第 IV 部

ネットワークトラフィック統計情報の収集と解析

第 4 部
ネットワークトラフィック統計情報の収集と解析

第 1 章　MAWI ワーキンググループ

MAWI（Measurement and Analysis on the WIDE Internet）ワーキンググループは、トラフィックデータの収集と解析を研究対象とした活動を行っている。

MAWI WG では WIDE Project の特徴を活かした研究をするため、「広域」「多地点」「長期的」の 3 つの項目に重点を置いたトラフィックの計測・解析を行っている。

1.1 広域で行う

インターネットの最大の特徴は、大規模な広域ネットワークにある。しかし、トラフィック情報には組織の機密保持やプライバシ保護の問題がともなうため、とくに不特定多数のトラフィックを含む広域データを第三者が入手するのは困難である。

1 つの企業や組織内といった狭い範囲でデータを取ることは各組織でできるが、広域バックボーンでのデータ収集はバックボーンを持っている WIDE だからこそ可能である。

1.2 多地点で見る

インターネットのもう 1 つの大きな特徴は、自律したネットワークが相互接続して経路制御を行い、エンド・エンドで通信制御を行う分散システムにある。したがって、ある地点でトラフィックを観測しても、ネットワーク全体の状態を捉えることは不可能である。観測者はあくまでその観測点から見たインターネット像が得られるだけで、別の観測点からはまったく別の世界が見えているかもしれない。

MAWI WG では、多地点で観測したデータを照らし合わせることによって、より広い範囲のネットワークの状態を把握する手法や、それを俯瞰で可視化することによって直観的にわかりやすく観測する手法について研究を行っている。

1.3 長期間行う

ネットワークのトラフィックの挙動は、TCP の特定のアルゴリズムが関係するようなミクロなものから、過去 10 年間のトラフィック量の推移のようなマクロなものまで、幅広い時間スケールにわたっている。また、ミクロな挙動についても、インターネットのマクロなレベルの発展にともなって、次第に変わっていく。

したがって、1 日や 1 週間といった短期間の計測も重要だが、何年間という長いスパンでデータを取り続けることが非常に重要になる。しかし、長期的にデータを収集し、その蓄積を持つことは、ある日誰かが思いついてできるものではない。そこで、ワーキンググループとしてメンバが協力して継続的なデータ収集を行っていくことが必要である。

計測技術はほとんどの研究分野で必要となるため、MAWI WG は WIDE 内の他のワーキンググループと連係をとりながら活動を行っている。具体的には、

- グローバルな視点からの DNS の挙動解析（DNS WG と共同）
- IPv6 普及度の計測（v6fix WG と共同）
- ネットワークトポロジの観測（netviz WG と共同）
- 長期的な経路変動の観測（routeview WG と共同）
- sFLow/NetFlow を使ったトラフィック計測（roft WG と共同）
- AI3 の衛星トラフィックの計測（AI3 WG と共同）

などが挙げられる。

また、国際協調として

- CAIDA（http://www.caida.org/）
- University of Waikato（http://www.cs.waikato.ac.nz/）
- ICANN RSSAC（http://www.icann.org/committees/dns-root/）
- ISC OARC（https://oarc.isc.org/）
- USC/ISI（http://www.isi.edu）
- INRIA（http://www.inria.fr/index.en.html）

などと共同して研究活動を行っている。

今年度の報告書では、まず第 2 章において、集約型トラフィックプロファイラを使った国際線トラフィックの傾向を報告する。このツールは、WIDE バックボーンのトラフィックをニアリアルタイムかつ長期的にモニタリングする目的で 2001 年に開発され、それ以来利用されてきている。また、急増する分散型 DoS アタックの早期検出にも役立っている。

第 3 章では、BGP 経路情報の収集と分析について報告する。WIDE Project での BGP の経路情報の収集手法を説明し、2003 年 8 月からの約 1 年分のデータをもとに、経路情報の変動を解析する。経路情報のデータを長期的に保存することによって、過去にさかのぼって経路の状態を検証することが可能になるので、今後もさまざまな研究への利用が考えられる。

第 4 章では、IPv6 インターネットの品質を調査するデュアルスタック計測について報告する。本格的な IPv6 への移行のためには、IPv6 インターネットの品質改善が急務である。もし、一部のサイトが問題を起こしているだけなら、それらを直せば IPv6 インターネットの品質を大きく全体的に改善できる。まずは現状を把握するために、2004 年の 1 月から IPv6 インターネットと IPv4 インターネットの品質比較調査を行っている。この研究は、WIDE、CAIDA、ワイカト大学の共同研究として行われており、現在では、v6fix WG の活動の一環にもなっている。

第 5 章では、CAIDA の NeTraMet による Root DNS サーバ群の応答時間計測を、慶應大学と東京大学で運用している実験について報告する。これと並行して、WIDE で開発したアクティブ計測による Root DNS サーバの観測も行っており、相互にデータを照合することでより正確な実態の把握が可能になる。

第 6 章では、2004 年 4 月と 8 月に行った CAIDA との 2 度の計測ワークショップについて報告する。CAIDA と WIDE は、2003 年度から正式に計測に関する包括的な共同研究を行っている。また、2005 年度も CAIDA との共同研究を継続し、DNS 計測、IPv6 トポロジ計測、BGP の解析などを行っていくほか、人材交流も予定している。

第 7 章では、フランスの INRIA、タイの AIT、WIDE の 3 組織で、2004 年 9 月にフランスで開催したワークショップについて報告する。このワークショップは、幅広い研究分野の交流を目的として行われ、計測もその中の重要分野であったため、ここで報告することにする。

2005 年度は、引続き DNS 計測、IPv6 トポロジ計測、BGP の解析などを中心とし、また国際的な共同研究をより積極的に進める予定である。

第 2 章　WIDE 国際線のトラフィック傾向

2.1 はじめに

WIDE インターネットのような広域なネットワークを運用し続けていくためには、トラフィックモニタリングを多地点、かつ長期間行い、ネットワークの現状に適した通信機器の設置、設定を行う必要がある。

しかし、現存するネットワークモニタリングツールは長期に渡ってトラフィックの傾向を収集し続けることが難しい。

そこで、WIDE Project MAWI WG では収集したトラフィックを効果的に集約することによって、ネットワークの特徴を抽出することのできるトラフィックモニタリングツール AGURI[39] の設計、実装を行った。

AGURI（Aggregation-based Traffic Profiler）は、1) トラフィック中の特徴的なフロー傾向を残しつつ、2) 短期間から長期間に渡って利用可能なトラフィックモニタリングツールである。

AGURI は以下に示す 4 種類のネットワークサマリ情報を作成する。

- 送信元 IP アドレス
- 受信先 IP アドレス
- IP バージョン＋プロトコル＋送信ポート番号
- IP バージョン＋プロトコル＋受信ポート番号

この 4 種類のネットワークサマリを定期的に出力することによって、ある短時間のネットワーク状態の特徴を知ることができる。

さらに、AGURI は一度 AGURI で作成したネットワークサマリからもデータを入力することができ、複数のサマリを同時に入力することもできるので、ある短時間のサマリを組み合わせて AGURI に入力することによって、可変長の時間のネットワーク状態の特徴を知ることができる。

2.2 収集データ

WIDE Project では以下に示す 2 地点において国際線のデータを収集している。

1. samplepoint1 trans-Pacific line (18 Mbps CAR on 100 Mbps link)
2. samplepoint2 US-Japan line (Japan side 60 Mbps POS)

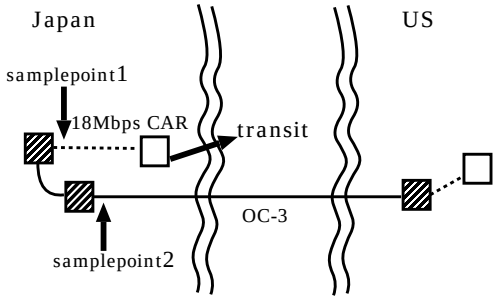

図 2.1. データ収集地点

WIDE Project で利用している 2 本の国際線のうち、1 本は他 AS と BGPpeer を張っているポイントの WIDE インターネットの入り口でデータ収集を行っている（samplepoint1）。

ほかの 1 本は WIDE の利用している国際線の日本側（samplepoint2）でデータ収集を行っている。

2004 年度の WIDE 報告書では、samplepoint1 と samplepoint2 で収集した WIDE 国際線の年間トラフィック傾向を図 2.2 から図 2.29 に示す。

図の出力は時期を四半期ごとに、対象を 1) 送信元 IP アドレス、2) 宛先 IP アドレス、3) 送信元ポート番号、4) 宛先ポート番号とする（表 2.1、表 2.2）。

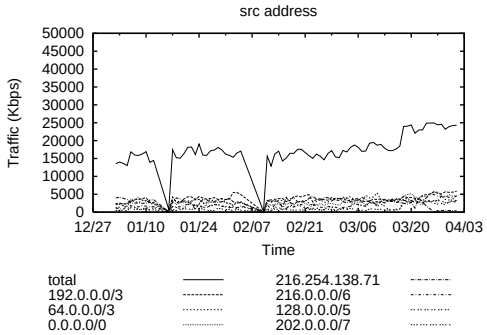

図 2.2. 送信元 IP アドレス（1 月−3 月）

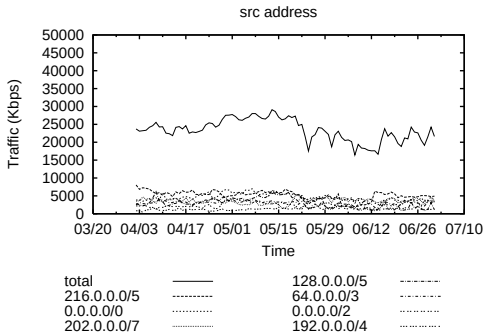

図 2.3. 送信元 IP アドレス（4 月−6 月）

ただし、samplepoint1 で収集されたデータのうち、収集機器の不具合のため 1 月中旬、2 月初旬のデータが収集されていない。また、samplepoint2 で収集されたデータは、2004 年 9 月 25 日以降の設定変更に対応出来なかったため、9 月 25 日以降のデータが欠落している。そのため、samplepoint2 に関しては 1 月から 9 月までのデータを示す。

表 2.1. トラフィック傾向一覧表（samplepoint1）

	1 月−3 月	4 月−6 月	7 月−9 月	10 月−12 月
送信元 IP アドレス	図 2.2	図 2.3	図 2.4	図 2.5
宛先 IP アドレス	図 2.6	図 2.7	図 2.8	図 2.9
送信元ポート番号	図 2.10	図 2.11	図 2.12	図 2.13
宛先ポート番号	図 2.14	図 2.15	図 2.16	図 2.17

表 2.2. トラフィック傾向一覧表（samplepoint2）

	1 月−3 月	4 月−6 月	7 月−9 月
送信元 IP アドレス	図 2.18	図 2.19	図 2.20
宛先 IP アドレス	図 2.21	図 2.22	図 2.23
送信元ポート番号	図 2.24	図 2.25	図 2.26
宛先ポート番号	図 2.27	図 2.28	図 2.29

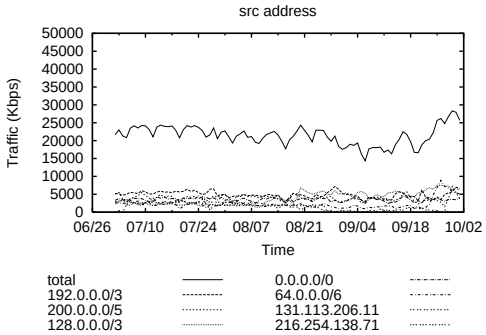

図 2.4.　送信元 IP アドレス（7 月−9 月）

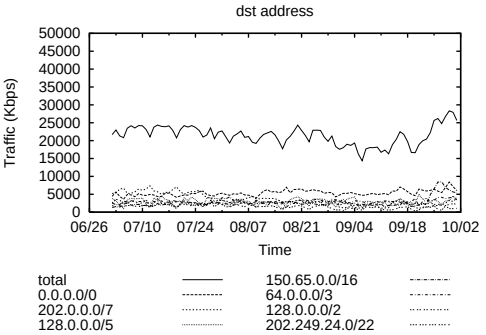

図 2.8.　宛先 IP アドレス（7 月−9 月）

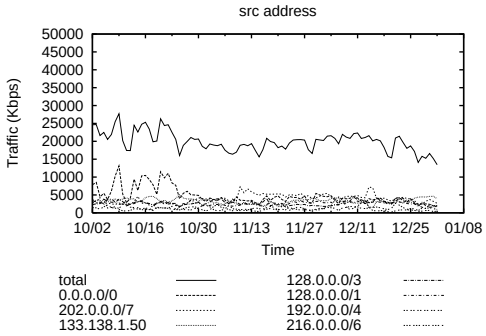

図 2.5.　送信元 IP アドレス（10 月−12 月）

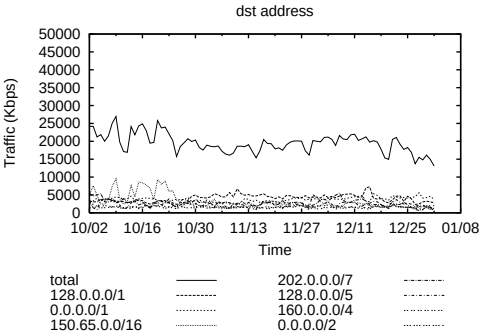

図 2.9.　宛先 IP アドレス（10 月−12 月）

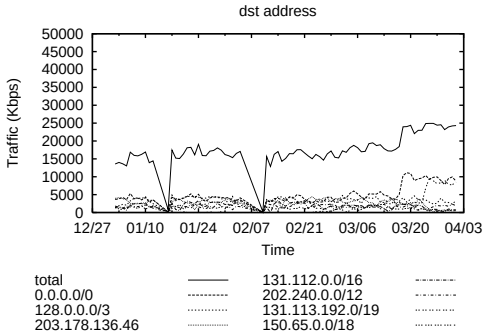

図 2.6.　宛先 IP アドレス（1 月−3 月）

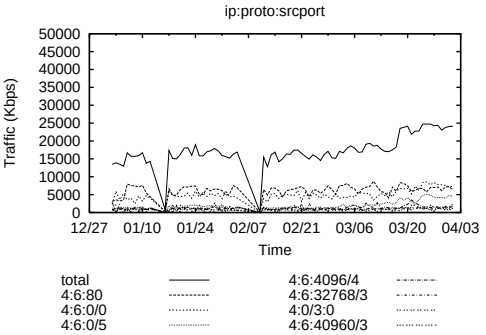

図 2.10.　送信元ポート番号（1 月−3 月）

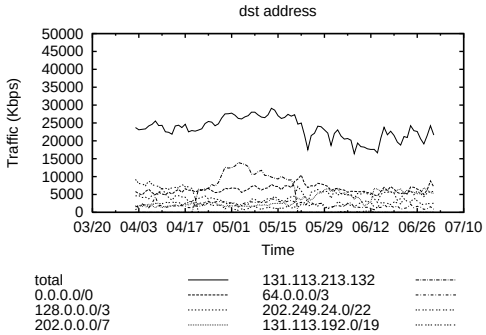

図 2.7.　宛先 IP アドレス（4 月−6 月）

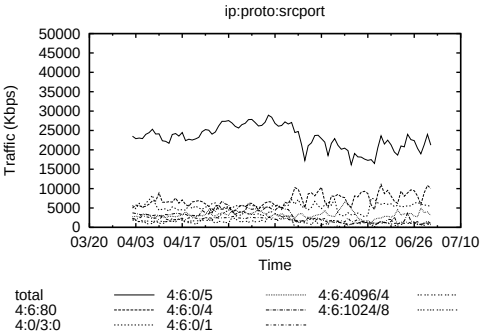

図 2.11.　送信元ポート番号（4 月−6 月）

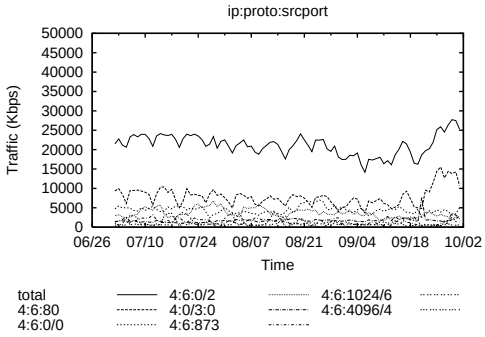

図 2.12. 送信元ポート番号（7 月−9 月）

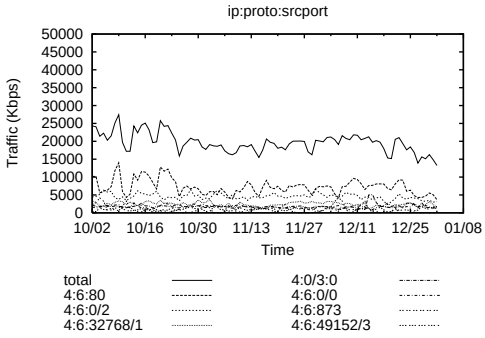

図 2.13. 送信元ポート番号（10 月−12 月）

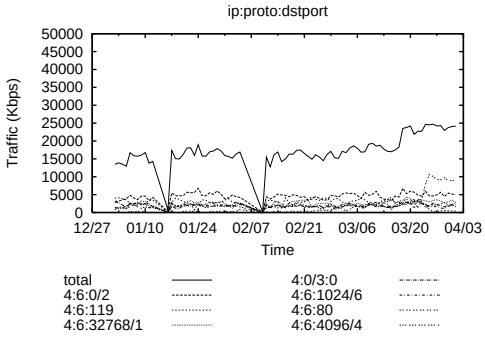

図 2.14. 宛先ポート番号（1 月−3 月）

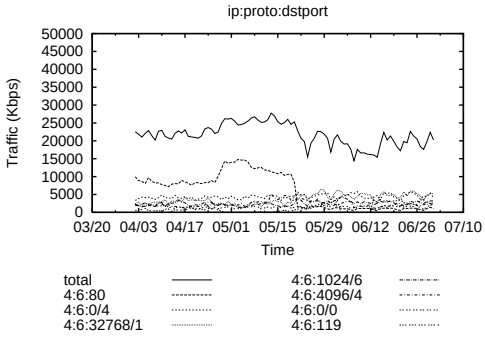

図 2.15. 宛先ポート番号（4 月−6 月）

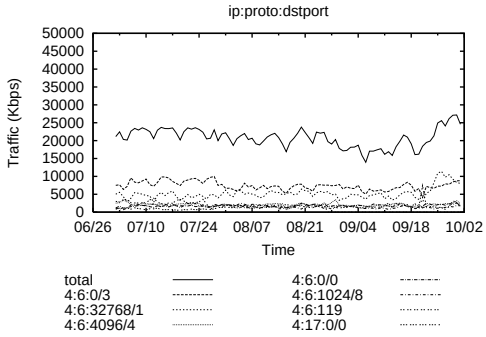

図 2.16. 宛先ポート番号（7 月−9 月）

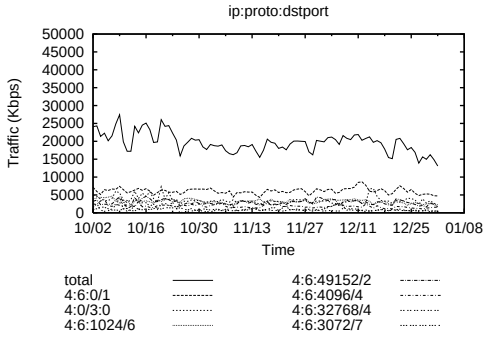

図 2.17. 宛先ポート番号（10 月−12 月）

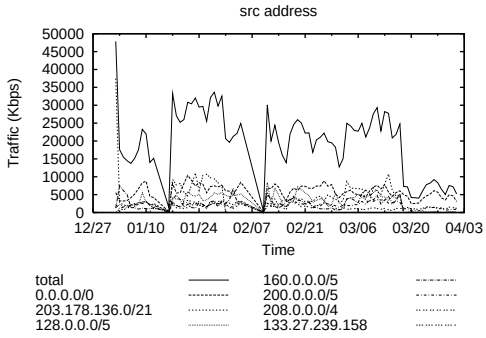

図 2.18. 送信元 IP アドレス（1 月−3 月）

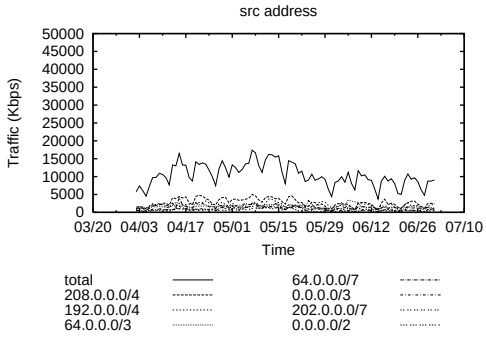

図 2.19. 送信元 IP アドレス（4 月−6 月）

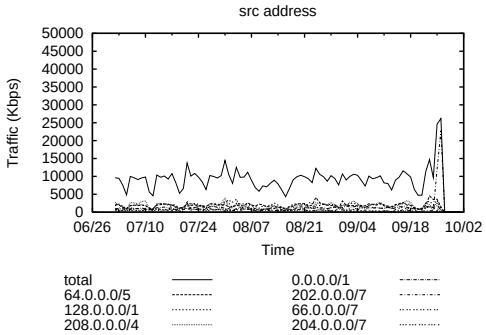

図 2.20.　送信元 IP アドレス（7 月−9 月）

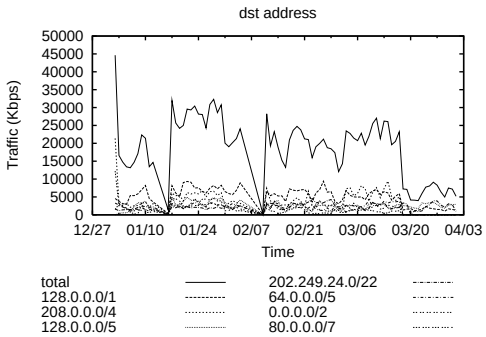

図 2.21.　宛先 IP アドレス（1 月−3 月）

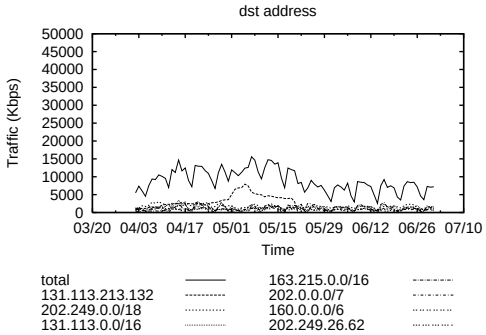

図 2.22.　宛先 IP アドレス（4 月−6 月）

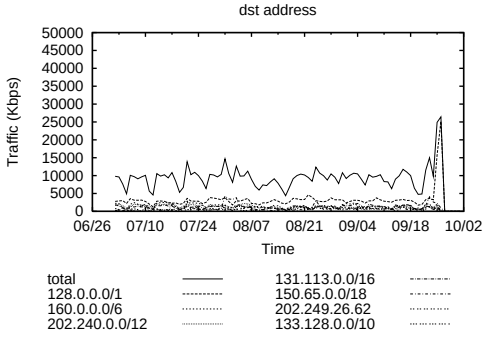

図 2.23.　宛先 IP アドレス（7 月−9 月）

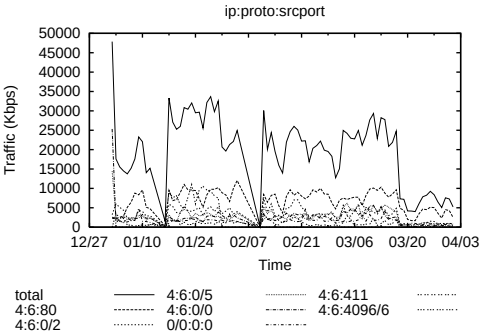

図 2.24.　送信元ポート番号（1 月−3 月）

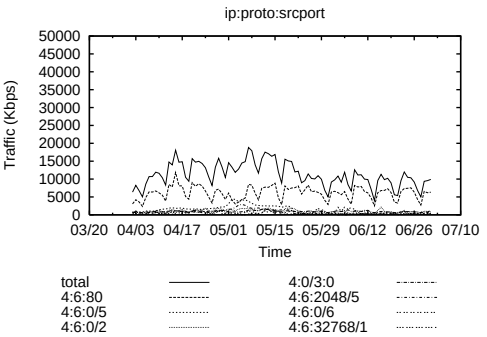

図 2.25.　送信元ポート番号（4 月−6 月）

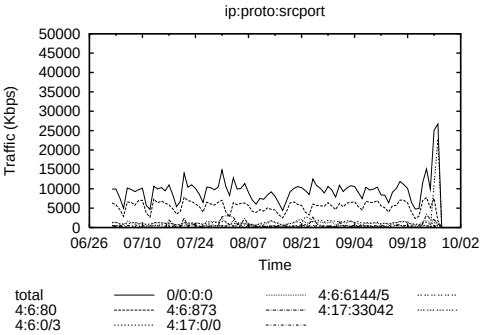

図 2.26.　送信元ポート番号（7 月−9 月）

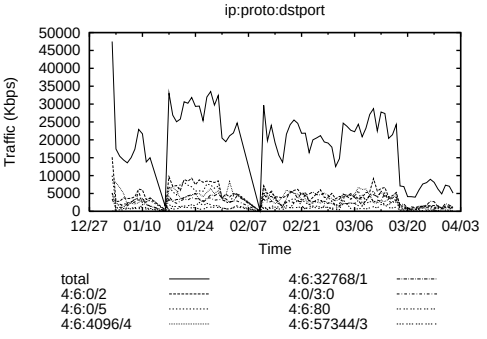

図 2.27.　宛先ポート番号（1 月−3 月）

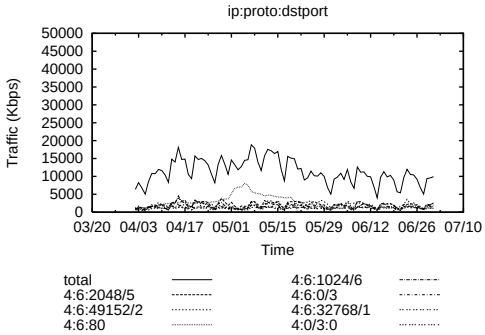

図 2.28. 宛先ポート番号（4 月−6 月）

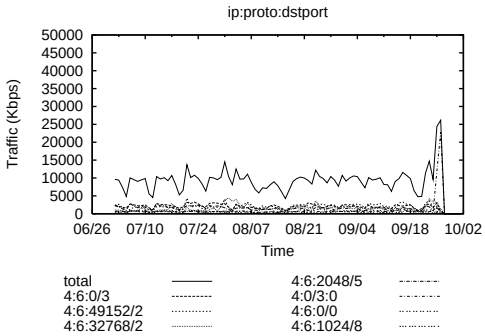

図 2.29. 宛先ポート番号（7 月−9 月）

図 2.2 から図 2.29 に示された長期的トラフィック傾向から抽出できた情報を表 2.3、表 2.4 に示す。

これらの表 2.3、表 2.4 にある “4:6:80” とは IP バージョンが 4、プロトコル番号が 6（つまり TCP）、送信元ポート番号が 80（つまり HTTP）ということを示している。

ここに示した図は 2 つの情報を持っている。

- 折れ線グラフ

  回線を占めているトラフィックの属性を視覚的に見ることができる。

  今回取り上げた WIDE インターネット国際線の例では、全トラフィック量の推移と HTTP データの割合を把握できる。

- 項目

  折れ線グラフの下にリストアップされる項目数は、AGURI によって設定することができる。この項目は全トラフィック中の占有率順にリストアップされるため、回線を使用している組織や使われているアプリケーションを検知することができる。

  送信元、宛先 IP アドレスからは、特定の組織の IP アドレス空間と特定のホストを検出できた。

表 2.3. 識別された IP アドレス

graph	IP アドレス	hostname
図 2.2、2.4	216.254.138.71	media.kyoto-u.feed.nnrp.primus.ca
図 2.4	131.113.206.11	news.fbc.keio.ac.jp
図 2.5	133.138.1.50	mobiquitous.csl.sony.co.jp
図 2.6	203.178.136.46	news.sakyo.wide.ad.jp
図 2.6、2.7、2.23	131.112.0.0/16	titech.ac.jp
図 2.6、2.8、2.9、2.23	150.65.0.0/16	jaist.ac.jp
図 2.7、2.22	131.113.213.132	keio.ac.jp 配下のホスト
図 2.7、2.8、2.21	202.249.24.0/22	ai3.net
図 2.18	203.178.136.0/21	wide.ad.jp
図 2.18	133.27.239.158	keio.ac.jp 配下のホスト
図 2.22,2.23	202.249.26.62	ai3.net 配下のホスト

表 2.4. 識別されたポート番号

graph	ポート番号	プロトコル／アプリケーション
図 2.10–2.17、図 2.24–2.29	4:6:80	HTTP
図 2.14、2.15、2.16	4:6:119	nntp
図 2.27	4:6:411	DC++[54]

とくに 2002 年度の WIDE 報告書と比較した場合、2002 年度に観測された ai3.net を宛先としたトラフィックを引き続き抽出できた。それに加え、titech.ac.jp、jaist.ac.jp、keio.ac.jp など AS2500 番に接続されている学術組織のアドレスブロックを抽出できた。

また、今年は特定のホストにトラフィックが集中している様子も観察できた。抽出された ‘131.113.213.132’ という IP アドレスは、Agobot.FO と呼ばれるワームの攻撃先になっていたホストの保持するアドレスであり、‘202.249.26.62’ という IP アドレスはインドネシア BRAWIJAYA 大学に設置されている Web キャッシュサーバである。

送信元ポート番号からは、特定のポートを使用したアプリケーションを検出できた。

今年度も引き続き HTTP トラフィックが流行している事に加えて、NNTP トラフィックも抽出できた。これは IP アドレスのトラフィック特徴にて news.sakyo.wide.ad.jp が抽出されたことと関連深い。また、今年度は仮想 hub ネットワークを用いたファイル共有アプリケーションである ‘4:6:411’（IPv4:tcp:port 411 番）を samplepoint2 において抽出することができ、観測地点によってアプリケーションの流行が異なる結果となった。

以上のように、折れ線グラフで表された情報とリストアップされた項目から、全トラフィックを構成している特徴的な要素を抽出することができた。

2.3 結論

本章では、AGURI を用いた WIDE インターネット国際線のトラフィック傾向を述べた。

WIDE インターネットのような広域なネットワークを運用し続けていくためには、トラフィックモニタリングを多地点、かつ長期間行い、ネットワークの現状に適した通信機器の設置、設定を行う必要がある。

しかし、現存するネットワークモニタリングツールは長期に渡ってトラフィックの傾向を収集し続けることが難しい。

WIDE Project MAWI WG では収集したトラフィックを効果的に集約することによって、ネットワークの特徴を抽出することのできるトラフィックモニタリングツール AGURI を用い長期に渡る国際線のトラフィック傾向を明らかにした。

実際に AGURI を用いて WIDE インターネット国際線でデータを収集し、対象とした国際線のトラフィックの傾向を明らかにした。

WIDE Project では、AGURI の開発を進めるとともに、WIDE インターネットのバックボーンにおいて AGURI を運用し続けている。これらのデータは `http://mawi.wide.ad.jp/mawi/`から参照可能である。

第 3 章　BGP ルーティング情報の収集と分析

MAWI WG の活動の一貫として、従来より WIDE ネットワークの BGP ルータの経路情報の収集が行われている。BGP 情報は、MAWI WG と WNOC-Sendai（仙台 NOC）が協力して収集を行っており、2003 年 8 月 27 日に開始され、これまでに 1 年分以上のデータを収集してきた。本章は、これまでに収集された BGP 経路情報の分析結果を報告するものである。なお、本活動はもともと、ネットワーク情報の分析をより高度に行うために BGP 経路情報を活用する目的で始められたものであるが、RouteView WG が設立されたことにより、上記の研究は今後 RouteView WG において行い、成果の報告も行う予定である。本章では上記の関連を考慮し、特に MAWI WG の研究領域である統計的情報に絞って報告を行う。

3.1 概要

WIDE（AS2500）の BGP ルータが持つすべての経路（full route）情報を定期的に収集し、結果を保存している。データを蓄積している PC は仙台 NOC に設置しているが、仙台 NOC には full route を持つ BGP ルータが存在しないため、当該 PC において zebra を用いて BGP プロセスを起動し、LA-NOC に存在する Cisco ルータと ebgp multihop によって private peering を確立することで full route を取得している。

仙台 NOC において peering を行っている PC の諸元は以下の通りである。

```
! -*- bgp -*-
!
! BGPd configuratin file
!
hostname pc10.sendai.wide.ad.jp
password ******
enable password ******
!
router bgp 64514
 bgp router-id 203.178.138.26
 neighbor 203.178.136.20 remote-as 2500
 neighbor 203.178.136.20 ebgp-multihop 8
!
dump bgp all /db1/bgpdata/%Y.%m/BGPALL/all.%Y%m%d.%H%M 2h
dump bgp updates /db1/bgpdata/%Y.%m/UPDATES/updates.%Y%m%d.%H%M 15m
dump bgp routes-mrt /db1/bgpdata/%Y.%m/RIBS/rib.%Y%m%d.%H%M 2h
```

図 3.1. bgpd の設定ファイルの内容

- Name/IP: pc10.sendai.wide.ad.jp（203.178.138.26）
- OS: FreeBSD 4.10-Release-p5（2004/12/23 現在）
- Disk Space: 20GB＋120GB(RAID)＋700 GB（RAID）
- BGP daemon: zebra-0.94_2（2004/12/23 現在）（zebra + bgpd）

peering 先は、

- cisco1.LosAngeles.wide.ad.jp（203.178.136.20）

である。full route を Mirroring する設定としている。図 3.1 に bgpd の設定ファイル（bgpd.conf）の内容を掲げる。

データの蓄積は、Routeviews Project（`http://www.routeviews.org/`）が蓄積している、BGP パケットの full dump、UPDATE パケットの dump、および RIB データの 3 種類のデータを 2 時間ごとに蓄積しているのに加えて、管理コンソールから show ip bgp を実行して得られる output も 1 時間ごとに収集、蓄積している。

データ蓄積は、2003 年 8 月 27 日から現在まで継続して行っている（PC のメンテナンスなどのため、データの存在しない期間も若干ある）。現在、総データ量は 141.7 GB 程度となっている。

3.2 統計情報

以下の統計情報について分析を行った。なお分析期間はほぼ 1 年間とし、開始当初から 2004 年 8 月末までのものである。データとして、上記で説明した、1 時間ごとに取得している show ip bgp の結果を用いている。

- 学習したプレフィックスの総数
- 上記のプレフィックスに対応する、相異なる AS パスの総数
- AS パスの平均長
- プレフィックスおよび AS パスの安定性

3.2.1 プレフィックス数の変動

BGP ルータが学習した、相互に独立なプレフィックスの総数の変動を 1 時間ごとにプロットした結果が図 3.2 である。

1 年間でほぼ 15000 個増加している。また 2004 年 3 月にプレフィックス数が一時的に減少している様子がわかる。

3.2.2 相異なる AS パス数の変動

図 3.2 で示したプレフィックスに対応する AS パスで、相互に独立なものの総数の変動を 1 時間ごとにプロットした結果が図 3.3 である。すなわち異なるプレフィックスに対して同一の AS パスが学習されている場合には重複して数えることはしない。また、学習した全 AS パスの総数を示したものではない。つまり単一のプレフィックスに対応する複数の冗長な AS パスに関してはカウントしていない。

1 年間でほぼ 3000 個増加している。また図 3.2 同様、2004 年 3 月に AS パス数が一時的に減少している様子がわかる。

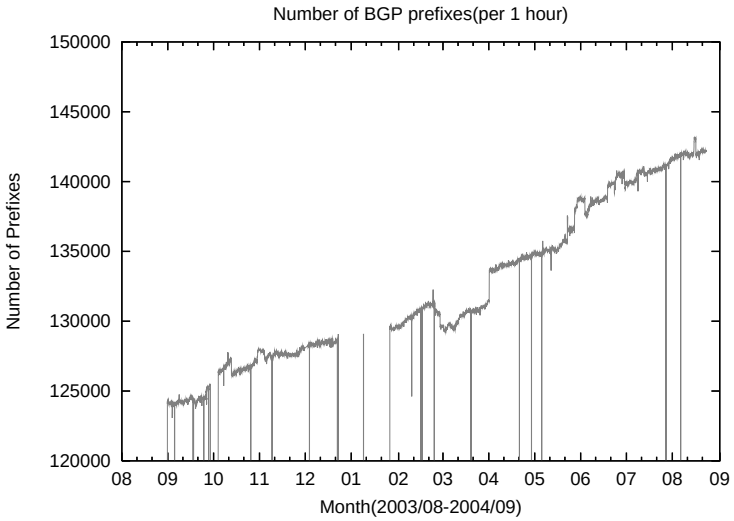

図 3.2.　Number of prefixes

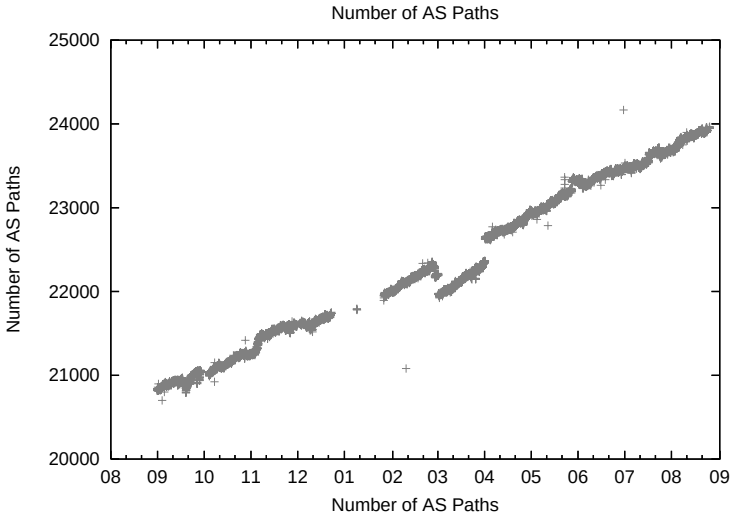

図 3.3.　Number of different AS paths

3.2.3 平均 AS パス長の変動

図 3.4 は、上記の図 3.3 で対象とした AS パスに対して、AS パス長の平均を取ったものである。傾向として安定していることが分かる。

なお、AS パスにおいては、いわゆるダミーが挿入されることにより、しばしば同一の AS 番号が連続することがある。これらを削除したものと削除しなかったものを同一にプロットした。結果、両者の差は 0.5 程度であった。

3.2.4 プレフィックス数および AS パス数の安定性

長期に渡って同一のプレフィックスや AS パスが観測されるとき、それらを「安定」であると呼ぶ。安定的なプレフィックスやパスがどの程度の割合で存在するのかを確かめた。

なお、「安定」を示す指標には 2 種類ある。1 つは「流行性（prevalence）」と呼べるものであり、当該期間に何回観測されたかという指標である。もう 1 つは「持続性（persistency）」と呼ばれるもので、連続して観測される期間が長いかどうかを示す指標である。

図 3.5 は個別のプレフィックスについて、1 年間を通しての累積観測回数を示したものである。これは上記の「流行性」を評価するものといえる。観測機会は、取得失敗がなかったとすると 1 年間で 8760 回となるので、これが累積観測回数の最大値である。ま

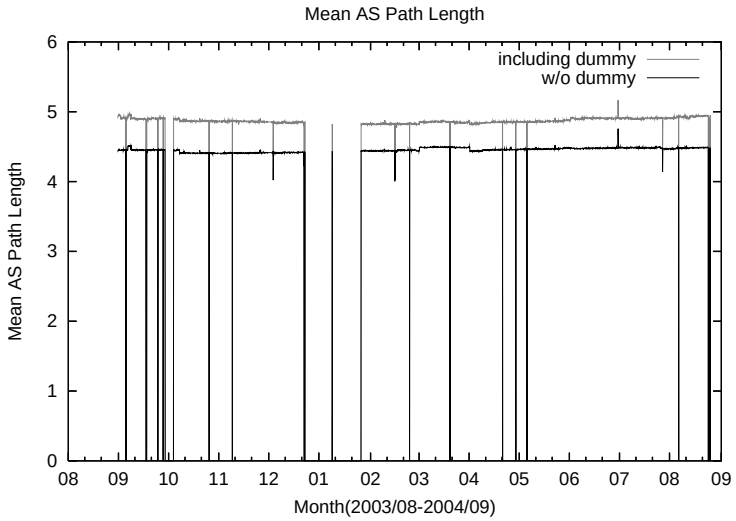

図 3.4. Mean AS path length

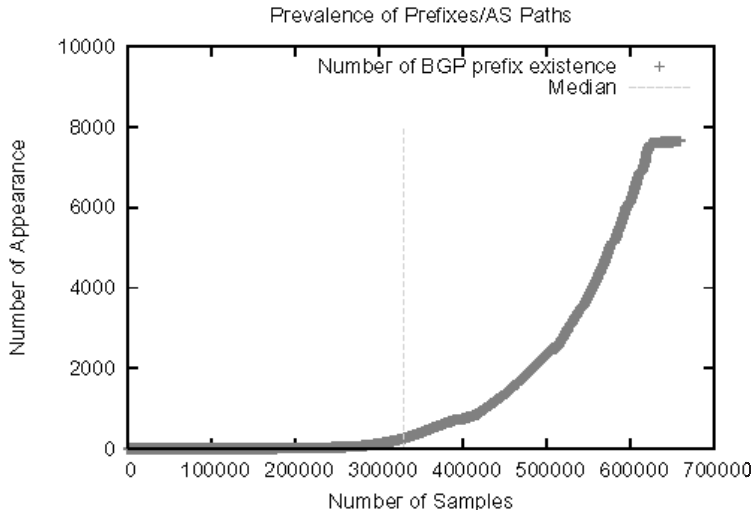

図 3.5. Prevalence of prefixes

た 1 年間を通して、プレフィックスおよびプレフィックス長が相異なるケースが 1 回でも観測された場合はすべて個別のものとして計数している。トータルのケース数は 658334 であった。

このうち、カウント数が 8760 に近い、すなわちほぼ 1 年間を通して観測されたものも少なからずあることがグラフからわかる。一方で、カウント数の少ない、すなわち「流行性」の低いものも多い。計測数のメジアン値をとると 257 回であることがわかる。これはほぼ 1 週間分の計測機会に相当する。つまり、半分以上のプレフィックスは 1 週間分は BGP 経路情報上に存在していたこととなる。

図 3.6 は「持続性」を評価したグラフである。観測のある時点から、個別のプレフィックスがどの程度連続して観測されたかをプロットしている。持続期間は 1 日、および 1 週間と設定した。グラフでは、ある観測点において、その時点で上記の期間以上の長さで連続して観測されているプレフィックスに関してカウントしたグラフを、全体のプレフィックス数と比較している。連続して観測されていたプレフィックスがある観測点から観測されなくなり、しばらくして再び観測されるようになった場合については一切考慮せず、再び観測されるようになった点を始点として持続期間を再カウントする。ただし、前述の通り、データ収集 PC のメンテナンスなどの理由からデータには部分的に抜けがあることがある。この

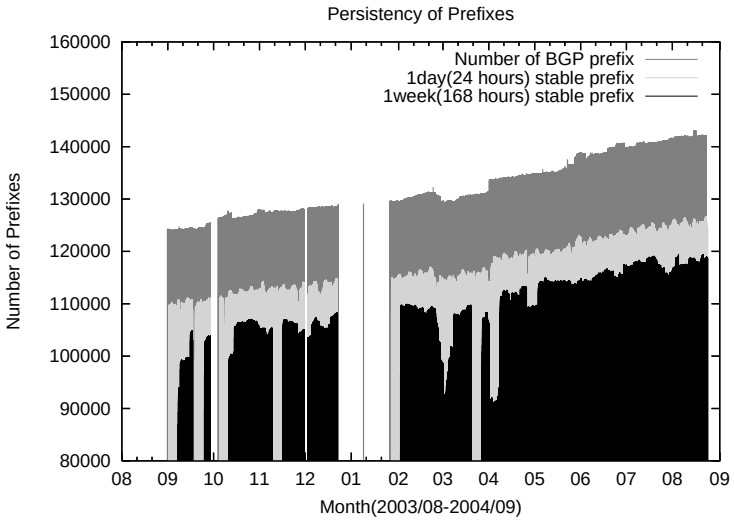

図 3.6.　Persistency of prefixes

点にそれらのプレフィックスが存在していたかどうかは判断できないため、便宜上、観測 1 回分の不連続性については無視することとした。

結果として、全体の 88%程度に相当する 110000 以上のプレフィックスについてはほぼ 1 日以上連続して観測され、さらに全体の 83%程度に相当する 100000 以上のプレフィックスに関しては 1 週間以上連続して観測されていることが分かる。

3.2.5 2004 年 3–4 月に見られたギャップについて

前述した、2004 年 3 月から 4 月にかけてのプレフィックス数および AS 数のギャップについて考察する。経路数が元の傾向に回復したのは 2004 年 4 月 1 日から 2 日にかけてであった。この区間においてプレフィックスの変動の差分を分析すると、おおむね AS701（UUNET）、AS11537（ABILENE）、AS7660 の関与する変動であることがわかった。これを元に、それらの AS 番号を AS パスに含むものについて、観測個数の変動を当該ギャップ区間においてプロットしたものが図 3.7 である。

当該期間には AS701 を origin とする経路が 15000 程度上昇しているのに対し、AS11537、AS7660 の両者で 12000 程度の経路が減少している。これはネットワークの切り替えに際して、AS701 が経路の広告

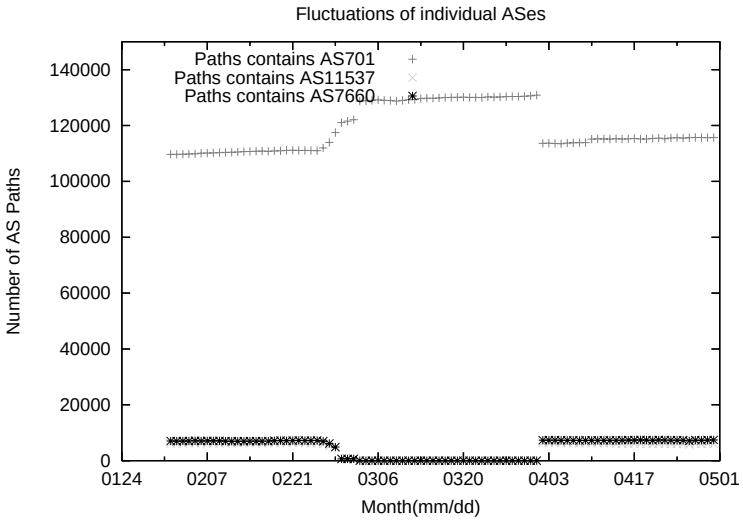

図 3.7.　Fluctuations of some ASes

を代行したためと推測される。なお、当該時期にはJGNネットワークに関連するネットワークの切り替えが実際に行われており、この観測結果はこれにともなう変動を表しているものと考えられる。

3.3 今後の方針

データの収集は今後も継続して行う方針である。

データの分析に関しては、同様の分析はさまざまな組織で行われていることから、よりさまざまな切口の分析を行う必要があると考える。特徴的なパラメータの1つに、割り当てられているIPアドレスのうちどの程度の割合のアドレスの経路が広告されているか、というものが考えられる。これは実際に使用されているアドレスの割合を表すことになる。

MAWI WGではIPv6のBGP情報の収集も行われているので、IPv4およびIPv6のBGP情報の統計を比較することも加味しながら今後の分析を進めることとなろう。

第4章 Identifying IPv6 Network Problems in the Dual-Stack World

(This paper appeared in SIGCOMM'04 Workshops, Aug. 2004, Portland, Oregon, USA.)

4.1 Introduction

The IPv6 Internet is in a state of transition from a collection of experimental research networks, such as the 6bone[99], toward a collection of production networks. One of the major hurdles limiting IPv6 adoption is the existence of poorly managed experimental IPv6 sites that negatively affect the perceived quality of the IPv6 Internet. To promote the use of IPv6, many operating system IP stacks prefer IPv6 to IPv4 when both protocols are available to be used in communicating with another system. A *dual-stacked* system is a system with both IPv4 and IPv6 protocol stacks available and configured. When an IPv6 user encounters an IPv6 system across a relatively poor IPv6 network, the user-perceived performance is considerably degraded. When this occurs, a frustrated user may hastily conclude that the problem lies with IPv6.

As IPv6 connectivity becomes available, some advanced users start experimenting with IPv6, possibly by using IPv6-in-IPv4 tunnels. Typically, they find IPv4 connections performing better than IPv6 connections. As initial experimental interest fades away, some users stop using IPv6 altogether, and may unintentionally leave poorly managed IPv6 networks behind. If use of IPv6 fails, communication automatically falls back to IPv4. Many users are not aware of their use of IPv6, nor problems in the IPv6 network. IPv6 network problems are often overlooked because of the transparent design of IPv6 systems.

Making the IPv6 Internet fully functional will require a major change. No simple solution appears, other than fixing each individual path problem as it is identified. Although traditional tools such as `ping` and `traceroute` are useful for investigating IPv4 and IPv6 independently, we can gain a better understanding of IPv6 problems with tools specifically designed to compare IPv6 and IPv4 measurements. By comparing IPv6 and IPv4 paths, we can focus on problems that are present only in the IPv6 path.

We are exploring methods to illustrate IPv6 network problems that provide insight to network operators and system administrators. Our approach makes use of the availability of both of the IPv4 and IPv6 protocol stacks to compare the two types of paths. Our results appear promising for understanding the status of IPv6 deployment and for improving the quality of the IPv6 Internet.

One can measure and diagnose problems in the IPv6 Internet using similar techniques to those used in the IPv4 Internet. Most network management tools available for IPv6 are simple replacements of the tools developed for IPv4. Because the IPv6 Internet is being deployed via tunnels over as well as in parallel (native) with the existing IPv4 Internet, we can develop new techniques specifically designed to manage both networks. Our focus is on dual-stack tools that measure and

compare IPv4 and IPv6 paths to provide insight to network operators and system administrators.

4.2 Methodology

Our methodology is simple. First, by monitoring DNS messages, we create a list of systems with IPv6 and IPv4 addresses in actual use. Second, we measure delay with `ping` to each address in order to select a few nodes per site based on the IPv6:IPv4 round-trip time (RTT) ratios. Finally, we run `traceroute` with Path MTU (PMTU) discovery[207] to the selected sites, and visualize the results for comparative path analysis.

4.2.1 Dual-Stack Node Discovery

It is challenging to produce an address list that provides a reasonable coverage of dual-stacked sites and systems in the world. Existing studies often select targets semi-manually from a larger set such as a client list obtained from server access logs. Our approach is to monitor DNS responses and record those with AAAA Resource Records (RRs). A AAAA (quad-A) record maps an IPv6 address to a hostname in a similar way to how an A record maps an IPv4 address to a hostname. We assume that a DNS response with AAAA records indicates that an IPv6 address is likely to be in actual use without prejudging the service it offers.

We define a dual-stack node as a system that has both IPv4 and IPv6 protocol stacks implemented and configured for operation. Our measurement targets are only those nodes with both IPv4 and IPv6 addresses registered in the DNS. Since an IPv6 address can be automatically configured, having an IPv6 address configured does not necessarily indicate an intention to use IPv6. When a system has an IPv6 address registered in the DNS, we assume it is intended to provide some service over IPv6. Although there are cases where a hostname points to topologically different IPv4 and IPv6 addresses, we do not distinguish them, since they are the same service from a user's point of view.

To find dual-stack nodes in real use, we passively monitor for DNS responses and record hostname and IPv6 address pairs appearing in the answer, authority, and additional sections. Many IPv6-capable clients first search for IPv6 addresses of a hostname, and then for IPv4 addresses of the same name. The answer section contains the Resource Records that answer a query, for example, a AAAA record containing an IPv6 address for a given hostname. The authority and additional sections provide auxiliary information about the authoritative name servers for the hostname and/or address. We extract any name server information from the authority and additional sections because we prefer DNS servers as measurement targets, since they are generally well-maintained and robust to occasional measurement.

From the list obtained, we extract nodes that have legitimate global unicast IPv6 addresses, and perform DNS lookups for both IPv4 and IPv6 addresses for the hostname. This is to confirm that the nodes actually have both IPv4 and IPv6 addresses for the given hostnames. This process provides a list of target dual-stack nodes for use in the dual-stack ping measurement.

4.2.2 Dual-Stack Ping

Our dual-stack ping is a script that obtains the IPv4 and IPv6 RTT delays for a set of target nodes by running `ping` and `ping6`.

ICMP-based RTT measurement bears well-known limitations: many firewalls filter ICMP packets, and some routers process ICMP packets in the slow forwarding path, rendering measured RTT artificially larger than that of other packets. Nonetheless, `ping` provides an estimation of the comparative difference between IPv6 and IPv4 that is close enough for our purposes.

From the dual-stack ping results, we can identify the percentage of dual-stack nodes reachable only by IPv4 even though they have AAAA records. When a node is unreachable by both IPv4 and IPv6, the target node may be off-line,

or there may be a network problem not specific to IPv6. The number of nodes reachable only by IPv6 is not reliable since many sites filter ICMP for IPv4 but not for IPv6. Conversely, it is unlikely that ICMP is filtered only for IPv6. Therefore, we assume that when a node is reachable only by IPv4, it is an indication of IPv6 network problems that need further investigation.

From this set of nodes we select a few representative nodes per site. By the current IPv6 address assignment rules, we assume an organization has a fixed prefix length of 48 bits, which is a Site-Level Aggregation or SLA[127], where a site is loosely defined as an organizational unit in a single geographical location.

We select up to two representative nodes for each /48 using the following rules. Nodes reachable by both IPv4 and IPv6 are classified by their IPv6:IPv4 RTT ratios into 3 groups; Large ($ratio > 1.25$), Small ($ratio < 0.8$), and Equal ($0.8 \leq ratio \leq 1.25$). One node is selected from each group, except when both the Large and Small groups are not empty then the Equal group is omitted.

A node with the largest (smallest) RTT ratio is selected as a representative node for the Large (Small) group. For the Equal group, we select one with the shortest string length for the concatenated numeric-address and hostname. This works well for selecting suitable targets since important servers tend to have a manually assigned shorter address form (e.g., prefix::1) and a shorter hostname (e.g., ns.example.com).

If the /48 has no node reachable by both IPv4 and IPv6 but there is a node reachable only by IPv4, we select the representative node using the same heuristics as used for the Equal group.

Often only one node is selected for a site because all nodes in the site share the same network path. If a specific node has a large RTT, we select it along with another representative node, to facilitate comparative analysis in distinguishing a node problem from a site problem.

We also take the distribution of the IPv6:IPv4 RTT ratio among the nodes reachable by both IPv4 and IPv6. We categorize the distribution into different geographical regions to observe regional differences. We base our classification on the publicly available IP address assignment database provided by the Regional Internet Registries (RIR). The resulting statistics provide an estimation of the quality of the IPv6 network relative to that of IPv4.

4.2.3 Dual-Stack Traceroute and Visualization

The third step is to identify specific problems and their causes through discovering and visualizing the forward topology. Most problems lie in routing, e.g., routing loops, vanishing routes, and roundabout routes. A roundabout route is not always caused by a routing problem *per se*, but by a lack of peering or IPv6-capable paths. Because IPv6 exchange points and paths are still fairly limited, a packet could travel much further with IPv6 than the same packet might travel with IPv4. One of our goals is to identify a lack of peering or paths for IPv6. Another related problem is poorly configured tunnels that disregard the underlying topologies. Tunnels are useful during the early stages of IPv6 deployment, but poorly configured tunnels, especially in the backbone, present performance problems and other issues when left untended after infrastructural changes.

It is difficult to identify path problems by simply running the traditional `traceroute` program, since it often requires comparative analysis of multiple paths using knowledge of the underlying topology. Our method employs visual comparison of IPv4/IPv6 path pairs to intuitively recognize path anomalies. If necessary, we can use traditional `traceroute` to further investigate details of a path in question.

Our dual-stack traceroute tool is `scamper`[185], successor of `skitter`[129]. Both `skitter` and `scamper` are designed for large scale topology measurement, run multiple traceroutes in parallel at

a specified packet-per-second rate, and terminate a trace as soon as the destination is detected to be unreachable. In addition, `scamper` can probe both IPv4 and IPv6 addresses, and has the ability to perform PMTU discovery.

We use PMTU discovery to identify IPv6-in-IP4 tunnels, since a drop in MTU at an intermediate router indicates a possible tunnel entry point. It is useful to identify tunnels, especially those ignoring the underlying IPv4 topology. The tunnel discovery is also useful for troubleshooting since problems in tunnels are often caused by the underlying IPv4 networks and hard to debug with IPv6 tools alone. Colitti *et al.* use PMTU discovery for tunnel detection in [43] and propose several techniques to infer and confirm the existence of IPv6-in-IPv4 tunnels. We use only PMTU discovery because tunnel detection is not the goal and is only used as auxiliary information for path analysis.

The visualization script reads the output of `scamper`, and creates graphs comparing IPv4 and IPv6 path pairs. The graph juxtaposes IPv4 and IPv6 path pairs for neighboring destinations, and plots intermediate hops according to their RTTs.

4.3 Results

Data was collected by measurement from three locations in June, 2004. The three locations are 1) WIDE[326], a research network in Tokyo, Japan; 2) IIJ[138], an ISP providing commercial IPv6 services in Tokyo, Japan; and 3) Consulintel[45], in Madrid, Spain, directly connected to MAD6IX that is part of Euro6IX[91]. These three measurement points are arguably among the best connected IPv6 sites in the world, and are referred to as the WIDE, IIJ, and ES sites in this paper.

4.3.1 Dual-Stack Node Discovery Results

We set up several DNS monitors within the WIDE network from April to June in 2004. By monitoring AAAA records, we obtained 11,834 unique hostname and IPv6 address pairs. Table 4.1 shows a breakdown of the obtained IPv6 address prefixes.

We extracted a total of 8,347 pairs for '2001::/16' and '3ffe::/16' since only global unicast IPv6 addresses are of interest. We then performed DNS lookups by hostname for A and AAAA records, and found that 4,711 pairs actually have both A and the matching AAAA. After removing invalid IPv4 addresses (e.g., RFC1918 addresses and local addresses) and duplicates with identical IPv6 and IPv4 address pairs but with different host names, we obtained 4,086 target dual-stack nodes.

Table 4.2 shows the distribution of the target dual-stack nodes by their country code, representing 47 countries. We obtain the country code for each pair by matching the IPv6 addresses against the allocated IPv6 prefix in the RIR's database. We use the country code of the address block assignee in the database entry. Limiting this approach is that the real location of a node may be different from the registered country. In addition, we do not consider the associated IPv4 addresses at all.

Table 4.1. IPv6 address prefixes captured within the WIDE network

prefix	prefix use	pairs
2001::/16	Aggregatable Global Unicast for sub-TLA	6,585
3ffe::/16	6bone	1,762
2002::/16	6to4	241
::ffff/96	IPv4-mapped IPv6 address	97
::/96	IPv4 compatible IPv6 address	31
fe80::/10	Link-local Unicast	6
fec0::/10	Site-local Unicast	2
other	reserved or unassigned address	3,110

Table 4.2. Number of dual-stack targets by country code based on their IPv6 address

JP:1155	ID:79	NO:34	KR:17	LU:9	PH:4
NL:497	FI:68	CZ:30	MY:17	RU:8	TN:4
US:464	IT:68	DK:29	BR:16	TH:8	YU:4
DE:431	SK:68	TW:27	HU:13	ZA:8	AR:2
FR:251	CH:59	AT:25	LT:13	BE:6	RO:2
UK:186	PL:57	EU:21	CN:10	SG:6	CL:1
CA:144	AU:41	EE:18	MX:10	GR:4	IL:1
SE:93	IE:39	PT:18	ES:9	HK:4	

4.3.2 Dual-Stack Ping Results

We performed the dual-stack ping from the three locations, from the WIDE and IIJ sites on June 10 and from the ES site on June 23, using the same list obtained by the dual-stack node discovery within the WIDE network. Table 4.3 lists the numbers of unreachable and reachable nodes by IPv4 and IPv6 from the WIDE site. The results from IIJ are almost identical, and the results from ES are similar to WIDE's. About 66% are reachable by both IPv4 and IPv6. However, about 16% are reachable by IPv4 but not by IPv6 even though they have AAAA records. These sites would force communicating peers to timeout with IPv6 before falling back to IPv4. In Table 4.3, the nodes are classified into four regions by matching their IPv6 address prefixes to the RIR database; 'jp' for Japanese nodes, 'apnic' for non-jp APNIC nodes, 'arin' for ARIN and LACNIC nodes, 'ripe' for RIPE NCC nodes. Japanese nodes are separated from other APNIC nodes since their node number is large and most of the Japanese nodes are in Tokyo, so that their RTT is usually less than 10 msec from the WIDE and IIJ sites. Since the number of LACNIC nodes is so small, we merge them with the ARIN nodes.

When we examined the two middle groups, those that had only IPv4 or IPv6 responding addresses, there was an unusual difference in the ratio of addresses in these two groups across different RIRs. The ratios in Japan and RIPE NCC were around 0.6, 0.57 and 0.67 respectively. In contrast, ARIN was about half that with 0.23 and APNIC was almost four times with 2.43. The low level of IPv6 responding in ARIN could be the result of the low level of commitment to IPv6 in the US, which makes up a large portion of ARIN's membership. The surprisingly strong ratio of responding IPv6 addresses in APNIC might be the result of stronger support for their relatively large IPv6 blocks in comparison to their small IPv4 allocations.

Figure 4.1, 4.2, 4.3 show the scatter graphs of the observed RTTs. We plot a node's IPv4 RTT on the X-axis and its IPv6 RTT on the Y-axis. Each graph plots about 2,700 nodes that were

Table 4.3. Number of unreachable and reachable nodes by dual-stack ping from WIDE.

IPv6		unreach	unreach	OK	OK
IPv4		unreach	OK	unreach	OK
total	4086 (100%)	370 (9.0%)	634 (15.5%)	384 (9.4%)	2698 (66.0%)
jp	1155 (100%)	83 (7.2%)	126 (10.9%)	72 (6.2%)	874 (75.7%)
apnic	213 (100%)	37 (17.4%)	28 (13.2%)	68 (31.9%)	80 (37.6%)
arin	645 (100%)	80 (12.4%)	168 (26.1%)	38 (5.9%)	359 (55.7%)
ripe	2042 (100%)	162 (7.9%)	306 (15.0%)	204 (10.0%)	1370 (67.1%)

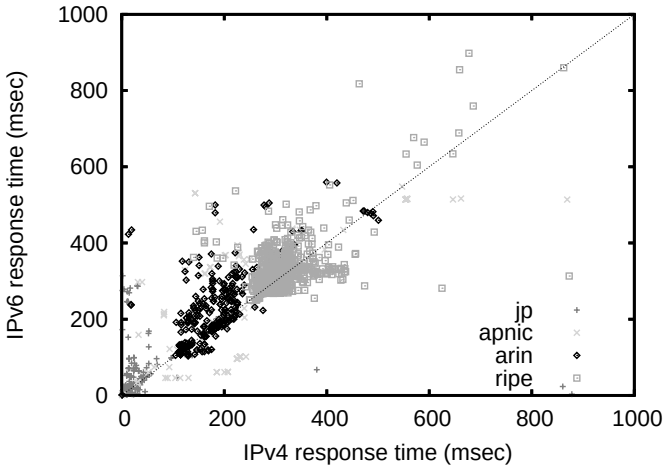

Fig. 4.1. Distribution of IPv6/IPv4 RTT from WIDE

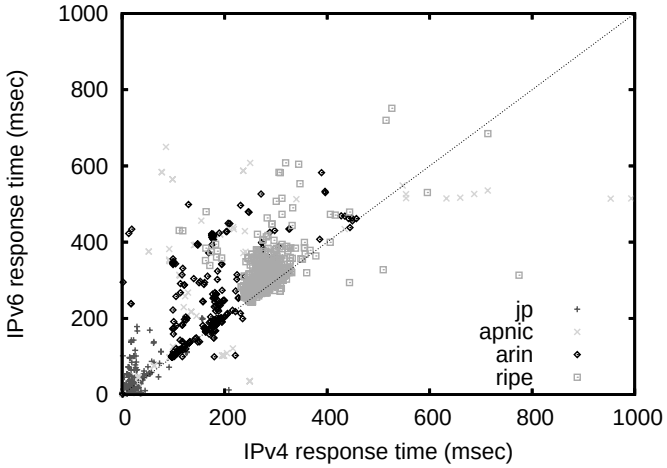

Fig. 4.2. Distribution of IPv6/IPv4 RTT from IIJ

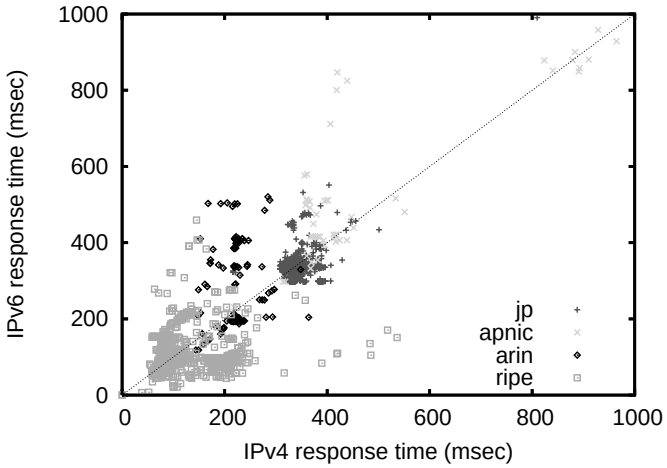

Fig. 4.3. Distribution of IPv6/IPv4 RTT from ES

reachable by both IPv4 and IPv6. When the response time of IPv6 is equal to that of IPv4, we plot the node on the unity line, $y = x$. For nodes above this line, IPv4 outperforms IPv6, and for nodes below this line, IPv6 outperforms IPv4. We again categorize the nodes into four regions.

These results indicate that the majority of the nodes have similar RTT for both IPv4 and IPv6. A number of individual nodes far above the unity line have IPv6 performance issues specific to the node or the site. The clusters above the unity line indicate the existence of roundabout paths within the backbone network.

Compared to WIDE, IIJ has fewer nodes below the unity line, probably due to Acceptable Use Policies (AUPs) of academic IPv6 networks. The ES site has a large cluster of RIPE nodes below the unity line, likely connected through Euro6IX[91]. The majority of nodes are around the unity line; the percentage of nodes whose IPv6:IPv4 RTT ratio is less than 1.25 is 80.1% for WIDE, 74.3% for IIJ, and 82.5% for ES.

APNIC nodes have large variance in RTT ratios due to its topological diversity; many APNIC countries are connected to Japan through the US or Europe, and many satellite links connect islands. Also, some networks are funded to promote IPv6, such that there are nodes with a direct IPv6 path but with an IPv4 path that must go through the US.

Next, we select representative nodes for each /48 using the rules described in Section 4.2.2. For the WIDE site, we selected 1,334 nodes out of 4,086 nodes for 1,469 /48s. For the IIJ and ES sites, we selected 1,320 and 1,310 nodes, respectively. We selected fewer nodes than the number of 48-bit prefixes since we selected no nodes in sites not reachable by IPv4. The reduction rate is about 1/3 for these results, but it improves if more nodes are available per site.

4.3.3 Dual-Stack Traceroute Results

We ran `scamper` to the representative nodes with PMTU discovery on June 11, 2004 from the WIDE and IIJ sites, and on June 16 from the ES site. To visualize the results, a set of

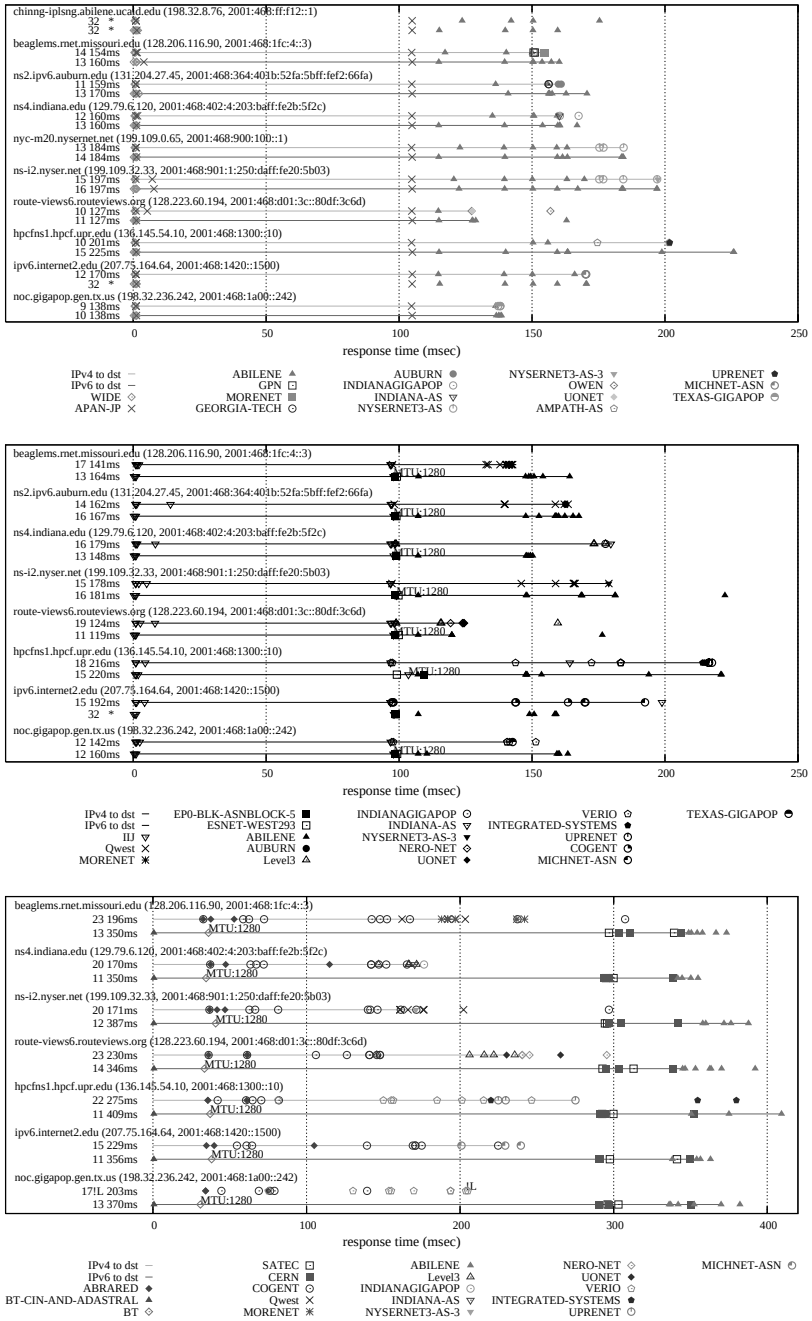

Fig. 4.4. Path visualization towards 2001:468::/16 from WIDE (top), IIJ (middle) and ES (bottom)

scripts divide the `scamper` output into smaller target groups and create a graph for each group. Each graph contains 10 target nodes, yielding about 130 graphs for each measurement; the script also creates a web page for each graph along with scrollable `scamper` text output. While there have been a number of attempts to visualize traceroute-derived topology[129, 243], we are not aware of published work that extensively compares IPv4 and IPv6 paths. In the graph we map IP addresses into Autonomous System (AS) numbers to simplify the presentation[130, 187, 261].

Figure 4.4 shows example outputs of the `scamper` visualization towards 2001:468:X::/48, the nodes within the ABILENE address block (selected simply because it is less controversial for publishing results). The top graph is from the WIDE site, the middle graph is from the IIJ site, and the bottom graph is from the ES site. The target nodes are slightly different for each measurement site since they are selected based on the dual-stack ping results of each site.

For each target node in the graph, two lines are drawn from the source to the destination, the upper line for IPv4 and the lower line for IPv6. A missing line indicates the destination is unreachable. To the left of the line, the graph shows the total hop number and destination RTT.

The graphs plot intermediate hops at their RTTs from the source. We map IP address of a hop to its AS number by finding the best matching prefix and origin AS in the publicly available Routeviews BGP table[264]. There were 165,289 prefixes for IPv4 and 520 prefixes for IPv6 in the BGP table at the time of measurement.

When a drop in MTU is detected, the graph marks the MTU on the right side of the hop; it suggests a likely tunnel between this hop and the previous hop. If `traceroute` terminated with an error, the graph marks the error code at the hop using the `traceroute` notations (e.g., '!X' for communication administratively prohibited).

In the WIDE and IIJ graphs, most destinations have similar RTTs for IPv4 and IPv6. In the WIDE graph, the IPv6 paths are similar to the IPv4 paths, therefore they appear to be IPv6-native dual-stack paths. In contrast, the IIJ graph shows IPv6 paths going through ASes different from IPv4 paths, which is more common in the current IPv6 Internet. AS-level comparison yields insight into path differences since many IPv6 paths do not follow their IPv4 counterparts.

In the ES graph, the IPv6 paths are much longer in time than the IPv4 paths. All the IPv6 paths share the long hop after the hop with 1280 MTU, consistent with a tunnel that makes a detour to the destinations. Note that this is not a typical path to the US from the ES site; we observed better paths to other US destinations in other graphs but did not include them in this paper.

Table 4.4 shows the distribution of the Path MTU size detected by `scamper`. Since the target nodes include nodes reachable by `ping` but not by `ping6`, the number of nodes unreachable by `ping6` is shown at the bottom. A 1454-byte MTU is common for PPPoE, and 1280 and 1480 bytes are default MTU sizes for popular tunnel implementations. WIDE stands out with a high number of 1500-byte IPv6 PMTUs, likely a result of their efforts to promote native IPv6 connections.

Table 4.4. distribution of Path MTU size

	IPv4			IPv6		
PMTU	WIDE	IIJ	ES	WIDE	IIJ	ES
1500	761	751	732	575	76	33
1492	6	6	8	—	—	—
1480	2	2	1	64	96	15
1476	2	1	1	8	2	—
1472	1	1	1	2	—	—
1456	—	—	1	—	—	—
1454	90	95	82	—	—	—
1450	—	—	—	—	2	—
1448	2	2	2	—	—	—
1446	1	1	1	—	—	—
1400	—	—	1	—	—	—
1280	1	1	1	184	622	500
1258	1	—	1	—	—	—
unknown	46	43	44	47	21	83
unreach	421	417	433	454	501	679
unreach by ping6	—	—	—	249	276	273

4.4 Conclusion

It is essential to IPv6 deployment to improve the quality and performance of the IPv6 Internet. In order to illustrate IPv6 network problems for network operators, we are developing tools to compare IPv6 measurements with corresponding IPv4 measurements. Our techniques include the dual-stack node discovery for finding dual-stack nodes, the dual-stack ping for selecting representative nodes, and `scamper` for detailed path analysis. Our test results indicate that we can improve IPv6 network quality by identifying and fixing a limited amount of erroneous settings.

Our tools are still under development and need improvements. We plan to fully automate the measurement procedure in order to perform regular measurements and archive results. This long-term measurement strategy will provide a way to evaluate the progress of IPv6 deployment. We would also like to increase the coverage of measurement points including developing countries. Another important step is to establish procedures to notify responsible parties of problems we find using our tools.

The results of our measurements, along with our tools, are available from `http://mawi.wide.ad.jp/mawi/dualstack/`.

第 5 章　NeTraMet による Root DNS サーバ群の計測

5.1 NeTraMet とは

NeTraMet とは、CAIDA[46] によって開発された、トラフィックフローを計測するためのツールである。CAIDA は、主にインターネットにおけるさまざまな計測に関する研究を行っているグループである。

NeTraMet は、 RFC2720[23]、 RFC2721[22]、RFC2722[26]、RFC2723[24]、RFC2724[117] にて標準化された、Real Time Flow Measurement（RTFM）にしたがって作成されている。フロー観測の記述言語として、Simple Ruleset Language（SRL）を用いており、柔軟なフロー計測を可能としている。

NeTraMet ツールは、実際にフロー計測を行う NeTraMet コマンドと、計測結果を SNMP にて取り出してファイルに記述する NeMaC コマンドの 2 つのコマンドからなる。Linux、FreeBSD、NetBSD といった一般的なフリー UNIX 系 OS の上で動作する。計測を行うためには、NeTraMet が動作するホストにて次の条件が必要となる。(1) 計測対象となるトラフィックが、NeTraMet を動作させるインタフェースにて観測できる。(2) 特権（ルート権限）にて NeTraMet を動作させることができる。

すなわち、NeTraMet は実トラフィックを監視して計測するという静的な計測ツールであるため、スイッチングハブにおけるポートミラー設定やトラフィック分岐装置などによってトラフィックを複製し、ホストにて監視できる環境が必要となる。

NeTraMet パッケージの最新版は、`http://www2.auckland.ac.nz/net/NeTraMet/` から入手可能である。

5.2 NeTraMet による計測

MAWI WG では、NeTraMet を使って Root DNS サーバ群への名前解決要求ならびに応答クエリの計測を行っている。これは、複数地点から Root DNS サーバ群への到達性を計測し、Root DNS サーバのエニーキャスティングの有効性を検証しようという活動の一部として行われている。Root DNS サーバのエニーキャスティングとは、複数地点に同じ IP アドレスを持つ Root DNS サーバを設置して、同じアドレスブロックを BGP にて広告することによって、クエリの分散処理を行う技術である。これによって、RTT が小さくなり、サービス妨害攻撃や故障への耐性が増すと考えられている。

一方、NeTraMet の開発元である CAIDA においても、Root DNS サーバ群ならびに gTLD DNS サーバ群へのクエリの計測を行っている。この計測は CU Boulder、University Auckland、University of California San Diego（UCSD）の 3 地点で行われている。計測結果は論文 [25, 27] に示されている。

CAIDA による最新の計測結果は、`http://www.caida.org/cgi-bin/dns_perf/main.pl` にて確認できる。

5.3 WIDE Project における計測

前節にて述べたとおり、WIDE Project においても NeTraMet による Root DNS サーバへのクエリ計測を行っている。設置地点は、慶応義塾大学と東京大学の 2 地点である。慶応大学においては、慶応大学湘南藤沢キャンパスと、そのネットワークの上流となる WIDE Project との中間に位置するスイッチにおいてポートミラー設定を行い、NeTraMet を設置した。東京大学においては、東京大学とその上流となる学術情報ネットワーク（SINET）との間において光分岐装置を設置し、NeTraMet を設置した。

すなわち、慶応大学においては湘南藤沢キャンパスにて発生する RootDNS サーバへのクエリを計測し、東京大学においては東京大学全体から発生する Root DNS サーバへのクエリを計測することとなる。

計測するためのホストの仕様は、次のとおりである。慶応大学は CPU に Pentium III 800 Mhz、メモリ 256 MB を搭載したホストに、1000base-SX インタフェースを用いて計測を行う。東京大学では Xeon 3 Ghz 2 個、メモリ 1 GB のホストに、同じく 1000base-SX インタフェースを用いて計測を行う。なお、NeTraMet パッケージのバージョンはどちらも Ver 5.1b2 を利用している。これは、Ver 4 以前の NeTraMet では、DNS クエリの RTT を正確に測定することができないというバグがあったからである。

計測対象は A.root-servers.net から M.root-servers.net までの 13 個の Root DNS サーバである。計測に利用した SRL を図 5.1 に示す。

この SRL 設定によって、各 Root DNS サーバへの問い合わせならびに応答を記録し、それぞれの数や応答を得られるまでにかかった RTT を記録している。この計測結果は NeMaC によって 5 分単位で記録される。

NeMaC によって記録されるデータの例をは図 5.2 に示す。この図では見やすいように適切に改行を入れた。#Format で始まるコメントの段落に、NeMaC が記録する数値の意味が明記されている。19 という数値にて始まるそれぞれの段落が、1 つの Root DNS サーバに関する 5 分間の計測結果を示している。

WIDE Project においては、この NeTraMet による Root DNS サーバ群へのクエリ計測を 2003 年 9 月から試験的に開始し、2004 年 1 月から 2 地点における本格的な計測を開始した。

```
define DNS = 53;
define PP_ICMP_ECHO     =   1;

define PP_UDP_DNS       =  11;

define PP_TCP           = 192;
define PP_OK_SYNACK     =   1;  #   ->SYN,  <-SYN+ACK pairs
define PP_OK_SYNRST     =   2;  #   ->SYN,  <-SYN+RST pairs

define PP_OK_MULTI      =   8;  #   ->DATA, <-ACK for more than one packet
define PP_OK_SINGLE     =  16;  #   ->DATA, <-ACK 'lone' packet
define PP_OK_INGROUP    =  32;  #   ->DATA, <-ACK single packet in a group

define A_ROOT = 198.41.0.4/32;        #  Verisign, Dulles, Va
define B_ROOT = 192.228.79.201/32;    #  ISI, Marina del Ray, Ca
define C_ROOT = 192.33.4.12/32;       #  Cogent, Herndon, Va
define D_ROOT = 128.8.10.90/32;       #  U Maryland, Md
define E_ROOT = 192.203.230.10/32;    #  NASA Ames, Ca
define F_ROOT = 192.5.5.241/32;       #  ISC, Palo Alto, Ca
define G_ROOT = 192.112.36.4/32;      #  DoD NIC, Vienna, Va
define H_ROOT = 128.63.2.53/32;       #  ARL, Abdereen, Md
define I_ROOT = 192.36.148.17/32;     #  KTH, Stockholm
define J_ROOT = 192.58.128.30/32;     #  Verisign, Dulles, Va
define K_ROOT = 193.0.14.129/32;      #  RIPE NCC, Amsterdam
define L_ROOT = 198.32.64.12/32;      #  IANA, Los Angeles, Ca
define M_ROOT = 202.12.27.33/32;      #  WIDE, Tokyo

define B_ROOT_OLD = 128.9.0.107/32;       #  ISI, USC, Ca
define J_ROOT_OLD = 198.41.0.10/32;       #  NSI, Herndon, Va

define TestDestAddress =
   if DestPeerAddress == A_ROOT
      { store FlowKind :=  1\;  store FlowClass := 0\; }
   else if DestPeerAddress == B_ROOT || DestPeerAddress == B_ROOT_OLD
      { store FlowKind :=  2\;  store FlowClass := 0\; }
   else if DestPeerAddress == C_ROOT
      { store FlowKind :=  3\;  store FlowClass := 0\; }
   else if DestPeerAddress == D_ROOT
      { store FlowKind :=  4\;  store FlowClass := 0\; }
   else if DestPeerAddress == E_ROOT
      { store FlowKind :=  5\;  store FlowClass := 0\; }
   else if DestPeerAddress == F_ROOT
      { store FlowKind :=  6\;  store FlowClass := 0\; }
   else if DestPeerAddress == G_ROOT
      { store FlowKind :=  7\;  store FlowClass := 0\; }
   else if DestPeerAddress == H_ROOT
      { store FlowKind :=  8\;  store FlowClass := 0\; }
   else if DestPeerAddress == I_ROOT
      { store FlowKind :=  9\;  store FlowClass := 0\; }
   else if DestPeerAddress == J_ROOT || DestPeerAddress == J_ROOT_OLD
      { store FlowKind := 10\;  store FlowClass := 0\; }
   else if DestPeerAddress == K_ROOT
      { store FlowKind := 11\;  store FlowClass := 0\; }
   else if DestPeerAddress == L_ROOT
      { store FlowKind := 12\;  store FlowClass := 0\; }
   else if DestPeerAddress == M_ROOT
      { store FlowKind := 13\;  store FlowClass := 0\; }


optimise 3;

   if SourcePeerType == IPv4 save;
   else ignore;

   if SourceTransType == UDP save;
   else ignore;

   TestDestAddress;  # Sets FlowKind
   if FlowKind == 0 nomatch;
   else {

      if DestTransAddress == DNS save;  # Avoid 'match on non_DNS flow' msg
      else ignore;

      save ToTurnaroundTime = 120.11.0!0 & 4.2.10!7000;
      count;
      }

set dns_root_wide;
statistics;
format
  FlowRuleSet FlowIndex FirstTime SourcePeerType SourceTransType
  "   " FlowKind FlowClass
  "   " ToPDUs FromPDUs
  "   " ToLostPDUs FromLostPDUs
  "   (" ToTurnaroundTime
  ")";
```

図 5.1. SRL 設定

```
#flows starting at 15:00:00 Mon 3 Jan 2005 (UTC)

#Format: flowruleset flowindex firsttime sourcepeertype
#sourcetranstype flowkind flowclass topdus frompdus tolostpdus
#fromlostpdus (toturnaroundtime1) Time:

19 4 1039 1 17 2 0 6253468 5318492 1152512 217536 (5 2 10 7000 63 11 0
0 1913 1904 1902 1900 1839 1903 1903 1902 1902 1874 1917 1914 1843
1834 1903 1828 1862 1830 1830 1903 1904 1903 1835 1834 1830 1829 1829
1829 1903 1900 1904 1902 1834 1915 1904 1903 1900 1899 1903 1899 1933
1925 1905 1829 1904 1907 1908 1907 1922 1904 1904 1902 1904 1905 1834
1838 1914 1868 1858 1827 1913 1837 1915)

19 5 1040 1 17 9 1 12735380 12663671 678923 607214 (5 2 10 7000 63 11
0 0 3025 2976 3055 3017 3048 3045 3081 2978 3011 3011 3010 3055 3063
3071 3056 2967 3013 3057 3040 3022 2943 3035 3020 3056 2941 3040 3053
3030 3030 3041 3026 3055 3027 3058 3046 3063 3030 3013 3042 3041 3023
3036 3028 3045 3046 3061 3076 3053 3055 3042 3025 2940 3047 3038 3051
3063 2988 2942 3065 3025 3030 3017 3048)

19 6 1041 1 17 3 1 16703780 16469505 923629 689354 (6 2 1965 7000 120
11 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 3 1 1 1 0 2 0 1 0 2 2 2 1 1 2 2 2
3 0 2 1 1 1 2 2 1 1 0 2 1 1 1 0 2 1 0 0 0 3 2 0 2 0 1 5 0 0 2 5 2 5 4
2 2 2 3 4 1 3 3 1 3 4 6 1 2 4 3 3 5 0 5 6 0 2 3 2 2 3 2 2 1 1 0 1 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0)

19 7 1046 1 17 13 0 37799759 36803333 1532996 536570 (6 2 7874 7000
120 11 0 0 648 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0)

19 8 1046 1 17 6 1 13515243 13363098 757920 605775 (5 2 10 7000 75 11
0 0 2651 2647 2510 2494 2486 2484 2484 2515 2589 2562 2550 2566 2566
2576 2586 2571 2562 2532 2550 2564 2552 2574 2587 2551 2545 2547 2569
2544 2587 2551 2549 2674 2636 2655 2655 2646 2646 2659 2641 2528 2554
2586 2536 2564 2580 2575 2486 2471 2495 2490 2495 2504 2505 2662 2660
2529 2575 2594 2595 2472 2518 2495 2486 2555 2571 2579 2580 2586 2577
2561 2575 2566 2550 2527 2522)

19 9 1049 1 17 11 1 13839538 13330392 1106929 597783 (5 2 10 7000 38
11 0 0 2499 2507 2614 2571 2485 2612 2500 2498 2496 2537 2534 2500
2509 2520 2483 2511 2531 2529 2537 2494 2504 2511 2542 2566 2574 2575
2481 2611 2565 2604 2605 2476 2524 2591 2592 2600 2506 2516)

19 10 1049 1 17 5 1 13773863 13606764 755134 588035 (5 2 10 7000 51 11
0 0 2679 3025 2702 2699 2680 2712 2785 2651 2679 2674 2731 2692 2654
2722 2721 2694 2779 2779 2737 2566 2690 2686 2699 2819 2754 2820 2694
2675 2722 2692 2647 2646 2646 2716 2692 2687 2670 2670 2669 2735 2749
2747 2639 2637 2659 2692 2749 2632 2699 2697 2645)

19 11 1055 1 17 8 1 10480084 10353402 692623 565941 (5 2 10 7000 96 11
0 0 2116 2028 2074 2088 1932 1945 1953 1957 1969 1957 1982 1983 1983
1998 2035 2040 2055 2084 2121 2089 2004 2015 2012 2070 2076 2077 2008
2014 2043 2079 2099 2060 2009 2048 2022 2060 2091 2070 2101 2125 2084
2090 2108 2109 2114 2019 2003 2079 2029 1999 2003 2030 2015 2018 2030
2002 2034 2054 2075 2094 2116 2010 2009 2034 2039 2027 2055 2073 2111
2114 2090 2062 2067 2074 2055 2015 2023 2044 2043 2025 2098 2035 2014
2014 2030 2003 2076 2084 2005 2017 2042 2052 2010 2013 2054 2050)

19 12 1073 1 17 13 1 58832809 57840355 2323094 1330640 (5 2 10 7000 75
11 0 0 2754 2739 2879 2836 2575 2574 2573 2571 2800 2840 2749 2852
2884 2882 2882 2752 2885 2931 2932 2632 2626 2595 2589 2584 2579 2827
2735 2732 2712 2721 2908 2961 2834 2750 2750 2614 2614 2621 2579 2595
2594 2592 2591 2792 2907 2775 2711 2872 2851 2911 2892 2952 2571 2632
2596 2610 2619 2619 2582 2581 2571 2571 2626 2624 2604 2632 2617 2604
2600 2571 2500 2550 2544 3013 2835)

19 13 1082 1 17 3 0 6711146 4381917 2520040 190811 (5 2 10 7000 64 11
0 0 2143 2099 2047 2095 2143 2124 2100 2098 2095 2090 2158 2158 2164
2168 2094 2112 2196 2144 2131 2154 2133 2047 2076 2090 2135 2094 2178
2131 2089 2093 2094 2097 2179 2094 2100 2144 2070 2074 2081 2071 2080
2064 2099 2071 2065 2074 2100 2072 2124 2068 2083 2047 2144 2068 2065
2070 2133 2149 2191 2175 2130 2048 2125 2145)

19 14 1082 1 17 1 0 6782804 6413979 671041 302216 (5 2 10 7000 76 11 0
0 2079 2062 2079 2083 2080 2078 2079 2079 2079 2070 1995 2058 2079
2083 2081 2081 2083 2079 2080 2079 2082 2060 2059 2063 2062 2061 2063
2060 2060 2060 2061 2078 2078 2079 2081 2080 2080 2081 2079 2080 2079
1919 2080 2080 2069 2083 2085 1998 2082 2082 2083 2081 2082 2083 2081
2080 2081 2082 2082 2082 2084 1997 2080 2079 2082 2069 2082 2082 1997
2080 2062 2063 2073 2078 2059 1997)

19 15 1083 1 17 12 1 16445117 16233637 901572 690098 (5 2 10 7000 57
11 0 0 2128 2161 2168 2144 2129 2158 2115 2223 2239 2160 2143 2208
2225 2125 2128 2155 2141 2149 2155 2112 2124 2118 2118 2118 2153 2161
2225 2126 2156 2163 2158 2111 2136 2164 2131 2144 2120 2136 2171 2195
2229 2135 2160 2190 2229 2114 2148 2140 2139 2153 2129 2150 2143 2153
2129 2138 2159)

19 16 1121 1 17 4 1 14702378 14518514 817334 633470 (5 2 10 7000 80 11
0 0 2341 2306 2244 2263 2340 2293 2296 2295 2356 2356 2360 2394 2380
2329 2329 2363 2255 2306 2243 2243 2274 2275 2299 2295 2211 2204 2358
2354 2295 2284 2281 2229 2208 2198 2243 2293 2306 2181 2278 2370 2244
2354 2349 2275 2259 2216 2183 2310 2221 2308 2244 2290 2284 2285 2285
2286 2363 2199 2306 2320 2319 2353 2359 2320 2338 2321 2320 2320 2341
2321 2369 2344 2320 2349 2320 2361 2351 2281 2283 2271)
```

図 5.2.　NeMaC 記録結果

5.4 計測結果

NeTraMet による測定結果の例として、図 5.3 に 2004 年 8 月 7 日の慶応大学における測定結果を示す。

また、図 5.4 に、2004 年 12 月 31 日の慶応大学における測定結果を示す。

縦軸が RTT（ms）もしくは Root DNS サーバに向けて出された名前解決要求のクエリ数を示し、横軸が時間を示している。グラフ中の「＋」の点が RTT を示し、「×」の点がクエリ数を示す。

どちらの日ににおいても、m.root-servers.net に対する RTT が最小となっている。これは m.root-servers.net は WIDE Project によって運営されており、WIDE Project のネットワークから近い場所に位置しているためである。

注目すべきなのは、i.root-servers.net に対する結果である。2004 年 8 月 7 日と 2004 年 12 月 31 日の i.root-servers.net に関する結果のみを図 5.5 に示す。8 月と 12 月の結果で RTT が大きく異なっている。これは、i.root-servers.net が分散拠点を東京に設置し、エニーキャスト [119] を利用したサービスを開始したためである。このため、8 月の時点では海外にある i.root-servers.net のホストに送られていたクエリが、東京の分散拠点にて処理されるようになったため、RTT が小さくなったと考えられる。

さらに、東京大学における計測結果を図 5.6 に示す。

慶応大学に比べ、東京大学のほうが観測されるクエリの全体数が多い。そのため、慶応大学の測定結果に比べて、m.root-servers.net に多量のクエリが送信されていることがはっきりと判別できる。

WIDE Project における NeTraMet の計測結果は、`http://dnstap.nc.u-tokyo.ac.jp/NeTraMet/` にて公開している。

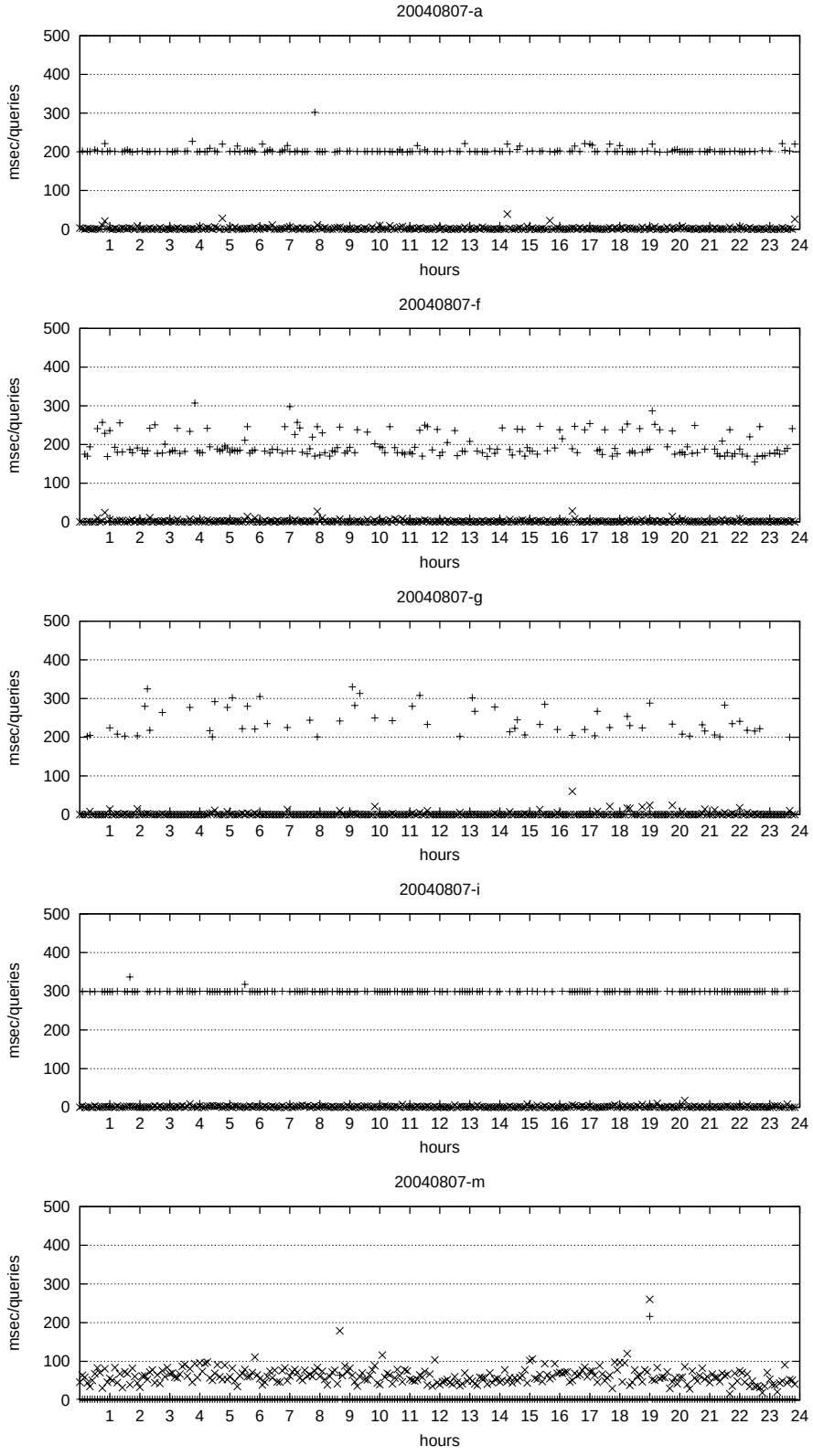

図 5.3.　2004 年 8 月 7 日：慶応大学

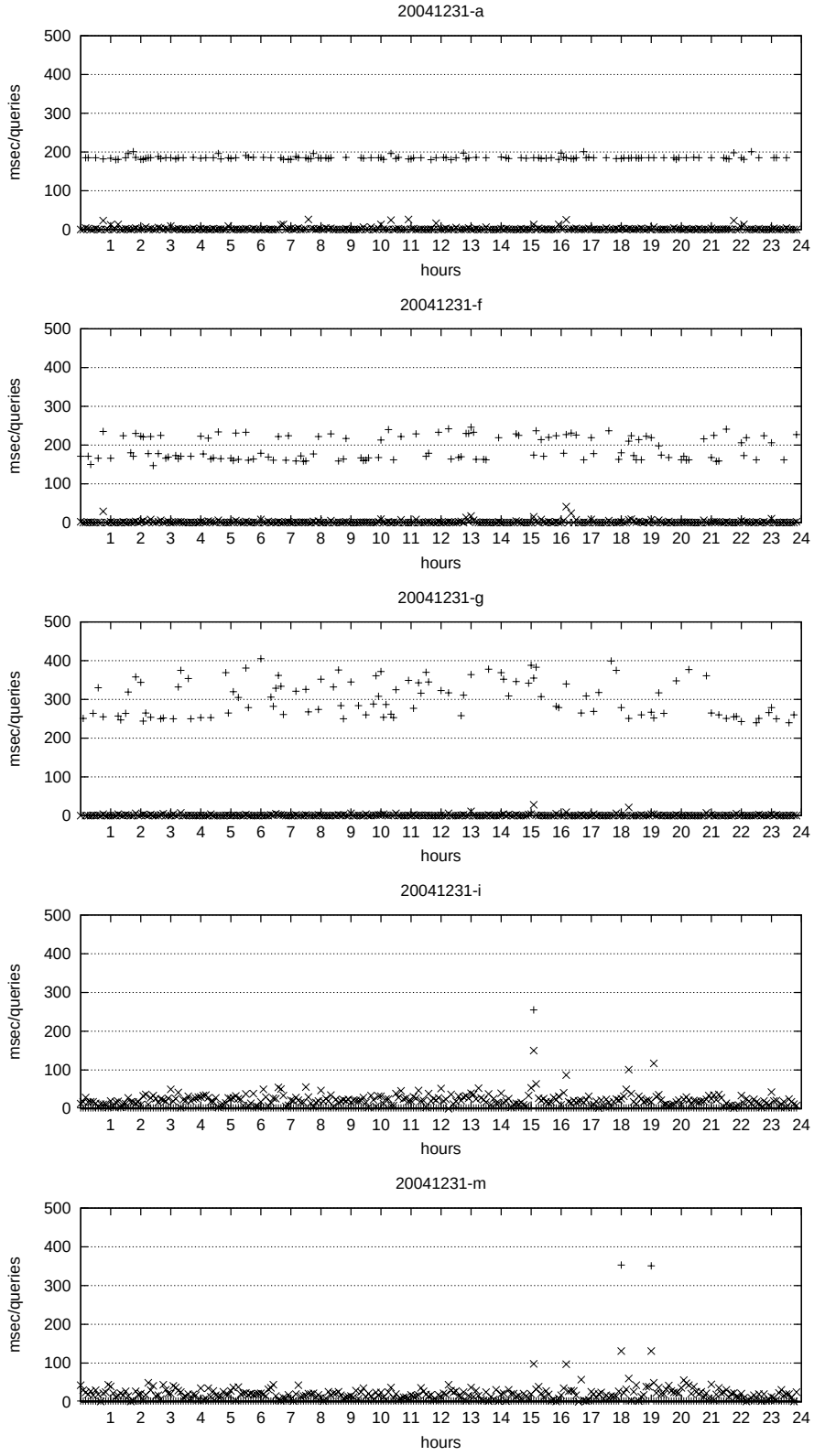

図 5.4. 2004 年 12 月 31 日：慶応大学

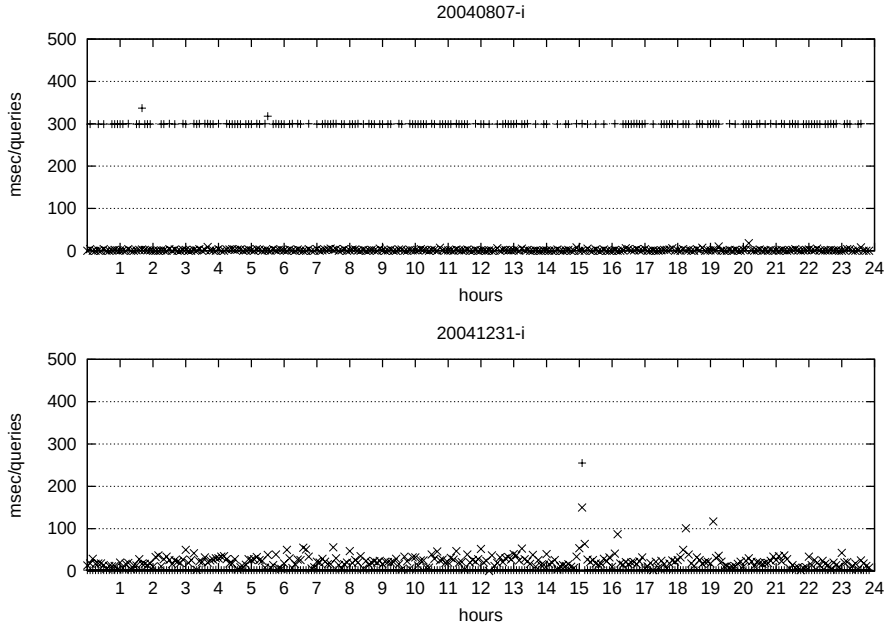

図 5.5.　i.root-servers.net に対する計測結果の比較

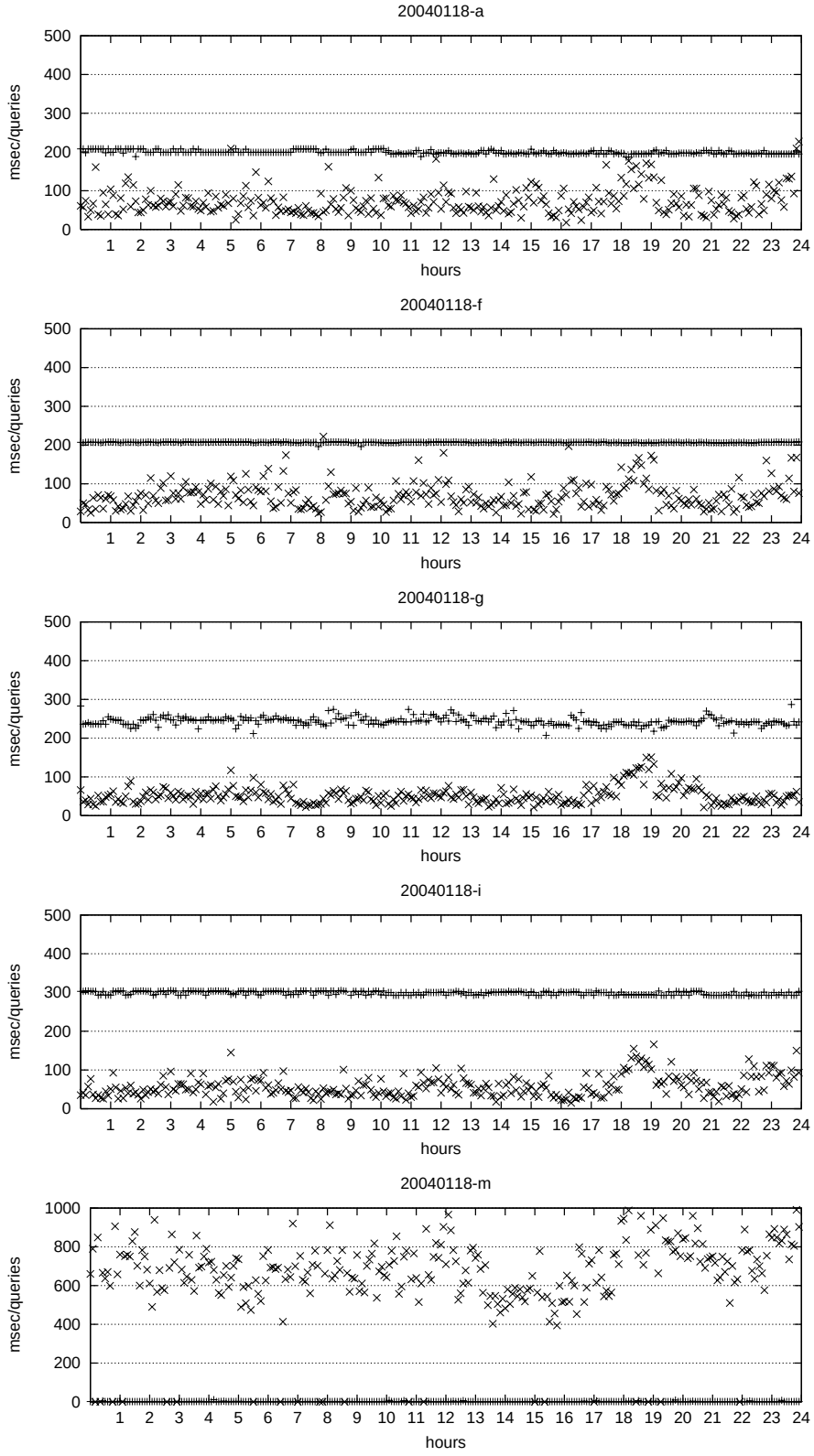

図 5.6. 2004 年 1 月 18 日：東京大学

5.5 結論

このように、NeTraMet を利用して Root DNS サーバへのクエリを計測することができる。しかし、今回の計測でわかったこととして、次の問題点が挙げられる。

- NeTraMet の設置ならびに運用コスト
  NeTraMet を設置し運用するためには、ポートミラーやトラフィック分岐装置などによってトラフィックを複製し、監視する必要があるため、それ相応の設置コスト、運用コストが必要となる。
- エニーキャスト
  多くの分散拠点によってエニーキャスティングが行われた場合、RTT やクエリ数の変化はつかむことができるが、どの分散拠点にて処理されているのかをつかむことが難しい。
- NeTraMet のバッファ不足
  NeTraMet が SRL による柔軟なフロー処理を行うため、単にダンプを行って監視する場合に比べ、処理が重くなる傾向にある。また、複数のインタフェースを監視対象とした場合や、たくさんのトラフィックが発生した場合、フローを記憶しておくためのバッファが不足し、正確に記録できない場合が発生する。今回の計測においても、東京大学での計測においてバッファ不足が発生した期間があった。NeTraMet のログによると、2004 年の 5 月から 12 月まで不定期に発生していた。NeTraMet プロセスに割り当てられるメモリ量の上限を増やし、NeTraMet プロセス起動時に -t オプションにてバッファサイズを増加させることによって対処した。

第 6 章　2004 年度 CAIDA/WIDE 計測ワークショップ報告

6.1 概要

CAIDA（the Cooperative Association for Internet Data Analysis）と WIDE Project は、2003 年度から計測に関する包括的な共同研究を行っている。現在、DNS 計測とトポロジ計測の分野で共同研究を行っており、これらについては、4 章と 5 章で詳細を報告した。共同研究の一環として、2003 年に 2 回、2004 年に 2 回のワークショップを行い、相互の活動を理解し、協力体制を作るようにしている。ここでは、2004 年 4 月と 8 月に行ったワークショップの概要を報告する。

6.2 第 3 回 CAIDA/WIDE 計測ワークショップ

第 3 回 CAIDA/WIDE 計測ワークショップは、2004 年 4 月 22–23 日に USC/ISI で実施した。主な計測テーマは、DNS、IPv6、トラフィックモニタリング、BGP などである。

プログラム

April 22（Thursday）

Session 1: Review of 2003 activities

- Jun Murai（WIDE）、WIDE activities
  10 Gbps クラスのネットワークの展開など、最近の WIDE Project の主な活動を紹介。
- Kenjiro Cho（WIDE）、WIDE measurement activities
  WIDE Project の計測活動の概要として、DNS、IPv6 topology、BGP simulation、netflow/sflow、10G、StarBED、distributed IX、AI3 satellite などを紹介。
- kc claffy（CAIDA）、CAIDA report
  2003–2005 年の CAIDA の研究計画について主に、1) macroscopic topology & routing、2) workload characterization、3) DNS、4) performance、5) IMDC-Trends、6) security の 6 つの研究領域がある。

Session 2: DNS measurements and modeling

- Yuji Sekiya（WIDE）、Comparison of active and passive measurement
  DNS 応答時間計測の手法の比較について。
- Nevil Brownlee（CAIDA）、Measurements and laboratory simulations of the upper DNS hierarchy
  PAM2004 で発表した DNS キャッシュの効果のモデル化について。
- Duane Wessels（CAIDA）、Data collection and analysis for DNS-OARC components
  ISC OARC の活動として開発している DNS 統計情報収集ツール DSC について。
- Akira Kato（WIDE）、Measurements at M-root server

M-root には、US から日本の 5 倍のクエリが来ていることなどを報告。

- Bradley Huffaker（CAIDA）、Sources of strange queries at F-root
  F-root で観測されている異常なクエリの送信元の解析の報告。
- Matthew Luckie（WAND）、IPv6 DNS misconfigurations
  IPv6 アドレスは長いため、DNS の設定をタイプミスしているケースが多く見られるという報告。
- Francisco J. Martin（Oregon State University）、Toward rapid diagnosis and repair of DNS problems
  DNS の問題を自動検出するため、モデル化を行っている。
- Genevieve Bartlett（ISI）、Deploying DNSSEC at a root server
  B-root で行っている DNS 計測の紹介。

April 23（Friday）

Session 3: IPv6 measurements

- Matthew Luckie（WAND）、Active measurements of IPv6 topology: scamper project
  開発中の scamper ツールの紹介と新機能の説明。
- Kenjiro Cho（WIDE）、Dual-stack world: a view from .jp
  日本からデュアルスタック計測を行った結果の報告。

Session 4: Traffic Monitoring

- Bartek Wydrowski（Caltech）、FAST TCP —Internet Congestion Control
  FAST TCP の紹介と、性能評価の試みについて。
- Seiichi Yamamoto（WIDE）、Traffic monitoring by sflow
  WIDE での sFlow を使ったトラフィック計測計画について。
- Colleen Shannon、Security data collection at CAIDA
  外部のモニタから得られたパケットトレースの解析と、使われていないアドレス空間を用いて観測を行うネットワークテレスコープの報告。

Session 5: BGP and Routing

- Kengo Nagahashi、BGP simulation environment on Starbed
  StarBED での BGP シミュレーションの報告。
- Dima Krioukov、Compact routing model
  経路情報の圧縮に関する理論的な研究。
- Kenjiro Cho、Server placement/selection for scale-free networks
  サーバ配置やサーバー選択アルゴリズムを動的な環境で評価するため、スケールフリーネットワークでシミュレーションする手法について。

6.3 第 4 回 CAIDA/WIDE 計測ワークショップ

第 4 回 CAIDA/WIDE 計測ワークショップは、2004 年 8 月 6–7 日にカリフォルニア大学サンディエゴ校のサンディエゴスーパーコンピュティングセンターで実施した。主なテーマは、DNS 計測、トポロジと経路計測、トラフィック計測などである。

プログラム

August 6（Friday）

14:00–14:30 arrivals and introductions

14:30–18:15 DNS measurements and modeling

- Duane Wessels（CAIDA）、Is your caching resolver polluting the Internet?
  SIGCOMM2004 NetTs で発表予定の F-root での DNS クエリ解析の報告。
- Yuji Sekiya（WIDE）、Passive and active DNS measurements
  DNS 応答時間の計測手法の比較についての進捗報告。
- Bill Manning（ISI）、Secure Resolver Objects
  resolver の正しい動作を記述し、検証するプロジェクトについて。
- Suzanne Woolf（ISC）、NS ITR: Improving the Integrity of DNS
  CAIDA/ISC で NSF に提案している DNS 計測プロジェクトの紹介。
- Nevil Brownlee（CAIDA）、NeTraMet:

Recent developments in a production measurement environment
NeTraMet のガーベッジコレクタの改良による性能改善について。

August 7（Saturday）

9:45–12:25 Topology and Routing

- Kenjiro Cho（WIDE）、Dual-stack tools and results
  3 ヵ所からのデュアルスタック計測結果の比較。
- Brad Huffaker（CAIDA）、Active Topology Probing at CAIDA
  skitter/scamper の計測データのデータベースと、これを利用するためのツールについて。
- Dima Krioukov（CAIDA）、Progress in inferring business relationships between ASes
  AS 間のトラフィック相関を解析して、ビジネス的な繋がりを推測する手法について。
- Kengo Nagahashi（WIDE）、BGP simulation on Starbed
  StarBED 上での BGP シミュレーションの進捗報告。

12:25–15:30 Traffic monitoring and other research topics

- Arne Oslebo（UNINETT）、SCAMPI & LOBSTER—European measurement projects
  高速ネットワークモニタリング用カード SCAMPI の開発と、それを用いた計測インフラ LOBSTER についての報告。
- Hiroshi Esaki（WIDE）、IP infrastructure study in Japan
  複数 ISP の協力を得て、バックボーントラフィックを調査する計画について。
- Akira Kato（WIDE）、t-lex: Tokyo Lambda EXchange www.t-lex.net
  T-LEX の紹介と、ラムダネットワークと計測に関する話。
- Jun Takei（WIDE）、AI3: Satellite Internet tested in Asia
  東アジアのインターネット状況と、AI3 プロジェクトでも計測が重要なテーマになっていることを報告。
- David Moore（CAIDA）、How to build a better netflow
  SIGCOMM2004 で発表予定の NetFlow の改善に関する研究。
- Seiichi Yamamoto（WIDE）、s-flow interop test report
  Networld + Interop 2004 tokyo で行った sFlow のインターオペラビリティテストの報告。
- kc claffy（CAIDA）、Internet Measurement Data Catalog
  計測データをカタログ化して、研究利用を推進するプロジェクトの報告。
- Matt Zekauskas（Internet2）、The Abilene Observatory
  ABILENE の計測システムに関する報告。

6.4 今後の予定

次回のワークショップは、2005 年 3 月 11–12 日に USC/ISI で開かれる。今後も引続き、DNS 計測やトポロジ計測を中心とした共同研究を続けるが、新しい分野での研究も検討中である。また、2005 年度には人材交流なども予定している。

第 7 章　First French Asian Workshop on Next Generation Internet

7.1 概要

2004 年 9 月 21 日から 23 日にかけて、フランスの国立研究機関である INRIA[1]において、INRIA、AIT[2]、WIDE の 3 組織による国際共同研究に関するワークショップが開催された。今回のワークショップには総勢約 40 名の研究者が集まり、WIDE からも約 10 名のメンバが参加した。会議では各組織から現在取り組んでいる研究テーマとその現状についてのプレゼンテーションが行われた。INRIA と WIDE は古くからの交流があり、UDLR の標準化で INRIA 側の代表者であった Dr. Walid Dubbous が今回の

1　INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE
2　Asian Institute of Technology (Thailand)

会議の coordinator であった。

以下に、本ワークショップで取り上げられたテーマの概要を示す。

- Mobility
- Ad-hoc networks
- Measurements
- Audio and video support over the Internet
- Multicast
- Dynamic networks and ubiquitous networking
- Internet development
- Distance learning

会議の前半ではまず、各組織における現在の研究テーマおよびその内容の紹介が行われ、後半において今後どのようにそれぞれの組織がコラボレートしていくかの議論がなされた。

発表者および発表内容を次節に示す。

7.2 Workshop Programs

Location
INRIA Sophia Antipolis
2004, route des Lucioles – BP 93
06902 Sophia Antipolis – France

Workshop date: Tuesday, September 21st–Thursday, September 23rd 2004.

7.2.1 Tuesday, Sep. 21st

08:45–09:00 Welcome address by Michel Cosnard, Director, INRIA Sophia Research Unit

09:00–13:00 General Overview Session

- 09:00–09:45 WIDE activities by Jun Murai
- 09:45–10:30 AIT activities by Kanchana Kanchanasut
- 10:30–11:00 Coffee Break
- 11:00–11:45 INRIA and other French activities by Walid Dabbous
- 11:45–13:00 Short Presentations Session

  Presenters:
  - Jean-Marie Bonnin, Armor team
  - Ali Boudani, Armor team
  - Hossam Afifi, Plante team
  - Activities on FLUTE and LDGM, Plante team
  - Antoine Clerget, Udcast
  - Thomas Clausen, Hipercom team
  - Nol Crespi, INT
  - Activities on Intelligent Car, INRIA Visa team
  - Osamu Nakamura, WIDE
  - Keisuke Uehara, WIDE
  - Thierry Ernst, WIDE
  - Yojiro Uo, WIDE
  - Ryuji Wakikawa, WIDE
  - Kenjiro Cho, WIDE
  - Jun Takei, WIDE

13:00–14:30 Lunch Break

14:30–18:00 Technical Presentations Session

Session Chair: Kenjiro Cho

- 14:30–15:00 Large scale Multicast (Kanchana Kanchanasut, AIT)
- 15:00–15:30 Reliable Multicast (Patcharee Basu, AIT)
- 16:00–16:30 Large Scale Virtual Environments (Thierry Parmentelat, INRIA)
- 16:30–17:00 Coffee Break
- 17:00–17:30 Multicast channel announcement (Hitoshi Asaeda, INRIA)
- 17:30–18:00 Internet Traffic measurements at Armor group (Graldine Texier, GET)
- 18:00–18:30 The Metropolis French project on Internet Measurement (Kav Salamatian, LIP6, CNRS)

7.2.2 Wednesday September 22nd

09:00–11:00 Technical Presentations Session

Session Chair: Walid Dabbous

- 09:00–09:30 Towards self-organized networks (Serge Fdida , CNRS)
- 09:30–10:00 Internet Topology Inference (Chadi Barakat, INRIA)
- 10:00–10:30 The OLSR Protocol (Philippe Jacquet, INRIA)
- 10:30–11:00 IPv6 Multicast (Bernard Tuy, Renater)

11:00–11:30 Coffee

11:30–12:45 Discussion Session

Session Chair: Jun Murai
– Future directions and collaboration
– eWorking groups definition

12:45–14:00 Lunch

14:00–14:30 Message from Jun Murai (Left Jun, Osamu and Auto-ID members)

14:30–18:00 Technical Presentations Session
Session Chair: Kanchana Kanchasut

14:30–15:00 Satellite Internet and School On the Internet (Jun Takei, WIDE)

15:00–15:30 WIDE Measurement activities and IPv4/IPv6 results (Kenjiro Cho, WIDE)

15:30–16:00 Mobile Ad-hoc Networking (Ryuji Wakikawa, WIDE)

16:00–16:30 Coffee

16:30–17:00 Network Mobility (Thierry Ernst, WIDE)

17:00–17:30 InternetCAR and Internet Mobility (Keisuke Uehara, WIDE)

17:30–18:00 DVTS—Digital Video over Internet (Ryuji Wakikawa, WIDE)

7.2.3 Thursday, September 23rd

09:00–11:00 Working groups meetings

11:00–11:30 Coffee

11:30–12:30 Wrap-up

12:30–14:00 Lunch

7.3 The working groups

今後ヨーロッパとアジアを結ぶ共同研究プロジェクトを推進していくため、それぞれの組織の研究テーマの紹介が終わった段階で、どの研究テーマを共同研究のテーマにしていくべきかが議論された。その結果、次の研究 WG が構成されることとなった。

- **The Mobility working group** gathers researchers interested in the network mobility domain (nemo and nautilus project), ad-hoc networks, security for ad-hoc networks and Internet car. Collaboration on theses domains already exist between French and Japanese researchers and it is expected that it will be strengthen.
- The collaborations on **Measurements** were discussed within the corresponding working group. The WIDE group already organize a joint workshop with CAIDA. It was proposed to invited French researchers to this workshop.
- **The Multicast and AV streams working group** corresponds to ongoing work in different teams and collaboration is expected to continue/start.
- **The Internet development working group**, it corresponds to ongoing activity in the context of the SOI (School on the Internet) that could involve French professors using a high speed connection between Paris and AI3 through Tokyo. The activity within this domain could be used by the Measurements researchers if helpful.
- **The Ubiquitous networking**: There were no ongoing activities on the French side concerning the Ubiquitous networking topic, but there was an interest to collaborate on this topic.

7.4 Administrative follow-up

会議の終了にあたり今後のプログラムの進め方が議論された。まず次の全体会合については、2005 年の秋の時期に 2004 年同様に INRIA において行うことが合意された。さらに実際に研究を進める WG の interim meeting 的な意味合いで 2005 年の春の時期に AIT が host となり Thailand において会合の場を設けることが提案された。この会議に参加するかは WG ごとに判断するものとし、詳細は WG のメーリングリストにおいて決めることとなった。さらに、WIDE Project の関わるほかの会議（CAIDA workshop、AI3 meeting）などとの joint meeting の可能性についても議論され、その可能性についてそれらの会議日程が決まる時期に改めて検討することとなった。

メーリングリストについては、AIT がセットアップすることとなり現在稼働している。

会議の最後に、研究協力のもう 1 つの道として、交換留学や訪問研究員についても道を開いていくことが確認された[3]。

3　現在すでに WIDE Project からは INRIA および AIT に研究者が送られている。