

Copyright © 2009 IEEE.

Reprinted from
IEEE Transactions on Visualization and Computer Graphics
Vol. 15, No. 5 (Sep/Oct 2009), pages 759–776.

This material is posted here with permission of the IEEE. Internal or personal use of this material is permitted. However, permission to reprint/republish this material for advertising or promotional purposes or for creating new collective works for resale or redistribution must be obtained from the IEEE by sending a blank email message to info.pub.permission@ieee.org.

By choosing to view this document, you agree to all provisions of the copyright laws protecting it.

A Survey of Radial Methods for Information Visualization

Geoffrey M. Draper, Yarden Livnat, *Member, IEEE*, and Richard F. Riesenfeld

Abstract—Radial visualization, or the practice of displaying data in a circular or elliptical pattern, is an increasingly common technique in information visualization research. In spite of its prevalence, little work has been done to study this visualization paradigm as a methodology in its own right. We provide a historical review of radial visualization, tracing it to its roots in centuries-old statistical graphics. We then identify the types of problem domains to which modern radial visualization techniques have been applied. A taxonomy for radial visualization is proposed in the form of seven design patterns encompassing nearly all recent works in this area. From an analysis of these patterns, we distill a series of design considerations that system builders can use to create new visualizations that address aspects of the design space that have not yet been explored. It is hoped that our taxonomy will provide a framework for facilitating discourse among researchers and stimulate the development of additional theories and systems involving radial visualization as a distinct design metaphor.

Index Terms—Information visualization, multivariate visualization, visualization techniques and methodologies, interactive data exploration and discovery.

1 INTRODUCTION

THE goal of information visualization is to communicate technical information in a graphical, interactive, and understandable way. This paper focuses on an emerging paradigm in information visualization, namely, radial visualization. We use the term *radial visualization* to describe any interactive system that arranges data in an elliptical fashion. Radial visualization is an increasingly popular metaphor in information visualization and HCI research; indeed, many recent papers in the field of information visualization make use of radial visualization to at least some extent. Examples include the *Hyperbolic Browser* [52], *Radial Traffic Analyzer* [49], and *VisAlert* [55]. The prevalence of radial visualization is perhaps due to its aesthetic appeal, its compact layout, and its ability to put selectable data within easy reach of the user [27]. Whatever the reason, the sheer popularity of this visualization metaphor makes it a subject worthy of further analysis. Somewhat surprisingly, little effort has been devoted to studying radial visualization as a distinct methodology of its own.

The present work aims to fill this identified gap. To our knowledge, this paper is the first such effort to gather, review, and analyze the vast body of research involving radial visualization. We first provide a historical review of radial visualization, tracing it from its modest roots in the centuries-old discipline of statistical graphics. We then identify the

principal types of problem domains to which modern radial visualization techniques have been applied. A taxonomy for radial visualization is proposed in the form of seven design patterns encompassing nearly all recent works in this area. We discuss several examples of each pattern, drawn from both academic and commercial arenas. From an analysis of these patterns, we then distill a series of design dimensions, or considerations, that system builders can use to create new visualizations that address aspects of the design space that have not yet been explored. Finally, we suggest possible avenues of future work in radial visualization by applying the design patterns and dimensions presented herein to additional problem domains.

In so doing, we are not suggesting that radial visualization is a panacea for all information visualization problems. Rather, we hope instead to raise the research community's awareness of radial visualization, elevating it from an interesting visual gimmick to a unique methodology of its own. Information visualization researchers will thus be able to evaluate radial techniques in the context of others, and select whichever visualization metaphor is most appropriate for a given problem.

2 A HISTORY OF RADIAL GRAPHICS IN STATISTICS

The term *radial visualization* appears to have been coined by Hoffman et al. [35] in the 1990s, but the underlying concepts are firmly rooted in the statistical graphics literature of the 19th century. Today, techniques such as the pie chart, starplot, and radar plot are frequently used in business and the media to communicate numerical data visually. These graphics are the common ancestors of virtually all the radial visualization methods found in the state-of-the-art research, so we begin our survey with them.

2.1 Pie Charts

A *radial display* is a visualization paradigm in which information is laid out in a circular or elliptical pattern. Perhaps, the earliest use of a radial display in statistical

- G.M. Draper is with Brigham Young University Hawaii, Department of Computer and Information Sciences, BYUH #1956, 55-220 Kulanui St., Laie, HI 96762. E-mail: gmd2@byuh.edu.
- Y. Livnat is with the University of Utah, Scientific Computing and Imaging Institute, 3750 WEB, 72 S Central Campus Drive, Salt Lake City, UT 84112. E-mail: yarden@sci.utah.edu.
- R.F. Riesenfeld is with the University of Utah, School of Computing, 50 S Central Campus Dr. #3190, Salt Lake City, UT 84112. E-mail: rfr@cs.utah.edu.

Manuscript received 30 Apr. 2008; revised 24 Sept. 2008; accepted 15 Jan. 2009; published online 26 Jan. 2009.

Recommended for acceptance by G. Robertson.

For information on obtaining reprints of this article, please send e-mail to: tcvg@computer.org, and reference IEEECS Log Number TVCG-2008-04-0060. Digital Object Identifier no. 10.1109/TVCG.2009.23.

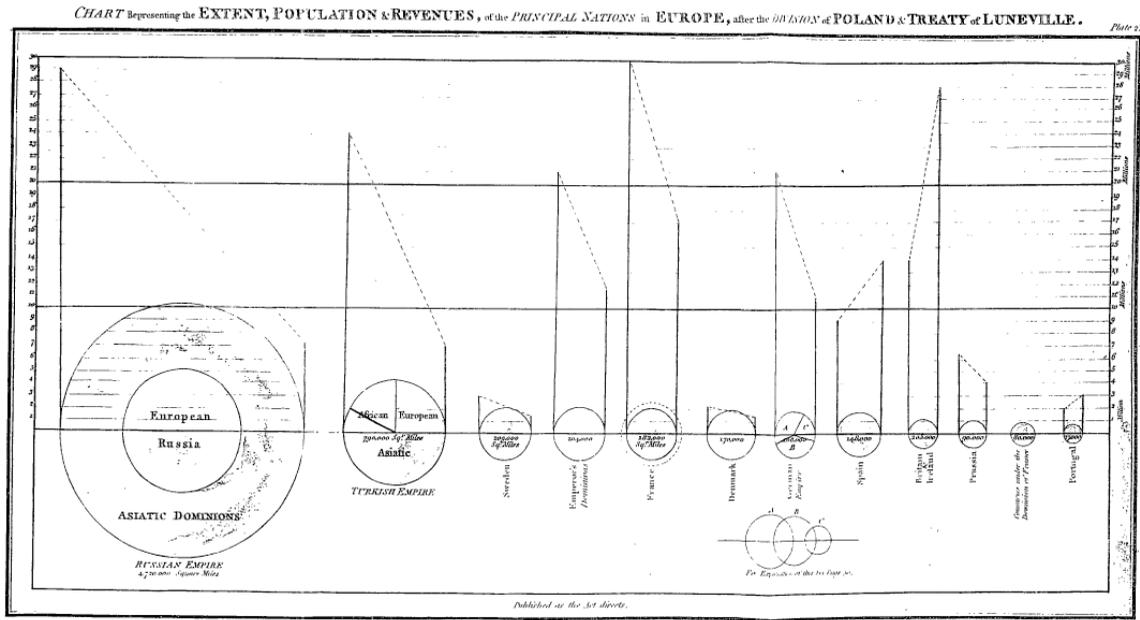


Fig. 1. A chart representing the extent, population, and revenues of the principal nations in Europe (from William Playfair's *The Statistical Breviary* [63]).

graphics was the *pie chart*. The first known occurrence of a pie chart (Fig. 1) was in William Playfair's 1801 treatise, *The Statistical Breviary* [63], a detailed listing of the population, and wealth in the nations of 19th century Europe. Spence [70] reviews the historical background of Playfair's work and its influence on modern statistical graphics.

Although commonly used in mainstream media, the pie chart has some limitations. In particular, when the wedges in a pie chart are almost the same size, it is difficult to determine visually which wedge is largest (Fig. 2). A bar chart is generally better suited for this task. In fact, Wilkinson [83] points out that a pie chart is simply a stacked bar chart in polar coordinates. Although the chart shown in Fig. 2a is typical of the most common variety of pie chart, other variants exist as well. Wilkinson describes an alternate pie chart whose area is not divided up into wedges, but rather concentric rings. In contrast to a traditional pie chart, comparisons among the different entities are based on the length of the various radii (Fig. 2b), rather than on the angular magnitude of a wedge. Although a ring with a longer radius will usually have a larger area; for concentric rings, the area also depends on the offset of each ring from the center of the circle. This potential for confusion is likely one reason for this variety of pie chart not having met widespread acceptance.

A variant of the pie chart was also used by Nightingale [61] in the 1850s to communicate the poor sanitary conditions of British Army hospitals during the Crimean War. Each wedge corresponds to one month, and the area of the wedge represents the total number of deaths incurred at the hospital during that month. The Nightingale chart, now known as a *rose diagram* or *polar area chart*, is chiefly different from a pie chart in that each wedge of Nightingale's chart is equiangular. Accordingly, the differences in area among wedges stem from the fact that each wedge has a different radius, rather than a different angular magnitude. In Fig. 3, reprinted from [61], the large light-colored portion of each wedge shows the

proportion of deaths in a given month that were due to preventable diseases rather than wounds incurred in battle. The chart on the left shows how the death rate dropped dramatically after Nightingale implemented sanitary reforms at the hospital.

2.2 Star Plots

The *star plot* is another form of radial diagram that figures prominently in statistical graphics. This form of chart is alternately known as a *kiviat diagram* or a *spider web*, and is designed for viewing multivariate systems in a compact form. A star plot is constructed by mapping each variable to one of several axes radiating from a common center point. The axes are spaced equiangularly from each other and the range of each variable is scaled such that each axis is of the same length. The data points are then drawn at their appropriate location on the axes, and lines are drawn connecting them. Star plots are the radial equivalent of parallel coordinates [40].

There are minor stylistic differences in the ways that star plots may be rendered. In one variant, like the one shown in Fig. 4a, the axes are drawn but the circumference is not. Others draw the circumference, but omit the axes (Fig. 4b). Still other star plots employ a combination of these two extremes [25].

Moreover, multiple star plots may be superimposed to allow comparisons among several distinct data sets with common field names. This gives rise to a type of graphic called the *radar plot* [83]. Radar plots are most effective when the area enclosed by one entity is entirely contained within another area (Fig. 5) or when transparency is used, allowing comparison of relative areas without occlusion.

2.3 Sociograms

One of the first modern applications of radial displays was for *sociometry*, "the study and measurement of interpersonal relationships in a group of people [58]." The *target*

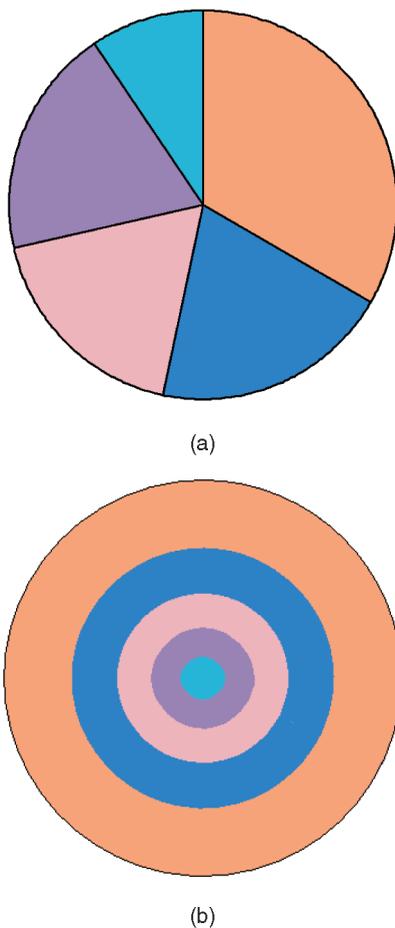


Fig. 2. The pie chart. Notice that it is difficult to discern relative magnitude among similarly sized entities. (a) Standard pie chart. (b) Concentric pie chart.

sociogram, introduced by Northway [62], depicts people as small circles, and relationships between people as lines between these circles (Fig. 6).

Radial visualization extends the radial display concept to include the interactive manipulation of data. Most often, this means that the radial visualization has been implemented as part of a computer program, but this is not necessarily so. For example, Northway [62] also discusses an interactive analog to the target sociogram called the *peg board sociogram* (Fig. 7). In it, the people are represented by pegs that may be moved anywhere on the board, and relationships are represented by rubber bands stretched between pegs. In this manner, the placement of people and relationships can be modified interactively.

3 TYPICAL USAGE DOMAINS

The examples cited in the previous section create a contextual framework from which modern radial visualization techniques and systems are derived. Radial visualization is a broad field, with applications targeting a variety of subject areas. We do not wish to imply that radial visualization is the right solution for every visualization need; however, there are certain application domains for which it seems especially well suited. Based on our survey of the field, we have identified four basic application areas,

or kinds of data, which have been successfully visualized using radial techniques. They are:

- Hierarchical structures (trees), e.g., [37], [38], [53], [76]: Hierarchy visualization remains one of the most important problems in information visualization [3], [85]. Commonly used data sets include file systems and organizational hierarchies.
- Relationships among disparate entities, e.g., [5], [19], [49], [55]: In many kinds of multidimensional data, the relationships among the several variables are often not apparent. Examples include computer network traffic and alerts, demographic surveys, and social networks.
- Ranking of search results, e.g., [18], [33], [72], [79]: The ubiquity of search engines in modern computing make this application of radial visualization particularly compelling. While the actual problem of ranking search results is more a pure algorithmic problem than a visualization problem, visualization is nonetheless an effective means of communicating the relative rankings to the user so that a final decision can be made.
- Serial periodic data, e.g., [12], [74], [82]: This refers to data that is continuous while manifesting a predictable repetitive structure. The most common example is time-series data.

Note that this list is neither exclusive nor exhaustive. Other visualization strategies exist for these domains, just as radial visualization may likewise be applied to other domains not mentioned here. However, the application areas listed above are representative of radial visualization systems built in the past two decades.

4 DESIGN PATTERNS

While all current radial visualization designs ultimately stem from centuries old statistical graphics, current radial visualizations exhibit a remarkable diversity in their visual design and methods of interaction. Based on our survey of the literature, we propose seven high-level design patterns, or archetypes, which describe nearly all radial visualization systems built in recent years. The patterns are listed in Table 1.

These design patterns emerged gradually as we reviewed and analyzed the vast body of visualization techniques in which radial methods are utilized to some extent. We first observed three basic recurring themes based on their visual appearance. Some visualizations were “spiky,” consisting mainly of line segments radiating from a common central point. Others were more dense, filling most or all of the area of a circle. Still others packed most of the nodes around the circumference. These characteristics gave rise to the three main divisions shown in Table 1: **Polar Plot**, **Space Filling**, and **Ring**.

However, this simple trichotomy alone is insufficient to capture the high degree of variance among visual interfaces utilized in radial visualization. Accordingly, we further subdivided our schema into seven themes, or design patterns, to better reflect the nuances exhibited by current systems.

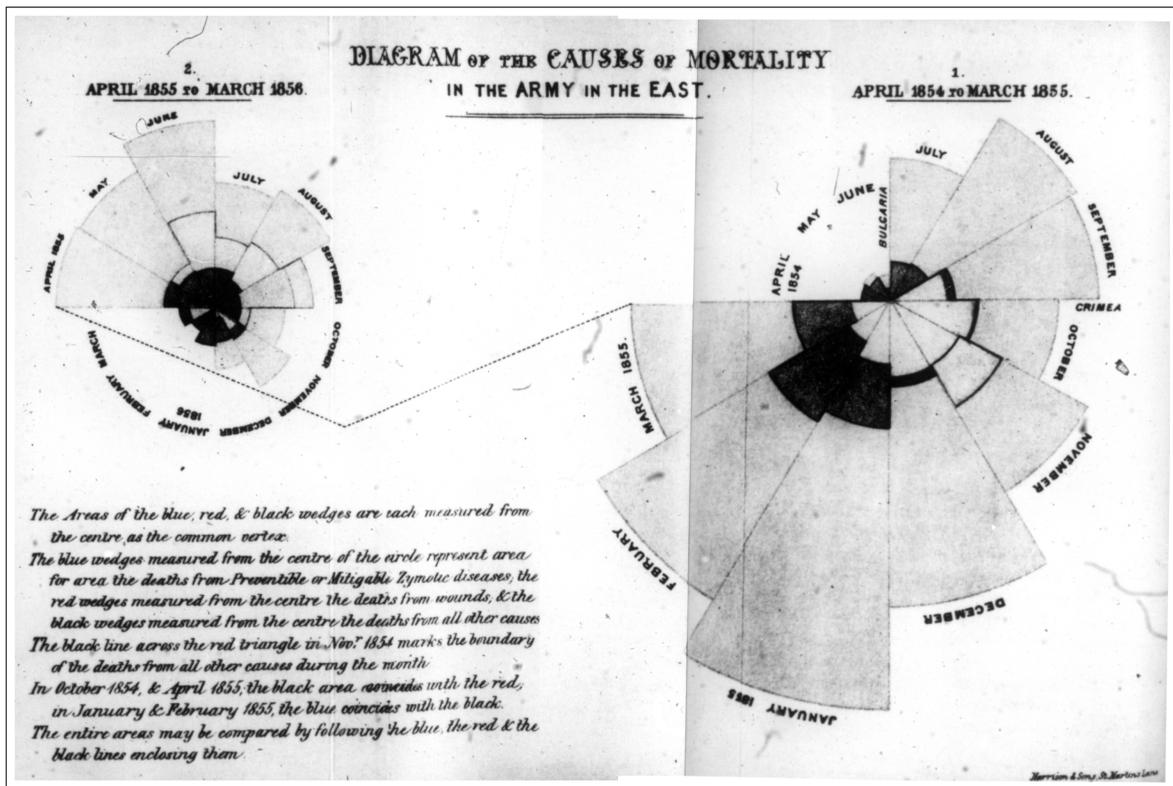


Fig. 3. Florence Nightingale's polar area chart.

4.1 Polar Plot

The Polar patterns are divided into two camps: **Tree** and **Star**. In either one, the center of the graphic carries some special meaning. For example, it may represent the root node of a tree, the origin of a coordinate system, or search terms in a query. Other nodes extend radially outward from the center. Unlike a star plot, the radii of a Polar Plot visualization are not necessarily all the same length. The linear distance between a node and the center typically carries some semantic significance. Furthermore, some meaning is often ascribed to the size at which a node is rendered.

4.1.1 Tree Pattern

Defining characteristics. Origin at or near center of canvas; line segments radiating outward from origin; other segments branching off from these; may be nested many levels deep.

Common uses. Viewing and browsing hierarchical data structures, such as trees; viewing relationships among disparate entities.

The Tree pattern traces its roots to the *radial graph layout* algorithm used in some graph-drawing contexts. Early references to radial graph layouts include the "centrifugal representation" of Carpano [13] and the "ring diagram" of Reggiani and Marchetti [66], as well as the influential work by Eades [24]. Radial graph layouts are usually rendered in 2D, especially for information visualization contexts, but 3D variants also exist [14], [31], [44], [60], [67].

Inspired in part by the artwork of Escher [32], the *Hyperbolic Browser* [52], [54], [53] is a good example of an interactive radial visualization that follows the Tree pattern.

However, the Hyperbolic Browser allows the display and navigation of many more nodes than traditional tree layouts. This is done by laying out the hierarchy of the graph on the hyperbolic plane, which is then rendered as a standard (euclidean) circle. This has the desirable effect that nodes on the periphery of the user's area of interest appear smaller, allowing the user to focus on the nodes he or she is most interested in, while still preserving the overall context of how the current node fits into the global topology of the graph.

Extending the concepts introduced by the Hyperbolic Browser, the *MoireGraphs* visualization by Jankun-Kelly and Ma [43] facilitates comparison among nodes at the same tree level. A key idea to this end is their additional constraint that all nodes at the same tree level must share the same radial distance from the center point (Fig. 8). *MoireGraphs* also support a wider range of navigation techniques than the Hyperbolic Browser. Similar methods can be found in the RINGS system [76]. Emphasizing interactivity and efficient space usage, RINGS facilitates the browsing of large hierarchical structures such as filesystems.

Similar in appearance and spirit to Northway's target sociograms (see Section 2), Brandes et al. [8] present a diagram for visualizing the communication patterns among organizations within a municipality (Fig. 9). The radius of an icon is based on the *centrality*, or importance, of that actor (entity) within the organization. Directed edges between icons represent one-way communication; undirected edges denote two-way communication. Additional semantic information is communicated by the size, color, and shape of an individual icon. Interestingly, the angular position, or *theta*, of an icon is computed purely for aesthetics and carries no direct

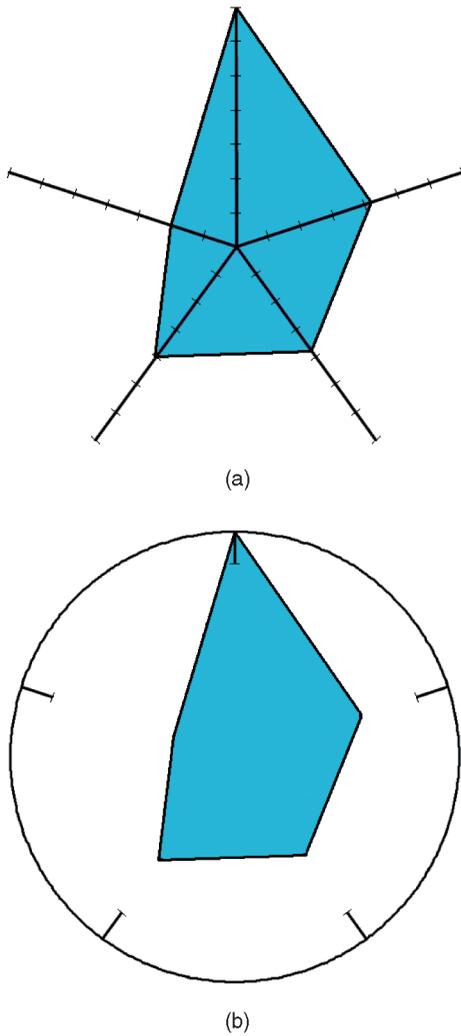


Fig. 4. Two equivalent renderings of a star plot. (a) Star plot with visible axes. (b) Star plot with invisible axes.

semantic meaning. However, the indirect benefit is that it helps reduce edge-crossings, which would otherwise obscure meaning [65].

The *Visual Thesaurus* [77] from Thinkmap, Inc., is a commercial Java applet based on the Tree pattern. The user keys in a word, and the system responds with an animated

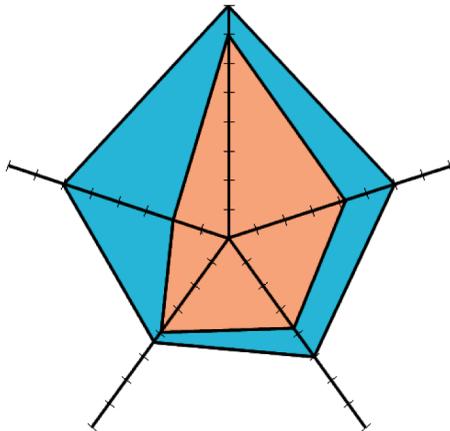


Fig. 5. Radar plot showing differences in relative area.

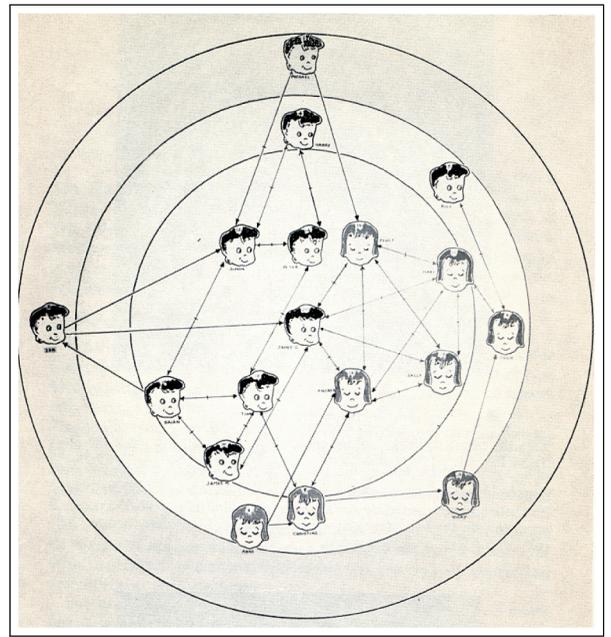


Fig. 6. The target sociogram, an early example of radial display (reprinted from [62] by permission of University of Toronto Press).

visualization of that word and all its synonyms arranged in a radial tree structure (Fig. 10). The user can then further refine the query by selecting one of the nodes in the display.

4.1.2 Star Pattern

Defining characteristics. Origin at or near center of canvas; line segments radiating outward from origin; no branching.

Common uses. Ranking of search results; viewing relationships among disparate entities.

Our next design pattern is the Star pattern. The Star pattern finds its origins in the star plot, as discussed in

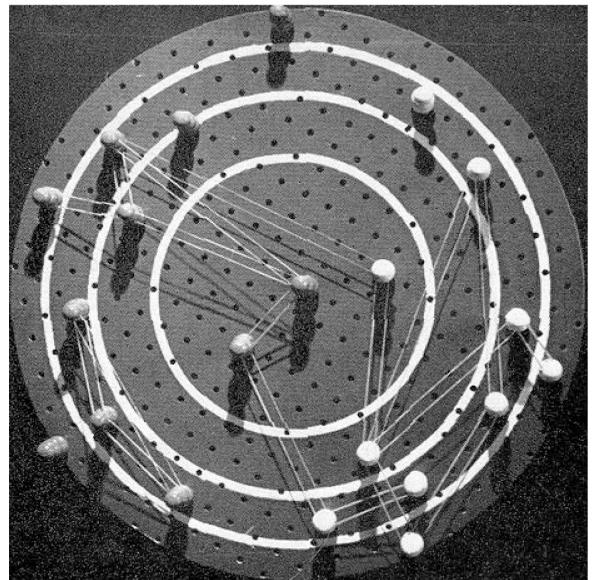


Fig. 7. The peg board sociogram, an analog radial visualization (reprinted from [62] by permission of University of Toronto Press).

TABLE 1
Design Patterns for Radial Visualization

	Pattern	Examples
Polar Plot	<i>Tree</i>	[43], [53], [76]
	<i>Star</i>	[33], [34], [71]
Space Filling	<i>Concentric</i>	[18], [49], [85]
	<i>Spiral</i>	[12], [20], [82]
	<i>Euler</i>	[5], [37], [59]
Ring	<i>Connected</i>	[19], [55], [69]
	<i>Disconnected</i>	[9], [23], [74]

Section 2.2. In contrast to the Tree pattern, in which there may be any number of intermediate nodes between the root and the leaves, systems using the Star pattern typically connect leaf nodes to the root with a single straight line segment. This is at once a simplification and a specialization. On the one hand, Star-based visualizations tend to be smaller and less visually complex than those employing the Tree pattern. However, the Star pattern also offers less variety in its visual representation and affords fewer options for interactivity.

The *Starstruck* [34] system uses the Star pattern for showing relationships among themes within a document. The center of the visualization represents a given document and each theme is depicted as a small icon at the end of a line segment emanating from the center point. The length of the line represents the strength of that theme in the document. The distance from one line to another represents how closely the two themes are related one to another (Fig. 11).

The *Neighbourhood Explorer* [71] system applies the Star pattern to the domain of real estate. It allows the user to browse through a graphical list of homes for sale. Multiple axes extend radially outward from a picture of a *reference home* in the center of the visualization. Each axis represents a different home attribute, such as price, number of bedrooms, or acreage. Pictures of other homes are positioned on the axes based on how they compare with respect

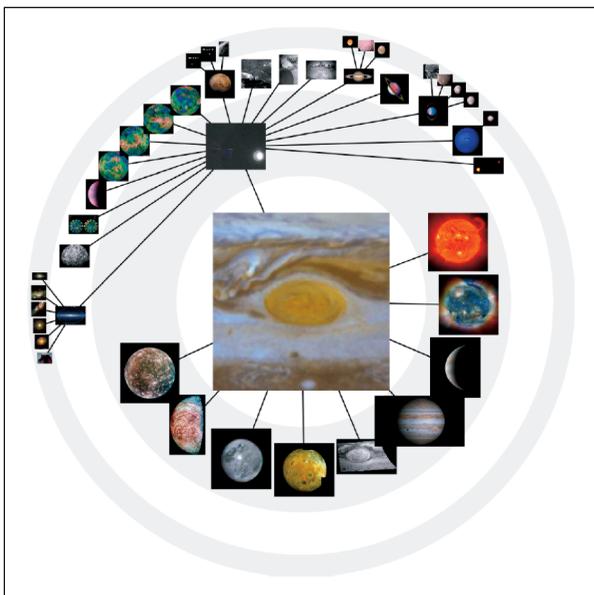


Fig. 8. MoireGraph (reprinted from [43]). ©2006 IEEE.

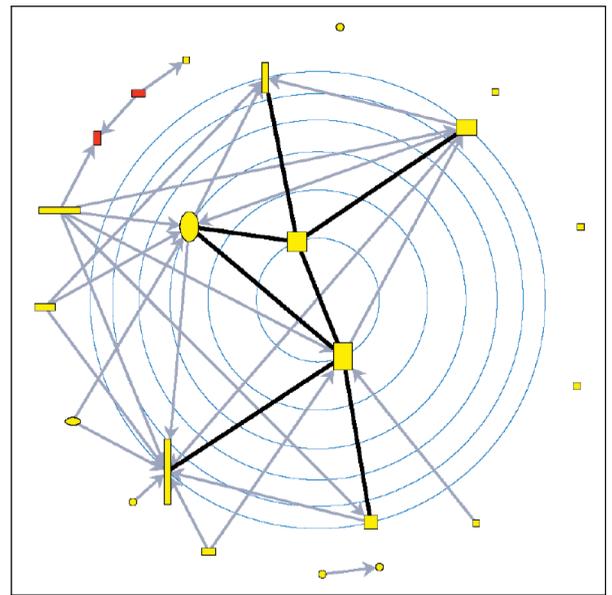


Fig. 9. Policy network diagram (reprinted from [8]). ©2003 IEEE.

to a selected attribute relative to the reference home. A comparison home can appear on more than one axis.

In the *Sparkler* system [33], Havre et al. propose a Star-based visualization for performing queries on a collection of documents. Their visualization supports the depiction of multiple queries at once. Icons representing the various queries are placed in the center of the graphic. The results to each query are represented as point-clusters at various radii around the circle. The distance between a particular result and its query icon indicates the estimated relevance of that result to the query.

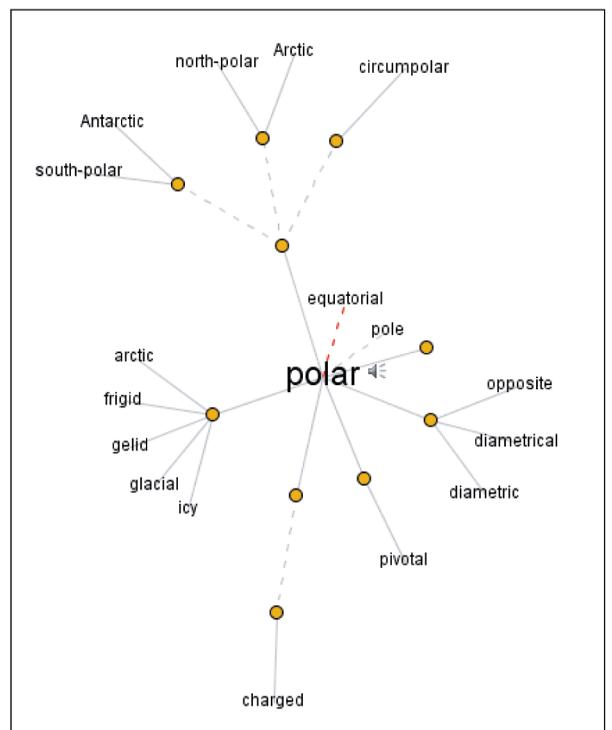


Fig. 10. The Visual Thesaurus. (Image courtesy Thinkmap, Inc.)

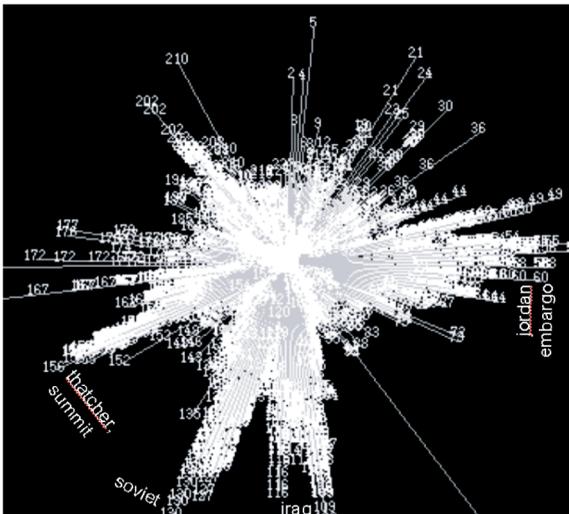


Fig. 11. Starstruck, a thematic document visualization (reprinted from [34]). ©1998 IEEE.

Ishihara et al. [42] apply the Star pattern to the task of displaying query results. As with other systems, the results are rendered as icons positioned at various distances from a central node. Their approach is distinguished from other systems in that the distance of each node to the center is based on the *timestamp* rather than a qualitative measure of “relevance.” Less current results are rendered farther from the center.

Star Coordinates [47] is another multidimensional visualization strategy bearing a strong resemblance to the other Star-based systems described above. Likewise, *Star Coordinates*’ data points are plotted relative to one of several attribute axes. In contrast, however, the points are not guaranteed to fall within the range suggested by the length of the axis. Indeed, the points may lie anywhere in the graphic, so determining which attributes correspond to a given point can be ambiguous. This visualization thus bears some visual similarity to scatterplots [6]. Optionally, the points can be scaled and normalized to fit within the parameters of the axis.

The *DataMeadow* visualization by Elmqvist et al. [25] uses a series of linked *DataRose* diagrams to facilitate queries. A *DataRose* is essentially a star plot with a slider on each axis to dynamically adjust the terms of the query. The output of one *DataRose* can be fed into the input of another *DataRose*, allowing compound queries to be created (Fig. 12).

We end our discussion of the Polar Plot patterns with an example from HCI rather than information visualization *per se*. In contrast to traditional GUI menus in which all the options are arranged in a straight line, a *marking menu* [51] presents the options in a radial arrangement. Indeed, marking menus are sometimes alternately called *radial menus*. A marking menu has the attractive characteristic that every selection option is equidistant from the cursor. Thus, experienced users can access the desired functionality more quickly and accurately [11] than with linear menus, because they are freed from having to iterate through a sequential list of undesired options. A marking menu can also have submenus; if a particular menu option triggers a submenu, then a new marking menu appears (Fig. 13). Thus, all that is

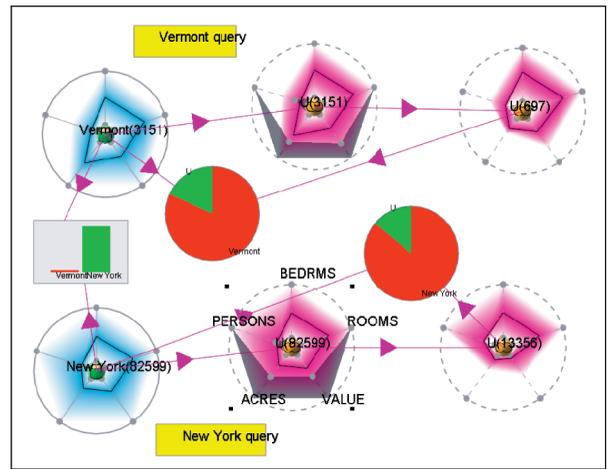


Fig. 12. DataMeadow visualization showing multiple linked DataRoses (reprinted from [25]). ©2007 IEEE.

required for a user to navigate a hierarchy of menu options is to move the cursor in a specific pattern. After repeated use, the user may become reflectively familiar with the required motions for certain frequently used options that menu navigation occurs with hardly a glance. In this sense, marking menus are similar to *gesture-based interfaces* [68], except that in a gestural interface, no menu is shown; the user must memorize the gesture required to initiate a request.

Marking menus do not fit cleanly into either the Tree or the Star pattern. In the simplest case, a marking menu follows the Star pattern in that each menu option is directly connected to the root node. However, the existence of submenus opens up the possibility for deeply nested menu structures more appropriately represented by the Tree pattern.

4.2 Space Filling

The nonuniform distribution of nodes in the Tree and Star patterns leads to the criticism that they do not utilize space efficiently. The next three patterns, **Concentric**, **Spiral**, and **Euler**, known collectively as *Radial Space Filling* (RSF) [85] patterns, attempt to overcome this. RSFs combine the solid appearance of a pie chart (see Section 2.1) with the polar plot techniques described above. As in the Tree and Star patterns, the center point of an RSF usually carries some significance. However, instead of laying out information like the spokes of an uneven wheel, RSFs typically arrange their data as concentric circles, tight spirals, or evenly spaced clusters. The defining characteristic of an RSF is that

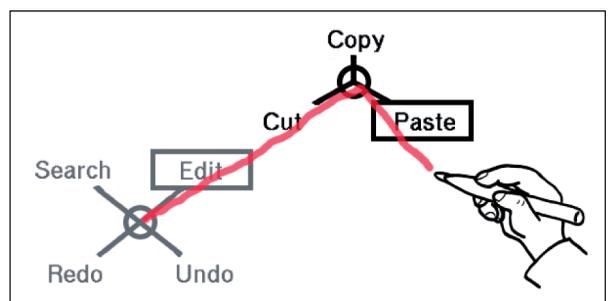


Fig. 13. Marking menus (Adapted from Kurtenbach and Buxton [51]).

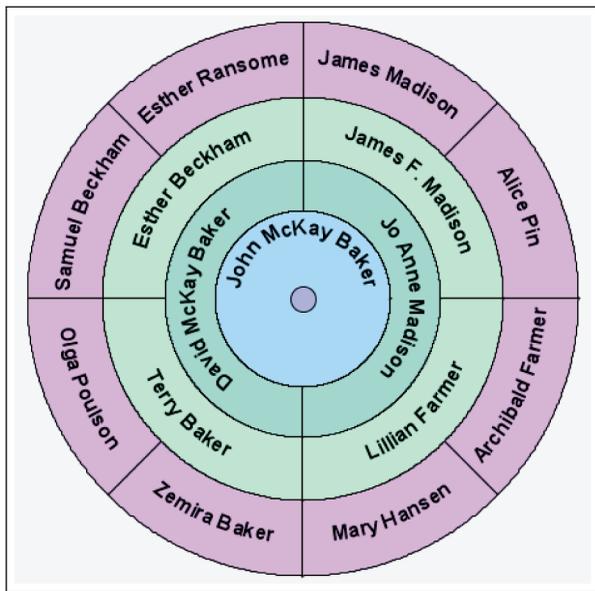


Fig. 14. Fan chart, a radial display for genealogical graphs.

it fills up its allotted space. This usually, but not necessarily, means that it fits in a clean circle, without jagged edges. Indeed, although the term “space-filling” is commonly used to describe this class of radial visualization, Stasko and Zhang [73] acknowledge that it is somewhat of a misnomer, given that radial visualizations rarely fill the entire (rectangular) display area.

4.2.1 Concentric Pattern

Defining characteristics. Origin at or near center of canvas; concentric rings fan outward; each ring divided into multiple sectors.

Common uses. Viewing and browsing hierarchical data structures, such as trees; viewing relationships among disparate entities.

Perhaps the most prevalent of the RSFs, the Concentric pattern may be considered the space-filling analog of the Tree pattern.

The *fan chart* is a well-known visualization technique among genealogy enthusiasts. A fan chart places the root (descendant) at the center of the circle, with branches (ancestors) radiating outward in concentric rings (Fig. 14). Fan charts are an attractive alternative to standard pedigree charts for viewing an individual’s ancestry because the densest parts of the tree are drawn on the outer rings of the circle, where the available space is greatest. Most commercial genealogy software can draw fan charts in a variety of configurations, typically as full circles, semicircles, or quarter circles. Although traditionally rendered as a static diagram, interactive variants have also been proposed [22].

TreeMaps [45] are not generally considered a radial visualization. Nevertheless, a circular variant of TreeMaps called *Polar TreeMaps* was proposed by Johnson [46]. The *Filelight* system [38] (Fig. 15) is a usable filesystem browser based in part on the Polar TreeMap metaphor.

The *InterRing* system [85] presents a visualization strategy for hierarchical data similar to *Filelight*’s, but the emphasis of *InterRing* is on the mode of interaction rather than on the

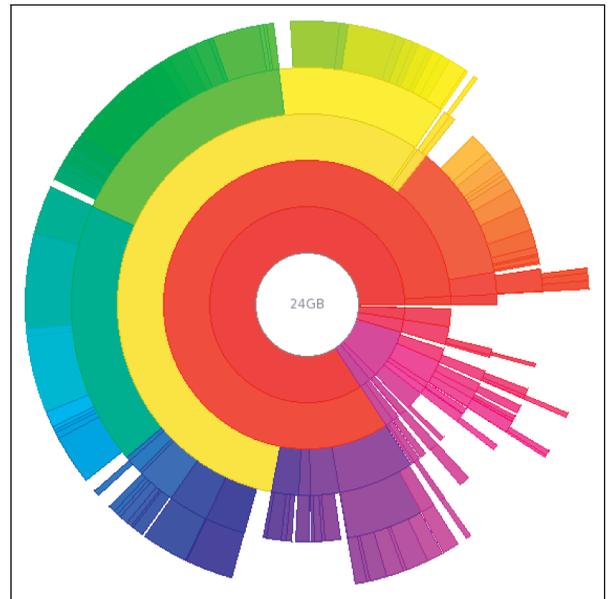


Fig. 15. The *Filelight* filesystem browser is an example of a concentric RSF that does not fit into a clean circle. (Image courtesy methylblue.com.)

information being displayed. It is interesting to note that of all the radial visualizations reviewed herein, *InterRing* is one of very few that supports interactively *modifying* the underlying data structure. The majority allows the user only to browse the data in a “read only” fashion.

Despite the Concentric pattern’s usefulness for examining hierarchical data, one weakness is that the leaf nodes in a large hierarchy tend to be rendered too small [85]. A number of solutions have been proposed to this problem. For example, Chuah [16] allocates a greater proportion of the circle to nodes with more children. Andrews and Heidegger [1] employ a split-screen approach, where one half of the screen shows a semicircular diagram and the other half shows a “zoomed-in” view of a leaf-node section of the diagram, so that details of interest can be explored more fully. Stasko and Zhang [73] use a similar strategy, except that in their system, the global view of the data is scaled down, and the zoomed-in details are either wrapped around the circumference of the global view, or placed inside the center of the visualization. This avoids the distraction of a split-screen view. Cava et al. [15] use an approach similar to [1], one which they call a *Bifocal Tree*. The main visual difference is that they use a Tree layout (see Section 4.1.1) instead of a Concentric RSF.

Another common application of the Concentric RSF pattern is to visualize relationships within multidimensional data. In the *circle segments* technique [2], [48], each dimension is mapped to a wedge in the circle. Within each wedge, different values for the attribute are represented by colored sectors. This general idea was later applied to time-based data in Keim et al.’s *Circle View* metaphor [50]. Each wedge in the circle is divided into multiple sectors representing time intervals. Each sector is color-coded according to the value of the attribute during that time interval. The *Radial Traffic Analyzer* system [49] extends the circle segments concept to the problem domain of visualizing network traffic. Radial

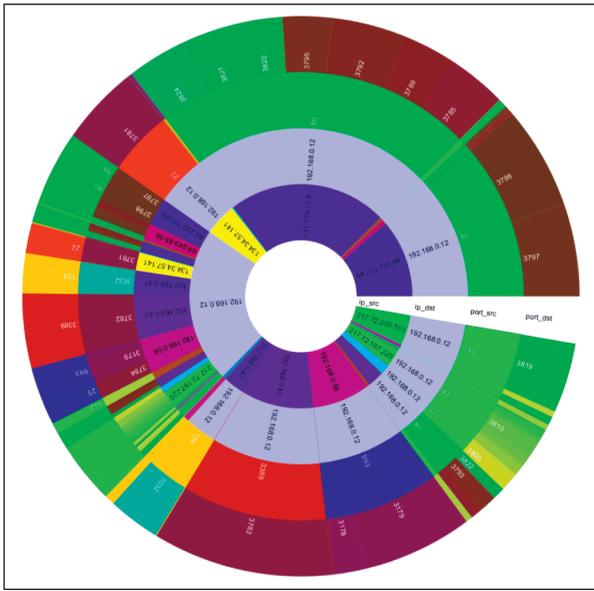


Fig. 16. Radial Traffic Analyzer (reprinted from [49]). ©2006 IEEE.

Traffic Analyzer utilizes four concentric colored rings: the two inner rings represent the source and destination IP addresses, while the outer two rings represent the source and destination ports. IP addresses and ports are color-coded (Fig. 16).

4.2.2 Spiral Pattern

Defining characteristics. Origin at or near center of canvas; spiral-shaped glyph emanating from origin.

Common uses. Viewing serial periodic data, such as time-based data.

In the Spiral pattern, nodes are placed along the curve of a spiral-shaped glyph. The spiral may either be a thin curve or a solid swath of color; this is an implementation detail and does not significantly affect the structure of the pattern.

In the definitive paper on Spiral RSFs, Carlis and Konstan [12] observe that by arranging serial periodic data in a spiral shape, certain patterns in the periodicity of the data are easier to detect than in traditional line-plot diagrams.¹ Their definition of serial periodic data is the data that “has a serial (continuous) dimension that exhibits periodicity.” Examples of serial periodic data include the growth rate of trees, students’ time allocation throughout their academic careers, or even the regular, predictable progression of tones from one octave to another on a piano keyboard. Carlis and Konstan’s visualization places serial data along the curve of a spiral, but enforces that each “period” of the data occupies one complete revolution of the spiral, with corresponding elements within each period occupying the same angle. Carlis and Konstan use a diagram similar to Fig. 17 to track the annual feeding patterns of chimpanzees. The 12 axes correspond to the 12 months in a year, with each complete loop corresponding to one year. By following one axis from

1. The use of spirals to visualize periodic data was reintroduced in 2001 by Weber et al. [82]. Although typically quite similar to Carlis and Konstan’s 1998 paper, Weber et al. do not reference it, so one may surmise that both groups likely invented spiral graphs independently.

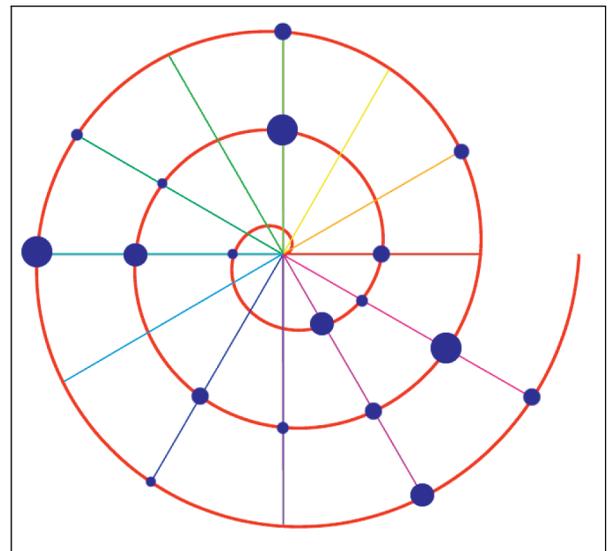


Fig. 17. Radial visualization of serial periodic data (adapted from Carlis and Konstan [12]).

the center outward, one can see the data’s variance over a number of years at the same time each year.

The *SpiralView* system [7] uses a similar visualization, but applies it to the problem of monitoring alerts on a computer network. By employing a periodic Spiral layout, *SpiralView* enables analysts to spot recurring patterns of network activity.

An even earlier example of a Spiral interface can be found in the NIRVE system [18] for displaying results of a document-based search. It introduces several motifs that would appear in later efforts; namely, the placement of more relevant results near the center of the display. However, NIRVE uses a 3D approach, which means that some results will obscure others, and parallax is necessary to derive meaningful information. This visualization has also been criticized for using space inefficiently [72].

Torres et al. [79] use a radial visualization to browse results from an image-based search engine. In contrast to many current image search engines, the terms of the query itself are images, not text. Hence, the search compares how similar the resultant images are to the reference (query) image. In their prototype implementation, the user can select either a Concentric interface or a Spiral interface. In either case, those images that resemble the reference image more strongly appear closer to the center of the display.

RankSpiral [72] is another spiral interface for search engines. As with Torres et al. [79], the top search results appear near the center of the visualization. An interesting feature of *RankSpiral* is that the title of each Web page found in the search is displayed alongside the icon representing the page. Occlusion events are averted by widening the spiral sufficiently to accommodate the title text.

Spiral RSFs have also found application in portable devices and ambient systems [64]. In one of the first applications of radial visualization to handheld computers, the *SpiraList* system [39] applies a Spiral layout to the problem of locating people in a handheld computing device’s address book. Taking advantage of the device’s

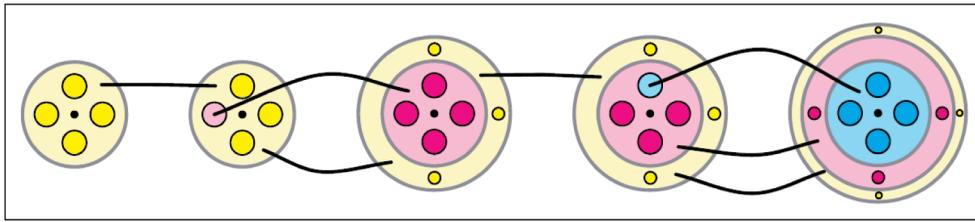


Fig. 18. Browsing a tree structure using MoireTrees (reprinted from [59]). ©2005 IEEE, Eurographics.

touchscreen capability, a thumb is used to scroll through names in the address book. In the *SpiraClock* ambient system by Dragicevic and Huot [20], the familiar metaphor of an analog clock face is mapped to a Spiral RSF. *SpiraClock* displays upcoming events around the circumference of a spiral and continuously updates them to reflect the passage of time. *SpiraClock* has the additional benefit that it does not require a great deal of screen real estate to be effective, and can run as a small applet or widget in the periphery of the user's workspace.

Despite its name, the *Spiral Calendar* system [57] is a Spiral interface only in the loosest sense [12]. In this case, the spiral shape is just a decorative side effect; it is not central to the interpretation of the data nor the interactivity of the system. The *Spiral Calendar* shows a 3D view of a calendar widget spiraling toward the viewer, progressing from coarse to fine resolution (i.e., years, months, weeks, and days).

4.2.3 Euler Pattern

Defining characteristics. Multiple circles placed inside (or adjacent to) a larger circle; may be nested many levels deep.

Common uses. Viewing and browsing hierarchical data structures, such as trees; viewing relationships among disparate entities.

The third RSF design pattern is the Euler pattern, which is essentially the interactive analog to the Euler diagram [4].

The *Zoomology* system [37] is the archetypal example of the Euler pattern for radial visualization of hierarchical data. In contrast to the more common approach of placing the root node in the center of the visualization, *Zoomology* treats the outer ring as the root, with the child nodes embedded within. In general, each node's children are rendered as inner circles. The user interface permits the user to zoom into any section of the diagram (hence, the name). The authors compare the operation of zooming into ever-smaller rings to "flying through a tunnel," suggesting the importance of smooth animation to preserve the user's sense of context from one level of detail to the next.

Combining the ideas of *Zoomology* with *MoireGraphs* [43], the *MoireTrees* system [59] also supports the notion of zooming into one of several child nodes. However, *MoireTrees* aims to preserve an even greater sense of context than *Zoomology*. Specifically, for any node in the hierarchy, *MoireTrees* displays the entire path from the root node to the current node. Each time, the user drills down one level into the hierarchy, the previous level is shown as a ring encompassing the current level. Thus, after multiple expansions, the path from the current node to the root is

shown as a series of concentric rings around the circumference of the display (Fig. 18).

Although originally conceived as a 3D tree layout algorithm, a 2D mapping of the *PhylloTrees* method [14] yields a space-filling radial diagram consisting of tightly packed nodes. Its algorithm is based on phyllotactic patterns found in nature; e.g., the arrangement of seeds in a sunflower head [81].

In an interesting spin on the Euler pattern, Van Berendonck and Jacobs' *Bubbleworld* system [5] is designed to display results of queries on a collection of documents. In contrast to *Zoomology*, *Bubbleworld* uses clusters of adjacent rings (or "bubbles") on a plane to denote relationships among topics. Key words from the user's search terms are positioned as nodes around the bubble, and icons of various shapes and colors are clustered within the bubble's interior to show the results of the query. Icons are positioned inside the bubble according to the relevance of their underlying document to a specific key word. The user can refine a query, spawning additional bubbles which are rendered tangentially to a key word on the parent bubble's circumference. In this way, the visualization provides an explicit trail of which key words link to related queries.

We close our discussion of the RSF patterns by briefly mentioning a space-filling variant of the marking menu. Known as a *pie menu*,² its options are rendered as wedges rather than edges and vertices. Pie menus closely resemble pie charts and do not fit cleanly into any of the three RSF patterns.

4.3 Ring Based

Our last two design patterns for radial visualizations are the **Connected Ring** and **Disconnected Ring** patterns. The Ring patterns borrow ideas from both the Polar Plot and Space Filling patterns. As in the RSFs, the visualization is usually constrained to fit inside the confines of a circle. Nodes of interest are placed around the circumference of a ring. The center point of the circle usually has little significance on its own, although additional information may nonetheless be displayed within the ring's interior.

4.3.1 Connected Ring Pattern

Defining characteristics. Nodes positioned around circumference of ring; line segments connecting nodes; additional nodes optionally appear in ring's interior.

Common uses. Viewing relationships among disparate entities.

2. The terminology used in the literature is not always consistent. Occasionally, the term "pie menu" is also used to refer to a standard (nonspace-filling) marking menu.

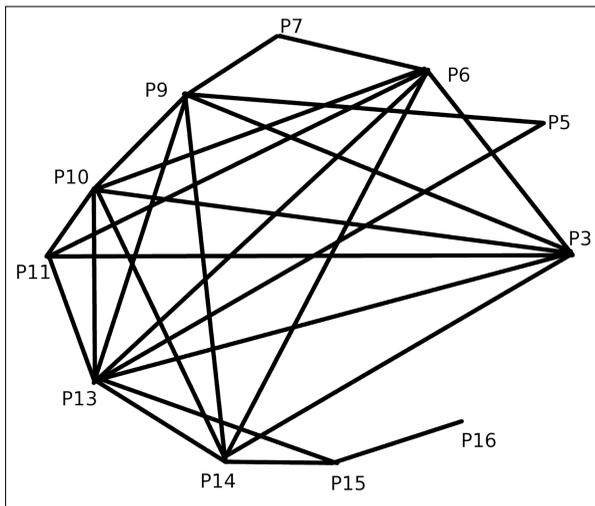


Fig. 19. Text Relation Map (Adapted from Salton et al. [69]).

The most common application of the Connected Ring pattern is to show relationships among nodes in a data set. This information is typically shown by visually linking data points together. Typically, the links are rendered as line segments between two nodes, although spline-based curves are also sometimes used.

An early example of the Connected Ring design pattern is the *text relation map* proposed by Salton et al. [69]. The application domain here is the automatic generation of hyperlinks between articles in a digital encyclopedia. A text relation map (Fig. 19) shows the traversal of a link from one section of text to another. The different articles are represented as nodes along the circumference, and links between articles are rendered as lines from one node on the circumference to another. Salton also introduced a *paragraph similarity map* for a similar purpose. In a paragraph similarity map, articles are represented as curved sectors along the circumference. The radial magnitude of a sector corresponds to the article’s length relative to other articles. If a paragraph in one article references content in another article, a line segment is drawn between the two sectors. The endpoints of the line segment along the two sectors are positioned relative to the location of the referring paragraphs within the body of the article.

The *Daisy* system [19] is a commercially available software package that employs a Connected Ring-based visualization scheme (Fig. 20). Groups of related entities are represented as sectors around the circumference. The interior of the ring is reserved for straight lines depicting relationships or links between the entities on the circumference. Moreover, small histogram bar charts are placed on each node in the circumference, conveying additional information.

NetMap [30] is another commercial visualization system, designed primarily for law enforcement agencies, insurance companies, or other organizations that need to find links among large volumes of disparate data. *NetMap*’s visualization metaphor is similar to *Daisy*’s, in many respects, in that related variables are grouped sequentially around the circumference of a ring. As with *Daisy*, a line is drawn between two nodes to indicate a relationship between

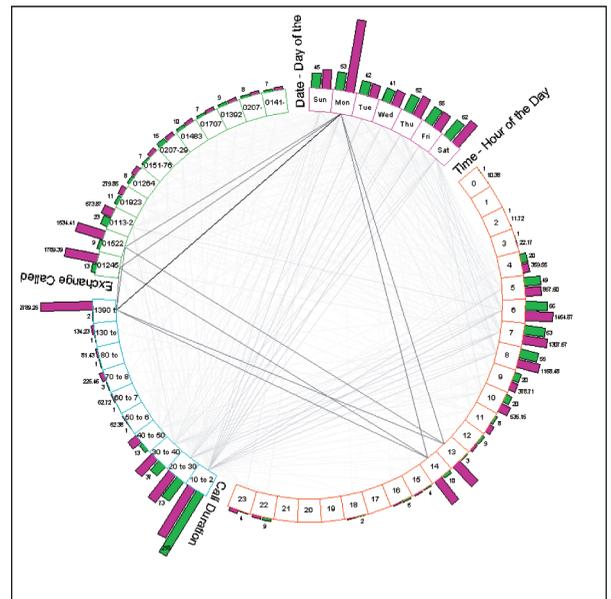


Fig. 20. Daisy Analyzer (Image courtesy of Daisy Analysis, Ltd.)

them. However, *NetMap* does not include histograms along the circumference.

Tominski et al. [78] introduce two multidimensional radial visualizations, the *TimeWheel*, and the *MultiComb*. In *MultiComb*, multiple variable plots are arranged around a circle. *MultiComb* supports two arrangements of the plots. In one case, the plots run parallel to the circumference of the circle. In the other, the plots extend outward from the center of the circle (Fig. 21a), similar to *Daisy*’s histograms. In *TimeWheel*, an axis representing an independent variable (such as time) is placed in the center, and axes representing multiple dependent variables are positioned around the central axis, forming an approximate ellipse (Fig. 21b). Relations between the dependent and independent variables are rendered as colored lines.

In the Connected Ring visualizations discussed thus far, the primary means of correlating data points are to draw a line between a node on one position of the circumference to another node elsewhere on the circumference. While this is effective for many applications, one drawback to this approach is that the interior of the ring becomes a veritable cloud of intersecting lines, making certain relationships difficult to spot.

TimeWheel, as described above, addresses this issue somewhat by connecting points on the circumference to a central node in the ring’s interior, rather than to other points on the circumference. This reduces the average length of the line segments, which, in turn, reduces the visual clutter of the interface.

The *VisAlert* system [29], [55], [56] takes this approach one step further. The problem domain of *VisAlert* is computer network intrusion detection. The interior of the ring is reserved for a schematic of the network topology, while possible alerts are placed around the circumference. For each alert experienced by a node on the network, a line is drawn connecting the alert’s icon to the affected node’s icon (Fig. 22). By placing additional information (i.e., the network map) *inside* the ring, this area is transformed from

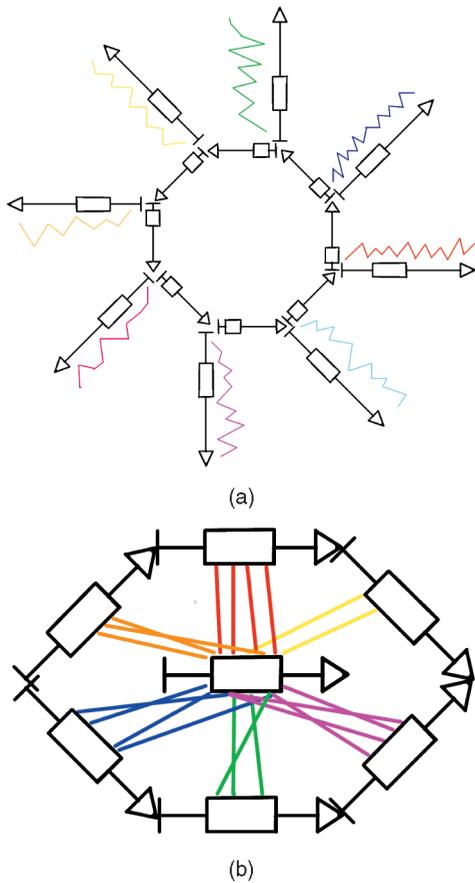


Fig. 21. Two visualizations for multidimensional visualization (Adapted from Tominski et al. [78]). (a) MultiComb. (b) TimeWheel.

a passive zone of connecting lines, to a canvas on which important domain-specific data are displayed.

The *Manager's Dashboard* system [21] employs a visualization metaphor similar to *VisAlert's*, but with a greater

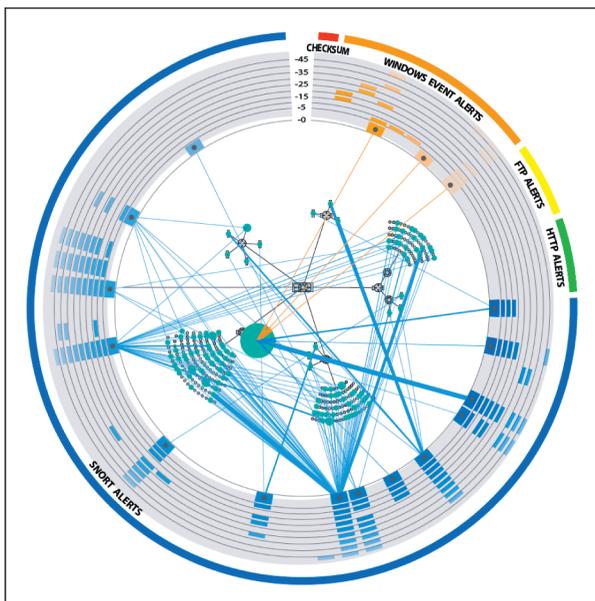


Fig. 22. *VisAlert's* user interface (reprinted from [56]). ©2005 IEEE.

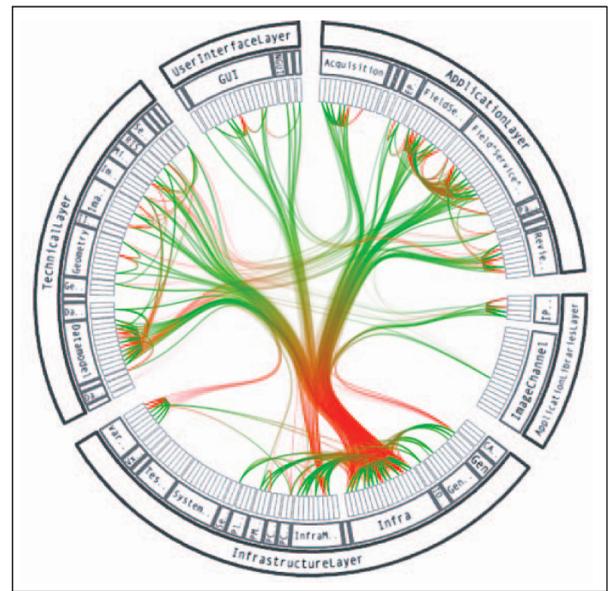


Fig. 23. Bundling edges to reduce visual clutter (reprinted from [36]). ©2006 IEEE.

emphasis on interactivity. Queries are initiated by dragging an icon from the circumference into the ring's interior. Line segments are then rendered, connecting the icon(s) in the interior to icon(s) on the circumference wherever a relationship exists between the entities represented by the icons.

Regardless of whether the lines converge on the ring's circumference (as in *Daisy* [19] and *NetMap* [30]) or in the ring's interior (as in *VisAlert* [55] and *Manager's Dashboard* [21]), one must ask how a system that relies on converging line segments to indicate correlation can avoid eventual scalability issues as the number of nodes and lines grows. One approach to this problem, suggested by Holten [36], is to "bundle" similar edges together near the center of the circle and branch out gradually toward the circumference (Fig. 23). The edges are rendered as B-spline curves so that the branches appear less abrupt to the human eye. (A similar bundling technique is often employed by airline route maps, in which a vast majority of flights originate from a relatively small number of "hub" cities.)

We close our discussion of Connected Ring systems with examples of this pattern applied to 3D. Most 3D radial systems are graph-drawing implementations, and therefore are instances of the Tree pattern. However, the problem domain of representing geographic relationships on a world map is a natural fit both for the Connected Ring pattern and 3D. The *GeoSOM* system by Wu and Takatsuka [84] has been used to visualize the flow of international trade; however, each country's position on the map is determined by similarity in attributes rather than actual geographic location. The *VIBall* system [88] takes a somewhat different approach. It depicts Internet traffic between countries as a series of lines projected onto a geographically accurate world map. In contrast to many 3D visualization applications, *VIBall* is a true 3D interface: two video projectors are used for displaying the map of the world onto a white plastic sphere. The sphere is set up on rollers, so that it can be rotated via direct manipulation, and the projected display is updated appropriately.

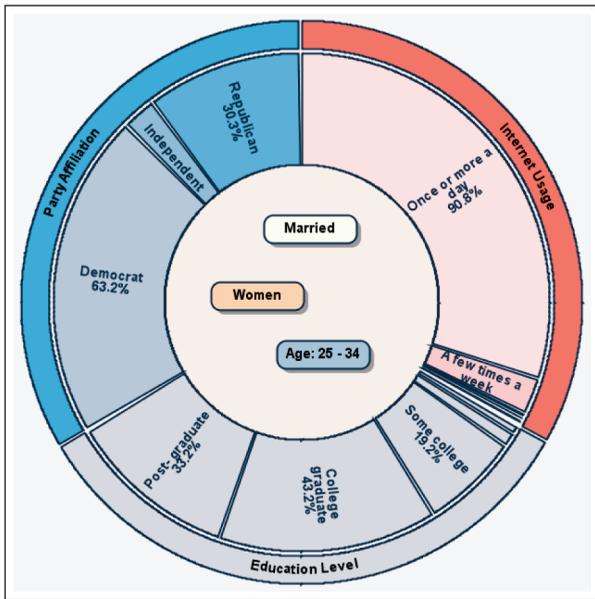


Fig. 24. Viewing poll results for a given subpopulation.

4.3.2 Disconnected Ring Pattern

Defining characteristics. Nodes positioned around circumference of ring; additional nodes optionally appear in ring's interior.

Common uses. Viewing relationships among disparate entities.

The Disconnected Ring pattern emerged in response to the inevitable scalability issues associated with rendering links between icons. Even with the edge-bundling technique [36] described in the previous section, eventually, the display becomes too cluttered as the number of edges increases. This, then, is the basic difference of the Disconnected Ring pattern: no explicit links are rendered. It follows that the icons themselves must bear the burden of communicating salient information about the data, either through their shape, position, color, or labels.

A recent example of the Disconnected Ring pattern is the *SQiRL* system [23]. Designed to visualize the results of opinion polls, a percentage breakdown of respondents' answers to selected questions is placed around the circumference. By default, the percentages reflect the survey population as a whole, but specific subpopulations can be designated by placing icons representing refinement criteria within the ring's interior (Fig. 24).

The *Event Tunnel* system [74] exhibits many of the qualities of a Disconnected Ring pattern, but extends them to 3D. The application domain here is the visualization of tasks' progress over time. Each unit of time (for example, a day) is represented as a ring, with the rings stacked so as to form a cylinder. At each level of the ring, icons representing simultaneous events are placed around the circumference. The cylinder is oriented toward the display such that the user looks "down" the cylinder, with recent events closer to the front and past events stretching away into the distance. Related events are grouped by color. In this way, the user can see how long a particular task took, and whether its

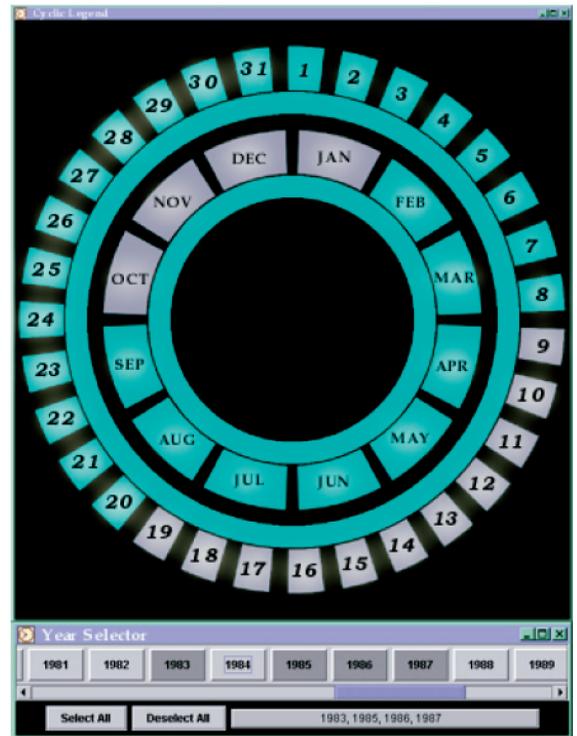


Fig. 25. A radial date-selection widget (reprinted from [9]). ©2000 IEEE.

progress was continuous or punctuated with bursts of activity between periods of inactivity.

We conclude our discussion of the Disconnected Ring pattern with two more examples of radial widgets taken from the broader field of HCI. In a novel alternative to the traditional calendar widget, Brewer et al. [9] introduce a ring-based navigation widget (Fig. 25) as a part of their geographic information browser system. This widget displays controls for selecting the month and date in concentric rings, thus allowing the user to select any date in the year via a compact radial interface. Moreover, by displaying every month and date in a unified space, the user avoids having to traverse a linear sequence of months as in a typical calendar widget.

A radial menu that exhibits some qualities of the Disconnected Ring pattern is the *tracking menu* [28]. Similar in appearance to a marking menu or a pie menu, the tracking menu places icons for one kind of functionality around the circumference of the menu, while reserving the interior area of the menu for supportive options.

Table 2 presents a mapping of the design patterns proposed in this section to the usage scenarios reviewed in Section 3. We have provided references to example systems where the patterns and applications coincide. It should be noted that some systems may exhibit attributes of more than one pattern. In these cases, we identify it with the pattern that, in our judgment, is the most dominant. An empty cell in Table 2 simply means that we did not find a matching system in our search, not that a certain pattern is inherently unfit for any particular purpose. On the contrary, gaps in this table merely suggest opportunities for future research. For example, it is conceivable that a system using the Connected Ring pattern could be used for visualizing

TABLE 2
Typical Applications of Radial Design Patterns

	Tree	Star	Concentric	Spiral	Euler	Connected	Disconnected
<i>Hierarchical structures</i>	[53], [76]	-	[22], [38], [85]	-	[37], [59]	-	-
<i>Ranking of search results</i>	-	[33], [34], [42]	[79]	[18], [72], [79]	[5]	-	-
<i>Serial periodic data</i>	-	-	-	[12], [82]	-	-	[74]
<i>Relationships among disparate entities</i>	[77]	[71]	[49]	-	-	[19], [30], [55], [69]	[23]

hierarchical structures or search results, but to our knowledge this has not yet been attempted.

5 DESIGN DIMENSIONS

The patterns in Section 4 describe the visual themes prevalent in recent radial visualization systems. In this section, we focus on various design *dimensions* [64] that system builders must consider in architecting new systems. These dimensions (or choices) are more abstract than the patterns, and consider not only the arrangement of widgets on a display, but also the needs of the application areas for which current systems are built. The dimensions are:

- unit of visualization,
- significance of centroid,
- level of interactivity,
- access permissions, and
- 2D versus 3D.

While it might seem counterintuitive to introduce these dimensions *after* the presentation of the design patterns, we believe that an understanding of the basic themes and usage scenarios of existing radial visualization systems is prerequisite to appreciating the issues surrounding the application of the design dimensions. The systems mentioned in Section 4 each address these considerations in their own way. Some of these dimensions may be applicable to information visualization systems in general, but we focus the present discussion to their specific application to radial visualization. We now describe each dimension in more detail, and discuss their relationship to the design patterns introduced in the previous section.

5.1 Unit of Visualization

The *unit of visualization*, in this context, refers to the graphical components that make up the radial visualization. Among the systems surveyed herein, we have already seen a great deal of diversity with respect to this dimension.

For example, in most instances of the Tree and Star patterns, the basic units are glyphs connected by line segments. Examples include Starstruck [34], MoireGraphs [43], and Hyperbolic Browser [53]. This approach is likewise used by some instances of the Spiral RSF pattern, such as the prototype by Carlis and Konstan [12]. Line segments are also a central feature of systems using the Connected Ring pattern, like VisAlert [55] and Daisy [19].

Other systems use glyphs or icons to represent nodes, but rely on position and proximity to communicate relationships with other data points, rather than connecting line segments. The most obvious example from traditional statistical graphics is the scatterplot [6]. This approach is likewise employed by a number of the systems reviewed herein,

across a spectrum of design patterns. However, it is perhaps most evident in the Euler RSF pattern, which by definition consists of a collection of disjoint nodes. See, for example, Zoomology [37] and PhylloTrees [14]. Moreover, the Spiral RSF pattern is also compatible with this approach; in these instances, the spiral shape is implicit in the positioning of the nodes rather than by an explicit spiral glyph. See, for example, RankSpiral [72] and SpiralView [7]. Systems using the Disconnected Ring pattern also rely on the position, size, and shape of glyphs to communicate information.

The third basic unit of visualization employed by radial visualization system is the sector. In most instances, the sectors do not stand alone, but are placed adjacent to other sectors to form a complete ring. Sectors are used primarily in the Concentric RSF pattern to form the concentric rings that make up the circle. Typically, systems using this pattern are designed to fill a complete circle, with no gaps. However, this is not necessarily the case. The Filelight system [38] is an example of a Concentric RSF that does not require the circle to be complete. In most examples of the Ring patterns (both Connected and Disconnected), the rings themselves consist of discrete sectors. A notable exception is the Event Tunnel [74], in which the rings are rendered as thin circles upon which small spherical icons are positioned.

In addition to selecting the unit of visualization from an aesthetic standpoint, builders of radial visualization systems must also consider the units' *representational fidelity* [64]. For example, will the units—glyphs, icons, or sectors—represent entities (rows) or attributes (columns) in a database? Will they represent numeric quantities, textual results, or abstract concepts? Since the display area of any information visualization system is limited, the choice made here can greatly affect the visualization's scalability.

5.2 Significance of Centroid

The next design dimension is concerned with what role, if any, the centroid (or center point) of the display canvas will play in the semantics of the visualization. This dimension most directly relates to the Tree, Star, Concentric, and Spiral patterns. In traditional graph visualization, the center point usually represents the root node of a tree. In some systems, such as Visual Thesaurus [77] and Neighbourhood Explorer [71], the central node represents a frame of reference against which connected nodes may be compared. Alternately, the center of the graphic may provide a descriptive context for the remainder of the diagram. For example, although not shown in Fig. 16, the innermost circle of the Radial Traffic Analyzer [49] sometimes contains the name of the geographic region whose network traffic is currently being visualized. In many time-series spiral visualizations [12], [82], the center of the spiral denotes the starting point of the time interval under consideration. As one can see, the

meaning of the centroid is domain-specific and varies across applications and design patterns.

In the systems which adhere to the Connected Ring and Disconnected Ring patterns, the ring's interior space is typically reserved for application-specific purposes. In diagrams such as the text relation map [69] and commercial systems like Daisy [19] and NetMap [30], the interior of the ring is occupied by line segments connecting related nodes on the circumference. VisAlert [55] instead uses the area inside the ring to render a network topology map. Systems like Manager's Dartboard [21] and SQiRL [23] put no information in the ring's interior by default, instead reserving that space for user-specified queries. Strictly speaking, the Event Tunnel [74] leaves the interior of the ring completely empty, but due to the necessary projection of the 3D cylinder onto a 2D display area, in practice, the interior space of each ring is filled with the projection of the rings beneath it.

It is worth noting that although current instantiations of the Connected Ring and Disconnected Ring patterns have placed more emphasis on the interior of the ring as a whole, rather than on the exact centroid, these were implementation-specific design choices, not a requirement of the patterns. Future systems built using these patterns may well ascribe specific meanings to the ring's center point.

5.3 Level of Interactivity

Interactivity plays an important role in any information visualization method, and radial visualization is no exception. From our analysis of the systems surveyed herein, we conclude that none of the radial design patterns are intrinsically "better" for interactive applications than another. In general, each pattern can either be used as a standalone static graphic, or as a part of an interactive system.

Yang et al. [85] suggest five interaction techniques for radial visualization. Although these operations were originally proposed in the context of RSF-based systems for visualizing hierarchical data, the principles behind them extend to the other design patterns as well. The operations are: **selection**, the process of selecting certain nodes for further operations; **reconfiguration**, the process of modifying the data; **distortion**, the process of temporarily enlarging certain objects of the display while maintaining context; **drill-down/roll-up**, the process of exposing/hiding nodes in the display; and **pan, zoom, and rotation**, the process of modifying the focus, scale, and orientation of the display.

Most of the systems surveyed in Section 4 support one or more of these operations. For instance, systems that use the Tree pattern will often support a combination of distortion and drill-down/roll-up techniques to permit smooth browsing of large hierarchies. MoireGraphs [43] and RINGS [76] are typical examples. Visualizations designed for viewing search results, such as RankSpiral [72], generally permit the selection of responses for further analysis. As a typical example of the Connected Ring pattern, the VisAlert system [55] allows the user to pan and zoom the network map inside the ring's interior, as well as expand and collapse sectors along the circumference.

5.4 Access Permissions

For many radial visualization tasks, the goal is to view and comprehend the data, not necessarily modify it. Hence, read-only access to the underlying data is sufficient. Our survey of radial visualization techniques confirms this assumption. We found very few systems that support the interactive editing of data. Two exceptions are InterRing [85] and Manager's Dartboard [21]. This does not necessarily mean that other systems do not have editing features, just that we were unable to determine it from their published descriptions. Note, however, that a visualization system's lack of "write" access to the data should not be construed as an inherent weakness. It is often the case that the integrity of the visualization requires that the user *not* be able to modify the data. This is true for many types of data, for example, election results, census data, network traffic, etc. System builders must consider the problem domain and intended purpose of a proposed radial visualization system before deciding whether to incorporate editing operations into the user interface.

5.5 2D versus 3D

The debate over 2D versus 3D methods for information visualization remains unresolved [17], [75]. Nevertheless, a vast majority of radial visualization systems available today, both academic and commercial, are based on a 2D design. Those that do support 3D are chiefly implementations of novel tree layout algorithms, such as Cone Trees [67] and PhylloTrees [14]. As previously mentioned, the VIBall [88] is unique in that it is a tactile 3D system, rather than a 2D projection of a 3D image. We believe that the perceived bias toward 2D interfaces for radial visualization is due in large part to the formidable occlusion issues related to 3D graphics [26].

6 LOOKING FORWARD

The dimensions outlined in Section 5, coupled with the contents of Table 2, suggest that the design space for radial visualization has not yet been fully explored. New visualization systems can be created by applying one or more patterns to new problem domains. For example, recent work by Burch and Diehl [10] combines elements from both the Concentric RSF and Connected Ring patterns to visualize dynamic compound digraphs. Future systems of this type may justify the inclusion of a new design pattern into the taxonomy presented herein.

Thus, we do not claim that the seven design patterns presented herein encompass the entire range of possibilities for radial visualization. It is a relatively young discipline, and only in recent years, has it made significant inroads into the broader field of information visualization. By considering the design dimensions outlined in Section 5 and applying these choices to new data sets, future system builders will create additional radial visualizations which go beyond the design patterns described in this work.

The visualization of hierarchical data is an important topic in information visualization, so we expect that those design patterns which lend themselves to hierarchical data (namely, Tree and Concentric RSF) will continue to see widespread use and extension. We foresee that future radial

visualization systems will be able to visualize ever larger hierarchies. However, we do not expect the resolution of commodity display screens to grow dramatically in the near future. Rather, we anticipate that improvements in the capacity of radial visualization systems will be due to the development of new techniques for distortion and drill-down/roll-up, coupled with normal increases in computational power.

Regarding the dimension of interactivity [87], we likewise expect this to improve in many ways. More and more radial visualization will be done via collaborative tabletop interfaces [41], portable computers, and touchscreen displays. We do not expect these interaction techniques to supplant the desktop mouse/keyboard paradigm; this will remain a good general purpose platform for everyday tasks. Rather, we expect that advanced visualization applications will gradually migrate to more specialized devices.

The sheer abundance of systems utilizing some form of radial visualization would suggest that users find this approach satisfying; however, some quantitative usability studies have also been done. Barlow and Neville [3] report a study in which users were able to navigate a hierarchical data set more quickly using a radial chart than with other tree representations. In addition, Vliegen et al. [80] suggest that radially rendered TreeMaps leverage users' familiarity with pie charts, thus lowering the barriers to their acceptance. A usability test conducted by Yee et al. [86] indicates that radial animation helps users maintain context better than linear animation in the same visualization. Certain ring-based visualizations [21], [23] were also shown to be easy to learn for novice users, while still providing valuable insights into the data. These results, while compelling, are not enough. Additional quantitative evaluations of radial visualization methods are needed to shed further light on the general and specific effectiveness of these techniques. This is a valid avenue for further research in this area.

7 CONCLUSION

Although radial visualization is becoming an increasingly common metaphor in information visualization, the underlying concepts trace their origins to the well-established field of statistical graphics. In this paper, we have examined recent work in radial visualization from three perspectives. First, we look at the kinds of problems for which radial visualization has been used. Second, we propose seven design patterns for radial visualization, based on our analysis of current systems. Third, we suggest five dimensions, or design choices, that builders of radial visualizations must consider in creating new systems. We conclude by offering insights regarding how radial visualization techniques may be extended in the future.

We envisage that this survey will raise the community's awareness of radial visualization and elevate it from an interesting visual gimmick employed in an ad hoc manner by a handful of information visualization systems, to a *bona fide* methodology in its own right. While no single approach is right for every problem, researchers may wish to consider the unique properties of the radial metaphor in designing future visualizations. We hope that the analysis and taxonomy

presented herein will motivate and facilitate continued research in the creative development of this constantly evolving collection, leading to new schemes that better serve the manifold challenges in information visualization.

ACKNOWLEDGMENTS

This work was conducted while Geoffrey Draper was a PhD candidate at the University of Utah, and was supported in part by the Scientific Computing and Imaging Institute at the University of Utah, and by the US Air Force Research Laboratory, the US Air Force Material Command, and the US Department of Defense under Contract No. F30602-03-C-0257 through the Advanced Research and Activity's (ARDA) Advanced Intelligence Community Information Assurance Research Thrust. Additional support was given by the Graduate School and School of Computing at the University of Utah. The authors also thank the anonymous reviewers for their insightful suggestions which greatly improved this paper.

REFERENCES

- [1] K. Andrews and H. Heidegger, "Information Slices: Visualising and Exploring Large Hierarchies Using Cascading, Semi-Circular Discs," *Proc. InfoVis Late Breaking Hot Topic Papers*, pp. 9-12, 1998.
- [2] M. Ankerst, D.A. Keim, and H.-P. Kriegel, "Circle Segments: A Technique for Visually Exploring Large Multidimensional Data Sets," *Proc. IEEE Visualization Conf., Hot Topic Session*, 1996.
- [3] T. Barlow and P. Neville, "A Comparison of 2D Visualizations of Hierarchies," *Proc. InfoVis '01*, pp. 131-138, 2001.
- [4] M.E. Baron, "A Note on the Historical Development of Logic Diagrams: Leibniz, Euler and Venn," *The Math. Gazette: The J. Math. Assoc.*, vol. 53, no. 384, pp. 113-125, May 1969.
- [5] C.V. Berendonck and T. Jacobs, "Bubbleworld: A New Visual Information Retrieval Technique," *Proc. Asia-Pacific Symp. Information Visualisation (APVis '03)*, pp. 47-56, 2003.
- [6] J. Bertin, *Semiology of Graphics*. Univ. of Wisconsin Press, 1984.
- [7] E. Bertini, P. Hertzog, and D. Lalanne, "SpiralView: Towards Security Policies Assessment through Visual Correlation of Network Resources with Evolution of Alarms," *Proc. IEEE Symp. Visual Analytics Science and Technology (VAST '07)*, vol. 2, pp. 139-146, Oct. 2007.
- [8] U. Brandes, P. Kenis, and D. Wagner, "Communicating Centrality in Policy Network Drawings," *IEEE Trans. Visualization and Computer Graphics*, vol. 9, no. 2, pp. 241-253, Apr.-June 2003.
- [9] I. Brewer, A.M. MacEachren, H. Abdo, J. Gundrum, and G. Otto, "Collaborative Geographic Visualization: Enabling Shared Understanding of Environmental Processes," *Proc. InfoVis '00*, pp. 137-141, 2000.
- [10] M. Burch and S. Diehl, "Timeradartrees: Visualizing Dynamic Compound Digraphs," *Proc. 10th Joint Eurographics/IEEE-VGTC Symp. Visualization (EuroVis)*, 2008.
- [11] J. Callahan, D. Hopkins, M. Weiser, and B. Shneiderman, "An Empirical Comparison of Pie vs. Linear Menus," *Proc. Special Interest Group on Computer-Human Interaction Conf. Human Factors in Computing Systems*, pp. 95-100, 1988.
- [12] J.V. Carlis and J.A. Konstan, "Interactive Visualization of Serial Periodic Data," *Proc. 11th Ann. ACM Symp. User Interface Software and Technology (UIST '98)*, pp. 29-38, 1998.
- [13] M.-J. Carpano, "Automatic Display of Hierarchized Graphs for Computer-Aided Decision Analysis," *IEEE Trans. Systems, Man, and Cybernetics*, vol. SMC-10, no. 11, pp. 705-715, Nov. 1980.
- [14] S. Carpendale and A. Agarwala, "Phyllotrees: Harnessing Nature's Phyllotactic Patterns for Tree Layout," *Proc. InfoVis '04*, p. p3, 2004.
- [15] R.A. Cava, P.R.G. Luzzardi, and C.M.D.S. Freitas, "The Bifocal Tree: A Technique for the Visualization of Hierarchical Information Structures," *Proc. Workshop Human Factors in Computer Systems (IHC)*, 2002.

- [16] M.C. Chuah, "Dynamic Aggregation with Circular Visual Designs," *Proc. InfoVis '98*, pp. 35-43, 151, 1998.
- [17] A. Cockburn, "Revisiting 2D vs 3D Implications on Spatial Memory," *Proc. Fifth Conf. Australasian User interface*, pp. 25-31, 2004.
- [18] J. Cugini, C. Piatko, and S. Laskowski, "Interactive 3D Visualization for Document Retrieval," *Proc. Workshop New Paradigms in Information Visualization and Manipulation, ACM Conf. Information and Knowledge Management (CIKM)*, 1996.
- [19] Daisy Analysis Ltd, *Daisy 2003*, <http://www.daisy.co.uk/>, Nov. 2007.
- [20] P. Dragicevic and S. Huot, "SpiraClock: A Continuous and Non-Intrusive Display for Upcoming Events," *Proc. Extended Abstracts on Human Factors in Computing Systems (CHI '02)*, pp. 604-605, 2002.
- [21] G. Draper, Y. Livnat, and R. Riesenfeld, "A Visual Query Language for Correlation Discovery and Management," *Proc. Second Ann. Visual and Iconic Language Conf. (Vall '08)*, pp. 14-23, 2008.
- [22] G.M. Draper and R.F. Riesenfeld, "Interactive Fan Charts: A Space-Saving Technique for Genealogical Graph Exploration," *Proc. Eighth Ann. Workshop Technology for Family History and Genealogical Research*, 2008.
- [23] G.M. Draper and R.F. Riesenfeld, "Who Votes for What? A Visual Query Language for Opinion Data," *IEEE Trans. Visualization and Computer Graphics*, vol. 14, no. 6, pp. 1197-1204, Nov./Dec. 2008.
- [24] P. Eades, "Drawing Free Trees," *Bull. of the Inst. of Combinatorics and its Applications*, pp. 10-36, 1992.
- [25] N. Elmqvist, J. Stasko, and P. Tsigas, "DataMeadow: A Visual Canvas for Analysis of Large-Scale Multivariate Data," *Proc. IEEE Symp. Visual Analytics Science and Technology (VAST '07)*, vol. 2, pp. 187-194, Oct. 2007.
- [26] N. Elmqvist and P. Tsigas, "A Taxonomy of 3D Occlusion Management for Visualization," *IEEE Trans. Visualization and Computer Graphics*, vol. 14, no. 5, pp. 1095-1109, Sept./Oct. 2008.
- [27] P.M. Fitts, "The Information Capacity of the Human Motor System in Controlling the Amplitude of Movement," *J. Experimental Psychology*, pp. 381-391, 1954.
- [28] G. Fitzmaurice, A. Khan, R. Pieké, B. Buxton, and G. Kurtenbach, "Tracking Menus," *Proc. 16th Ann. ACM Symp. User Interface Software and Technology (UIST '03)*, pp. 71-79, 2003.
- [29] S. Foresti, J. Agutter, Y. Livnat, S. Moon, and R. Erbacher, "Visual Correlation of Network Alerts," *IEEE Computer Graphics and Applications*, vol. 26, no. 2, pp. 48-59, Mar./Apr. 2006.
- [30] J. Galloway and S.J. Simoff, "Network Data Mining: Methods and Techniques for Discovering Deep Linkage between Attributes," *Proc. Third Asia-Pacific Conf. Conceptual Modelling (APCCM '06)*, pp. 21-32, 2006.
- [31] M. Gross, T. Sprenger, and J. Finger, "Visualizing Information on a Sphere," *Proc. InfoVis '97*, pp. 11-16, 1997.
- [32] B. Grünbaum, "Mathematical Challenges in Escher's Geometry," *M.C. Escher: Art and Science*, H.S.M. Coxeter, M. Emmer, R. Penrose, and M.L. Teuber, eds., North Holland, pp. 53-67, 1985.
- [33] S. Havre, E. Hetzler, K. Perrine, E. Jurrus, and N. Miller, "Interactive Visualization of Multiple Query Results," *Proc. InfoVis '01*, pp. 105-112, 2001.
- [34] B. Hetzler, P. Whitney, L. Martucci, and J. Thomas, "Multi-Faceted Insight through Interoperable Visual Information Analysis Paradigms," *Proc. InfoVis '98*, pp. 137-144, 161, 1998.
- [35] P. Hoffman, G. Grinstein, K. Marx, I. Grosse, and E. Stanley, "DNA Visual and Analytic Data Mining," *Proc. Eighth Conf. Visualization*, pp. 437-ff., 1997.
- [36] D. Holten, "Hierarchical Edge Bundles: Visualization of Adjacency Relations in Hierarchical Data," *Proc. InfoVis '06*, pp. 741-748, 2006.
- [37] J.Y. Hong, J. DAndries, M. Richman, and M. Westfall, "Zoomology: Comparing Two Large Hierarchical Trees," *Proc. Poster Compendium of IEEE Information Visualization*, 2003.
- [38] M. Howell, *Filelight*, <http://www.methylblue.com/filelight/>, Aug. 2007.
- [39] S. Huot and E. Lecolinet, "SpiraList: A Compact Visualization Technique for One-Handed Interaction with Large Lists on Mobile Devices," *Proc. Fourth Nordic Conf. Human-Computer Interaction (NordICHI '06)*, pp. 445-448, 2006.
- [40] A. Inselberg and B. Dimsdale, "Parallel Coordinates: A Tool for Visualizing Multi-Dimensional Geometry," *Proc. IEEE Visualization Conf.*, pp. 361-378, 1990.
- [41] P. Isenberg and S. Carpendale, "Interactive Tree Comparison for Co-Located Collaborative Information Visualization," *IEEE Trans. Visualization and Computer Graphics*, vol. 13, no. 6, pp. 1232-1239, Nov./Dec. 2007.
- [42] M. Ishihara, K. Misue, and J. Tanaka, "Ripple Presentation for Tree Structures with Historical Information," *Proc. Asia Pacific Symp. Information Visualization (APVIS '06)*, pp. 153-160, 2006.
- [43] T. Jankun-Kelly and K.-L. Ma, "Moiregraphs: Radial Focus+Context Visualization and Interaction for Graphs with Visual Node," *Proc. InfoVis '03*, pp. 59-66, 2003.
- [44] C.-S. Jeong and A. Pang, "Reconfigurable Disc Trees for Visualizing Large Hierarchical Information Space," *Proc. InfoVis '98*, pp. 19-25, 1998.
- [45] B. Johnson and B. Shneiderman, "Tree-Maps: A Space-Filling Approach to the Visualization of Hierarchical Information Structures," *Proc. Second Conf. Visualization*, pp. 284-291, 1991.
- [46] B.S. Johnson, "Treemaps: Visualizing Hierarchical and Categorical Data," PhD thesis, Univ. of Maryland, 1993.
- [47] E. Kandogan, "Visualizing Multi-Dimensional Clusters, Trends, and Outliers Using Star Coordinates," *Proc. Seventh ACM SIGKDD Int'l Conf. Knowledge Discovery and Data Mining*, pp. 107-116, 2001.
- [48] D.A. Keim, "Information Visualization and Visual Data Mining," *IEEE Trans. Visualization and Computer Graphics*, vol. 8, no. 1, pp. 1-8, Jan.-Mar. 2002.
- [49] D.A. Keim, F. Mansmann, J. Schneidewind, and T. Schreck, "Monitoring Network Traffic with Radial Traffic Analyzer," *Proc. IEEE Symp. Visual Analytics Science and Technology (VAST '06)*, pp. 123-128, 2006.
- [50] D.A. Keim, J. Schneidewind, and M. Sips, "CircleView: A New Approach for Visualizing Time-Related Multidimensional Data Sets," *Proc. Working Conf. Advanced Visual Interfaces (AVI '04)*, pp. 179-182, 2004.
- [51] G. Kurtenbach and W. Buxton, "The Limits of Expert Performance Using Hierarchic Marking Menus," *Proc. Computer-Human Interaction (CHI '93)*, pp. 482-487, 1993.
- [52] J. Lamping and R. Rao, "Laying Out and Visualizing Large Trees using a Hyperbolic Space," *Proc. Seventh Ann. ACM Symp. User Interface Software and Technology (UIST '94)*, pp. 13-14, 1994.
- [53] J. Lamping and R. Rao, "The Hyperbolic Browser: A Focus+Context Technique Based on Hyperbolic Geometry for Visualizing Large Hierarchies," *J. Visual Languages and Computing*, vol. 7, pp. 33-55, 1996.
- [54] J. Lamping, R. Rao, and P. Piroli, "A Focus+Context Technique Based on Hyperbolic Geometry for Visualizing Large Hierarchies," *Proc. Special Interest Group on Computer-Human Interaction (SIGCHI) Conf. One Human Factors in Computing Systems*, pp. 401-408, 1995.
- [55] Y. Livnat, J. Agutter, S. Moon, R.F. Erbacher, and S. Foresti, "A Visualization Paradigm for Network Intrusion Detection," *Proc. IEEE Workshop Information Assurance and Security*, pp. 30-37, 2005.
- [56] Y. Livnat, J. Agutter, S. Moon, and S. Foresti, "Visual Correlation for Situational Awareness," *Proc. InfoVis '05*, pp. 95-102, 2005.
- [57] J.D. Mackinlay, G.G. Robertson, and R. DeLine, "Developing Calendar Visualizers for The Information Visualizer," *Proc. Seventh Ann. ACM Symp. User Interface Software and Technology (UIST '94)*, pp. 109-118, 1994.
- [58] Merriam-Webster, Inc., Merriam-Webster Online Dictionary, <http://www.m-w.com/>, Aug. 2007.
- [59] M.J. Mohammadi-Aragh and T. Jankun-Kelly, "Moiretrees: Visualization and Interaction for Multi-Hierarchical Data," *Proc. EuroGraphics—IEEE VGTC Symp. Visualization*, K.W. Brodli, D.J. Duke, and K.I. Joy, eds., pp. 59-66, 2005.
- [60] T. Munzner, "H3: Laying Out Large Directed Graphs in 3D Hyperbolic Space," *Proc. InfoVis '97*, pp. 2-10, 1997.
- [61] F. Nightingale, *Notes on Matters Affecting the Health, Efficiency, and Hospital Administration of the British Army: Founded Chiefly on the Experience of the Late War*. Harrison and Sons, 1858.
- [62] M.L. Northway, *A Primer of Sociometry*, second ed. Univ. of Toronto Press, 1967.
- [63] W. Playfair, *The Statistical Breviary; Shewing, on a Principle Entirely New, the Resources of Every State and Kingdom in Europe*. Wallis, 1801.

- [64] Z. Pousman and J. Stasko, "A Taxonomy of Ambient Information Systems: Four Patterns of Design," *Proc. Working Conf. Advanced Visual Interfaces (AVI '06)*, pp. 67-74, 2006.
- [65] H. Purchase, "Which Aesthetic Has the Greatest Effect on Human Understanding?" *Proc. Fifth Int'l Symp. Graph Drawing (GD '97)*, pp. 248-261, 1997.
- [66] M.G. Reggiani and F.E. Marchetti, "A Proposed Method for Representing Hierarchies," *IEEE Trans. Systems, Man, and Cybernetics*, vol. 18, no. 1, pp. 2-8, Jan. 1989.
- [67] G.G. Robertson, J.D. Mackinlay, and S.K. Card, "Cone Trees: Animated 3D Visualizations of Hierarchical Information," *Proc. SIGCHI Conf. Human Factors in Computing Systems*, pp. 189-194, 1991.
- [68] D. Rubine, "Specifying Gestures by Example," *Proc. ACM SIGGRAPH '91*, pp. 329-337, 1991.
- [69] G. Salton, J. Allan, C. Buckley, and A. Singhal, "Automatic Analysis, Theme Generation, and Summarization of Machine-Readable Texts," *Science*, vol. 264, no. 5164, pp. 1421-1426, 1994.
- [70] I. Spence, "No Humble Pie: The Origins and Usage of a Statistical Chart," *J. Educational and Behavioral Statistics*, vol. 30, no. 4, pp. 353-368, 2005.
- [71] R. Spence, *Information Visualization*, first ed., pp. 85-88. ACM Press, 2001.
- [72] A. Spoerri, "RankSpiral: Toward Enhancing Search Results Visualizations," *Proc. InfoVis*, 2004.
- [73] J. Stasko and E. Zhang, "Focus+Context Display and Navigation Techniques for Enhancing Radial, Space-Filling Hierarchy Visualizations," *Proc. InfoVis '00*, pp. 57-65, 2000.
- [74] M. Suntinger, H. Obwegger, J. Schiefer, and M.E. Gröller, "The Event Tunnel: Interactive Visualization of Complex Event Streams for Business Process Pattern Analysis," *Proc. IEEE Pacific Visualization Symp. (PacificVIS '08)*, pp. 111-118, Mar. 2008.
- [75] M. Tavanti and M. Lind, "2D vs. 3D, Implications on Spatial Memory," *Proc. InfoVis '01*, pp. 139-145, 2001.
- [76] S.T. Teoh and K.-L. Ma, "RINGS: A Technique for Visualizing Large Hierarchies," *Proc. Int'l Symp. Graph Drawing '02*, pp. 268-275, 2002.
- [77] Thinkmap, Inc., *Visual Thesaurus*, <http://www.visualthesaurus.com/>, Aug. 2007.
- [78] C. Tominski, J. Abello, and H. Schumann, "Axes-Based Visualizations with Radial Layouts," *Proc. ACM Symp. Applied Computing (SAC '04)*, pp. 1242-1247, 2004.
- [79] R.S. Torres, C.G. Silva, C.B. Medeiros, and H.V. Rocha, "Visual Structures for Image Browsing," *Proc. 12th Int'l Conf. Information and Knowledge Management (CIKM '03)*, pp. 49-55, 2003.
- [80] R. Vliegen, J.J. van Wijk, and E.-J. van der Linden, "Visualizing Business Data with Generalized Treemaps," *Proc. InfoVis '06*, pp. 789-796, 2006.
- [81] H. Vogel, "A Better Way to Construct the Sunflower Head," *Math. Biosciences*, vol. 44, pp. 179-189, 1979.
- [82] M. Weber, M. Alexa, and W. Müller, "Visualizing Time-Series on Spirals," *Proc. InfoVis '01*, pp. 7-13, 2001.
- [83] L. Wilkinson, *The Grammar of Graphics (Statistics and Computing)*, first ed. Springer, 1999.
- [84] Y. Wu and M. Takatsuka, "Visualizing Multivariate Network on the Surface of a Sphere," *Proc. Asia-Pacific Symp. Information Visualization (APVIS)*, 2006.
- [85] J. Yang, M.O. Ward, and E.A. Rundensteiner, "InterRing: An Interactive Tool for Visually Navigating and Manipulating Hierarchical Structures," *Proc. InfoVis '02*, pp. 77-84, 2002.
- [86] K.-P. Yee, D. Fisher, R. Dhamija, and M. Hearst, "Animated Exploration of Dynamic Graphs with Radial Layout," *Proc. InfoVis '01*, pp. 43-50, 2001.
- [87] J.S. Yi, Y. ah Kang, J.T. Stasko, and J.A. Jacko, "Toward a Deeper Understanding of the Role of Interaction in Information Visualization," *IEEE Trans. Visualization and Computer Graphics*, vol. 13, no. 6, pp. 1224-1231, Nov. 2007.
- [88] B. Yip, S. Goyette, and C. Madden, "Visualising Internet Traffic Data with Three-Dimensional Spherical Display," *Proc. 2005 Asia-Pacific Symp. Information Visualization (APVIS '05)*, pp. 153-158, 2005.



Geoffrey M. Draper received the PhD degree in computer science at the University of Utah. His research interests include information visualization, human-computer interaction, graphics, and vintage computing. He recently accepted a faculty position in computer science at Brigham Young University-Hawaii.



Yarden Livnat received the PhD degree in computer science from the University of Utah. He is a research scientist at the Scientific Computing and Imaging Institute, University of Utah. His research interests include visual analytics with emphasis on situational awareness, scientific visualization, and software common components architecture. He is a member of the IEEE.



Richard F. Riesenfeld received the PhD degree from the School of Computer and Information Science, Syracuse University in 1973. He is a professor of computer science at the University of Utah, where he joined the faculty in 1972. His research specialties are computer-aided geometric design and computer graphics.

► For more information on this or any other computing topic, please visit our Digital Library at www.computer.org/publications/dlib.