perfSONAR Data Representation and Analysis

# Abstract

The internet is now a basic service similar to water and electricity. At the very core of the internet is the “computer network” consisting of routers and switchers. Billions of dollars are potentially lost if the network is down for a day. For just Google company, almost $100 million can be lost from advertisements if it stops operating for a day [5]. Computer network is an important infrastructure for commerce, defense, social networks, media and etc.  
Thus, it is essential to monitor and act proactively to ensure the network is in operation. Rather than just looking at current traffic, it is also to predict the network traffic. So one of the way is to use an automated prediction/forecasting algorithm to predict how the network should look like. Holt-Winters is chosen to be the algorithm with its capability to perform exponential smoothing.

In this project, I have worked with the network engineer of Singapore advanced research and education network(SingAREN) to develop a tool to predict the network traffic. The tool gets information from the API and displayed it on a table and graph. Similar information is grouped together. Network engineers can use the provided graph implementation and prediction to make informed choices based on the data analyzed and represented. Error information are shown to network engineers with three statuses (Critical, Warning, Minor). As data from the network can be huge and sieving through the data can be a huge hassle, with the new information in hand, they can create faster and possibly better solutions for the network through the help of the project.

Table of Contents

[Abstract 1](#_Toc477437936)

[Background 2](#_Toc477437937)

[Acknowledgements 3](#_Toc477437938)

[1. Introduction 3](#_Toc477437939)

[1.1 Introduction to perfSONAR 3](#_Toc477437940)

[1.2 Aims of the project 3](#_Toc477437941)

[1.3 Scope 3](#_Toc477437942)

[2. Requirements 4](#_Toc477437943)

[2.1. Functional Requirements 4](#_Toc477437944)

[One way delay 4](#_Toc477437945)

[Bandwidth 4](#_Toc477437946)

[2.2 Non-Functional Requirements 4](#_Toc477437947)

[Performance Requirements 4](#_Toc477437948)

[Usability Requirements 4](#_Toc477437949)

[Reliability Requirements 4](#_Toc477437950)

[Supportability and Maintainability Requirements 5](#_Toc477437951)

[2.3 Interface Requirements 5](#_Toc477437952)

[User Interface 5](#_Toc477437953)

[Hardware Interfaces 5](#_Toc477437954)

[Software Interfaces 5](#_Toc477437955)

[2.4 Software Environment 5](#_Toc477437956)

[3. Documentation 6](#_Toc477437957)

[3.1 Use Case 6](#_Toc477437958)

[One way delay 6](#_Toc477437959)

[Bandwidth 7](#_Toc477437960)

[3.2 Sequence Diagram 8](#_Toc477437961)

[3.3 Quality Management 9](#_Toc477437962)

[3.4 Architecture 10](#_Toc477437963)

[General Services 10](#_Toc477437964)

[3.5 Testing 11](#_Toc477437965)

[Black Box Testing 11](#_Toc477437966)

[White Box Testing 12](#_Toc477437967)

[4. Front End Development 13](#_Toc477437968)

[AngularJS 13](#_Toc477437969)

[Why choose AngularJS? 13](#_Toc477437970)

[ZingChart 13](#_Toc477437971)

[5. Deployment 13](#_Toc477437972)

[6. Description of individual pages 14](#_Toc477437973)

[6.1 One way Delay Page 14](#_Toc477437974)

[7. Conclusion 15](#_Toc477437975)

[8. Data Dictionary 15](#_Toc477437976)

# Background

A computer network is a telecommunications network which allows nodes to share resources among each other. Networked computing devices exchange data using a link and the connections made can either be cabled or wireless. The Internet is a well-known computer network. In a network, there are hosts and receivers. A host can also be a receiver and vice versa.   
A network such as Internet is extremely important as it supports a huge amount of applications and services such as YouTube, Facebook, Twitter and etc.

# Acknowledgements

I would like to thank Prof Francis Lee and Simon from SingAREN for assisting and guiding me during the project development. Also, a huge thanks to Nazri for developing the initial project and helping to get the project going.

# Introduction

Network monitor is the use of a system that constantly monitors a computer network for slow or failing components and notifies the network administrator in case of outages. 1 It keeps track of information such as response time, uptime, availability and etc. The information is important to research disciplines as it provides valuable insight about a network which can be hard to keep track manually. Some examples of a network monitor include Wireshark and tcpdump.

perfSONAR is also a network monitor which can run network tests across multiple domains. It is based on a service oriented network architecture. This architecture “proposes to use independent pieces of software called “services” which can be orchestrated to collaborate in order to reach a common goal.” 2 A service oriented architecture as compared to other architectures have several advantages which include independent services which prevents software from being difficult to maintain. Also, services can be dropped or added which increases the program’s flexibility and robustness. It provides more functions than a generic network monitor with examples including tracepath, Iperf and ttcp.

Although perfSONAR can be used to generate results, not much of the results generated are being analyzed. Meaningful analysis can be done to filter and sort the results accordingly. Graphs and bar charts can be plotted against results to show relationships between different attributes. It helps to visualize and thus reinforcing the results.

## 1.1 Introduction to perfSONAR

PerfSONAR (Performance Service Oriented Networking Monitoring Architecture) is an open sourced network measurement tool which is developed internationally. It has a wide range of features including network metrics to measure the performance between each node. The tools will perform tests at regular intervals and results are stored in the database which can be retrieved through an API. Thus, network engineers can retrieve valuable information about the network.

perfSONAR acts as a beacon that provides value to others around the world by simply allowing others to test the node.3 It can also be used to create a grouping of test known as a mesh. A configuration file is shared among the group so that it can be connected together. 3

## 1.2 Aims of the project

The aim of this project was to make data tracking simpler and more efficient with perfSONAR. The main features that were done on this project were populating the data that perfSONAR generated on a suitable medium and to analyze the data so that relationships between data can be drawn easily. Other features include tracking of a specific node and all of the information were collated and shown in a table form or graph form.

## 1.3 Scope

Retrieving the One Way Delay, Bandwidth and packet loss data to create visualizations and forecasting.

The project will retrieve the network measurements over the last twenty-four hours.

The project will inform users of packet loss reading and its metakey data if it exceeds the threshold which is defined as such: (Minor ,0.2) (Warning, 0.6) (Critical,0.8)

The project is a web based application to be run in a Javascript enabled browser.

The project will display the following types of visualizations

1. Line Graph representation highlighting One Way Delay(Minimum) and reverse traffic with Forecasting.
2. Line Graph representation highlighting Bandwidth(Throughput) and reverse traffic.
3. Area Graph representation highlighting Bandwidth(Throughput) Forecasting.

# 2. Requirements

## 2.1. Functional Requirements

Functional requirement defines a function of a system or its component that defines what a system is supposed to achieve.

### One way delay

The project will retrieve information from API to AngularJS to process and display on webpage.

Displaying information such as Standard deviation, metakey, minimum, maximum and etc.

Displaying information such as packet Loss on a different request.

Processing HW Forecast on the particular node.

Displaying a plotted graph relating to the particular node and forecasting.

### Bandwidth

Retrieving information from API

Processing HW Forecast on the particular node.

Process HW Forecast with index of 0.85.0.9,1.1,1.15.

Displaying a plotted graph relating to the particular node and forecasting.

## 2.2 Non-Functional Requirements

Non-Functional requirement defines the criteria that can be used to judge the operation of a system, rather than specific behaviors.

### Performance Requirements

Response time of the system must be done within 1 second.

Maximum processing time must be done within 10 seconds.

### Usability Requirements

The system should be interactive. User should not have difficulty in using the application without prior training or guides required.

The system must provide user with informative feedbacks after each action is performed.

### Reliability Requirements

The system must be available at all time.

The system will ensure the data integrity even in the case of system failure.

In case of system failure, the project can be restored within 1 hour.

The system must accurately show the user of the details the user requested at the point in time.

### Supportability and Maintainability Requirements

The system is designed to be able to integrate with various platforms with web browser.

Maintenance must be done monthly to ensure the smooth operation of the project.

The project is designed in a way that any future enhancements can be made easily by adding a webpage in.

## 2.3 Interface Requirements

### User Interface

The Schneiderman’s Eight Golden Rules were followed to design the interface for the project to provide a good interaction interface for the users.

### Hardware Interfaces

There are no direct hardware interfaces for this project.

### Software Interfaces

This project communicates with perfSonar RESTful API to retrieve network information. Browsers such as Google Chrome are used.

## 2.4 Software Environment

The project was developed with python as the backend server. The backend server is used to provide the support like persistent storage and to handle business logic. Front end and back end server interacts through the help of AJAX which allows web applications to be interactive as refreshing the page is not needed. By using python, OS such as Linux and Windows can deploy the server, making it extremely handy and convenient to port over between OS. Flask is used as the backend library for python and it is a framework for web applications.

The front end was developed with AngularJS and operates on normal browsers which supports JavaScript. More will be discussed in the section on Front End development.

# Documentation

## 3.1 Use Case

### One way delay

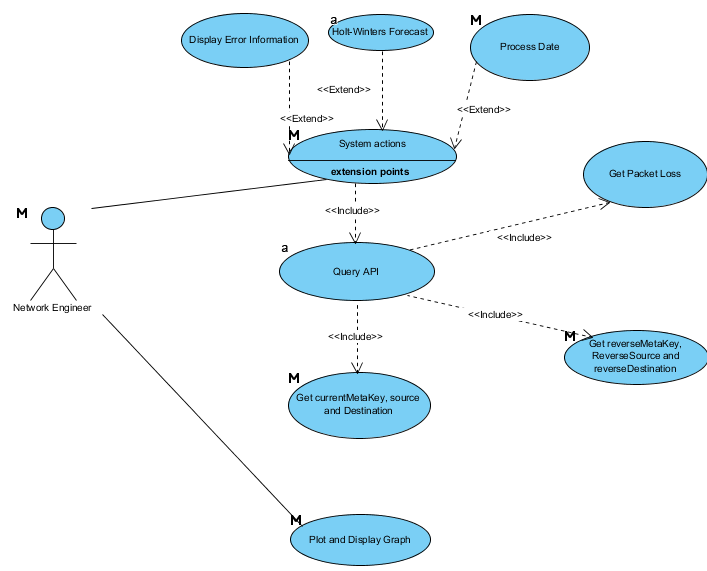


Figure 1. Use case of One way Delay Page

### Bandwidth

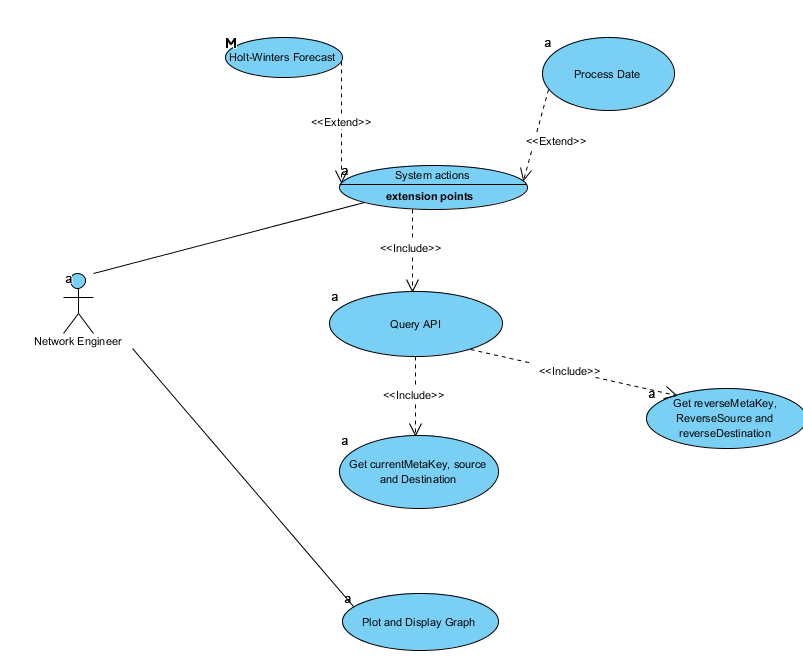


Figure 2. Use case of Bandwidth Page

## 3.2 Sequence Diagram

#### Bandwidth

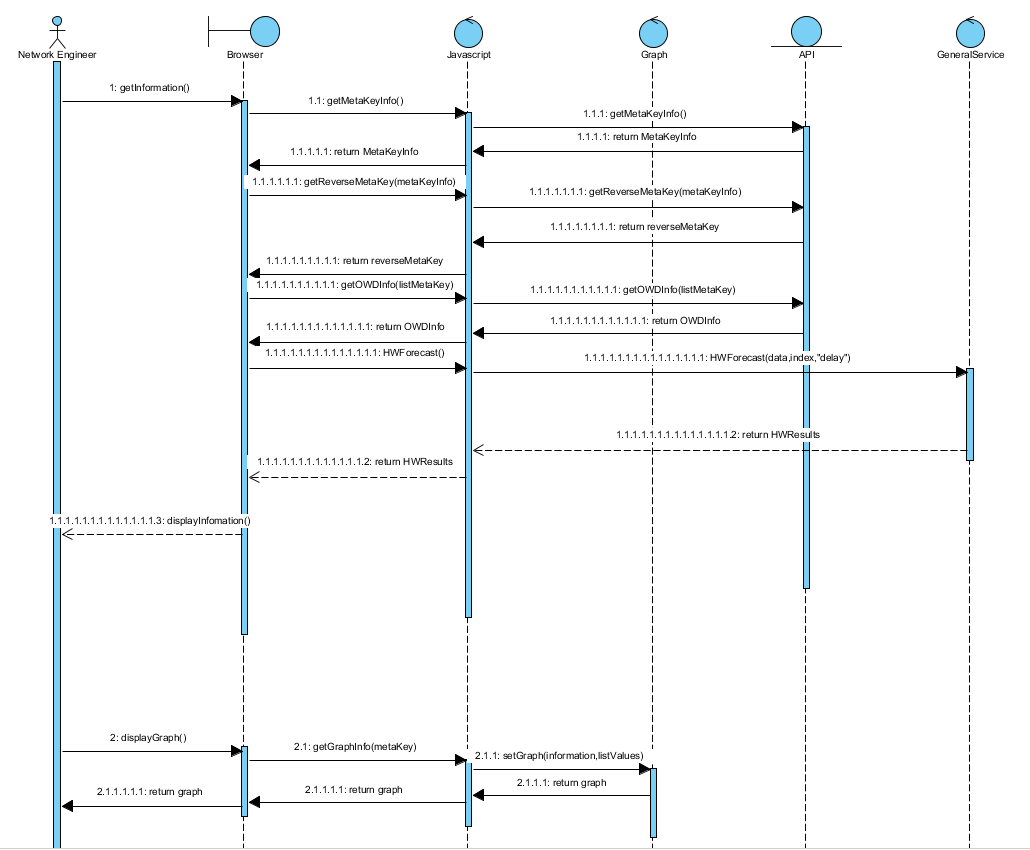


Figure 3. Sequence Diagram of Bandwidth

#### One way Delay

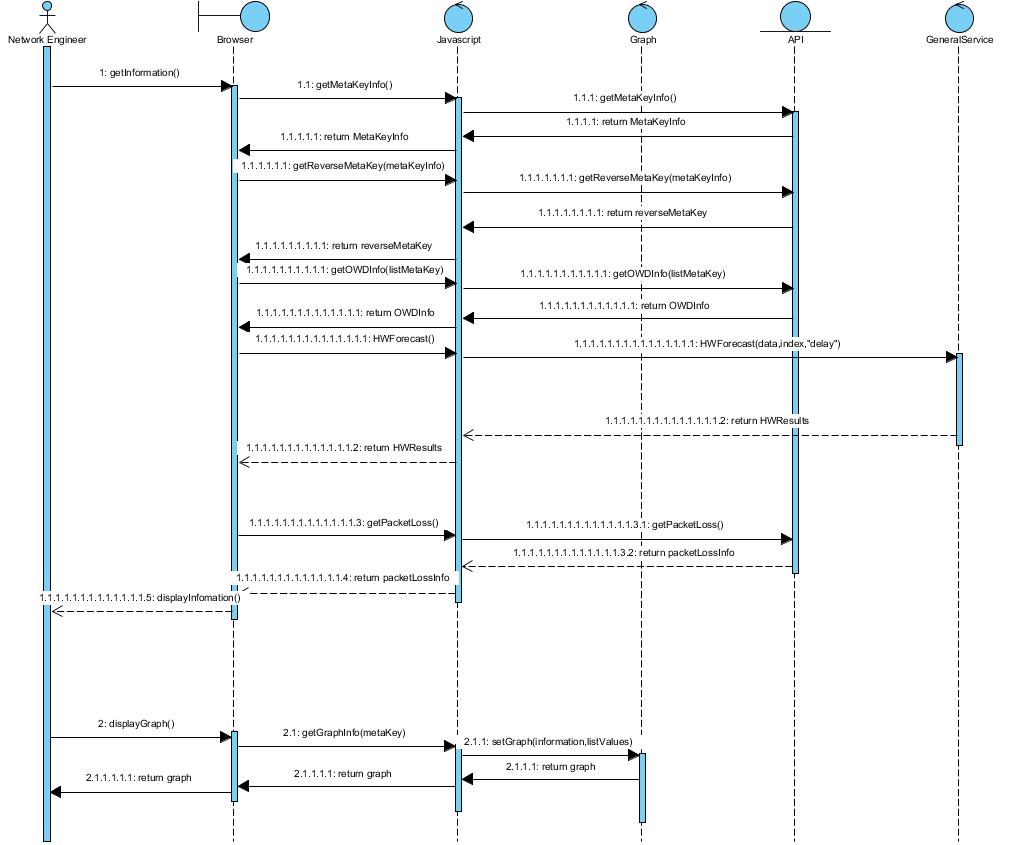


Figure 4. Sequence Diagram of One Way Delay.

## Quality Management

Flexibility

More features could be added into the project and include more functions such as statistics or aggregation for the information displayed. It is important that the project is able to add on modules easily. By adopting the services method, other modules could easily extend the services without affecting other existing components.

Availability

As this system has the capability to monitor the network, it is important that it can support the requests that the user wants and not crash. Users can check the network status immediately and get the error nodes out immediately so that the least impact will be made to the network.

## Architecture

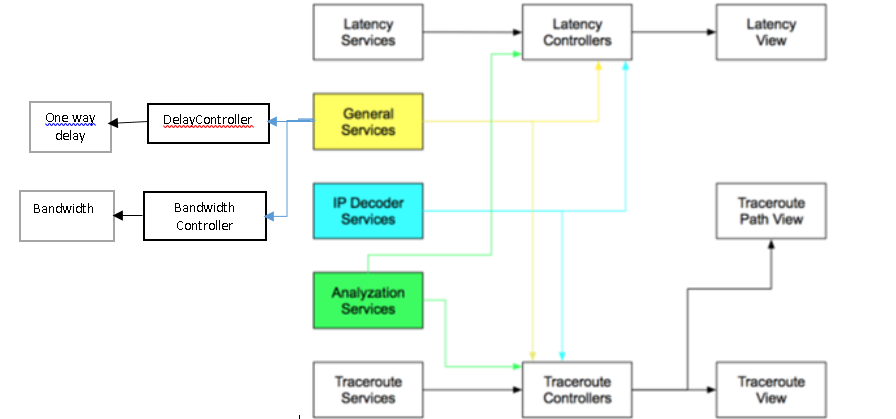


Figure 5. Software Architecture. Credits to Nazri.

### General Services

It contains all the services that are for all the modules in the application. It is used by all the modules except Traceroute view.

UnixTimeConverterService

A service that converts the time into Unix format.

CurrentTimeUnixService

A service that converts the current time into Unix format

HostService

A service that returns the IP address of the centralized Esmond.

UniqueArrayService

A service that takes in an array and returns back an array with only unique values. This is used determine unique nodes to add into the visualization.

Holt-Winters  
A service that takes in a set of values and alpha and returns a set of forecasted value based on the alpha and default set.

Delay Controller

It handles the information retrieval from API to the browser page. Also, it uses the Holt-Winters service from General service to forecast the graph set to pass to ZingGraph. Information manipulation are also done in the controller which is used to determine whether packet loss is above the threshold

Bandwidth Controller

It handles the information retrieval from API to browser page. Also, it uses the Holt-Winters service from General service to forecast the graph set to pass to ZingGraph. Four sets of values are then returned back to ZingGraph for Area Graph plotting.

## 3.5 Testing

Unit testing is a software testing method by which individual units of source code, sets of one or more computer program modules together with associated control data, usage procedures, and operating procedures, are tested to determine whether they are fit for use. It can isolate each part of the program and show that the individual parts are correct.

### Black Box Testing

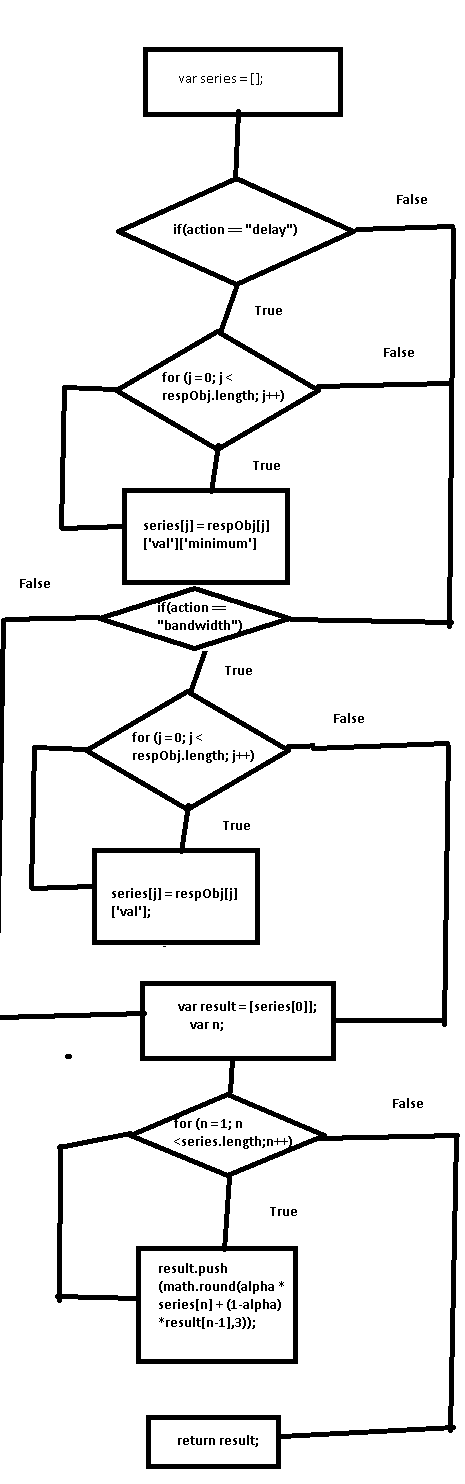
The test revolves around functional requirements

The test uses Boundary Value Analysis which is a software test design technique that involves determination of boundaries for input values and selecting values that are at the boundaries and just inside/ outside of the boundaries as test data.[6]

Packet loss values are set at   
Minor value - 0.2  
Warning value - 0.6   
Critical value - 0.8  
Set of test values at 0.199,0.201,0.599,0.601,0.799,0.801.

Test results  
0.199 – Nothing  
0.201 – Minor  
0.599 – Minor  
0.601 – Warning  
0.799 – Warning  
0.801 - Critical

### White Box Testing

Cyclomatic complexity = 5 + 1 = 6  
Method HWFunction(respObj,alpha,action)  


# Front End Development

Based on the previous development, the project is a web based application with emphasis on visualization of graphs and to inform on network with high error/ packet loss rate. AngularJS is a JavaScript framework used for the client side browser to interact with the API.

## AngularJS

AngularJS is a framework for web application which lets the user use HTML as the template language and extend HTML's syntax to express the application's components. AngularJS's data binding and dependency injection eliminate much of the code the user would otherwise have to write. And it all happens within the browser, making it an ideal partner with any server technology. [7]

### Why choose AngularJS?

AngularJS is built around the belief that declarative code is better than imperative when it comes to building UIs and wiring software components together, while imperative code is excellent for expressing business logic. [7]  
AngularJS decouples DOM from app logic which improves the testability of the code and it decouples the client side of the app from the server side which allows reuse for both sides of the project.

AngularJS also solves some problems such as registering callbacks which clutters the code, making it hard to debug. It decreases the amount of coding and it makes it easier to see how the application works. Manipulating HTML DOM is error-prone and problematic. By declaratively describing how the UI should change as your application state changes, the app is freed from low-level DOM manipulation tasks. Most applications written with AngularJS never have to programmatically manipulate the DOM.  With AngularJS, bootstrapping can be done using services, which are auto-injected into the application.

## ZingChart

ZingChart is a JavaScript chart library which can be customized to a user’s needs. The reason to use ZingChart instead of other chart libraries is because it provides the API to customize the graph such that visualizations are clear-cut and easy to see. Also, it is free to use.

# Deployment

The operating system for deployment can be in Linux, Windows and OS X since all of them supports Python and Nginx, which is required by the project in order to operate.

A deployment server has been setup by SingAREN and is in Ubuntu 16.04.1 LTS and is a virtualized machine.

To deploy the project, several dependencies and programs must be downloaded and installed. The most important program is Node.js. Then install npm via Node.js. Once both are installed, we can now install bower and gulp by using the command in Node.js:

npm install bower -g | npm install gulp -g

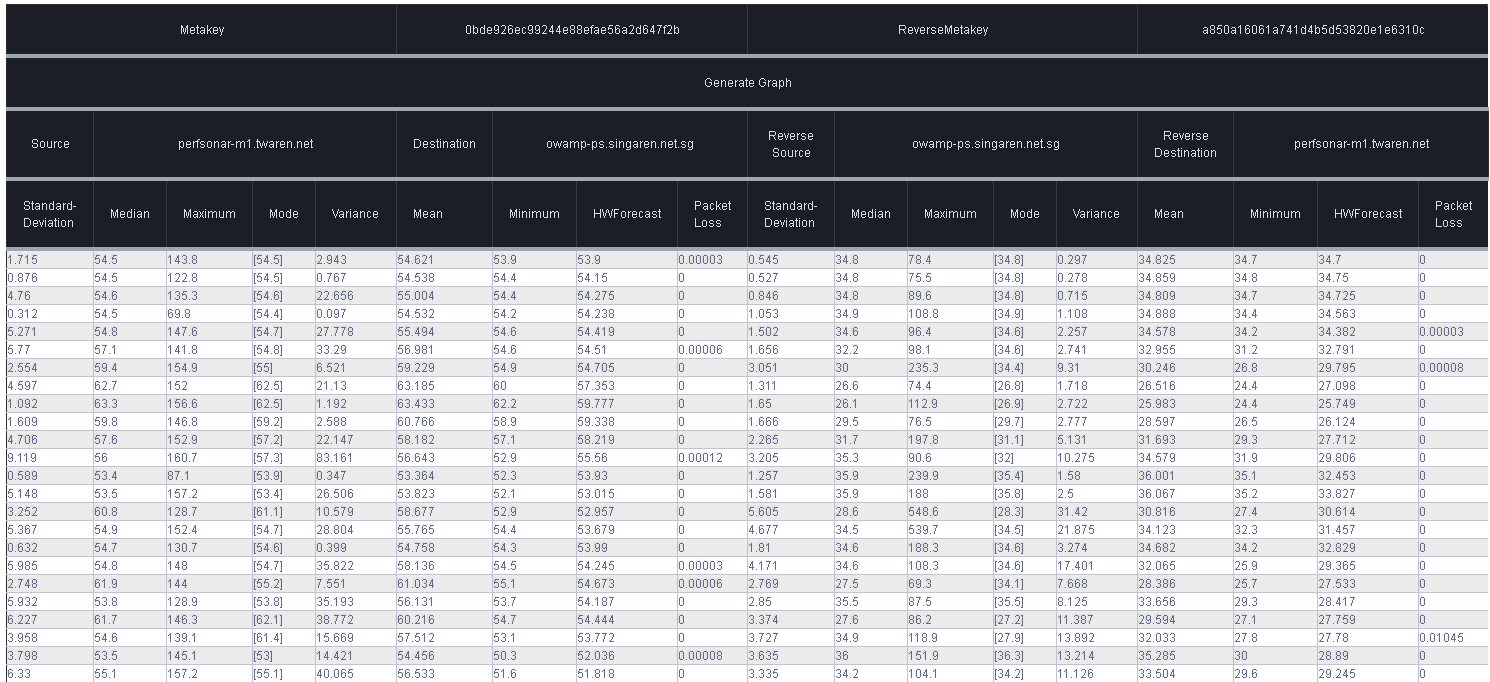
Also, ensure that the path variables in the System path are correct.

As for python version, the working version is Python 2 and Pip is required to install more dependencies.

The following dependencies are: python-dev, nginx and python-pip.

# Description of individual pages

## 6.1 One way Delay Page

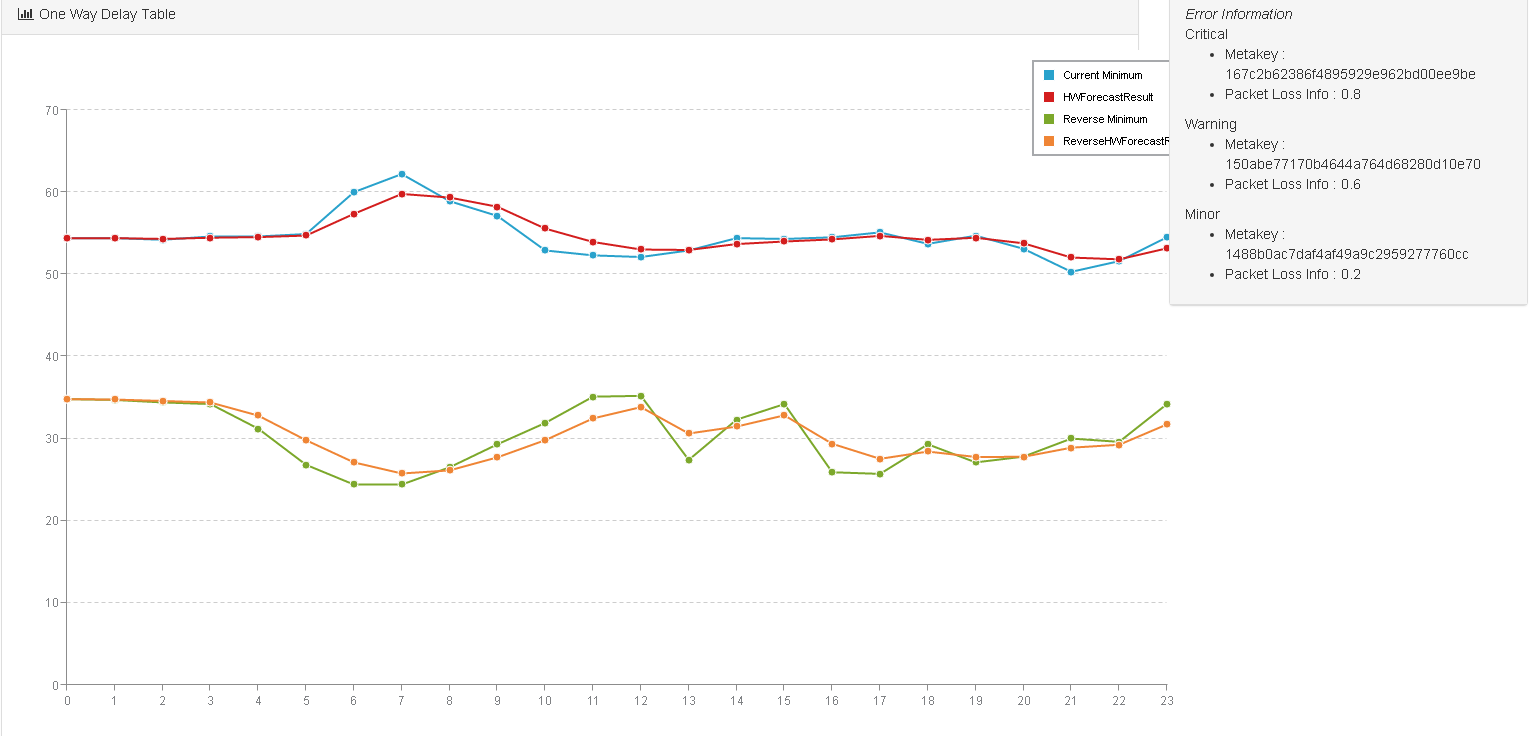


This is a sample entry on the one-way delay page. The metakey and reversemetakey is the unique identifier that allows the page to retrieve the particular node’s information. Source is the place where the data/information is coming from. Destination is the place where the data/information is going to. Reverse source and destination is the reverse way that the data/information is flowing. Information such as standard deviation, median,maximum,mode,variance,mean,minimum, HWForecast and packet loss are available for the network engineer.

HWForecast is predicted using the base value from minimum as given a delay from one place to another, the bottleneck is usually the minimum time that takes for the data/information to flow from one place to another.

Packet Loss is estimated to five decimal place. Any value that is more than five decimal values might be too insignificant to be considered.

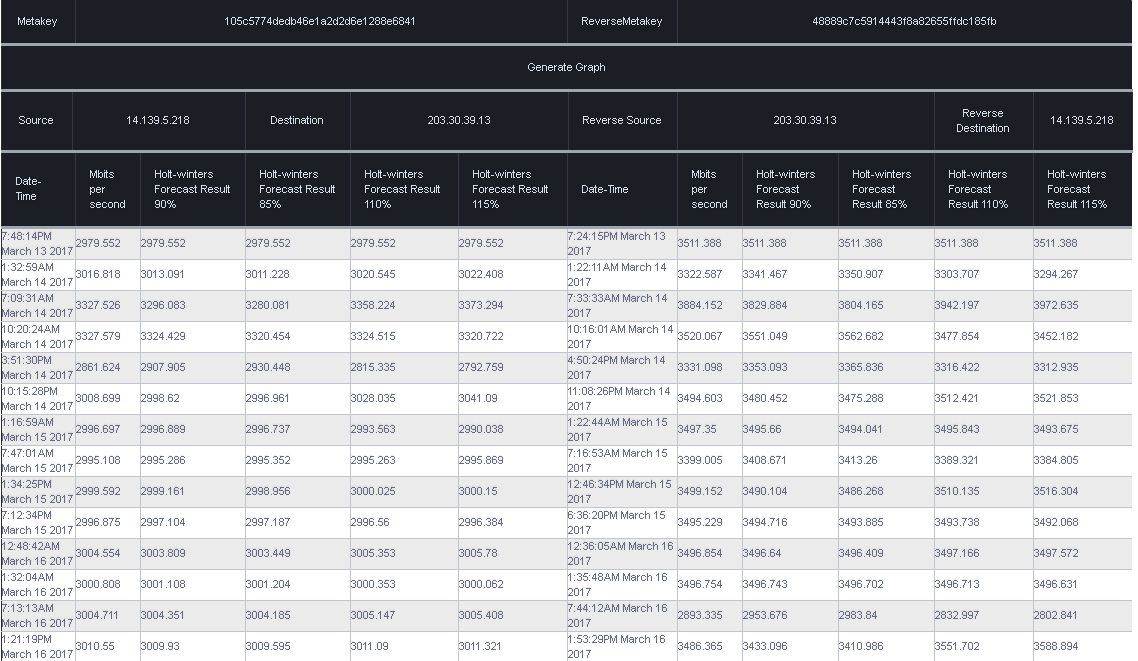
The network engineer can click on the generate graph to show the relevant information on a graph.



The graph will display the minimum and the HWForecast information. Based on this graph, the network engineer can know the trend of the network and debug subsequently if there’s a sudden dip on the graph.

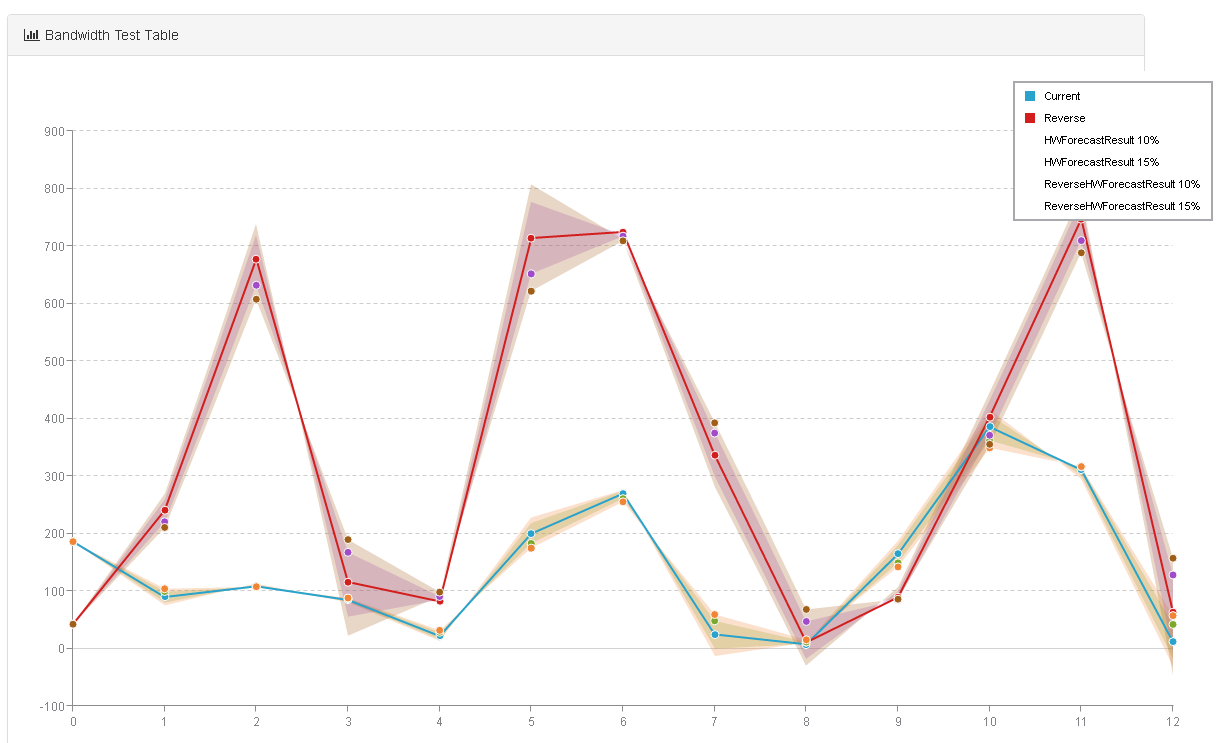
Error information shows the packet loss information which exceeds a certain threshold. A minor status is between 0.2-0.6 packet loss rate. A warning status is between 0.6-0.8 packet loss rate and Critical status is above 0.8 packet loss rate.

## 6.2 Bandwidth page



This is a sample entry on the bandwidth page. The metakey and reversemetakey is the unique identifier that allows the page to retrieve the particular node’s information. Source is the place where the data/information is coming from. Destination is the place where the data/information is going to. Reverse source and destination is the reverse way that the data/information is flowing. Information such as Date-Time, Mbps and HWForecasts are available for the network engineer.

The network engineer can click on the generate graph to show the relevant information on a graph.



For the bandwidth part, a range of 10% and 15% difference is set by forecasting. The network engineer can therefore easily tell if the bandwidth drops or rises outside the forecasting range and does the necessary adjustments to make it back into the range.

# Forecasting Algorithm

## Holt-Winters Forecasting

Forecasting can be used to reduce decision risk by supplying additional information about the possible outcomes.[6] As data is fed into the model, the model will mold the data into a similar shape as the model and that is the forecast used. Time-series forecasting assumes that there are random errors and a combination of patterns and the goal is to segregate the pattern and the error to provide a more accurate forecasting.

Exponential smoothing

Exponential smoothing is used to revise a forecast constantly to provide a more recent result. Weight is used and as the results get older, the weights in the forecast gets lesser and vice versa.

# Conclusion

Network is an important part of our daily life and network measurement tools are a must to ensure quality of service. Combined efforts from the tools and engineers will allow monitoring of network and analysis of performance to be simple.

PerfSONAR has a wide range of network features that measures the health of the network and the architecture of perfSONAR which is service oriented means that it has the ability to scale.

By having this project, the time taken to track down a specific node in the network drops tremendously. Also, difficult analysis that are done manually by plugging in data from perfSONAR is automated thus removing any source of human error in transferring the data and increasing work efficiency. “Pictures were remembered better than words overall and both semantic processing and intentional learning resulted in better recognition than non-semantic encoding.” 4

# Data Dictionary

|  |  |
| --- | --- |
| perfSONAR | Performance Service Oriented Networking Monitoring Architecture |
| SingAREN | Singapore advanced research and education network |
| Python | Python is a widely used high-level programming language |
| LTS | Long term support |
| Nginx | NGINX is a free, open-source, high-performance HTTP server and reverse proxy |
| Pip | pip is a package management system used to install and manage software packages written in Python |
| Cyclomatic complexity | measures the amount of decision logic in the program module |
| ZingGraph | Javascript charts library |

References:

1"Network monitoring," in Wikipedia, Wikimedia Foundation, 2016. [Online]. Available: https://en.wikipedia.org/wiki/Network\_monitoring. Accessed: Sep. 3, 2016.

2 " mnm-team.org," in PerfSONAR: A Service Oriented Architecture for Multi-domain Network Monitoring, 2005. [Online]. Available: http://mnm-team.org/pub/Publikationen/hbbd05/PDF-Version/hbbd05.pdf. Accessed: Sep. 3, 2016.

3 "perfSONAR How does it work?," 2016. [Online]. Available: http://www.perfsonar.net/about/how-does-it-work/. Accessed: Sep. 4, 2016.

4 C. L. Grady, A. R. McIntosh, N. M. Rajah, and F. I. M. Craik, "Neural correlates of the episodic encoding of pictures and words," Proceedings of the National Academy of Sciences, vol. 95, no. 5, pp. 2703–2708, Mar. 1998. [Online]. Available: http://www.pnas.org/content/95/5/2703.full. Accessed: Sep. 4, 2016.

[5] <https://thenextweb.com/insider/2012/01/18/how-much-would-facebook-google-or-twitter-lose-if-they-shut-down-for-one-day/>

[6]<https://labs.omniti.com/people/jesus/papers/holtwinters.pdf>

[7] <https://docs.angularjs.org/guide/introduction>