CS284 Design Specification: An Application for Real-time Network Visualization

Leah Chatkeonopadol: chatkeon@gmail.com | Michael Nekrasov: mikrasov@gmail.com

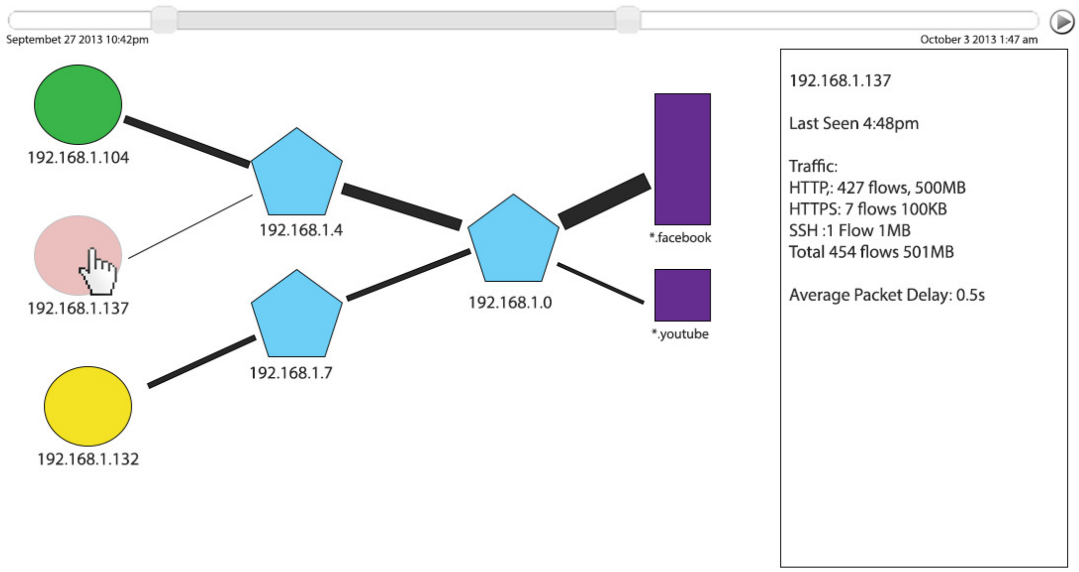
As an ever-growing number of devices (from cellphones to toasters) are becoming internet enabled, it becomes increasingly important to understand and manage their interaction on a network. This is especially true for developing regions where limited bandwidth necessitates careful resource allocation. Problems like ineffective caching and unfair network utilization greatly impact network performance and are of high concern to the administrators of these networks. A visual depiction of network state and potential problems can help administrators quickly identify and address these problems. Our project aims to tackle this problem by developing a real-time visualization system for monitoring network behavior.

Previous work done in network monitoring and real-time visualization includes systems such as VISUM [1], ReTiMon [2], and Gephi [3]. While these systems perform services similar to our project, they are all separate programs that require additional software and are designed for specific, limited platforms. In contrast, our system will be browser-based and thus extremely portable and accessible. We also emphasize different key characteristics of the analyzed data and propose some novel features, such as a “Tivo”-like slider for specifying the time range of displayed data.

The architecture of our system will consist of a parser and a user interface for visualization. Data, stored in pcap files, will be fed into the parser for processing, and its key characteristics will be saved in summary form for easy retrieval later. Then the analyzed data will be displayed to the user as described below.

The parser will be based on the Node.js node-pcap-parser library. For maximum usability and interoperability, the visualization will be done using HTML5 compliant javascript on Node.js. This will allow easy use of the tool without the installation of additional software. We will leverage existing javascript libraries such as “dagre” and “D3” to simplify coding and allow the application to grow upon established technologies.

The initial focus of our application is on flow analysis. We will display the clients on the network, the destinations (at least the most highly utilized ones), and the relative quality and utilization of each path. By hovering over particular nodes, the user will be given additional information detailing the breakdown of utilization including the protocol used. Hovering over a node will also reveal all associated paths. In addition, we will extract and display basic network metrics, such as protocol usage, throughput, and average delay. An example mock-up of the application is shown below.



The visualization will be time based. Users will be able to hit play and see the current state of the network (in 1 second increments). Users will also be able to use a slider to specify the range for which they want to see previous data. As clients stop utilizing the system their nodes will begin to fade from view. Nodes not utilized for the current window length will not be shown. Nodes that are experiencing problems (for example starvation or high latency) will be highlighted in red. Link thickness will show degree of utilization.

For this project we will focus on the visualization engine. The data collection problem will be handled in a later project. To evaluate our system we plan to use network traces collected in Macha, Zambia. We will use this data set to simulate real-time data flowing through our network. This data set exhibits many interesting properties like narrow bandwidth, multiple clients repeatedly accessing the same content, and high latency, which should test the ability of our system to highlight real-world network problems.

The main challenges we foresee in the implementation of this project are determining how to process the data efficiently and presenting analyzed information in a meaningful and timely manner. For a real-time system, we will need to process and display data at a rate faster than the network flow. This demands efficient algorithms for both analyzing data and rendering images. Other complications we may encounter include scalability issues and effective implementation of the slider functionality allowing users to view data from previous time periods.

We will test our system by processing data from Macha and comparing our results to the previous work done by [4, 5, 6]. If our system is able to reveal the same characteristics, this will verify that our work is capable of accurately visualizing real-world data. Should we be able to find other appropriate data sets, we will perform similar tests, wherein we process the data with our system and compare the results with previous analyses.

We plan to implement our system according to the following timetable.

- **Week 1 (10.14):** Create parser to read a pcap file and extract destination, source, packet size, and protocol.
- **Week 2 (10.21):** Create basic visual display with connected nodes and build framework for feeding data to the display.
- **Week 3 (10.28):** Add a slider and “TiVo”-like functionality so that the visualization can show previous data.
- **Week 4 (11.4):** Add object fade, color queues for problems, and line thickness for link utilization.
- **Week 5 (11.11):** Add additional network metrics and polish the interface.

References:

[1] C. Ho, K. Ramachandran, K. Almeroth, and E. Belding-Royer. A Scalable Framework for Wireless Network Monitoring. In *WMASH*, Philadelphia, PA, October 2004.

[2] S. Das and V. Kone. ReTiMon - A Real Time Network Monitor, unpublished.

[3] M. Bastian, S. Heymann, and M. Jacomy. Gephi: An Open Source Software for Exploring and Manipulating Networks. In *Proceedings of the Third International ICWSM Conference*, 2009.

[4] D. Johnson, E. Belding, K. Almeroth, and G. van Stam. Internet usage and performance analysis of a rural wireless network in Macha, Zambia. In *NSDR*, San Francisco, CA, 2010.

[5] D. Johnson, E. Belding, and G. van Stam. Network traffic locality in a rural African village. In *ICTD*, Atlanta, GA, 2012.

[6] V. Pejovic, D. Johnson, M. Zheleva, E. Belding, L. Parks, and G. van Stam. The Bandwidth Divide: Obstacles to Efficient Broadband Adoption in rural Sub-Saharan Africa. In *International Journal of Communication*, vol. 6, pp. 2467-2491, 2012.