Eurographics Conference on Visualization (EuroVis) (2015)
R. Borgo, F. Ganovelli, and I. Viola (Editors)

A Survey of Multi-faceted Graph Visualization

Steffen Hadlak†[1], Heidrun Schumann[2], and Hans-Jörg Schulz[1]

[1]Fraunhofer IGD Rostock, Germany
[2]University of Rostock, Germany

Abstract

Graph visualization is an important field in information visualization that is centered on the graphical display of graph-structured data. Yet real world data is rarely just graph-structured, but instead exhibits multiple facets, such as multivariate attributes, or spatial and temporal frames of reference. In an effort to display different facets of a graph, such a wealth of visualization techniques has been developed in the past that current surveys focus on a single additional facet only in order to enumerate and classify them. This report builds on existing graph visualization surveys for the four common facets of partitions, attributes, time, and space. It contributes a generic high-level categorization of faceted graph visualization that subsumes the existing classifications, which can be understood as facet-specific refinements of the resulting categories. Furthermore, it extends beyond existing surveys by applying the same categorization to graph visualizations with multiple facets. For each of the introduced categories and considered facets, this overview provides visualization examples to illustrate instances of their realization.

Categories and Subject Descriptors (according to ACM CCS): I.3.3 [Computer Graphics]: Picture/Image Generation—Line and curve generation

1. Introduction

Data often combines various aspects, such as spatial and temporal frames of reference, or multiple attribute values per data item. We term such data "multi-faceted" in accordance with Kehrer and Hauser, who introduced this term to describe this characteristic for scientific data [KH13]. The importance and interplay of data facets are reflected in tailored visualization techniques for multi-faceted data. In this report, we provide an overview of visualizations specifically tailored to multi-faceted graph-structured data.

All visualizations for graph-structured data have in common that they encode in some form the **graph's structure** – i.e., its nodes and edges – as this sets it apart from other kinds of visualization. There are different ways to systematize graph layouts and visualization methods. The most prevalent one is to categorize them according to algorithmic considerations [BETT99, HMM00, KW01, Tam13], but there also exist categorizations according to the input data (trees vs. networks, directed vs. undirected, etc.) [BETT94], to the principal visual encoding of the output visualization [vLKS*11, SS06], and to the user tasks the visualization supports [APS14].

On top of the structure, additional facets of graphs are frequently included in their visualization. Existing surveys on graph visualization commonly focus on one additional facet to be shown, while other facets are considered as secondary constraints or subtypes:

- The **graph's partitions** – i.e., any grouping or clustering of the nodes and/or edges – are explicitly addressed as part of the research on *Compound Graph Visualization*, for which a number of common layout techniques exist that handle the specifics of partitioned graphs [BC01, VBW15].
- The **graph's attributes** – i.e., its node properties and edge weights – play a fundamental role in the field of *Multivariate Network Visualization* that treats them as a representation challenge in their own right [KPW14].
- The **graph's dynamics** – i.e., its time-varying structure – is the subject of the field of *Dynamic Graph Visualization* [BBDW14, KKC14a]. The distinction between static and dynamic graph visualization is considered a primary visualization challenge [Che06].
- The **graph's spatialization** – i.e., fixed node positions and sometimes even fixed routes for its edges – is usually considered a subdomain of *Cartography* and most of the literature on (geo)spatial graphs has appeared in this context [Rod05, Wol13].

† corresponding author: steffen.hadlak@igd-r.fraunhofer.de

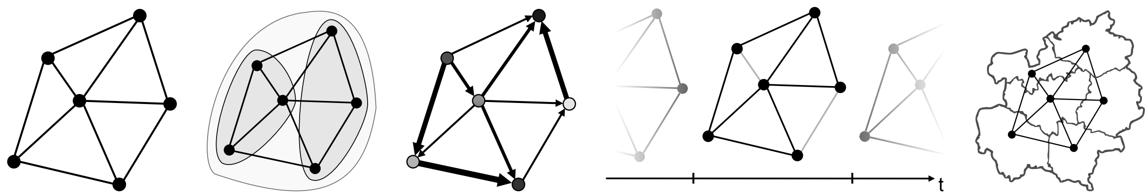


Figure 1: Facets of graph-structured data that are commonly included in graph visualizations.

As a result, the mentioned overview articles and surveys provide to a large degree targeted classifications for the particular facet on which they focus. This stands in contrast to many scenarios in which multiple facets of a graph are displayed – e.g., visualizations of attributed, spatio-temporal graph structures as they occur for example in wireless mesh networks [HSS11, HSCW13]. These multi-faceted scenarios require a broader view of graph visualization that incorporates the commonly separated facets of graph-structured data.

In this report, we aim to reconcile these different perspectives with each other, thereby contributing a high-level overview or "meta-survey" of the mentioned existing surveys for faceted graph visualization. Our main thesis underlying this report is that each facet of a graph can conceptually be considered as being visualized separately and then being composed into a final multi-faceted visualization. This compositional viewpoint resonates to some degree with most of the existing surveys – most prominently in Kerracher et al.'s design space for dynamic graph visualization [KKC14a]. In Sec. 2, we distill this idea into a systematization approach that allows us to create a uniform categorization of faceted graph visualizations for a diverse set of facets. Sec. 3 presents such a categorization for combinations of the graph layout with representations of one additional facet out of the four data facets that are commonly encountered with graph data: partitions, attributes, temporal and spatial context. These are schematically depicted in Fig. 1. This categorization effectively reframes the classifications put forward by the existing surveys in a consistent and relatable schema. As Sec. 4 shows, our composition approach also allows us to go beyond the existing surveys of single-faceted graph visualization by combining them into graph visualizations with multiple different facets. Finally to span the full breadth of multi-faceted graph visualization, Sec. 5 discusses visualizations of multiple instances of the same facet, before Sec. 6 concludes this report by highlighting some open research questions.

2. Our Systematization Approach

There exist countless instances of visualization techniques for multi-faceted graph data, which makes it impossible to survey these techniques one by one in this report. To solve this problem, we used a systematization approach that is:

- *output-oriented* by focusing on the visual result and thus abstracting from other aspects, such as the different ways to produce them (algorithmics) or to use them (user tasks),
- *high-level* or *generic* by abstracting from the concrete visual displays of individual facets and describing their composition instead, and
- *exemplifying* visualizations for each composition and discussing them in detail, rather than trying to list them all.

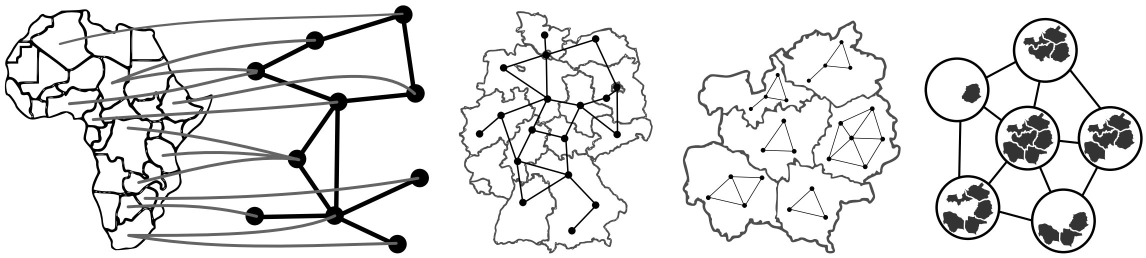
These aspects are detailed in the following in order to clarify the methodology we use in this report.

2.1. An Output-oriented View on Graph Visualization

In graph visualization, the focus lies traditionally on the algorithms that produce a graph layout with their visual properties being a-priori constrained (e.g., uniform edge lengths) or a-posteriori measured and optimized (e.g., number of edge crossings). This focus stems from the inherent complexity of the graph layout problem, which is intractable in case of contradicting visual constraints and still remains NP-complete when prioritizing them [DPS02] or when reducing the problem to a simpler graph class, such as trees [MS04]. Thus a lot of research is devoted to developing layout heuristics that reduce the problem's algorithmic complexity while maintaining a high visual quality of the outcome.

Over the last decade, this focus has slightly shifted towards considerations of the utility of the generated graph layouts for various user tasks. This is only natural, as different tasks impose different requirements on the visualization. For example, an interactive exploratory traversal of a network requires other visual properties than a static overview for its presentation. Hence, recurring tasks for graph visualization have been identified [LPP*06] and subsequently refined for the various graph facets – e.g., for partitioned graphs [SSK14], for multivariate graphs [PPS14], and for dynamic graphs [APS14, KKC14b].

In order to handle this variety of the diverse layout algorithms for generating and the numerous ways of using graph visualizations, we recognize that end users are mainly concerned with the final "look-and-feel" of a visualization We thus adopt an output-oriented perspective that aims to systematize the observable visual encodings, instead of the possibly large number of different ways to produce it or to use it.



(a) juxtaposition (b) superimposition (c) nesting

Figure 2: Composition mechanisms for two facets in display space exemplified for different graph structures and their geospatial context. Compositions realize different ways of putting focus on a representation: while the *juxtaposition* (a) provides a balanced view, the *superimposition* (b) features a clear, underlying base representation that determines the positioning for all overlaid facets. This is also the case for *nesting* (c), as the nested facet has to obey the positioning and space constraints of the base representation. Depending on the base representation, the possible utilization of the outcome can differ significantly. For example, both instances of nesting (c) show the same data – yet, the nested subgraphs on the left convey which nodes belong to a given geospatial area, whereas the nested areas on the right show which areas belong to a given node.

2.2. A High-level Systematization through Composition

As the graphical encoding is largely different from facet to facet, each facet would require its own categorization, based on their respective graphics. This is basically the abstraction level used by most of the existing surveys, which focus on a single additional facet – e.g., the surveys for dynamic graphs [BBDW14, KKC14a] use such visual distinctions. In order to provide a uniform systematization for all different data facets, we adopt a higher-level approach that focuses on the different ways for combining visualizations of graph facets instead of what these visualizations are. There are a number of possible combinations and we discern them by their *base representation* and *composition modality*.

The **base representation** of a combination names the primary graph facet whose depiction governs the central aspects of the composited visualization, whereas any other facet is merely added onto this base representation. For example, when combining the graph structure with its attributes, the base representation can either be a graphical layout of the structure with added visual cues for the attributes; or it can be a multivariate visualization of the attributes that is enhanced with edges to show the structure. The base representation can be more or less pronounced, but can usually be determined if the graph is large enough to prevent equal treatment for each facet due to limited screen space. Whereas for smaller datasets, it may be possible and desirable to show all facets in a balanced way without a discernable focus on one of them. This is either done by placing the representations of both facets in multiple coordinated views [Rob07] and connecting them via linking and brushing [BMMS91], or both facets are considered to be nodes of a bipartite graph [ADH98] and interleaved in a combined visualization.

The **composition modality** denotes whether the combination of the facets is realized through a *spatial composition* that utilizes the display space, or through a *temporal composition* that utilizes display time. For both composition modalities exists a variety of realizations. For the **spatial composition** of visualizations, the literature enumerates different possibilities, such as juxtaposing two visualizations, superimposing them, or nesting them [JE12, GAW*11]. Fig. 2 illustrates these three composition methods for the case of different graph structures being combined with a geospatial facet. For the **temporal composition**, notable results in the direction of establishing an agreed-upon set of possible realizations have been made [KK95, Fis10]. In principle when combining two facets in display time, we iterate over the elements of one facet and display the corresponding other facet. For example, in dynamic graph visualization, we could either iterate over the temporal facet and show the graph structure present at each time point to analyze the *dynamic network*, or we could traverse the structural facet and highlight all time points at which a node is present to investigate the *network dynamics*. We discern between two degrees of freedom for steering such an iteration: *predefined compositions* run automatically, as in an animation, and freely *adjustable compositions* allow the user to determine the sequence of views on the fly via appropriate GUI controls.

In general, spatial and temporal composition are challenging and despite much research on them, there remain a number of hard algorithmic questions to which no final answers have been found yet. A common problem of the spatial composition is edge clutter of a superimposed or nested graph structure that obscures the underlying base representation. The most prominent solution to this problem is the use of edge bundling [ZXYQ13]. Whereas the most challenging problem of the temporal composition is to create a base representation that remains coherent over the course of the iteration and preserves the mental map [Bra01]. This is usually achieved by confining layout changes to only those local regions in which underlying data changes occur, while keeping the global layout stable [BIM12, MELS95]. In case of both composition modalities being used for different facets, it can also occur that both challenges must be addressed concurrently – for example in the form of an edge bundling that is stabilized over the course of an animation [HEF*14].

2.3. Exemplification instead of Enumeration

While other surveys aim to provide a complete enumeration of existing visualizations (e.g., [Sch11]), this is impossible for multi-faceted graph visualization techniques, as there are simply too many. This is underlined by the fact that each of the considered facets forms the subject of its own domain in visualization:

- Displaying partitions and clusters is addressed in *Euler diagrams* [Rod13] and *set visualization* [AMA*14].
- Showing (numerical) attributes of data is the concern of *multivariate visualization* [WB97, FH09].
- Representing dynamic data is the challenge of *visualization for time-oriented data* [AMST11, Wil12].
- Depicting spatial data is mainly understood as showing **geo**spatial data and addressed in *cartography* [KO10] and *geographic visualization* [DMK05, DMT08].

The combinations of the many individual techniques these domains comprise into multi-faceted visualizations is sheer endless and modern multi-view visualization systems can easily be configured to produce hundreds of different combinations. Therefore, we concentrate on only a few examples per combination possibility and facet, which allows us in turn to provide a more detailed description of them. They stand as representatives for other, often similar visualization techniques that fall into the same category.

3. Visualizations of the Graph Structure and One Additional Facet

In this section, we present the proposed categorization by systematically discussing all possible combinations of a visualization of the graph structure (G) with a visualization of **one** additional facet ($*$). Together with the composition mechanisms introduced in the previous section, this generates the following five combinations:

Spatial composition of two facets means that they must both be accommodated in the same visualization, with the base representation determining the principal visual encoding. We denote this composition by straight arrows – a one-sided arrow $\leftarrow$ pointing towards the base representation in which the other one is incorporated, and a double-sided arrow $\leftrightarrow$ when both representations are combined in a balanced way without a particular focus on one of them.

- **[$G \leftarrow *$]** Using the **graph structure** in the base representation implies that an underlying graph layout is enhanced via superimposition or nesting with a visual representation of the respective other facet.
- **[$G \leftrightarrow *$]** A balanced display of two facets can be achieved by juxtaposing them – either with each facet in its own display space as *linked views* or by interleaving them in the same display space as a *bipartite graph*.
- **[$* \leftarrow G$]** Using the **other facet** in the base representation implies that this facet is visualized and the graph structure is added onto it by means of superimposition or nesting.

Temporal composition of two facets means that we only show the base representation and iterate over the other facet. As the facet over which the iteration runs remains mostly invisible, there cannot be a balanced composition in display time, because the visible base representation clearly determines the visual appearance. Another difference to the spatial composition is that the visualization is not produced once, but reproduced for each iteration step. This aspect of an iteratively changing display is embodied in the circular arrow $\circlearrowleft$ that we use to denote this composition.

- **[$G \circlearrowleft *$]** Using the **graph structure** in the base representation implies that the user iterates over the respective other facet and sees the corresponding parts of the structure – e.g., looping through a list of spatial regions and showing the subgraph(s) that belong to each region.
- **[$* \circlearrowleft G$]** Using the **other facet** in the base representation implies that the user traverses the graph structure by iterating over the nodes and/or edges and thus adapts the display of the other facet to show only those items that relate to it – e.g., stepping through the nodeset of the graph to see which spatial regions they lie in.

In the following, these five types of composition are applied to the four facets of partitions ($* = P$), attributes ($* = A$), time ($* = T$), and space ($* = S$). The resulting categorization for each facet is compared to the classifications derived by other surveys for the individual single-faceted graph visualizations.

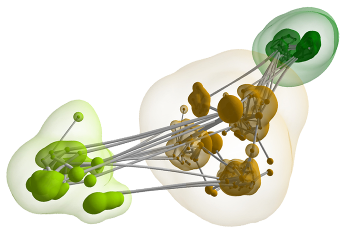
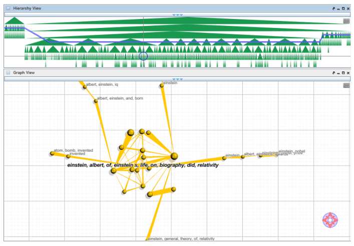
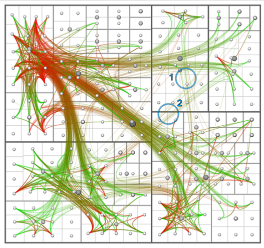
3.1. Partitions: Compound Graph Visualization

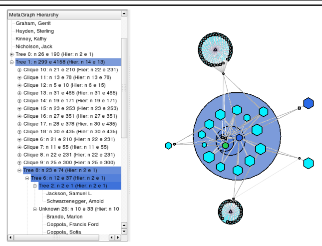
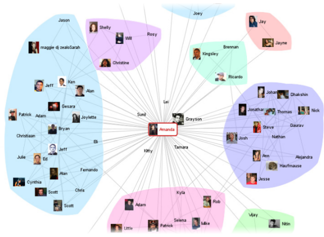
Combining the graph structure and its partitioning in one visual display is often done to show graphs that would be too large to be shown in an unclustered, fully detailed way. Such a partitioning can either be a classification that is given with the graph-structured dataset or a computed clustering for which a wide range of methods exists [Sch07]. Often, partitions are given and shown in form of a hierarchy (e.g., a *dendrogram*) with larger partitions including smaller ones.

3.1.1. Spatial Compositions

[$G \leftarrow P$] **Graph structure as base representation:** The first figure in Table 1 shows the level-of-detail visualization technique [BD07] that takes a given 3D layout of the graph structure and constructs semi-transparent implicit surfaces around groups of nodes of the same partition. Note that in order to yield discernable clusters, the underlying layout must separate them well [NL05]. The nodes are classes of an object-oriented software and the edges denote method calls among

Table 1: Examples for combinations of structure and partitions in one visualization.

	structure as base representation	balanced representation	partitions as base representation
spatial composition	 [$G \leftarrow P$] Level-of-Detail Visualization [BD07]	 [$G \leftrightarrow P$] Coordinated Graph Visualization [TAS09]	 [$P \leftarrow G$] Hierarchical Edge Bundling [Hol06]

	structure as base representation	partitions as base representation
temporal composition	 [$G \circlearrowleft P$] Grouse [AMA07]	 [$P \circlearrowleft G$] Vizster [HB05]

them. The partitions are formed by the inheritance hierarchy in which the classes are embedded. Roughly speaking, the transparency of each implicit surface is increased the closer it is to the viewpoint and decreased the further away it is. This way, surfaces, which are too close, become invisible as they are completely transparent; and surfaces, which are too far away, become invisible as they are occluded by an already opaque "parent surface". Only surfaces at a certain distance are visible, which corresponds to a horizontal cut through the partitioning hierarchy – i.e., a *maximal anti-chain* [AvHK06].

[$G \leftrightarrow P$] **Balanced representation:** The second figure in Table 1 shows the graph structure and its partitioning in two adjacent views as they are provided by the CGV system [TAS09]. The dataset is a network of search queries (the nodes) that are linked via edges if they occurred within the same period. Its partitioning was computed using the Markov clustering algorithm [vDAG12] and the clusters are shown as *metanodes* in the graph view. Both views are linked, so that selecting a cluster in the hierarchy will change the graph visualization to bring it into view. Whereas selecting a metanode in the graph view will unfold its contained (meta-)nodes, corresponding to a *drill-down* operation in the partition hierarchy [EF10] that moves the anti-chain downward. Further examples are the ASK-GraphView system [AvHK06] and the TreeMatrix technique [RMF12], which combine representations of the partitioning with a matrix visualization of the graph structure.

[$P \leftarrow G$] **Partitions as base representation:** The third figure in Table 1 shows a software class hierarchy whose inheritance relationship (i.e., the partitions) is laid out as a treemap [JS91] first and then the method calls among the leaves are added in the form of edges. These edges are further bundled [Hol06] to reduce the visual clutter that is generated by such a superimposition [FWD*03]. In contrast to the two previous combinations, this technique prevents overcrowding of the visualization by bundling the edges into metaedges, but leaves the individual nodes visible and does not combine them into metanodes. Another example is the SWViz application [ACJM03] that represents the clustered graph by a hierarchical nesting of node-link visualizations. Connections between nodes from different clusters are neglected and instead visualized as edges between the clusters.

3.1.2. Temporal Compositions

[$G \circlearrowleft P$] **Graph structure as base representation:** The fourth figure in Table 1 shows the visualization tool Grouse [AMA07]. It displays parts of the Internet Movie Database where the nodes are actors, and the edges denote that two actors have appeared in a movie together. The left side shows a tree view of a computed cluster hierarchy, effectively presenting a GUI control with which users can steer the iteration over the clusters, changing the shown graph structure step by step. The right side shows the graph structure within the selected cluster in detail in the center and all connected clusters as metanodes in its context. A further example is the GMine system [RTT*06] that also follows the idea of focusing on a specific cluster for the exploration. To help the visual inspection of a cluster, distortion techniques such as semantic fisheye views [LEH05] can be employed to increase the display space of the cluster in focus.

[$P \circlearrowleft G$] Partitions as base representation: The fifth figure in Table 1 shows Vizster [HB05], a visualization system for social network exploration. In a social network, nodes are persons and edges denote friendship, kinship, or any other social relation between two persons. Its initial view selects a current user and displays all persons who are directly connected to her – i.e., her so-called *ego-centered network* [WF94, ch.2.3.3]. Depending on their interconnections, these "friends" are partitioned into communities and color-coded. The user can interactively traverse the graph by selecting another person (node) among the shown ones, which will then become the center of a new ego-centered network and show its communities. This way, one can iterate in a step-wise, user-adjusted fashion over the graph structure.

3.1.3. Relation to Existing Surveys

While one of the existing overview articles on compound graph visualization [BC01] is solely concerned with the algorithmic side of the layout generation and thus cannot be directly related to our five categories, the other one [VBW15] shows interesting parallels:

- [$G \leftarrow P$] encompasses the classes *contour overlay* and *line overlay* that draw the graph first and superimpose the partitions as Jordan curves or simple lines, respectively. Furthermore, it captures the very similar class called *embedded*, which basically adds metaedges among the partitions, thus turning them into nodes (i.e., metanodes) themselves. Finally, it also subsumes the class of *visual node attributes*, which denotes a nesting of more or less elaborate representations of partition membership in the nodes – e.g., in the form of glyphs.
- [$G \leftrightarrow P$] maps directly to the class called *juxtaposed* in which each facet is drawn in a dedicated part of the drawing area – either in a completely separate way or in a synced manner that aligns the positioning of both facets.
- [$P \leftarrow G$] relates directly to the class called *partitioning* that maps the partitions onto space-filling partitions of the drawing area and superimposes the graph accordingly.

Our categories [$G \circlearrowleft P$] and [$P \circlearrowleft G$] are not captured by this survey, as it does not consider the iteration over the partitions or the graph structure via animation or user interaction.

3.2. Attributes: Multivariate Graph Visualization

Showing the graph structure together with node attributes and/or edge weights is a prevalent theme in graph visualization. As it was the case for partitions, such attributes can either be part of the dataset or they can be computed for all nodes/edges, usually to quantify local aspects of the graph structure [WFC*06]. While a single attribute is easily mapped onto a visual variable such as node size or color, the incorporation of multiple attributes can be realized as follows.

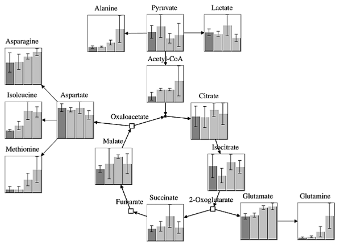
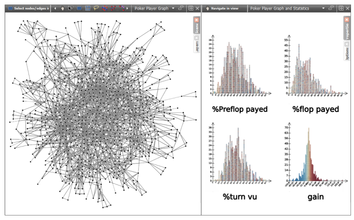
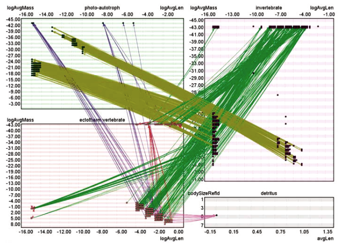
3.2.1. Spatial Compositions

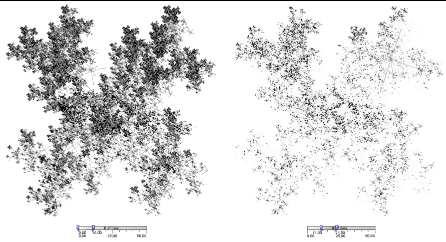
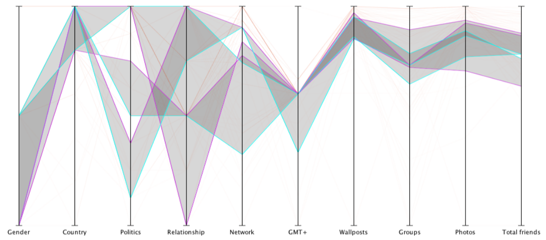
[$G \leftarrow A$] Graph structure as base representation: One of the most common compositions is the use of a standard graph layout for the structure and embedded attributes within the nodes. This is shown in the first figure in Table 2 with a visualization taken from the DBE-GRAVISTO network analysis and graph visualization system [BHK*05]. The graph structure represents parts of the metabolic network of the Narbon bean and the shown detail represents the citric acid cycle. For each metabolite (node) of this network, different attributes are shown in an embedded bar chart. These are measured substance levels for each metabolite in different lines of the bean, with dark gray being the wild type and light gray being several transgenic lines. Other examples of this category employ various kinds of glyph-based attribute visualizations to embed in nodes [ME09] or edges [EDG*08].

[$G \leftrightarrow A$] Balanced representation: This spatial composition through linked views is probably the most straightforward one, as it does not require additional effort in generating a new visualization, but can be realized by means of two standard visualizations – a layout of the graph structure and a multivariate visualization of the attributes. The second figure in Table 2 shows the Tulip graph visualization system [Aub04] with the graph structure on the left side and a series of histograms on the right side. The dataset in this example is a poker network in which the nodes are poker players and the edges denote that one player lost and paid to the other. The node attributes shown in the histograms include common player metrics, such as the number of hands played and the total gain obtained, and edge weights denote the amount of money paid. Besides linked views, there also exist techniques that interleave both facets in the same display space. For example in JauntyNets [JKZ13], attribute values are handled as an additional nodeset and each of these "value nodes" is connected to the graph nodes exhibiting its corresponding attribute value.

[$A \leftarrow G$] Attributes as base representation: The third figure shown in Table 2 depicts a food web using semantic substrates [AS07]. Nodes represent different species and directed edges are predator to prey relations. For each species, attributes are given for its metabolic category, its average mass and length. The layout uses these attributes to position the nodes. In the example, it first defines regions of the drawing space according to the metabolic categories. It then put a scatterplot into each region, using average length as x-axis and average mass as y-axis. Each node can thus be placed according to its attributes. As such a positioning often introduces overplotting between nodes having the same attribute values, it is also possible to relax the layout by applying a node overlap removal [TS13]. Similar techniques to semantic substrates are PivotGraph [Wat06] and GraphDice [BCD*10].

Table 2: Examples for combinations of structure and attributes in one visualization.

	structure as base representation	balanced representation	attributes as base representation
spatial composition	 $[G \leftarrow A]$ DBE-GRAVISTO [BHK*05]	 $[G \leftrightarrow A]$ Tulip [Aub04]	 $[A \leftarrow G]$ Semantic Substrates [AS07]

	structure as base representation	attributes as base representation
temporal composition	 $[G \circlearrowleft A]$ Point-based Representation [SHS11]	 $[A \circlearrowleft G]$ Paired Parallel Coordinates [SHQ08]

3.2.2. Temporal Compositions

$[G \circlearrowleft A]$ **Graph structure as base representation:** The two views shown in the fourth figure in Table 2 were produced using a point-based representation that layouts a tree in a very space-efficient manner [SHS11]. By adjusting the GUI control underneath the visualization, different ranges of interest for an associated attribute can be set to iteratively adjust the displayed structure to only show the nodes of the hierarchy whose attribute values are within that range. In the example, the shown hierarchy is a large hierarchical topic index for websites and the attribute mapped on display time is the number of weblinks in each of its categories.

$[A \circlearrowleft G]$ **Attributes as base representation:** The fifth figure in Table 2 shows paired parallel coordinates [SHQ08] – a parallel coordinates visualization of node attributes. The user can select nodes in an adjacent display (not shown in the figure) and for each node, a polyline is added to the parallel coordinates. If an edge connects two of the shown nodes, the space between their respective polylines is colored gray. The figure shows nodes and edges from a social network with associated node attributes, such as the number of posted photos, friends, wall posts, and groups of which they are a member. The user can traverse the graph structure node by node or edge by edge to get a step-wise impression of their attributes. Another example for this category is the network lens [JDK10] for traversing the graph structure and showing the associated attributes next to the nodes within a lens.

3.2.3. Relation to Existing Surveys

The classes from the recent overview on multivariate network visualization [KPW14, ch.1.2] relate in part to our categories:

- $[G \leftarrow A]$ includes the *integrated approaches* that nest attributes in the nodes of a graph layout.
- $[G \leftrightarrow A]$ covers the *multiple and coordinated views*.
- $[A \leftarrow G]$ corresponds to the *semantic substrates* and *attribute-driven layouts*, which are more fine-grained distinctions within this category.

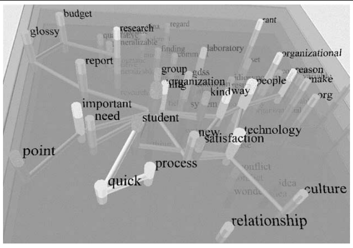
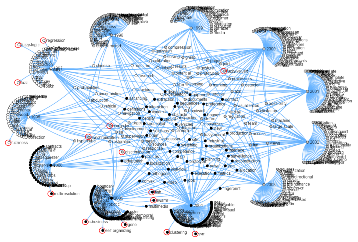
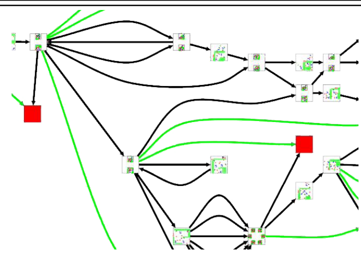
In addition, the recent overview also features the visualization class *hybrid approaches*, in which at least two of the other classes are combined – e.g., *semantic substrates* and an *integrated approach*. Yet, the two facets of structure and attributes can only be integrated once via either one of these classes. Thus *hybrid approaches* occur either when other facets such as time or partitions are added (cf. Sec. 4), or when multiple instances of facets are integrated (cf. Sec. 5).

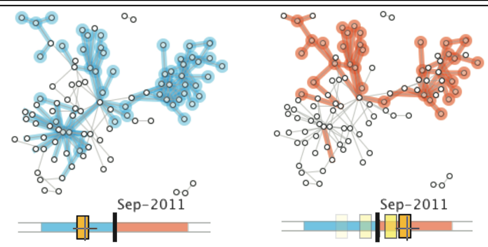
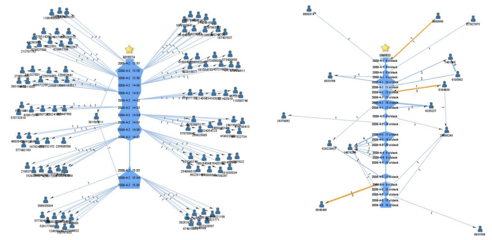
Finally, our two categories $[G \circlearrowleft A]$ and $[A \circlearrowleft G]$ are not captured by these five classes, as they incorporate interaction in the form of user-adjustable iteration, while the classification of the overview considers only "visual mappings".

3.3. Time: Dynamic Graph Visualization

Visualizing graph data that evolves over time is a major research challenge, as laying out a single graph structure is already hard, but doing it for multiple structures – one for

Table 3: Examples for combinations of structure and time in one visualization.

	structure as base representation	balanced representation	time as base representation
spatial composition	 [$G \leftarrow T$] Visual Unrolling [BC03]	 [$G \leftrightarrow T$] Topic Shift Visualization [TDKB07]	 [$T \leftarrow G$] Porgy [PMD12]

	structure as base representation	time as base representation
temporal composition	 [$G \circlearrowleft T$] GraphDiaries [BPF14]	 [$T \circlearrowleft G$] 1.5D Visualization [SWW*15]

every time point – in a coherent and discernable way is ultimately harder. That is why a large body of research exists on this topic with a variety of different visualization solutions.

3.3.1. Spatial Compositions

[$G \leftarrow T$] **Graph structure as base representation:** The first figure in Table 3 shows a 2.5D visualization [BDS04] that maps the temporal facet onto the third dimension. It basically stacks semitransparent 2D graph layouts on top of each other in their temporal order. As a result, the layouts corresponding to earlier time points are barely visible at the bottom of this stack, while the most recent at the top are well discernable. The shown dataset is of a discussion, where the nodes are nouns extracted from statements being made and the edges denote whether the nouns were used in the same sentence. The timeline is simply the progress of the discussion with each time step being a full statement of a participant of the discussion. The idea of such a 2.5D stacking is very common and can be found in multiple visualizations [GW06, BPF14]. Another common way to realize this particular composition is via nesting of the temporal facet as a line plot inside the nodes or edges in the same sense as it is done for attributes [SLN05]. The pixel matrix [SWS10] uses this idea to encode the presence or absence of edges at various time points in a matrix display.

[$G \leftrightarrow T$] **Balanced representation:** The second figure in Table 3 interleaves both facets in the same drawing space, instead of juxtaposing their independent views. This is done by showing not only the nodes of the graph, but also encoding the time points as nodes of a different kind and connecting each "time node" with the graph nodes that are present at the respective time point. The figure shows a topic shift visualization [TDKB07] of a corpus of scientific paper abstracts. The visualization contains a number of terms derived from these abstracts as graph nodes and their respective years of publication as "time nodes". A term is connected to the year(s) in which it occurred in an abstract. Note that the visualization does not feature any edges among the graph nodes, as the evolution of the edges is hard to encode likewise – for example, by linking each edge of the graph structure via "time edges" to certain "time nodes", i.e., years.

[$T \leftarrow G$] **Time as base representation:** The temporal facet as an underlying base is mostly utilized by *small multiples* [Tuf90] that use a linear or grid-structured tiling of the display space to encode the time points. This underlying grid serves as the base representation in which the graph structure is embedded at each time point. Techniques that use such small multiples are, for example, DiffAni [RM13] and MatrixFlow [PS12]. Yet this particular combination becomes even more useful when the time is not simply linear (i.e., a time**line**), but complex enough to be explicitly displayed itself – e.g., in the case of branching time. The third figure in Table 3 displays a part of such a visualization from the Porgy environment [PMD12] that layouts time in the form of a state-transition-diagram and embeds the graph structure at each state. It is used for graph rewriting systems that apply rewriting rules (transitions) to certain states. In the shown case, the graph structure is a biochemical network of proteins and interactions between them. Through this combination and some additional compression, such as shortening paths of transitions that do not branch, all possible state transitions for a network are shown in one visualization.

3.3.2. Temporal Compositions

[$G \circlearrowleft T$] Graph structure as base representation: This category stands for the common mapping of "time onto time". The fourth figure in Table 3 shows GraphDiaries [BPF14], a visual interface that combines a layout of the graph structure at a given time point with an interactive slider as GUI control to steer which time point is shown. The temporal stability of the layout is achieved by using a common initial layout from which the individual layouts are derived for each time point. Both figures show a co-authorship network at a selected point of its evolution. In the left figure, nodes are color-coded in blue, which have been added since the previous time point; whereas in the right figure, nodes are color-coded in orange, which will be removed at the next time point. There are many further examples building on graph animations that mainly differ in the strategy for maintaining the layout's stability and thus the user's mental map [BIM12, Sec.3].

[$T \circlearrowleft G$] Time as base representation: The fifth figure in Table 3 shows a 1.5D visualization [SWW*15] that displays the timeline for a selected node in the center of the view. Other nodes are connected with lines to the respective time points at which these nodes were direct neighbors of the selected node. By traversing the graph structure and selecting different nodes, their connectivity over time can be investigated in a step-wise manner. The figure depicts a communication network, where the nodes are mobile phone users and the edges are SMS texts. By looking at the connection patterns over time, spammers can be identified, as they connect to a different set of users at each time point (left figure) instead of repeatedly connecting to mainly the same group of people, like a regular user (right figure). Instead of its adjacent nodes, one can also show a node's incident edges over time [Rei10].

3.3.3. Relation to Existing Surveys

For the dynamic graph visualization surveys [BBDW14, KKC14a], we can find the following correspondences:

- [$G \leftarrow T$] incorporates the classes *superimposition* [BBDW14, KKC14a], *layered* [BBDW14], and *additional spatial dimension* [KKC14a], which all denote stacked combinations in some shape or form. In addition, the classes *nested* [KKC14a] and *intra-cell* [BBDW14] also fall into this category as they capture the embedding of the temporal facet in general or in matrices, respectively.
- [$G \leftrightarrow T$] includes the classes *integrated* [BBDW14] and *time as node* [KKC14a], which subsume combined and tightly interwoven visualizations of both facets.
- [$T \leftarrow G$] corresponds to the class of *juxtaposition* [BBDW14, KKC14a], which is comprised of visualizations using small multiples.
- [$G \circlearrowleft T$] relates directly to the classes *animation* [BBDW14] and *sequential views* [KKC14a].

In addition, a class of visualizations called *merged* [KKC14a] is given, which describes a transformation into the domain of multivariate graph visualization by computing a *union graph*, as it is discussed in Sec. 3.5.

Our category [$T \circlearrowleft G$] appears in neither of the two surveys, probably because only few visualizations exist in this category. A reason for this underrepresentation may be that these visualizations violate the *congruence principle* [TMB02], which states that it is more intuitive to map "time onto time" and "space onto space" than the other way around.

3.4. Space: Spatial Graph Visualization

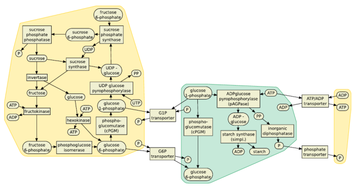
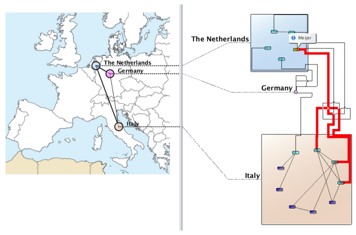
A spatial facet is usually thought of as a geospatial frame of reference for the data, which can be given at various levels of granularity: from all positions being fixed at point locations to mere inclusion relations with a set of areas or regions. On top of that, other forms of spatial facets have recently emerged – for example for graphs from the biomedical domain, the spatial context can also be the organs of the human body or the compartments of the biological cell, instead of countries of the world. This detaches the tight relation of visualization of spatial facets to the cartographic domain and opens up new representation possibilities as they are among the following examples, as well.

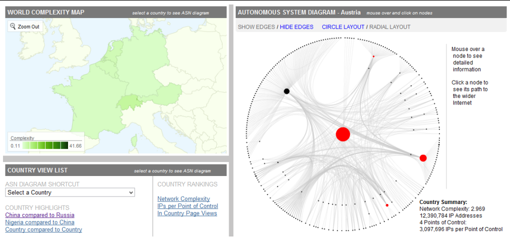
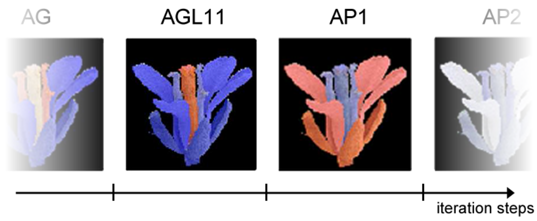
3.4.1. Spatial Compositions

[$G \leftarrow S$] Graph structure as base representation: The first figure in Table 4 has been generated with the generic layout algorithm [SDMW09]. It first draws the graph structure and then superimposes spatial groupings through set visualization mechanisms, as we have discussed them in Sec. 3.1. The figure depicts a metabolic pathway with a bipartite graph structure (nodeset$_1$= compounds, nodeset$_2$= reactions) and denotes the involvement of a compound in a reaction with an edge. Each reaction is associated with a particular cell compartment in which it occurs, including transport reactions between the compartments. Another approach is to represent the spatial facet as an embedded glyph [JRS12].

[$G \leftrightarrow S$] Balanced representation: The second figure in Table 4 depicts the MOM system [GDLP09] showing a co-authorship network. The geospatial context is displayed on the left side with an overlaid graph that connects European countries that cooperated in co-authoring papers. The graph structure is shown on the right side with authors as nodes and joint collaboration on a paper encoded as edges. Note that these juxtaposed views are not linked via brushing and highlighting, but with static visual links [CC07] that preserve the connections between the two views even on a printout.

Table 4: Examples for combinations of structure and space in one visualization.

	structure as base representation	balanced representation	space as base representation
spatial composition	 [$G \leftarrow S$] Generic Constraint-based Layout [SDMW09]	 [$G \leftrightarrow S$] Matched One-to-Many graphs (MOM) [GDLP09]	 [$S \leftarrow G$] KDE-based Edge Bundling [HET12]

	structure as base representation	space as base representation
temporal composition	 [$G \circlearrowleft S$] Network maps [RLFP11]	 [$S \circlearrowleft G$] HIVE [JRS12]

[$S \leftarrow G$] **Space as base representation:** The third figure in Table 4 shows a visualization of flight routes over the US. The nodes of this graph are airports and the edges are flights over a given period of time. These flights were originally shown as straight lines, which produced massive edge clutter leaving almost none of the underlying map visible. To reduce this clutter, a kernel-density-estimation-based edge bundling has been applied [HET12]. For coping with different spatial scales, it is possible to use insets for first nesting detailed spatial views into spatial overviews, before overlaying it with the graph [BKA*15]. In the case that entire graphs are associated with regions of the map, embedding techniques have been developed that aim to adapt the graph layout to arbitrary spatial areas [HTSS10].

3.4.2. Temporal Compositions

[$G \circlearrowleft S$] **Graph structure as base representation:** The fourth figure in Table 4 shows a visualization of the world wide web broken down into subnetworks by country [RLFP11]. The nodes represent autonomous systems such as internet service provides or large companies handling the data transport and the edges represent consumer-provider relationships indicating a data exchange between them. The left side features a world map as an interactive overview and GUI control in which the user can select countries of interest and thus steer which subgraph to show in the detail view on the right side. This way, the user can iterate step-by-step over the spatial facet and investigate the respective subnetwork.

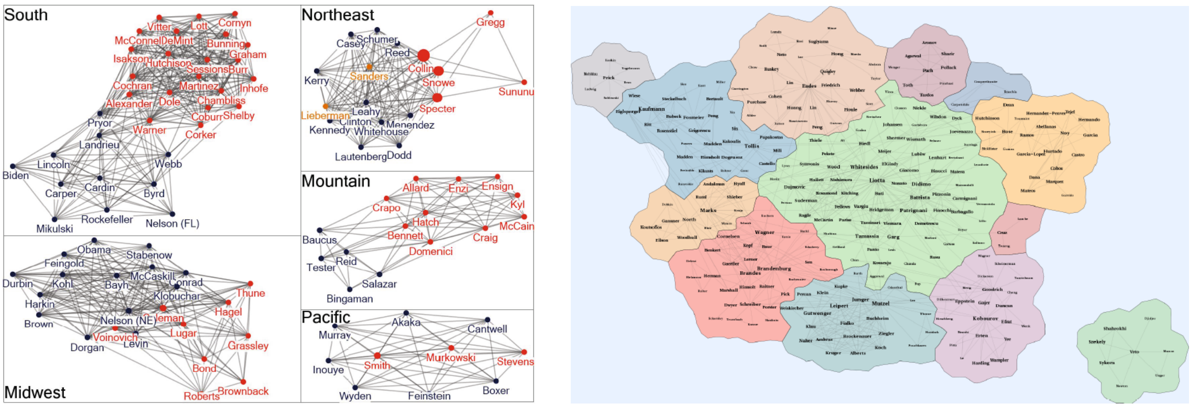
[$S \circlearrowleft G$] **Space as base representation:** The fifth figure in Table 4 shows the HIVE system [JRS12] depicting floral organs of thale cress in a 3D rendering. The associated graph structure (not shown) is a gene regulatory network of genes (the nodes) and regulatory processes, such as inhibition or activation (the edges). Selecting a node – i.e., a gene – will highlight its expression across the various floral organs in the "spatial view". Iterating through the different genes maps the graph structure to display time, showing the different regions in which the gene expression is high (red) or low (blue).

3.4.3. Relation to Existing Surveys

The few existing overview articles on this field of graph visualization [Rod05, Wol13] focus mainly on algorithmic aspects of the visualization and different structural features to be visualized, but not on the visual aspects themselves. Thus, it is not possible to establish direct relations between our five categories and the distinctions they make.

3.5. Transformational Approaches

From a few of the examples given in the previous four sections, it becomes clear that a *correspondence* exists between some of the facets. For example, nodes that belong to the same spatial region can also be thought of as belonging to the same partition (i.e., a partition based on geographical regions) or as having the same categorical attribute (i.e., the name of the region). These correspondences can be used to



(a) Group-in-a-box Layout [SD13]

(b) GMap [GHK10]

Figure 3: Visualization examples that utilize the correspondence between geographical regions of a spatial facet and sets of a partitioning: (a) The Group-in-a-box Layout [SD13] is in fact a technique for partitioned graphs. So, when clustering a given spatial graph according to its geographical regions, it can be used to show the graph as a grouped collection of subgraphs. (b) GMap [GHK10] takes a given partitioned graph and spatializes its partitions into regions of an abstract map.

adapt visualizations to one facet despite them being originally developed for another facet. This is illustrated in Fig. 3 for space and partitions. It is noteworthy that their correspondence goes both ways, albeit turning partitions into spatial regions of an abstract map is considerably harder.

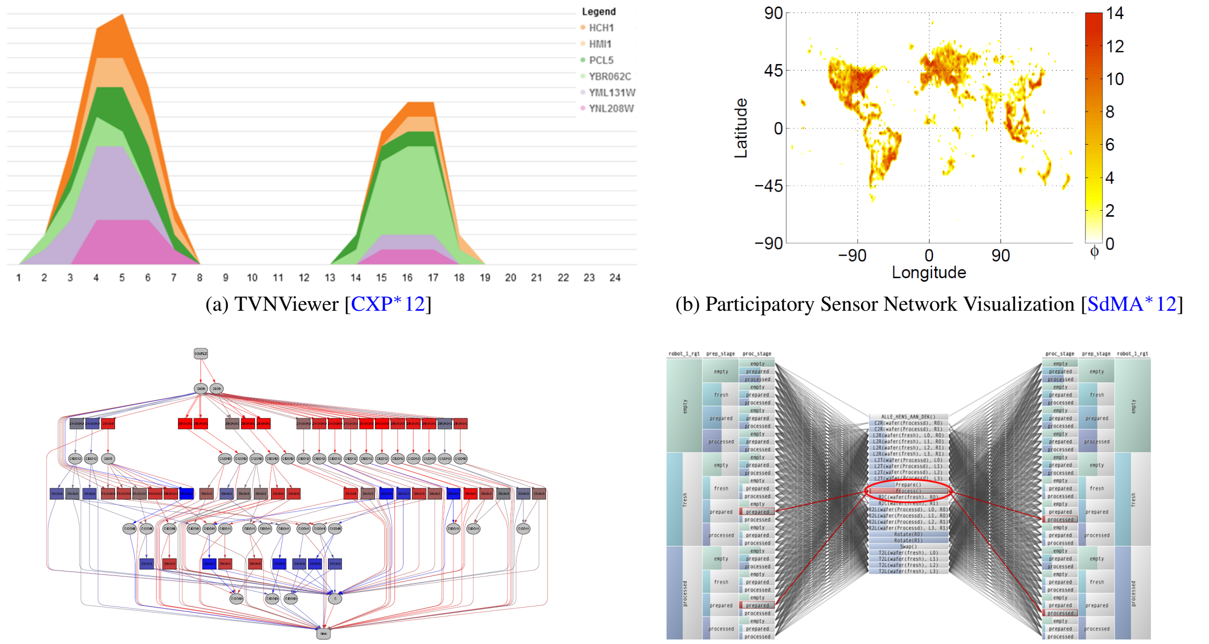
If no apparent correspondence exists, we can still *transform* one facet into another by means of computation. For example, we can express the key characteristics of the graph structure by means of structural metrics [WFC*06, BB05] and thus effectively transform the structural facet into numerical attributes. This permits displaying them with multivariate visualization techniques that do not explicitly show the graph structure at all – e.g., with scatterplots [WFC*06] or time-value plots [JSS*13]. Another possibility is to maintain the graph structure, but to transform one of the other facets to use a different kind of visualization. The most common variant is the generation of a *supergraph* or *union graph* [DGK01], transforming the temporal facet of a graph structure into an attribute: it combines all nodes and edges that exist in the graph at least at one point in time into a static graph structure and annotates them with an attribute that contains the time point they were first added to the graph. This way, it can be visualized using a multivariate graph visualization instead of a dynamic graph visualization. The union graph and other transformational approaches are shown in Fig. 4.

3.6. Summary

In sum, the categorization yielded from our combination approach and used in Sections 3.1 through 3.4 gives a comprehensive coverage of single-faceted graph visualizations. However, we should mention that in some cases it can be challenging to determine the underlying base-representation. One instance that illustrates this challenge is the force-directed layout of clustered graphs [EH00], whose end result seems equally likely to be either a nested drawing of clusters with added nodes and edges, or a node-link layout of the nodes and edges with added clusters. A reason for this is the generality of our high-level categorization that comes with its broad coverage of different facets. Many of the existing surveys of single-faceted graph visualizations feature further classes to capture such nuances. From the comparisons of our categorization with the existing surveys, one can find additional classes of visualizations in these surveys for one or more of the following three reasons:

- A number of visualization classes are actually **more detailed subdivisions** of one of our categories – for example, as they further discern between node-link diagrams and matrix displays. This can be taken as an indication that many visualization techniques exist for a particular facet and category, so that the existing classifications found it necessary to further subdivide them.
- A visualization class is mainly concerned with the **transformation of the data** so that it can be visualized with a different type of visualization altogether.
- A visualization class does not only involve a single facet, but either **multiple different facets** or **multiple instances of the same facet**.

The first of these aspects is out of scope for our report, which is focused on incorporating all facets under one categorization and does not cater towards particularities of individual facets, and the second of them has already been discussed in the previous section. This leaves the latter of these three aspects, which will be discussed in the following sections.



(a) TVNViewer [CXP*12]

(b) Participatory Sensor Network Visualization [SdMA*12]

(c) Union Graph Visualization [RUK*10]

(d) Multivariate Graph Visualization [PvW08]

Figure 4: Examples of transforming one facet into another in order to utilize approaches from different (graph) visualization domains: (a) The TVNViewer [CXP*12] transforms the structural facet of a dynamic graph into attributes by counting the number of different node types at each time point. It then displays the evolution of these numbers over time in a stacked chart. (b) The Participatory Sensor Network Visualization [SdMA*12] transforms the structural facet of a geospatial network into an attribute by determining the node count for each position on the map and displaying it as a heatmap. (c) The Union Graph Visualization [RUK*10] transforms the temporal facet into an attribute by denoting the first time point at which each node/edge appeared and color-coding these onto a static graph layout. (d) The Multivariate Graph Visualization [PvW08] transforms attributes of a graph into partitions by grouping nodes with similar attribute values, displays the partitioned nodesets side by side, and connects them if an edge exists between nodes of these partitions.

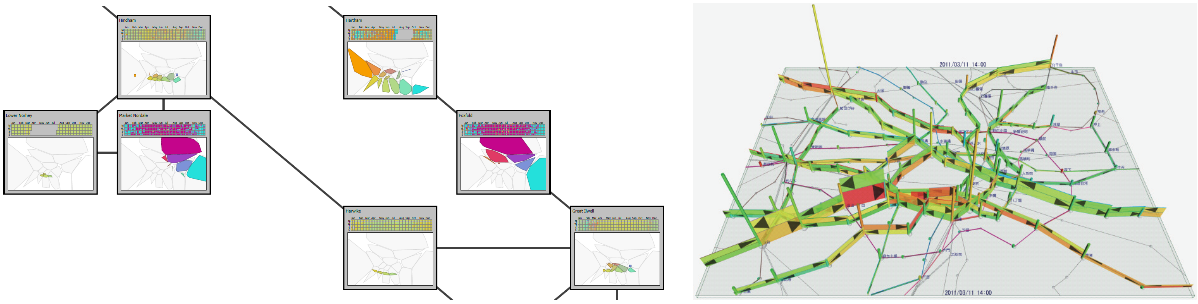
4. Visualizations of the Graph Structure and Multiple Additional Facets

Multi-faceted graph visualizations incorporate more than just a single facet on top of the graph structure. In this section, we show how they can be composed from single-faceted graph visualizations and visualizations of individual facets by applying the five combinations that we have used before.

4.1. Spatial Composition

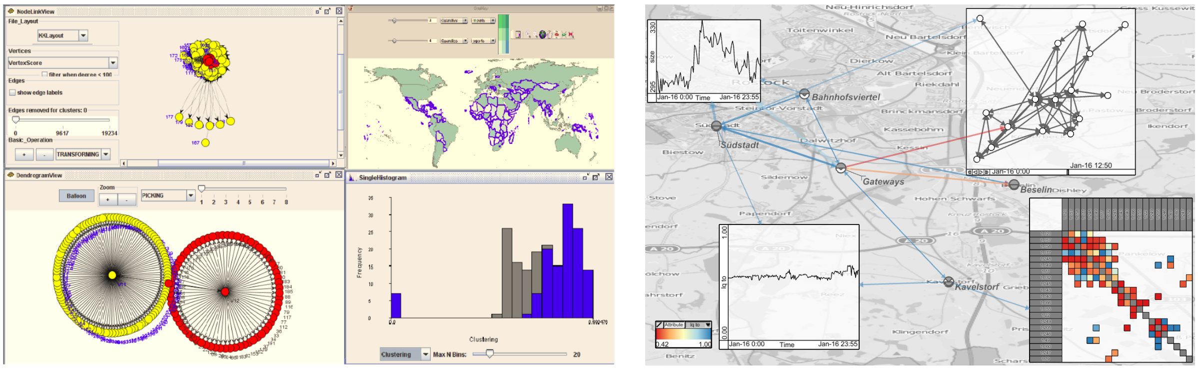
Graph structure as base representation: This combination adds one or more facets onto a base representation of the graph structure. This base representation can either be a plain graph layout consisting of the graph structure by itself or an already combined visualization that includes the structural facet alongside other facets. Fig. 5 shows an example for each of these two cases. The example in Fig. 5a shows part of an octi-linear graph layout with embedded views [SBM*14]. It depicts a sensor network in the underlying base representation G. Each node (i.e., sensor) embeds a combined calendar view of the temporal facet T and similarity visualization of a clustering P. As these embedded views take up more display space than a mere glyph, the visualization features different levels of detail, fully showing the embedded views only when the user zooms in. Similar nesting approaches have also been developed for other base representations, such as treemaps [TS07] and matrices [YEL10]. The example in Fig. 5b visualizes passenger flow (edges) between metro stations (nodes) in Tokyo (spatial reference) [IYT*13] – effectively forming a composited base representation of the graph structure and its spatial context [$S \longleftarrow G$]. On top of this representation, the third dimension is used to encode node attributes and edge weights A, which denote passenger numbers. The colors of the resulting bars and bands show how these attribute values compare to the expected values.

Balanced representation: Using multiple linked views in a balanced combination of multiple facets is the most prevalent solution if more than three facets are integrated with each other [JRA09]. Fig. 6a shows an example of a combination of four facets from a world trade network [LMYH11]. It features four linked views – one for each facet:



(a) Octi-linear layout with embedded views [SBM*14] (b) Metro Passenger Flow Visualization [IYT*13]

Figure 5: Examples for spatial compositions of multiple facets in which the graph structure is part of the base representation.



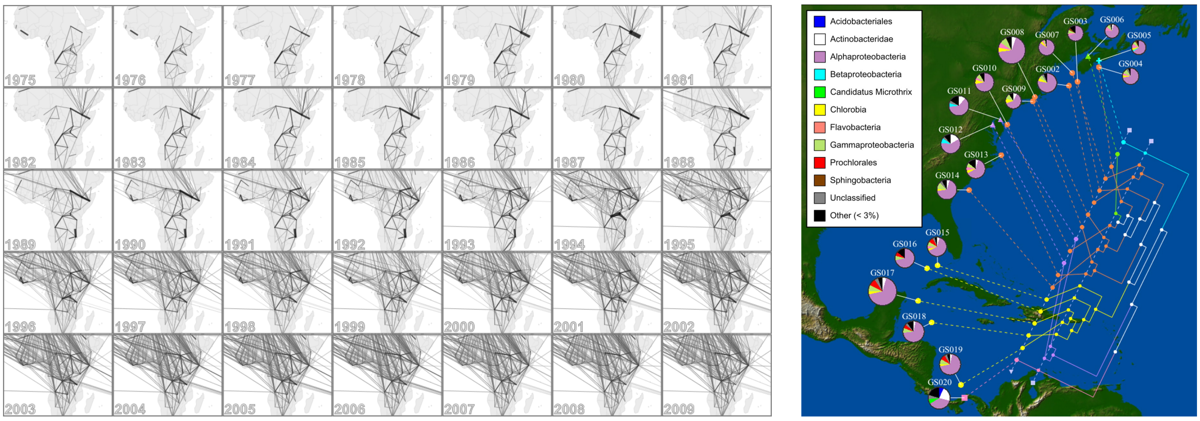
(a) World Trade Network Visualization [LMYH11] (b) In Situ Visualization [HSS11]

Figure 6: Examples for balanced compositions of the graph structure with other facets.

- the top left view shows the graph structure with countries (nodes) and trade relationships (edges),
- the bottom left view shows a partitioning of the graph by displaying its clustering hierarchy in a radial tree visualization, in this case showing two clusters – the left cluster (yellow) consists of the economically less important countries and the right cluster (red) of the more important ones,
- the top right view shows the spatial facet in a choropleth map with all of the countries from the left cluster being highlighted in purple, and
- the bottom right view shows a node attribute's distribution (in this case the clustering coefficient) and again highlights the portion of nodes from the left cluster in purple.

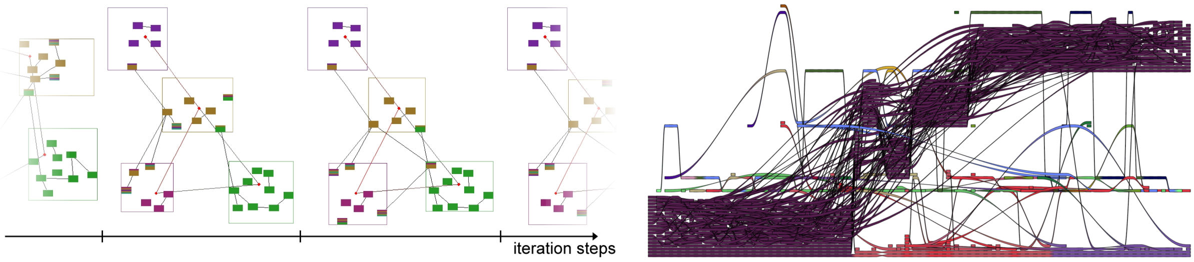
In the same spirit, such linked views cannot only be used as a top-level composition strategy among independent views. They can also be used as multiple embedded views that are balanced among themselves, but nested in the same base representation – effectively forming a hybrid of a balanced composition among themselves and a nesting into a common base representation. Fig. 6b shows such a hybrid combination with a spatial graph layout $[S \longleftarrow G]$ as base representation and a number of different embedded "in situ" views [HSS11]. These views can be independently defined, so that each can show a different facet of the graph. The figure depicts a wireless mesh network with all four additional facets given: The nodes are routers with geospatial positions, the edges are connections between them, and edge weights denote the dynamically changing signal strengths. In addition, routers are grouped (partitioned) according to their geospatial locations (e.g., by city districts). The embedded views show different visualizations, such as time-value plots of average signal strengths and node-link diagrams of structural details for certain partitions. Other examples, such as VisAlert [FAL*06] or TimeRadarTrees [BD08] follow a more interleaved approach representing graph structure, attributes, and temporal context in the same display space.

Other facet as base representation: This combination adds a visualization of the graph structure onto a base representation. Again, we can have two cases of the base representation either showing only the graph facet, or showing an already combined visualization of multiple facets. Fig. 7 shows a visualization for each of these two cases in (a) and (b), respectively. The example in Fig. 7a shows flow map visualizations of migration movements (edges) among countries (nodes) in Africa over 30 years [BBL12]. The combined structural and spatial facet $[S \longleftarrow G]$ are essentially embed-



(a) Small Multiples of Flow Maps [BBL12] (b) GenGIS [PPC*09]

Figure 7: Examples for spatial compositions of multiple facets in which the graph structure is not part of the base representation.



(a) Dynamic Drawing of Clustered Graphs [FT04] (b) Egocentric Storylines [MCSM13]

Figure 8: Examples for temporal compositions of multiple facets: (a) the graph structure is part of the base representation, (b) the graph structure is the facet over which the iteration runs.

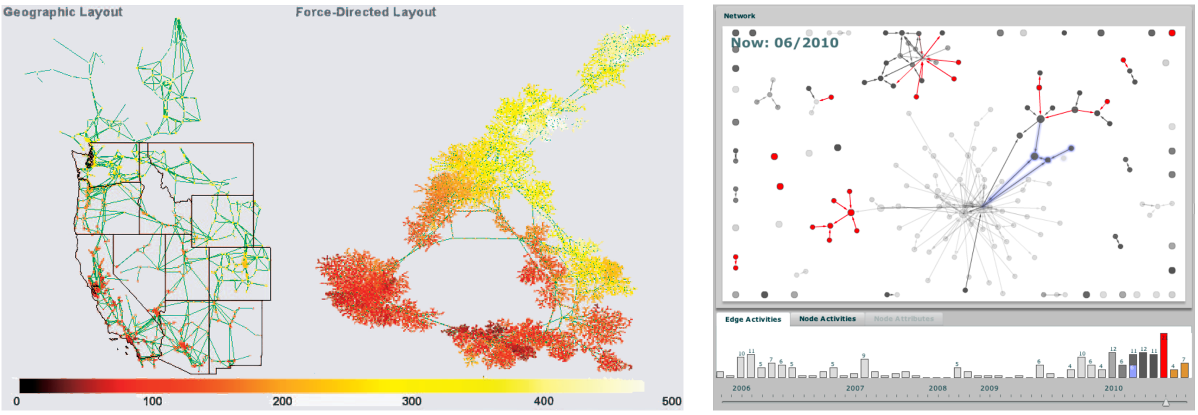
ded along the timeline T in a quasi-linear small multiples visualization, as it was introduced in Sec. 3.3 for $[T \leftarrow G]$. Whereas the example in Fig. 7b depicts a visualization from the GenGIS system [PPC*09] showing sampling sites and the proportion of different bacteria for each in superimposed pie charts. Into this base representation, a hierarchy G is embedded that shows the similarities between the sites based on their shared phylogenetic diversity. Other examples take a base representation of the graph's partitions and embed a single-faceted graph visualization – e.g., a $[S \leftarrow G]$ combination including the spatial facet [RMBA13] or a $[G \leftrightarrow T]$ combination including the temporal facet [vdEHBvW14].

4.2. Temporal Composition

Iteration is usually applied to only one facet. While it is possible to map multiple facets on display time (e.g., iterating over all partitions and showing for each partition an animation of its embedded dynamic network), we do not know of any visualization that uses such a temporal interleaving.

Graph structure as base representation: This combination shows the graph structure together with one or more other facet(s) and iterates over a third facet, which is usually time. In Fig. 8a, the base representation shows a partitioned graph and its temporal facet is mapped onto display time for animation. Such an animation becomes more challenging the more facets are involved, as not only the node positions must be stabilized, but all other facets as well. In this case, the shown dynamic drawing algorithm for clustered graphs [FT04] uses the idea of "virtual nodes" for each cluster. It treats clusters internally like extra nodes with the same stabilization mechanisms as the graph nodes, but shows them as rectangles.

Other facet as base representation: This combination uses a multi-faceted visualization as a base representation and traverses the graph structure. The example in Fig. 8b shows an ego-centered storyline visualization [MCSM13] with a chosen node at the bottom and related nodes stacked on top of it according to their similarity – i.e., nodes closer to the top are less similar to the chosen node. Time is encoded from left to right and partitions are color-coded onto the nodes. Horizontal bands represent nodes that change their vertical position over time, i.e., their similarity to the selected node. In the figure, a node is selected that changes its cluster membership twice – first belonging to the purple cluster, then



(a) GreenGrid [WSM*09]

(b) TempoVis [ATMS*11]

Figure 9: Examples for multiple representations of a single facet: (a) multiple representation of the structure – (left) on top of a map, (right) using a force-directed layout, (b) multiple representation of time – (top) as animation over the structure, (bottom) as a timeline of bars indicating connectivity.

to the red, and finally to the violet one. This makes it very similar to other nodes of the purple cluster in the beginning, which are close to it, but as it changes cluster membership, this similarity weakens and the purple nodes move to the top. A GUI (not shown) allows choosing other nodes to investigate and thus to traverse the graph structure. A similar technique can be used for investigating the edges of a graph [IA12].

5. Multiple Instances of Graph Facets

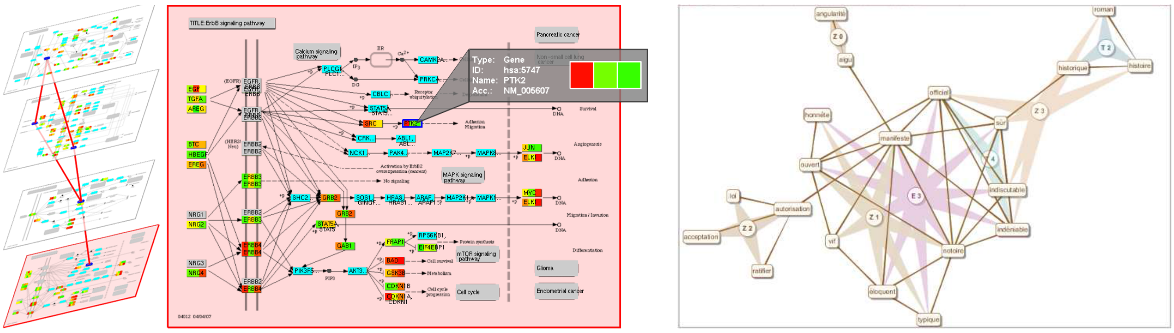
So far, graph facets have been shown only once in the composited visualizations. Yet, one can also show multiple instances of facets in one visualization – either by showing the same given facet multiple times, or by showing multiple instances of the same facet given by the data. In the following, we give two examples for each: for multiple instances of the graph structure and for multiple instances of another facet.

5.1. Multiple Representations

Showing the same facet multiple times is not unusual, as it is either the case that more information needs to be encoded together with the same facet than one can possibly fit in one view; or that one wants to balance a composition of base representation and embedded representation by complementing it with a composition that works the other way around. The latter enables users to look up relations between facets from both sides, as it was illustrated in Fig. 2c. The following examples give an idea of such multiple representations.

Multiple representations of the graph structure: Fig. 9a shows the information visualization system GreenGrid [WSM*09] for planning and monitoring power grids. Such a grid consists of electrical power stations and transformers (nodes) that are interconnected via transmission systems (edges). On the one hand, this graph structure is shown on the left side in its spatial context by superimposing it on a map visualization in a $[S \leftarrow G]$ combination and color-coding the base voltages onto the nodes. On the other hand, the same graph structure is also shown on the right side in a graph layout that highlights the topological dependencies of the structure, spreading out clusters of nodes that were overplotted in the densely populated areas in the map view. Both views are linked with each other, so that selecting nodes will highlight them with visual links in both view. Providing multiple representations of the structure is also important in other domains – e.g., for the analysis of flight delay data [KAW*14].

Multiple representations of another facet: Not only the graph structure itself, but also one of its associated facets can be duplicated and included more than once in a visualization. In the example in Fig. 9b, a snapshot of the TempoVis tool [ATMS*11] is shown. In its upper view, it shows the structure of an online community with users (nodes) and their conversations (edges). The temporal facet is mapped onto display time in a $[G \circlearrowleft T]$ combination, so that the dynamics of the network are shown via animation, which can be steered with the slider at the very bottom. The lower view explicitly shows the temporal facet that was already included in the upper view, but remained invisible there. For each time step, it shows a bar that encodes the number of activities (i.e., edges) in the online community and thus gives an overview of network connectivity across the entire temporal facet. If a particular time step exhibits unusually high or low activity, the user can set the slider accordingly and inspect the associated network of conversations for this incident. Other facets, such as the graph's partitioning can also be shown multiple times, for example, to show the structural relations between the partitions and details of their temporal trends [HSCW13].



(a) Caleydo [SKKS08]

(b) Donatien [HD12]

Figure 10: Examples for representation of multiple instances of the same facet: (a) showing multiple instances of the graph structure, (b) showing multiple instances of partitionings.

5.2. Multiple Instances

We cannot only duplicate a single facet and show it multiple times, but we can also have multiple instances of a facet given. On the one hand, this can be the graph structure itself – i.e., the same nodeset being connected by a number of different sets of edges [BPD11]. On the other hand, this can be any of the other facets – e.g., multiple partitionings or multiple parallel timelines as produced by multi-run simulations. In both cases, this falls into the field of visual graph comparison [AWW09, ABR*13]. The following two examples illustrate each of the above cases.

Multiple given instances of the graph structure: The example in Fig. 10a shows the Caleydo framework [SKKS08] with a stacked visualization of pathways. Pathways are in essence chemical reaction networks with enzymes and compounds as nodes and the chemical reactions that transform them as edges. Such pathways capture a specific function and all compounds and reactions related to it. Yet pathways are rarely agreed upon and besides the canonical pathways, a number of alternative models of how biochemical reactions perform a particular function are given. Showing these partially different pathways is a common visualization task in biomedical research. Caleydo allows doing so via brushing and linking, as the user can select a compound in a pathway and all other occurrences are highlighted. One pathway from the stack can be shown in detail on the right. A similar approach uses 3D primitives in the stacking for a more compact visualization of related pathways [BDS04].

Multiple given instances of another facet: One graph structure can be partitioned in a number of ways and the example in Fig. 10b shows a way to display and compare such multiple partitionings with each other. The depicted visualization system Donatien [HD12] uses a balanced approach to do so: alongside the graph nodes and edges, it also displays the clusters as nodes and connects them to the graph nodes that belong to them. Cluster nodes of the same color belong to the same partitioning, i.e., they have been produced by the same cluster algorithm. This encoding allows for a direct visual comparison of different partitionings of the same graph structure. Similarly, GrouseFlocks [AMA08] provides functionality to interactively change the partitioning of a network and analyze their outcome in a more iterative manner.

6. Conclusion and Directions for Future Research

The presented overview of multi-faceted graph visualization aims to survey a vast field of visualization. To do so, despite the extent of the research in this field, we have devised a categorization based on different visual combination modalities and merely highlighted selected visualizations for each category. We have further presented our efforts to relate the existing surveys in this field to our categorization. While there exists no one-to-one mapping between those surveys and our categories, the overlap we could establish is considerable and the differences are mainly due to particularities of individual facets. We deem this "meta survey" to be an important step towards a better understanding of the space of possible visualization solutions for multi-faceted graphs altogether. This overview further points into directions for future research in the following three aspects:

Graph visualization techniques: We can identify categories for which so far only very few visualizations exist. This is the case for temporal compositions, but also for the balanced spatial composition with interwoven nodesets.

Graph visualization surveys: When looking at the availability of surveys for faceted graph visualization, to the best of our knowledge for the domain of (geo-)spatial graph visualization, neither an output-oriented overview of their state-of-the-art, nor a task taxonomy exists to date.

Graph visualization facets: Other facets exist for which so far only very few faceted graph visualizations are known, if at all – for example, provenance, uncertainty, heterogeneity, or text/annotations. Thus visualizing graph provenance, graph uncertainty, graph heterogeneity, or graph annotations are worthwhile directions for systematic investigations.

For each of these three aspects, our categorization can provide guidance: For user studies of novel techniques, it can point to visualizations from the same category against which to compare. For new surveys of faceted graph visualization, its five categories can provide a first blueprint which can then be further refined and subdivided for the case at hand. And finally for investigating other facets, the spatial and temporal compositions can give an idea of possible ways to include them with the graph structure in a faceted graph visualization.

Acknowledgments

This work has been funded by the federal state of Mecklenburg-Vorpommern and EFRE within the project "Basic and Applied Research in Interactive Document Engineering and Maritime Graphics".

References

[ABR*13] ALPER B., BACH B., RICHE N. H., ISENBERG T., FEKETE J.-D.: Weighted graph comparison techniques for brain connectivity analysis. In *Proc. of CHI'13* (2013), pp. 483–492. 16

[ACJM03] AUBER D., CHIRICOTA Y., JOURDAN F., MELANCON G.: Multiscale visualization of small world networks. In *Proc. of IEEE InfoVis'03* (2003), pp. 75–81. 5

[ADH98] ASRATIAN A. S., DENLEY T. M. J., HÄGGKVIST R.: *Bipartite Graphs and their Applications*. Cambridge University Press, 1998. 3

[AMA07] ARCHAMBAULT D., MUNZNER T., AUBER D.: Grouse: Feature-based, steerable graph hierarchy exploration. In *Proc. of EuroVis'07* (2007), pp. 67–74. 5

[AMA08] ARCHAMBAULT D., MUNZNER T., AUBER D.: Grouseflocks: Steerable exploration of graph hierarchy space. *IEEE TVCG 14*, 4 (2008), 900–913. 16

[AMA*14] ALSALLAKH B., MICALLEF L., AIGNER W., HAUSER H., MIKSCH S., RODGERS P.: Visualizing sets and set-typed data: State-of-the-art and future challenges. In *EuroVis'14 State-of-the-Art Reports* (2014), pp. 1–21. 4

[AMST11] AIGNER W., MIKSCH S., SCHUMANN H., TOMINSKI C.: *Visualization of Time-Oriented Data*. Springer, 2011. 4

[APS14] AHN J.-W., PLAISANT C., SHNEIDERMAN B.: A task taxonomy for network evolution analysis. *IEEE TVCG 20*, 3 (2014), 365–376. 1, 2

[AS07] ARIS A., SHNEIDERMAN B.: Designing semantic substrates for visual network exploration. *Information Visualization 6*, 4 (2007), 281–300. 6, 7

[ATMS*11] AHN J.-W., TAIEB-MAIMON M., SOPAN A., PLAISANT C., SHNEIDERMAN B.: Temporal visualization of social network dynamics: Prototypes for nation of neighbors. In *Proc. of SBP'11* (2011), pp. 309–316. 15

[Aub04] AUBER D.: Tulip – A huge graph visualization framework. In *Graph Drawing Software*, Jünger M., Mutzel P., (Eds.). Springer, 2004, pp. 105–126. 6, 7

[AvHK06] ABELLO J., VAN HAM F., KRISHNAN N.: ASK-GraphView: A large scale graph visualization system. *IEEE TVCG 12*, 5 (2006), 669–676. 5

[AWW09] ANDREWS K., WOHLFAHRT M., WURZINGER G.: Visual graph comparison. In *Proc. of IV'09* (2009), pp. 62–67. 16

[BB05] BONCHEV D., BUCK G. A.: Quantitative measures of network complexity. In *Complexity in Chemistry, Biology, and Ecology*, Bonchev D., Rouvray D. H., (Eds.). Springer, 2005, pp. 191–235. 11

[BBDW14] BECK F., BURCH M., DIEHL S., WEISKOPF D.: The state of the art in visualizing dynamic graphs. In *EuroVis'14 State-of-the-Art Reports* (2014), pp. 83–103. 1, 3, 9

[BBL12] BOYANDIN I., BERTINI E., LALANNE D.: A qualitative study on the exploration of temporal changes in flow maps with animation and small-multiples. *Comp. Graph. Forum 31*, 3pt2 (2012), 1005–1014. 13, 14

[BC01] BROCKENAUER R., CORNELSEN S.: Drawing clusters and hierarchies. In Kaufmann and Wagner [KW01], pp. 193–227. 1, 6

[BC03] BRANDES U., CORMAN S. R.: Visual unrolling of network evolution and the analysis of dynamic discourse. *Information Visualization 2*, 1 (2003), 40–50. 8

[BCD*10] BEZERIANOS A., CHEVALIER F., DRAGICEVIC P., ELMQVIST N., FEKETE J.-D.: GraphDice: A system for exploring multivariate social networks. *Comp. Graph. Forum 29*, 3 (2010), 863–872. 6

[BD07] BALZER M., DEUSSEN O.: Level-of-detail visualization of clustered graph layouts. In *Proc. of APVIS'07* (2007), pp. 133–140. 4, 5

[BD08] BURCH M., DIEHL S.: TimeRadarTrees: Visualizing dynamic compound digraphs. *Comp. Graph. Forum 27*, 3 (2008), 823–830. 13

[BDS04] BRANDES U., DWYER T., SCHREIBER F.: Visualizing related metabolic pathways in two and a half dimensions. In *Proc. of GD'04* (2004), pp. 111–122. 8, 16

[BETT94] BATTISTA G. D., EADES P., TAMASSIA R., TOLLIS I. G.: Algorithms for drawing graphs: an annotated bibliography. *Computational Geometry 4*, 5 (1994), 235–282. 1

[BETT99] BATTISTA G. D., EADES P., TAMASSIA R., TOLLIS I. G.: *Graph Drawing*. Prentice Hall, 1999. 1

[BHK*05] BORISJUK L., HAJIREZAEI M.-R., KLUKAS C., ROLLETSCHEK H., SCHREIBER F.: Integrating data from biological experiments into metabolic networks with the DBE information system. *In Silico Biology 5*, 2 (2005), 93–102. 6, 7

[BIM12] BRANDES U., INDLEKOFER N., MADER M.: Visualization methods for longitudinal social networks and stochastic actor-oriented modeling. *Social Networks 34*, 3 (2012), 291–308. 4, 9

[BKA*15] BRODKORB F., KUIJPER A., ANDRIENKO G., ANDRIENKO N., VON LANDESBERGER T.: Overview with details for exploring geo-located graphs on maps. *Information Visualization* (2015). to appear. 10

[BMMS91] BUJA A., MCDONALD J. A., MICHALAK J., STUETZLE W.: Interactive data visualization using focusing and linking. In *Proc. of IEEE Vis'91* (1991), pp. 156–163. 3

[BPD11] BECK F., PETKOV R., DIEHL S.: Visually exploring multi-dimensional code couplings. In *Proc. of IEEE VisSoft'11* (2011), pp. 1–8. 16

[BPF14] BACH B., PIETRIGA E., FEKETE J.-D.: GraphDiaries: Animated transitions and temporal navigation for dynamic networks. *IEEE TVCG 20*, 5 (2014), 740–754. 8, 9

[Bra01] BRANKE J.: Dynamic graph drawing. In Kaufmann and Wagner [KW01], pp. 228–246. 3

[CC07] COLLINS C., CARPENDALE S.: VisLink: Revealing relationships amongst visualizations. *IEEE TVCG 13*, 6 (2007), 1192–1199. 9

[Che06] CHEN C.: *Information Visualization. Beyond the Horizon*. Springer, 2006, ch. Graph Drawing Algorithms, pp. 65–87. 1

[CXP*12] CURTIS R., XIANG J., PARIKH A., KINNAIRD P., XING E.: Enabling dynamic network analysis through visualization in TVNViewer. *BMC Bioinf. 13*, 1 (2012), 204–216. 12

[DGK01] DIEHL S., GÖRG C., KERREN A.: Preserving the mental map using foresighted layout. In *Proc. of Data Visualization* (2001), pp. 175–184. 11

[DMK05] DYKES J., MACEACHREN A. M., KRAAK M.-J. (Eds.): *Exploring Geovisualization*. Elsevier, 2005. 4

[DMT08] DODGE M., MCDERBY M., TURNER M. (Eds.): *Geographic Visualization*. John Wiley & Sons, 2008. 4

[DPS02] DÍAZ J., PETIT J., SERNA M.: A survey of graph layout problems. *ACM Computing Surveys 34*, 3 (2002), 313–356. 2

[EDG*08] ELMQVIST N., DO T.-N., GOODELL H., HENRY N., FEKETE J.-D.: ZAME: Interactive large-scale graph visualization. In *Proc. of IEEE PacificVis'08* (2008), pp. 215–222. 6

[EF10] ELMQVIST N., FEKETE J.-D.: Hierarchical aggregation for information visualization: Overview, techniques, and design guidelines. *IEEE TVCG 16*, 3 (2010), 439–454. 5

[EH00] EADES P., HUANG L. M.: Navigating clustered graphs using force-directed methods. *Journal of Graph Algorithms and Applications 4*, 3 (2000), 157–181. 11

[FAL*06] FORESTI S., AGUTTER J., LIVNAT Y., MOON S., ERBACHER R.: Visual correlation of network alerts. *IEEE Computer Graphics & Applications 26*, 2 (2006), 48–59. 13

[FH09] FUCHS R., HAUSER H.: Visualization of multi-variate scientific data. *Comp. Graph. Forum 28*, 6 (2009), 1670–1690. 4

[Fis10] FISHER D.: Animation for visualization: Opportunities and drawbacks. In *Beautiful Visualization*, Steele J., Iliinsky N., (Eds.). O'Reilly, 2010. 3

[FT04] FRISHMAN Y., TAL A.: Dynamic drawing of clustered graphs. In *Proc. of IEEE InfoVis'04* (2004), pp. 191–198. 14

[FWD*03] FEKETE J.-D., WANG D., DANG N., ARIS A., PLAISANT C.: Overlaying graph links on treemaps. In *Poster Compendium of IEEE InfoVis'03* (2003), pp. 82–83. 5

[GAW*11] GLEICHER M., ALBERS D., WALKER R., JUSUFI I., HANSEN C. D., ROBERTS J. C.: Visual comparison for information visualization. *Information Visualization 10*, 4 (2011), 289–309. 3

[GDLP09] GIACOMO E., DIDIMO W., LIOTTA G., PALLADINO P.: Visual analysis of one-to-many matched graphs. In *Proc. of GD'09* (2009), pp. 133–144. 9, 10

[GHK10] GANSNER E. R., HU Y., KOBOUROV S.: GMap: Visualizing graphs and clusters as maps. In *Proc. of IEEE PacificVis'10* (2010), pp. 201–208. 11

[GW06] GAERTLER M., WAGNER D.: A hybrid model for drawing dynamic and evolving graphs. In *Proc. of GD'06* (2006), pp. 189–200. 8

[HB05] HEER J., BOYD D.: Vizster: Visualizing online social networks. In *Proc. of IEEE InfoVis'05* (2005), pp. 32–39. 5, 6

[HD12] HASCOËT M., DRAGICEVIC P.: Interactive graph matching and visual comparison of graphs and clustered graphs. In *Proc. of AVI'12* (2012), pp. 522–529. 16

[HEF*14] HURTER C., ERSOY O., FABRIKANT S., KLEIN T., TELEA A.: Bundled visualization of dynamic graph and trail data. *IEEE TVCG 20*, 8 (2014), 1141–1157. 4

[HET12] HURTER C., ERSOY O., TELEA A.: Graph bundling by kernel density estimation. *Comp. Graph. Forum 31*, 3pt1 (2012), 865–874. 10

[HMM00] HERMAN I., MELANÇON G., MARSHALL M. S.: Graph visualization and navigation in information visualization: A survey. *IEEE TVCG 6*, 1 (2000), 24–43. 1

[Hol06] HOLTEN D.: Hierarchical edge bundles: Visualization of adjacency relations in hierarchical data. *IEEE TVCG 12*, 5 (2006), 741–748. 5

[HSCW13] HADLAK S., SCHUMANN H., CAP C., WOLLENBERG T.: Supporting the visual analysis of dynamic networks by clustering associated temporal attributes. *IEEE TVCG 19*, 12 (2013), 2267–2276. 2, 15

[HSS11] HADLAK S., SCHULZ H.-J., SCHUMANN H.: In situ exploration of large dynamic networks. *IEEE TVCG 17*, 12 (2011), 2334–2343. 2, 13

[HTSS10] HADLAK S., TOMINSKI C., SCHULZ H.-J., SCHUMANN H.: Visualization of attributed hierarchical structures in a spatio-temporal context. *IJGIS 24*, 10 (2010), 1497–1513. 10

[IA12] ITOH M., AKAISHI M.: Visualization for changes in relationships between historical figures in chronicles. In *Proc. of IV'12* (2012), pp. 283–290. 15

[IYT*13] ITOH M., YOKOYAMA D., TOYODA M., TOMITA Y., KAWAMURA S., KITSUREGAWA M.: Visualization of passenger flows on metro. In *Poster Session at IEEE VAST'13* (2013). 12, 13

[JDK10] JUSUFI I., DINGJIE Y., KERREN A.: The network lens: Interactive exploration of multivariate networks using visual filtering. In *Proc. of IV'10* (2010), pp. 35–42. 7

[JE12] JAVED W., ELMQVIST N.: Exploring the design space of composite visualization. In *Proc. of IEEE PacificVis'12* (2012), pp. 1–8. 3

[JKZ13] JUSUFI I., KERREN A., ZIMMER B.: Multivariate network exploration with jauntynets. In *Proc. of IV'13* (2013), pp. 19–27. 6

[JRA09] JERN M., ROGSTADIUS J., ASTROM T.: Treemaps and choropleth maps applied to regional hierarchical statistical data. In *Proc. of IV'09* (2009), pp. 403–410. 12

[JRS12] JUNKER A., ROHN H., SCHREIBER F.: Visual analysis of transcriptome data in the context of anatomical structures and biological networks. *Front. in Plant Science 3*, 252 (2012). 9, 10

[JS91] JOHNSON B., SHNEIDERMAN B.: Tree-Maps: A space-filling approach to the visualization of hierarchical information structures. In *Proc. of IEEE Vis'91* (1991), pp. 284–291. 5

[JSS*13] JOHN M., SCHULZ H.-J., SCHUMANN H., UHRMACHER A. M., UNGER A.: Constructing and visualizing reaction networks from pi-calculus models. *Formal Aspects of Computing 25*, 5 (2013), 723–742. 11

[KAW*14] KO S., AFZAL S., WALTON S., YANG Y., CHAE J., MALIK A., JANG Y., CHEN M., EBERT D.: Analyzing high-dimensional multivariate network links with integrated anomaly detection, highlighting and exploration. In *Proc. of IEEE VAST'14* (2014), pp. 83–92. 15

[KH13] KEHRER J., HAUSER H.: Visualization and visual analysis of multifaceted scientific data: A survey. *IEEE TVCG 19*, 3 (2013), 495–513. 1

[KK95] KRAAK M.-J., KLOMP A.: A classification of cartographic animations: Towards a tool for the design of dynamic maps in a GIS environment. In *Proc. of the Seminar on Teaching Animated Cartography* (1995), p. 118. 3

[KKC14a] KERRACHER N., KENNEDY J., CHALMERS K.: The design space of temporal graph visualisation. In *Short Paper Proc. of EuroVis'14* (2014), pp. 7–11. 1, 2, 3, 9

[KKC14b] KERRACHER N., KENNEDY J., CHALMERS K.: Tasks for temporal graph visualisation. *arXiv.org e-print service 1402.2867* (2014). 2

[KO10] KRAAK M.-J., ORMELING F.: *Cartography: Visualization of Spatial Data*, 3rd ed. The Guilford Press, 2010. 4

[KPW14] KERREN A., PURCHASE H., WARD M. (Eds.): *Multivariate Network Visualization*. Springer, 2014. 1, 7, 19

[KW01] KAUFMANN M., WAGNER D. (Eds.): *Drawing Graphs: Methods and Models*. Springer, 2001. 1, 17

[LEH05] LI W., EADES P., HONG S.-H.: Navigating software architectures with constant visual complexity. In *Proc. of IEEE VL/HCC'05* (2005), pp. 225–232. 6

[LMYH11] LUO W., MACEACHREN A. M., YIN P., HARDISTY F.: Spatial-social network visualization for exploratory data analysis. In *Proc. of LBSN'11* (2011), pp. 65–68. 12, 13

[LPP*06] LEE B., PLAISANT C., PARR C. S., FEKETE J.-D., HENRY N.: Task taxonomy for graph visualization. In *Proc. of BELIV'06* (2006), pp. 1–5. 2

[MCSM13] MUELDER C., CRNOVRSANIN T., SALLABERRY A., MA K.-L.: Egocentric storylines for visual analysis of large dynamic graphs. In *Proc. of IEEE BigData'13* (2013), pp. 56–62. 14

[ME09] MCDONNEL B., ELMQVIST N.: Towards utilizing GPUs in information visualization: A model and implementation of image-space operations. *IEEE TVCG 15*, 6 (2009), 1105–1112. 6

[MELS95] MISUE K., EADES P., LAI W., SUGIYAMA K.: Layout adjustment and the mental map. *Journal of Visual Languages and Computing 6*, 2 (1995), 183–210. 4

[MS04] MARRIOTT K., STUCKEY P. J.: NP-completeness of minimal width unordered tree layout. *Journal of Graph Algorithms and Applications 8*, 2 (2004), 295–312. 2

[NL05] NOACK A., LEWERENTZ C.: A space of layout styles for hierarchical graph models of software systems. In *Proc. of SoftVis'05* (2005), pp. 155–164. 4

[PMD12] PINAUD B., MELANÇON G., DUBOIS J.: PORGY: A visual graph rewriting environment for complex systems. *Comp. Graph. Forum 31*, 3pt4 (2012), 1265–1274. 8

[PPC*09] PARKS D. H., PORTER M., CHURCHER S., WANG S., BLOUIN C., WHALLEY J., BROOKS S., BEIKO R. G.: GenGIS: A geospatial information system for genomic data. *Genome Research 19*, 10 (2009), 1896–1904. 14

[PPS14] PRETORIUS A. J., PURCHASE H. C., STASKO J. T.: Tasks for multivariate network analysis. In Kerren et al. [KPW14], pp. 77–95. 2

[PS12] PERER A., SUN J.: MatrixFlow: Temporal network visual analytics to track symptom evolution during disease progression. In *AMIA Annual Symposium Proceedings* (2012), vol. 2012, pp. 716–725. 8

[PvW08] PRETORIUS A. J., VAN WIJK J. J.: Visual inspection of multivariate graphs. *Comp. Graph. Forum 27*, 3 (2008), 967–974. 12

[Rei10] REITZ F.: A framework for an ego-centered and time-aware visualization of relations in arbitrary data repositories. *arXiv.org preprint:1009.5183* (2010). 9

[RLFP11] ROBERTS H., LAROCHELLE D., FARIS R., PALFREY J.: Mapping local internet control. In *Proc. of CCW'11* (2011). 10

[RM13] RUFIANGE S., MCGUFFIN M. J.: DiffAni: Visualizing dynamic graphs with a hybrid of difference maps and animation. *IEEE TVCG 19*, 12 (2013), 2556–2565. 8

[RMBA13] ROZENBLAT C., MELANÇON G., BOURQUI R., AUBER D.: Comparing multilevel clustering methods on weighted graphs: The case of worldwide air passenger traffic 2000-2004. In *Methods for Multilevel Analysis and Visualisation of Geographical Networks*, Rozenblat C., Melançon G., (Eds.). Springer, 2013, pp. 141–154. 14

[RMF12] RUFIANGE S., MCGUFFIN M. J., FUHRMAN C. P.: Treematrix: A hybrid visualization of compound graphs. *Comp. Graph. Forum 31*, 1 (2012), 89–101. 5

[Rob07] ROBERTS J. C.: State of the art: Coordinated & multiple views in exploratory visualization. In *Proc. of CMV'07* (2007), pp. 61–71. 3

[Rod05] RODGERS P.: Graph drawing techniques for geographic visualization. In *Exploring Geovisualization*, Dykes J., MacEachren A. M., Kraak M.-J., (Eds.). Elsevier, 2005, pp. 143–158. 1, 10

[Rod13] RODGERS P.: A survey of Euler diagrams. *Journal of Visual Languages and Computing 25*, 3 (2013), 134–155. 4

[RTT*06] RODRIGUES JR. J. F., TONG H., TRAINA A. J. M., FALOUTSOS C., LESKOVEC J.: GMine: A system for scalable, interactive graph visualization and mining. In *Proc. of VLDB'06* (2006), pp. 1195–1198. 6

[RUK*10] ROHRSCHNEIDER M., ULLRICH A., KERREN A., STADLER P. F., SCHEUERMANN G.: Visual network analysis of dynamic metabolic pathways. In *Proc. of ISVC'10* (2010), pp. 316–327. 12

[SBM*14] STEIGER M., BERNARD J., MITTELSTÄDT S., LÜCKE-TIEKE H., KEIM D., MAY T., KOHLHAMMER J.: Visual analysis of time-series similarities for anomaly detection in sensor networks. *Comp. Graph. Forum 33*, 3 (2014), 401–410. 12, 13

[Sch07] SCHAEFFER S. E.: Graph clustering. *Computer Science Review 1*, 1 (2007), 27–64. 4

[Sch11] SCHULZ H.-J.: Treevis.net: A tree visualization reference. *IEEE Computer Graphics & Applications 31*, 6 (2011), 11–15. 4

[SD13] SHNEIDERMAN B., DUNNE C.: Interactive network exploration to derive insights: Filtering, clustering, grouping, and simplification. In *Proc. of GD'13* (2013), pp. 2–18. 11

[SdMA*12] SILVA T., DE MELO P., ALMEIDA J., SALLES J., LOUREIRO A.: Visualizing the invisible image of cities. In *Proc. of the IEEE GreenCom'12* (2012), pp. 382–389. 12

[SDMW09] SCHREIBER F., DWYER T., MARRIOTT K., WYBROW M.: A generic algorithm for layout of biological networks. *BMC Bioinformatics 10*, 1 (2009), 1–12. 9, 10

[SHQ08] SHANNON R., HOLLAND T., QUIGLEY A.: *Multivariate Graph Drawing using Parallel Coordinate Visualisations*. Tech. rep., University College Dublin, 2008. 7

[SHS11] SCHULZ H.-J., HADLAK S., SCHUMANN H.: Point-based visualization for large hierarchies. *IEEE TVCG 17*, 5 (2011), 598–611. 7

[SKKS08] STREIT M., KALKUSCH M., KASHOFER K., SCHMALSTIEG D.: Navigation and exploration of interconnected pathways. *Comp. Graph. Forum 27*, 3 (2008), 951–958. 16

[SLN05] SARAIYA P., LEE P., NORTH C.: Visualization of graphs with associated timeseries data. In *Proc. of IEEE InfoVis'05* (2005), pp. 225–232. 8

[SS06] SCHULZ H.-J., SCHUMANN H.: Visualizing graphs: A generalized view. In *Proc. of IV'06* (2006), pp. 166–173. 1

[SSK14] SAKET B., SIMONETTO P., KOBOUROV S. G.: Group-level graph visualization taxonomy. In *Short Paper Proc. of EuroVis'14* (2014), pp. 85–89. 2

[SWS10] STEIN K., WEGENER R., SCHLIEDER C.: Pixel-oriented visualization of change in social networks. In *Proc. of the IEEE/ACM ASONAM'10* (2010), pp. 233–240. 8

[SWW*15] SHI L., WANG C., WEN Z., QU H., LIN C., LIAO Q.: 1.5D egocentric dynamic network visualization. *IEEE TVCG 21*, 5 (2015), 624–637. 8, 9

[Tam13] TAMASSIA R. (Ed.): *Handbook of Graph Drawing and Visualization*. CRC Press, 2013. 1, 20

[TAS09] TOMINSKI C., ABELLO J., SCHUMANN H.: CGV – an interactive graph visualization system. *Computers & Graphics 33*, 6 (2009), 660–678. 5

[TDKB07] THIEL K., DILL F., KOTTER T., BERTHOLD M.: Towards visual exploration of topic shifts. In *Proc. of IEEE ISIC'07* (2007), pp. 522–527. 8

[TMB02] TVERSKY B., MORRISON J. B., BETRANCOURT M.: Animation: Can it facilitate? *International Journal of Human-Computer Studies 57*, 4 (2002), 247–262. 9

[TS07] TU Y., SHEN H.-W.: Visualizing changes of hierarchical data using treemaps. *IEEE TVCG 13*, 6 (2007), 1286–1293. 12

[TS13] TSHAGHARYAN G., SCHULZ H.-J.: A graph-based overview visualization for data landscapes. *Computer Science and Information Technology 1*, 3 (2013), 225–232. 6

[Tuf90] TUFTE E. R.: *Envisioning Information*. Graphics Press, 1990. 8

[VBW15] VEHLOW C., BECK F., WEISKOPF D.: The state of the art in visualizing group structures in graphs. In *EuroVis'15 State-of-the-Art Reports* (2015). to appear. 1, 6

[vDAG12] VAN DONGEN S., ABREU-GOODGER C.: Using MCL to extract clusters from networks. In *Bacterial Molecular Networks: Methods and Protocols*, van Helden J., Toussaint A., Thieffry D., (Eds.). Springer, 2012, pp. 281–295. 5

[vdEHBvW14] VAN DEN ELZEN S., HOLTEN D., BLAAS J., VAN WIJK J.: Dynamic network visualization with extended massive sequence views. *IEEE TVCG 20*, 8 (2014), 1087–1099. 14

[vLKS*11] VON LANDESBERGER T., KUIJPER A., SCHRECK T., KOHLHAMMER J., VAN WIJK J., FEKETE J.-D., FELLNER D.: Visual analysis of large graphs: State-of-the-art and future research challenges. *Comp. Graph. Forum 30*, 6 (2011), 1719–1749. 1

[Wat06] WATTENBERG M.: Visual exploration of multivariate graphs. In *Proc. of CHI'06* (2006), pp. 811–819. 6

[WB97] WONG P. C., BERGERON R. D.: 30 years of multidimensional multivariate visualization. In *Scientific Visualization*, Nielson G., Hagen H., Müller H., (Eds.). IEEE, 1997, pp. 3–33. 4

[WF94] WASSERMAN S., FAUST K.: *Social Network Analysis: Methods and Applications*. Cambridge University Press, 1994. 6

[WFC*06] WONG P. C., FOOTE H., CHIN JR. G., MACKEY P., PERRINE K.: Graph signatures for visual analytics. *IEEE TVCG 12*, 6 (2006), 1399–1413. 6, 11

[Wil12] WILLS G.: *Visualizing Time: Designing Graphical Representations for Statistical Data*. Springer, 2012. 4

[Wol13] WOLFF A.: Graph drawing and cartography. In Tamassia [Tam13], pp. 697–736. 1, 10

[WSM*09] WONG P. C., SCHNEIDER K., MACKEY P., FOOTE H., CHIN G., GUTTROMSON R., THOMAS J.: A novel visualization technique for electric power grid analytics. *IEEE TVCG 15*, 3 (2009), 410–423. 15

[YEL10] YI J. S., ELMQVIST N., LEE S.: TimeMatrix: Analyzing temporal social networks using interactive matrix-based visualizations. *International Journal of Human-Computer Interaction 26*, 11 & 12 (2010), 1031–1051. 12

[ZXYQ13] ZHOU H., XU P., YUAN X., QU H.: Edge bundling in information visualization. *Tsinghua Science and Technology 18*, 2 (2013), 145–156. 3

Biographies

Steffen Hadlak received his master's (2009) and doctorate degree (2014) from the University of Rostock. He is currently working at Fraunhofer IGD Rostock on visualizing multivariate networks in space and time for different applications such as bioinformatics and communication network analysis. His research interests include GPU-based rendering and alternative interaction techniques as well as their application to Information Visualization. More about his research can be found at http://www.informatik.uni-rostock.de/~hadlak/.

Heidrun Schumann received her master's, PhD, and habilitation degrees in 1977, 1981, and 1989, respectively, from the University of Rostock, where she is heading the Computer Graphics Research Group since 1992. Her research covers Information Visualization, Visual Analytics, and Rendering. Her current projects, supported by funding agencies and industry, include scalable frameworks for information visualization and adaptive visual interfaces. More about her research can be found at http://www.informatik.uni-rostock.de/~schumann/.

Hans-Jörg Schulz received a master degree in computer science in 2004 and his PhD in 2010 from the University of Rostock. He is currently working as a senior researcher at the Fraunhofer IGD Rostock on industrial and academic projects in the domain of Big Data Analytics. His research interests include graph visualization, visualization design spaces, and Visual Analytics for biomedical and systems biological applications. More about his research can be found at http://www.informatik.uni-rostock.de/~hs162/.