
Christian Niles, Natalie Jeremijenko
Center for Advanced Technology
New York University
can207@nyu.edu, nat@cat.nyu.edu

Abstract

This paper details the design of a one-dimensional interface element, the “Collated Path”, an effort to create a very light-weight information manipulation tool that could be used within a user’s own dataset (e.g., their history list), which could then be used in combination with other searches. It is optimized to present views of the Semantic Web useful for everyday applications. Emphasis was placed on creating a tool with simple elements that could be combined to create sophisticated views. Whereas existing web information tools such as browser history, and search tools like Sherlock present a limited search/retrieve view, the Collated Path is designed to create a view that encourages the interaction and tangibility of such data sources enabling effective integration into the user’s workspace. The model uses the metaphor of “pages”, representing web resources, which are “collated” in two variables according to some retrieved or generated values, such as time accessed, popularity, or relevance to search terms. In addition, the design is general enough to accept values from an RDF model, the W3C’s metadata standard for the web. The conceptual framework is not restricted to web pages, but the “page” metaphor connects the concept with previous research that have promoted document piles [2], and other visualization methods, while giving it a more grounded basis for discussion.

1. Introduction

Unlike traditional information centers like a local library, the web as we know it today lacks any kind of system to organize or describe what information it contains. Search engines, price comparison sites, and other web tools must visit every possible page and guess their meaning and relationships to other pages. In addition, current tools fail to give users any usable interfaces for sense making and overview about the pages we visit. Web browsers, for example, display a single page at a time, replacing the current page with the next without providing quick access to previous pages or other contextual information, beyond “back” and “forward”. However, it is extremely common to look at pages collectively during a task, as when we perform a web search and visit and revisit pages as we encounter new information. As the standards body for the web, the W3C has addressed the organizational problems of the web in the form of the “Semantic Web”, a vision for a web in which information has meaning for machines as well as humans, creating a more coherent and descriptive information environment. The Collated Path addresses the need to create better interfaces to exploit such information and create an interactive representation of the context of web resources and our usage of them. An interface to
the Semantic Web must cross the machine/human boundary and allow the user to learn from the interface to the data as well as the data itself.

The fundamental concept of the Semantic Web is to make the information available on the web more meaningful, by making it accessible to automated tools that can augment our experience. We have already become dependent on search engines, price-comparison tools, and other filters to help us make sense of the vast collection of information on the web. If web information were more readable for these tools, the usefulness of the web could be increased. In addition, our tools can allow us to make sense of previously inaccessible data, like a thousands long list of previously viewed web pages. Though the Semantic Web is often described in examples with large or global data sets like search engines and intelligent agents, our research has focused on allowing interaction within a more directed data set such as a user’s usage history or current task. Our design addresses the integration goal of the Semantic Web by allowing such information to be integrated into common applications and reused across tasks within an application. For example, the Collated Path can be used to integrate recent documents and events into the periphery of the user’s workspace, facilitating reuse and awareness.

2. Design Goals

Current tools, as stated before, are inadequate to enable interaction and manipulation of resources in a Semantic Web environment. Current implementations fail to address the goals we feel are imperative to allow dynamic user-directed views of the Semantic Web:

- Establish a context for the user’s current position within their task or usage history.
- Encourage the manipulation and integration of the data set.
- Allow for casual organization of their data using methods such as ‘piling’ and informal overview.
- Present a cognitively lightweight interface,
- Allow a degree of ambiguity that exists in knowledge systems and user’s work habits.

These goals are focused towards amplifying the use of web resources because meaning (semantics) is often a product of use. By failing to allow the use of such data, current tools inhibit the sense making processes that allow for the construction of meaning. Research into users’ revisitation patterns has shown that pages often do have a shared, collective meaning, because users revisit pages at a rate of 58%, especially recent pages, implying that these pages are often used in conjunction, and with a purpose [3]. Integrating a view of the relationships within a user’s current web activity into the workspace would then prove useful, allowing peripheral access, informal overview, and manipulation of resources relevant to the task at hand.

In contrast, most browsers privilege a resource-centric model by only allowing direct access to the current and previous page, and placing history and bookmark information a few steps away, in a menu or a window separate from the current page. Traversing a link completely replaces the current page with the linked page, and ignores any relationship between the two. A web search will produce a result page that displays a set of matches to the given query. Visiting a match replaces the result page, and returning to the result page will involve backtracking through the user’s path, replacing the visited match. Besides keeping a mental record of the various visited pages, the user has the option to open each match in a new window, open a history applet, or save each result in a bookmark; all are cognitively heavy and require the user to do the bookkeeping.

History and Bookmark information stored in static trees and lists can be only sorted according to certain fixed parameters, mostly derived from the URL, and are designed to locate single pages rather than expose a set of interesting pages. The process for using such information using these tools begins by formulating an information need and resolving it into a set of possible matches, bounded by two points (between two dates for example). After opening the list, and navigating to an appropriate, manageable list, the user must then translate their set of pages into the titles given by the pages, through trial and error. In short, the cognitive overhead is much too high because it requires the user to know to a fairly certain degree the exact page to retrieve before beginning the search.

Recent browsers have moved history and bookmark system into a side panel, displayed on the left side of the main browser window. The side panel, like most browser tools, only displays the title of the page thus limiting any comparison among other variables including the search term. Additionally, trees and lists obscure the degree of relation between resources they display because the distance and size of every page is the same. In a list sorted by date, the second item could be 10 minutes older than the first, or 10 hours
older but it will appear almost identical to the one above it.

In the case of bookmark systems, the use of a static tree structure is especially limiting, because the strict containment property of trees forces each bookmark be contained by a single folder, causing a cognitive overhead to maintain order as organizational needs change. Previous research has discussed a need to allow more informal or casual methods of organization than trees within information interfaces, like ‘piling’ [2,8,9,11]. These studies have shown that informal organizations are common in work processes, and casual structures support iterative and dynamic views of information relative to the task at hand. This is especially true in a system like the Internet where shared meaning between diverse resources may not be apparent until later use and further discovery. We feel that allowing informal views of resources not only exposes the meaning and structure of the Semantic Web, but also facilitates the use of and interaction with such information. Informal views allow the user to make quick generalized relationships between resources as they work, which helps to cope with the ambiguity and diverse composition of a distributed system like the Internet, which should be exposed, not hidden by such an interface.

3. Towards a Conceptual Framework

The focus of our research, as stated above, has been to visualize aspects of the Semantic Web for the purpose of facilitating multiple interpretations. Simple, interfaces are needed in order to achieve the integration goals stated in the Semantic Web Activity Statement and allow the benefits of the Semantic Web to be an integral part of end-user applications. We have chosen to address these needs in terms of a one-dimensional interface. The one-dimensional interface allows us to achieve our goals to create a compact and cognitively lightweight interface to the Semantic Web that allows casual organization and informal overview. Also, since the data set of the web is diverse, content should be encapsulated in a simple metaphor that can aggregate many different types of data, which may only share some comparable property, like last access date.

The metaphor of the Collated Path is a “Page”. The metaphor is easily understood because of its connection with web pages as well as paper documents. Yet, a “Page”, or “document” is an abstract concept, which can represent any kind of information resource. A Page could represent a message posted to a mailing list, a stock portfolio, or a fragment within an XML document. In short, a Page is any resource that is interesting and comparable.

The Collated Path takes a data set of Pages, which are “collated” into some meaningful order. The order is determined by a floating-point value, which allows a more relative comparison between pages, such as “Page A is 2.4578 days older than Page B”. Such comparisons are lost in other visualizations, such as current history tools, and the Aurora/Flash projects [4]. Our design uses two such collations to render the Page. On the one-dimensional line, the first collation defines page position on the line relative to other pages (the x-axis). The size (y) of Pages is not a function of position (x), and a second collation defines the size of the page relative to the other pages. The collators of these two criteria can be equal, in which case the pages will appear sorted (since size is a function of position).

Figures 1 and 2 illustrate a sample layout of a set of pages with the collators equal to each other (produce the same values). The smallest valued pages appear towards the left, and the higher to the right. A case where the collators were different, for example the x collator collated by date, and the size collator by relevance to some search terms, would create

Figure 3: A Collated Path using different “Collators” to map position and size to its “pages”.
visualization where the size of neighboring elements could be very different (see Figure 3).

The advantages of this framework are simplicity and extensibility. The visualization is quite simple and easy to decipher, without ambiguity within the details of the representation. Higher dimensional interfaces such as the Hyperbolic Tree [5] often introduce ambiguity with less structured data because arbitrary data sets do not have immediately recognizable mappings into n-dimensional space. Positioning information on a path is simple and easily controlled, allowing direct-manipulation and modification of the parameters that affect it, while still keeping the representation recognizable and coherent. The use of a timeline style presentation draws on user’s familiarity with traditional timeline methods of positioning historical objects in relation to each other, but has the added benefit of being highly compact. Other works, such as LifeStreams and Time-Machine Computing have argued that temporal models of organization are more effective than spatial, and our visualization abstracts the timeline to display arbitrary information including dates [9,8]. An alternate model considered was the path model, based on breaking web activity into discrete paths, or groups, of meaning [10]. Yet, this model proved too specific to certain data sets to allow diverse expression, especially in a compact form. In order to allow discrete differentiation, paths must be resolved to the spatial metaphor, which we’ve mentioned is too bulky for the task. In addition, other work has argued that the applicability of paths to larger work goals is often negligible [3].

We consider the Collated Path extensible model because the comparison routines are high-level and simply return a floating-point value. Using a floating-point comparison allows for relative comparison, beyond simply “greater than” and “less than”. Because the comparisons have a high precision, it allows for comparisons to be layered, or chained, so that two or three comparators could be combined to form higher order comparisons like “give me all the messages on this mailing list related to cooking within the past year, privileging authors with high post counts”. This structure will not distract the user from being able to layer interpretation of the relationship of images of documents, which is a more open-ended interpretation possibility. Implementing a comparator is also very similar to creating a standard comparison function as used in many programming languages. Comparison functions like these simplify the human/computer interface into a language both are fluent in. The machine will not be able to interpret why one Page might have a higher value than another, but that information gets passed on to the user who can bring a nuanced context dependent interpretation to the display. Our tools don’t always need to understand all of the information it processes, just enough to effectively present it.

We expect that many data sources will be able to exploit the functionality of some standard collators, such as Date, Numeric, String, etc.. However what these collations mean will vary by application and usage. Here we list some possibilities the Collated Path gives for representing these meanings.

Temporal Collation
Many current tools use temporal information to organize information. Browser history and versioning systems are two such examples. As mentioned earlier, others have argued that time-centric metaphors for information organization are more effective than spatial ones [8,9]. We expect temporal collation to be the predominant or the default collator. Browser history has a direct connection with temporal metaphors, and studies have shown that within recurrent systems like the web, recency is a prime indicator of use [2].

Semantic Collation
Given a set of pages it may be interesting to collate the pages according to their relevance to some search terms, moderation level, or other semantic data. Such data could be collected from outside sources such as an annotation engine, or generated internally. In the case of a newsgroup, for example, a Collated Path could be used to position the pages in temporal order, from oldest to most recent, and let page size be determined by relevance or moderation level. This would create a view that would display relevance according to time. A similar collation could apply to news feeds, allowing the user to highlight those stories relevant to a certain topic.

User-Defined Collation
In certain cases, it may be interesting to allow the user to directly set the values of pages. For example, while size could be mapped to any value, the position of pages could be set by allowing the user to position them interactively, thus creating user-defined meaning in terms of piles. Such an application might be a non-hierarchical bookmark system allowing iterative and dynamic relationships, while allowing the machine to map another variable like use rates to the size of the Pages.
Compound Collation
There are certain applications where we do not want an exact representation of the relative values of Pages. For example, in applications of temporal collation, pages with very recent time values may be of more interest than older pages. In this case, we’d like to transform the visualization to privilege recent pages, and a compound collation would do the trick. A compound collation accepts the output of another collator as input, such as a temporal collator and transforms it to create its output. To achieve the example above, we’d apply the transformation $x^n$ to the output of the position collator, with $x$ being an output value and $n$ being a desired degree, such as 2. An example of this is shown in Figure 1. Another possibility for compound collation is to use a third collator to weight the input values. For example, we could combine a User-Directed collation with another collation, allowing the user to apply weights to specific areas of the line in order to get greater detail.

With these, and other collators, we expect the range of visualizations to be quite large, and the Collated Path to be quite expressive, even beyond its single dimension.

4. Exploratory Design and Observations

To understand how our framework might perform an exploratory implementation was created in the Java programming language. The intent was to identify what relationships the model would express within the data set, what visual cues would represent the relationships, and to address any issues in the visualization. Figure 4 diagrams the common features of the Collated Path having both collators set equal (also compare with Figures 1,2 and 3).

One of the most interesting features of the Collated Path is that the items (weighted by floating point numbers), do not have an even distribution in either their position or size. Such differences in distribution help express relationships between pages, and the set as a whole. When such differences are severe, gaps appear in the line creating disjoint groups of pages. Such a phenomenon would occur in a Temporal Collation, for example, when periods of idleness occurred in the data set, and in a Semantic Collation when a certain subset of pages are of much higher relevance than the rest of the set. When the same collator determines both the position and size of the pages (as in Figures 1,3,4), an implied curve is created by the edges of the pages (schneiderman). The implied line is useful as a guide to navigation since it reinforces comparison. For example an implied line allows the user to see where a page exists in relation to the pages to its left and right, because the line acts as a guide for measurement. The implied curve shown in Figure 4 is a line, because a strict application of the framework described earlier will be linear, such that the position of a page on the line is proportional to its value relative to other pages. A median value will get mapped to the center of the path and be half as large as the largest value. The curve need not be linear however. For example, in the case where we have used a compound collation to apply a transformation to the positioning collator, such as $x^n$, the curve will resemble the graph of the n-th root of $x$, as in Figure 1.

As mentioned above, the relative sizing and positioning of elements allows for informal overview of elements, due to its simplicity and quick recognition. If the Collated Path shown in Figure 1 were to be interpreted as displaying a browser history list, it is easy to discern where, in general, certain pages lay in relation to each other. In addition, when gaps are created, the set is broken into distinct subsets that further aid in breaking the aggregate data set into distinct groups, and as mentioned above, the implied curve aids in quick comparisons to nearby objects. When the data set becomes large, or pages are extremely similar, pages may be displayed close together and the differences between them will be displayed equally small, eliminating the ability to perform edge-detection and visual comparison important to the search process [2]. As a solution, we have designed but not yet fully implemented a simple zoom functionality which, when a user clicked on an area on the path, that area would be emphasized, scrunching the areas to the left and right. This is easily implemented as a compound collation. A core benefit of a zooming interface in this context is that the user does not have to set up bounds for a view (i.e. between

Figure 4: A labeled model of the Collated Path's distinguishing features.
two values), but only click on the area of interest, keeping the focus on the interesting material. To return
to a wider view the click would be just below the axis
line. Again zooming out around the area of interest.
There is little aim required with this navigation
method, in so much one doesn’t have to ‘hit’ a menu
slider. The intent is to allow the cognitive focus to
remain on the document of interest while integrating
powerful navigation. [12,13,14]

5. Related Work

There are several related projects that explore
other alternatives for reinforcing semantic and
contextual cues in browsing.
The Mozilla Project has work-in-progress creating
more integrated and interesting views of user-oriented
data [4]. Starting with Netscape Navigator 4.5, a
“Smart Browsing” feature was incorporated to provide
a list of pages related to the current page. In addition,
the Mozilla Aurora Project is developing a user
interface element that will unify user resources such as
email, history, and other RDF data sources. Another
project in planning stage is the “Flash Panel”, a list
that displays diverse “event” information, such as
email notifications, instant messages, or data feeds like
stock quotes. All of these however, still use
hierarchical trees and lists to display their information,
providing a familiar, but less expressive interface than
we feel is needed.

Our choice of a one-dimensional interface builds
on the crucial work done on LifeStreams [9] and Linda
before it. LifeStreams also allows access to diverse
document types, outside of tree structures, like file
systems. LifeStreams demonstrated the power of using
a single dimension, time, as the meta field around
which to display equal value to every document sort. It
is this commitment to the simplicity of the information
structure that is equally a commitment to reordering
around a parameter of interest. This is the critical
difference with our system and investigation. By
scaling the size of the items, we introduce another
“half-dimension”, breaking the ‘equal visual value’
principal implemented in LifeStreams products. This
threatens the strict simplicity that privileges
manipulability. However, our design intuition is that
an X/Y plot, (in which the x-axis is by default time) is
such a naturalized visual form for information that it
does not complicate the ability to visually compare
the image positions. Although it changes the relative size
of the document images it does not alter their x-value
position on the dominant axis. The overlapping pages
ensure that the bottom left corner is an absolute visual
reference while allowing more semantic value without
adding cognitive work. The problem with making
visual interfaces is that one needs to exploit very
familiar visual display forms, precisely because of
their familiarity. This seemingly small difference
makes it an additional contribution.

Although each system is generalizable, the designs
address different domains of use. The emphasis of this
project, and our future work, is focused on facilitating
the use and interpretation of information that the user
is not always familiar with. The
Mirrorworlds/LifeStreams project, on the other hand,
is oriented towards more familiar and personal
information sources, stored on local media. We expect
this difference to emerge more in further investigation.
Our hypothesis is that LifeStreams is optimized as a
general, albeit powerful, file retrieval while the
Collated Path is a ‘tool to think with’, to layout
different relationships, to use in an active sense
making process. Consequently the navigation and
zoom strategy is quite radically different.

6. Future Work

Future work will be directed toward integrated the
Collated Path into a working environment to facilitate
detailed investigation of actual use. We will implement
the Collated Path as a component for the Mozilla
browser, which will provide an environment in which
we can take advantage of multiple data sources such as
the web browser and email client. This will also allow
us enough functionality to conduct our evaluations in
an actual workspace and capture the vagaries of
combining this tool in the uncontrolled context of
multiple competing windows, multiple demands on the
person using the software, in the real context of trying
to get work done.

The following situations will be structure around the
following comparison: open and closed research tasks;
self and other generated file collection; textually and
visually driven information types.

Open-ended research is a context in which a person is
looking for particular information but also learning
from the subject of investigation more generally;
developing a sense of the different approaches to the
problem, how popular each approach is, who is
prolific in the area, how it applies in other fields, etc.
It is a task something like writing a paper, in which
you learn as much from the search as you do from the
actual object of search. This sort of information would
be legible from our tools, for example. doing a search
of one-dimensional interface papers, one could see the clustering along the x-axis (time) in relation to the number of publications, peaking around the early 90s under the influence of Mirrorworlds. One could tell peak areas of publication around SIGCHI, versus the publication in SIGGRAPH, because one is clustered in April, and one set in May. One could set the citation rate as the Y-axis (size) and therefore recognize how influential a document has been. If you wanted to more carefully investigate this, one could collate the citation rate along X to get a sense of the citation behavior of the field.

Closed-ended research is the searching for a specific piece of information. Our hypothesis is that in this sort of investigation can also benefit from the Collated Path because it allows multiple ways to approach and see an overview of results, maximizing the chance for finding a specific piece of information. For example, if a compound collation is available the data can be weighted by many diverse variables. The user also has the option to rearrange the views to answer more interpretive queries like “the documents I am looking for was published around 98, by the EPA or some other federal funding agency, and had quite a few authors. It had a diagram in it that was perfect”. The user could first map the page size by date, and relevance to the EPA, and number of authors, and visually find the page. This combines not only the machines ability to sort and search, but facilitates the user’s visual interpretation abilities.

We are also interested in observing and understanding how people use the files generated by their own usage (i.e. the history window of the web browser), compared to a set of random web documents, or a set of pages from an organization’s internal website. Our hypothesis is that in the case of self-generated material there may be a greater emphasis on visual familiarity when choosing a page. The reverse may be true for a searching amongst the more visually consistent intranet of a corporate site, where the size of the pages would provide more contrast than the visual representation of the page. We would expect a set of random, possibly unvisited, web pages to be set between these two. In addition, the data set of the search will affect how well a visual icon can be created, and how recognizable certain can be to the user. This feedback may provide us with a metric that indicates when to privilege the visual manipulative strategies provided, versus the collation facilities provided.

The final comparison we are interested in is between more textual based information like email and message boards versus large visually driven information like a photo album. We hope this will provide further details relating to how spatial, imagistic and organizational layouts interact. For example, we know that even text documents can be recognized visually, allowing one to recognize different documents by their structure; compare an academic paper to a header file of a program’s source code. It may or may not demonstrate that images and text documents can be treated equivalently as visual resources.; Our hypothesis is that images are less visually useful as thumbnails in an unfamiliar data set, because there is a less predictable structure of comparison.

These and other questions will be observable from the daily insitu use of the Collated Path, illuminating the complex interrelationships among web resources and the ways in which we use and make sense of them. Our tool is not intended to proscribe any top-down sense making strategies but to reinforce diverse approaches with a simple tool. It is this sense of privileging many different sense-making processes that is the great potential of the Semantic Web.

7. Conclusions

The Collated Path has promise as a general tool for navigating relationships and comparisons between resources in the Semantic Web. We feel it addresses specific needs of the Semantic Web to privilege the relationships between resources as much as the resources themselves. It has value in its ability to compress subtle differences into a compact representation, while still allowing complex relationships to be visualized. It demonstrates our hypothesis that using simple representation strategies facilitates complex manipulation strategies. We understand sense making to be an active process that involves many different resources and interpretations. Further, the simplicity of the design makes it usable in many mainstream applications, while it takes advantage of advanced features of the Semantic Web to model aggregate data sources from diverse sources directed by both humans and machines.

8. References

[1] Semantic Web Activity Statement
World Wide Web Consortium (W3).
<http://www.w3.org/2001/sw/Activity>


