User Tasks for Evaluation

Untangling the Terminology Throughout Visualization Design and Development

Alexander Rind, Wolfgang Aigner, Markus Wagner
St. Pölten Univ. Applied Sciences, Austria
{ alexander.rind, wolfgang.aigner, markus.wagner } @fhstp.ac.at

Silvia Miksch, Tim Lammarsch
Vienna University of Technology, Austria
{ miksch, lammarsch } @ifs.tuwien.ac.at

ABSTRACT
User tasks play a pivotal role in evaluation throughout visualization design and development. However, the term 'task' is used ambiguously within the visualization community. In this position paper, we critically analyze the relevant literature and systematically compare definitions for 'task' and the usage of related terminology. In doing so, we identify a three-dimensional conceptual space of user tasks in visualization. Using these dimensions, visualization researchers can better formulate their contributions which helps advance visualization as a whole.

Categories and Subject Descriptors
H.5.2 [**Information Interfaces and Presentation**]: User Interfaces—*evaluation/methodology, theory and methods*

General Terms
Human Factors, Theory

Keywords
Reflections, taxonomy of tasks, interaction, terminology

1. INTRODUCTION
Tasks are an issue discussed frequently in the visualization community as they are central to how we can show the usefulness of our work. Yet there is continuing confusion over what the term 'task' means in a visualization context.

Tasks are pivotal for evaluation throughout visualization design and development [14; 20]: the understanding of environments and work practices is an essential step for problem-driven visualization projects, which encompasses the characterization of domain problems and their abstraction using generic tasks [20; 32]. For evaluating user experiences or performance, test users need to be presented with tasks as stimuli that are carefully chosen to explain the research hypothesis [7]. When transcribing case studies or field observations for understanding visual data analysis and reasoning processes the tasks performed need to be coded using a scheme [27]. But task are also needed for purely technique-driven visualization work with evaluation by algorithmic performance measures or qualitative result inspection [11] to set results in context. Finally, tasks can play a role at visualization runtime, for example for analytical provenance [9; 24] or recommending task-specific views [31; 36].

To support visualization researchers, task frameworks such as categorizations or taxonomies are useful to describe, prescribe, and compare visualization artifacts and data analysis processes [3; 4]. This is illustrated by the central role of tasks in widely used theoretical frameworks such as the Task by Data Type Taxonomy [33], the Data-Users-Tasks Design Triangle [19], and the Nested Model [20]. Once tasks are structured using a common categorization framework, they make it possible to extract general guidelines from individual research and development efforts [18] and to identify gaps currently not supported by techniques or not examined by evaluation studies. Thus tasks are a critical building block to advance the field of visualization.

However, there are many nuances of what a task can be. For example, a visualization tasks may be as open-ended as 'detect anomalies in recent public health data' or 'identify the main drivers of climate change' but also as crisp as 'find yesterday's most profitable product' or "buy a train ticket" [32, p. 2433]. Already in 1994 it was widely acknowledged that "the notion of 'task' is increasingly difficult to pin down" [28, p. 410] and nowadays the word 'task' is still regarded as "deeply overloaded in the visualization literature" [20, p. 921]. Based on our own experience throughout multiple visualization design and evaluation projects and inputs from fellow researchers we regard this confusion as unsatisfactory. A commonly agreed understanding and terminology of tasks are needed. Therefore, this position paper studies tasks along three conceptual dimensions (Figure 1):

Abstraction (concrete/abstract): Concrete tasks can be described using generic categories from task frameworks. For example, Andrienko and Andrienko [2] would classify the concrete task 'find the day of largest revenue' as the abstract task 'indirect lookup'. This allows systematic study on multiple levels of abstraction and the reuse of visualization methods for tasks of the same generic category.

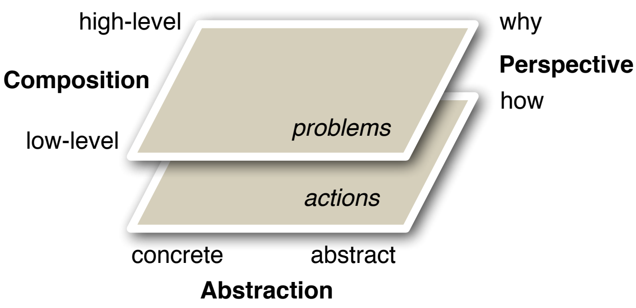

Figure 1: Overview of the conceptual space of user tasks in visualization with composition, perspective, and abstraction as orthogonal dimensions.

Composition (high-level/low-level): Tackling a task, it is common practice to break it down into smaller subtasks. Thus the level of composition can range from long-term challenges like 'end poverty' to mini steps like 'find outliers in economic data'. By the same token, task frameworks often distinguish high-level, low-level, and sometimes levels in-between [5; 30].

Perspective (why/how): On the one hand high-level visualization tasks are typically non-routine problems (e.g., 'find suspicious patterns'). On the other hand the interactive features of visualization artifacts and categories of some low-level task frameworks are formulated as executable actions (e.g., 'zoom to the orange cluster'). Since there is usually no direct relation – no decomposition – between problems and actions, we distinguish between a *Why?* perspective and a *How?* perspective.

The paper is organized as follows: Section 2 critically analyzes the relevant literature in visualization and human-computer interaction (HCI) and systematically compares task definitions and related work. Emerging from that a three-dimensional conceptual space of user tasks in visualization is introduced in Section 3. In Section 4 we examine implications and provide examples of how these dimensions allow visualization researchers to describe their contributions more precisely.

2. LITERATURE SURVEY

In an effort to distill a consistent terminology for research and development in visualization, we compare different definitions of what tasks are and what role they should play in visualization.

In colloquial English a 'task' is understood as "a piece of work that has been given to someone : a job for someone to do".[1] The Merriam-Webster Dictionary emphasizes that tasks are characterized as being externally assigned, having a deadline, and being hard or unpleasant. Surveying the HCI and visualization literature we can however discover a shifted focus and a multitude of nuances of this general definition. Next, we analyze these nuanced definitions across the three dimensions presented above. These dimensions are deeply interconnected as we explain in Section 2.4 but we will show in Section 3 that looking at them separately helps to untangle the terminological confusion.

2.1 Abstraction

For a systematic investigation of tasks in visualization as in the example scenarios of the introduction, an abstraction of concrete tasks is needed. First, we can distinguish between a single performance of a task by a single person at a single point in time and abstract description of a task. In controlled experiments, the term 'trial' is used for "each occasion a participant performs the task" [7, p. 298]. Second, there are various possibilities to generalize tasks or group them to categories. A task can be expressed in terms of the users' domain or using abstract terms [20, p. 921]. For example, Munzner [20, p. 921] distinguishes between a task formulated in terms of the domain, which she calls 'problem', and an abstract task, which she denotes as 'operation'. To formulate a task on a middle abstraction level, the characteristics of the analyzed data type can be used (e.g., behaviors of the data [2], time-oriented data [15], multivariate networks [29], and temporal graphs [13]). Alternatively, tasks can be formulated in terms closer to specific visualization techniques (e.g., interactive dynamics [10]). The terms used in the literature for abstract tasks include 'generic', 'general', but also 'high-level'.

2.2 Composition

It appears as a general notion that users break down tasks into smaller, better manageable *subtasks* [8]. Preece et al. [28, p. 412] speak of a "decomposition of goals into subgoals and tasks into subtasks as the user moves downwards through a hierarchy of systems". One possible notation to specify such hierarchical tasks models are ConcurTaskTrees [25]. Norman [23, p. 15] introduces a composition level above tasks: "an activity is a coordinated, integrated set of tasks" and emphasizes that design should be centered on these activities. Gotz and Zhou [9, p. 46] describes how "analysts typically follow a divide-and-conquer approach, performing several sub-tasks to achieve the requirements of a single top-level task". In the visualization literature, the terms 'high-level' and 'low-level' are often used to distinguish such tasks [5] but there is no consistent understanding.

Tasks at the leaf-level of the hierarchy are often treated distinctively. Preece et al. [28, p. 411] define an *action* "as a task that involves no problem solving or control structure component" and is performed by "the human physically interact[ing] with a device". Tasks at a similar level of composition are also referred to as 'simple tasks' or 'unit tasks' [28]. Fuchs [8, p. 10] uses 'basic tasks' for the leaf-level and differentiates them from 'actions', which "describ[e] the functional properties beyond the conceptual task decomposition". He defines an 'action' as "an atomic operation that is executed upon an artifact, by an entity that is involved in the completion of the task (user, computer, . . .)". Similarly, the action tier of Gotz and Zhou [9, p. 46] represents "an atomic analytic step performed by a user with a visual analytic system". However, they consider yet another level below actions: "events correspond to the lowest-level of user interaction events (for example, a mouse click or a menu item selection) which carry very little semantic meaning" [p. 43].

[1] http://www.merriam-webster.com/dictionary/task, accessed Aug 27, 2014.

Norman [23, p. 15] describes tasks as "composed of actions, and actions [as] made up of operations". ConcurTaskTrees [25] distinguish at the leaf-level between 'user tasks', 'application tasks', and 'interaction tasks'. They refer to tasks composed of such subtasks as 'abstract tasks'. The selection of tools also plays an important role in task decomposition. Both Preece et al. [28] and Cooper et al. [6] use it to distinguish between 'goals' and 'tasks'. For Fuchs [8] it is the 'action' that involves an artifact such as a task-specific visual representation.

2.3 Perspective

If we understand the task, like in colloquial English, as some work to be done, we can characterize it from two perspectives:

On the one hand, the *Why?* perspective describes a task by the problem addressed or the intended outcomes. Preece et al. [28, p. 411] define an external task or goal "as a state of a system that the human wishes to achieve". Roth [30, p. 2357] distinguishes further between "an ill-defined task, or goal, motivating use of the visualization" and "a well-defined task, or objective, that can support the goal". In visualization, the *why* perspective is often formulated as a question or a query to be answered based on the data at hand. Andrienko and Andrienko [2, p. 49] "use the word 'tasks' to denote typical questions that need to be answered by means of data analysis", Amar et al. [1] categorize tasks by questions or queries asked, and Munzner [20, p. 921] denotes a task described in domain terms as 'problem'. For Gotz and Zhou [9, p. 46] the "task tier captures a user's high-level analytic goals".

On the other hand, the *How?* perspective addresses the means or activities by which users perform the task. Preece et al. [28, p. 411] define a task, in particular an internal task, "as the activities required, used or believed to be necessary to achieve a goal using a particular device". Fuchs [8, p. 10] understands a task "as a single, conceptually distinguishable but not necessarily atomic step within a composite activity or work flow". Cooper et al. [6, p. 15] write "both activities and tasks are intermediate steps (at different levels of organization) that help someone to reach a goal or set of goals". For Schulz et al. [31, p. 2366] visualization tasks are "activities to be carried out interactively on a visual data representation for a particular reason", for Brehmer and Munzner [5, p. 2376] "abstract tasks are domain- and interface-agnostic operations performed by users", and Roth [30, p. 2357] distinguishes a third category: "a system function, or operator, that may support the objective".

2.4 Summary

What makes the notion of 'task' so hard to understand are not only the differences in these three dimensions but also their interconnectedness. For example, when we follow a concrete user session, we could observe how the user transforms a high-level *why* task formulated in terms of the application domain into low-level *how* tasks matching operations of the visualization system [9]. However, it is difficult to tackle tasks in a more abstract way needed to draw generalizable guidelines and to characterize tasks on the levels in-between. Bridging between the extremes and taking an abstract view on a middle level appears as a promising direction [37; 5; 31].

Previous work has started to elaborate these dimensions: On the one hand, Munzner [20] proposes a 2x2 matrix of task concepts by abstractions and composition. On the other hand, Roth [30] distinguishes between different concepts for *why* (goal, objective) and *how* (operator). Pike et al. [26] combines the *why/how* dichotomy and the level of composition. Brehmer and Munzner [5] and Schulz et al. [31] represent tasks in a three-dimensional respectively a five-dimensional design space that encompasses both *why* and *how*.

What is missing, is a more fine-grained commonly agreed understanding and a terminology that can distinguish between *why* and *how* tasks. In particular, we need to emphasize that visualization often addresses open-ended problems. When solving such problems, the users can follow different strategies and there are often no definite mappings between the *why* and the *how*. Furthermore, HCI scholars have long warned against a design approach that focuses on hierarchical analysis of operational tasks and not on the goals and characteristics of users [3; 6; 23; 28].

3. PROPOSED CONCEPTUAL SPACE

As a result of this survey of the literature, we conclude that the term 'task' is not consistently defined. To avoid the ensuing ambiguity we first propose two distinct terms to explicitly address the *why* and *how* perspective. For the *why* task, we borrow the term 'problem' from Munzner [20] and derive its definition from the Merriam-Webster Dictionary[2] and from Preece et al. [28]:

> A problem is a question the user raises for inquiry, consideration, or solution while aiming to achieve his or her goals.

For the *how* task, we adopt the widely used term 'action' and we derive its definition from Gotz and Zhou [9]:

> An action is a discrete step towards solving a problem.

Second, we propose to consider two continuous dimensions for abstraction and composition in addition to the *why/how* dichotomy. The *compositions* of problems and actions form hierarchies for which we propose a continuous scale from high-level to low-level. *Abstraction* captures that different problems or different actions can be grouped to a common category. Here we distinguish on a continuous scale between *concrete* and *abstract*.

Abstraction versus Composition. Terms like 'high-level' and 'abstract' are not used consistently in the literature surveyed in Section 2. These dimensions might also be confused as high-level tasks are often extremely vague

[2] http://www.merriam-webster.com/dictionary/problem, accessed Jun 30, 2014.

such as 'improve business'. Nevertheless abstraction can be clearly distinguished from composition: a low-level task is a part of a high-level task, whereas an abstract task is a more generic category of a concrete task. Thus, a low-level task demands less work than its high-level task, while the concrete and the abstract task demand the same work.

Perspective versus Abstraction. Both problems and actions can be described at different levels of abstraction. Interestingly, abstraction along users' domains lends itself rather to problems and abstraction along the features of visualization techniques fits more to actions, whereas abstraction along data type applies to both perspectives.

Perspective versus Composition. Likewise, both problems and actions can be composed or decomposed: On the *why* perspective, users break down large problems to increasingly smaller *subproblems* intentionally or unintentionally in order to make ill-defined problems manageable using visualization tools [8; 22]. On the *how* perspective, blocks of consecutively observed actions can be combined to *action sequences*, at multiple levels of composition. With experience, users will develop *action strategies* to solve common subproblems with a tool and it is possible to analyze such strategies by observing users [27]. In addition, actions can be decomposed further to a level below intentional problem solving such as individual user interface events [9].

Even though the composition hierarchies of problems and actions are connected, it makes sense to draw a clear line between them along the *why/how* dichotomy because problems and actions are different concepts with different properties. Furthermore actions can be found at the same composition level as middle- to low-level problems with the mapping varying largely depending on visualization tools and user experience.

Examples. 'Understand election results' would be an example of a high-level problem, which could be broken down to include a low-level problem like 'what is the geographic distribution of the districts with largest voter participation'. This problem could be addressed by an action to encode this variable to color on a choropleth map. Multiple mouse clicks might be needed to change the encoding. A higher level action strategy involve both grouping small multiples of the map and changing the color-encoding.

The high-level problem of 'understand election results' could be formulated more concretely as 'understand the 2014 elections to the European parliament' or more generically as 'explore multivariate geographic data'. The low-level problem could be abstracted as synoptic pattern search according to Andrienko and Andrienko's task typology [2]. The action could be categorized as 'encode' according to Yi et al.'s user intent categorization [37]. A less generic action categorization could group all actions that 'change color coding' on choropleth maps across visualization tools. Finally, the provenance of a user session would be concrete high-level action sequence.

The conceptual space spanned by these three orthogonal dimensions can be used to compare individual tasks as well as task categorizations. For example, the framework by Andrienko and Andrienko [2], the framework MacEachren [15], and the analysis questions for comparative genomic by Meyer et al. [17] would all capture problems at a similar low composition level but at largely different levels of abstraction ranging from generic, to time-oriented data, to a concrete application domain.

4. DISCUSSION

In this section, we interpret the dimensions of the conceptual space proposed above and observe their implications.

The visualization process – as we conceptualize it – starts with a user, data, and a problem. The problem can originate from the user's personal goals or from the goals of the user's organization. Under consideration of the *tools* available to the user and his or her *background knowledge* about the problem, the data, and the tools he or she breaks the problem down to manageable *subproblems* and develops a *plan*, which is a sequence of actions that he or she believes will solve the problem. While performing these actions, the user updates the plan iteratively as the background knowledge, the availability of tools, or even their goals can change [28; 21; 22; 6; 26]. With experience, users will develop *action strategies*, generic and low-level plans, to solve common subproblems with a tool. It is possible to analyze such strategies by observing users [27] and to improve user performance and experience based on use insights from such user studies.

Suitable Levels of Composition. Actions take a pivotal level in this concept. As discrete steps they should represent the lowest level of activity of which the user is consciously aware. Below this level, the user activates a number of *events* in the visualization user interface that are interpreted by interaction techniques. For example, panning using a scrollbar involves a mouse-down, several mouse-drag, and a mouse-up event. However, it is a matter of context and in particular the user's experience, which granularity the user regards as discrete steps [28; 21]. One user's strategy might be another user's action and this might also change within a single user over time as he or she progresses from novice to expert. For example, a user might learn to click through menus and dialog boxes unconsciously because he or she frequently needs to change a setting that is hidden there.

Likewise, problems can be broken down to increasingly lower levels. However, decomposition into trivially small subproblems such as 'read value 1, read value 2 ...' is often not practical and differs from how users realistically solve problems. Combining human perception and visualization techniques, they can detect patterns at a larger scope. For example, they can spot clusters in a scatter plot or judge the trend in a line plot without consecutively reading the values encoded for individual data records. Therefore, it is important to consider problems at an adequate level of composition, in particular when evaluating visualization techniques.

Open-Endedness of Problems. The term 'task' often has connotations as being externally assigned and scheduled within a workflow. However, this work focuses on visualization we do not look at tasks from a perspective of project management or workflow management systems but concentrate on the scope steered by the user to solve a reasonably open-ended problem. We assume that the user engaged with a problem has some intellectual interest in the outcomes. We would not accept an assignment like 'pan to the year 1983' or 'click on the blue rectangle' as a problem. Such assignments are purely perceptual and mechanical and do not carry the notion of the user raising an inquiry. Furthermore, problems in the visualization field involve questions based on *data*.

This intellectual interest in problems and their many-to-many mapping to action sequences are our motivations to distinguish between the perspectives *why* and *how*. They are also a major difference to the traditional view of tasks in HCI that follows the assumption of a one-to-one mapping between these perspectives [3]. Yet, such an assumption holds only for operational tasks and not for such creative tasks as they are needed for the open-ended problems addressed by visualization [16; 34].

4.1 Tasks—use with care

We assume that much of the confusion and criticism on tasks stems from this traditional view of operational tasks. For example, some visualization experts reportedly [2, p. 148] even "believe that having a defined task is not (or not always) necessary". Other HCI experts like Preece et al. [28, p. 410] warn that "the idea of a task is useful in system development – as long as it is used with care" because otherwise current action sequences would be reinstantiated in future systems, which are too rigid to serve the real problems of users.

However, we agree with Miksch and Aigner [19], Munzner [20], Andrienko and Andrienko [2], and many other visualization experts that tasks are a useful concept for evaluation throughout visualization design and development. Yet, we propose that the terms 'problem' and 'action' should be used more often instead of the ambiguous term 'task'. Furthermore, we recommend to understand perspective, level of composition, and level of abstraction as distinct dimensions.

To underline the role of problems and actions and to further illustrate their proposed usage, we discuss them in some evaluation settings already listed in Section 1:

Domain Characterization and Abstraction. An essential aspect for understanding of environments and work practices [14] is to figure out the *problems* of domain experts. As *high-level problems* might be too vague, it is often necessary to gather problems at multiple levels of *composition*. Subsequently it is necessary to transform *concrete problems* formulated in domain language to *abstract problems* that can be matched to categories of task/problem frameworks. The two outer layers of Munzner's Nested Model [20] describe these steps.

Stimuli for Experiments. User experiences or performance can be evaluated by letting test users interact with visualization artifacts in a controlled environment. The stimuli presented to the test users are typically *problems* and can be formulated as question like 'which <items> fulfill <. . . >', as imperative like 'identify <items> that fulfill <. . . >', or even prescribe the answering method like 'click on <items> that fullfill <. . . >'. As argued above the evaluation of predefined action sequences like 'filter by <. . . >, then zoom to <. . . >' is less relevant for user studies in visualization.

Depending on the study's hypotheses a suitable composition level must be found because *low-level problems* can be too trivial and *high-level problems* too open-ended for quantitative analysis of time and errors. For such experiments other evaluation methods, e.g., qualitative analysis of insights [35], are more suitable.

Research hypotheses are often specific to an *abstract problem* such as 'less errors for comparison' whereas stimuli need to present a *concrete problems* that act as representative examples for the abstract problem. In some cases a concrete problem is also translated to a concrete problem in a different domain so that a sufficiently large population of test persons can be recruited.

Setting Context. When case studies are performed to evaluate visualization artifacts under more realistic conditions it is necessary to describe the *problem* tackled and it is useful to set it in context using *abstract problems*. This also applies to evaluation by algorithmic performance measures or qualitative result inspection [11].

Analyze Interaction. Observation of user interaction in case studies with domain experts allows to investigate visual data analysis and reasoning processes. Here, researchers can use *actions* to transcribe interaction logs and identify generalized *strategies* based on *abstract actions* [27]. Such action logs make also sense as history mechanism to provide analytical provenance at visualization runtime [9; 24].

Recommendation. Given the user's *problems* the visualization artifact can also recommend suitable settings at runtime. For example Tominski et al. [36] propose to choose one of eight color schemes based on a three-dimensional problem space and Schulz et al. [31] use their five-dimensional task framework to recommend visual representation techniques for climate impact data.

5. CONCLUSIONS AND FUTURE WORK

In this paper we critically analyze the usage of the term 'task' in visualization and human-computer interaction literature. We propose to use 'problem' and 'action' as a more suitable terminology that reduce ambiguity and allow visualization researchers to better formulate their contributions.

However, this position paper can only be a proposal to members the visualization community. Before it is adopted and can have an impact, the terminology needs to be agreed upon. We hope to spark discussion at BELIV 2014 and VIS 2014 and that these lead towards a community consensus glossary defining terms such as 'problem', 'action', 'visualization artifact', or 'interaction technique' which would be comparable to a similar effort of the temporal database community in the 1990s [12].

In addition, we identify a three-dimensional conceptual space of user tasks in visualization with abstraction, composition, and perspective as orthogonal dimensions. While tasks can be formulated across this concept space we observe that some subspace is more suitable for visualization design and evaluation than the rest. Identifying these levels of composition and abstraction for problems and actions is a challenge for future research. While the paper at hand is primarily based on definitions and theoretical frameworks a thorough survey of evaluation tasks used in actual user studies can be conducted and coded using our three-dimensional concept space. Further work can be directed at filling the gaps of the potentially suitable subspace.

6. ACKNOWLEDGMENTS

This work was supported by the Austrian Science Fund FWF via KAVA-Time (#P25489) and HypoVis (#P22883) and by the Austrian Federal Ministry of Science, Research, and Economy via CVAST, a Laura Bassi Centre of Excellence (#822746).

References

[1] R. Amar, J. Eagan, and J. T. Stasko. Low-level components of analytic activity in information visualization. In *Proc. IEEE Symp. Information Visualization, INFOVIS*, pages 111–117, 2005. doi:10.1109/INFVIS.2005.1532136.

[2] N. Andrienko and G. Andrienko. *Exploratory Analysis of Spatial and Temporal Data: A Systematic Approach*. Springer, Berlin, 2006. doi:10.1007/3-540-31190-4.

[3] M. Beaudouin-Lafon. Designing interaction, not interfaces. In *Proc. Working Conf. Advanced Visual Interfaces, AVI*, pages 15–22. ACM, 2004. doi:10.1145/989863.989865.

[4] B. B. Bederson and B. Shneiderman. Theories for understanding information visualization. In *The Craft of Information Visualization: Readings and Reflections*, pages 349–351. Morgan Kaufmann, San Francisco, CA, 2003.

[5] M. Brehmer and T. Munzner. A multi-level typology of abstract visualization tasks. *IEEE Trans. Visualization and Computer Graphics*, 19(12):2376–2385, 2013. doi:10.1109/TVCG.2013.124.

[6] A. Cooper, R. Reimann, and D. Cronin. *About Face 3: The Essentials of Interaction Design*. John Wiley & Sons, 2012.

[7] C. Forsell and M. Cooper. An introduction and guide to evaluation of visualization techniques through user studies. In W. Huang, editor, *Handbook of Human Centric Visualization*, pages 285–313. Springer, New York, 2014. doi:10.1007/978-1-4614-7485-2_11.

[8] G. Fuchs. *Task-based Adaptation of Graphical Content in Smart Visual Interfaces*. Ph.D. thesis, Universität Rostock, Apr. 2011.

[9] D. Gotz and M. X. Zhou. Characterizing users' visual analytic activity for insight provenance. *Information Visualization*, 8(1):42–55, Mar. 2009. doi:10.1057/ivs.2008.31.

[10] J. Heer and B. Shneiderman. Interactive dynamics for visual analysis. *Queue*, 10(2):30:30–30:55, Feb. 2012. doi:10.1145/2133416.2146416.

[11] T. Isenberg, P. Isenberg, J. Chen, M. Sedlmair, and T. Möller. A systematic review on the practice of evaluating visualization. *IEEE Trans. Visualization and Computer Graphics*, 19(12):2818–2827, 2013. doi:10.1109/TVCG.2013.126.

[12] C. Jensen, C. Dyreson, M. Böhlen, J. Clifford, R. Elmasri, S. Gadia, F. Grandi, P. Hayes, S. Jajodia, W. Käfer, N. Kline, N. Lorentzos, Y. Mitsopoulos, A. Montanari, D. Nonen, E. Peressi, B. Pernici, J. Roddick, N. Sarda, M. Scalas, A. Segev, R. Snodgrass, M. Soo, A. Tansel, P. Tiberio, and G. Wiederhold. The consensus glossary of temporal database concepts – February 1998 version. In O. Etzion, S. Jajodia, and S. Sripada, editors, *Temporal Databases: Research and Practice*, pages 367–405. Springer, Berlin, 1998. doi:10.1007/BFb0053710.

[13] N. Kerracher, J. Kennedy, and K. Chalmers. Tasks for temporal graph visualisation. *arXiv:1402.2867 [cs]*, Feb. 2014. URL `http://arxiv.org/abs/1402.2867`.

[14] H. Lam, E. Bertini, P. Isenberg, C. Plaisant, and S. Carpendale. Empirical studies in information visualization: Seven scenarios. *IEEE Trans. Visualization and Computer Graphics*, 18(9):1520–1536, Sept. 2012. doi:10.1109/TVCG.2011.279.

[15] A. M. MacEachren. *How Maps Work*. Guilford Press, New York, 1995.

[16] E. Mayr, M. Smuc, and H. Risku. Many roads lead to Rome: Mapping users' problem-solving strategies. *Information Visualization*, 10(3):232–247, July 2011. doi:10.1177/1473871611415987.

[17] M. Meyer, T. Munzner, and H. Pfister. MizBee: A multiscale synteny browser. *IEEE Trans. Visualization and Computer Graphics*, 15(6):897–904, Nov. 2009. doi:10.1109/TVCG.2009.167.

[18] M. Meyer, M. Sedlmair, P. S. Quinan, and T. Munzner. The nested blocks and guidelines model. *Information Visualization*, 2014. Published online before print, doi:10.1177/1473871613510429.

[19] S. Miksch and W. Aigner. A matter of time: Applying a data-users-tasks design triangle to visual analytics of time-oriented data. *Computers & Graphics*, 38:286–290, 2014. doi:10.1016/j.cag.2013.11.002.

[20] T. Munzner. A nested model for visualization design and validation. *IEEE Trans. Visualization and Computer Graphics*, 15(6):921–928, 2009. doi:10.1109/TVCG.2009.111.

[21] B. A. Nardi. Studying context: A comparison of activity theory, situated action models, and distributed cognition. In B. A. Nardi, editor, *Context and Consciousness: Activity Theory and Human-Computer Interaction*, pages 69–102. MIT Press, 1996. Preprint retrieved from CiteSeer.

[22] D. A. Norman. Cognitive engineering. In S. W. Draper and D. A. Norman, editors, *User Centered System Design*, pages 31–61. Lawrence Erlbaum, 1986.

[23] D. A. Norman. Human-centered design considered harmful. *interactions*, 12(4):14–19, 2005. doi:10.1145/1070960.1070976.

[24] C. North, R. Chang, A. Endert, W. Dou, R. May, B. Pike, and G. Fink. Analytic provenance: process+interaction+insight. In *Proc. 2011 Ann. Conf. Ext. Abstracts Human Factors in Computing Systems, CHI EA*, pages 33–36. ACM, 2011. doi:10.1145/1979742.1979570.

[25] F. Paterno, C. Mancini, and S. Meniconi. ConcurTaskTrees: A diagrammatic notation for specifying task

models. In S. Howard, J. Hammond, and G. Lindgaard, editors, *Proc. IFIP TC13 Int. Conf. Human-Computer Interaction, INTERACT*, pages 362–369. Springer, 1997. doi:10.1007/978-0-387-35175-9_58.

[26] W. A. Pike, J. Stasko, R. Chang, and T. A. O'Connell. The science of interaction. *Information Visualization*, 8(4):263–274, 2009. doi:10.1057/ivs.2009.22.

[27] M. Pohl, S. Wiltner, S. Miksch, W. Aigner, and A. Rind. Analysing interactivity in information visualisation. *KI – Künstliche Intelligenz*, 26:151–159, 2012. doi:10.1007/s13218-012-0167-6.

[28] J. Preece, Y. Rogers, H. Sharp, D. Benyon, S. Holland, and T. Carey. *Human-Computer Interaction*. Addison-Wesley, Harlow, UK, 1994.

[29] J. Pretorius, H. C. Purchase, and J. T. Stasko. Tasks for multivariate network analysis. In A. Kerren, H. C. Purchase, and M. O. Ward, editors, *Multivariate Network Visualization*, LNCS 8380, pages 77–95. Springer, Jan. 2014. doi:10.1007/978-3-319-06793-3_5.

[30] R. E. Roth. An empirically-derived taxonomy of interaction primitives for interactive cartography and geovisualization. *IEEE Trans. Visualization and Computer Graphics*, 19(12):2356–2365, 2013. doi:10.1109/TVCG.2013.130.

[31] H.-J. Schulz, T. Nocke, M. Heitzler, and H. Schumann. A design space of visualization tasks. *IEEE Trans. Visualization and Computer Graphics*, 19(12):2366–2375, Dec. 2013. doi:10.1109/TVCG.2013.120.

[32] M. Sedlmair, M. Meyer, and T. Munzner. Design study methodology: Reflections from the trenches and the stacks. *IEEE Trans. Visualization and Computer Graphics*, 18(12):2431–2440, Dec. 2012. doi:10.1109/TVCG.2012.213.

[33] B. Shneiderman. The eyes have it: A task by data type taxonomy for information visualizations. In *Proc. IEEE Symp. Visual Languages, VL*, pages 336–343, 1996. doi:10.1109/VL.1996.545307.

[34] B. Shneiderman. Creativity support tools: Accelerating discovery and innovation. *Commun. ACM*, 50(12):20–32, Dec. 2007. doi:10.1145/1323688.1323689.

[35] M. Smuc, E. Mayr, T. Lammarsch, W. Aigner, S. Miksch, and J. Gärtner. To score or not to score? tripling insights for participatory design. *IEEE Computer Graphics and Applications*, 29(3):29–38, May 2009. doi:10.1109/MCG.2009.53.

[36] C. Tominski, G. Fuchs, and H. Schumann. Task-driven color coding. In *Proc. 12th Int. Conf. Information Visualisation, IV*, pages 373–380. IEEE, July 2008. doi:10.1109/IV.2008.24.

[37] J. S. Yi, Y. A. Kang, J. T. Stasko, and J. A. Jacko. Toward a deeper understanding of the role of interaction in information visualization. *IEEE Trans. Visualization and Computer Graphics*, 13(6):1224–1231, 2007. doi:10.1109/TVCG.2007.70515.