

# 2H-01 インターネットにおける End-to-end パフォーマンス測定解析システムのフレームワーク

串田高幸\*

日本アイ・ビー・エム株式会社東京基礎研究所

柴田義孝

岩手県立大学ソフトウェア情報学部

## はじめに

最近、インターネットは、社会の共通のインフラストラクチャーとして、日々の様々なアクティビティーに利用されている。さらに企業間のビジネスにも利用されはじめてきている。しかし、もともとインターネットは、ベストエフォート型パケット交換ネットワークのアーキテクチャであるため、End-to-end のパフォーマンスを保証するための機構をもっていない。一方、インターネットにおいて、ネットワーク側で End-to-end パフォーマンスが保証されないにも関わらず、インターネットは、急激に広まっており、多くの人が利用しはじめている。また、最近では社会の重要なトランザクションの交換にも利用されてはじめている。このようにインターネットの利用される分野が、今まで以上に重要な分野になってきている。この結果、たとえインターネットが End-to-end のパフォーマンスが保証されていないアーキテクチャであったとしても、アプリケーションにおいて、現在の End-to-end パフォーマンスがどの程度のものか、また今後、どの程度のパフォーマンスが得られるのかについて正確に見積ることが必要性が増大してきている。

図 1 は、End-to-end のホスト間とインターネットのパフォーマンスについての概念図である。この図において Packet Flow が左から右に流れているが、途中のパスでバンド幅が異なるために、パスの狭いところ（バンド幅が小さいところ）では、パケットの流れにボトルネックが生じている。そのため、End-to-end パフォーマンスの観点からみると、全体のパフォーマンスは途中経路の最も狭いところに依存してしまうことになる。このように End-to-end のパフォーマンスの観点からみると、パスの途中にボトルネックがあるとアプリケーションあるいはサービスからみて、End-to-end のパフォーマンスが常に変動していることになる。

本研究では、最初に End-to-end パフォーマンスを見積もるためのフレームワークについて定義する。このフレームワークでは、インターネットにおいて End-to-end パフォーマンスを見積もるために測定用パケットをネットワークに入れるアクティブ方式を使う。また、今まで多くの他の測定結果を特徴をパラメータ化して記録しておき、現在の測定した結果と比べて End-to-end パフォーマンスを見積もる。

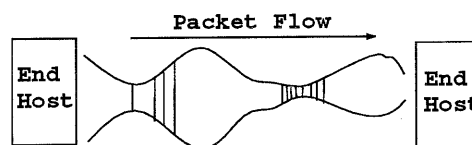


Figure 1: End-to-end performance

## フレームワーク

このフレームワークでは、End-to-end パフォーマンスを見積もるためにデータの測定とパラメータ化、結果を上位のアプリケーションから利用するためのコンポーネントの 2 つシステムから構成される。また、インターネットで測定解析されるデータとしては、1. 過去の測定解析データ（一般的な特徴量も含む。）と 2. 現在の測定解析データの 2 つがある。1. の過去の測定解析データを見積もって、パラメータ化して記録するために「測定解析システム」を構成する。また、1. のパラメータ化した値と 2. の現在の測定データの解析結果を使って「パフォーマンス・コンポーネント」を構成する。この「パフォーマンス・コンポーネント」は、アプリケーションとプロトコルスタックの間で動作してパフォーマンスを見積もってアプリケーションに知らせる。

## 基本測定に関して

一般に End-to-end パフォーマンスの評価を行なう場合に必要測定項目は、1. バンド幅、2. パケット損失率、3. パケットの遅延の 3 つの値である。これらの 3 つの値を測定装置によって取得してデータを解析処理してパラメータ化する。これらのパラメータから特徴量を解析する。測定方法であるが、バンド幅は、転送された量 (Byte) と転送にかかった時間 (sec) の比で表わされる。また、パケット損失率は、転送されたパケットに対して損失したパケットの比を 100 パーセント率で表したものである。さらにパケットの遅延は、パケットを転送しているときに起っているパケットの転送遅延である。パケットの遅延には、最低限の転送時間があり変動が、その転送時間に重なるために、さらに遅くなる。

## 測定解析システム

図 2 が、測定解析システムである。Measurement Instruments (1) は、“Packet Generator”, “Packet Receiver”, “Parameterizer”, “Analyzer”, “Reporter” があって、また、Measurement Instruments (2) は、“Packet Generator”, “Packet Receiver”, “Parameterizer” がある。

\*Framework of End-to-end performance measurement and analysis system for Internet applications, Takayuki Kushida (kushida@trl.ibm.co.jp) and Yoshitaka Shibata (shibata@iwate-pu.ac.jp)

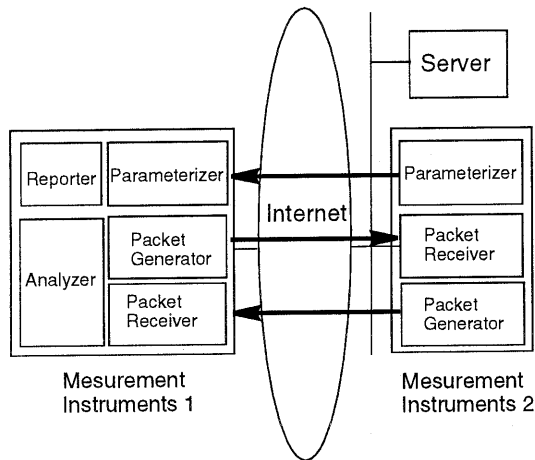


Figure 2: Measurement and analysis system

Time Stamp	IP address for TS	....	Data Padding
------------	-------------------	------	--------------

Figure 3: Measurement packet

Measurement Instruments (1) の“Packet Generator”は、測定用パケットを生成して、その測定パケットを対向している Measurement Instruments (2) の“Packet Receiver”に送信する機能を持つ。また、Measurement Instruments (2) の“Packet Generator”から送られてきた測定用パケットを受信して測定に必要なデータを“Parameterizer”に送る。また、“Packet Receiver”は、“Packet Generator”に対して Measurement Instruments (1) に測定データを送るように指示を出す。Measurement Instruments (2) の“Parameterizer”では、“Packet Receiver”からデータを受け取ってパラメータ化して、Measurement Instruments (1) に転送する。一方、Measurement Instruments (1) の“Parameterizer”は、Measurement Instrument (2) の“Parameterizer”から送られてきたパラメータを受信して、Measurement Instruments (1) のパラメータとともに“Analyzer”に送る。“Analyzer”では、それぞれの Parameterizer から来たパラメータを解析する。また、“Reporter”は、測定解析結果をユーザーに報告する機能及び解析結果を次のデータのために保管する機能を持つ。測定用パケットは、図3に示した形式のパケットを End Host 間で利用する。このパケットには、“Time Stamp”及び“IP address for Time stamp”を入れる。この測定用パケットにおいて、複数の経路を測定する場合には、通過した経路の Time Stamp 及び IP address を入れて、それぞれのフィールドを作る。また、測定用パケットを固定長パケットとするためにバイト数が足りない場合には、最後尾に Data Padding を挿入して同じ長さのデータとして揃える。

### パフォーマンス・コンポーネント

アプリケーションプログラムからインターネットのパフォーマンスを利用する場合には、測定解析結果をパラメータ化して、ライブラリとしてアプリケーションから利用できるように構成する必要がある。ここでは、過去の測定解析結果のパラメータと実時間で測定解析したパラメータを適切な形で組み合わせてパフォー

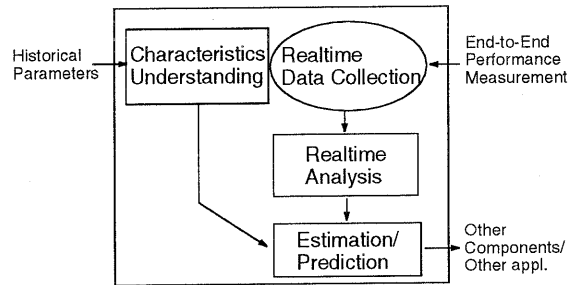


Figure 4: Performance Component

マンスの見積を行なう。図4は、パフォーマンス・コンポーネントの構成図である。この図において Historical Parameters は、前の節で説明した「測定解析システム」においてパラメータが記録されているので、そのパラメータを入力として使用する。また、この Historical Parameter を Characteristics Understanding コンポーネントに入力して、特徴量を Estimation/Prediction コンポーネントに出力する。一方、End-to-end パフォーマンスの測定を実時間で行なった結果は、Realtime Data Collection に集められていて、Realtime Analysis コンポーネントで解析される。データ解析された結果は、Estimation/Prediction コンポーネントで出力される。2つの結果が入力された Estimation/Prediction コンポーネントでは、他のコンポーネントあるいはアプリケーションに対して適切なパラメータとして通知することができる。このようにパフォーマンス・コンポーネントでは、ネットワークの過去の測定結果から得られた特徴量と現在の状況を比較して適切な値にして、他のコンポーネントあるいはアプリケーションに通知する。

### 議論

測定専用を使うパケットをネットワークに入れて、その結果によって End-to-end パフォーマンスを測定する場合、測定パケットがネットワークの資源を消費することになる。例えば、アナログモデムを利用している公衆回線のようにリンクのバンド幅が小さい場合は、測定パケットの占める割合が相対的に大きくなるが、全体の傾向として End Host からのアクセス速度が、速いと測定パケットの占める割合が相対的に小さくなる。このように End Host からのアクセス速度がより速くなってくれば、ネットワーク資源に対して測定パケットの影響が小さくなるので、測定パケットが占める比率の問題は、将来はほとんどなくなると考えられる。

一般にネットワークの測定を行なう場合、測定したパケットの誤差について考慮する必要がある。ネットワーク測定装置の誤差に関しては、パケットカウンターとネットワークパケットを捕獲するとき CPU ロードが高くなることによって測定誤差が出ることが報告されている。また、測定の時間に関しては、ハードウェアクロックを使用して精度を高める必要がある。CPU ロードによって測定が影響をなるべく受けないように、本研究では、実際に利用するシステムとは別に測定解析システムを別々に用意して、実際に利用するコンポーネントでは、実時間の参照のために測定解析だけを行なうようにシステムを構成している。