Data-Driven and User-Driven Multidimensional Data Visualization*

Rober Morales-Chaparro, Juan C. Preciado, and Fernando Sánchez-Figueroa

Quercus Software Engineering Group, Universidad de Extremadura
{robermorales,jcpreciado,fernando}@unex.es

Abstract. Data Visualization on the Web is one of the main pillars for understanding the information coming from Business Intelligence based systems. However, the variety of data sources and devices together with the multidimensional nature of data and the continuous evolution of requirements is making this discipline more complicated as well as passionate. This paper outlines a process for obtaining a multidimensional data visualization driven by both, the data and the user, providing an automatic code generation. While the designer is automatically provided with a wide range of possible visualizations for a given data set, the user can change the visualization in several ways: the dominant dimension, the kind of visualization and the data set itself by adding, removing or grouping variables.

Keywords: Data Visualization, Web Engineering, Business Intelligence.

1 Introduction

Data Visualization is becoming more and more important in Web applications for Business Intelligence. Not only is important extracting the relevant information for the company but also showing it in the appropriate way. Company managers want to see their business' situation in a quick and easy way, in order to make decisions correctly, efficiently and on-time.

Different Data Visualization techniques are being widely used for this purpose [Bro08, TSD10]. Their interactive nature enables users to explore patterns, test hypotheses, discover exceptions, and explain what they find to others [Rob08].

These techniques have proven to be useful when the requirements do not change over time. However, for those applications with evolving requirements, this kind of systems generates a growing dissonance between what the users want to know, and what the application can show. Step by step, original requirements differ more and more from the actual necessities.

Under this situation, company managers have two options: managing more than one application and/or document to take decisions or contacting again the software company that developed the application to adapt it to the new requirements. The former has the problem of being an error-prone and tedious task, while the latter has the risk of not providing the information on-time for the company purposes.

* This work has been developed under the Spanish Contract MIGRARIA - TIN2011-27340 funded by Ministerio de Ciencia e Innovación.

A. Harth and N. Koch (Eds.): ICWE 2011 Workshops, LNCS 7059, pp. 159–166, 2011.

If the users know exactly how to see the information, why not letting them to drive and customize the presentation just to obtain the information in the most appropriate way? Several authors have identified this challenge as "interdisciplinary collaboration": there should be a communicative balance between visualization masters and application domain experts [KEM07]. So, it is time for user-driven visualization on the Web.

However, the variety of current data sources (query languages, APIs, etc.) together with the plethora of different devices for visualization and the ultimate practices coming from social applications for tagging information, are making data visualization on the Web even more challenging. The existence of semantic and/or contextual information around data is very useful for visualization purposes [Ber67]. This fact opens the opportunity to reason about data, bringing the possibility of automating Data Visualization. So, it is time for data-driven and user-driven visualization on the Web.

When combining several kinds of data, the visualization can be different depending on the dominant dimension (i.e. it is not the same showing the "hair color of some people", that showing "number of people with a certain hair color"). From a user-driven point of view, it would be interesting to play with different visualizations for the same set of data or even changing the dominant dimension just to find the information he is interested in. Under this multidimensional nature of data, we can say it is time for data-driven and user-driven multidimensional visualization on the Web.

Precisely, the main contribution of this paper is presenting a data-driven and user-driven process to visualize multidimensional data on the Web. The main benefits of the proposal are twofold. On the one hand, the possibility for the designer to reuse one visualization among different applications; on the other hand, the possibility for the user (and the designer) to have automatically more than one visualization for the same data set. Far from giving details, this paper outlines the whole process we are following in our research.

The rest of the paper is organized as follows. Section 2 introduces a motivating example while Section 3 presents the process. Finally, Section 4 outlines conclusions and related works.

2 Motivating Example

The restaurants manager of an international airport has a visualization software solution that shows up important data to her. The dashboard shown in Figure 1 summarizes data about time and target country of departures.

This solution fit well her needs in the past. However, now she wants to change the thematic of the airport restaurants, to be of cultures around the world: the menu at a given time will depend on the nationality of most of the passengers in the airport at that time. So it is needed to obtain which countries are the destinations/origins of most flights at different moments of the day. Desirably, she would be able to query the system, and the system able to answer.

From the system point of view: there are neither new data nor new casuistry among them or new actors. In addition, from the manager point of view: she knows better than everyone around the world how is the new visualization she needs. Despite this, her visualization software is not able to show her the data in the appropriate way. She needs a system that i) knows very well the nature of the data, ii) offers the possibility of choosing between different visualizations, playing with the number of items to be showed and the dominant dimension that drives the visualization.

Such a system is briefly introduced in next section. This system should be able to automatically show the dashboard of Figure 2.

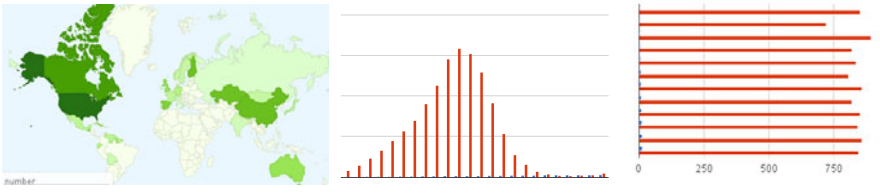

Fig. 1. Original dashboard. Left: average departures by target country. Center: average departures by hour. Right: average departures by month.

Fig. 2. Desired dashboard. At 9am and 3pm, frequency by target country.

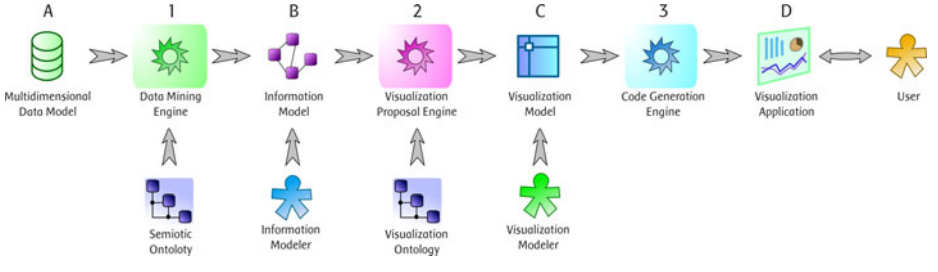

Fig. 3. Sequential steps of the process, from the data to the user

3 Proposal

The process is divided into three main sub-engines: data mining, visualization proposal and code generation (Figure 3). Next we briefly explain the whole process.

3.1 Data-Driven Concerns and the Multidimensional Challenge

Dimensions are the attributes of every object from the data set (i.e. name, age, gender,…), or the aggregations of attributes (i.e. average age, distinct name, count, etc.). Together with the name, their semantic annotations include (a) the type, (b) the range if applicable, (c) the relevance, (d) the relationship with other dimensions and (e) the organizational level (roughly: qualitative ⊂ ordinal ⊂ quantitative) [Ber67].

Each visualization has two parts. The first is constrained by the dominant dimension, which is usually mapped with the 2D position of the objects, i.e. used as the axis. The second is the selection of visual variables (mainly: color, intensity, size, orientation, grain and shape). Their accuracy for human perception is very well documented [CM84]. This part is usually summarized in a legend.

The "data mining engine" (Figure 3-1) is the entry point of the process. Its goal is to propose visualizations with the maximum utility and accuracy of perception. The utility is the capacity of showing the snapshot of the data that reveals potential non-obvious patterns. The accuracy of perception is about choosing the visual variables that best represents the selected attributes. To get these two goals, it automates every concern inferable from the multidimensional data. The input of the engine is a "data model" with semantic annotations (Figure 3-A). The output is an "information model", which stores all the decisions taken in the phase (Figure 3-B).

This phase is supported by a semiotic ontology, which contains the rating between the semantic markup of the dimensions and the visual variables, i.e., it stores that a quantitative dimension fits well with size but not with color.

The first task of the phase is building the most relevant perspectives: 1. Which is the dominant dimension of the data? 2. Which are the most relevant dimensions (which potentially will reveal patterns or trends)? and 3. Is it suitable to make groups, filters or annotations, to see the data better?

Then, it tentatively performs the matching of dimensions with visual variables: 1. Taking into account the dominant dimension: what is the best disposition of the object set? 2. For each output dimension: what visual properties do fit better? 3. Seeing the size of the dataset: is it needed managing filters, or perhaps the focus and the context?

In our motivating example, the input of the engine is the data model in Figure 4. Table 1 shows the first stage: the relevant perspectives that the system has found based on the data mining analysis of the data model. Table 2 represents the second task of the phase: the best visual variables for the dimensions of those perspectives. The decision takes into account, mainly, their organizational level as scored in the ontology (not shown). After that, the data-driven phase ends with an "information model": the composition of tables 1 and 2 (not shown).

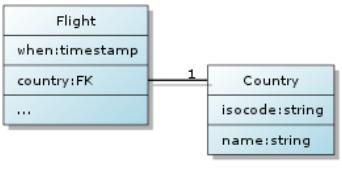

Fig. 4. Annotated data model

Table 1. Proposed perspectives

Perspective	Dimensions		
	dominant	outputs	group
Flights by hour	When	Count	Hour
Flights by month	When	Count	Month
Flights by country	Country	Count	Hour

Table 2. Map between attributes and visual variables

Dimensions	⟶	Organizational level	⟶	Variables
Count		Qualitative		Intensity, size
When		Ordinal		Hor. Axis
Country		Qualitative		map

3.2 Model-Driven Flow

At this stage, the modeler has the option of improving the skeleton of the "information model". After this optional refinement, a "visualization proposal engine" selects the best patterns to display the perspectives (Figure 3-2).

Patterns are the formalization of a reusable graphical representation. The suitable patterns and their requirements are stored in the visualization ontology. For instance, one pattern is the "bar chart", which can be used for lots of different data sets, as long as the structure required for its usage is very common. While the input of the phase is the "information model", the output is a skeleton of a "visualization model" (Figure 3-C). Over that, the modeler can change the patterns and edit their preferences.

In our example, it is easy to see what the ontology will propose, seeing the output scoring at table 3: 1. Flights by hour → Bar chart; 2. Flights by month → Bar chart; 3. Flights by country → Intensity map.

Two facts must be observed: on the one hand, how the pattern "bar chart" can be used more than once. On the other hand, there is more than one pattern that could fit well for a given perspective; these are precisely two of the main benefits outlined in Section 1.

Table 3. Scoring of some patterns about displaying the indicated perspectives

	bar chart	*column chart*	*line chart*	*pie chart*	*heat map*	*time line*	…
flights by hour	**0.8**	0.7	0.5	0.4	0.0	0.3	…
flights by month	**0.8**	0.7	0.5	0.4	0.0	0.3	…
flights by country	0.5	0.4	0.0	0.3	**0.9**	0.0	…

The "code generation engine" (Figure 3-3) is intended to generate the runtime code for the system, using all the information previously collected. This is done by model transformations. It uses the "visualization model" as input, and outputs the final code of the application. Previous studies [JJK08] shows up the fact that web-native display technologies (HTML5, SVG, etc.) have the potential to expand the impact of visualization in some cases. So, we use them for the final code.

3.3 User Experience

Using the formal storage provided by ontologies, and, also, the formal representation of models, the proposed framework can now allow the user: a) maintaining the visualization patterns, editing their preferences (color, disposition, etc.) b) moving (maintaining the dominant dimension) from one visualization pattern to another, if the current one does not fit well with her interests. Afterwards, she optionally can perform a). c) changing the dominant dimension to get a new perspective, and then optionally b) and a). d) changing the dataset she wants (probably editing the filter), and then optionally c), b) and a).

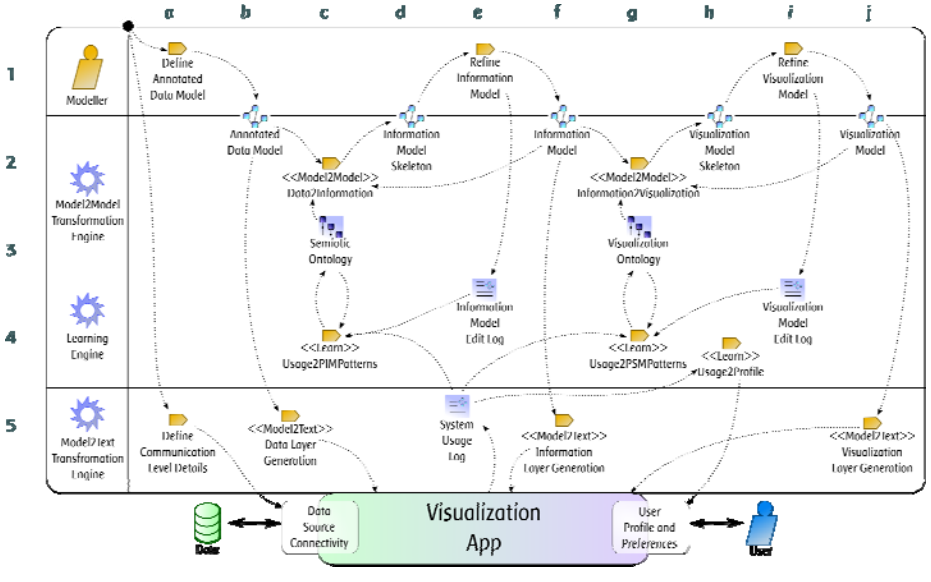

Fig. 5. SPEM diagram, with phases and artifacts of the methodology

Now, the example of the airport should be revisited from this perspective. Suppose that the user chooses the interaction referred as c). As expressed in the initial example: she wants to see "flights by hour and country". Then, the "data mining engine" finds new visual metaphors for the new dimensions and the "visualization proposal engine" searches for visualization patterns that best ranks the new combination. The user finally ends with the desired dashboard, like the one shown in Figure 2. A running demo of the motivation example can be seen at http://visualligence.com/airport_gv/. Although it has been manually developed, follows the process in order to show the viability and the possibility of automation.

3.4 SPEM Representation

The different parts of the proposal have been formalized in a SPEM representation (Figure 5). From row 1 to 3 it can be observed the correlation with the diagram in Figure 3. On row 5 there is the correspondence with the "code generation engine". The figure also reveals a learning process from selections performed by both, the designer and the user (row 4). The more frequent a pattern is selected, the more probably it will be automatically suggested for similar problems in the future. Also, it is useful storing those selections in the users' profile: so the system can adapt better to their preferences.

4 Conclusions and Related Work

This paper has presented a data-driven and user-driven approach for visualizing multidimensional data on the Web. Far from giving details, the paper has outlined the whole process with all its phases. This process is being integrated in RUX [LPSF07], a tool aimed to model advanced user interfaces. This integration is not over yet.

Table 4 shows some related works. Different features have been considered for them (user-driven, data-driven, model-driven, multidimensional, tool available and Web oriented). Although there are relevant works such as [BBC+11], to our knowledge there is not an existing approach considering all these issues.

Table 4. Summary of related proposals

	User	*Data*	*MDE*	*High-dim*	*Tool*	*Web*
[MLG+10]	✓	✓			✓	
[The03]	✓			✓	✓	
[BSL+01]	✓			✓	✓	
[SCB98]	✓			✓	✓	
[FPSSO96]		✓		✓	✓	
[Mac86]		✓		✓	✓	
[SH00]	✓			✓	✓	
[BBC+11]		✓	✓	✓	✓	✓
Our proposal	✓	✓	✓	✓	✓	✓

References

[BBC+11] Bozzon, A., Brambilla, M., Catarci, T., Ceri, S., Fraternali, P., Matera, M.: Visualization of Multi-domain Ranked Data. In: Search Computing, pp. 53–69 (2011)

[Ber67] Bertin, J.: Semiologie Graphique: Les Diagrammes, Les Reseaux, Les Cartes (1967)

[Bro08] Brooks, M.G.: The Business Case for Advanced Data Visualization (2008)

[BSL+01] Buja, A., Swayne, D.F., Littman, M., Dean, N., Hofmann, H.: XGvis: Interactive data visualization with multidimensional scaling. Journal of Computational and Graphical Statistics, 1061–8600 (2001)

[CM84] Cleveland, W.S., McGill, R.: Graphical perception: Theory, experimentation, and application to the development of graphical methods. Journal of the American Statistical Association 79(387), 531–554 (1984)

[FPSSO96] Fayyad, U., Piatetsky-Shapiro, G., Smyth, P., et al.: Knowledge discovery and data mining: Towards a unifying framework. In: Proc. 2nd Int. Conf. on Knowledge Discovery and Data Mining, Portland, OR, pp. 82–88 (1996)

[JJK08] Johnson, D.W., Jankun-Kelly, T.J.: A scalability study of web-native information visualization. In: Proceedings of Graphics Interface 2008, pp. 163–168. Canadian Information Processing Society (2008)

[KEM07] Kerren, A., Ebert, A., Meyer, J.: Human-Centered Visualization Environments. Springer, Heidelberg (2007)

[LPSF07] Linaje, M., Preciado, J.C., Sánchez-Figueroa, F.: Engineering Rich Internet Application User Interfaces over Legacy Web Models. IEEE Internet Computing 11(6), 53–59 (2007)

[Mac86] Mackinlay, J.: Automating the design of graphical presentations of relational information. ACM Transactions on Graphics 5(2), 110–141 (1986)

[MLG+10] Matković, K., Lež, A., Gračanin, D., Ammer, A., Purgathofer, W.: Event Line View: Interactive Visual Analysis of Irregular Time-Dependent Data. In: Taylor, R., Boulanger, P., Krüger, A., Olivier, P. (eds.) Smart Graphics. LNCS, vol. 6133, pp. 208–219. Springer, Heidelberg (2010)

[Rob08] Roberts, J.C.: The Craft of Information Visualization (January 2008)

[SCB98] Swayne, D.F., Cook, D., Buja, A.: XGobi: Interactive Dynamic Data Visualization in the X Window System. Journal of Computational and Graphical Statistics 7(1), 113 (1998)

[SH00] Stolte, C., Hanrahan, P.: Polaris: a system for query, analysis and visualization of multi-dimensional relational databases. In: Proceedings of IEEE Symposium on Information Visualization 2000, INFOVIS 2000, pp. 5–14 (2000)

[The03] Theus, M.: Interactive data visualization using mondrian. Journal of Statistical Software 7(11), 1–9 (2003)

[TSD10] Turban, E., Sharda, R., Delen, D.: Decision support and business intelligence systems. Prentice Hall Press, Upper Saddle River (2010)