました。

表 1　代表的なプロトコルの例

よく使用するコンピュータ	プロトコルの総称	プロトコルの種類
UNIX マシン	TCP/IP	TCP, UDP, IP, ICMP, ARP, TELNET, FTP, SNMP, SMTP, NNTP, HTTP, …
MS-DOS パソコン	NetWare	IPX, SPX, NPC, …
Mac パソコン	AppleTalk	DDP, RTMP, AEP, ATP, ZIP …
IBM 社メインフレーム	SNA	SDLC, RSCS, 3270, …
富士通メインフレーム	FNA	RJEP, HICS, F6650, …
DEC 社ミニコン	DECnet	DPR, NSP, SCP
ISO 標準プロトコル	OSI	FTAM, MOTIS, VT, CMIP, CLNP, CONP, …

2.4　OSI の参照モデル

プロトコルのお話をする場合に必ず避けて通れないのが OSI（Open System Interconnection）の階層化モデルです。これは ISO（International Organization for Standardization）が OSI プロトコルを作成する時にプロトコルは「7 つの機能で出来ていて各機能はお互いに上下関係で組み合わされている」と説明したのが OSI の参照モデルです。現在、世の中にあるプロトコルで明確に 7 層に分割できるものはあまりありませんが機能としては 7 層の機能をすべて備えてます。たとえば、後でお話する TCP/IP については 4 層ないしは 5 層に分けることが可能です。表 2 に OSI の参照モデルの意味、表 3 にその例を紹介します。

表 2　OSI の参照モデルの意味

層	層の名前	機　　能
7	アプリケーション層	利用者に実際の通信サービスを提供する。仮想端末機能やメール、ニュース、ファイル転送機能などがあります。
6	プレゼンテーション層	異なるデータの表現形式をある一定の形式にしたり、暗号化や圧縮化をしたりします。
5	セッション層	転送経路の確保と転送の中断や再開、送信権の制御をします。
4	トランスポート層	データを確実に相手に届けるための転送確認、誤り制御、転送順序の確認を行います。
3	ネットワーク層	直接通信できない場合の通信経路の選択とデータの中継をします。
2	データリンク層	直接通信できる装置間で通信を行います。また、転送を保証するために誤り制御も行います。
1	物理層	通信データを電気的に表し相手に届けます。

表 3　OSI の参照モデルに当てはめた例

OSI の層		ラーメン屋の場合	TCP/IP の場合
7	アプリケーション層	ラーメンの注文 ラーメン代の支払い	TELNET(TELecommunications Network) FTP(File Transfer Protocol) SMTP(Simple Mail Transfer Protocol) NNTP(Network News Transfer Protocol) HTTP(Hyper Text Transfer Protocol) NFS(Network File System) NIS(Network Information System) など
6	プレゼンテーション層	言葉の定義 ミソラーメン、 生、いつもの、 カラショー	
5	セッション層	いらっしゃいませ、 ご注文は。 ご注文ありがとう ございました。	
4	トランスポート層	もう一度どうぞ。 お客さん早く決めてよ。	TCP(Transmission Control Protocol) UDP（User Datagram Protocol）
3	ネットワーク層	伝言、メモ	IP（Internet Protocol）
2	データリンク層	日本語、英語、文法	イーサネット、FDDI、ATM、 フレームリレー、ISDN、HDLC、 PPP、SLIP など
1	物理層	声、電話	同軸ケーブル、ツイストペアケーブル、 光ケーブル、無線、電話、RS-232C など

2.5　TCP/IP とは

TCP/IP とは（Transmission Control Protocol/Internet Protocol）の略で、もともとはアメリカの国防総省（DOD）の主導で DARPA（Defense Advanced Research Project Agency）という機関が設立され、国防のためのコンピュータネットワークの研究から生まれたプロトコルです。ところが、カルフォルニア大学バークレイ校で 1981 年ごろにコンピュータのオペレーティングシステムである UNIX（4.1BSD）にこのプロトコルを実装したことから、UNIX の標準通信プロトコルとして広がり、最近は UNIX 以外のコンピュータでも標準的に使えるようになりました。現在では世の中で最も普及した「事実上の標準」プロトコルとしての地位を築いています。TCP/IP の特長を上げると以下のようになりますが何と言っても最近流行の「インターネット」の標準プロトコルである所が重要な点です。

TCP/IP の特長

1. コンピュータ間で最もよく使われているプロトコル。

   パソコン、ワークステーション、メインフレーム、スーパコンピュータ何でも使える。
2. 国際ネットワーク「インターネット」を構成するプロトコル。
3. 仕様が RFC（Request For Comments）として公開されており、誰でも入手できる。

   コンピュータメーカ固有のプロトコルでないため、誰でもコンピュータに組み込める。
4. ネットワーク利用者が使いながら改良、開発して来たプロトコル。

   使い勝手がよい、幅広い互換性、日々進歩するプロトコル。
5. 動作検証済、動作実験済のソフトウェアがすぐに入手できる。

   動作検証が済まないと標準仕様と認められない。そのため、仕様ができた時にはソフトウェアが完成している。

3 イーサネット上での TCP/IP プロトコル

前節ではプロトコルの概念と TCP/IP プロトコルの特長について述べてきました。それでは、TCP/IP プロトコルは神戸大学の KHAN（神戸大学情報ネットワークシステム）上をどのような形で流れているのでしょうか。ここでは、大学内の建屋に設置されているイーサネットケーブルを例に取って説明します（ちなみにこれから説明する事を OSI の参照モデルに当てはめると、第 1 層の物理層と第 2 層のデータリンク層に当てはまります)。

3.1 イーサネットの種類

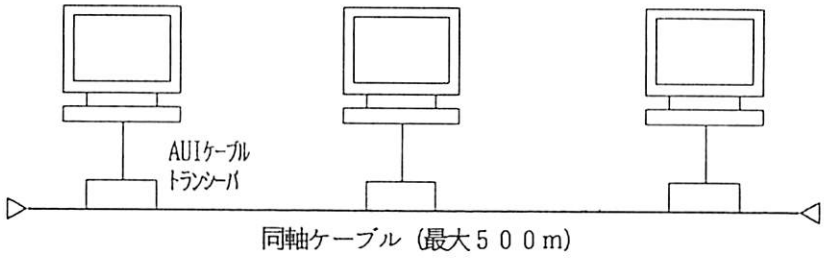
まず、イーサネット（Ethernet）の簡単な紹介をしましょう。LAN（Local Area Network）と言うとイーサネットと言う言葉が思い浮かびますが実はこの「イーサネット」と言う言葉は元々の開発元である米国 XEROX 社の登録商標なのです。その後このイーサネットの規格を元にして IEEE（米国電気技術者協会）や ISO で IEEE 802.3 や ISO 8802.3 と言う LAN の規格が規定され、LAN そのものの規格は IEEE 802.3 や ISO 8802.3 にほぼ集約されました。しかし習慣的にこの IEEE 802.3 や ISO 8802.3 規格の LAN についてもイーサネットと呼んでいます。(IEEE 802.3 と ISO 8802.3 の規格は同じです。またイーサネットと IEEE 802.3/ISO 8802.3 は少し異なる所もありますがほぼ同じです)

イーサネットの種類には以下のような 3 種類があり、神戸大学の建屋内に設置されているのは 10BASE5 です。

1. 10 BASE 5

- 太い同軸ケーブル（イエローケーブル）を通信経路として使用した LAN

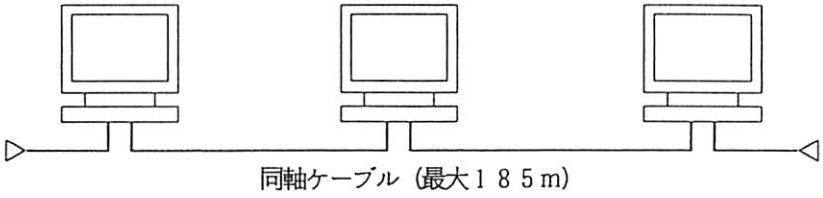
  ケーブルの長さは最大 500m。端末台数 100 台。トランシーバでケーブルに接続。



2. 10 BASE 2

- 細い同軸ケーブル（Thin ケーブル）を通信経路として使用した LAN

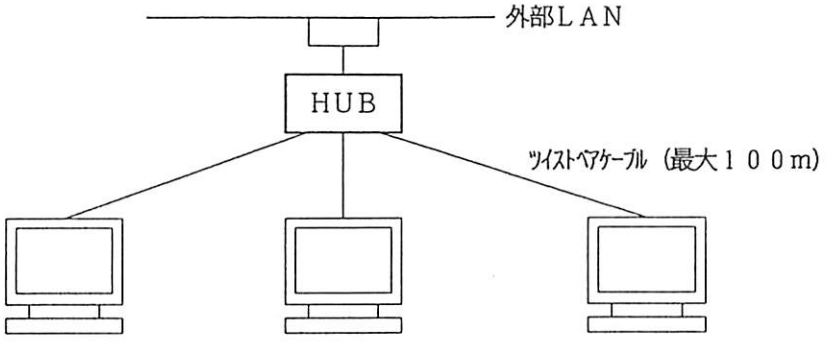
  ケーブルの長さは最大 185m。端末台数 30 台。トランシーバ付 LAN ボードを使用。教室内 LAN や事務室内 LAN のような室内 LAN に適する。



3. 10 BASE-T

- 集線装置（HUB）とツイストペアケーブルで LAN を構成

  集線装置から端末までは最大 100m。一台の集線装置に最大 30 台程度の端末を接続。集線装置は 10 BASE 5 や 10 BASE 2 で外部へ接続。



3.2 データの階層構造

前節の OSI の参照モデルの所でプロトコルは幾つかの機能階層で構成されると言う説明をしました。そして TCP/IP プロトコルでは 4 層ないし 5 層に分けられると言う説明をしました。その一番分かり易い例がイーサネット上でのデータの階層です。図 1 が示すようにイーサネット上を流れる「イーサネットフレーム」の中には「 IP パケット」が含まれ、IP パケットの中には「 TCP セグメント」が含まれると言うような階層構造を取っています。そして、イーサネットフレームが OSI の参照モデルで言う所の 1 ～2 層（物理層、データリンク層）、IP パケットが 3 層（ネットワーク層）、TCP セグメントが 4 層（トランスポート層）で見た時のデータの形式に当たり、コンピュータとコンピュータが通信する時にはこれらの層どおしで互いにやり取りが行われます。

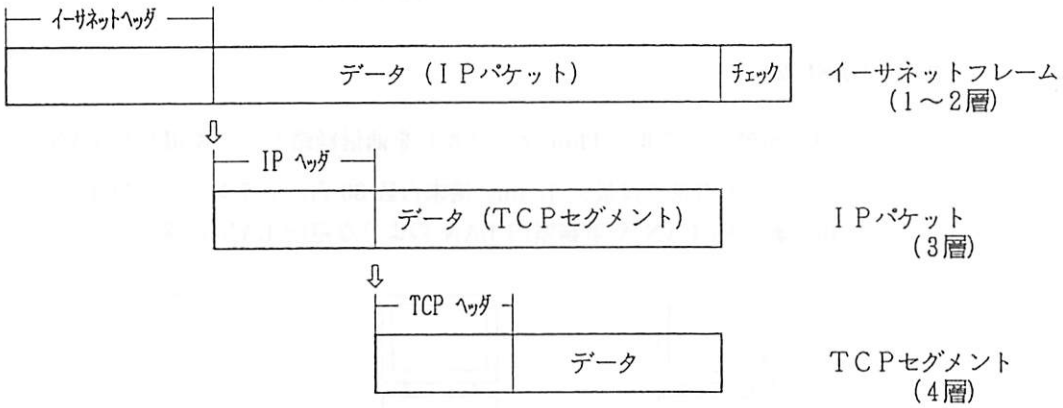


図 1　TCP/IP データの階層構造

3.3 イーサネットフレームの構造

それではイーサネットフレームの内容をもう少し詳しく見てみましょう。図 2 がその詳細図です。イーサネットフレームの先頭には 8 バイトの端末間で通信の同期を取るためのプリアンブルがあり、次に宛先 Mac アドレス、送信元 Mac アドレス、プロトコルタイプ、そして、データ（即ち IP パケット）、そして最後に伝送エラーチェックのためのシーケンスが含まれています。

この中でプロトコルタイプはデータ部分に含まれるプロトコルのタイプを示し、たとえば、TCP/IP では X'0800' や X'0806' の値が、Apple Talk の時は X'809B' や X'80F3' の値が入っています。

次に、Mac アドレスはハードウェアアドレスやイーサネットアドレスとも言い、アドレスはコンピュータや端末、 LAN 接続のためのボードなどが製造メーカから出荷される時にあらかじめ付けられ、決して重複しない固有の値を持ちます。ただし、相手 Mac アドレスとしてすべてのビットが 1（X'FFFFFFFFFFFF'）を指定した時は特別な意味を持ち、 LAN 上に接続される全ての機器に対して送信したい時、すなわちブロードキャストを示します。

8 バイト	6バイト	6バイト	2バイト	46 ~ 1500 バイト	4バイト
プリアンブル	宛先 Macアドレス	送信元 Macアドレス	プロトコル タイプ	データ	エラーチェック シーケンス

1. プリアンブル：ハードウェアの同期を取るための信号
2. プロトコルタイプ：使用しているプロトコル

```
X'0000'～X'05FF'   IEEE802.3 形式フレーム
X'0800'、X'0806'   TCP/IP
X'8137'、X'8138'   NetWare
X'809B'、X'80F3'   AppleTalk
```

3. 宛先・送信元 Mac アドレス：各機器ごとにつけられる固有のアドレス。オール 1 はブロードキャスト。上位 3 バイトはIEEE より割当られたメーカ識別番号、下位 3 バイトはメーカ内で固有番号（順番）。上位 3 バイトの値は、

```
X'00000E'   富士通
X'080020'   Sun
X'080046'   Sony
X'00000C'   CISCO       など。
```

図 2　イーサネットフレームの形式

3.4　イーサネットフレームの取り込み方法

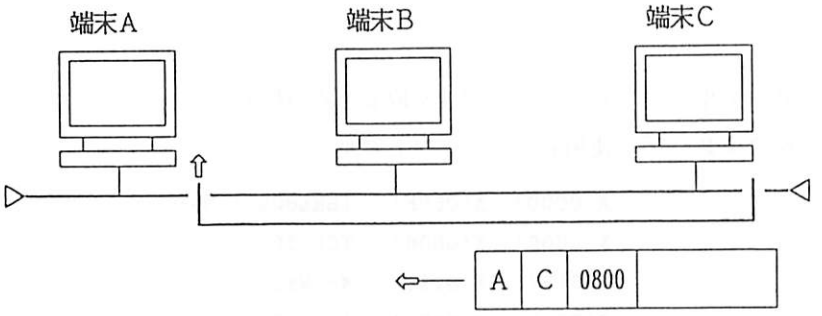
次にイーサネット上を流れているイーサネットフレームの取り込み方について説明しましょう。たとえば次ページ図 3 のように、一本の LAN 上に端末 A、B、C が接続され、それぞれの端末が Mac アドレス MacA、MacB、MacC を持つネットワークがあったとします。

まず、特定の端末どうしで通信する場合はどのようになるのでしょうか。端末 C が端末 A に対してイーサネットフレームを送るとした場合、端末 C が送信するイーサネットフレームの宛先 Mac アドレスには MacA が、送信元 Mac アドレスには MacC が、そしてプロトコルタイプには TCP/IP を示す X'0800' が入ることになります。そして端末 C から送信されたフレームを端末 A は宛先 Mac アドレスが MacA のため自分宛のフレームと認識し取り込み、端末 B は自分宛でないと認識して無視します。フレームを取り込んだ端末 A はプロトコルタイプを見て TCP/IP であるため、TCP/IP の通信制御ソフトウェアにフレーム内のデータ部分、すなわち IP パケットを渡します。

次にブロードキャストの場合はどのようになるのでしょうか。先程と同じように端末 C がイーサネットフレームを送信します。その時のフレームの宛先 Mac アドレスにはブロードキャストを示すオール 1 が、送信元 Mac アドレスには MacC が、プロトコルタイプには X'0800' が入ります。次にそのフレームを端末 A も端末 B も宛先 Mac アドレスがブロードキャスト

のため取り込み、プロトコルタイプを確認してから TCP/IP の通信制御ソフトウェアにデータ部分を渡します。

1. 特定の宛先の場合



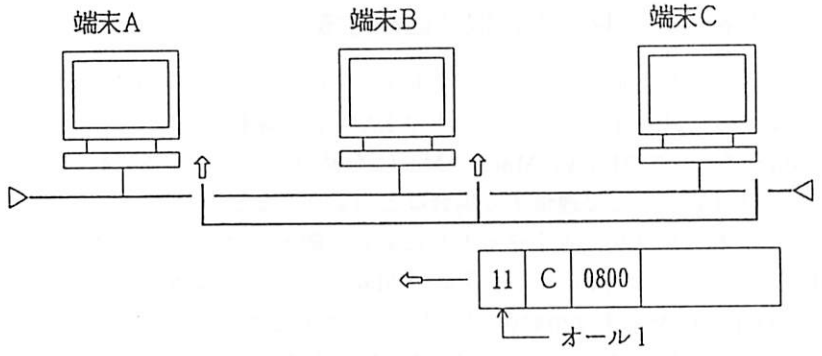
(1) 端末 C が端末 A へフレームを送信

(2) 端末 A は自分宛のためフレームを取り込む

(3) 端末 A はプロトコルタイプが X'0800' のため TCP/IP と認識し、TCP/IP 通信プログラムにフレームを渡す

(4) 端末 B は自分宛でないためフレームを無視する

2. ブロードキャストの場合



(1) 端末 C がブロードキャストフレームを送信

(2) 端末 A、端末 B はブロードキャストのためフレームを取り込む

(3) 端末 A、端末 B はプロトコルタイプが X'0800' のため TCP/IP と認識し、TCP/IP 通信プログラムにフレームを渡す

図 3　イーサネットフレームの取り込み方法

3.5 TCP/IP 通信の具体的な例

TCP/IP 通信の具体的な例を SUN ワークステーションの snoop コマンドで見た例を以下に紹介します。この例ではワークステーション netman（Mac アドレス：X'00000E221862'）から opensunx（Mac アドレス：X'0800201CB51F'）へ英小文字の'l' と言う文字を 1 文字送信した時のイーサネットフレームの 16 進ダンプとそれをプロトコル翻訳表示した場合を示します。IP 階層以上の説明はこの後で出てきますが、翻訳表示の場合、TCP/IP プロトコルがイーサネット、IP、TCP の各階層で構成されているのが良く分かります。

1. 通信内容を 16 進ダンプ表示した場合

```
1   0.00000        netman -> opensunx      TELNET C port=1161 l
         0: 0800 201c b51f 0000 0e22 1862 0800 4500    .. ......".b..E.
        16: 0029 9f0d 0000 1e06 0354 851e 7816 851e    .).......T..x...
        32: 781b 0489 0017 0918 e68c d450 433e 5018    x..........PC>P.
        48: 1000 2d89 0000 6ca1 3502 0134              ..-...l.5..4
```

注：上記のダンプにはフレームの先頭にあるプリアンブル部分（8 バイト）が含まれていません。

2. 通信内容を翻訳表示した場合

```
ETHER:  ----- Ether Header -----
ETHER:  Packet 1 arrived at 18:25:4.73
ETHER:  Packet size = 60 bytes
ETHER:  Destination = 8:0:20:1c:b5:1f, Sun
ETHER:  Source      = 0:0:e:22:18:62,
ETHER:  Ethertype = 0800 (IP)
IP:   ----- IP Header -----
IP:   Version = 4
IP:   Header length = 20 bytes
IP:   Type of service = 0x00
IP:   Total length = 41 bytes
IP:   Identification = 40717
IP:   Flags = 0x0
IP:   Fragment offset = 0 bytes
IP:   Time to live = 30 seconds/hops
IP:   Protocol = 6 (TCP)
IP:   Header checksum = 0354
IP:   Source address = 133.30.120.22, netman
IP:   Destination address = 133.30.120.27, opensunx
TCP:  ----- TCP Header -----
TCP:  Source port = 1161
```

```
TCP:  Destination port = 23 (TELNET)
TCP:  Sequence number = 152626828
TCP:  Acknowledgement number = 3562029886
TCP:  Data offset = 20 bytes
TCP:  Flags = 0x18
TCP:  Window = 4096
TCP:  Checksum = 0x2d89
TCP:  Urgent pointer = 0
TELNET:  ----- TELNET -----
TELNET:  "l"                                    ← 送信した文字 'l'
```

4 IP

突然 IP と言う言葉が出てきましたが、IP（Internet Protocol）とは今までお話しているTCP/IP プロトコルの OSI 参照モデルで言うネットワーク層に当たるプロトコルの名前です。従って IP は TCP/IP プロトコルの一部分と言うことになります。IP は端末から出た IP パケットを中継装置（ルータ）を経由して相手先に届ける役目をします。ここでは IP の機能を紹介するとともに、IP アドレスやサブネットアドレスなどと言う、お手持ちのパソコンなどを KHAN に接続する時に必要となる言葉を紹介します。

4.1 IP の特長

前前節の「TCP/IP とは」の所で説明した通り、TCP/IP は元々軍のネットワークを作るためのプロトコルとして作られました。そのため、色々な面白い機能があり、かつ柔軟性に優れています。

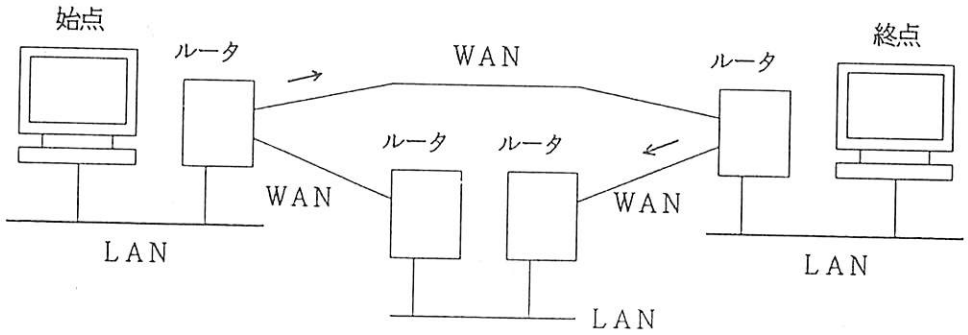
たとえば IP が担当するネットワーク層の機能においてもルーティングと言う機能があります。これは IP パケットが中継装置により中継されながら始点の端末から終点の端末まで送られる時、各中継装置で次の IP パケットの送り先を勝手に決めながら順次転送して行く機能です。たとえば、アメリカの西海岸のロサンゼルスから中部のヒューストンを経由してワシントン DC まで行く軍用の回線が 1 回線しか無かったとしたらどうでしょう。ヒューストンにソ連の（いやロシアの）核ミサイルが落ちればロスとワシントン間の通信は途絶えます。しかし、ロスとワシントンの間にシカゴ経由の回線がもう 1 回線あれば、ロスの中継装置はヒューストンの中継装置に何か障害が起きたのを認識し、自動的にシカゴ経由に切り換えます。このように、障害に対して柔軟な対応ができると言うのも TCP/IP の特長です。

以下に IP の特長について列記します。

IP の特長

1. OSI 参照モデルのネットワーク層を担当するプロトコル。
2. 中継装置（ルータ）を経由して最終目的地まで IP パケットを届ける。その時、中継装置の判断で次の IP パケットの送り先が決められる。

3. データの誤りのチェックや抜け、順番間違いのチェック、送信スピードの制御などはしない。とにかく、最終目的地までパケットを速く送り届ける。データの内容の保証はしない（データの内容の保証などはもっと上位の処理、具体的には TCP で行う）。
4. IP パケットは LAN（イーサネットなど）だけでなく、電話回線や専用回線、ISDN、無線、通信衛星などの多種多様の通信メディアを通過する。
5. 始点から最終目的地までの経路は一つとは限らない。また、行きと帰りが同じ経路を通るとも限らない、いつも同じ経路を通るとも限らない（これは中継装置の判断で経路が動的に決められるため、下図参照）。

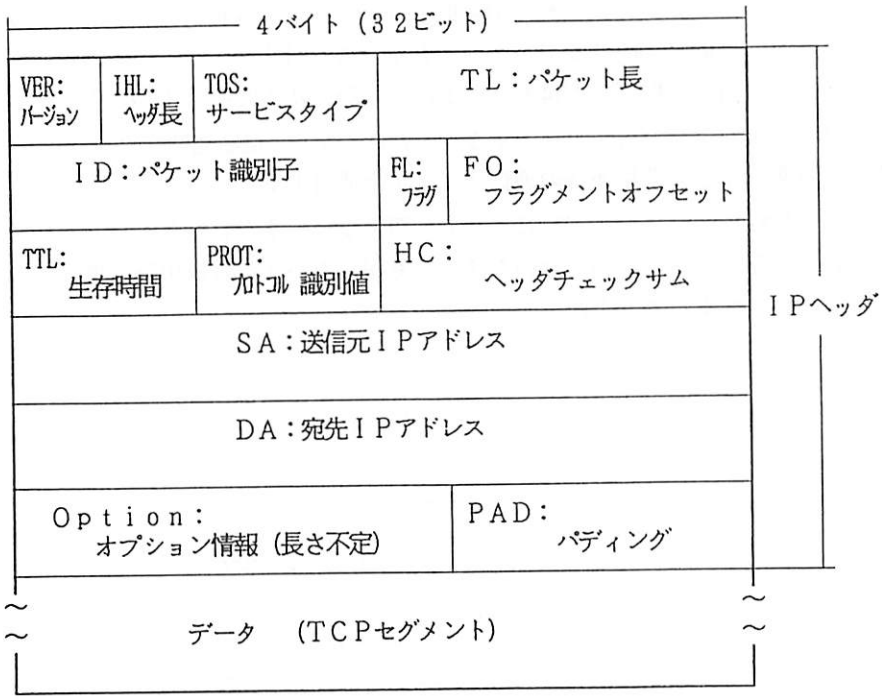


4.2 IP パケットの構造

IP パケットの構造を次ページ図 4 に示します。IP パケットの頭の部分（IP ヘッダ）には主要な情報として送信元 IP アドレスや宛先 IP アドレス、上位プロトコルの識別番号などが含まれており、これはイーサネットフレームの中に宛先 Mac アドレス、送信元 Mac アドレス、プロトコルタイプが含まれているのと同じ構造をしています。

4.3 IP アドレスとは

前節では Mac アドレスという、各端末固有につけるアドレスの話が出てきましたがここでは IP アドレスと言うアドレスの紹介をします。IP アドレスは Mac アドレスと同じように各端末ごとに付けられる固有の 4 バイトの長さをもったアドレスであり、これも世界中で同じ値は存在しません。それでは、なぜ同じ端末に Mac アドレスと IP アドレスと言う二つのアドレスが必要なのでしょう。それは、IP パケットは始点の端末から終点の端末まで中継装置を経由しながら送られると言う話をしました。また、IP パケットはイーサネットだけでなく、電話や専用線、無線などのいろいろな通信メディアを経由すると言う話もしました。この通信メディアの中には専用線のように Mac アドレスを持たないものや電話のようにアドレスの体系が全く Mac アドレスと異なるものもあります。そこで、IP アドレスは始点と終点の端末のアドレスを示し、Mac アドレスはイーサネット上だけで使われイーサネット上の相手端末（または相手となる中継装置）のアドレスを示すと考えれば良く分かります。



1. **パケット識別子**：個々の IP パケットにつけられる識別子。
2. **生存時間**：このパケットがネットワーク上存在してもよい時間（秒）。通常ルータを 1 台通過するたびに 1 づつ引かれ、ゼロになったらそのパケットを捨てる。パケットがループし最終目的地に届かなくなった場合などに捨てられるようする。この生存時間の機能を旨く使い IP パケットのルート情報を取るコマンドに traceroute コマンドがある。
3. **プロトコル識別子**：上位層のプロトコルタイプ番号。

- 1（10 進）：ICMP
- 6（〃）：TCP
- 17（〃）：UDP

図 4　IP パケットの構造

もっと例えて言うなら、人の名前や生年月日、本籍地は結婚などでかわるのを除けば一生かわりませんが、その人の住所や勤め先、肩書、所属、愛称、あだ名などは時と場合により、どんどん変わって行きます。即ち、人の名前などの一生かわらないものを IP アドレス、住所などの時と場合によりかわったり、無くなったりするのが Mac アドレスと考えていただければよく分かるでしょう。

それでは IP アドレスの特長を以下に示します。

1. 個々の端末（ワークステーション、パソコン等）に個々に割り振るアドレス。
2. 世界中で 1 つしか存在しない。世界的には NIC（Network Information Center）で一元管理。日本国内では JPNIC（JaPan NIC）で一元管理されている。（注：IP アドレスを表す時はアドレスの各バイトを 10 進で小数点で区切って表します）
   IP アドレスの例
   - 神戸大学：133.30.xxx.xxx
   - 大阪大学：133. 1.xxx.xxx
   - 富士通　：164.71.xxx.xxx
3. 端末が接続されるネットワークを示す「ネットワーク部」とそのネットワーク内で端末ごとに付ける「ホスト部」とで構成される。
4. アドレスを割り振る組織の大きさによって、クラス（A～C）、および特殊クラス（D～E）がある（図 5）。（神戸大学には 133.30 から始まるクラス B のアドレスが割当てられている）

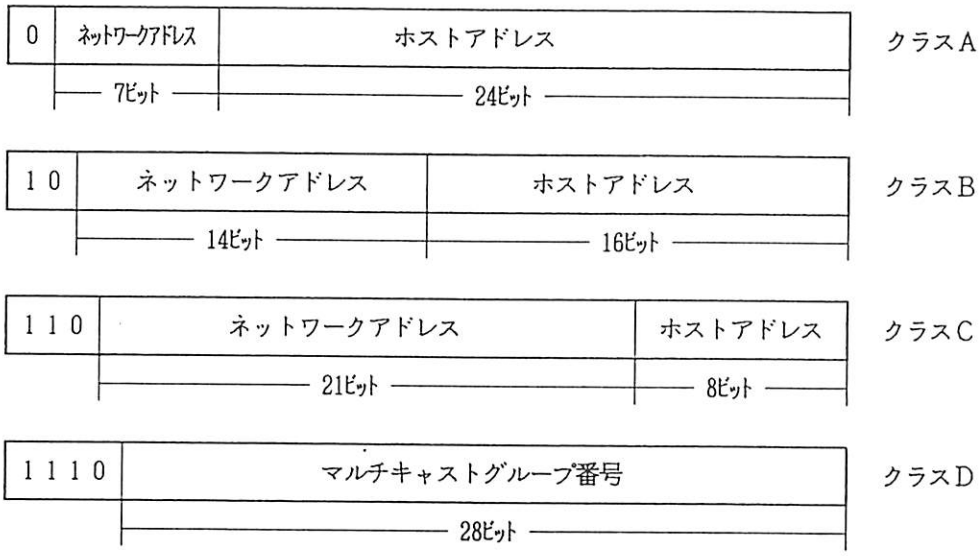


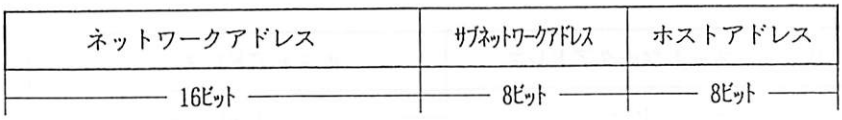
図 5　IP アドレスのクラス

ところで一つ重要な事を言い忘れました。それは IP アドレスは非常に貴重でかつ限りある資源であると言うことです。上記で IP アドレスは世界中で決して重複する事なく NIC によって割り当てられるといいましたが、IP アドレスは 4 バイトの値ですので世界中で 2 の 32 乗（約 42 億 9 千万）個しか取れません、そのうち神戸大学には 2 の 16 乗（6 万 5 千）個が割当られていますが、地球上の人ひとりに 1 個も割当られないのです。また、最近のネットワークの普及やネットワークに接続されるパソコン、ワークステーションの急激な普及により、いくら IP アドレスを節約したり、割当方法を工夫したりしても、近い将来、だいたい 2013 年ごろには完全に IP アドレスが足らなくなることが予想されています。みなさんもその事を良く理解し、必要最小限の IP アドレスを取得するように心掛けましょう。

4.4 サブネットワーク

前項の説明で IP アドレスは端末が接続されるネットワークを示す「ネットワーク部」とそのネットワーク内で端末ごとに付ける「ホスト部」で構成されると言う話をしました。そして神戸大学にはネットワーク部が 133.30. であるクラス B の IP アドレスが 1 個割当てられていると言う話もしました。ところが神戸大学内にはネットワーク部を割り当てるべきネットワークが約 60 本存在します。そこで神戸大学では 16 ビットあるホスト部の上半分（8 ビット）をサブネットワーク部とし、大学内部では 24 ビットのネットワーク部と 8 ビットのホスト部の組み合わせで運用しています。また、IP アドレスのうち、「ネットワーク部がどこまでを占めるのか」と言うことを示す値をサブネットマスクと言います。

神戸大学でのサブネットの割当



1. ネットワークアドレス
   神戸大学が NIC より指定されたアドレス。 → 133.30
2. サブネットワークアドレス
   各建屋ごとに設置されているイーサネットごとに指定。
   → 133.30.10　総合情報処理センター
   133.30.12　自然系図書館
   133.30.56　理学部 A 棟（1〜2 階）など
3. サブネットマスク
   どこまでがネットワーク部かを示す値、神戸大学では 255.255.255.0 を指定する。

4.5 特殊な IP アドレス

前節の Mac アドレスの説明の所で Mac アドレスのうち、すべてのビットが 1 の物は特殊なアドレスで全ての端末に対するイーサネットフレームの送信、すなわちブロードキャストを示すとありましたが、IP アドレスについても全く同じ考え方があります。IP アドレスのう

ち、ホスト部がオール 1 の物がブロードキャストを示し、すべての端末に対し IP パケットを送信したい時に使用されます。

以下にその説明と実際イーサネット上を流れているパケットのダンプを示します。

1. ホストアドレスがオール 0 (0)
   古い形式のブロードキャストアドレス（現在は使用しない)。予約アドレスであり使用してはならない。
2. ホストアドレスがオール 1 (255)
   ブロートキャストアドレスを示す。たとえば総合情報処理センター内イーサネット上の場合は → 133.30.10.255 となる。
3. IP アドレスすべてがオール 1 (255.255.255.255)
   これもブロードキャストアドレスを示す。

(1) ブロードキャストのダンプ表示

```
1   0.00000 133.30.120.6 -> 133.30.120.255 RIP R (24 destinations)
         0: ffff ffff ffff 0000 9310 54c5 0800 4500    ..........T...E.
        16: 0200 7010 0000 3c11 119b 851e 7806 851e    ..p...<.....x...
        32: 78ff 0208 0208 01ec 7e4b 0201 0000 0002    x.......  K......
        48: 0000 851e 7800 0000 0000 0000 0000 0000    ....x...........
        64: 0001 0002 0000 851e 5a00 0000 0000 0000    ........Z.......
        80: 0000 0000 0001 0002 0000 851e 2400 0000    ............$...
```

(2) ブロードキャストの翻訳表示

```
ETHER:  ----- Ether Header -----
ETHER:  Packet 1 arrived at 20:30:55.73
ETHER:  Packet size = 526 bytes
ETHER:  Destination = ff:ff:ff:ff:ff:ff, (broadcast)←  宛先 Mac アドレスも
ETHER:  Source      = 0:0:93:10:54:c5, Proteon              ブロードキャスト
ETHER:  Ethertype = 0800 (IP)
IP:     ----- IP Header -----
IP:     Version = 4
IP:     Header length = 20 bytes
IP:     Type of service = 0x00
IP:     Total length = 512 bytes
IP:     Identification = 28688
IP:     Flags = 0x0
IP:     Fragment offset = 0 bytes
IP:     Time to live = 60 seconds/hops
IP:     Protocol = 17 (UDP)
```

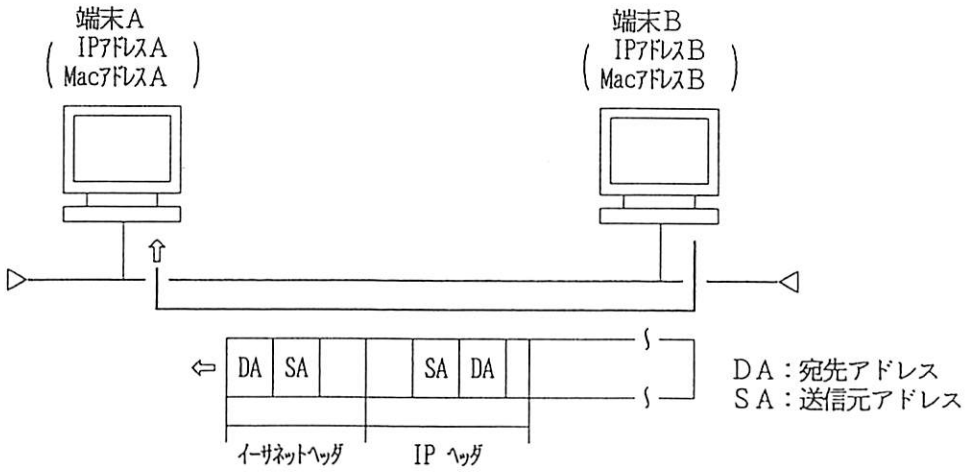
```
IP:   Header checksum = 119b
IP:   Source address = 133.30.120.6,
IP:   Destination address = 133.30.120.255,          ← 宛先 IP アドレスは
UDP:  ----- UDP Header -----                            ブロードキャスト
UDP:  Source port = 520
UDP:  Destination port = 520 (RIP)
UDP:  Length = 492
UDP:  Checksum = 7E4B
RIP:  ----- Routing Information Protocol -----
RIP:  Opcode = 2 (route response)
RIP:  Version = 1
RIP:  Address                          Port   Metric
RIP:  133.30.120.0    133.30.120.0      0      1
RIP:  133.30.90.0     133.30.90.0       0      1
RIP:  133.30.36.0     133.30.36.0       0      16 (not reachable)
```

4.6 イーサネット上での IP 通信

それでは、イーサネット上で IP パケットをやり取りする手順はどのようになるのでしょうか。図 6 を参照ください。イーサネット上に端末 A と端末 B が接続されているとします。それぞれの端末の IP、Mac アドレスは端末 A は IP アドレス A、Mac アドレス A 、端末 B は IP アドレス B、Mac アドレス B とします。端末 B から端末 A 宛に送られた IP パケットはどのようにして端末 A に届くのでしょうか。

まず、端末 B からイーサネットフレームが送信されます。その時、フレーム上の宛先 IP アドレス、Mac アドレスにはそれぞれ IP アドレス A、Mac アドレス A が、また、送信元 IP アドレス、Mac アドレスにはそれぞれ IP アドレス B、Mac アドレス B が設定されます。また、イーサネットヘッダ上のプロトコルタイプには当然 TCP/IP であることを示す値 X'0800' が設定されます。

次に端末 A はイーサネット上を流れて来た端末 B からのフレームを受信します。その時、端末 A は宛先 Mac アドレスが自分宛なのでフレームを取り込み、プロトコルタイプを見て TCP/IP 通信プログラムに IP パケットとして渡します。そして端末 A の TCP/IP 通信プログラムは IP パケットの宛先 IP アドレスが自分宛なので、その IP パケットをさらに上位の通信プログラム（たとえば TCP 処理プログラム）に渡します。ところで、宛先 Mac アドレスが自分宛で、宛先 IP アドレスが自分宛で無い時はどのようになるでしょう。このような事は通常は起こらないと思われますが、実はごく普通に起きることです。それは受信側が IP パケットの中継装置（ルータ）である時です。この時は宛先 IP アドレスが最終目的地の端末の IP アドレス、宛先 Mac アドレスが中継装置の Mac アドレスとなります。その時、中継装置に取り込まれた IP パケットは最終目的地の IP アドレスを持つ端末または最終目的地に近い中継装置に向かって再送信（これをパケットのフォワーディングと言います）されます。



1. 端末 B は端末 A 宛のフレームを送信する。その時、宛先 Mac アドレス、宛先 IP アドレスには端末 A を指定する。
2. 端末 A は Mac アドレスが自分宛なのでフレームを取り込み、TCP/IP 通信プログラムへ渡す。
3. 端末 A の TCP/IP 通信プログラムは IP アドレスが自分宛の場合はその IP パケットを取りこむ。また、他人宛の場合はその IP アドレスを持つ端末へ再送信する（パケットのフォワーディング）。

図 6　イーサネット上での IP 通信

4.7　ARP プロトコル

前項でイーサネット上の端末と通信を行う場合、宛先 IP アドレスと宛先 Mac アドレスの両方に相手のアドレスを入れないといけないと言いました。しかし、通常、相手の IP アドレスは分かっても Mac アドレスは分かりません。なぜなら、IP アドレスは各端末ごとに割当られたアドレスを端末の利用者や端末の管理者が設定しますが、Mac アドレスは端末製造メーカが出荷時に付けてくるものだからです。また、いちいち相手のアドレスを 2 つも指定しないといけないと言うのも面倒です。TCP/IP には相手の IP アドレスから Mac アドレスを知るためのプロトコル、ARP プロトコル（Address Resolution Protocol）が用意されています。

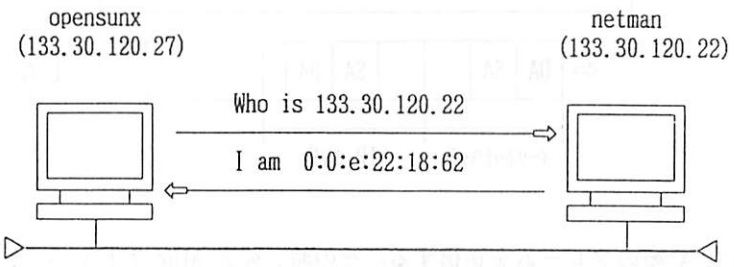
これは、通信相手の Mac アドレスを知りたい端末がイーサネットに接続されているすべての端末に向かって、「この IP アドレスを持つ端末はだーれ？」とブロードキャストを使って問い合わせると、対応する IP アドレスを持つ端末が「ぼくだよ。ぼくの Mac アドレスは XX」と答えてくれると言うものです。この時、「だーれ」と問い合わせるのを ARP リクエスト、「ぼく」と答えるのを ARP リプライと言います。

これらの ARP リクエストと ARP リプライで得られた IP アドレスと Mac アドレスの対は各端末上の ARP テーブルと言う所に保存され、2 度目の送信からは ARP リクエストを出

さなくなりますが、暫く使わないとその情報はテーブルから消去されるようになっています。消去されるまでの時間は端末によって異なりますが、たとえばSunワークステーションの場合は30秒となっています。

以下にopensunxと言う端末（IPアドレス、133.30.120.27）がnetmanと言う端末（IPアドレス、133.30.120.22）を捜している例とARPテーブルの例を示します。

（1）ARPの通信例



- 16進ダンプ表示

```
1  10.43104     opensunx -> (broadcast)  ARP C Who is 133.30.120.22, netman ?
         0: ffff ffff ffff 0800 201c b51f 0806 0001    ........ .......
        16: 0800 0604 0001 0800 201c b51f 851e 781b    ........ .....x.
        32: ffff ffff ffff 851e 7816                   ........x.

2   0.00348    netman -> opensunx  ARP R 133.30.120.22, netman is 0:0:e:22:18:62
         0: 0800 201c b51f 0000 0e22 1862 0806 0001    .. ......".b....
        16: 0800 0604 0002 0000 0e22 1862 851e 7816    .........".b..x.
        32: 0800 201c b51f 851e 781b 0d00 3520 7801    .. .....x...5 x.
        48: 0805 7801 8078 0182 5a00 030a 43dd c260    ..x..x..Z...C..`
```

- プロトコルの翻訳表示

```
ETHER:  ----- Ether Header -----
ETHER:  Packet 1 arrived at 19:37:10.47
ETHER:  Packet size = 42 bytes
ETHER:  Destination = ff:ff:ff:ff:ff:ff, (broadcast)
ETHER:  Source      = 8:0:20:1c:b5:1f, Sun
ETHER:  Ethertype = 0806 (ARP)
ARP:  ----- ARP/RARP Frame -----
ARP:  Opcode 1 (ARP Request)
ARP:  Sender's hardware address = 8:0:20:1c:b5:1f
ARP:  Sender's protocol address = 133.30.120.27, opensunx
ARP:  Target hardware address = ?              ← netman の Mac アドレス？
```

```
ARP:  Target protocol address = 133.30.120.22, netman

ETHER:  ----- Ether Header -----
ETHER:  Packet 2 arrived at 19:37:10.47
ETHER:  Packet size = 64 bytes
ETHER:  Destination = 8:0:20:1c:b5:1f, Sun
ETHER:  Source      = 0:0:e:22:18:62,
ETHER:  Ethertype = 0806 (ARP)
ARP:  ----- ARP/RARP Frame -----
ARP:  Opcode 2 (ARP Reply)
ARP:  Sender's hardware address = 0:0:e:22:18:62  ← netman の Mac アドレス
ARP:  Sender's protocol address = 133.30.120.22, netman
ARP:  Target hardware address = 8:0:20:1c:b5:1f
ARP:  Target protocol address = 133.30.120.27, opensunx
```

（2）ARP テーブルの例

```
host% arp -a
133.30.124.3 at 00:00:0c:08:e1:76
133.30.124.4 at 00:00:0e:34:06:9c
133.30.124.5 at 00:00:0e:34:0a:80
133.30.124.10 at 00:00:0e:34:0a:44
133.30.124.13 at 00:00:0e:34:0c:48
133.30.124.15 at 00:00:0e:34:0b:a0
+-----------+    +---------------+
 IP アドレス        Mac アドレス
```

4.8 IP ルーティング機能

それでは、IP のもっとも重要で花形の機能、ルーティングについてご紹介します。この機能は IP の特長の説明の所であったように、IP パケットを中継装置（ルータ）を経由しながら始点から終点まで確実に送り届ける機能です。それでは、具体的な例を参考にしながらルーティング機能を説明しましょう。

図 7 を参照ください。これは、ある旅行者がニューヨークの国連本部から神戸大学まで旅行する時のルートを示したものです。これ以外にもたくさんのルートが考えられますが省略します。ニューヨークの国連本部を出発した旅行者はまず JF ケネディ国際空港に行きます。そして、空港で日本の空港の様子たとえば「成田空港が赤軍派によって乗っ取られていないか」「関西空港が海に沈没してしまっていないか」（すべて冗談です）を尋ね、次の行き先を決定します。次に関西空港に着いた旅行者は空港でまた、大阪駅に JR で行く方が良いのか、船で K-CAT に行けば良いのかを尋ね、次の目的地に向かいます。このようにして、ニューヨークの国連本部を出発した旅行者は神戸大学に着くことができるでしょう。

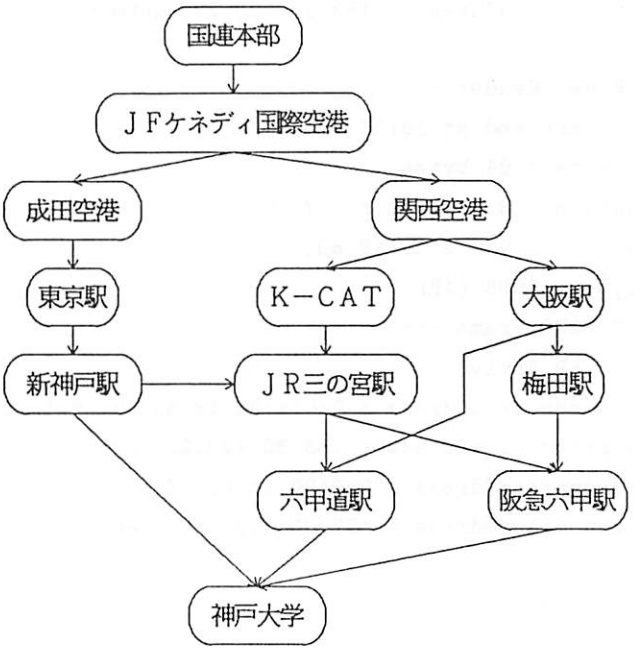


図 7　ニューヨークの国連本部から神戸大学までのルート

この例の中で旅行者を「IP パケット」、空港などの行き先を尋ねる所を「中継装置（ルータ）」と言うふうに置き換えると IP パケットのルーティング機能の説明となります。それでは、実際に IP ルーティングが行われている例をお見せしましょう。1 つ目は神戸大学から日本の村山首相いる首相官邸まで、もう 1 つはアメリカのホワイトハウスまでのルーティングの例です。

日本の首相官邸までのルートでは神戸大学から、大阪、東京、大手町というふうにルーティングされ最後に首相官邸に届いているのがよく分かります。また、ホワイトハウスまでのルートでは中継装置（ルータ）を 7 台ぐらい経由したあたりで太平洋を越え、どこかで西海岸から東海岸まで行って、最後にワシントン DC のホワイトハウスに着いています。

このように、日本の首相官邸も、アメリカのホワイトハウスも距離に関係なくアクセスできる所がネットワーク（特にインターネット）のすばらしい所です。また、ホワイトハウスに行く方が首相官邸に行くより中継装置の段数が少なく、かつ速い所がアメリカと日本のネットワーク事情を示しているようで面白い所です。

（1）日本国首相官邸までのルート

```
opensun% traceroute kantei.go.jp
traceroute to kantei.go.jp (202.32.34.2), 30 hops max, 40 byte packets
 1  133.30.8.254 (133.30.8.254)  12 ms  3 ms  2 ms
 2  kobesig.kobe-u.ac.jp (133.30.120.10)  4 ms  3 ms  3 ms
 3  kobe.bb.sinet.ad.jp (150.99.34.1)  5 ms  6 ms  5 ms
 4  150.99.69.1 (150.99.69.1)  23 ms  34 ms  21 ms
 5  new-osaka-S1/1.bb.sinet.ad.jp (150.99.164.1)  23 ms  23 ms  23 ms
```

```
 6  new-nacsis.bb.sinet.ad.jp (150.99.100.1)  70 ms  69 ms  68 ms
 7  new-nacsis-fddi3/0.bb.sinet.ad.jp (150.99.90.1)  69 ms  70 ms  69 ms
 8  otsuka.sinet.ad.jp (150.100.127.1)  83 ms  82 ms  82 ms
 9  newgate.sinet.ad.jp (150.100.1.250)  86 ms  84 ms  82 ms
10  wnoc-tyo.wide.ad.jp (133.4.3.2)  767 ms  718 ms  517 ms
11  wnoc-tokyo-cisco2.wide.ad.jp (133.4.3.16)  719 ms  486 ms  199 ms
12  202.249.3.34 (202.249.3.34)  321 ms  282 ms  233 ms
13  otemachi.iij.net (192.244.176.205)  347 ms  305 ms  102 ms
14  kanteigw.iij.net (192.244.180.86)  161 ms  426 ms  196 ms
15  sh.kantei.go.jp (202.32.34.2)  870 ms  121 ms  129 ms
```

（2）アメリカのホワイトハウスまでのルート

```
opensun% traceroute whitehouse.gov
traceroute to whitehouse.gov (198.137.240.100), 30 hops max, 40 byte packets
 1  133.30.8.254 (133.30.8.254)  12 ms  3 ms  3 ms
 2  kobesig.kobe-u.ac.jp (133.30.120.10)  4 ms  3 ms  3 ms
 3  kobe.bb.sinet.ad.jp (150.99.34.1)  6 ms  5 ms  5 ms
 4  150.99.69.1 (150.99.69.1)  23 ms  23 ms  21 ms
 5  new-osaka-S1/1.bb.sinet.ad.jp (150.99.164.1)  24 ms  25 ms  23 ms
 6  new-nacsis.bb.sinet.ad.jp (150.99.100.1)  70 ms  68 ms  69 ms
 7  nacsis-gate.sinet.ad.jp (150.99.99.12)  71 ms  74 ms  73 ms
 8  sl-stk-5-S3/4-1984k.sprintlink.net (144.228.45.13)  315 ms  211 ms  253 ms
 9  sl-dc-6-H1/0-T3.sprintlink.net (144.228.10.1)  533 ms  287 ms  275 ms
10  Boone1.VA.ALTER.NET (192.157.65.227)  279 ms  275 ms  280 ms
11  Falls-Church4.VA.ALTER.NET (137.39.43.97)  286 ms  299 ms  279 ms
12  Falls-Church1.VA.ALTER.NET (137.39.8.2)  287 ms  277 ms  303 ms
13  TISMd-gw.ALTER.NET (137.39.238.194)  292 ms  292 ms  287 ms
14  WhiteHouse.Gov (198.137.240.100)  318 ms  290 ms  330 ms
```

4.9 ルーティングテーブル

ルーティングテーブルとはIPパケットの中継装置（ルータ）やTCP/IPの通信ができる端末、コンピュータが持つ「次にIPパケットを送るべき相手」を指定したテーブルです。中継装置や端末はIPパケットを送る時は必ずこのテーブルを見て、IPパケットを送るべき相手を決定します。もし、IPパケットを送るべき最終目的地が同じイーサネット（正確には同じネットワーク）に接続されている場合は直接相手にIPパケットを送りますが、相手が異なるネットワークに接続されている時はルーティングテーブルを参照し相手に最も近いと思われる中継装置（ルータ）にIPパケットを送ります。

それではルーティングテーブルはどのようにして作られるのでしょう。ルーティングテーブルの作成方法にはスタティクルーティングとダイナミックルーティングと言う方法があります。

1. スタティクルーティング
   この方法では予め固定的にルートを設定します。非常に簡単な方法ですが、相手が多くなると設定が非常に困難になったり間違ったりする、また、ルートが固定的に設定されるので指定されたルートで障害が起きた場合は代替ルートに切り替わらず通信が途絶えてしまうと言うふうな欠点があります。
2. ダイナミックルーティング
   この方法は他の中継装置（ルータ）から送られてくる情報（ルーティング情報）を使って定期的にルーティングテーブルを更新する方法です。この方法では予めルート情報を設定しておく必要は無いし、もし、通信ルートに障害が起きた場合でも、その通信ルートを使って送られてくるルーティング情報そのものが途絶えるため、別のルートに自動的に切り換えられるという長所があります。従って、幹線系ネットワークにおいて特別な理由が無い限り、ルーティングはダイナミックルーティングを使用するのが一般的となっています。

最後に、スタティックルーティングの機能の一部に「デフォルトルーティング」と言う機能があります。この機能は端末や中継装置の繋がるネットワークから外部に出るルートが 1 箇所しか無い場合や、ルートをルーティングテーブルにすべて登録すると大きくなり過ぎるため、一部を「ルートの省略値」として指定する時に使用します。デフォルトルートを指定した場合、ルーティングテーブルに無い相手先はすべてこのデフォルトルートが示す相手先に送られます。

現在、ルーティング情報は巨大化の一途をたどっており、94 年 11 月現在で、日本国内だけで約 1600（半年前の 4 月ごろには約 800 で、いかに最近の伸びが激しいか分かります）、世界では 2 万から 3 万と言う個数になっています。また、神戸大学の内部では FDDI や ATM と言う幹線系の LAN ではダイナミックルーティング、各建屋に設置されている支線イーサネットでは、幹線 LAN への出口が 1 箇所しか無くかつ、むだなルーティング情報を支線まで流す必要がないことから、デフォルトルートとして幹線 LAN の中継装置（ルータ）を指定するスタティックルーティングを使用するようにしています。

以下にルーティングテーブルの例（抜粋）を載せます。実際は約 1600 個ぐらいのルーティングテーブルとなっています。

```
host% netstat -r
Destination       Gateway           Flags   Refcnt   Use      Interface
0.0.0.0           133.30.120.11     UG      0        0        ei11
133.30.16.0       133.30.120.254    UG      0        0        ei11
133.30.64.0       133.30.120.1      UG      0        47317    ei11
133.30.80.0       133.30.124.17     UG      0        3        fddi21
133.30.128.0      133.30.124.11     UG      0        0        fddi21
192.218.128.0     133.30.120.11     UG      0        0        ei11
202.38.145.0      133.30.120.10     UG      0        0        ei11
+-----------+     +-------------+                              +-------+
```

目的ネットワークアドレス　中継ルータアドレス　　中継ルータが接続されるボード名

（注）目的ネットワークアドレスが 0.0.0.0 はデフォルトルートを示す。

4.10 DNS サーバまたはネームサーバ

今まで通信する相手の指定は IP アドレスと言う数字で指定すると言ってきました。ところが、ただの数字の列では非常に覚えにくいうえに、ひとつ値が違うととんでも無い相手を指定してしまうと言う欠点があります。そこで、通信する相手はわかりやすい名前（ホスト名）で指定し、それを IP アドレスの電話帳のような物で IP アドレスに変換してやれないかとの考えが出てきました。これを実現しているのが DNS（Domain Name System）サーバまたはネームサーバと呼ばれるものです。DNS サーバはネットワーク上に存在し、各端末からくるホスト名から IP アドレスへの変換要求やその逆の要求に対して、電話帳のように辞書を引き返答します。たとえば、IP ルーティングの実例の所で指定した、**kantei.go.jp** や **whitehouse.gov** のようなホスト名は DNS サーバにより、202.32.34.2 や 198.137.240.100 と言う IP アドレスに変換されて問い合わせ先に返答されます。

ホスト名はローカルホスト名と世界中でひとつしか存在しないドメイン名をピリオド（.）で繋いだものであり、神戸大学のドメイン名は **kobe-u.ac.jp** です。従って、host1 と言う名前のローカルホスト名を持つワークステーションのホスト名は **host1.kobe-u.ac.jp** となります。このように、ホスト名はローカルホスト名に世界中でひとつしか存在しないドメイン名を付けることにより、世界中でひとつしか存在しない名前となります（ドメイン名も IP アドレスと同じように NIC（Network Information Center）より割当てられます）。

5 TCP

次に TCP/IP の TCP についてお話しましょう。TCP とは Transmission Control Protocol の略で、OSI の参照モデルで言うとちょうど 4 層のトランスポート層に当たり、正確なデータの転送と通信路の多重化という役目を持っています。また、ここでは説明しませんが、TCP/IP プロトコルには TCP の代わりに、TCP より機能を簡単にした UDP（User Datagram Protocol）と言うプロトコルもありその機能が簡単と言う特性を生かし、一部の目的で使われています。

5.1 TCP の特長

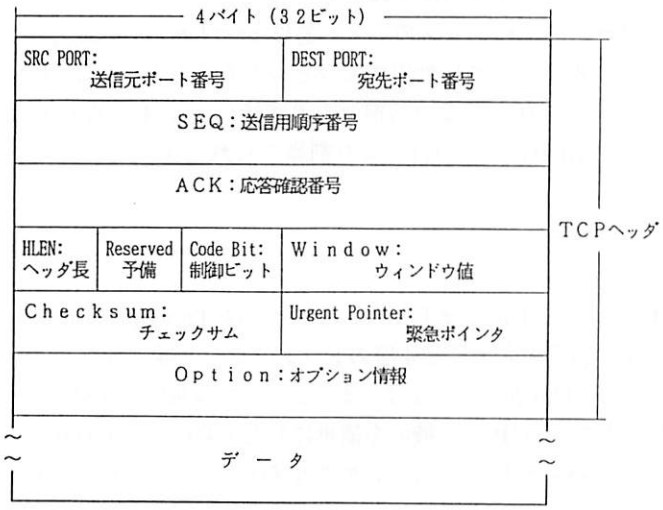
前節の IP の特長の説明で少し不思議に思われたかと思いますが、IP では転送するデータの中身については保証しないと言う話がありました。それでは、データの保証は誰がやるのでしょうか。実はこの TCP がしっかり行います。また、最近のパソコンでは MS-Windows のようなウィンドウシステムが使え、パソコンを端末として使う時でも、同じ相手ホストに対して同時に複数の接続、たとえば、telnet をしながら FTP をし、かつ、Mosaic を使用すると言うことが可能になりました。その場合、1 本の、あるホストと、あるパソコン間の TCP/IP 接続を多重化して使用する必要性が出ます。このように、TCP はデータの保証と多重化の役目を持っています。

TCP の特長

1. OSI 参照モデルのトランスポート層を担当するプロトコル。
2. データの誤り、データの抜け・重複、順序間違いなどのチェック、および応答確認、再試行を行い、通信の信頼性を保つ（そのため、UDP よりプロトコルの処理が重い）。
3. 送信のスピード制御（フローコントロール：流量制御）を行う。
4. 端末間で通信する前に通信のための仮想的な通信路を作る（コネクション型）。
5. 始点と終点間で多重化を行い、同時に複数の通信が行えるようにする。

5.2 TCP セグメントの構造

TCP セグメントの構造を図 8 に示します。



1. SEQ：送信用順序番号
   送信元が送ったデータの位置を示す番号。通信の最初に初期化され、送信したバイト数づつ加算される。
2. ACK：応答確認番号
   次のデータを受信する時に、相手から送られてくるべきデータの順序番号。送信側がこの値を受取り次に送ろうとしている送信用順序番号より小さい時は前のデータが正常に届かなかったことを示している（再送要求）。
3. Window：ウィンドウ
   受信側が次にどれだけ、データを送ってもらってもよいか送信側に伝える値。
4. Checksum：チェックサム
   受信したセグメントが壊れていないかチェックするためのチェックデータ。

図 8　TCP セグメントの構造

TCP セグメントの頭の部分（ TCP ヘッダ）には多重化のための送信元・宛先ポート番号、セグメントの順序確認のための送信用順序番号、受信側が送信側にデータが正常に届いたのを知らせる応答確認番号、データの流量制御のためのウィンドウ値、そして、送られて来たデータの信頼性を確認するためのチェックサムからなります。

5.3 伝送データのチェックと再送

それでは TCP ではどのようにして伝送データの保証がなされているのでしょうか。次ページの図 9 を参照ください。端末 A から端末 B へデータを転送するとします。まず 1 の正常な動作の場合を見て見ましょう。端末 A から端末 B に向かって 100 バイトのデータを送りました。その時、TCP ヘッダ内の送信用順序番号には順序番号として 0 が入っているとします。データを受信した端末 B は受信したデータの正当性をチェックサムで確認し正常であるため、ACK（応答確認番号）として次に送って貰いたいデータの番号 100 を返します。その ACK を端末 A が受信することにより、端末 A は端末 B が正常に受信したことを知ります。

次に伝送エラーが発生した時はどうでしょう。2 をご覧ください。端末 A は 100 番からのデータを 200 バイト送信します。そのデータを受信した端末 B はデータの正当性をチェックサムで確認し、伝送エラーがあった事を知ります。そして端末 B は端末 A に再度同じデータを送るよう、 ACK に再送して欲しいデータの先頭番号である 100 を入れて端末 A に返します。それを受けた端末 A は伝送エラーがあったのを知り、端末 B に前と同じデータを再度送信します。

次に端末 A が送信したデータそのものが伝送途中で消失し、端末 B に届かなかった場合はどうでしょう。3 をご覧下さい。端末 A が送信したデータは途中で消失し端末 B へ届きません。そのため、端末 B は当然のことながら、端末 A に ACK を返しません。端末 A はいくら待っても端末 B から ACK が返って来ないのでしびれを切らして、端末 B に再度同じデータを送ります。実際はこのしびれを切らして再度データを送信するまでの時間、いわゆる「タイムアウト」になる時間は端末間のそれまでの正常な場合の応答時間を元に計算された値が使用されます。また、この再送ルールは端末 B に正常にデータが届き ACK を返したのに係わらず、ACK が途中で消失し端末 A に返らなかった場合にも適用されます。(ただし、この場合、端末 A が同じデータを再度、端末 B へ送信するため、端末 B には同じデータが 2 度届くことになります。この時は TCP ヘッダの中の送信用順序番号を見ればデータがダブっているのが分かります。)

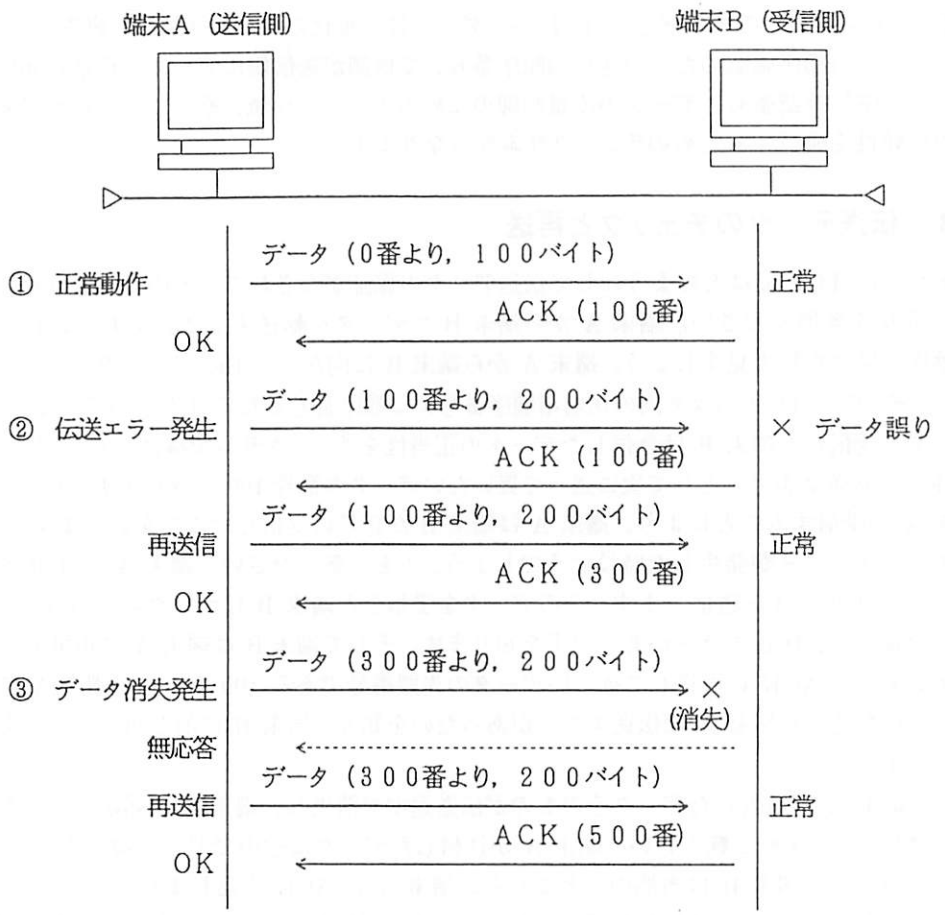


図 9　伝送データのチェックと再送

5.4　伝送データの流量制御

送信側から受信側にデータを転送する時、送信側から送信したデータをすべて受信側が受けられるでしょうか。いえそんな事はありません。たとえば、送信側が非常に速いコンピュータで受信側が非常に遅いコンピュータのような場合、送信したデータがすべて受信できるとは限りません。また、受信側が何かの都合で送信を一時停止して欲しい場合もあります。その時は受信側が送信側に対して、送信するデータの量を減らしたり、送信を一時停止するよう依頼することができます。

この機能はデータの流量制御（フローコントロール）と言い、TCP ヘッダ内の「ウィンドウ」フィールドを使用して行います。図 10 を参照ください。端末 A は端末 B に 200 バイトのデータを送りました。そのデータを正常に受け取った端末 B は端末 A に対して ACK を返しますが、その時、端末 B は次に受信できるデータ量、ここでは 500 バイトを TCP ヘッダ

のウィンドウフィールドへ入れて ACK として、端末 A に返します。この ACK を受け取った端末 A はウィンドウフィールドを見て、あと 500 バイト送れることを知ります。そして、端末 A は端末 B に対して端末 B からの ACK を受ける事なく、500 バイトのデータを送ります（このように相手の受信確認を受ける事なく、どんどんデータを送り付ける事をウィンドウコントロールと言います）。

次に受信側がデータの送信を一時停止して貰いたい場合はどのようにすればよいのでしょうか。それは ACK を返す時にウィンドウの値として 0 を返します。そうすれば、送信側は受信側が受信できないと判断し送信を一時停止します。そして、受信側が送信を再開して欲しい時は、ウィンドウに適当な値を入れて、送信側に送れば、データの送信が再開されることになります。

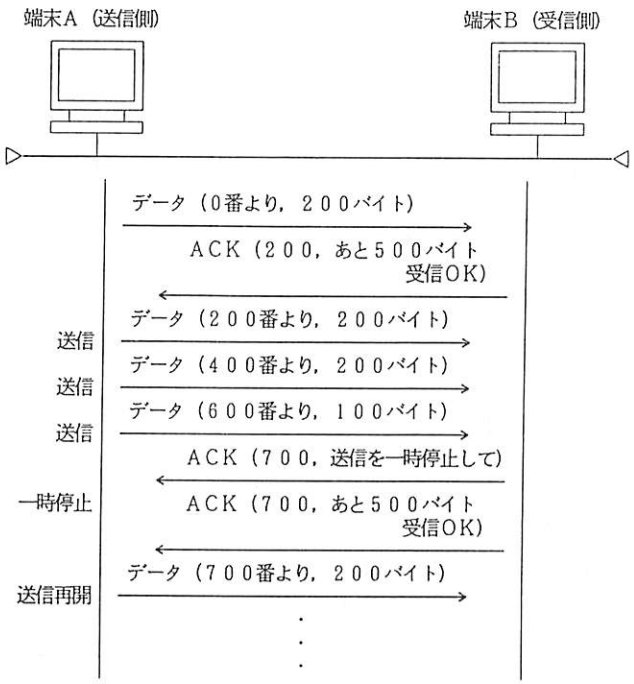


図 10　伝送データの流量制御

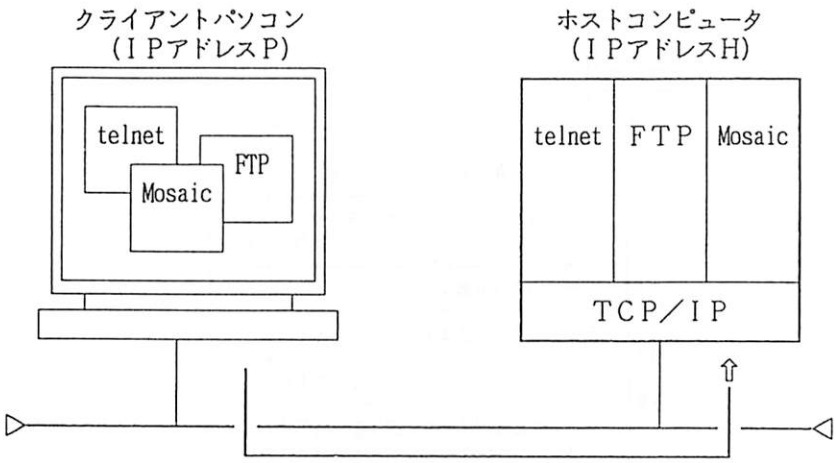
5.5　通信の多重化

TCP の特長の所でも述べましたが最近のパソコンでは MS-Windows のようなウィンドウシステムが使え、パソコンを端末として使う時でも、同じ相手ホストに対して同時に複数の接続、たとえば、telnet をしながら FTP をし、かつ、Mosaic を使用すると言うことが可能になりました。このような場合、ホストコンピュータと端末となっているパソコンは 1 本の TCP/IP によるつながり（ TCP/IP リンク）で接続されることになります（次ページ図 11 参照）。ところが、このままでは、どの端末から IP パケットが来たのかは、送信元の IP アドレスを見れば分かりますが、その端末のどのクライアントアプリケーション、たとえば、telnet、FTP、

Mosaic 、からデータが来たのか分かりません。また、パケットを受け取ったホストコンピュータもどのサーバアプリケーションにデータを渡して良いのか分かりません。ちょうど、郵便が「神戸市灘区六甲台町 1 － 1 神戸大学殿」で届いたようなものです。届いた郵便が誰宛なのか分からないのです。

そこで、TCP ではポート番号と言うものを使い、誰宛のデータか、また、誰から来たのかと言うことを識別しています。ポート番号はサーバ（親）となる時は値が「よく知られたポート番号（Well-known port number)」として決められており、例えば、telnet サーバが 23 番、FTP サーバが 21 番、Mosaic サーバ（WWW サーバ）が 80 番となっています。また、子供側（クライアント）側は 224 番以上の空いている番号を使用する事になっています。

このようにして、同じ端末間でたくさんのアプリケーションを同時に使用した場合でも、お互いのデータが混乱なく、相手のアプリケーションに伝わるようになっています。



1. ホストコンピュータの悩み

   クライアントパソコンから届いた IP パケットの宛先アドレスがみんな「IP アドレス H」なので、届いたデータをどのアプリケーションに渡せば良いか分からない。また、クライアントパソコンのどのアプリケーションから来たのかも分からない。

2. そこで解決策

   ホストコンピュータのアプリケーションに番号を付け、クライアントにその番号（ポート番号）を宛先として指定してもらう。また、送信元番号で送り元アプリケーションを明らかにしてもらう。

3. ポート番号の例

   21（10 進）FTP
   23　　telnet
   25　　SMTP（メールシステム）
   80　　Mosaic
   119　　NNTP（ニュースシステム）

図 11　ポート番号の必要性

6 最後に

ついに「最後に」なりました。プロトコルと TCP/IP の話はこれで終わりにします。プロトコルや TCP/IP について、まだまだ、お話することもあるとは思いますが、これ以上お話しても私の知識の無さがバレルだけですのでおしまいにしたいと思います。もし、これ以上勉強なされたい方は最後に参考文献の一覧を添付しますので、そちらをご参照ください。なお、この資料に書かせて頂いた内容については、私の勉強不足により、ウソや解釈間違い等がたくさんあると思いますが、大筋の所では合っているのではないかと思っています。最後に、この資料が読者の皆様方のプロトコルと TCP/IP の理解に少しでも役に立ち、神戸大学情報ネットワーク KHAN の益々の発展のお役に立てれば幸いです。

参考文献

以下にこの資料を作るのに参考にした資料、もしくはより良い理解に役に立つと思われる資料の一覧を記載します。(ただし、すべて私の知っている範囲でのお話です。)

(1) 通信プロトコルのしくみがわかる本 (株式会社工業調査会)
著者が女性で通信プロトコルや OSI の参照モデル、モデムの規格等、一通のことがやさしく説明されています。生活の中のプロトコルの例もこの本がネタです。初心者にはお勧めの本です。

(2) マスタリング TCP/IP 入門編 (オーム社)
この資料を作るための TCP/IP 関連のネタの多くはこの資料から出ています。TCP/IP 関連の新しい情報も載っておりお勧めの本です。

(3) 新プロトコルハンドブック (朝日新聞社)
プロトコルの辞典です。内容は難しいですが、イザと言う時に持っておくと便利な本です。欠点は値段が高いこと。

(4) インターネットワーク入門 (工学図書)
あまり面白くないが、何かの役に立ちます。内容は新プロトコルハンドブックぽいです。

(5) COM M シリーズ、図解 TCP/IP 入門 (オーム社)
比較的分かりやすく、TCP/IP の説明がされています。内容はマスタリング TCP/IP 入門編より詳しいです。

(6) インターフェイス別冊、OpenDesign No.3 イーサネットと TCP/ IP (CQ 出版社)
TCP/IP プロトコルの内容が、実際のプロトコルのダンプ情報を元にしながら詳しく述べられています。また、telnet や FTP などの上位プロトコルに対しても説明されており、其なりに役に立つ資料です。

(7) UNIX マガジン (アスキー)
「ユニマガ」として UNIX ユーザに親しまれている雑誌です。この中に連載されている「UNIX Communication Notes」がネットワークに関するいい資料になります。特に、IP ルーティングについて知りたい方は 93 年 1 月号から 4 月号ぐらいを参考になされると

良いでしょう。

(8) KUINS WORKSHOP'90 報告集（京都大学学術情報ネットワーク機構）
この資料は少し古いですが、私が最もよく勉強した資料です。ネットワークのサブネット化について非常に分かりやすく書かれています。現在、どのようにすれば入手できるのか分かりません。

(9) RFC（Request for comments）
多分、最後に行き着く所はこの RFC しかありません。RFC の文書は色々の所から入手できますが Nifty サーブのインターネットフォーラムにもあります。ただし、私はこの資料は大嫌いです。（英語のため）