Integrating Semantics into the Visualization Process*

Sebastián Escarza[1], Martín L. Larrea[1], Dana K. Urribarri[1], Silvia M. Castro[1], and Sergio R. Martig[1]

1 **Laboratorio de Investigación y Desarrollo en Visualización y Computación Gráfica (VyGLab)**
Departamento de Ciencias e Ingeniería de la Computación (DCIC)
Universidad Nacional del Sur (UNS)
Bahía Blanca, Buenos Aires, Argentina
{se,mll,dku,smc,srm}@cs.uns.edu.ar

Abstract

Most of today's visualization systems give the user considerable control over the visualization process. Many parameters might be changed until the obtention of a satisfactory visualization. The visualization process is a very complex exploration activity and, even for skilled users, it can be difficult to arrive at an effective visualization. We propose the construction of a visualization prototype to assist users and designers throughout the stages of the visualization process, and the integration of such process with a reasoning procedure that allows the configuration of the visualization, based on the entailed conclusions. We are working on a formal representation of the Visualization field. We aim to establish a common visualization vocabulary, include the underlying semantics, and enable the definition of visualization specifications that can be executed by a visualization engine with ontological support. An ontological description of a visualization should be enough to specify the visualization and, thus, to generate a runtime environment that is able to bring that visualization to life. The visualization ontology defines the vocabulary. With the addition of inference rules to the system, we can derive conclusions about visualization properties that allow to enhance the visualization, and guide the user throughout the entire process toward an effective result.

1998 ACM Subject Classification I.3, I.3.6, I.2.4

Keywords and phrases semantic driven visualization, ontology, visualization, knowledge representation

Digital Object Identifier 10.4230/DFU.Vol2.SciViz.2011.92

1 Introduction

Visualization tools have strong dependency on the visualized data domain. This leads to a great heterogeneity in the field. If we compare, for example, a flow visualization with a graph visualization, we find differences between them in many respects. Data topologies in flow visualization are usually highly structured regular grids whereas in graph visualization are graphs. Data items in flow visualization have spatial locations that should be represented in the visual representation, but in graph visualization the layout of elements is a freedom

* This work was supported in part by the PGI 24/N020 and 24/ZN19, Secretaría General de Ciencia y Tecnología, Universidad Nacional del Sur, Bahía Blanca, Argentina.

Scientific Visualization: Interactions, Features, Metaphors. *Dagstuhl Follow-Ups, Vol. 2.*
Editor: Hans Hagen; pp. 92–102
Dagstuhl Publishing
Schloss Dagstuhl – Leibniz Zentrum für, Germany

degree whereas the node-link structure is the key aspect to represent. As a result of such heterogeneity, different approaches need to be applied depending on the case.

However, there are many aspects in common across the visualization techniques. Data derivation, cleaning and filtering, and layout algorithms, are a few examples of tasks that are performed in similar ways in almost every visualization tool. A key issue to unify the Visualization area is the identification of such common aspects in order to perform developments that enable the reuse along the visualizations. Many efforts exist to gather these aspects, such as [25], [4], [26], [7], [3] and our Unified Visualization Model (UVM) [17].

The UVM is a reference model that gives users and designers a unique mental model in terms of which express their needs. It defines a theoretical framework for describing the intermediate states and transformations of the data, from its raw origin in the application domain to the final view construction. Additionally, the UVM enables the explicit definition of user interactions through the definition of tasks, basic interactions, and low level operators and operands.

The UVM and the other proposals are steps toward the foundation of a Visualization Base Theory and an extensive work has been done to unify the vocabulary in the field and overcome the heterogeneity. However, achieving an effective visualization is still more an art than a science. Although, with a framework for describing each aspect, each technique, and each possible choice in Visualization, how to configure the visualization process to obtain an effective visualization is a very difficult task, even for skilled users. Visualization systems can give the user considerable control over the visualization process. This freedom makes it difficult to choose a proper configuration that allows to obtain the desired results.

To overcome this difficulty, we propose to integrate semantics into the visualization in order to guide the user in the selection of the visualization parameters. We found necessary to include semantics for describing the data, the visual representation, the interactions, and the visualization process itself, plus a reasoning mechanism to derive the features of an effective visualization from such semantics. In consequence, our proposal relies upon three major components: a description about how each visualization technique represents each aspect of the data (i.e. a visualization ontology), a specification about the meaning of the raw data (i.e. a visualization-oriented data taxonomy), and a mechanism that performs the selection of the most suitable visual parameters to be applied to a concrete scenario (i.e. a rule system for semantic-based visualization).

The general objective of our proposal is the integration of the visualization process with a reasoning procedure that allows the configuration of the visualization, based on the entailed conclusions. In particular, we are working to obtain an operational visualization prototype with semantic support that allows to assist users and designers throughout all the stages of the visualization process. As part of our ongoing work, we are defining a visualization ontology based on the UVM, and formalizing a data taxonomy. Additionally, we are working on a preliminary set of rules that enable the entailment of conclusions that suggest to the user the selection of an appropriate configuration for an effective visualization.

This work is organized as follows. In Section 2, we present some related work on integrating semantics in visualization. Next, in Sections 3, 4, and 5, we describe the main components of our ongoing work. Finally, we give some closing remarks.

2 Related Work

The integration of semantics in visualization has evidenced an ever increasing interest in the community. Much work has been done to formalize the visualization topics and define visualization reference models. In [8], the authors argument the need for increasing the rigourousness in visualization descriptions to explicitly define a visualization ontology, and give some clues about how it can be achieved. In this sense, data and/or task oriented taxonomies such as [25], [4], [7], [3], and the UVM [17] are partial views of the semantic shared conceptualization for Visualization that must be explicitly modeled by every valid visualization ontology.

There are some good examples of how semantic information can be integrated into visualization tasks ([32], [15], [18], and [31]). However, in all these examples, the role of the semantics is to improve the integration, querying, and description of the data in the visualization; in none of these cases the semantics associated with the data is used to create the visualization.

Despite that, there are cases where semantics is used, although limited in some way, for aiding in the visualization creation. In [13], it is presented a semi-automatic visualization assistant that helps users to construct perceptually sound visualizations for large multidimensional datasets. In [10], it is proposed a framework that uses context information and a set of rules to automatically select a suitable visualization method. In [16], the authors present a method for aiding in the visualization pipeline design. A database of pipeline configurations is used to properly complete the configuration of the user's pipeline. In [9], the authors combine a domain ontology with a visual representation ontology to automatically select a proper visualization for web data.

Several efforts to define generalized visualization ontologies have also been made. A seminal paper on this matter is [2]. In that work, a top level ontology is outlined and future tendencies are given. Moreover, authors in [6] discuss the role of the information and the knowledge in visualization and identify possible trends. In [14] a graphics ontology representing the semantics for a 3D-scene graph is proposed to define the semantics of graphical primitives, using a very similar approach to ours.

Additionally, there are semantic specifications for particular visualization aspects such as the user domain data classification, the visual representation, and the visual mapping. Some examples are size-based data classifications [30, 29], a taxonomy for visualization algorithms that is based in assumptions over their inputs [28], the characterization of visual variables to represent visual representations at a higher level of abstraction [5], the use of fuzzy logic semantics to replace the traditional transfer function setup in illustrative volume rendering [21], and a specific semantic model created by a machine learning mechanism that uses representative dataset collections as training sets [24].

Finally, we can mention a rule-based related work. PRAVDA (Perceptual Rule-Based Architecture for Visualizing Data Accurately) [1] is a rule based architecture for assisting the user in making choices of visualization color parameters. This architecture provides appropriate choices for visualization, based on a set of underlying rules [23, 22], which are used to select a colormap.

3 Toward a Visualization Ontology

An ontology for visualization is the first necessary step to enable the building of visualization tools with underlying ontological support. An ontology is a formal, explicit specification of a shared conceptualization [12]. That shared conceptualization is given in visualization by the reference models discussed earlier, which provide the foundations and the necessary theoretical background. Our purpose is to formalize the theoretical framework given by the UVM and make it explicit and manipulable by a software platform automatically. To

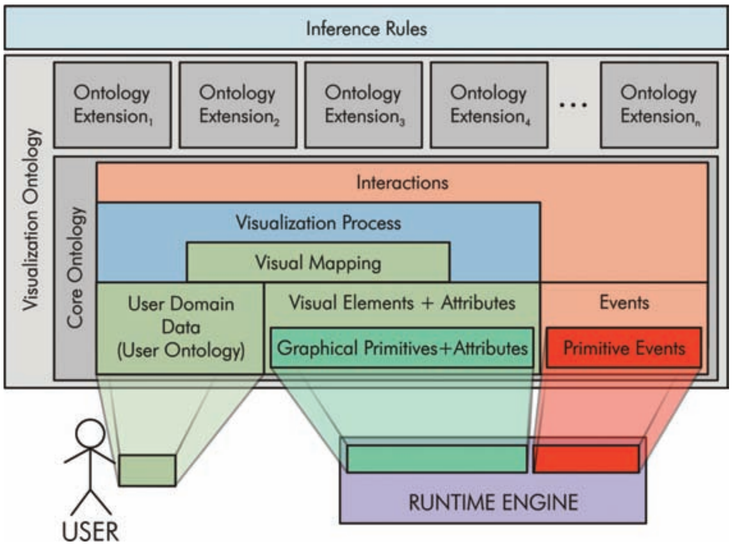

Figure 1 The ontology architecture. The figure shows the main components of the core ontology, the ontology extensions, and the inference rules. Users provide their own domain ontology and the selected runtime engine adds its primitives.

emphasize the shared nature of our ontology, we have adopted the Ontology Web Language (OWL) [20] as the language to define it. OWL is the W3C standard ontology language for the Semantic Web initiative and has been under active development for about six years. The utilization of a heavily supported standard facilitates the sharing and the interoperability of our proposal, and enables us to exploit the vast set of tools that have been implemented. In particular, we are interested in the definition of an OWL DL visualization ontology (i.e. the subset of OWL whose semantics is based in Description Logics), in order to keep the reasoning decidable.

The design of the ontology follows a modular approach (see Fig. 1). It has a core composed of a minimal set of basic concepts and relationships that can be extended by concepts and relationships of higher level of abstraction. The definitions in each ontology extension are based directly or transitively on the core definitions. Ontology extensions have a particular purpose and may be used depending on the user needs. In this way, only the extensions that model aspects of interest for the current application are used. This simplify the reasoning process, facilitates the decision making about the visualization configuration, and allows to focus on the outstanding topics of Visualization required for such application. This setting of core definitions plus their respective extensions leads to a layered and more extensible architecture.

From a static perspective, a visualization can be described in terms of the data in the user domain, the visual representation, and the visual mapping. From a dynamic perspective, the visualization can be perceived as a process of transformation that takes the user domain data as input, processes it to get the visual representation, and can be affected by the user interactions. The core of the ontology supports these both perspectives and it is presented in the following subsections (see Fig. 1).

3.1 User Domain Data Representation

Data in the user domain and its relevant properties are characterized by a domain ontology provided by the user. This ontology is imported and its concepts and relationships are made available to establish visual mappings from them.

Users are the experts in their domains. They know all the subtleties of the knowledge area that they want to visualize. Thus, it is reasonable that they provide their own representation for that area. Additionally, for a particular application domain, many domain ontologies may exist differing in how they represent such a domain, its purpose, level of detail, specificity, ontology commitment, etc. We prioritize the maximal flexibility in our design by allowing users the inclusion of their most adequate OWL domain ontology to represent their data and metadata. By requiring an OWL DL user ontology, we enable the use of all the OWL DL constructors, and ensure a successful integration between user and visualization ontologies.

3.2 The Visual Representation

The definition of the visual representation must describe the spatial substrate (how the elements are arranged in the visual representation space), the visual elements, and their attributes. Concepts for *space* and *geometric transformation* are used to describe the visualization layout.

In order to enable the automatic execution of the visualization represented by an ontological description, the visualization system must be able to perform the rendering of the elements represented by the involved concepts. But rendering engines may have different rendering capabilities. To overcome this issue without losing flexibility in our design, our system allows modular runtime engines for rendering and event handling. Each runtime engine provides concepts describing its supported graphical primitives (e.g., vertex based primitives, nurbs, etc) with its attributes (e.g., color, opacity, shader models, etc), the supported transformations (e.g., translations, rotations, etc), the user events that can be handled (e.g., mouse click, mouse drag, keystrokes, etc.), and every platform dependent aspect. These primitive concepts do not rely on any other concepts and they have a semantics given by how they are processed at runtime. Primitive concepts can also be combined to define derived concepts for describing more complex scenes (see Fig. 1).

3.3 The Visual Mapping and The Visualization Process

Concepts for the data in the user domain and the visual representation are combined to define the visual mapping. From a static viewpoint, the visual mapping consists mainly in ontological relationships that associates data items and attributes in the user domain with graphical elements and attributes in the visual representation. In many cases, the mapping follows a data structure or topology, or some data values are mapped specifically depending on their datatype.

However, the visual mapping can be the result of a complex data transformation. From a dynamic viewpoint, the visualization can be perceived as a process that takes the data in the user domain (i.e., the input data or raw data) and process it, giving back the view as a result. This processing is decomposed in a visualization network or pipeline where the intermediate data stages, the transformations, and the interconnections between them are described (see Fig. 2).

The visualization ontology defines concepts and relationships to describe the visualization stages and transformations. In each transformation, a formal description of the calculations performed there, and the conditions that must hold to carry out them, are provided.

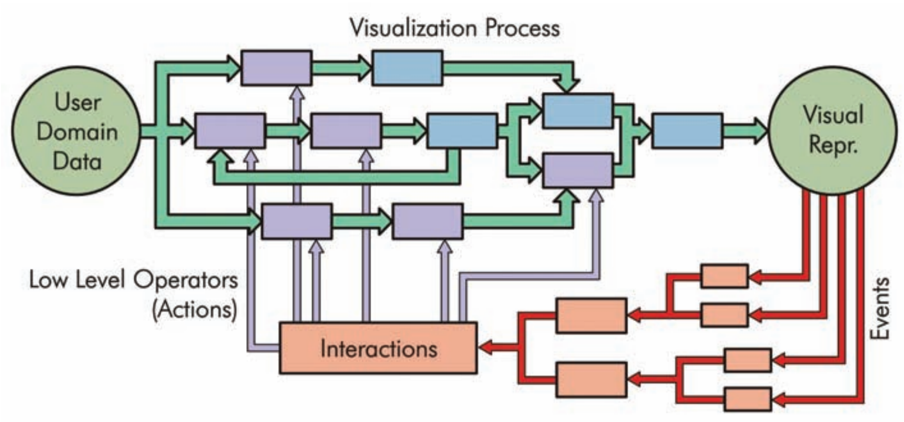

Figure 2 The visualization process and the interactions. The figure shows a visualization network that obtains a visual representation from the input data in the user domain, and the interaction feedback loop. A set of events are combined to characterize the basic interactions. These interactions affect the visualization network through low level operators provided in each stage.

These descriptions involve the specification of control structures such as the sequence, the conditionals, etc. Additionally, each stage provides low level operators that enables its reconfiguration. To represent these aspects the Process Ontology of OWL-S can be used. OWL-S, the Semantic Markup for Web Services [19], is an OWL ontology that describes semantic web services. The description of the behavior of such web services could be useful to describe the semantics of the stages and transformations in the visualization process.

The formal description of the visualization process enables its dynamic reconfiguration, the re-execution of the proper stages in response to interactions, and the ability to reason about how to connect the available stages and transformations to obtain the desired visual representation.

3.4 Interactions in Visualization

After the visual representation is generated, the user can interact with the visualization triggering background processes that recalculates and re-executes several parts of the visualization. Then, the user receives some feedback from the visualization and the interaction cycle is repeated again and again.

Each interaction in the visualization starts when the user produces an event on the visual representation. The events that the visualization can handle are associated to basic interactions. These interactions reconfigure the visualization process through low level operators defined in each stage of the visualization network (see Fig. 2).

Ontologically speaking, the basic interactions are mappings from events to low level operators. Events that can be handled by the runtime engine are provided as a set of concepts. Low level operators are part of the set of concepts used to define the visualization process. Hence, the basic interactions are represented in the ontology by relationships that map those two set of concepts. In this way, each time the user produces the set of events that triggers some interaction, the runtime engine will report them, and then, the semantic information about such interaction will be used to perform the appropriate reconfiguration of the visualization network represented by the ontology.

3.5 Extensions of the Visualization Ontology

In the previous subsections the main parts of the ontology core have been discussed. The core concepts and relationships offer the ability to model the key elements of the visualization, but they present a low abstraction level for the user data model and the underlying software platform. In visualization, many decisions take into account more abstract information. Concepts such as shape, transfer function, visualization technique, among others, are more appropriate to reason and decide the optimal way to represent some particular dataset.

Extensions to classify datasets, handle transfer functions, manipulate color spaces and transformations, and define information channels in visualization (such as shape, opacity, color, etc.) can be defined by combining core concepts and used independently only when they are required. For example, a user interested in transfer functions can use such extensions without importing others. In this way, there is no need to perform the reasoning over the full visualization ontology. Only the relevant concepts are considered to assist the user. Additionally, since extensions are defined in terms of core concepts, and such concepts have support in the runtime engine, the extensions automatically acquire runtime support contributing to the automatic execution of the visualization represented by the ontology.

4 Building Up a Visualization-Oriented Data Taxonomy

Data classification is the categorization of data for its most effective and efficient use. Data can be classified according to any criteria. For example, it can be divided according to its topical content, file type, operating platform, average file size, creation date, etc. We need a suitable data classification to integrate in the UVM and in our ontology. An explicit description of the data provides the semantics necessary to enable the reasoning and the selection of effective visualization techniques to visualize it.

A visualization-oriented data taxonomy should consider not only the size in bytes of the dataset but also issues like:

- the amount of items, to determine how important it is the visual scalability of the technique to be used,
- the existing relationships among items, to determine how important is to use a technique that displays explicitly such relationships (e.g. a graph-based technique),
- the amount of attributes, to determine how important is to use a multidimensional visualization technique,
- and finally, the amount of different objects to be represented and their complexity, to determine the necessary interactions to explore and analyze the data.

These aspects are key features to select the most adequate technique to visualize a dataset. A dataset taxonomy also helps in the selection of the graphical elements and their attributes. For example, the techniques to visualize high-dimensional textual data are very different from the techniques required to visualize a vector field, and the graphical elements to represent tuples are very different from those used to represent vectors. Figure 3 shows the UVM's transformations where the taxonomy-driven selections are performed.

In the context of our Visualization Ontology, we need to provide concepts and relationships to enable the formal definition of visualization-oriented data taxonomies. This implies the description of the metrics needed to characterize the datasets accordingly to the aspects previously mentioned. Based on such metric values, it is possible to define hierarchies for dataset classification. Then, for each category, a set of proper visualization techniques can be made available ensuring an adequate visual representation. All these concepts and

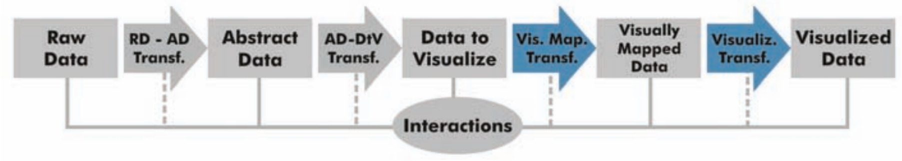

Figure 3 The role of a visualization-oriented data taxonomy in the context of the UVM's pipeline. The taxonomy is involved in the selection of how the elements and its attributes are represented (Visual Mapping Transformation), and it is essential to choose an appropriate visualization technique (Visualization Transformation).

relationships will be integrated into the visualization ontology through one or more extensions created for this particular purpose.

5 Fulfilling Semantics Through Rules: Semantic-Based Visualization

An ontology describes the meaning of the domain that it represents. But such ontology cannot be useful if that meaning is not exploited in all its extent. An OWL specification of concepts and relationships only describe the axioms of the knowledge base. It has information about what holds and what does not, and is able to be queried for ontology consistency, concept satisfiability, concept subsumption, and instance checking. These operations are valuable for validation purposes and provide richer "object oriented" axiomatic descriptions. However, to assist the user in his search for an effective visualization, it is necessary to include inference rules into the system. This addition will allow to derive conclusions about visualization properties that will be used to enhance the visualization and guide the user throughout the process.

Inference rules are able to represent the way in which the system can derive new information. They are combined with the axiomatic descriptions in the visualization ontology to entail new facts that can be queried and used to suggest how to visually represent some dataset. Rules also allow the use of variables in their specification giving the ability to infer through a pattern matching based mechanism. Such a mechanism, not present at ontological level, is essential to bind visualization parameters, derive new information, and increase the overall expressive power. Additionally, inference rules allow a richer information management by making reference to concepts of a higher abstraction level, i.e., concepts described by specific extensions of the ontology. For example, rules stating that some visualization technique is appropriate for some particular kind of dataset, are rules that make reference to the concepts for the visualization-oriented data taxonomy discussed before.

From combining the semantics given by the visualization ontology with a carefully chosen set of inference rules, the available family of queries become more according to our objective: assist the user to get an effective visualization. For example, one could ask the system if some technique is suitable for the input data, if some color scheme is appropriate to make evident the differences between some attributes, if the visual mapping shows appropriately the presence of some data correlation, etc.

In order to include rules into our proposal in an elegant and consistent manner, we are analyzing the use of a rule language that has a good integration with OWL. Unfortunately, the rule language for the Semantic Web is not standardized yet. The most encouraged proposal under consideration is the Semantic Web Rule Language (SWRL) [27]. This language adds rules written in RuleML (an XML-based language for the denotation of rules) to OWL DL. The result is a language based in horn clauses which has the expressive power of a First Order

Logic, but, in consequence, it is undecidable. Another proposal is the use of Description Logics Programs [11], a combination of OWL DL with the decidable portion of a horn-like rule language. This last proposal emphasizes decidability over expressiveness resulting in a more limited but decidable language.

Although the rule support for the Semantic Web is not mature yet, the tendency indicates that some kind of horn-like clauses will be used to enable full reasoning over the web. In this context, we expect that our proposal also make use of such a kind of rules.

6 Final Remarks

We have presented a proposal to integrate semantics into visualization to aid users to obtain an effective visualization. The semantic specification is based on the definitions given by a visualization reference model (the UVM), and contributes to the formalization of the visualization base theory. Such specification requires a visualization ontology, plus a set of inference rules to entail information to properly configure the visualization. The ontology was designed following a modular design that separates the core ontology describing basic concepts, from the higher abstraction level concepts defined in particular purpose extensions. Additionally, the ontology design distinguishes between primitive and derived concepts to provide runtime support for the visualization represented by the ontology.

Several aspects compose the semantic description for a visualization. Static aspects such as the data in the user application domain, the visual representation, and the visual mapping, are combined with concepts describing the stages and transformations of the visualization process and the interaction feedback provided by a visualization tool. All this information is gathered and used to establish valid conclusions about how to configure a visualization that represents adequately and accurately the input dataset.

In addition to the core visualization ontology, we have outlined an ontology extension to describe a dataset taxonomy, and have described the use of inference rules in the reasoning process. These two topics are essential to make possible, to our system, the selection of a suitable visualization technique as a function of the input dataset characteristics.

This work is the first of a long series of steps toward the construction of a visualization system that helps users represent their data in an effective manner. We are currently working in the design and implementation of a service-oriented visualization model that will extend the UVM through the inclusion of semantic information and reasoning. Currently, we are working on the formalization of the UVM and its ontological representation, the definition of the core ontology, the development of a service-oriented version of the UVM, the characterization of the input data and an inference model based on semantic reasoning. In the future, we expect to complete the ontology definition with specific extensions to handle data categorization, color management, and information channels among other topics. Also, we pursue the implementation of a modular prototype for a concrete evaluation of the exposed topics, and the validation of the proposed formalisms by the Visualization community.

References

1 L. D. Bergman, B. E. Rogowitz, and L. A. Treinish. A rule-based tool for assisting colormap selection. In *VIS '95: Proceedings of the 6th conference on Visualization '95*, page 118, Washington, DC, USA, 1995. IEEE Computer Society.

2 K. W. Brodlie, D. A. Duce, D. J. Duke, and al. Visualization Ontologies: Report of a Workshop held at the National e-Science Centre. Technical report, e-Science Institute, 2004.

3 Ken Brodlie and Nurul Mohd Noor. Visualization Notations, Models and Taxonomies. In Ik Soo Lim and David Duce, editors, *EG UK Theory and Practice of Computer Graphics (2007)*, pages 207–212, Bangor, United Kingdom, 2007. Eurographics Association.

4 S. K. Card and J. Mackinlay. The structure of the information visualization design space. In *INFOVIS '97: Proceedings of the 1997 IEEE Symposium on Information Visualization (InfoVis '97)*, page 92, Washington, DC, USA, 1997. IEEE Computer Society.

5 M. S. T. Carpendale. Considering Visual Variables as a Basis for Information Visualisation. Technical report, University of Calgary, Calgary, AB, 2003.

6 Min Chen, David Ebert, Hans Hagen, Robert S. Laramee, Robert van Liere, Kwan-Liu Ma, William Ribarsky, Gerik Scheuermann, and Deborah Silver. Data, information, and knowledge in visualization. *IEEE Comput. Graph. Appl.*, 29(1):12–19, 2009.

7 Ed H. Chi. A taxonomy of visualization techniques using the data state reference model. In *Proceedings of the IEEE Symposium on Information Visualization (InfoVis'00)*, pages 69–75. IEEE Computer Society Press, 2000.

8 David J. Duke, Ken W. Brodlie, David A. Duce, and Ivan Herman. Do you see what i mean? *IEEE Computer Graphics and Applications*, 25(3):6–9, 2005.

9 O. Gilson, N. Silva, P.W. Grant, and M. Chen. From web data to visualization via ontology mapping. *Computer Graphics Forum*, 27:959–966(8), May 2008.

10 Maria Golemati, Constantin Halatsis, Costas Vassilakis, Akrivi Katifori, and University of Peloponnese. A context-based adaptive visualization environment. *Information Visualisation, International Conference on*, 0:62–67, 2006.

11 Benjamin N. Grosof, Raphael Volz, Ian Horrocks, and Stefan Decker. Description logic programs: Combining logic programs with description logic. In *WWW 03 Proceedings of the 12th international conference on World Wide Web*, pages 48–57. ACM, 2003.

12 Thomas R. Gruber. A translation approach to portable ontology specifications. *Knowledge Acquisition*, 5(2):199–220, June 1993.

13 Christopher Healey, Robert St. Amant, and Jiae Chang. Assisted visualization of e-commerce auction agents. In *In Proceedings graphics interface 2001*, pages 201–208, 2001.

14 Evangelos Kalogerakis, Stavros Christodoulakis, and Nektarios Moumoutzis. Coupling ontologies with graphics content for knowledge driven visualization. In *VR '06: Proceedings of the IEEE conference on Virtual Reality*, pages 43–50, Washington, DC, USA, 2006. IEEE Computer Society.

15 V Kashyap, K-H Cheung, D Doherty, M Samwald, MS Marshall, J Luciano, S Stephens, I Herman, and R Hookway. An ontology-based approach for data integration - an application in biomedical research. In J Cardoso, M Hepp, and M Lytras, editors, *Real-world Applications of Semantic Web Technology and Ontologies*, Semantic Web and Beyond. Springer-Verlag, Heidelberg, 2008.

16 David Koop. Viscomplete: Automating suggestions for visualization pipelines. *IEEE Transactions on Visualization and Computer Graphics*, 14(6):1691–1698, 2008.

17 Sergio Martig, Silvia Castro, Pablo Fillottrani, and Elsa Estvez. Un modelo unificado de visualización. In *Proceedings of the IX Congreso Argentino de Ciencias de la Computación*, pages 881–892, 2003.

18 Trong Dung Nguyen, Tu Bao Ho, and DucDung Nguyen. Data and knowledge visualization in knowledge discovery process. In *VISUAL '02: Proceedings of the 5th International Conference on Recent Advances in Visual Information Systems*, pages 311–322, London, UK, 2002. Springer-Verlag.

19 OWL. OWL-S: Semantic Markup for Web Services, Last visited on September 2009. `http://www.w3.org/Submission/2004/SUBM-OWL-S-20041122/`.

20 OWL. OWL Web Ontology Language Overview, Last visited on September 2009. `http://www.w3.org/TR/owl-features/`.

21 Peter Rautek, Stefan Bruckner, and Eduard Groller. Semantic layers for illustrative volume rendering. *IEEE Transactions on Visualization and Computer Graphics*, 13(6):1336–1343, 2007.

22 B. E. Rogowitz and L. A. Treinish. Using perceptual rules in interactive visualization. In B. E. Rogowitz and J. P. Allebach, editors, *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series*, volume 2179 of *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series*, pages 287–295, May 1994.

23 Bernice E. Rogowitz and Lloyd A. Treinish. Data structures and perceptual structures. In Jan P. Allebach and Bernice E. Rogowitz, editors, *Human Vision, Visual Processing, and Digital Display IV, Volume 1913, Number 1*, pages 600–612. SPIE, 1993.

24 Christof Rezk Salama, Maik Keller, and Peter Kohlmann. High-level user interfaces for transfer function design with semantics. *IEEE Transactions on Visualization and Computer Graphics*, 12(5):1021–1028, 2006.

25 Ben Shneiderman. The eyes have it: A task by data type taxonomy for information visualizations. *IEEE Symposium on Visual Languages*, 0:336, 1996.

26 Chris Stolte and Pat Hanrahan. Polaris: A system for query, analysis and visualization of multi-dimensional relational databases. In *INFOVIS '00: Proceedings of the IEEE Symposium on Information Vizualization 2000*, page 5, Washington, DC, USA, 2000. IEEE Computer Society.

27 SWRL. SWRL: A Semantic Web Rule Language Combining OWL and RuleML, Last visited on September 2009. `http://www.w3.org/Submission/SWRL/`.

28 M. Tory and T. Moller. Rethinking visualization: A high-level taxonomy. In *Information Visualization, 2004. INFOVIS 2004. IEEE Symposium on*, pages 151–158, 2004.

29 Antony Unwin, Martin Theus, and Heike Hofmann. *Graphics of Large Datasets: Visualizing a Million (Statistics and Computing)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006.

30 E. Wegman. Huge data sets and the frontiers of computational feasibility. *Journal of Computational and Graphical Statistics*, 4(4):281–295, 1995.

31 Z. M. Weng and D. Bell. Integrating visual ontologies and wavelets for image content retrieval. In *DEXA '98: Proceedings of the 9th International Workshop on Database and Expert Systems Applications*, page 379, Washington, DC, USA, 1998. IEEE Computer Society.

32 Zhao Xu, Huajun Chen, and Zhaohui Wu. Applying semantic web technologies for geodata integration and visualization. In Jacky Akoka, Stephen W. Liddle, Il-Yeol Song, Michela Bertolotto, Isabelle Comyn-Wattiau, Samira Si-Said Cherfi, Willem-Jan van den Heuvel, Bernhard Thalheim, Manuel Kolp, Paolo Bresciani, Juan Trujillo, Christian Kop, and Heinrich C. Mayr, editors, *ER (Workshops)*, volume 3770 of *Lecture Notes in Computer Science*, pages 320–329. Springer, 2005.