PERFORMANCE OR CAPACITY?

DIFFERENT APPROACHES FOR DIFFERENT TASKS

Alexander Gilgur, Steve Politis

(Facebook, Inc.)

With critical contribution from
Josue Kuri, Alexander Nikolaidis, Grace Smith, Yuri Smirnoff, Tyler Price, Paul Sorenson

The person is always happy who is in the presence of something they cannot know in full.
Nicolas Chamfort

ABSTRACT

We often talk about performance and capacity as one thing, and indeed they complement each other in a powerful balancing loop: higher capacity improves performance, decreased performance indicates insufficient capacity, which needs to be provisioned for. However, we often miss the fact that measurement and aggregation approaches that are used in performance monitoring are not always useful for capacity planning, while approaches that we use in capacity planning are often meaningless for performance analysis. This paper explores this gap and discusses ways to reconcile the two tasks.

INTRODUCTION: THE UNCERTAINTY PRINCIPLE

Regardless of whether it is used to describe elementary particles or elements of large systems, the Uncertainty Principle states that we cannot simultaneously describe the position and the movement of an element of our system. This is also in accord with the concepts illustrated by Maxwell's Demon and described mathematically by Boltzmann and later by Gödel.

APPROACHES

Whether we take Maxwell's or Boltzmann's optimistic approach – that the system on the whole behaves in ways that are describable mathematically and therefore knowable – or we take the Schrödinger's procrastinating approach – that we can never know if the health of the cat in the trunk is dictated by the fact that we opened the trunk – we do have two distinct ways to model any system:

- top-down: describe the system's behavior as a whole and then dive into the behavior of its individual elements
- bottom-up: describe the behavior of the system's elements and then compile them into a model of the behavior of the system as a whole.

As performance analysts, we identify anomalies in the behavior of the IT system. As capacity planners, we predict future behavior of the IT system and size the resources to support this behavior in the future. In this paper, we analyze both of these approaches (top-down and bottom-up) and their advantages, drawbacks, and shortcomings from the point of view of performance analyst and capacity planner.

Top-Down Approach

The top-down approach seems to be the most obvious and is often easiest to understand and present. When the "big picture" is followed by a drilldown, we see which changes in the behavior of the system's elements matter for the system as a whole and which are irrelevant.

An illustration of Top-Down approach is in Figure 1

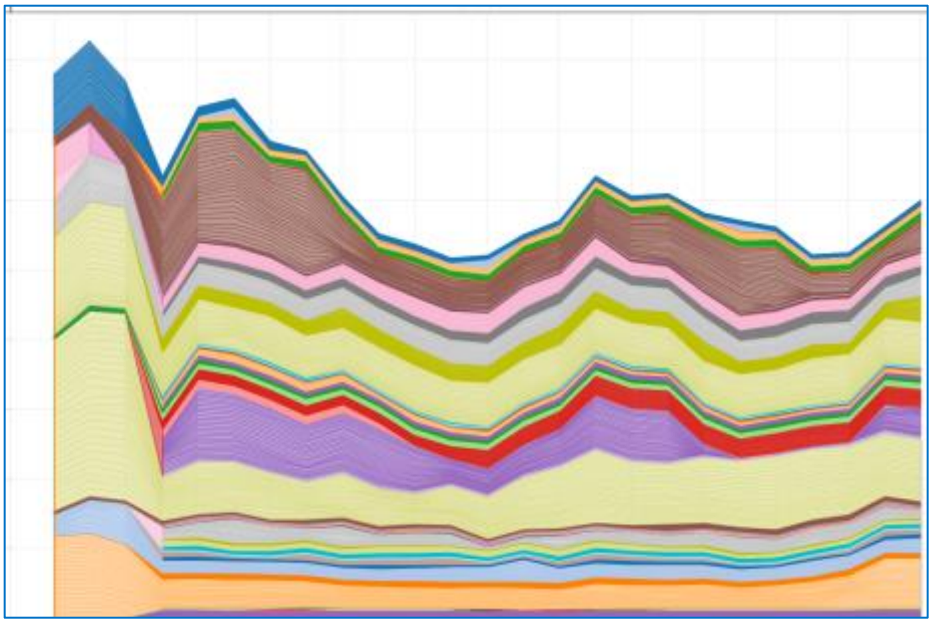


Figure 1: Top-Down Approach (an illustration)

Performance Analytics

For the performance analyst, this approach offers certain advantages, such as:

(i) Compared to bottom-up approach, its implementations lead to significantly lower usage of the computational resources

(ii) We do not need to worry about selecting aggregation function: we already see the end results of all aggregation and multiplexing.

(iii) We immediately see the "grand-scheme-of-things" impact of changes in the behavior of the system's elements.

Capacity Planning

For the capacity planner, this approach involves predicting behavior of the system as a whole, sizing total capacity, and redistributing the capacity once the overall behavior is predicted. The first two advantages outlined above are important for the capacity planner as well. In addition, there is an added value of getting an "optimistic" estimate of the monetary impact of provisioning the new capacity.

However, the shortcomings of top-down approach often outweigh its advantages:

(i) Any "redistribution of wealth" involves uncertainty. For example, applications with low demand in the early stages of their adoption have been known to "go viral" and overwhelm the system provisioned using the top-down approach, e.g., the www.healthcare.gov site or more recently the Pokémon Go game.

(ii) Quantile Compression [GILG2015] will also lead to artificially shortening of the tails of the demand distribution.

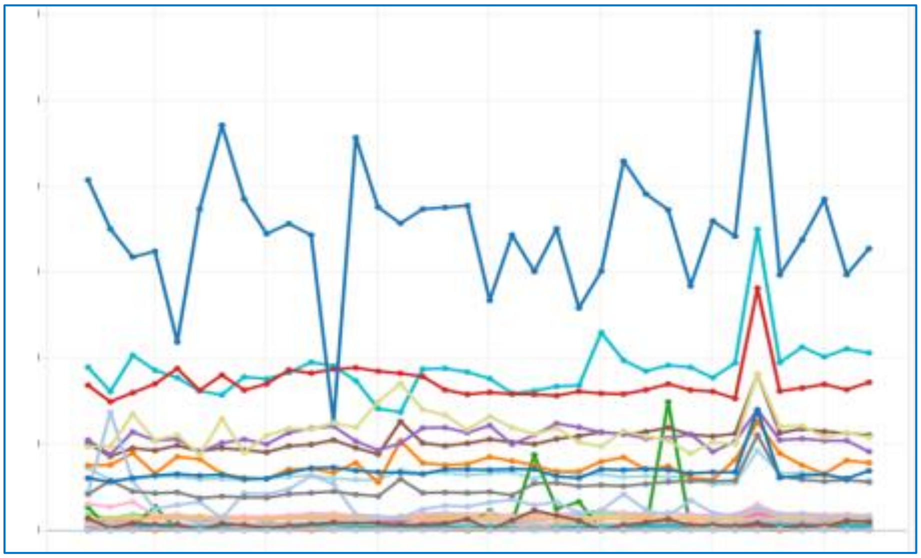
(iii) Finally, Central Limit Theorem will force the distribution of aggregated demand to appear more symmetrical than it really is, leading to underestimation of the tails of the distribution.

Bottom-Up Approach

The bottom-up approach involves playing the role of Maxwell's Daemon: individually analyzing and predicting the behavior of each of the elements of the IT infrastructure system followed by aggregating the predictions. It can be application[1] (service) level; it can be, for a datacenter, cluster or even individual blade level; for a network, it can be the level of (source, destination, SLA) tuple (frequently known as "flow"); for storage, it can be LUN level; etc.

In its pure form, it is the lowest atomic aggregation level that makes sense: e.g., each service's behavior on each cluster in a datacenter, or the effect of the number of application users on edge network traffic for each application and each peering router (or each PoP).

An illustration of the Bottom-Up approach is in Figure 2.



[1] Here "application" and "service" are used interchangeably. Elsewhere in this paper, with the exception of the "Application" section title, the word "application" refers to user-facing software, whereas the word "service" refers to internal programs. In a network setting, applications run on the network edge, while services usually run on the content-distribution-network (CDN) backbone.

Figure 2: Bottom-Up Approach (an illustration)

Performance Analytics

For the performance analyst, this approach too offers certain advantages, such as:

(i) Trend identification

(ii) Anomaly detection

However, it has its drawbacks:

(i) We do not know the practical significance of the anomalies and trends

(ii) We lose sight of the effect of interactions and interdependencies among system elements

Capacity Planning

For the capacity planner, the advantages of the bottom-up approach usually outweigh the drawbacks:

(i) We work with the actual distributions and trends
(ii) We can ([GILG2015]) compensate for quantile compression and predict the actual bandwidths required to ensure unencumbered system operation.

On the other hand, the often present uncertainty about how much multiplexing and interdependency is involved can lead the capacity planner using the bottom-up approach to overestimating the budget for the system as a whole.

Hybrid Approaches

Hybrid approaches can be anything in between the top-down and the bottom-up approach, e.g., a service owner will typically be more interested in the service-level approach to performance analytics and capacity planning: regardless of which clusters are used, the service may need a total of $X\ GiB$ of RAM and Y GPU cores.

On the other hand, an IT performance analyst may be interested in how much RAM is being used on a given cluster and how many machines in that cluster need to be rebooted.

In the “Application” section of this paper, we demonstrate how the hybrid approach can be used in network demand forecasting.

MEASURES AND AGGREGATIONS

The *Z*Sigma* Methodology

The $Z * \sigma$ methodology evolved from Walter Shewhart's ideas for Statistical Process Control (SPC) and was introduced in the 1980s, first by Motorola and later expanded by General Electric, Intel, and a number of other big manufacturing companies. The method became known as "Six Sigma" (https://www.isixsigma.com/) and is very much in use by many businesses where tightening the processes to meet the demanding specifications is one of the goals of continuous improvement. The general idea is to keep the process metrics such that even 6 standard deviations from their respective means they will still be within specifications. This leads to at most 3 defects per million opportunities, which in most industries is considered good process control.

General Considerations

The overwhelming advantages of using mean and variance as the measure of choice for key performance indicators (KPIs) are:

(i) They are additive: $\mu(X1 + X2) = \mu(X1) + \mu(X2)$; $var(X1 + X2) = var\ (X1) + var(X2)$.
(ii) The ratio of mean to standard deviation is the equivalent of the signal to noise ratio.
(iii) The math we use in performance analysis and capacity planning is greatly simplified by using this methodology.

However, it makes certain assumptions:
(i) Underlying distributions are normally distributed (Gaussian).
(ii) Mean and standard deviation of samples are attributable to the mean and standard deviation of the population
(iii) The underlying processes are stationary.
(iv) There are no outliers.

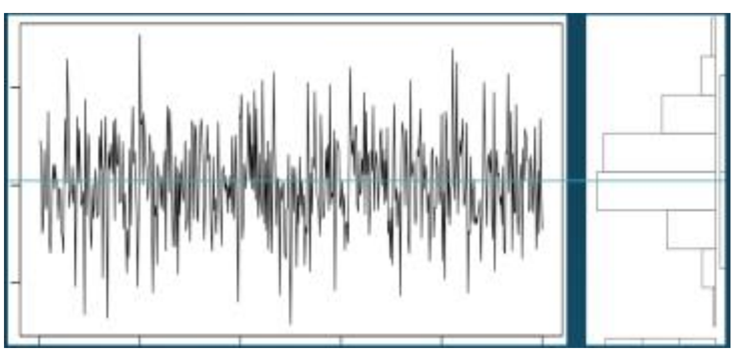


Figure 3 (a): *Z*Sigma* approach will work

These assumptions are very dangerous in IT performance analysis and capacity planning:
(i) Distributions in the IT world are rarely even symmetrical, even less Gaussian.

a. This can be compensated by:
   i. applying Central Limit Theorem.
   ii. using algebraic transformations, e.g., Box-Cox or logarithmic.
b. It is a slippery slope.

(ii) Some distributions observable in IT do not have a mean or a standard deviation.
a. To be certain, we can always calculate the mean and standard deviation of a sample or a group of samples, and the group's means will even be asymptotically normally distributed.
b. However, heavy-tailed distributions are typically observed for latencies and for traffic. Such distributions (e.g., Pareto-like) may have no mean or standard deviation defined.

(iii) Stationarity may be observable at the high level, but rarely at the low level
a. This limits our ability to meaningfully use the additivity of means and variances.
b. Stationarity can be found by finding a stable relationship between the key performance indicators (KPIs) and their drivers (e.g., a business metric).

(iv) There are always outliers.
a. Outliers will pull the means in their direction.
b. This, in turn, will lead to expansion of variance in the direction away from outliers.

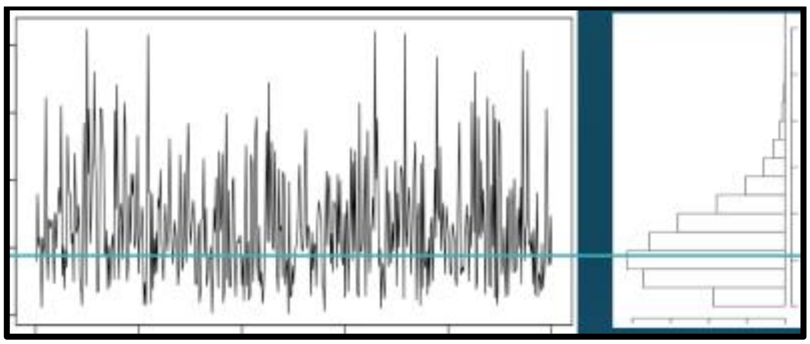


Figure 3 (a): *Z*Sigma* approach will not work directly

Can we still use the Six Sigma approach, with a change in the methodology?

The short answer is yes, we can (see, e.g., [GILG2014]).

Different Zs

One fairly popular, due to its simplicity, technique is to use different values of *Z* to estimate the upper bound vs. the lower bound of the data. This does help offset the asymmetry issues. However, it does nothing about the other, more serious, underlying problems in this approach.

We can coerce it to be used in capacity planning. For performance analysis it is not very useful.

Data Transformations

A number of algebraic transformations have been developed, from log-transformation to left-skew and right-skewed distribution, to power transformations (e.g., the Box-Cox technique). Care must be taken, if working with transformed data, to reverse-transform the model prediction, especially when forecasting for Capacity Planning.

Peak Load

It makes sense, and is usually the right thing to do, for performance and capacity analytics to find the underlying metric(s) that dictate how much capacity is being used, or how the system behaves in response to these metrics (e.g., [GILG2014]). For example, in retail and payment business, it makes sense to plan capacity for the peak-sales season (e.g., in the US, between Thanksgiving and Christmas). For information and connectivity businesses, global and local events like Football (Soccer) World Cup, Olympic Games, elections, etc. will likely be among the defining factors.

However, Peak Load has been one of the more popular techniques in performance analysis and capacity planning. Peak load is an indirect proxy for the business metric, but if used correctly, it can give us a good idea of how the system behaves and how much bandwidth will be needed next time a high-demand event happens.

For Performance Analytics

It makes perfect sense for the performance analyst to use the system's response to busy-hour or busy-minute demand in measuring whether the system is well-behaved (top-down approach):

(i) it automatically accounts for multiplexing and all other interactions.
(ii) A drilldown allows the analyst find which applications or IT elements have been hit the most.

However, it also has disadvantages, even with the top-down approach:

(i) We do not know that the peak-hour or peak-minute KPI value (traffic / CPU load / memory usage) was not an outlier.
(ii) We cannot possibly see quantile compression and have no way of knowing whether the system is getting congested ([GILG2015]).
(iii) When reporting, to get the 95th percentile of historical KPI, we need at least 3 weeks of daily data.

For Capacity Planning

For capacity planning, things get even more complicated. Daily busy-hour or busy-minute KPI has certain advantages:

(i) It provides a time series of demand data that can be used to forecast the future demand.
(ii) It is very useful with the bottom-up approach.

However, for capacity planning too, this method has serious drawbacks:

(i) If used with Top-Down approach:
    a. This measure diminishes the impact of services / applications / IT elements whose KPI values were low during the peak time interval.
    b. Peak-Load collection dampens underlying trends.
    c. Forecast will disregard services / applications / IT elements where the KPI is growing faster than for others.
    d. It levels out the trend (Figures 4 a, b)
(ii) If used with Bottom-Up approach forecasting approach:

a. If we use each element’s busy-hour / busy-minute data, we ignore the multiplexing and will overestimate the demand.
b. If we use each element’s KPI during the overall high-KPI time slot, and an element is not always there (e.g., a batch service only runs on Mondays and Wednesdays for one hour), we run the risk of misguiding ourselves into thinking that an element does not really exist.

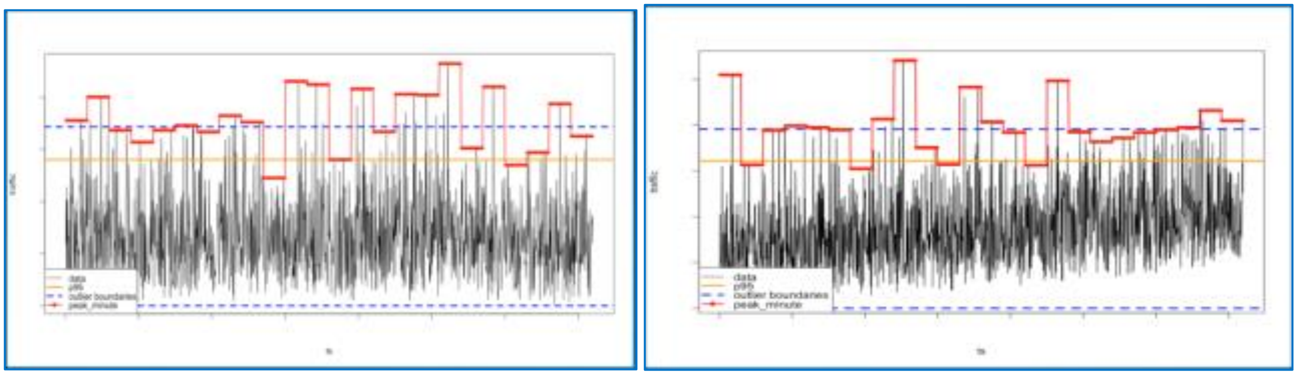


Figure 4: Red line corresponds to the busy-minute of the day.
(a) Untrended KPI (b) Trended KPI

95th Percentile

The 95th percentile, or p95, has been in use by capacity planners and performance analysts for a long time.

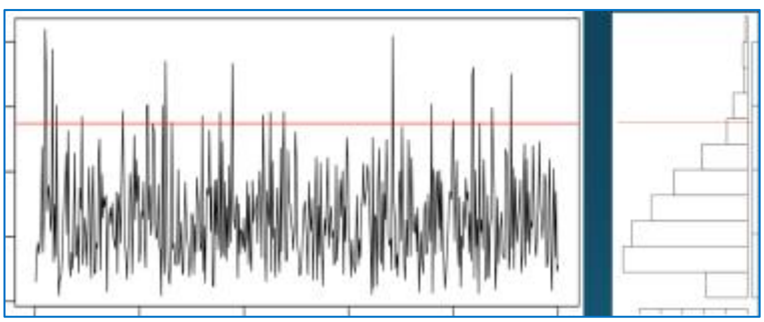


Figure 5: p95 Illustration

Like all other measures, it has its advantages and disadvantages:

Advantages

(i) Shape of the distribution does not matter.
  a. Even if mean and variance are not defined, percentiles are always there.
  b. Skew does not matter.
(ii) Outliers do not matter.
(iii) Only 5% of the data points will cause alerts in monitoring.
(iv) Math for predictive analytics is all worked out (quantile regression).

Disadvantages

(i) Percentiles are NOT additive:

a. $\sum p95\,(x_i) > p95(\sum x_i)$

(ii) We do not know which of the 5% of the data points above p95 are outliers.

Outlier Boundary

A lot can be, and has been, said in the literature about the art and the science of outlier detection. From the Six-Sigma-based methodology of using a given percentile as an outlier to Tukey's method of marking outliers, the jury is still out on what should be considered the definitive method of outlier detection.

Regardless of the method, the outlier boundary is important for different reasons depending on the task we are working on.

Performance Analysis and Monitoring

For performance analysis and monitoring, we want to detect anomalies and change points, learn what happened to the system during these events, and include these points into the data set, along with an annotation describing how it had happened before, why, and what we learned then.

Capacity Planning

From the capacity planning perspective, outliers represent events that were not expected. We cannot be expected to size the system for outliers – unless the KPI distribution is such that the outlier boundary is so low that a significant portion of the data are outliers.

The preferred nonparametric Tukey's outlier detection method has been visualized in R and python languages via the boxplot method (Figure 6). It involves very simple algebraic operations:

1. Identify the Quartiles
2. Compute the Inter-Quartile Range:

$$IQR = p75 - p25$$

3. Compute the High and Low Outlier Boundaries:

$$UpperBound = p75 + \beta * IQR$$
$$LowerBound = p25 - \beta * IQR$$

where the default value of the parameter $\beta = 1.5$

Outlier Boundary vs. Percentile

Figure 6a illustrates a scenario where the data are slightly right-skewed. In this case, the outlier boundary (the dotted blue line) is above the p95 (the orange solid line). This is a well-behaved system, and it is safe to use the outlier boundary for both performance monitoring and capacity planning: we can size the system for its outlier boundary.

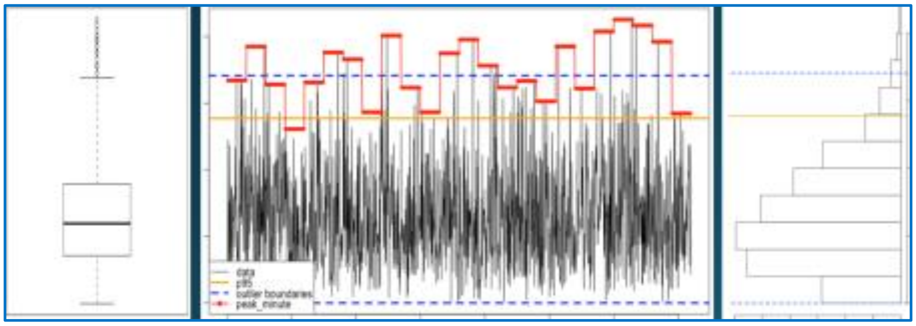


Figure 6a: an illustration of Tukey's outlier detection method.

In Figure 6 (a, b), outlier boundaries are marked by dotted blue lines; the orange line corresponds to p95; the red line is the busy-minute data

When the KPI distribution is similar to Figure 6a (in this example, the $skew = 0.13$), it is very sensible to use Tukey's outlier boundaries ("fences") as not only dynamic thresholds for performance monitoring, but also as the benchmark for capacity planning.

However, it is not unheard of to see in a network user-facing traffic with very long heavy tails (e.g., Figure 6b)

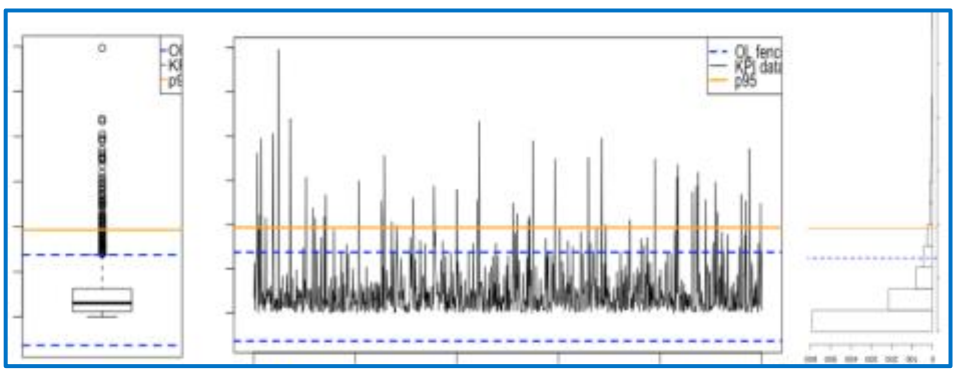


Figure 6b: high outlier boundary (blue dotted line) is below p95 (orange solid line)

The example is Figure 6b illustrates a more asymmetrical distribution. Here $skew = 0.26$, and the difference in skew was enough to render outlier boundary computed with $\beta = 1.5$ irrelevant.

Some work in the direction of smart calculation of the parameter β has been done in relatively recent years (e.g., [SSEO2006]; [VNDR2004]; etc.); however, these studies are for the most part distribution-specific and complicated. We are now working in this direction as well.

METRICS

A metric is what is important to the user, or has a significant effect on what is important to the user. In IT, we can boil it all down to three types of metric:

(i) Throughput;

(ii) Latency;
(iii) Delivery.

They are intertwined via queueing theory models, which have undergone a lot of development over the last century, pioneered by A.K. Erlang, adjusted for the IT systems by Jeffrey Buzen ([BUZN1971]), Neil Gunther([GUNT2009]), and implemented successfully by, e.g., Ferrandiz & Gilgur [FERR2012]. In addition, Bonald & Roberts[BNLD2012] and Nikolaidis et al [NKLD2013] proposed using the Erlang formulae to model the operation of complex networking environments, including the Internet.

A Note on Network-Specific Metrics

With regard to network operation, it is important to note that when it comes to performance analysis, it is more meaningful to measure packets per second, for the following reasons:

- The unit of work for a network is a packet.
- TCP sends (and blocks, and re-sends) packets, not bits.
- TCP sends packets of different sizes, from pings, to handshakes, to payloads. While we may be able to say that packets are being sent in a Poisson process, the additional uncertainty from the variety of packet sizes ensures that the old adage "Network traffic is spiky" – if measured in bits per second – is true.

As a result, situations like illustrated in Figure 7 are not uncommon.

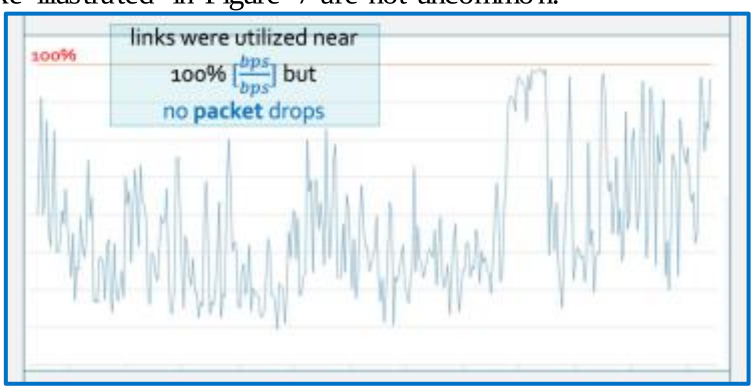


Figure 7: Near-100% utilized link experienced no packet drops (an illustration)

For capacity planning, on the other hand, because bandwidth is defined by the physical properties of the fiber or radio channel, which boil down to information theory, bits per second became the standard metric.

TRENDS AND CORRELATIONS

Trends in Performance Monitoring and Capacity Planning

Gilgur et. al. [GILG2015] discuss extensively the issues that arise when we use the wrong assumptions with the data, as well as the dangers of forcing the data into the technique, instead of fitting the technique to the data. The authors also cover some viable solutions to these problems. Other publications on the topic include, but are not limited to, e.g., [TRBN2006], and others. An interesting approach is also proposed by Papagiannaki et al ([PTZD2003]). The authors apply a time-series forecasting technique (ARIMA) to separately forecast upper-bound, midline, and lower-bound values of their data sets.

The practical recipe for basic capacity planning that has over the years been derived from these discussions can be summarized in Figure 8.

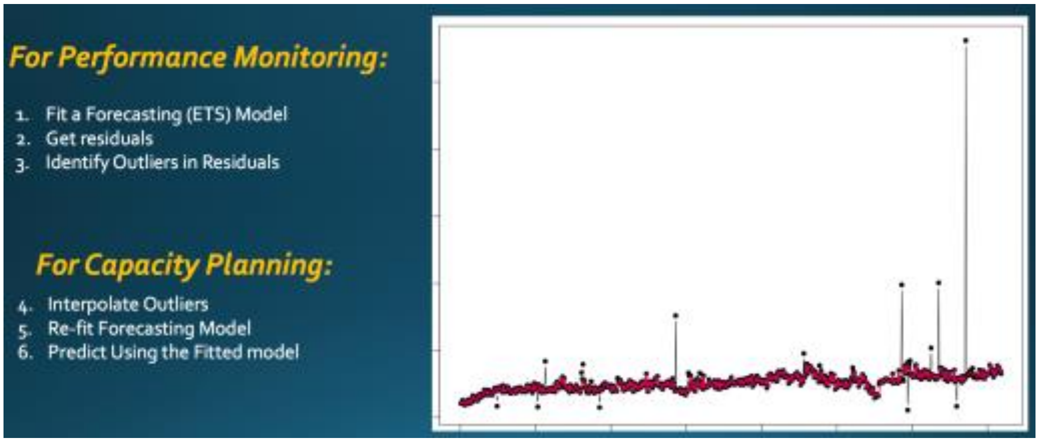


Figure 8: Using Time Series Analysis to identify outliers in trended data.

Accounting for outliers is a somewhat separate topic; usually an outlier indicates an external interaction. The right approach is to find external and internal factors that control outliers and predict the metric as a function of these factors (see discussion below).

Correlated Trends in Performance Monitoring and Capacity Planning

For correlated time series, things can get very complicated very fast. A robust solution has been proposed (and implemented) by Gilgur, Ferrandiz, and Beason [GILG2012]: if we know the causality direction – e.g., which business metric(s) drives which KPI(s) – we can fit regression and apply time-series analysis to its residuals.

For Capacity Planning, we can use this approach to predict the behavior of the KPI at any time in the future, as long as the forecast of the business metric is known. For example, power consumption is known to be a predictor of network backbone traffic, whereas number of concurrent users is a predictor of edge traffic.

For Performance Monitoring, this approach can be adapted to Predictive SPC [GILG2014]: predicting KPI values using the same model applied to baseline and to new data and measuring the deviation allows us to holistically monitor the system's performance and alert proactively to subtle changes in the system's behavior: e.g., instead of change in an application's latency, we can alert the user to a significant change in its sensitivity to the number of concurrent users.

APPLICATION: A HYBRID APPROACH TO CAPACITY PLANNING

As we discussed earlier in this paper, the capacity planner sometimes has a long run of top-level data (e.g., busy-hour; daily peaks; hourly averages; etc.), but a very short (e.g., a few months) set of higher-granularity data. Yet we have to plan capacity for many years ahead. One way to deal with it is described below using a hypothetical example.

Data

Network Traffic:

- N years of daily busy-hour data $(1\ value * each\ day)$ used in monitoring.
- $n << 12 * N$ months of high-granularity $(every\ minute * \ each\ service * each\ flow)$ data.

Question

What will each service's demand be within each flow N years from now?

Solution

General Considerations:

Using a pure bottom-up approach $(service \rightarrow flow \rightarrow overall)$ in forecasting this traffic into a long-range horizon is wrong: there is not enough service-level data. A hybrid approach (e.g., bottom-up for flows and top-down for services) will work here.

Bottom-Up: Flows

If flows are defined as a (source, destination, QoS) tuple (service name is not part of flow definition), then the bottom-up approach for flows is very straight-forward: assuming mutual independence of flows and their observable independence ([BUZN2015]) from services within each flow, we can apply time series analysis as outlined in Figure 8 to produce traffic forecast as shown in Figure 9.

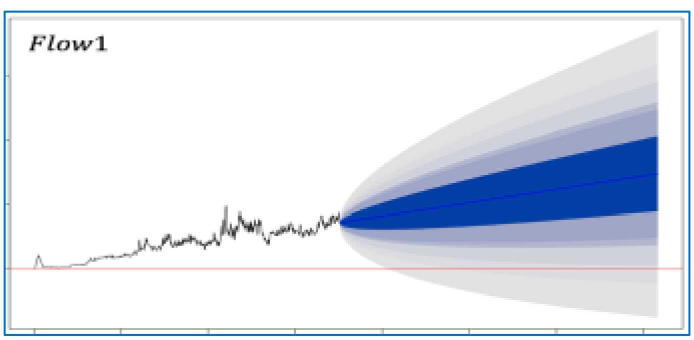


Figure 9: Flow Forecast

Repeating the process for each flow gives us a set of range forecasts, after which we can apply Monte-Carlo or other techniques to get the overall range forecast at any desired aggregation level.

However, this will tell us nothing about each services' demand trends.

Top-Down: Services Within Each Flow

The idea is to take advantage of the fact that the time series of service demand is too short to have developed a meaningful trend. As long as we can extract enough information from the per-service data to build distributions (empirical cumulative distribution functions, or ECDFs) of service demands within the flows.

From these distributions, the weights of each percentile of the ECDF are simple enough to compute, and applying these weights to the flow forecast – e.g., using Monte-Carlo – provides forecast for each service in each flow. The high-level view of such algorithm is presented in Figure 10.

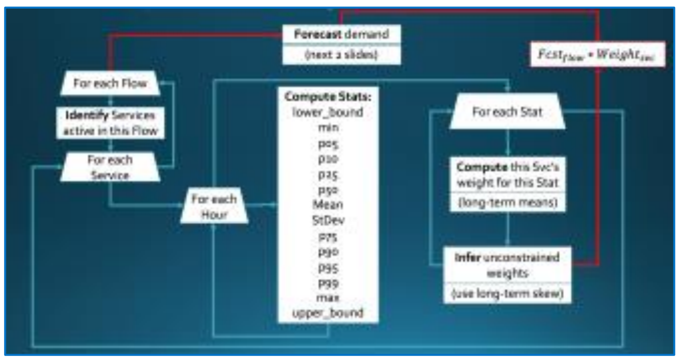


Figure 10: Hybrid Forecasting Methodology

Finishing touches

From here, we can aggregate service traffic forecasts; correct (adjust) them by building regression models to business metrics and other KPIs; establish Predictive SPC, etc.

CONCLUSIONS

Queueing theory, via Erlang and Erlang-style (e.g., [FERR2014], [NKLD2013]) models, or via Monte-Carlo simulations when Erlang model assumptions are not true (see, e.g., [BUZN2015] for more detail), holistically bridges the gap between performance and capacity.

However, when it comes to data requirements, we need to be very careful. Depending on whether the goal is to monitor and analyze performance or to analyze and predict demand for capacity, approaches to data collection, aggregation, and cleanup are very different, and while there is no one-size-fits-all approach, we have outlined in this paper the challenges and the solutions for both tasks.

REFERENCES

1. [BUZN2015] Jeffrey P. Buzen (2015) Rethinking Randomness. CreateSpace/Amazon, ISBN: 978-1-508-43598-3.
2. [GILG2015] Gilgur, A., Gunn, S., Browning, D., et. al. (2015) Percentile-Based Approach to Forecasting Workload Growth. Proceedings of the Performance and Capacity International conference by Computer Measurement Group, Inc. (CMG'15). San Antonio, TX, November 2015.
3. [PTZD2003] Papagiannaki, K., Taft, N., Zhang, Z.-L., Diot, C. (2003). Long-Term Forecasting of Internet Backbone Traffic: Observations and Initial Models. IEEE INFOCOM, 2003.
4. [GILG2012] Gilgur, A., Ferrandiz, J., Beason, M. (2012) Time Series: Forecasting and Regression: 'And' or 'Or'? 38th Annual International Conference of the Computer Measurement Group, Inc. Las Vegas, NV, December 2012.
5. [SSEO2006] Songwoo Seo (2006) A Review and Comparison of Methods for Detecting Outliers in Univariate Data Sets – Master's Thesis. University of Pittsburgh
6. [IGLE2001] Iglewitz, B., Banerjee, S., A Simple Univariate Outlier Identification Procedure. Proceedings of the Annual Meeting of the American Statistical Association. 2001
7. [VNDR2004] Vandervieren, E., Hubert, M. (2004) An Adjusted Boxplot for Skewed Distributions. Proceedings of COMPSTAT'2004 Symposium.
8. [BUZN1971] Jeffrey P. Buzen (1971) Analysis of System Bottlenecks Using a Queueing Network Model. ACM-SIGOPS Workshop on System Performance Evaluation. ACM, New York, 1971
9. [GILG2014] Gilgur, A., Ferrandiz, J. (2014) Predictive SPC. Talk presented at the Second Annual ASA Conference for Statistical Practice (CSP 2014). Tampa, FL, February 2014.
10. [GUNT2009] Mind Your Knees and Queues - Gunther. MeasureIT, Issue 62, 2009
11. [BNLD2012] Bonald, T., Roberts, J.W. (2012) Internet and the Erlang Formula – ACM SIGCOMM, 2012
12. [FERR2012] Ferrandiz, J., Gilgur, A. (2012) Level of Service Based Capacity Planning – Proceedings of the Performance and Capacity International Conference by the Computer Measurement Group (CMG'12)
13. [NKLD2013] Nikolaidis, A., Asselin, S., Auster, M., Bragg, N (2013) Capacity Planning for Dynamic Inter-Data Center Networking via Erlang Modeling. IET Conference Proceedings. Stevenage: The Institution of Engineering & Technology. (Sep 22, 2013)
14. [TRBN2006] Trubin, I. (2006) System Management by Exception. Presented at the Annual International Conference of the Computer Measurement Group (CMG 2006). Reno, NV. December 2006.
15. [FERR2014] Ferrandiz, J., Gilgur, A. (2014). Capacity Planning for QoS. Journal of Computer Resource Management. Issue 135 (Winter 2014). pp. 15-24