Combining Timeline and Graph Visualization

Robert Morawa
Interactive Media Lab
Technische Universität Dresden
Dresden, Germany
robert.morawa@tu-dresden.de

Annett Mitschick
Interactive Media Lab
Technische Universität Dresden
Dresden, Germany
annett.mitschick@tu-dresden.de

Tom Horak
Interactive Media Lab
Technische Universität Dresden
Dresden, Germany
tom.horak@tu-dresden.de

Raimund Dachselt
Interactive Media Lab
Technische Universität Dresden
Dresden, Germany
dachselt@acm.org

Ulrike Kister
Interactive Media Lab
Technische Universität Dresden
Dresden, Germany
ukister@acm.org

ITS 2014, November 16–19, 2014, Dresden, Germany.
ACM 978-1-4503-2587-5/14/11.
http://dx.doi.org/10.1145/2669485.2669544

Abstract
Timelines are as important for presenting temporal data as node-link diagrams are relevant for displaying graphs and relations in general. Yet, both are rarely combined. We present *Time Shadows* to precisely indicate a node's place in time, revealing associated temporal data and relations. We also introduce *Time Beads*. Created as a focus and context interaction technique for time-based graphs, Time Beads allow to continuously or discretely change the level of detail and to set multiple arbitrary focus regions if needed. We implemented both techniques in a prototype and conducted an initial user study.

Author Keywords
Time-visualization; Information Visualization; Semantic Data.

ACM Classification Keywords
H.5.2. [Information Interfaces and Presentation (e.g. HCI)]: User Interfaces.

Introduction
Understanding temporal data and phenomena is key to long-term planning, as well as decision- and sensemaking. This applies to stock prices, chemical reactions, historical events and many other domains.

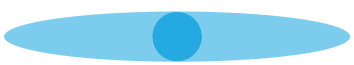

Figure 1: *Time Shadow*: The transparent layer around the node indicates its duration.

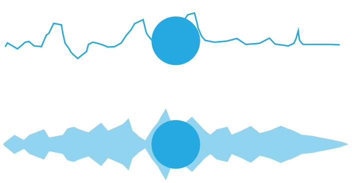

Figure 2: *Time Shadow* with sparklines or polygons can display quantitative data.

There are two main goals for analyzing temporal data; to get a sense of the past development and to detect trends and patterns for the future. This calls for dedicated interactive features like the ability to browse, to compare different regions in time and to choose hierarchical layers such as decades, years or months [1]. To achieve this, dedicated temporal visualizations tend to provide the user with overview and detail views [2], focus and context tools like vertical zoom lenses [4], or multiple timeline views [6].

Temporal data may contain multiple types of relations. The relation in time is the most important and relevant, often mapped to the spatial position as dominant visual variable [3], the elements displayed as bars or dots. Commonly, visualizations show similarity via facets [7] or color-coding [5]. More complex relations require different means. The tool *Continuum* uses mainly frames, indented lists and bars to visualize hierarchies and quantitative data [2], while Jensen uses edges and different kinds of arrows to map different relations [6].

However, complex networks like semantic data tend to be visualized as node-link diagrams. Already the creation of a meaningful, readable graph layout proves to be very challenging. A spatial timeline imposes a strict layouting constraint. This causes problems for drawing nodes that occupy a time interval, and for drawing edges of temporal distant nodes, especially when combined with zoomable interfaces.

In this paper, we propose two techniques to deal with time-based node-link diagrams. We explain the design process, including an initial study, and the concepts of *Time Shadow* and *Time Beads*. We present the prototype used in our user study, Nornir, and finally discuss the results of our work.

Design Process

The following concepts were developed using an iterative design process. The visualization of time-dependent node-link diagrams was targeted toward large multi-touch displays. Our first concepts were implemented in a proof-of-concept prototype, which featured semantic data from the DBpedia project[1].

The prototype was used for an initial study with seven participants (four male, average age: 24). We gave each user seven tasks, e.g., to read off the life span and identify relations of Albert Einstein. The interactions were video-recorded, and we took notes of the behavior and comments of the users. Each session finished with a short questionnaire.

We used the feedback to further revise the concepts and identify their strengths and weaknesses. The following sections feature the refined concepts.

Time Shadow

When displaying nodes in a spatial mapping of time, their shape has to be considered. Simple nodes can lead to confusion, especially if they represent an item with a longer duration; is the node placed at the start, the middle or the end of the interval? However, bars would cause problems when drawing edges as they occupy much more space.

Our proposal to solve this problem, the Time Shadow, is a semi-transparent layer behind an arbitrary shaped node. Using its length and spatial position, it visualizes the duration of an item as shown in Figure 1, for example the time a chemical reaction requires to finish or the lifetime of a person. Time spans may exist that are still ongoing,

[1] http://dbpedia.org

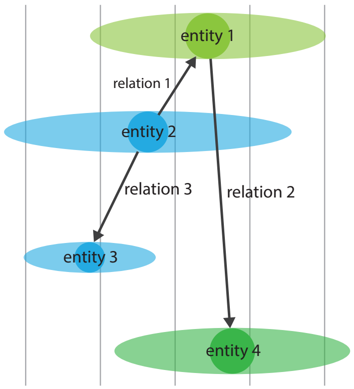

Figure 3: Time Shadows can be used easily to extend a node-link diagram.

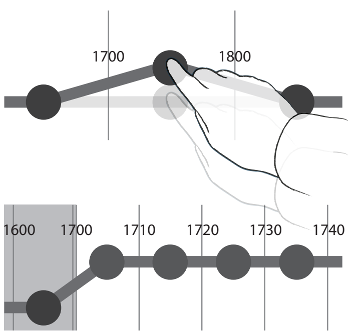

Figure 4: A Time Bead can be popped by tapping or dragging it upwards.

e.g., if the person is still alive. In this case, Time Shadows can be blurred out instead of ending on a specific point.

Time Shadow can have various shapes which can be used to visualize different classes of durations. Most remarkable, sparklines or irregular polygons can be used to display associated quantitative data, as visualized in Figure 2, their amplitudes indicating published works of an artist or the stock prices of a company.

For our study, we used ellipse shaped Time Shadows. Users required some time to identify the Time Shadow as the duration of an item, but then heavily relied on it while reading the graph. Often they were used to better grasp semantic relations that featured a time-based constraint. For example, someone whose Time Shadow started after Albert Einsteins ended, could not have influenced him in any way. Also, the task to read off the date of birth and death of Albert Einstein was the best rated of our study.

Time Shadows offer many advantages for layouting and sensemaking. They exactly define start and end of a time interval, something that is not possible with common circular nodes. This supports the understanding of temporal relations in general. Edges may cross the shadow without being associated with the node, as shown in Figure 3, in contrast to bar-shaped nodes. Nodes can have different shapes, and even vary their position within their Time Shadow, if it benefits the overall layout without changing the meaning. The Time Shadow's size makes their associated nodes easier to touch and harder to occlude either with a finger or other nodes.

However, the increased size can also lead to a cluttered visualization, when too many shadows overlap and reduces the amount of nodes that can be drawn on the canvas. This increases the importance of a meaningful and readable layout. Surveying the results of our initial study, Time Shadows are a promising approach to visualize the time aspect of entities.

Time Beads

Zoom-based overview and detail interaction is common within time visualization, but faces problems when displaying node-link diagrams. A node may have connections to nodes that are well outside the zooming level, since the time variable heavily constraints the layout. Edges would link to the edge of the screen or to an off-screen visualization, which omits the temporal context the linked nodes are placed in.

Time Beads are a visualization a focus and context interaction technique, and are inspired by TimeZoom [4]. Time Beads utilizes graph nodes to represent time intervals, potentially ranging from millenia to milliseconds. They chained at the lower edge of the screen as a timeline, the boudaries separated by thin lines. Time is hereby understood as a tree, where a millenium has its centuries as children, each century its decades and so on.

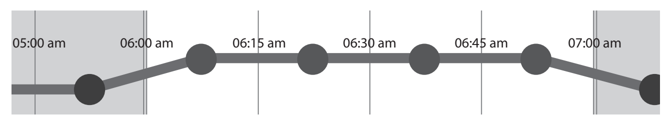

Figure 5: *Time Beads* display time intervals of different hierarchical layers.

Interaction is based on this underlying structure. With a simple tap or drag to the top, a Bead can be *popped*, disappearing and revealing its children, as visible in Figure 4. Since all Time Beads have the same default weight and distribute equally onscreen, the user stretches a focus region while the other regions compress.

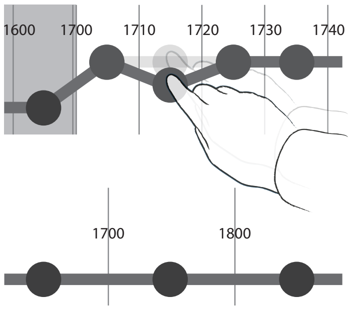

Figure 6: When pulling down a Bead, its hierarchical layer collapses.

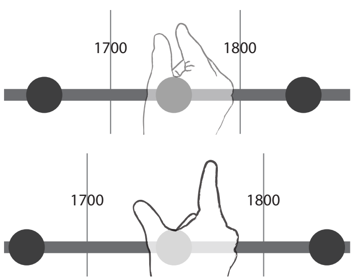

Figure 8: A spread gesture on an interval expands it.

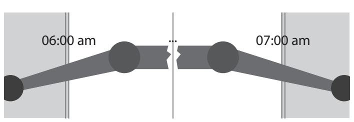

Figure 9: A 2.5D projection emphasizes the transition between layers.

If the user chooses to expand one of the children further, this Bead pops and its right and left siblings *stack*. A stack represents the time interval of all stacked Beads and can be popped to distribute them again. Beads can be stacked manually by dragging one Bead over another, as shown in Figure 7. Every Bead between them will be added to the stack. Beads can also be stacked at the edge of the screen by dragging them there, which essentially stretches the entire remaining timeline.

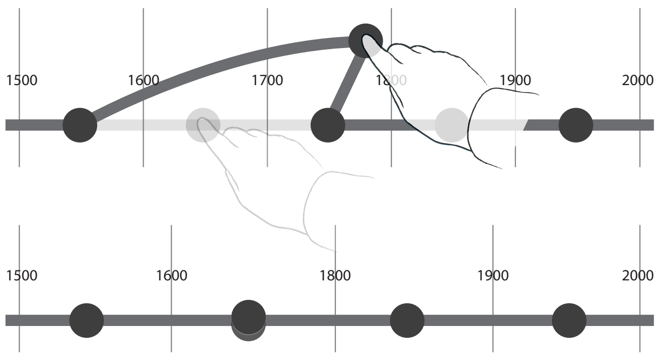

Figure 7: Multiple Time Beads can be stacked manually by dragging one Bead over another.

The user can pull down a Bead to *collapse* it, all its siblings and their children. They will be replaced by the single parent node, as visualized in Figure 6. These three techniques allow a discrete navigation along the timeline. Since Time Beads are only loosely chained, the user can open multiple focus regions to compare data at different points in time without additional tools like fisheye lenses.

To clearly distinguish different hierarchical layers, we use different visualization techniques. Beads and their interval receive a greyscale coloration, where parents are colored darker than children. Also parent Beads are closer to the bottom of the screen than their children.

While the mapping proved to be beneficial in our initial study, the interaction style was not deemed intuitive. In fact, most users at first tried to pinch-to-zoom or tap on the background of the node-link diagram. When users were browsing the graph, they generally did not interact with the Time Beads, instead they primarily used them to solve time-related tasks. Also, some users did not recognize the transition between the heterogeneous layers.

To improve Time Beads and the overall interaction, we concluded to further regard multi-touch interactions on the whitespace of the graph, but still the goal was to keep the gesture set as simple as possible. The basic concepts for discrete navigation are maintained and the basic gestures for popping, collapsing and stacking also work on the whitespace, targeting the underlying Time Bead.

The user can use the spread to zoom gesture to increase the weight of the Time Bead, as shown in Figure 8. After exceeding a certain threshold, the Bead pops, distributing its weight evenly among its children. In return, pinching decreases the weight of a Bead. The same gesture can be used bimanually. The user selects the boundaries of a time interval and scales it according to her needs, stretching or compressing the Time Beads. This enables the user to precisely focus and scale a region of interest. A double tap with the whole hand or five fingers evenly distributes weight of the Beads again, which can be necessary after several spreading gestures.

To emphasize the transition between hierarchical layers, we propose two techniques. First, we double the boundary line between these transitions. Second, we map the whole visualization to a 2.5 D projection to put an emphasis on

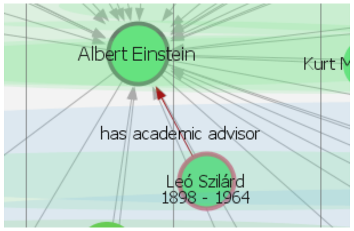

Figure 10: The prototype shows the relations of one pivotal node, here Albert Einstein.

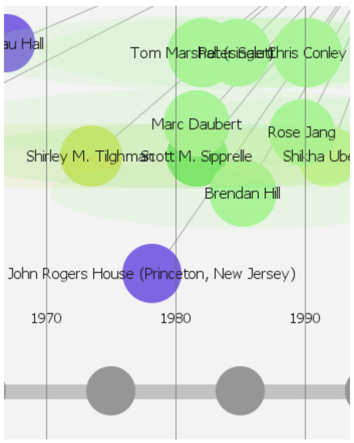

Figure 11: A snapshot of DBpedia was used, in which the user were able to explore.

the distortion effect, indicated in Figure 9. Smaller time intervals are displayed closer to the user and greater intervals, including stacked Beads, farther away. A smooth transition between the layers causes the Time Shadows, nodes and edges to skew. Since most focus regions are evenly distributed, the distortion should not dampen the readability.

Time Beads offer many advantages. The technique allows to navigate in a discrete as well as in a continuous fashion. Time Beads enable the user to directly compare multiple time intervals without additional gestures or tools. This serves the two main goals for analyzing time-related data, to compare and to find patterns. All nodes and edges are visible onscreen, reducing the need to browse.

Still, this inherently limits the amount of data shown simultaneously on screen and may lead to a cluttered visualization. Additionally, multiple sequential interactions can lead to a very heterogeneous timeline, which is harder to read than a linear one because the user has to be aware of the transitions between focus levels. Visual feedback as suggested above can counteract this effect.

Prototype: Nornir

We implemented this concept in a prototype called Nornir, again using semantic data of the DBpedia project. We used the Java-based prefuse[2] framework to display the visualization and the Apache Jena framework to query against a local Fuseki server, which held an image of the semantic graph. Mostly, Nornir ran on a Samsung SUR40 tabletop. In a later exhibition, we used a 27" Perceptive Pixel display, for its resolution and precise touch recognition. Nornir received favorable feedback from visitors, some of them addressing specifically its educational potential.

Nornir displays only the network of a single, pivotal node at a time, the starting point in our implementation being Albert Einstein. The user can pop a node to view its network and navigate through the graph. The layout algorithm positions the pivotal node in the vertical center, sorts linked nodes vertically according to their semantic relation and places them with minimal overlap. Node size depends on the total number of nodes and the node's outgoing connections. Other methods like a history, filters, selecting items and class-dependent color coding are basic visualization techniques that were also included.

Discussion

Combining timeline and node-link visualizations creates exiting opportunities. Node-link diagrams are a well-researched mean for visualizing complex explicit relations. An interactive timeline serves as a strict data-based constraint and emphasizes the implicit temporal relations. This provides an additional context for understanding the data, and can help to understand a relation without actually reading edge labels, since it may feature inherent temporal constraints. For example, persons may only influence others who lived after their date of birth.

Especially the Time Beads concept offers room for extension. Since all Beads are nodes, they can visualize more information. The size of a Bead could encode the amount of nodes the interval contains, or the strength of directed edges could indicate the amount of connections between the intervals like in PivotGraph [8].

One participant of the initial study inspired the idea to safe nodes by dragging them below the Time Beads,

[2] http://prefuse.org/

dividing the visualization in a space for browsing and an analytic space where all nodes of interest can be stored. The positioning of Time Beads is very flexible and can be used to separate space in regions with different meanings.

Conclusion

There are few approaches for combining time-based visualizations with node-link diagrams. We introduce *Time Shadow* and *Time Beads* as two techniques supporting visualization of time-based node-link diagrams. Using semantic data provided by the DBpedia project, both concepts were tested in an initial study.

It confirmed that Time Shadows provide a temporal context for graph nodes which is used to better understand time-based and semantic relations. They are simple, do not interfere with edge-drawing and can be used for displaying associated quantitative data.

Time Beads offer a solution to problems that zoom-based interaction techniques face when displaying node-link diagrams that are mapped to a timeline. They allow to stretch the timeline in discrete steps or to continuously enlarge a region of interest. Additionally, the user can magnify multiple regions.

For future work, a larger user study should be designed, to clearly evaluate the trade-offs of the Time Shadow an Time Beads techniques. While the pivotal navigation proves to be sufficient for browsing data, other means to interact with the node-link diagram should be considered that support more analytic approaches. Furthermore, it should be investigated, how both concepts fare on other screen sizes such as interactive walls or smartphones.

Acknowledgements

This work was partially funded by the German Research Foundation (DFG) through the project GEMS (DA 1319/3-1).

References

[1] Aigner, W., Miksch, S., Müller, W., Schumann, H., and Tominski, C. Visualizing time-oriented data — a systematic view. *Computers & Graphics 31*, 3 (2007), 401–409.

[2] André, P., Wilson, M. L., Russell, A., Smith, D. A., Owens, A., and Schraefel, M. Continuum: designing timelines for hierarchies, relationships and scale. In *Proc. UIST*, ACM (2007), 101–110.

[3] Carpendale, M. Considering visual variables as a basis for information visualisation. *Computer Science TR #2001-693-16 16* (2003).

[4] Dachselt, R., and Weiland, M. TimeZoom: a flexible detail and context timeline. In *CHI EA*, ACM (2006), 682–687.

[5] Dörk, M., Carpendale, S., and Williamson, C. Visualizing explicit and implicit relations of complex information spaces. *Information Visualization 11*, 1 (Jan. 2012), 5–21.

[6] Jensen, M. Visualizing complex semantic timelines. *NewsBlip Technical Report NBTR2003-001, http://newsblip.com* (2003).

[7] Plaisant, C., Mushlin, R., Snyder, A., Li, J., Heller, D., and Shneiderman, B. LifeLines: using visualization to enhance navigation and analysis of patient records. In *Proc. AMIA Symposium*, American Medical Informatics Association (1998), 76.

[8] Wattenberg, M. Visual exploration of multivariate graphs. In *Proc. CHI*, ACM (2006), 811–819.