Visualizing large-scale IP traffic flows

Florian Mansmann, Fabian Fischer, Daniel A. Keim, Stephen C. North

University of Konstanz, Germany
AT&T Research, USA
Email: {mansmann,fischerf,keim}@inf.uni-konstanz.de, north@research.att.com

Abstract

Hierarchical Network Maps are a scalable approach to the presentation of IP-related measurements on the global Internet. This study focuses on how to extend them for emphasizing the source destination relationship of network traffic aggregated on IP prefix, autonomous system, country, or continent. Edge bundles consisting of several spline curves visually group traffic that shares common ancestor nodes along the IP/AS hierarchy.

1 Introduction

Today, signature-based and anomaly-based intrusion detection is considered the state-of-the-art for network security. However, fine-tuning parameters and analyzing the output of these methods can be complex, tedious, and even impossible when done manually. If this situation was not challenging enough, current malware trends suggest an increase in security incidents for the foreseeable future. The health of the network infrastructure clearly depends on the effectiveness of both manual and automated methods to analyze, comprehend, and disseminate understanding of large network data sets.

Hierarchical Network Maps (HNMaps) are an approach to the presentation of IP-related measurements on the global Internet. They are based on a hierarchy ($prefix \rightarrow AS \rightarrow country \rightarrow continent$) on top of all Internet subnet prefixes and displayed using a space-filling visualization technique. This pixel-conservative approach is appropriate as display space is a scarce resource when displaying about 200,000 IP prefixes at once.

In previous work, we considered network statistics observed at a single vantage point (arriving at or leaving from a particular gateway) and displayed it either by its source or destination in the IP address space. In this study, we consider traffic being transferred through multiple routers, such as in service provider networks. The use of *edge bundles* enables visually displaying and detecting patterns through accumulative effects within nodes of the AS/IP hierarchy. The novelty of this approach lies in its capability to support the formation of a mental model that places each autonomous system or IP prefix on a map while linking nodes according to the traffic under consideration, and at the same time limiting visual clutter.

The rest of this paper is structured as follows: we briefly discuss the employed database technology to speed up the backend of our application and review related work. Our HNMap appraoch is then discussed and extended through so-called edge-bundles. Afterwards, we examine generation of random data and apply the presented methods before assessing the overall contribution.

2 Efficient querying of large IP-related data sets

To support visualization that is fast enough for interactive data exploration, we need not only to consider efficient rendering techniques, but also be aware of database technology as precondition for flexibility in querying different data sets as well as for speed. We therefore briefly regard the *multi-dimensional data model* ([13]) which stores large amounts of *facts* with associated numerical *measure* in *data cubes* that are particularly well-suited for data analysis (in contrast to storing of transactional data). Queries aggregate measure values over a range of dimensional values to provide results such as the number of security events aggregated on each AS in a given country (dimension IP hierarchy) at a certain day (dimension time). Figure 1 shows the user interface to specify the database query and visualization parameters.

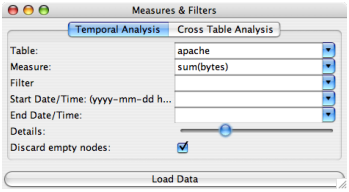


Figure 1: Specifying the database query and visualization parameters.

Using the snowflake schema, a separate table is created for each level of the dimension IP address and the tables' entries are linked accordingly. For example, each IP address is linked to the tuple of its advertised IP prefix, which in turn is linked to its AS, etc. Pre-joining the lowest level of the IP hierarchy with all its upper level entries avoids expensive join operations during interactive analysis and thus considerably speeds up queries. *IP addresses* are grouped by *IP prefix* → *autonomous system* → *country* → *continent* and we thus obtain the hierarchy as shown in Table 1.

Table 1: IP/AS hierarchy

Level	Name	Entries
1	continents	7
2	countries	190
3	autonomous systems	23054
4	prefixes	197427

The snowflake schema is not limited to the dimension IP hierarchy, but can be - depending on the analysis field - easily extended to various dimensions, such as time, protocol, ports, or type of event.

3 Related Work

A common way of displaying hierarchical data is in layouts where child nodes are placed inside parent node boundaries. Such displays provide spatial locality for nodes under the same parent, and visually emphasize the sizes of sets at all levels in the hierarchy. Usually, leaf nodes may have labels or additional statistical attributes that may be encoded graphically as relative object size or color.

The most important layouts of this type are *Treemaps* – space-filling layouts of nested rectangles, of which there are several main variants. The earliest variant was the *slice-and-dice Treemap* [8]. Here, display space is partitioned into slices whose sizes are proportional to to the sum of the nodes they contain. At each hierarchy level this procedure is repeated recursively, rendering child nodes inside parent rectangles while alternating between horizontal and vertical layouts. This is not difficult to program and can run efficiently, but long, thin rectangles arise, which are hard to perceive and compare visually.

Squarified Treemaps [3], remedy this deficiency by using rectangles with controlled aspect ratios. Rectangles are prioritized by size, so large ones are treated as the most critical ones for layout. This improves the appearance of Treemaps, but does not preserve the input node order, which is also a problem in some applications. This drawback was noticed, and *Ordered Treemaps* [2] were introduced to address it.

An alternative, non-Treemap layout algorithm which is not space filling was applied in [7] to the visualization of computer security data. Their proposed method maps hosts to rectangles, and subnets to larger enclosing rectangles. In contrast to this method, the approach proposed here has the goal of integrating both geographic and abstract layouts in the same view, and scaling up to the entire IPv4 address space.

Another related area is recent work on rectangular layouts in cartography, such as rectangular cartograms [5] which optimize the layout of rectangles with respect to area, shape, topology, relative position, and display space utilization. A genetic algorithm has been applied to find a good compromise between the objective functions describing the above mentioned properties. This technique renders layouts offline, not interactively and does not yet exploit hierarchical structures.

HNMap [11] supports the formation of a mental model of measurements reflecting the global Internet. It combines several layout techniques to cope with a large, multilevel AS/IP hierarchy in a tool that runs fast enough for interactive response. In another study [10], we evaluated alternative layouts and reported a case study with network data from a web server, a university network gateway, and an intrusion detection system (IDS) from a service provider to gain deeper insight into these large data sets.

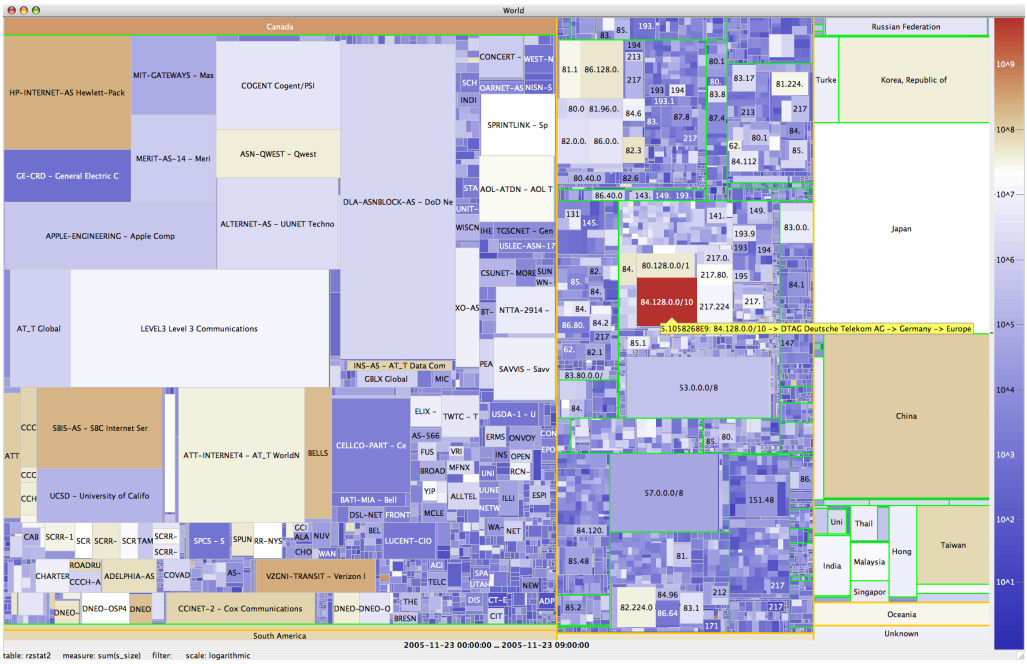


Figure 2: Multi-resolution HNMap approach to display aggregated IP-referenced measurements of prefixes (middle), ASes (left), countries (right), and continents (bottom) using a bi-color scale.

Drawing network traffic as lines on top of maps is well-known and has been studied in the cartographic community. Unfortunately, highly interconnected nodes lead to a lot of visual clutter and make it difficult to recognize any structure. Becker et al. [1] attack the problem by inventing *line shortening* and visualizing an adjacency matrix instead of geographic objects. Brushing has also been recommended [15].

A further approach to show the movement of objects from one location to another are so-called *flow maps*. Traditionally, these flow maps were hand drawn to reduce visual clutter introduced by overlapping flows. Phan et al. presented a method for generating well-drawn maps, which allow users to see the differences in magnitude among the flows while minimizing the amount of clutter [14]. However, interpretation of flow maps with multiple vantage points stays challenging.

A recent study [6] handles the problem of visual clutter elegantly: a hierarchical classification of nodes is exploited for bundling lines that connect leaf nodes, to visually emphasize correlations. Inspired by this work, we draw *edge bundles* on top of HNMaps in this paper. This enables us to view end-to-end relationships in network data sets, instead of limiting our analysis to the outgoing or incoming traffic from a single vantage point.

4 Hierarchical Network Map

In visualizing times series of network statistics, setting node (rectangle) sizes proportional to a time series variable leads to confusing displays, due to repositioning of nodes between HNMap frames. Figure 2 shows the multi-resolution HNMap approach addressing spatial memory by fixing node positions to facilitate tracing through time. A reasonable approach to this is to make node sizes proportional to the number of IP addresses contained, which is static in our experiments.

The general visualization paradigm of our technique places child nodes within the bounds of their parent node's rectangle. This results in a grouping operation which adds semantic meaning to the otherwise unstructured data. The benefit is obvious for

the parent child relationship among the continent and country as well as the AS and prefix nodes. However, the relationship between countries and ASes is not always clear. For this, we rely on statistics for each IP within an AS, which we obtained from a commercial GeoIP database [12]. Unfortunately, the country information within the registration services of ARIN, RIPE, AFRINIC, APNIC, LACNIC as obtained from [16] were uncomplete and sometimes misleading (a lot of ASes are registered to EU rather than a country).

The upper two levels of the AS/IP hierarchy are geographic entities. In general, geographic visualization is often very compelling– two-dimensional maps are familiar to most people as a convention for representing three-dimensional reality. Remarkably enough, mental models derived from maps are effective for many tasks even when extreme scales and nonlinear transformations are involved. Many approaches have been investigated for showing geographically-related as well as more abstract information on maps [4].

As a similarity measure for the AS level, we calculated the middle IP address of each AS by averaging the weighted middle IP address of all its advertised IP prefixes. This information is only meaningful to a certain extent as ASes sharing similar prefixes are positioned next to each other, but it does not take connectivity among ASes into account.

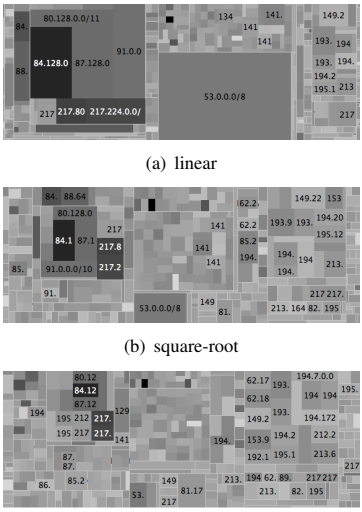
The lowest level of the hierarchy consists of the prefixes which have a clearly defined order. Our system therefore offers the capability to display this level using the Ordered Strip Treemap [2] with a line-wise sorting order. However, the HistoMap 1D method scales better to the large hierarchy data at hand with respect to visibility of small prefixes, layout preservations, and rendering performance (cf. [10]). The hierarchy levels can be interactively drilled-down or rolled-up. The mouse cursor highlights the measure for one particular hierarchy node and labels for the current hierarchy level as well as all parent nodes. When loading a data set, pruning nodes with little or no traffic frees space for the current analysis and retaining layout stability at the same time becomes an important aspect [10]. Since this paper is a continuation of our previous work, we discard nodes with no traffic in the HNMaps presented in this study.

Large differences in sizes between IP prefixes, ASes, countries, and even continents turn visual comparison of the respective rectangles into a challenge, especially when dealing with ordinary computer displays as opposed to wall-sized displays. We opted for a compromise by scaling the IP prefix sizes (number of contained IP addresses) and therefore indirectly also the upper level aggregates using square-root or, alternatively, logarithmic scaling:

$$f_{sqrt}(x) = \sqrt{x} \tag{1}$$

$$f_{log}(x) = \log x + 1 \tag{2}$$

The effects of node size scaling are illustrated in Figure 3. Note that the size of the upper level rectangles is determined through the sum of the scaled child nodes.



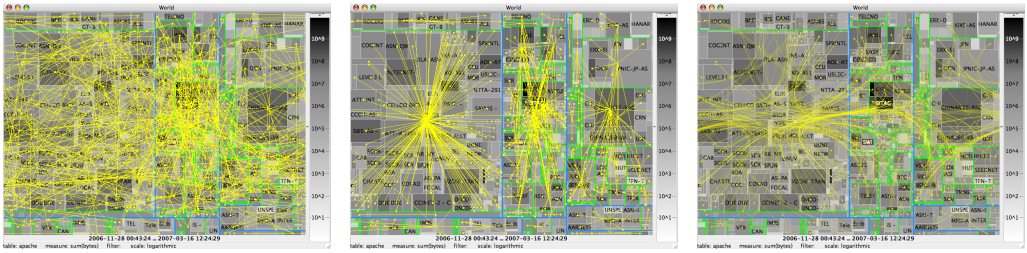
(a) linear

(b) square-root

(c) logarithmic

Figure 3: Effects of scaling on the space-filling layout demonstrated on some IP prefixes in Germany.

In the proposed visualization, the value of a measure of interest is encoded as color, using a fixed color scale and logarithmic normalization: $colorindex(v) = log(v+1)/log(v_{max}+1)$. The user is free to move the transition point (white) in the bi-color scale (blue to red) to focus his analysis on a range of values. When drawing edge-bundles, we substitute the blue-red color scale with a white-black color scale for the HNMap in the background to improve visibility of the bundles.



(a) Straight connection lines (b) Exploiting hierarchical structure (c) Compromise through edge bundles

Figure 4: Comparison of different strategies to draw adjacency relationships among nodes of the IP/AS hierarchy on top of the HNMap.

In some cases, analyzing an absolute measure of network traffic (e.g., number of connections or bytes transferred) provides hardly any insight in the time-varying dynamics of the data. Therefore, we calculate and display the change over time in some analysis scenarios.

5 Drawing routed traffic for multiple sources and destinations

Figure 4 shows three different approaches to draw adjacency relationships among nodes of the IP/AS hierarchy. (a) The problem of visual clutter is obvious when straight lines are drawn. (b) Simply connecting with the upper level nodes of the hierarchy removes visual clutter, but adds a lot of ambiguity, e.g. the lines connecting the left star (U.S.) and the stars in Europe (middle) – one for every country – are overdrawn several hundred times. (c) An interesting compromise is to use so-called edge bundles and transparency effects to combine the advantages of the latter approaches.

5.1 Edge Bundles

A recenty study proposed a way of using spline curves to draw adjacency relationships among nodes organized in a hierarchy [6]. Figure 5 illustrates how we use hierarchy structure to draw a spline curve between two leaf nodes. P_{start} (green) is the center of the source rectangle representing a node in the IP/AS hierarchy and P_{end} (red) the center of the destination rectangle. All points P_i (green, blue, and red) from P_{start} over $LCA(P_{start}, P_{end})$ (least common ancestor) to P_{end} form the cubic B-spline's control polygon. Because many splines share the same LCA (in this case the center point of the world rectangle), we do not use them for the control path.

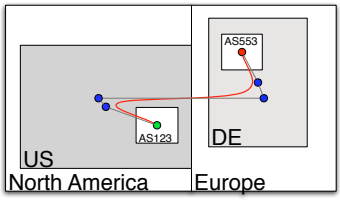


Figure 5: Spline (red) with control polygon (gray)

The degree of the B-spline used to control the bundling strength. In our experiments, a degree of 6 has proven to be a good choice for the cases where we have enough control points.

5.2 Edge Coloring

In general, we considered two options for coloring the edges, namely (a) use of color to convey the amount of traffic transferred and (b) use of color to distinguish edges. For the first case (see Fig. 6), we employed a heat map color scale from yellow to red using 50 to 0 percent alpha blending to visually weight the high traffic links. Naturally, the less important splines were drawn first. For the second case (see Fig. 7), we chose a HSI color map with constant saturation and intensity which is silhouetted against the background. The largely varying colors make tracing of single splines easier, but we noticed that users were strongly distracted by the colorful display.

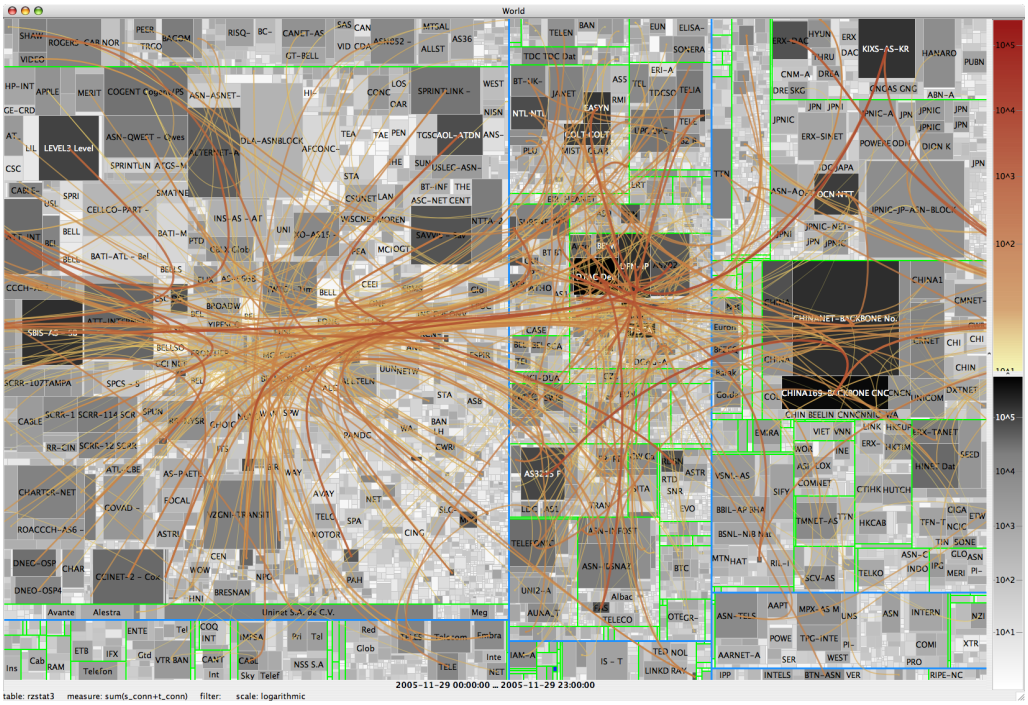


Figure 6: HNMap with edge bundles showing the 500 most important connections of one-day incoming and outgoing traffic of our university gateway (the anonymized destination/source is semi-randomly chosen). Color combined with transparency and line width communicate the amount of traffic of each spline.

5.3 Interaction design

In our opinion, the task of tracing splines is not solvable by means of coloring as soon as their number excels about one hundred nodes. We therefore tried to tackle the problem through interaction. Basically, there are two possible interaction scenarios. The first scenario is that a particular spline or region is selected, refined, and visually highlighted. Selecting a spline or a region by the start and end points of the splines is relatively intuitive to implement using the spatial data structure of the HNMap application.

However, the second scenario comprises marking a whole bundle of splines and was discarded since the implementation becomes tedious at this place (probably a point in polygon test for each pixel of all splines).

After specifying the start or end points of the splines using mouse interaction, we redraw all splines using their previous RGB color values – the not selected splines with higher alpha blending and the selected ones without transparency effects on top. This allows the user to easily trance the few highlighted splines to their end.

5.4 Data Simulation

As it was impossible to obtain real traffic data from service providers for publishing, we settled on using real netflow data from our university gateway and substituted the internal source IP prefix with a randomized one according to algorithm 1.

This randomization schema is based upon the assumption that nodes with more traffic are more likely to communicate with several other nodes, whereas the opposite applies to low traffic nodes.

Figures 6 and 7 demonstrate the outcome of our randomization schema based upon a one-day traffic load of our university gateway. The images highlight the high-level connectivity information while still being able to recognize the low-level relations

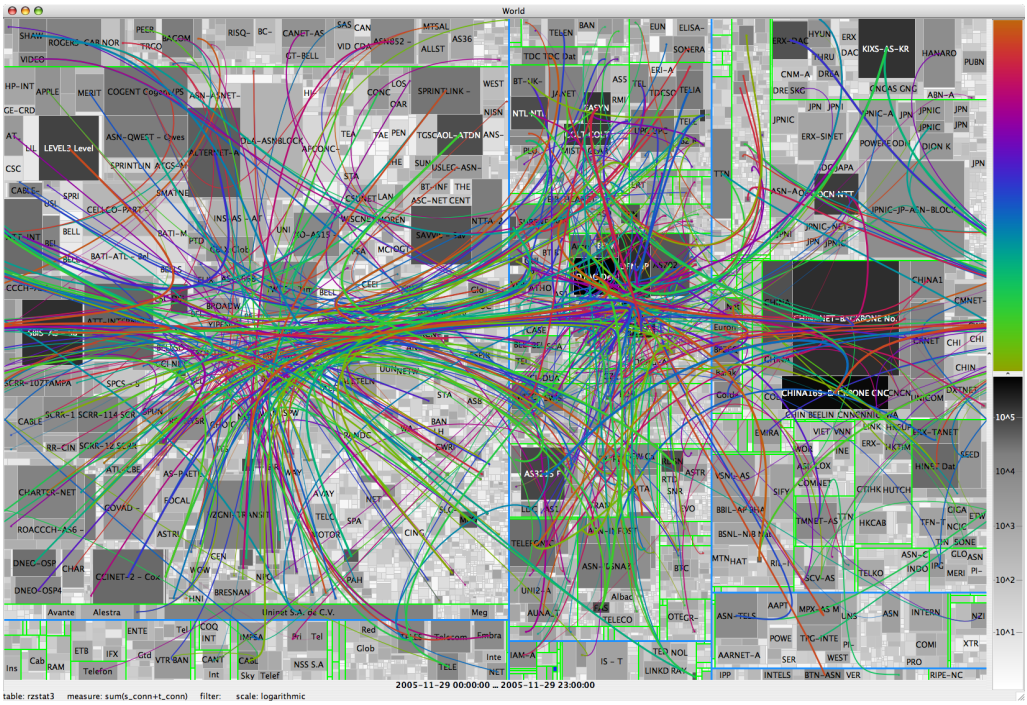


Figure 7: HNMap with randomly colored edge bundles makes splines more distinguishable. The amount of traffic is expressed only through spline width.

begin
  $SortedList \leftarrow$ all unprocessed nodes and weights
  while $|SortedList| > 0$ **do**
    $(n_{min}, t_{min}) \leftarrow SortedList.removeMin()$
    $(n_{rand}, t_{rand}) \leftarrow SortedList.RandomSample()$
    $drawSpline(n_{min}, n_{rand}, t_{min})$
    $SortedList.update(n_{rand}, t_{rand} - t_{min})$
end

Algorithm 1: Randomization schema to simulate backbone provider traffic.

to a large extent. The strong bundling effects in North America (center left) are explainable through the proximity of two control points to each other (North America, United States).

6 Findings and Evaluation

Edge bundles offer the possibility to convey source destination relationships on top of the HNMap and thus leverage the application from a purely measurement based analytical approach to a more complete view on large-scale network traffic. Since it is very challenging to trace single splines as soon as a certain volume of traffic links are placed on the map, we experimented with the RGB and alpha values of the spline colors. Mouse interaction is used to select splines at their start or end points in order to silhouet them from the remaining splines.

When monitoring larger networks, focusing the analysis on a particular type of traffic, for example communication of hijacked computers of a botnet, helps to significantly reduce the number of nodes and connections to be displayed.

During the analysis, further detailed information of other attributes of the data set at hand can be displayed using bar charts or the Radial Traffic Analyzer [9]. One major drawback of our approach is

that conveying significance of splines or to distinguishing them through color disqualifies the visual variable color for being used to indicate direction of traffic flows.

7 Conclusion

HNMaps visually represent network traffic aggregated on IP prefixes, ASes, countries, or continents and can be used for exploration of large IP-related data sets. In this study, we extended HNMaps through edge bundles to emphasize the source destination relationship of network traffic. Rather than representing new visualization or analysis techniques, we combined two existing methods and applied them to large-scale network traffic to gain deeper insight.

In order to facilitate tracing of splines, which represent source destination relationships among networks or abstract nodes of the IP hierarchy, we compared two alternative coloring schemes. Moreover, mouse interaction was proposed to visually enhance network traffic links of interest.

References

[1] R. A. Becker, Stephen G. Eick, and A. R. Wilks. Visualizing network data. *IEEE Transactions on Visualization and Computer Graphics*, 1(1):16–21, March 1995.

[2] Benjamin B. Bederson, Ben Shneiderman, and Martin Wattenberg. Ordered and quantum treemaps: Making effective use of 2d space to display hierarchies. *ACM Trans. Graph.*, 21(4):833–854, 2002.

[3] M. Bruls, K. Huizing, and Jarke J. Van Wijk. Squarified treemaps. In *Proceedings of the Joint Eurographics and IEEE TCVG Symposium on Visualization*, pages 33–42, 2000.

[4] Martin Dodge and Rob Kitchin. *Atlas of Cyberspace*. Addison-Wesley, 2001.

[5] Roland Heilmann, Daniel A. Keim, Christian Panse, and Mike Sips. RecMap: Rectangular Map Approximations. In *InfoVis 2004, IEEE Symposium on Information Visualization, Austin, Texas*, pages 33–40, October 2004.

[6] Danny Holten. Hierarchical Edge Bundles: Visualization of Adjacency Relations in Hierarchical Data. *IEEE Transactions on Visualization and Computer Graphics*, 12(5):741–748, 2006.

[7] Takayuki Itoh, Hiroki Takakura, Atsushi Sawada, and Koji Koyamada. Hierarchical visualization of network intrusion detection data. *IEEE Computer Graphics and Applications*, 26(02):40–47, 2006.

[8] B. Johnson and Ben Shneiderman. Tree-maps: A space filling approach to the visualization of hierarchical information structures. In *VIS '91: Proceedings of the 2nd IEEE Conference on Visualization*, pages 284–291, 1991.

[9] Daniel A. Keim, Florian Mansmann, Jörn Schneidewind, and Tobias Schreck. Monitoring network traffic with radial traffic analyzer. In *Proc. of IEEE Symposium on Visual Analytics Science and Technology 2006 (VAST 2006)*, pages 123–128, 2006.

[10] Florian Mansmann, Daniel A. Keim, Stephen C. North, Brian Rexroad, and Daniel Shelehedal. Visual analysis of network traffic for resource planning, interactive monitoring, and interpretation of security threats. *IEEE Transactions on Visualization and Computer Graphics*, 13(6), 2007.

[11] Florian Mansmann and Svetlana Vinnik. Interactive Exploration of Data Traffic with Hierarchical Network Maps. *IEEE Transactions on Visualization and Computer Graphics*, 12(6):1440–1449, 2006.

[12] Maxmind, LLC. Geoip database, 2007. `http://www.maxmind.com`.

[13] T. B. Pedersen and C. S. Jensen. Multidimensional database technology. *IEEE Computer*, 34(12):40–46, 2001.

[14] Doantam Phan, Ling Xiao, Ron Yeh, Pat Hanrahan, and Terry Winograd. Flow map layout. In *INFOVIS '05: Proceedings of the Proceedings of the 2005 IEEE Symposium on Information Visualization*, page 29, Washington, DC, USA, 2005. IEEE Computer Society.

[15] Ben Shneiderman and Aleks Aris. Network visualization by semantic substrates. *IEEE Trans. Vis. Comput. Graph.*, 12(5):733–740, 2006.

[16] Team Cymru. IP to ASN Lookup Page, April 2007. `http://www.cymru.com/BGP/asnlookup.html`.