Measuring and Analyzing Service Levels: A Scalable Passive Approach

Manjari Asawa
Broadband Information Systems Lab,
HP Labs, Palo Alto.
E-mail: manjari@hpl.hp.com

Internet service providers are increasingly trying to differentiate themselves in terms of service performance that they provide to their users. To measure and quantify the service performance from end-users perspective, proper metrics and measurement methods are needed. Traditional measurement techniques are not sufficient for this purpose.

In this paper, we have developed a scalable service level monitoring methodology to assess user satisfaction without injecting any measurement traffic. Specifically, we suggest web throughput as a service level metric, outline possible ways to measure it and discuss advantages of passive observations of actual user activity. We further propose a statistical data analysis method that analyzes passive throughput measurements, and quantifies user satisfaction/dissatisfaction and the confidence that the provider may have on the collected data, i.e. data reliability. The proposed technique is based on the premise that the service provider is interested in continuously monitoring the service levels being offered to a majority of the users over a long enough time. We present results of a real-world experiment that demonstrates that with careful data analysis, passive measurements are effective in detecting service problems. Our experiments also indicate that 90% of the time, the results of reliable passive measurements agree with those of active measurements, without generating any additional measurement traffic. The underlying approach may also provide a communication vehicle between service sales/marketing and operations/capacity planing aspects of service provisioning.

1.0 Introduction

Businesses and home users of information technology are increasingly beginning to view internet service providers (ISPs) and Information Technology (IT) in terms of the services by looking at the information the users receive and the performance the users perceive [14]. This is in contrast to the traditional way of viewing providers in terms of the computing/networking resources and the underlying mechanism. Viewing ISPs and IT as service providers implies that the service levels provided to the end users will become important[1].

Service level agreements (SLA) form a major mechanism for quantifying service benefits and the end users' satisfaction/dissatisfaction/expectation with the delivered ser-

1. It is important to point out that the concept of service levels is closely related to the concept of quality of service (QoS), defined as "collective effect of service performance which determine degree of satisfaction of a user of the service [10]." This definition encompasses many areas, including subjective customer satisfaction. However, within the ITU-T recommendation, the aspects of QoS are restricted to the identification of parameters that can be directly observed and measured at the point at which service is accessed by the user.

vice[11]. An example SLA may define targets such as service availability, service performance etc. As the name suggests, an SLA is an agreement between users and a service provider, and can be explicit or implicit. Service providers need to set SLAs as part of the design of the service, ensure that they are achievable by the underlying technology, and be flexible enough to allow changes to the services as the needs change.

Current works on management of service levels are focused at defining SLAs ([11],[15]) discussions on usefulness and importance of SLAs [14] and on developing various architectures under which SLAs could be provisioned ([3],[12]). Research in the related subject of QoS is primarily focused on mechanisms to enable QoS in the context of individual architectural layers [6]. There is also a limited work on QoS management focusing on management architecture for QoS[4]. To the best of our knowledge, discussions on how service levels or user performance should be monitored seems to be lacking. As John Gallant, Editor, Network World puts it [9]:

> "The sad truth is that many administrators don't know how their network and applications really are performing from an end user's viewpoints. As renowned quality expert Dr. W. Edwards Deming preached, you can't improve quality without effective tools for measuring what you are already doing. Baselining your operations will be important for getting investments in new tools and living up to SLAs."

In this paper, our goal is to develop a good way of expressing and measuring service levels for data services, outline possible approaches for measuring the proposed metric, discuss advantages and disadvantages of each measurement approaches, and also develop methods to analyze collected data to draw useful conclusions about whether the targeted service levels are being achieved or not.

The paper is organized as follows: in Section 2.0, we define and motivate the need for Service Level Objectives (SLOs) which are intimately linked to SLAs. In Section 3.0 we propose user-perceived throughput as a service level metric for data services. In Section 4.0, we outline different methods for throughput measurements and their advantages and disadvantages. In Section 5.0, we explain requirements that any measurement and analysis methodology which measures performance to a subset of user population (i.e. samples) must meet to be able to draw meaningful conclusions about the aggregate user population. In Section 6.0, we devise a method for drawing conclusions about the aggregate user population from the sampled measurements when the service provider has limited control over the availability of sampled measurements. We have also applied the above proposed methodology to a real-world trial with a broadband data service provider, and the results are summarized in Section 7.0. Finally, we present conclusions in Section 8.0 and discuss areas for future research in Section 9.0.

2.0 Service Level Objectives (SLOs)

While SLAs are important from a user's point of view, keeping individual records of service levels of each individual user and ensuring that each individual user achieves its agreed service level is very complex and expensive. This is because current networks are mostly best-effort with very few mechanisms (such as RSVP) in place to control service performance on a per user basis [2]. Hence, performance of an individual user depends on the activities of other users. Furthermore, even if mechanisms for individual user control exists, operating a system near its capacity and meeting agreed service level

for each individual user by sophisticated scheduling, resource allocation, control etc., as well as keeping records of service levels of each individual user requires a huge amount of processing and storage. Service levels provided with such mechanisms will also be very sensitive to user demands.

Due to the above reasons, service providers are more likely to come up with targeted Service Level Objectives (SLOs) for the aggregate user population, and hope to meet SLAs by proper selection of SLOs and by providing enough capacity and continuous monitoring to make sure that targeted SLOs are being achieved. SLOs are chosen such that if internal operations achieve performance equal to or better than specified SLOs, then there is a sufficiently high likelihood that most of the time the individual users will achieve performance equal to or better than the specified SLAs[1]. To make sure that SLAs are met, SLOs will in general require better performance than specified SLAs. For example, an SLA might specify that most user should achieve throughput greater than or equal to 128 Kbps 95 percent of time, and the corresponding SLO may specify that aggregate user population should receive throughput greater than or equal to 150 Kbps 98 percent of the time.

Independent of relationship between SLAs and SLOs, SLOs can provide a communication mechanism between sales/marketing and traditional operations/planning aspects of service providers. This is because while service provisioning/marketing entities focus on individual users, operations/planning entities tend to have an aggregate user population view. Also, to decide a targeted SLO baseline, a service provider must take into account economic factors under the domains of business planning as well as technological factors under the domains of capacity planning/operation. In the first category are factors such as how much users are willing to pay for the service, the expectations and requirements of intended user classes and competition from other alternative service providers in terms of performance and cost. In the second category are factors such as the typical user PC configuration, typical processing load on the PCs, the available service capacity, expected number of simultaneously active users during peak and off-peak hours, user generated traffic pattern, server configurations and load on the servers, etc. If SLO values computed from the technological perspective are not acceptable from the business perspective, or vice versa, additional capacity planning or service modification measures should be adopted to tie the two aspects of service operations.

In the rest of the paper, our focus will be on SLOs. For SLOs to be useful, it is essential that SLOs be realistic, stated in terms of a metric, and be measurable. Traditional network and system measurements are not sufficient for service level measurements., and new measurement methods are needed. Efficient mechanisms are also needed for comparing achieved performance with specified SLOs to help a service provider determine whether they should provide more capacity and/or modify their services.

1. Confidence interval techniques described in the standard statistical analysis literature [5] can be used to ensure proper translation from SLAs to SLOs. If ergodicity [13] could be assumed, then the performance seen by a particular user with time will be similar to the performance seen across users at a particular time instance and the performance guaranteed to the aggregate user population will also apply to most of the users The exact relationship will in general depend on the system architecture and is beyond the scope of this paper.

3.0 Web Throughput as a Service Level Metric

There are many possible different performance metrics that can be used for assessing service levels of various information services. Since a majority of applications currently being introduced are based on web or web-like mechanisms for data transfer, and even legacy applications are moving to web or web-like architecture, we propose to use web throughput as a service level metric. We define web throughput as the amount of *useful* web data that is transferred *reliably* over a network connection between a pair of source and destination nodes per unit of time. As per this definition, web throughput takes into account the overhead of protocol headers, acknowledgments and retransmissions at the underlying layers. It is important to point out that while we have focused on web as an example service due to its popularity, similar definition of throughput is equally well applicable to any other service requiring reliable data transfer.

Throughput has several qualities that makes it useful as a service level metric for web and data services[1]. First, besides being an indicator of user satisfaction, throughput measurements can also indicate system problems that warrant timely maintenance and planning. Second, a majority of other data applications such as E-mail, News etc. also require reliable communications and depend on many of the same system conditions, and hence degradation in web throughput may indicate problems with the other services. Last, many factors that affect measured throughput (such as packet loss, congestion, heavy load, etc.) also affect interactive applications, and hence degradation of throughput may also indicate performance problems with interactive applications.

4.0 Throughput Measurements - methods and issues

Throughput can be measured[2] by either transmitting test data actively to and from user PCs or by passively observing the traffic generated by different users over a period of time. In the former case, since the throughput measurements generate additional load into the network, we refer to this approach as *active measurement*. When the throughput measurement method itself does not stimulate additional traffic into the network to assess achievable throughput, we refer to it as *passive measurement*.

The active measurement method relies on any one of several available tools which perform a user web access and record the number of bytes transferred and the time taken.

1. User perceived response time has also been suggested in literature as a possible service level metric for reliable data transfer. In certain time-critical or interactive applications requiring reliable data transfer, users may also be concerned about response time. However, response time depends very closely on the size of transfer, and requires the operator to give varying resources depending on the size of transfer. Hence, we have decided to focus on the throughput. However, the approach presented here with appropriate modifications can be used to properly interpret measured data when response time is of primary consideration.

2. In this paper, we define throughput as the number of data bytes transferred divided by the time between the transmission of the first bit of data by the source and the reception of the last bit by the destination. Since the first event happens at the transmitter and the second event at the receiver, measurement of the above time interval can not be done precisely at the source or the receiver itself. Measurement of time at the receiver or the transmitter can only approximate the defined throughput due to finite propagation time. All the tools discussed in this section approximates this definition of throughput.

This method has the advantage of complete control over the measurement process, and hence is especially useful during problem diagnosis for individual subscribers. However, active testing also has the disadvantage that additional network traffic needs to be generated solely for measurements. Due to traffic generation, this method worsens any congestion problem with the overhead proportional to the number of users that are monitored, and therefore should not be used for measurements during congestion.

The alternate method of measuring throughput passively relies on measuring and recording performance information during user activities. Hence, no additional traffic needs to be generated for measurements, and throughput values can be collected continuously throughout the operation of the service. Furthermore, the passive throughput measurement technique measures throughput for actual user traffic, rather than for perturbed workloads (as is necessary for active throughput measurements). However, it is important to note that the passive throughput measurement approach does not subsume all the functions of active throughput measurement. During times when the network and servers are under utilized, passive throughput measurements cannot provide any data, and active measurements become necessary for monitoring possible fault conditions.

Passive measurements can be done in two ways. The first method, which we call *client-side passive measurement*, relies on measuring and recording performance at the user's machine, which can be uploaded to a measurement server for further analysis. Tools like Net.Medic from VitalSigns already facilitate such measurements at the client side, and one only needs to integrate such tools with a data upload mechanism. However, this requires cooperation among users and the provider of the service and creates the additional traffic for data upload. The second approach, which we call *server-side passive measurement,* is based on the observation that if the amount of service data transferred is large enough, throughput observed by the web server applications is likely to be a good approximation of the throughput observed by the client side web application. Furthermore, many web servers such as Netscape already log subscriber accesses, and each log entry includes information about the time of access, the IP address from which the subscriber accessed the system, the data transfer size, and the total time for the data transfer. For each data transfer, the throughput achieved can be computed as the ratio of the transfer size to the transfer time.

There are tradeoffs in selecting client-side vs. server side measurement techniques. Client-side passive measurement has the advantage that it can accurately record the performance as experienced by the user. The disadvantage is that mechanisms for performance measurements are needed at the client side and the measurement data needs to be uploaded from user-sites to a central data analysis/measurement site. Data collection at a central site in server-side measurement avoids these difficulties. However, measurements at the server may not be accurate representatives of the performance seen by the user, and may require translation of the measurement data at the server-side to the corresponding service levels at the user side. Server measurements also do not capture queueing delays of user requests before reaching server software and transmission delays from users to server. Furthermore, since the data collection requires processing power and hence takes away some resources, it may not be feasible to initiate data logging in each server. When a web-proxy can be deployed, measurements at the proxy offer interesting tradeoffs between the above two techniques. We will discuss proxy measurements in detail in Section 7.0.

In general, measured values of throughput vary even during the normal operation of the system. This is because, throughput depends on many factors, such as PCs and server configurations, load on PCs and servers, TCP parameters settings, TCP implementation, data transfer size, network characteristics, network capacity and round-trip delay, network load, and error characteristics of the network links, etc. We have investigated the variation pattern of throughput with variations in each of the above factor in [1].

Due to the above factors, if the number of available throughput measurements are very small, then service-level prediction for users other than for whom the measurements are available will be quite limited in its accuracy. However, when many measurements with respect to many different users are available, then proper interpretation of data can provide indications of significant changes in operational conditions that result in unacceptable performance to most of the user population (e.g., periods of extremely low, almost unacceptable, throughput caused by the occurrence of streams of network errors). In the next section, we devise a method by which service providers can draw conclusions about the state of aggregate user satisfaction[1] with respect to SLOs based on the proper interpretation of passively measured throughput values.

5.0 Measurement and Analysis Methodology requirements

From the discussions in the previous section, we can draw many inferences regarding requirements for any methodology to be used for measuring and analyzing throughput values measured passively (either at the server or at the client). While the same requirements apply on actively collected measurement data, due to complete control on the data collection they can be easily met. The requirements are as follows:

1. ***Need for frequent measurement:*** Since single instantaneous throughput measurements cannot be trusted to provide true indications of typical user perceived performance, throughput measurements must be performed frequently to provide meaningful conclusions. The measurement overhead costs can be made small by using passive measurements rather than active measurements whenever possible (see Section 4.0).

   Whether the tests are active or passive, to draw reliable conclusions about user satisfaction, some statistical analysis is needed to remove the variations across user population and across time. Furthermore, the time window for statistical analysis should be carefully chosen. Choosing too small a window may highlight sporadic problems, whereas choosing too large a window may prevent interesting problem trends from being noticed (e.g., occurrence of low throughput at specific times during each day).

2. ***Need to restrict data transfer size(s) used in measurements*:** In general, observed throughput is a monotonic nondecreasing function of data transfer size. The larger the data transfer size, the greater is the impact of the network errors and congestion losses on the measured value of throughput. Hence, in setting an SLO, the data trans-

1. As per SLOs definition, if the performance meets or exceeds specified SLOs, then with a high probability performance to most of the users exceeds specified SLAs, which can be explicit or implicit. Successful SLAs assures that the users are satisfied if they receive performance better than or equal to the performance specified in SLAs.

fer size should be carefully chosen such that it captures service degradation effects, and a specified constant data transfer size (or a small range of data transfer sizes) should be used for the measurements. This can be easily done during active measurements, where the measurements are under operator control. For passive measurements, only a range of data transfer sizes such that the expected throughput is similar for all transfer sizes in the range, should be used. Additional consideration with range selection is that the chosen range should include enough data points

3. ***Need to remove variability in user PC configuration and state:*** When performing throughput measurements to assess user satisfaction, one must remember that measured throughput depends on configuration and state (e.g. running applications) of the user PC, and that measured throughput values will vary with time even when they are measured against the same PC. Hence, throughput performance of a particular PC at a particular time may not be indicative of performance being achieved by the other PCs on the network, and even of the same PC at different time instance.

   Due to the above reasons, useful conclusions about service level status can only be drawn by measuring throughput against multiple PCs chosen at random. Furthermore, to avoid biasing results by few PCs, one needs to select a new set of test PCs at each testing interval. While this can be easily done for active measurements, due to lack of control over passive measurements, one needs to carefully filter data generated by the passive method to remove variability in users' PCs and application performance. Furthermore, to avoid unnecessary false alarms caused by one (or a few) bottleneck users, the analysis method should also remove the bottleneck effects of a few PCs.

4. ***SLO Baseline Selection:*** To obtain meaningful indicators about prevailing service conditions, the observed throughput values must be compared against baseline expectations. As mentioned earlier, an SLO is the baseline performance level that the service provider strives to achieve. For instance, for data services, an operator may wish to ensure that over a one hour interval, there is a high probability that aggregate user population will receive throughput greater than or equal to 2 Mbps for at least 80% of the time.

   The time interval over which statistical analysis and comparison with the baseline are performed should be specified in the SLO. The criteria when warnings/alarms are triggered, the urgency of the alarms, etc., are under service provider control. In general, alarm/warning generation is aimed at (i) alerting service provider when users are unsatisfied with the performance of the service they are receiving, (ii) indicating the degree to which the service has degraded (a measure of "badness"), thereby also indicating the urgency with which repair is necessary.

5. ***Comparing Observed and Baseline Throughput Values:*** While monitoring throughput trends and aggregate user satisfaction, the observed throughput values need to be compared against the operator specified SLO. At first it might be tempting to do straightforward comparison of measured values of throughput with the baseline value and generate a warning as soon as the measurements fall below the baseline. The problem with the above approach is that it does not account for the facts that some throughput measurements may not be an accurate representative of user-perceived performance, that a user who is himself/herself the cause of throughput degradation might be contributing to many measurements, and that many measurements might have been made in a small time interval when the service was temporarily bad. Therefore, to deduce correct conclusions about the network condition and average user satisfaction, statistical analysis techniques are necessary for

comparing the measured throughput values with the baseline over a period of time while taking into account all the variability associated with throughput measurements.

We present several examples to illustrate why a straighforward comparison of the measured throughput values and the specified SLO values may draw incorrect conclusions about aggregate user performance. In doing so, we also derive a set of requirements that any service level analysis approach must address. Figure 1 defines the terminology we use for the examples that follow. Different color bars represent throughput measured to different user PCs. The horizontal time axis is divided into sampling intervals, which are small enough that the service conditions do not change significantly during a sampling interval *S*. In practice, the value of S will be determined by the rate of change of network condition with respect to time. The value of analysis period *P* is determined by the length of time over which if the performance is not satisfactory, the operator would like to be alerted, and will be specified in SLOs.

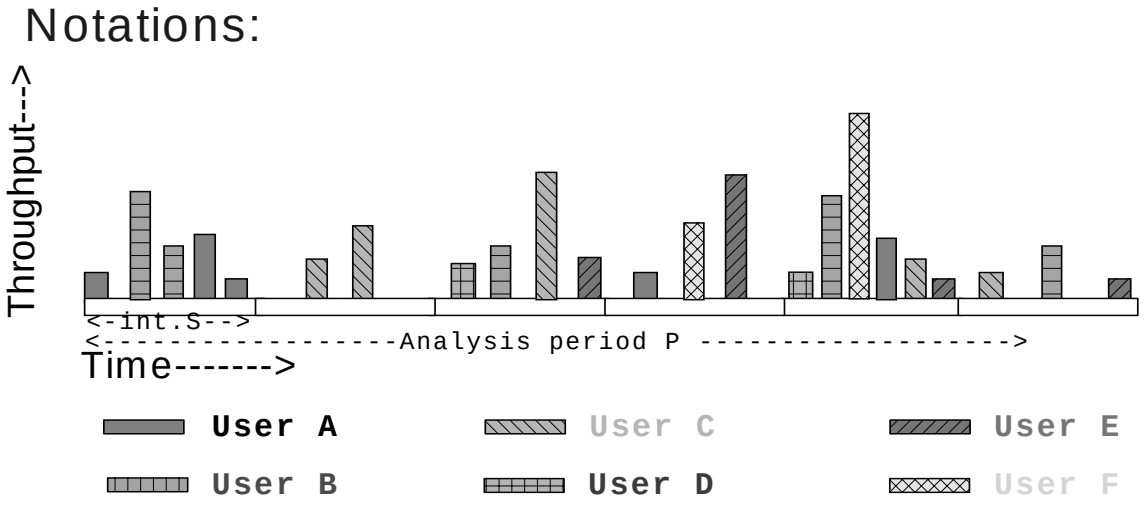

Figure 1 Notations used in the throughput analysis approach

The following examples depict scenarios that the throughput analysis approach must take into account.

1. *Measurement data reliability should be based on the number of users generating data points (Figure 2):* The analysis approach must account for the number of users generating data points in the analysis period. Higher number of users generating data points implies better representation of user population by the available data points. Hence, when the number of users are high, the operator can rely more on available data points for assessing typical user satisfaction.
2. *Relative number of data points from various users should be appropriately accounted:* Figure 3 motivates the need for the throughput analysis approach to take into account the relative contributions of different user users to the throughput measurements. In the two scenarios depicted in Figure 4, roughly the same number of measurements are obtained. However, in the first case, only two users contribute to a

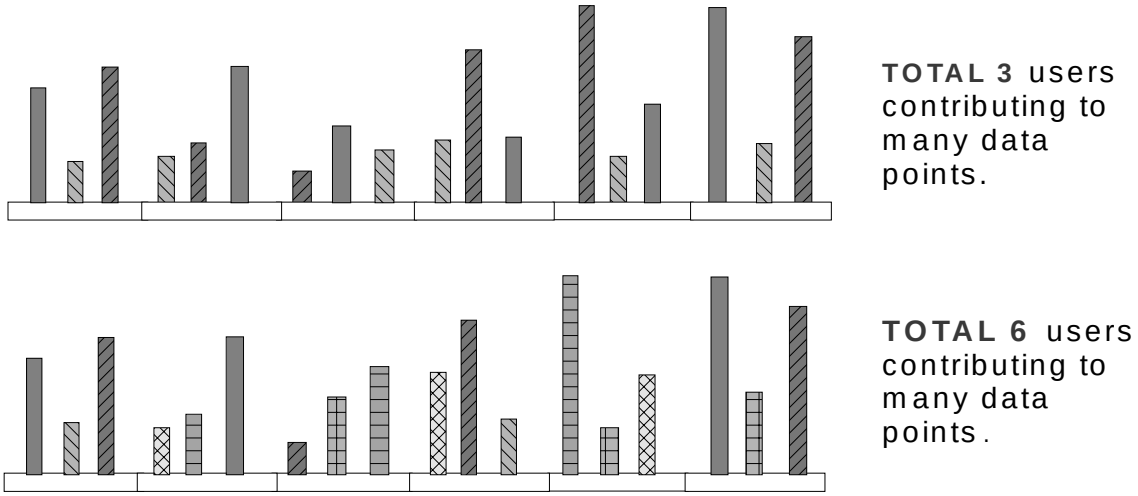

Figure 2 Contrast between instances when different number of PCs contribute to the same number of throughput measurements

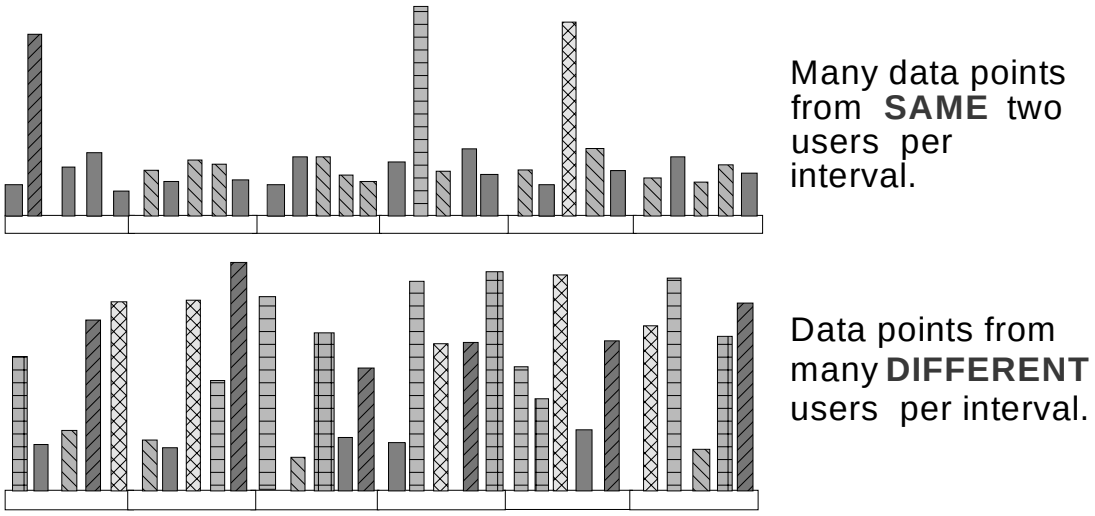

Figure 3 Contrast between instances when few PCs dominate data points vs. many PCs uniformly contributing to same number of data points.

majority of the measurements, whereas in the second case, six users contribute almost equally to the measurements. In the first case, the low throughput values may be attributable to the users themselves, rather than to the service provider's system. Consequently, in deciding status of user satisfaction, the analysis approach must not only take into account the number of users contributing to a majority of the measurements, but must also account for the relative number of data points from each user.

3. *Number of measurements from distinct users in a sampling interval should be appropriately accounted:* Figure 4 compares two sets of measurements of the same net-

work during the same time period. The data in the second case is clearly a more reliable representative of the status of most users than the first case.

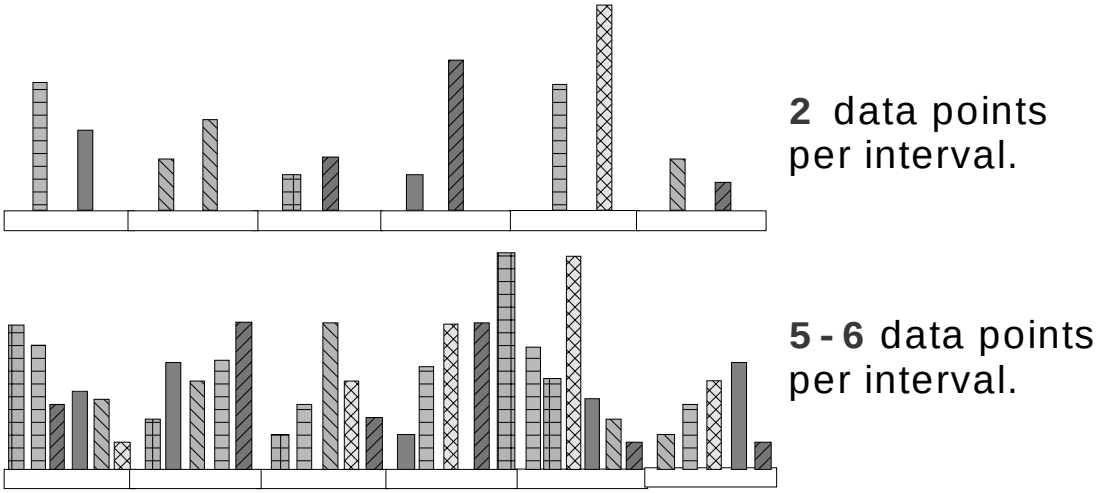

Figure 4 Comparison of scenarios with different number of measurements from distinct PCs in each sampling interval

4. *Clustering of measurement data points should be accounted and the methodology should filter its effects:* Figure 5 depicts this condition. In the first case, most of the measured throughput values are clustered in the fourth samlping interval. During fourth sampling interval, service performance was temporarily degraded due to some transient noise problems. Throughput remained uniformly high during the other sampling intervals. A throughput analysis approach that analyzes the accumulated data without regard to clustering of data points in time may conclude that service performance was degraded over the entire analysis period, even though the performance was degraded only during a small fraction of time.

6.0 Proposed Data Analysis Methodology

Our proposed data analysis method is a statistical method of analyzing data and is based on the premise that the service provider is interested in proactively monitoring the service levels being offered to a majority of the users over a long enough time and that the operational staff does not wish to be alerted about problems that are specific to individual users, or about transient[1] service problems. Problems affecting only a very few users or which are transient need to be handled by some other mechanism than the one presented here, and it might be more efficient to handle these problems reactively rather than proactively.

1. By transient we mean that the time interval over which problem exists is small compared to the time period specified in an SLO (1 hour in the above example), and the problem resolves itself without any human intervention. An example might be the case when there are suddenly a flood of queries to primary DNS, which results in temporary bottlenecks, but resolves itself once the DNS cache builds up, and the load reduces.

The problem is difficult because one needs to appropriately filter for temporal variability (transient problems), and for spatial variability (variability across various users). Furthermore, there are no well established models for user web accesses. Most of the work in statistics [5] and in quality control ([7], [8]) literature has focused on accounting for either the temporal variability or spatial variability when well-established models for the variability exists. In the absence of well-established models and a strong theoretical foundation, we propose a heuristic statistical analysis approach that accounts for all the possible variations, and show validity of our approach by real-world experiments.

In the proposed statistical analysis approach, SLO is specified as follows:

- *Targeted baseline* against which the observed throughput values can be compared. To provide for flexibility in defining a desired performance vs. another which should be absolutely achieved, we provide for specification of both. Hence, an example baseline is defined as: it is desired that aggregate user throughput within an hour should be above $B_{X, desired}$ for more than X percent of time. The throughput must also be above $B_{X, minimun}$ for more than X percent of the time. Hence, $B_{X, desired}$represents the desired throughput, and $B_{X, minimum}$ represents the minimum acceptable throughput that any user should experience. One can also interpret $B_{X, desired}$ as the desired performance levels achieved by group of users, whereas $B_{X, minimun}$ represents the performance that should be met by almost all the users.Desired and minimum acceptable performance could be identical.
- *Data transfer size* (or a range) that is used for measurements (e.g., 50 Kbytes). This is denoted by *T.*
- *Analysis period:* the time period over which throughput values are compared against the baseline (e.g., once every hour in the above baseline example). Let us denote the analysis period by *P.*

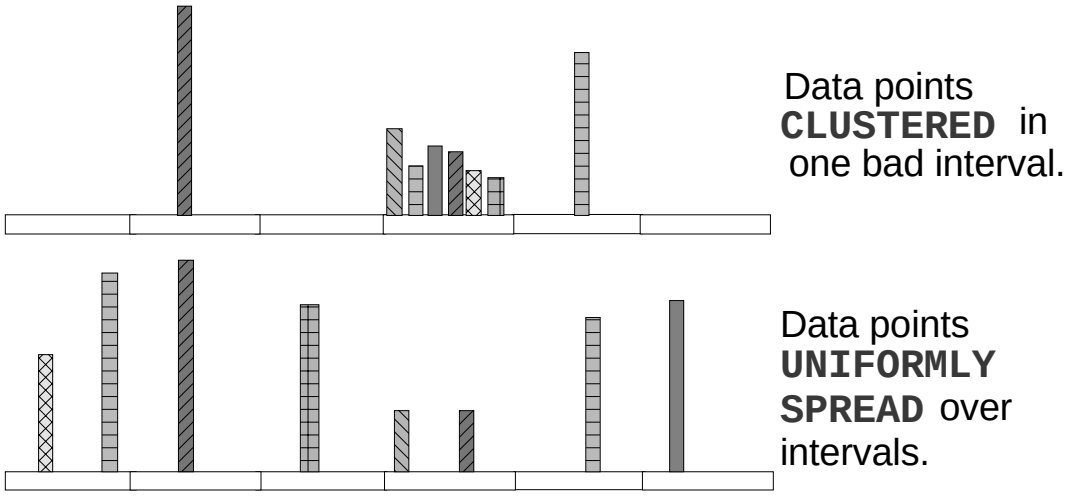

Figure 5 Demonstration of the possibility of throughput measurements being clustered in one sampling interval

To address the problems associated with clustering of data with time and the ability of temporary situations to affect most of the data points, a sampling interval, *S*, is also specified (e.g. 5 minutes).

To minimize the measurement overheads, the throughput monitor can use a combination of passive and active measurement. Passively collected measurements should first be filtered with respect to data transfer size *T* to include only those measurement points for whom the associated data transfer size lies in the specified range. During every sampling interval, the throughput monitor can check if the passive measurements have yielded sufficient data, and if not, can schedule active measurement. Alternatively, active measurements can be performed periodically, but at a much lower frequency than if they were the only measurements. All further data analysis is performed on the combined filtered data.

The proposed throughput analysis approach computes a data reliability parameter, *DRP*, and a representative throughput value, Q_X, and uses them to compute user population dissatisfaction factor *DSF*. *DRP* represents confidence that an operator should have on the measured throughput values as representing performance perceived by most of the users. Q_X is obtained by filtering the raw measurement data to account for simultaneous multiple connection and clustering effects. *DSF* represents service degradation with respect to the specified SLO. More specifically, the reliability parameter helps in selecting an appropriate baseline point in the range $B_{X,\,desired}$ and $B_{X,\,minimum}$, against which the measured throughput, Q_X, should be compared. The selected baseline point draws closer to desired baseline performance, i.e. $B_{X,\,desired}$when we have high confidence on the data as being a representative of user population. To avoid generation of unnecessary alarms, selected baseline draws closer to minimum acceptable performance, i.e. $B_{X,\,minimum}$, as the confidence that the collected data represents entire user population decreases.

To derive a reliability parameter for an analysis period, the following steps should be preformed in the order they are enumerated:

1. **Averaging:** The throughput values measured from the same user over a one minute interval are averaged as the total bytes transferred divided by the total time taken. The commonly used web data transfer protocol, HTTP, opens simultaneous multiple connections per web request to get inline images etc., and each of these connections can achieve varying throughput. Since user web experience depends on aggregated data transfer, the averaged throughput value better represents user's web experience. Let us denote the time period over which averaging is performed by *A*. In general, a good choice for *A* is one minute. Averaging of requests within 1 minute effectively averages simultaneous data transfers due to multiple client connections, and yields one throughput number per PC per web request[1] All further analysis is performed on the above average data.
2. **Computation of effective number of users:** To account for the pitfall of drawing incorrect conclusions in a situation where many data points are from one user, but

1. It is also possible that a user launches subsequent web requests within a very short time (less than a minute). In this case, we will be averaging across multiple web requests by the same user. Since service conditions are not expected to change much within such a short time interval, it is still appropriate to aggregate such requests.

very few from all the other users, we account for the number of users contributing to significant number of data points. For this, we compute the number of all the distinct users contributing to the measurements in an analysis period. Let us denote this number by *n*.

Now, a user reliability parameter denoting confidence with respect to different users who have contributed to data points (higher confidence on more uniform data contribution by various users) is computed. Let us denote this confidence parameter by *URP. URP* should be a monotonic nondecreasing function of *n* and should be always between 0 and 1. Furthermore, if the number of users are more than the minimum number of distinct users necessary to have statistical confidence on the obtained data points as representing entire user population (as per law of large numbers), then *URP* should be equal to 1. Let us denote the minimum number of distinct users by *N.* A simple choice of function having the above desirable properties is:

$$URP = min\left(1, \frac{n}{N}\right)$$

In order to take into account possible nonuniformity of data point generation by various users we need to also account for the relative number of measurements obtained from each user. For this, the above *URP* computation can be modified so that *n* represents the minimum subset of users that collectively contribute to a majority of the measurements (e.g., *n* is chosen as the minimum number of users which contribute to 75% of data points. Here, 75% includes most of the data points, but not all).

3. **Accounting for and removal of cluster effect:** To avoid the problems associated with clustered data points with respect to time, the time-axis is divided into small sampling intervals and an interval reliability parameter (*IRP)* is associated with each sampling interval S within the analysis period *P* under consideration. A cluster parameter, *CP,* is also associated with the analysis period *P.* Let us denote the number of measurements in the sampling interval *i*, S_i, by m_i.

   CP should have the following properties: first, when the number of measurements in each sampling interval within the analysis period *P* are same, *CP* should be equal to confidence on the data in each sampling interval, i.e., IRP_i for all *i*. Second, when IRP_i differs across all *i*, CP should be proportional to the average confidence across intervals, i.e. average IRP_i. However, from the average IRP_i value alone, it is not possible to predict whether there is a significant variation between IRP_i values for different S_is. Since for the same average IRP_i, uniform values of IRP_i across the intervals indicate more reliable data than widely varying values across intervals, *CP* should also be inversely proportional to the fluctuations in IRP_i values and should decrease with increased discrepancies in the number of data points across S_is. One possible function choice for *CP* with the above properties is:

$$CP = \frac{avg_i(IRP_i)}{1 + max_i(IRP_i) - avg_i(IRP_i)}$$

   The reliability parameter of S_i, namely IRP_i, should be a function of the number of measurements m_i in S_i, should increase monotonically with decresing rate of increase

with increase in m_i (i.e. a convex function of m_i) and should always be between 0 and 1. One possible choice for such a function is:

$$IRP_i = 1 - \frac{1}{1 + m_i}$$

With such a choice of IRP_i, CP is also always between 0 and 1.

While *CP* accounts for the clustering effect, it doesn't solve the biasing problem, the problem where the overall percentile calculation are biased by data points clustered in one or few sampling intervals. To solve the biasing problem, we must restrict the maximum number of data points that are selected from each sampling interval by further filtering the averaged data of step 1. If any interval S_i has more data points than the maximum, we sort the data points based on the associated throughput values and choose a certain number, say T, of median values as a representative throughput values for S_i.

4. **Computation of overall data reliability:** The *URP* and *CP* values computed for each analysis period serve as the basis for computing the overall data reliability parameter, *DRP,* for the analysis period *P.* Since greater confidence in the measured values as representative of true user population should imply higher *DRP, DRP* dissatisfaction factor should be proportional to both *URP* and *CP.* One simple choice of function with such property is

$$DRP = URP \bullet CP$$

5. **Representative throughput computation:** Next, we need a representative throughput value for the analysis period. For this, we compute the *X* percentile value of filtered throughput data from step 3 over the analysis period. If over an analysis period, there are less than 5 filtered data points, we disregard the computation and inform the operator that there are very few data points to be able to draw any meaningful conclusions about aggregate user satisfaction.

   If there are a sufficient number of data points, then service performance is deemed to be unsatisfactory and an alarm is generated if the computed percentile value is less than the provider specified baselines $B_{X,\,desired}$ and $B_{X,\,minimum}$ weighted by the overall data reliability parameter *DRP.*The weighting should avoid unnecessary alarm generation due to biasing effects of poor performance offered to a few subscribers. Intuitively, it means that when we have lower confidence on the collected sample data as being representative of entire user population, we should compare it with the minimum performance that the operator will like to provide to every user. When there is a high confidence on the collected data as true representative of user population, we should compare the collected data with the desired performance objective.

   An example of alarm generation condition with the above desired properties is that if the computed *X* percentile of data filtered in step 3, denoted by Q_X , is such that:

$$Q_X < DRP \cdot B_{X,\,desired} + (1 - DRP) \cdot B_{X,\,minimum}$$

   then an alarm is generated.

6. **Computation of dissatisfaction factor:** A dissatisfaction factor, *DSF, c*an also be associated with the above generated alarm whenever service performance is unsatis-

factory. *DSF* should increase with the increase in the difference between the representative value of throughput and the baseline (which is $DRP \cdot B_{X,\,desired} + (1 - DRP) \cdot B_{X,\,minimum}$). We define *DSF* as the ratio of the above averaged difference multiplied by the number of points that are below the baseline, and the above baseline multiplied by the total number of data points**.** The motivation for the above approach is that user satisfaction/dissatisfaction is proportional to the percentage difference from the desired baseline.

In the above framework, we have proposed a methodology for analyzing and monitoring operational conditions in terms of service levels, and hence SLOs. We have also provided a web-interface to facilitate instantaneous view of the satisfaction of the aggregate user population with respect to a specified baseline, reliability on the collected passive data as representative of aggregate user population, and the effective measured service performance. These results can help a service operator decide whether a particular condition requires immediate attention or not. If the results indicate high user dissatisfaction and high data reliability, then an immediate action should be most likely taken. If user dissatisfaction or data reliability is low, then the service operator may prefer to wait for some more time and observe if the condition is persistent before taking an action.

The graphical view of above results can also give an indication of the type of problem. If problems occur occasionally then it might be an operational problem and need to be solved by repair, whereas if the problem persists despite possible maintenance, then it is most likely a capacity problem. Capacity planning can be done by applying the proposed methodology on longer terms (weeks and months). For example, Q_X and *DRP* values can be recorded every hour, and then compared against longer term baseline over a longer time window (for example, weeks or months). This method reduces the amount of data that needs to be stored, and also displays useful information to network managers in a meaningful and concise form. If the analysis indicates service performance to be unacceptable or close to the minimum promised performance in a consistent basis with no operational problem, then the service operator should plan to either increase the capacity or modify the provided service.

To appropriately handle any service, the environment under consideration should be properly understood. Our proposed methodology eases the task of making decisions by representing data in a meaningful way. At this point, it is also important to point out that there are several other possible approaches that can be used for generating warnings/ alarms, determining warning/alarm criticalness, and for capacity planning. The method which is most suitable depends on the characteristics of a particular environment. In the next section, we have applied the proposed methodology to a real-world broadband data service environment and have demonstrated its usefulness and validity in the presence of all possible variations in the real-world.

7.0 Broadband Data Service Experiment

To demonstrate that the proposed methodology is both practical and useful in the real-world, we applied it to a real world broadband data service trial, and we summarize our observations in this section. The description also illustrates steps and issues that may arise while using the proposed methodology to understand aggregate user satisfaction.

Our group was involved in studying performance of a data service trial with a broadband service operator. The system architecture for their web based data services is shown in Figure 7. The system provided two-way data transmission. The user population was divided into domains, with each domain served by the same access network. Hence, within a domain, service levels delivered by access network are somewhat similar. All web accesses must pass through the proxy server.

Susceptibility of the access network to a variety of communication problems due to noise was causing many performance problems in the data services. To monitor and maintain service performance, the service operator needed tools that could assess user performance, analyze performance trends, and advise about management requirements of the system. Our proposed methodology of user performance assessment found an immediate application in their environment. Since the service operator was solely responsible for the performance seen on the local access network, we focused on the access network performance. We selected the averaging interval, *A*, to be 1 minute, sampling interval, *S*, to be 5 minutes and the analysis period, *P*, to be one hour. In the absence of business or operational guidelines for an appropriate SLO, we based baselines $B_{X,\,desired}$ and $B_{X,\,minimum}$, on measured data. Furthermore, our analysis was applied after the trial was over, and hence we had limited control on the data collection.

Since, all web accesses must pass through the proxy, we used passive measurement logs at the proxy. In Section 7.1, we characterize traffic passing through the proxy and accuracy proxy logs. In Section 7.2, we describe active throughput measurements that were performed at the same time as passive data was being collected at the proxy. Section 7.3 describes the methodology we used for baseline selection. In Section 7.4 we validate our proposed methodology against active measurements. Finally, Section 7.5 explores sensitivity of proposed methodology with respect to its parameters, *A*, *S* and *P*.

7.1 Proxy traffic characterization and data accuracy

The installed web proxy itself measured the residence time of each web request (from the time the request is received by a proxy thread to the time the service of the request is completed by the proxy), and the amount of data transferred by each web request. Hence, the logged values could be used to compute throughput observed by the proxy. Also, access network web throughput performance is the same as the performance seen by the user when the web request is served by the proxy cache (HTTP code 200 with proxy cache hit). If a significant fraction of web accesses were served by proxy-cache, and if user perceived cache-hit performance was similar to the proxy recorded cache-hit performance, then the proxy logs could be used to passively assess performance experienced by the aggregate user population. Our proxy log analysis showed that approximately 40 percent of web requests were served by proxy cache.

While relating the proxy recorded cache-hit performance to the user perceived cache-hit performance, one faces the following two difficulties. First, the proxy recorded residence time is measured by the proxy thread that receives the HTTP requests, and hence the proxy measured residence time did not take into account the time to establish a TCP connection and the time spent by the request waiting in the proxy server's TCP listen queue. To solve this, we used netstat data (netstat was run periodically) to check if the proxy TCP queue was building up and/or if the proxy was overloaded.We found that in the proxy was rarely a bottleneck and hence it did not impact any calculations.

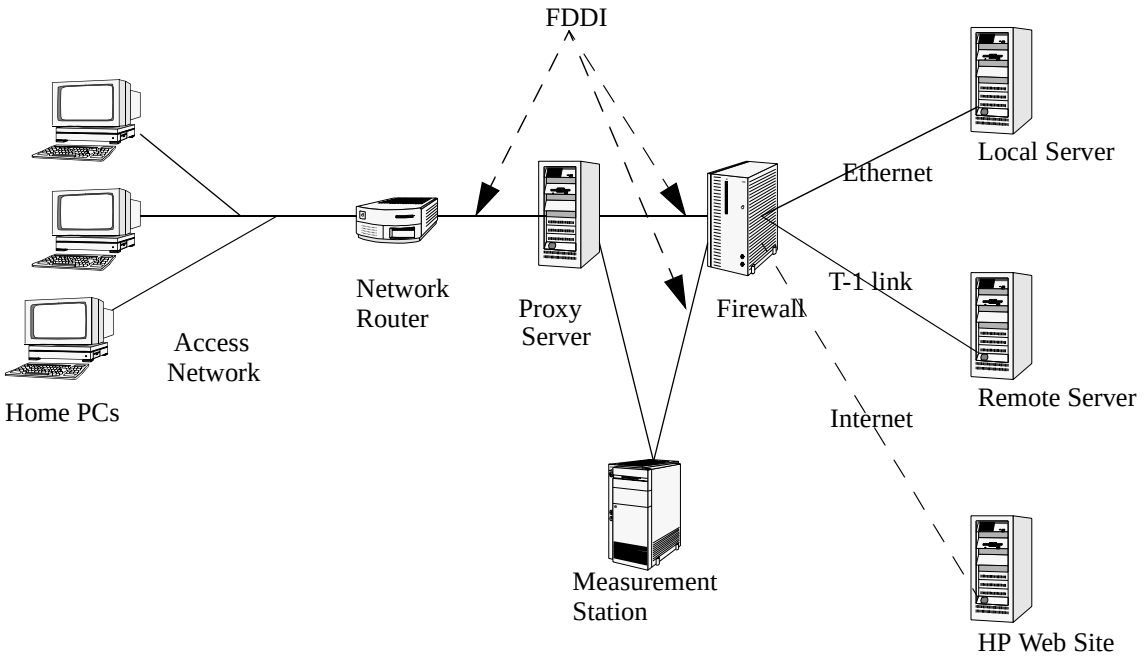

Figure 6 Broadband data trial architecture.

The second difficulty was that the proxy did not record the time to send the last socket buffer data. This was due to the fact that once all the data is transferred to the TCP layer, HTTP server process marks it as completed. Hence the recorded time did not include the time taken to send the last TCP socket buffer data, which was of size 32 KByte. Hence, we considered only those logs for which the logged data transfer size was greater than 32 KBytes, and subtracted 32 KBytes from the transfer size in throughput computation.

With the above filtering and computation, we found that for large data transfer sizes, proxy recorded throughput performance and throughput performance of an emulated web user at the measurement station were within a small percentage of each other. However, the performance differed greatly for data transfer sizes just above 32 KBytes. The reason being that the TCP connection set-up time was accounted for in measurements from the users, and was not accounted in measurements at the proxy. Therefore, we decided to look only at the access logs with data transfer sizes greater than 36 K[1].

1. There are tradeoffs in the selection of appropriate data transfer range for consideration. The higher the data transfer size, the closer the measured proxy log throughput will be to the actual throughput seen by the user. However, the higher the threshold for data transfer above which logs are considered for assessing user satisfaction, fewer will be the available data points. Since, reliability on data as a measure of aggregate user population perceived performance increases with the number of data points, we want this threshold for data transfer size to be such that we do get a significant number of data points for the analysis. Our experiments indicated that about 5 to 12 percent of cache hits were greater than 36 KBytes, and that 36 KBytes (32 KBytes socket buffer + at least 4 KBytes data transfer) yields good compromise between the above two conflicting requirements.

To summarize, for collecting passive measurement data points, we considered only those proxy recorded web-access measurements which were served by proxy cache, were requested from users within a single domain, and were greater than 36 Kbytes.

7.2 Active measurements

We are interested in the relationship between true performance levels received by a group of users within a domain and conclusions from the proposed methodology about the aggregate user population satisfaction. To know the first quantity, we needed user-side measurement of the received performance from all the users and a mechanism to transfer this data to the service provider. Since this required modifications at the users' side, outside our control, for validation of the proposed methodology we used active measurements that our group had performed.

In the active measurements, throughput tests were performed every sampling interval from the measurement station to a few selected user PCs within a particular domain. PCs were selected randomly among the list of active PCs, except that PCs which were found to be very slow were not selected. A PC could be slow because of many reasons, such as being a low-end machine, improper configuration, a resource intensive application execution etc. With this method of PC selection, the probability of the selected PCs being the cause of performance bottleneck was low, and any measured performance degradation was most likely due to performance degradation in the access network.

We analyzed about 2 months worth of the proxy and the active measurement logs. We studied the relationship between the proxy recorded performance and the active measurement performance for few selected users and found them to be similar. Using other measurements deployed by our group and communications with field personnel, we selected 10 days during which no problems existed or were reported. We then computed an active baseline, B_{active} ,as the median of active tests during these 10 days, and found it to be 1200 Kbps.

We also analyzed active measurement data per analysis period *P*, and indicated the service performance during an hour to be unacceptable if all the user measurements were below B_{active} , and acceptable if all the measurements were above B_{active} . Since we used the active measurement data mostly for validation of the proposed methodology (Section 7.4), we disregarded those analysis periods during which some PCs were below B_{active} and some were above. This was because in these cases, it was not clear whether the access network was experiencing a problem or not.

7.3 Baseline computation

To account for the throughput variation with the data transfer size in $B_{X,\,desired}$ and $B_{X,\,minimum}$ computation, we assume that the distribution of data transfer size from cache hits remains the same with respect to time. This assumption implies that the baselines should remain relatively constant with time, and that the baselines computed for a representative sample of time can be used for assessing performance during other times.

For baseline computations, we applied the proposed data analysis algorithm on the filtered proxy data of Section 7.1 for the same 10 days that we had chosen for B_{active} computation in Section 7.2. As mentioned before, in the applied analysis, we set

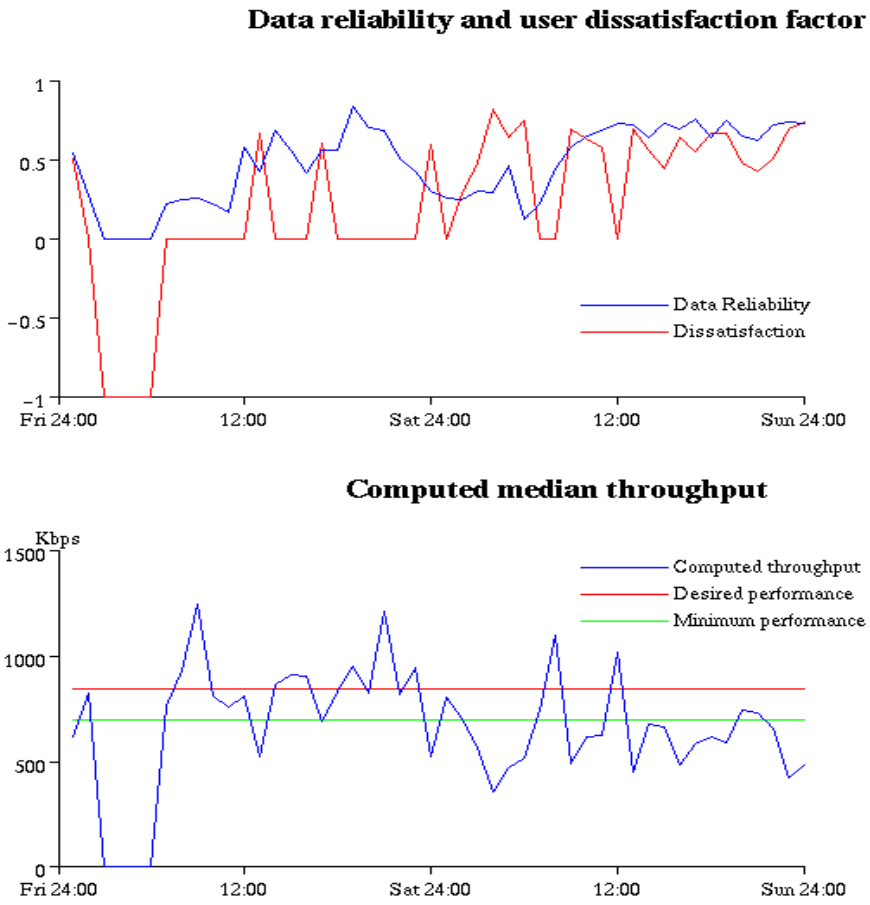

Figure 7 Graphical display of analysis results.

the sampling interval *S* to be 5 minutes and the analysis period *P* to be one hour. Baseline for a particular day, donated by $B_{X,day}$, is computed as the ratio of the sum of reliability parameter *DRP* multiplied by the computed throughput Q_X over the hours of the day and of the sum of *DRP* over hours for the same day. With this, $B_{X,day}$ is essentially a weighted average of performance over the day under consideration.

Next, we computed minimum and desired baselines across all days. Since we did not have any known problems and performance was considered acceptable during the 10 days chosen earlier, we computed minimum acceptable performance, i.e. $B_{X,minimum}$ as the minimum of $B_{X,day}$ over the chosen days. Desired performance, i.e. $B_{X,desired}$ was computed as the average of $B_{X,day}$ over the same 10 days. We chose not to compute $B_{X,desired}$ as the maximum of $B_{X,day}$, since maximum was the best performance observed over chosen days, and it is not reasonable to expect replication of best performance all the time. Our computation found $B_{X,desired}$ to be equal to 850 Kbps and $B_{X,minimum}$ to be equal to 700 Kbps.

Values of the active and the passive baselines differ since active and passive measurements were made at different points using different measurement tools. Active measurements were performed from the measurement station to the user PCs while the passive measurements were made at the proxy server using real-user access logs. In an experiment where our group looked at the proxy logs in real-time and performed active measurement to the PC contributing to the proxy log entry, the two measured values showed similar relationship as the above computed baselines.

Our implemented web interface showing results of the application of the above proposed methodology is shown in Figure 7. From time 3:00 to 6:00 on Friday there were not enough data points for the analysis to be useful, and hence data reliability and throughput are 0, and user dissatisfaction factor is -1, which means that the number has no meaning in the absence of sufficient data points. Service performance was satisfactory almost all the time on Friday, whereas due to access network problems, performance was unsatisfactory almost all day on Saturday.

7.4 Validation with active measurements

For purpose of validation, we performed a per analysis period analysis of filtered proxy logs of Section 7.1 to assess aggregate user population satisfaction/dissatisfaction with respect to computed baselines of Section 7.3, and compared it with results of the active measurements performed in Section 7.2. Comparison results are shown in Figure 9. The chart on the figure can be broken down into following five regions:

1. **Region A:** Points in this region represent hourly time periods during which both active and passive methodology indicated that the aggregate user population service level is above their corresponding baselines. Specifically, during these hours all the monitored PCs in active measurement tests achieved throughput above B_{active} , and the passive methodology indicated dissatisfaction factor to be zero.
2. **Region B:** Points in this region correspond to the time periods during which active measurements indicate acceptable service levels, while passive measurements indicate unacceptable service levels.
3. **Region C:** Points in this region corresponds to time instances when both active and passive measurements indicated unacceptable service levels. Specifically, during these hours throughput to all the actively monitored PCs was below B_{active} , and passive methodology also indicated positive user dissatisfaction.
4. **Region D:** Points in this region corresponds to time intervals when active measurements indicated unacceptable service level whereas passive measurements indicated acceptable service level.
5. **Region E:** Points on the Y-axis represents time intervals when active measurements could not conclusively indicate obtained service level. i.e. some PCs achieved throughput above and some PCs achieved throughput below B_{active} . Point at the origin corresponds to time periods when there were not enough passive data points to enable meaningful conclusions about user satisfaction. Points in this region are not relevant for validation purpose.

Passive and active methodologies correlate if they draw the same conclusions about status of aggregate user satisfaction. Points in region A and C corresponds to time periods when the conclusions are same, and points in the region B and D corresponds to the

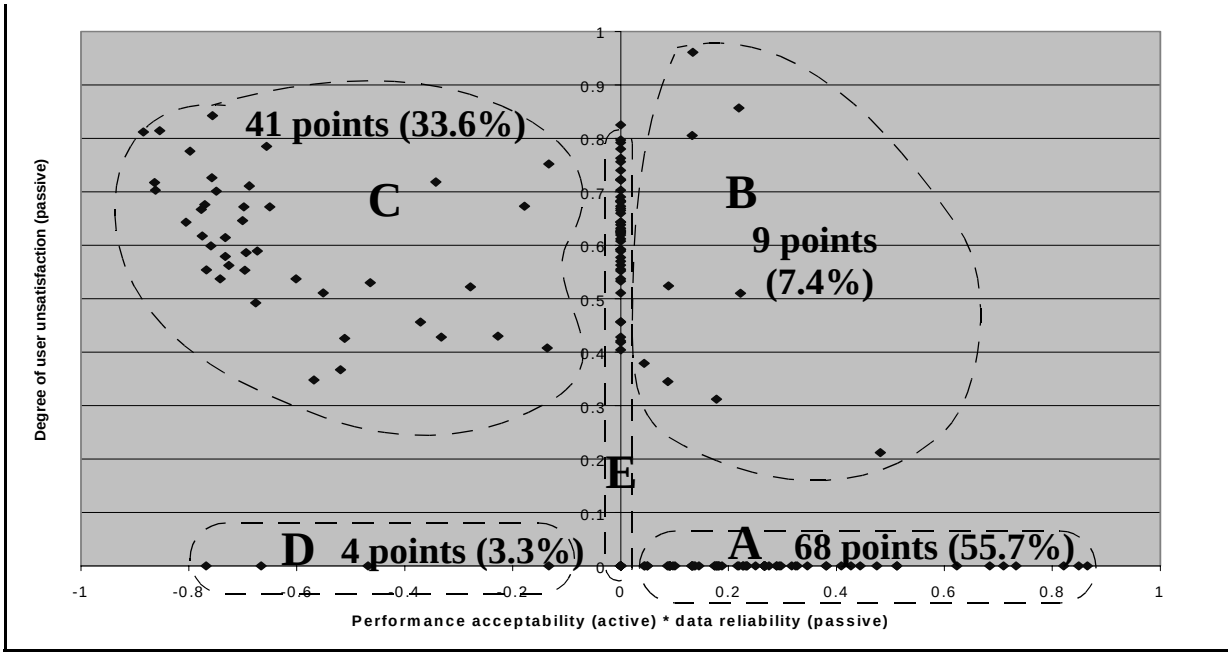

Figure 8 Comparison of conclusions drawn by active and passive methodologies.

time periods when the conclusions are opposite. No conclusions were drawn about service conditions during time periods corresponding to the points on the Y-axis, and hence they are disregarded. Our results show that the conclusions differed (4 + 9) / (4 + 41 + 9 + 68) = 0.1065 fraction of time i.e. 10.65 percent. Furthermore, most of the disagreement between conclusions occur during time periods with low passive data reliability.

To put the above result in perspective, we should recall that both active and passive measurement methodologies are measuring performance to a subset of user population and these two subsets may or may not contain the same PCs, and hence some disagreement should be expected. We also expect the disagreement among results to be more during time periods with low passive data reliability. Figure 9 demonstrates that passive proxy logs with careful analysis can draw similar conclusions as a random active sampling methodology for almost 90 percent of the time, and that the two methods correlate even more during high passive data reliability periods.

The above results show validity and usefulness of passive data collection methodology at the proxy. Since, this method does not create any additional load onto the network, it is much more efficient than active measurement methodology. Furthermore, since the two very different sampling and measurement methodologies (active and passive) draw the same conclusions about aggregate user satisfaction for most of the time, it strengthens our belief that the passive method does represent typical user satisfaction fairly well.

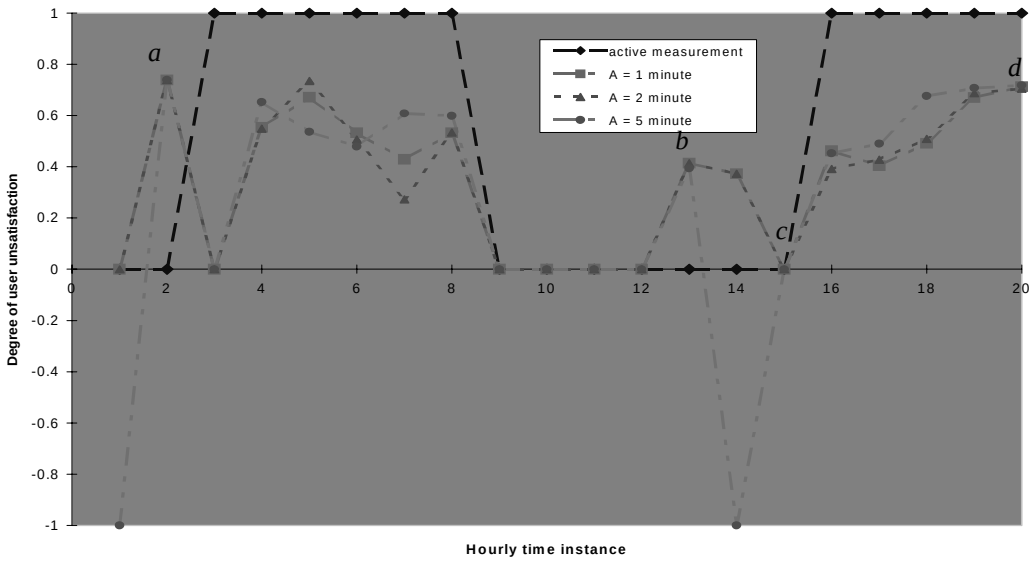

Figure 9 Sensitivity of proposed passive data analysis methodology with respect to choice of averaging interval *A*. *S* = 5 minutes, *P* = 1 hour.

7.5 Sensitivity analysis:

In this section, we present our results for sensitivity study of the proposed analysis with respect to averaging interval, sampling interval and analysis period.

To study sensitivity with respect to a choice of averaging interval, we chose the sampling interval *S* to be five minutes and analysis period *P* to be one hour. With this choice, we get a reasonable number of measurements in every sampling interval and also have 12 sampling intervals within an analysis period. We varied averaging interval *A* to be equal to 1, 2 and 5 minutes. Figure 9 shows the results of passive measurements and the proposed analysis methodology along with the results of active measurements. Negative value along the y-axis represents that there were not enough data points for the analysis to be able to draw any meaningful conclusions about aggregate user population. Since the averaging condenses many measurement points into one, and the proposed analysis methodology disregards any analysis period with fewer than a certain number of measurement data (5 in our case), averaging over longer time (such as 5 minutes in the above example) can make our proposed methodology indicate unavailability of sufficient data points for the analysis. However, if sufficient data points exist for drawing meaningful conclusions by passive data analysis methodology (intervals (a,b) and (c,d)), the results for various averaging intervals are highly correlated and the conclusions are not very sensitive to choice of averaging interval.

Results of passive data analysis with sampling interval equal to 2, 5 and 10 minutes are shown in Figure 10. Again, the results are not very sensitive to the choice of sampling interval. However, since there were very few data points per sampling interval when

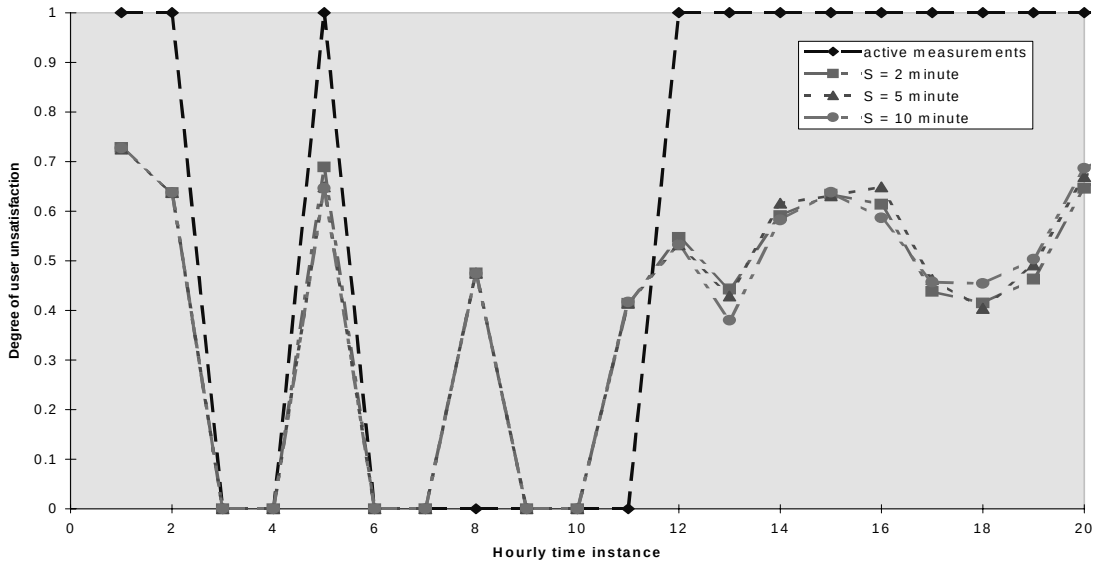

Figure 10 Sensitivity of proposed passive data analysis methodology with respect to choice of sampling interval. Averaging interval is of 1 minute, and analysis period is of 1 hour.

sampling interval was chosen to be 2 minutes, and 10 minute sampling interval resulted in not having enough sampling periods within analysis period. Therefore, we chose sampling interval to be equal to 5 minutes for our experiments.

Results of analysis with respect to 30 minutes, 1 hour, 3 hour and 6 hour analysis periods, *P*, are shown in Figure 11. When P = 30 minutes, many analysis periods did not have sufficient number of data points. Furthermore, results of analysis with longer analysis periods were somewhere in between results of analysis with associated shorter time intervals. For example, results of analysis with P = 2 hour were in between the two results obtained by analysis with P = 1 hour during the associated hours. Both of these results were not surprising, since number of data points per period decreases with decrease in analysis period, and results of analysis over longer time are in some sense average of analysis with shorter time intervals.

8.0 Conclusions

In this paper, we have proposed throughput as a user service level performance metric for data services, outlined possible approaches for measuring throughput, and discussed advantages and disadvantages of each measurement approach. We have also developed an approach and analysis methodology to analyze collected data to assess aggregate user satisfaction with respect to specified service levels. Our approach provides a com-

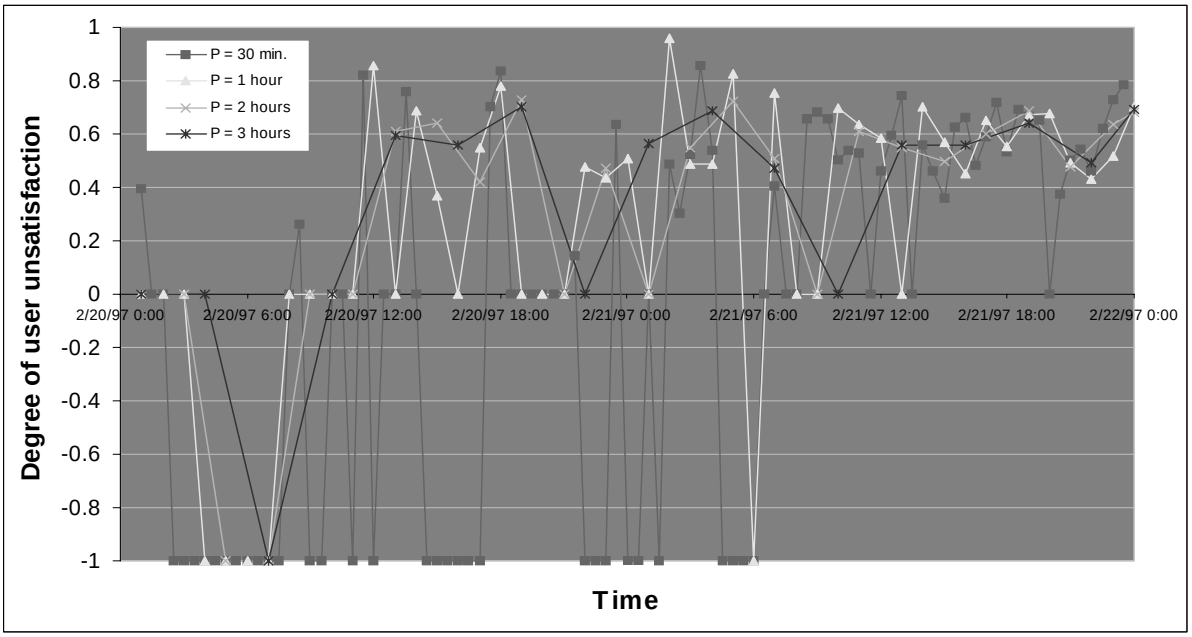

Figure 11 Sensitivity of proposed data analysis methodology with respect to choice of analysis period *P*. Averaging and sampling intervals are respectively of 1 and 5 minutes.

munication vehicle between service sales and marketing, and operations management and capacity planning functions.

Our proposed data analysis approach is especially useful with any passive throughput measurement mechanism, and is applicable independent of whether the throughput is measured by observing user activity passively at the client-side or at the server-side. Since server-side measurements do not require any changes in users' software and do not load network with downloading of performance data from all the users, server-side measurements are operationally easier to use than client-side passive measurements.

We have also described our deployment and data analysis experiment in a real-world trial with a broadband data service provider. In our trial, we have used a web proxy server for server-side passive measurements. We have provided examples of the decision-making process for each step in applying the above methodology. We have also integrated the passive monitoring facility with the analysis prototype, and have provided a web-interface for observing analysis results in graphical format. Our proposed passive data analysis methodology was quite effective in assessing service levels of aggregate user population.

9.0 Future Work

The work presented in this paper is applicable in many contexts and can be extended in the following ways:

1. **Extensions for enterprise networks:** in general, in an enterprise environments, local networks are fast and do not cause performance degradation. Hence, looking at the proxy cache hit records to find performance to our alarm generation mechanism may not be particularly useful. However, if local network becomes the cause of performance degradation, then the proposed method is useful, since our measurements of local enterprise web proxy server show that about 20-25% of accesses were served by proxy cache. It may also be useful to extend the proposed methodology to assess performance of internet service provider used for external connectivity, and for identifying capacity problems in intrants with high-speed LANs inter-connected by slow-speed wide-area links.
2. **Extensions for wireless cable:** wireless cable network architecture is similar to the network architecture of the broadband experiment discussed in Section 7., and proposed method is directly applicable.
3. **Extensions to other QoS parameters**: Since, passive measurements have many advantages over active measurements, it will be worthwhile to explore what other service levels can be measured passively, and how. In our view, service level parameters such as delay and jitter can be measured passively by changes in users' software. To assess aggregate user population satisfaction with respect to the identified parameter, the proposed methodology with suitable modifications can be used.

10.0 Acknowledgments

Author would like to thank Srinivas Ramanathan for useful conversations on the subject of this paper. Author will further like to thank Debbie Caswell, Gita Gopal, Ed Perry and Sharad Singhal for their inputs that substantially contributed to improve presentation quality.

11.0 References

[1] M.Asawa and S. Ramanathan, Throughput measurement methodologies. *working paper*, March 1996.

[2] F.Baker, J.rowcroft, R.Guerin, H.Schulzrinne, L.Zhang, Reservations about Reservations, Proceedings of the Fifth IFIP International workshop on Quality of Service, pp 325-331, 1997.

[3] P.Bhoj, D.Caswell, S.chutani, G.Gopal and M.Kosarchyn, Management of new federated services, Proceedings of the Fifth IFIP/IEEE International Symposium on Integrated Network Management, 541-552, 1997.

[4] R.Bless, M.Jacob and C.Schmidt, Service-Tailored QoS management in High Performance Networks, Proceedings of the Fifth IFIP International workshop on Quality of Service, pp179-190, 1997.

[5] G. Box, W. Hunter and J.Hunter, Statistics for Experimenters, Wiley, 1978.

[6] A.Campbell, C. Aurrecoechea, L. Hauw, “A Review of QoS Architectures”, Proceedings of 4th IFIP International Workshop on Quality of Service, IWQS’96 (invited paper), Paris, France, March, 1996.

[7] B.S.Dhillon, Quality, Control, and Engineering Design, Marcel Dekker, Inc. 1985.

[8] A.J.Duncan, Quality control and industrial statistics, IRWIN, fifth edition, 1986.

[9] J.Gallant, Are you ready for those management challenges?, Editorial, Network World, pp 48, August 4, 1997.

[10] ITU-T recommendation E.800, Terms and definitions related to the quality of telecommunication services.

[11] G.W.Miller, Service Level Agreements: Good fences make good neighbors, Conference Proceedings of the Computer Measurement Group, pp 553-557, December 1987.

[12] G. Pacifici and R.Stadler, An architecture for performance management of multimedia networks, Proceedings of the Fourth IFIP/IEEE International Symposium on Integrated Network Management, pp 174-186, 1995.

[13] A.Papoulis, Probability, random variables, and stochastic processes, McGraw-Hill, 1965.

[14] D. Rohde, ISPs put a premium on Net performance, Network World, September 15, 1997.

[15] V.J. Salminen, Leveraging Service Level Agreements, Conference Proceedings of the Computer Measurement Group, pp 995-1000, December 1989.