目次

基礎知識

データ

・郵便物を届けるように、運びたいデータ以外の情報もまとめて一緒に送る

・データ以外の情報を**制御データ**という

・データと制御データまとめたものをデータユニットという

レイヤ

・レイヤごとに機能や役割が異なるのでレイヤごとに考える

ネットワークメディア

・PCは**LANケーブルに繋がるようにはできてない**

・ つなげる機器のこと**NIC**という

・1対1のPCを繋げてもネットワークだが複数つなげる場合はデータの流れを信号のように制御する必要がありレイヤごとに機器が存在する

基礎知識

MACアドレス

・NICごとに付与される物理アドレス

・**物理アドレスはユニーク**

・ユニークなのでデバイスを識別できるが、どこのネットワークに存在するかの情報はなし(IPアドレスとの違い)

基礎知識

IPアドレス

・**ネットワークとの接続点**ごとにつける(NICを交換しても同じ、NICが2つあればIPアドレスは2つ)

・ IPアドレスは**誰かに割り振ってもらう必要がある**(動的：DHCP / 静的：ネットワーク管理者に教えてもらい手動で設定)

・最終的に届けたい**相手のネットワーク番号＋ホスト番号で構成**

・32ビット4オクテットでクラスがある(ホスト番号0：ネットワークアドレス / ホスト番号255：ブロードキャストアドレス)

クラス	第１オクテット	10進数で表記した場合
A	0	0.x.x.x ～127.x.x.x
B	1 0	128.x.x.x ～191.x.x.x
C	1 1 0	192.x.x.x ～223.x.x.x
D	1 1 1 0	224.x.x.x ～239.x.x.x
E	1 1 1 1	240.x.x.x ～255.x.x.x

・インターネットワークに属していない**独立しているネットワークや、ネットワーク内にあるネットワーク**にあるデバイスも、ネット接続された時のことを考え、以下ルールに基づく273個の内から**プライベートIPアドレス**を設定される必要がある

※NICも以下はプライベート用に予約されているので割り振らない

第１オクテット	第２オクテット	第３オクテット	第４オクテット
10	ホスト番号		
172	16～31	ホスト番号	
192	168	0～255	ホスト番号

[Table23-01:プライベートアドレス]

基礎知識

サブネットワーク

・**ネットワークの中に小さなネットワークを定義**できる

・クラスで定義された**ホスト番号の範囲の一部**をサブネットワーク番号にする

・**サブネットの数を増やすと各サブネットに属せるホスト数は減少**する

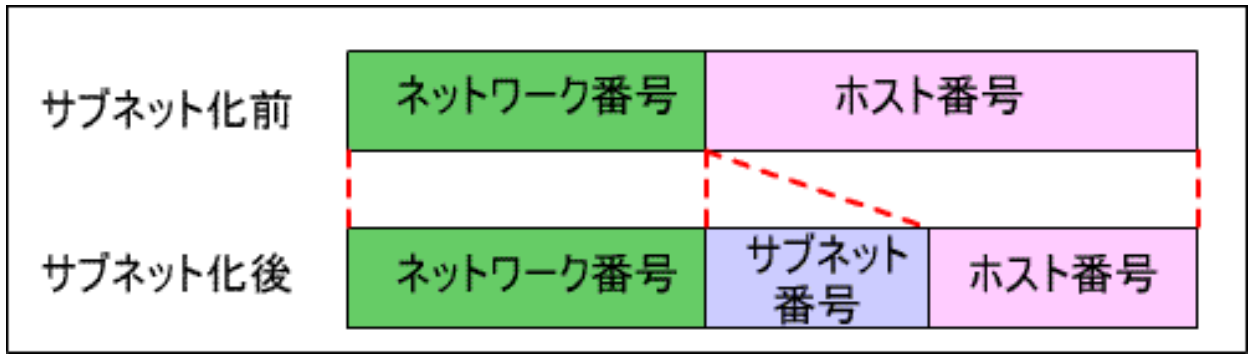
※ネットワーク番号はNICから割り振られたものなので変更不可

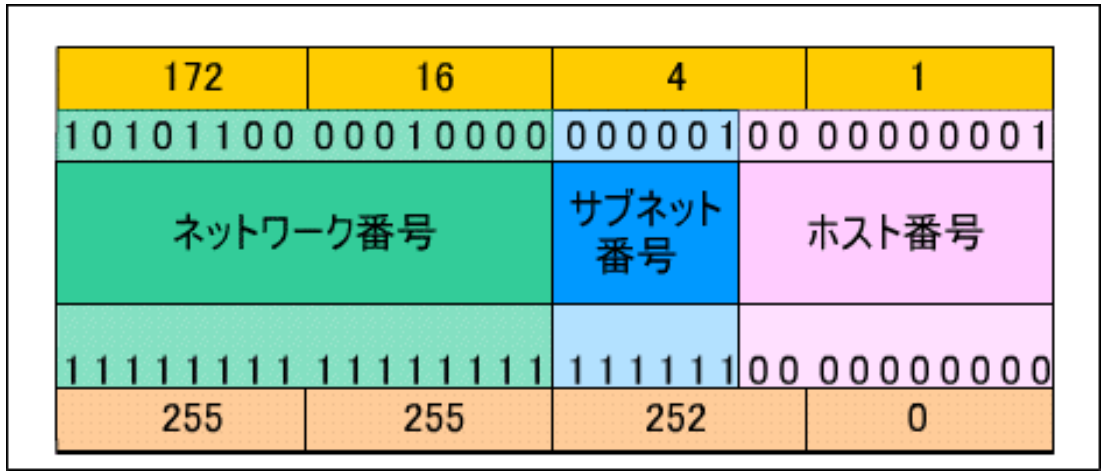
※サブネット番号は、元ホスト番号なので管理者が適当につけれる

・IPアドレス上の**どこの範囲がサブネットかを示すものをサブネットマスク**といい32ビット４オクテットで構成

・ビットが1部分の範囲がネットワーク、0部分の範囲がホスト番号

・サブネットマスク計算 (https://note.cman.jp/network/subnetmask.cgi)





基礎知識

リピータ、ハブ

・データは電気信号はあらゆる理由で減衰する

■リピータ

・繋げると**信号の強さを戻し整える**

※データの制御せず全て流す

■ハブ

・**LANの差込口が複数ある**のでLANで複数のPCに繋げる

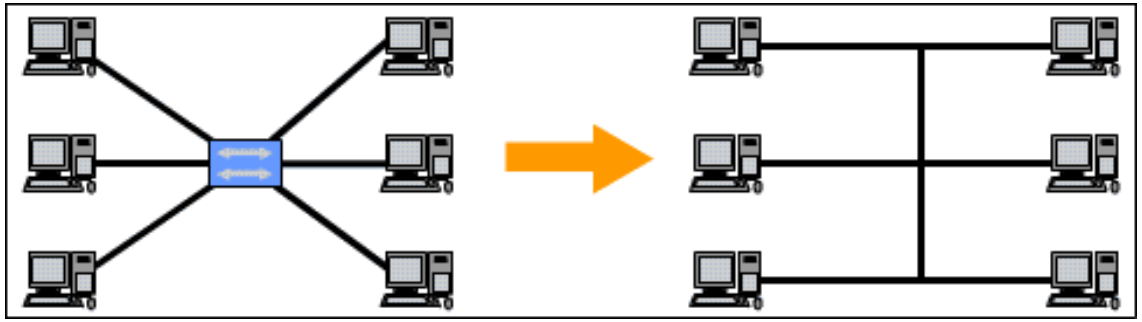
・複数PCと1対1で繋がっているのと同じなため、**ハブに届いたデータは繋がっている全てのPCに送られる**

・イーサネット規格通信は**ブロードキャスト通信**をするので、データが発信された場合**繋がっている全てのデバイスにデータが届く**

・そのため同時に**複数のPCがデータを送信した時は衝突してデータが壊れる**

※ハブはマルチポートリピータと呼ばれ信号の強さを戻し整えることもする

※ハブで繋がれているデバイスの範囲は衝突ドメイン、セグメントと呼ばれる



Q:イーサンネットの考え方あってる？

基礎知識

ブリッジ、スイッチ

イーサネット(LAN)規格のブロードキャスト通信が基本のセグメント(衝突ドメイン)を複数繋げる

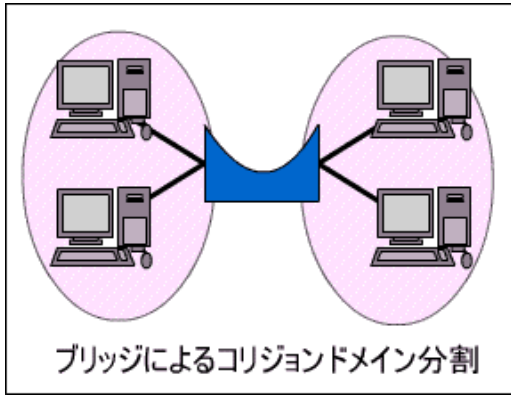
※衝突ドメインを区切りトラフィックが減り通信効率を上げられる

■ブリッジ

・2つのポートがあり、**ポートごとにMACテーブルを持つ**

・受信パケットの**MACアドレスを確認し一致するMACアドレスが存在するセグメントが接続されているポートにデータを送る**

※一致するMACアドレスがない場合、または送信先MACアドレスが同一ネットワークないの場合はパケット破棄

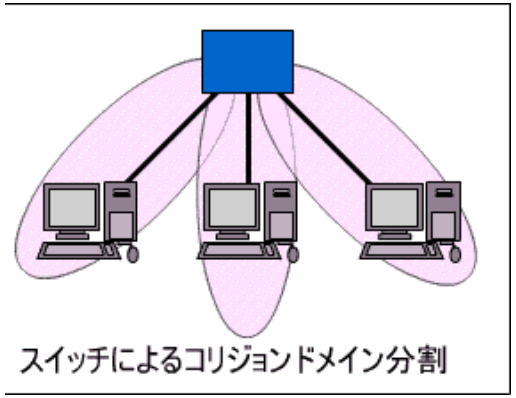


■スイッチ

・**3つ以上のポート**があり、**ポートごとにMACアドレステーブルを持つ(ネットワークごとにMACアドレステーブルを持つ)**

・受信パケットの**MACアドレスを確認し一致するMACアドレスが存在するセグメントが接続されているポートにデータを送る**

※一致するMACアドレスがない場合、または送信先MACアドレスが同一ネットワークないの場合はパケット破棄



ネットワークに入るとき

自分のIPアドレス取得 (DHCPサーバとの通信)

■DHCPサーバとの通信

・設定された**プール**(割り振り可能なIPアドレスの範囲)からIPアドレスを自動的にクライアントに割り振る機能

※このIPは振ってはいけないという設定も可

・リース期限も決める必要がある

・**オプション**を設定できる(サブネットマスクなど)

・設定手順

DHCPDISCOVER：(**ク**)サーバを見つけるための**ブロードキャスト送信**

DHCPOFFER：(サ)クライアントへ候補アドレスを送信

DHCPACK：(サ)使ってOKを送信‥OKの場合

DHCPNAK：(サ)使ってNGを送信‥NGの場合

クライアントは要求していたIPアドレスを設定

・DHCPからふられたIPアドレスのリース期限確認方法

/var/db/dhcpclient/leases/ 以下にあるファイルに記載されている

ネットワークに入るとき

デフォルトゲートウェイのIPアドレス取得(DHCPサーバとの通信)

■デフォルトゲートウェイ

・**異なるネットワークへパケットを送る際のネットワークの出入口**のデバイスのIPアドレス

・通常は、所属するネットワークの**ルータのポートに設定されているIPアドレスを指す**

(同一ネットワークの1つのデバイスという扱い)

※デフォルトゲートウェイのIPアドレスは、外部ネットワーク通信時のインターネット層でARP要求でMACアドレス取得に使われる

・デフォルトゲートウェイのIPアドレス確認

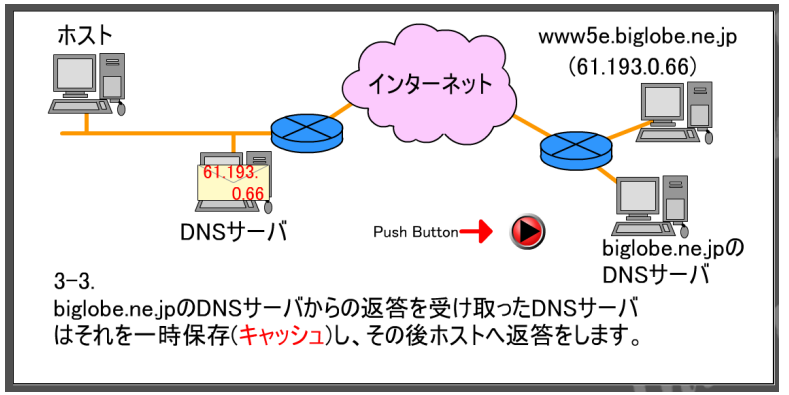
ip route

データを送受信するとき アプリケーション層

アプリケーション層の処理

※概要：詳細は３分間ネットワーキングの後半学習後

※現状はブラウザからサイトにアクセス→DNSで名前解決→HTTPリクエストの流れを想定



■ホスト名＋ドメイン名

・www.example.jp＝ホスト名.ドメイン名＝コンピュータ名.組織名＝組織の管理者管理.NIC管理

※並びは逆だがIPアドレスと構成は同じ

・ドメイン名(組織名)はユニーク

・上記のようなホスト名＋ドメイン名の組み合わせを**FQDN（Fully Qualified Domain Name）**という

■DNSサーバ

・**ドメイン登録業者のDNSサーバを利用する / レンタルサーバ業者のDNSサーバを利用する**(by参照URL)

・DNSサーバは**各組織に1つずつ**存在し、**自分の組織のドメイン名とホスト名のみ管理**している(by3分間)

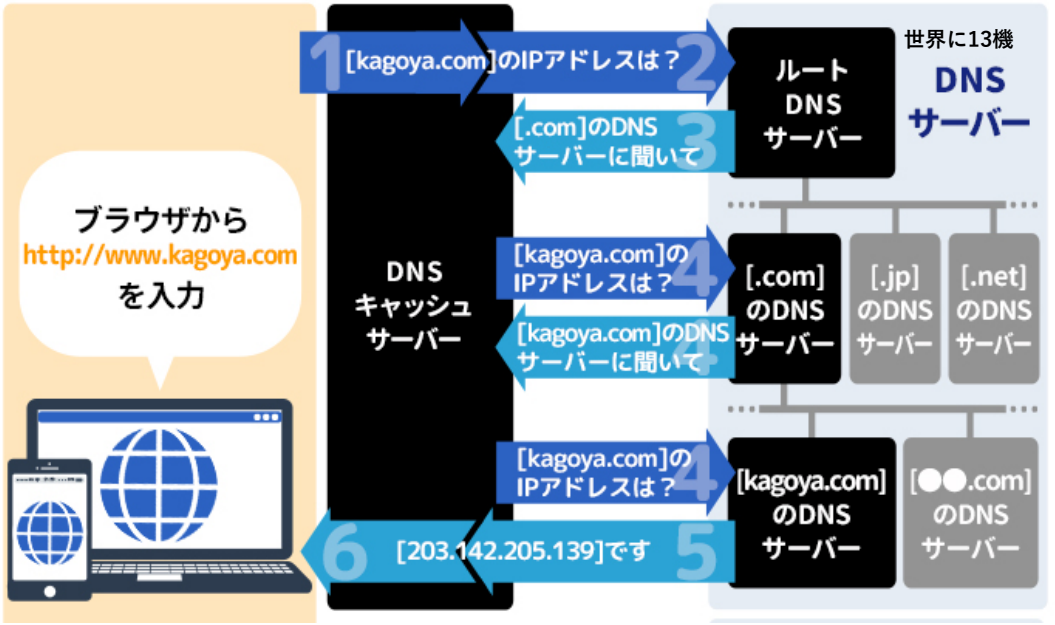
・異なるネットワークのホスト名.ドメイン名のリクエストを受けた場合は、

インターネット経由でその組織のDNSサーバにきき、IPアドレスをレスポンスしてもらう

・クライアントに問い合わせ先のIPアドレスをレスポンスする前に次回に備えて期限付きでキャッシュする

※DNSサーバは世界中のホスト名.ドメイン名を管理する分散型データベース

(キャッシュ蓄積し、ルートDNSサーバの代わりに返答する)



https://www.kagoya.jp/howto/rentalserver/dns-server/

```
ドメイン名からIPアドレスを確認する
nslookup google.com

Server:         103.5.140.1
Address:        103.5.140.1#53

Non-authoritative answer:
Name:   google.com
Address: 172.217.27.78
```

← このドメイン名のDNSレコードガ登録されているDNSサーバのIPアドレスとポート番号

← Non-authoritative answer　は、回答がキャッシュDNSサーバからきたということ
※そもそもDNSを指定しない限り、ルートDNSから回答が来ることはあるのか？

← このドメインのIPアドレス

Q:「DNSサーバは世界中のホスト名.ドメイン名を管理する分散型データベース」どういう意味？
Q:上の図のDNSキャッシュサーバーと、自分のドメインを登録したDNSは同じ？
Q:問い合わせるDNSキャッシュサーバーはどう決まるの？

データを送受信するとき トランスポート層 (TCP)

トランスポート層の処理

TCP概要編

■通信を確実、正確に行う役割の層

・**事前にデータの仮想的な通信経路を確保**し確実にデータを届ける

・アプリケーション間の通信を制御(**エンドツーエンド**と言われデータの最終的なやりとり)

※インターネット層: ネットワーク間の通信を制御 (送りっぱなし)

※ネットワークインタフェース層: (ネットワーク内の)デバイス間の通信を制御 (送りっぱなし)

・アプリケーションの通信状態

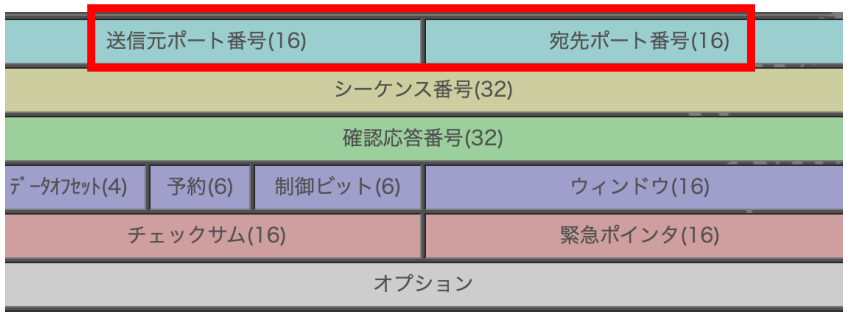
netstat -an

データを送受信するとき　トランスポート層 (TCP)

トランスポート層の処理

ポート番号編(ウェルノンポート)

TCPヘッダをつけてセグメントへとカプセル化



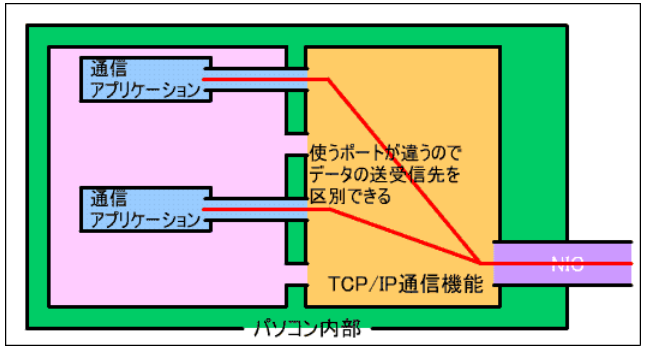
・ポートとは通信データを流すための架空の差込口 (「開いている」「閉じている」という表現)

・ 各アプリケーションは16ビット(65,536個)の値の中から1つを選んでデータの送受信口とする (**どのアプリケーションかを特定する番号**)

※それぞれ0から番号が振られる

・厳密にいうとアプリケーション別というよりは**プロセス別**であるため、例えば**ブラウザのタブごとに別々のポート番号を持っている**

・アプリケーションがポートと接続して通信を行う仕組みを**ソケット**という (**OSが持つソケットライブラリという機能**)



┗送信元ポート番号

・1024以上でランダムな番号が入る

┗宛先ポート番号

・リクエストしたいサービスのウェルノウンポートを設定

■ウェルノウンポート

・よく使われる**サービスごとに事前に決められているポート番号**

・ 0~1023番の番号

・サービスを提供するサーバ側で、提供するサービスに合わせて設定しておく必要がある

※サーバ側が提供サービスのポート番号を該当のウェルノウンポートに設定していない＝**そのサービスを提供していない**

ポート番号	アプリケーション
20・21	FTP
23	telnet
25	SMTP
53	DNS
67・68	DHCP
80	HTTP
110	POP3
161・162	SNMP

代表的なウェルノンポート

データを送受信するとき　トランスポート層 (TCP)

トランスポート層の処理

制御ビット編(SYN/ACK/FIN)

TCPヘッダをつけてセグメントへとカプセル化

・いきなりデータを送るのではなく、事前に宛先と通信できることを確認すること

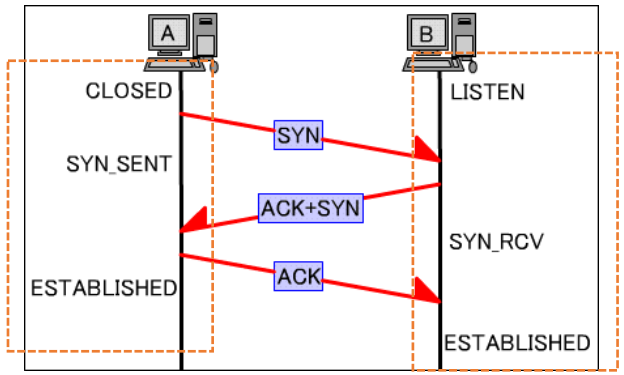
■コネクションの確立

・3回のデータ転送で双方向の通信路を確保(**スリーウェイハンドシェイク**)

・1回目 **SYN**　　: 転送許可要求 (A→B)

・2回目 **ACK+SYN**: 転送許可＋転送許可要求 (B→A)

・3回目 **ACK**　　:転送許可 (A→B)



TCPのコネクションの状態についての備考

・CLOSED: 相手からの通信を受け取らない状態

・LISTEN: 待ち状態

・ESTABLISHED: 双方がこの状態になるとコネクション確立

■コネクションの切断

・切断する際も双方の合意が必要！(自分の転送終わっても相手からの転送が終わったとは限らないから)

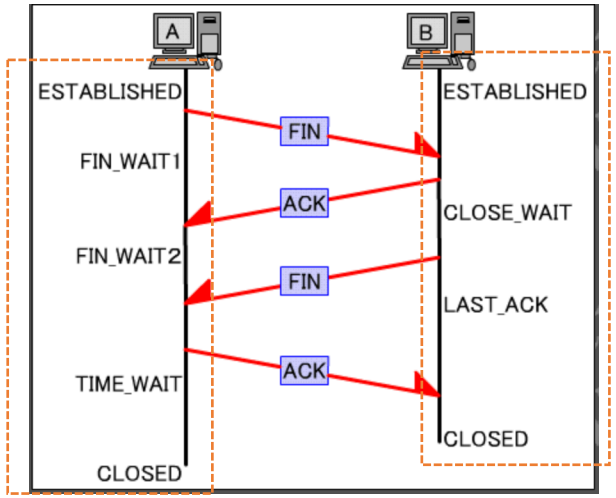
・1回目 **FIN** ： 終了要求 「もう終わらせよう」(A→B)

・2回目 **ACK** : 終了許可「いいよ」 (B→A)

・3回目 **FIN** ： 終了要求 「僕からももう終わらせたいけど、どうかな？」(B→A)

・4回目 **ACK** : 終了許可「いいよ」 (A→B)

※確立時と異なり、AからBへのFINに対するBの回答ACKと、BからAに対するBのFIN要求は別々に行われる(**1回多い**)



TCPのコネクションの状態についての備考

先にFIN要求した側視点(A視点)

・FIN_WAIT1: 相手からのACK待ち状態

・FIN_WAIT2: **相手からのFIN待ち状態**

・TIME_WAIT: 相手からにACKが届いたか確認待ち状態

データを送受信するとき トランスポート層 (TCP)

トランスポート層の処理

シーケンス番号/MSS最大値/確認応答番号編①

・TCP/IPでは長いデータを分割して送る (送るデータの呼名: **セグメント** / 送るデータのサイズ: **MSS**)

【コネクション確立中：シーケンス番号/確認応答番号/MSS最大値の決定】

・**スリーウェイハンドシェイク実行時にやり取りされるTCPヘッダ内の情報のため、コネクション確立完了と同時に決定**される

┗シーケンス番号

・SYN時：シーケンス番号に**ランダムな値を設定** (※ +1されて確認応答番号として返ってくる)

・ACK＋SYN時：受け取った**確認番号＋1**をシーケンス番号に設定

・ACK時：送る → **Aが設定したシーケンス番号の初期値＋1の値値 をシーケンス番号に設定し送信** で決定

┗MSS最大値

・SYN時：MSS値の**最大値を設定**

・ACK＋SYN時：受け取ったMSS値値を確認→ **双方の 送れる and 受取れる バイト数** で決定

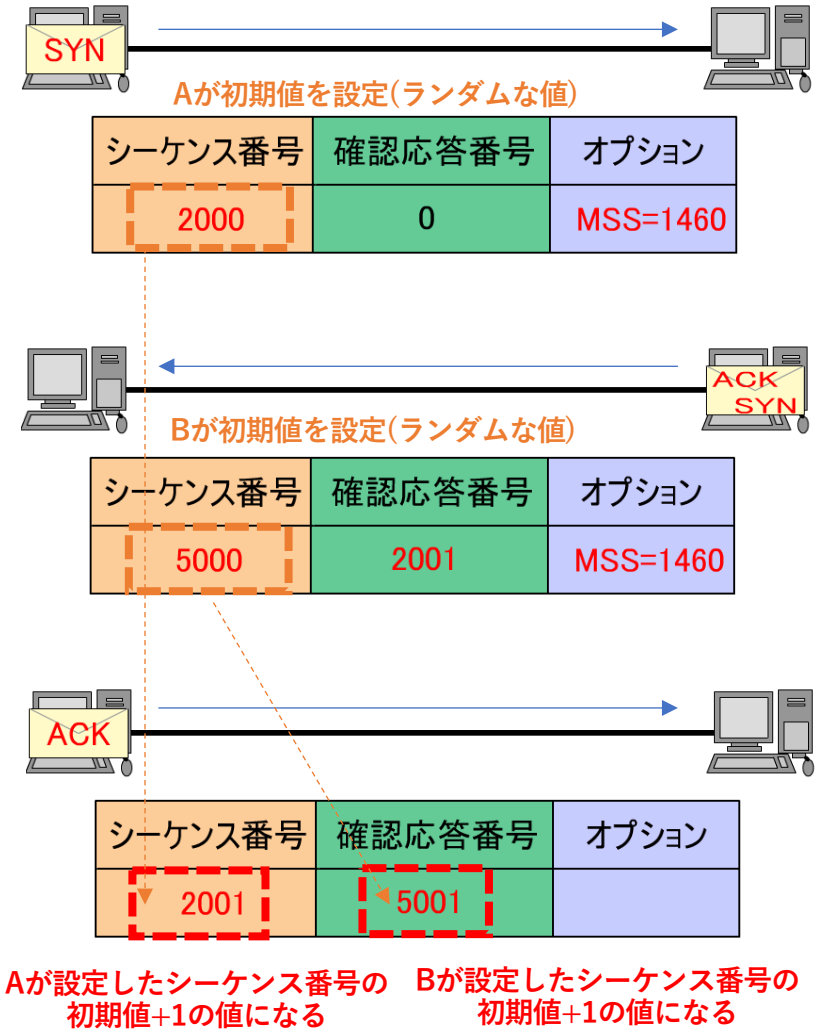
┗確認応答番号

・SYN時：0を設定

・ACK＋SYN時：**シーケンス番号にランダムな値が入ってくる**

・ACK時：送る → **Bが設定したシーケンス番号の初期値＋1の値 を確認応答番号に設定し送信** で決定

TCPヘッダをつけてセグメントへとカプセル化



データを送受信するとき　トランスポート層 (TCP)

トランスポート層の処理

シーケンス番号/MSS最大値/確認応答番号編②

TCPヘッダをつけてセグメントへとカプセル化

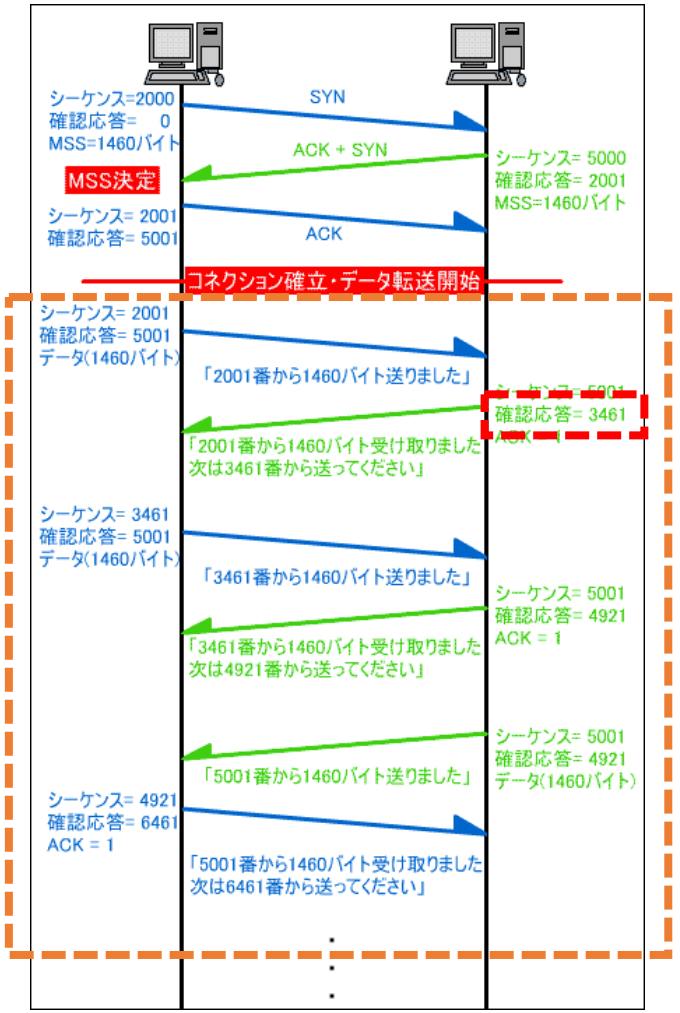
【コネクション確立後の通信中：シーケンス番号と確認応答番号の更新】

<u>┗A視点: シーケンス番号</u>

- **(A→B) これから送るデータの先頭バイト番号**
- **Bから受け取った確認応答番号の＋1の番号で、** AがTCPヘッダのシーケンス番号を**更新**

<u>┗ B視点: 確認応答番号</u>

- **(B→A) 次に送って欲しいデータの先頭バイト番号**
- **Aから受け取ったシーケンス番号の＋MSS番号で、** BがTCPヘッダの確認応答番号を**更新**



確認応答番号＝

受け取ったシーケンス番号＋MSS最大値

「受け取りました。次はこのバイト番号から送ってね。」

※ 受け取り確認(フロー制御)

※ 次回受け取るシーケンス番号

memo(2020/09/27)

ウィンドウ制御、輻輳制御は割愛

データを送受信するとき トランスポート層 (UDP)

トランスポート層の処理

UDPの概要編(TCPとの比較)

【比較/特徴】

高速

・**コネクションを確立しないコネクションレス**という通信方式のため確認応答の時間がない

・**UDPヘッダは8バイトしかなく(TCPヘッダの)4割程度のサイズ**のため通信時に**小さいデータ**で済む

※TCPは20バイト

ブロードキャスト向き

・コネクション確立をしないので、**複数のデバイスに効率よく同時(もしくはリアルタイム)にデータを送れる**

※１対多の通信時に全てのデバイスとコネクション確立と確認応答をすると送受信量が多くなりメモリの消費が激しい

信頼性が低い

・コネクションレスでのためデータを送りつけるだけで、届く保証も届いた返答ない

UDPヘッダをつけてセグメントへとカプセル化

送信元ポート番号(16)	宛先ポート番号(16)
セグメントサイズ(16)	チェックサム(16)

【UDP向きのサービス】

・高速: 高速性やリアルタイムなやりとりが必要なアプリケーション

・ブロードキャストが必要なアプリケーション

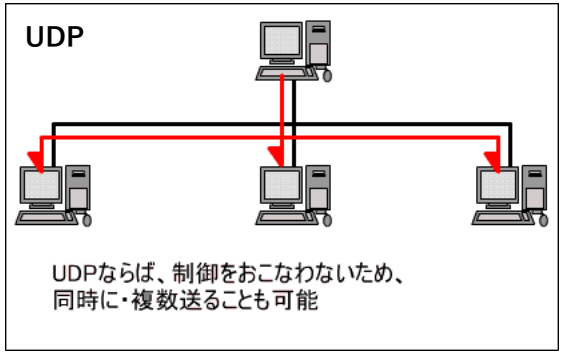
・信頼性の必要のないアプリケーション

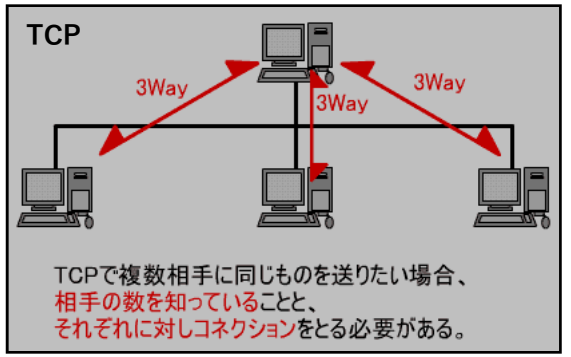
例)

・動画ストリーミングサービス

・DNS

※頻繁に使われる ＆ パケットサイズが小さいので速度重視

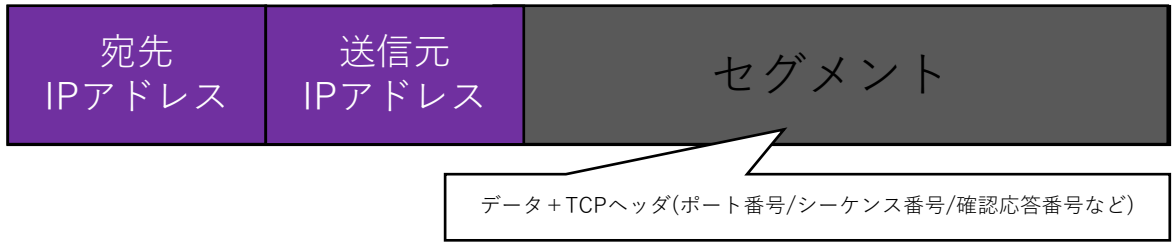




データを送受信するとき　インターネット層

インターネット層の処理

セグメント → パケットへとカプセル化



【宛先MACアドレス取得(ARPプロトコル)】

■ARPテーブル

・宛先IPアドレスの取得が完了した時点で実行され、**IPアドレスとMACアドレスの対応表であるARPテーブル**を確認する

・static: 手動でエントリを設定/dynamic: 動的にエントリを設定

┗ARPテーブルに該当のIPアドレスのエントリがある場合

そのまま

┗ARPテーブルに該当のIPアドレスのエントリがない場合

ARP要求が実行

・「arp -a」でARPテーブル表示

・「arp –s (IPアドレス) (macアドレス)」でARPテーブルへの追加

・「arp –d (IPアドレス) 」でARPテーブルから削除

■ARP要求(この処理で宛先MACアドレスが確定)

・宛先IPアドレスに対応する宛先MACアドレスを取得する作業

┗宛先IPアドレスのネットワーク番号が同一ネットワークの場合

上位レイヤの情報を必要としないレイヤ3独自の**ARPパケットをブロードキャスト**で通信をし、宛先デバイスからレスポンスとして、

そのデバイスのMACアドレスを受け取り、ARPテーブルのエントリとして追加

※同一ネットワーク内で通信する際は、都度行う必要がある。

┗**宛先IPアドレスのネットワーク番号が異なる**ネットワークの場合

直接デフォルトゲートウェイにARPし、レスポンスとして、**デフォルトゲートウェイのMACアドレス**を受け取り、ARPテーブルのエントリとして追加

※一般的にはデフォルトゲートウェイ＝ルータであることがほとんどである。

参考記事: https://ascii.jp/elem/000/000/629/629331/

Q: 既にデフォルトゲートウェイのIP取得時にARP要求がされているのか

Q: イーサネットだからブロードキャストという理屈であってるか

データを送受信するとき　インターネット層

インターネット層の処理

セグメント → パケットへとカプセル化

・IPヘッダーを追加し、パケットへとカプセル化する(完成系は右上参照)

・宛先IPアドレス　：※ 既に上位階層で**DNSサーバとの通信で取得済み**

・送信元IPアドレス：※ 既に上位階層で**DHCPサーバとの通信で取得済み**

データを送受信するとき　ネットワークインタフェース層

ネットワークインタフェース層の処理

パケット → フレームへとカプセル化

・イーサネットヘッダーを追加し、フレームへとカプセル化する(完成系は右上参照)

・宛先MACアドレス：**ARPテーブル(レイヤ3でエントリ情報追加済)**を参照し、**宛先IPアドレスのエントリ情報からMACアドレスを取得**

・送信元MACアドレス：※ 既に**デバイスのNICに付与されているため取得済み**

データを送受信するとき　ネットワークインタフェース層

ネットワークインタフェース層の処理

・電子信号化してでぶっ飛ばす

パケット → フレームへとカプセル化

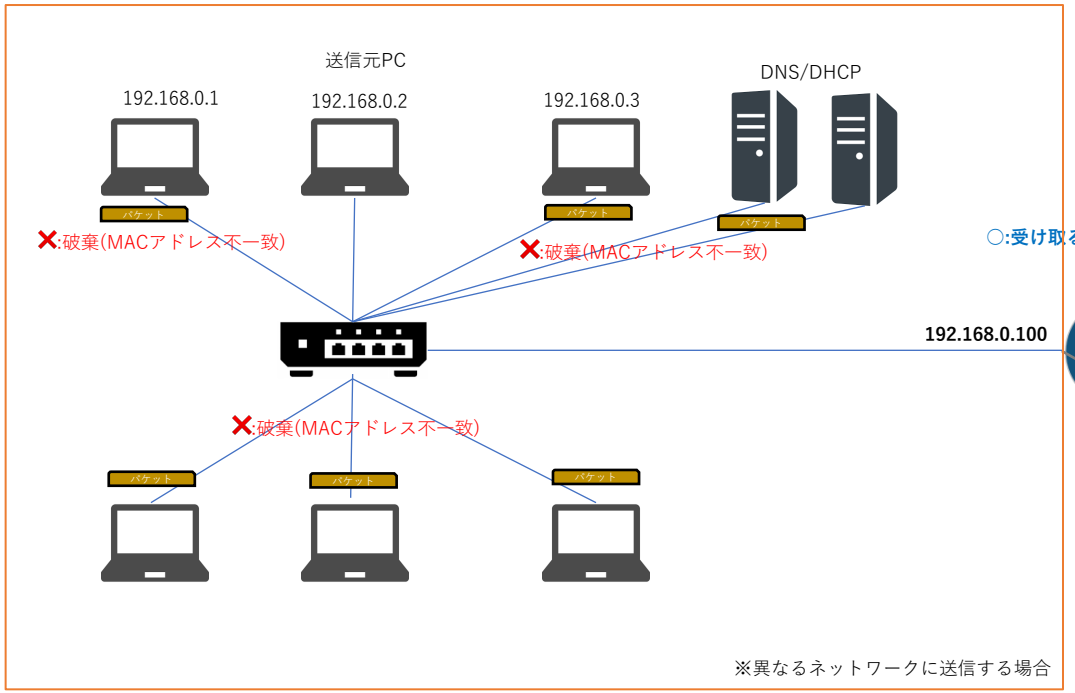
データを送受信するとき　イーサネット通信規格 / MACアドレス

同一ネットワーク内でのデータの動き(LAN . ver)

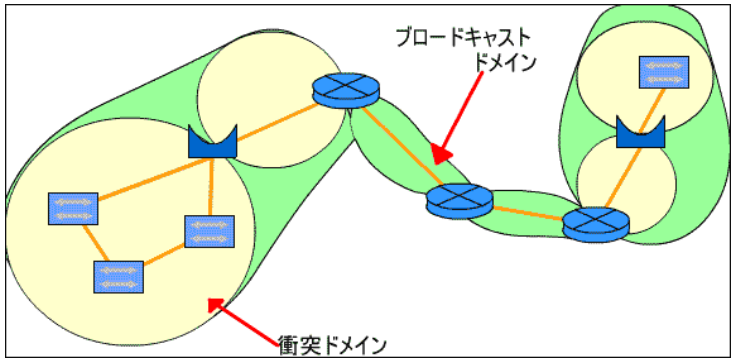
・イーサネット通信のため**ブロードキャストで通信**

・ネットワーク内で、受け取ったパケットの宛先MACアドレスが一致した場合のみ受信、それ以外は破棄

【ネットワーク: 192.168.0.0】



ルータの各ポートはIPアドレスを持ち
それぞれのネットワークに属している
(ブロードキャスト通信は受け取れる)
↓
デフォルトゲートウェイであることが多い



ブロードキャスト通信の影響範囲
※ルータはブロードキャストを通さない。

Q: フレームかパケット、どちらで呼べば良いのか

Q: イーサネットってブロードキャストであってるか、宛先MACアドレスわかっている場合はでも？

データを送受信するとき　TCPIP通信規格 / IPアドレス

モデムとルータの違い(wifiルーターも)のわかりやすい記事
https://cs.zaq.ne.jp/jpluszaq/articleDetail?cn=20161019172302

異なるネットワーク間でのデータの動き

■ルーティングテーブル

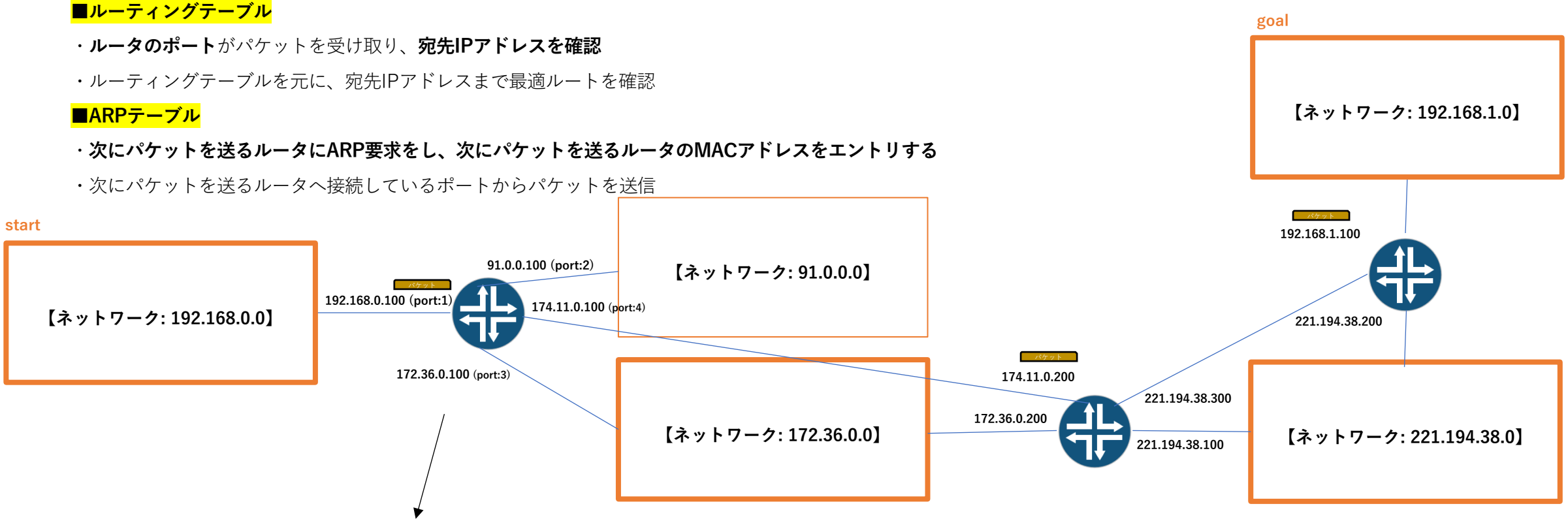
・**ルータのポート**がパケットを受け取り、**宛先IPアドレスを確認**

・ルーティングテーブルを元に、宛先IPアドレスまで最適ルートを確認

■ARPテーブル

・**次にパケットを送るルータにARP要求をし、次にパケットを送るルータのMACアドレスをエントリする**

・次にパケットを送るルータへ接続しているポートからパケットを送信



宛先ネットワーク	次のルータ	距離	ポート
91.0.0.0	なし	0	2番ポート
172.36.0.0	なし	0	3番ポート
221.194.38.0	174.11.0.200	1	4番ポート
192.168.1.0	174.11.0.200	2	4番ポート

参考: http://www5e.biglobe.ne.jp/aji/3min/28.html

Q: ルータはネットワークだけでなくルータ同士も繋がっているため、めっちゃたくさん繋がっている？

Q: 距離のカウントは経由するルータの数であっているか？

Q: ルータはネットワークだけでなくルータ同士も繋がっているため、めっちゃたくさん繋がっている？

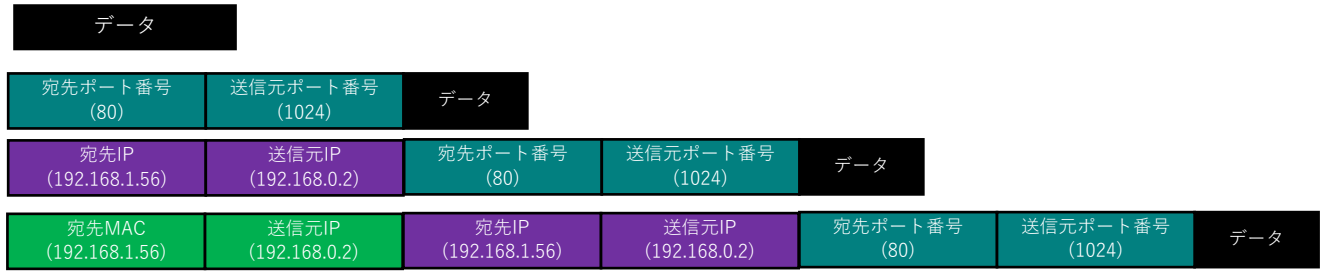
Q:最終的にはどこかのルータがブロードバンドルータ？プロバイダ？に繋がる？

データを送受信するとき 制御データの流れ

クライアント

【デバイス情報】
IPアドレス(192.168.0.2)
MACアドレス(34:8:bc:25:40:2)

【ネットワーク情報】
ネットワークアドレス(192.168.1.0)



トランスポート層

インターネット層

ネットワークインターフェース層

クライアントがブラウザでwww.sample.com のWEBデータをHTTPサーバから取得するまでの流れ

※ネットワーク内はLAN接続されている

Q: フレームかパケット、どちらで呼べば良いのか

Q: イーサネットってブロードキャストであってるか、宛先MACアドレスわかっている場合はでも？

データを送受信するとき　制御データの流れ

クライアント

IP(192.168.0.2)
MAC(34:8:bc:25:40:2)

【ネットワーク: 192.168.1.0】

【ネットワーク: 192.168.0.0】

○:受け取る(MACアドレス一致)

IP(192.168.0.100)
MAC(9e:43:34:48:42:4a)

IP(192.168.0.100)
MAC(9e:43:34:48:42:4a)

HTTP
192.168.1.56

データ

宛先ポート番号 (80)	送信元ポート番号 (1024)	データ

宛先IP (192.168.1.56)	送信元IP (192.168.0.2)	宛先ポート番号 (80)	送信元ポート番号 (1024)	データ

宛先MAC (192.168.1.56)	送信元IP (192.168.0.2)	宛先IP (192.168.1.56)	送信元IP (192.168.0.2)	宛先ポート番号 (80)	送信元ポート番号 (1024)	データ

トランスポート層

インターネット層

ネットワークインターフェース層

クライアントがブラウザでwww.sample.com のWEBデータをHTTPサーバから取得するまでの流れ

※ネットワーク内はLAN接続されている

Q: フレームかパケット、どちらで呼べば良いのか

Q: イーサネットってブロードキャストであってるか、宛先MACアドレスわかっている場合はでも？