First publ. in: Visualization and Data Analysis, Proceedings of SPIE, Vol. 5669 (2005), pp. 327-338

The CircleSegmentView - A Visualization for Query Preview and Visual Filtering

Peter Klein[a] and Harald Reiterer[a]

[a] University of Konstanz, Department of Computer and Information Science, Universitätsstr. 10, 78467 Konstanz, Germany

ABSTRACT

Users of Information Retrieval systems have often been the target group of Human-Computer Interaction researchers. A lot of effort has been spent inventing new forms of visualizations to support the information seeking process.[1] Information Retrieval and Information Visualization are tight coupled fields of research.[2] Together with psychology (which answers questions like 'how' do users search) and usability engineering (answering questions like 'what' do user expect from user interfaces and their behaviour) the research on improving information seeking systems goes on. This paper will concentrate on a meta-data driven, user-centered approach for the query formulation stage. In contrast to the intense research on result-set visualizations we will focus on the development of a visualization which supports human search behaviour at the query stage. Additionally this visualization proved that it can compete with other visualizations like the scatter-plot as a visual filter in the result-set presentation.

Keywords: Query Preview, Meta-data, User-centered, Human Factors, Information Visualization

1. INTRODUCTION

In the field of Information Visualization and Visual Data Mining a lot of effort has been spent on inventing new visualizations that utilize the human visual capabilities.[1] Functions of Information Visualization and Retrieval Systems include browsing, searching, refinement and presentation. However, increasingly users demand more features and tools to help them interactively mine data, generate patterns and conduct analysis on data.[3] We address some of these features with our newly created visualization called 'CircleSegmentView' (CSV). To build a visualization that supports the human search behaviour in the information retrieval process it is essential to use meta-data that describes the raw data. Following Marchionini,[4] this kind of information is refered as 'descriptive meta-data', meta-data that is external to the meaning of the document and pertains more to how it was created. Another type of meta-data characterized the subject matter that can be found within the document's content. This type is refered as 'semantic meta-data'. It can increasingly be found in closed environments like geo-information sytems (GIS), medicine databases or library cataloques where hierarchical taxonomies of terms (ontologies) describe vertain knowldege topics. A third kind of meta-data are attributes that specify the document repository under certain points of view like the amount of available documents with a specific feature. This data is usually computed on the database and allows (when offered to the user) filtering and preview of the power of the estimated result set. Visualizations usually visualize only this 'meta'-part; e.g. the placement of a data dot in a scatter-plot depends on two (three if it is a 3D scatter-plot) meta-data values. So the user can only act on the data knowing that data D has value X=x and Y=y. In this paper we will assume, that the meta-data generation and analysis is already concluded (as described in the visualization reference model[5]) and we focus on the mapping of the meta-data to the visualization. For this meta-data driven, user-centered approach we adopted the following design focus:

1: Build a visualization for expert users that is capable of mapping a lot of different meta-data and supports the user's way of searching. We define the expert user as frequent user, trained on the interface who performs information requests on a regular bases (like for example an information broker).

2: The visualization can compete with traditional form-based queries.

3: The visualization can compete with other visualizations like a scatter-plot as a visual filter and supports data-mining qualities (pattern recognition).

The remainder of this paper is organized as follows. After this introduction, section 2 will give a short overview of the systems architecture. The data transformation, meta-data generation and the configuration tools are described in section 3. Main theses, system description and introduction to the different possibilities of the CSV will be presented in section 4. The user studies will be described in section 5. The related work that has influenced our own will be presented in section 6. Conclusions and outlook are given in section 7.

2. SYSTEM CONCEPTION

2.1. Architecture

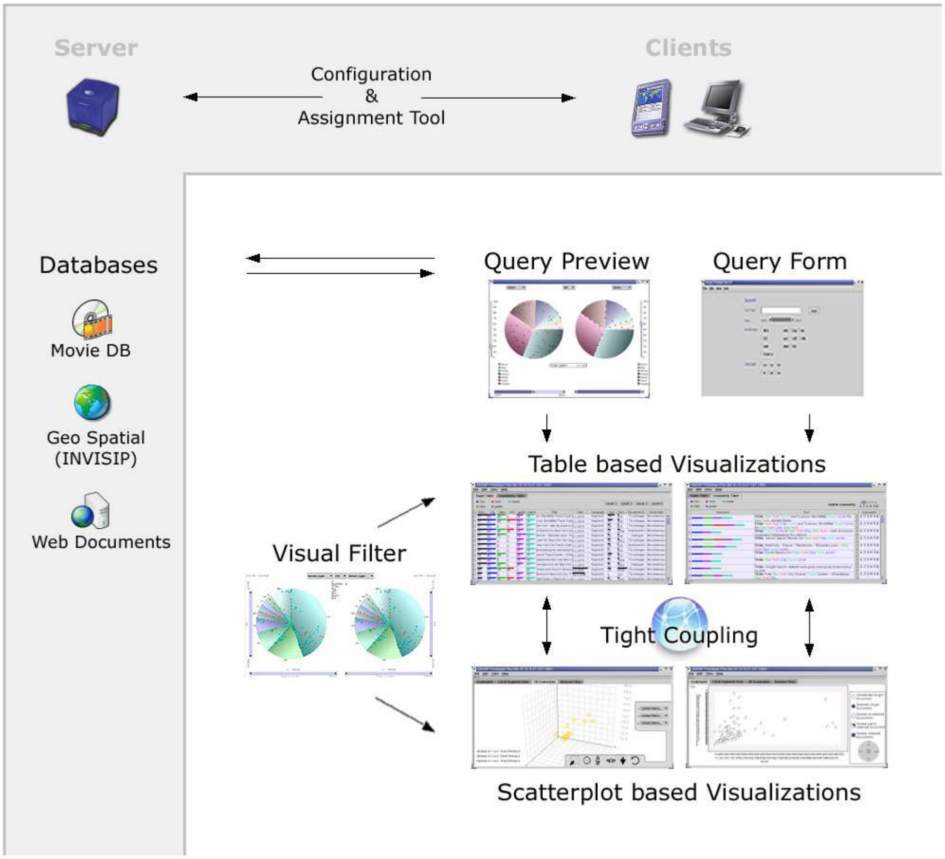
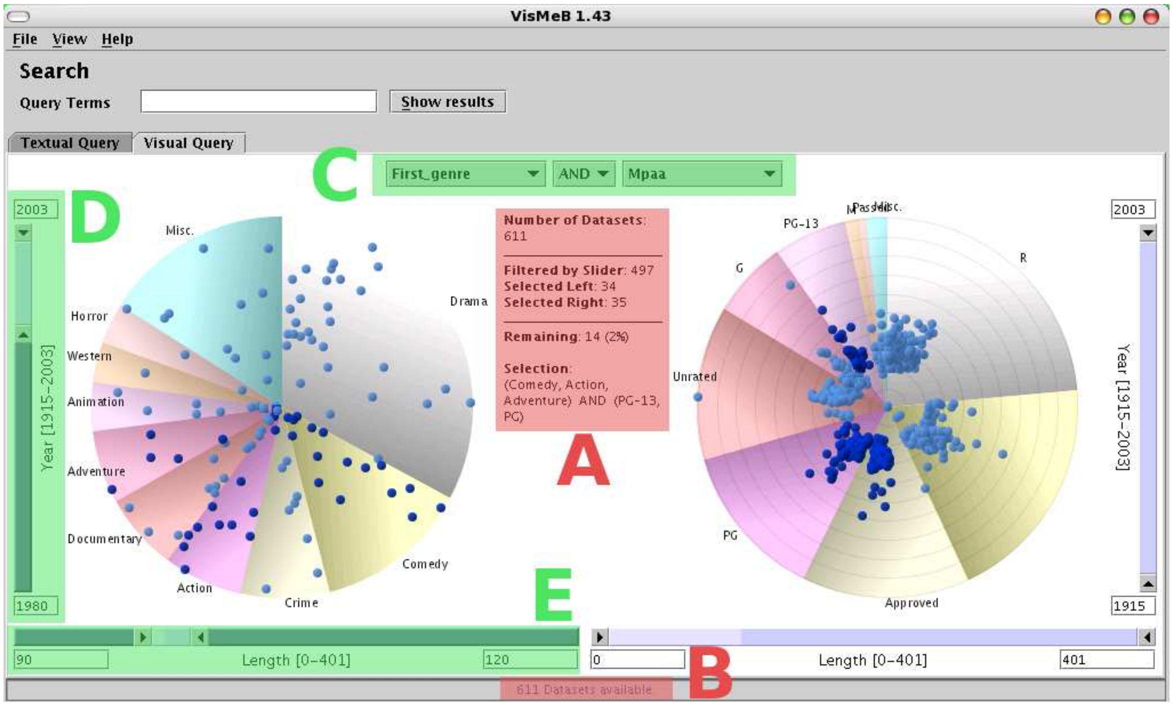


Figure 1. Architecture of VisMeB

The CircleSegmentView is part of a larger visualization framework called VisMeB*. VisMeB as the name suggests is not only so much around searching with keywords, but more over around rapid browsing and reacting to large collections of data. VisMeB is based on Client/Server architecture (see figure 1) and implemented in Java$^{\mathrm{TM}}$. It can be connected to a wide variety of different data-sources (virtually to any database that offers an JDBC driver). At the moment we use VisMeB with the following information spaces:

- A geo-meta-data base (ISO 19115 for geo-meta-data) in the INVISIP project.
- A web database containing the results of our topic specific web crawls.

*Visual Meta-data Browser was developed in the EC funded project INVISIP: The project INVISIP was part of the EU Information Society Technologies Programme and facilitates processes and parties of location site planning: www.invisip.de



Number of Datasets: 611

Filtered by Slider: 497
Selected Left: 34
Selected Right: 35

Remaining: 14 (2%)

Selection:
(Comedy, Action, Adventure) AND (PG-13, PG)

611 Datasets available

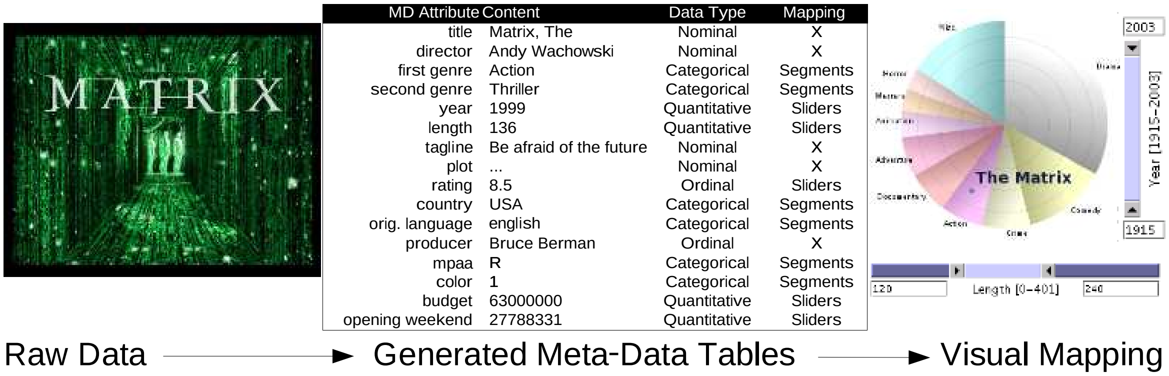
Figure 2. CSV - user interface components: A and B - preview areas with information about the selected and filtered data; C - drop down boxes which contain the different categories for each pie menu and the boolean operator to connect them; D and E - the Alpha-sliders to manipulate the ranges of the numeric meta-data.

- A movie database and
- the university of Konstanz campus media library (that provides media other than books and journals, e.g. DVD, microfiche, ...)

Besides the raw data and its derived meta-data the server stores and manages the different sessions, configurations and the mapping of table columns to visualizations. The result visualizations on the client contain table based visualizations (so called SuperTable variants: LevelTable and GranularityTable) as well as scatter-plots (2D and 3D variants).[6] The visualizations are tightly coupled with each other to support the advantages of each visualization concerning its abilities to rapidly filter/select or zoom of documents/information spaces. This suite is completed by a visual configuration and assignment tool which empowers the user to change rapidly the data source, the layout of the visualizations, the mapping of the data columns to various visualizations inside the tables as well as the initial views or the filter dialogs.
In this framework the CSV plays two important roles. On the first hand it was designed as a visualization that supports the user at the query formulation stage. It should combine a powerful interface for expert users with query preview / dynamic query capabilities. During the development it became clear, that this visualization was also very effective in the result set visualization as visual filter tool.

2.2. User Interface and Visualizations

The CSV was planned to introduce the user to the visual information seeking process. It should offer an overview of the information space and also some general features of the collection before the user can specify a query. Besides this information the visualization should enable to limit and filter the result set to manageable sizes. The evaluation results[7] on search behaviour for the INVISIP domain showed a clear tendency towards filtering out irrelevant data based on some meta-data attributes as early as possible. Theses findings can be backed up by



MD Attribute	Content	Data Type	Mapping
title	Matrix, The	Nominal	X
director	Andy Wachowski	Nominal	X
first genre	Action	Categorical	Segments
second genre	Thriller	Categorical	Segments
year	1999	Quantitative	Sliders
length	136	Quantitative	Sliders
tagline	Be afraid of the future	Nominal	X
plot	...	Nominal	X
rating	8.5	Ordinal	Sliders
country	USA	Categorical	Segments
orig. language	english	Categorical	Segments
producer	Bruce Berman	Ordinal	X
mpaa	R	Categorical	Segments
color	1	Categorical	Segments
budget	63000000	Quantitative	Sliders
opening weekend	27788331	Quantitative	Sliders

Figure 3. Meta-data extraction/generation and visual mapping.

the research results of Chen and Dumais,[8] who found that users can handle search results much more efficiently when presented in a categorized instead of a (common) list based view.
The idea of Query Previews[9] where the user has a visualization of the data and a set of controls (such as sliders), by which subsets of the data table can be selected was adapted. The CSV (see figure 2 for the actual user interface), which emphasizes the distribution of the data also gives continuous feedback about the size of the result set and works as a dynamic query interface. Additionally the users benefit from this because of the prevention of zero-hit or mega-hit results.
For categorical meta-data the effectiveness of control depends on the data type and on the cardinality of the attributes. So it should be carefully considered which data from our databases matches the requirements of the query preview.

The visual representation of the categorized information space is solved by pie charts. The whole circle represents the information space. The different segments show the distribution of the data concerning one type of meta-data. The extent of each segment hints at the size of this categorical attribute. The segments 'start' at top (twelve o'clock position) with the largest segment and continue clockwise in descending order. The available categories can be selected through a drop-down list which is placed above the circle (see figure 2 C). For lucid reasons the number of wedges is limited to ten (nine plus the remaining wedges that are summarized in a tenth slice labeled 'Misc'). The document representing dots are placed inside the segment using two other numeric meta-data. One meta-data is mapped on the radius (e.g. rank or popularity) and another one on the angle (e.g. release date, or length). The position is rendered, so that greater values lay on the border (for the radius) and at highest angle (counter-clockwise for the angle). This setting was chosen to offer more space for relevant data (high values were considered more important in our scenarios). Concerning the optimal utilization of available space space the user has also the possibility to chose different placement methods. The Alphaslider context menu offers three different placement algorithms: linear, square root and logarithmic. The effects can be seen in Figure 4; all three pie menus show the same information space. The area representing the intersection of both high values is supported by a colour gradient to gain easy access to that information (e.g. the 'newer' and 'larger' documents lay in the brighter drawn area if the chosen meta-data would be 'release year' and 'size in bytes'). The selected sets of documents in both pie charts can be linked by a boolean operator. At the moment we support the boolean 'AND' and 'OR' operator. To influence the ranges of the numeric meta-data needed to place the data-dots we use two Alpha-Sliders,[10] see figure 2 D and E. The first one (vertically orientated) specifies a range for the angle (e.g. 'year' from 1980..2003 as can be seen in figure 2), the second (horizontally orientated) a range for the radius (e.g. 'length' from 90..120). By clicking on the Alpha-Sliders a pop-up window appears and lets us change the assignment for the radius or the angle. The minimum and maximum values for the sliders are derived directly from the database. Instead of using the sliders, the user also has the possibility to enter the border values in two text fields placed at the end of the sliders. These text fields always contain the actual border values.

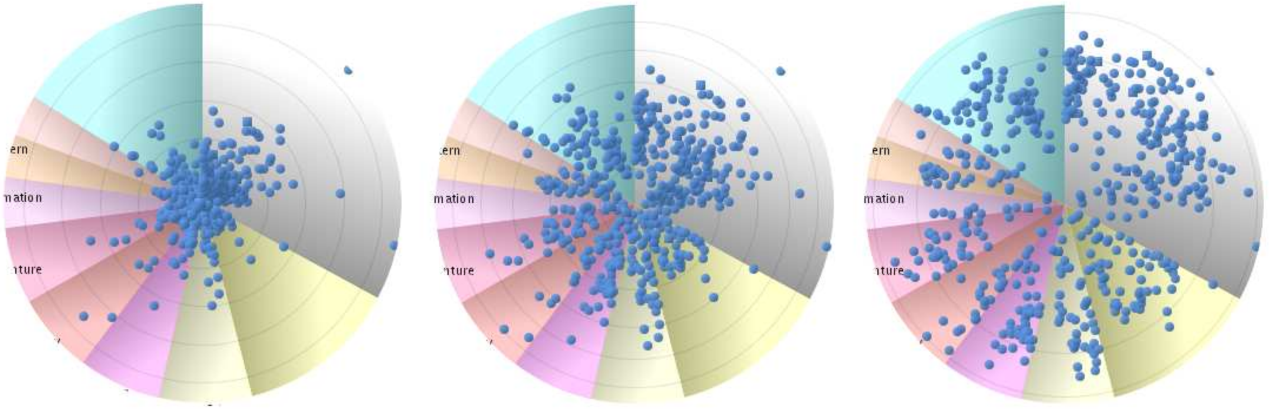


Figure 4. Different Placements for the dots. From left to right: linear, square root and logarithmic placement.

3. META-DATA GENERATION AND CONFIGURATION

An expressive visualization depends on the transformed raw data to encode all the data relations intended and no other data relation. This transformation step usually takes place on the DBMS level. The generation, maintenance and evaluation of these meta-data remind on a usability engineering life-cycle (like Mayhew[11]). And indeed the single steps look analogous. This process usually is very intellectual challenging and normally done by individuals rather than software. Simple counting and calculating of distribution however are preferred for software generated meta-data as well as calculating relevances or rankings. The next step is to map these generated meta-data to visualizations. Usually it helps if the data is collected in a separate view instead of offering the user many database tables. At this stage VisMeB's configuration and assignment tool offers the possibility to map meta-data to visualizations (see figure 5). In a first step the general information about the database and its connection is requested. After accessing the database we are presented a split view: a frame on the left displays the tables in a tree hierarchy. On the right frame the visualizations are also rendered in a tree. Root on the right is a label called 'Visualizations', first children include the main visualizations like 'LevelTable' or 'Scatter-plot' but also a node for the 'Initial Values'. The last level contains the user interface widgets, for example the tables have their 'columns', the scatter-plots have their 'axes' and the CSV has the allocation for the combo boxes and for the two alpha sliders. The user can now connect the database tables to the visualizations by simply dragging a column to the desired node on the right. These mappings are restricted to avoid critical mappings that a visualization couldn't handle, for example it's not possible to connect a text array to a CSV combo-box (nominal data wouldn't make any sense in this case). Additionally it's possible to select meta-data that occur in the form fill-in filter menu or in the form fill-in query interface. The user can also mark table columns that should be included for the keyword ranking mechanism[†].

After all assignments are finished one can now save this configuration and test the settings. Preferably this is done involving the target group. Leaving the usability engineering life-cycle at this point would be critical. The final product is stored and can be shared with other members/work groups.

4. USAGE SCENARIOS AND INTERACTION TECHNIQUES

Typical tasks for the CSV are the reducing of the search space, finding patterns/outliers in a result set or simply get an overview of the distribution. The following example gives an overview of the intended usage scenarios. This scenario is based upon a movie database which provides information (read: meta-data) about films. It is motivated by the fact that movie databases are a common playground for information visualizations and do not require special knowledge on the data/meta-data standards. The usage of the system does not differ if using a

[†] A simple tf*idf algorithm

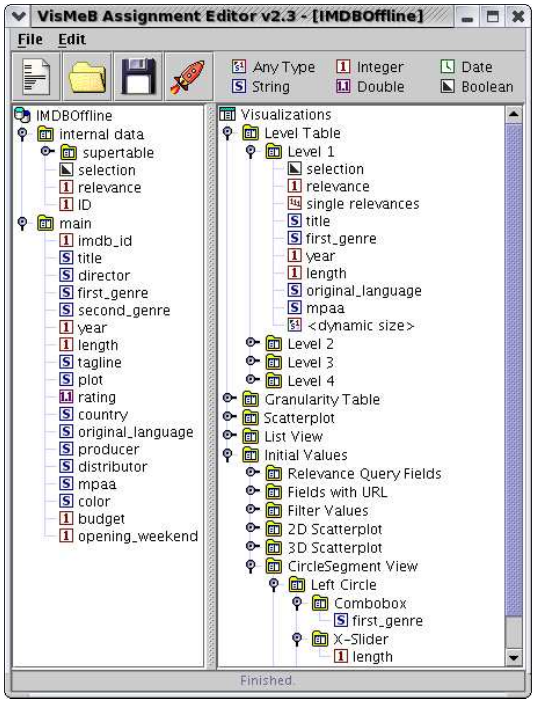
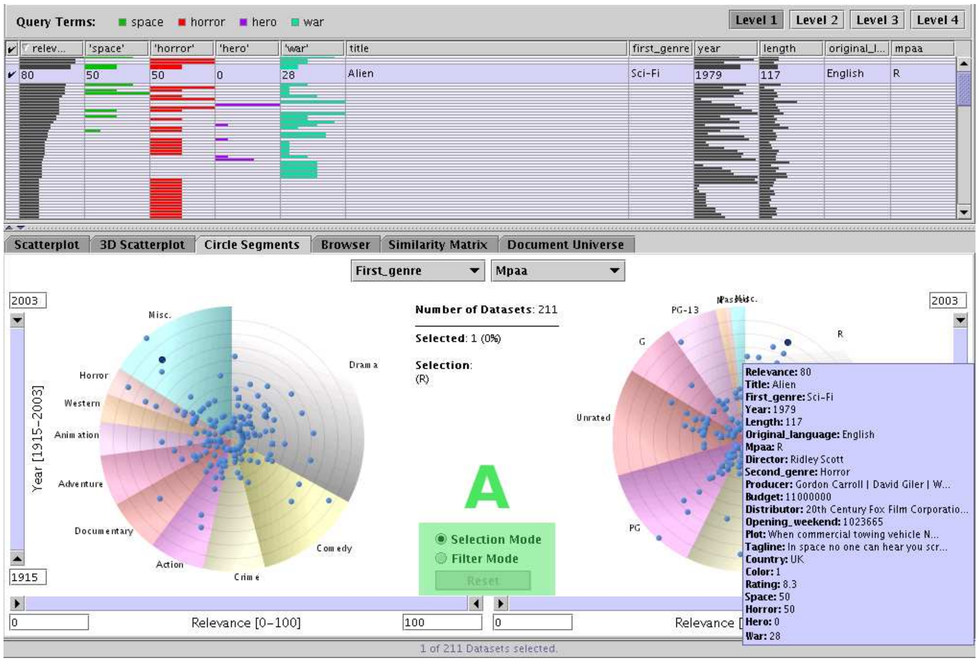


Figure 5. The visual configuration and assignment tool.

Meta-data	Data Type	Movie ID
title	Nominal	The Matrix
director	Nominal	Andy Wachowski
first genre	Categorical	Action
second genre	Categorical	Thriller
year	Quantitative	1999
length	Quantitative	136
rating	Ordinal	8.5
country	Categorical	USA
orig. language	Categorical	english
producer	Nominal	Bruce Berman
distributor	Categorical	AVH
mpaa	Categorical	R
color	Categorical	1
budget	Quantitative	USD 63.000.000
opening weekend	Quantitative	USD 27.788.331
plot	Nominal	...
tag line	Nominal	...

Table 1. Example of meta-data provided by our movie database and used in VisMeB (extracts).

meta-data standard like Dublin Core, ISO19115 or MAB2.
After extraction and generation of meta-data (see figure 3) the underlying database table looks like table 1.



✓	relev...	'space'	'horror'	'hero'	'war'	title	first_genre	year	length	original_l...	mpaa
✓	80	50	50	0	28	Alien	Sci-Fi	1979	117	English	R

Number of Datasets: 211

Selected: 1 (0%)

Selection:
(R)

◉ Selection Mode
○ Filter Mode
Reset

Relevance: 80
Title: Alien
First_genre: Sci-Fi
Year: 1979
Length: 117
Original_language: English
Mpaa: R
Director: Ridley Scott
Second_genre: Horror
Producer: Gordon Carroll | David Giler | W...
Budget: 11000000
Distributor: 20th Century Fox Film Corporatio...
Opening_weekend: 1023665
Plot: When commercial towing vehicle N...
Tagline: In space no one can hear you scr...
Country: UK
Color: 1
Rating: 8.3
Space: 50
Horror: 50
Hero: 0
War: 28

Figure 6. The CSV as a filter working in conjunction with VisMeBs other visualizations. Area 'A' shows the added UI elements. Selection of single documents: Selected and focused documents in the CSV are highlighted in the table. Tool-tips give details on demand.

4.1. Visual Query

A characteristic scenario (used in figure 2) for an user of a movie database is to select a film for her evening entertainment. The user must consider some parameters like 'how long should the running time of the film be' (had a rough day in the office?), 'is the choice admissible for the mate', and so on. The CSV provides an overview, the universe of all the movies and also some general features of the collection. The initial view of the CSV user-interface is preconfigured so that the circles are segmented by the attributes 'first genre' and 'MPAA'[‡] and the numeric meta-data are set to 'year' (mapped to the angle) and 'running time' (mapped to the radius). The viewer has now the possibility to 'zoom' into the collection using the Alpha-sliders. The user restricts the running time of the movie between 90 and 120 minutes because she wants a full length movie that doesn't last too long. She does this by setting one end of the two-sided-slider to 90min and the other end to 120min; only those cases of the data table whose 'length' variable lies between these limits will be displayed in the results. The preview area tells her that still over 300 movies are available. She continues to reduce the result set by limiting the year of publication to the interval of 1980 to 2003 by using the vertical slider. There are still over hundred movies available. It is visually apparent, for example, that a good share of the movies in the collection are 'dramas'. But in order to get some easy fun (and going along with her mate) she selects the genres 'comedy', 'action' and 'adventure'. Only 34 movies remain. This set is further belittled by the assumption, that possibly her 13 year old niece could visit her and so she selects in the second circle only movies that are approved for children ('PG-13', 'PG', 'Approved'). The boolean 'AND' operator that knots the two pie charts calculates a fourteen movies remaining (see figure 2). If the user were to relax one of the criteria only a little (say reducing the lower boundary of 'year' to 1970 she'll get more movies to choose from (in this case: 21).

4.2. Visual Filter

When the user works with VisMeB's result visualizations she's able to use the CSV as a visual filter. For this example let's assume the user gave the system 'space', 'horror', 'hero' and 'war' as relevant keywords and didn't further set any restrictions on the meta-data. The interface of the CSV now gets some additional widgets (see figure 6 A): a radio button that toggles the behaviour of the interface widgets (segments, sliders) and a 'reset' button which restores the initial result set and the initial widget settings. When used in 'selection mode' a single click on a segment selects all documents that are visible in this set. The selection is also visible in the tables through a darkening of the appropriate row (for more information on the SuperTables interaction techniques see[12]). Changing sliders doesn't alter the amount of selected documents. If the user wants an updated view she has to click the segment again after she modified the sliders. Focal items are made visually distinctive in some way: first the data dots are rendered in a darker color, second a tool-tip is provided (see figure 6) and last but not least the data is highlighted in the appropriate row in the SuperTable.
Another feature appears when the user selects a segment. Now he sees the selected data also on the second pie chart. This is very helpful if one wants to see the distribution of the selection under another criteria; e.g. how are the R-rated movies distributed among the 'genres'.
Additionally the user gets a context-menu (right mouse button) which offers useful options like: 'Filter selected', 'Select all visible', 'Deselected all visible', 'Invert visible selection', 'Select all', 'Deselect all', 'Invert selection' and the different placement scales (linear, square root and logarithmic).

When used in 'filter mode' the segments and sliders work as assumed: they filter out everything, that is not visible. This has an instant impact on the SuperTable (data rows vanish or appear), the layout of the circles (data dots move, vanish and appear) and of course on the preview area (showing the actual set size). The method of instantly responding in the display to the dynamic movement of the slider (and the selection of segments) allows users rapidly to explore the multidimensional space of films.
The placement of the data dots through our positioning algorithm and these interaction techniques empowers the user to rapidly scan for outliers, patterns or distributions under different meta-data criteria.

5. EVALUATIONS

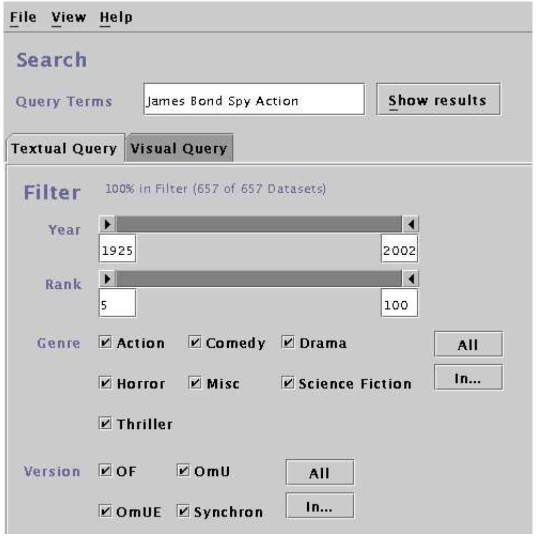
One of the main goals for the VisMeB project was to develop usable visualizations. During the whole design and implementation of the VisMeB framework we followed an usability engineering life-cycle.[13] The tasks and users were clearly specified by the INVISIP project. The specifications (task analysis, etc.) for the other scenarios (movie, web, library) were provided by our own research. The different evaluations were also defined by our work packages. We concluded formative user tests with paper mock-ups, HTML and Java^{TM} prototypes. Additional focus groups, interviews and web based questionnaires completed these studies.
One of the final studies was a comparative evaluation between the CSV and a common form based query interface. The test objective was to prove that the CSV can compete with a form concerning the efficient and effective solving of common tasks. We selected a setting that was based upon a movie database[§] so the users hadn't to care about a powerful meta-data standard like the ISO19115 for geo-meta-data as in the INVISIP project. The independent variable is the user interface type and the treatments are:

- A form fillin interface (see figure 7) and
- the CircleSegmentView

With the CSV we clearly aim at the expert user. Because of its status as a prototype it was impossible to acquire expert users. We decided to generate expert users through a training session. After an introduction to the user interface and the interaction techniques the test persons had to solve some trainings tasks with guidance. These sessions last between 10 and 15 minutes. This was definitely not enough as we saw during the tests. Some users showed clearly that they were unable to use the full potential of the CSV (as well as of the form based

[‡] voluntary movie rating system from the Motion Picture Association of America.

[§] A stripped down version of the one used in these examples



Query Terms: James Bond Spy Action

100% in Filter (657 of 657 Datasets)

Year 1925 2002

Rank 5 100

Genre ☑ Action ☑ Comedy ☑ Drama ☑ Horror ☑ Misc ☑ Science Fiction ☑ Thriller

Version ☑ OF ☑ OmU ☑ OmUE ☑ Synchron

Figure 7. The form fillin interface used in the evaluation.

interface). Because of this we plan to perform a follow up study with more intense training sessions.

The users (n=20; experienced PC users) had to solve nine different tasks. The tasks can be divided in three categories:

A Questions about the distribution of documents.

B Questions about single documents.

C Comparison of different document attributes.

This set of tasks should emphases the strength (T1, T3) and weaks (T2) of the CSV. Each category was represented by three tasks. Each subject was tested on both of the interfaces, but the order of the interfaces was reversed for half of the users. The tasks of each category slightly differ from GUI to GUI, for example:

A1 - CSV How many action films are in the database?

A1 - Form How many comedy films are in the database?

To measure the efficiency we took the time to complete a task. Tasks that last more than 5 minutes or were abandoned by the user were marked as 'not solved'. If the user needed a guidance or hint we also marked this task as 'not solved'.
Our hypotheses for this setting were very conservative:

H0 Both interfaces provide the same efficiency for solving the tasks.

H1 One of the query interfaces is significant better suited.

Looking at the raw data one could easily see that a lot of tasks were solved faster with the CSV (figure 8). We used the Shapiro-Wilk[14] test to examine for distributional adequacy. Only tasks A1, A2, B3 and C2 fulfilled the requirements. Next we calculated a sign test by Dixon that works on our results. To prove this statistical relevant we ran Wilcoxon's matched pairs rank test.

Task	Description	Winner
1	Distribution	Draw
2	Needle in Haystack	Form
3	Set comparison	CSV

Table 2. Results for the Evaluation

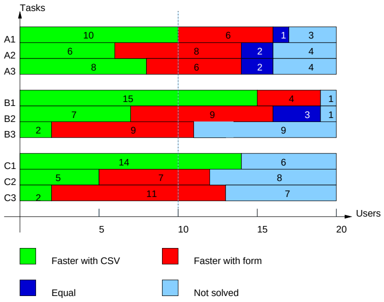


Figure 8. Raw data of the comparative usability test.

The results of this analysis are summarized in table 2. We gave the users the opportunity to state their subjective satisfaction in a post-test questionnaire. We asked for the intuitiveness, effectiveness, joy of use and if they could imagine to work with the CSV on a regular basis. The CSV clearly lost on the topic of 'intuitive and comprehensive user interface'. In all other topics the CSV got more positive than negative ratings. Very promising was the result of the last question: "Could you imagine to work with the CSV on a daily basis?" which 19 from 20 users answered with 'yes'. Anyhow, the free associations of the users were clearly in favor of the form fillin interface. We could track this result to the fact, that the subjects were used to forms and that the CSV interface was not familiar to them. Also the powerful and feature rich user interface demands more training.

6. RELATED WORK

The CSV uses a similar approach as[15] in their 'Circle Segments' but in contrast to them we use each circle to show only the distribution of one dimension and not of all dimensions in the data set. We also offer the possibility to combine two circles with the help of boolean operators. The filter functionality with its visual data mining capacities was inspired by the work of[16] and.[17] 'Multidimensional Data Visualization' as constituted in VisMeb can be found in the 'Visage' System.[18] 'Selective Dynamic Manipulation of Visualizations' (SDM)[19] is a set of interaction techniques for visualizations. Dynamic queries[20][21] can be seen as the predecessor of query previews. Online data visualization has certain features that require special care when it comes to implementation: Distribution over the network forces us to take bandwidth issues in concern. Basic knowledge about this topic provides.[22] VisMeB is intended to be a scalable visualization framework for different application domains. A similar approach could be found in.[23] To reach this purpose we have implemented the Visual Configuration and Assignment Tool that allows adapting our framework to different multidimensional data sets.

7. CONCLUSION AND OUTLOOK

In this paper we introduced a visualization specially designed to work as a query preview and as a visual filter. The whole VisMeB Framework is easily adoptable to a wide range of data sources. If the data source provides meta-data according to a standard (e.g. Dublin Core or ISO19115 for geo-meta-data), almost all of the configuration work is done: only the mapping from the views to the visualization is left over (see section 3). If on the other hand we are faced only with raw data it is unavoidable to analyse the data and generate meta-data and views.

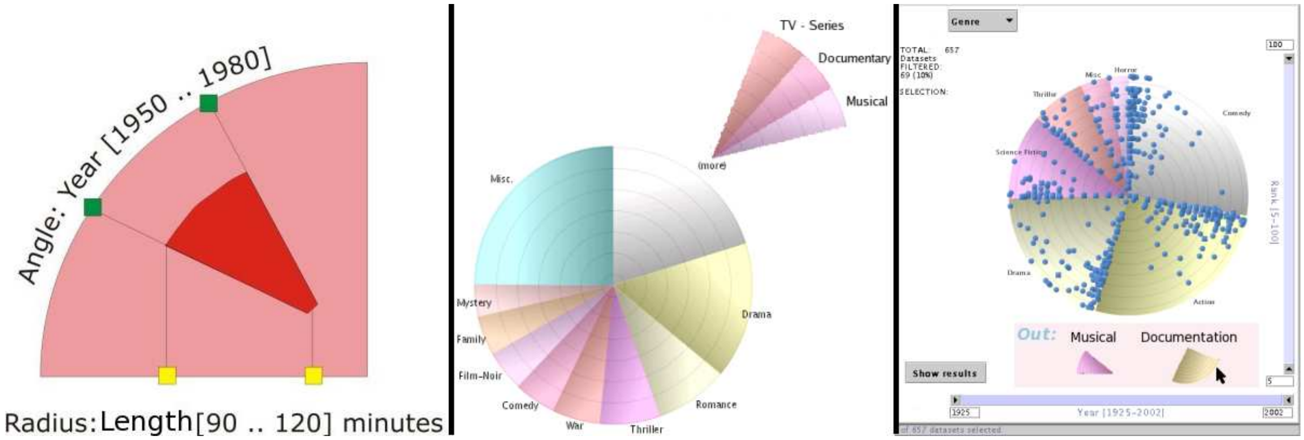


Figure 9. Intuitive widget as a replacement for both Alpha-sliders (left). Mockup for a sub-sections or 2nd stage segmentation (center). Drag'n drop for slices: Visualization recalculates the remaining data dots (right).

The CSV is focused on human search behaviour and on a user-centered design process. The visualization itself is based on various information visualization paradigms like:

- "Space is perceptually dominant, it is good for discriminating values and picking out patterns".[24] → Positioning algorithm, data relations mapped to visual encodings.
- Dynamic queries.[9] → Instant update on distribution data.
- 'Details on demand', 'overview and detail'. → Tool-tips, interaction with SuperTable.
- Brushing and linking. → Interacting with VisMeB's other visualizations.
- Overview first, zoom and filter, then details on demand.[25] → Information Retrieval pipeline as stated in section 2.

The evaluations so far were countenancing and we plan to perform more usability tests with a slightly changed setup to compensate for the novelly user interface, e.g. more training session, long-term observation and a comparative evaluation to find out about the suitability as a visual filter (e.g. scatter-plot vs. CSV).
The future development include examination and design for new interface widgets, that work more intuitively than for example the sliders. A first mock-up can be seen in figure 9 on the left. Here the information space and its possible limitation through two numeric meta-data is rendered as a quarter pie and the intervals can be altered by small boxes on the radius (bottom) and on the arc. The remaining information space (rendered as dark red area) can be moved by the user (similar to the behavior of the alpha-slider).
Improvements concerning the preview could include the direct labeling of the segments, so the user not only sees which category is represented by which segment, but also how many selected/unselected data is in this wedge (e.g. Comedy: 20/45 selected).
Another interesting idea is the sub-segmentation of a wedge (see figure 9, center). Clicking on a segment lets pop-up a quarter pie with subsegments, for example clicking on 'More' reveals in this pop-up a detailed sub-segmentation like 'TV-Series', 'Documentary' and 'Musical'.
Some drag'n drop of segments (to filter hole segments out of the calculation: The user drags the slice 'Comedy' from the pie and the freed space is used up by the remaining slices) are features that should be considered in a next release (see figure 9, right).

REFERENCES

1. D. A. Keim, "Information visualization and visual data mining," *IEEE Transactions on Visualization and Computer Graphics* **8**(1), pp. 1–8, 2002.
2. R. Baeza-Yates and B. Ribeiro Neto, *Modern Information Retrieval*, Addison Wesley (ACM Press), 1999.
3. J. Mostafa, "Workshop - information visualization for retrieval and analysis (ivira)," 2003.
4. G. Marchionini, *Information seeking in electronic environments*, Cambridge Series on Human-Computer Interaction, Cambridge University Press, 1995.
5. S. Card, M. J.D., and B. Shneiderman, *Information Visualization: Using Vision to think*, Morgan Kaufmann Publishers, 1999.
6. P. Klein, H. Reiterer, F. Müller, and T. Limbach, "Metadata visualization with vismeb," in *Proceedings of the 7th International Conference on Information Visualization (IV 03)*, pp. 600–605, 2003.
7. T. Limbach, P. Klein, F. Mueller, and H. Reiterer, "Visualising metadata: Level- vs. granularity concept in the supertable/scatterplot framework," in *HCI International 2003*, 2003.
8. H. Chen and S. Dumais, "Bringing order to the web: Automatically categorizing search results," in *Proceedings of the CHI 2000 Conference on Human Factors in Computing Systems*, pp. 142–152, 2000.
9. E. Tanin, C. Plaisant, and B. Shneiderman, "Browsing large online data with query previews," in *Proceedings of the Symposium on New Paradigms in Information Visualization and Manipulation (NPIVM) 2000*, ACM Press, 2000.
10. C. Ahlberg and B. Shneiderman, "The alphaslider: A compact and rapid selector," in *Proceedings of the CHI '94 Conference*, pp. 365–371, ACM Press, 1994.
11. D. J. Mayhew, *The Usability Engineering Lifecycle: A Practitioner's Handbook for User Interface Design*, Morgan Kaufmann, 1999.
12. P. Klein, F. Müller, H. Reiterer, and M. Eibl, "Visual information retrieval with the supertable+scatterplot," in *Proceedings of the 6th International Conference on Information Visualization (IV 02)*, pp. 59–66, 2002.
13. M. Rosson and J. Carroll, *Usability Engineering: Scenario-Based Development of Human-Computer Interaction*, Academic Press, San Diego, 2002.
14. S. Shapiro and M. Wilk, "An analysis of variance test for normality," in *Biometrika*, **52**, pp. 591–611, 1965.
15. M. Ankerst, D. Keim, and H.-P. Kriegel, "Circle segments': A technique for visually exploring large multidimensional data sets," in *Proceedings of Visualization '96, Hot Topic Session, San Francisco, CA, 1996*, 1996.
16. E. H. Chi, "A taxonomy of visualization techniques using the data state reference model," in *Proceedings of the IEEE Information Visualization Symposium*, pp. 69–75, 2000.
17. B. Shneiderman, "The eyes have it: A task by data type taxonomy of information visualizations," in *Proceedings of the IEEE Symposium on Visual Languages*, pp. 336–343, 1996.
18. S. F. Roth, P. Lucas, J. A. Senn, C. Gomberg, M. B. Burks, P. Stroffolino, and C. Dunmire, "Visage: A user interface environment for exploring information," in *Proceedings of the IEEE Information Visualization Symposium*, pp. 3–12, 1996.
19. M. C. Chuah, S. F. Roth, J. Mattis, and J. Kolojejchick, "Sdm: Selective dynamic manipulation of visualizations," in *Proceedings of the ACM UIST95 Conference Symposium*, pp. 61–70, 1995.
20. C. Ahlberg and B. Shneiderman, "Visual information seeking: Tight coupling of dynamic query filters with starfield displays," in *Conference Proceedings Human Factors in Computing Systems*, J. O. E. B. Adelson, S. Dumais, ed., pp. 313–317, ACM Press, (New York), 1994.
21. C. Ahlberg and E. Wistrand, "Ivee: An information visualization and exploration environment," in *Proceedings of the IEEE Information Visualization Symposium*, pp. 63–73, 1995.
22. E. Tanin, R. Beigel, and B. Shneiderman, "Incremental data structures and algorithms for dynamic query interfaces," *SIGMOD Rec.* **25**(4), pp. 21–24, 1996.
23. M. Kreuseler and H. Schumann, "A flexible approach for visual data mining," *IEEE Transactions on Visualization and Computer Graphics* **8**(1), pp. 39–51, 2002.
24. A. MacEachren, *How maps work*, The Guilford Press, New York, 1995.
25. B. Shneiderman, *Designing the user interface*, Addison-Wesley, 3 ed., 1999.