

# 応用情報技術者試験

---

## テクノロジ系Ⅳ

---

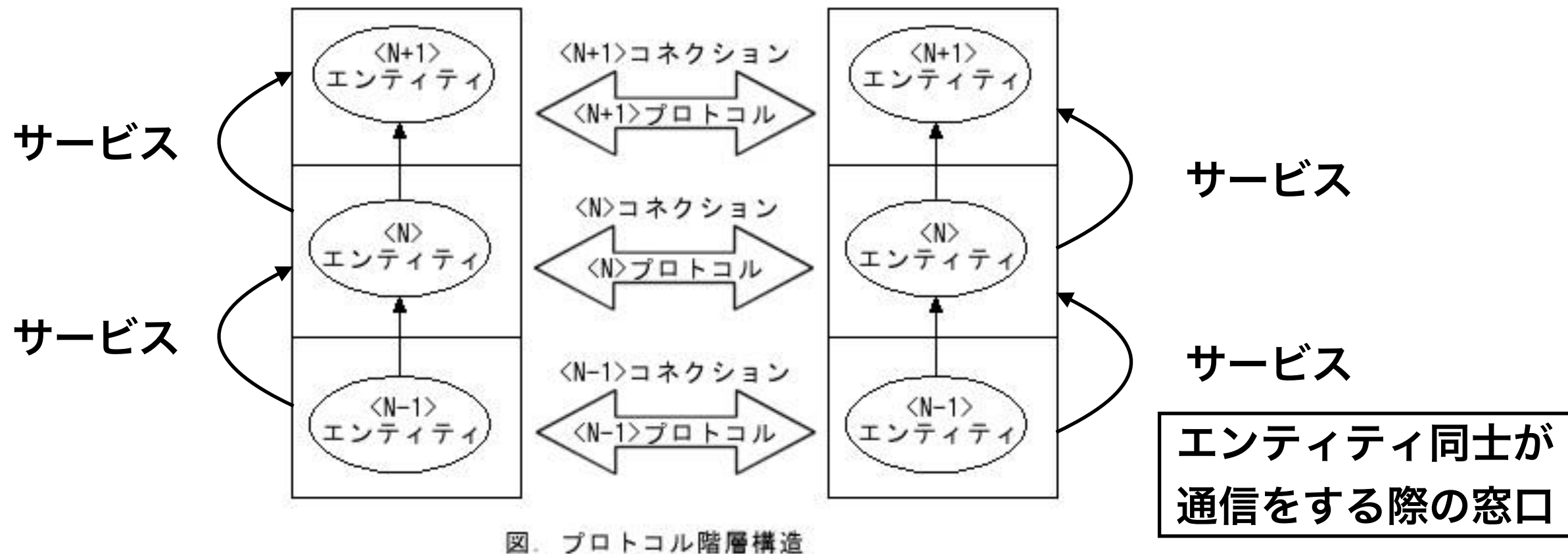
# 3-4-3

## 通信プロトコル

優先度： 高

プロトコル

# プロトコル



- **プロトコル**：通信を行う際の取り決めのこと
- OSI参照モデルでは、各階層を**N層**と表現
- \* **エンティティ**：N層に存在する通信機器などの実体

# 目次

- TCP/IP

- \* MACアドレス・コネクション

- TCP/IPの中核プロトコル

- \*TCP・UDP・ポート番号

- IPアドレス

- アプリケーション層のプロトコル

- アドレス変換技術

- IPv4・IPv6

**基本的には情報ネットワークI・II  
(ネットワークアーキテクチャ・  
ネットワークアプリケーション)  
の復習内容です**

# TCP/IP

# TCP/IP

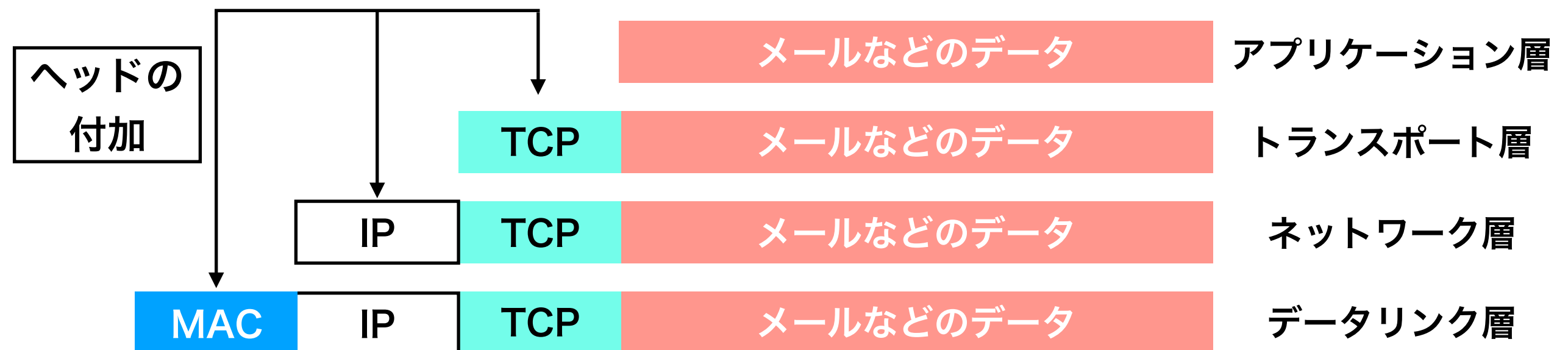
- ネットワーク層のプロトコルであるIPを中心としたプロトコル群  
(**プロトコルスイート**)

\* プロトコルスイートでは以下のような独自の階層を持つ

| OSI参照モデル   | TCP/IP         | 実装例                                            |
|------------|----------------|------------------------------------------------|
| アプリケーション層  | アプリケーション層      | HTTP, SMTP, FTP, DHCP, DNS, SNMP, POP3, Telnet |
| プレゼンテーション層 |                |                                                |
| セッション層     | トランスポート層       | TCP, UDP                                       |
| トランスポート層   | インターネット層       | IP, ICMPなど                                     |
| ネットワーク層    | ネットワークインタフェース層 | イーサネット                                         |
| データリンク層    |                |                                                |
| 物理層        |                |                                                |

# TCP/IPの通信

- TCP/IPでは**パケット交換方式**でデータのやり取りが行われる
  - \* データを**パケット**と呼ばれる単位に区切り, 各パケットに**ヘッダ**をつけて送信する
    - ▶ TCP/IPでは, 各階層毎にこのヘッダが付加され, 次の階層へと渡される
    - ▶ ヘッダに含まれる情報: その階層で必要となる送信元や送信先, 大きさ, 順番など



# ヘッドを付加したパケットの 呼び名

|          |                                                                                                |
|----------|------------------------------------------------------------------------------------------------|
| TCPセグメント | <b>ポート番号</b> によって、トランスポート層の通信を行うTCPヘッダを付加したパケット                                                |
| IPパケット   | <b>IPアドレス</b> によってネットワーク層の通信を行うIPヘッダを付加したパケット                                                  |
| MACフレーム  | <b>MACアドレス</b> によってデータリンク層の通信を行うMACヘッダを付加したパケット. データリンク層の規格イーサネットでは <b>イーサネットフレームワーク</b> と呼ばれる |

# MACアドレス

- **MAC(Media Access Control)アドレス**

- \* **イーサネット**や**FDDI**で使用するNIC(ネットワークインタフェースカード)に割り当てられる**6**バイトの情報(番号)

- ▶ 先頭**24**ビットのOUI(ベンダID)と後続**24**ビットの固有製造番号から構成される

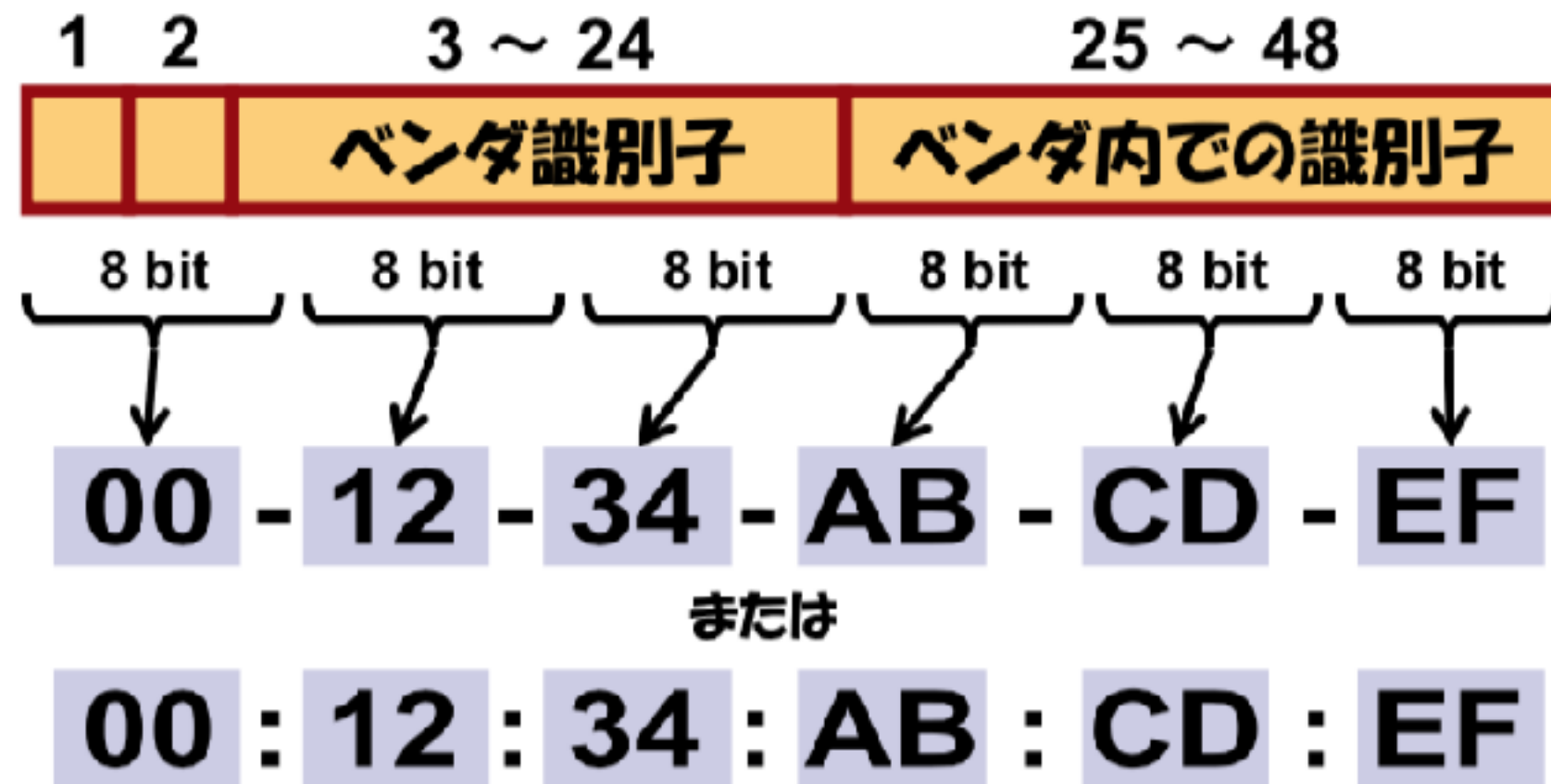
- データリンク層で使用

- \* **同じネットワークに接続された隣接ノード間の通信で相手を識別**

- MACアドレスは機器が固有に持つ番号

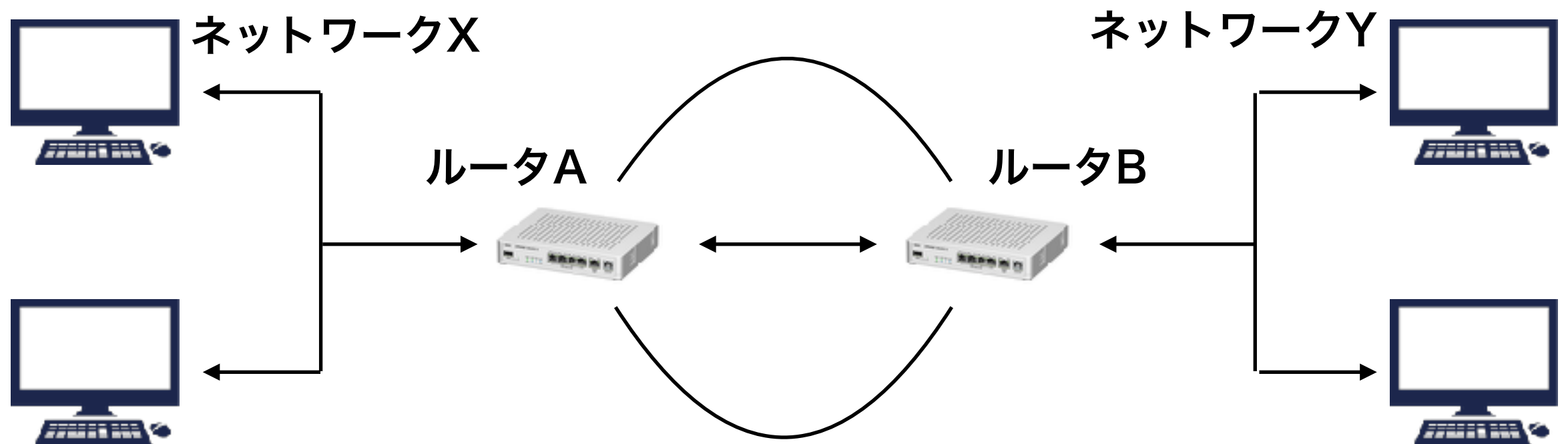
- \* **必ず一意**に定まるように**IEEE**が管理している

# MACアドレスの表記法



# ネットワーク間の通信

- 同じネットワークに接続されたノード間は**データリンク層**の通信
- ネットワークを超えたノード間は**ネットワーク層**の通信



|              | ノードA   | ルータA | ルータB     | ノードB     |
|--------------|--------|------|----------|----------|
| 宛先IPアドレスの遷移  | (ノードB) | →    | →        | →        |
| 宛先MACアドレスの遷移 | (ルータA) | →    | (ルータB) → | (ノードB) → |

# MACアドレスに関するプロトコル

- **ARP**(Address Resolution Protocol)

- \* IPアドレスからMACアドレスを得るためのプロトコル

- ▶ ARP要求パケットを**ブロードキャスト**(全員向けのパケット)で送出
    - ▶ IPアドレスだけが該当する場合は, ARP応答パケットを**ユニキャスト**(相手だけにに向けたパケット)で送出

- **RARP**(Reverse Address Resolution Protocol)

- \* MACアドレスからIPアドレスを得るためのプロトコル

- ▶ ハードディスクがなく, 自分のIPアドレスだけを保持しておけないPCなどが利用

# コネクション型とコネクションレス型

## ■ コネクション型

- 通信相手とコネクションを確立し、通信を始める。通信が終わったら切断する。

### □ ストリーム型



## ■ コネクションレス型

- 通信相手とコネクションを確立せず、いきなり通信を始める。

### □ データグラム型



# コネクション型



- **送った順序通りに確実に**相手に届ける
  - 順序制御, 再送制御
- **信頼性は有るが処理が複雑**
- 確実に相手に届いて欲しい通信, 信頼性の必要な通信に用いる



# コネクションレス型



- データが**確実に届く保証はない**
- **信頼性はない**が**処理が単純**
- データが少し抜けても影響の少ない通信, リアルタイム性の必要な通信に用いる



# TCP/IPの中核プロトコル

TCPとUDPの説明が8行くらいしかなかったので、  
塚田先生のスライドと別の参考書を用いて  
ボリュームミーにしてみました

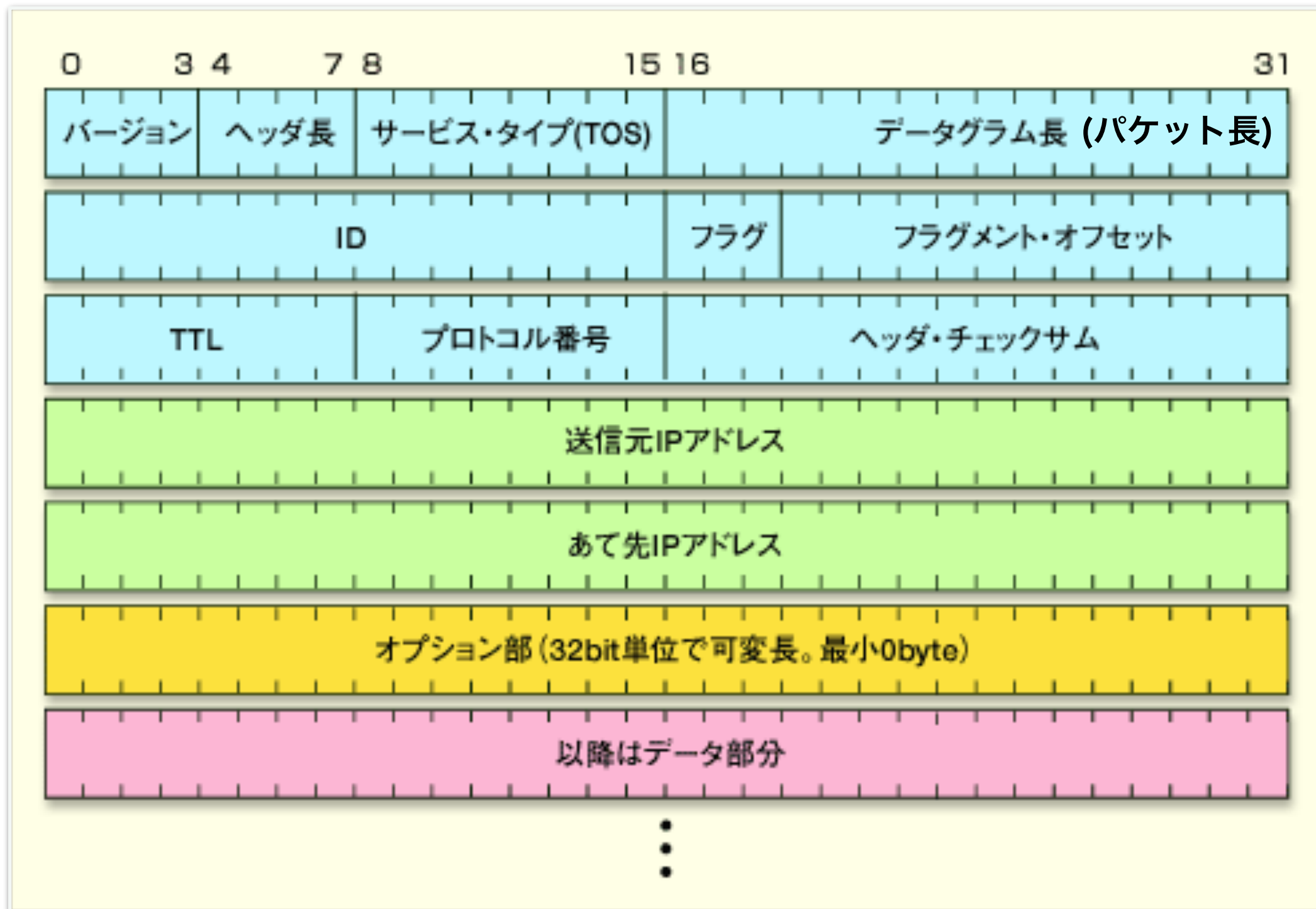
# ネットワーク層のプロトコル

- **IP**(Internet Protocol)

- パケット通信技術である
- コネクションレス型通信である
- IPアドレスを使った経路制御を行う

- ネットワーク層のプロトコルは, ネットワークを超えた通信を提供する機能を持つ
  - \* ルータがIPアドレスを手がかりに通信の振り分けを行う
  - \* 宛先IPアドレスがネットワーク外である場合, 宛先により近いルータへ転送を繰り返すことで, パケットを目的地に到達させる(**ルーティング**)

# IPヘッダの構成(IPv4)



# トランスポート層

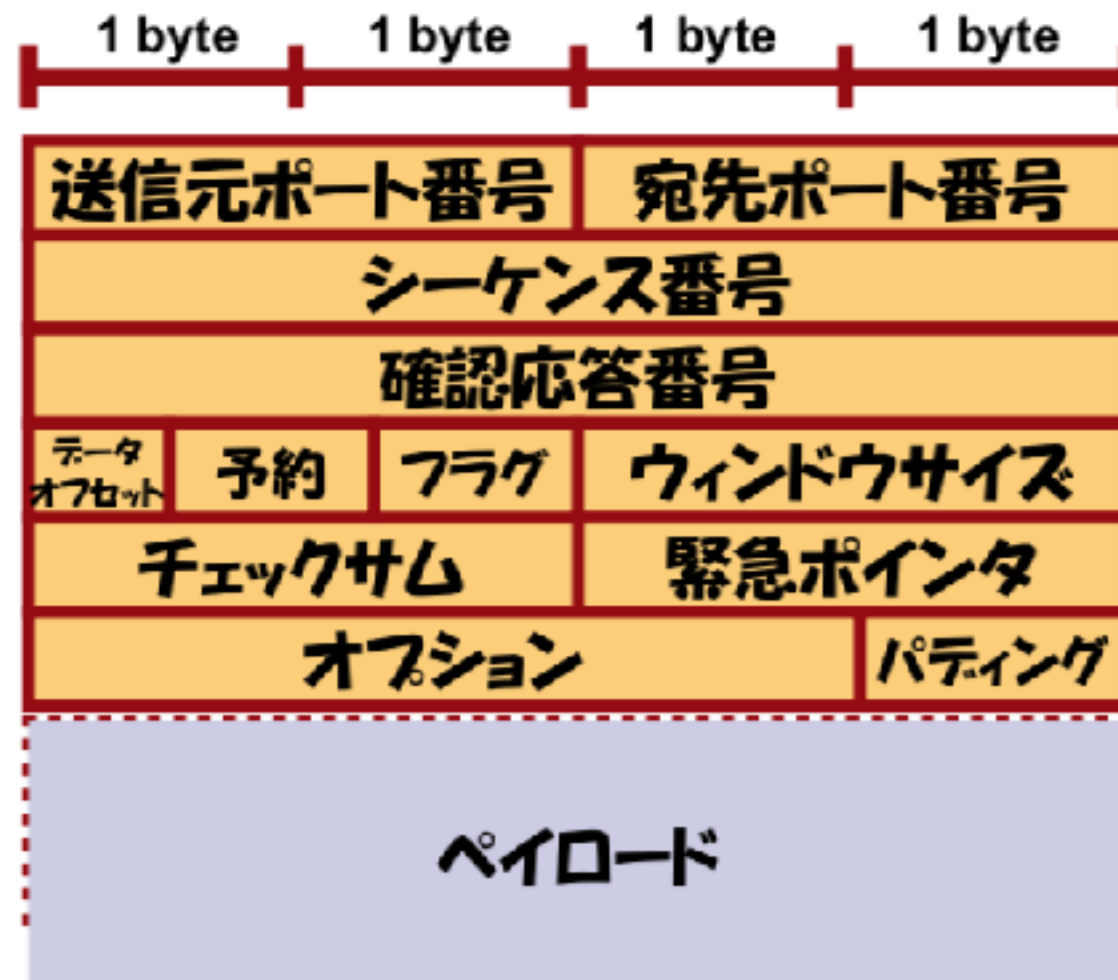
- **トランスポート層**：IPを補完し、データ送信の品質や信頼性を向上させるための層
  - \* プロトコルには**TCP**と**UDP**がある

# TCP

- **TCP(Transmission Control Protocol)**
  - \* 下位層に位置するIPがデータ通信の安全性を保証しない  
コネクションレス型通信
    - ▶ その補完のために送達管理, 伝送管理の機能を持った  
**コネクション型プロトコル**
- TCPではコネクション確立のため, コネクションの**確立要求**  
**パケット(SYN)**と**確認応答パケット(ACK)**のやり取りを行  
う**3ウェイハンドシェイク**を行う

# TCPヘッダの構成

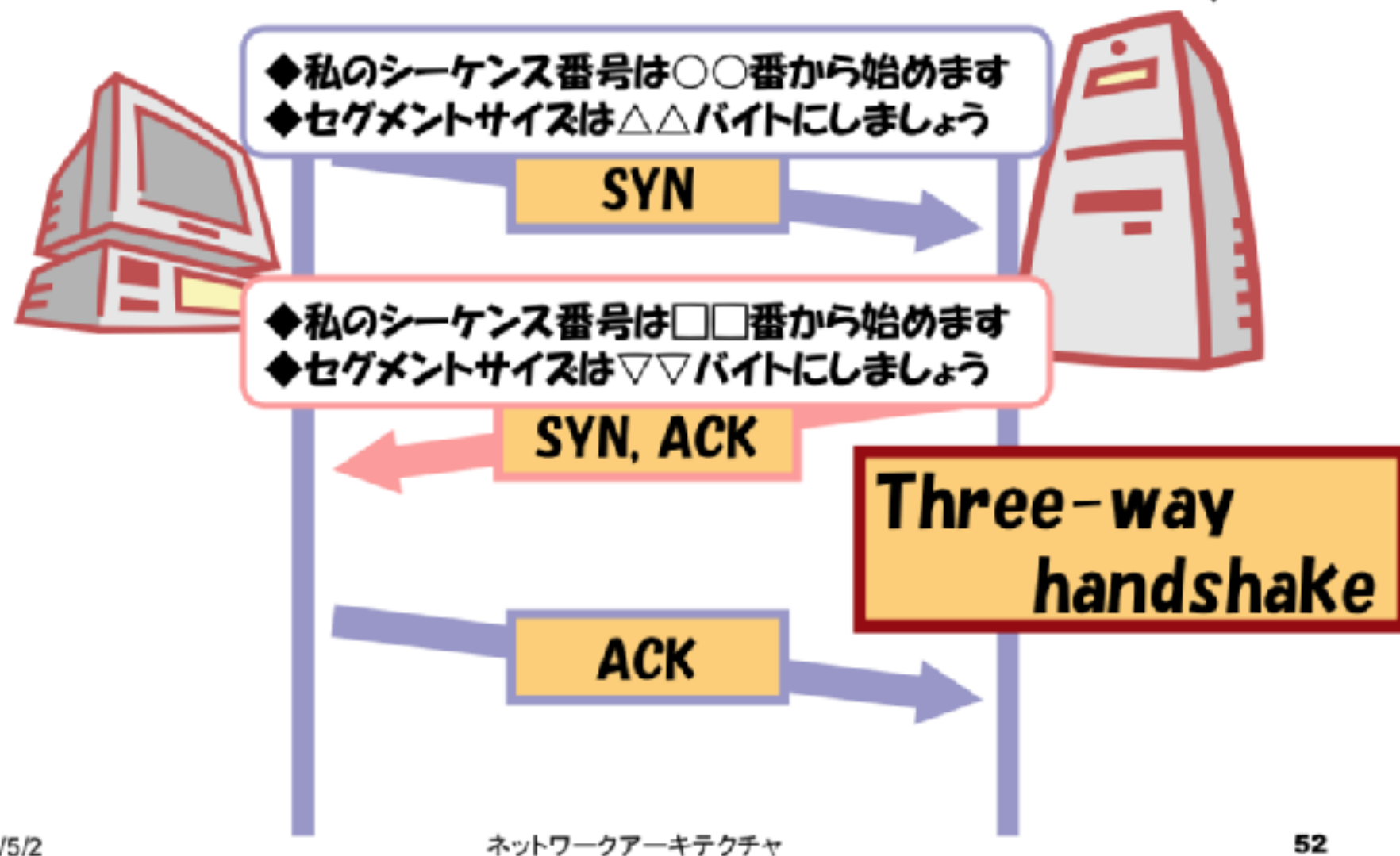
## TCPヘッダ



# 3 ウェイハンドシェイク

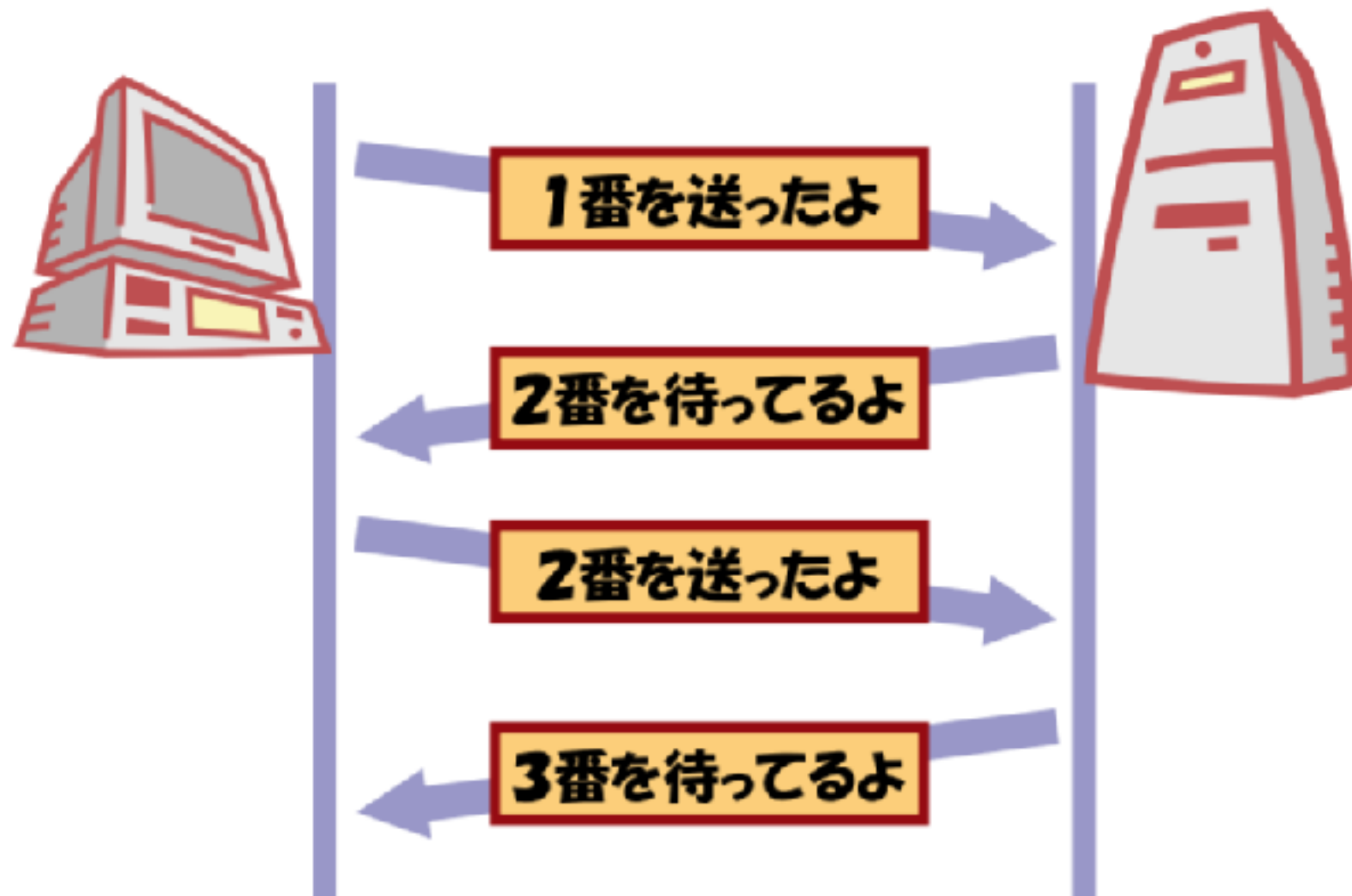
コネクション確立

(TCP Connection Establishment)



# シーケンス番号

## シーケンス番号



# UDP

- UDP(User Datagram Protocol)

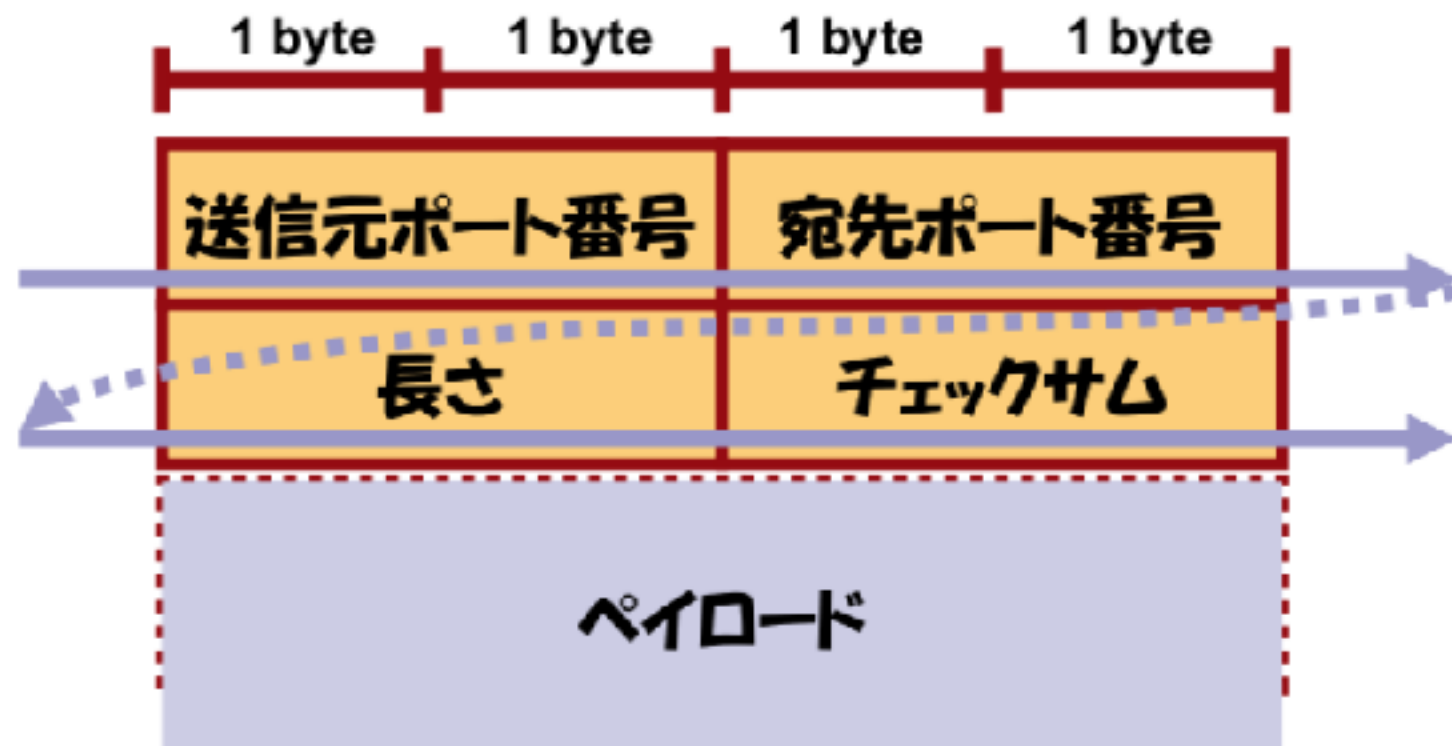
- \* 送達管理を行わない**コネクションレス型**通信

- ▶ TCPでやり取りしていたシーケンス番号やACK番号などの情報は扱わない

- データ落ちが発生した場合の再送なども行わないので, **高速**な分, **信頼性は低くなる**

# UDPヘッダの構成

## UDPヘッダ

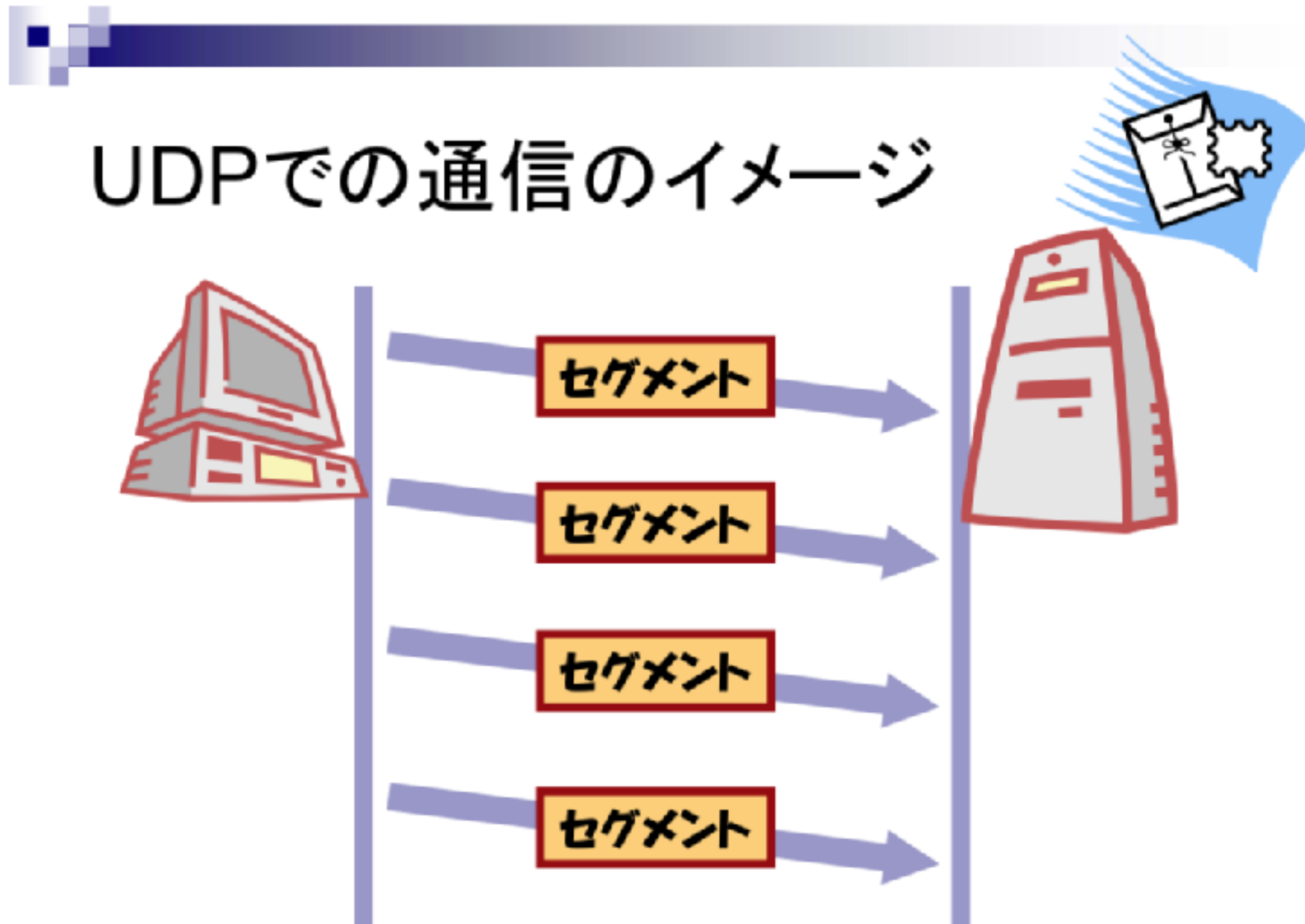


先頭



実際にはセグメントの先頭からこのように並んでいる

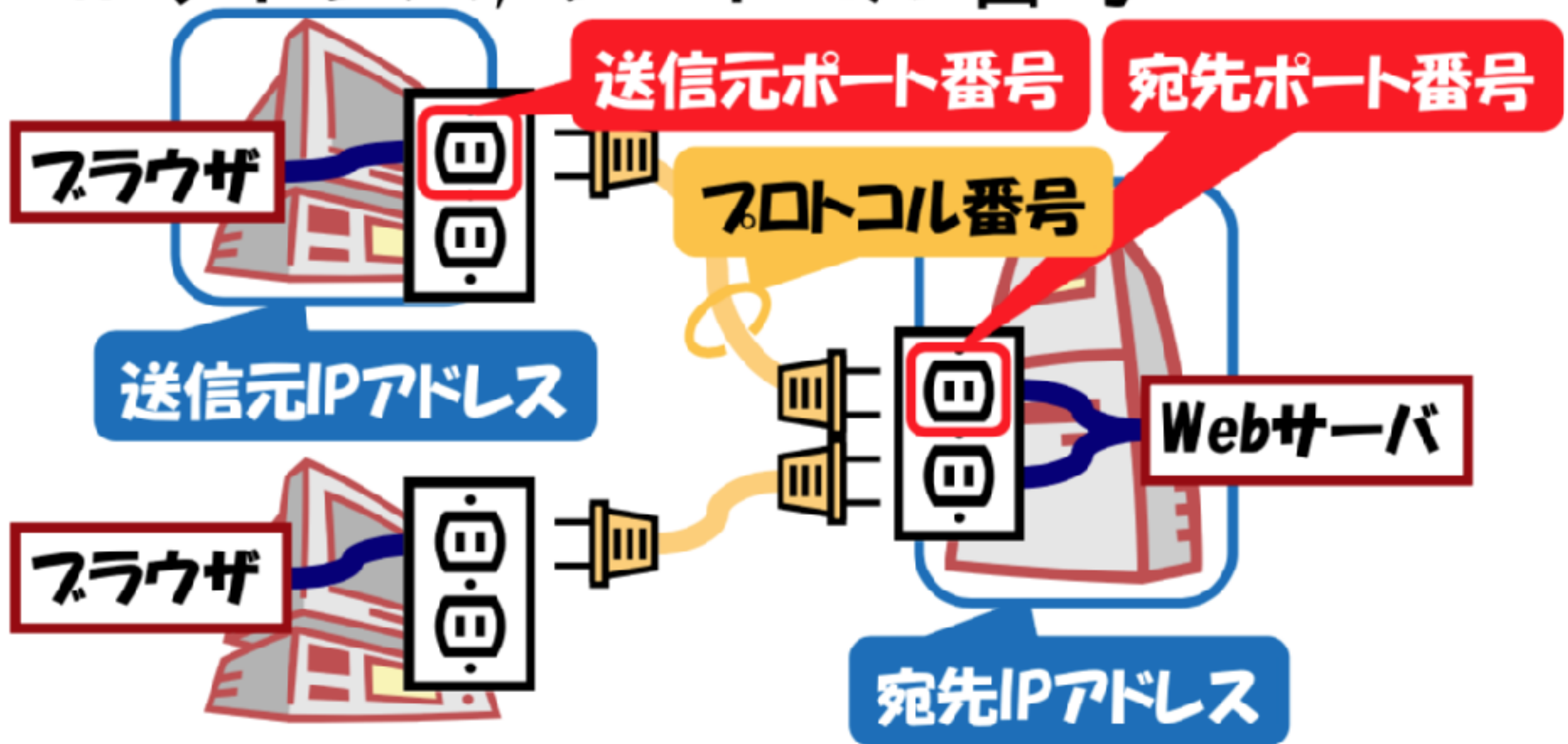
# UDP通信のイメージ



# アプリケーション間の通信

- トラnsポート層のもう1つの役割：アプリケーション間の通信を実現
  - \* 各アプリケーションは、トラnsポート層のヘッダにある「宛先ポート番号」を見ることで自分宛てのデータか否か判断
    - ▶ **ポート番号**：トラnsポート層において、ノード内のアプリケーションの識別に利用する識別子番号. **0～65535**の範囲で指定
    - ▶ よく利用されるアプリケーションについてのポート番号は、**Well-Knownポート**として標準化されている. TCP・UDPそれぞれ**0～1023**番までの番号を割り当てている

ポート番号,  
IPアドレス, プロトコル番号



# IPアドレス

# IPアドレスの実体

- IPアドレス: ネットワーク層のプロトコルIPで利用されるノードを特定するためのアドレス
- IPアドレスは各国のNIC(Network Information Center)が割り当てなどの管理を行っている
  - \* インターネットに直接接続する場合には, ISP(Internet Service Provider)に申請し, IPアドレスを取得する

# IPアドレスの表記

- **IPv4**では, IPアドレスとして**32桁**の2進数を利用

\* 通常, 8ビットずつ区切って10進数表記にする

|        |            |          |          |          |
|--------|------------|----------|----------|----------|
| 2進数表記  | : 11011011 | 01100101 | 11000110 | 00000100 |
| 10進数表記 | : 219.     | 101.     | 198.     | 4        |

10進数表記 : 219.

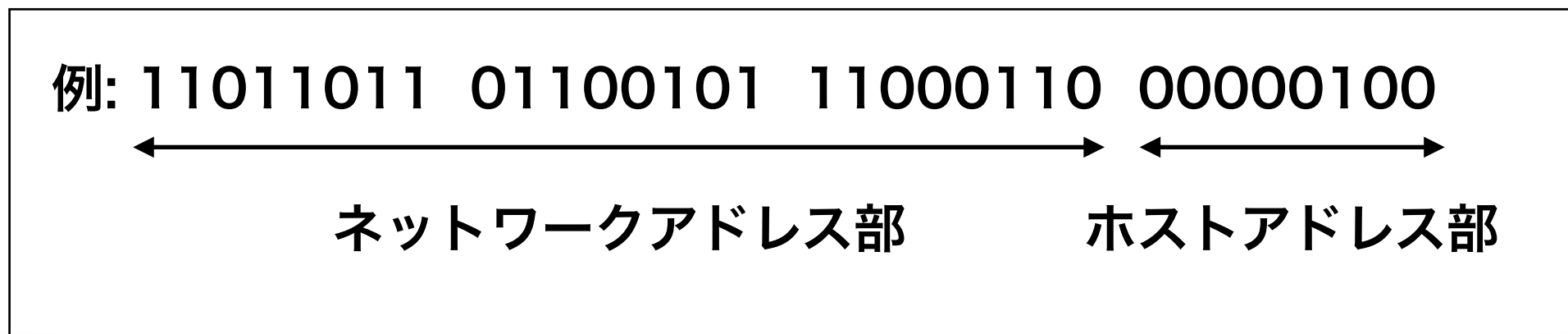
101.

198.

4

# IPアドレスの構成

- IPアドレス: ネットワークアドレス部とホストアドレス部で構成

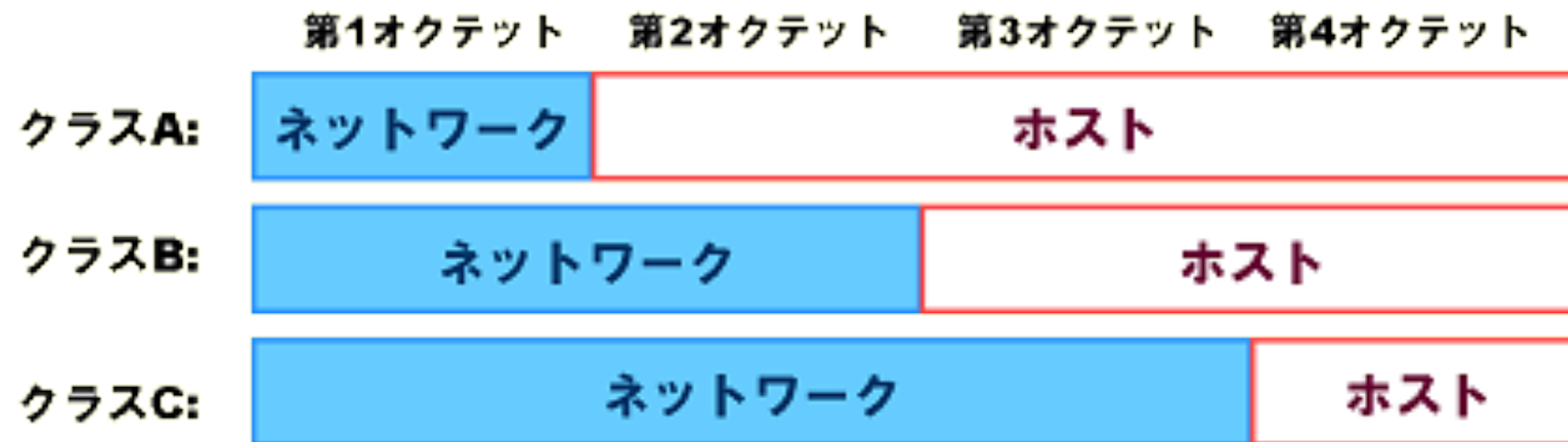


- **ネットワークアドレス部:** 企業において組織単位にネットワークがある場合, それぞれのネットワークを一意に識別するためのネットワークアドレスを表す部分
  - \* 同じネットワークに属しているノードのネットワークアドレスは同じ
- **ホストアドレス部:** 同じネットワークに属するノードを一意に識別するためのホストアドレスを表す部分

# IPアドレスクラス

- 従来では, IP IPアドレスをその先頭4ビットまでの値によって4つの種類に分けるアドレスクラスと言う概念が採用されていた
  - \* ホストアドレス部のビット数が多いほど, 各ネットワークで使用できるIPアドレスは多くなるが, 利用せず無駄になるIPアドレスが発生しやすい

## IPアドレスクラス



| クラス  | IPアドレス範囲                  | 1範囲あたりのアドレス個数 |
|------|---------------------------|---------------|
| クラスA | 0.0.0.0～127.255.255.255   | 16,777,216    |
| クラスB | 128.0.0.0～191.255.255.255 | 65,536        |
| クラスC | 192.0.0.0～223.255.255.255 | 256           |

# 特殊なIPアドレス

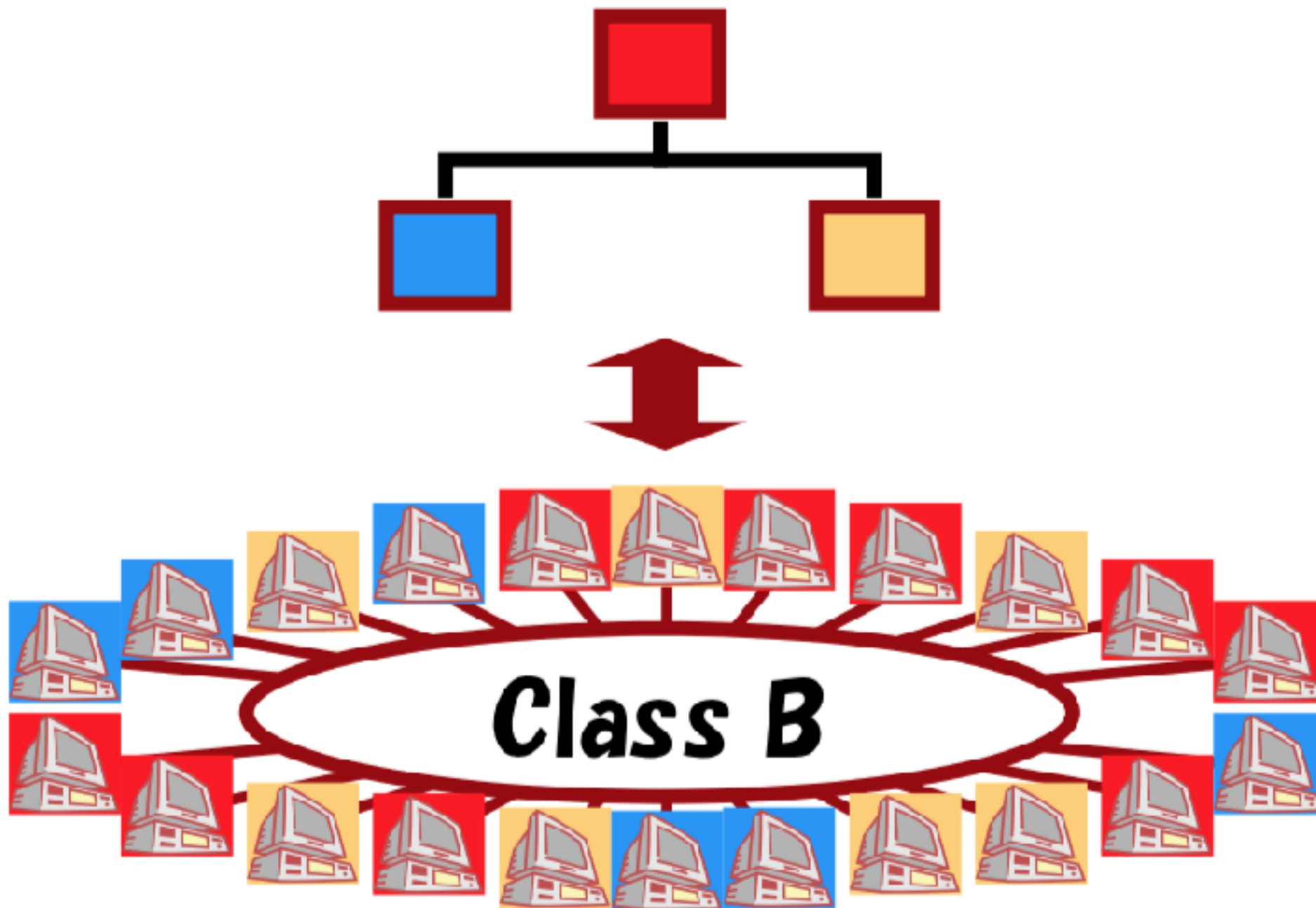
- **ネットワークアドレス**: ホストアドレス部が全て「**0**」のアドレス.
  - \* ネットワーク自体を指す
- **ブロードキャストアドレス**: ホストアドレス部が全て「**1**」のアドレス
  - \* ネットワーク内のすべてのノードあてを示す
- **ループバックアドレス**: 自分自身を表すIPアドレス
  - \* 「122.0.0.1」

# サブネットマスク

- クラスに縛られずにネットワーク部とホストアドレス部を分ける
  - \* アドレスクラスの欠点を補う目的
- 32ビットの情報
  - \* 「**1**」の部分：ネットワークアドレス部
  - \* 「**0**」の部分：ホストアドレス部
- IPアドレスのホストアドレス部のうち、数ビットをネットワークアドレス部に割り当てることができる
  - \* 同一ネットワーク内に小さなネットワークを作ることができる

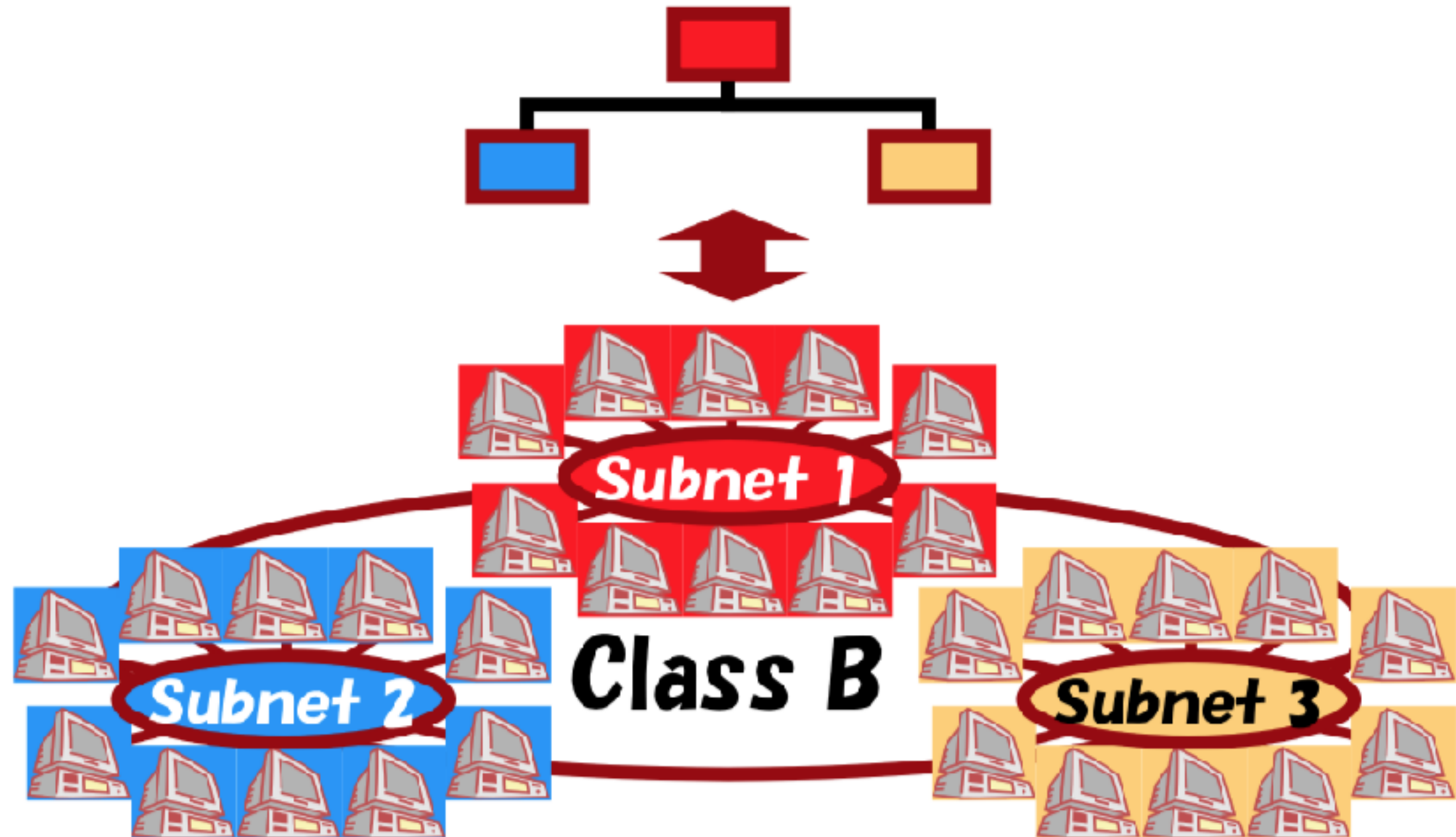
# サブネットの意義

—サブネットを使わないと...—



# サブネットの意義

—サブネットを使うと...—



# プレフィックス表記

- ネットワークアドレス部とホストアドレス部の区切りをネットワークアドレス部の桁数で示す方法

\* 例：ネットワーク部が29桁のIPアドレス

▶ 233.xxx.255.0/**29**

# CIDR



- IPアドレスの効率的な運用を促進するために考えられた方法
  - \* サブネットマスクを使用
- サブネットマスクによりネットワークアドレス部をネットワークサイズに応じた任意の長さに変更可能

# アプリケーション層の プロトコル

# アプリケーション層のプロトコル

- **HTTP**(HyperText Transfer Protocol)
  - \* WebブラウザとWebサーバとの間で, **HTML**(HyperText Markup Language)などの**コンテンツの送受信**を行うプロトコル
- **SMTP**(Simple Mail Transfer Protocol)
  - \* インターネットでは**メールを転送**するプロトコル
- **POP**(Post Office Protocol)
  - \* ユーザがメールサーバから自分のメールを取り出すときに使う. メールを**クライアントにダウンロード**する

# アプリケーション層のプロトコル

- **IMAP**(Internet Message Access Protocol)
  - \* メースサーバ上のメールにアクセスして操作するためのプロトコル. **メールをサーバ上に保存**したまま管理する
- **DNS**(Domain Name System)
  - \* インターネット上の**ホスト名・ドメイン名とIPアドレス**を対応づけて管理する. 分散データベースシステムで, ルートサーバから階層的にデータ管理する
  - \* ゾーン情報(元となるDNSレコード)を持つ**プライマリサーバ**と, その完全なコピーとなる**セカンダリサーバ**との間で**ゾーン転送**を行い, データの同期を実行する

# アプリケーション層のプロトコル

- FTP(File Transfer Protocol)
  - \* ネットワーク上で**ファイルの転送**を行うプロトコル
- **DHCP**(Dynamic Host Configuration Protocol)
  - \* コンピュータがネットワークに接続するときに必要な**IPアドレス**などの情報を**自動的に割り当てる**プロトコル
- NTP(Network Time Protocol)
  - \* ネットワーク上の危機を**正しい時刻に同期**させるためのプロトコル

# アドレス変換技術

# アドレス変換

- 近年IPv4の枯渇問題により, IPアドレスを節約するために, IPアドレスを節約して変換することが一般的になった
  - \* **プライベートIPアドレス** : 社内LANなど内部でしか利用できないアドレス
  - \* **グローバルIPアドレス** : 外部と接続するときに使用する
- RFC1918が推奨するプライベートIPアドレスの範囲は以下の通り

|        |             |   |                 |
|--------|-------------|---|-----------------|
| クラスA : | 10.0.0.0    | ~ | 10.255.255.255  |
| クラスB : | 172.16.0.0  | ~ | 172.31.255.255  |
| クラスC : | 192.168.0.0 | ~ | 192.168.255.255 |

# アドレス変換

- プライベートアドレスをグローバルアドレスに変換する仕組み
  - \* **NAT**(Network Address Translation)
  - \* **NAPT**(Network Address Port Translation)

|                    |                                                                                         |
|--------------------|-----------------------------------------------------------------------------------------|
| NAT                | グローバルIPアドレスとプライベートIPアドレスを1対1で相互に変換する. 複数のノードが同時にインターネットに接続する場合, 同じ数のグローバルIPアドレスが必要      |
| NAPT<br>(IPマスカレード) | NATの考え方にTCP/UDPポートのポート番号を組み合わせ, 1つのグローバルIPアドレスで, プライベートIPアドレスを持つ複数のノードが同時にインターネットに接続できる |

IPv4 • IPv6

# IPv4からIPv6へ

- IPv4の枯渇問題の解決：IPv6の登場
  - \* IPアドレスが32ビットが**128ビット**に
    - ▶ 十分なアドレス空間が確保されている
- IPアドレスの自動設定
  - \* DHCPサーバがなくてもIPアドレスを自動設定できる
- セキュリティの強化
  - \* **IPsec**のサポートが必須なので、セキュリティが確保され、ユーザ認証やパケット暗号化が可能になった

# IPv4 ・ IPv6

