A Reference Model for Adaptive Visualization Systems

Kawa Nazemi, Christian Stab, and Arjan Kuijper

Fraunhofer Institute for Computer Graphics Research,
Fraunhoferstrasse 5, 64283 Darmstadt, Germany
{kawa.nazemi,christian.stab,arjan.kuijper}@igd.fraunhofer.de

Abstract. One key issue of both Information Visualization as well as Adaptive User Interfaces is information overload. While both disciplines have already devised well performing algorithms, methods and applications, a real merging has not taken place yet. Only a few attempts bring the surplus values of both disciplines together, whereas a fine-grained investigation of visualization parameterization is not investigated. Today's systems focus either on the adaptation of visualization types or the parameterization of visualizations. This paper presents a reference Model for Adaptive Visualization Systems (MAVS) that allows the adaptation of both the visualization type and the visualization parameterization. Based on this model, a framework for the adaptive visualization of semantics data will be derived. A use case describing the interaction with an "adaptive visualization cockpit" covering different visualization metaphors concludes the paper.

Keywords: Adaptive Visualization, Information Visualization, Intelligent Visualization, Visualization Reference Model, Ontology Visualization.

1 Introduction

Information overload is a well-known phenomenon in the current information age [1]. Different research areas face this phenomenon with their own methods and try to reduce the interaction cost [2], cognitive overload [3] and information amount, using content- and collaborative-based recommendation techniques. The community of intelligent and adaptive systems investigates this problem with novel and promising algorithms and techniques for user-, content- and activity analysis and adaptation. On the other hand, the Information Visualization (IV) community develops appropriate techniques for presenting information graphically.

For the community of IV, the way from data-oriented visualization to a more human-centered information presentation plays a key role. In 2007, one of the ten main challenges for IV was the inclusion of semantics or contexts [4]. In the following year, especially the human as an implication and decision factor for IV was placed in the foreground of the research [1]. The increased involvement of user's intentions and preferences in the forming process of IV is also noticeable in the challenges and scopes of Visual Analytics in 2008, where the adaptation of IV systems was proclaimed. Thus, one of the most important challenges is the development of "novel interaction algorithms incorporating machine recognition of the actual user intent and

J.A. Jacko (Ed.): Human-Computer Interaction, Part I, HCII 2011, LNCS 6761, pp. 480–489, 2011.

appropriate adaptation of main display parameters such as the level of detail, data selection, etc. by which the data is presented" ([5], p. 162).

Researchers in the field of Adaptive Systems [6] represent a similar position and are increasingly recognizing the importance of the usage of visualization techniques [7]. This research area disposes of a comprehensive pool of methods, systems and algorithms for recognizing and analyzing user related information. With these methods Adaptive User Interfaces (AUIs) facilitate the handling with complex information and support users during their work process [8]. Different existing systems e.g. intelligent help systems [9], personalization of web page navigations [10] or command line proposal lists [11] are already using these methods and tailor the user interface to the given usage context. These methods and systems do mainly not refer to bridge the gap between IVs and AUIs. Actually only a few attempts try to bring the areas of IV and AUIs together [12]. These approaches focus either on the adaptation of the visualization type, using a single set of parameters [13] or on the adaptation of visual parameters of a single visualization type [7]. Despite the research in both areas, a system or a reference model for adaptation of visualization types and their visual parameters could not be found. Further the aspect of composing multiple visualizations [14] in adaptive systems is not investigated yet.

In this paper we introduce a reference Model for Adaptive Visualization Systems (MAVS) that provides the opportunity to adapt both the visualization type as well as visual parameters. Beside a detailed description of the reference model we present an implementation of the MAVS as a framework. Finally, we introduce an application example that not only shows the adaptation based on an example but also opens new research questions for the adaptation of visualizations.

The remaining paper is structured as followed: In the next section, the related work is presented in three sections: Reference models, frameworks and applications. Afterward the reference model is described followed by a detailed description of its implementation. An application example of the implemented framework concludes the description of our work.

2 Related Work

The related work is sub-divided in three parts: 1) Reference Models will describe existing work in an abstract level. 2) Frameworks will describe existing work on the implementation of the reference models and 3) Applications will describe existing systems that combine Adaptive Systems and Information Visualizations.

From the IV point of view Card et al. [15] have presented a general visualization reference model, which has been applied successfully to different visualizations. This model contains different phases that are needed for mapping and transforming raw data into a visual representation. The user may control the way in which the data is transformed, or mapped to visual structures. So the reference model focuses on the creation of visualizations and does not investigate any kind of adaptations. Based on the established reference model of Card et al. different extensions exist today [5,16], which investigate different issues of visual data representation more in depth. An extension with the investigation of a visual adaptation is only investigated by Aaltonen and Lehikoinen [17], who redefined the data table and visual structures of the original model. They incorporated rules into the data table and renamed it as context table. In addition the visual structure was divided into layers (overlapping visual representations). The main difference is the usage of rules for the context. These rules consist of simple structured logical inferences and logical- and comparison operators. The aspect of a user-centered or task-based adaptation in a context-free environment (non-mobile) is not investigated.

The Information Visualization community provides a large set of frameworks for visualizing information in different ways. The most famous is Prefuse introduced by Heer et al. [18], a framework for creating dynamic visualizations of structured and unstructured data, based on the visualization reference model of Card et al. [15]. All existing frameworks support the development and creation of visualizations and are often more a programming platform than a basis for adaptive visualizations. The issue of automatic or semi-automatic adaptation is not investigated. The only framework, we could find, which considers the aspect of adaptation is e-Viz [19]. Whereas e-Viz addresses the modeling, scheduling and managing of visualization computational tasks and rendering graphics. e-Vis does not provide a framework or reference model for adaptation of visualization types and visualization parameters to meet the users' demands.

To round out the related work concrete applications will be introduced in this part. Applications are the most concrete alternatives to the described reference models and frameworks and conclude the related work section. Already different approaches have been proposed to support an automatic or semi-automatic adaptation of UIs. Cicero, for example is a component-oriented architecture [19], where a central UI adaptation manager is used. On the contrary agent-based environments [21], where UI models are transformed and rendered into different platforms, adapt user interfaces to different impact factors. The only applications that adapt the visualization to certain impact factors, e.g. user interactions or user-goals are the following applications: Gotz and Wen [13] analyze user interactions and extract behavioral-patterns that are used to recommend different visualization types (Line Graph, Fan Lens, Parallel Coordinates or Bar Charts) to the user. But a visualization type has always characteristics and parameters, e.g. color of entities, order, size, layout etc., which can and should be used for communicating designated information adaptively. On the contrary Ahn & Brusilovsky are adapting visual parameters of a single visualization type and visualize the user-specific relevance of a query [7]. In this case a single static visualization type is used to represent the searched information.

The related work could point out that the number of existing adaptive visualizations is very limited and these approaches focus either on the adaptation of the visualization type, using a single set of parameters or on the adaptation of the parameters of a given single visualization type. Despite the research in both areas, a system or a reference model for adaptation of visualization types and their parameters could not be found. Further the aspect of composing multiple visualizations [14] in adaptive systems is not investigated yet.

3 A Reference Model for Adaptive Visualization Systems

In this section, we present a Reference Model for Adaptive Visualization Systems *MAVS*. The MAVS (Fig. 1) consists of three basic Components: *Input*, *Adaptation* and *Output*.

These Components are composed by several *modules*, which are defined by interfaces and show a unique behavior to their Component. Internally, there can be different conformant implementations, allowing a very easy reuse. In the reference model, we will assume that there is a toolbox of implementations available at runtime, allowing the adaptation engine to select the best fitting one for a certain situation. Additionally these modules can be configured by *parameters*. For instance these parameters are defined by certain impact and influence factor, e.g. knowledge or behavior of the user or the structure and amount of the underlying data.

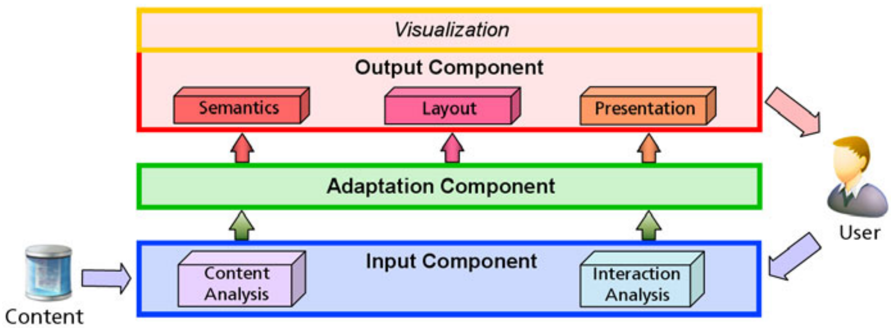

Fig. 1. Reference Model for Adaptive Visualization Systems (MAVS)

The *Input Component* receives different inputs, which are used as implication factors for the *Adaptation Component*. It consists of two different modules, for analyzing these factors: The *Content Analysis Module* prepares the underlying data for the visual presentation and extracts certain criterions and properties of the data for the Adaptation Component. The *Interaction Analysis Module* is responsible for extracting information from different sensors about the user and prepares it for an appropriate usage in the Adaptation Component.

The *Adaptation Component* receives information of the Input Component and adapts several modules of the Output Component accordingly. This includes the selection of the best fitting modules and their parameterization in the Output Component.

The *Output Component* consists of the three modules *Semantics*, *Layout* and *Presentation*. The Semantics Module represents the structure of the data transformed for visualization. Further it contains information about the amount and attributes of the data. The *Layout Module* is responsible for placement and structuring the information on screen (e.g. graph-layout). The *Presentation Component* holds certain visual parameters (e.g. color, size etc.) and provides them to the Layout Module. The instantiation of these three modules together builds visualization for the user.

4 The SemAdapt Framework

In this section, we present *SemAdapt*, a framework for adaptive visualizations based on the concepts described above. This section will focus on the three components input, output, and adaptation.

4.1 Input Component

The Input Component of SemAdapt receives two different impact factors for the visualization adaptation and parameterization: User interactions and data.

Interaction Analysis: The main input component of SemAdapt is responsible for capturing information about the individual user and to extract user information from interaction events. These interaction events appear as natural consequence of the visualization operation and are captured with different contextual information about the interaction type, the layout method and the content. Altogether, an interaction event has the form of a triple `<type, layout, data>`. For example a click with the left mouse button on a graphical representation of the element Fraunhofer produces the following triple

```
(device.mouse.button.left.click, semadapt.semap,
thing.organizations.research.appliedsciences.Fraunhofer)
```

The gathered user interactions serve as input for an instantiation of the Interaction Analysis Module that extracts information about the user behavior. Based on a quantitative analysis on different grades of abstraction, the instantiated module captures the preferences of the user according to the input device, the visualization type and the content [25]. Additionally, the module calculates predictions with the KO*/19-Algorithm and detects user activities as recurring and similar interaction sequences [26]. All this information about the user behavior is passed to the Adaptation Component of SemAdapt to select a visualization type and an individual parameterization of the output modules.

Content Analysis: Beside the user interactions, the data as input and impact factor for the adaptation is considered in SemAdapt. The semantically annotated data can contain geographic or time-dependent information, which is used to identify visualizations that are able to visualize these factors in an appropriate way. Further the amount of the data and its structure is considered. A high amount of ontological concepts forces to choose a visualization that figures the hierarchy in an abstract way and provides a higher-level of interactions with data. Beginning with a search in knowledge domains, the data is analyzed in different steps building a hierarchy and extracting the relevant attributes of data (e.g. timestamps and geographical attributes). The procedure of the data analyzing is described in [27] in detail.

4.2 Adaptation Component

The *adaptation component* is the central component of the interactive system presented in this paper. In order to control the whole system, the adaptation component has interfaces to all components and modules. It gathers information from the Input

Component and transforms it for the Output Component. The gathered user information are persisted in a three dimensional user model [28], that contains tables with preference values for the activities, visualization types and contents.

The whole adaptation process is based on the (static) *visualization capability model* and the (dynamic) *user preferences model*. For each visualization the *visualization capability model* defines its ability to present certain data types (e.g. time-dependent data or concept hierarchy), activities and additional features like time or spatial dependent data. This model is static because the capabilities of a visualization are predefined and do not change.

The adaptation of the Output Component is performed in two steps. In a first step, based on the visualization capability model, the set of visualizations capable to present the user preferences with respect to visualizations, content, and activities are selected. In a second step, these visualizations are ordered according to the visualization preferences of the user. Herewith, the adaptation engine defines the overall layout of the User Interface as well as the parameters for each module in the Output Component. For instance a certain visualization type, the SeMap [22] is chosen in the Layout Module and the chosen visualization is parameterized (e.g. color, placement, size of the icons etc.) with the Presentation Module.

4.3 Output Component

Visualizations can be described according to the following characteristics: *what* is displayed, *where* is it displayed, and *how* is it displayed? Accordingly, each visualization in SemAdapt implements the modules semantics (data), layout and presentation.

The *semantics module* (*data module*) defines which data is visualized. It contains information about the data (what is the data about), its structure (e.g. hierarchy, incoming and outgoing relations) and amount. The main task of this module is to parameterize the data and provide the opportunity to adapt visualization in the lowest level. In semantically annotated data this module has further the separation functionality of different abstraction level (e.g. concept and instance level). With the information about data, data-related adaptation can be realized, e.g. content recommendation or adaptation of level of details.

The *layout module* defines where and how the data will be visualized. Dependent on the user preferences, different graphical metaphors like text, different graph-layout algorithms may be chosen. For each metaphor, different layout algorithms like cone tree or tree-map are provided. It should be noted that only the geometrical layout is defined on this layer, but not yet the visual appearance.

The *presentation module* defines more precisely how the data will be presented. It is the visual layer of the visualizations and parameterizes the visual look, by setting e.g. the texture, color, or size. The input of the presentation is the geometry as calculated by the layout module. The output component of SemAdapt contains a set of different visualizations. Each of these visualizations has its own pipeline of semantics (data), layout and presentation. The visualizations are grouped into three categories:

General visualized components (GVCs) are abstracted visualizations and form the UI. GVCs are primarily responsible for the selection, placement and initialization of visualization types in their layout layer. Their semantics layer separates the data from the visualization. It decides which part of the data should be visualized with which visualization. The presentation layer is responsible for the separation of the visual representations. It just decides which visualizations should have the same visual representation. It is necessary e.g. for comparative visualization. Further the GVCs decide the interactive connections between the visualization. If a user interacts with visualization v1, should the visualization v2 react to the user interaction?

Semantics visualization components (SVCs) visualize the structure of the semantic data and provide the possibility to interact. SVCs are the real visualization components and consist of one or more visualization algorithms, e.g. force directed or concentric radial. While SVCs get their data from the Semantics Layer of the GVCs, their own Semantics layer decides about the number, level-of-detail and ranking of the data to be visualized. The layout of a SVC has different parameterizations, e.g. the centralization of a topic (in concentric radial) or the order of concepts (in SeMap [22]). The presentation layer parameterizes the visual representation, e.g. by choosing colors or sizes for the instances, relations and concepts.

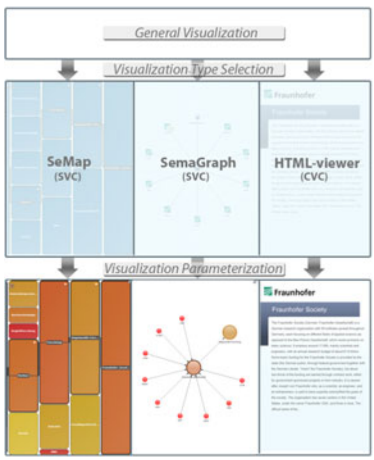

Fig. 1. Three-stepped adaptation of the output component

Content Visualization Components (CVCs) are responsible for the presentation of the content referenced by the semantics. Examples for CVCs are pictures or HTML-viewers. They are interactively coupled to one or more SVCs and inherit the semantics, presentation and layout form theses coupled SVCs. The output component provides a three-stepped adaptation of the visualization. The figure above shows these three steps. In the first step the GVC provides a number of visualization types based on the input and the adaptation component. In the figured case two SVCs (SeMap and SemaGraph) and one CVC (HTML-viewer) are chosen by the GVC. In the next step each of the chosen visualizations creates its own Layout and visualizes the same information in different abstraction levels. Therefore an adaptation of the Semantics (content) and the Layout is performed. The last step adapts the visual representation (Presentation layer) of the information. In the given case, size, order and color are adapted and parameterized to the computed requirements of the user and data respectively.

Several visualizations for SemAdapt were already developed and are used for adaptive visualization of semantically annotated data. These visualizations were introduced in some of our previous works [14,22,23,24].

5 A SemAdapt Application Example

In this chapter we describe the functionality of the SemAdapt framework based on a knowledge exploration example. Knowledge exploration is an important process for adopting knowledge with information systems, whereas graphical representation of the knowledge can help to optimize the learning process and reduce the cognitive overload [14]. For evaluating the system with ground-truth data, we are using the Freebase data base [8] extended with an own schema. Freebase contains open linked data, which can easily be mapped to a formal data specification, whereas *Domains* are the highest concepts (of an ontology), *Types* inherit from *Domains*, the individuals are called *Topics*. The relations are defined as *Properties*. The starting point of our example is the interaction of a user on a graphical representation of the concept *Fraunhofer*, which is analyzed in the Input Component as described above. This interaction produces a search query on the freebase data base. The results are analyzed based on the procedure described in [27].

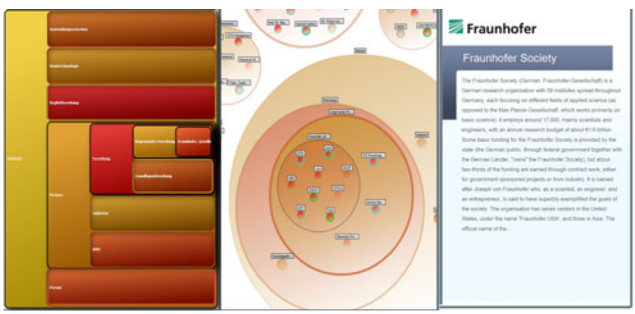

Fig. 4. Example for Visualization Adaptation

The results are information about the data, its structure and the amount of results in different semantics categories. Based on these results a set of visualizations is chosen, that meets the requirements of the analyzed data. Further the interaction is analyzed in a user-centered way based on the procedure described above. The results are a current set of information about the user and the data, which is further compared with the user model described in [28]. The result of the comparison is a table with scalar values for the fitting of different visualization, abstracted data layer (ontological concepts) and activities. Based on the computed values several visualization types and parameterizations are set, whereas a user interface may consist of more than one visualization [14].

6 Conclusion and Future Work

In this paper a reference model and framework for adaptive visualization has been presented with the following main innovative features. The overall conceptual model is based on the distinction of three main components and herewith, supports perspectives to adaptive visualization. In combination with the modular structure of the system's conceptual architecture, this allows the goal-oriented adaptation of specific parts of the system with appropriate level of detail. The user interface consists of a so-called visualization cockpit, which integrates different visualizations, chosen and adapted according to the user preferences. This concept supports both independent views as well as user preferred interactions in closely coupled visualizations. SemAdapt, so far, focuses on the adaptation of visualization type and its parameterization.

One main area of future research is the extension of the framework to handle raw data and metadata. The cockpit concept will be extended so that different data types can be visualized in parallel. Major aspects of future work are editing and annotation and their synchronization along different data types. From an adaptation point of view, this will lead to a substantial extension of both the user model as well as the capability model. Another principal extension of SemAdapt will be the conceptualization of adaptive visualization in Web 2.0, where user groups with different roles and preferences view and edit in shared environments.

References

1. Keim, D.A., Mansmann, F., Schneidewind, J., Thomas, J., Ziegler, H.: Visual Analytics: Scope and Challenges. In: Simoff, S.J., Böhlen, M.H., Mazeika, A. (eds.) Visual Data Mining. LNCS, vol. 4404, pp. 76–90. Springer, Heidelberg (2008)
2. Hui, B., Gustafson, S., Irani, P., Boutilier, C.: The need for an interaction cost model in adaptive interfaces. In: Proceedings of the Working Conference on Advanced Visual interfaces, AVI 2008, pp. 458–461. ACM, New York (2008)
3. Cao, Y., Theune, M., Nijholt, A.: Modality effects on cognitive load and performance in high-load information presentation. In: Proceedings of the 13th International Conference on Intelligent User Interfaces (IUI 2009), pp. 335–344. ACM, New York (2009)
4. Thomas, J.: Visual Analytics: a Grand Challenge in Science-Turning Information Overload into the Opportunity of the Decade. Keynote talk. In: IEEE InfoVis 2005, Minneapolis (2005)
5. Keim, D., Andrienko, G., Fekete, J.-D., Görg, C., Kohlhammer, J., Melançon, G.: Visual Analytics: Definition, Process, and Challenges. In: Kerren, A., Stasko, J.T., Fekete, J.-D., North, C. (eds.) Information Visualization. LNCS, vol. 4950, pp. 154–175. Springer, Heidelberg (2008)
6. Brusilovsky, P.: Adaptive navigation support. In: Brusilovsky, P., Kobsa, A., Nejdl, W. (eds.) Adaptive Web 2007. LNCS, vol. 4321, pp. 263–290. Springer, Heidelberg (2007)
7. Ahn, J.-w., Brusilovsky, P.: Adaptive visualization of search results: Bringing user models to visual analytics. Information Visualization 8/3, 167–179 (2009)
8. Danine, A., Lefebvre, B., Mayers, A.: TIDES - Using Bayesian Networks for Student Modeling. In: ICALT 2006, pp. 1002–1007. IEEE Computer Society, Los Alamitos (2006)
9. Noguez, J., Sucar, L.E.: A Probabilistic Relational Student Model for Virtual Laboratories. In: ENC 2005: Proceedings of the Sixth Mexican International Conference on Computer Science, pp. 2–9. IEEE Computer Society, Los Alamitos (2005)

10. Anderson, C.R., Domingos, P., Weld, D.S.: Relational Markov Models and their Application to Adaptive Web Navigation. In: KDD 2002: Proceedings of the Eighth ACM SIGKDD 2002, pp. 143–152 (2002)
11. Davison, B., Hirsh, H.: Predicting Sequences of User Actions. In: Predicting the Future: AI Approaches to Time Series Problems, pp. 5–12. AAAI Press, Menlo Park (1998)
12. Leuski, A., Allan, J.: Interactive information retrieval using clustering and spatial proximity. User Modeling and User Adapted Interaction 14(2-3), 259–288 (2004)
13. Gotz, D., Wen, Z.: Behavior-driven visualization recommendation. In: IUI 2009: Proceedings of the 13th International Conference on Intelligent User Interfaces, pp. 315–324. ACM, New York (2009)
14. Nazemi, K., Burkhardt, D., Breyer, M., Stab, S., Fellner, D.W.: Semantic Visualization Cockpit - Adaptable Composition of Semantics-Visualization Techniques for Knowledge-Exploration. In: Proc. 13th International Conference ICL on Interactive Computer Aided Learning, ICL 2010, Hasselt, Belgium (2010)
15. Card, S.K., Mackinlay, J.D., Shneiderman, B.: Readings in Information Visualization: Using Vision to Think, 1st edn. Morgan Kaufmann, San Francisco (1999)
16. May, T., Kohlhammer, J.: Towards closing the analysis gap: Visual generation of decision supporting schemes from raw data. Computer Graphics Forum, 911–918 (2008)
17. Aaltonen, A., Lehikoinen, J.: Refining visualization reference model for context information. Personal Ubiquitous Comput. 9, 381–394 (2005)
18. Heer, J., Card, S.K., Landay, J.A.: Prefuse: a toolkit for interactive information visualization. In: Proceedings of the SIGCHI. ACM, New York (2005)
19. Brodlie, K.W., Brooke, J., Chen, M., Chisnall, D., Hughes, C., John, N., Jones, M., Riding, M., Roard, N., Turner, M., Wood, J.: Adaptive Infrastructure for Visual Computing. In: Proc. of Theory and Practice of Computer Graphics (2007)
20. Arens, Y., Hovy, E.: The design of a model-based multimedia interaction manager. Artif. Intell. Rev. 9(2-3), 167–188 (1995)
21. Kao, T.-H., Yuan, S.-M.: Designing an XML-based context-aware transformation framework for mobile execution environments using CC/PP and XSLT. Computer Standards & Interfaces 26(5), 377–399 (2004)
22. Nazemi, K., Breyer, M., Hornung, C.: SeMap: A Concept for the Visualization of Semantics as Maps. In: Stephanidis, C. (ed.) UAHCI 2009. LNCS, vol. 5616, pp. 83–91. Springer, Heidelberg (2009)
23. Stab, C., Breyer, M., Nazemi, K., Burkhardt, D., Hofmann, C., Fellner, D.: SemaSun: Visualization of Semantic Knowledge based on an improved Sunburst Visualization Metaphor. In: Proceedings of World Conference on Educational Multimedia, Hypermedia and Telecommunications, pp. 911–919 (2010)
24. Stab, C., Nazemi, K., Fellner, D.W.: SemaTime - Timeline Visualization of Time-Dependent Relations and Semantics. In: Bebis, G., Boyle, R., Parvin, B., Koracin, D., Chung, R., Hammound, R., Hussain, M., Kar-Han, T., Crawfis, R., Thalmann, D., Kao, D., Avila, L. (eds.) ISVC 2010. LNCS, vol. 6455, pp. 514–523. Springer, Heidelberg (2010)
25. Nazemi, K., Stab, C., Fellner, D.W.: Interaction Analysis for Adaptive User Interfaces. In: Huang, D.-S., Zhao, Z., Bevilacqua, V., Figueroa, J.C. (eds.) ICIC 2010. LNCS, vol. 6215, pp. 362–371. Springer, Heidelberg (2010)
26. Nazemi, K., Stab, C., Fellner, D.W.: Interaction Analysis: An Algorithm for Activity and Prediction Recognition. In: Proc. 2nd IEEE International Conference on Intelligent Computing and Intelligent Systems, ICIS 2010. IEEE, Los Alamitos (2010)
27. Nazemi, K., et al.: Interacting with Semantics: A User-Centered Visualization Adaptation based on Semantics Data. In: Jacko, J.A. (ed.) HCII 2011, Part I. LNCS, vol. 6761, pp. 491–500. Springer, Heidelberg (2011)
28. Nazemi, K., et al.: Modeling Users for Adaptive Semantics Visualizations. In: HCII 2011, Part I. LNCS, vol. 6761, pp. 491–500. Springer, Heidelberg (2011)