Semantic Lenses: Seamless Augmentation of Web Pages with Context Information from Implicit Queries

Martin Rotard ∗ Mark Giereth Thomas Ertl
Visualization and Interactive Systems Institute (VIS)
Universität Stuttgart
Universitätsstraße 38, 70569 Stuttgart, Germany

Abstract
We propose a novel method to support analytical processes by visually integrating context information directly into web pages. The context information is obtained from knowledge bases or services that provide additional information about certain concepts and instances of a knowledge domain. The approach performs implicit queries on the underlying information sources using the key phrases of the web page and visually presents the results to the user in a seamless way. The technique aims at improving the integration of different information sources into the user's task. The method has two phases, an augmentation and an interaction phase. In the augmentation phase, for each key phrase the availability of additional context information is checked. All phrases for which context information is available are highlighted by adding special markups and scripts to the web page. In the interaction phase, the user can click on a highlighted phrase to receive the associated context information. The information is displayed directly in the web page in an overlay that we call a semantic lens.

Key words: Visual Analytics, Implicit Queries, World Wide Web, Semantic Web, Human-Computer Interaction

1. Introduction
Visual representations and interaction techniques are a key topic in visual analytics research. In analytical processes huge amounts of complex data have to be collected, filtered, and provided to the users in an appropriate way (22). Interdisciplinary knowledge is necessary to support the interpretation of specialized information. In semi-automatic analytical processes users need to connect information from different sources in order to perceive relationships between entities of interest. However, when using different retrieval tools and methods, users often have to switch between various user interfaces.

With implicit queries it is possible to automatically take advantage of the user's context for getting explanatory or additional information. We refer to this kind of information as context information. Users can be visually informed about the existence of such information without having to switch the context. On demand the available context information can be seamlessly integrated and visualized. Thus, the concept of implicit queries is a useful approach when working with semi-automatic analytical processes.

Although there might be an overhead in executing queries whose results are not used later, we expect a great benefit from the implicit query concept because of the following reasons: It can be checked very quickly if there is context information for certain entities available at all. Users are able to detect the existence of multiple meanings for one and the same

Preprint submitted to Elsevier 22 December 2006

* Corresponding author  
Email address: rotard@vis.uni-stuttgart.de (Martin Rotard).
entity, which is important especially when working in interdisciplinary fields. Further, the users might be motivated to inspect context information, when their availability is visualized.

In our approach we assume that the user context is given by a web page within a certain set of knowledge domains. For each domain we assume the existence of knowledge bases or services that provide additional information about the concepts and instances of a domain. The key phrases of a web page act as parameters of implicit queries. They are matched with the information contained in the underlying knowledge bases or retrieved from underlying services. The availability of additional information is indicated by highlighting the matched phrases in the web page.

When users interactively select a highlighted phrase, the context information is presented visually by embedding it into a special panel that we call a semantic lens. This representation allows displaying the information integrated into the current web page without changing the layout. The context information is presented as compound document containing graphical and textual representations that provide background or detail information about the entity.

The rest of this paper is structured as follows. The next section describes related work and shows the differences from our approach. Section 3 presents applications and scenarios in which the proposed methods are used. Then the overall system architecture is described in more detail in section 4. Interaction techniques and the results of a preliminary user evaluation are presented in the sections 5 and 6. In section 7 we discuss our approach and possible limitations. The paper ends with our conclusions and an outlook to future work.

2. Related Work

The representation of complex data structures is one of the key problems in information visualization research. Special graph layouts help to improve the understanding of the data (25; 11; 9). Visual graph analysis techniques support the users in identifying the overall structure of a graph and some of its key features (20; 24).

The term semantic lens is based on the concept of magic lenses where a panel can be moved like a magnifying glass over the objects on a screen in order to provide a special view on the objects. In contrast to magic lenses, the visualizations presented in semantic lenses are based on the information retrieved from knowledge stores. The goal of semantic lenses
is to provide quick access paths to related context information that is seamlessly integrated into web pages and thus to support the users in better understanding the content.

The concept of using lenses in user interfaces has been published already for different fields of application. Magic lenses were published by Bier et al. and Stone et al. (2; 21). Their work is based on earlier research that was done in more specialized techniques like fisheye views by Furnas (6) and by Perlin and Fox (15). Lenses have been integrated into web browsers to interactively change the rendering of web pages by Phelps and Wilensky (16).

Furthermore, Janecek et al. describe a technique called Semantic Fisheye Views (SFEV) that can integrate rich semantic models into the search process, such as WordNet (8).

Link augmentation is a hypermedia technique in adaptive hypertext systems. Bailey et al. combine link augmentation with a model of the user’s spatial context to achieve cross-domain adaptive navigational support (1). A method called fluid links that provide additional information at a link source to support readers in choosing among links and understanding the structure of a hypertext (26).

In particular, when describing knowledge domain concepts and instances using the Resource Description Framework (RDF) (10) as data model, the visual structure of the resulting RDF graphs may not be intuitive to understand. Great challenges are the visual presentation of and the navigation in RDF graphs. Different tools have been developed to support the visualization of RDF graphs, such as IsaViz (17), Welkin (13), or GViz (5), to name only the most prominent.

Our approach differs from previous RDF visualization tools in the following aspects:
- Only relevant parts of a knowledge base are visualized using the key terms of a web page as search criteria.
- Semantic lenses provide both, graph-based visualizations to outline the overall structure of the data and textual presentations for additional or detail information.
- Our approach allows a seamless integration into web pages by using Web Integration Compound Documents (14), that embed Scalable Vector Graphics (SVG) (4) and other markups in
XHTML documents.

- Semantic lenses can also be used to integrate non-RDF content and allows e.g. for previews of web pages, images, or PDF documents.

3. Applications

In the field of visual analytics it is important to illustrate applications and scenarios that can be used with the proposed methods. In this section we present three prototypical web based applications for different knowledge stores and services that take advantage of the augmentation with implicit queries and of the semantic lenses.

In the following applications the context information is provided either by RDF knowledge stores or by web services, such as the Open Patent Services\(^1\) of the European Patent Office, or Wikipedia\(^2\).

The first application uses a bibliographic database that has been crawled from the web. It highlights the author names in a web page, and presents the related publications and co-authors as a node-link graph, when clicking on the author’s name. The second application uses the Open Patent Service. It highlights the patent numbers and presents legal event information about patent families. The third application highlights the names of Wikipedia article titles and presents the full text article in a semantic lens.

3.1. Bibliographic Lenses

The bibliographic augmentation works on gathered BibTeX data that are crawled from web sites. The data is mapped to an RDF representation and added to a knowledge store. Furthermore, existing databases that consist of relations between authors, publication name, publication location, publication year, etc. may be used within this scenario.

Using the bibliographic data the names of authors can be highlighted in web pages by presenting the availability of data set entries. When clicking on a name, the related publications and the co-authors of this person are presented in the semantic lens as a node-link diagram.

As an example figure 1 shows the context information embedded in a semantic lens for an author. In the example, the lens is integrated into a search result web page from Google. To determine the authors all phrases on the web page are matched against the instances that occur in the knowledge base as authors of a published paper.

---

3.2. Patent Lenses

Patents are an expression of intellectual property as well as a representation of knowledge. As such, they are an important aspect for national and international economies. Reducing the retrieval and inspection time for patent documents can also significantly reduce the patent application and examination costs.

Patent lenses provide a structured view on patent metadata such as patent families, citations, applicant/inventor networks, or legal event information. When looking at web pages containing patent numbers (e.g. when doing a patent inquiry), this application highlights potential patent publication numbers by applying regular expressions. When selecting such a number, the Open Patent Service is called and the legal events as well as the family information is retrieved. The result is converted into an RDF representation based on the Patent Metadata Ontology (7) and serves as input for the context information visualization service (cf. section 4.3).

Using the timeline widget provided by the SIMILE project 3, the visualization service renders the data as a timeline showing the temporal sequence of the legal events. When the user clicks on an event, its details are presented in an info box (figure 2). This kind of presentation allows a quick overview of the legal status of a patent and its associated family members directly within the web page without forcing the user to perform a context switch.

3.3. Wikipedia Lenses

Wikipedia has become one of the most important information sources when searching for definitions and background information. But using the Wikipedia search form forces the users to leave their current context. In order to improve the integration of Wikipedia into the user’s context, Wikipedia lenses allow the presentation of related Wikipedia articles directly within the current web page (18).

The original web page is augmented with links to Wikipedia articles by matching the phrases of the web page against the Wikipedia titles in the knowledge store. The results of the inverse query are highlighted in the web page and linked to the article on the Wikipedia server (see figure 3). Optional filters may be defined to select the article categories to include, e.g. only articles about biology and medicine. For each article the categories have to be stored in the RDF knowledge store to enable this feature. This information is extracted from the Wikipedia XML database dumps. Furthermore, the main categories for each article have to be determined.

The users can click on a highlighted phrase to present the latest Wikipedia article in a semantic lens. To optimize the available screen space in the lens only the body of the article is presented not including the navigation bars on the top and on the left.

4. System Architecture

In this section we describe the general system design in more detail. The system is implemented as a web application that can be used with standard browsers 4. It has three major components: The first component is the augmentor service which preprocesses a web page, checks the availability of context information by implicit queries, and augments the original web page with highlighting and linking information. The second is the context information visualization service which generates a visual presentation of the available context information. The third component, the semantic lens engine, handles the user interactions and the visual presentations on the client side. An overview of the components and their invocation sequence is shown in figure 4.

The services are called via HTTP. The augmentor service can be invoked with the following simple JavaScript statement that can be added to the browser’s bookmark list:

javascript:location.href='http://SERVERName/Augmentor?url='+encodeURIComponent(location.href)

When users visit a web page and request context information about the phrases on the page, the augmentor service is invoked with the URL of the page as argument (figure 4, step 1).

The augmentor service consists of three subcomponents: The text processing engine downloads a web page, extracts the textual content from the page, determines the language by using characteristic stop words and trigrams, and analyzes and

3 http://simile.mit.edu/timeline

4 Currently we have optimized our application for the Mozilla Firefox browser, since Firefox supports compound document by inclusion for XHTML and SVG. For other browsers, compound documents have to be build by reference and plugins might be necessary.
lemmatizes the content using the statistical TreeTagger (19). Heuristic filtering, e.g. nouns, can be applied to reduce the amount of phrases\(^5\) for lookup. After this step, the remaining phrases are used to generate queries on the index.

4.1. Implicit Queries

Key phrases of a web page are used to automatically generate queries on underlying indexed knowledge bases. Multiple queries (several hundreds for each phrase) have to be matched with millions of small sized terms, e.g. Wikipedia article titles. This differs from classical information retrieval where only one query is matched with a document set containing millions of medium to large sized documents.

The implicit query scenario is outlined in figure 5 and can be described more formally as follows: The goal is to match a text having a word sequence \(w_1, \ldots, w_m\), of length \(m > 0\) with a set of index terms \(\{t_1, \ldots, t_n\}\), where each term \(t_i\) has a word sequence \(w_{i1}, \ldots, w_{ix}\) of length \(i_x > 0\). Typical values are \(m < 2,500\), \(i_x < 5\), and \(n > 1,000,000\). The result of the matching procedure is a list of matching phrases and their positions in the text (e.g., the table in figure 6).

Overlapping matches are possible, so that one word of a phrase can be contained in different index terms. In figure 6, for example, the word \(w_{i+2}\) at position \(i + 2\) is part of the terms \(t_2, t_4,\) and \(t_5\).

\(^5\) In this context, we define that a phrase is a sequence of words in a natural language contained in the content of a web page. We define a term to be a word sequence in an index.

A phrase can have different meanings, and thus different context information associated to it. Different meanings are characterized by having multiple terms associated with one phrase, e.g. the phrase \((w_i, w_{i+1})\) has two meanings, one defined by term \(t_1\) and one defined by term \(t_3\).

Overlapping and multiple matches have to be taken into consideration, when phrases are visually highlighted and links to the associated context information are generated. We group overlapping matches into the longest continuous match and provide for each group a menu, in which the user can select an appropriate meaning.

4.2. Web Page Augmentation

The last step within the service is augmenting the original web page with additional HTML tags and JavaScript code, so that each matched phrase is highlighted and initialized to trigger the creation of one or more corresponding semantic lenses. The initialization includes the generation of linking in-
formation that either triggers a context information query in one or more knowledge stores and a subsequent visualization procedure, or links to a resource having further information about the phrase, for example a link to an encyclopedia.

An augmented web page is an enriched version of the original web page and also contains JavaScript code to handle the user interaction on the client side that we call the semantic lens engine. An augmented page is returned by the augmentor service as response to the client request (figure 4, step 5).

4.3. Visualization of Context Information

When the user clicks on a highlighted phrase, the associated context information is generated and presented to the user. The service is called with three parameters: (1) the phrase for which context information is created, (2) the knowledge stores, where the information has to be looked up, and (3) the presentation description specifying how the context information has to be collected and presented.

Since the result of context information queries is an RDF graph, we implemented the visualization service based on the Fresnel display vocabulary (3). This allows for a generic way of defining the presentation of context information and sharing this presentation knowledge between semantic lens applications.

The task of presenting RDF graphs in a human-readable way mainly consists in specifying what information contained in an RDF graph should be presented and how this information should be presented. Fresnel’s two basic concepts are lenses and formats. Lenses define which properties of one or more RDF resources to display and the order of presentation. Formats determine how to render the resources, their properties and values.

Context information can be presented graphically, for example as node-link diagrams, or textually, for example using XHTML. In our approach, we combine the two paradigms in order to generate compound documents containing graphical elements in SVG inside general XHTML documents. The context information visualization service returns a compound document as response to the user’s request (figure 4, step 8).

4.4. Semantic Lens Engine

On the client side, the compound document is integrated in an overlay, which we call a semantic lens. The semantic lens engine and the available interaction techniques are described in more detail in section 5.

4.5. Customization

The framework can be customized for different applications by performing two steps: First the augmentation has to be adapted. This can be done in various way. When working on RDF knowledge bases, one way would be to specify queries that identify the instances to be highlighted. The simplest interface to the augmentation service is a list of index terms. This has been done for the Wikipedia example.

The second step involves the specifying of Fresnel descriptions to define what context information to present and how to present it. For the rendering a set of serializers is provided, e.g. a renderer for generating node-link diagrams as shown in figure 1, or a renderer for generating timelines as shown in figure 2.

5. Interaction Techniques

The interaction techniques are one of the most important aspects for a new tool to be widely accepted. In this section we describe the interactive handling of the semantic lens. Figure 7 shows the functions of the handle on the semantic lens. The right part of the handle is visible when the content in the context information is presented as graph.

![Fig. 7. Handle of the semantic lens](image-url)

By clicking on a highlighted key term the lens is launched with the context information near the
corresponding key term. The lens can be minimized and closed by pressing the corresponding button on the handle. The user is not restricted to have just one semantic lens. Our system can handle multiple lenses that can be integrated into a web page and can overlap each other. When the user scrolls the web page the lens remains fixed in that position related to the context. On double clicking the lens handle the context information is presented in full screen.

At the top of the handle there is a viewfinder. To show the semantic context information of a key term the viewfinder can be used by moving it over the key term. By using the viewfinder key terms can be selected that are inside a hyperlink. Therefore any open semantic lens can be used. Additionally, a new lens can be opened with a button in the toolbar of the browser.

A graph displayed as context information can be larger than the entire display area of the semantic lens. Therefore zooming functionality is necessary, which can be found in the two buttons on the handle of the lens. The viewport can also be moved by dragging the background of the graph. While the user moves the viewport of the graph the mouse cursor changes its symbol as feedback for the user. Since the graph is rendered as SVG the text in the context information graph is accessible and can be searched. The button on the handle with the square icon toggles the transparency of the background of the lens in three steps between full, semitransparent, and opaque.

Each semantic lens builds up a history of the context information when the content is a graph. This history can be navigated forward and backward with the corresponding buttons on the right side of the handle. Each node of the graph can contain a hyperlink to the corresponding context information graph of its key term. This enables the users to explore the context information by browsing the graph structure and by navigating in the history.

Data from different knowledge stores can be combined in one graph of the context information. A tool tip can be assigned to each node that presents additional information (e.g. contact information, full titles, hierarchy level, etc.) that comes from another knowledge store (figure 1).

6. Preliminary Evaluation

The usefulness and the usability of the semantic lens was tested in a short evaluation with twelve employees and students at our university. Before we started each test we introduced the subjects into the 'thinking-aloud' method. The duration of each test was ten minutes. Before the test the subject were given a short tutorial on how to augment the web page by using the buttons in the toolbar of the browser and how to handle the lens with the viewfinder. After the free exploration phase the subjects where asked to fill out a questionnaire, in which they had to rate the usefulness as well as the ease of use in several questions. Concluding the subjects were asked to propose some applications that would profit of the semantic lens concept.

Our evaluation led to some preliminary results. All subjects noticed the highlighted key terms and intuitively clicked on them. Then the first semantic lens appeared and showed the context information to the highlighted key term. All users started to explore the functionalities of the lens and most of them pointed out that the handling is easy to use. Some of them had stated problems in using the viewfinder. The navigation within the context information was found to be easy and intuitive.

The questions about the usefulness of the semantic lens where rated positive without exception. In particular, the implicit queries where pointed out to be very useful. Especially the key term selection by using the viewfinder was new but easy to handle for the users. The navigation inside a graph of the context information was not discovered by all subjects.

7. Discussion

In the following sections we discuss pros and cons of our approach, in particular the system architecture and limitations.

7.1. Pros and Cons

The construction of semantic lenses in the way described in section 4 has obvious advantages:
- Semantic lenses provide a flexible mechanism for integrating knowledge sources into the context of web pages.
- The lenses are displayed adjacent to the phrases in the web page but can be moved to any position on the page. This also allows them to be
used as magic lenses for different kinds of content, like context information, web page previews, PDF documents, images, etc.

- With our approach it is possible to have multiple independent semantic lenses integrated into one page, which enables the comparison of context information from different data sources.
- The integration of semantic lenses does not disturb the layout of the web page and in case of a semitransparent presentation does not occlude the web page and saves precious screen space.
- Our approach does not depend on having special browser plugins that have to be developed separately for each browser.

There are also the following shortcomings in our approach:

- Client functionality is programmed in terms of JavaScript and additional HTML tags. When working with different browsers, existing inconsistencies have to be handled by different program code.
- There are certain limitations, when augmenting web pages using HTTPS. Also, HTTP sessions can cause problems since the server is not aware of the session context. Frames and existing JavaScript code on the web page can also cause problems.
- Current SVG viewers still do not support hardware accelerated rendering. Complex graphs will be visualized slowly.

7.2. Limitations

Due to the ambiguity of the natural languages, it is possible that a phrase has a different context on the web page than in the knowledge base. In the current stage the mapping between phrases on a web page and instances in a knowledge store is purely syntactic and only morphological features are considered.

Furthermore, there are still problems for matching names concerning semantic and syntactic ambiguities. Semantic ambiguities exist for example, when different persons have the same name (cf. (23)). Examples for syntactic ambiguities are abbreviations (e.g. Jeffrey D. Ullman, J. D. Ullman, J. Ullman, or Jeff Ullman), nicknames (e.g. Michael vs. Mike), permutations (e.g. Liu Bin vs. Bin Liu), different transcriptions (e.g. Andrei vs. Andrej vs. Andrey), accents (e.g. Muller vs. Müller vs. Mueller), ligatures (e.g. Weiß vs. Weiss), case (e.g. Al-AAli vs. Al-Aali), or hyphens (e.g. Hans-Peter vs. Hans Peter) (cf. (12)). In general, when working with data of one domain one can expect fewer semantic and syntactic ambiguities.

When looking at scalability issues, we found, that the limiting factor is the time needed to execute queries on the underlying RDF stores. This applies for the retrieval of index terms and also to the retrieval of context information. Unfortunately, it is difficult to make a general statement about the performance of the system, since it heavily depends on the application as well as on the structure of the knowledge bases or services.

8. Conclusions and Future Work

For applications in the field of visual analytics we have proposed a new method for visually integrating semantic resources into standard web pages. Our method is based on implicit queries on configurable knowledge stores by using the phrases of the current web page. The results are highlighted in the web page. We have described a special panel that we call a semantic lens that can be integrated directly into the current web page. In this lens the context information related to the selected key phrase is visualized interactively. To select the highlighted phrases a viewfinder on the semantic lens can be used. Furthermore, web page previews of hyperlinked documents can be displayed in the lens by using the viewfinder.

We have discussed our proposal and pointed out that our user interface has advantages for the user in the handling and the presentation. Furthermore, there are disadvantages in some technical aspects especially the performance on large knowledge bases. We have presented the results of a preliminary evaluation. Although the semantic lens was new to all test persons it turned out to be very useful. The user test results are very promising and show us that some work has to be done in the graph layout and the presentation of the compound documents.

In the preliminary evaluation we have asked the subjects to propose some applications where our concept could be used in and their answers were multifaceted. Since viewers for Rich Site Syndication (RSS) can be easily integrated into web browsers, our concept also allows to augment RSS feeds. Even e-mails that are presented with unified messaging systems can be combined with our system. Further-
more, messages of Usenet discussion groups that are embedded into web pages could be processed.

Future plans are the integration of existing web services that could be used in visual analytics, such as Google, Ebay, Amazon, etc. Finally we would like to do a broader user evaluation with more subjects.

9. Acknowledgments

We thank Martin Kraus, Thomas Klein, Guido Reina, and Mike Eissele for their constructive feedback and for their ideas to improve our approach.

References


