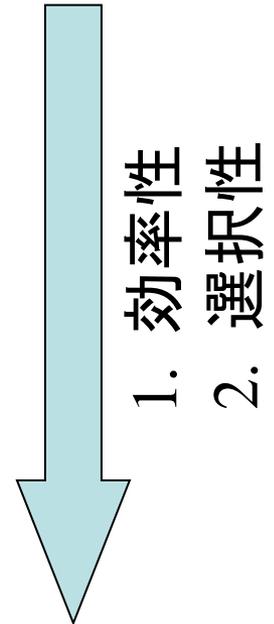


1.2 インターネットの技術的特徴

1. コンピュータの接続の仕方

- 永久的な線を準備する。
 - チャンネル、ケーブル
- 必要な時に(仮想的な)線を準備する
 - 電話、(古い)パケット通信
- データを小包にして送りつける。



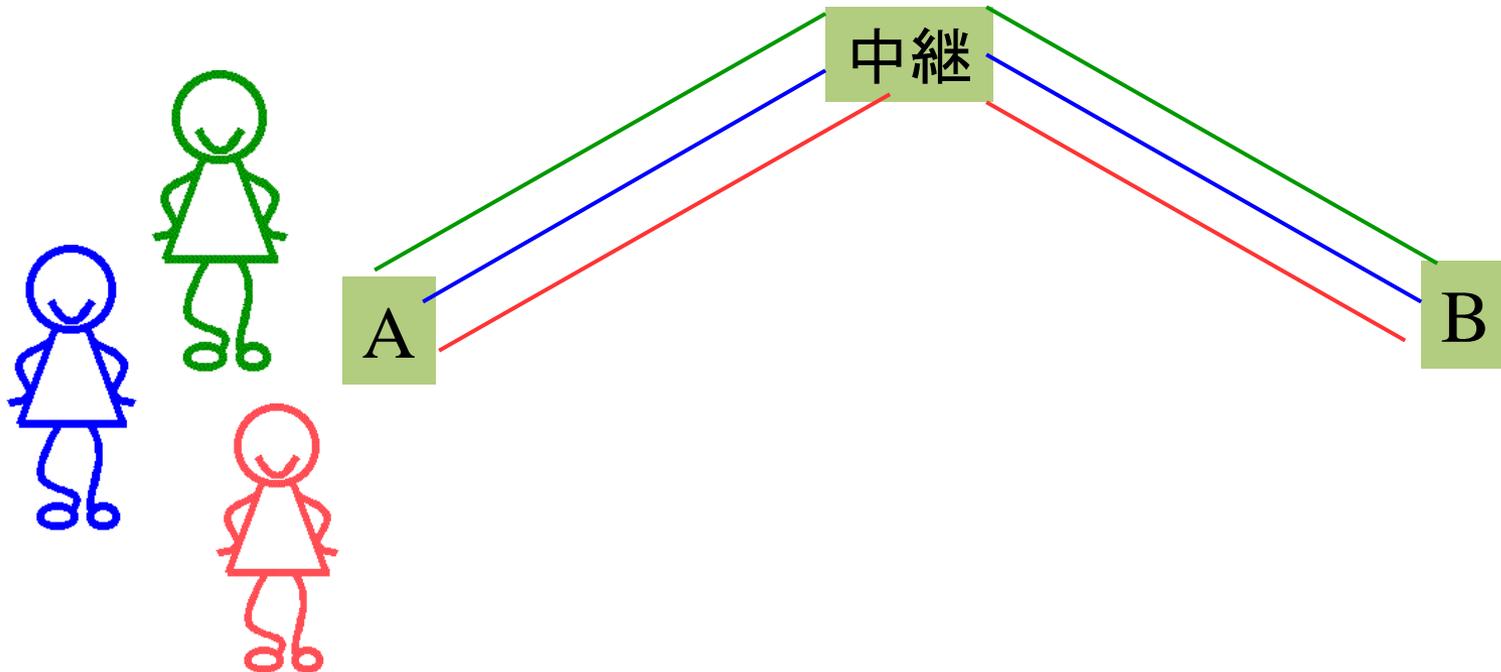
基本的には、この3つの方法の組み合わせ

パケット交換

- 電話網：回線交換
 - 電話局が壊れると通信継続不能
- パケット交換：情報を決められた以下の大きさに分割し、それぞれに宛先をつけて独立に配送
 - 情報の一部が失われても後続には影響なし
 - 中継局が壊れても迂回して通信継続可能

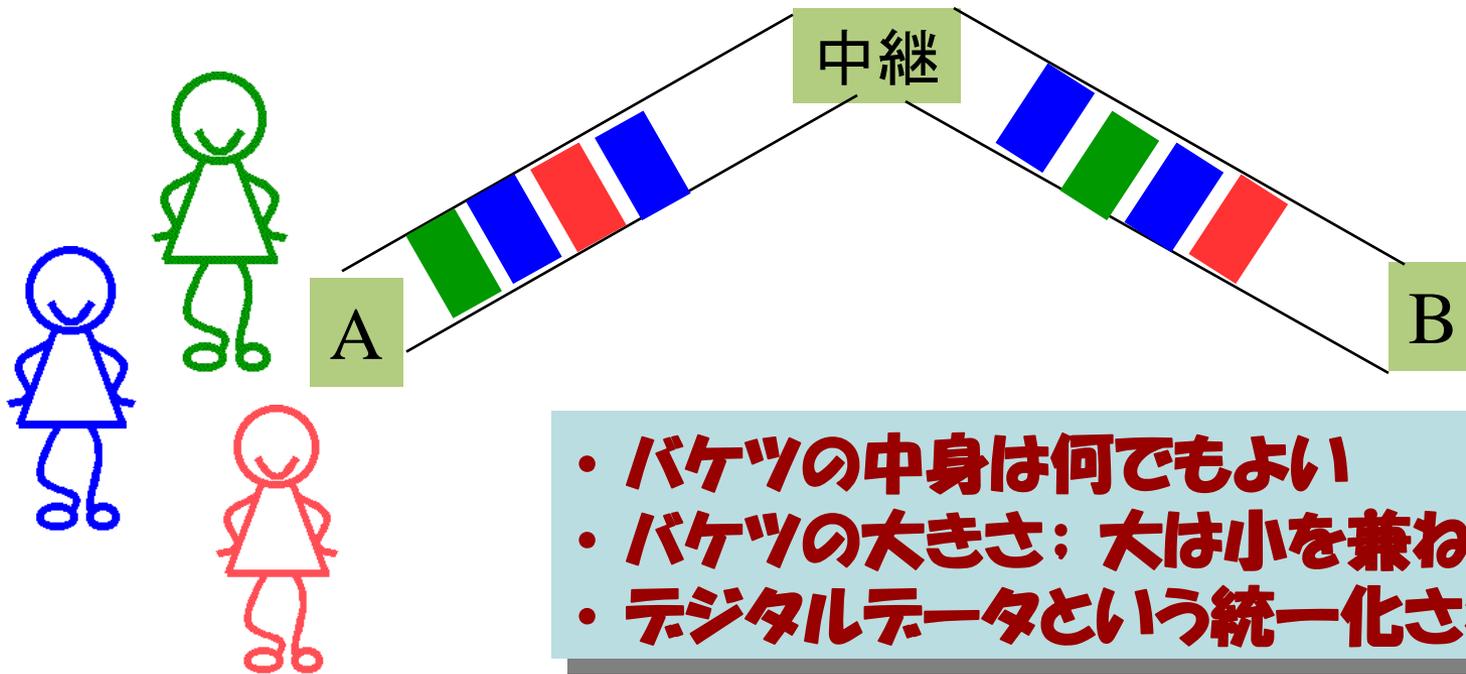
回線交換方式

- 2地点間を回線で結ぶ



パケット交換方式

- 2地点間を**パケット**で結ぶ



- **バケツの中身は何でもよい**
- **バケツの大きさ: 大は小を兼ねる**
- **デジタルデータという統一化された中身**

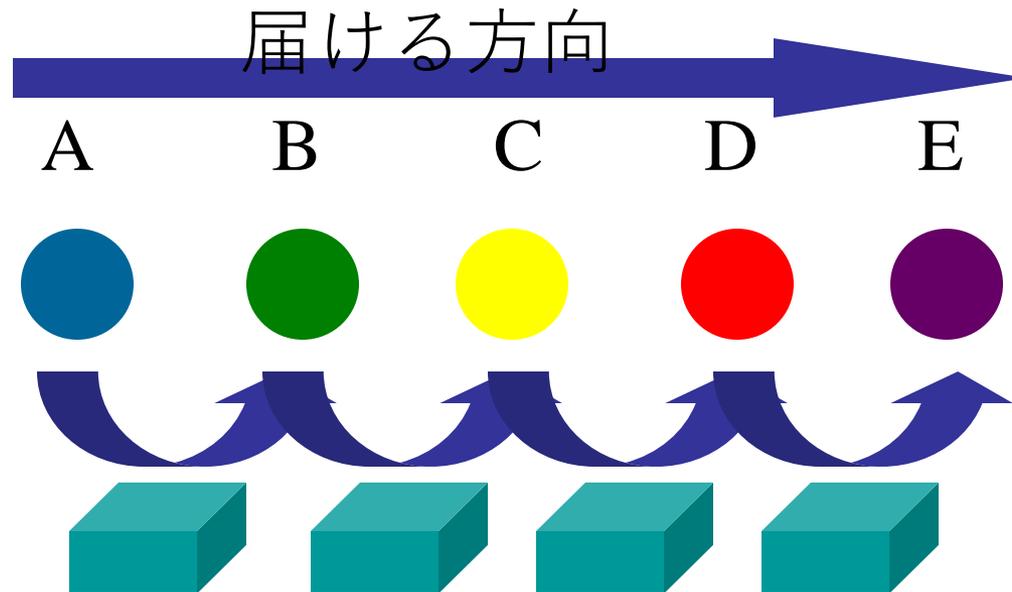
IPパケットを用いた バケツリレーモデル

- バケツリレー
 - 目的
 - バケツに汲んだ水を火のあるところへ
 - 空になったバケツを水のあるところへ
 - 動作
 - バケツを隣の人に渡す



バケツリレーのモデル

- 隣接するエンティティだけが手渡せる
 - 少しずつ目的地に向かって進む
 - 手渡すまでの責任を持てば良い

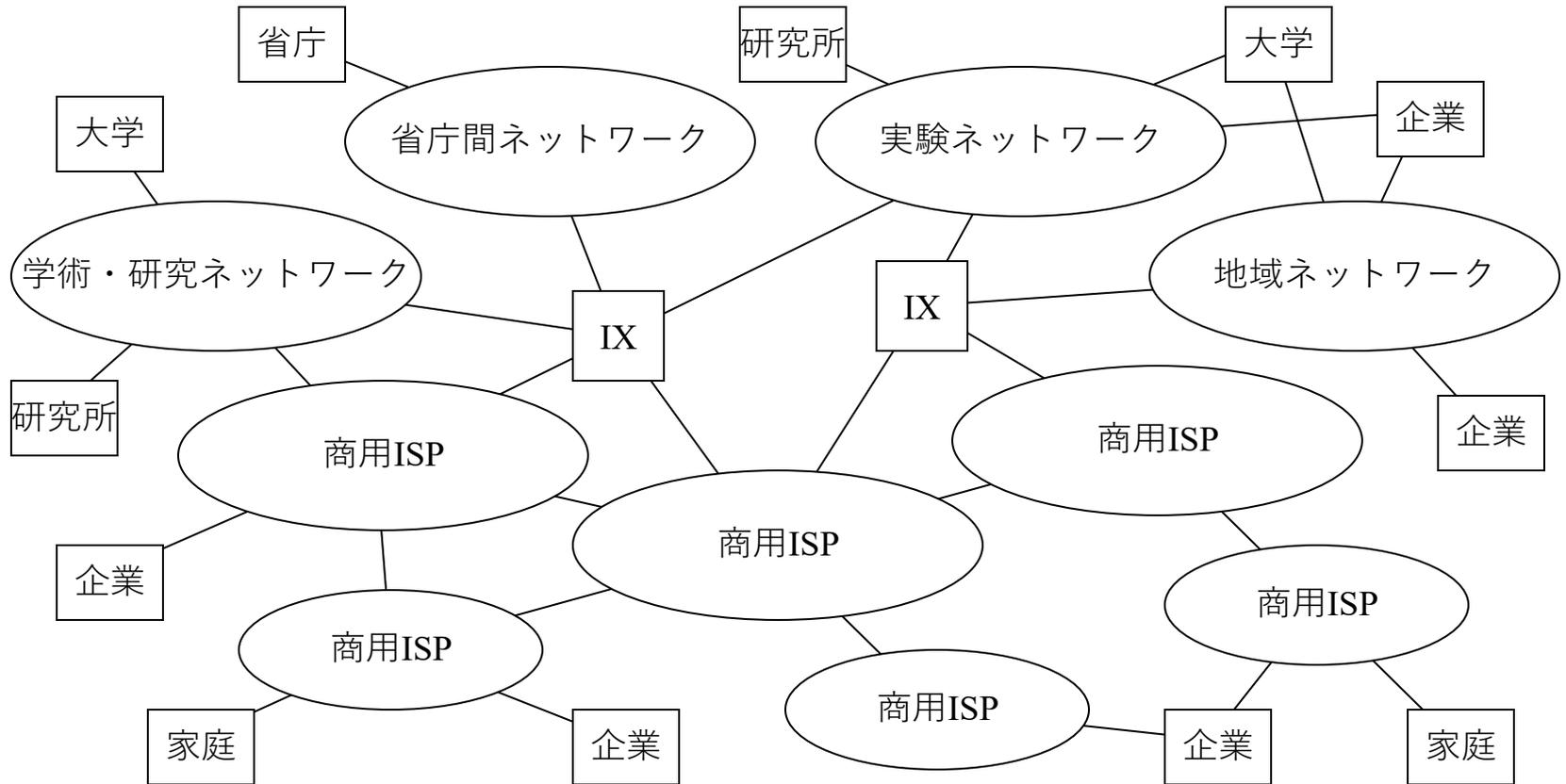


『エンド・エンドには、責任を持たない』

1. 状態管理量が、爆発しない
2. 情報の同期が不要

さて、回線交換 と パケット交換
どっちが、良い？

ネットワークのネットワーク



IX: Internet eXchange

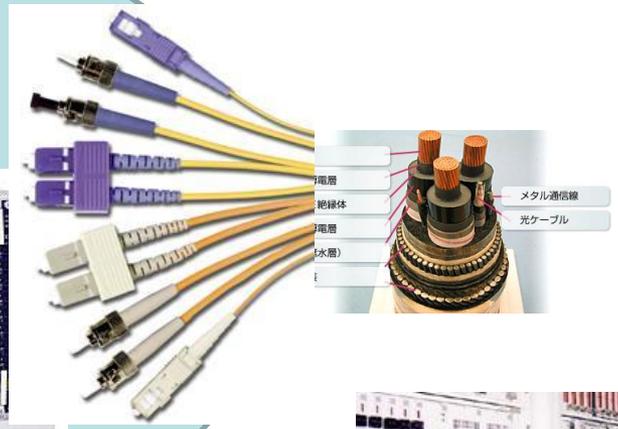
(あとから書かれた)

Architectural Principles of the Internet

- the goal is connectivity,
繋がるのがゴール。
- the tool is the Internet Protocol, and
その武器が IP(Internet Protocol)
- the intelligence is end to end rather than
hidden in the network
知能・知性はエンド(ユーザ)側に！
ネットワークは愚か(Simple/Stupid)でいい

1.3 インターネットの構造

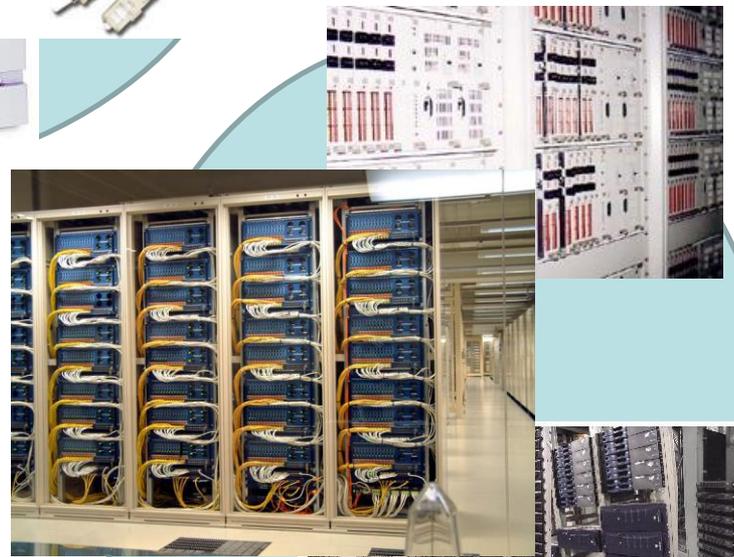
電話局

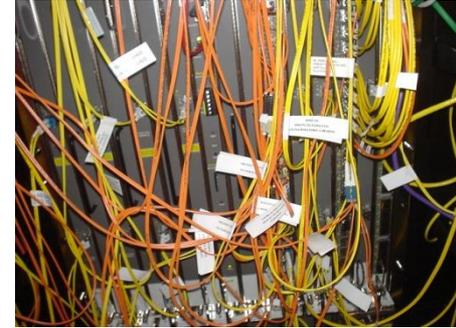


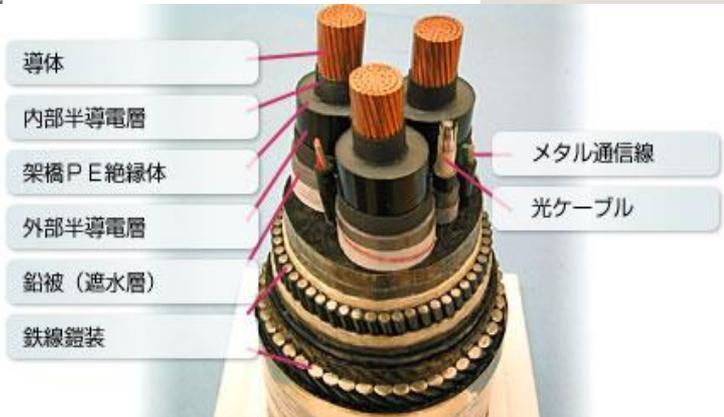
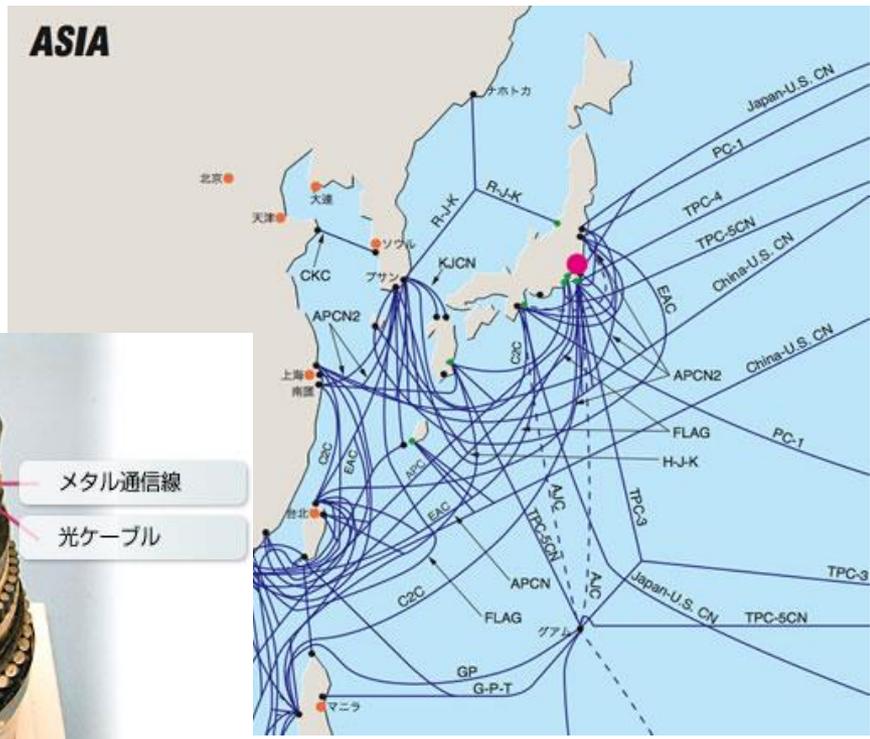
自宅



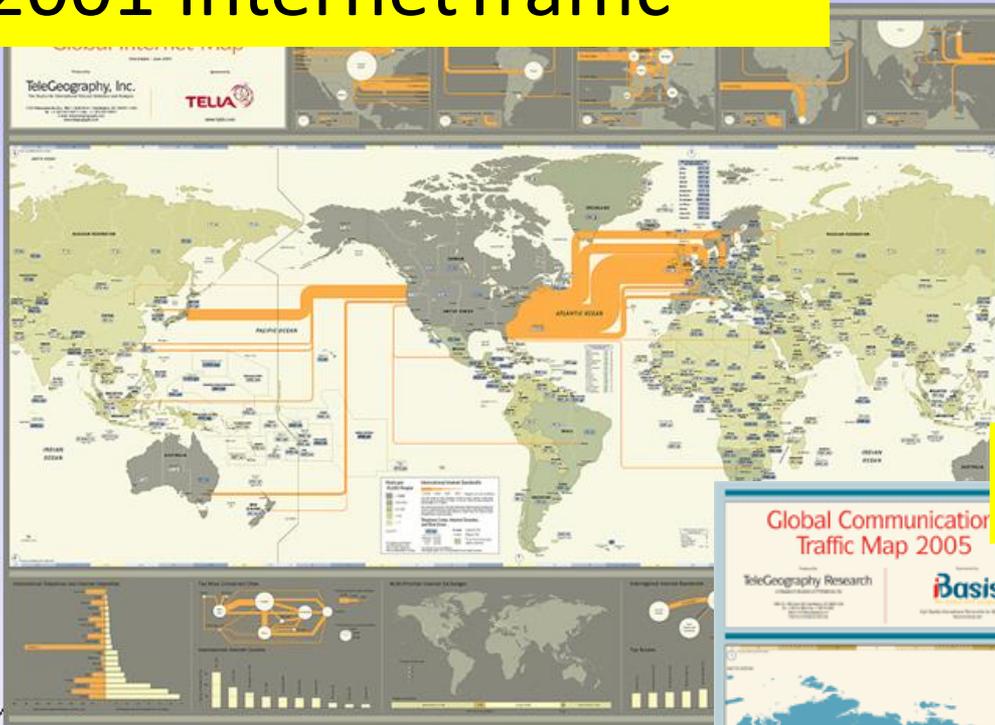
データセンタ





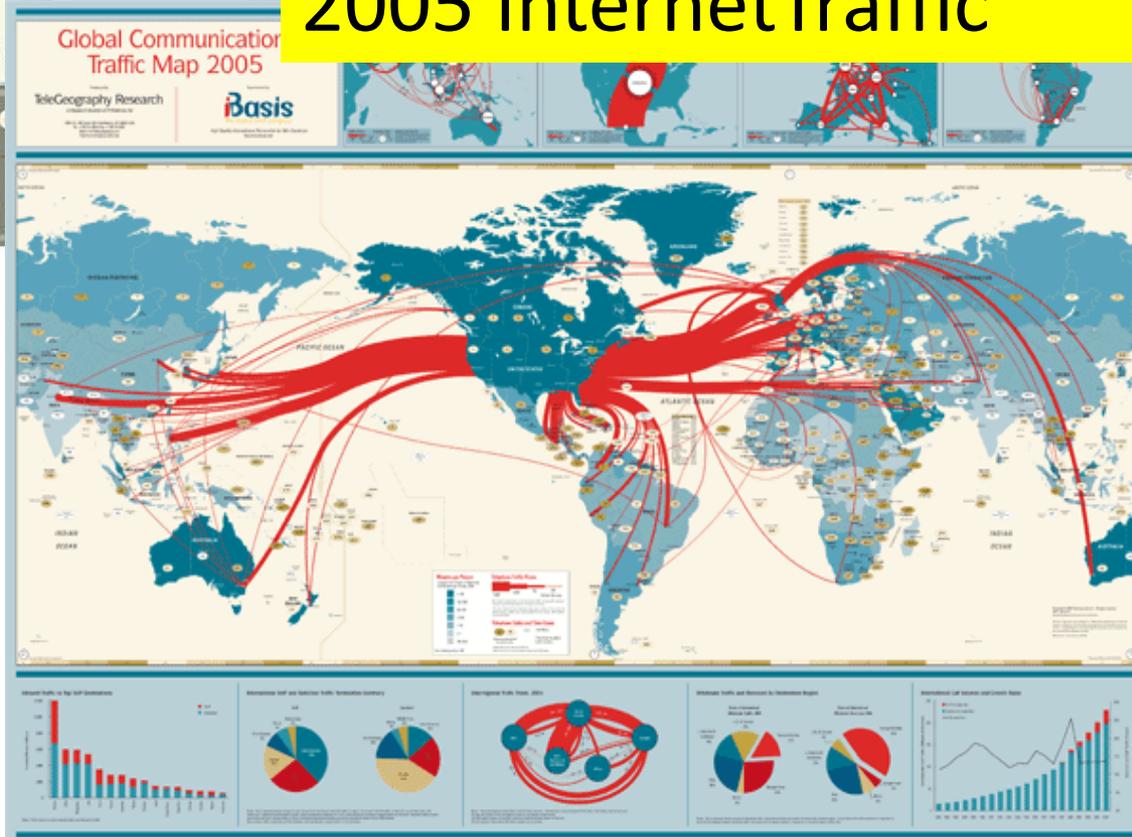


2001 Internet Traffic

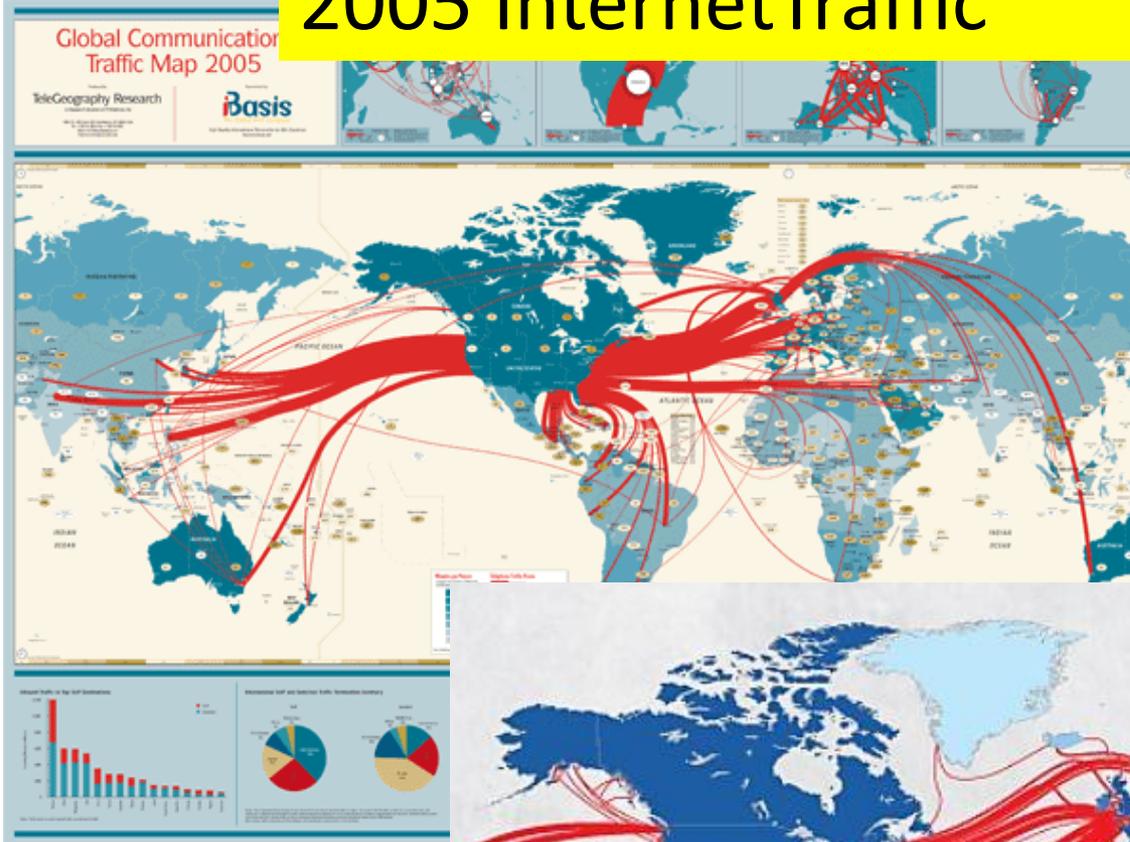


2003

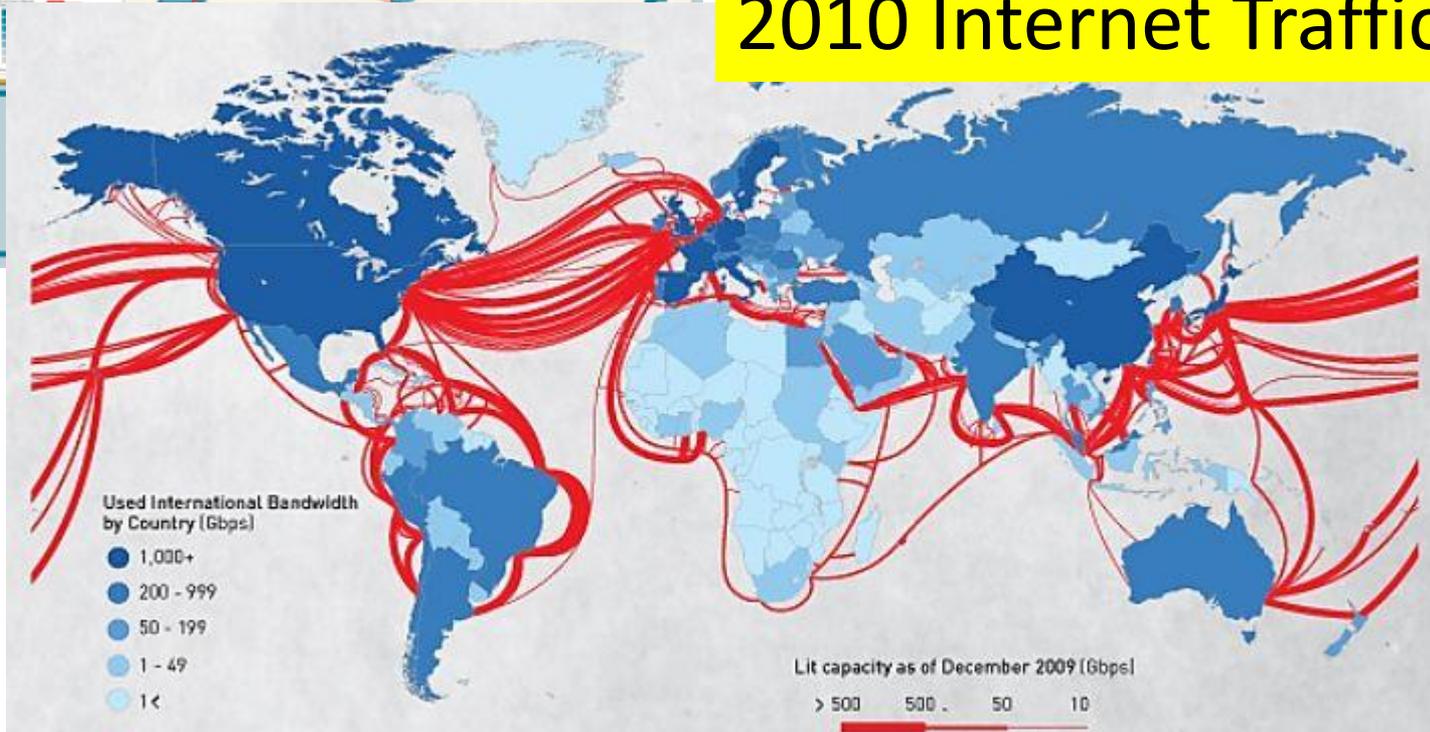
2005 Internet Traffic



2005 Internet Traffic



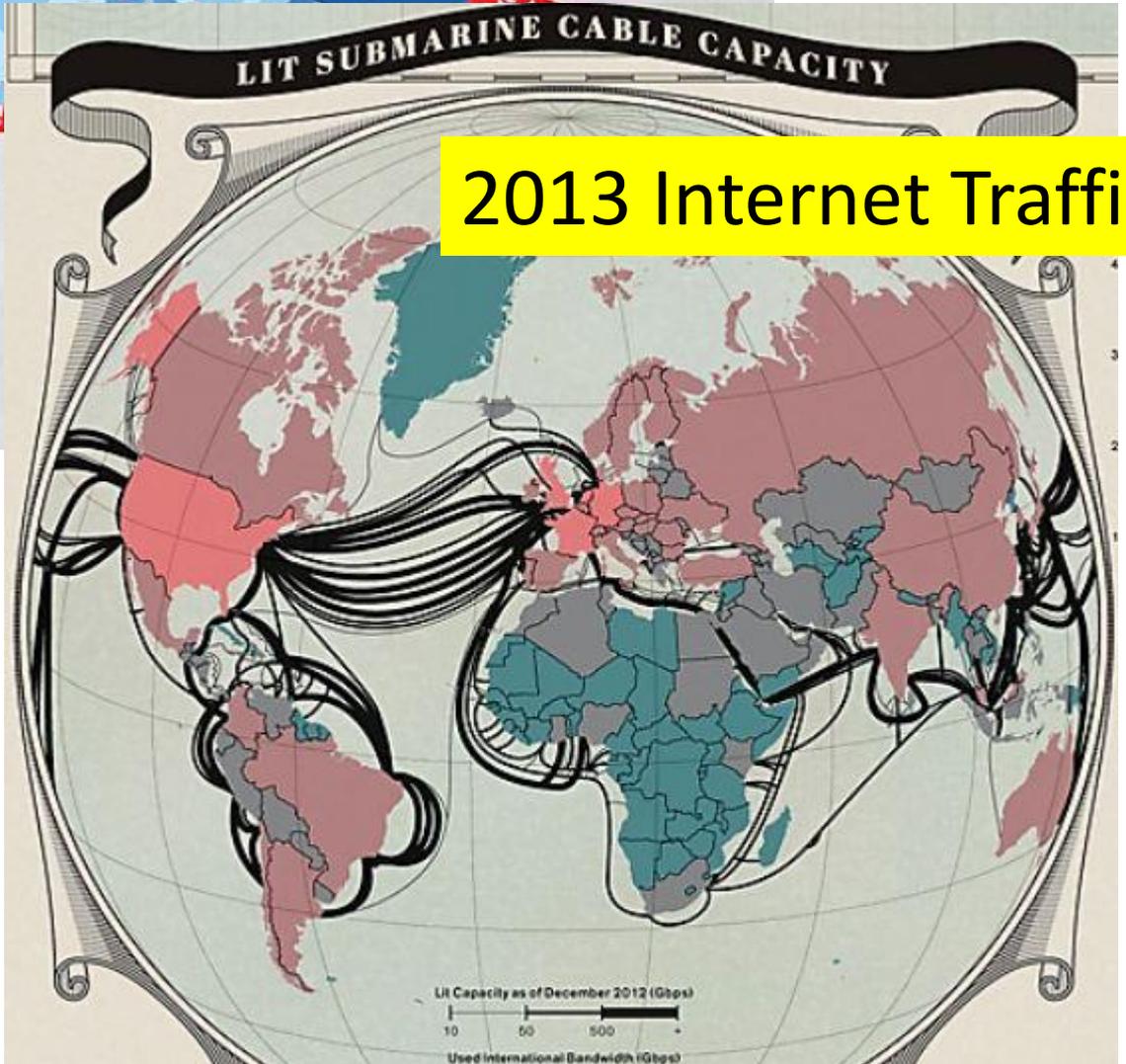
2010 Internet Traffic



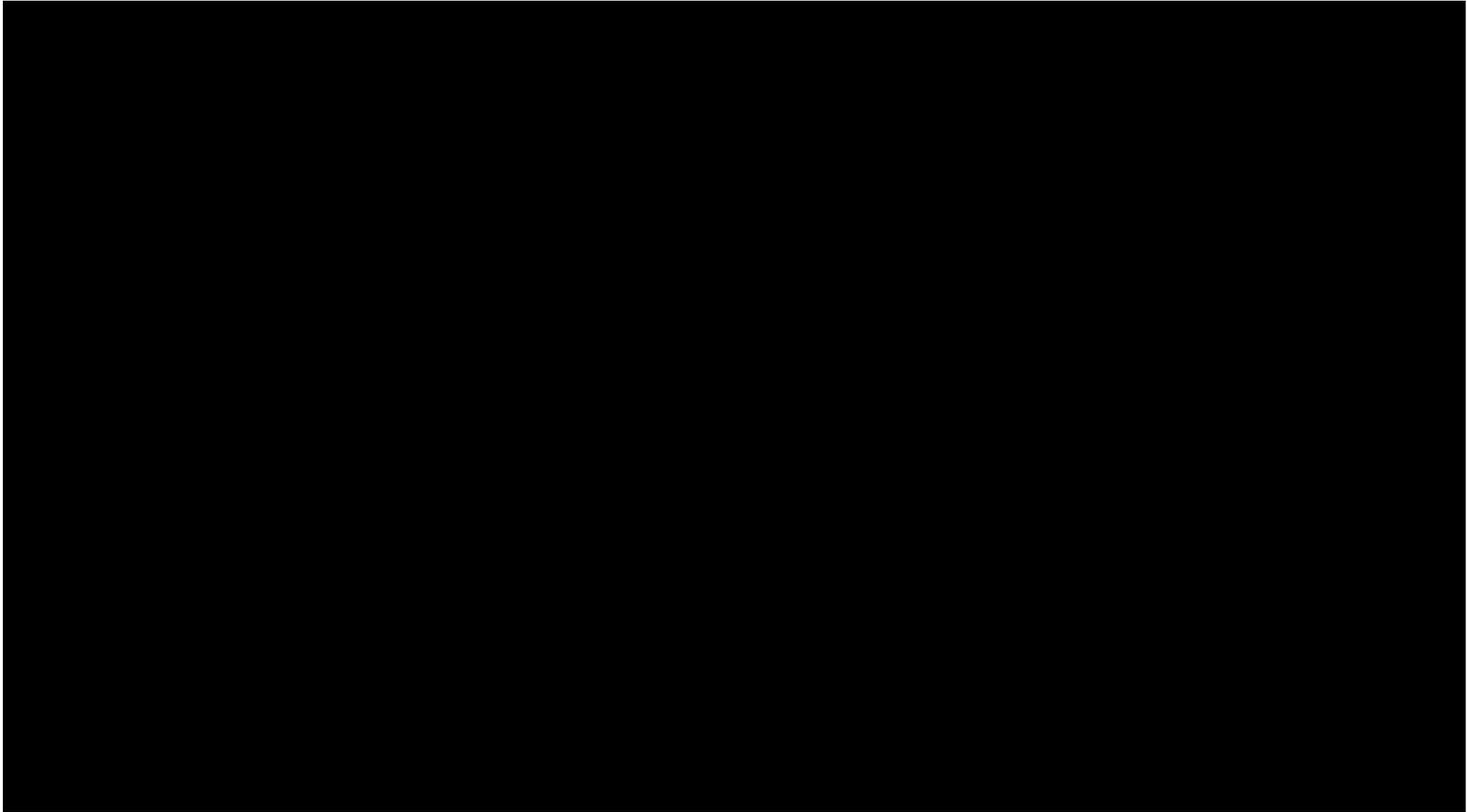
2010 Internet Traffic



2013 Internet Traffic



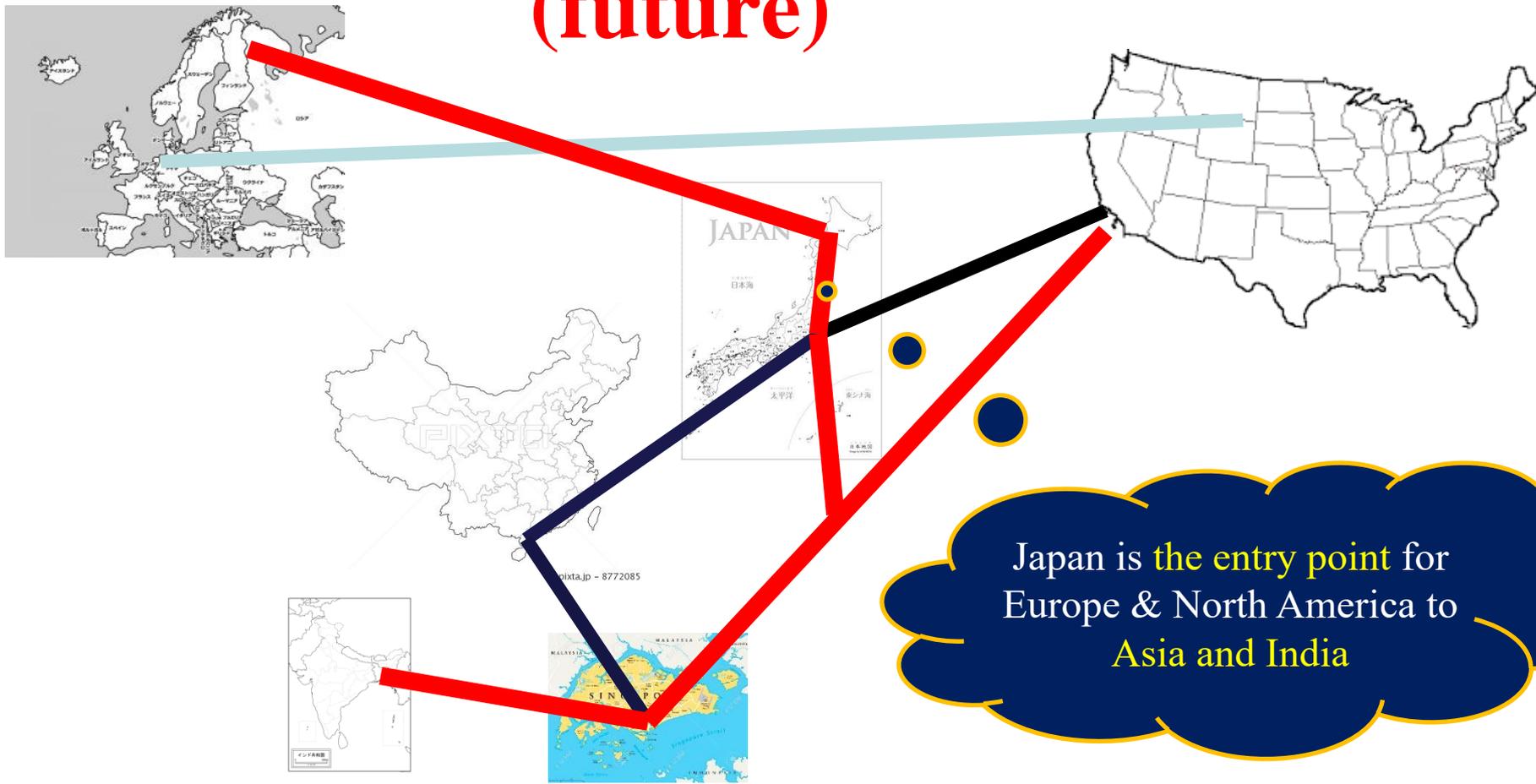
<https://www.visualcapitalist.com/wired-world-35-years-of-submarine-cables-in-one-map/>

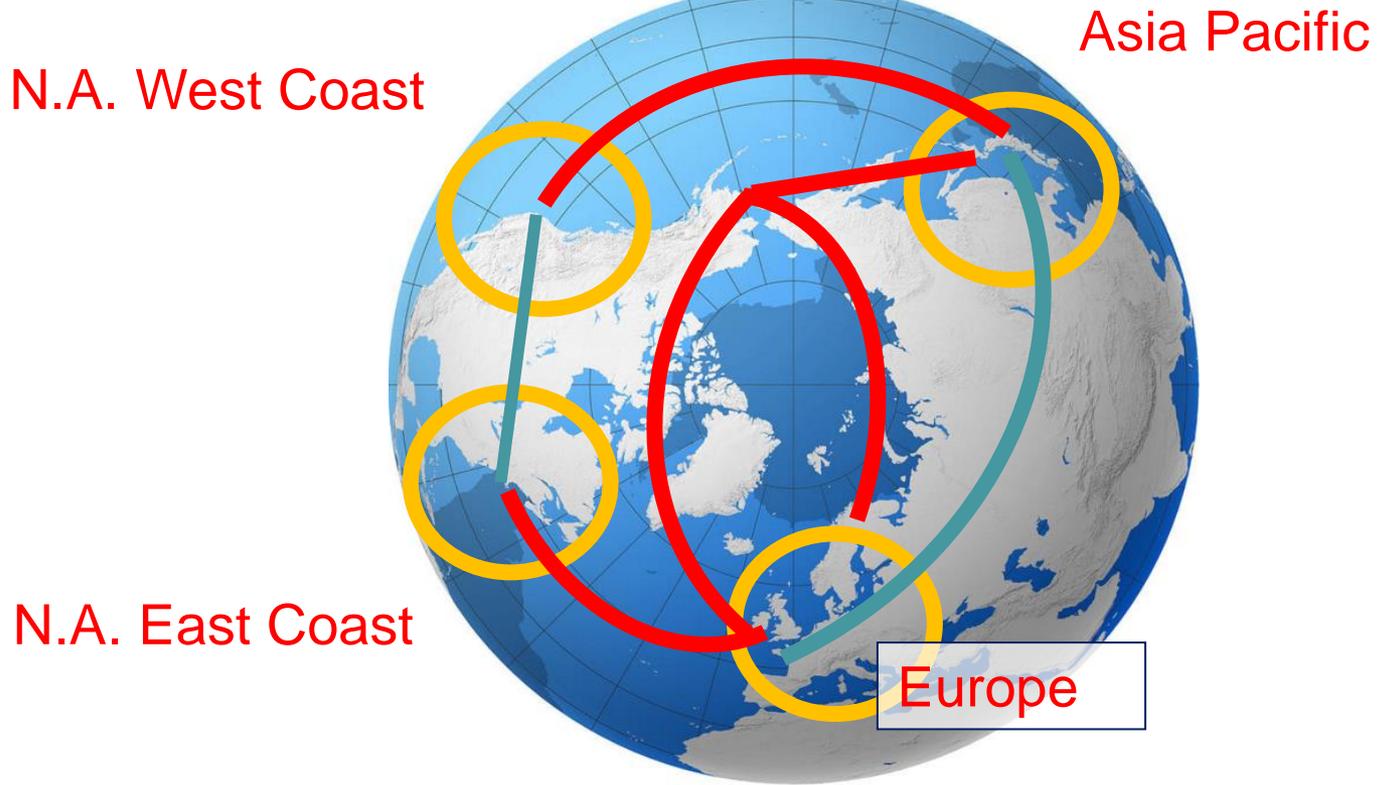


Topology of DC hubs and network (now)



Topology of DC hubs and network (future)





HIGH LEVEL OVERVIEW



CURRENT DESIGN

- The cable transits the Northwest Passage offering a new route in subsea cables
- Significant latency reduction versus routes available today
- 16 fiber pair system
- Landings: dual landings in Japan (Hokkaido and Tokyo region), Europe (Ireland, Norway/Finland) and North America (Prudhoe Bay)
- Branching Units to support remote northern regions

ESTIMATED LATENCIES

Segment	RTD/ms
Japan (Chikura) – Ireland	152
Japan – Norway/Finland	166
Japan – Alaska (Prudhoe Bay)	70
Alaska – Ireland	82
Alaska – Norway/Finland	96

PROJECT TIMELINE, PRELIMINARY:

2022

Supply Contract-in-Force (CIF)

2022

–

2023

Marine Route Survey

2023

–

2025

Cable Manufacturing & Installation

2026

Ready for service



1.4 情報量を表す単位

情報量を表す単位

- ビット[bit]: 0/1の2進数1桁で表すことのできる情報量($=\log_2(\text{場合の数})$)、および実際に0/1で表した桁数
- bps: ビット／秒(=bit per second)
- バイト[byte, B]: 英数字1文字を表すことのできるビット数。現在ではほとんどの場合8ビット(=オクテット[octet])

単位の接頭語(その1)

10^{24}	ヨタ	Y
10^{21}	ゼタ	Z
10^{18}	エクサ	E
10^{15}	ペタ	P
10^{12}	テラ	T

10^9	ギガ	G
10^6	メガ	M
10^3	キロ	k
10^2	ヘクト	h
10^1	デカ	da

単位の接頭語(その2)

10^{-1}	デシ	d
10^{-2}	センチ	c
10^{-3}	ミリ	m
10^{-6}	マイクロ	μ
10^{-9}	ナノ	n

10^{-12}	ピコ	p
10^{-15}	フェムト	f
10^{-18}	アト	a
10^{-21}	ゼプト	z
10^{-24}	ヨクト	y

単位の接頭語(その3)

- コンピュータのメモリ容量を表す時は
 - $2^{10}=1,024$ をキロ
 - $2^{20}=1,048,576$ をメガ
 - $2^{30}=1,073,741,824$ をギガ
 - $2^{40}=1,099,511,627,776$ をテラ
 - $2^{50}=1,125,899,906,842,624$ をペタ
 - $2^{60}=1,152,921,504,606,846,976$ をエクサ

と呼ぶ習慣があり、ディスク容量などにも用いられてきたが、メモリ容量以外では用いられなくなりつつある

- 10^3 をk、 2^{10} をKと書き分けることもある

2. インターネットにおける通信

2.0 階層構造化

階層プロトコル

- 各種の機能を誤り無く作り上げるのは大変
- モジュールを入れ替え可能にしてアップグレードを可能にする。
→モジュール化

cf. パソコンのソフトウェア

アプリケーションプログラム
オペレーティングシステム
デバイスドライバ
ハードウェア

- それぞれのソフトウェアは直下のモジュールの機能だけを使用
- とくに、ハードウェアの差異はデバイスドライバで吸収し、オペレーティングシステム以上は機種非依存

階層プロトコル

アプリケーションプロトコル	
TCP	
IP	
イーサネット	無線LAN

- それぞれの層のソフトウェアは直下の層の提供する機能だけを使用
- 上位層は下位層非依存

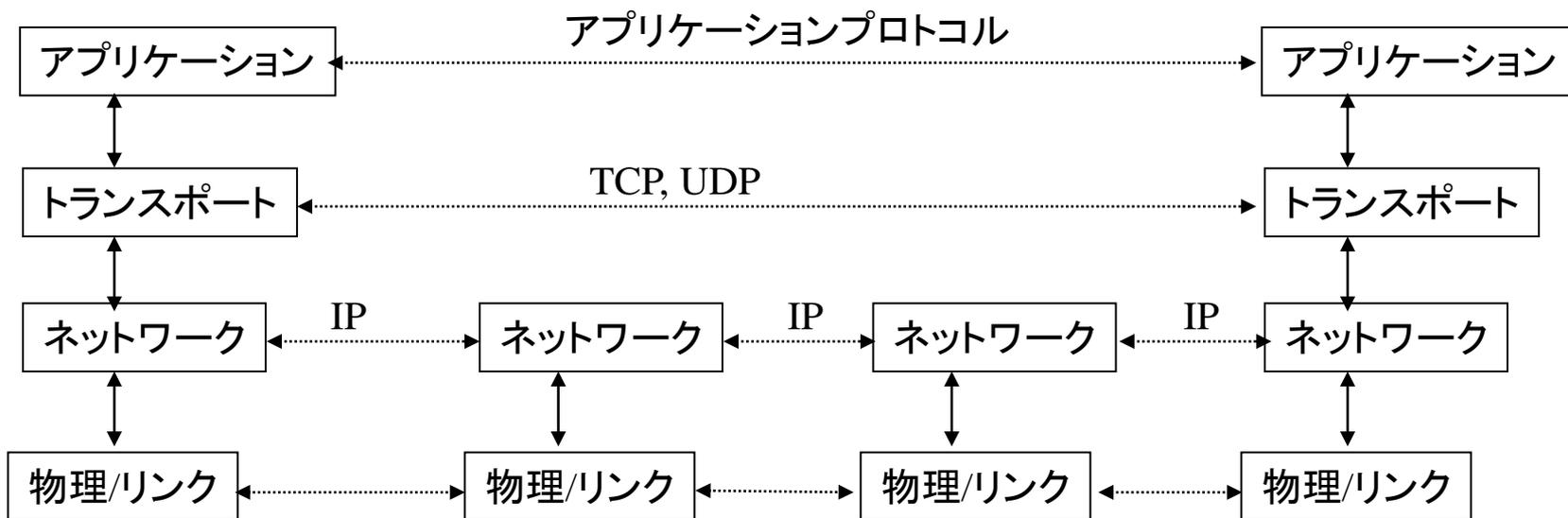


図1-12 TCP/IPの4レイヤモデル

用語の定義

- エンティティ
→ プロトコルを実現するプロセス
- インタフェース
→ 上下層間の呼び出し手順、パラメータなど
- サービス
→ 各層の提供する機能

階層プロトコル

- $\langle N \rangle$ 層は、 $\langle N-1/N \rangle$ インタフェースを通じて提供される $\langle N-1 \rangle$ サービスを用いて $\langle N \rangle$ エンティティ間で $\langle N \rangle$ プロトコルに基づく通信を行うことにより、 $\langle N/N+1 \rangle$ インタフェースを通じてより付加価値の高い $\langle N \rangle$ サービスを $\langle N+1 \rangle$ 層に対して提供する

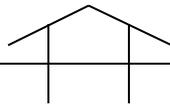
コンピュータA

コンピュータB

<N+1>層

<N/N+1>
インタ
フェース

<N>サービス



<N>層

<N-1/N>
インタ
フェース

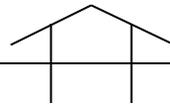


<N>プロトコル

<N>エン
ティティ

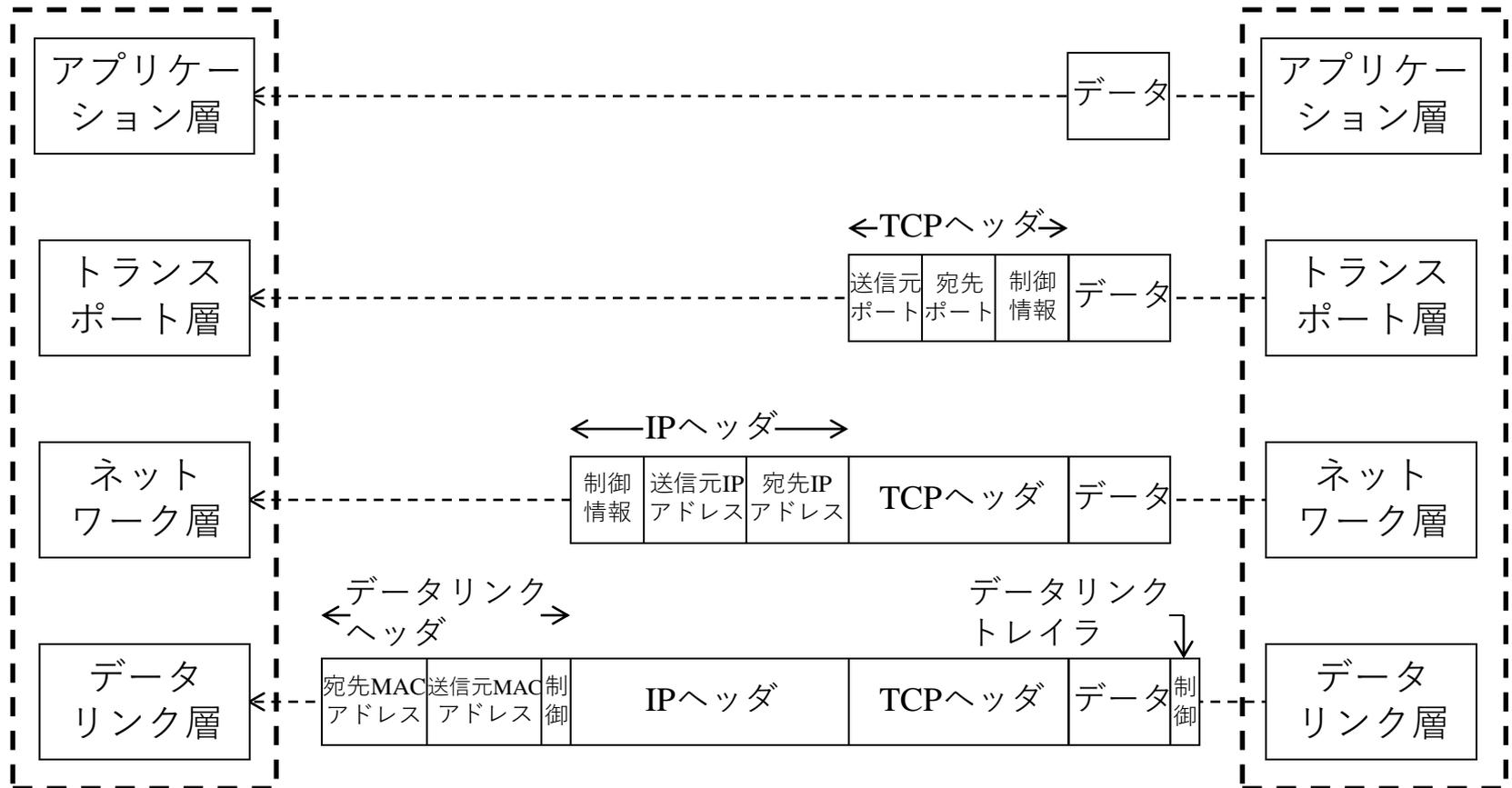


<N-1>サービス



<N-1>層

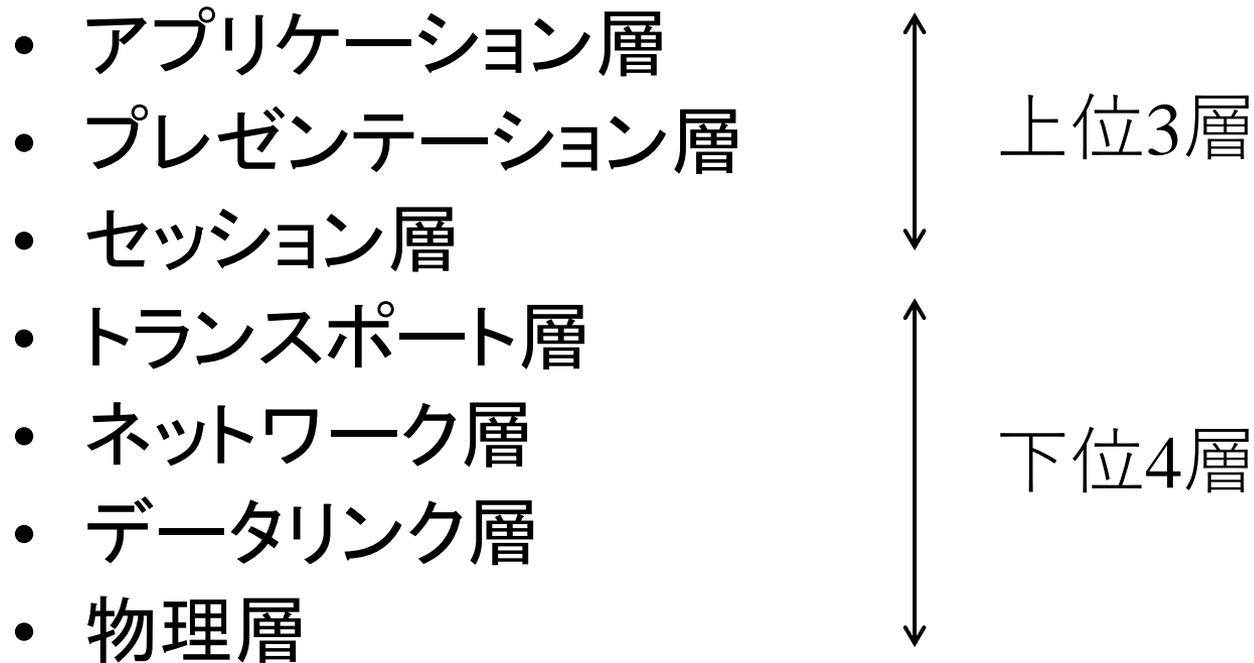
階層プロトコルにおける データ伝送



OSI参照モデル

- ISO (International Standardization Organization: 国際標準化機構) で OSI (Open Systems Interconnection: 開放システム間接続) のためのプロトコルの標準化を行うに当たって、まずプロトコル階層のモデル化を行った

OSI 7層モデル



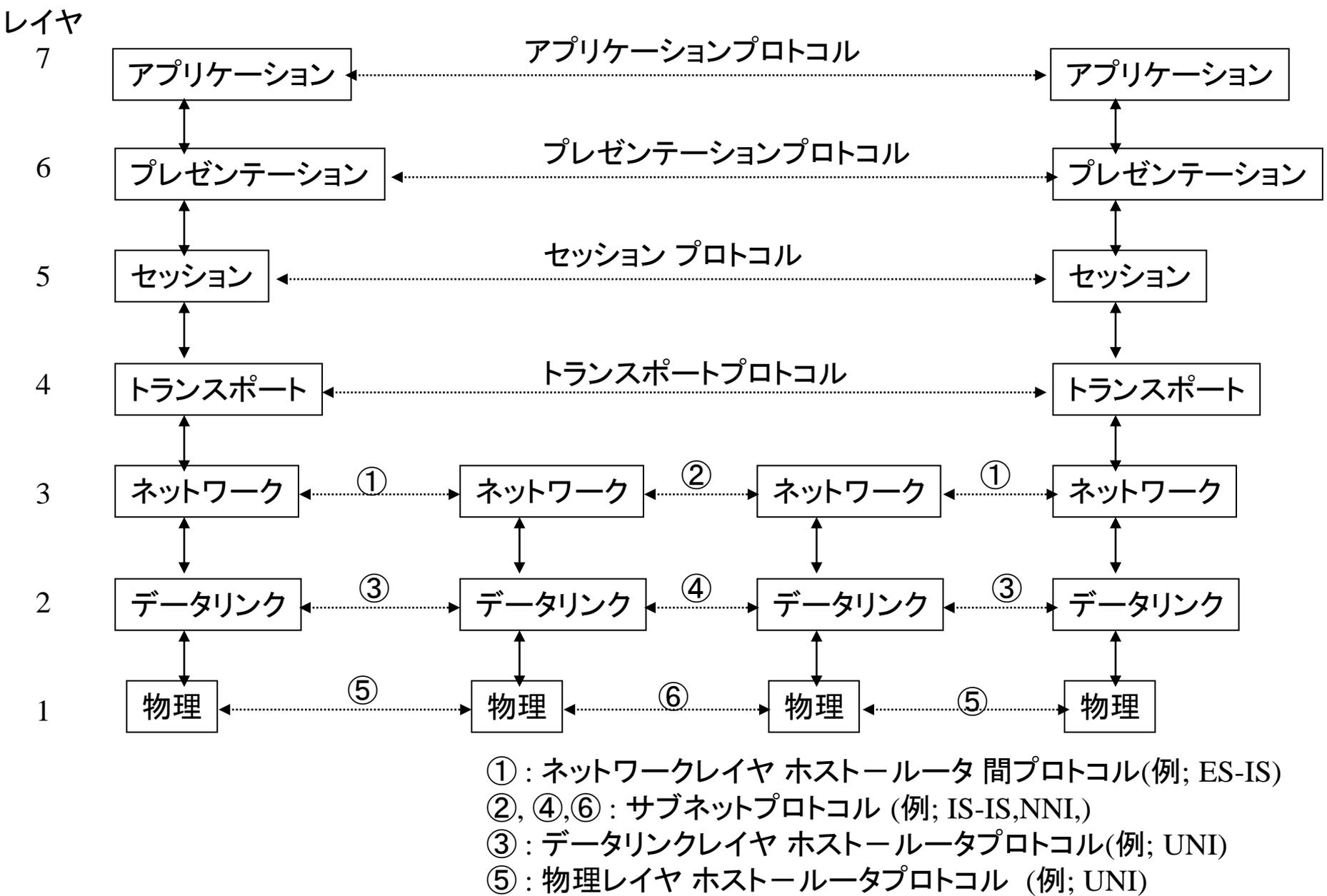


図1-11. OSI 参照7レイヤモデル

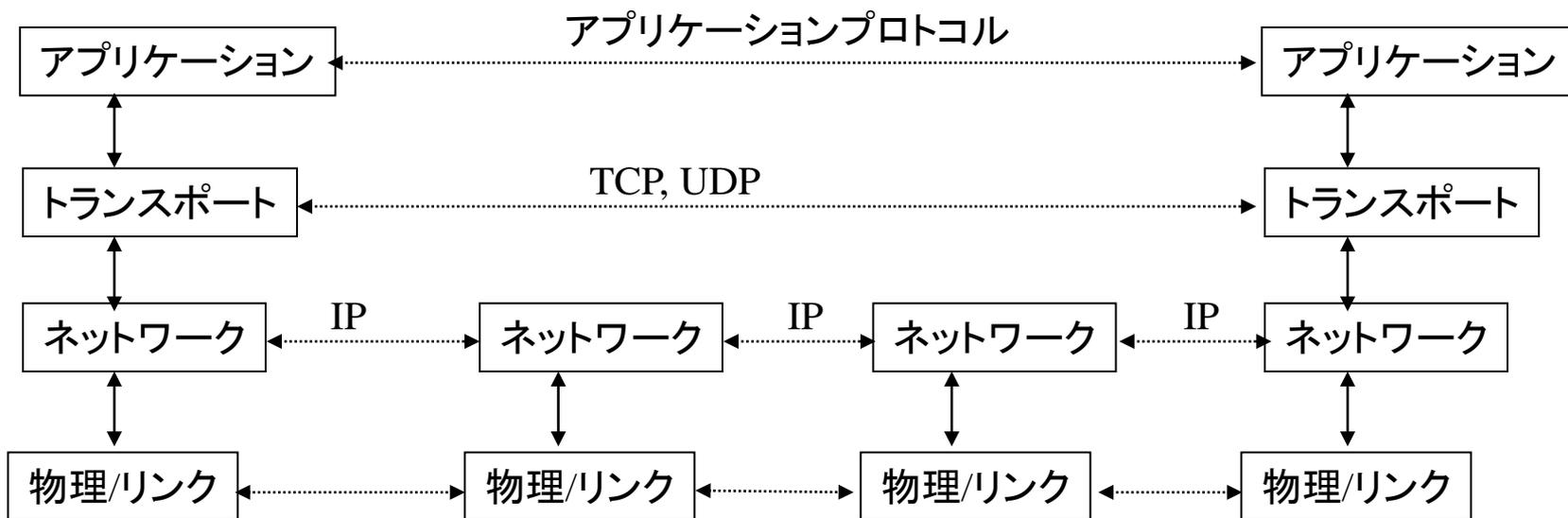


図1-12 TCP/IPの4レイヤモデル

1. 階層化は何のため？

- なぜ、データリンクとネットワークを分けるのか？

→結局は、「選択肢」、「冗長性」の確保

→広い意味での、「(事業)継続性」

(*)BCP; Business Continuity Plan.

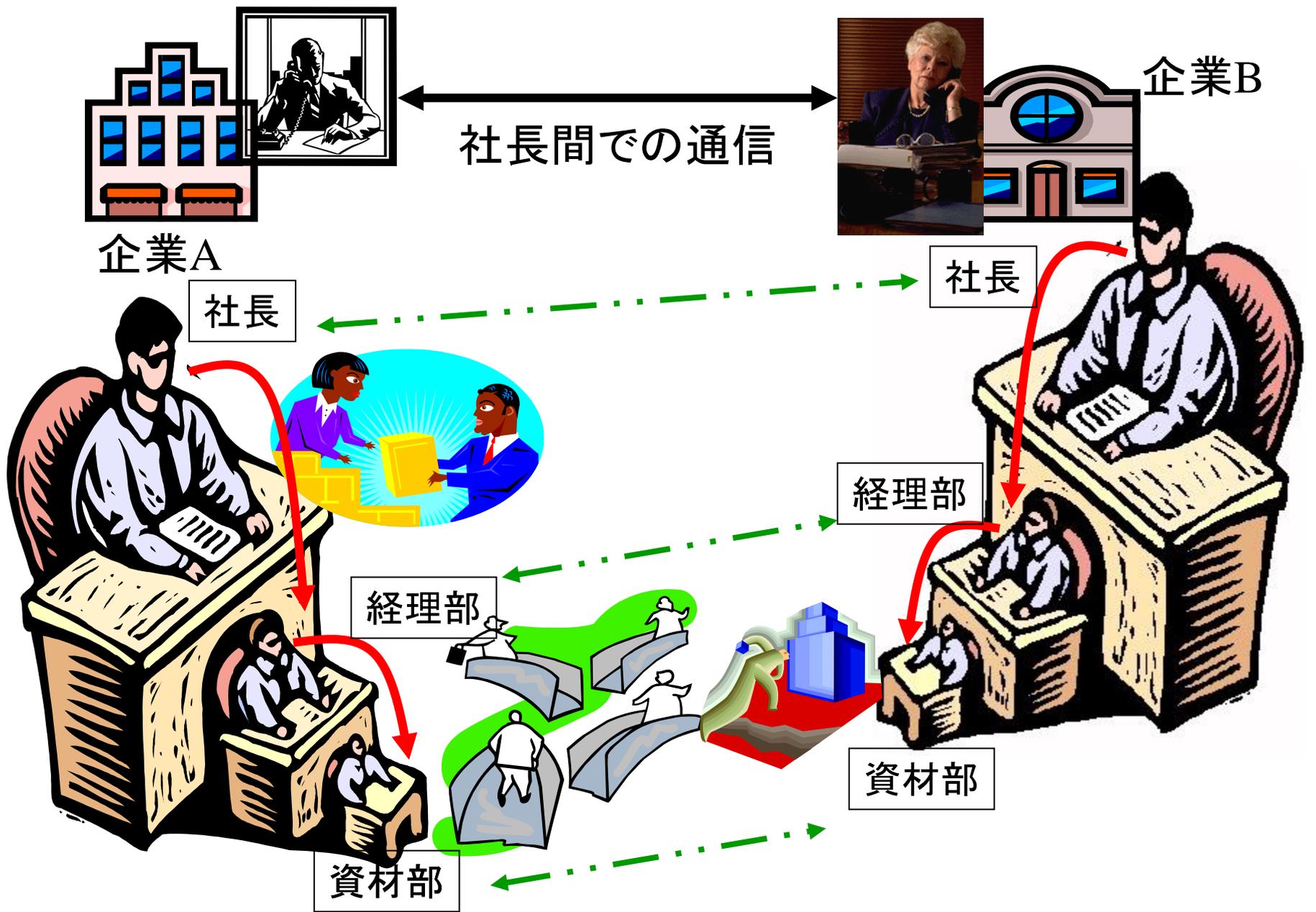


図1-7. 企業間でのメッセージ通信を例にとったレイヤ構造の概念

階層化の嬉しい点

- オープン化
 - 取替え可能
 - 選択肢を与える
 - 競争原理の発生
 - 機器の継続的確保
 - 分担が可能
 - 一人で何でも行う必要がなくなる
 - 選択肢が増える(e.g., Third Partyからの提供)

2.1 通信プロトコル

アプリケーションプログラム間での データの受け渡し

- 同一コンピュータ上
→メモリ上に置いてポインタを渡す
- 近くにあるコンピュータ間
→ファイルに格納して渡す
- 離れたコンピュータ間
→通信回線を経由して送る

通信回線を経由して送る場合の 問題点

- 相手のコンピュータがダウンしている
- 回線上で雑音によりデータが化ける
- 相手のコンピュータが忙しくてデータを取りこぼす
- 送った通りの順序でデータが届かない
- 途中の混雑によりデータが失われる

通信プロトコル

- データを誤りなく(時間内に)届けるために
送り手と受け手の間であらかじめ取り決めておく通信の手順
 - 誤り制御
 - フロー制御
 - 順序制御
 - 輻輳制御

誤り制御

- 送る情報に冗長性を持たせる
 - 誤り訂正
 - ex. 3連続送信
 - 誤り検出→再送
 - ex. パリティ、CRC (Cyclic Redundancy Check)

誤り制御方式の比較

- メタル回線、無線では誤りはバースト的に起こる
 - 誤り訂正のためには大きな冗長性が必要
 - 多くの場合は誤り検出／再送の方が適当

誤り制御方式の比較(続き)

- 再送のオーバーヘッドがデータを送る手間と比べて大きい場合には誤り訂正の方が良い場合もある

ex. 光ファイバ、衛星

→帯域・遅延積(=回線上に乗るビット数)

フロー制御・輻輳制御

- ある程度データを送ったところで、しばらく待つ
→ウィンドウ方式、クレジット方式
- データを送るたびに少しずつ待つ
→レート制御方式

物理層

- 電気的特性、変調方式、コネクタ形状など
ex. RS-232C、V.90

データリンク層

- 直接回線で接続されているコンピュータ間で(誤り無く)データを届ける
 - 媒体アクセス制御(MAC)副層
 - 回線が共有されている場合に媒体へのアクセスを調停する
 - リンクレベル制御(LLC)副層
 - 誤り制御、フロー制御

ネットワーク層

- 宛先コンピュータに至る経路が複数ある場合に適切にデータを送り届ける
 - 経路制御
 - 経路情報の交換
 - 経路選択
 - 輻輳制御

トランスポート層

- 送信元コンピュータから宛先コンピュータまでエンド・エンドで正しくデータを送り届ける
 - 誤り制御、フロー制御
 - 順序制御
 - 輻輳制御

セッション層

- セッション管理
- 同期制御
 - チェックポイントイング

プレゼンテーション層

- 同じ「意味」のデータを届ける
 - 文字コード
 - ASCII/EBCDIC、JIS/Shift JIS/EUC
 - データ表現
 - 1の補数/2の補数、浮動小数点形式
- (暗号化)

アプリケーション層

- 複数のアプリケーションで共通に使うことのできる機能
 - ファイル転送
 - 仮想端末(画面制御)
 - 電子メール、電子掲示板
 - 名前サービス、時刻合わせ
 - WWW (World Wide Web)
 - ネットワーク監視・管理

TCP/IPのプロトコル階層

