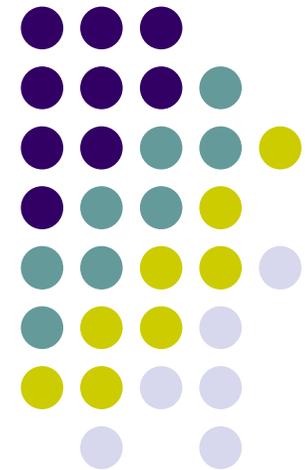


インターネットの仕組み

～ TCP/IPとアプリケーション ～

株式会社日本レジストリサービス (JPRS)
宇井 隆晴 (ui@jprs.co.jp)





はじめに

- 本チュートリアルの対象
 - いつも何気なく使っているインターネットの仕組みが知りたいユーザ
 - インターネット関連業務に携わることになった初心者
 - IPアドレス、TCP/IP、DNSという言葉は聞いたことがあるけど、よくわからない・知りたい人
- 注意
 - 初心者向けに幅広い分野を説明するために、技術的な正確さよりもわかりやすさを優先する場合があります。

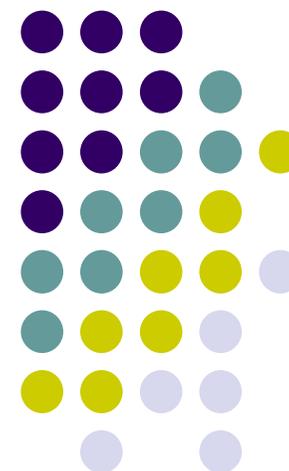


全体構成

- インターネットの歴史
- インターネットとプロトコル
- ネットワークのいろいろ
- インターネットプロトコルの核心
- ネットワークとアプリケーションの接続
- すべてはアプリケーションのために
- ドメイン名とIPアドレスの管理
- アプリケーションはどう動く
- インターネットと社会との融合

インターネットの歴史

「ARPANET」から「The Internet」へ





インターネットはなぜ生まれたか

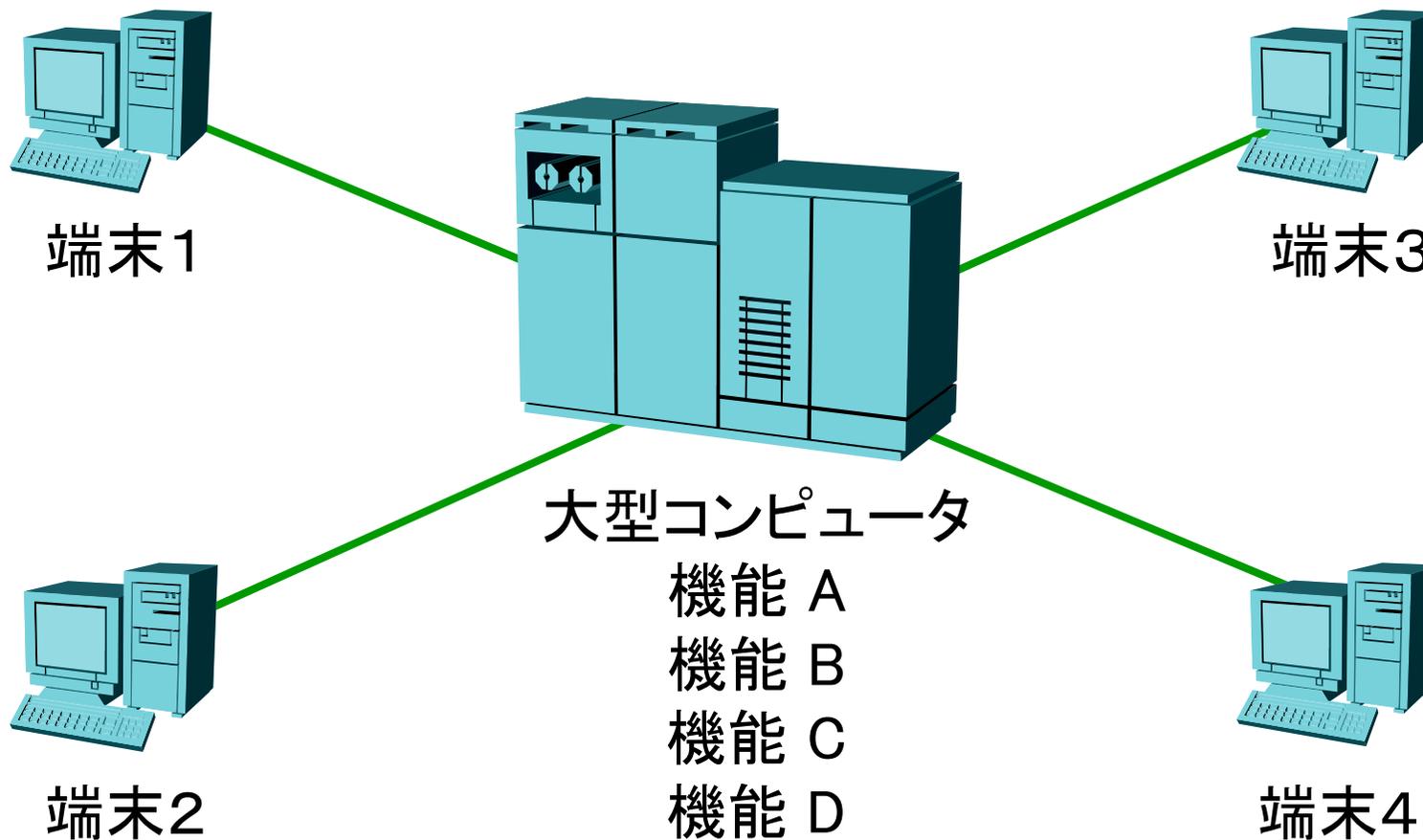
- 東西冷戦の産物
 - ARPA(Advanced Research Project Agency)
 - アメリカ国防総省 高等研究計画局
 - ソ連に勝る軍事科学技術開発を推進するための組織
 - 冷戦構造を背景に、莫大な予算を駆使
 - ARPANET(=インターネットの原型)
 - 1958年 国防総省内にARPA設置
 - 1962年 ARPA内にIPTO(情報処理技術部)設置
 - 1969年 ARPANET稼動



ARPANETが目指したもの

- 「核戦争に耐えられるコンピュータネットワーク」
 - 当時のコンピュータネットワークは「スター型」が主流
 - 中央の大型コンピュータにすべての機能が集中
 - 端末から中央に接続して機能を利用
 - 「スター型」の弱点
 - 中央の大型コンピュータが停止すると何もできない
 - 端末から中央への接続が途絶えると何もできない

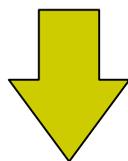
スター型ネットワーク





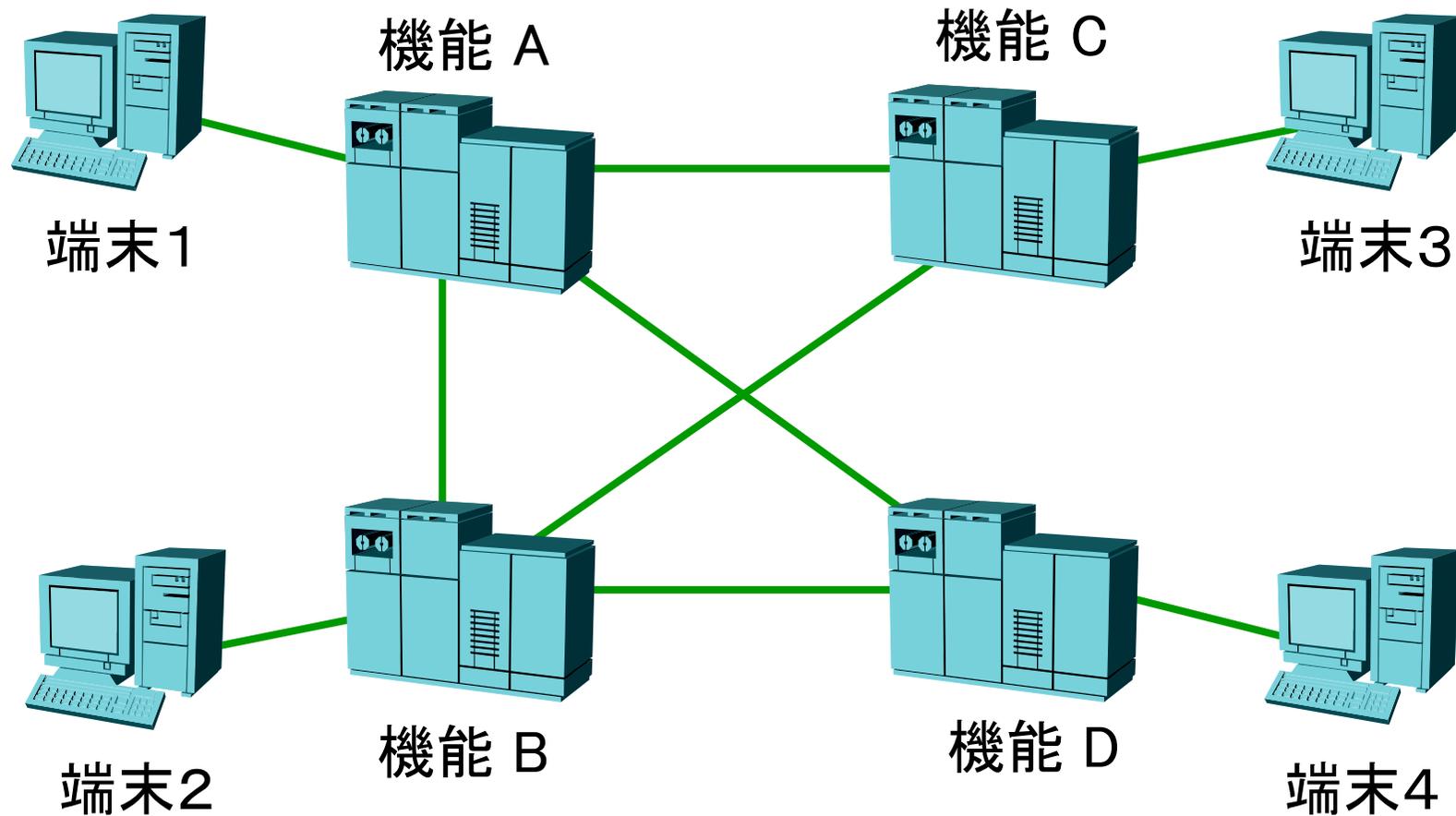
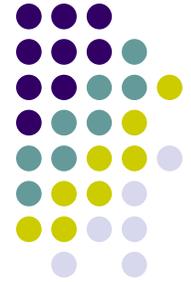
ARPANETの思想

- 核戦争に耐えるためには…
 - 機能を集中させずに、ネットワーク上に分散配置
 - 局地的な攻撃によりネットワークの一部が破断しても、通信が確保できる冗長性のあるネットワーク構成
 - ネットワークの成長、広域展開を支えることができる、単純で信頼性の高い通信技術



- パケット交換式広域分散ネットワーク
＝インターネットの原型

広域分散ネットワーク





…という建前の元に

- 政治的建前と技術的本音
 - 「核戦争に耐えるネットワーク」は国防総省から予算を獲得するための建前
 - 研究に携わった技術者の本音は、
 - コンピュータは便利だけど、専門的な処理はそれができるコンピュータを設置しているセンターに行かなければならないし、他の研究所のデータをもらうには磁気テープを郵送してもらわなくちゃならない。
 - 様々なベンダの、地理的にも離れているコンピュータを接続して、遠隔処理をしたり、情報資源を共有したい。
 - 「みんなで幸せになろうよ！」



ARPANETの拡大

- 1969年 最初のARPANET稼働
 - カリフォルニア大学ロサンゼルス校(UCLA)・サンタバーバラ校、スタンフォード研究所、ユタ大学を接続
- 1970年 ゼロックス社パロ・アルト研究所(PARC)設立
 - 1970年代 通信規格「イーサネット(Ethernet)」を開発
- 1972年 ARPAがDARPAと改名
- 1973～74年 プロトコル「TCP/IP」開発
 - DARPAのBob Kahnとスタンフォード大学のVinton Cerf
- 1980年代初頭 DARPAがUNIXへのTCP/IP実装を支援
 - 1983年 4.2BSDがリリースされる
 - UNIX＋イーサネット＋TCP/IP が標準的な形に



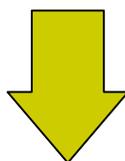
ARPANETからインターネットへ

- 1982年 NSF(全米科学財団)がCSNET開始
 - ARPANETで培われた技術を元にスーパーコンピュータの接続を開始
- 1983年 ARPANETから軍事部門(MILNET)分離
- 1986年 NSFがCSNETをNSFNETとして再構築
 - 大学間を接続する学術研究目的の**非商用ネットワーク**を構築
 - 全米の基幹ネットワークに
- 1990年 ARPANETが役割を終えて廃止
- 1991年 CIX(商用インターネット協会)設立
- 1993年～1994年 NSFNETの運用が民間に移管
 - インターネットを**営利目的利用に開放**



日本の政府主導学術ネットワーク(1)

- 1975年 N1ネットワーク稼動
 - 国立大学大型コンピュータネットワーク
 - 東京大学と京都大学を接続
 - 1980年にはほとんどの国立大学を接続
 - ARPANETとは異なる独自のプロトコルを採用
 - 電子メールが使えなかった



- 1999年 運用終了



日本の政府主導学術ネットワーク(2)

- 1986年 学術情報センター設立
 - 2000年 国立情報学研究所
 - 1987年 学術情報ネットワークパケット交換網開始
 - 国際標準として規格化が進められていた「OSI」を推進
 - しかしOSIはデファクト・スタンダードとなっていたTCP/IPを駆逐することはできなかった。
- ↓
- 2002年 運用終了
 - 1992年 インターネットバックボーン「SINET」運用開始



日本の草の根主導のネットワーク

- 1984年 JUNET誕生
 - 東京大学、東京工業大学、慶應義塾大学を接続
 - 電話線とUUCP (UNIXの通信方法)による接続
 - KDD研究所の協力で(こっそり)海外接続を実現、電電公社武蔵野研究所の協力で(こっそり)国内無料通信を実現
 - 87年5月、Inetクラブにより(正式に)海外接続を実現
- 1988年 WIDEプロジェクト誕生
 - 専用線とTCP/IPによる「日本のインターネット」



WIDEから日本のインターネットへ

- WIDEによる研究インターネットの構築
 - 研究者の草の根活動では専用線による本格的なインターネットの構築ができない
 - 当時は64kbpsの専用線が月額数十万円もした
 - 企業との共同研究という名目で資金を集め、日本のインターネットを構築したのが「WIDEプロジェクト」
- 1990年代から商用プロバイダサービスが広がる
 - 1993年10月 AT&T Jemsがサービス開始
 - 1993年11月 IIJがサービス開始
 - 1994年06月 富士通がサービス開始
 - 1995年04月 東京インターネットがサービス開始



インターネットの爆発

- WWW (World Wide Web) が出現
 - Mosaic、Netscapeがキラーアプリケーションに
 - インターネットの使い方が大きく変化
- Microsoft Windows95が標準でTCP/IPを搭載
 - インターネットはUNIXを使う大学の研究者のものではなく、一般のパソコンユーザが使うものに
- 個人向けプロバイダサービスの開始
 - BEKKOAME、Rimnetなど、安価な個人向けサービスの普及
 - OCN、ODNなどキャリア系サービスによる全国展開
- 個人のインターネット利用環境は定額・高速化
 - ダイヤルアップ接続からADSL、FTTHに
 - 従量課金から定額課金へ

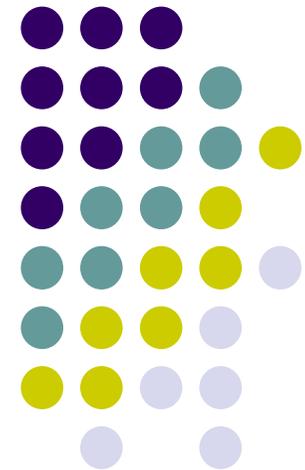


参考図書

- 「電網創世記」
 - 滝田誠一郎、実業之日本社
- 「インターネット」「インターネットII」
 - 村井純、岩波新書

インターネットとプロトコル

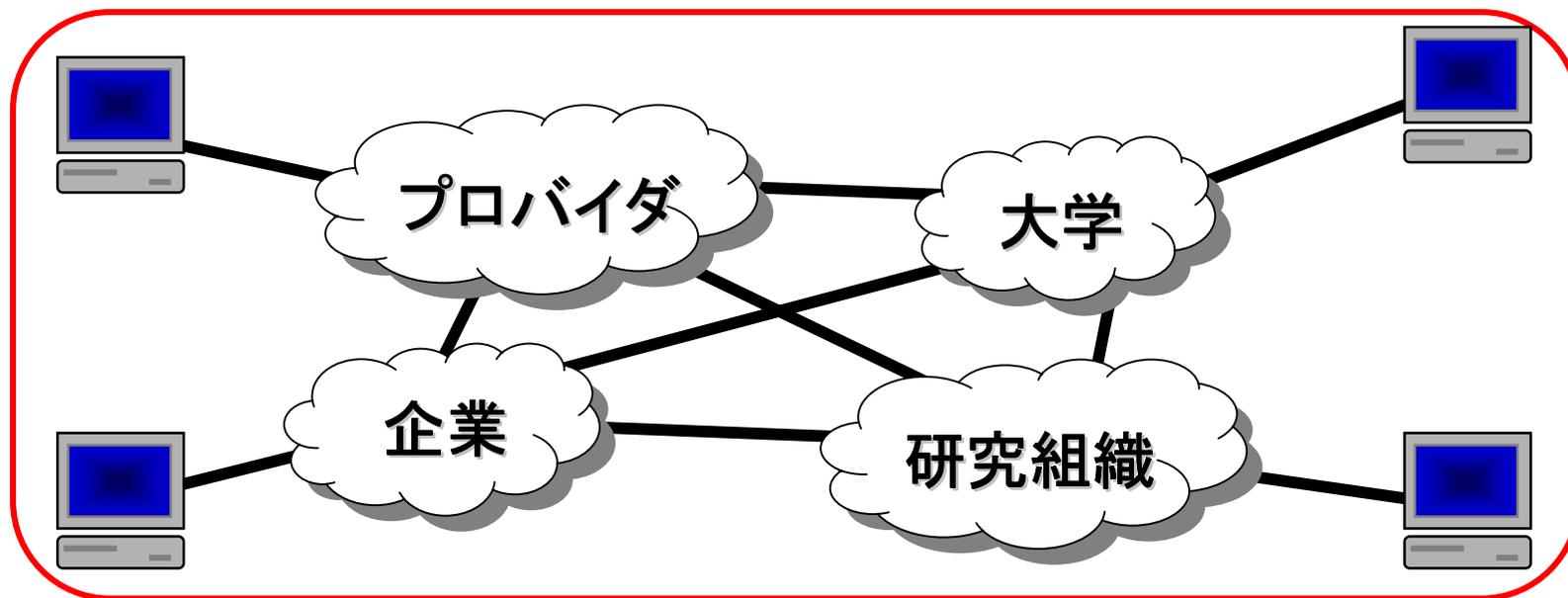
「通信」を成立させるために必要なもの





インターネットとは何か

- 「Internet」 = 「Inter」+「Network」
- 世界中のコンピュータ・ネットワークを接続した1つの大きなネットワーク





通信を成り立たせるための約束

- インターネットには異なるベンダの、異なるOSを搭載した様々な種類のコンピュータが接続されている
- これらが通信を行うためには、その手順を決めておく必要がある
- この手順を定めた約束事を「プロトコル」と言う



プロトコル

- 日常世界におけるプロトコル
 - 手紙: 表に郵便番号・住所・氏名を書いて、ポストに投函する
 - 電話: 相手の電話番号を入力し、「もしもし〇〇ですが…」と話し出す
 - 会談: あらかじめアポイントをとり、時間場所を取り決め、5分前には到着する
- コンピュータネットワークにおけるプロトコル
 - ネットワーク上の通信相手を指し示す方法
 - データパケットを電気信号へ変換する方法など



インターネットのプロトコル階層



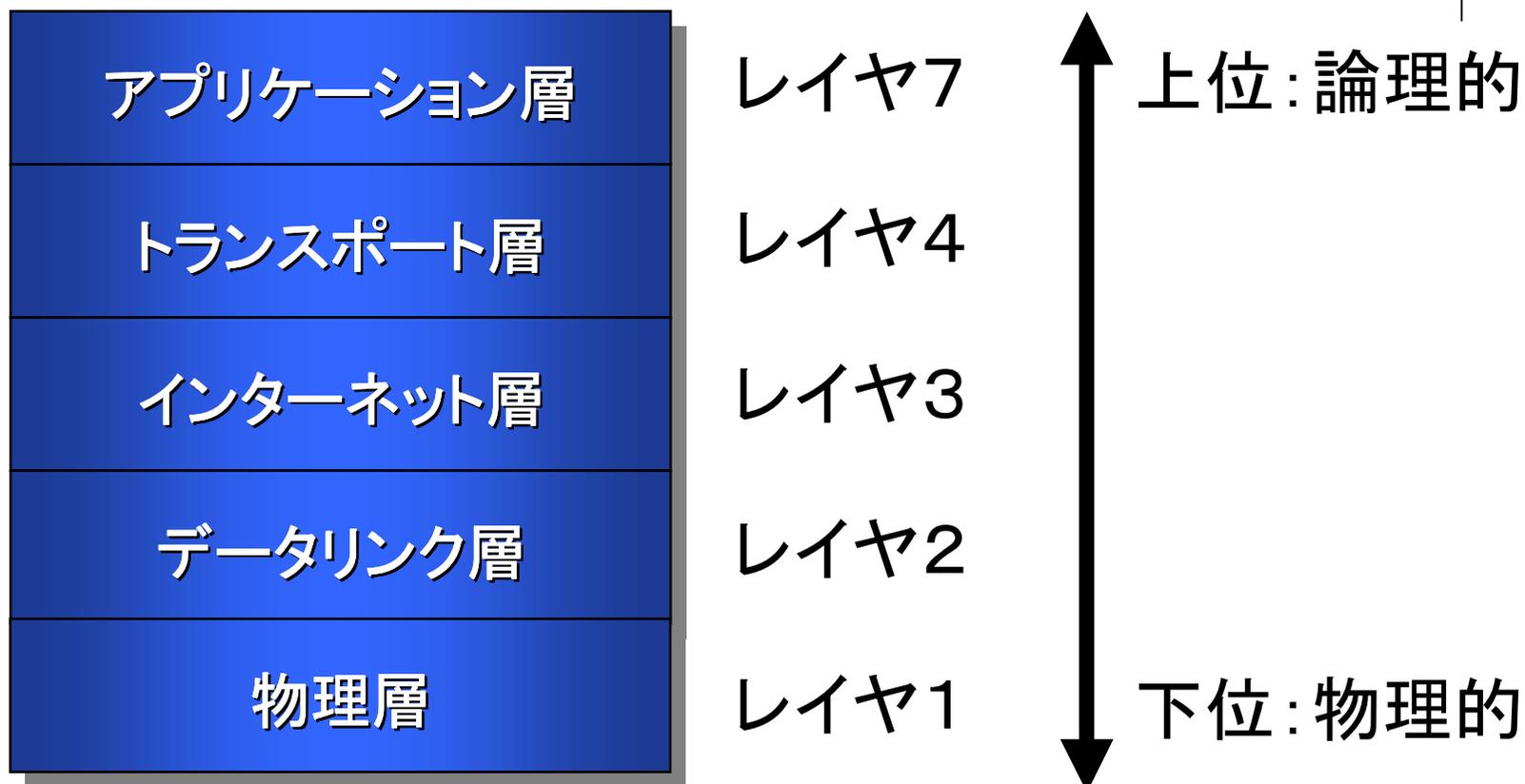
※書籍等では層の名称が違ったり7層に分けたものもありますが、本チュートリアルでは上記のような名称と定義し、また5層に分けて説明します。

- インターネットのプロトコルは階層的に設計されています。
- 層のことを「レイヤ」と呼びます。

- 各層の役割は？
- なぜプロトコルは1つではなくて、階層化されているの？



プロトコル階層の上下関係





ハードウェア層の役割



- ネットワークを構成する電氣的・物理的要素を定義
 - ソケット・コネクタの形
 - ケーブルの構造、特性
 - 信号の電圧、波形
- メーカーの違いによらない、物理的な接続を確保
- 物理的なネットワークの形成が役割



データリンク層の役割



- 信号処理手順を定義
 - データ通信手順
 - パケットサイズ、構造
 - 物理アドレスの規定
- 情報を、信号に変換して物理層に渡す
- 物理層からの信号を情報として復元する
- 論理的なネットワークの形成が役割



インターネット層の役割



- インターネット的な通信手順を定義
 - 論理アドレスの定義
 - 相手までの経路制御
- 情報に宛先情報を付加してデータリンク層に渡す
- データリンク層からの情報が自分宛のものかどうかを判別する
- ネットワークの中継地点では、適切なネットワークに情報を転送する



トランスポート層の役割



- アプリケーション間での接続の確立が役割
 - 通信相手アプリケーションの識別
 - 情報パケットの順序整理
- 情報をパケットに分割し、インターネット層に渡す
- インターネット層からのパケットを並び替えて連結して、適切なアプリケーションに渡す



アプリケーション層の役割



- アプリケーション内での手順を定義
 - Webへのアクセス手順
 - メールの送受信手順
- アプリケーションメッセージをトランスポート層に渡す
- トランスポート層からの情報を受け取って処理する



インターネットプロトコル群

アプリケーション層	HTTP (Web)	SMTP (Mail送信)	POP (Mail受信)
トランスポート層	TCP		UDP
インターネット層	IP		
データリンク層	イーサネット	ATM	FDDI
物理層			

※:他にも多くのプロトコルがありますが、代表的なもののみを記述しています。

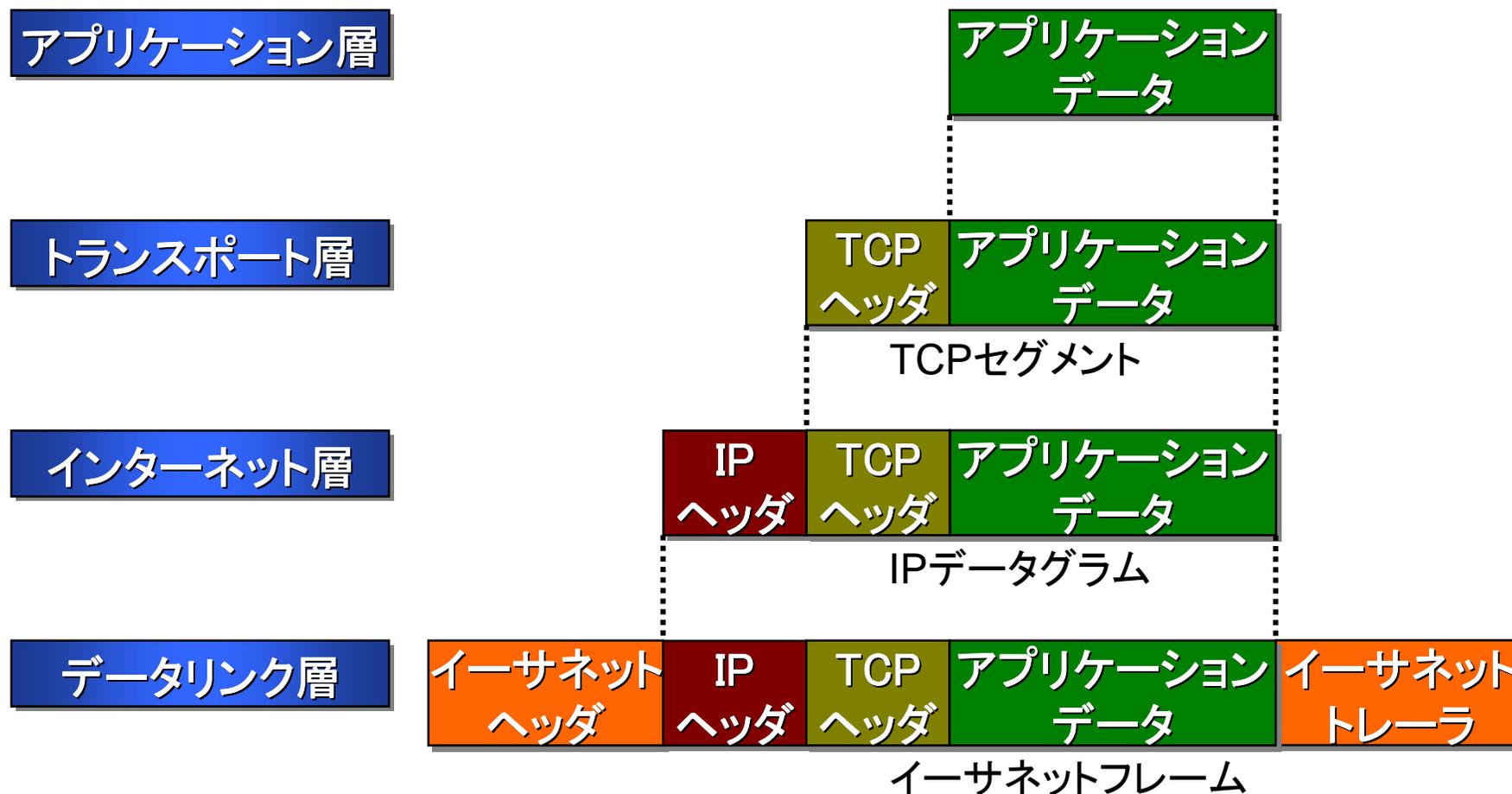


TCP/IP

- インターネットを構成するプロトコルは、
 - トランスポート層のTCP・UDPと
 - インターネット層のIPを軸に、
 - 上位にアプリケーションを、
 - 下位に物理ネットワークを配している。
- TCP/IPとは、
 - 狭義には中核となるトランスポート・インターネット層を、
 - 広義にはインターネットのプロトコル群全体を指す

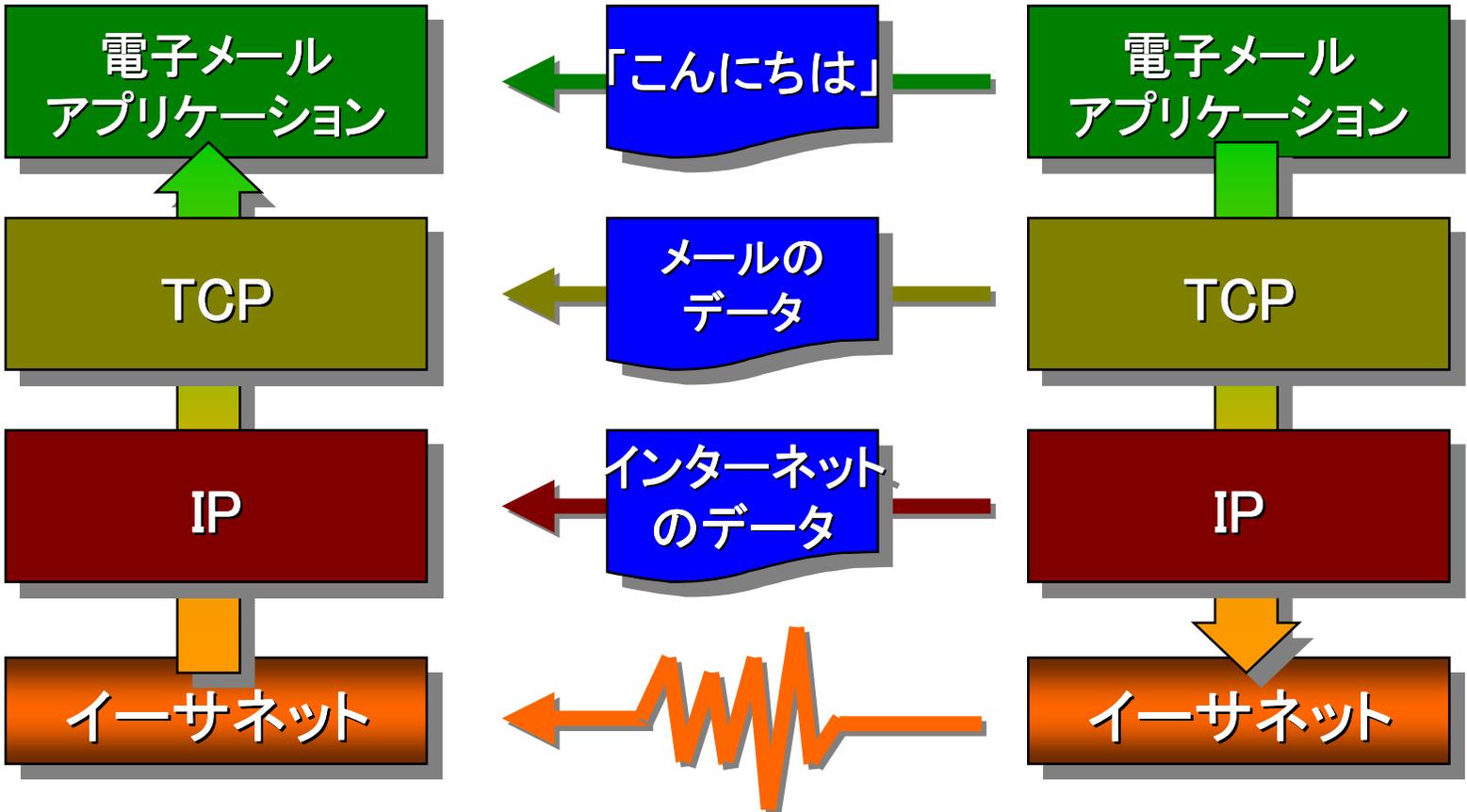


階層的プロトコルとカプセル化





電子メール送受信のプロトコルモデル



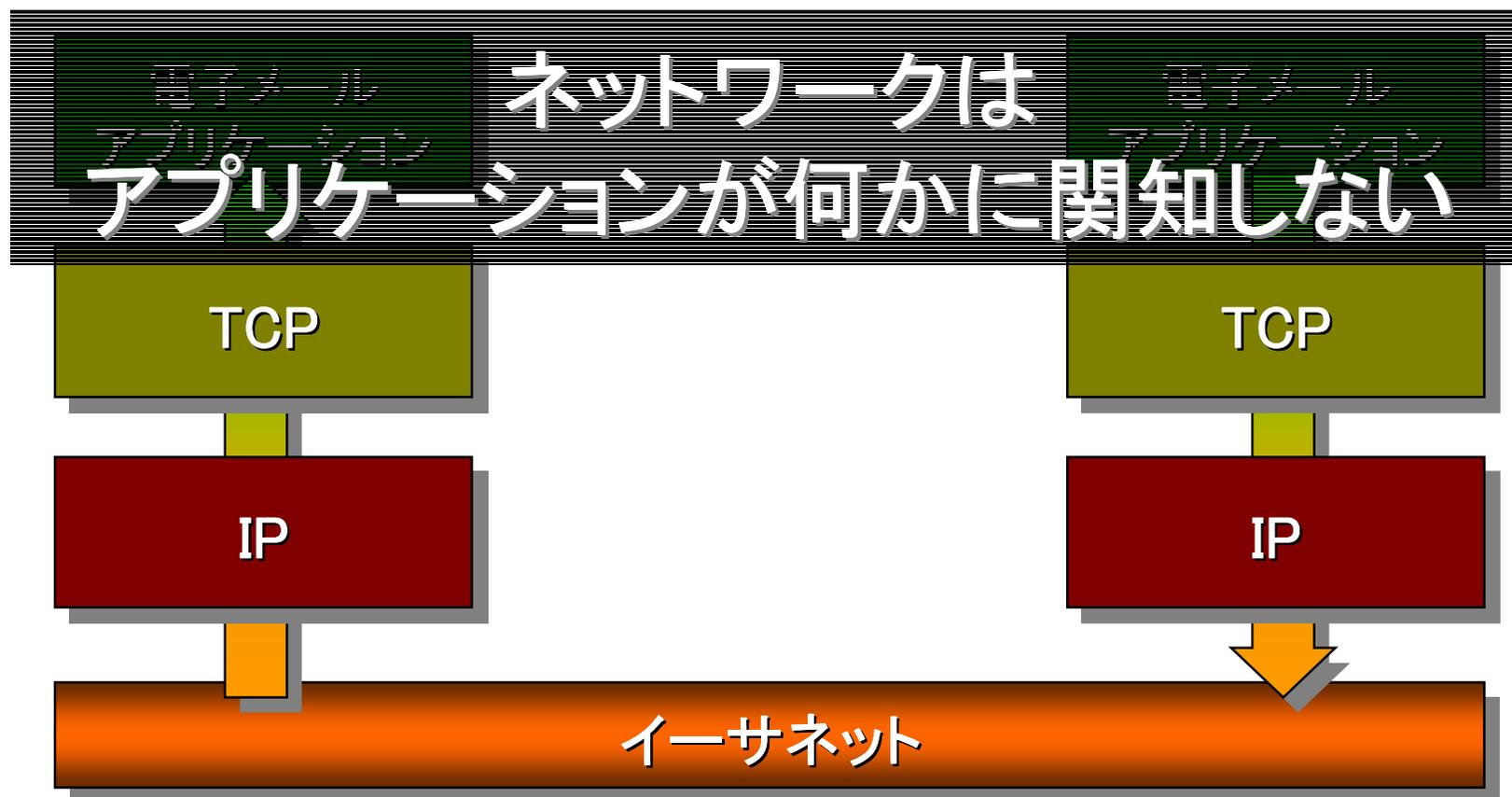


階層的プロトコルの理由





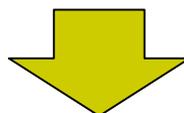
階層的プロトコルの理由





階層的プロトコルのメリット

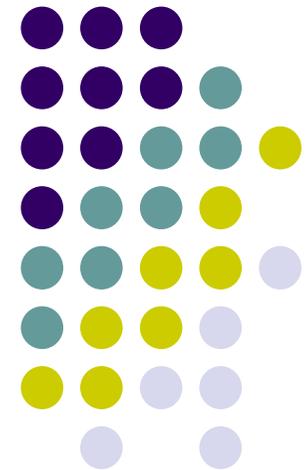
- 隣のレイヤへのデータの渡し方さえ守れば、離れたレイヤが何であるかに関知しなくてよい。
 - 電子メールアプリケーションは、それがどういうネットワークで運ばれるのか、知る必要はない。
 - イーサネットは、何のアプリケーションのデータを運んでいるのかを意識する必要はない。



- 同一レイヤ間での仮想的な接続を提供
- 各レイヤの役割の明確化による実装単純化
- レイヤ内での実装変更の容易さ

ネットワークのいろいろ

物理層・データリンク層





インターネットは異種混合

- 物理層・データリンク層の役割は、
 - データを信号に変換して、媒体を通して目的の機器まで伝達すること
- ところで、「インターネット」＝「イーサネット」だと思いませんか？
 - インターネットにはイーサネット以外のネットワークもいろいろと接続されている
 - 異なるネットワークを接続することがインターネットの目的でもある



物理層で使われる媒体

UTPケーブル
(Unshielded Twisted Pair)



光ファイバケーブル





データリンク層プロトコル: Ethernet

- LANで最も一般的に使われている方式
- 正式にはIEEE802.3として規格化されている
- 接続機器はMACアドレスという固有の番号を持つ
- CSMA/CD方式の典型
 - Carrier Sense Multiple Access with Collision Detection
 - 通信路を見張って、誰も話してなければ話し出す
 - 運悪く誰かと同時に話し出したら話すのをやめて、ちょっと待つ
- 通信速度、ケーブルの種類によるバージョンが存在
 - 10BASE-5、10BASE-2
 - 10BASE-T、100BASE-TX、1000BASE-T、etc...



その他のデータリンク層プロトコル

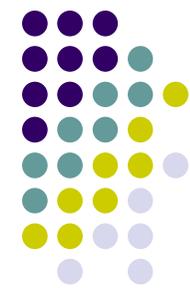
- ATM
 - 物理層に光ファイバを用いた、155～622Mbpsのネットワーク
 - データを固定長の「セル」に分割し、ハードウェアのATMスイッチによって仮想的な通信路を確立させる方式
- FDDI
 - 物理層に光ファイバを用いた、100Mbpsのネットワーク
 - トークンリング方式
 - リング状のネットワークに発言権を示す「トークン」を流す
 - トークンを持っている者だけが発言できる
 - 発言の必要がなければ隣の人にトークンを渡す
- 無線LAN (IEEE802.11x)、PPP、などなど



ネットワークをつなぐ：物理層

- 物理層での接続とは、
 - 同じ種類のネットワークを電氣的に接続すること。
 - 異なる種類のネットワークの接続は物理層ではできない。上位層を経由する必要がある。
- リピータ (Repeater)
 - 信号の中身に関係なく、無条件に電気信号を伝達する。
 - 一般的に用いられるハブ (Hub) はリピータの一種。

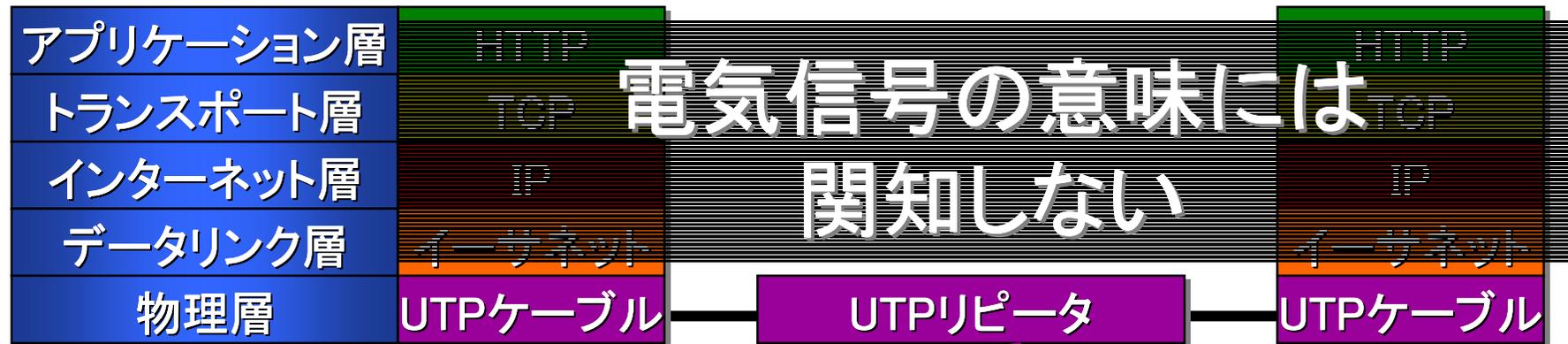
物理層での接続



転送元ホスト

リピータ

転送先ホスト



電気信号



ネットワークをつなぐ：データリンク層

- データリンク層での接続とは
 - 同じ種類のネットワークを信号的に接続すること
 - データリンク層が同一で、物理層の異なるネットワークの接続を行うことが可能。
- ブリッジ (Bridge)
 - 信号の宛先MACアドレスを見て、伝達すべきもののみ伝達する。
 - 一般的に用いられるスイッチングハブはブリッジの一種。

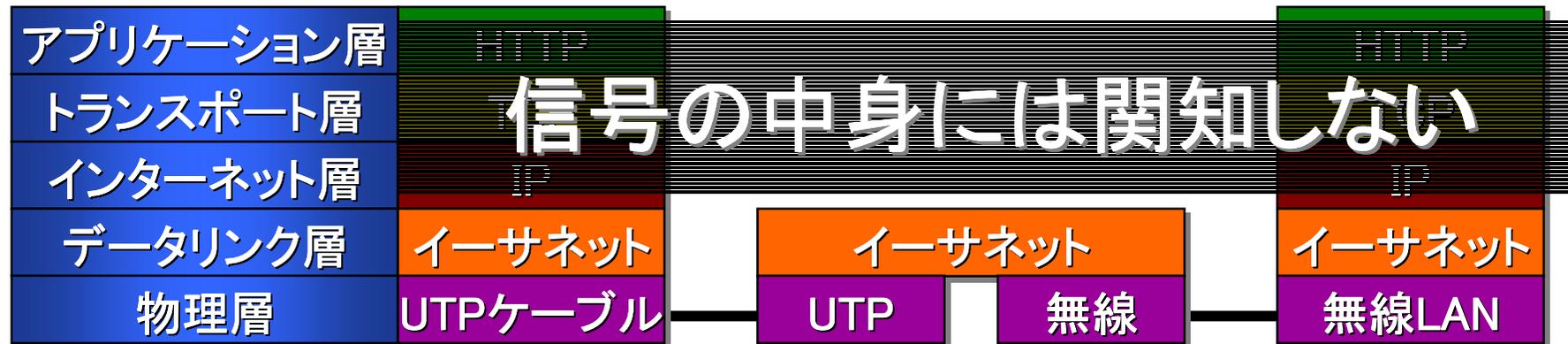
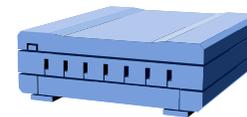
データリンク層での接続



転送元ホスト

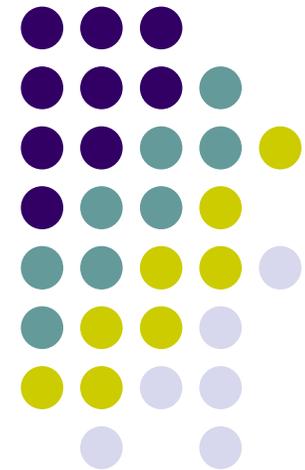
ブリッジ

転送先ホスト



インターネットプロトコルの核心

インターネット層

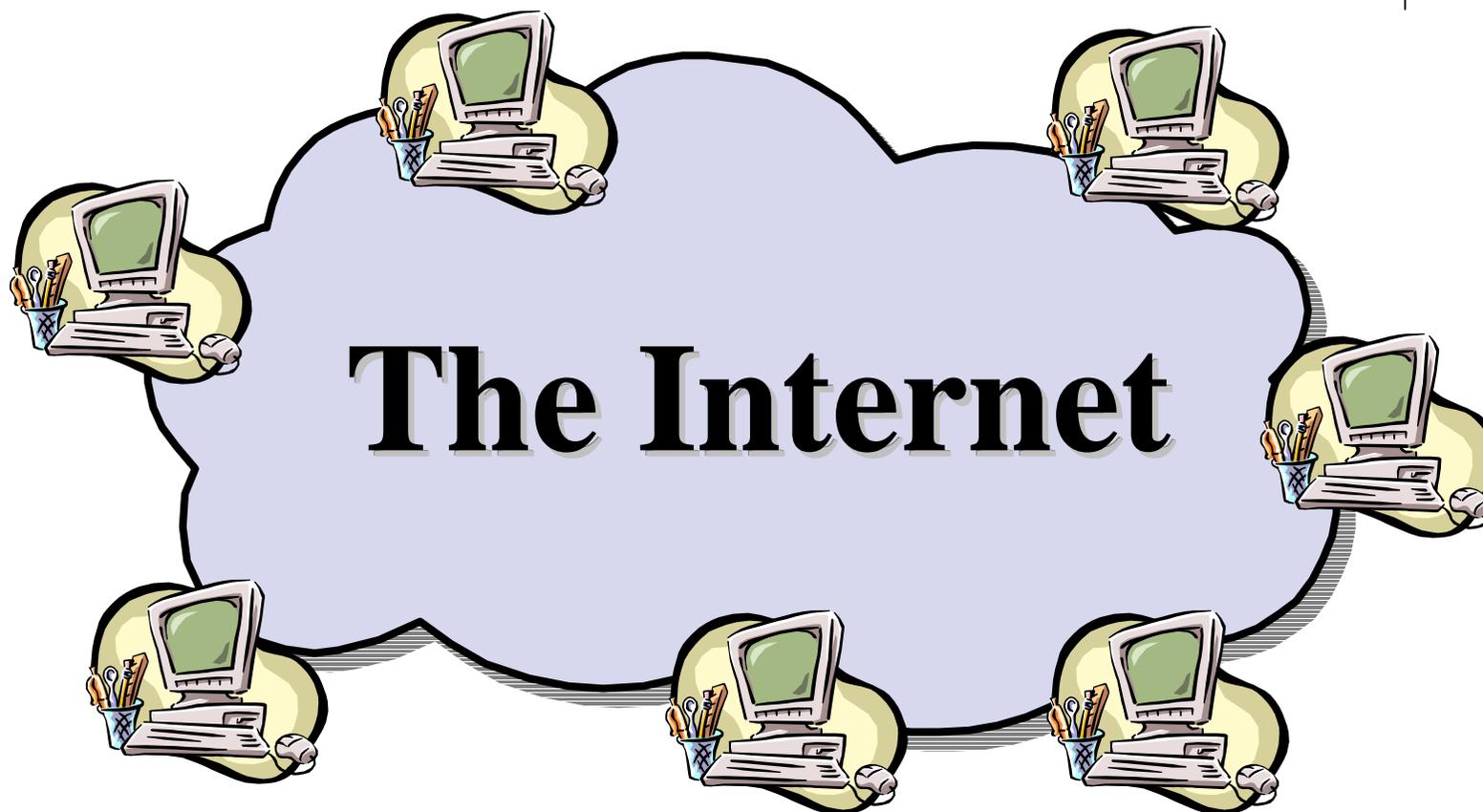




IP = Internet Protocol

- インターネット層のIPこそが、インターネットのプロトコルの中心
- IPの役割
 - 伝達する情報を「パケット」という小包にする。
 - パケットには情報の送り先などの「ヘッダ情報」を付加する。
 - ヘッダ情報が付加されたパケットが「IPデータグラム」
 - IPデータグラムを始点ホストから終点ホストまで運ぶことがIPの大きな役割

送り先をどう特定する？





IPアドレス

- 送り先の特定方法
 - 電話なら電話番号
 - 手紙なら住所・氏名
 - IPネットワーク(インターネット)なら「IPアドレス」
- IPを使って(=インターネット)通信をするものにはすべてIPアドレスが必要
- IPアドレスは、インターネットにおける住所
 - 異なる機器が同じIPアドレスを使ってはいけない



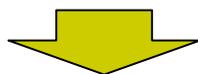
IPアドレスの表記

- IPアドレスは32ビット(32桁)の2進数

1100000010101000000000000000001010

- 人が覚えやすいように、8ビットごとにピリオドで区切って10進数で表記する

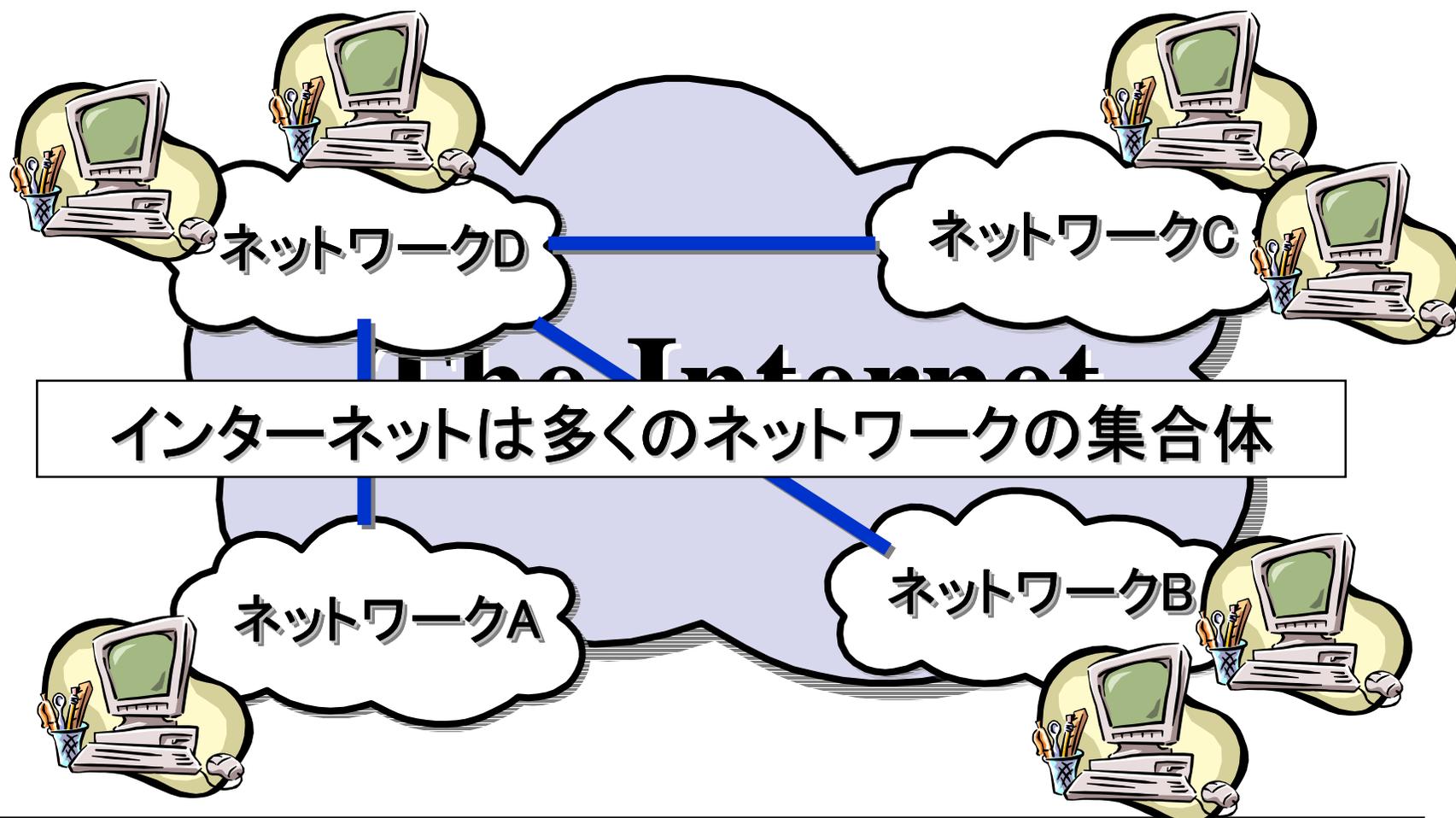
11000000 . 10101000 . 00000000 . 00001010



192 . 168 . 0 . 10



通信相手はどこにいるの？





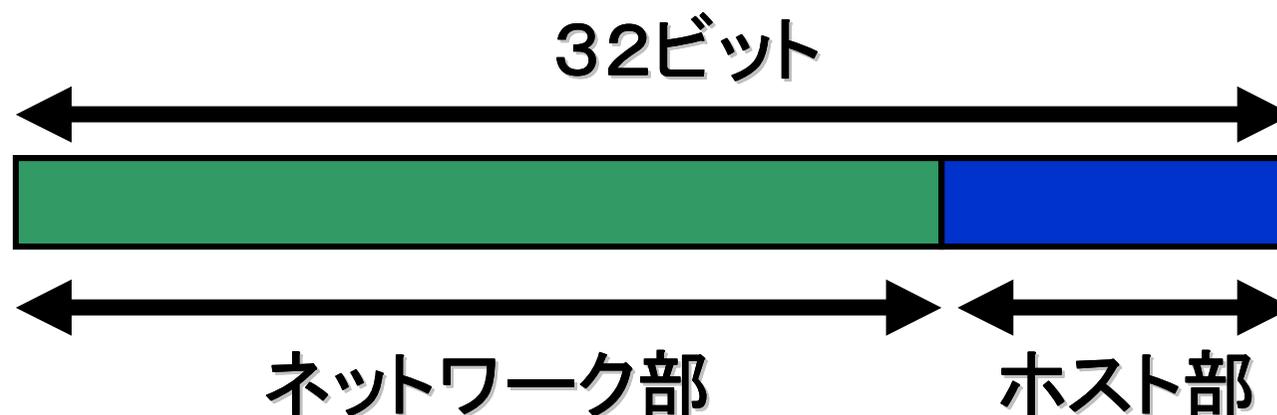
ネットワークの識別

- IPアドレスによって通信相手を特定するだけでは、IP データグラムを届けることはできない
- インターネットはネットワークの集合なので、相手がどのネットワークにいるのかが分からなければいけない
- IPアドレスでは、相手のネットワークの識別と、相手ホストの特定の両方を行うことができる。



IPアドレスのネットワーク部とホスト部

- ネットワーク部・ホスト部に分かれている

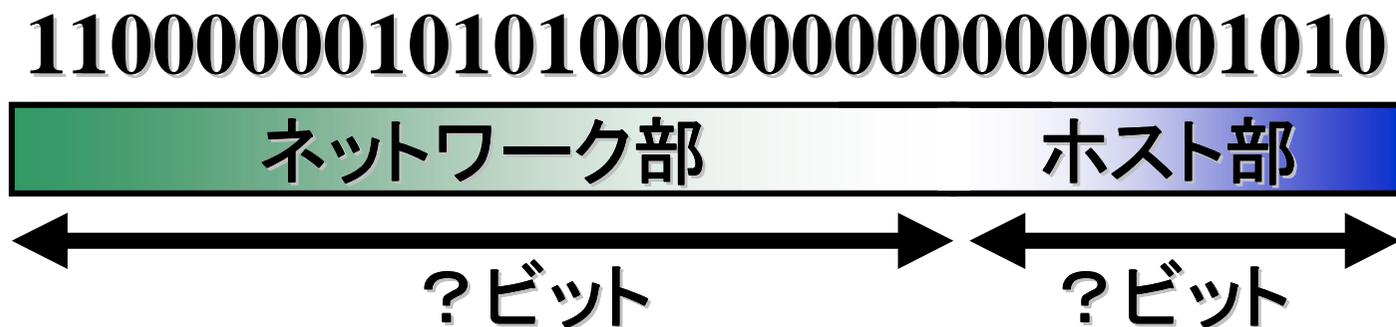


- ネットワーク部
 - インターネット全体の中でのどのネットワークかを識別
- ホスト部
 - そのネットワークの中のどのホストかを識別



どこまでがネットワーク部？

- IPアドレスは0と1の羅列にすぎない
 - IPアドレスを見ただけではネットワーク部の長さがわからない



- ネットマスクと呼ばれる目印で境目を表現



ネットマスクの表記

- IPアドレスの表記に合わせた32ビット形式
 - ネットワーク部を全て1、ホスト部を全て0と表記
 - IPアドレスと併記することでネットワーク部の長さを表現

IPアドレス 11000000.10101000.00000000.00001010
ネットマスク 11111111.11111111.11111111.00000000

1が先頭から24個並んでいる → ネットワーク部の長さは24ビット

このネットマスクを10進数に直すと、255.255.255.0 になる

インターネット層での接続



転送元ホスト

ルータ

転送先ホスト



IPヘッダ TCPセグメント



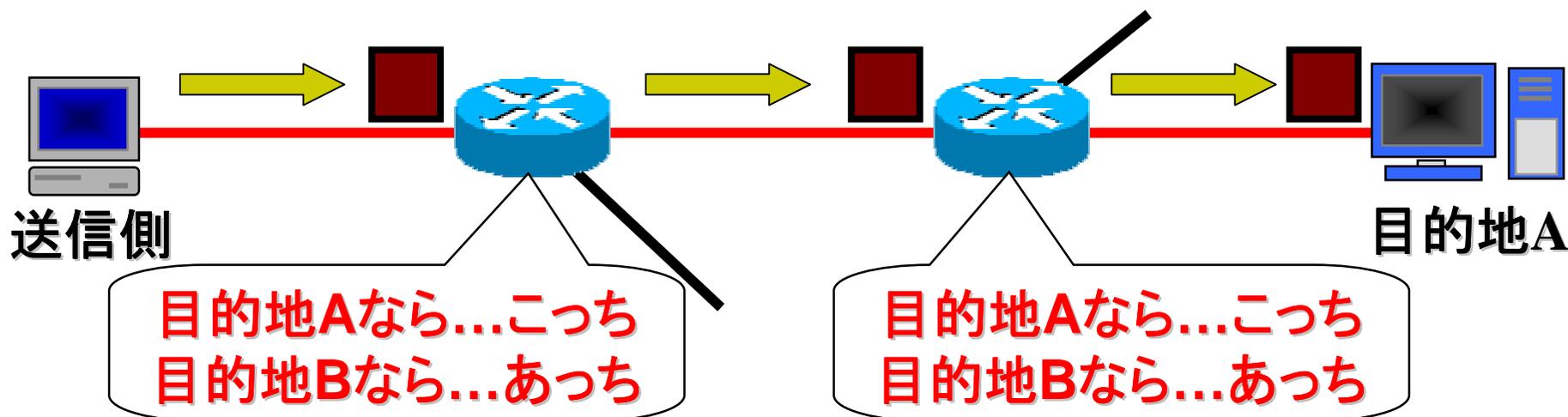
ルータと経路制御

- 宛先のネットワークがわかっても、IPデータグラムをどうやってそこまで運んでいくのか？
- ネットワークとネットワークを接続しているのが「ルータ」
- IPデータグラムはルータによってネットワークを渡り歩いて目的地までたどり着く。
- IPデータグラムを次にどのルータに投げればよいのか？
→ 経路制御(ルーティング)
- ルータは宛先ネットワークと、それに対応する中継ルータの一覧表を保持 → 経路表(ルーティングテーブル)

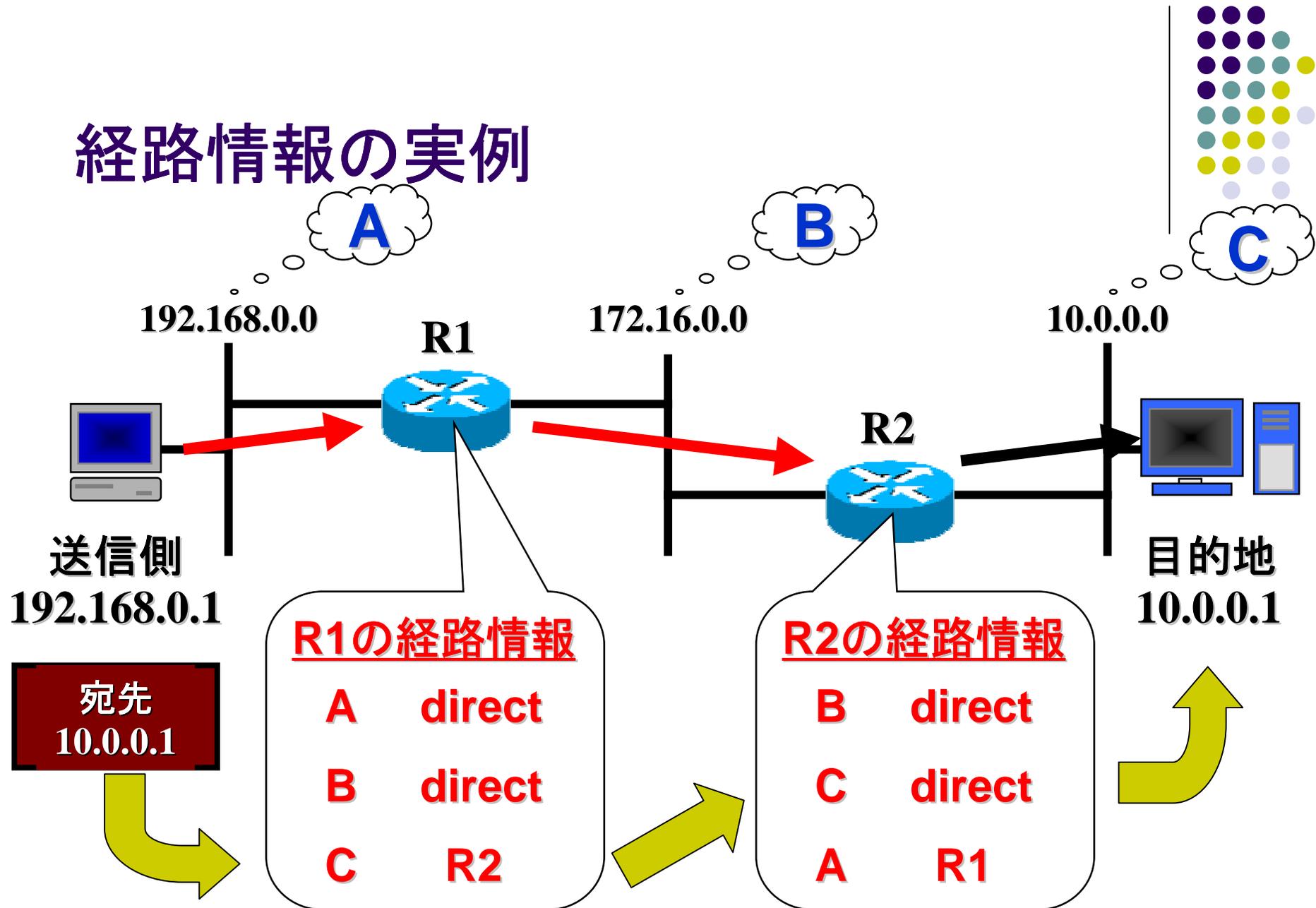


経路制御の仕組み

- ルータがIPデータグラムを目的地に運ぶ
 - 宛先IPアドレスのネットワークアドレスを見て判断



経路情報の実例





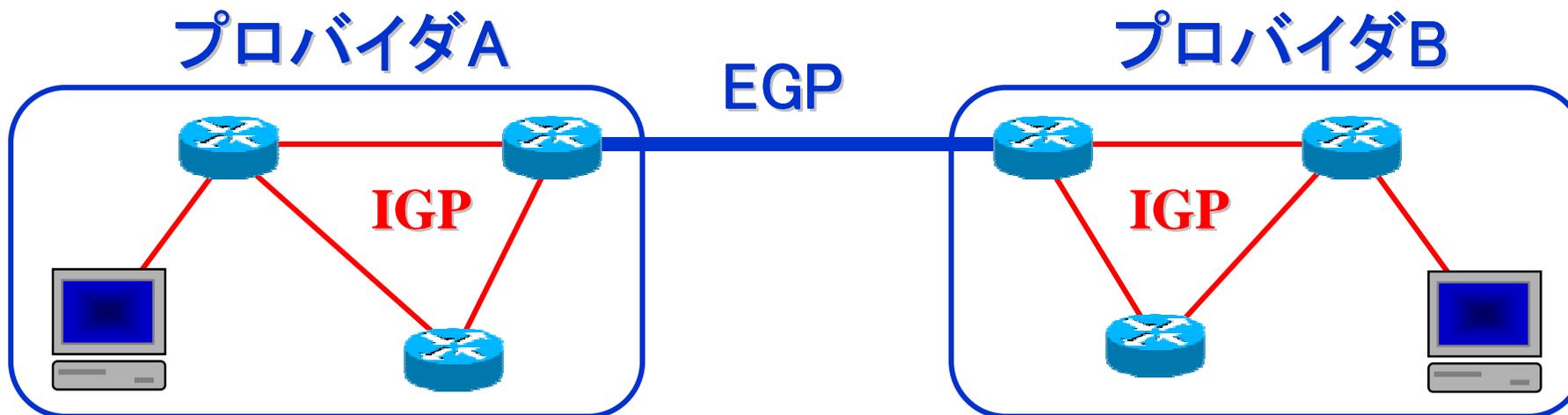
経路情報の自動生成

- 経路情報はインターネットの拡大と共に増加
 - 経路情報を手動で設定しては、ネットワークの数の増加と、日常的な構造の変化に対応できない。
 - 自動的に経路情報を作る仕組みが考案された。
- 「ルーティングプロトコル」の基本
 - ルータに接続されているネットワークの情報を保持
 - 同一ネットワークのルータと経路情報を交換
 - 離れたネットワークの情報もルータ同士の情報交換でルーティングテーブルに追加される



インターネット上のルーティング

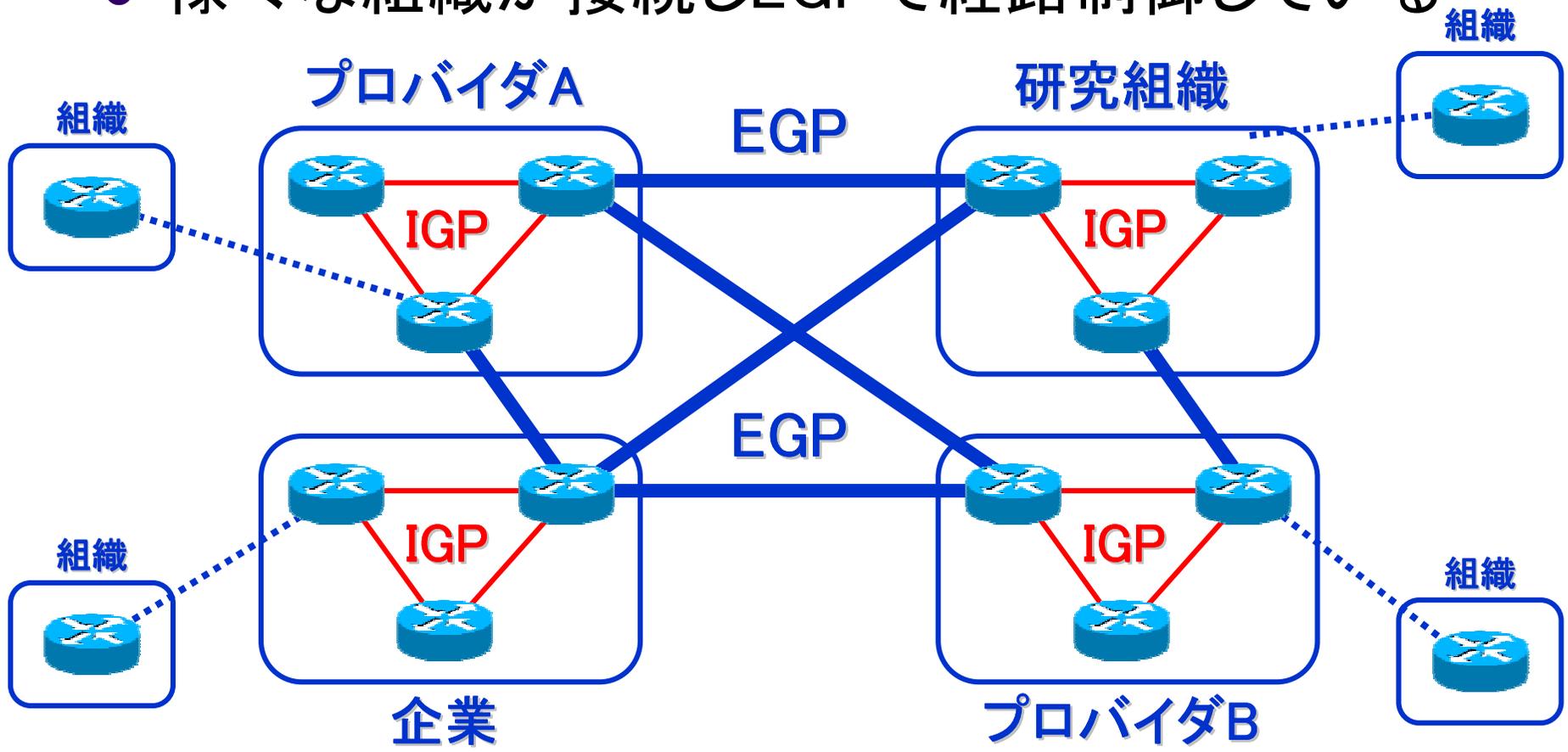
- 組織間は大まかな経路制御
 - EGP (External Gateway Protocol)
- 組織内部は細かい経路制御
 - IGP (Internal Gateway Protocol)





インターネット全体の構造

- 様々な組織が接続しEGPで経路制御している





代表的なルーティングプロトコル

- IGP

- 小～中規模なネットワーク向け
- RIP (Routing Information Protocol)
- OSPF (Open Shortest Path Fast)

- EGP

- 大規模なネットワーク向け
- 主にISP同士の接続に使われる
- BGP (Border Gateway Protocol)



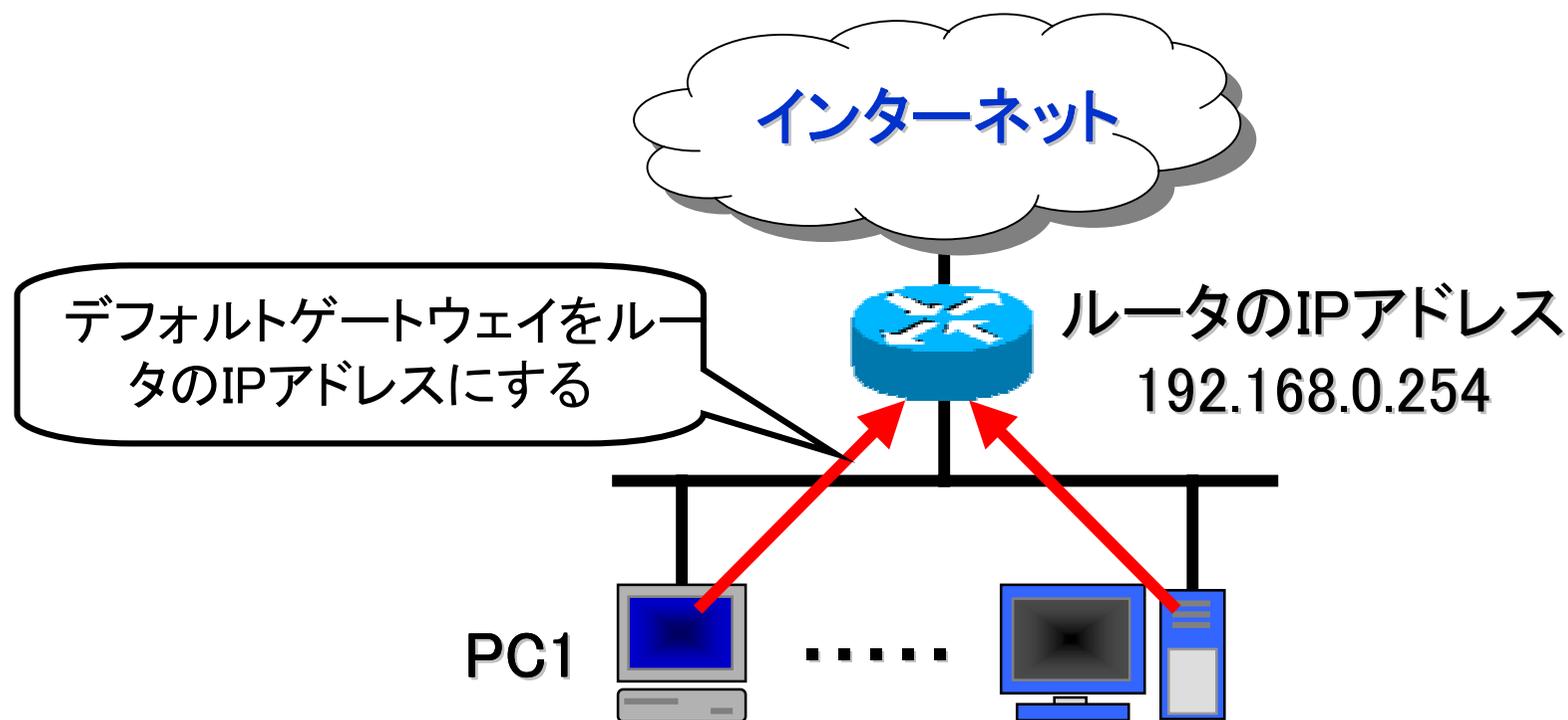
その他の経路情報

- デフォルトゲートウェイ
 - デフォルトルートとも呼ばれる
 - 経路情報に存在しない宛先に適用される経路
- インターネットへの出口が1つしかない時に使用される
 - 末端のPC
 - インターネットへの出口が1つしかないルータ



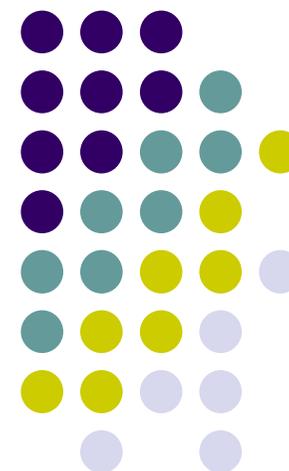
デフォルトゲートウェイの実例

- デフォルトゲートウェイをルータに向ければあとの経路制御はルータに任せられる



ネットワークとアプリケーション の接続

トランスポート層





トランスポート層の役割

- インターネット層までの活躍で、データを送り先のホストまで届けることはできた。
- でも…
 - ホストの中で動いているいろいろなアプリケーションの中から、目的のアプリケーションにデータを渡すにはどうしたらよいの？
 - 情報が伝送の途中で失われたりした場合、どうしたらよいの？
- そこでトランスポート層の出番



トランスポート層のプロトコル

- TCPとUDPの2つ
 - TCP … Transmission Control Protocol
 - UDP … User Datagram Protocol
- TCP/IPというとTCPとIPの組み合わせ、と思いがちですが、そうではありません
 - UDPとIPを使った通信は「UDP/IP」とは言いません
 - 様々なプロトコル全体を指して「TCP/IP」と言います

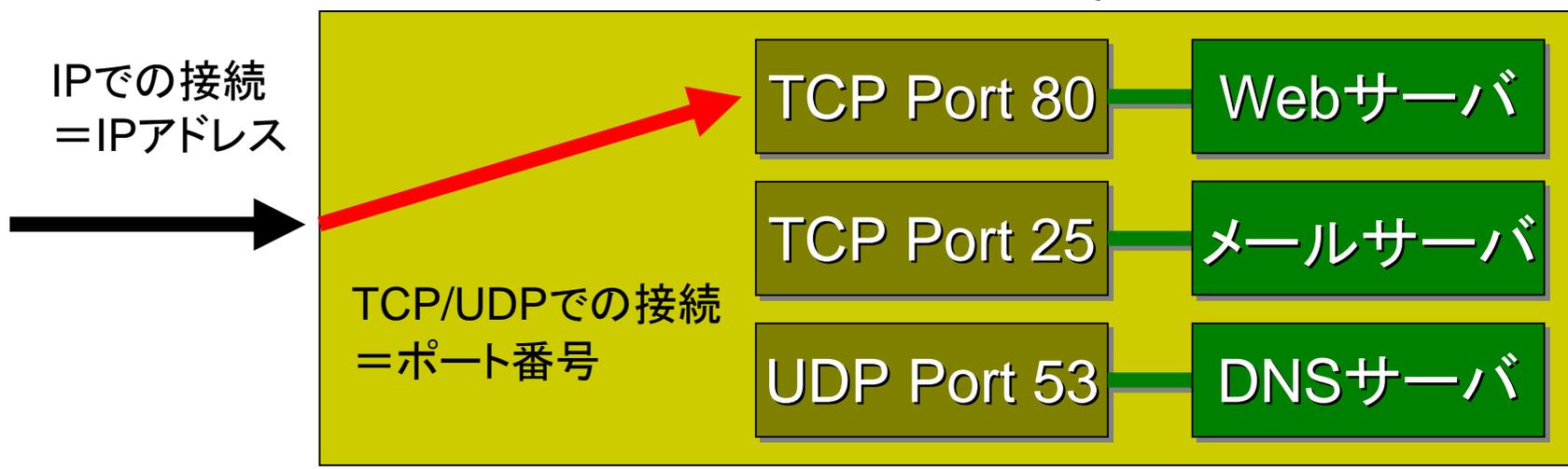


ポート番号: アプリケーションの通信窓口

- IPアドレスでホストを識別したその先にあるもの
 - アプリケーションの通信窓口を識別するために用意されているのが「ポート番号」



ホストA
192.168.0.10





ポート番号

- 代表的なアプリケーションには世界共通のポート番号が割り当てられている
 - HTTP → TCP ポート80番
 - SMTP → TCP ポート25番
 - DNS → UDP ポート53番
- メールを送る場合ならTCPを利用しポート番号を25番にセットしてデータを送信する
- 0～1023の番号をWell Known Portと呼ぶ
 - 標準的なアプリケーションが利用するために予約されているポート番号
 - 勝手に他の用途に使ってはいけない



TCPとUDPの違いは？

- どちらも、IPで運ばれたデータをポート番号で識別されるアプリケーションに届ける、という機能は同じ。
- 違いは「信頼性の確保」に対する姿勢
 - IPデータグラムは、経路の途中で壊れたり失われたりする。
 - IPデータグラムは、先に送信したものよりも後から送信したものが、順序が入れ替わって先に到着することがある



TCPは「コネクション型」

- TCPでは、最初に通信をするアプリケーション同士が通信用の仮想的なパイプを作る。
 - このパイプは、入り口でパケットに細切れにしたデータも、出口で元通りに復元する
 - 復元するときには、パケット到着の順序が入れ替わっていても、ちゃんと元通りに並び替える
 - このパイプを通るデータが途中で壊れたり失われたりした場合は、自動的に再送要求をする
 - 相手がデータを正しく受け取ったことを確認する

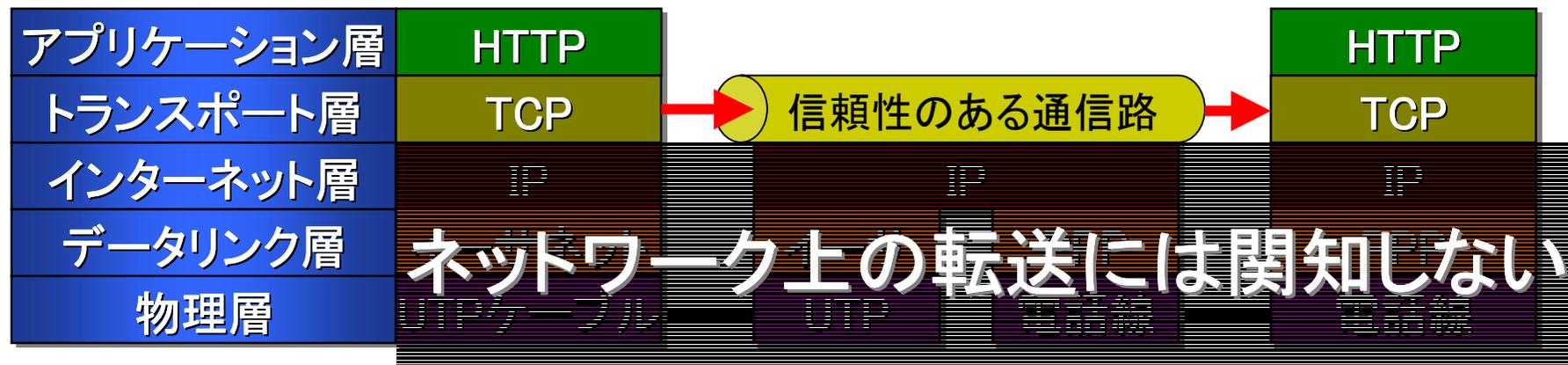
TCPによる接続



転送元ホスト



転送先ホスト



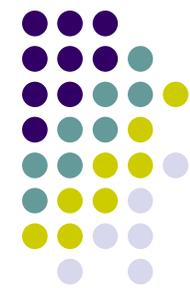
TCPヘッダ アプリケーションデータ



UDPは「コネクションレス型」

- UDPでは、パケットに細切れにしたデータ(UDPデータグラム)を相手にどんどん投げつける。
 - 細切れにしたパケットは、相手に届くときも細切れのまま。
 - 順番が入れ替わっても知らない。
 - 途中で壊れたりなくなったりしても知らない。
 - 相手がデータを受け取れる状態かどうかも知らない。
 - 相手がデータを受け取れなかったとしても知らない。

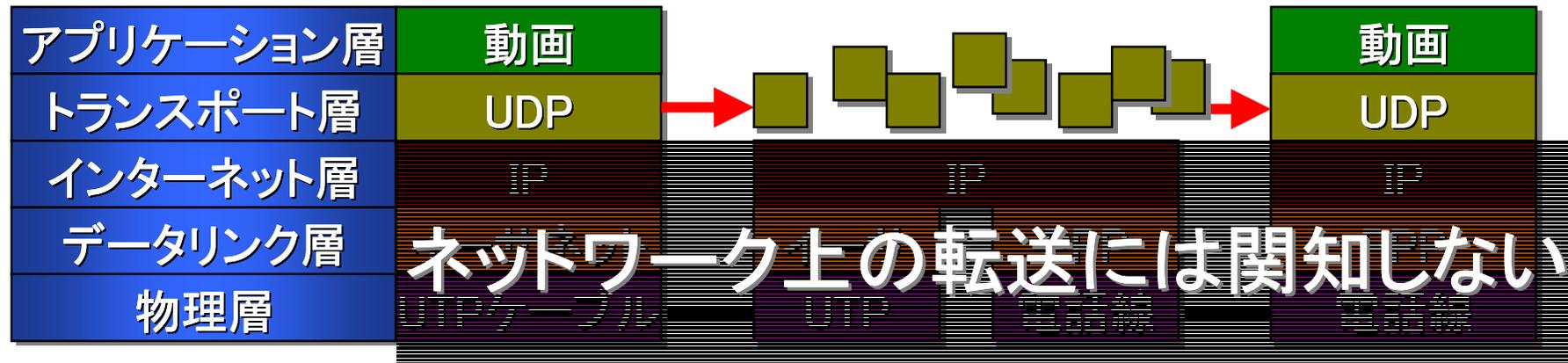
UDPによる接続



転送元ホスト



転送先ホスト



UDPヘッダ アプリケーションデータ

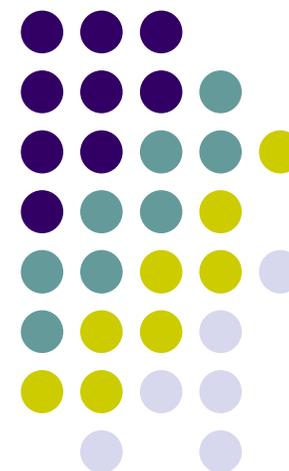


TCPかUDPか

- TCPとUDPの違い
 - TCPは信頼性が確保できるが、その分の性能的オーバーヘッドも大きい
 - UDPは身勝手だけど、伝送効率が高い
- 利用するサービスによって使い分ける
 - 通信の確実性が必要なものはTCP
 - 電子メールやWebコンテンツなど
 - データ単位が小さかったり、多少情報を失ってもリアルタイムな通信を優先する場合はUDP
 - 映像・音声の配信や、DNSクエリなど

すべてはアプリケーションの
ために

アプリケーション層





結局のところ

- 物理層からトランスポート層までは、すべてアプリケーション層のためにある。
- アプリケーション層から見れば、データが相手のアプリケーションに正しく届けばよいだけ。
 - 「ネットワークさん、がんばってね」
- というわけで、アプリケーション層の勉強では、他の層のプロトコルはとりあえず忘れましょう。



アプリケーションプロトコルの例

- DNS (Domain Name System)
 - ドメイン名とIPアドレスを変換する
- SMTP (Simple Mail Transfer Protocol)
 - 電子メールを送信する
- POP (Post Office Protocol)
 - 電子メールを受信する
- HTTP (Hyper Text Transfer Protocol)
 - Webコンテンツを転送する

アプリケーション層での通信相手の識別



- メールを送るにも、Webを見るにも、通信相手を指定する方法が必要。
 - インターネット上でのホストを識別するものとしてインターネット層のIPアドレスを利用することができる。
 - が、IPアドレスは人間には使いにくい。
 - おまけに、ネットワーク構造の変化でIPアドレスは変わることがある。
- アプリケーション層で人間が使うために考えられたのが「**ドメイン名**」



ドメイン名での通信相手識別

- WebのURLの例

<http://www.example.co.jp/~taro/>

<http://jprs.jp/>

- 電子メールアドレスの例

taro@example.co.jp

ui@jprs.co.jp



アプリケーション「DNS」の役割

- ドメイン名を使って様々なアプリケーションが通信を行う
 - しかし、下位層のIPでは、通信相手を特定するためにIPアドレスが必要
- ドメイン名をIPアドレスに変換するアプリケーションとして「DNS」が存在する
 - つまり、DNSは、他のアプリケーションを支える、基本的なアプリケーション

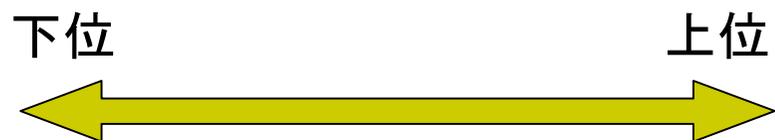


ドメイン名のプロトコル

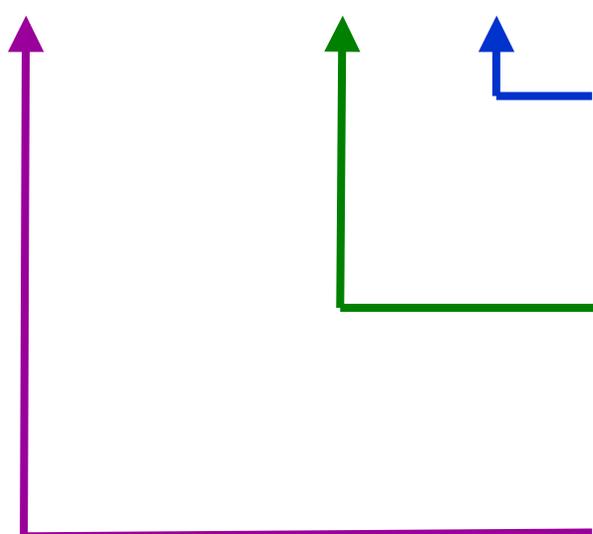
- ドメイン名として使える文字
 - アルファベット、数字、ハイフン
 - 日本語の平仮名・片仮名・漢字など、各国言語文字
- 文字列解釈
 - アルファベットの大文字・小文字は区別されない
- ドメイン名階層
 - ピリオドで区切られた階層構造を持つ
 - 一番右のラベルが最上位、左に行くにしたがって階層が下がる



ドメイン名の階層



example . co . jp



第1レベルドメイン
(TLD: Top Level Domain)

第2レベルドメイン
(SLD: Second Level Domain)

第3レベルドメイン
(3LD: 3rd Level Domain)



DNSの機能

- クライアント・サーバモデル
 - クライアント
 - リゾルバ(多くのOSに組み込まれている)
 - 名前解決要求をサーバに送信する
 - サーバ
 - DNSサーバ(ネームサーバとも言う)
 - ドメイン名とIPアドレスの変換結果をクライアントに返す





DNSの特徴

- DNSサーバは1台じゃない
 - 世界中のホストがDNSを利用するため、1台では耐えられない
 - ドメイン名の階層構造に応じた、分散データベースとして構築されている
- DNSサーバの役割分担
 - コンテンツサーバ
 - ドメイン名とIPアドレスの対応表を持っている
 - もしくは、下位層のDNSサーバのアドレスを持っている
 - キャッシュサーバ
 - リゾルバからの要求を受け、DNS階層をたどって答を得る

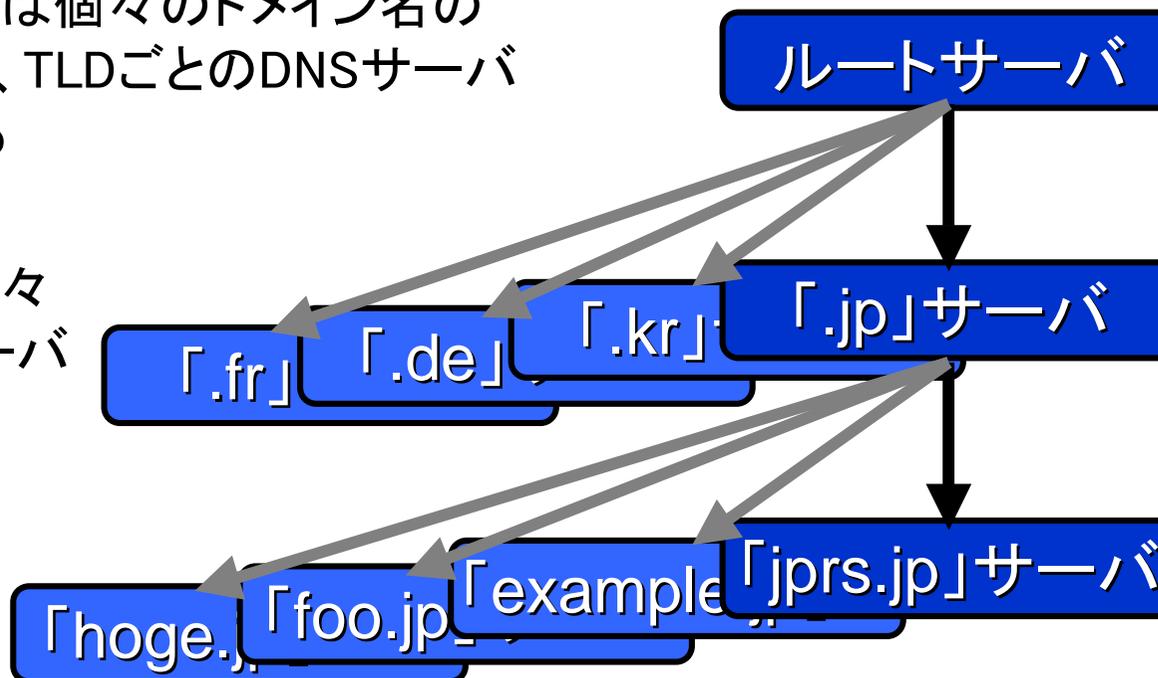


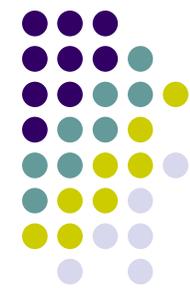
DNSの階層構造

最上位のルートサーバは個々のドメイン名の情報を持っていないが、TLDごとのDNSサーバのアドレスを知っている

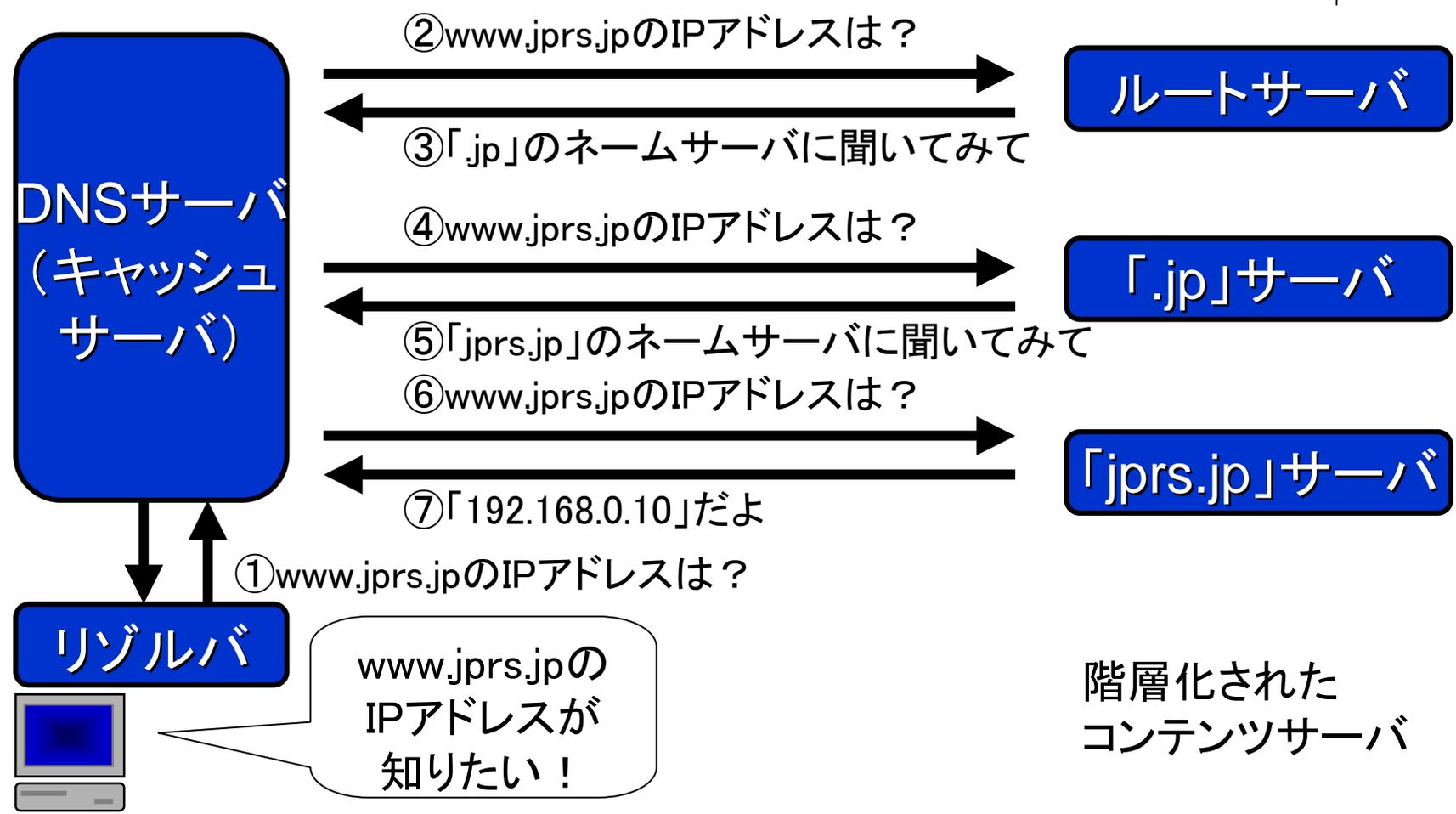
TLDのDNSサーバは個々のドメイン名のDNSサーバのアドレスを知っている

個々のドメイン名のDNSサーバが、IPアドレスなどの情報を持っている



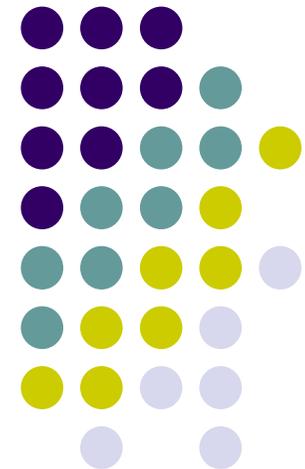


DNSでドメイン名をIPアドレスに変換



ドメイン名とIPアドレスの管理

インターネットの資源管理構造

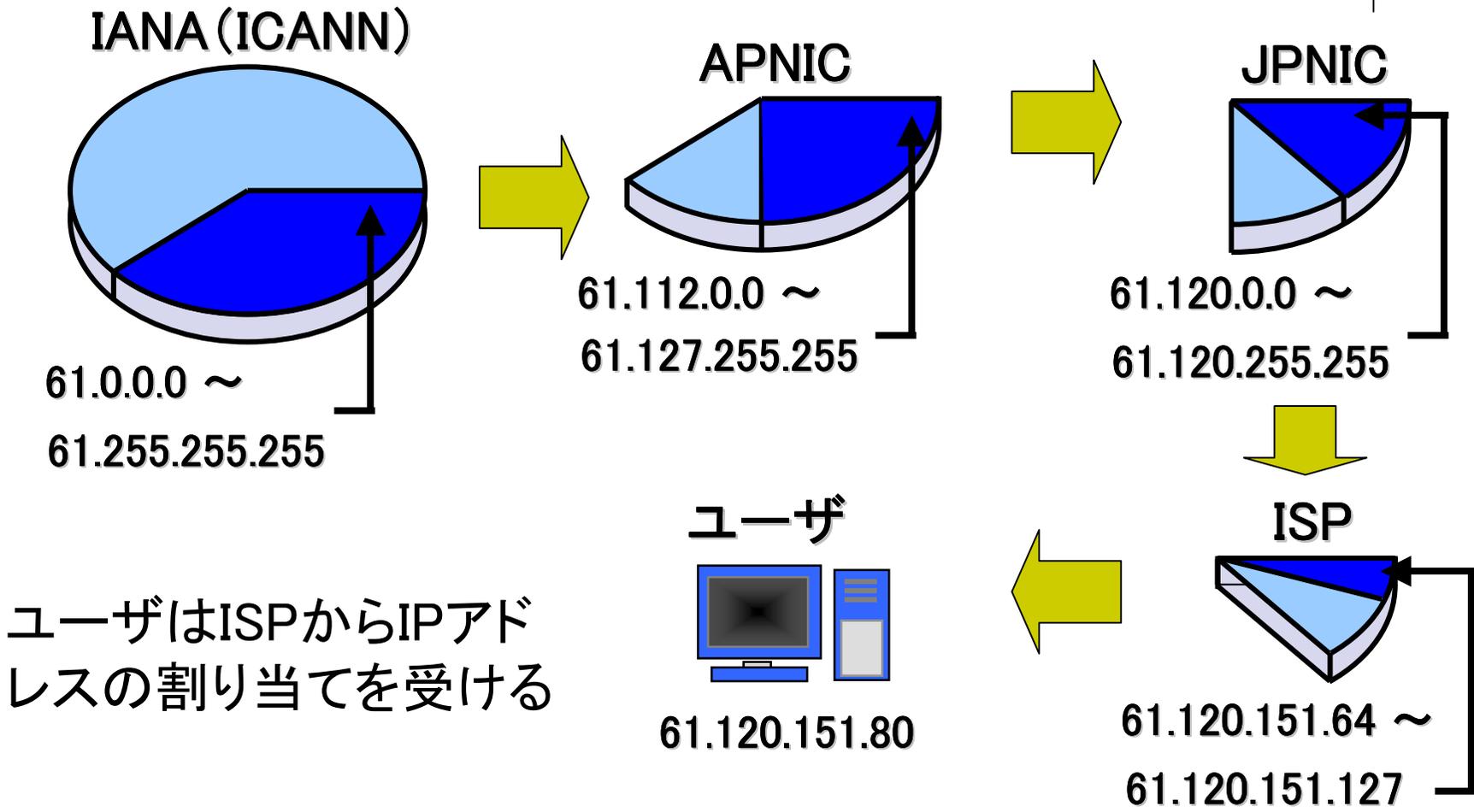


ドメイン名もIPアドレスも 「一意性」が必要



- ドメイン名とIPアドレスは、インターネットの上でホストを識別する名前
 - 同じIPアドレスのホストが複数存在してはいけない
 - 同じドメイン名のホストが複数存在してはいけない
- 同じアドレス・ドメイン名が存在すると、どっちと通信していいのかわからなくなってしまう
- だから、
 - 好きなIPアドレスを勝手に使っていいわけではない
 - 好きなドメイン名を勝手に使っていいわけではない

IPアドレスの管理構造





トップレベルドメイン名 (TLD) の種類

- ccTLD (country code TLD)
 - 世界中の各国に1つずつ定められたTLD
 - 日本は「.jp」
 - 他に、韓国「.kr」、中国「.cn」、ドイツ「.de」、イギリス「.uk」、など
- gTLD (generic TLD)
 - 国とは関係のないTLD
 - 「.com」、 「.net」、 「.org」、 「.info」、 「.biz」など
 - 「.gov」、 「.edu」、 「.mil」はARPANET時代からの名残でアメリカ専用となっている



ドメイン名の管理構造

- 各TLDごとに、その一意性管理を行う「レジストリ」という組織が存在する。
 - TLDが「jp」であるJPドメイン名では、一意性管理をJPRSがレジストリとして担っている。
 - JPドメイン名を使うためには、使いたいドメイン名を指定事業者（ISPなど）を通して申し込むことが必要。
- TLDレベルのDNSサーバはレジストリが運用している

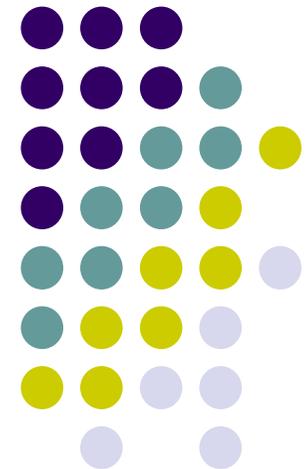


JPドメイン名の種類

- 属性型JPドメイン名
 - 1つの組織で1つだけ登録できる
 - 会社や大学など、組織種別によってSLD(9種)を規定
 - example.co.jp … 民間企業
 - example.ac.jp … 大学、研究機関など
 - example.ne.jp … プロバイダなど
 - example.go.jp … 政府組織
- 汎用JPドメイン名
 - 誰でも、どんな組織でも、個人でも、いくつでも
 - example.jp、jprs.jp、宇井隆晴.jp

アプリケーションはどう動く？

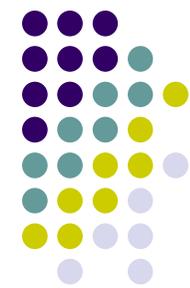
Webブラウザからのアクセスを例に
TCP/IPのおさらい





Webブラウザでホームページを見る

- アプリケーション層から見た一連の動作
 - ユーザが見たいURLを指定する
 - DNSでWebサーバのIPアドレスを調べる
 - Webサーバに接続し、HTTPでコンテンツを要求する
 - WebサーバからHTTPでコンテンツが送られてくる
 - Webブラウザでコンテンツを表示する
- 2つのアプリケーション層の protocols を利用
 - DNS … ドメイン名とIPアドレスを変換
 - HTTP… WebのHTMLコンテンツの送受信



アプリケーション層での動作

①URLを指定する

「http://www.example.co.jp/」

②DNSサーバでIPアドレスを調べる

「www.example.co.jpのIPアドレスは？」

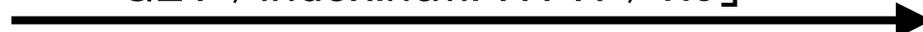


③DNSサーバがIPアドレスを返す

「192.168.0.10 です」

④Webサーバにコンテンツを要求する

「GET /index.html HTTP/1.0」



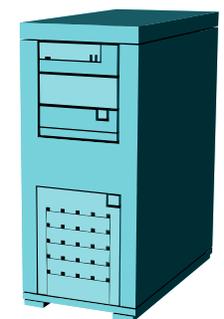
⑥HTMLを表示する

⑤Webサーバがコンテンツを返す

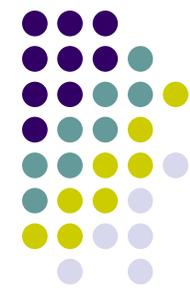
HTML
コンテンツ



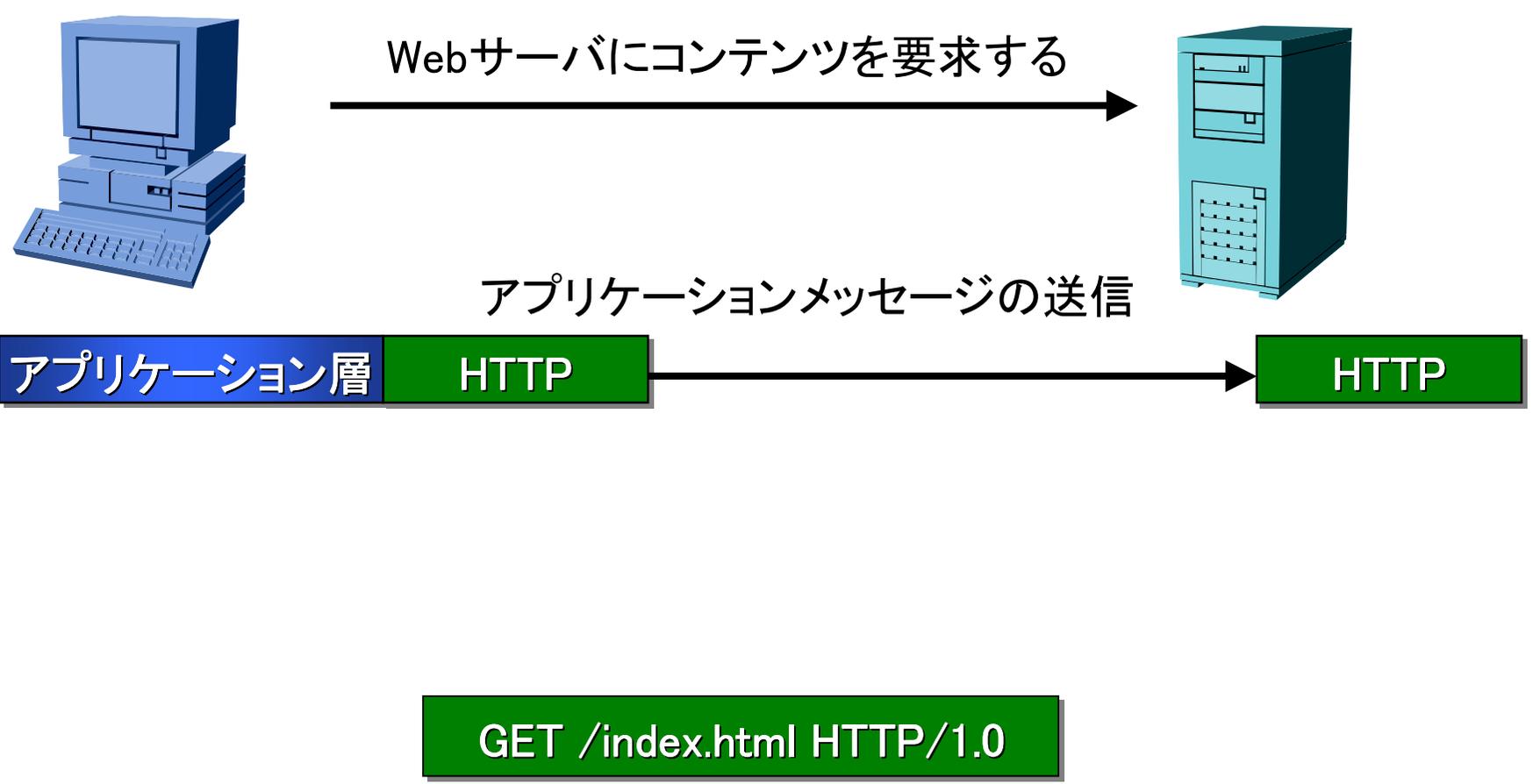
DNSサーバ

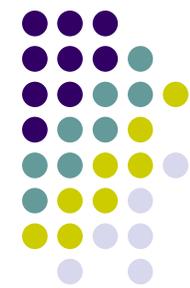


Webサーバ



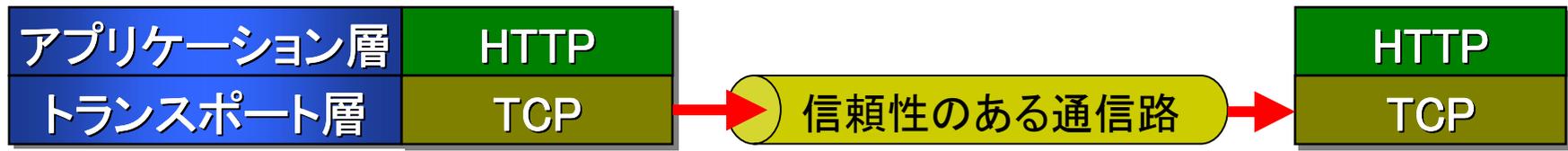
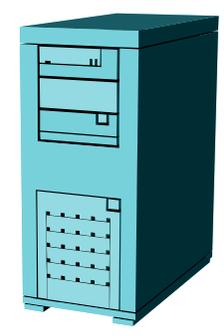
Webサーバへの要求送信





トランスポート層での動作

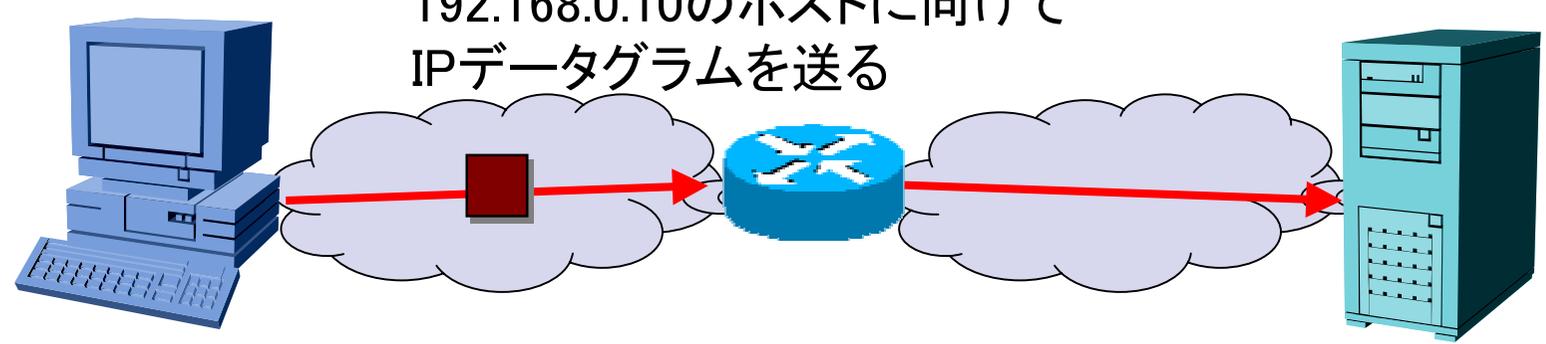
192.168.0.10のTCP port 80番に接続して
TCPセグメントを送る

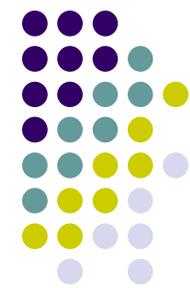




インターネット層での動作

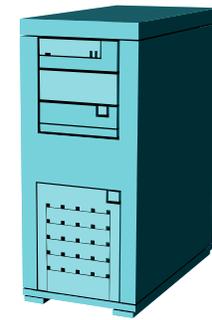
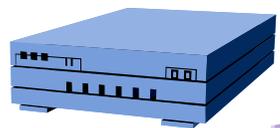
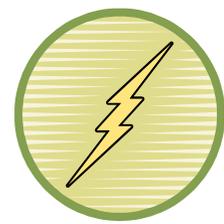
192.168.0.10のホストに向けて
IPデータグラムを送る



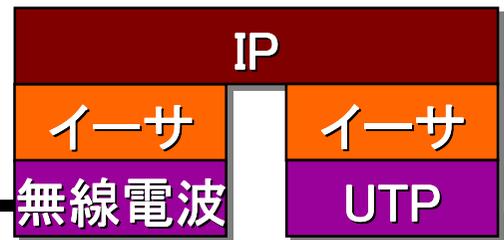


データリンク層・物理層での動作

信号を物理ネットワークを通して伝える



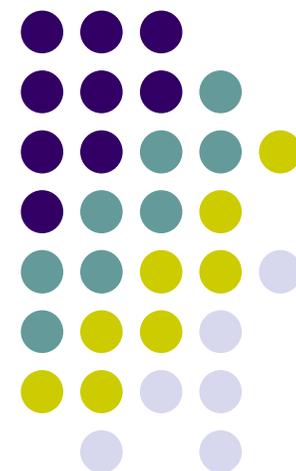
アプリケーション層	HTTP
トランスポート層	TCP
インターネット層	IP
データリンク層	イーサネット
物理層	無線電波



HTTP
TCP
IP
イーサ
UTPケーブル

インターネットと社会との融合

社会基盤としてのインターネット





インターネットでできること

- 性能・技術の進歩
 - 最初は文字のやりとりしかできなかったものが、画像、音声、動画など、様々なメディアを扱うことができるようになった
- 特別なものから当たり前のものへ
 - インターネットは、研究者が使うものから、誰もが生活の中で使うものへ
 - 音楽を聴く、本を読む、テレビを見る、などの行動と同じレベルに「インターネットを使う」



既存の仕組みとの融合、置き換え

- インターネットの上で育ってきたサービス
 - 電子メールやチャットなど
- 既存の媒体をインターネットが置き換えようとしているもの
 - 固定電話とIP電話の関係
 - 郵便と電子メールの関係
 - 出版とオンラインメディア
 - 実生活とオンラインゲーム？



「使う」ものから「使っている」ものへ

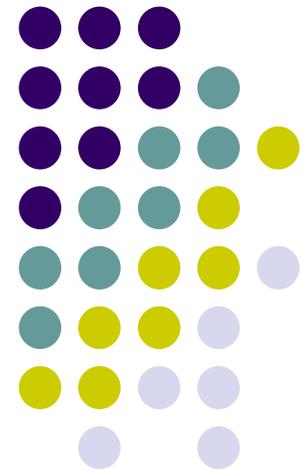
- 今はまだインターネットは意識して「使う」もの
 - 「インターネットでWebを見よう」
 - 「インターネットでメールを送ろう」
- これがいつのまにか「使っている」ものへ
 - 「〇〇さんに電話しよう」
 - 実はその電話、インターネットを使っています
 - 「テレビであの映画をリクエストして見よう」
 - そのサービス、インターネットを使っています



社会基盤としてのインターネット

- 「あると便利なもの・役立つもの」から「なくてはならないもの」へ
- インターネットが生活、企業活動、社会、経済を支える基盤としてなくてはならないものになった時、インターネットは社会基盤(インフラ)になった

おわりに





まとめ

- インターネットは、軍事・学術研究分野から一般に広がり、社会基盤となりました。
- どんなに高度で複雑なアプリケーションでも、インターネットプロトコルの役割は同じです。
- 各層のプロトコルは、それぞれが奥深いものです。また、アプリケーションは次々と新しいものが開発されています。
- 全体像を掴むことで、個々の技術を理解しやすくなります。

