

From Beautiful to Useful: A Multi-Scale Visualization of Users Movie Ratings

Romain Vuillemot
LIRIS, INSA Lyon, France
romain.vuillemot@insa-lyon.fr

Verónica Peralta
PRiSM, University of Versailles, France
veronika.peralta@prism.uvsq.fr

Research report

July 2007

Abstract. This report presents researches prototyped and experimented in the frame of the APMD¹ Project (2004-2007) test platform. This platform provides data describing movies and users' ratings on such movies, obtained from two public datasets : IMDb² and MovieLens³. Our prototype provides an interactive query environment for analyzing this data, which presents query results in an innovative geo-spatial-like way.

The structure of this report is a 2-page research article and a technical specification article which have been submitted to the IEEE Infovis 2007 Contest⁴.

Keywords : Information Visualization, Data Masses, User Preferences.

¹Personalized Access to Data Masses : <http://apmd.prism.uvsq.fr/>

²The Internet Movie Database : <http://www.imdb.com/>

³Movie Recommendation Website : <http://movielens.umn.edu/>

⁴Conference Website : <http://conferences.computer.org/infovis/infovis2007/>

From Beautiful to Useful: A Multi-Scale Visualization of Users Movie Ratings

Romain Vuillemot*
LIRIS, INSA Lyon, France

Verónica Peralta†
PRiSM, Versailles, France

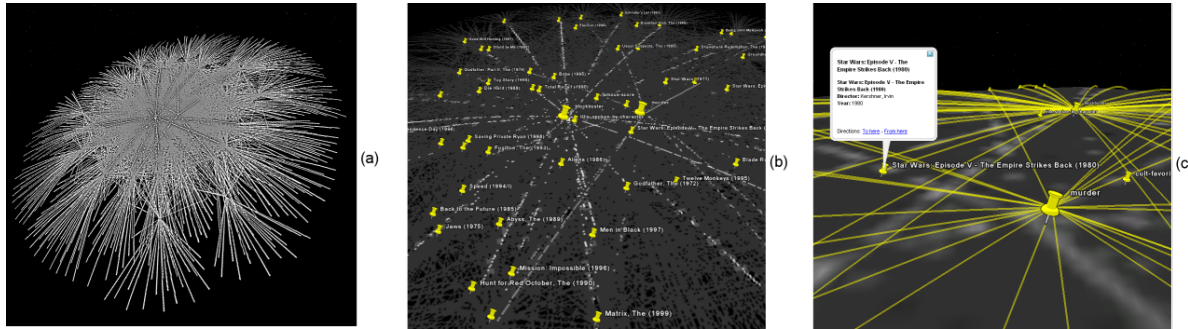


Figure 1: From left to right, from beautiful to useful: (a) movies and keywords relationships overview (with an appealing rendering), (b) movies and keywords relationships (with movies titles and keywords) and (c) movies filtering and details available by user interaction.

ABSTRACT

Interactive environments lack of attractiveness and sex-appeal. While nowadays so many digital arts are available, they have not been included in navigation processes yet. In this paper we suggest to include both artistic and inspiring depictions of data, while proposing an interactive query environment. We present our visualization of the InfoVis 2007 contest data set, focusing particularly on movies rated by users. Our contribution is twofold. Firstly, we extend the contest data set with individual user ratings on movies and we store data in a relational database, allowing SQL-like queries. Secondly, we support multi-scale analysis of query results based on graphical representation of data. This allows high-level analysis of query results and detailed-analysis of specific results by zooming in the data of interest, filtering and getting details on demand. Our proposal incorporates post-processing visualization and hijacking geo-spatial environments in order to explore data. Our approach distinguishes two main phases: (i) construction of a relational database about movies and ratings, (ii) development of an interactive query environment that presents query results in an innovative geo-spatial-like way. Section 1 briefly describes the preparation of the data set, and then, Sections 2 and 3 present data visualization issues and solutions.

1 DATA SET PREPARATION

The original XML-formatted contest data set has been transformed into a relational database. Such a transformation allows us to benefit from Database Management System functionalities, specifically, from efficiently answers to user queries.

We augmented the data set with other publicly available movie features (extracted from IMDb) and user ratings on movies (extracted from MovieLens). The additional movie features consist

in more than forty attributes describing movies (e.g. genres, languages, keywords) and persons involved in movies (e.g. actors, directors, producers). User ratings consist in individual users' evaluations for each movie (rather than average ratings available in the original data set). We sorted a total of 52 tables and 15 aggregation views.

The challenge when studying movie databases is to find correlations between movie features and user behaviors. In that respect, we crossed movie features with the evaluations of each user obtaining a collection of preference rules of the form $attribute = value : support$; for example $genre = Action : 0.80$ means that 80% of movies evaluated by a certain user are Action movies. Statistics on the obtained rules are shown in Figure 2. The attributes that resulted to be more representative of user preferences were keywords and genres. We specially considered these attributes for analyzing data. We also think that social trends come with huge quantity of data, thus we will focus on large subsets of data in our forthcoming analysis.

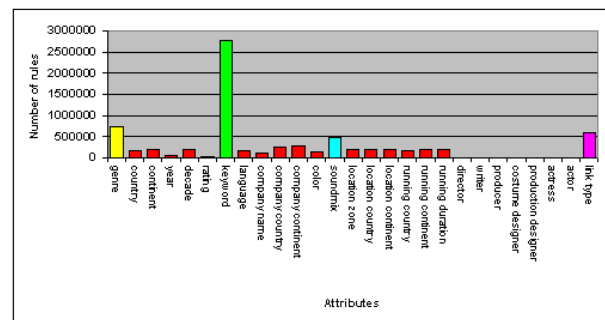


Figure 2: Number of preference rules obtained by movie feature

2 INFORMATION VISUALIZATION

The result of a query is represented as a Graph. Vertices represent nominal attributes (e.g. movies, keywords, users). Edges represent

*e-mail: romain.vuillemot@insa-lyon.fr

†e-mail: veronika.peralta@prism.uvsq.fr

relations among attributes (e.g. a user evaluated a movie). Ordinal attributes (e.g. ratings, budgets, revenues) are represented as graphical properties of edges or vertices (e.g. color). Movie ratings don't hold an explicit layout, thus we picked-up auto-organizing graph layouts based on relationships between movies and attributes of interest. For example, movies having similar keywords are placed nearby in Figure 1b.

Advantages of this representation are: (i) the possibility of seeing great amounts of data at a glance, (ii) the graphical representation of data relationships (instead of presenting long lists of tabular data) and ordinal attributes, and (iii) the neighborhood of data having similar relationships. The major drawback of such a technique is the lack of knowledge the user gets on the data location and their persistence: two layouts of an identical data set might result differently and not always optimally. We managed to circumnavigate that by means of consistent and intuitive coloring. We used LGL [1] to display data resulting from queries.

Post-Processing Image Rendering

We found mandatory to apply an extra step upon the usual visualization generating process [2], using Image Analysis techniques. Image Analysis aim to extract properties from an image, with similar models than the human eye. Among all the existing techniques, we used 2D image filtering capabilities, in order to (i) extract features to assist the user and (ii) provide an artistic looking image while still keeping the same data layout. The trick of image filtering is that every pixel of the image takes the product sum of surrounding pixels. For instance, blur is done by taking the weighted average of the current pixel and its 8 neighbors. Technically, the vicinity is integrated by means of convolution matrix such as the following (Gaussian blur filter kernel example):

$$\begin{pmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{pmatrix}$$

We found across a wide variety of filters that none of them is specifically optimal to a situation; it has to be strategized according to user's tastes. Figure 1a shows a Laplace filter effect combined to a color filling.

3 HIJACKING GEO-SPATIAL ENVIRONMENTS

A Geo-spatial environment provides efficient and intuitive interfaces, by means of multi-scales data exploration. Based on Earth metaphor, it doesn't require a particular learning: everybody inerrantly knows how to "handle earth", from mechanical laws to pictures and miniatures models one might have had as kid. Thus, interactions are easily discoverable even for a non-geo-spatial application. We re-used this paradigm for non-geo-spatial tasks.

2D-Visualizations are mapped into a virtual 3D-sphere, which provides both WIMP (menus, icons) and POST-WIMP (zoom, pan, rotate) interactions. Interactions are quick and permit the user to be lost or to wander randomly to explore visualizations and adopt the tool. We focus on two very interesting features: *Focus+Context* (my means of spherical distortion) and *Multi-Scale* (according to user's altitude).

Focus+Context keeps the focal point at full size and detail, while having view on the surrounding with spherical distortion. By adding a small external map, with a higher point of view we can help the user to enhance his capabilities. By enlarging the map on the globe the spherical distortion will increase.

Multi-Scale is linked to user's *altitude* and indicates how close or far is the user to the visualization. Thus, a strategy is set up coordinating appropriate visualizations at each step of view. Our three major goals are based on known techniques [3] and can be formulated as: (i) attract the user, (ii) give him good insight of the

data, and (iii) let him filter data himself. These goals are achieved by three levels of data visualization, each one conveying a different kind of information but preserving data layout:

Beautiful Overview Layer: is a highly abstracted visualization with structural information only, which has been post-processed to provide an appealing look. The goal is to catch the user's attention, but also providing quantitative analytics elements. Then, the user can identify clusters and areas of interest, and takes the decision to get details by lowering his altitude.

Zoom Layer: is a transitional layer which is very reactive, which starts to partially provide details. No more artistic experience here; the result is raw and some qualitative elements are available. The user is not stuck at this level range: he can go back or forth quickly, especially if he made a mistake or if he already knows what his target is.

Useful Details Layer: is an exhaustive layer about details that the user can dynamically filter (with a check box). The post-processed image is slightly blurred and becomes a background support, with some drawing on top of it to convey detailed informations. Vectorial drawings (with high resolution) are performed and coupled with pinpoints on the map.

Figure 1 illustrates a typical multi-scale analysis of movies keywords.

4 EXPERIMENTATIONS AND CONCLUSIONS

We implemented our system with client-server architecture: data visualization is performed on a distant server and the client application maps the visualization on the 3D-sphere that provides very-reactive environment and dynamic local data selection.

We evaluated our approach for several types of user queries, which are detailed in our contest web page. Our experimentations validated our intuition that graphical, multi-scale and inspiring visualization aids understanding data and provides an alternative point of view for decision-making analysis. The data selection interface has to be enhanced in order to allow users to write their own requests.

Many fields were bridged together (Databases, Computer Graphics, Information Visualization, Geo-spatial Information Systems) and we went a bit further than just a conceptual mock up, by recombining widely spread software.

Our main perspective is to augment the server with more libraries and layouts techniques, in order to provide a wide range of visualizations technique and to offer a Visualization-On-Demand service (VizOD). Another perspective is to define user *visual* profile criteria, according to user's tastes, perceptual capabilities and culture.

ACKNOWLEDGEMENTS

This research was partially supported by the French Ministry of Research and New Technologies under the ACI program devoted to Data Masses (ACI-MD), project #MD-33.

REFERENCES

- [1] A. T. Adai, S. V. Date, S. Wieland, and E. M. Marcotte. Lgl: creating a map of protein function with an algorithm for visualizing very large biological networks. *J Mol Biol*, 340(1):179–190, June 2004.
- [2] E. H. Chi. A taxonomy of visualization techniques using the data state reference model. In *INFOVIS*, pages 69–76, 2000.
- [3] B. Shneiderman. The eyes have it: A task by data type taxonomy for information visualizations. In *VL '96: Proceedings of the 1996 IEEE Symposium on Visual Languages*, page 336, Washington, DC, USA, 1996. IEEE Computer Society.

InfoVis 2007 Contest

From Beautiful to Useful: A Multi-Scale Visualization of Users Movie Ratings

Contest webpage: <http://liris.cnrs.fr/romain.vuillemot/infovis/webpage.html>

Authors and Affiliations:

- Romain Vuillemot, LIRIS, INSA Lyon, Lyon, France, romain.vuillemot@insa-lyon.fr
- Verónica Peralta, PRiSM, University of Versailles, Versailles, France, veronika.peralta@prism.uvsq.fr

Tools:

Server-side:

- LGL (version 1.1 <http://apropos.icmb.utexas.edu/lgl/>): Large Graph Layout Software
- Oracle (version 10g XE <http://www.oracle.com/technology/xe/index.html>): Database Management System.
- The Gimp (<http://www.gimp.org/>): The GNU Image Manipulation Program and Script-Fu scripts (Scheme language).
- Apache (<http://www.apache.org/>), PHP (<http://www.php.net/>): PHP scripts, various file formats converters (.lgl, .gml, .kml, ..) and graph manipulation program specifically written.

Server is running on GNU/Linux Fedora Core 6, Ram 512 Mo, AMD AthlonTMXP 2500+ (1.8 GHz) 512 KB cache memory.

Client-side:

- Google Earth (release 4 <http://earth.google.com/>): 3D-virtual globe program mapping the earth.
- Mozilla Firefox (version 2.0 <http://www.mozilla.com/firefox/>): Web browser.

Client machine is any recent computer.

Data Set Preparation

Our approach consists in the analysis of user ratings as a gauge to raise trends and obtain new insights and correlations about movies. To this end, we augmented the original XML-formatted contest data set with two publicly available data sets:

- **Movie features from IMDb.** The IMDb web site (<http://www.imdb.com/>) provides 49 text files in ad-hoc format (called lists) containing different characteristics about movies (e.g. genres, languages, keywords) and persons involved in movies (e.g. actors, directors, producers). Lists contents is continually updated and enlarged; at the moment we extracted data (October 5th 2006) the movie list included more than 858.000 movies.
- **User ratings from MovieLens.** The MovieLens web site (<http://movielens.umn.edu/>) provides 3 text files in tabular format (ratings.dat, movies.dat and users.dat) containing data about 1.000.209 ratings for 3883 movies by 6040 anonymous users.

We transformed and loaded the three data sets into a relational database. In order to integrate data we needed to match movie titles (which were light different in the three data sets); several heuristics were

executed obtaining an integrated database with information about 3883 movies (i.e. all movies included in the MovieLens dataset). We sorted a total of 52 tables and 15 aggregation views. We still kept a US-centered data set but with dates ranging from 1918 to 2000.

TASK 1: Innovative Design by Means of Auto-Organized Graph Layout combined to Post-Processed Images

The key idea of our design is to construct a graph resulting from SQL queries, in a way that the user gets more insights than the single result. To accomplish that, we focus on large auto-organizing graphs layout techniques (with LGL) which relatively quickly display graphs minimizing visual overlapping. We performed an extra treatment step on the resulting image by means of The GIMP Script-Fu batch interpreter (using Scheme Language), such as filters to enhance user's visual abilities.

The result is appealing, but also efficient since we keep the same data layout: filters integrate vicinities of pixels, but remain roughly at the same location. We focused on two different image filtering techniques: Laplacian filter and Gaussian blur filter.

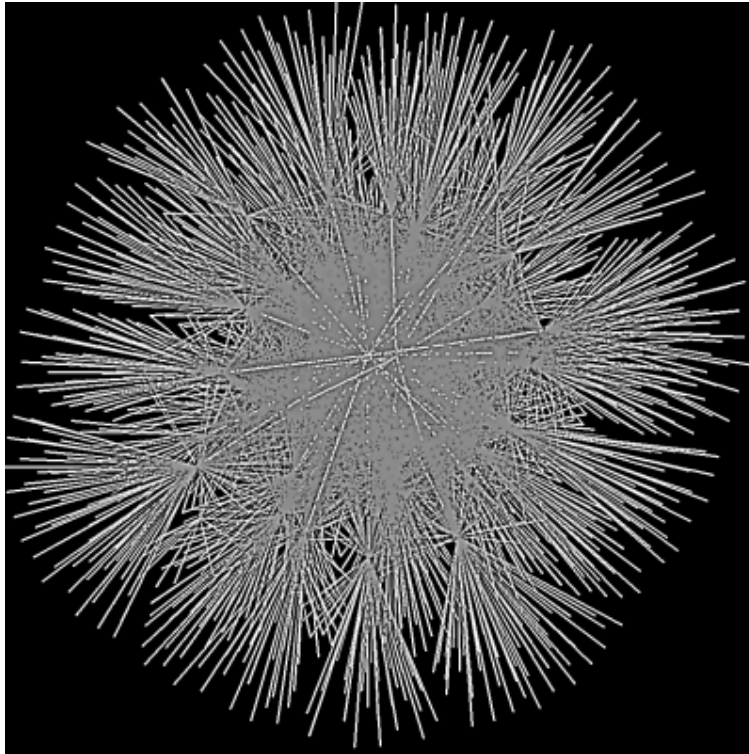
1.1 Laplacian Filter

- **Process 1.1:**

- **The underlying data structure is a graph** where vertices are attributes such as movies, users and keywords. Edges represent an existing relation between attributes. Edges colors are set according some criteria such as vertices degrees.
- **Data layout is auto-organized** according to edges relationships, in order to raise homologies among results.
- **We post-processed the resulting image** from LGL rendering tool with a **Laplacian filter** that extract contours (we filled inner parts with gray) by means of the following convolution matrix:

$$\begin{pmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{pmatrix}$$

- **Image 1.1 :**



Post-processing graph visualization with Laplacian filter and color filling provides an attracting abstraction effect.

- **Insight 1.1:**
 - We provided a reality augmentation by contours detection, helping the user to visualize all the available information at the first sight.
 - The result looks intriguing, showing a whole rather than details. Its artistic looking makes it also attracting, then the user is eager to learn more about information contained within.
 - An external data file keeps track of the informations to allow details addition (See Task 2).
- **Caption for exhibit 1.1:**
 - Post-processing graph visualization with Laplacian filter and color filling provides an attracting abstraction effect.

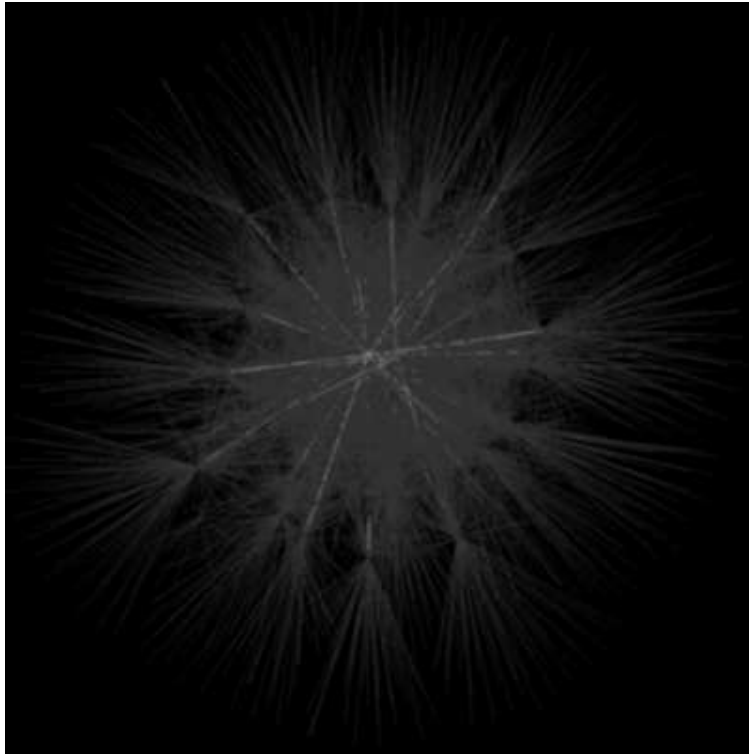
1.2 Gaussian Blur

- **Process 1.2:**

The same process as Process 1.1 has been used, but with a Gaussian Blur filter this time (with no filling). The convolution matrix is as follow:

$$\begin{pmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{pmatrix}$$

- **Image 1.2:**



Post-processing graph visualization with Gaussian blur filter effectively reveal data cluster and trend.

- **Insight 1.2:**
 - We provided a data selection by fading outliers, helping the user to visualize major clusters in a blurry way. Trends appear if there is a high color contrast.
 - The result can't be watched too much time (eyes hurt) but is fine as a background, in order to keep contextual elements.
- **Caption for exhibit 1.2:**
 - Post-processing graph visualization with Gaussian blur filter effectively reveal data cluster and trend.

TASK 2: A Multi-Scale Strategy Combined to a Geo-Spatial Environment as Effective Exploration Interface

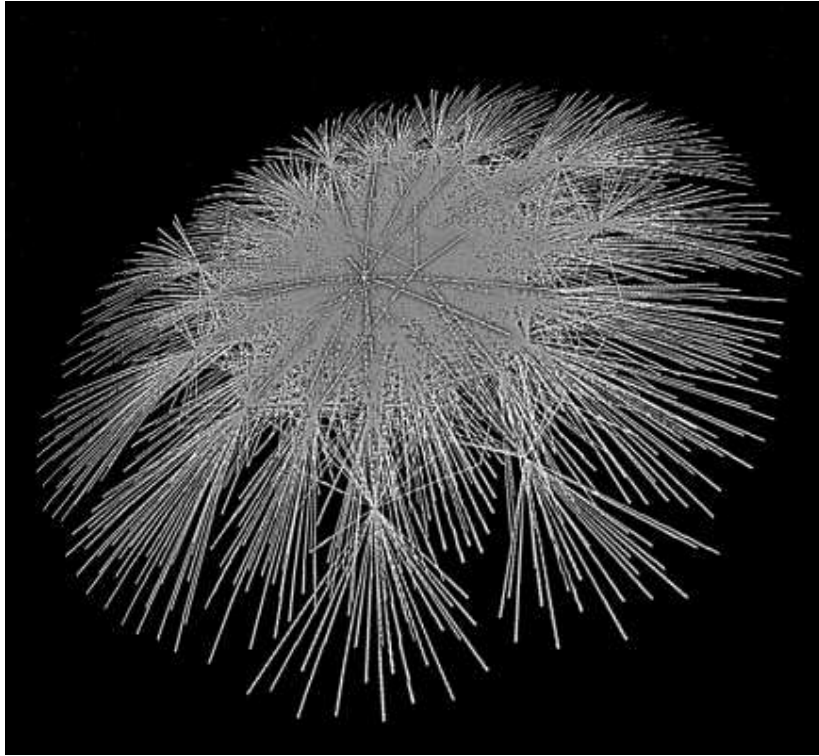
Our design looks interesting, but lacks of efficiency since it provides the user with only structural information. We suggest a 3-layered multi-scale strategy with an overview layer (Laplacian filter), zoom layer (raw image output) and details layer (Gaussian blur filter).

We selected Google earth (GE) as an interactive environment (with all geo-spatial features disabled). Other 3D-globe software like Microsoft Virtual Earth (<http://www.microsoft.com/virtualearth/>) or NASA World Wind (<http://worldwind.arc.nasa.gov/>) would have been fine, but hacking community and APIs are more active with GE.

GE is installed and run by the client, and connects to a remote server on which are hosted and generated visualizations. GE will seamlessly coordinate layers according to user's altitude. Server runs an Apache web server, hosting PHP server-side script writing KML (Keyhole Markup Language, GE inner language) on the fly, to trigger queries, data layout, .. and filter scripts (The GIMP) to map on GE. The average time of the whole process is between 30s and 60s

2.1 Overview Layer

- **Process 2.1:**
 - A very high altitude is associated to the Laplacian filtered image.
- **Image 2.1:**

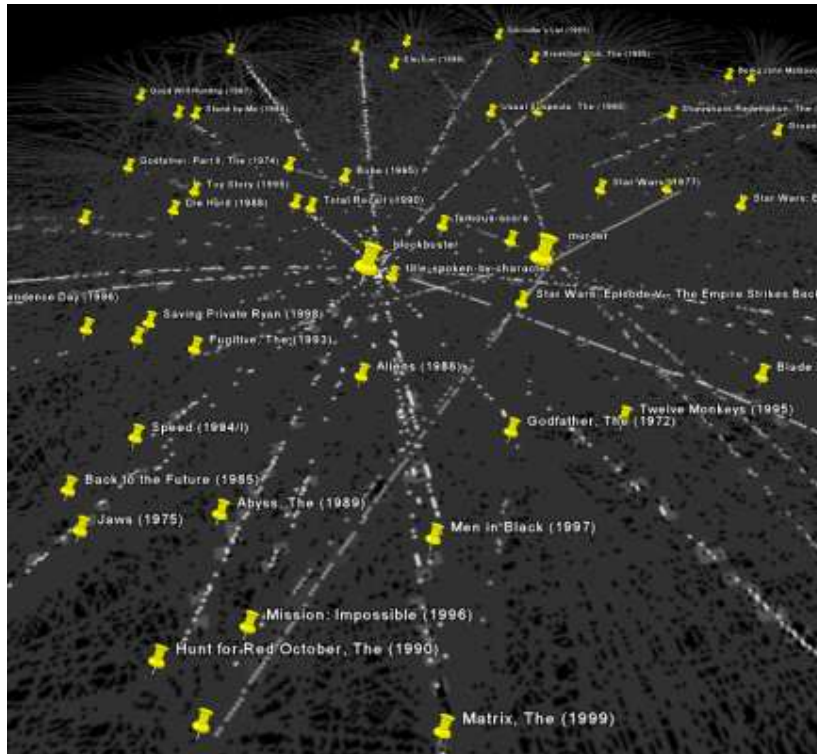


The visualization overview provides an appealing map that can be quickly explored.

- **Insight 2.1:**
 - Laplacian filled filter gives first the user a knowledge of the whole structure of the visualization. Then he can focus either on gray parts (masses of data to explore) or focus on outlines (data away from clusters).
 - GE usability makes every move very smooth and high altitudes allow to quickly rotate and have different angles of view.
- **Caption for exhibit 2.1:**
 - The visualization overview provides an appealing map that can be quickly explored.

2.2 Zoom Layer

- **Process 2.2:**
 - A mid-level altitude is associated to the raw (no post-processing) image output from LGL.
 - We added some extra features in GE which can be enabled/disabled by means of check box click on the sidebar. Features can be movies titles and are automatically retrieved by GE and displayed. The drawing is vectorial, meaning quality is very good but too many drawings slows the interactions. That's why we needed to generate images on a remote server and map them on the globz, rather than providing a vectorial only solution.
- **Image 2.2:**

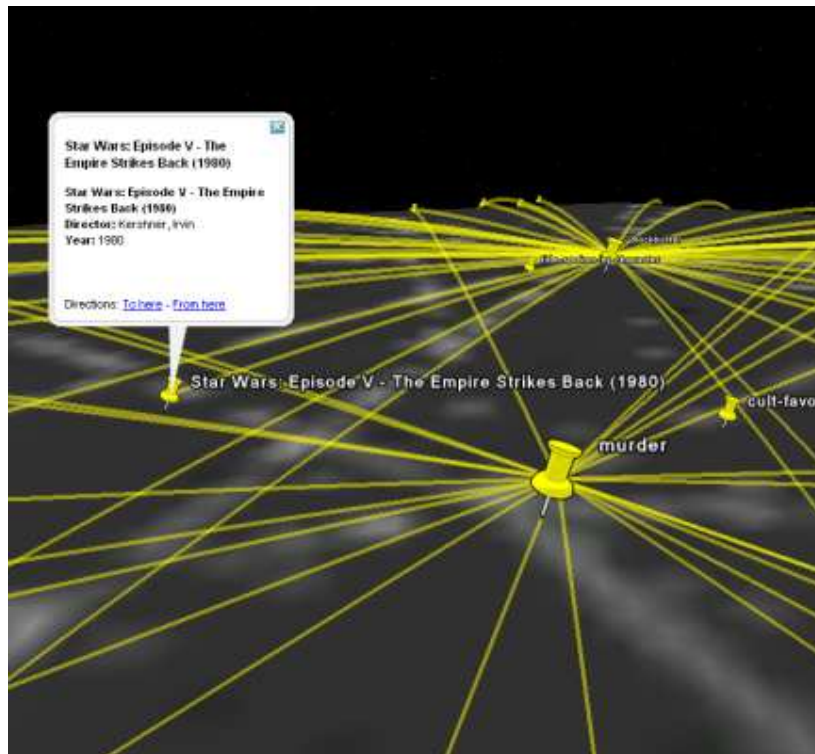


Zooming provides access to raw image output with details.

- **Insight 2.2:**
 - The original image is provided for efficiency (no filter distortion), and then the user can make up his mind on his own. Still he can either go easily back to the abstraction or go quickly to an area of interest he knows for sure he wants to get there.
- **Caption for exhibit 2.2:**
 - Zooming provides access to raw image output with details.

2.3 Details Layer

- **Process 2.3:**
 - From last level to the ground, the mapped image was post-processed with a Gaussian blur filter.
 - GE also retrieves more details by means of KML files, as described in the previous process.
 - Lines on the ground between pushpins help to dynamically indicate relationships (while the blurred background image remains the same).
- **Image 2.3:**



Gaussian blur visualizations help to keep track of the context while dynamically adding and removing details.

- **Insight 2.3:**

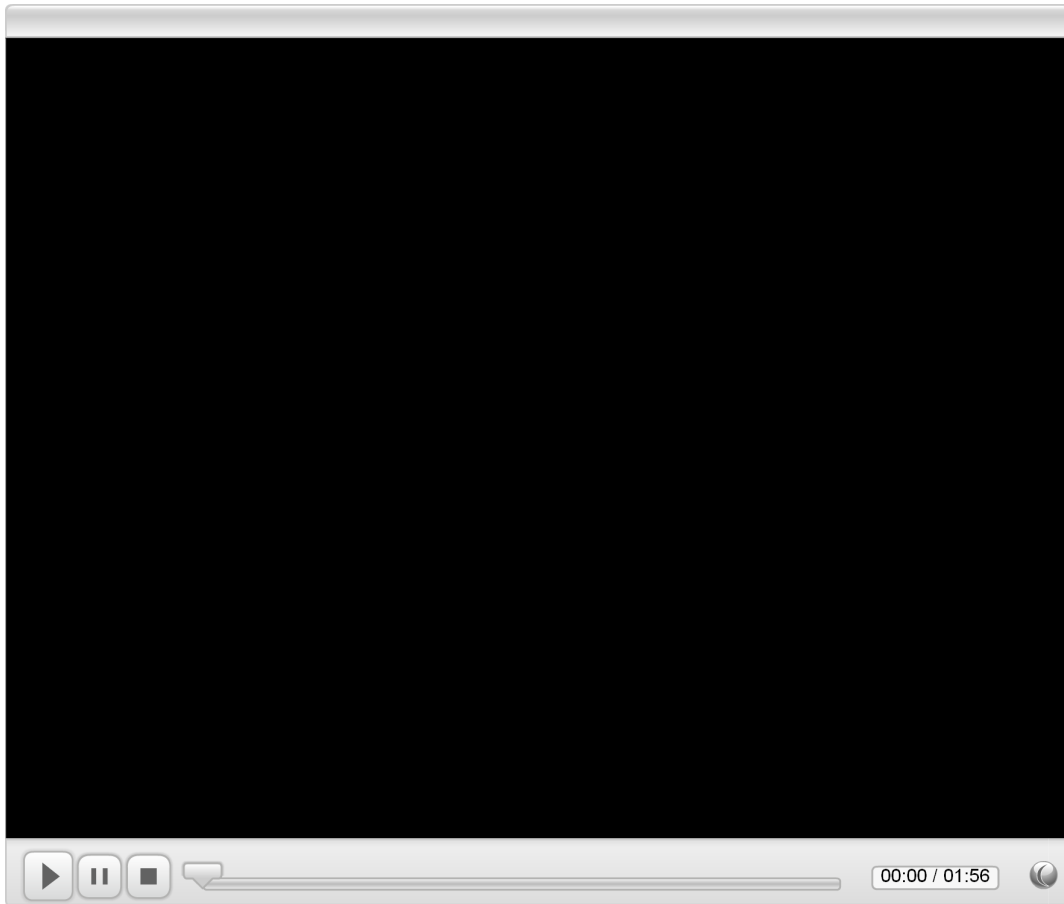
- Providing a Gaussian blur filtered visualization as a background prevent any user cognitive overload.
- The user obtains freedom to add or remove details that helps him to construct or reinforce his mental model.
- Automated navigation can play a scenario, to let the user following a pre-defined path and concentrate on visualizations rather than interactions. Movie posters image instead of pins would have been a great feature, to get quicker previews of the data (not implemented).

- **Caption for exhibit 2.3:**

- Gaussian blur visualizations help to keep track of the context while dynamically adding and removing details.

2.4 Demonstration Video & Prototype Use

The video (without sound) below shows a typical use of our system. **You are welcome to try it out yourself clicking on the following link (http://vizod.insa-lyon.fr/I_kmls.php) and open the file directly with Google Earth.**

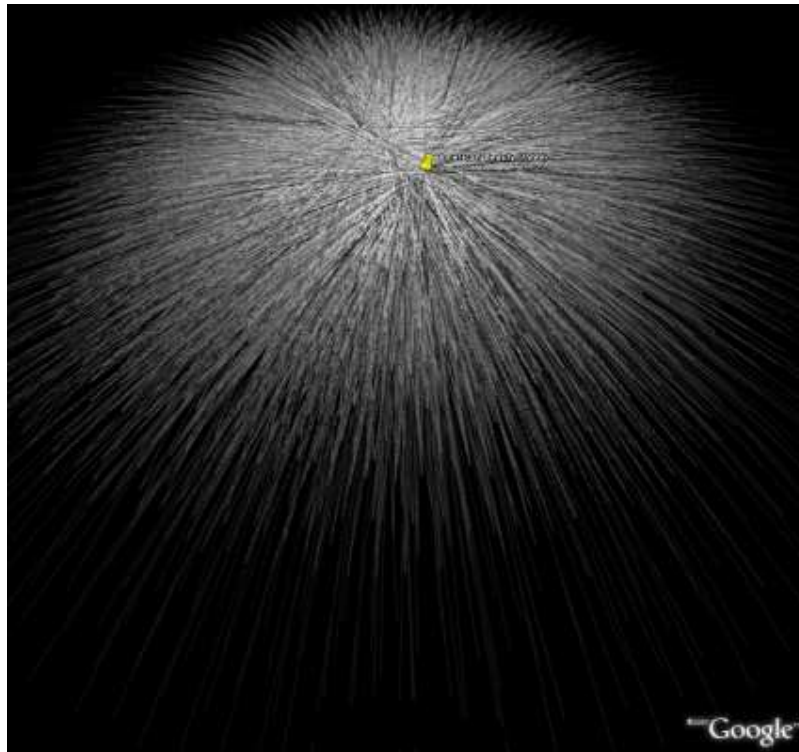


Demo video (without sound). Quicktime version available here (<http://liris.cnrs.fr/romain.vuillemot/infovis07/demoqt.mov>) (49Mo).

TASK 3: Users movie rating insights

Task 3.1: Which movie is the most rated by top 50 contributors?

- **Process 3.1:**
 - We selected the top 50 contributors as edges. For each edge we connected movies which have at least another contributor in common.
- **Image 3.1:**

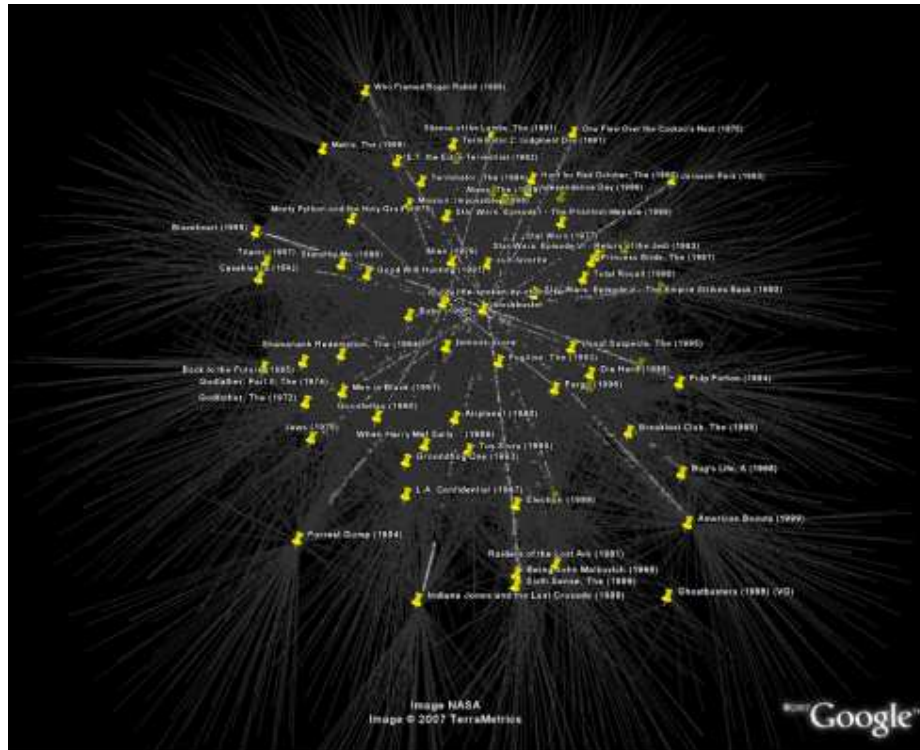


American Beauty is likely the most *common* movie shared by most contributors

- **Insight 3.1:**
 - Movies the most rated by contributors gather in the middle of the graph. Most rated movies are either action movies or social phenomena as *American Beauty*. Large circles with shades of gray shows that top users don't have that many movies in common.
 - **NOTE: it would have been interesting to select a known community of users, to get relationships among them. That allows to get info on which are the most commons movies, or which movies are rated by only two persons, ..**
- **Caption for exhibit 3.1:**
 - *American Beauty* is likely the most *common* movie shared by most contributors

Task 3.2: Which are the keywords from the most rated movies (e.g. movies with the most ratings, not always the bests)?

- **Process 3.2:**
 - We selected the 15 most rated movies as edges. For each edge, we connected keywords. The vertices color indicates the number of movies connected to a keyword, from dark (low) to light (high) colors.
- **Image 3.2:**

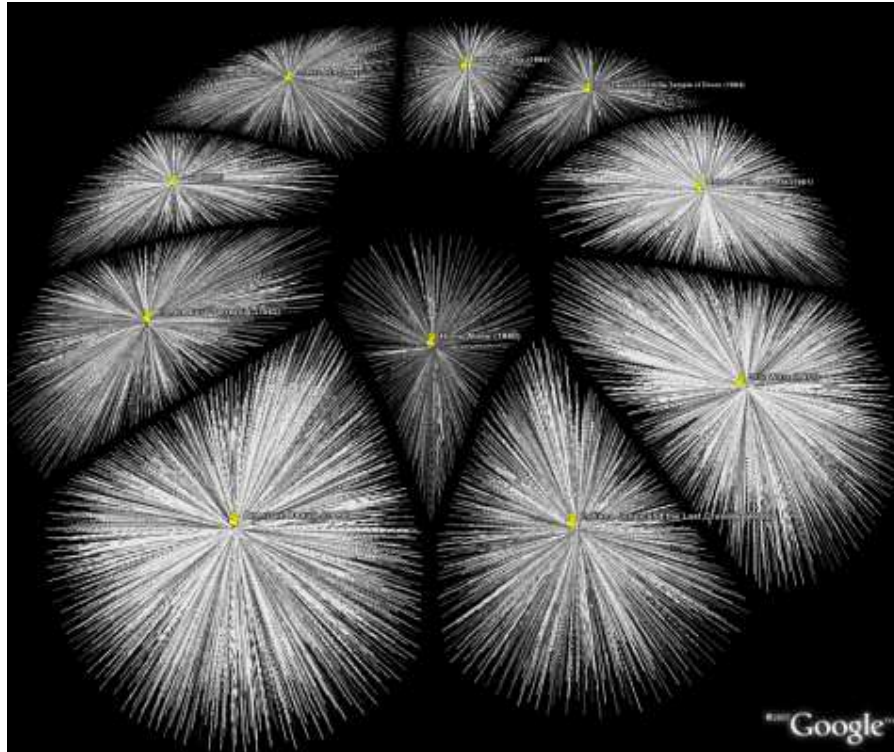


Blockbusters and *Murder* are most-used keywords from movies with highest number of ratings.

- **Insight 3.2:**
 - It is not surprising that keywords such as *Blockbusters* and *Murder* are the most salient ones, according to the high budgets dedicated to these film genres.
 - Our visualization shows important differences in the number of keywords attributed.
- **Caption for exhibit 3.2:**
 - *Blockbusters* and *Murder* are most-used keywords from movies with highest number of ratings.

Task 3.3: Is there any correlation between *most profitable* movies and users behavior?

- **Process 3.3:**
 - We selected movies with highest difference REVENUE-BUDGET, aka the most profitable movies (data not always available for all entries). Roots are movies and leaves user ratings. Ratings vary from 1 to 5 (best), so does the edge color from dark to white.
- **Image 3.3:**

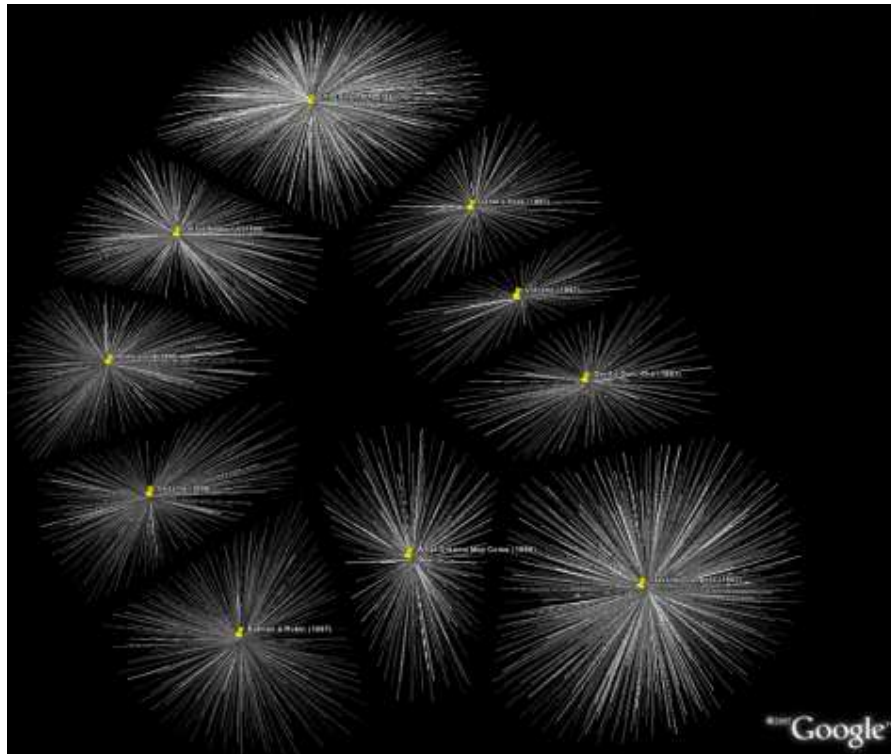


Most profitable movies tend to be rated more and with highest grades.

- **Insight 3.3:**
 - Here comes *American Beauty* again with business, popularity and opinion success. More than half of the movies have had sequels, but not all of these are visible, meaning they might have been less profitable. *Home Alone* is singular because of his lack of popularity which can be explained by focusing on youth people (not using the internet).
- **Caption for exhibit 3.3:**
 - Most profitable movies tend to be rated more and with highest grades

Task 3.4: Is there any correlation between *less* profitable movies and users behavior?

- **Process 3.4:**
 - Opposite as Process 3.3: we selected top 10 movies with highest deficit.
- **Image 3.4:**



Movies may have had many positive ratings while not being a business success.

- **Insight 3.4:**

- *The Fifth Element* and *Starship Troopers* have the same trend: user's massive and mainly positive interests, but not making money. Our opinion is that this is explained by high contrasts among the ratings. One can also stress that these movies were mostly *second-degree* and got lots of bad *first-degree* critics in magazines, which ranked them badly straight.
- Other movies are simply bad ones .

- **Caption for exhibit 3.4:**

- Movies may have had many positive ratings while not being a business success.

COMMENTS

An invisible innovation we made is a brand new architecture: centralization of the visualizations generation, and delocalization of interactions. The aim is that data layout takes a lot of time, thus if performed at unique place one may concentrate all the machine power there. Then the local client gets the image and can add extra informations on top of it with higher quality, while keeping the same data layout. Re-organizing a graph layout takes time as well as can disturb the user in case of radical structure change.

Our biggest disappointment is not having been able to use the terrain as a dimension to display data. Results are currently on a flat earth, where it could have been fantastic to build up mountains and peaks according to movies budget or revenue. Our wish was to hack the internal DTM (Digital Terrain Model) or DEM (Digital Elevation Model) rather than adding extra polygons where texture mapping is difficult.

We are conscious using third party tools might be harmful, in the long run. But advantages is that it makes us not focusing on the software engineering part, neither on the learning process/documentation since we re-use the knowledge users already got while using GE for another purpose. Our focus being mainly on usability and experimentations. Also NASA World wind was also a good candidate with more open terms of use. We just hope Google won't go evil too soon.