Cross-Industry Working Team

Internet Service Performance: Data Analysis and Visualization

July 2000

XIWT

The Cross-Industry Working Team (*XIWT)* is a membership organization consisting of a diverse group of communications, computer system, information and service providers who have joined together to develop a common technical vision for the National Information Infrastructure (NII).

XIWT publishes White Papers intended to improve the quality and accelerate the evolution of the NII by establishing a common understanding about the technical issues among those involved with its development and use.

This paper is the second in a series devoted to the critical issue of Internet performance. The principal authors were among a set of experts from *XIWT* member companies and from invited organizations, who together comprise our Internet Performance Working Team (*IPERF*).

Understanding and managing the performance of the Internet has emerged as a critical requirement for the evolution of networked computing, the applications it enables, and the impacts they will have. Future papers will report on empirical results of data collected from the measurement infrastructure described in this paper, as it is extended through a joint industry-government research program, "IPEX: Collaboration for Measurement, Validation and Research Into Anomaly Detection", that will be launched in the 3rd quarter of 2000.

For additional information about XIWT and IPEX, please contact:

Cross-Industry Working Team
Corporation for National Research Initiatives
1895 Preston White Drive
Reston, Virginia, 20191-5434

Tel: 703-620-8990 Fax: 703-620-0913
Email info-xiwt@cnri.reston.va.us
Web: http://www.xiwt.org

Charles N. Brownstein, Executive Director
Email: cbrownst@cnri.reston.va.us

Abstract

The continued viability of the Internet and the resulting economic benefits depend on a level of performance to meet the demands of existing and emerging applications. If performance degrades or reliability becomes uncertain, the user experience will suffer and the application – whether voice over IP, streaming audio or video, or simple web browsing – will, at best, not function as intended, or, at worst, not function at all. Given the emergence of the Internet as critical infrastructure, its contribution to economic output, and the growing number of businesses, organizations, institutions, and consumers who rely on its proper functioning, it is increasingly important for Internet performance to be understood. This understanding is best realized by collecting and analyzing measurement data to determine baseline performance, detect anomalies as they occur, and establish trends that can be used for planning.

This paper is a companion to an earlier Cross-Industry Working Team white paper "Customer View of Internet Service Performance: Measurement, Methodology, and Metrics". *The methodology described in that paper has been implemented in order to collect Internet performance data across a number of sites. Those data are used to establish the feasibility of the approach by illustrating how they can be aggregated and visualized to gain insight into Internet performance. Demonstrating the applicability of the approach to establishing baselines, detecting anomalies, and identifying trends is specifically addressed.*

For network managers, operators, systems engineers, technical managers, and all those involved in designing and implementing network planning processes, this paper provides a feasible approach for collecting and understanding relevant performance data. It demonstrates that the proposed measurement framework is useful and can be used as a starting point for implementations. It provides insight into data aggregation approaches and suggests techniques that may be adapted to the specific needs of network operators. Furthermore, for those involved in policy definition and strategic planning, this paper helps underscore the complexities of measuring and interpreting performance and offers one approach to managing such complexity.

1.0 Introduction

As more and more corporations, small businesses, government and private organizations, educational institutions, consumers, and individuals rely on the Internet to conduct business, engage in commercial activities, and simply communicate, the pressure applied by these users on this network to meet their performance and reliability demands also increases. This pressure is exacerbated by the deployment of high-speed access technologies, such as cable modems and ADSL. When these demands are coupled with observations that the Internet is emerging as critical infrastructure (XIWT 1998) and its contribution to economic output is significant (U.S. DOC 1999), it is clear that the performance of the Internet is a critical issue. The Internet's continued viability is closely tied to its ability to support the applications that use it. If performance degrades or reliability becomes questionable, the user experience will suffer. When the degradation is severe enough – although the specific ramifications will vary with the individual user and application – the end result will be the same: the application will not be used, and the benefit will be lost.

Internet performance can best be understood by collecting and analyzing measurement data. In XIWT (1998), a measurement architecture, measurement methodology, and common set of metrics were proposed for assessing, monitoring, negotiating, and testing compliance for service quality (between an Internet service provider [ISP] and its customers). The members of the Cross Industry Working Team Internet Performance Working Team (XIWT/IPWT) have undertaken a measurement initiative, based on the proposed architecture, to collect a comprehensive set of data. In this paper, those data are used to establish the feasibility of the approach and to demonstrate its applicability to establishing baselines, detecting anomalies, and identifying trends. To this end, this paper presents and analyzes data that were collected using the proposed methodology. Such analyses can provide insight into the current performance of the Internet as well as realistic performance values that could be used in service level agreements (SLAs). Insights into tradeoffs between methodology alternatives and different types of statistics can also be assessed.

This paper is targeted at the full spectrum of individuals for whom the performance of the Internet (or private intranets) is an issue. For network managers and operators, the paper provides a feasible approach for collecting and understanding relevant performance data. For systems engineers, technical managers, and those involved in designing and implementing network planning processes, it demonstrates that the proposed measurement framework is useful and can be used as a starting point for implementations; furthermore, it provides insight into data aggregation approaches and suggests techniques that may be adapted to the specific needs of network operators. Finally, for those involved in defining policies and strategic planning, this paper helps underscore the complexities of measuring and interpreting performance and offers one approach to managing such complexity.

Internet performance is a broad topic; this paper focuses on a few primary issues:

- establishing baselines,
- detecting anomalies, and
- identifying trends.

Establishing baselines is important for determining whether the infrastructure can support new applications and services and, if so, to what degree. It is integral to answering the question: *Will the application work?* The risks associated with investing in the development and deployment of Internet-based services must be managed by understanding, a priori, whether the performance and reliability of the application will meet end user expectations. Without understanding the *past* performance of the Internet (or private, IP-based networks), these risks cannot be managed.

Detecting anomalies addresses whether the existing infrastructure is currently meeting the performance and reliability requirements of the application on a day-to-day basis. Anomaly detection is closely associated with monitoring real-time and near-real-time performance of the network and responding to incidents such as outages that can affect performance. Essentially it answers the question: *Is the application working satisfactorily?* Ensuring that end user expectations for performance and reliability are continually met is critical to the successful offering of IP-based services.

Identifying trends helps predict the future performance of the infrastructure. It is closely associated with network engineering and planning, and ensuring that there is sufficient capacity to support the applications and end users. Understanding trends is essential to answering the question: *Will the application continue to work?* Advance planning ensures that sufficient capacity will continue to exist and end user expectations will continue to be met.

This paper is organized as follows. The remainder of this section provides a high-level overview of the proposed measurement architecture and reviews related efforts to characterize Internet performance. Section 2 introduces the proposed methodology and data analysis approach, describes the methodology and experimental setup, formalizes the relevant statistics, and explains how the data can be aggregated to produce relevant trends and baselines. Section 3 applies the methodology to the collected data to produce baseline results. Application of the approach to detecting anomalies is addressed in section 4. Section 5 uses the collected data to produce trends and to address planning. Finally, section 6 provides a summary and an outline for future work.

Note that this paper does not attempt to establish baselines or trends but to provide insight on how they might be obtained.

1.1 Measurement Methodology

This subsection provides a brief overview of the proposed XIWT measurement methodology. For additional details and clarifications, refer to XIWT (1998).

As illustrated in figure 1, the measurement architecture uses a “black box” approach wherein the metrics and measurements are defined in terms of externally visible properties.

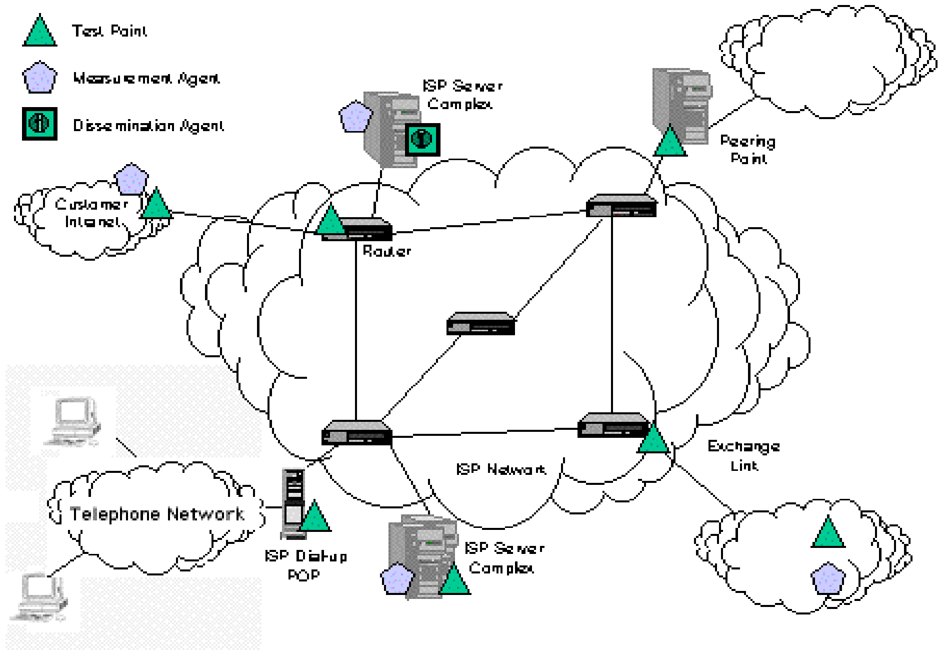

FIGURE 1. Measurement Architecture

The architecture is defined in terms of the following logical components:

Test point. Test points are hosts that either collect performance data or have been configured to respond to measurement queries. Providers may decide to install separate hosts as test points enabling measurements to reduce the load on critical network elements such as routers. The architecture assumes that appropriately configured test points have been identified at various places in the network and on customer sites.

Measurement agent. A measurement agent is software that runs on a host and actually initiates the various measurements. Measurement agents communicate with the test points to conduct the measurements or collect data present at the test points. Measurement agents may also examine log files and simulate service usage to measure performance.

Dissemination agent. A dissemination agent provides the results of the measurements that have been collected to interested parties and/or to an archive site. Based on need, the results from the various agents may be combined/correlated before dissemination or may be left “raw.”

Performance and reliability metrics quantify the end user's perceptions of service performance and reliability. Metrics of interest include:

Roundtrip delay. Roundtrip delay is defined as the interval between the time a measurement agent application sends a packet to a test point and the time it receives an acknowledgment that the packet was received by the test point. Roundtrip delay includes any queueing delays at the endpoints or the intermediate nodes, but does not include any Domain Name System lookup times by the measurement application.

Packet loss. Packet loss is defined as the fraction of packets sent from a measurement agent to a test point for which the measurement agent does not receive an acknowledgment from the test point. This includes packets that are not received by the test point as well as acknowledgments that are lost before returning to the measurement agent. Acknowledgments that do not arrive within a predefined roundtrip delay at the measurement agent are also considered lost.

Reachability. A test point is considered reachable from a measurement agent if that agent can send packets to the test point and, within a short, predefined time interval, receive acknowledgment from the test point that the packet was received. For example, if each measurement sample consists of multiple pings, the test point is considered reachable from the measurement agent if the latter receives at least one acknowledgment from the test point.

Availability. The network between a measurement agent and a test point is considered available at a given time t, if during a specified time interval Δ around t, the measured packet loss rate and the relevant statistics on the roundtrip delay (such as median or maximum) are below predefined thresholds. Network service availability is defined as the fraction of time the network is available from a specified group (one or more) of measurement agents to a specified group of test points.

Other metrics related to performance and reliability have been defined in XIWT (1998), but those are not pertinent to this document.

1.2 Related Work

Many public, private, and commercial efforts aim at understanding the performance of the Internet through measurement collection activities. Several of these are described below.

Visual Networks (1999) and Keynote (2000) are two companies that offer products and services that can be used to assess the performance of applications and the quality of service offered in dial-up networks. Although highly relevant to end users, ISPs, and web hosting services, these commercial efforts do not offer the detailed data analysis and statistics generation capabilities that are needed for customers to fully understand end-to-end Internet performance.

The PingER project led by the Stanford Linear Accelerator Center (SLAC) is an effort to monitor and understand the parts of the Internet used in high energy nuclear and particle physics (HENP) research (SLAC 2000). The project began in 1995 and is funded by the U.S. Department of Energy; as of January 2000, Ping ER monitoring involved 71 countries on six continents. The monitoring site (typically a particle physics research lab such as SLAC) sends pings[1] to a remote site (typically a university or institution collaborating on an experiment at the monitoring site), and the packet loss and roundtrip time reported by ping is gathered from each of the 20 monitoring sites and written to a database at Fermilab. The analysis of the data centers on trends in packet loss and changes in roundtrip time. These metrics allow the network researchers to identify troubled links where network congestion is causing packets to be dropped at routers along the way or routes to change, possibly taking a much longer path. This information helps network engineers and managers make decisions about routing and resource allocation. Beyond troubleshooting, the project is used to gauge the feasibility of the computing model for future HENP experiments that will generate petabytes or even exabytes (10^{15} to 10^{18} bytes) of data, much of which will need to be distributed to the experiment's collaborators at universities and institutions around the world.

The Cooperative Association for Internet Data Analysis (CAIDA) [6] is a “collaborative undertaking among organizations in the commercial, government and research sectors aimed at promoting greater cooperation in the engineering and maintenance of a robust, scalable global Internet infrastructure” (CAIDA 2000). As part of this initiative, CAIDA has developed a series of measurement and analysis tools that can be used to better understand Internet traffic. Most of these tools can be downloaded from the CAIDA website <http://www.caida.org/tools/>, and some of the CAIDA measurement data can be made available to researchers.

2.0 Methodology and Data Analysis

XIWT/IPWT member companies are participating in an experiment to collect data that could provide insight into the performance of the Internet between member sites. The purpose of this exercise is multifold:

- to demonstrate the validity of the proposed measurement methodology using an actual implementation;
- to gain experience with such implementations;
- to collect data that is indicative of Internet performance and that could be applied to SLAs, application performance analysis, monitoring, and trend analysis;
- to gain insight into aggregating techniques for data to provide meaningful results; and
- to provide the foundation for additional work, such as using passive (as opposed to active) measurements and gathering end-to-end application performance data (as opposed to network performance data).

The first phase of this effort was to baseline Internet performance between the member companies. The methodology, the analysis of the data, and the results from this effort (and Internet performance baseline) are described in the following subsections.

[1]A ping is a tool, included with most operating systems, that uses control packets to measure roundtrip delay between two endpoints (Postel 1981).

2.1 Methodology

The XIWT/IPWT uses a modified version of the PingER software,[2] described above, to measure roundtrip delay, packet loss, and availability between pairs of hosts. About a dozen measurement hosts are deployed at IPWT member sites; these hosts are both test points and measurement agents. Every 30 minutes,[3] each host pings every other host as well as itself in order to detect anomalies in its own operation. A set of 11 pings of 100 bytes each is sent first; the first ping is discarded to eliminate possible effects such as priming of caches. This is followed by a set of 10 pings of 1,000 bytes each. In addition to the two sets of pings, a traceroute command is sent to each remote host.[4] Once a day, an archive host (the dissemination agent) retrieves ping and traceroute data from each of the measurement hosts and stores the data in a database.

Each ping packet received by a source host contains a value for the roundtrip delay between that host and the destination host. If a ping is not returned to the source host within a specified timeout period, then it is assumed the packet has been “lost” either on the outbound or return path. If none of the pings in a set is received by the source, the destination host is considered unreachable from the source at that time. The percentage of packets lost and the roundtrip delay are important indicators of how well certain applications will perform. By retaining data on all the pings, the archive site is able to calculate a variety of statistics—for example, mean, median, minimum, maximum, quartile—and can perform this calculation over any aggregated set of data; for example, aggregation over all hosts or over a particular period of time. See XIWT (1998), section 4 for additional background on metrics. The traceroute data are currently used to help diagnose the cause of anomalous results in the ping data (see section 4 below).

This method of measurement was chosen because of its simplicity and the availability of the ping tool on all machines. It does, however, have certain limitations. Ping uses the Internet Control Message Protocol (ICMP), and does not necessarily have the same performance as TCP, UDP, or other IP protocols. For example, ICMP packets can be given lower priority on some routers, or they can be blocked by firewalls. Despite these limitations, ping data still provide good insight into the performance of a network. Furthermore, the discussions about statistics, data aggregation, and visualization in this document remain valid, regardless of the underlying measurement techniques.

The connectivity between the hosts as of October 1999 is illustrated in figure 2. Each oval represents an independent network on the Internet.[5] Traversing an independent network may require multiple hops. Measurement nodes are depicted by name and are connected to the independent networks by high-speed links (T1 rates or higher).

To obtain information on the complete set of raw data, refer to: <http://www.xiwt.org/IPERF.html>.

---

[2] The principle modification is that the result of each ping is collected, as opposed to collecting statistics on the 10 pings.

[3] The specific parameter values given here are those used when the measurements were taken.

[4] A traceroute provides information about the nodes a packet encounters along the path from the source host to the destination host, as well as the times the packet reaches those nodes (Stevens 1994).

[5] An independent network is one or more autonomous systems managed by the same network provider.

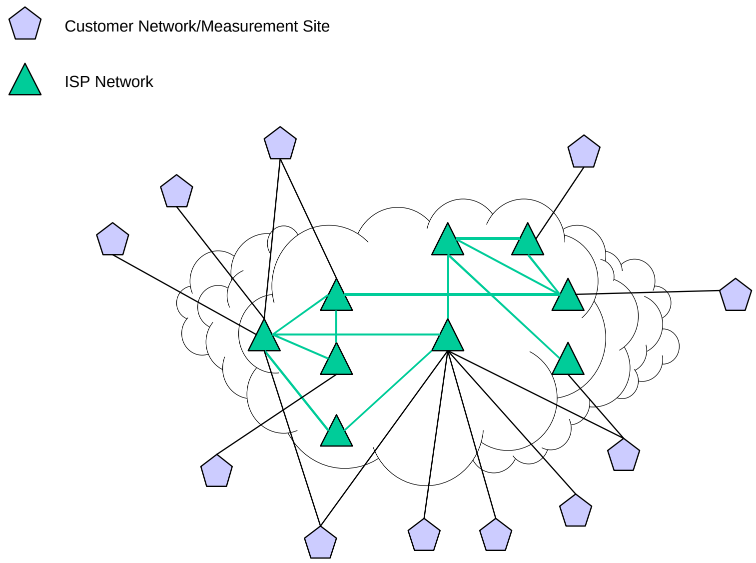

FIGURE 2. Network Topology

2.2 Statistics and Data Aggregation

The amount of data collected at the archive site can be enormous, especially if there are numerous source-destination pairs and if the data has been collected over a reasonably long period of time. The challenge is to visualize the data so they can be interpreted intelligently. Furthermore, only certain subsets of the data may be of interest (for example, delay between two hosts during the busiest hour or the packet loss to a particular destination). In this subsection, the various ways in which the data may be aggregated and plotted are summarized. Additional details and aids to the visualization of different types of aggregation are provided in appendix B. Note that there are many approaches and statistical methods for aggregating data; the approach used here was chosen for its simplicity and relevance; further, as will be seen in sections 3 and 4, it can result in very useful baseline and trend plots.

As described above, a set of ping samples are collected every half-hour between each source and destination pair. This set of samples is simply a set of delay measurements, which, for example, could look like:

$$\{78\text{ ms}, 85\text{ ms}, 72\text{ ms}, \infty, 64\text{ ms}, 53\text{ ms}, 81\text{ ms}, 93\text{ ms}, 101\text{ ms}, 67\text{ ms}\}$$

where ∞ (infinite delay) is simply used to represent the fact that a ping response was not received within the timeout interval (the default timeout for ping is typically 20 seconds).

Over the course of a day, 48 of these sample sets are collected, as illustrated in the following table.

TABLE 1. Sample Data Collected over a Single Day (Between a Single Source and Destination Pair)

	Time	Sample Sets
1	12:00 a.m.	{78 ms, 85 ms, 72 ms, ∞, 64 ms, 53 ms, 81 ms, 93 ms, 101 ms, 67 ms}
2	12:30 a.m.	{42 ms, 77 ms, 68 ms, ∞, ∞, ∞, 95 ms, ∞, 43 ms, 41 ms}
3	1:00 a.m.	{...}
4	1:30 a.m.	{...}
	.	
	.	
	.	
47	11:00 p.m.	{...}
48	11:30 p.m.	{...}

Of course, these data are collected between each source-destination pair; consequently, over a long period of time, the volume of data will be extremely large. To be useful, this large volume must be reduced to a form that reveals statistically meaningful trends. Two complementary techniques can be employed to achieve this: statistics generation and data aggregation.

A variety of statistics can be generated from a set of samples. With respect to delay, common statistics include the following:

The median. The median of a set of samples is simply the sample for which there are an equal number of samples with a lesser value and an equal number with a greater value. For the 12:00 a.m. set given above, the median is 78 ms:

53 64 67 72 | 78 | 81 85 93 101

Appendix A explains how the median is computed when there is an even number of samples in the set.

The mean. The mean of a set of samples is the same as the average value, which, for the 12:00 a.m. set, can be calculated to be:

$$\frac{53 + 64 + 67 + 72 + 78 + 85 + 93 + 101}{9} = 77.1 \text{ ms}$$

The maximum. The maximum sample is straightforward to obtain: 101 ms.

The minimum. The minimum sample is also straightforward to obtain: 53 ms.

Note that lost packets are not included in the above delay statistics. Other statistics, such as percentiles and inter-quartile ranges, can also be calculated from a set of delay values; appendix A provides additional details.

For many applications, such as voice over IP, loss can be as important to service quality as delay. Furthermore, the throughput of a TCP connection, which is used for Web (http) and file transfers (ftp), is very dependent on the loss characteristics at the IP layer. Loss is simply the ratio of unsuccessful pings to the total number of pings. For the 12:00 a.m. set of delay samples, the loss metric can be calculated as:

$$L = \frac{1}{10} = 10\ \%$$

To compute statistics on the loss metrics (such the median, minimum, maximum, and so forth), a set of loss metric data points is needed; that is, aggregation of data points to compute loss statistics is required. One obvious and useful approach is to aggregate the data collected between a single source and multiple destinations.[6] For example, at 9:00 a.m. on a particular day between a source and 10 destinations, the following data could have been collected:

Destination No.	Loss
1	0%
2	0%
3	10%
4	30%
5	0%
6	100%
7	40%
8	0%
9	10%
10	0%

Statistics for these loss metrics can be readily calculated as follows (note, however, that destination 6 would be excluded since 100% loss by definition means that the destination is unreachable).

The median. For the set given above, the median is 0%

0% 0% 0% 0% | 0% | 10% 10% 30% 40%

The mean.

$$\frac{0 + 0 + 0 + 0 + 0 + 10 + 10 + 30 + 40}{9} = 10\ \%$$

[6]Analogously, another approach is to aggregate data from multiple sources to a single destination.

The maximum. The maximum sample is straightforward to obtain: 40%.

The minimum. The minimum sample is also straightforward to obtain: 0%.

Baseline plots for delay and loss and other metrics (such as availability) can be obtained by plotting the statistical values as a function of time interval. The aggregation of data over which the statistics are generated and the time intervals to be plotted determine the type of baseline produced. In the delay examples above, no aggregation of data occurred: the statistics were generated directly from a set of samples (yielding a one-to-one correspondence between a measurement interval and a set of statistics). Aggregating sample sets before computing statistics is necessary in producing baselines that provide insight into the performance of the network and that can be used for predicting performance, detecting anomalies, or determining trends. Some common baselines are summarized in table 2; Appendix B illustrates how the corresponding plots are produced.

TABLE 2. Typical Baselines

Baseline type	Aggregation interval	Comments
Time of day Plot of delay (or loss) as a function of time	**None** Statistics are calculated directly from the individual sample sets with no aggregation of data. (see figure 26)	The time intervals plotted dictate the nature of the plot. For example, every sample during a 24-hour period or every 10:30 a.m. sample for the past 30 days would produce plots with useful, yet different information on performance. Such a plot may not provide a meaningful baseline since performance at instants in time can fluctuate widely.
	Day of week (for example, Tuesdays) For each time interval, statistics are calculated by aggregating all sample sets collected on Tuesdays at that time (see figure 27)	If the measurement interval was 30 minutes, this baseline would also be called a 30-minute baseline since the x-axis would be demarcated into 30-minute intervals (12:00 a.m., 12:30 a.m., etc.). If the measurement interval was 30 minutes, a 60-minute baseline could also be created by aggregating two measurement intervals (the x-axis would then be demarcated at 60-minute intervals (12:00 a.m., 1:00 a.m., etc.). The number of previous Mondays to use in the aggregation can vary depending on data availability and perceived relevance of historical data. Four to six is deemed reasonable. As the name implies, time of day baselines are most useful for understanding how performance can vary with the time of day.
Daily Plot of delay (or loss) as a function of the day of the week	**24-hour** For each day of the week, all the sample sets are aggregated to calculate the statistics for that day (see figure 28)	The number of days' (previous weeks) worth of data to aggregate again depends on data availability and perceived relevance. A 24-hour daily baseline produces statistics on a large amount of data. For example, if 30-minute measurement intervals with 10 samples per interval were used and if four weeks of data were collected, then each delay statistic is based on $10 \times 48 \times 4 = 1920$ samples.

TABLE 2. Typical Baselines (Continued)

Baseline type	Aggregation interval	Comments
	Busy period For each day of the week, all sample sets in a specific interval of the day (for example, between 9:00 a.m. and 10:00 a.m.) are aggregated to calculate the statistics for that day (see figure 29)	In some cases, network performance over a smaller interval than a 24-hour day is of interest. The interval of interest is often the busy period when traffic loads tend to peak. Generating daily baselines for this period is also possible. If the busy period were 10:00 a.m. to 2:00 p.m., this baseline would be known as a 10:00 a.m.-2:00 p.m. daily baseline. It is conceivable that the busy period of interest is a single measurement interval (in which case the baseline would be called a 10:00 a.m. daily baseline, for example). Daily baselines are very useful for understanding how performance varies with day of week.
Weekday or weekly Plot of delay (or loss) as a function of the week of the year.	**24-hour** For each week, all sample sets for that week are aggregated; for a weekday baseline, only the Monday to Friday data are used; for the weekly, all seven days of data are used (see figure 30)	This baseline is referred to as a 24-hour weekday (weekly) baseline. It differs from the time of day or daily baselines in that the x-axis can cover a large span of time and thus show how performance has changed over time. This type of baseline is useful for trend analysis.
	Busy period For each week, all sample sets in a specific interval (for example, between 10:00 a.m. and 2:00 p.m.) are aggregated (see figure 31)	As with daily baselines, it is possible that only a subset of time, such as the busy period is of interest. If the busy period were 10:00 a.m. to 2:00 p.m., this would be referred to as a 10:00 a.m.-2:00 p.m. weekday (weekly) baseline.
Rolling Plot of delay (or loss) as a function of the trending interval	**24-hour** For each trending interval, all sample sets for that interval are aggregated	A trending interval is a set of days, typically an integral number of weeks (for example, four or eight). The x-axis is usually labeled with the day of the year and represents the last day in the trending interval. These baselines are known as 24-hour 28-day (or 56-day) rolling baselines and are used for determining long-term trends in performance.
	Busy period For each trending interval, all sample sets in that interval and in the busy period (for example, between 9:00 a.m. and 10:00 a.m.) are aggregated (see figure 32)	If long-term trends for the busy period are of interest, then aggregation intervals can be limited. This type of baseline would be referred to as a 9:00 a.m.-10:00 a.m. 28-day (or 56-day) rolling baseline. If desired, specific days such as holidays or weekends could be omitted from the aggregation interval.

Though not explicitly addressed in table 2, the descriptions assumed a single source-destination pair. The baselines described are equally applicable to multiple source-destination pairs (appendix B provides an example).

2.3 Data Visualization

The visualization of performance metrics and the baselines associated with these metrics are important for gaining insight into the information that is contained in what can be a very large database. Metrics can be broadly divided into two categories: single-dimensional and two-dimensional. Single-dimensional metrics are the simplest and are a function of a single performance metric (such as delay, loss, or jitter). Two-dimensional metrics, such as availability, are a function of two metrics (such as delay and loss). The dimension of the metric dictates the approaches to visualization.

2.3.1 Single-Dimensional Metrics

First, consider a baseline for single-dimensional metrics such as delay or loss. To maximize the information contained in the baseline, standard statistics (Chambers et al. 1983) can be plotted, as shown in figure 3. In addition to the median, statistics relating to the quartile ranges are also plotted; including the inter-quartile range, the upper and lower adjacency values, and the outliers. Details on these statistics can be found in appendix A. Although plotting statistics such as the 10th and 90th percentiles is feasible, these statistics are more sensitive to the outliers. The quartiles are less sensitive and, when plotted with the adjacency values and outliers, provide very good graphical insight into the nature of the distribution.

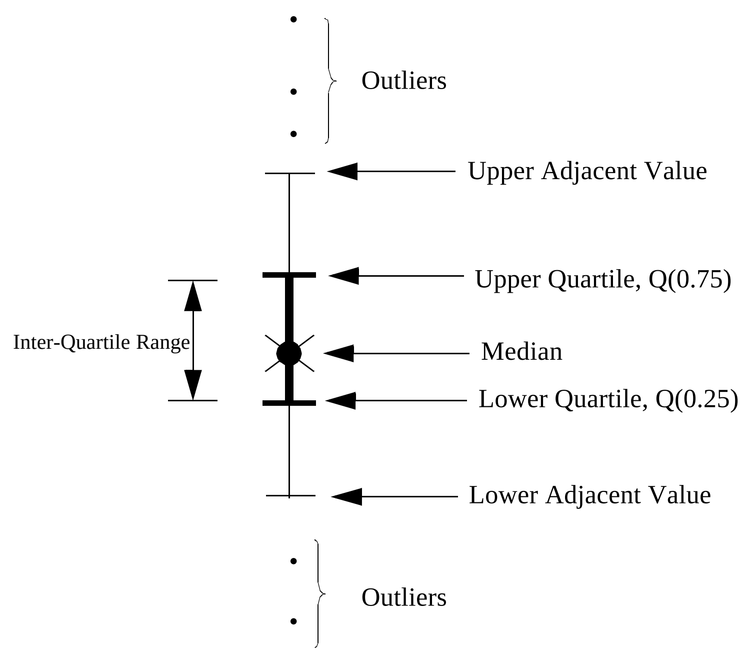

FIGURE 3. Data Point for Baseline Plots

Figure 4 shows an example baseline using the data point defined in figure 3.

Thus far, the discussion has focused on delay and packet loss. A baseline using the data point defined in figure3 is also applicable to a two-dimensional metric (such as availability), providing that the latter is reduced through a form of aggregation to a single-dimensional metric. An explanation of how this can be accomplished for availability follows.

2.3.2 Two-Dimensional Metrics

Visualization of two-dimensional metrics can be accomplished in two ways:

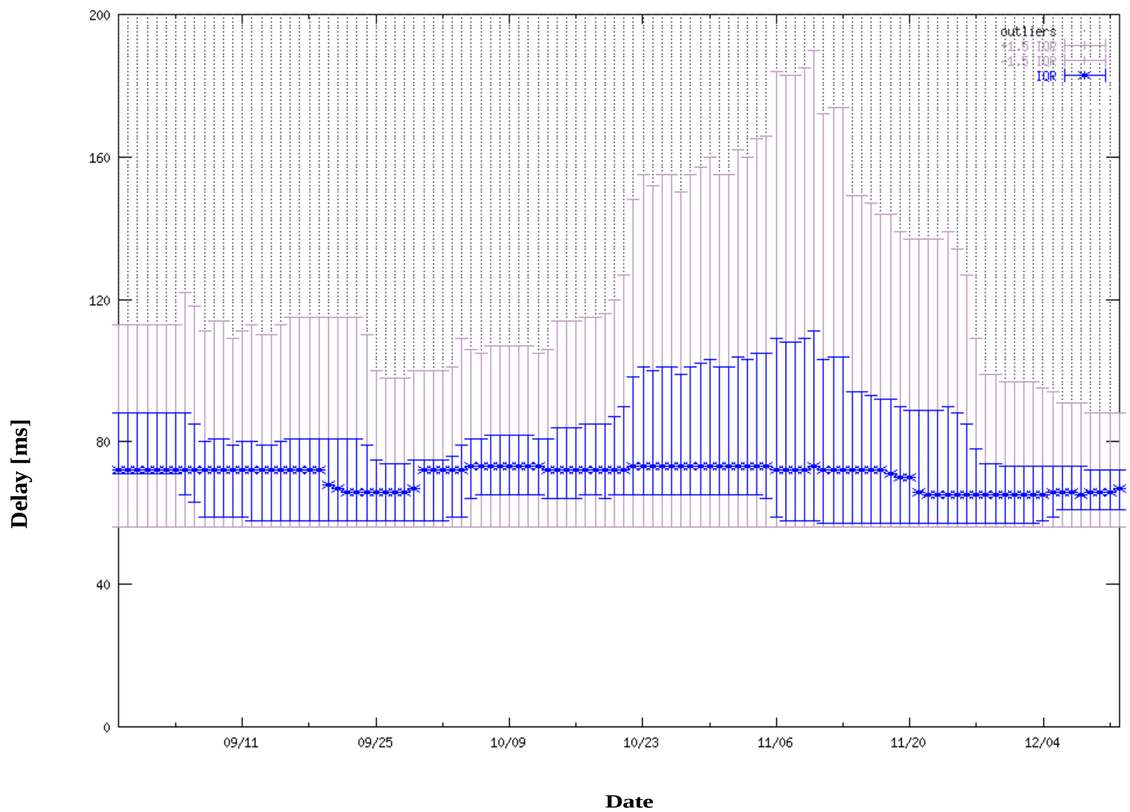

FIGURE 4. An Example Baseline (28-Day Rolling Baseline, Single Source-Destination Pair)

1. direct plots of the two metrics (one metric represented on the x-axis, the other on the y-axis); or
2. plots of some function of the two metrics (which reduces visualization to a single dimension).

Availability is used as an example. Availability is the fraction of time when the delay and loss rate of pings sent to a destination are within selected thresholds. For illustration purposes, three arbitrary thresholds (defining three levels of availability) are set:

Good	Delay <100 ms *and* loss <5%
Unavailable	Delay >400 ms *or* loss >20%
Poor	Otherwise

The labels *good*, *poor,* and *unavailable*, as well as the thresholds that define them, are only for illustration. In reality, the number of thresholds and threshold values would reflect the requirements of specific applications.

Figure 5 shows an example of the direct approach to plotting a two-dimensional metric. It plots ping data from one source to one destination during a period of 10 weeks. Each point on the graph is a {*loss*, *delay*} pair, where *delay* is the roundtrip time of a ping and *loss* is the average loss in the set to which the delay measurement belongs. For example, consider the following set of delay measurements:

{78 ms, 85 ms, 72 ms, ∞, 64 ms, 53 ms, 81 ms, 93 ms, 101 ms, 67 ms}

This set has an average loss of 10 percent and will produce the following 10 {loss, delay} pairs:

{10%, 78 ms}
{10%, 85 ms}
{10%, 72 ms}
{10%, ∞}
{10%, 64 ms}
{10%, 53 ms}
{10%, 81 ms}
{10%, 93 ms}
{10%, 101 ms}
{10%, 67 ms}

Pairs such as these are used to plot availability. Pairs with an infinite delay are usually not plotted.

As shown in figure 5, average loss ranges from 0 percent to 80 percent (average loss is always a multiple of 10 percent because there are exactly 10 pings in a set). Finite delays range from approximately 60 ms to 900 ms.

Figure 5 also plots the thresholds for good, poor, and unavailable, as defined above. Using these thresholds, availability was good 87 percent of the time, and poor 9 percent of the time. In the remaining 4 percent of the time, the destination was unavailable because of excessive loss or excessive delay.

Note that the good, poor, and unavailable metrics are single-dimensional metrics and can be visualized as described in subsection 2.3.1. This concept is expanded upon below.

Like delay and loss, the single-dimensional availability metrics can be aggregated in different ways to obtain the baselines. Since the two-dimensional metrics must be aggregated over some interval to be reduced to a single-dimensional metric, this reduction interval can dictate the type of baseline produced.[7] One obvious approach is to use a 24-hour period as the reduction interval; this is shown in figure 6, which plots the good, poor, and unavailable metrics from Host 21 to Host 2 during the month of September. Note that this is analogous to a time of day baseline with no aggregation interval (see table 2 and figure 26) where every sample is plotted (except that the x-axis time scale is on the order of days, rather than hours).

[7]Recall that the loss metric also required a reduction interval to convert a set of delay metrics into a single loss metric; however, it was implicitly determined that the logical reduction interval for loss was a single measurement interval. For availability, a single measurement interval may have insufficient data to yield meaningful good, poor, or unavailable metrics.

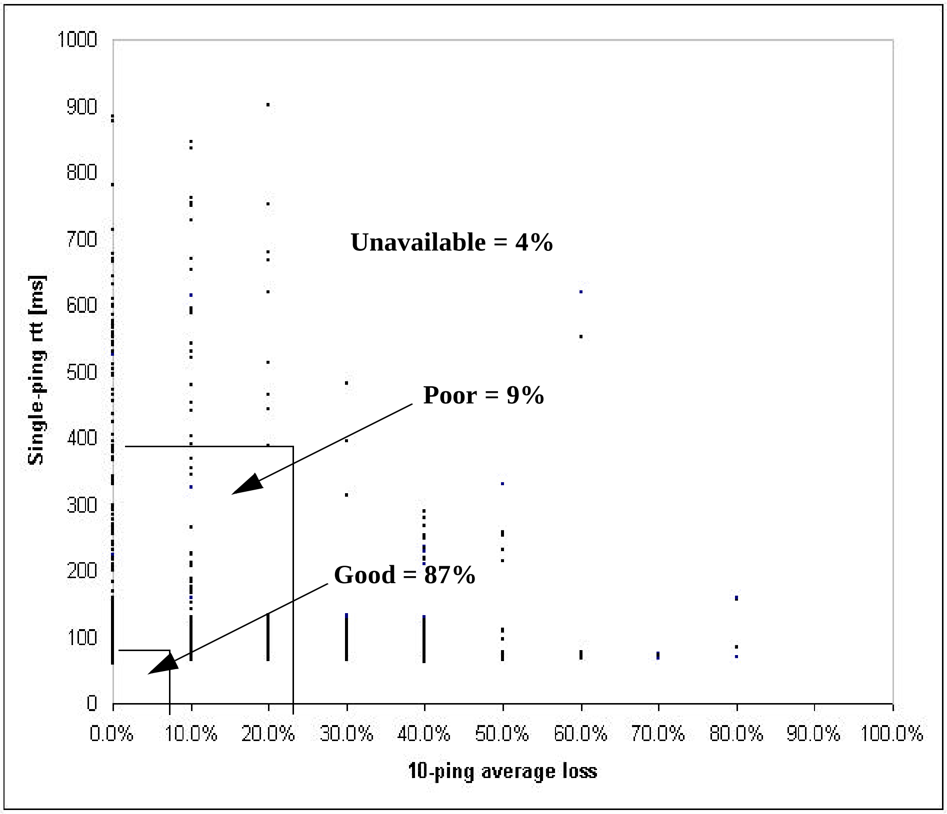

FIGURE 5. Availability From Host C to Host D (Over a 10-Week Period)

Once a reduction interval has been chosen and the two-dimensional availability metric is reduced to a single dimension, all the typical baselines (table 2) and the single-dimensional visualization approach (figure 3) are applicable. The only caveat, of course, is that the baseline aggregation interval (column 2 in table 2) must be greater than the reduction interval used to reduce from two dimensions to a single dimension.

Availability can also be visualized by aggregating over all or some meaningful subset of the host pairs, or, as was proposed for the loss metric, between a single source and multiple destinations. (Conversely, availability of a given host to the rest of the network can be computed by combining ping data from all sources to the destination host in question.) It might also be useful to compare the relative availability of the hosts. For example, figure 7 plots the availability of 18 hosts in a network over a 24-hour period. Hosts being pinged are labeled 1 to 20 and include self-ping data.[8] Host 16 had 100 percent good availability over the 24-hour period, which means that all ping delays and 10-ping packet loss averages were within the good thresholds. Host 6 was not connected to the network that day; therefore, is shown as unavailable for the whole period.

[8]Twelve measurement agents/test points and eight additional test points are being pinged. Hosts 17 and 18 are not plotted since their data were not available.

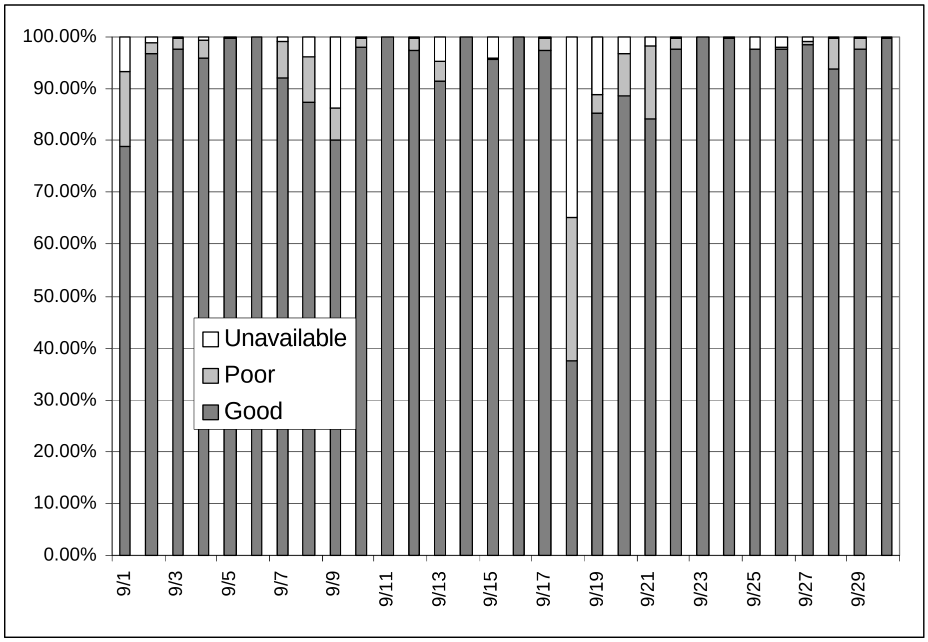

FIGURE 6. **Daily Availability From Host 21 to Host 2 Over 10 weeks**

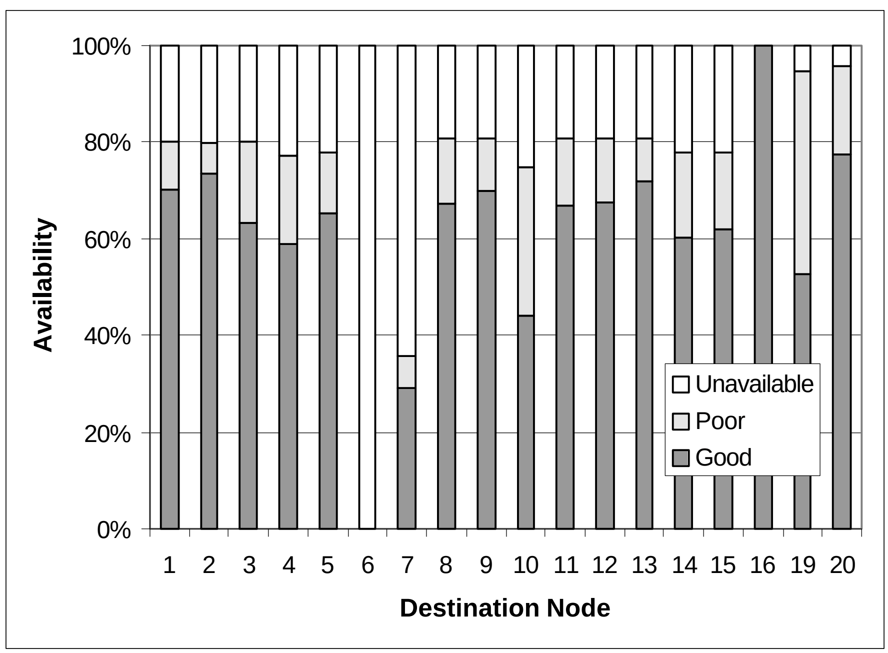

FIGURE 7. **Each Host's Availability to All Nodes**

The approaches described above to visualizing availability can be modified or expanded in many ways. Different sets of delay and loss thresholds can be used to evaluate the network for different types of services. For large networks with hundreds of nodes, it may be difficult to show all nodes on a single graph; thus, it might be necessary to sort them and show only the low performers or those that do not meet certain minimum availability criteria. To distinguish between busy periods and off-hours, it might be useful to produce separate availability graphs for different periods of the day. Also, although defined around delay and loss, availability is not limited to these metrics: other performance metrics, such as jitter, could be included in the definition, provided that measurements and sensible thresholds were available.

3.0 Baselines

This section presents examples of the baselines introduced in subsection 2.2. The plots shown and interpreted in this section are based on actual Internet performance, using data collected as described in subsection 2.1. As well as providing insight into Internet performance (or at least performance of a very small portion of it), these plots demonstrate the usefulness and applicability of the common baselines. Although the examples focus on delay and loss, these baselines are applicable also to the aggregated (single-dimension) availability metrics.

3.1 30-Minute Baseline

The first example of a baseline is the 30-minute baseline, where delay and loss are aggregated over a specific weekday (in this case, Thursdays) and plotted as a function of time for a single source-destination pair. These 30-minute baselines are based on data collected over four consecutive Thursdays.

For the first source-destination pair, as seen in figure 8, there is very little variation over the course of the day, with a slight rise in the mean, quartiles, and adjacency values during the middle of the day. This is consistent with the notion that traffic loads – and hence delays – are higher during the middle of the day than at other times. This baseline also shows the lower bound on the roundtrip delay between the pair: there are no outliers below the lower adjacency value; thus, the minimum delay is approximately 80 ms. Note that this minimum is fairly static and hardly varies with time of day. It is conjectured that, in this case, the minimum roundtrip delay represents the (fixed) transmission and processing delays between the pair. While this baseline shows that delays can be on the order of hundreds of milliseconds, the bulk of the traffic experiences roundtrip delays of between 80 and 120 ms.

It is interesting to compare this baseline with that between a different source-destination pair, as shown in figure 9. In figure 9, there is a much more pronounced time-of-day fluctuation with a peak in the early afternoon. The rising median implies that a significant amount of traffic is being affected. Note also that the minimum rises, indicating that all traffic (save for a couple of outliers) is experiencing queueing delays during the peak period.

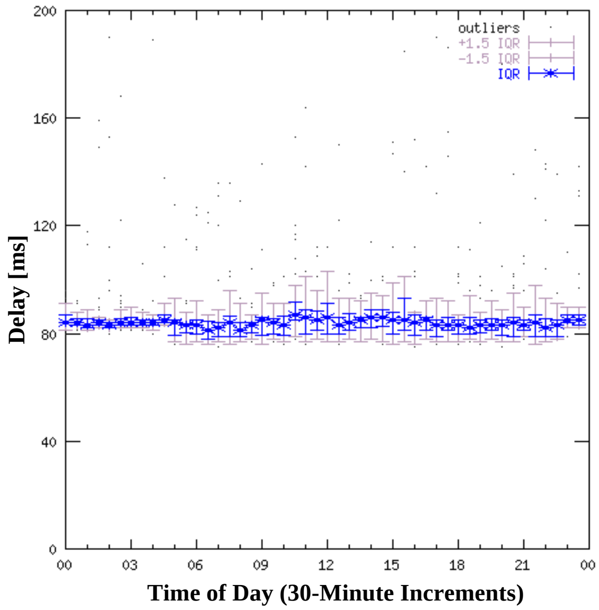

FIGURE 8. Example of a 30-Minute Baseline: Delay.
Single source-destination pair: C,D. Data aggregated over Thursdays between July 1and July 28, 1999.

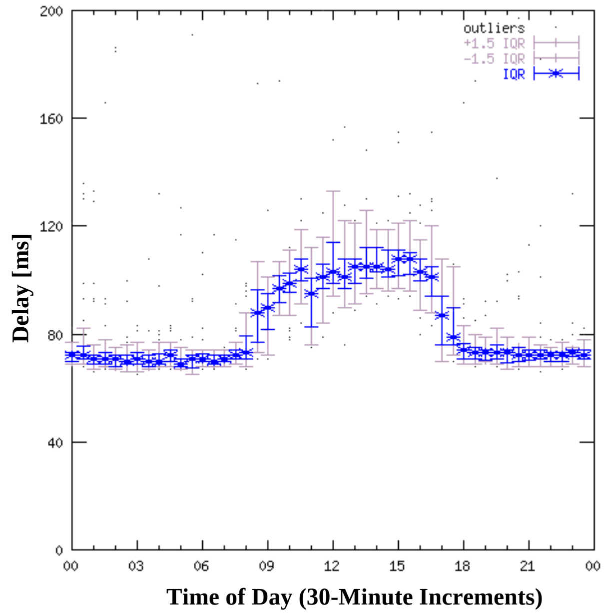

FIGURE 9. Example of a 30-Minute Baseline: Delay.
Single source-destination pair: E,C. Data aggregated over Thursdays between July 1, 1999 and July 28, 1999.

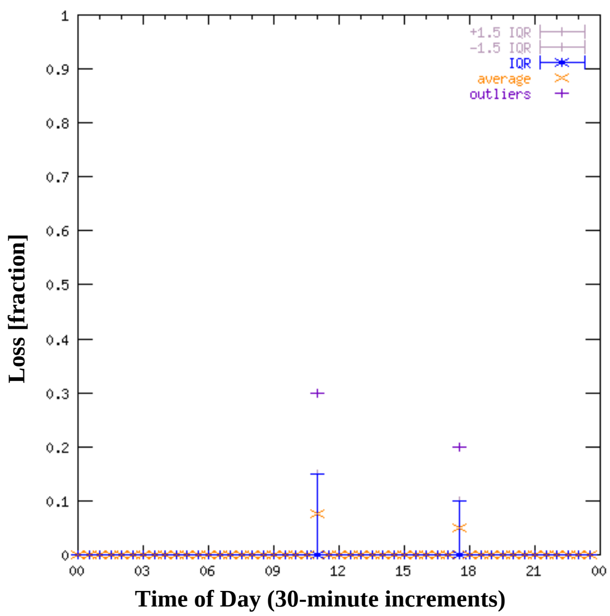

FIGURE 10. Example of a 30-Minute Baseline: Loss.
Single source-destination pair: C, D. Data aggregated over Thursdays between July 1, 1999 and July 28, 1999.

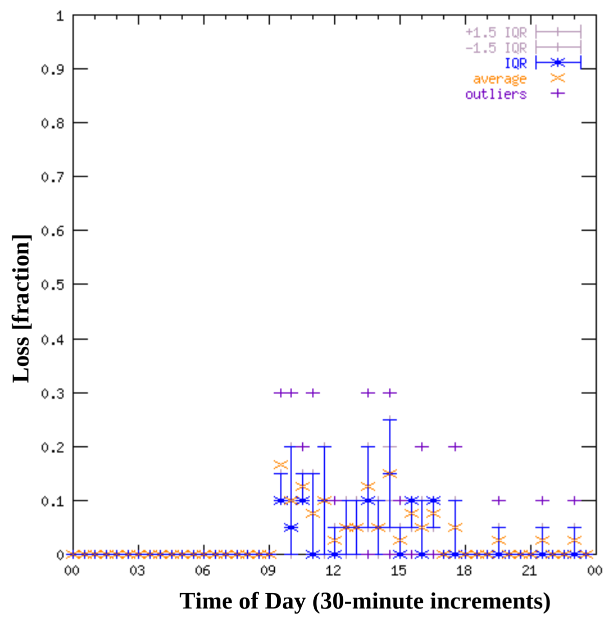

FIGURE 11.Example of a 30-Minute Baseline: Loss.
Single source-destination pair: E, C. Data aggregated over Thursdays between July 1, 1999 and July 28, 1999

A baseline such as that shown in figure 9 may be useful for determining how application performance can fluctuate. For example, consider an application that performs database queries over a network. If the timers in the application are set based on a roundtrip delay allocation of 100 ms, between approximately 9:00 a.m. and 5:00 p.m., 50 percent of the queries will “fail” due to timeouts.[9]

Figures 10 and 11 show the corresponding loss baselines. It is important to note that the loss statistics are based on an order of magnitude of less data than the delay statistics (since the loss metric must be computed from the results of a set of ping data, as explained in subsection 2.2.2). Figure 10 shows some fluctuation in loss as a function of time of day, but since the loss rates are so low, it is difficult to conclude that there is a correlation between loss and time of day. In figure 11, however, the loss rates are much higher, and it is clear that loss rates *do* vary with time of day.

Figures 9 and 10 also illustrate how the information provided by the median and the average differ. Generally, the median and the quartiles are useful in understanding overall distribution, but the average provides better insight into the characteristics of the loss. To understand why, consider the following set of loss metrics:

{0%, 0%, 0%, 0%, 0%, 0%, 0%, 0%, 0%, 90%}

For this set, the median is clearly 0 percent and the average is 9 percent. Since the loss metric is bounded between 0 percent and 100 percent (as compared to delay, which is bounded only by the—relatively large—ping timeout value), most of the loss information is in the “tail” of the distribution. Since the median tends not to reflect the outliers that form the tail of the distribution, the median is not a good statistic for gaining insight into loss performance. And, since most of the loss information is in the tail of the distribution, it is reflected better in the average.

The relative distance between the median and the average, however, is useful for understanding the distribution. When the median and average are far apart, it implies that there are a small number of large outliers (which influence the average, but not the median). When they are close together, the distribution can be thought of as balanced; that is, the distribution is symmetrical (about the median).

Loss baselines similar to those illustrated here could be used to assess the potential performance of an application between a source-destination pair. For example, if an application such as Voice over IP requires loss rates to remain below, say, 5 percent, figure 11 indicates that performance would suffer greatly between mid-morning and late afternoon.[10]

3.2 Daily Baselines

Daily baselines are plots of delay and loss as a function of day of week. In these examples, a 24-hour aggregation interval is used and data have been collected over a four- week period. Note that the increased density of the outliers, compared to the 30-minute

[9] A savvy application designer would use the baseline to set the timeout values.
[10] The fact that loss rates are based on roundtrips would need to be factored into any real analysis.

baselines, is a simple artifact of aggregating a much larger set of data (there are naturally more outliers, although the percentage of outliers might be the same).

As shown in figure 12 (which uses the same raw data as for the 30-minute baselines of subsection 3.1), a daily baseline can exhibit very little day-to-day variation in the median or inter-quartile range; this is indicative of a network that has sufficient capacity to absorb fluctuations in traffic. In figure 13, the daily baseline between a different source and destination pair indicates that there is weekday dependency on the inter-quartile range: on Sundays, the inter-quartile range is very small but is decidedly larger on weekdays (Monday through Friday). Since the median is not substantially affected, this implies that, on weekdays, the delay distribution has a heavier tail.

The loss plots show analogous characteristics. Between the first source-destination pair (figure 14) there is very low loss and very little day-to-day variation. Between the second source-destination pair (figure 15) the loss variation is much more pronounced, with a definite day-to-day variation: there is a major increase between Sundays and Mondays, with a decreasing trend on the other days of the week. Further data analysis would be needed to explain the nature of this baseline.

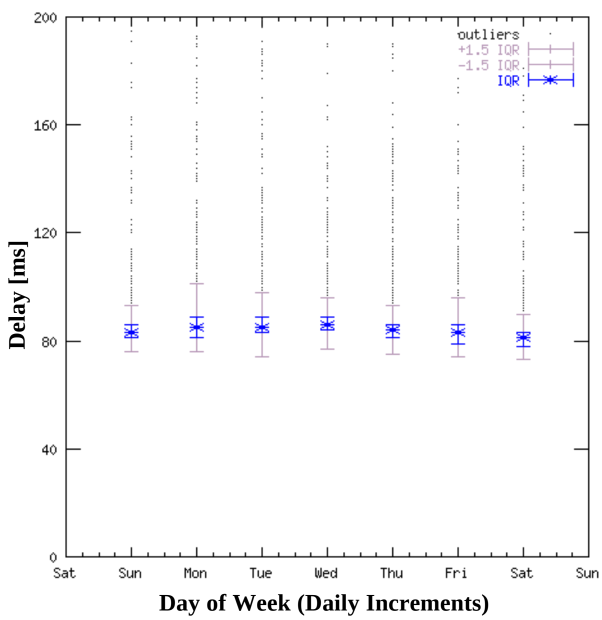

FIGURE 12. Example of a Daily Baseline: Delay.
Single source-destination pair: C, D. Data aggregated over Thursdays between July 1 and July 28, 1999.

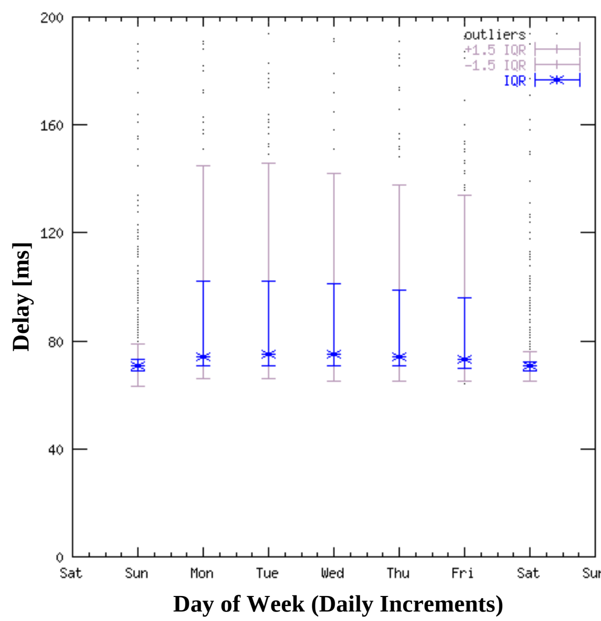

FIGURE 13. Example of a Daily Baseline: Delay.
Single source-destination pair: E, C. Data aggregated over Thursdays between July 1 and July 28, 1999.

3.3 Weekly Baselines

As explained in subsection 2.2, a weekly baseline is an aggregation over a five- or seven-day week, with each day aggregated over 24 hours or some smaller interval of interest. Each data point on the plot represents one week. Figures 16 and 17 show examples of weekly baselines for delay and loss, respectively. In these examples, the local network at

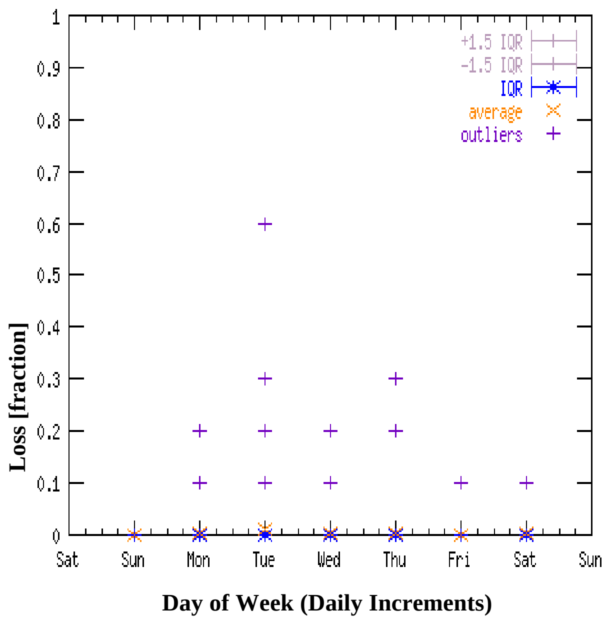

FIGURE 14. Example of Daily Baseline: Loss.
Single source-destination pair: C, D. Data aggregated over Thursdays between July 1 and July 28, 1999.

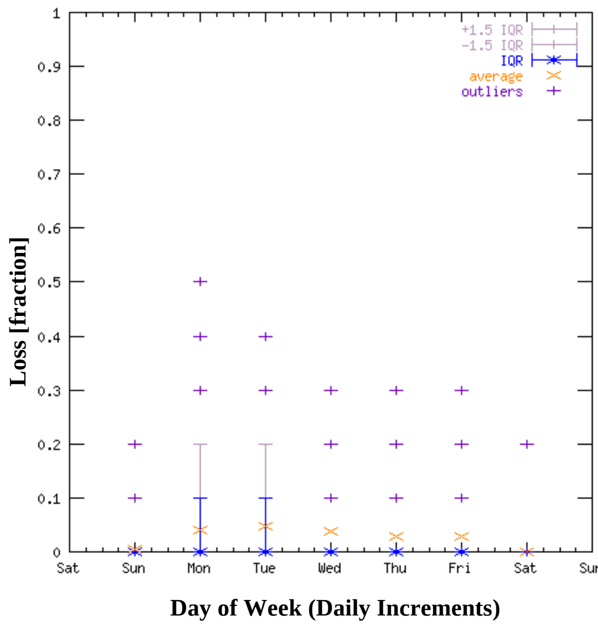

FIGURE 15. Example of a Daily Baseline: Loss.
Single source-destination pair: E, C. Data aggregated over Thursdays between July 1 and July 28, 1999.

the destination was known to have a congested link, and access to this network had known periods of high usage due to the nature of an application that was being hosted. The weekly delay baseline shows how the inter-quartile range varies; it also shows that the median, although confined to a narrow range, can have signification fluctuation within that range. The loss baseline also shows a significant amount of fluctuation.

3.4 28-Day Rolling Baseline

Twenty-Eight-day rolling baselines are useful for gaining insight into historical performance and, consequently, into whether current performance is within norms or if an anomaly has occurred. The use of this rolling baseline to determine trends or anomalies is further illustrated in sections 4 and 5. Each date on the x-axis represents the last day in the 28-day trending interval, and the statistics are based on the aggregation of data over those 28 days. The delay and loss plots shown in figures 18 and 19 are 24-hour 28-day rolling baselines.

Given the aggregation interval, there is a significant amount of data; hence, the density of the outliers is quite high. The somewhat periodic nature of the inter-quartile and upper adjacency value fluctuations is again an artifact of the traffic patterns for the application being hosted at the destination network. From a network management perspective, this type of baseline could be used to conclude that performance is cyclic, and—although the

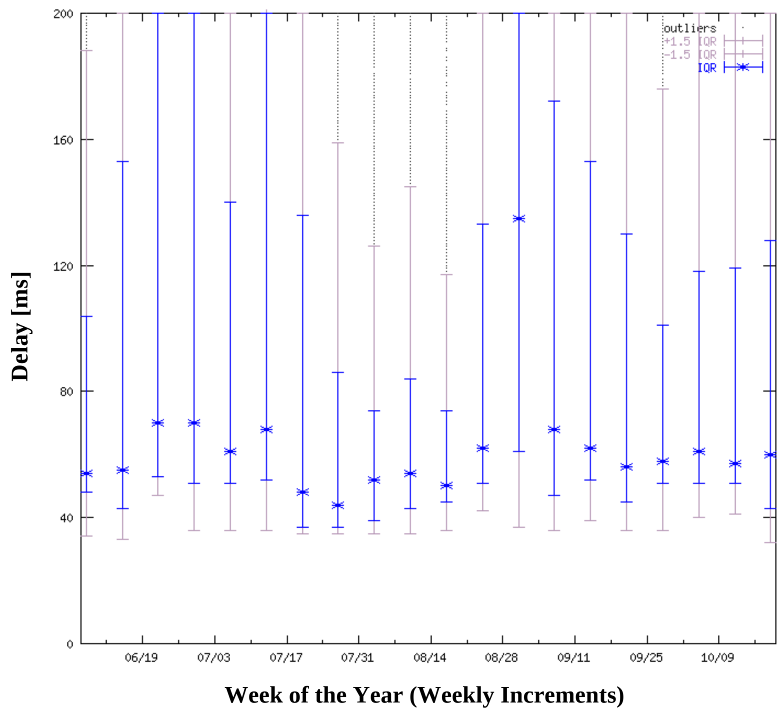

FIGURE 16. Example of a Weekly Baseline: Delay. *Single source-destination pair: A, F.*

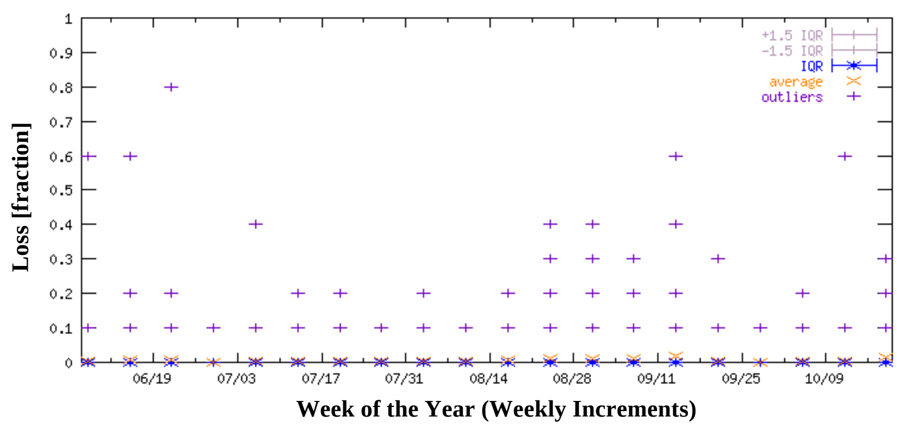

FIGURE 17. Example of a Weekly Baseline: Loss. *Single source-destination pair: A, F.*

median is relatively stable—it is normal for significant portions of traffic to experience delays that are an order of magnitude larger than the median.

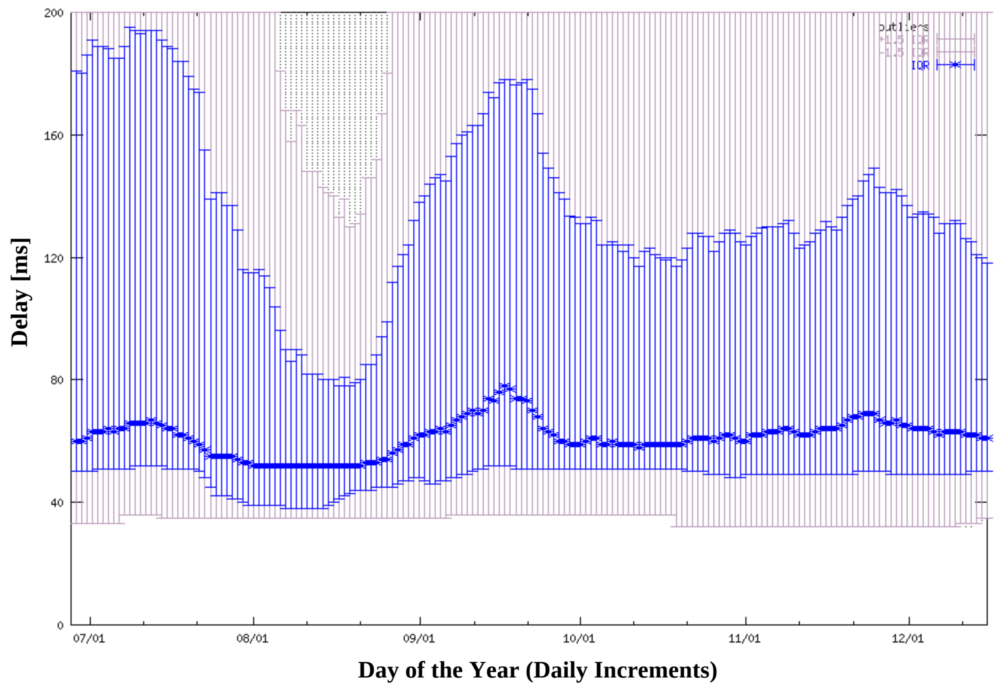

FIGURE 18. Example of a 28-Day Rolling Baseline: Delay. *Single source-destination pair: A, F.*

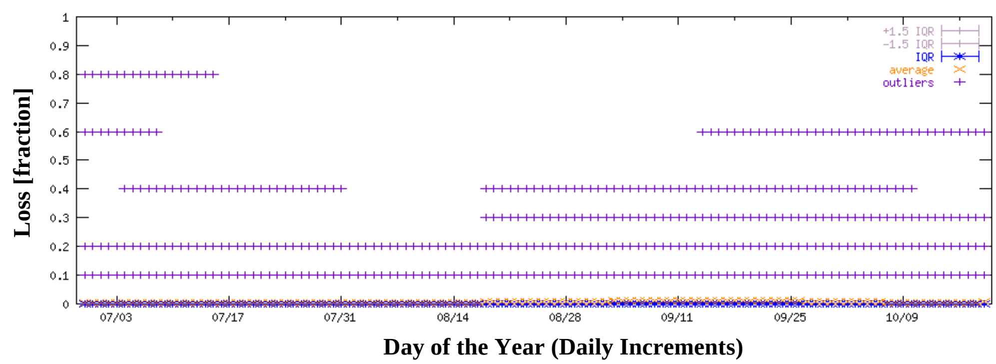

FIGURE 19. Example of a 28-Day Rolling Baseline: Loss. *Single source-destination pair: A, F.*

4.0 Anomalies

One of the benefits of developing baselines and collecting measurement data at regular intervals is the ability to detect anomalies. Roughly speaking, if a measurement or short series of measurements drastically differs from the baseline, there is a good chance that some “event” has occurred that has affected the performance for the duration of that event. Common examples of events that would cause an anomaly include network outages, configuration changes, and focused overloads. Events that regularly occur would be automatically reflected in the baseline and thus would not be considered an anomaly (for example, a route flap or routine maintenance).

Ideally, anomaly detection would be automated: measurements would be monitored, and, if statistical values exceeded certain deviations from the baseline, alarms would be generated (or at least an email would be sent to the network administrator). It is outside the scope of this paper to recommend criteria for detecting anomalies, however, examples can easily be provided. If, for example, the median of a sample set exceeds the inter-quartile range of the baseline, then an anomalous event can be considered to have occurred.

Figures 20 and 21 demonstrate anomaly detection. Figure 20 is a 28-day rolling baseline: each point summarizes the performance over the preceding 28 days. So for this example, on August 8, it is normal for the median delay between 9:00 a.m. and 12:00 p.m. to be 70 ms and for the inter-quartile range to be only a few milliseconds. In figure 21, the time of day data are plotted (with a 12:00 a.m.- 9:00 a.m. aggregation only), showing the performance for each day. On August 9, the median delay for the 12:00 a.m.- 9:00 a.m.interval was measured to be over 100 ms. This is clearly outside the inter-quartile range for the baseline (as calculated for the preceding 28 days) and therefore suggests that an event causing an anomaly has occurred. At this point, a network administrator could investigate, possibly by using other tools to confirm that an event has occurred and establishing whether it was transient or will be more lasting. In the example shown, the anomaly lasted nine days, after which performance returned to normal.

In practice, data associated with anomalies would not be incorporated into the rolling baseline (as was done in figure 20).

5.0 Trends

It is well known that the Internet is undergoing exponential growth in both size and traffic. To manage this growth, network planners use various tools to track utilizations and determine when links must be upgraded or additional equipment deployed. Similar exercises can be undertaken to manage network performance and to ensure that applications that require certain performance assurances will continue to work. In figure 20, for example, there would not have been any performance issue with an application that required the median roundtrip delay to be less than 90 ms: there is no evidence that delays are increasing.

Contrast this with figure 22. Although a regression analysis would be helpful in formally establishing trends, it appears that the median, inter-quartile, and adjacency values are increasing with time. If thresholds for performance are known, these statistics could be extrapolated to determine when in the future the application would be expected to fail (and plans could be implemented to avoid such a situation).

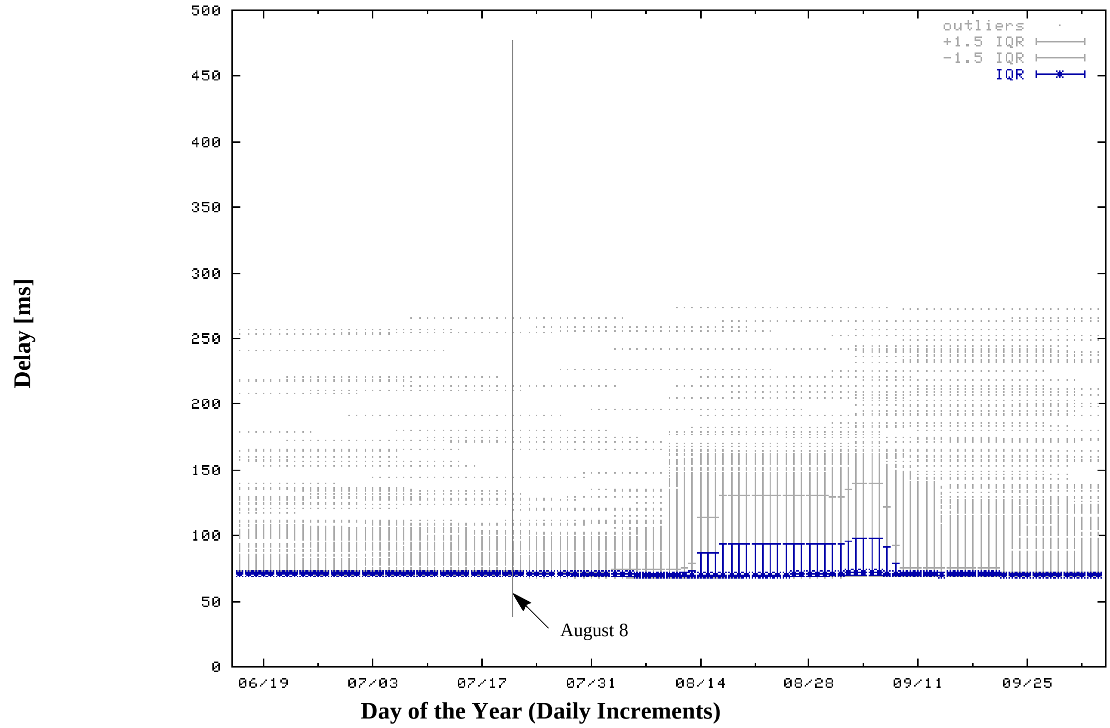

FIGURE 20. Anomaly Detection: 12:00 a.m. - 9:00 a.m. 28-Day Rolling Baseline (Delay). *Single source-destination pair:* **D, A.**

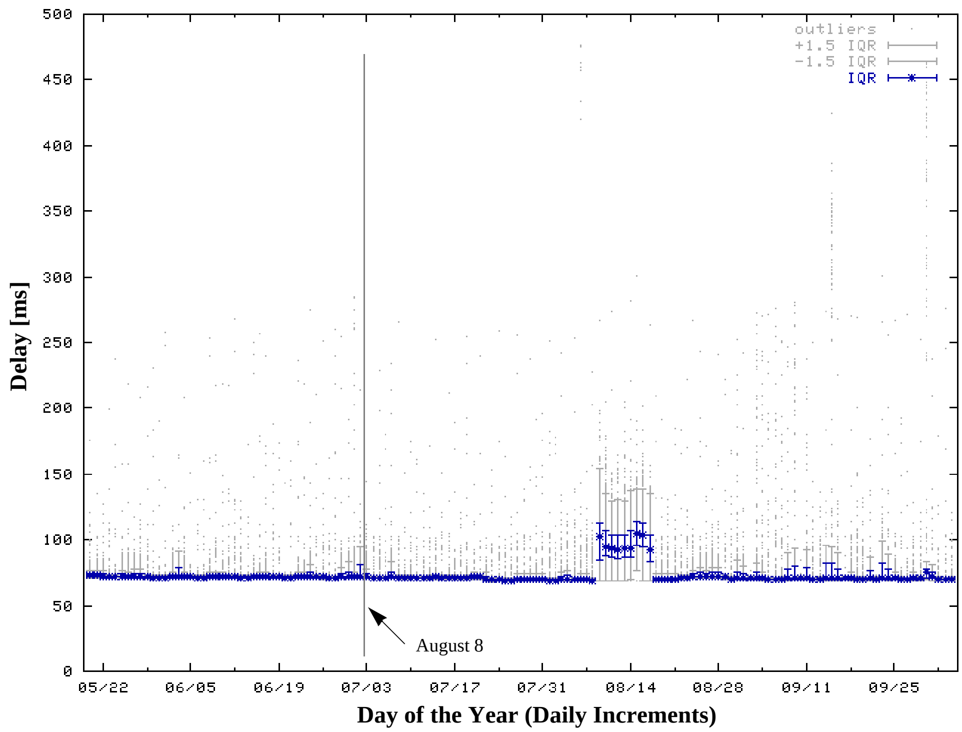

FIGURE 21. Anomaly Detection: Daily Performance (Delay). *Single source-destination pair: D, A.*

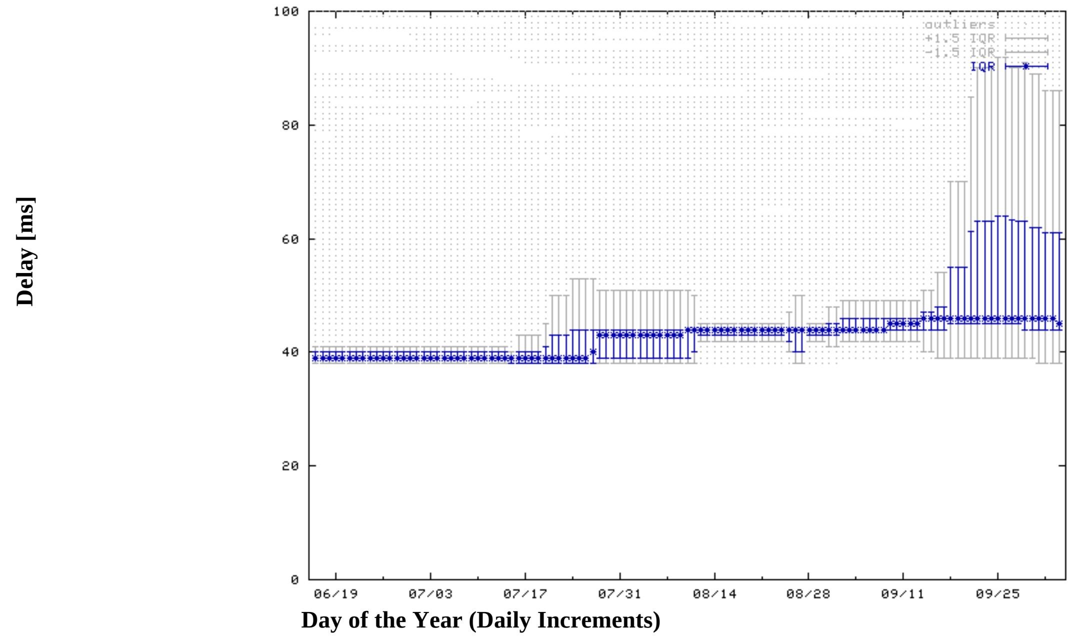

FIGURE 22. Delay Trend (28-Day Rolling Baseline). *Single source-destination pair: A, G.*

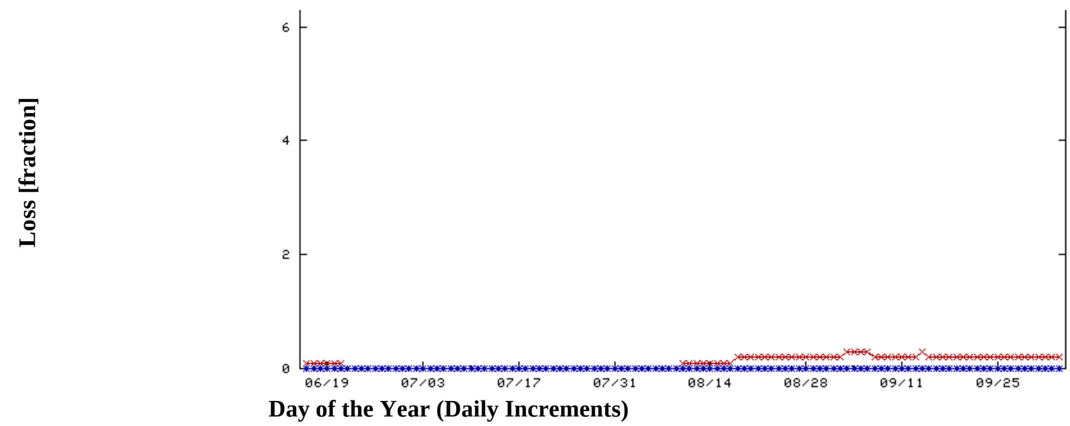

FIGURE 23. Loss Trend (28-Day Rolling Baseline). *Single source-destination pair: A, G.*

6.0 Summary and Conclusions

The continued viability of the Internet and resulting economic benefits depend on its performance meeting the demands of the existing and emerging applications that will use it. If performance degrades or reliability becomes uncertain, the user experience will suffer, and the application – whether it be voice over IP, streaming audio or video, or simple web browsing – will, at best, not function as intended, or, at worst, not function at all. Given the economic impetus to ensure that applications will work, performance must be understood.

Internet performance can best be understood by collecting and analyzing measurement data. In this paper, a measurement initiative has been described and the data collected from that initiative have been used to establish the viability of XIWT/IPWT's previously proposed methodology. Specifically, this paper has:

- Defined relevant statistics and aggregation methods for three major performance metrics: roundtrip delay, loss, and short-term availability. Short-term availability, which combines the important delay and loss metrics, introduces the notion of a "grade of service," which is useful for customers, planners, and others who are involved in the design and management of IP-based networks. The statistics and aggregation methods proposed are extensible in the sense that other metrics that may be relevant to the performance of specific applications, such as jitter or throughput, can be analyzed using the same approach.
- Demonstrated how the statistics could be used to baseline performance, highlighting the various type of baselines and showing, with actual measurement data, how these baselines offer differing insights into network performance and can be used to assess how well an application might perform if deployed.
- Identified how the statistics and baselines can be used for anomaly detection and trending. As illustrated by example, baselines can be used to detect an anomaly as it occurs, allowing network managers to react quickly and ensure that application performance will be restored promptly. Baselines can also be used to detect trends in performance, which can be used by network planners so that sufficient resources are deployed and the network is properly engineered to ensure that the application will continue to work.

The examples used in this paper have been drawn from a small subset of the data collected by the XIWT/IPWT. The raw data can be used for further analysis of aggregation techniques and to understand the performance of the Internet between the sites participating in the data collection.

The work focused primarily on establishing the feasibility of a measurement methodology and demonstrating its usefulness for understanding Internet performance. Recommendations for extending and broadening this effort include initiatives to answer the following questions:

- Given a network topology and applications of interest, where should agents be deployed?
- How should other metrics, such as jitter and throughput, be measured and analyzed?
- Can a passive measurement approach, that provides the same insights into performance be implemented?
- How should application-level performance measurements for applications such as voice over IP, streaming video, and web browsing be collected and analyzed?
- How should this methodology be refined for specific performance issues; for example, should finer grained measurements be taken or should correlations with traceroute data be undertaken?
- Are there commercial measurements that adapt to this measurement methodology?
- What are the approaches to data reduction, minimizing the storage requirements yet retaining a sufficient level of performance information?

7.0 References

Chambers, J. M., W.S. Cleveland, B. Kleiner, and P.A. Tukey. 1983. Graphical Methods for Data Analysis. Boston: Duxbury Press.

Cooperative Association for Internet Data Analysis (CAIDA). 2000. <http://www.caida.org/home/background.xml>.

Cross-Industry Working Team, (XIWT). 1998. Customer View of Internet Service Performance: Measurement, Methodology, and Metrics. <http://www.xiwt.org>.

Keynote.2000. <http://www.keynote.com>

Postel, J. 1981. Internet Control Message Protocol. STD 5, RFC 792. University of Southern California/Information Sciences Institute.

Stanford Linear Accelerator Center (SLAC). 1999. <http://www-iepm.slac.stanford.edu/pinger/>.

Stevens, W. Richard. 1994. TCP/IP Illustrated, Volume 1: The Protocols. Reading, MA: Addison-Wesley.

U.S Department of Commerce, (U.S. DOC). 1999. The Emerging Digital Economy. <http://www.ecommerce.gov/ede/report.html>.

Visual Networks. 1999. <http://www.visualnetworks.com/corp/corpslaform.htm>.

Appendix A: Formal Specification of Statistics

Let the set of samples collected during some measurement interval be denoted by:

$$\{d_1, d_2, \ldots, d_N\}$$

where d_k is the k^{th} data sample (round trip delay for the ping query and response) and N is the number of samples collected.

Without loss of generality, assume that the samples within a measurement set are ordered such that $d_i \le d_j$, $i < j$. Delay statistics, for instance, can include the following:

Median: The median for the set of samples is defined as:

$$\begin{cases} d_{\lceil N/2 \rceil} & N \text{ is odd} \\ \dfrac{d_{N/2} + d_{N/2+1}}{2} & N \text{ is even} \end{cases}$$

Mean: The mean for the set of samples is defined as:

$$\frac{1}{N}\sum_{n=1}^{N} d_n$$

Maximum: The maximum sample with a sample set is simply: d_N .

Minimum: The minimum sample with a sample set is also simply defined: d_1 .

Inter-quartile range: Let $Q(0.25)$ and $Q(0.75)$ denote the 25 percentile and 75 percentiles of the data, respectively. The inter-quartile range is thus:

$$\mathrm{QR} = Q(0.75) - Q(0.25)$$

Upper adjacent value: The upper adjacent value is defined to be the largest observation (sample) that is less than or equal to:

$$Q(0.75) + 1.5 \times \mathrm{IQR}$$

Lower adjacent value: The lower adjacent value is defined to be the smallest observation (sample) that is greater than or equal to:

$$Q(0.25) - 1.5 \times \mathrm{IQR}$$

Outliers: To provide insight into the nature of the distribution, samples that fall outside the range of the adjacencies are plotted as individual data points.

Note that it is always implicitly assumed that statistics operations are performed on a set of data samples from which the infinite delay measurements (no ping response) are removed. Further details on calculating these and related statistics can be found in Chambers et al. (1983).

Appendix B: Data Aggregation and Baselines

In table 2 a set of common baselines were described. In this appendix, how those baselines were produced is detailed by visualizing the aggregation. The complete set of data collected between all source-destination pairs over some period of time (perhaps many months) can be illustrated as shown in figure 24.

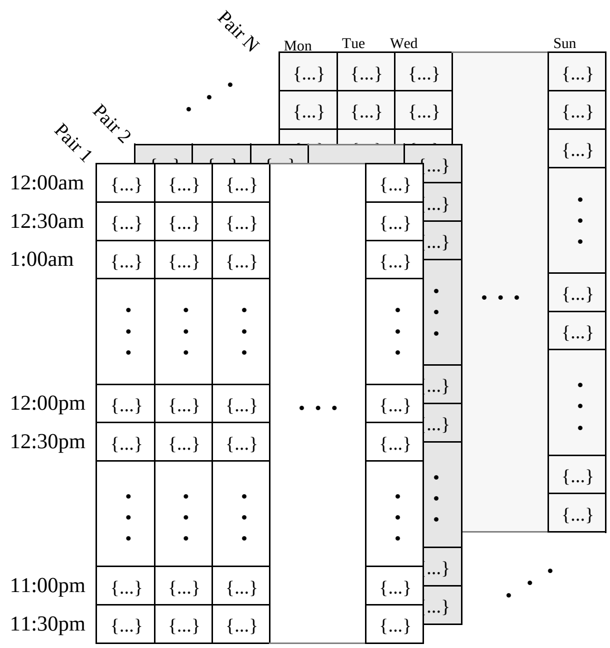

FIGURE 24. Conceptualization of a Large Set of Samples (All Source-Destination Pairs)

As in table 1, {...} is simply shorthand notation to represent the round-trip delays associated with the pings during a measurement interval.

For loss, each measurement interval results in a single loss metric, as explained in Section 2.2. The resulting set of loss metrics can thus be visualized as shown in figure 25

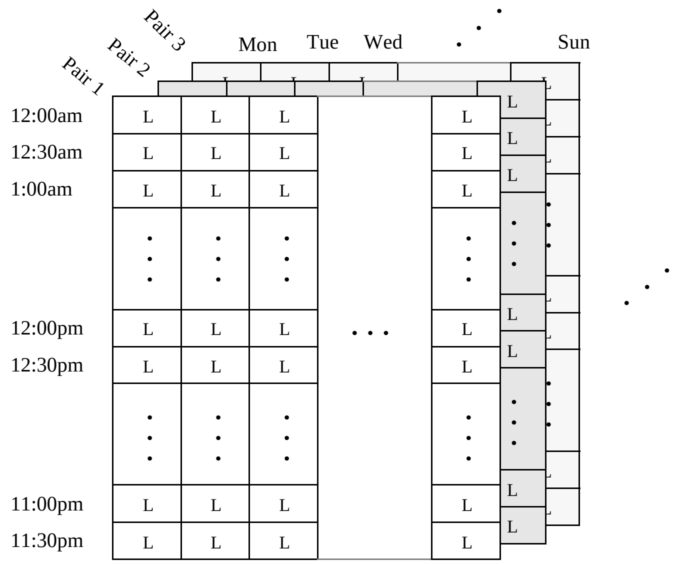

FIGURE 25. Conceptualization of a Large Set of Loss Samples (All Source-Destination Pairs)

B.1 Time of Day Baselines

Figure 26 illustrates how a subset of the data in figure 24 can be used to plot a time of day baseline without any aggregation of data.

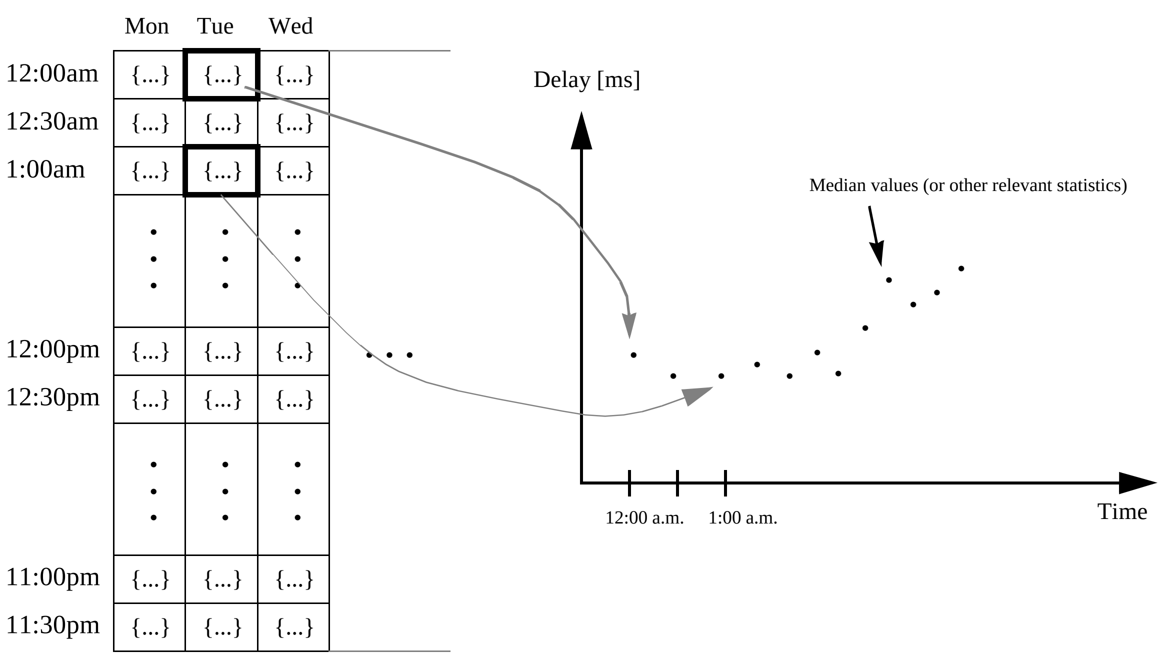

FIGURE 26. Time of Day Baseline (No Aggregation, Single Source-Destination Pair)

A 30-minute baseline (using day of week aggregation of data) is shown in figure 29.

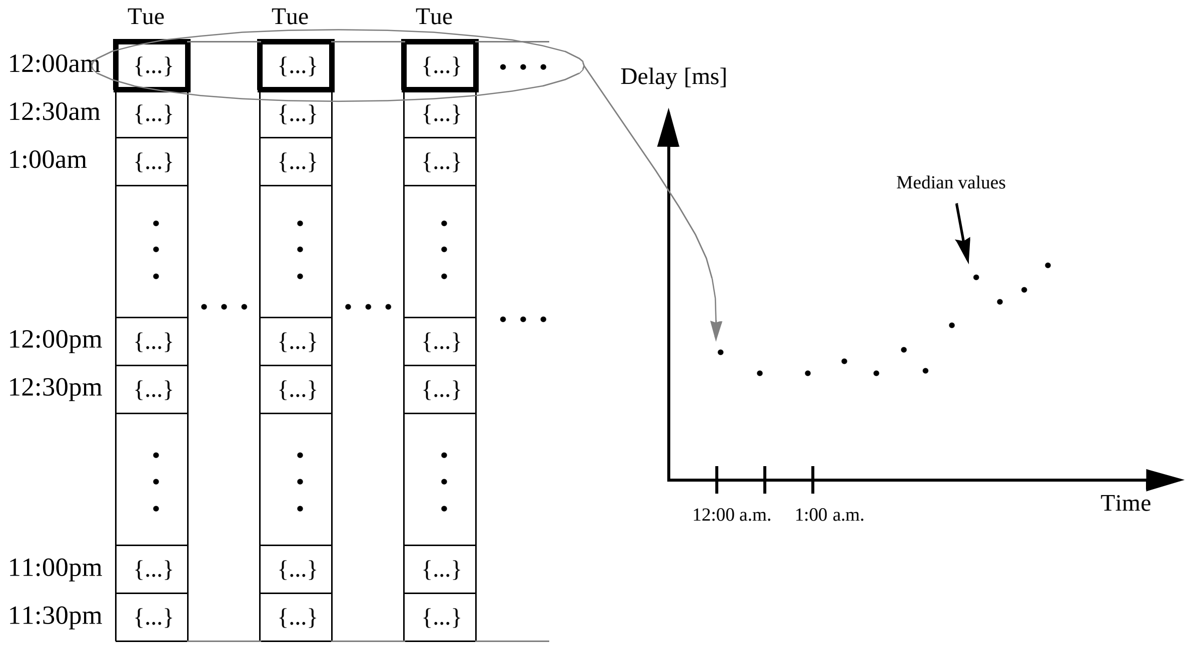

FIGURE 27. **30-minute Baseline (Single Source-Destination Pair)**

B.2 Daily Baselines

A daily baseline, with 24-hour aggregation is illustrated in figure 28.

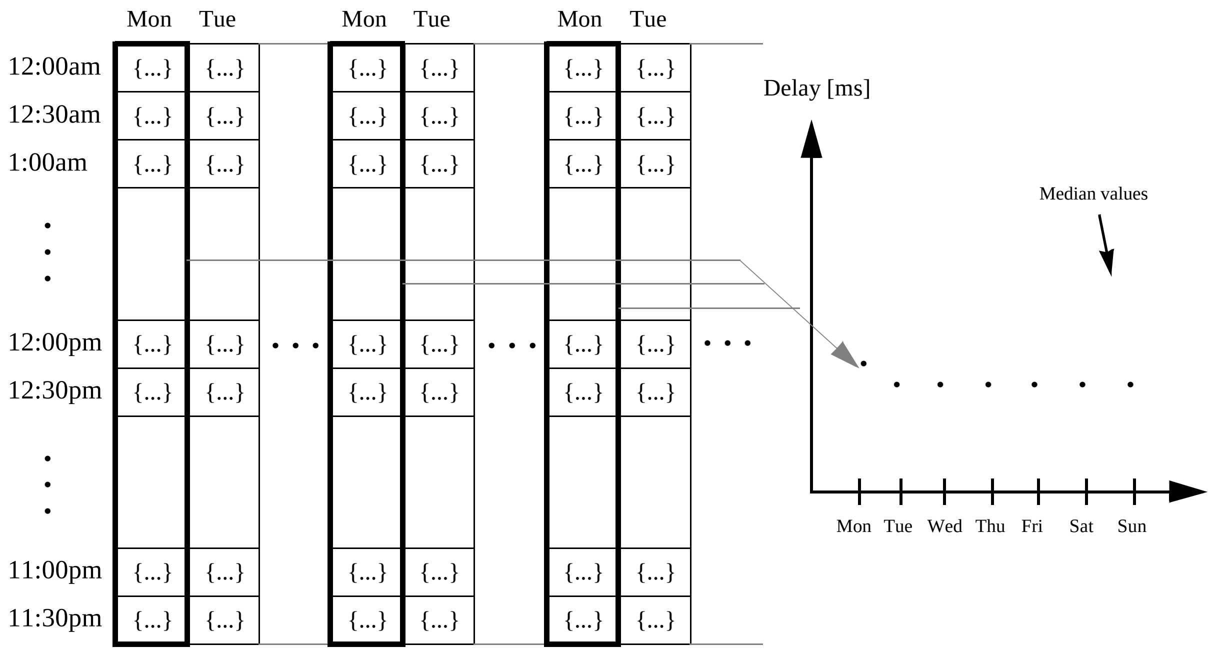

FIGURE 28. Daily Baseline (24-Hour Aggregation, Single Source-Destination Pair)

An example of a daily baseline using busy period aggregation (9:00 a.m. to 10:00 a.m.) is shown in figure 29.

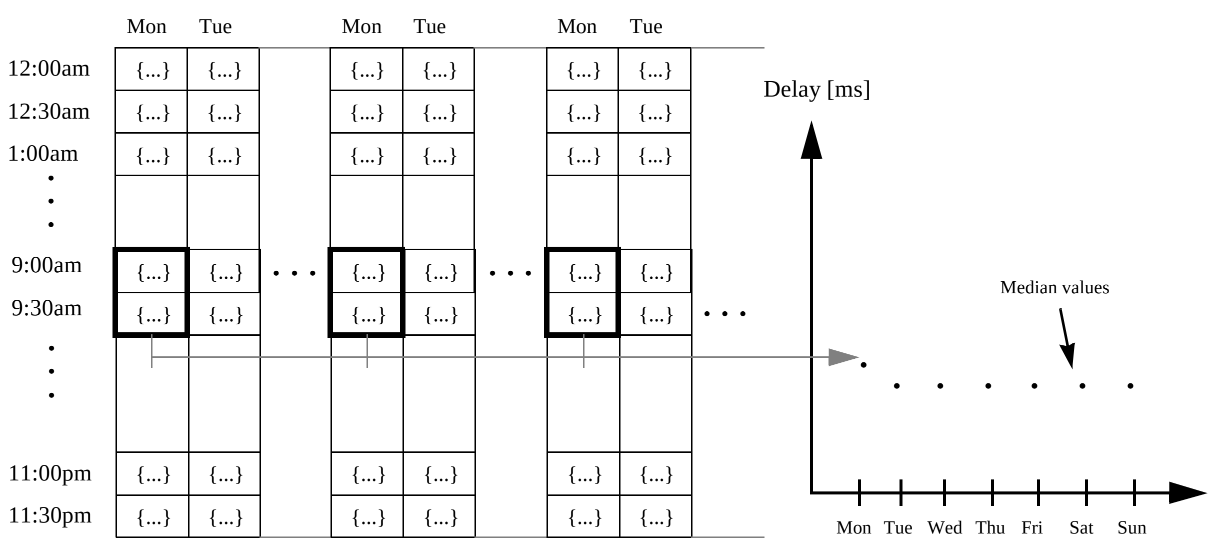

FIGURE 29. Daily Baseline (9:00 a.m.-10:00 a.m. Aggregation, Single Source-Destination Pair)

B.3 Weekday Baselines

A weekday baseline (Monday to Friday) with 24-hour aggregation is shown in figure 30.

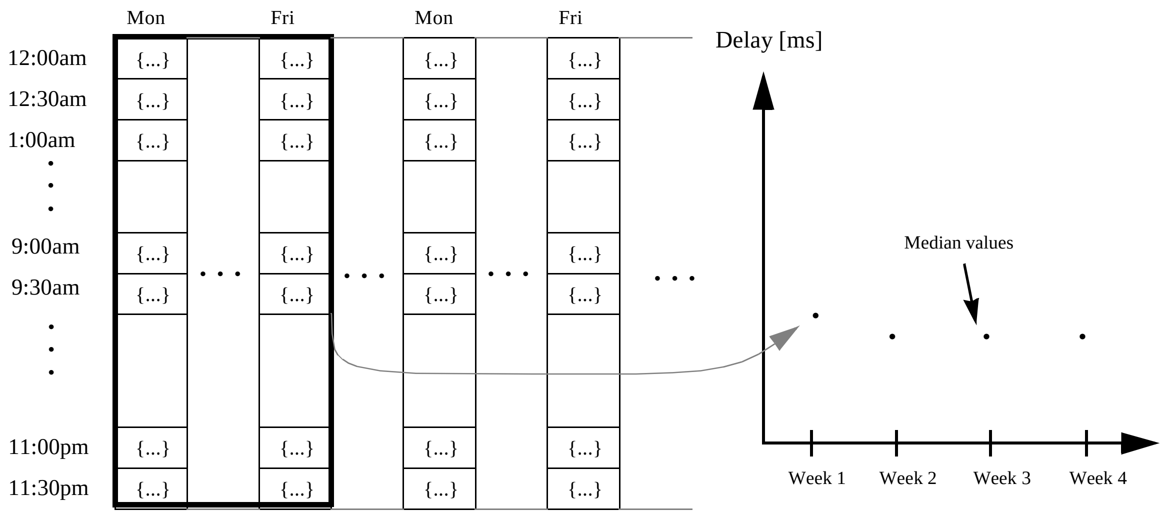

FIGURE 30. Weekday Baseline (24-Hour Aggregation, Single Source-Destination Pair)

Once more, busy period aggregation is possible by using only a subset of each day's data, as shown in figure 31.

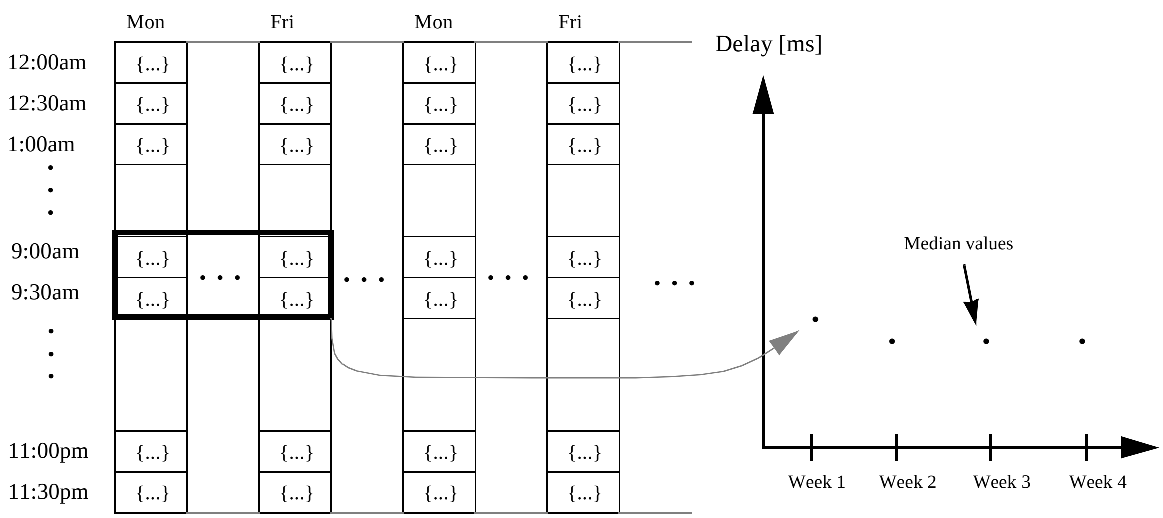

FIGURE 31. Weekday Baseline (9:00 a.m.-10:00 a.m. Aggregation, Single Source-Destination Pair)

B.4 Rolling Baselines

Figure 32 illustrates how a 28-day rolling baseline (for a 9:00 a.m. to 10:00 a.m. busy period) is aggregated. Note that Saturdays and Sundays could be excluded if only a weekday rolling average were of interest.

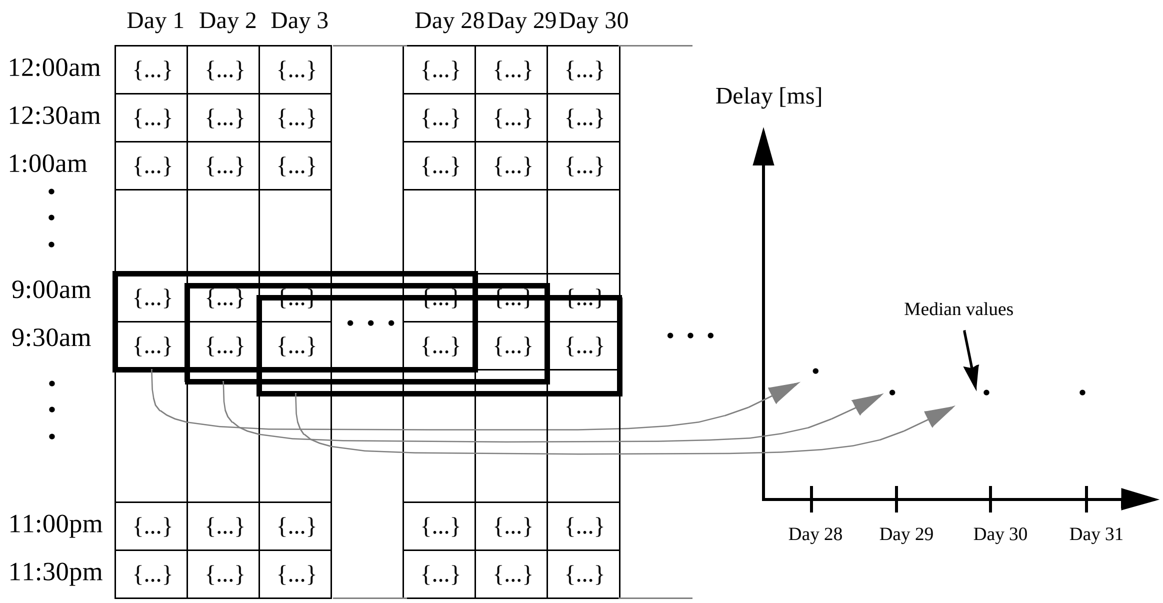

FIGURE 32. 28-Day Rolling Baseline (9:00 a.m.-10:00 a.m. Aggregation, Single Source-Destination Pair)