On the Technical Realization of Legal Visualizations

Hans-Georg Fill

University of Vienna – Department of Knowledge and Business Engineering
A-1210 Vienna, Bruenner Strasse 72
hans-georg.fill@dke.univie.ac.at

Schlagworte: Visualization, Visual Languages, Metamodeling

Abstract: The IT-based realization of legal visualizations can be viewed from different aspects. By showing the nature and combination of these aspects current state-of-the-art visualization approaches are classified. Derived from these insights a conclusion on the use of adequate IT concepts is drawn.

1. Motivation

The generation of graphical representations in the area of legal visualization can today be supported by several technical means. After a possible first conception of the visualization by using a “paper and pencil” approach the exchange of visualizations with other parties often requires the use of electronic documents. Equally the automated or semi-automated analysis of complex visualizations (as e.g. required for an evaluation of legislative process analyses and simulations) cannot be accomplished with traditional paper formats. Therefore an appropriate technical solution has to be selected and applied to the particular visualization task. The following elaborations shall give a short overview on the aspects that have to be taken into account and present some state-of-the-art solutions for an IT-based creation.

2. Aspects of the Creation of Visualizations

When creating visualizations it can be basically differentiated between three approaches: The first approach that can be found is characterized by a *subjective*, individual nature of the creation process. It reverts to a large extent to the *experience* of the creator of visualizations and is similar to an “art and crafts” procedure. Therefore it cannot be assessed in this case from an objective point of view how the visualization is actually created, which parameters have been considered

and what is the exact meaning of the elements of the visualization[1]. With the introduction of *semi-formal* approaches these deficiencies can be partly overcome: Their main difference to the aforementioned approach is that parts of the creation process are explicitly documented in a generally understandable format. Although in several research areas of visualization this would already imply the use of mathematical notations the view that is taken here shall not be solely restricted to this proceeding. This partial documentation can thus either be described by mathematical logic or may also be expressed using other, commonly understandable and sufficiently expressive language. The third and most exact approach of describing the creation of visualizations is denoted as *formal* approach. Here, it is aimed for an unambiguous description of the whole creation process that also includes the exact definition of every part of the description itself. This can be achieved by reverting to mathematical notations such as logic or algebraic approaches.

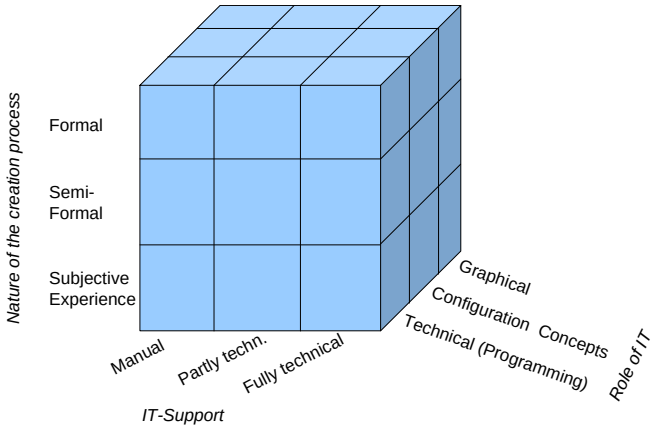

Figure 1: Combination of the Aspects for Creating Visualizations

The information technological support for the creation of visualizations can also be described by three dimensions: *Manual*, *partly technologically supported*, and *fully technologically supported*. The focus is put here on the creator of the visualization: Therefore a manual

[1] Cf. also the discussion in Burkhard, R., Is it Now Time to Establish Visualization Science as a Scientific Discipline? (2006), 10th Conference on Information Visualization, IEEE.

creation of a visualization corresponds to a realization with no or little technical support. This also encompasses the assignment of the visualization task to a third party, e.g. a visualization expert or an automatic visualization service that delivers the visualization based on a user request. When the creation process is partly technologically supported some activities are accomplished by using information technology, when the process is fully supported all activities rely on IT support.

Finally, the role of information technology in the creation process can be further detailed: Several IT-based visualizations are created in a *purely graphical* format. This means that the graphical representations contained in the visualization are either only syntactically described (in the case of vector graphics) or only delivered as a set of points (pixel-based graphics) that does not allow for a later analysis or modification of the visualization. In contrast to this approach visualizations may also be created using purely *technical*, i.e. *programming* approaches. In this case it is required to specify the visualization in a programming language as e.g. Java or C++. The advantages of this approach are that the creator can realize every type of visualization and at every level of graphical expressiveness. Furthermore, the visualizations can be later altered or even analysed using information technology. The third role of information technology can be described by the term *configuration concepts*. It lies in between the programming approach and the purely graphical approach. Its main characteristic is that the visualization is described by a model that can be altered and analyzed but that limits at the same time the visual expressiveness and thus eases the handling by the user. This type of visualization is mainly based on concepts from the area of metamodeling and visual languages. The possible combinations of the three aspects are shown in figure 1.

3. Examples for the IT-supported Creation of Visualizations

In the following several examples for this last aspect of the role of information technology shall be discussed. For the purely graphical approach a number of commercial and open-source tools exist that permit the creation of almost any type of graphical representation[2].

---

[2] Especially noteable are the open-source tools for vector graphics Inkscape (http://www.inkscape.org/) and the GNU Image Manipulation Program for pixel graphics (http://www.gimp.org/).

For the technical, programming approach it can be reverted to the abovely mentioned programming languages or to particular description languages such as SVG for vector graphics, SMIL for the authoring of interactive audiovisual presentations or Gnuplot for the depiction of diagrams[3].

For the realization of configuration concepts it is either possible to combine traditional vector graphics approaches with additional algorithms (e.g. by using SVG together with ECMAScript) or to revert to approaches from visual languages and metamodeling. Notable here are visual language descriptions using mathematical formalisms[4] as well as intuitive, symbolic approaches[5]. In the area of metamodeling it could be shown with the approach of Semantic Visualization how the assignment of visualizations to metamodels can be done based on semantic relationships between metamodels, ontologies and visualizations[6]. Thereby traditional metamodel approaches can be enhanced to further ease the creation process of visualizations.

4. Comparison of Approaches and Conclusion

A comparison of these three possibilities shows that the visual expressiveness is optimally satisfied by purely graphical or programming approaches but limited by configuration concepts. The user handling of the visualization process is comparatively easy with approaches based on configuration concepts but requires technical or in-depth application-specific knowledge in the other two cases. A model of the visualization for further automated analyses can only be provided by configuration concepts or programming approaches but only with significant additional effort for purely graphical approaches.

As a general guideline for the creation process of visualizations it can be concluded that configuration concepts should be favoured due to their high reusability and analysis capabilities, followed by vector graphics based graphical and finally programming approaches.

---

[3] SVG: http://www.w3.org/Graphics/SVG/, SMIL: http://www.w3.org/Graphics/SVG/, Gnuplot: http://www.gnuplot.info/

[4] Cf. for example: Stapleton, G. et al., Enhancing the Expressiveness of Spider Diagram Systems (2006), Workshop on Visual Languages, KSI, Grand Canyon.

[5] Cf. Favetta, F.. et al., Visualization and Exploration of Landscape Architecture Project Information (2006), Workshop on Visual Languages, KSI, Grand Canyon.

[6] Cf. Fill, H.-G., Visualization for Semantic Information Systems (2006), Doctoral thesis, University of Vienna.