Goal-Based Selection of Visual Representations for Big Data Analytics

Matteo Golfarelli[1,2], Tommaso Pirini[1], and Stefano Rizzi[1,2(✉)]

[1] DISI, University of Bologna, V.le Risorgimento 2, 40136 Bologna, Italy
{matteo.golfarelli,tommaso.pirini,stefano.rizzi}@unibo.it
[2] CINI, Via Salaria 113, 00198 Roma, Italy

Abstract. The H2020 TOREADOR Project adopts a model-driven architecture to streamline big data analytics and make it widely available to companies as a service. Our work in this context focuses on visualization, in particular on how to automate the translation of the visualization objectives declared by the user into a suitable visualization type. To this end we first define a visualization context based on seven prioritizable coordinates for assessing the user's objectives and describing the data to be visualized; then we propose a skyline-based technique for automatically translating a visualization context into a set of suitable visualization types. Finally, we evaluate our approach on a real use case excerpted from the pilot applications of TOREADOR.

Keywords: Big data · Visual analytics · Skyline queries

1 Introduction

As a consequence of the wide diffusion of big data technologies and of the increasing amounts of valuable data generated by sensors, devices, social media, etc., companies of all sizes have become aware of the opportunities lying with *big data analytics* (BDA), where advanced analytic techniques operate on big data sets aimed at complementing the role of traditional OLAP and data warehouses [15]. However, the lack of in-house technical skills often prevents companies from really benefiting of BDA, or even discourages them from taking this direction because of the outsourcing costs. In this context, the H2020 TOREADOR (TrustwOrthy model-awaRE Analytics Data platfORm) Project adopts a *model-driven architecture* (MDA [11]) to streamline BDA processes and make them widely and easily available to companies following a BDA-as-a-service approach. Following the basic principles of MDAs, TOREADOR builds on three models to support BDA [2]:

1. **Declarative Model**: an abstract and platform-independent model that specifies the user goals (*what* BDA should achieve) in terms of data collection,

This work was partly supported by the EU-funded project TOREADOR (contract n. H2020-688797).

S. De Cesare and U. Frank (Eds.): ER 2017 Workshops, LNCS 10651, pp. 47–57, 2017.
https://doi.org/10.1007/978-3-319-70625-2_5

preparation, analysis, and visualization. It corresponds to the *computation-independent model* in MDA terminology.
2. **Procedural Model**: a platform-neutral, vendor-independent model that specifies the algorithms for data preparation and for parallelizing and executing the analytics, as well as the way to present the results to users (*how* BDA should work). It corresponds to the *platform-independent model* in MDA terminology.
3. **Deployment Model**: the computational components and other resources for the process on a specific target execution platform (e.g., Hadoop-as-a-service). It corresponds to the *platform-specific model* in MDA terminology.

Remarkably, as required by the MDA paradigm, each model is (semi-) automatically derived from the previous one.

As sketched in Fig. 1, within the TOREADOR framework the three models are grouped into five conceptual areas: *preparation*, *representation*, *analytics*, *processing*, and *visualization*. The focus of this paper is on the visualization area, in particular on (i) how to specify the users objectives and describe the dataset to be visualized within the declarative model (e.g., comparison-oriented visualization of 4-dimensional numerical data with low-cardinality domains), and (ii) how to translate this specification into a concrete platform-independent solution (e.g., bar chart) within the procedural model, which will be eventually translated into a deployment model on the target execution platform (e.g., stacked-to-group bar chart in the D3 Java library, `d3js.org`). Specifically, the main contributions of this paper are:

– As part of the declarative model we define a *visualization context* based on seven prioritizable coordinates for assessing the user's objectives and conceptually describing the data to be visualized (Sect. 3).
– We describe a skyline-based technique for automatically translating a visualization context from the declarative model onto the procedural model in the form of a set of suitable visualization types (Sect. 4).

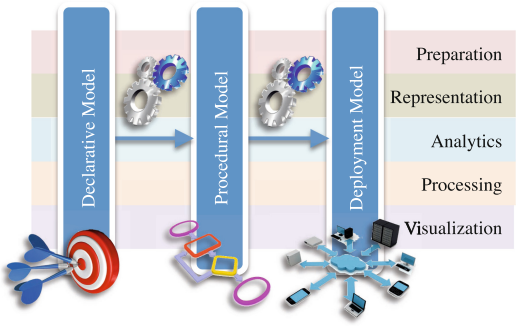

Fig. 1. The TOREADOR framework

The paper outline is completed by Sect. 2, which discusses the basic related literature, and by Sect. 5, which evaluates our approach through a real use case excerpted from the pilot applications of TOREADOR and draws the conclusions.

2 Related Work

Visualization has a key role in BDA to enable users to understand the problem, generate hypotheses and define the solution, as well as to steer the analysis process in dealing with massive, incomplete, and incorrect data [9].

Several papers propose principles and taxonomies to classify the different approaches to visualizing data and interacting with them. A seminal paper in this field is the one by Shneiderman, who proposes a classification taxonomy for data visualization based on two coordinates: *task* (e.g., overview, zoom, and details-on-demand) and *data type* (e.g., multi-dimensional, tree, and temporal) [16]. Another influential paper is [10], where Keim proposes a different classification of data visualization and visual data mining techniques by considering, besides the data type, the *visualization technique* and the *interaction and distortion technique*. A few years later, Abela listed four possible *goals* for visualization, namely relationship, comparison, distribution, and composition [1].

More recently, Börner surveyed the main classifications proposed in the literature and made a significant effort to integrate them into a single, consistent framework [4]. Her visualization framework is based on six coordinates, namely *insight need type* (which integrates [7,18]), *data scale type* (based on [17]), *visualization type* (based on [3,16]), *graphical symbol type*, *graphical variable type*, and *interaction type* (which integrates [10,16]). A more detailed classification of data types, including for instance datetime components and IRIs, is introduced in [14] with reference to the visualization of linked open data; the paper also relates each common type of chart to the user goals it is most compliant with. Finally, in [6] a new coordinate is introduced to visualize linked open data: the *user type*. Users are distinguished into *lay-users* and *techies*.

Despite the richness and detail of the classifications available, to the best of our knowledge only few papers focus on the criteria for deciding which type of chart is best suited for a given combination of data type, dimensionality, user goal, etc. In [1], a simple decision tree is proposed to select the best visualization according to the user's goal and to the main features of data (namely, the number of variables, the cyclicality, and the size). A description of the pros and cons of different charts to be used in the security domain is provided in [12]; the specific aspects of data considered include their dimensionality, cardinality, and type. A flow-chart is also provided to help users in choosing the right visualization for different goals and data dimensionality, but not all combinations are taken into account. In the context of big data, a framework for choosing the best visualization is outlined in [5]; specifically, the main types of charts are related to the user goals they fulfill and to the data dimensionality, cardinality, and type they support.

3 A Declarative Model for Visualization

In this section we describe the coordinates we use to enable users to declare their objectives and describe the dataset to be visualized. The method we followed to select these coordinates can be summarized as follows:

1. We analyzed the literature on the taxonomies of data visualization and interaction paradigms to derive a set of candidate *coordinates* (e.g., data type) and, for each coordinate, a set of candidate *values* (e.g., ordinal).
2. From these candidate coordinates/values we derived a set of questions to be submitted to users for requirement elicitation.
3. Based on the elicitation, we selected a final set of coordinates and values.

For requirement elicitation we adopted the Kano model [8], a useful tool for understanding needs and expectations of a stakeholder based on how they affect his/her satisfaction with a given product. The Kano model classifies requirements into *must-be*, *one-dimensional*, *attractive*, *indifferent*, and *reverse* based on their location along two dimensions, namely, the degree of satisfaction and the level of functionality. To position each requirement a questionnaire is submitted to each user (in our context, the key users of the pilot applications of TOREADOR), then the results are aggregated and evaluated. In the following we list the coordinates we selected, see Table 1 for the values each coordinate can take:

(1) *Goal*, which enables users to declare their analysis goal. This classification follows the one into *basic task types* proposed in [4].
(2) *Interaction*, which enables users to declare the type of interactions to be supported by the visualization. This classification derives from the one proposed in [4]; specifically, based on requirement elicitation, we selected a subset of most common and intuitive interaction types out of those proposed in [10].
(3) *User*, which enables users to declare their skill as in [6].
(4) *Dimensionality*, which enables users to declare the number of variables they wish to visualize. Here, as done in [1], we count all variables without distinguishing between independent and dependent variables.
(5) *Cardinality*, which enables users to qualitatively declare the cardinality of the data to be visualized like in [1].
(6) *Type*, which enables users to declare the type of each variable to be analyzed. The classification we adopt here is the one in [17], but as in [12] we distinguish independent from dependent variables.

Definition 1 (Visualization Context). *Let $O_1, \ldots, O_7$ be the sets of goals, interactions, users, dimensionalities, cardinalities, independent types, and dependent types, respectively, as listed in Table 1 ($O_6 \equiv O_7$); let $C = \{1, \ldots, 7\}$ and $O = \bigcup_{i \in C} O_i$. A* visualization context *is defined by a function $c : C \rightarrow O \cup \{\text{NULL}\}$ (where $c(i) \in O_i \cup \{\text{NULL}\}$) and by a weak order $\overset{c}{\succ}$ on the set C that expresses the priorities between the seven coordinates.*

Table 1. Visualization coordinates

Value	Objective	Example
Goal		
Composition	Highlighting the way in which distinct parts of data are composed to form a total	Stacked column chart
Order	Analyzing objects by emphasizing their ordering	Alphabetical list of names
Relationship	Analyzing the correlation between two or more objects or attribute values	Scatter plot
Comparison	Examining two or more objects or values to establish their similarities and dissimilarities	Column chart
Cluster	Analyzing data in such a way as to emphasize their grouping into categories	Dendrogram
Distribution	Analyzing how objects are dispersed in space	Histogram
Trend	Examining a general tendency of data variables	Line chart
Geospatial	Analyzing data values using a geographical map as a graphical context	Choropleth map
Interaction		
Overview	Gain an overview of the entire data collection	Dendrogram
Zoom	Focus on items of interest	Network map
Filter	Quickly focus on interesting items by eliminating unwanted items	Area chart
Details-on-demand	Select an item and get its details	Choropleth map
User		
Lay	Computer-literates who may have troubles in understanding complex visualizations	Line chart
Tech	Skilled users with a deeper understanding of BDA	Tree map
Dimensionality		
1-dimensional	A single numerical value or a string	Gauge
2-dimensional	One dependent variable as a function of one independent variable	Single-line chart
n-dimensional	Each data object is a point in an n-dimensional space	Bubble chart
Tree	A collection of items, each having a link to one parent item	Dendrogram
Graph	A collection of items, each linked to an arbitrary number of other items	Network map
Cardinality		
Low	From a few items to a few dozens items	Pie chart
High	Some dozens items or more	Heat map
Type		
Nominal	Qualitative, each data variable is assigned to one category	Pie chart
Ordinal	Qualitative, categories can be sorted	Histogram
Interval	Quantitative, it supports the determination of equality of intervals or differences	Line chart
Ratio	Quantitative, with a unique and non-arbitrary zero point	Scatter plot

Example 1. An example of visualization context is

$$c(1) = \text{Comparison}, \quad c(2) = \text{NULL}, \quad c(3) = \text{Tech},$$
$$c(4) = \text{n-dimensional}, \quad c(5) = \text{High}, \quad c(6) = \text{Interval}, \quad c(7) = \text{Ratio}$$
$$(3 \overset{c}{\sim} 6) \overset{c}{\succ} 1 \overset{c}{\succ} (2 \overset{c}{\sim} 4 \overset{c}{\sim} 5 \overset{c}{\sim} 7)$$

where the user expresses three levels of priority: high (for the user and independent type coordinates), medium (for the goal coordinate), and low (for the remaining coordinates). □

4 Going Procedural

To translate the visualization context stated by the user in the declarative model into a set of suitable visualization types in the procedural model, we first need to assess to which extent each visualization type is suitable for each value of each coordinate introduced in Sect. 3.

Definition 2 (Suitability Function). *A* suitability function *is a function* $\sigma : O \times V \rightarrow s$ *where* O *is the set of all coordinate values,* V *is the set of all visualization types, and* $s \in \{\text{unfit}, \text{discouraged}, \text{neutral}, \text{acceptable}, \text{fit}\}$ *is a suitability score.*

Our approach is general enough to be applicable to each possible visualization type v as long as a suitability evaluation is done for v based on our seven coordinates. Currently we consider a set of 25 widely used visualization types classified as shown below [4]:

- **Tables** are ordered arrangements of rows and columns in a grid, with data values stored in cells (e.g., pivot table —Fig. 2a).
- **Charts** visually depict quantitative and qualitative data without using a well defined reference system (e.g., tag cloud —Fig. 2b).
- **Graphs** plot quantitative and qualitative data using a well-defined reference system, such as Cartesian coordinates (e.g., bubble chart —Fig. 2c).
- **Maps** display data according to their spatial relationships and show how data are distributed geographically (e.g., heat map —Fig. 2d).
- **Network layouts** use nodes to represent sets of data records, and inter-node connections to represent relationships (e.g., dendrogram —Fig. 2e).

Then we defined a suitability function by assigning a score to each visualization type/coordinate value pair; the scores were derived from the literature (mostly from [1,4,12]). For instance, in Table 2 we show the suitability scores for three popular visualization types.

The next problem is that of using the suitability function to find, given a visualization context c, one or more "most suitable" visualization types. To this end we start by observing that, with reference to c, visualization type v

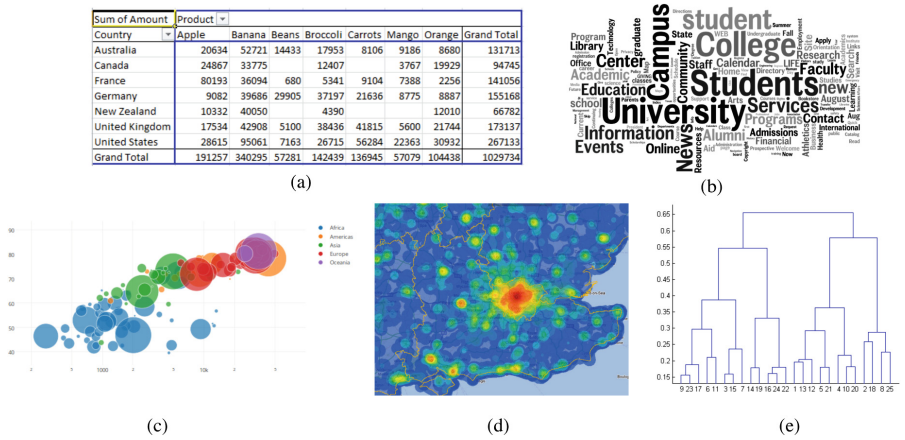

Fig. 2. A pivot table (a), a tag cloud (b), a bubble chart (c), a heat map (d), and a dendrogram (e)

Table 2. Suitability scores for three visualization types

		pie chart	bubble chart	heat map
Goal:	Composition	fit	unfit	unfit
	Order	neutral	unfit	unfit
	Relationship	unfit	fit	unfit
	Comparison	neutral	fit	acceptable
	Cluster	unfit	acceptable	acceptable
	Distribution	discouraged	fit	fit
	Trend	unfit	fit	unfit
	Geospatial	unfit	unfit	fit
Interaction:	Overview	fit	acceptable	fit
	Zoom	unfit	acceptable	fit
	Filter	neutral	neutral	neutral
	Details-on-dem	acceptable	neutral	neutral
User:	Lay	fit	acceptable	acceptable
	Tech	acceptable	fit	fit
Dimensionality:	1-dimensional	unfit	unfit	unfit
	2-dimensional	fit	unfit	unfit
	n-dimensional	unfit	fit	fit
	Tree	unfit	unfit	unfit
	Graph	unfit	unfit	unfit
Cardinality:	Low	fit	acceptable	acceptable
	High	discouraged	discouraged	fit
Independent Type:	Nominal	fit	unfit	neutral
	Ordinal	acceptable	neutral	acceptable
	Interval	discouraged	fit	fit
	Ratio	discouraged	fit	fit
Dependent Type:	Nominal	unfit	fit	unfit
	Ordinal	unfit	fit	discouraged
	Interval	unfit	fit	fit
	Ratio	fit	fit	fit

is evaluated through a 7-tuple $\langle \sigma(c(1), v), \ldots, \sigma(c(7), v) \rangle$ where each element expresses the suitability of v for c along one coordinate. On the other hand, the suitability scores introduced in Definition 2 are obviously related by a (strict) total order expressing a preference:

$$\text{fit} > \text{acceptable} > \text{neutral} > \text{discouraged} > \text{unfit}$$

So we can compare any two possible visualization types $v, v' \in V$ along each single coordinate: for the i-th coordinate, v is strictly better than v' if $\sigma(c(i), v) > \sigma(c(i), v')$.

Now, we have to combine the seven resulting one-dimensional preferences into a composite one for the whole 7-tuple. A popular way to cope with this problem is to look for tuples (corresponding in our case to visualization types) that are Pareto-optimal. A tuple is *Pareto-optimal* when no other tuple dominates it, being better in one dimension and no worse in all the other dimensions. In the database community, the set of tuples satisfying Pareto-optimality is called a *skyline* [13]. The definition of dominance is given below in flat (non-prioritized) form; it is given with reference to a subset of coordinate C' to be more easily generalized to the prioritized case in Definition 4.

Definition 3 (Flat Dominance). *Given visualization context c and two visualization types v and v', and given the set of coordinates $C' \subseteq C$, we say that v is* flat-substitutable *to v' on C', denoted $v \sim_{C'} v'$, iff $\sigma(c(j), v) = \sigma(c(j), v')$ for all $j \in C'$ such that $c(j) \neq$ NULL. We say that v* flat-dominates *v' on C', denoted $v \succ_{C'} v'$, iff (a) $\exists i \in C' : \sigma(c(i), v) > \sigma(c(i), v')$ and (b) for all other $j \in C'$ such that $c(j) \neq$ NULL it is $\sigma(c(j), v) = \sigma(c(j), v')$.*

Example 2. With reference to the visualization context in Example 1, we consider three visualization types: pie chart, bubble chart, and heat map. The three suitability 7-tuples to be compared are shown in Table 3; the scores are excerpted from Table 2. Considering all the six specified coordinates (the interaction coordinate is not specified), it is bubble chart $\succ_C$ pie chart and heat map $\succ_C$ pie chart. Specifically, bubble chart flat-dominates pie chart because it is better on all coordinates except dependent type and cardinality, on which it is equivalent; similarly for heat map. On the other hand, there is no flat-dominance or flat-substitutability relationship between bubble chart and

Table 3. Suitability tuples for three visualization types with reference to the visualization context in Example 1

	pie chart	bubble chart	heat map
Goal: Comparison	neutral	fit	acceptable
Interaction: NULL	—	—	—
User: Tech	acceptable	fit	fit
Dimensionality: n-dim	unfit	fit	fit
Cardinality: High	discouraged	discouraged	fit
Independent Type: Interval	discouraged	fit	fit
Dependent Type: Ratio	fit	fit	fit

heat map because the first is better on the goal coordinate, while the second is better on the cardinality coordinate. So overall, if coordinate priorities are not considered, both bubble chart and heat map would belong to the skyline while pie chart would not. □

The last step is that of considering the priorities $\overset{c}{\succ}$ declared by the user as part of the visualization context. To this end we resort to the concept of *prioritized skyline* given in [13] and redefine dominance as follows.

Definition 4 (Dominance). *Given visualization context* $c, \overset{c}{\succ}$ *and two visualization types* v *and* v'*, and given the set of coordinates* $C' \subseteq C$*, we say that* v dominates v' *on* C' *(denoted* $v \rhd_{C'} v'$*) iff either (a)* $v \succ_{max(C')} v'$ *or (b)* $(v \sim_{max(C')}) \wedge (v \rhd_{C' \setminus max(C')} v')$*, where* $max(C')$ *denotes the top coordinates in the* $\overset{c}{\succ}$ *order restricted to* C'*.*

Intuitively, if v is better that v' with reference to the coordinates that take highest priority for the user, then it is unconditionally better than v'; otherwise, if v is equivalent to v' with reference to those coordinates, we have to check if it is better with reference to the coordinates taking second priority, and so on.

Definition 5 (Skyline). *The* skyline *for* $c, \overset{c}{\succ}$ *is the set of visualization types in* V *that are not dominated by any other visualization type.*

Example 3. Considering again the visualization context in Examples 1 and 2, and taking now into account the coordinate priorities, it is bubble chart $\rhd_C$ heat map $\rhd_C$ pie chart. Indeed, since bubble chart and heat map are equivalent on the two top-priority coordinates (i.e., user and independent type), we have to check the second-priority coordinate (goal), on which bubble chart are better than heat map. So, taking into account priorities, the skyline only includes bubble chart. □

5 Evaluation and Conclusions

In this paper we have described an approach to automate the translation of the objectives declared by the user for visualizing the results of BDA into a set of most suitable visualization types. The approach enables users to specify a value for seven visualization coordinates, assigns a qualitative suitability score to each visualization type, then computes the skyline to determine the set of Pareto-optimal visualization types.

To evaluate our approach we have implemented a Java prototype whose interface supports the declaration of the visualization context and returns the prioritized skyline of visualization types. Then we have let the users of the three pilot applications of TOREADOR use this prototype to express a visualization context for their BDA use cases, and checked that they are satisfied with the visualization types proposed. For space reasons here we will describe only one use case out of the dozen use cases evaluated.

Fraud Detection. The goal of this use case is the identification of fraudulent clicks generated by bots in paid online advertising. Starting from a dataset describing the traffic through search engines and the related clickstreams, clustering and outlier detection algorithms are applied to determine a list of fraudulent IPs. The resulting data to be visualized describe the total number of clicks originated from the IPs of each country during 10 min slots of a single day. The visualization context declared by the users is

$$c(1) = \text{Trend}, \quad c(2) = \text{Filter}, \quad c(3) = \text{Lay},$$
$$c(4) = \text{n-dimensional}, \quad c(5) = \text{High}, \quad c(6) = \text{Ordinal}, \quad c(7) = \text{Ratio}$$
$$(1 \overset{c}{\sim} 3 \overset{c}{\sim} 4) \overset{c}{\succ} (2 \overset{c}{\sim} 5 \overset{c}{\sim} 6 \overset{c}{\sim} 7)$$

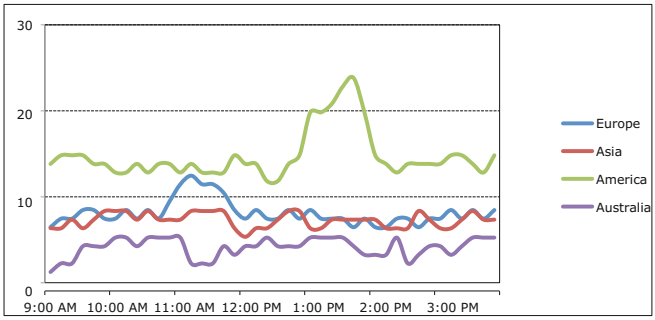

Fig. 3. Data visualization using a multiple line chart for the fraud detection use case

The skyline for the three top-priority coordinates of c includes multiple line chart, stacked line chart, and multiple line chart. However, when the remaining four coordinates are considered, only multiple line chart is left in the skyline (its suitability scores are neutral for filter, and fit for all other coordinate values). The resulting visualization is shown in Fig. 3, and was declared by the users to perfectly fit their needs. □

Our future work mainly concerns the translation from the procedural to the deployment level of the TOREADOR platform. Specifically, one the user has chosen her preferred chart (e.g., bubble chart) among those suggested, and based on the types of the single (independent and dependent) data variables to be visualized, the system will support the user in mapping each data variable onto a specific dimension of the chart (e.g., first variable onto X axis, second variable onto Y axis, third variable onto bubble color, fourth variable onto bubble size).

References

1. Abela, A.: Advanced Presentations by Design. Pfeiffer, San Francisco (2008)
2. Ardagna, C., Bellandi, V., Damiani, E., Bezzi, M., Hebert, C.: A model-driven methodology for big data analytics-as-a-service. In: Proceedings of the IEEE International Congress on Big Data, Honolulu, Hawaii (2017)
3. Bertin, J.: Semiology of Graphics. Esri Press, Redlands (1983)
4. Börner, K.: Atlas of Knowledge: Anyone Can Map. MIT Press, Cambridge (2015)
5. Chandra, J., Madhu Shudan, S.: IBA graph selector algorithm for big data visualization using defense data set. Int. J. Sci. Eng. Res. **4**(3), 1–7 (2013)
6. Dadzie, A.S., Rowe, M.: Approaches to visualising linked data: a survey. Semant. web **2**(2), 89–124 (2011)
7. Few, S.: Show Me The Numbers: Designing Tables and Graphs to Enlighten. Analytics Press, Berkeley (2004)
8. Kano, N., Nobuhiku, S., Fumio, T., Shinichi, T.: Attractive quality and must-be quality. J. Jpn. Soc. Qual. Control **14**(2), 39–48 (1984)
9. Keim, D.: Exploring big data using visual analytics. In: Proceedings of the EDBT/ICDT Workshops (2014)
10. Keim, D.A.: Information visualization and visual data mining. IEEE Trans. Vis. Comput. Graph. **8**(1), 1–8 (2002)
11. Kleppe, A., Warmer, J., Bast, W.: MDA Explained - The Model Driven Architecture: Practice and Promise. Addison-Wesley, Boston (2003)
12. Marty, R.: Applied Security Visualization. Addison-Wesley, Boston (2009)
13. Mindolin, D., Chomicki, J.: Preference elicitation in prioritized skyline queries. VLDB J. **20**(2), 157–182 (2011)
14. Peña, O., Aguilera, U., López-de-Ipiña, D.: Exploring LOD through metadata extraction and data-driven visualizations. Program **50**(3), 270–287 (2016)
15. Russom, P.: Big data analytics. Technical report, TDWI Best Practices Report (2011)
16. Shneiderman, B.: The eyes have it: a task by data type taxonomy for information visualizations. In: Proceedings of the IEEE Symposium on Visual Languages, pp. 336–343 (1996)
17. Stevens, S.S.: On the theory of scales of measurement. Science **103**(2684), 677–680 (1946)
18. Wehrend, S., Lewis, C.: A problem-oriented classification of visualization techniques. In: Proceedings of the IEEE Conference on Visualization, pp. 139–143 (1990)