Interactive Sankey Diagrams

Patrick Riehmann   Manfred Hanfler   Bernd Froehlich

Faculty of Media
Bauhaus University Weimar
Germany

Abstract

We present a system that allows users to interactively explore complex flow scenarios represented as Sankey diagrams. Our system provides an overview of the flow graph and allows users to zoom in and explore details on demand. The support for quantitative flow tracing across the flow graph as well as representations at different levels of detail facilitate the understanding of complex flow situations.

The energy flow in a city serves as a sample scenario for our system. Different forms of energy are distributed within the city and they are transformed into heat, electricity, or other forms of energy. These processes are visualized and interactively explored. In addition our system can be used as a planning tool for the exploration of alternative scenarios by interactively manipulating different parameters in the energy flow network.

CR Categories: H.5.2 [User Interfaces]
Subject Descriptors: Graphical user interfaces
Additional Keywords: Sankey diagram, flow diagram

1 Introduction

Sankey diagrams are traditionally used to visualize the flow of energy or materials in various networks and processes. They illustrate quantitative information about flows, their relationships, and their transformation. Sankey diagrams represent directed, weighted graphs with weight functions that satisfy flow conservation: the sum of the incoming weights for each node is equal to its outgoing weights. These diagrams are often drawn by hand or they are generated by drawing programs. These passive illustrations are of limited use for complex flow scenarios.

We have developed a system that allows users to interactively explore complex Sankey diagrams (Figure 1). The implementation is built on top of a simulation model, which generates the flow information from a set of input parameters using a data flow evaluation approach. The simulation model and the flow graph can be analyzed in detail using the flow tracing feature, which allows users to follow individual flows across the flow graph.

Our work was motivated by manually drawn flow diagrams used for displaying the energetic structure of a city. These diagrams are used to answer questions like

- How is the energy distributed?
- Where are the hugest losses?
- What kind of energy is used for which purpose?

The actual complexity of the different energy flows in a city is rather surprising and it is based on data collected on a yearly basis from the local power suppliers, gas stations, and other sources.

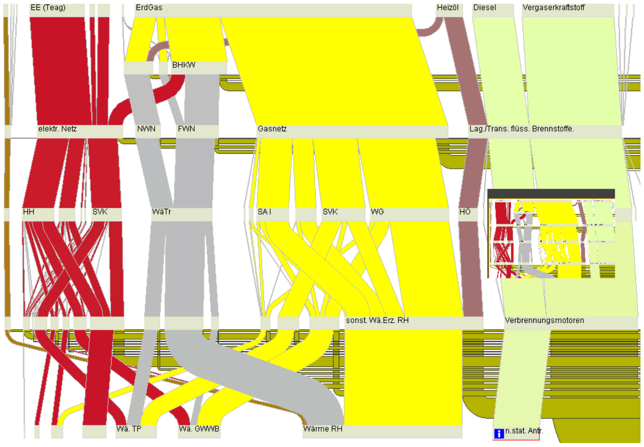

Figure 1: A Sankey diagram visualizing the energy distribution in a city. The nodes represent different types of energy and transformation processes. The width of the nodes and edges provide quantitative flow information.

In the past this data was entered into spread sheet programs to model the energy flow from primary energies coming into the city down to the energies actually used to heat the homes, light buildings, and cook meals. The spread sheet consisted of a large number of values and equations to precisely model the different consumer groups and uses of energy. The resulting spreadsheet was then used to manually draw an abstracted version of the resulting flow diagram. All the detail was lost in this final step and only an overview was provided. In addition, there are a lot of approximations and assumptions used in the whole process, which are not communicated by the final rendering.

The main contribution of our work is the development of interactive Sankey diagrams, which facilitate the exploration of complex flows. We have developed our system as a tool for the analysis, communication, and planning of the energy distribution in a city. Our interaction techniques allow the exploration of the diagrams on different levels of detail and the tracing of flows across the graph. An underlying simulation model supports the interactive change of parameters in the flow graph and triggers a recomputation of the flow quantities. Animated Sankey diagrams are efficient means to convey the drastic change of the use of different types of energy over time. The system uses XML for the representation of the underlying weighted graph. The implementation is based on Java and Java2D, which allows embedding in a web browser. First experiences with public users and experts confirm that the interactivity of our system facilitates the understanding of complex flow scenarios. Our tools and techniques may also be used for the visualization of other flows such as cost flow, flow of materials, and network traffic.

2 Related Work

"It may well be the best statistical graphic ever drawn." argues Edward Tufte in his 1983 book "*The Visual Display of Quantitative Information*" [21] referring to the drawing "Napoleon's Russian campaign of 1812" of Charles Joseph Minard. His map is a very early version of a Sankey diagram drawn already in 1861. Tufte pointed out that six variables were captured within this map. The two most essential variables are:

1. The size of Napoleon's troops is encoded as the thickness of the line at any given point in time.
2. There are (in this case geographical) points or nodes where significant changes happened.

C.J. Minard said about his creation: "The aim of my carte figurative is less to express statistical results, better done by numbers, than to convey promptly to the eye the relation not given quickly by numbers requiring mental calculation." An interesting, modern and computer supported recreation of his map was produced by Glennon and Goodchild [10] using Tobler's Flow Mapper [20]. An interactive version of Minard's map, which linked the nodes and edges on the map to other illustrations of statistical data, was produced with the SAGE [3,17] system.

The Irish engineer M. H. P. R. Sankey (*1853, +1921) used these types of drawings to illustrate energy flows[7]. Since the end of the 19th century, flow charts or Sankey diagrams were often applied in static visualizations of dynamic processes, like transport and traffic flows, trade relations or processes of production and conversion.

The visualization of connectivity and networks like telephone networks, parts of the Internet, are also based on weighted graphs. Examples are the interactive visualization of the global topology of the MBone by Munzner and Hoffman [15] or the network traffic in the SeeNet System developed by Becker, Wilks and Eick [1]. They pointed out that the two most common approaches representing traffic flow quantities are colors and the thickness of lines. They could not make extended use of the line thickness, since their visualizations were drawn on top of geographical maps, which would have been heavily occluded.

Cox and Patterson [4] used color encoded traffic flows in their "Visualization Study of the NSFNET". The visualization of the CESNET topology [2] and Krempel's visualization of the international trade with automobiles [13] use line thickness as a representation of flow quantities. Eick and Wills [6] encoded aggregate email flow quantities in the nodes' sizes and they used a color encoding technique for representing the flow volumes between the nodes. Commercial and free applications such as "S.Draw" [18] support users in drawing static Sankey Diagrams based on tabular data. Tobler's Flow Mapper [20] is able to create simple flow maps based on statistical data. His diagrams are embedded in geographical maps. These systems produce mostly static visualizations and they do not support features like flow tracing or different levels of detail.

Mansfield [14] showed how to produce Sankey diagrams using the Scalable Vector Graphics (SVG) language and data provided in XML and transformed using an XSLT style sheet. His Sankey diagrams support very basic interactions like popping an edge into the front if the mouse hovers over it. More complex interactions are difficult to support within SVG.

3 Energy Flow Structures in a City

Every city needs energy in various forms like gas or oil. The energy gets distributed within the city and is transformed into heat or electricity, for example. Those energies are powering the citizens' electrical devices, heat their houses and move their cars.

The distribution of such energies from the energy suppliers to the different users requires a complex network of cables, pipes, and gas stations in every city.

The collection of statistical data of the energy distribution in the city of Weimar has a long tradition, which dates back into the 1980ies. Manually drawn Sankey diagrams were already used in these days. Langner coined the term "das energetische Gesicht" of a city, which may be translated as the energetic face or imprint of a city. Each city has a characteristic energy flow diagram, which depends partially on the locally available energies, but in particular on the types and size of industrial companies in and around the city. The hand drawn Sankey diagrams were used to document, analyze and plan the energetic development of a city.

We had access to information dating back to 1990. In Germany electric energy and natural gas is provided to the homes by regional service providers, who have records of the total amount of energy fed into the city and the actually used energy in homes, companies, and public buildings. The difference is due to losses in cables and pipes. The public utilities have also statistical information about the different kinds of uses of the various forms of energy. Information of other energies had to be collected from local gas stations and from companies providing fuel oil or liquid gas to homes and companies. Some of the data was also estimated by energy experts from our university and Weimar's public utilities.

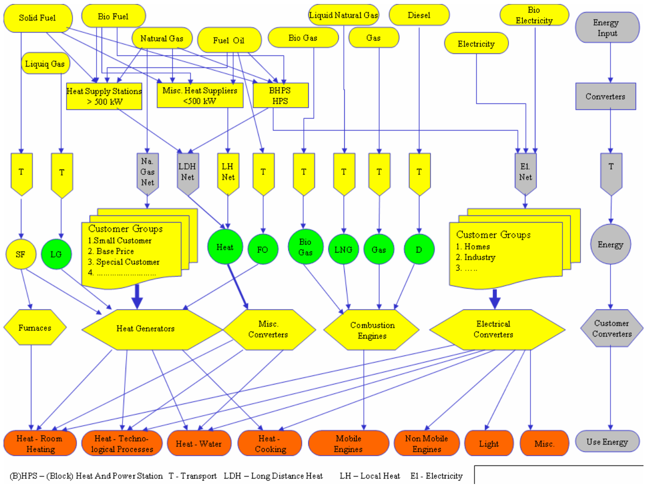

Figure 2: The underlying graph without weighted edges gives an idea of the complexity and relations of the energy distribution in the city of Weimar.

The graph in Figure 2 shows a detailed representation of the flow graph without weighted edges. The different levels (numbered from top to bottom) show the following information:

1. different types of energy provided to the city from national and regional energy suppliers, gas stations, and other sources.
2. large energy converters, such as power plants, which transform for example natural gas into heat or electricity.
3. ways of transporting the final energy to the customers.

4. customer groups, like homes, companies, and such.
5. energy converter groups at the customers' sites.
6. useful energy, which is the energy used to heat the apartments, light rooms, move cars around, etc

The hand drawn Sankey diagram in Figure 3 shows a simplified and abstracted version of the full graph shown in Figure 2. The transport layer (level 3) is not present and the energy converters at the customers' sites are represented as just one node. The loss edges are drawn to the left. This diagram provides already a lot of information, but it is only a fraction of the actual data shown in Figure 2.

For the development of our interactive Sankey diagrams we arrived at the following requirements by talking to the public utilities:

- There are two versions of the Sankey diagram required. A version for the experts, which provides detailed representation and manipulation capabilities such that it can be used for exploring different scenarios for the future. A much simpler and web-enabled version is required for public use, which provides information to the customers and can be used for educational purposes.
- Visualization at different levels of detail
- Zoom in and out
- Detailed numeric information on demand
- Tracing of flows through the graph
- A simulation model that allows the computation of the edge weights based on a minimal set of parameters.
- An authoring toolkit for adapting the system to different cities

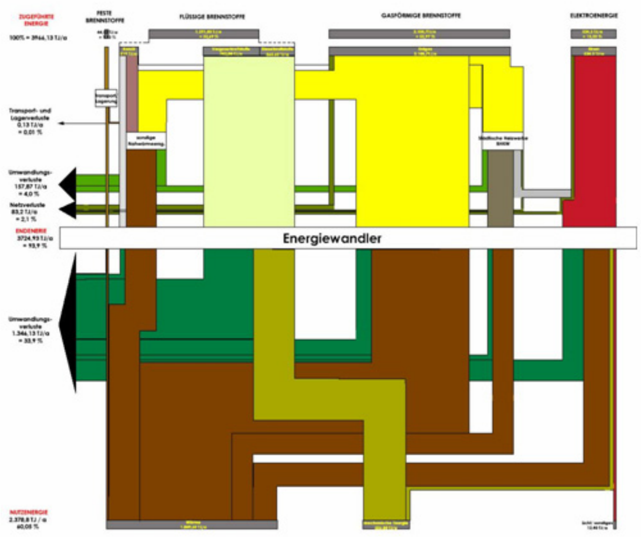

Figure 3: A hand drawn Sankey diagram of the energy flow in the city of Weimar

4 Interaction

Based on the requirement analysis we developed an interactive system for the exploration of our Sankey diagrams. For the definition of the flow graph we use an extended graph-editing tool, which allows users to create nodes and edges, set up the parameter of nodes, define colors for edges, and define equations relating quantities on incoming and outgoing edges in each node. This information is provided to the rendering component, which has full access to the underlying graph data.

Figure 4 shows the system in start up state, which provides an overview of the Sankey diagram at the lowest level of detail. The diagram is very similar to the hand drawn diagram in Figure 3. Only the loss edges are drawn to the right instead of to the left. From this starting point a variety of different techniques can be used to explore the diagram.

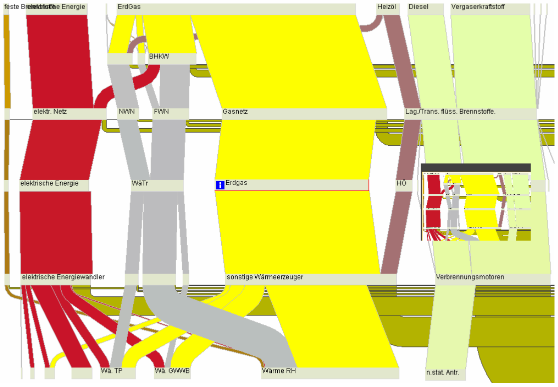

Figure 4: The start up display of the Sankey diagram

4.1 Grouping and Level of Detail

The nodes in the graph can be grouped to form a hierarchy. The grouping of nodes implies also a grouping of the adjacent edges (Figure 5).

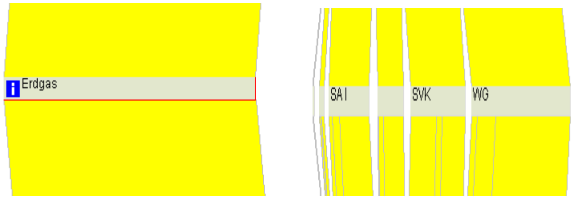

Figure 5: The left image shows the low level of detail versio of the node "Erdgas". The right image shows the .node divided into six nodes on the finer level of detail.

4.2 Overview and Detail

Overview and details is a common technique in information visualization systems. We decided to implement a combined display of the overview and magnified view as shown in Figure 6. Due to the limited size of the graph, a single fixed size overview window provides enough context for navigational purposes. Horizontal and vertical sliders may be used for panning as well as the frame of the magnified window in the overview display.

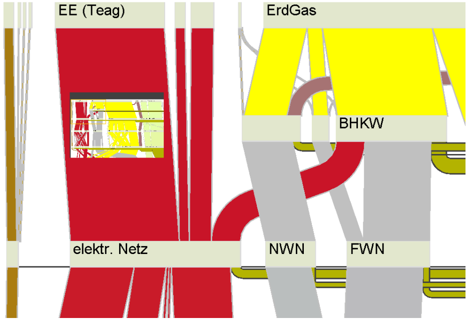

Figure 6: The overview and detail windows. The overview window can be used for panning of the detailed view.

4.3 Flow Tracing

One of the most important features of the interactive Sankey visualization is the tracing of flows throughout the graph. Users may select a node or edge in the graph and the contributions of all flows are highlighted and moved into the foreground. Figure 7 shows an example of a selected node.

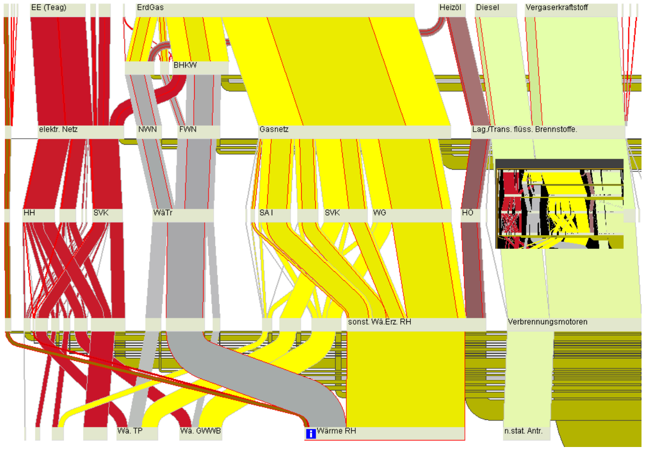

Figure 7: A node on the bottom is selected and the contributions from different branches of the flow graph are highlighted.

In our first implementation, we simply highlighted all the contributing edges without showing the actual contribution of each edge. Since each edge receives often a contribution from many sources and feeds many sinks, a large part of the graph was highlighted. This implementation was useful in cases, but it soon became clear that quantitative information of the flow contributions are much more interesting. Our current implementation shows the actual contribution of each edge of the flow graph. Each node may have one or more incoming and outgoing edges, which are highlighted. In the following examples we show how these situations are handled. Flows are traced from bottom to top. Depending on the actual configuration, several situations may occur

1. There is a single highlighted incoming and a single outgoing edge. In this case we are continuing the highlighted contribution directly through the node as shown in Figure 8.

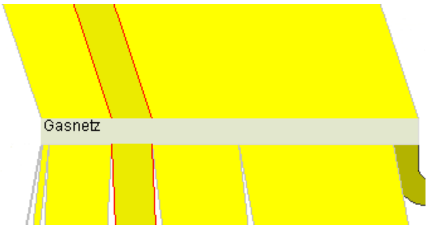

Figure 8: Continuity of a single traced edge through a node

2. There is a single highlighted incoming edge and there are multiple outgoing edges: The relative position of the incoming edge on the node is continued in the outgoing flow (Figure 9). In our current implementation the incoming fraction of a flow gets proportionally distributed to the outgoing edges and vice versa. For the currently used simulation model, the proportional distribution from output to input is correct, which is certainly not a valid assumption for the general case.

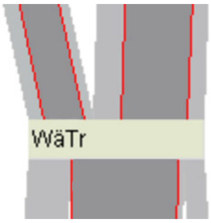

Figure 9: Relative position of the single incoming traced edge is passed on to the two outgoing edges

3. In all other cases, the sum of all incoming quantities is computed and proportionally distributed to one or more outgoing edges (Figure 10).

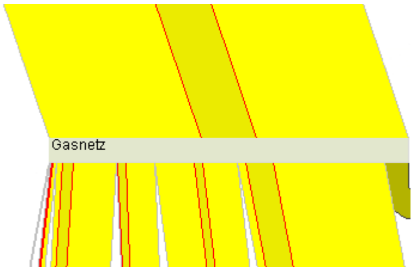

Figure 10: Multiple incoming edges are accumulated and continued in the middle of the outgoing edge.

A very important improvement of the flow tracing visualization was to color the highlighted flow fractions slightly differently than the actual flow (Figure 11). The traced flow appears as a continuous stream, which facilitates the flow tracking across different levels of the flow graph.

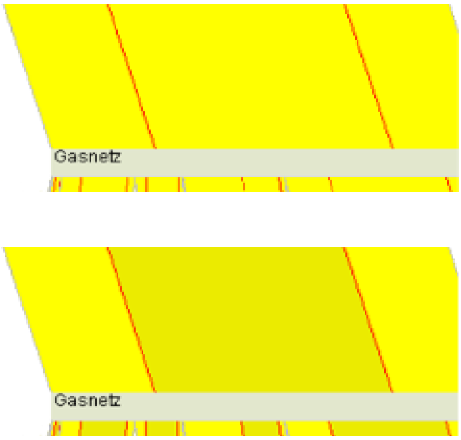

Figure 11: In the lower rendering, the traced flow is colored slightly darker than the surrounding flow, which simplifies visual flow tracing. The upper image does not use this feature.

4.4 Detail on Demand

Additional information about each node is provided on demand. This information is shown in a separate window and includes actual quantities flowing through the selected node, text information, an associated html page, or images (Figure 12).

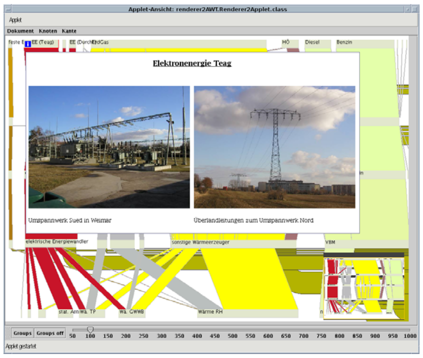
Applet-Ansicht: renderer2AWT.Renderer2Applet.class
Applet
Dokument Knoten Kante
Elektronenergie Teag
Umspannwerk Sued in Weimar
Überlandleitungen zum Umspannwerk Nord
Groups Groups off
Applet gestartet

Figure 12: The information window shows the power plant and a power pole.

4.5 Semi-automatic Layout

We implemented a semi-automatic layout, which can be manually adjusted. The order of the nodes on each level and the distance between the different levels of the flow graph needs to be provided. The horizontal positions of the nodes can be automatically calculated depending on their required space. In our case, we do not have the same amount of energy on each level, since the loss edges are drawn to the right. Our system suggests horizontal positions for the nodes, which are often manually adjusted to improve the legibility. A more automatic layout considering various requirements is clearly desirable.

For minimizing the number of edge crossings, the incoming or outgoing edges of a node can be automatically sorted by the horizontal position of their sources or targets.

4.6 Planning Tool

One of the most important features for the use of our system as a planning tool is the possibility to change parameters of the underlying simulation model. This model is basically a data flow graph, which can be evaluated on the fly. The parameters in each node can be changed, which results in a recomputation of the quantities for edges and nodes in the graph. Thus planners may explore different future scenarios – for example – by changing the efficiency of energy converters. Figure 13 shows an example.

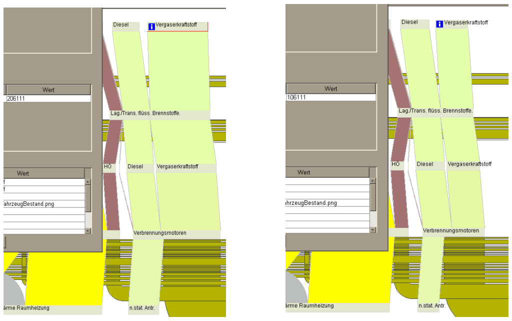

Figure 13: In the left image the provided energy in the top node on the right is changed. In the right image the updated flow diagram is shown.

4.7 Animation

The animation feature is still very basic. The idea is to animate the changes of the energy distribution over time. This feature can be used to visualize the history of the energy flows or to extrapolate into the future. Simple parameter studies can be supported as well. Currently we are able to support a key frame animation of two topologically equal graphs. An example is shown in Figure 14. For general animations over time, appearing and disappearing edges and nodes have to be considered, which is currently not implemented.

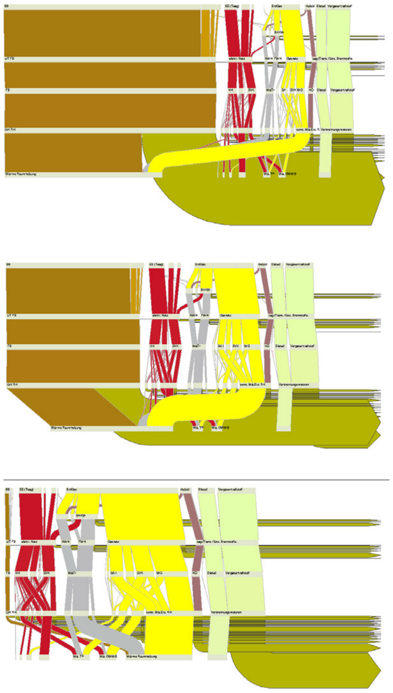

Figure 14: Three frames of the animation from 1990 to 2003. In 1990 a lot of coal was used shown in brown. Today coal is mostly replaced by natural gas shown in yellow.

5 Edge Layout

Drawing the edges appropriately is a surprisingly difficult problem. Besides the already mentioned edge sorting and semi-automatic arrangement of the nodes, there are still many issues to consider – in particular since we have two types of edges:

- regular edges connecting a node on one layer with a node on the next layer
- loss edges begin at nodes on an arbitrary level and they are drawn to the right.

5.1 Edge Drawing Order

Regular edges between two layers are sorted by their thickness and they are drawn with the thick edges on top of the thin edges. This seems counter intuitive, since the thick edges would occlude the thin edges. However the thick edges are the most important structures. Figure 15 shows the resulting images for the two different drawing orders.

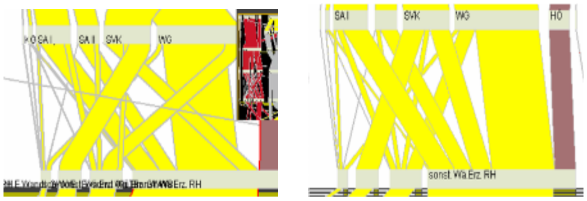

Figure 15: Drawing thick edges on top of thin edges moves the important information to the foreground as shown in the right image.

Loss edges may be sorted by size in vertical direction increasing from top to bottom as shown in Figure 16. Alternatively, they are sorted by the horizontal position of their corresponding node. In our experience both sorting orders work well and they are both legible. Sorting by size introduces more edge crossings, but it is visually more organized.

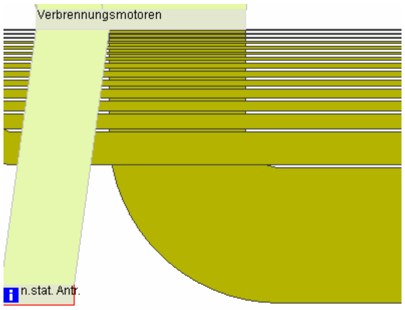

Figure 16: Loss edges are sorted by size

Drawing the loss edges in the same order as the regular edges resulted in too much occlusion of the thin loss edges. Thus we draw the thin loss edges on top of the thick ones. The overall drawing order (from back to front), which provided the best possible result for our test scenarios:

1. thick loss edges
2. thin loss edges
3. thin regular edges
4. thick regular edges

5.2 Edge Drawing

The width of an edge is supposed to represent the actual flow quantity going through the edge. If edges are not horizontal or vertical, they are not of constant width if they are drawn as straight connections. Figure 17 shows an example. Mansfield[14] suggested to construct the edges from concentric circles and parallel lines as shown in Figure 17. We adopted this edge drawing model for most cases. Only for almost vertical edges we simply use two parallel lines to connect the two nodes on different levels.

One of the most surprising findings was that the edges in a Sankey diagrams suffer from the so-called vertical-horizontal illusion[8]. Most humans perceive a vertical line as longer than a horizontal line of the same length. In our system we draw loss edges first in the vertical direction and then horizontally to the right. For thicker loss edges the width of the horizontal part has been perceived considerably thicker than the vertical part – see Figure 18. This sounds like the opposite of the vertical-horizontal illusion, but thickness of the line is measured orthogonal to the direction of the edge. This illusion was very apparent when the Sankey diagram was projected on a large projection screen. We have not tried to correct for the vertical-horizontal illusion, but it would be interesting to perform a user study to establish appropriate correction factors.

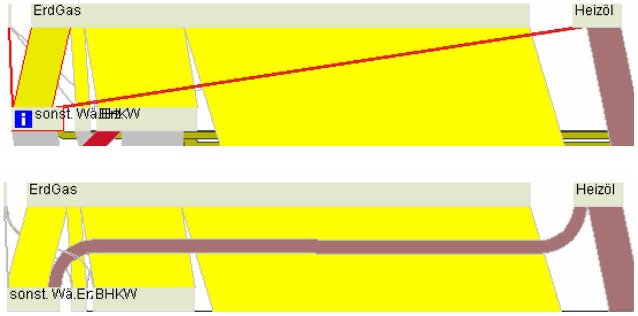

Figure 17: The edge width is incorrectly displayed by straight connections shown in the top picture. The bottom picture shows an improved version based on Mansfield's approach.

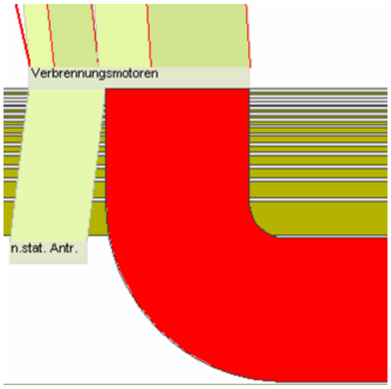

Figure 18: A vertical horizontal illusion was perceived by almost all users of our system for the red loss edge.

6 IMPLEMENTATION

We have tested many web-based platforms like SVG, Macromedia Flash and others for implementing our ideas. These platforms show promise at first and for small diagrams, but they suffer from severe limitations for complex interactive displays.

SVG as a vector image format could be an excellent base for static Sankey diagrams, but it does not support complex interaction scenarios. Interaction has to implement with embedded Java Script code and the performance for simple test scenarios was already insufficient.

Macromedia Flash provides better support for user interaction than SVG and a quite acceptable performance for simple Sankey diagrams. However, programming a fully interactive system including a simulation model would have been difficult and also performance would have been an issue.

In the end we decided to use the well documented web-based Java/Java2D platform. Java2D is bundled with each Java edition and provides a comfortable application programming interface for two-dimensional graphics applications. It is integrated with both graphical toolkits: Java AWT and SWING. The SWING integration suffers still from some performance problems, since it does software rendering only. Since SWING is the newer toolkit, it provides a much larger set of widgets, but after implementing advanced edge drawing techniques and animations, we realized that the SWING software rendering became a limiting factor. We had to resort back to AWT, which provides hardware supported graphics rendering.

The main components of our system are the editor for setting up the graph model, the actual interactive Sankey rendering component, and the simulation model. These components are based on the same data model. The simulation model performs the evaluation of the underlying data flow graph based on provided quantities and parameters.

6.1 The Data Model

The Graph eXchange Language GXL extends XML to describe common graphs. Our data model uses some of the GXL tags for modelling the underlying graph, but GXL was not expressive enough to represent all the required information for our system. Therefore we had to resort back to some standard XML structures, which can not be understood by a GXL compliant graph browser. For the description of the graph we use the following abstractions: subgraph, node, edge, port, and parameters. Subgraphs are used for representing the grouping information. A node consists in general of one or more incoming and outgoing edge ports and additionally some parameter ports. The output ports are often specified in terms of the input ports using an expression. The expression references input ports, parameters from parameter ports. In general a port can hold an expression, a parameter or a link to an edge as shown for an example in Figure 19.

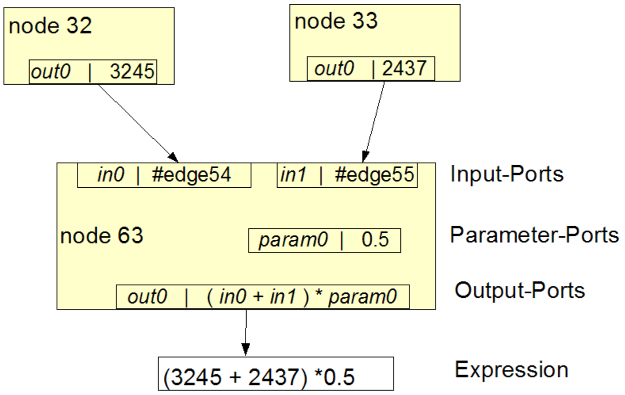

Figure 19: An example node that illustrates the idea of evaluation within one node.

6.2 The Simulation Model

The simulation model uses an evaluation component, which is based on a recursive process to substitute all the variables in the port expressions by scalar values. This substitution process is based on the underlying data model and follows the edges across the graph starting in each node of the graph until all the nodes are evaluated. Nodes that have been already evaluated, do not need to be evaluated again. The scalar values in each port are typically the width of the associated edges. An example is shown in Figure 19.

In some cases, data is known at different levels of the graph, which does not allow a simple top down graph evaluation approach. A good example is the calculation of losses in the city wide electricity net. We know the amount of energy, which was provided to the city by regional power suppliers. This data is represented in the top level of our graph. We also know the amount of energy that was consumed by the customers, because this is how the public utilities charge their customers. The difference between these values is due to losses in the cable net and transformation processes, hacked electricity meters, and other legal or illegal access to power lines before the meters. In such a case the loss is dependent from incoming and outgoing edges as shown in Figure 20.

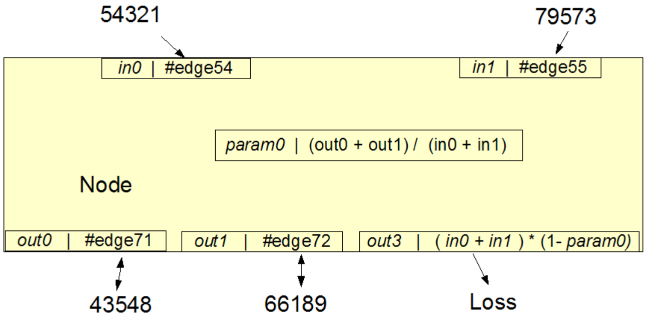

Figure 20: The loss calculation for port out3 can be seen in the port for param0, which depends on the data from all the other incoming and outgoing edges.

6.3 Authoring and Rendering Component

We had originally developed a system with an integrated authoring and rendering component. Soon it became clear that the editor had to be a separate component due to the total complexity of the integrated approach. The editor is now a basic graph editing toolkit, which allows users to comfortably set up the nodes, edges, groups, colors, and in particular the input/output relationships. These relationships are described by the expressions contained in the nodes, which are used by the evaluation mechanism. The editor is already connected to the simulation component, which computes the width for each node and edge. This information can be used in the editor to suggest a basic graph layout, which is exported with the graph data to the rendering component through the XML data structure.

For the rendering component we had to shadow the XML structure in our own data structure to provide faster access to the graph data. The shadow data structure uses a doubly linked connection between edges and nodes, such that recursive graph traversals starting from any node can be handled easily and efficiently. This object oriented layer gave us also more flexibility besides much improved performance.

7 Experience and Discussion

The system has been presented to many non-experts and also to a number of experts from public utilities. The overall response was very positive. The system was used by non-experts to understand the energetic flows and transformation processes in a city. The main use for the current system will be in the educational area and for communication purposes.

Our experts from the local public utilities are interested in the visualization and interactive exploration of Sankey diagrams as a tool to verify their data. The flow tracing provides visual insight into the underlying simulation model, which also contains a lot of estimated parameters. Some inconsistencies and small errors in the model were already revealed by using our flow tracing technique and the visualization at different levels of detail.

The use of our system as a planning tool has been tried for simple cases by the experts and was quite appreciated. They suggested that this feature should be much extended to become a full planning instrument. In particular the animations need to be much easier to produce to support parameter studies. One example that was mentioned as extremely useful was the following: The useful energy, which is the fraction of energy that is actually turned into light, kinetic energy for moving cars, heat for heating buildings is always needed and changes only slowly. The way this energy is produced can differ widely. The city may decide to add a small power plant to their energy network, which is driven by natural gas. This way they do not have to buy so much electricity from the national power suppliers. Besides producing electricity locally, heat as a byproduct may also be transferred to the homes directly. Such changes to the local energy network structure need to be explored and related to actual costs. Our system would have to be extended to support changing graph topologies for the animations and an easy to use authoring toolkit for such scenarios.

8 Conclusion and Future Work

In this paper we have presented a system for the visualization and interactive exploration of Sankey diagrams. Our system supports a variety of techniques, such as overview and detail, detail on demand, level of detail, flow tracing and animation over time. The dependencies between the edges, nodes, and parameters of our energy flow graph are modeled as a data flow graph, which can be interactively evaluated. Our implementation is based on Java and Java2D, which allows the embedding of our tool in a web browser.

Our implementation provides only basic computer-supported layout functionality for the placement of nodes of the Sankey diagram. A more automatic layout is desirable, which considers the requirements for a perceptually and cognitively functional visual representation for different user types. This is particularly challenging for complex hierarchical Sankey diagrams with changing topology over time.

There is a lot of uncertainty in the whole data acquisition process, since the data is collected from different sources. In addition, there are also a number of assumptions in the model, which result from statistical data about the kinds of energy uses at the customer sites. The visualization of these uncertainties in the context of a Sankey diagram are an interesting research challenge.

Our implementation is only a first step towards the interactive specification, visualization, exploration and animation of Sankey diagrams. Many more techniques need to be explored to provide a usable system for daily work. The potential for such systems is large, since Sankey diagrams can be used for the analysis of many different types of flows such as packet flow in the internet, material flow in a company, traffic flow, water flow, and flow of money. The visualizations of these flows may have slightly different requirements, but their analysis and understanding will be facilitated by interactive Sankey diagrams.

Acknowledgements

We thank Carlo Moritz Göllner and Henrik Thoms for their work on the initial version of our system, the Stadtwerke Weimar for providing the data and many helpful discussions, and the many students involved in collecting the data and designing hand drawn Sankey diagrams.

References

[1] Becker R. A., Eick S.G., Wilks A.R., *Visualizing Network Data*, IEEE Transactions on Visualization and Computer Graphics, pp. 16-28, 1995

[2] CESNET, Network Topology, http://www.ces.net/network/

[3] Chuah M.C., Roth S.F., *SageBook: Searching Data-Graphics by Content,* Proceedings of the SIGCHI conference on Human factors in computing systems, 1995, http://www-2.cs.cmu.edu/~sage/Papers/SageBook/SageBook.html

[4] Cox D., PattersonR., *Visualization Study of the NSFNET,* http://archive.ncsa.uiuc.edu/SCMS/DigLib/text/technology/Visualiza ion-Study-NSFNET-Cox.html

[5] Chen Ch., Information visualisation and virtual environments, Springer, 4-5 8,135 -145, 1999.

[6] Eick S.G, Willis G.J., *Navigating Large Networks with Hierarchies*, IEEE Proceedings Information Visualization, pp. 204-210, 1993

[7] F.A. Brockhaus, *Der Brockhaus*, 2004

[8] Fineman, M. *The Nature of Visual Illusion.* New York: Dover, p. 153, 1996.

[9] Friendly M., *The Graphic Works of Charles Joseph Minard,* http://www.math.yorku.ca/SCS/Gallery/minbib.html#Minard:1844

[10] Glennon A. ,Goodchild M., *A GIS Flow Data Model,* National Center for Geographic Information and Analysis, University of California, Santa Barbara, October 2004 http://dynamicgeography.ou.edu/flow/

[11] GXL, Graph eXchange Language, http://www.gupro.de/GXL/

[12] Jünger M., *Graph drawing software* , Springer, 2004.

[13] Krempel L,.Plümper T., International Division of Labor and Global Economic Processes: An Analysis of the International Trade in Automobiles, Journal of World-Systems Research, Vol. V, 3, 487-498, 1999

[14] Mansfield P. A., Ambachtsheer M. A.: Programmatic Rendering of Directed, Weighted Graphs. *In Proceedings of the SVG Open 2003.* http://www.svgopen.org/2003/papers/RenderingGraphs/.

[15] Munzner T., Hoffmann B., *Visualizing the global topology of the MBone,* Proceedings of the 1996 IEEE Symposium on Information Visualization (INFOVIS '96) , p85, 1996.

[16] Robinson A. H., The thematic maps of Charles Joseph Minard., Imago Mundi, 1967.

[17] SAGE: System for Automated Graphics and Explanation http://www-2.cs.cmu.edu/~sage/sage1.html

[18] S.Draw, http://www.sdraw.com

[19] Spence R., *Information Visualization*, 6 23-25 Addison-Wesley, 2001

[20] Tobler W, Jones D., Flow Mapper, http://csiss.ncgia.ucsb.edu/clearinghouse/FlowMapper/

[21] Tufte E. R., *The Visual Display of Quantitative Information*, Cheshire, CT: Graphics Press, 25 40-41 176-177, 1983.