

# Revealing Relationships in Search Engine Results

Cláudio Max. Zaina

M. Cecília C. Baranauskas

IC - Computing Institute

University of Campinas – UNICAMP

Campinas – SP – Brazil

{claudio.zaina, cecilia}@ic.unicamp.br

## ABSTRACT

The amount of information available in the Internet is so vast that finding the desired information in such an unstructured repository easily becomes a tedious task. Graphical cluster-based representations of results from search engines shift the user's mental load from slower thought-intensive processes of reading information got from linear lists of results to faster perceptual processes such as pattern recognition in a visual display. In this paper we investigate the subject by presenting the design proposal of a new system that uses concepts of Information Visualization to help the user of Internet search engines to identify the group of documents that are related to the needed information by examining or browsing documents in the group. Some results of preliminary usability tests for the system are also provided shedding some light to the subject.

## Keywords

Search engine results, Information Visualization, query results visualization, World Wide Web.

## INTRODUCTION

Internet is quickly becoming the *de facto* information source nowadays. Almost any subject has some explanation or reference in the Internet. The amount of information is so vast that a problem arises: How to find the desired information in such an unstructured repository? The most common tools used to cope with this problem are the search engines. Search engines are systems in the Web that, given a set of keywords entered by the users, return a result page. In this result page, the user gets a list of possibly useful documents, each one with a partial short description related to the document and a web address to serve as a link to the document. The list is supposedly ranked by the relevance of the documents in relation to the given keyword set. Unfortunately, due to the several meanings of words, some unrelated documents are listed between the ones that best matches the user's interests. Searching the list in order to pick up the potentially useful

documents easily becomes a tedious task. In this work we proposed a system design which applied some concepts of Information Visualization in order to make it easier for the user to identify the relationship between the documents of the results list. Through the system, the user is able to visually identify the documents that have somewhat similar contents and could help to solve the user's information need. This system application, named ReVEL (*Representação Visual de Elementos de Lista* – Visual Representation of List Elements), was submitted to preliminary tests with potential users in order to evaluate its usability.

In this paper we present ReVEL, its design features and properties; some results of usability investigation is also discussed. In the next section we review some literature work related to information visualization for search engines. In third section we describe ReVEL, discussing some of its design features. In the forth section we present some results of the usability tests and in the fifth section we conclude the work discussing some possibilities of future work.

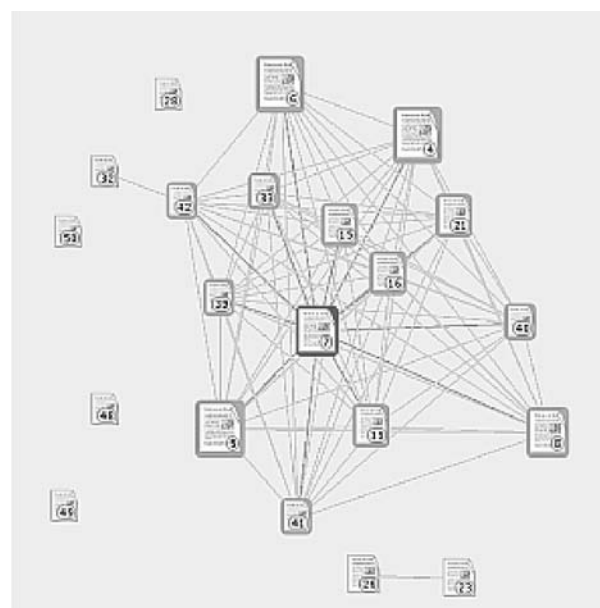
## RELATED WORK

The information presented in the lists resulting from search engines processing usually are textual and thus requires the user to sequentially read each one of the items to the end of the list, seeking for the most promising entries. The processing on the human side – reading – is detailed, serial, it demands high cognitive capacities and it takes time. But the information can be presented in a different visual way, one that uses some less demanding processing, which is parallel to other capacities, is fast and to a certain extent unconscious. This processing involves information that was coded as patterns, colors, widths, length, line orientations, position etc [3, p. 25]. The utility of visualization techniques derives in large part from their ability to reduce mental workload. Previous results from literature suggests that such reductions are dependent upon an appropriate mapping among the interface, the task and the user [12].

Different approaches have been made to represent an information search result in a way that the user could identify the desired information helped by the information representation. Some approaches categorize the results, identifying similar subject items and presenting them under the same category. An example of this type of solution is illustrated by the system presented in Chen [5]. In that



(a) Search engine results list



(b) ReVEL graph

Figure 1 – Different visualizations to present a query result

controlled search experiment, the interface based on categories was found to be superior to the interface based on lists of results in both subjective measures – open-ended questions about aspects of the interfaces – and objective measures – search time. In that case, if the query is too restrictive the items of the results list would fall entirely within one category and the outcome would be the equivalent to the results list returned by the search engine.

Once some approaches are based on different hypothesis and focus on different properties, they render different types of visualizations.

There are approaches that rely on Information Visualization (IV) concepts and they bind visual attributes to information attributes to allow the important information about the retrieved records to be represented visually [10].

*Results Wheel* [8] is a system that provides the user with information about the relevance of each keyword in a multi-keyword query. The authors concluded that, in a multi-term query, the user is aware that some keywords are more important than others to establish the ranking of search engine results for overall relevance. Unfortunately the ordinary user doesn't seem to use keywords enough to make *Results Wheel* as useful as it could be. [14] reports that 72.4% of the queries had 2 or less keywords. Amazingly, 20.6% of the queries had no term at all. An experiment evaluating user utilization of the *Infoseek* search engine system [9] reports similar results, with average query length of 2.2 words.

*CardVis* [10] presents HyperText Markup Language (HTML) documents retrieved from queries in specific databases as disconnected graphs, each one in a card. The vertices of a graph represent the pages and the edges

represents the links between the pages. Cards are then organized in a way that resembles playing cards: one highlighted card is kept at the top while relevant information in the others are apparent. The separation in cards, however, tends to hide the amount of documents there are in each card, once cards are overlapped with only a keyword being visible. There is another problem, if there are too many related documents, the visualization may get cluttered.

Some efforts are also being made to establish models for search results visualization [1] and to evaluate the effectiveness of search results visualization [14][12]. Our approach, described in the next section, defines an interface layer for the linear presentation of results got from the usual search engines. This interface is based on a graphical cluster-based representation, where information is presented graphically according to some relationships the documents may have regarding the information searched.

## REVEL

ReVEL is a system designed to present to the user results of a search engine query in a graphical way. This system depicts the items in the results list of a search engine query, as document icons connected as a net. This net is a graph where the vertices are the documents in the results list and the edges are the connections that the each document may have one another. Figure 1 depicts in (a) the results list returned by Google and in (b) the corresponding graph representation, both for the same query "search engine result list visualization". The connections in the graph are defined as the similarities calculated between each two of the documents of the results list, based on their text. By examining the graph, the user is able to identify

the groups of similar documents that constitute the results list. The representation also distinguishes the documents by size (how big it is) and by relationship (how connected the documents are). It is possible to adjust the automatic disposition of the document icons in the screen by dragging them in the visualization area. The user can browse the documents by double-clicking their icons. Each time an icon is selected, its short description is shown in a proper window, present in the interface. A table with details of the items of the results list is also available if the user wants to know some particular detail: whether the document has already been downloaded or analyzed, what is its address (URL) etc. The user may also examine the original results list, if he or she wants to. The search session can be remade any moment, just by entering different keywords in the search text field. There is also the possibility of marking some documents to be kept for the next search session. This way the user can evaluate how the documents of the new search session relate to the retained ones.

Graphical cluster-based representations shift the user's mental load from slower thought-intensive processes such

as reading to faster perceptual processes such as pattern recognition in a visual display [4].

### System Overview

The system is composed by four parts that operate independently. The general structure of the system is illustrated in Figure 2.

#### Search Manager

The Search Manager receives the keyword set and the specification of the desired search engine from the Interface, provided by the user. A query is assembled and submitted to the search engine requesting fifty entries, which is equivalent to five result pages (ordinary result page, with ten items, plus four more "next page"). Literature has reported that 63.7% of the sessions analyzed – same user query within a small range of time, (generally 5 minutes) – consisted of only one request: this means that only one query was entered and only one result screen was examined [13]. Once we request five times more, we are trying to provide the user with greater possibility of finding the desired information.

Each item of the results list is parsed and put in a document list in the same rank position. This document list is made available to the system and is used by the Download Manager and the Interface.

#### Download Manager

This manager sequentially downloads from the Internet the first fifty kilobytes of the documents in the results list. Fifty kilobytes was found to be a good enough size so that the similarity calculated is reliable and has a short download time. The Download Manager verifies if there are internal frames and, if so, it gets them too. If the URL refers to a document that is in a format different from HTML, and a HTML version is available, the Manager will get the HTML version, otherwise it'll ignore the entry. The downloaded document is then prepared to have its similarity calculated. All HTML code, frequent words – also known as "stop words" – and the trailing 's' from the plural words are skipped from the original document. Once these transformations are done, the new text document is made available to the system and the Computation Manager is notified. Literature has shown that about 20% of the available memory of the computer is reserved to cache the highest ranked documents from one query to the next. This way downloads are avoided if the query is made on the same subject, changing minimally the keywords. Documents are replaced in this cache by their age and frequency of use.

#### Computation Manager

To calculate the similarity between two documents, the Computation Manager uses the *idf*-weighted cosine coefficient as described in [11][2], often referred to as  $tf \cdot idf$ . The text documents are interpreted as vectors in an  $n$ -dimensional space, where  $n$  is the number of unique words in the collection of text documents. The coordinates of the vector of a given document are its term frequencies (*tf*). The *idf* (inverse document frequency) modification weights words according to their capacity to discriminate

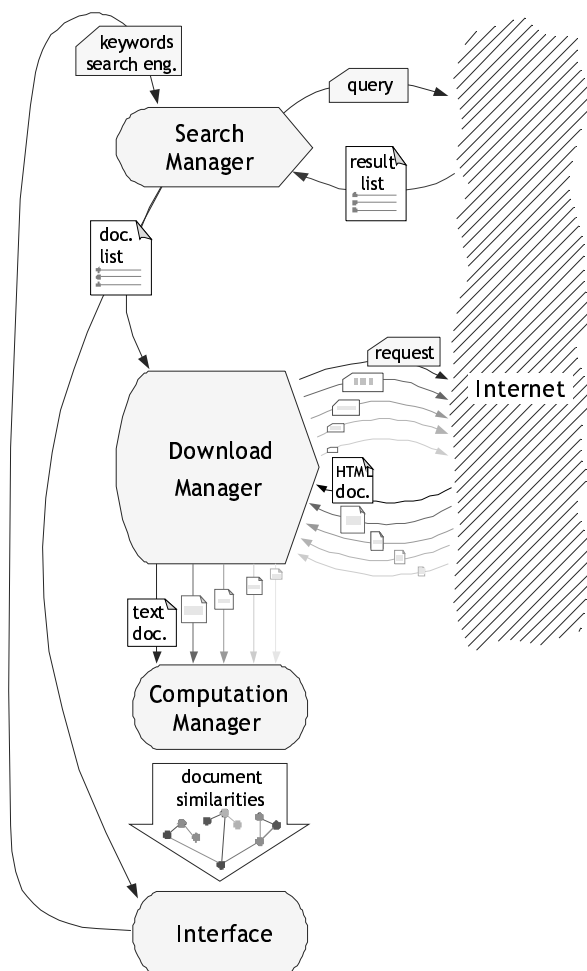


Figure 2 – ReVEL Structure

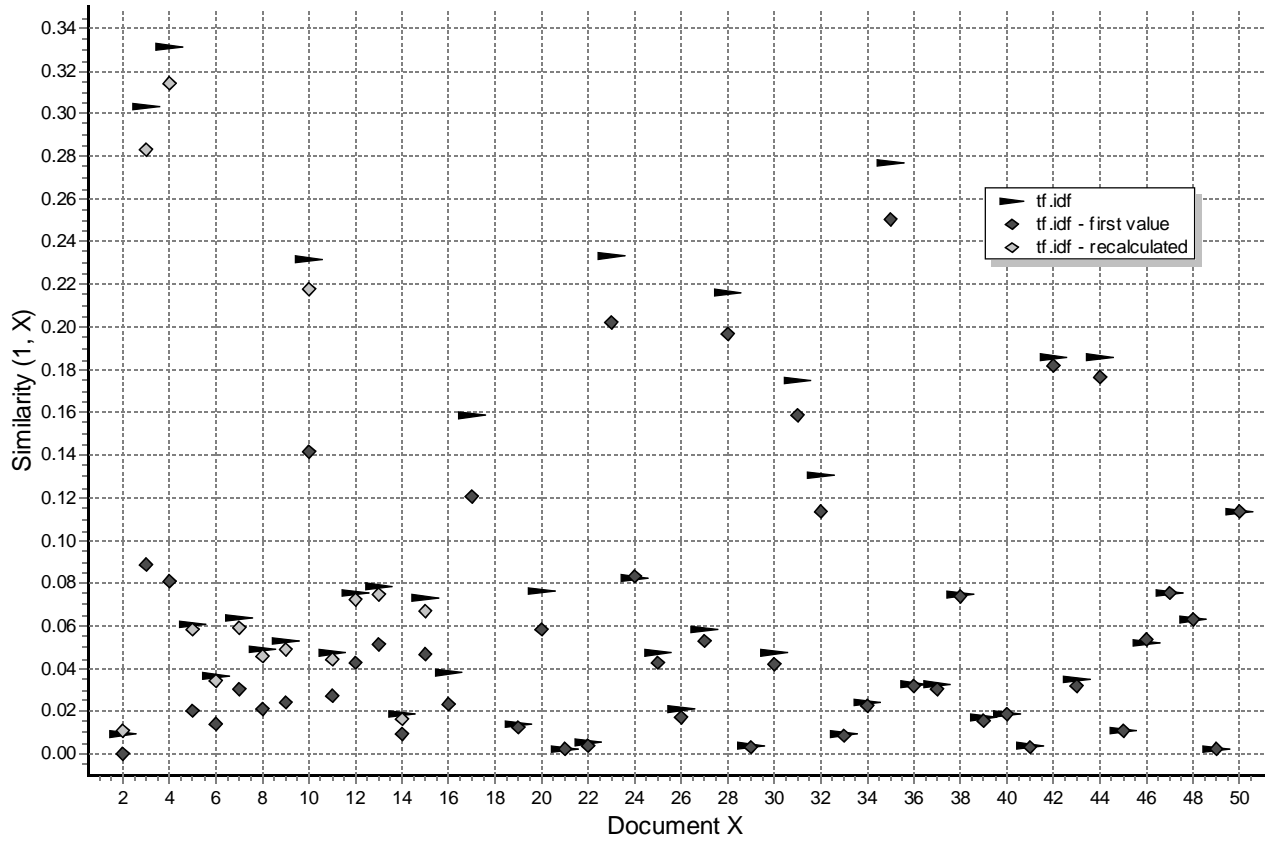


Figure 3 – *tf.idf* Similarity values for the first document in a search for “medusa”

documents. Words that appear in few documents of the collection have more discriminating capacity than those that are present in many of them. This capacity appears in the weighting factor  $idf(w)$  for the term  $w$ :

$$idf(w) = \log_{10} (N / df(w))$$

where  $N$  is the number of documents in the collection and  $df(w)$  is the number of documents in the collection where the term  $w$  appears. The similarity between document  $a$  and document  $b$  then is calculated as [11]:

$$\text{sim}(a, b) = \frac{\sum_{w=1}^n tf_a(w) \cdot tf_b(w) \cdot idf(w)}{\sqrt{\sum_{w=1}^n tf_a^2(w)} \cdot \sqrt{\sum_{w=1}^n tf_b^2(w)}}$$

A problem then arises: this formula requires that all the documents of the collection be available when the similarity between any two documents is calculated. But documents are being continually downloaded, the whole collection isn't already available. So we compute the similarity with the available documents and after forty seconds or after thirty documents are retrieved – what happens first – the similarities for the first fifteen documents are recalculated. As one can see in Figure 3, the first estimate of the real *tf.idf* is poor, but the second is

very close to the real value. As expected, the closer to the last document, the better gets the first estimate of the *tf.idf*.

The data structure with the similarities is shared with the Interface, that is responsible for presenting the relationships.

#### Interface

The Interface is responsible for the user interface, where the visualization takes place. In Figure 4 one can see an example session, with the keyword “medusa” – and a selected document.

The different areas of the user interface are letter marked in Figure 4 and their description follows:

- A** *Control*: Here one finds the text fields used to insert the keywords. There are also some buttons: to start and to cancel the search, to show the results list (*exibir como lista* – show list) and to stop the automatic organization of the document icons in the display area (*auto-organizar ícones* – icons self-organization).
- B** *Display*: Here the documents are depicted as document icons. Their size is proportional to their rank in the results list: the higher the rank, the bigger the icon. The icons are composed by an image and a number. The number is the rank of the document in the results list and the image reflects the document status. A list of the possible status is presented in Figure 5.

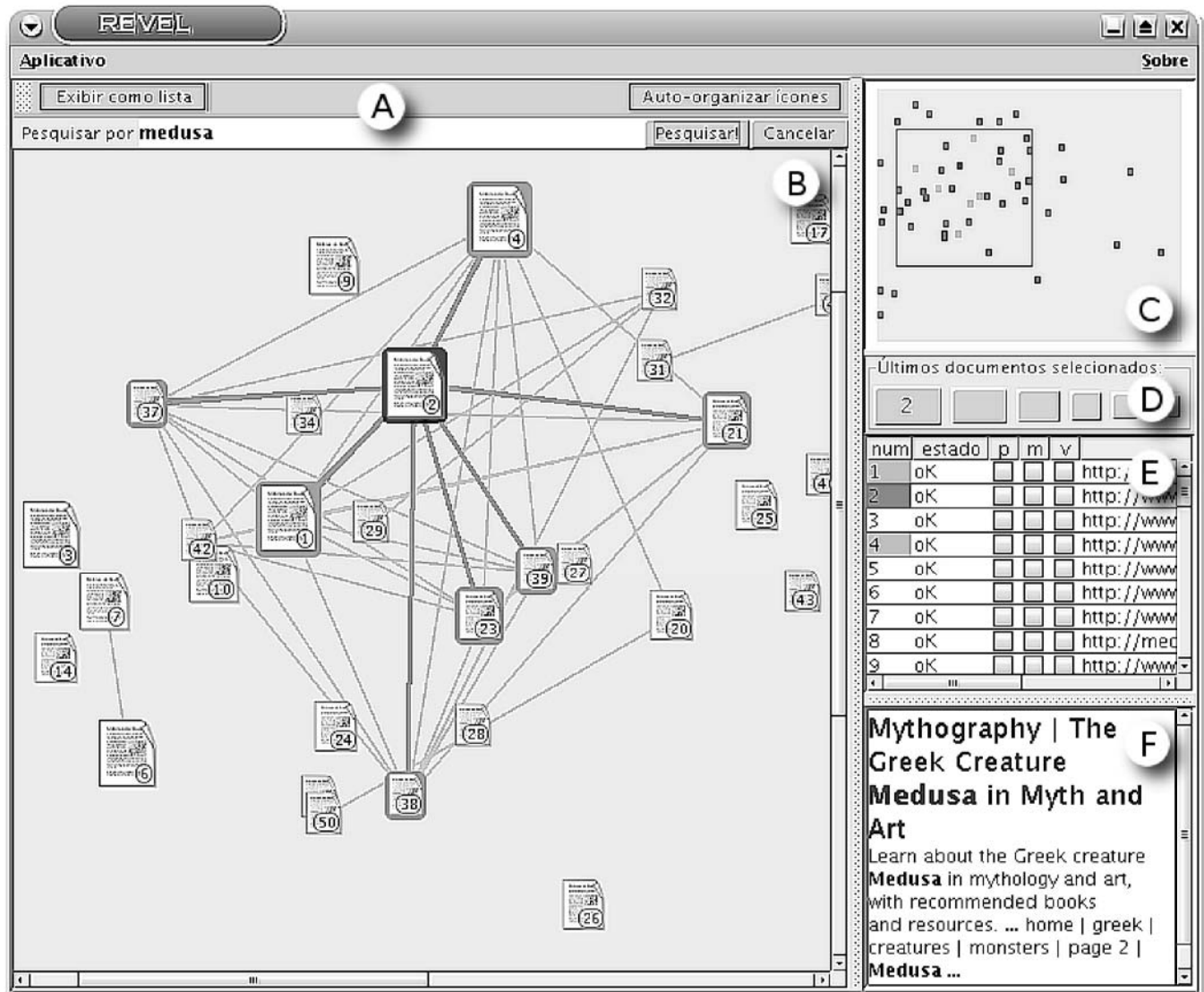


Figure 4 – Screen shot of the ReVEL system showing the query result for “medusa”

A clicked document gets highlighted in red, its short description appears in the Short Description Area (F) and the documents related to it have their connections highlighted in orange and are highlighted in orange themselves. There is a background thread that tries to adjust the position of the documents in a way that the size of the edges between them is as close as possible to the similarity between these documents, in a way that documents with higher similarity are expected to be drawn closer. This thread uses a force-directed method to position the icons in the Display Area [7]. This movement is not continuous: the icons are slowly repositioned in their new places and they stop, while a new turn of background calculation starts.

The user can move the icons positioning them him/herself. The user can pin the icon down by right-clicking it. This prevents its automatic positioning to take place.

If a document cannot be downloaded, it is marked as

in error and is pinned down in the leftmost part of top of the Display area.

- C** *Overview*: is a 1:6 scale image of the Display area. Here the user can see all the documents of the results list at once. If some document is selected in the Display area, it is highlighted here and its similar documents are highlighted as well.
- D** *Most Recently Selected Documents (MRSD)*: it is common, during a search session, to select (click) several documents for inspection. This sequence of six buttons aims to help the user to remember the last selected documents. The button/label size is also an indicative of its closeness in time: the bigger the buttons, more recently it was selected. In Figure 4, only one document had been selected until that moment – the second – so the other buttons were not yet functional.
- E** *General Information Table*: this table summarizes the information for each document of the results list. Here

the user can see the document rank (*num.*), whether it is selected or not (the background color of the number matches the document highlight in the Display Area), its status (*estado* – requested, downloading, in analysis, ok) and the URL. There are three checkboxes that allow the user to set/change behaviors of the document icons: **p** to pin the icon down, **m** (*manter*) to keep the document for the next search session and **v** (visited) to set as visited and call the browser for the current document or to erase this status by clearing the checkbox.

- F** *Short Description*: this area displays the short description of the document that appears in the results list of the search engine.

The Interface components have a tight coupling in order to help the user interaction by providing him/her immediate multiple feedback for an interaction. Tight coupling here means that interacting with any of the Interface components immediately updates all other related components. For instance, clicking on a document icon in the Display Area also highlights the references for this document in the Table and Overview areas besides updating information at the Short Description and MRSD areas. It is also possible to achieve the same result by other ways as, for instance, selecting the document by its number in the Table or by its text in the results list.

It is worth noticing that Information Visualization has been on everyone’s desktop for years in components of graphical interfaces [6].

Every interface component has a *tooltip* – a hover text. Some *tooltips* have information updated every moment. For instance, if the user hovers the mouse over the status *textfield* in the Table for the 8<sup>th</sup> document, possible messages could be ‘8<sup>th</sup> document was requested from Internet’, ‘8<sup>th</sup> document is being downloaded now’ etc depending on the status of this document in the moment.

## PRELIMINARY EVALUATION RESULTS

Initial usability tests were carried out with six graduate students from different subject areas. Their experiences with search engine ranged from novices to experts. Two students were from Geography, one from Civil Engineering and the three remaining from Computer Science. There was an equal number of men and women. The test had three tasks, supposedly of crescent difficulty. The first task was to find a URL where one could get the name of the Greek hero who killed Medusa. As the second task, the user had to discover who was the author of some given verses and in which work they appear. The third task required the user to find a special numeric value, the Brazilian gross national product in 2003. The usability test was tape recorded. When the last task was complete, the user was asked to answer a set of open questions about the system.

Some interesting points were revealed about the understanding the users made of some interface components.

Icons were identified as the documents retrieved by the query but their numbering were not immediately recognized by everyone as the rank in the results list. A Geography student even supposed it represented document chronological order. Similar misunderstanding took place with the edges connecting icons: a student supposed they meant “the continuation” of the document, as if they represented some kind of “next page” even though the edges were not represented as directed vectors. Half of the students – the Computer Science ones – reported the automatic icon positioning movement as disturbing. One of them added that one gets the impression that the system “is not ready, is still working”.

The most easily recognizable element was the Overview, probably because it is a common feature in multiple context interfaces. Some interacted as expected, clicking in the new desired *viewport*. Others tried to grab and drag the rectangle that indicates the region actually shown in the Display.






Icon	Status	Description
	requested	The document was requested to be downloaded from its URL.
	downloading	The document is being downloaded.
	in analysis	The similarity calculations are being performed for this document.
	ok	The process is complete. The document was processed and its similarities are in use.
	error	This document could not be retrieved by some problem outside ReVEL.

Figure 5 – Icon images for document status

The *tooltips* were found to be very useful for three of the students. One of them explored the *tooltip* for every element. One understood the concept of the Most Recently Selected Documents by using the *tooltip*. Nevertheless, one expert user student simply ignored them all. When asked about it, he said that “they generally say nothing new”.

One of the students happened to be color-blind. He complained that there are too many red hue items in the interface and this tires color-blind users. He suggested blue instead of red.

One user reported being easier to find information using the system and one said that, in spite of the Table being of a great help, “the clustering visualization doesn’t help”.

Surprisingly, the Table was very utilized by all the students. The three Computer Science users expanded the Table to its maximum and complained it couldn’t go even further. One of them said that “It was nice be able to see all the addresses” (URLs). Generally the Table was the element the students used the most, as if the Display was a visualization helper for the Table and not the other way round (at least by construction).

One student liked the ability to see the original results list and other expressed his satisfaction with how fast one can request the browser to open a document. It happens because it is enough to check the *visited* (V) checkbox in the table and a new browser window is opened with the document. That requires just a click.

The documents in error were supposed, by one student, to be the once clicked (selected) documents. Later she correctly noticed that that they were the documents marked with “error” in the Table.

A curious behavior surfaced separating the Geography users from the others: when browsing a page, they used to read it to the end. The other students took glimpses of the pages while the Geography ones read them as a whole.

Once the download process occurs in the background and the results list is (almost) immediately available to the user, not a single student complained about download time.

All students correctly finished tasks one and three but half gave up task number two, the one that wasn’t supposed to be the most difficult. That occurred probably because the verses were very frequently quoted and often misquoted, confusing the students. The verses are: “*Que é a vida? Um frenesi./ Que é a vida? Uma ilusão, / uma sombra, uma ficção; / o maior bem é tristonho, / porque toda a vida é sonho / e os sonhos, sonhos são.*” by Calderón de la Barca in “A Vida é Sonho”.

## CONCLUSIONS AND FUTURE WORK

Information Visualization has been used by different authors to provide better solutions to the task of getting information from search engines.

In this work we investigated the subject by presenting a system which makes use of concepts of Information Visualization in order to help users in their search for information. Usability tests were applied and users feed-

back suggests that the way to organize the graph icons could be improved, especially regarding the effect of movement. A possible solution would be to limit the icon movement, and this amount could be proportional to how long the document is ready. That would make the whole graph more stable and only the just completed documents would somehow move. The users also pointed out that the Table should occupy a bigger portion of the Interface space and that it is interesting to provide multiple tools in the same interface to help the user locate the needed information.

Although visual representations seem to be a promising approach to cope with the problem of searching information in a vast space, it remains an important issue to be discussed especially regarding the impact these solutions may have on interaction through the user interfaces of the tools. The next steps in this work involve an in-depth study regarding the proposed visual representation considering different categories of users and different interface layouts for ReVEL.

## REFERENCES

1. Alonso, O.; Baeza-Yates, R.; A Model and Software Architecture for Search Results Visualization on the Web. *Proceedings of the Seventh International Symposium on String Processing Information Retrieval (SPIRE'00)*. 2000. pp. 8-16.
2. Baeza-Yates, R.; Ribeiro-Neto, B. (1999). *Modern Information Retrieval*. ACM Press / Addison Wesley.
3. Card, S. K.; Mackinlay, J. D.; Shneiderman, B.; Readings in Information Visualization – Using Vision to Think. Morgan Kaufmann Publishers, Inc. 1999.
4. Carey, M.; Kriwaczek, F.; Rüger, S. M.; A Visualization Interface for Document Searching and Browsing. *Proceedings of the New Paradigms in Information Visualization and Manipulation NPVIM(00)*. 2000.
5. Chen, H.; Dumais, S.; Bringing Order to the Web: Automatically Categorizing Search Results. *CHI Letters*, vol. 2, issue 1.2000. pp. 145-150.
6. Dantzhich, M.; Visualization Is a State of Mind. *Proceedings of the 1997 workshop on New Paradigms in Information Visualization and Manipulation (NPVIM'97)*. 1997. pp. 29-31.
7. Fruchterman, T. M. J.; Reingold, E. M.; Graph Drawing by Force-directed Placement. *Software – Practice and Experience*, vol 21 (1 1). 1991. pp. 1129-1164.
8. Grewal, R. S.; Jackson, M.; Burden, P.; Wallis, J.; A Novel Interface for Representing Search-Engine Results. *IEEE Informatics Colloquium*. November, 1999. pp. 7/1-7/10.
9. Kirsch, S.; Infoseek's Experiences Searching the Internet. *ACM SIGIR Forum*, Vol 32, issue 2, 1998. pp. 3-7.
10. Mukherjea, S.; Hirata, K.; Hara, Y.; Visualizing the Results of Multimedia Web Search Engines. *Proceedings of the 1996 IEEE Symposium on*

- Information Visualization* (INFOVIS '96). 1996. pp. 64-65, 122.
11. Schultz, J. M.; Liberman, M.; Topic Detection and Tracking using idf-Weighted Cosine Coefficient. *Proceedings of the DARPA Broadcast News Workshop*, pp. 189-192.
  12. Sebrechts, M. M.; Vasilakis, J.; Miller, M. S.; Cugini, J. V.; Laskowski, S. J.; Visualization of Search Results: A Comparative Evaluation of Text, 2D and 3D Interfaces. *Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*(SIGIR '99). 1999. pp. 3-10.
  13. Silverstein, C.; Henzinger, M.; Marais, H.; Moricz, M.; Analysis of a Very Large Web Search Engine Query Log. *ACM SIGIR Forum*. Vol. 33 issue 1, 1999. pp. 6-12.
  14. Veerasamy, A.; Heikes, R.; Effectiveness of a Graphical Display of Retrieval Results. *Proceedings of the 20th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (SIGIR '97). 1997. pp. 236-245