

## 第1章 ネットワークと標準化

### 1.1 標準化

ネットワークについて議論する場合、標準化は非常に重要な概念である。なぜならば、ネットワーク上の機器（ノード）が通信を行う場合、通信上の約束事が予め決められていなければ、お互いに通信を行うことは一切不可能だからである。（図 1.1.1）

この通信上の約束事、即ち通信規約のことを一般に**通信プロトコル（Protocol）**と呼ぶ（または単にプロトコルと呼んでも良い）。現在インターネットや LAN（ラン：Local Area Network：同一エリア内のネットワーク）などで広く使われている基本的な通信プロトコルは TCP/IP（テーシーピー・アイピー）と呼ばれる通信プロトコルである。TCP/IP プロトコルは 1982 年には既に、現在使用されているものとほぼ同機能のものが完成している（Version4）。30 年近くも同一バージョンのソフトウェアが使い続けられるという事は驚くべき事であり、他にはあまり例を見ない。

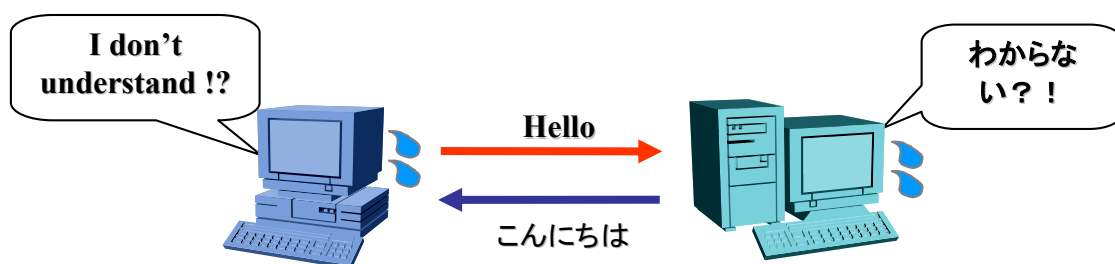


図 1.1.1 プロトコルが違うと通信は不可能

#### 1.1.1 デ・ファクト・スタンダードと ISO

いわゆる標準には、大きく分けると 2 通りある。政府などの公的機関が定める正式（公式）な標準（de jure standard）と、多くのユーザが使用することによって、結果的に標準となる「**事実上の標準**」（デ・ファクト・スタンダード：de facto standard）である。

正式な標準の策定には、様々な団体の利害調節のため多くの時間が費やされるのが普通である。コンピュータの世界での技術革新のスピードには目覚ましいものがあり、ある技術についてその正式な標準の確定を待っていると、その技術そのものが時代遅れのものになってしまう可能性がある。そこでコンピュータ業界などでは、正式な標準の確定を待たずに製品の開発・販売を行ってシェアの拡大を計り、自社ブランド技術が「事実上の標準」になることを目指すのである（最近では BLU-RAY と HD-DVD のシェア争いが最も典型的な例である）。なお、TCP/IP プロトコルは一企業の技術ではないが、「事実上の標準」の代表的な例である。

一方、ISO（International Organization for Standardization：国際標準化機構）は電気分野以外の工業製品全版の正式な標準規格（de jure standard）を定める国際的な標準

化組織である。また、電機分野については国際電気標準会議（IEC： International Electrotechnical Commission）が国際的な規格を決定する。

海外旅行先で買った乾電池や CD が日本から持っていった CD プレーヤーでそのまま使用できるのは、ISO/IEC がそれらの物理的な形状や電気的な特性を定めているからに他ならない。また ISO はネットワークに関しても様々な規格を定めている。

なお、ISO は” 同位 ” を表すギリシャ語の接頭語 “isos” が語源であり、International Organization for Standardization の略号ではない。

### 1.1.2 その他の標準化組織

- ITU (International Telecommunication Union：国際電気通信連合) は、無線通信と電気通信に関する国際標準の策定を行う国際連合の専門機関である。無線周波数利用の割り当てや国際電話の接続のための調整を行っている。ITU は 3 つの部門に分かれており、その中の電気通信標準化部門である ITU-T (アイティユーティ：  
ITU-Telecommunication Standardization Sector：旧 CCITT) は、電気通信分野の標準策定を専門に行う部門であり、モデムや xDSL などの標準化を行っている。ITU-T で策定された標準化案は「ITU-T 勧告」として公表される。
- IEEE (アイトリプルイー：Institute of Electrical and Electronics Engineers Inc.：米国電気電子学会) は米国の電気・電子関連の研究を行う学会である。学会活動と共に専門委員会による電気電子に関する標準化も行っている。IEEE で策定された標準化案は ANSI を通して ISO にも提案され、国際標準として採用される。ネットワークに関する標準化では、IEEE の 802 委員会による一連のシリーズである IEEE802 シリーズ（主に物理層とデータリンク層の規格）が特に有名であり、この規格は ISO802 シリーズとして国際標準にもなっている。つまり IEEE802 シリーズと ISO802 シリーズは同じものである。
- ANSI (アンジ：American National Standards Institute：米国規格協会) は米国の工業製品に関する規格を策定する団体である。日本の日本工業規格 (JIS) や日本工業標準調査会 (JISC) に相当する。米国の国内規格を規定する団体であるが、ANSI の規格がそのまま ISO の国際標準となる場合が多い。ANSI-C/C++などのプログラミング言語の規格化も行っている。

## 1.2 OSI 参照モデルと TCP/IP

1970～1980 年代、ネットワーク上では様々な通信プロトコルが使用されていた (TCP/IP, SNA, AppleTalk, NetWare など)。当然これらのプロトコル間には互換性はなく、お互いに通信を行うことは不可能であった。

1982 年、ISO はこれら問題を解決するために、通信プロトコルの正式な国際標準規格として OSI (Open Systems Interconnection：開放型システム間相互接続) プロトコルの策定を開始する。OSI プロトコルがほぼ完成する 1980 年代後半には、すでに TCP/IP が通信プロ

トコルの事実上の標準としての地位を固めつつあり、当時日本でも如何に TCP/IP から OSI へ移行するかという議論が活発に行われた。

しかしながら、(現在のネットワーク状況からも一目瞭然だが) 結局 OSI は失敗し、TCP/IP が名実ともにネットワークの標準プロトコルとしての地位を獲得する。OSI は失敗してしまったが、その中の **OSI 参照モデル (OSI Reference Model)** だけは、その優れた考え方故に現在まで生き残っている。

OSI 参照モデルは複数の通信プロトコル間の機能の標準的な物差しとして働く。ネットワーク機能全体を 7 層のモジュールに分解し、各層ごとに機能を独立させネットワーク全体を見通しの良いものになっている。OSI 参照モデルは実際の通信プロトコルを表したものではなく、あくまでも概念的なものである。しかしながら、ネットワークの機能を理解し議論するためには、この階層構造の概念を理解することが必須となる。

一方 実際の通信プロトコルである TCP/IP は 4 層の構造しか持たない。図 1.2.1 に OSI 参照モデルと TCP/IP の階層構造の関係を示す。TCP/IP ではプレゼンテーション層とセッション層が無いため、TCP/IP でネットワークプログラムを作成する際には、プログラムは自らこれらの機能を実現するプログラムコードを作成しなければならない。

図では、OSI と TCP/IP の各層の区切りがきっちりと対応しているように見えるが、実際には各階層の区切りは若干ずれたものとなっている。

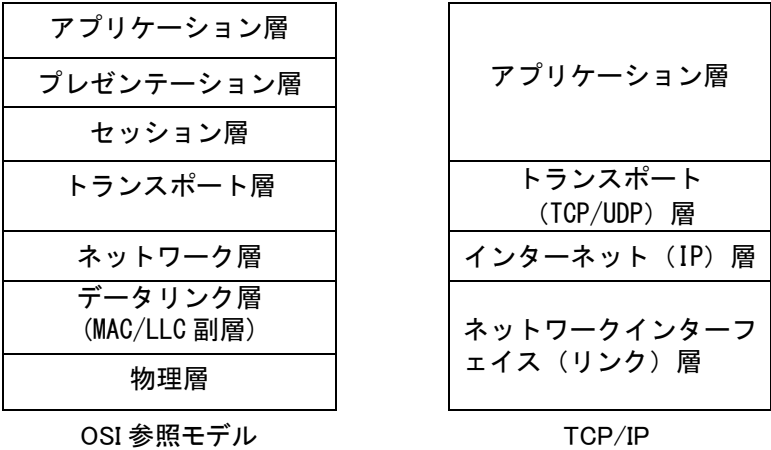


図 1.2.1 OSI 参照モデルと TCP/IP の階層構造

1.2.1 OSI 参照モデルの簡単な説明

OSI の参照モデルの各層について簡単に説明する。詳細な説明については2章以下で行う。

【物理層】

ケーブルへの接続方法、ビットの 0,1 の電圧などを規定し、実際に信号を伝送する。この層で交換されるデータの単位は**ビット**である。ケーブル (メディア) 自体は物理層、つまり OSI 参照モデルには含まれない。

### 【データリンク層】

同じネットワーク内に隣接する他の通信機器（ノード）へ信号を伝達する。このデータリンク層は論理的な制御を行う **LLC (Logical Link Control) 副層**と物理的な制御を行う **MAC (マック : Media Access Control) 副層**とに分けられる。MAC 副層で物理層での違いが吸収されるために、物理層以下で様々な形態のケーブルを利用することが可能となる。

同一ネットワーク内の通信機器は MAC 副層の 48bit の **MAC アドレス (マックアドレス)** で識別される。MAC アドレスは通常は **NIC (ニック : Network Information Card)** の ROM に焼き付けられているため、ハードウェアアドレスや**物理アドレス**とも呼ばれる。この層で交換されるデータの単位は**フレーム**と呼ばれる。

### 【ネットワーク層】

他のネットワーク上の通信機器（ノード）へ信号を伝達する。TCP/IP の場合、ネットワーク上のノードは 32bit の **IP アドレス**と呼ばれるアドレスで識別される。物理的な MAC アドレスに対して IP アドレスは**論理的なアドレス**であると言われる。交換されるデータの単位は**パケット**である。

### 【トランスポート層】

他のノード上のプロセスと通信（**プロセス間通信**）を行う。TCP/IP の場合は、プロセスは**ポート番号**と呼ばれる 16bit の符号なしの整数で識別される。交換されるデータの単位は**セグメント**である。

### 【セッション層】

プロセス間通信のセッション管理を行う。すなわち、通信の開始、継続、終了を管理する。

### 【プレゼンテーション層】

交換されるアプリケーションデータのコード系の設定、データ圧縮・伸張、暗号化・復号などを行う。

### 【アプリケーション層】

アプリケーションそのもの。

図 1.2.2 に OSI 参照モデルにおけるデータの処理順序を示す。図の向かって左側が送信ノードであり、右側が受信ノードである。

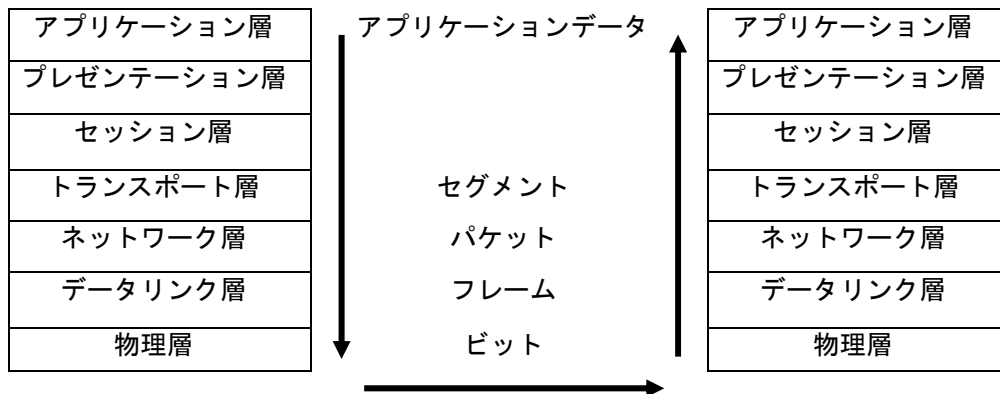


図 1.2.2 データの処理順序

### 1.2.2 OSI 参照モデルと郵便との対比

OSI 参照モデルの一般的な場合の各層の働きについて、郵便を例にとって対比してみる。ただし、元々全く違うシステム間の対比なので、完全な対応とはならないことに注意されたい。

#### 《送信側》

1. Bob は Alice に手紙を書くことにする。(アプリケーション層)
2. まず内容を日本語で書くか、英語で書くか迷い、結局日本語にする。もしかすると他人に読まれるかもしれないので、Alice との間で予め決めておいた暗号（言い回し）を使うことにする。(文字コードの確定と暗号化：プレゼンテーション層)
3. 「こんにちは。」で文章を便箋に書き初め、「それじゃまたね。」で文章を終える。ここでは電話での例の方が分かり易いかもしれない。電話に例えれば、「もしもし」で電話を開始し、「バイバイ。」で電話を終わる。ただし読者も経験があるかもしれないが、電話を切る場合にどちらから先に電話を切るかで迷う場合も少なくない。ネットワークアプリケーションでも同様に、セッションの開始や継続の管理はさほど難しくはないが、セッションをどのタイミングで終了させるかということは、かなり難しい場合がある。(セッションの管理：セッション層)
4. 便箋を封筒に入れ、封筒の表に「Alice 様へ」と書く。(通信相手の確定。相手の名前はポート番号に相当：トランスポート層)
5. さらにそれを別の封筒に入れ、封筒の表に Alice の住む町の住所と苗字（ファミリーネーム）を書く。ただし、同一町内に同じ苗字（ファミリーネーム）の家はないものとする。(ネットワーク上の通信ノードの確定。相手の住む町の住所と苗字は IP アドレスに相当：ネットワーク層)
6. Alice が同じ町内なら Alice の家の番地、違う町に住んでいるのなら郵便局の番地を調べる。5 の封筒をさらに別の封筒に入れ、封筒の表に届け先の番地を書く。(隣

接するノードへ信号を伝送する。番地は MAC アドレスに相当：データリンク層)

7. 徒歩, 自転車またはスクータで届け先の番地まで手紙を届ける。(信号の伝送：物理層)

#### 《受信側》

1. Alice 宅に郵便局経由または, Bob から直接郵便が届けられる。(信号の伝送：物理層)
2. 封筒の表の番地と自分の家の番地が一致していれば封筒の中から次の封筒を取り出し, 違っていれば破棄する。(番地は MAC アドレスに相当：データリンク層)
3. 取り出した封筒の表の住所と苗字が自宅宛であれば, 封筒の中の次の封筒を取り出し, 違っていれば破棄する。(住所と苗字は IP アドレスに相当：ネットワーク層)
4. 取り出した封筒の表に記載されている名前の人物 (ここでは Alice) に封筒の中の便箋を渡す。(名前はポート番号に相当：トランスポート層)
5. 手紙の文章の始まりと終わりを識別する。(セッションの管理：セッション層)
6. 手紙の文章の言語 (日本語か英語か) を識別し, Bob と予め決めておいた暗号 (言い回し) を元に戻す。(データの復号と文字コードの認識：プレゼンテーション層)
7. Bob からの手紙の内容を読む。(アプリケーション層)

### 1.2.3 カプセル化とカプセル化の解除

前節の OSI 参照モデルと郵便との対比で, 手紙を何度か封筒に入れ直す場面がある (《送信側》4~6)。これらは上位層のデータを下位層のデータの中に埋め込むことを表し, これらの処理をデータの**カプセル化**と呼ぶ。

TCP/IP のカプセル化の場合には, アプリケーションデータに対して, 順に TCP(UDP)ヘッダ, IP ヘッダ, フレームヘッダと呼ばれるヘッダデータが付加される (データの最後にトレーラと呼ばれるデータが付加される場合もある) (図 1.2.3, 図 1.2.4)。

逆に受信側で下位層から上位層にデータが渡されるときに行われる (前節の《受信側》2~4), 封筒の中から次の封筒を取り出す処理を**アンカプセル化 (カプセル化の解除)**と呼ぶ。これらは下位層のデータの中から上位層のデータを取り出すことを表す。

以上をまとめると,

- 下の階層にデータが渡されるに従って, ヘッダやトレーラが付加される [封筒に入れる] → **カプセル化**
- 上の階層にデータが渡されるに従って, ヘッダやトレーラが取り除かれる [封筒から取り出す] → **アンカプセル化**

となる。

図 1.2.5 に, OSI 参照モデルを郵便に例えた場合のカプセル化・アンカプセル化のイメージも示す。

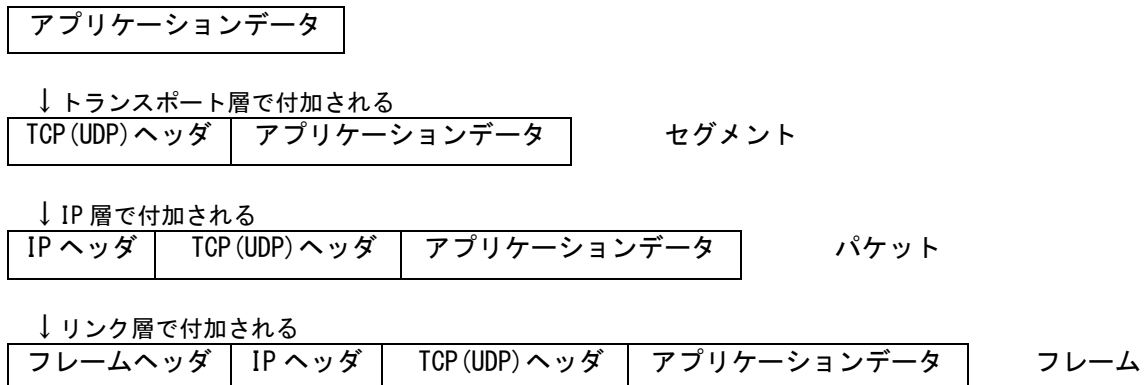
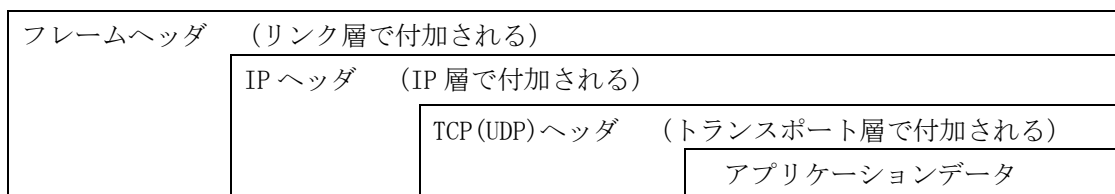


図 1. 2. 3 TCP/IP でのデータのカプセル化

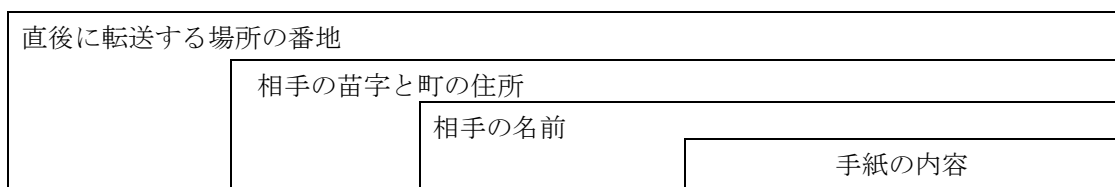
TCP/IP では 3 度ヘッダを付加する（トレーラが付く場合もある）。



フレームヘッダ：相手が同じネットワーク内 → 相手の MAC アドレス付加される  
                   ：相手が違うネットワーク → ルータの MAC アドレス付加される  
 IP ヘッダ：IP アドレスが付加される  
 TCP (UDP) ヘッダ：ポート番号が付加される

図 1. 2. 4 カプセル化のイメージ

郵便に例えると 3 重の封筒に入れる。



直後に転送する場所の番地：相手が同じ町内 → 相手の家の番地が付加される  
                                   ：相手が違う町 → 郵便局の番地が付加される

図 1. 2. 5 手紙に例えた場合のカプセル化のイメージ

### 1. 3 ネットワーク上の中継器

ネットワークを形成する場合、中継器なしにこれを行うことは不可能である。ネットワーク上での中継器は、中継を行う層により大きく 4 つに分類される (図 1. 3. 1)。

#### 1. 3. 1 物理層での中継器

物理層でビット列の中継を行う機器は一般にリピータと呼ばれる。特にスター型ネットワーク（詳しくは次章参照，なお現在の LAN は殆どがスター型である）で使用されるリピータをリピータハブ（またはシェアードハブ）と呼ぶ。リピータ（リピータハブ）は 0, 1

の電気信号の増幅のみを行い、結果としてケーブルの延長を実現する。電気信号をそのまま増幅・中継するので、雑音などもそのまま増幅・中継される。

しかし現在では、これよりも高性能な**スイッチングハブ**が非常に安価になったため、実際のネットワーク上ではリピータ（リピータハブ）は殆ど使われていない。市場にも殆ど出回っていないのが現状である。

### 1.3.2 データリンク層での中継器

データリンク層でフレームの中継を行う機器は一般に**ブリッジ**と呼ばれる。特にスター型ネットワークで使用されるブリッジを**スイッチングハブ**、または単に**スイッチ**と呼ぶ。またネットワーク層の L3 スイッチと区別するために **L2 スイッチ**と呼ぶ場合もある（L2 はレイヤー2、すなわち第2層を表す）。

ブリッジ（スイッチングハブ）は MAC アドレスを学習してフレームの中継するため、受信先の通信機器（ノード）が存在しないケーブル（通信ポート）には信号を流すことはなく、効率的な通信を行うことが可能である。

### 1.3.3 ネットワーク層での中継器

ネットワーク層でパケットの中継を行う機器は**ルータ**と呼ばれる。特にスター型ネットワークで使用される、スイッチングハブ機能を持ったルータを **Layer3 スwitchングハブ**、または **L3 スイッチ**と呼ぶ（ネットワーク層は第3層であるから）。

ルータはネットワークとネットワークを繋ぎ、IP アドレスによってパケットの転送経路を決定する。逆の言い方をすると、「**ルータはネットワークを分割する**」とも言える。ルータは 1-2-2 節の郵便局に相当する。

なお、リピータ（リピータハブ）をブリッジ（スイッチングハブ）で置き換えることは可能だが、ブリッジ（スイッチングハブ）とルータはその機能が大きく違うため、単純にブリッジ（スイッチングハブ）をルータで置き換えることはできない。

### 1.3.4 アプリケーション層での中継器

アプリケーション層での中継はプロトコル変換やアプリケーションデータの変換に用いられる。プロトコル変換やアプリケーションデータ変換を行う機器を正式には**ゲートウェイ**と呼ぶ。ただし一般的にはゲートウェイと言えばルータのことを指す場合が殆どであり、プロトコル変換やアプリケーションデータの変換を行うゲートウェイは特別に**アプリケーションゲートウェイ**、または**アプリケーションレベルゲートウェイ (ALG)**と呼ぶ場合が多い。



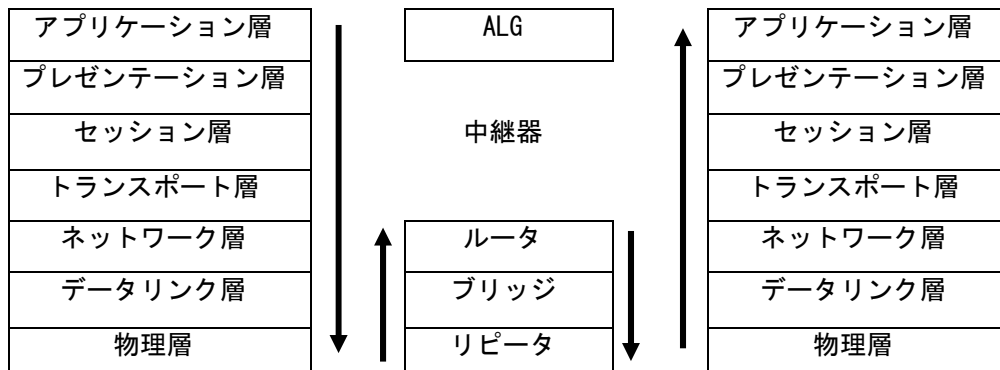


図 1.3.1 中継機器の位置づけ

#### 1.4 RFC (Request for Comments)

RFC (Request For Comments) は IETF (Internet Engineering Task Force) が取り纏めを行う、インターネットに関する様々な情報文章集である。代表的な内容としては、インターネットに関する技術的な仕様書、新しいサービスの提案、守るべきルール、用語集、ジョークなどがある。いわゆる正式な標準 (de jure standard) ではないが、「インターネットの法律書」と言っても過言ではない。

RFC は誰でも投稿することが可能で、標準化に関する議論なども完全にオープンに行われる。提案が RFC として採用された場合には一連の番号が割り振られ、その番号の元に管理される。一度採用された RFC は訂正や削除されることはなく、内容の修正や拡張などを行う場合には新規の RFC として新しい番号が割り当てられる。

例えばインターネットのメール (SMTP) に関する規格は RFC812, RFC2821, RFC5321 などに記述されている。

## 1.5 練習問題

### 【問題】

- 次のうち、公的な標準化組織ではないものはどれか？（複数選択）  
☐ a. ISO   ☐ b. OSI   ☐ c. IEEE   ☐ d. ANSI   ☐ e. ITU-T   ☐ f. ICANN
- OSI 参照モデルの各階層の名前を、下位層から順番に7層全て答えなさい。
- デ・ファクト・スタンダードの意味として正しいものはどれか？  
☐ a. 正式な組織により決定された規格  
☐ b. 今では無効になってしまった過去の規格  
☐ c. 正式な標準ではないが、大多数の人々が利用している規格  
☐ d. 現在、規格化が進行途中であるような規格
- インターネットの基本的な通信プロトコルは何か？  
☐ a. OSI   ☐ b. TCP/IP   ☐ c. WWW   ☐ d. RFC   ☐ e. HTTP
- データリンク層でデータの中継を行なう機器はどれか？  
☐ a. スイッチングハブ   ☐ b. ルータ   ☐ c. リピータハブ  
☐ d. ゲートウェイ
- カプセル化の説明として正しいものはどれか？  
☐ a. 下位層のデータを上位層のデータの中に埋め込む  
☐ b. 下位層のデータの中から上位層のデータを取り出す。  
☐ c. 上位層のデータを下位層のデータの中に埋め込む  
☐ d. 上位層のデータの中から下位層のデータを取り出す。
- IETF (Internet Engineering Task Force) が取りまとめるインターネットの規約集は何と呼ばれているか？  
☐ a. IEDoc (Internet Engineering Documentations)  
☐ b. POI (Protocols of Internet)  
☐ c. W3C (World Wide Web Consortium)  
☐ d. FAQ (Frequently Asked Question)  
☐ e. RFC (Request For Comments)

### 【解答】

- a: OSI はプロトコルの名称  
f: ICANN はインターネット上のリソースの管理を行なう組織の名称
- 物理層 → データリンク層 → ネットワーク層 → トランスポート層 → セッション層 → プレゼンテーション層 → アプリケーション層
- c
- b
- a
- c
- e



## コラム

### 【1Byte は 8bit か？】

1Byte と言えば通常では 8bit の事を指すが、本当に、常に 1Byte は 8bit なのだろうか？元来 1Byte は 1 つの文字を表すのに必要なビット数を表していた。この定義から考えれば、英数字と若干の特殊記号だけなら 7bit (128 文字) で全文字を十分に表現可能である。

実際にパソコンの文字コードの元祖である ASCII コードは 7bit である。この場合の 1Byte は 7bit となり、この状態を 7-bit byte と呼ぶ。初期の頃のコンピュータでは 1Byte が 7bit であるマシン (7-bit byte マシン) が多く存在していた。このようなマシンに 8bit の文字データを入力すると、8bit 目は使用しないか (無視するか)、エラー検査 (パリティチェック) 用のパリティとして利用されるので文字化けを起こす。

初期のインターネットではメールはマシン間をバケツリレー式に転送されていた。もしバケツリレーを行っているマシンの中に一台でも 7-bit byte マシンがあれば、8bit の文字コードは文字化けを起こして、メールを読むことができなくなる。この理由から初期のメールの規格では、メールの内容は全て 7bit コードであることが求められた。そのため日本語のメールを送る場合にも、7bit の日本語コードである JIS コードが日本語メールの標準文字コードとなった (現在の規格では 8bit コードを送ることも可能である)。

以上の事から分るように、実は常に 1Byte が 8bit であるとは限ないのである。ネットワークの世界では、この曖昧性を排除するために 8bit を 1 **Octet** (オクテット) と表現する場合もある (Oct は 8 を表す)。ただし、本書では特に断らない限りは、1Byte を 8bit として (8-bit byte) として話を進める。

## コラム

### 【どうして OSI プロトコルは失敗したのか？】

1980 年代後半に、当時主流であった TCP/IP プロトコルと置き換えるために、鳴り物入りで登場した OSI プロトコルは、結局あっけなく失敗してしまった。OSI プロトコルはどうして失敗してしまったのだろうか？ 理由は幾つもあると考えられるが、その中で代表的な物を 2 つ挙げたい。

まずはコストの問題である。OSI プロトコルと TCP/IP プロトコルは互換性がないので、ネットワークプロトコルを TCP/IP から OSI に変更すると、その上で動作していたアプリケーションを全て買い換えなければ（作り替えなければ）ならない。これは非常にコストがかかる作業である。そしてそのコストは各ユーザや組織が負担しなければならない。

次に接続性の問題である。ある組織が自分の所だけ TCP/IP から OSI に切り替えたとする。実はその瞬間から、その組織はインターネット上のほぼ全てのサービスを利用できなくなる。インターネット上の他の組織は相変わらず TCP/IP プロトコルを使用しているからである。

例えば、「グリニッジ標準時の XX 年 YY 月 ZZ 日の 0:00 に世界中のネットワークを OSI プロトコルに変更する」と強制できればこの問題は回避できるが、実際問題としてこのようなことは不可能である。

以上のような理由から TCP/IP プロトコルから OSI プロトコルへの移行は結局失敗し、現在まで TCP/IP が使い続けられている。

ところが、実は今日でも同じような問題が発生している。TCP/IP プロトコルでの IPv4 (IP Version 4) から IPv6 (IP Version 6) への移行問題である（ちなみに IPv5 は実験用のプロトコルであった）。現在では何年も掛けて徐々に通信機器やソフトウェアが、標準で IPv6 を搭載するようになって来ており、コストの問題はほぼクリアされつつある。後は接続性の問題であるが、もしこれが失敗した場合の影響は OSI プロトコルの失敗時の比ではないため（失敗した場合、インターネットは壊滅的な状況に陥ると予想される）、慎重に移行作業が進められている。詳しくは「12-8. IPv4 から IPv6 への移行」を参照せよ。