

# VISFACET: Facet Visualization Module for Modern Library Catalogues

Miriam Allalouf,<sup>1</sup> Dalia Mendelsson,<sup>2</sup> and Evgeniy Mishustin<sup>1</sup>

<sup>1</sup>Azrieli College of Engineering, Jerusalem

<sup>2</sup>The Hebrew University of Jerusalem

**Abstract.** The “next-generation” catalogues of academic libraries provide a discovery layer that contains faceted classification and search features and suggested topics selected by their rankings. To improve the discovery process, this paper demonstrates an interactive faceted visualization box, termed VisFacet, that extends the catalog interface and allows users to narrow or broaden their search results filtered by suggested topics or facets in an interactive manner. The VisFacet software visualization module was integrated into and contributed to the VuFind open source software system. VuFind is a development portal that enables libraries to customize their own catalogue interfaces and discovery layer. Thus, extending VuFind with VisFacet provides all catalogues using VuFind at their system’s core with the benefit of having an infographic interactive search box. We will describe the challenges encountered during the development of the VisFacet project, including the user discovery satisfaction questionnaire results.

## 1 Introduction

The traditional academic library catalogue enables end users to search for and find requested information. With the development of the Internet, and the introduction of Google as a major player in the information retrieval arena, the search patterns began to change. The Google-like search method influenced the way that library catalogues are being developed to offer a similar user experience [1], [2]. Traditional library catalogues were designed for the tasks of searching, identifying, selecting, and accessing information. The information retrieval is based on predefined indexes. The user had to be familiar with the specific fields involved in the information retrieval process. For example, users were expected to know how to construct complex search queries, utilize subject searching, and apply Boolean search operators [3]. Over time, the catalogue did change and a new generation of catalogues became more popular in academic libraries. The breakthrough of providing web-based online public access to library collections and resources did not require changes in the Integrated Library Systems’ (ILS) core. The new concept was developed only at the presentation layer of the Online Public Access Catalogue (OPAC). Additional indexing at the presentation layer provided the essential discovery infrastructure to the “next-generation” catalogues. Nowadays, libraries’ next-generation catalogues offer academic resources based on the advances of the information retrieval technologies; they are trying to meet the readers’ new expectations and enhance their experience by making library catalogues more user friendly, intuitive, and visually attractive [4]. The new discovery systems contain predictive search features (such as “Did you mean?”); user profile-aware content, such as tags, ratings, reviews, and comments; and a faceted classification.

The faceted classification feature that classifies items into multiple independent categorization schemes (or facets) is a topic from library science that has become

popular in information computing [5]. In this important approach, the information retrieval system allows the assignment of an object to multiple attributes, thus enabling the classification to be ordered in multiple ways, rather than in a single, predetermined order. For example, a collection of books might be classified using an Author facet, a Region facet, an Era facet, and so on. The Region facet consists of clusters of records from the list, each of which is associated with a region-related keyword. Performing a search of the catalogue, the system presents, along with the usual search engine results page (SERP), different predefined facets. By clicking on them, the user can narrow his search and refine the results set. All the suggested information is presented to users in textual format in several separated fields of the SERP. Using faceted navigation with suggested topics has proven to be one possible way of enhancing the user's navigation and exploration, but with the growing numbers of library materials, and users who demand easy access and high-performance systems, there is still a need for further development.

Information visualization is an emerging component in many scientific research areas such as digital libraries, data mining, and financial data analysis. Merčun and Žumer in [6] discuss the demonstrated importance of visual information in library catalogue presentations, particularly in simple visualizations such as tag clouds, time sliders for refining search, featured cover displays of new acquisitions, recently borrowed items, bookshelf display, and vocabularies such as "word cloud" in Aquabrowser [7]. In our opinion the next-generation catalogue, being an exploratory tool designed for discovery, must add the potential of information visualization for exploratory faceted search visualization to their capabilities. This paper describes the VisFacet plug-in framework we have designed and developed. VisFacet is an interactive, faceted search and navigation visualization system. It uses a book's existing metadata information to provide an intuitively understood and visually attractive graph of facets. It appears within and extends the SERP. The VisFacet software modules have been integrated into the VuFind system. There is currently no next-generation catalogue that integrates a visualized and interactive faceted search.

VuFind ([www.vufind.org](http://www.vufind.org)) is an open source, next-generation library resource portal that enables the development of a customized, faceted navigation system. VuFind provides a full set of modules to produce a customized cataloging system layer in which libraries can implement all the features they want, modify the existing modules to best fit their needs, and add new modules to extend their resource offerings. The Library Authority of the Hebrew University customized the VuFind system modules and built the HUfind next-generation catalogue. HUfind has been in production since 2012. Our VisFacet framework extends the VuFind system's capabilities with the visualization feature, and all catalogues that use VuFind at their system's core (such as HUfind) can benefit from VisFacet. Moreover, today there are several development platforms for modern cataloging of library systems, and none of them has a faceted visualization framework such as we are suggesting. VisFacet makes three main contributions:

1. It adds the visualized, interactive search capabilities of the facets to the VuFind system. The view includes objects filtered by a variety of similar topics and by facets such as Era and Region.
2. It adds a broadened search capability to the visualized view, which is a completely new feature.
3. It contributes the code to the VuFind open source project.

## 2 Related Works

In chapters thirteen and fourteen of their book [8], Shneiderman *et al.* provide a comprehensive survey of human-computer interaction in general, particularly in the context of information visualization for information search and retrieval. Latest trends in the information technology arena show that visualized representation of the suggested topics, where the user can see the relations between clusters drawn graphically and click on them, is more understandable and intuitive [9], [10], [6]. There are several works that explore data visualization as a way to enhance the digital library results page. ResultMaps [11] is a treemap-based search visualization system, developed as an extension to digital libraries. It uses hierarchical subject classification to map each repository document into a treemap and highlight items that correspond to the current query. ResultMaps works very well for small repositories consisting of hundreds to thousands of records, but does not scale well, since the interface quickly becomes unreadable with the growth of the repository. FacetMap [12] is another visualization approach to faceted navigation systems that enables the visualization by interacting with large databases. Though it is limited to the Windows operating system rather than the web. AquaBrowser [7] is a commercial next-generation library catalogue that provides its own data retrieval algorithms and unique features to satisfy the end user's high requirements. The "Discover" feature gives a similar visualization look to our VisFacet. It does not provide a narrow faceted-search capability: it performs a new search each time a user clicks on it. Tiara [13] is a text visualization system that was built to aid users in exploring and analyzing large text collections: given a user query, Tiara provides rich graphic interactions with informative and powerful visual text summaries. Thai *et al.* [14] suggest a new visual ordering of faceted visualization in which a matrix-based multidimensional visualization is used for modeling the relations between documents. There is a tight correlation between the information format, type of analysis the information requires, and the visualization design.

## 3 VisFacet Framework

The aim of this research was to enrich the discovery layer of the catalogue with an interactive side-box that presents the facets of the search results graphically. Fig. 1 presents the current textual view as provided by the VuFind package. The term for the search in this example is the word "king." The area above a set of results presents a list of **Suggested Topics** that were found in the records addressing the term "king," such as "history" and "kings and rulers." The area on the right provides us with the facets—namely, the books taken from the left column filtered by a variety of subjects. For example, within the "Author" category we can see a cluster that contains all the records (202) from the current set of search results that appear to be written by William Shakespeare. Thus, the search system allows the user to search from one search box and then narrow down the results by clicking on the various facets of the results.

The VuFind information retrieval engine, as well as that of most modern library catalogues, uses the MARC records as the core for its search engine index schema over the ILS core. MARC (MACHINE-Readable Cataloging) is the de-facto standard for the bibliographical records with metadata for each item (that is, book, journal, movie, and so on). The metadata are inserted as indexes in the VuFind internal database, along with their associated metadata. Given a term to search, the catalogue's search engine retrieves associated records from the search engine. At the same time, it retrieves a list of predefined facets (sometimes also called

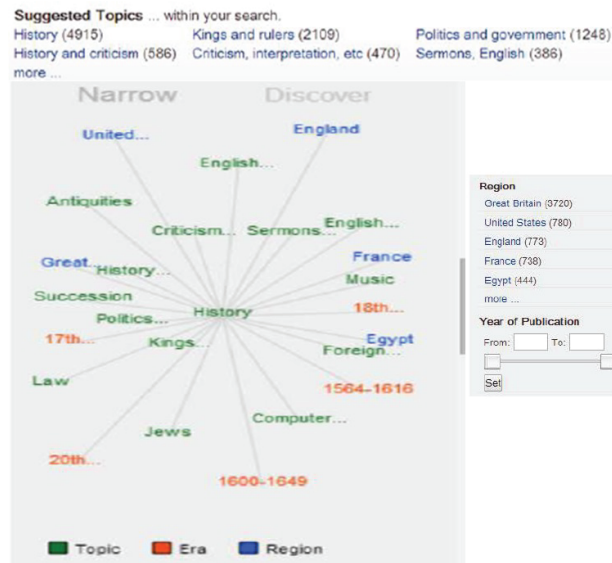
The screenshot displays the HUFIND search interface. At the top, the search term 'king' is entered in the search box, with options for 'Keywords' and 'Find Advanced'. Below the search bar, 'Suggested Topics' are listed, including 'History (4916)', 'Kings and rulers (2110)', and 'Politics and government (1246)'. The search results show 'Showing 1 - 20 of 22998 for search: 'king'', with a 'Sort' dropdown set to 'Relevance'. The results list includes several entries for 'Kings', such as 'Kings : introduction, revised version, with notes, index and map / Published 1904' and 'Kings : An account of books One and two of Homer's Iliad / by Logue, Christopher, 1926- Published 1992'. Each entry includes a 'Call Number', 'Location', and an 'Add to List' button. On the right side, there is a 'Limit Search' section with 'Libraries' (e.g., Mt. Scopus Library, Archaeology Library) and 'First Author' (e.g., Shakespeare, William, 1564-1616). Below this is a 'Region' section with 'Great Britain (3720)', 'United States (781)', 'England (773)', and 'France (738)'. At the bottom right, a 'Suggested Topics Cloud' visualization is shown, with 'Narrow' and 'Discover' options. The cloud contains various topics like 'United...', 'England', 'Antiquities', 'Criticism', 'Sermons', 'English...', 'Great History', 'Succession', 'Politics', 'History', '17th', 'Kings', 'Law', '18th', '1564-1616', 'Foreign', 'Computer', '1600-1649', '20th', 'Jews', and 'Ebook'. A legend at the bottom indicates 'Topic' (green), 'Era' (red), and 'Region' (blue).

Fig. 1. VuFind with VisFacet integrated.

clusters), each of which represents a logical conjunction of the current set of the results filtered by several predefined fields. For example, the *Topic* field is represented in the MARC records with the MARC code 650. If we choose to search for the topic “history,” the results list will display the records where the topic field, represented by 6##, holds the value of a number of records that include the field 6## with the word “history.” Choosing another subject like “second world war” will narrow the search, and all the resulting records that contain “history” but not “second world war” will be excluded from the current results list.

A visualized discovery box, which is intuitive, easy to interact with, and which integrates smoothly with web interfaces, is an essential addition to these evolving software systems. However, current commercial companies and academic libraries that develop tools for the customization of the discovery layer do not have this capability. Thus, an additional aim of this research was to extend the complex, modern catalogue software package with visualization capabilities while retaining its modularity and performance quality. Any visualization solution must integrate smoothly with the existing software architecture and be capable of using the retrieved information as any other other facet.

Our tool code was integrated into the VuFind open source system so that the VisFacet box is added to the user interface of the catalogue below the right facets section of the SERP (Fig. 1). The location can be changed to above the facets, should the system administrator wish to configure it that way. Section 3.1 describes the targets we addressed in the graphical design of VisFacet. In section 3.2 we describe VisFacet’s modules and how they were integrated into VuFind.



**Fig. 2.** Enlarged view of Visfacet. The suggested topic “History” for the searched term “King” appears in the middle. The “History” topic occurs in the metadata topic field 4915 times.

### 3.1 Visual Design

In the example presented in Fig. 2, the *topic* node that appears in the center of the star-like graph is “History,” a *topic* suggested by the system because it appears in the largest number of items that came up in a search for the keyword “king.” The rest of the topic nodes (in green), each representing a cluster of results, are connected to it in descending order in a spiral form according to their grading. *Era* in orange and *Region* in blue are two additional facets of our visualization. The color of each facet appears in the legend at the bottom. All the colors (including background) and sizes are configurable and adjustable. To save space, only the first word in a term appears near the nodes in the graph. When a user selects a node (by brushing the mouse over it) the full topic of the current node appears below the graph.

Any click on one of the nodes in the graph will narrow (assuming the toggle at the top is on the Narrow position) the current search results list according to the specific Topic, Era, or Region that was clicked. For example, clicking on the “France” Region facet will drop all records that do not contain France as their region. The **Narrow** toggle that appears at the top of the box is on by default, while the **Discovery** toggle is dimmed. Clicking on any of the nodes when **Discover** is on will trigger a completely new query with the clicked topic as a keyword. Since all of those topics were retrieved from the search engine index, this “Discover” function will never lead to an empty results set. The graph is redrawn with each

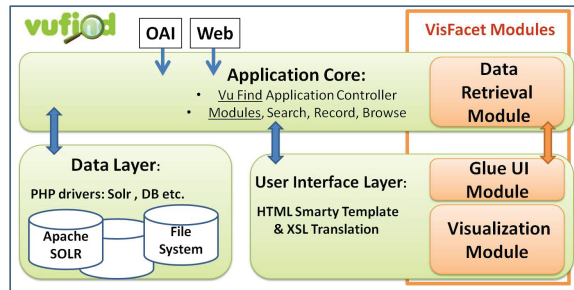
search operation, whether text or graphic. Readers of this paper are welcome to try it at <http://hufind.huji.ac.il/>.

The visualization of the suggested topics and the facets should help the end user discover a subtopic to focus on from a wide-area topic. To address this requirement we have applied several dimensions to the two-dimensional layout of the infographic box as follows:

1. **Link-Node form:** A star-like graph where the topics are connected with an explicit link to the center that intuitively highlights the relation between the topics.
2. **Spiral form and order:** Another dimension of connectivity arises from the fact that we have arranged all the topics in a spiral form, according to their frequency. For example, the topic *History* that occurs 4915 times in the resulting items appears in the middle of Fig. 2. It is closely connected with the *kings and rulers* node that occurs 2905 times in the resulting items. The other nodes appear in the spiral according to the number of occurrences that can be seen in the “Suggested Topics” box on the top of Fig. 1. The benefit of using a spiral form is that it spreads the topics evenly across any stretched box to optimize space usage as well as implicitly isolate the terms. Moreover, users gain a sense of the importance of the topics through their closeness to the center and thus may navigate between them using the spiral route.

To draw the spiral, we redesigned Vogel’s formulation [15] of Fermat’s Spiral [16] to fit our needs. The graphical objects are drawn within a canvas with predefined width and length. For each object in the list, the algorithm calculates its polar coordinates, which are composed of the angle and radius relative to the center of the graph.

3. **Colors** are used to distinguish among the categories, such as Era and Region.



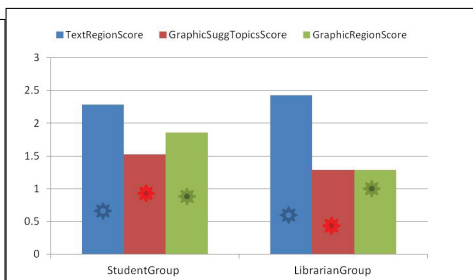
**Fig. 3.** VuFind with VisFacet integrated.

### 3.2 System Design and Implementation

Incorporating a visualization tool into the VuFind system, which is a generic engine for building and customizing library catalogues, is a natural evolution. The VuFind system has complicated software engineering and many code lines. Each part of the system is implemented in a different programming language that is best suited for its particular feature. Hence, the architecture and programming languages of VisFacet were selected and designed in terms of code coherency and performance. The VuFind architecture, shown on the left side of Fig. 3, consists of an application core and two main layers. The data layer contains a search engine index that can be distributed among several machines, or even different

Q1	<b>Text Search:</b> Does the Region facet helped to focus on a location? Answer options: 1. very helpful 2. gave ideas 3. little help 4. no help The rates for the answers are 3, 2, 1, 0 respectively
Q2	<b>Graphic Search:</b> Does <b>Suggested Topics</b> help to discover a topic? Answer options: 1. very helpful 2. gave ideas 3. little help 4. no help The rates for the answers are 3, 2, 1, 0 respectively
Q3	<b>Graphic Search:</b> Does the <b>Region Facet</b> help to focus on a location? Answer options: 1. very helpful, 2. gave ideas, 3. little help 4. no help The rates for the answers are 3, 2, 1, 0 respectively
Q4	How many steps did it take to focus on a topic in each search type?
Q5	Which search type better highlights the relationship between the <b>Suggested Topics</b> in terms of their number of occurrences? Answer options: 1. Graphic is better than text 2. Text search is enough 3. The graphic search is good only when combined with Text search
Q6	Which discovery is best? 1. Text 2. Graphic 3. Combined search

**Table 1.** A questionnaire that was presented to both groups: librarians and students. Each question presented in this table is followed by the possible answers and the rate for each option.



**Fig. 4.** Answers for questions 1-3 compared between the two groups. Each column presents the average rate of the answers for Q1 (TextRegionScore), Q2 (GraphicSuggTopScore) and Q3 (GraphicRegionScore). The flowers in the middle of each column show the Q1, Q2, and Q3 standard deviations: 0.6, 0.94, and 0.91 for the students, and 0.53, 0.44, and 1.05 for the librarians.

libraries, and be updated daily. The application core that runs on the server side and is responsible for bringing data from the data layer performs all the required processing and then passes it to the user interface layer. Finally, the user interface layer is responsible for arranging the data. The VisFacet visualization subsystem is divided into three main components, each incorporated into the appropriate layer of the VuFind system, as can be seen in Fig. 3. The retrieval module, was written in PHP, retrieves the data from the search engine index. It obtains information about a user's current search, performs its own internal processing, and then returns all the required information needed to render suggestions. This information is processed and organized for the visualization in the Glue UI module that runs in the browser on the client side and was written in Javascript. The Glue UI module binds its own functions to the visualization module and "listens" to interactions. The visualization module gets the organized data and translates it into the visualization objects, which will be presented graphically in a web browser. We incorporated the Processing.js package (<http://www.processingjs.org>) that is used to create images, animations, and online interactions by using the visual programming language named Processing. It also converts the Processing code into Javascript, thus allowing it to be run by any HTML5-compatible browser, including mobile browsers and current versions of Firefox, Safari, Chrome, Opera, and Internet Explorer. We implemented the visualization module including the spiral algorithm, described in 3.1, in the Processing proprietary language. Behind the scenes, the Processing.js library compiles our code to pure JavaScript.

#### 4 System Setup, User Study, and Evaluation Results

The VisFacet package was set up, integrated, and tested for two environmental systems: HUfind, which is a customized project based on VuFind, and VuFind itself. After it had passed HUJI library's approval regarding the usage patterns and performance tests, it was added to the production version of HUfind in May 2014. The site at <http://hufind.huji.ac.il/> allows thousands of HUfind users to use a graphic search. Note that it is enabled only when using browsers other than Internet Explorer. To learn more about user satisfaction, we wrote a first-stage evaluation questionnaire, which is described in this section. We designed the questionnaire to measure the experiences of users wishing to discover a new research topic to focus on. The questionnaire was sent to two groups of users: (1) an undergraduate student group composed of 21 students, and (2) a group of librarians composed of 10 experienced librarians who are familiar with the HUfind catalogue. The questionnaire itself consists of two search tasks: the **Text search** that

asks the user to search the suggested topics for the keyword *journalists* via the regular text interface and the **Graphic search** that requires the use of our infographic box. Each task requires the exploration of the **Suggested Topics** and the **Region facet** in order to focus on a direction. The search experience is questioned in each path with questions that appear in Table 1. The user is asked to compare both experiences and determine whether she prefers one, the other, or a combination of the two.

The columns in Fig. 4 present the average rate of the answers to Q1, Q2, and Q3, according to the group involved: librarians or students. On average, both groups preferred the Text search over the Graphic search. The answers for the graphic evaluation were mainly *gave some ideas* or *very little*. The librarians, who are used to text searches, were more skeptical about the graphic visualization, while the students liked it more than the librarians. Some of them preferred the graphic visualization, and therefore the standard deviation for their answers was higher. However, 16 students (76%) answered, in Q6, that they prefer having both types of search. This shows that they are open to this direction but want an improved display. We received the following two comments about VisFacet: (1) the search is less user friendly because it requires brushing long words with the mouse to make them fully visible, and (2) the importance of the order of nodes in the spiral form was not intuitively understood. We will relate to both problems in the next version. The number of steps taken by the student group in both types of search were very similar, 1.91 steps on average, with a standard deviation of 0.80-0.84. The number of steps taken by the librarians was 1.8 steps on average, which is slightly smaller than in the other group, and it was the same for both types of search. Since the questionnaire is not a real discovery task, it is difficult to draw conclusions from this result. To gain greater benefit from VisFacet, we plan to provide additional guidance for its use as well as monitor its use and find out how we can improve it.

## 5 Conclusion

We have implemented a subsystem that adds an interactive visualized and faceted search to a modern catalogue. Because our subsystem is a new feature of VuFind, we have explored the existing system to understand its current state and find the appropriate solutions for our design. VisFacet is an initial suggestion for faceted visualization that can be a real impetus for extending the visualization capabilities of modern library catalogues.

## Acknowledgments

We thank Demian Katz, the founder and main developer of the VuFind system, for his ongoing help in enabling the smooth integration of VisFacet into VuFind. We also thank Edith Falk, the chief librarian of the Hebrew University for all her support in this project, and Eli Hayun, the programmer at the Library Authority of HUII for helping with the integration with HUfind. We thanks Nurit Baltiansky, Mally Cohen, and Avi Allalouf for the discussions regarding the questionnaires.

## References

- [1] Emanuel, J.: Usability of the vufind next-generation online catalog. *Information Technology and Libraries* **30** (2011)
- [2] Ramdeen, S., Hemminger, B.M.: A tale of two interfaces: How facets affect the library catalog search. *JASIST* **63** (2012) 702–715
- [3] Merčun, T., Žumer, M.: New generation of catalogues for the new generation of users: A comparison of six library catalogues. *Program* (2008)



- [4] Tennant, R.: Lipstick on a pig. *Library Journal* **130** (2005)
- [5] Garfield, E.: A tribute to s.r. ranganathan, the father of indian library science. *Essays of an Information Scientist* **7** (1984)
- [6] Merčun, T., Žumer, M.: Visualizing for explorations and discovery. In: *Libraries in the Digital Age*. (2010) 104–115
- [7] Kaizer, J., Hodge, A.: Aquabrowser library: Search, discover, refine. *Library Hi Tech News* (2005)
- [8] Shneiderman, B., Plaisant., C., Cohen, M., Jacobs, S. In: *Designing the User Interface: Strategies for Effective Human-Computer Interaction* (5th Edition), Pearson Education (2014)
- [9] Tergan, S.O., Keller, T. In: *Knowledge and Information Visualization, Searching for Synergies*. (2005)
- [10] Thudt, A., Hinrichs, U., Carpendale, S.: The bohemian bookshelf: Supporting serendipitous book discoveries through information visualization. In: *Proceedings of the SIGCHI, ACM* (2012)
- [11] Clarkson, E., Desai, K., Foley, J.: Resultmaps: Visualization for search interfaces. *IEEE Trans. on Visualization and Computer Graphics* (2009)
- [12] Smith, G., Czerwinski, M., Meyers, B., Robbins, D., Robertson, G., Tan, D.S.: Facetmap: A scalable search and browse visualization. *IEEE Transactions on Visualization and Computer Graphics* **15** (2006)
- [13] Wei, F., Liu, S., Song, Y., Pan, S., Zhou, M.X., Qian, W., Shi, L., Tan, L., Zhang, Q.: Tiara: A visual exploratory text analytic system. In: *Proceedings of the 16th ACM SIGKDD Conference, ACM* (2010)
- [14] Thai, V., Rouille, P.Y., Handschuh, S.: Visual abstraction and ordering in faceted browsing of text collections. *ACM Trans. Intell. Syst. Techn.* (2012)
- [15] Vogel H.: A better way to construct the sunflower head. *Mathematical Biosciences* **44** (1979) 179–189
- [16] spiral, F.: ([http://en.wikipedia.org/wiki/fermat's\\_spiral](http://en.wikipedia.org/wiki/fermat's_spiral))