KIT Kanazawa Institute of Technology

情報ネットワーク
第9回

工学部　情報工学科

スケジュール

	日程	学習内容	教科書記載箇所
第1回	9/30	ネットワーク概論　その1	1章
第2回	10/7	ネットワーク概論　その2	1章
第3回	10/17	TCP/IP基礎知識	2章
第4回	10/28	データリンク　その1	3章
第5回	11/5	データリンク　その2	3章
第6回	11/11	IPプロトコル	4章
第7回	11/18	IPに関連する技術	5章
第8回	11/25	小テスト@多目的ホール	
第9回	12/2	TCPとUDP　その1	6章
第10回	12/9	TCPとUDP　その2	6章
第11回	12/16	ルーティングプロトコル	7章
第12回	12/23	アプリケーションプロトコル	8章
第13回	1/8	セキュリティ+付録（物理層）	9章
第14回	1/20	達成度確認試験@多目的ホール	
第15回	1/27	総合カラーニング（予定：情報処理サービスセンター講演@5-101）	

目標

コンピュータネットワークに関連する以下の項目について体系的に説明できる。

1.情報通信の基本的な仕組みについて

2.OSI参照モデルとTCP/IPモデルについて

3.物理層、データリンク層について

4.ネットワーク層、トランスポート層ついて　⇐今回

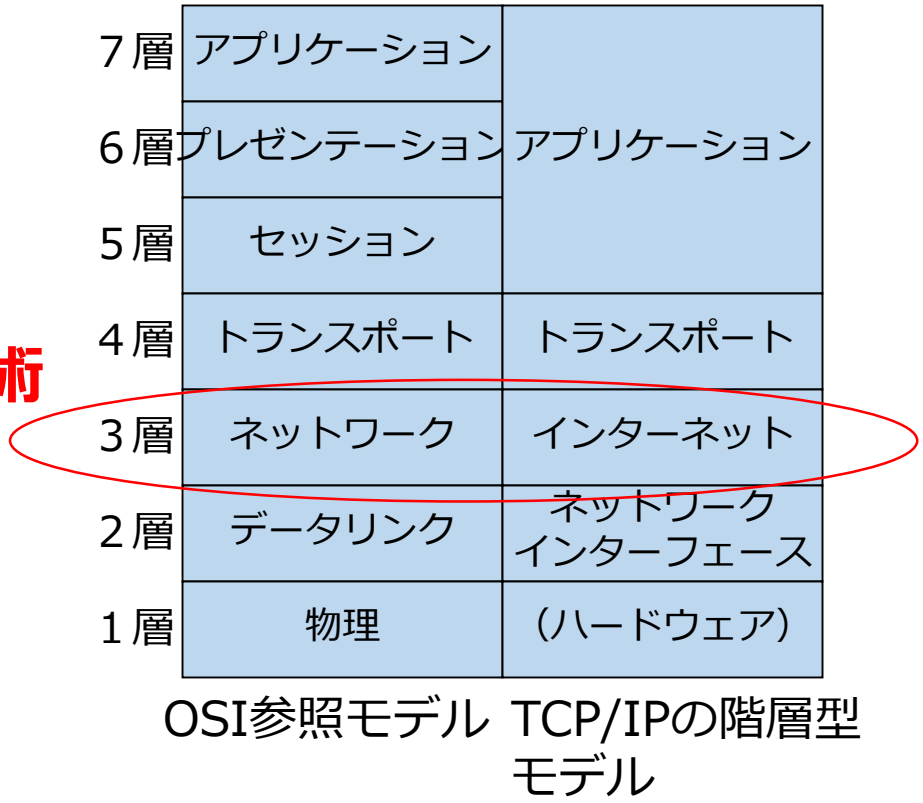
5.ルーティングプロトコル、アプリケーション層について

また、ネットワークに関する適切なコマンド、ツールを用いてネットワークの保守管理を行える。

第5章　IPに関連する技術

講義内容

5.1　DNS
5.2　ARP
5.3　ICMP
5.4　DHCP
5.5　NAT
5.6　IPトンネリング
5.7　その他のIP関連技術

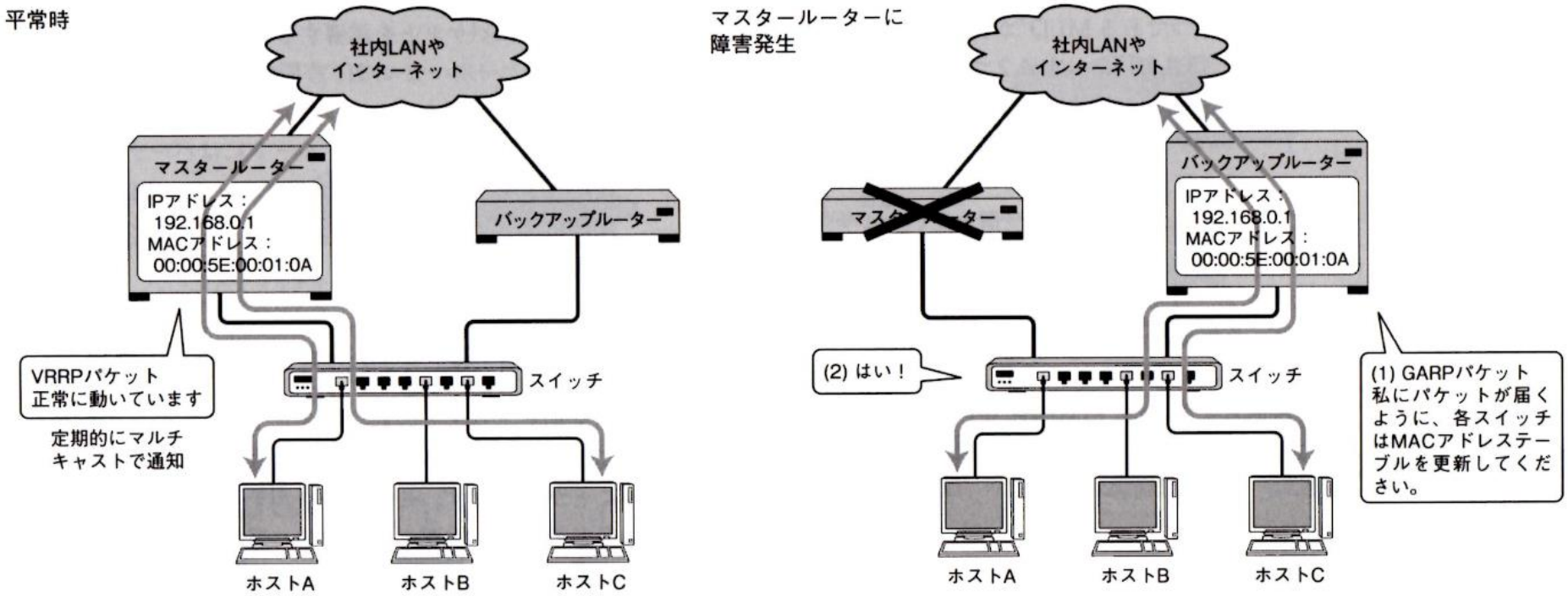


その他のIP関連技術

- VRRP（Virtual Router Redundancy Protocol）
- IPマルチキャスト関連技術
- IPエニーキャスト
- 通信品質の制御
- 明示的なふくそう制御
- Mobile IP

VRRP（Virtual Router Redundancy Protocol）

ルーターの冗長化技術
マスタールーターが故障しても、バックアップルーターに通信が切り替わる



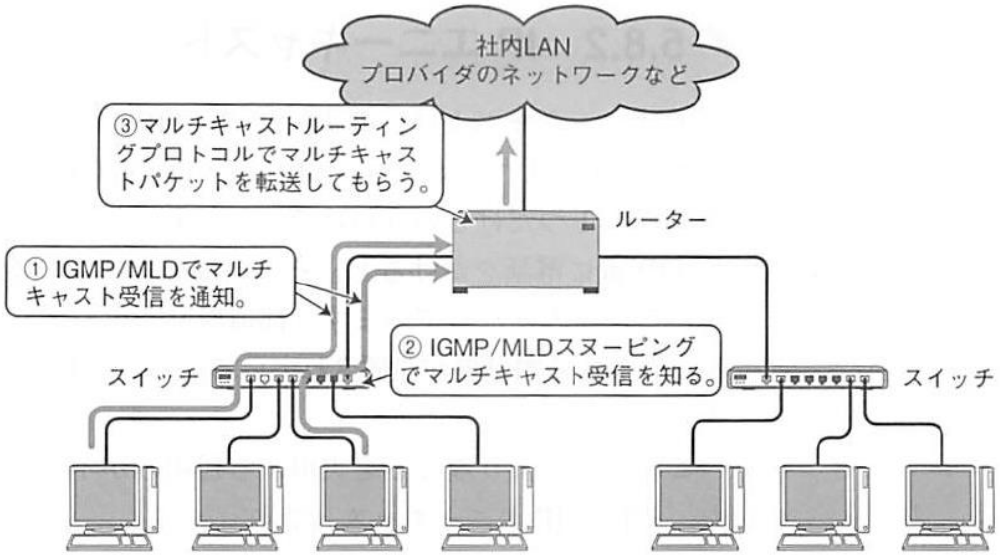
IPマルチキャスト関連技術

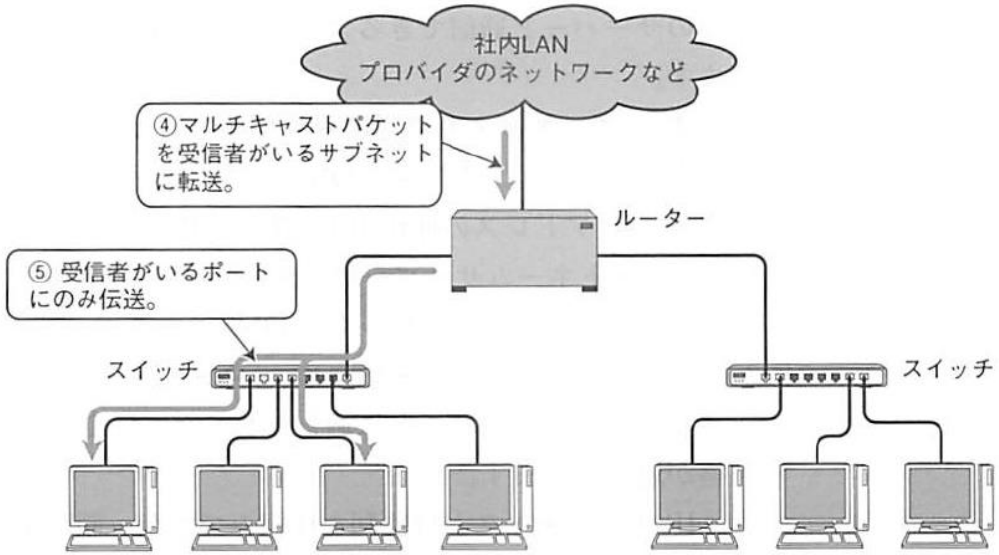
教科書　p.218

- 分散処理型のコンピュータネットワークが発展するにつれて、複数のホストへ同時にデータを送信し、効率を向上させる要求が高まっている
- マルチキャストは特定のグループに所属する全てのホストにパケットを送信するために利用される
  - ユニキャストやブロードキャストだと無駄が多い
- クラスDのIPアドレスが利用され、後ろ28ビットがマルチキャストするグループを特定する番号になる（あらかじめ用途が決まっているものもある）

クラスDのIPアドレス

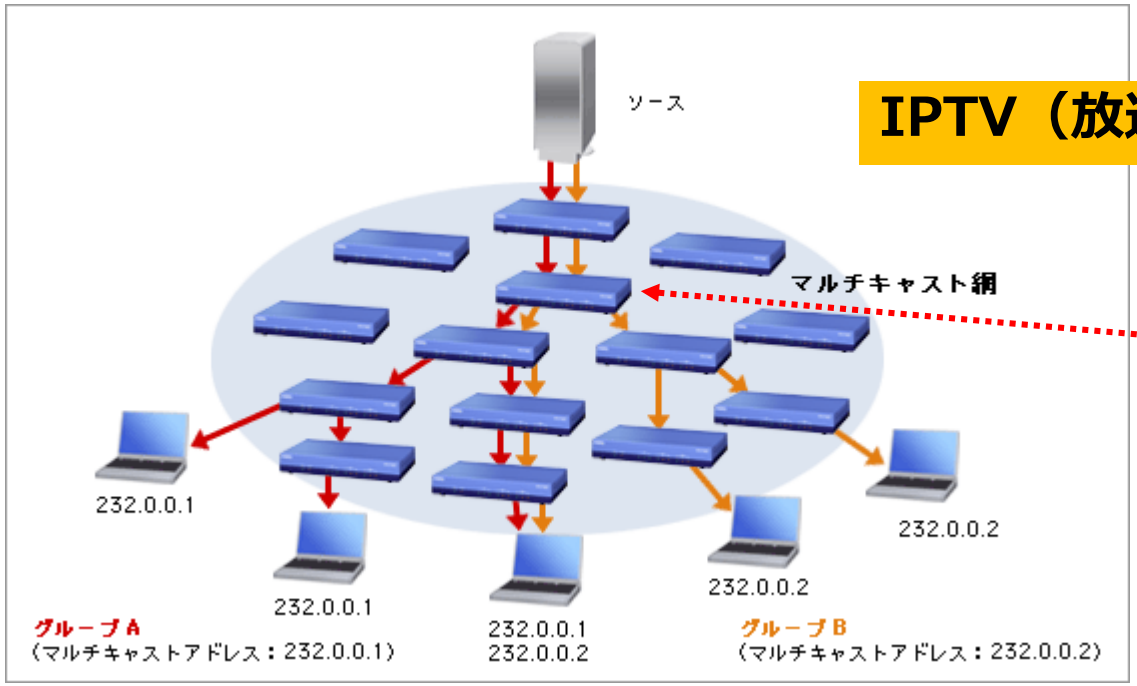
IPマルチキャスト関連技術





教科書　p.219　図5.26参照

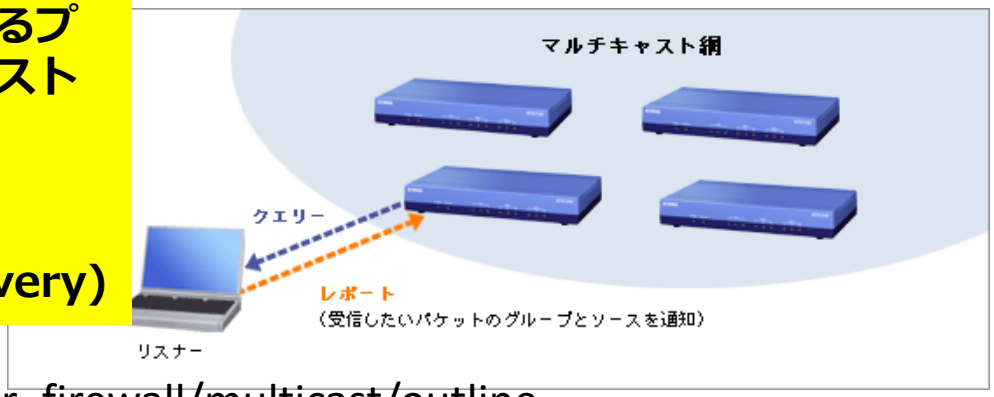
【参考】マルチキャストグループ



IPTV（放送）に使われている技術

出力ポート毎にグループA、B、両方、を選んでいる

マルチキャストグループへの参加者を管理するプロトコルを使い、リスナーがどのマルチキャストグループに属したいかをルータが把握する。
IPv4 … IGMP (Internet Group Management Protocol)
IPv6 … MLD (Multicast Listener Discovery)

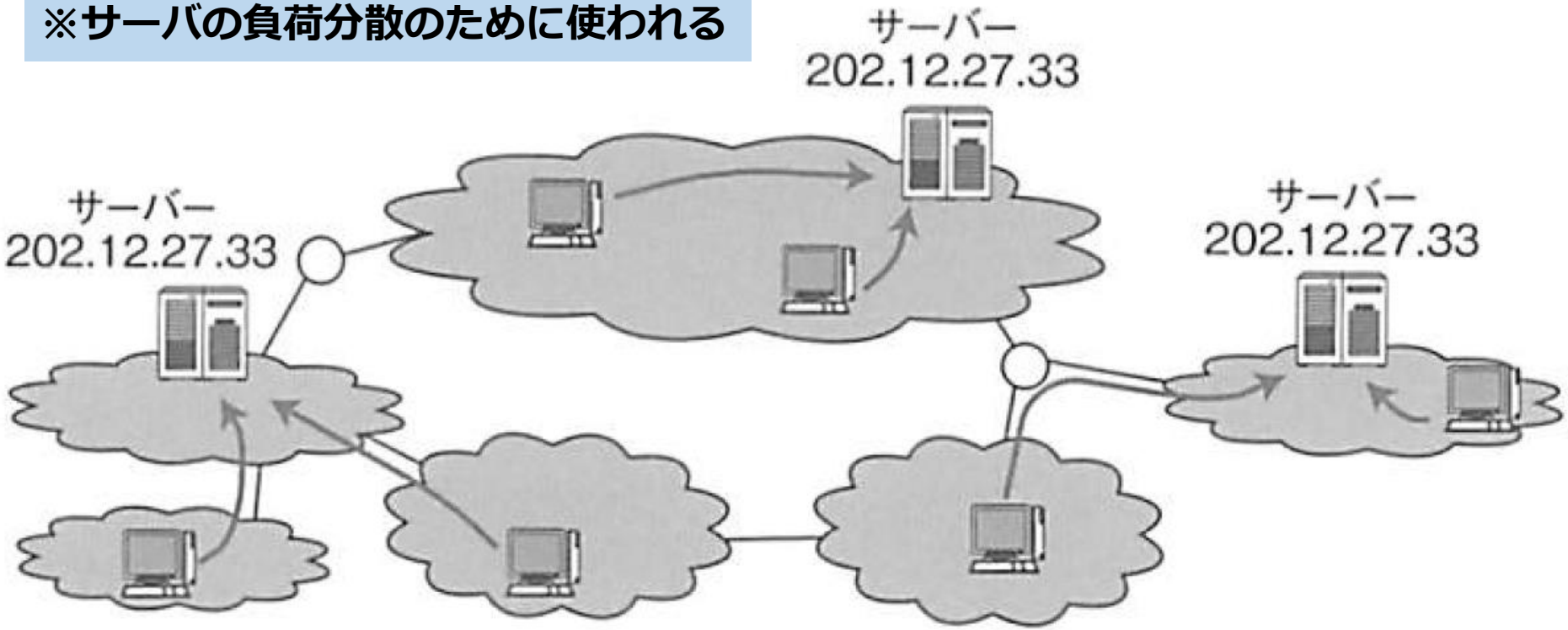


https://network.yamaha.com/setting/router_firewall/multicast/outline

IPエニーキャスト

教科書　p.220　図5.27参照

※サーバの負荷分散のために使われる



IP エニーキャストでは同じ IP アドレスが複数のサーバーに設定され、
クライアントは最寄りのサーバーからサービスを受けることができる。

通信品質の制御

教科書　p.221

通信の品質とは

- 近年IPプロトコルの実用性が認められ、様々な通信に用いられているが、IPプロトコルは「ベストエフォート」型なため、通信の品質保証はない
- また、ベストエフォート型の通信は、通信回線が混雑すると通信性能が著しく低下する→ふくそう（輻輳）
- しかしながら、VoIPなどIPを利用した音声通信などの要求が高くなり、IPを使った通信に品質（QoS:Quality of Service）の保証が求められるようになってきた

通信品質を保証する仕組み

- 品質保証は、基本的にはパケットに優先順位をつけルータなどで優先処理をすることにより実現される
  - 保証する品質以上のパケットが流れてきたときには、そのパケットは優先されずに破棄される
- この通信品質を制御する仕組みとして提案されている技術として、RSVP（Resource Reservation Protocol）を用いてきめ細かい優先制御を提供するIntServと、相対的でおおざっぱな優先制御を提供するDiffServがある

※パケットの特定の領域に優先度を示す数字をセットすることで実現している。

IntServとDiffServ

- IntServ
  - 特定のアプリケーション間の通信に対して通信品質の制御を提供する仕組み
    - ✓ RSVPというプロトコルを用いて、フローをセットアップする
  - IntServを実現するためには、個々のルータに通信品質を設定する必要があるため、大規模なネットワークでは運用が困難
- DiffServ
  - 特定のネットワーク内でおおざっぱに通信を制御するため、処理がしやすく、実用的な仕組みになっている
    - ✓ たとえば特定のプロバイダの中だけで、顧客ごとにランク付けをして、パケットに対する優先制御を行う
  - DiffServにより通信品質を制御するネットワークをDiffServドメインといい、このドメインの境界にあるルータが入ってくるパケットのヘッダのDSCP(Differentiated Services Code Point）フィールドを書き換え、優先制御を行う
  - 【参考】DiffServ QoS Model
    https://www.infraexpert.com/study/telephony7.html

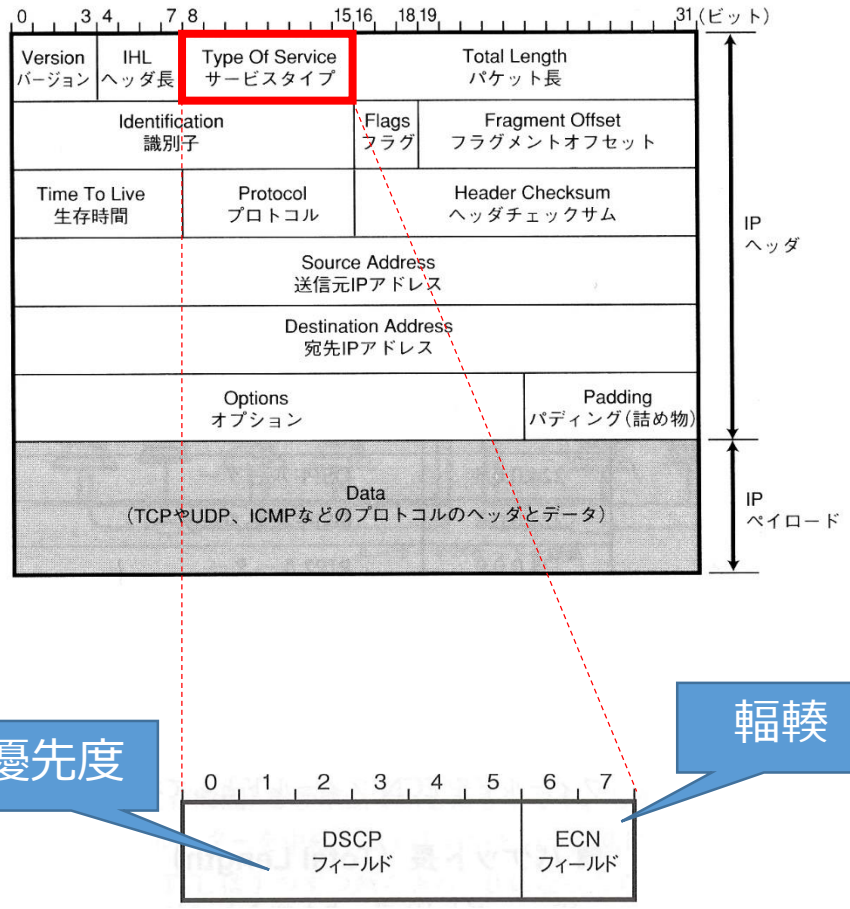
明示的なふくそう通知

教科書　p.223

- ふくそうが起きたときには、データパケットを送信しているホストは送信量を減らす必要がある
  - TCPではふくそう制御を行うが、ふくそうが発生しているかどうかをパケットの喪失で判断するため、パケットが喪失する前に、パケットの送信量を減らすことができない
- これを解決するためにIPパケットを使った明示的なふくそう通知の機能ECN（Explicit Congestion Notification）が追加された
  - ECNでは、ふくそう通知機能を実現するため、IPヘッダのTOS (Type Of Service)フィールドを置き換えて、ECNフィールドを定義し、TCPヘッダの予約ビットにCWRフラグとECEフラグというものを追加する

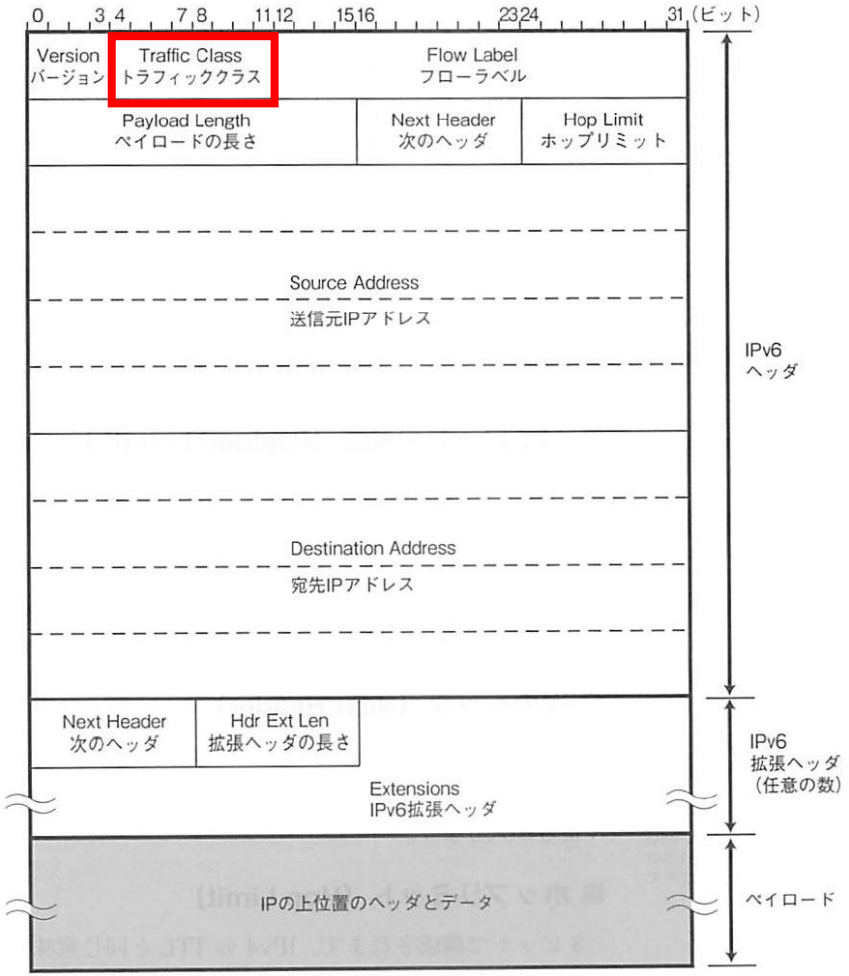
TOS (Type Of Service)フィールド

IPv4ヘッダ



教科書　p.177～参照

IPv6ヘッダ



【参考】Mobile IP

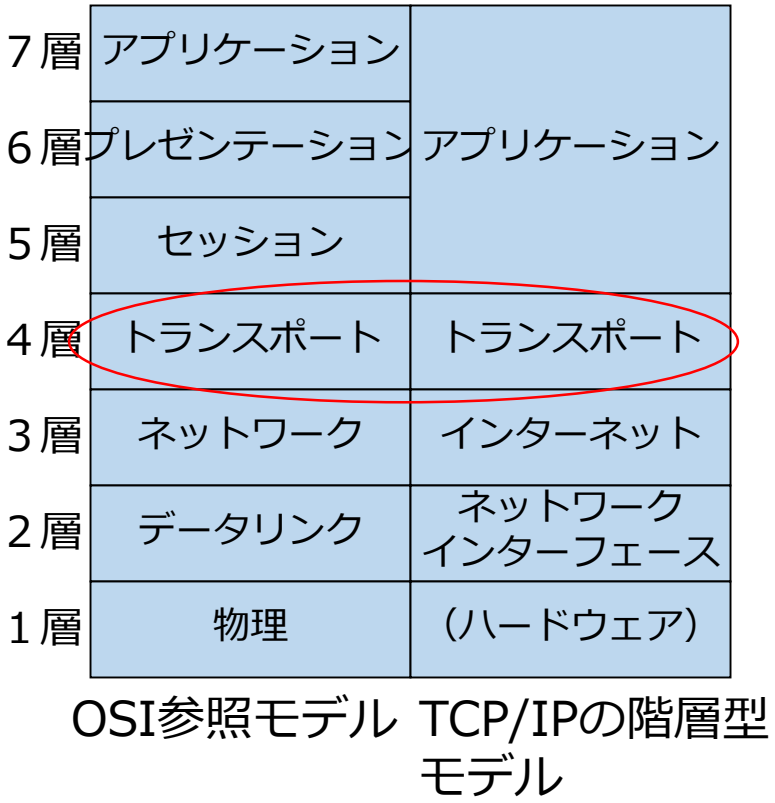
教科書　p.224

- Mobile IPとは、ホストが接続しているサブネットが変わっても、IPアドレスが変わらないようにする技術
- Mobile IPにより、今まで使っていたアプリケーションを変更することなしに、IPアドレスが変わる環境でも、通信することができる
- Mobile IPはIPトンネリングで実現される

第6章　TCPとUDP

講義内容

- 6.1　トランスポート層の役割
- 6.2　ポート番号
- 6.3　UDP
- 6.4　TCP
- 6.5　その他のトランスポートプロトコル
- 6.6　UDPのヘッダフォーマット
- 6.7　TCPのヘッダフォーマット



トランスポート層

教科書　p.228

- トランスポート層の役割
  - TCP/IPにおけるトランスポート層の役割は**下層（ネットワーク層）から受け取ったデータをどのアプリケーションに渡すかを識別すること**である
  - その識別は**ポート番号**によって行われる
- トランスポート層のプロトコル
  - **TCP**：コネクション型通信、信頼性がある
  - **UDP**：コネクションレス型通信、信頼性がなく、細かい制御はアプリケーションに任せる（高速性、リアルタイム性重視）

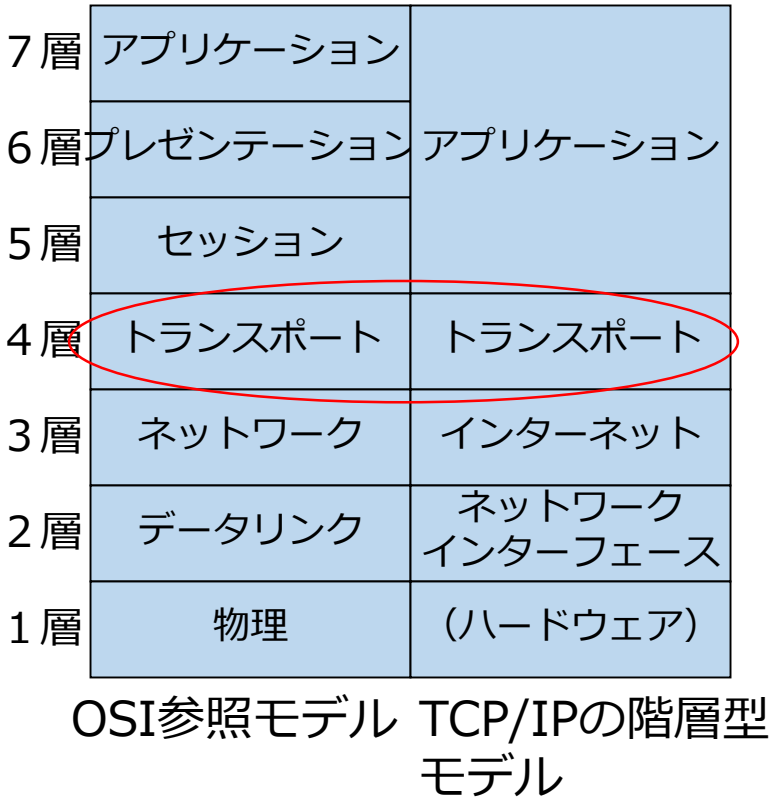
サーバー/クライアント

- TCP/IPの多くのアプリケーションプロトコルは一般にサーバー/クライアントモデルと呼ばれる形式で作られている
  - クライアント
    サーバーに対してサービスの要求を行う
  - サーバー
    あらかじめサーバープログラムが起動されており、クライアントからの要求を処理してサービスを提供する
- UNIX系OSでは、常時起動しているサーバープログラムはデーモン（Daemon）と呼ばれる
  - HTTPのhttpdやTELNETのtelnetdなど
  - とくにデーモンの代表としてクライアントからの要求を待つ、inetd（インターネットデーモン）をスーパーデーモンという
- トランスポート層はこれらのアプリケーションにデータを渡す

第6章　TCPとUDP

講義内容

- 6.1　トランスポート層の役割
- 6.2　ポート番号
- 6.3　UDP
- 6.4　TCP
- 6.5　その他のトランスポートプロトコル
- 6.6　UDPのヘッダフォーマット
- 6.7　TCPのヘッダフォーマット



ポート番号

教科書　p.232

- データリンクや、IPにはアドレスがあるが、トランスポート層でそれに対応するのがポート番号である
- ポート番号を使うことで、１台のコンピュータ上に複数のアプリケーションを識別できる
- それぞれのアプリケーションに割り当てられたアドレス（番号）ともいえる
- トランスポート層で受け取ったデータをどのアプリケーションに渡すかを決定する

  例…

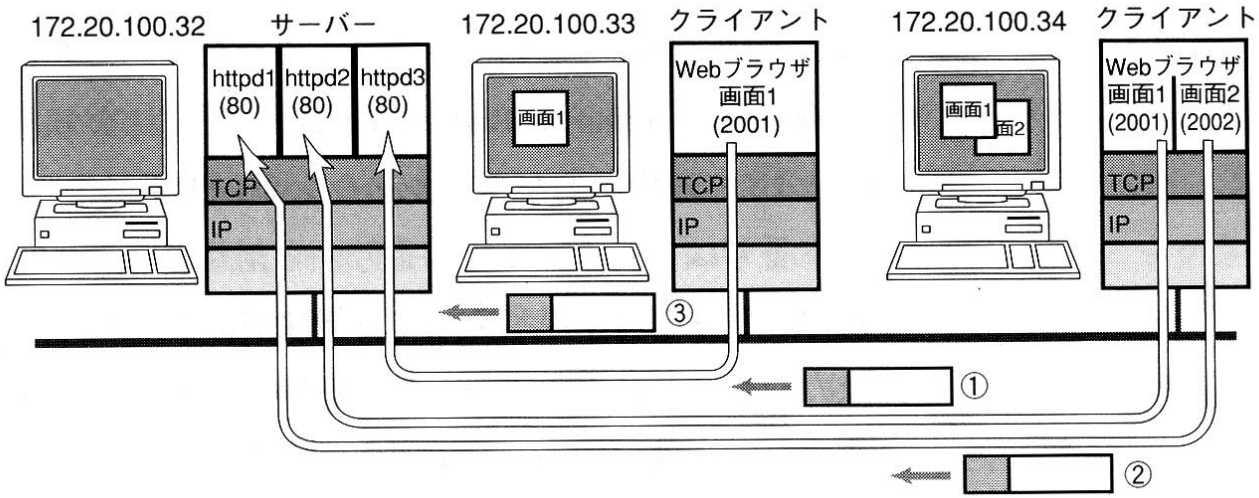
  httpd(HTTP):80番

  telnetd(TELNET):23番

  ftpd(FTP):21番

  その他（smtp, ntp, httpsなど、各種試験でよく出る）

ポート番号によるアプリケーションの識別



	IPヘッダ			TCPヘッダ		
①	送信元IPアドレス 172.20.100.34	宛先IPアドレス 172.20.100.32	TCP 6	送信元ポート番号 2001	宛先ポート番号 80	データ
②	送信元IPアドレス 172.20.100.34	宛先IPアドレス 172.20.100.32	TCP 6	送信元ポート番号 2002	宛先ポート番号 80	データ
③	送信元IPアドレス 172.20.100.33	宛先IPアドレス 172.20.100.32	TCP 6	送信元ポート番号 2001	宛先ポート番号 80	データ

送信元IPアドレス、宛先IPアドレス、プロトコル番号、送信元ポート番号、
宛先ポート番号の5つの数字で通信を識別する。

ポート番号の決め方

教科書　p.234

- 通信を行うためには各アプリケーションがどのポート番号を使うか、事前に決めておく必要がある
- 静的な割り当て
  - 一般的なアプリケーションには、あらかじめポート番号（0~1023）が割り振られている
  - ウェルノウンポートと呼ばれる（pp.236-237、表参照）

    http://www.iana.org/assignments/port-numbers

    UNIXの　/etc/services　ファイル
  - これらのポート番号の多くは、サーバーで使われる

ポート番号の決め方

- 動的な割り当て
  - クライアントなど、不定期にポートを確保する場合、オペレーティングシステムが自動で適当なポート番号を割り当てる
  - この動的割り当てにより、同じクライアントプログラムから複数のTCPコネクションを確立しても、通信を識別できる
  - 動的に割り当てるポート番号は49152~65535だが、多くのシステムではこれを無視して1024以上の使われていないポート番号が順番に使われる

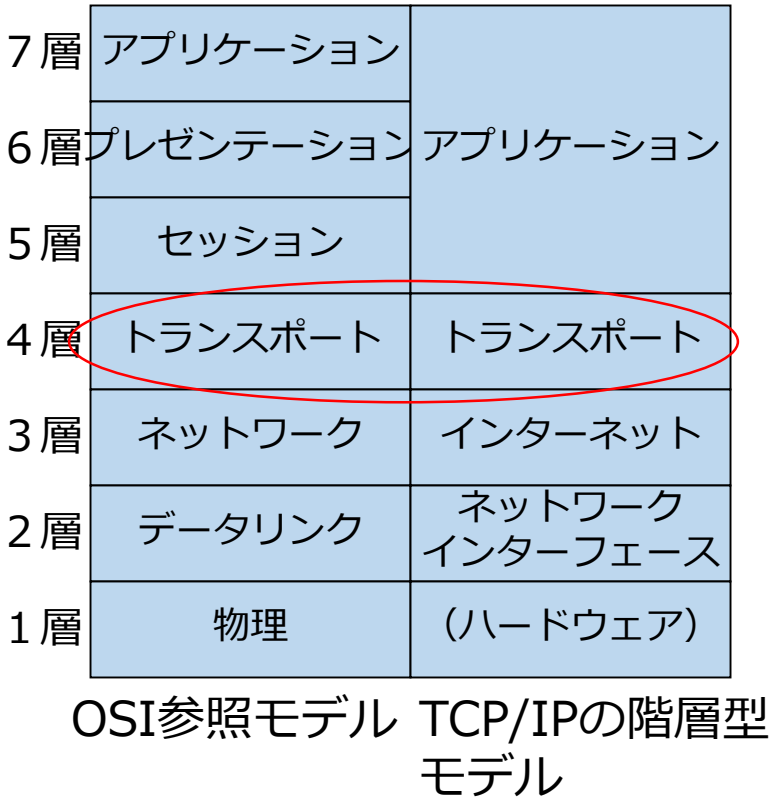
トランスポートプロトコルとポート番号

- 同一トランスポートプロトコルにおいて、ポート番号の重複は不可
  - １台の端末で、２つのアプリケーションが、同じポート番号を使用することはできない
- ただしポート番号の割り当ては、TCPとUDPで別々に行われる
  - TCPで13番を使っていても、UDPでは13番は使用することが可能

第6章　TCPとUDP

講義内容

- 6.1　トランスポート層の役割
- 6.2　ポート番号
- 6.3　UDP
- 6.4　TCP
- 6.5　その他のトランスポートプロトコル
- 6.6　UDPのヘッダフォーマット
- 6.7　TCPのヘッダフォーマット



UDP (User Datagram Protocol)

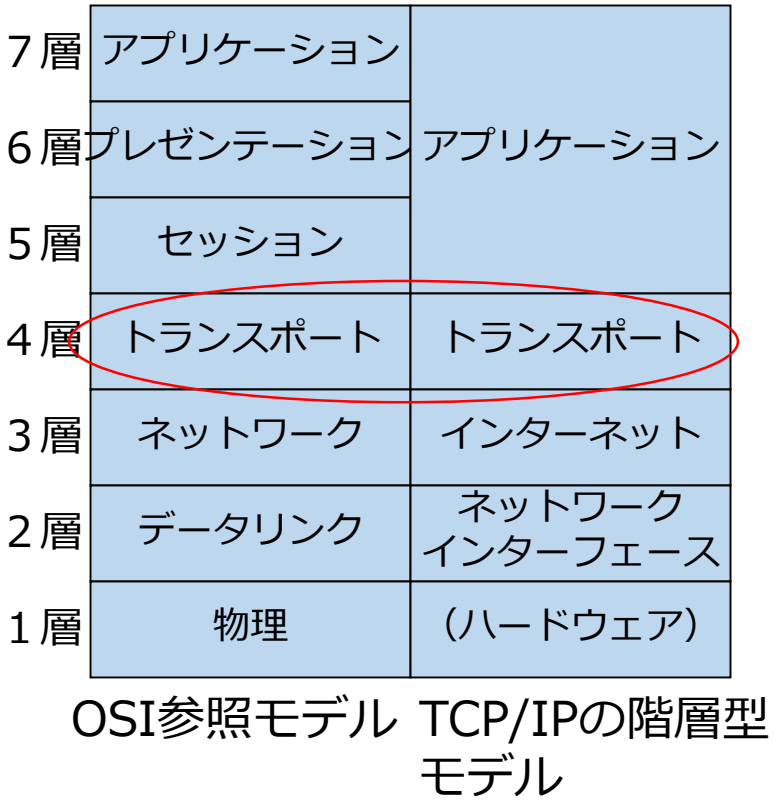
教科書　p.238

- UDPは複雑な制御は提供せず、IPを用いてのコネクションレス型通信を提供する
  - ネットワークが混雑していても、送信量の調節はしない
  - データが失われても、再送しない
  - 上記機能が必要な場合はアプリケーション側で制御する（つまりユーザーが管理する）
- UDPはこれらの性質から以下のような用途に向いている
  - 総パケット数が少ない通信
  - 動画や音声などのマルチメディア通信
  - LANなどの特定のネットワークに限定したアプリケーションの通信
  - 同報性が必要な通信（ブロードキャスト、マルチキャスト）

第6章　TCPとUDP

講義内容

- 6.1　トランスポート層の役割
- 6.2　ポート番号
- 6.3　UDP
- 6.4　TCP
- 6.5　その他のトランスポートプロトコル
- 6.6　UDPのヘッダフォーマット
- 6.7　TCPのヘッダフォーマット



TCP (Transmission Control Protocol)

教科書　p.239

- UDPは、アプリケーション側に制御をまかせ、最小限の機能を提供する
- それに対しTCPは、伝送制御の機能をもつ
  - パケットの喪失時に再送する
  - 順序が入れ替わった場合の処理
  - 相手先に届いたかどうかの確認応答
- これらの機能により、IPというコネクションレスのネットワーク上で、信頼性の高い通信を実現することができる

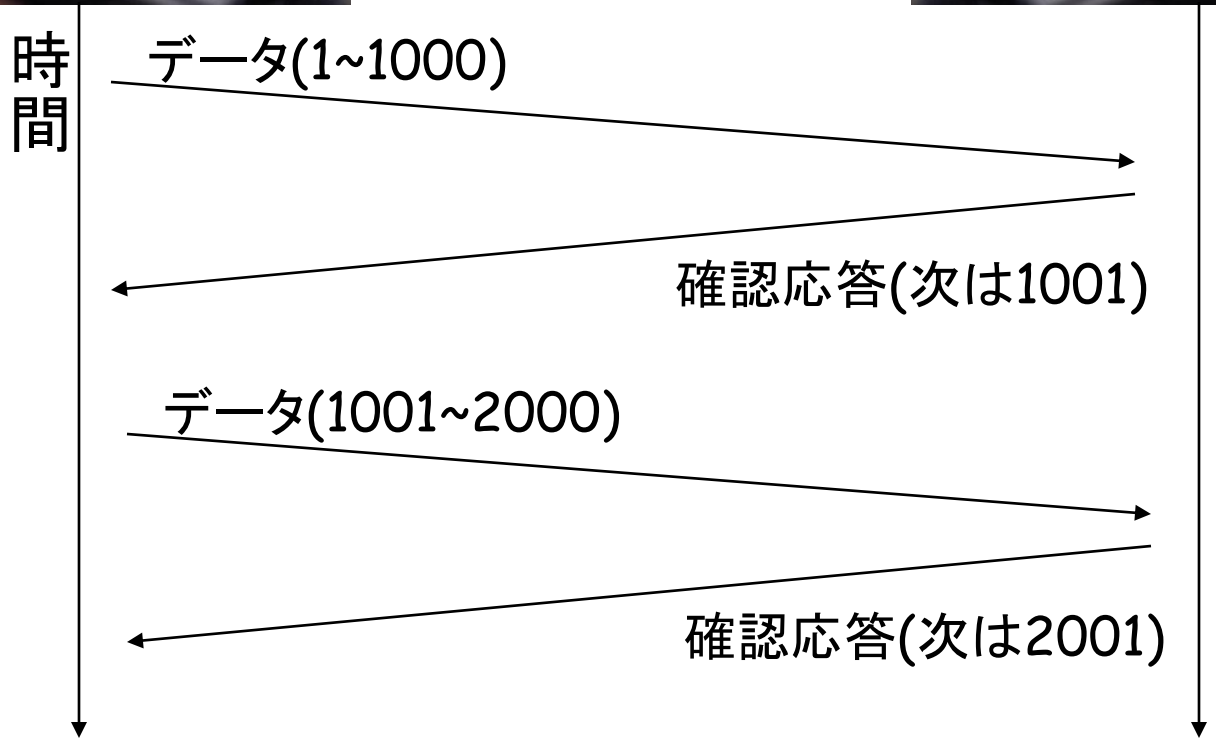
シーケンス番号と確認応答

教科書　p.241

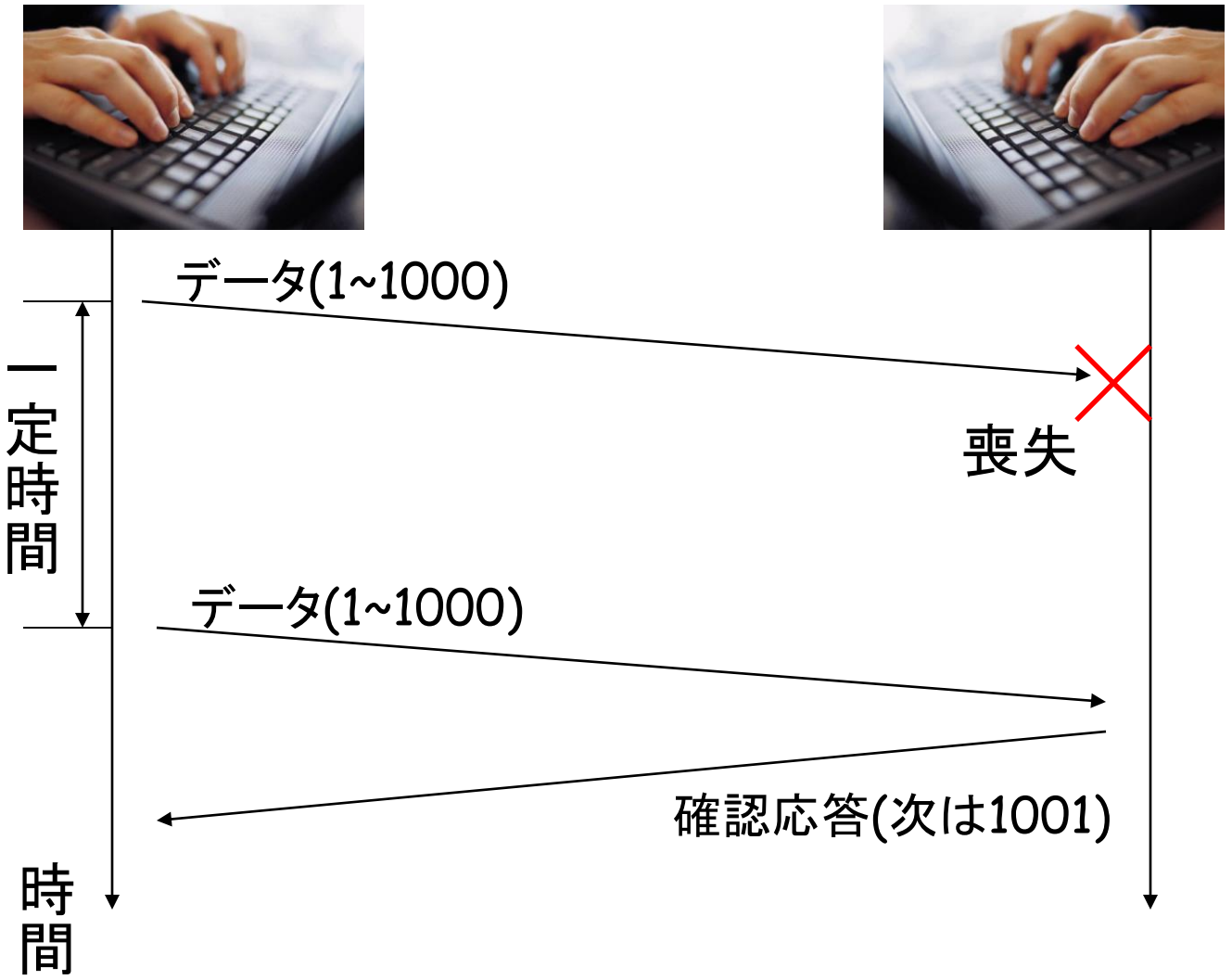
- TCPでは、送信したデータが受信ホストに到達したときACK（アック：Acknowledgementの略）と呼ばれる確認応答をする
- TCPでは、ある一定時間内に確認応答（ ACK ）が返ってこない場合には、データが喪失したと判断して再送する
- このとき、データが届いていても確認応答が返ってこない場合も再送が行われる

  →データが重複して送られてしまうことがある
- これらのデータの流れの制御やデータが重複することを防ぐために、送られるデータをオクテット単位でカウントして、送信時にヘッダに**シーケンス番号**を記載する

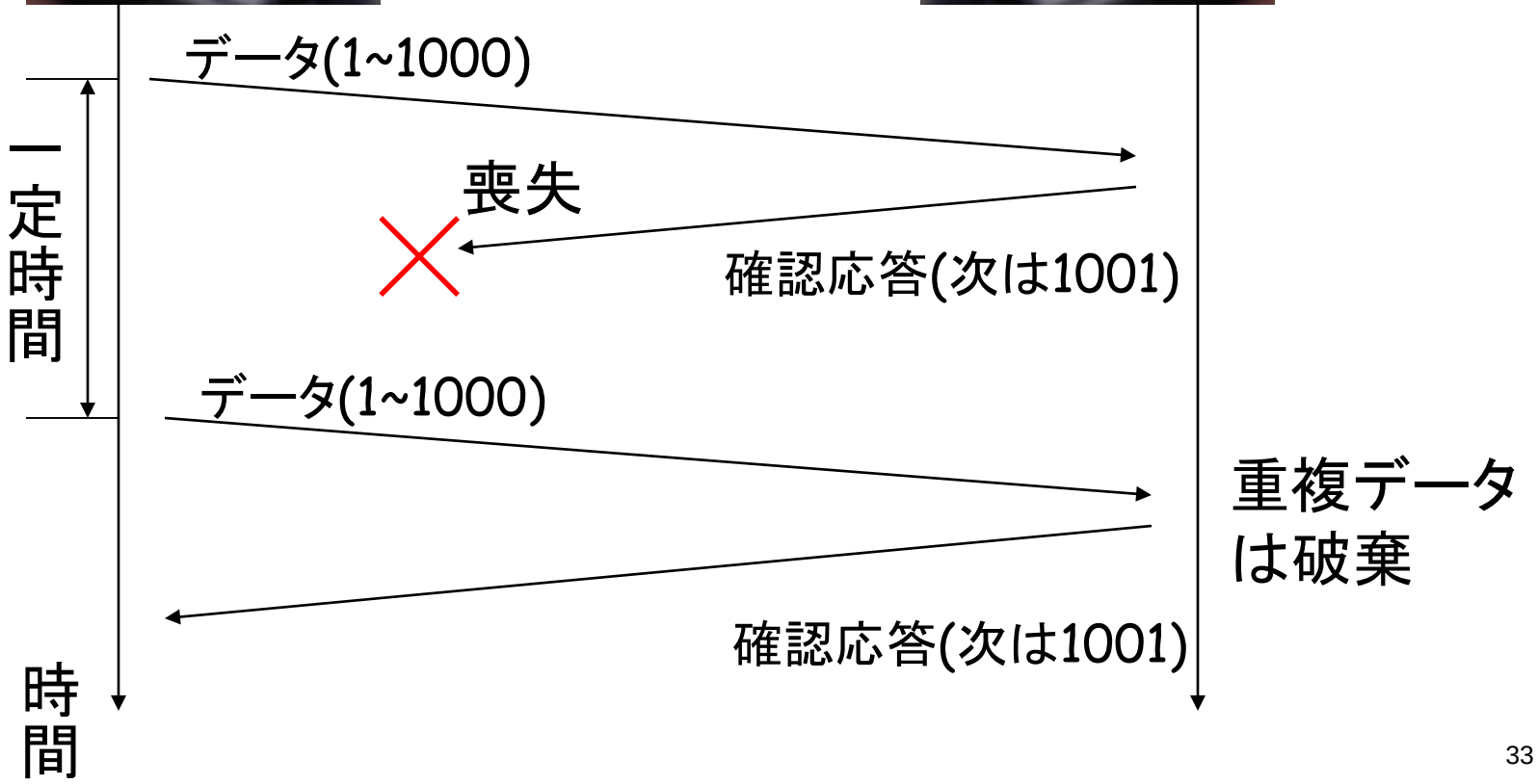
正常時のデータ送信



パケット喪失時



確認応答喪失時


データ(1~1000)
一定時間
喪失
確認応答(次は1001)
データ(1~1000)
重複データは破棄
確認応答(次は1001)
時間

再送タイムアウト

教科書　p.244

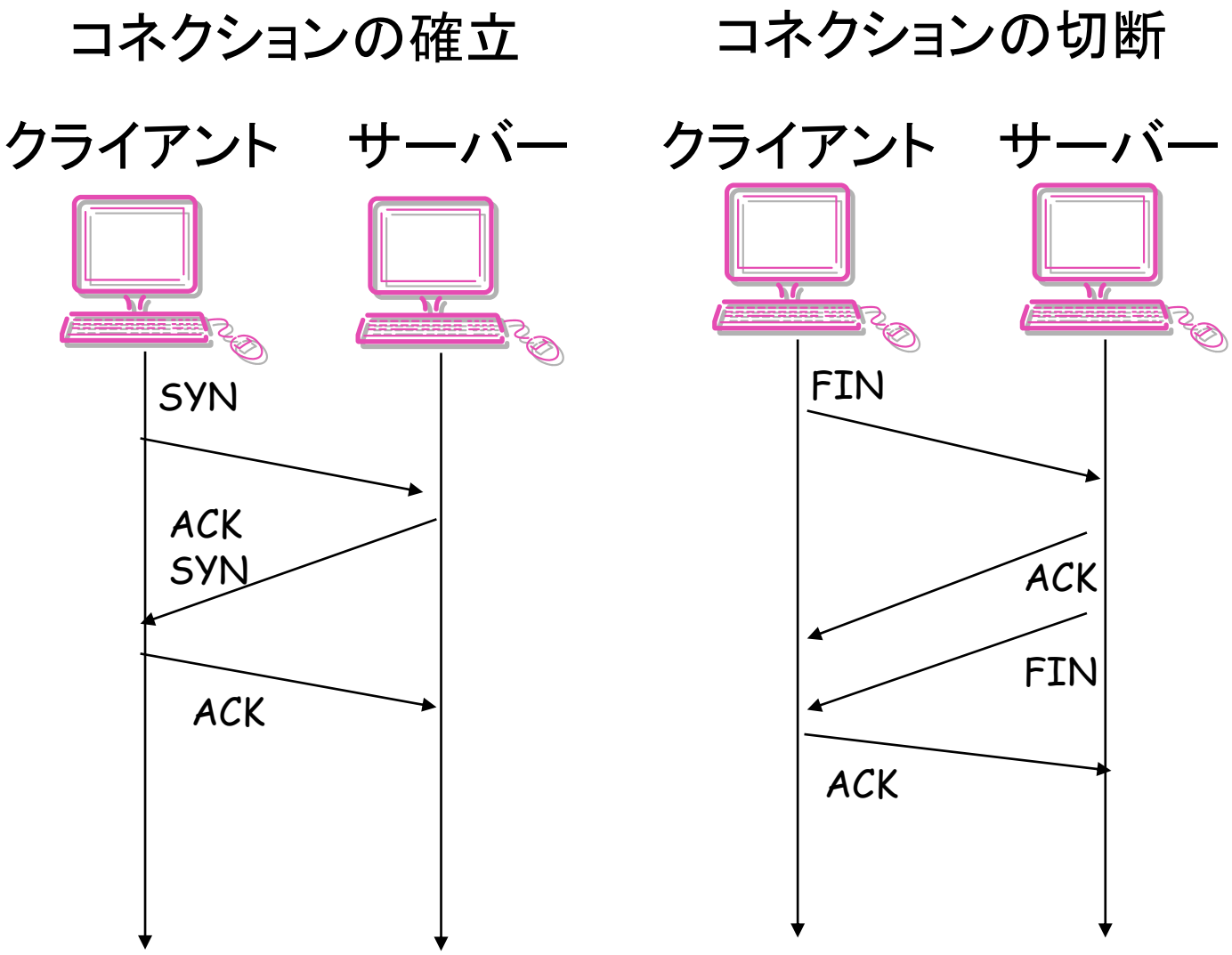
- 確認応答の到着を待つ時間
  - この時間を経過しても確認応答が到着しない場合、データの再送を行う
  - この時間を動的にどう決めればよいか？
- 時間の決定（p.244、図参照）
  - パケットを送信するたびに、時間を計測し最適値を決める（ラウンドトリップ時間＋ジッタ）
    - ✓ ラウンドトリップ時間：RTT(Round Trip Time), パケットが一往復するために必要となる時間
    - ✓ ジッタ：RTTのばらつき（分散）
  - 初期値は6秒、そのあとは0.5秒単位で制御されることが多い

コネクション管理

教科書　p.245

- コネクション型通信では、通信に先立って、相手とのコネクションを確立する
  - コネクション確立要求を送り、確認応答を待つ
  - 確認応答が送られてきたら、コネクションが確立、通信が可能となる
  - 確認応答がとれない場合は通信ができない
- TCPではコネクションを確立するときに3つのパケット（SYN、ACK&SYN 、 ACK ）をやり取りし（スリーウエイハンドシェイク）、切断するときに4つのパケット（FIN、ACK、FIN、ACK）がやり取りされる
- また、コネクションを確立時に通信を行うデータ単位、最大セグメント長（MSS: Maximum Segment Size）も決定する

TCPのコネクション確立と切断



実機演習

演習1：netstat コマンド

- ネットワークのコネクションとプロトコルに関する統計情報を表示させるコマンドを使ってみる

  cf.
  http://www.atmarkit.co.jp/fwin2k/win2ktips/234netstat/netstat.html

  または

  netstat /h

- コマンドプロンプトで、netstat –a と打ち、出力内容を記録せよ

  （-a オプション：現在待ち受け状態にあるTCPやUDPのポート一覧を表示）

  （-oオプション：ポート番号とその番号を使用するアプリのPIDを表示）

  ↑タスクマネージャと組み合わせるとそのポート番号を使うアプリが判る

netstat コマンドのオプション

- **netstat**
  現時点でのアクティブなコネクション状態を表示。即ち、TCPコネクション状態を表示
- **netstat -a**
  現在待ち受け状態にあるTCPやUDPのポート一覧を表示
- **netstat -o**
  ポート番号とその番号を使用するアプリのPIDを表示
- **netstat -b**　　（← bオプションは管理者権限で画面を起動してください）
  ポート番号とその番号を使用するアプリのプロセス名を表示
- **netstat -s**
  プロトコルごとの統計を表示
- **netstat -f**
  コネクション先のリモート（外部アドレス）のホスト名の完全修飾ドメイン名 (FQDN) ををを表示

【参考】遷移状態の説明

パッシブオープン：

TCPの接続要求を受け付ける側。いわゆるサーバ側は、このLISTEN状態でクライアントからの接続を待ち受けしている。例：Webサーバなど。

アクティブオープン：

TCPの接続要求を送信する側。LISTEN状態のTCPのポートに対して、接続を試みる側。例：Webブラウザなど。

通信確立：

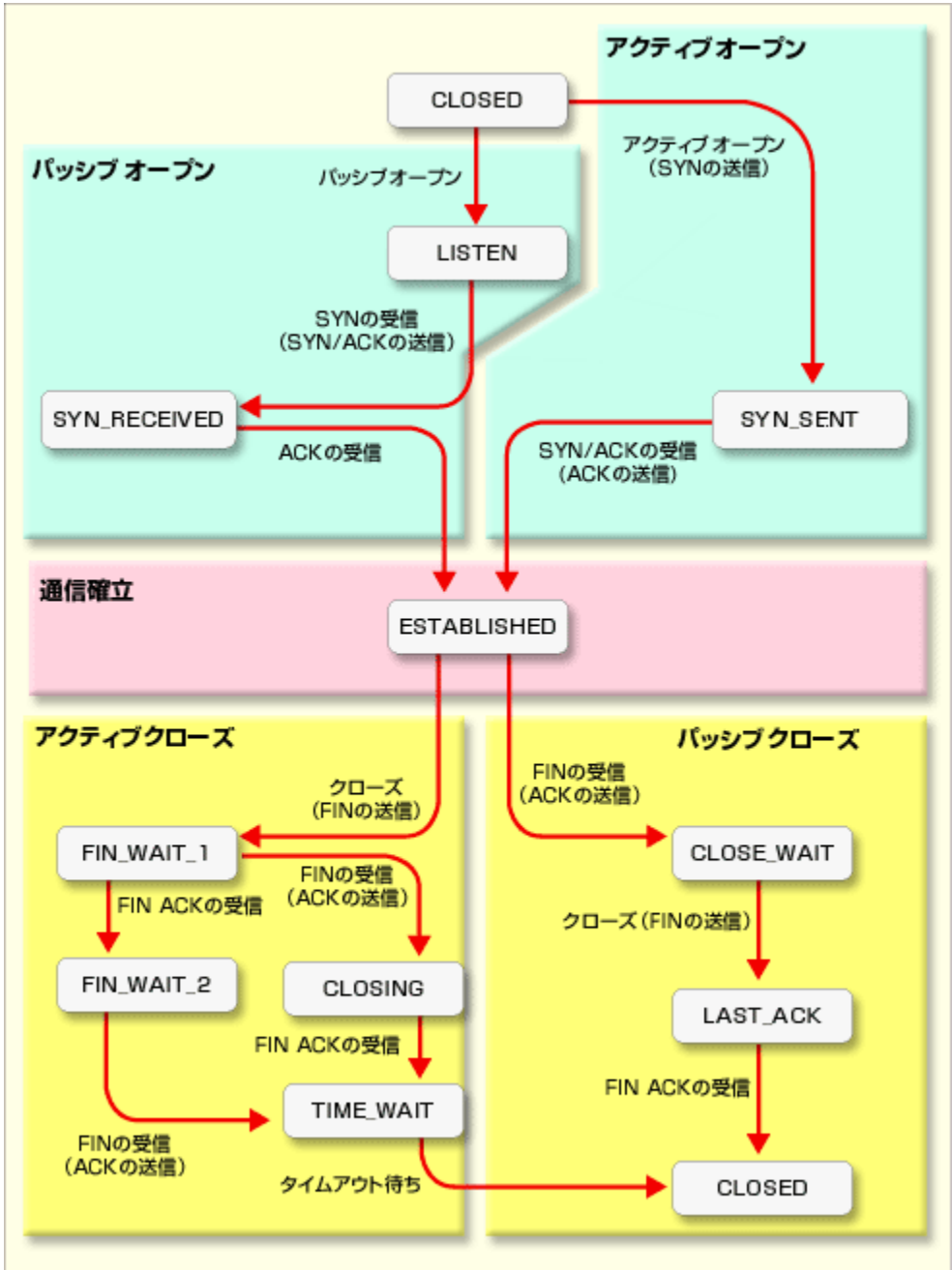
TCPの接続が確立して、通信中の状態。双方からデータを送信可能な状態（通信路がアクティブな状態）。

アクティブクローズ：

先に終了要求を送信する側。

パッシブクローズ：

相手から終了要求を受け取って、自分自身も終了処理を実行する状態。



Itmediaサイトから引用

演習

演習問題

① 通信品質制御はどのような手法によって実現されるか説明せよ

② トランスポート層の役割について説明せよ

③ サーバー/クライアントモデルとはどのようなものか説明せよ

④ ポート番号とは何かについて説明せよ

⑤ ウェルノウンポートをいくつか挙げなさい

レポート

- 実機演習の結果（画像ファイルやテキストファイルなど）と演習結果のPDFをzipで梱包して提出すること。
- 氏名を記載すること。
- 提出方法：e-シラバス経由

  ファイル名：クラス_番号_講義回数.xxx

  例：1EP7_01_09.zip
- 提出期限：12/8　23:59