インターネットインフラ特論
１３．分散情報システム設計におけるエンドツーエンド定理

太田昌孝

mohta@necom830.hpcl.titech.ac.jp

ftp://chacha.hpcl.titech.ac.jp/infra13.ppt

原論文(End to End Argument in System Design)のエンドツーエンド論法

- The function in question can completely and correctly be implemented only with the knowledge and help of the application standing at the end points of the communication system. Therefore, providing that questioned function as a feature of the communication system itself is not possible. (Sometimes an incomplete version of the function provided by the communication system may be useful as a performance enhancement.)

The function?

- 原論文での例は
  - careful file transfer
  - encryption
  - duplicate message detection
  - message sequencing
  - guaranteed message delivery
  - detecting host crashes
  - delivery receipts
- どれも、送受信者間の情報のやりとり
  - 相手の状態を知るか変えるかが、funciton

the knowledge and help of the application standing at the end points of the communication system?

- アプリケーションとは？レイヤ構造は？
  - In a system that includes communications, one usually draws a modular boundary around the communication subsystem and defines a firm interface between it and the rest of the system.

  と、送受信者とcommunication systemとの界面は1つ（レイヤはない）でfirm
- レイヤリングは際物扱い
  - It is fashionable these days to talk about "layered" communication protocols

a firm interface between it and the rest of the system?

- 界面は固定的で、オプション機能などは認めていないという意味か？
  - First, since the lower level subsystem is common to many applications, those applications that do not need the function will pay for it anyway. Second, the low-level subsystem may not have as much information as the higher levels, so it cannot do the job as efficiently

  と、低いレベルのサブシステムにおかれた機能は全てのアプリケーションが使わないといけない
- 低いレイヤに十分情報があれば、そこでfunctionはcompletely and correctlyに実現できそう
  - TCPによる信頼性あるバイトストリーム等

the knowledge and help of the application standing at the end points of the communication system?

- Communication systemの内部のプロトコルへの議論の当てはめは、このままではできない
  - 一般の分散システムへの拡張が必要
  - 論文では
    - We begin by considering the communication network version of the argument.
    - In a broader context the argument seems to apply to many other functions of a computer operating system, including its file system. Examination of this broader context will be easier if we first consider the more specific data communication context, however.

    と、難しそうだからか一般論には踏み込んでいないが、、、実は簡単

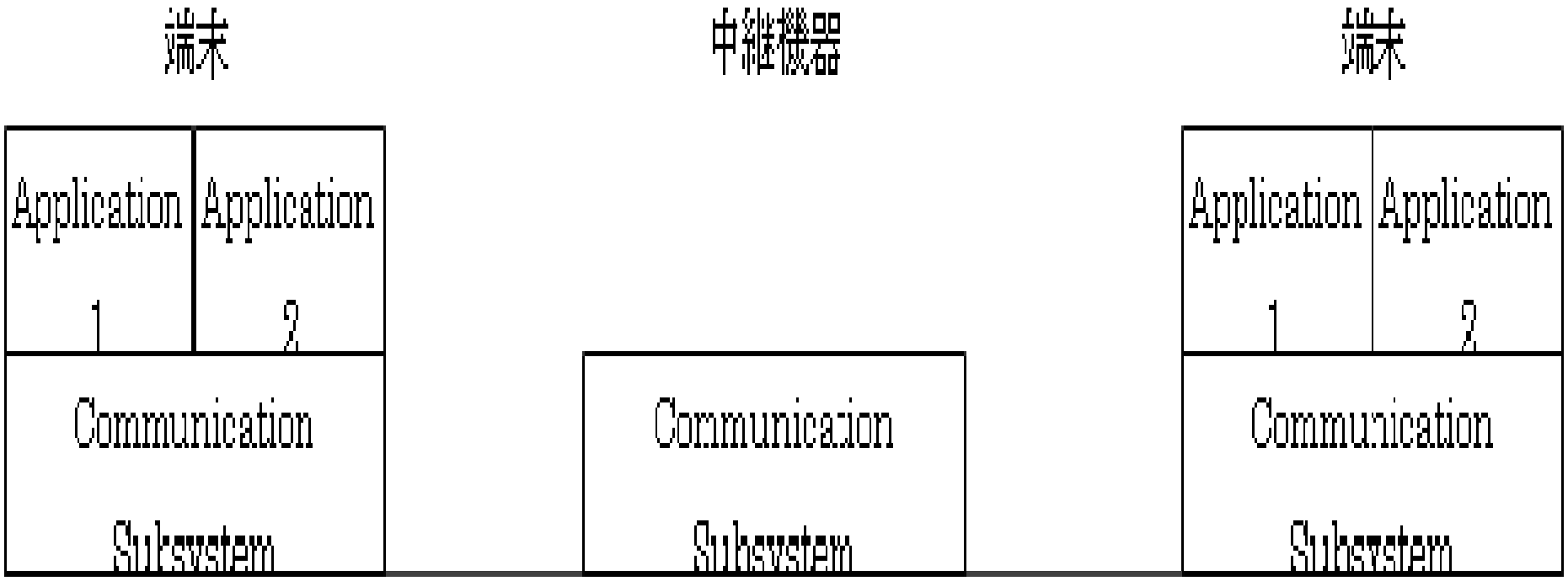

図1　エンドツーエンド論法で想定しているレイヤ構造

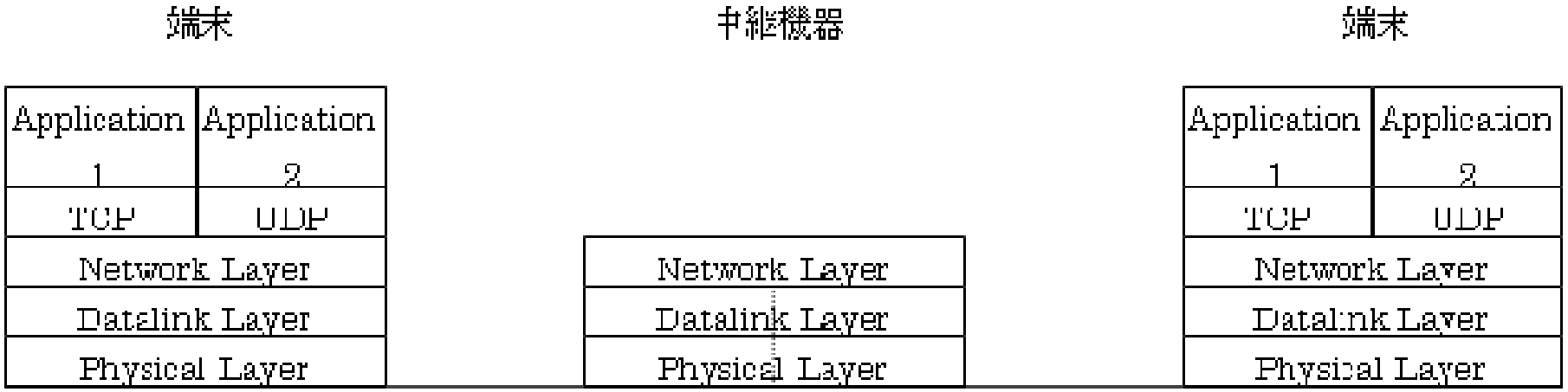

図 2　現代のインターネットのレイヤ構造

エンドツーエンド定理

- 分散情報システムにおいては、あるサブシステム（自サブシステム）が他のサブシステム（相手サブシステム、一般には複数）の状態を知ったり変えたりすることは、相手サブシステムの持つ知識と助けなくしては不可能であり、自サブシステムの作業を自サブシステムの知識を持たない他のサブシステムに代行させることも完全かつ正確には行えない。

レイヤリングを考慮したエンドツーエンド定理

- 分散情報システムにおいては、あるサブシステム（自サブシステム）が他のサブシステム（相手サブシステム、一般には複数）のあるレイヤの状態を知ったり変えたりすることは、相手サブシステムの当該レイヤの持つ知識と助けなくしては不可能であり、自サブシステムの作業を自サブシステムの知識を持たない他のサブシステムに代行させることも完全かつ正確には行えない。

エンドツーエンド定理の証明？

- 他のサブシステム（相手サブシステム、一般には複数）の状態を知ったり変えたりすることは、相手サブシステムの持つ知識と助けなくしては不可能

って、自明でしょ

- 自サブシステムの作業を自サブシステムの知識を持たない他のサブシステムに代行させることも完全かつ正確には行えない

も、通信にエラーがあれば、自明

一応の証明（というか、「分散情報システム」の定義？）（1）

- 分散情報システムとは、通信リンクによって結合されたサブシステムの集合である
  - 分散情報システムの要素を列挙しただけ
- 各サブシステムは、自らの状態を知り変えることができるが、他のサブシステムの状態を直接知ったり変えたりすることはできない
  - 「分散」してる以上「直接」できるわけがないのは当たり前
  - ここここから、定理の前半が直接導出される

一応の証明（というか、「分散情報システム」の定義？）（2）

- 各サブシステムは、自らに接した通信リンクを通じてその通信リンクに接している他のサブシステムと情報を送受でき、また通信リンクの自らに接した部分の状態を知ることができる
  - 通信リンクの意味を与えた
- 各サブシステムと各通信リンクは、雑音や故障の影響を受け、情報を送受できなくなったり、間違った情報を送受したりすることがある
  - ここから、定理の後半が直接導出される

おわりに

- エンドツーエンド論法の原論文の精査により
  - レイヤリングとの関係を明らかにし
  - 一般の分散情報システムに対して拡張した定理として証明した
- エンドツーエンド定理は
  - レイヤリングされたプロトコルの議論に、直接適用できる
  - ネットワーク内部のプロトコルの議論に、直接適用できる

completely and correctlyの定量化

- 無限の完璧さ・正確さには、無限の通信が必要
- 原論文では、この程度（定性的）
  - a checksum that has sufficient redundancy to reduce the chance of an undetected error in the file to an acceptably negligible value.
- リトライを繰り返す場合のエラー率を定量化：
  - （エンドエンドの通信エラー率）^（時間/RTT）

程度で改善してゆく

  - リトライなしではRTT程度に最新の情報取得を目指すべき

PMTUDへのエンドツーエンド定理の適用

- PMTUDの困難
  - CompleteでCorrectなMTUをどう取得する？
    - MTUが大きくなったことを、どの程度の頻度で調べるべきか？
- RTT程度！
  - 送信間隔がRTT以上なら、もっと低頻度でよい
  - RTTが計測できていることが前提
    - コネクションが前提
    - トランスポート層での実装が妥当
    - 結構頻繁で、ルータには高負荷かも

Unicast IGPへのエンドツーエンド定理の適用

- Functionは、各目的地への最短経路の計算
- 他のサブシステム（相手サブシステム、一般には複数）の状態を知ったり変えたりすることは、相手サブシステムの持つ知識と助けなくしては不可能

  は、当然として、、、
- 自サブシステムの作業を自サブシステムの知識を持たない他のサブシステムに代行させることも完全かつ正確には行えない

  つまり、ネットワークで計算（DV型)しては駄目で、LS型ならRTT程度で経路は収束する

  - OSPFの最短HELO間隔は長過ぎ、リンクのRTT程度が妥当？
    - ただし、HELOの帯域は全トラフィックの1(0.1?)%以下にすべき