Electronic Communications of the EASST
Volume 13 (2008)

Proceedings of the
Second International Workshop on
Layout of (Software) Engineering Diagrams
(LED 2008)

Positioning Map: a Visual Technique to Improve the Layout of Diagram Contextual Information

Nicolas Genon, Raimundas Matulevičius, Vincent Englebert and Patrick Heymans

8 pages

Guest Editors: Andrew Fish, Harald Störrle
Managing Editors: Tiziana Margaria, Julia Padberg, Gabriele Taentzer
ECEASST Home Page: http://www.easst.org/eceasst/ ISSN 1863-2122

Positioning Map: a Visual Technique to Improve the Layout of Diagram Contextual Information

Nicolas Genon, Raimundas Matulevičius, Vincent Englebert and Patrick Heymans

PReCISE Research Centre, Faculty of Computer Science, University of Namur,
rue Grandgagnage 21, 5000 Namur, Belgium
{nge, rma, ven, phe}@info.fundp.ac.be

Abstract: The presentation of information is a difficult activity. It requires to deal with model complexity, characterised among other things by the number of visual elements per diagram. In this position paper, we propose a *positioning map* constructed by combining the navigation and locator maps – two techniques resulting from the theoretical principles of effective communication. We believe that the positioning map is a better means than navigation and locator maps together to decompose models and to integrate their information cognitively. However, we still need to validate our proposal through empirical studies.

Keywords: Modelling language, model visualisation, cognitive psychology, effective communication.

1 Introduction

Visual modelling languages are common means to share information among different Information System (IS) engineering project stakeholders. To be efficient, a language needs to be properly used. Project participants need to be aware of the language syntax and semantics; they also need to respect conceptual modelling conventions [Ros78] ensuring that the language is applied to the right level of abstraction. Moreover, the language should be appropriate for the problem domain, should be able to convey the relevant information, and meet other organisational goals [Kro01]. Another aspect of the language application problem is representing information in a graphical form that facilitates unambiguous, precise and complete understanding. Based on computer-human interaction [NH98] and cognitive psychology [BG03], Moody has presented a set of principles of effective communication (PoEC) [Moo06a] [Moo06b]. For example, some of these principles describe how diagram elements can be visually structured in order to group related information; others describe how to highlight the most important model elements.

On average, the human mind is capable of remembering "seven, plus or minus two" [Mil56] pieces of information at the same time. However, as observed in [Bro87] [Moo03], visual diagrams are usually too *complex*, where complexity is understood as easy, quick and accurate reading and understanding of diagrams. This results in difficulties for the reader to correctly comprehend information provided in the diagram. In order to be understood more easily, the model needs to be decomposed into different parts (diagrams) presenting different concerns [KLM+97]. The resulting diagrams, then, need to be combined together mentally by the reader in order to recollect the overall information. This can also be a daunting task. To facilitate it, we need adequate conceptual and technical support.

In this paper, we are analysing Moody's principles [Moo06b] of (*i*) *modularity* – which explains how an information model needs to be divided into cognitively and perceptually manageable diagrams – and (*ii*) *cognitive integration* – which explains how to design navigation aids between decomposed diagrams and how to integrate pieces of information of diagrams into a coherent mental representation of the overall model. In particular, we investigate two specific techniques: the *navigation map* – which helps to decompose the overall model into separate diagrams and to visualise how these diagrams are linked together – and the *locator map* – which helps to see the active content of the diagram. These two techniques were elaborated to reduce the cognitive effort resulting from the application of the modularity principle. Nevertheless, a human effort is still required to integrate the information conveyed by the navigation and locator maps. We think that the situation can be improved by combining both techniques. Thus, our research question is: is it possible to use both techniques together? Our analysis results in constructing a *positioning map* that combines both aforementioned techniques and, hopefully, reduces the cognitive effort needed to comprehend the overall model.

In Section 2 we present our research method. Then, in Section 3, we detail modularity and cognitive integration principles. In Section 4, we combine navigation and locator maps to construct a positioning map. In Section 5, we provide a preliminary evaluation of our proposal. Finally, we present the future work and conclude the paper in Section 6.

2 Research Method

To achieve the objective of this study, we follow the research method illustrated in Figure 1. This research method supports a wider investigation of how computer-aided visual techniques can facilitate effective communication of IS diagrams. First, we study the concrete syntax of a panel of IS modelling languages in order to understand how to improve layout of diagrams created using these languages. Next, we investigate PoEC in order to find out how we can use the different techniques they suggest. Next, we envisage the possibility of combining different PoEC together. The last step includes validation of different proposals.

In this paper we discuss only step 3 where we investigate the principles of *modularity* and *cognitive information integration* [Moo06a] [Moo06b] and try to combine together *navigation map* and *locator map*. But first, we briefly describe the purpose of the survey on IS modelling language processed at step 1.

A modelling language consists of an abstract and a concrete syntax as well as a semantics [HR04]. In this work, we are concerned with *concrete syntax*, and so is the scope of PoEC. We have conducted a survey on the concrete syntax of IS modelling languages. The main objectives of this survey are: a) to identify visualisation problems that are common and visualisation problems that are specific among modelling languages; b) to ground the decisions about solving the aforementioned problems. At this time, we have analysed more than ten modelling languages, such as UML, ER, ORM, and several goal-oriented languages like i^*, Tropos and KAOS. In this paper, we focus on general issues addressed in ER diagrams (see Figure 2 and Figure 3).

In order to have a quick means to test the application of PoEC on concrete syntax, we are

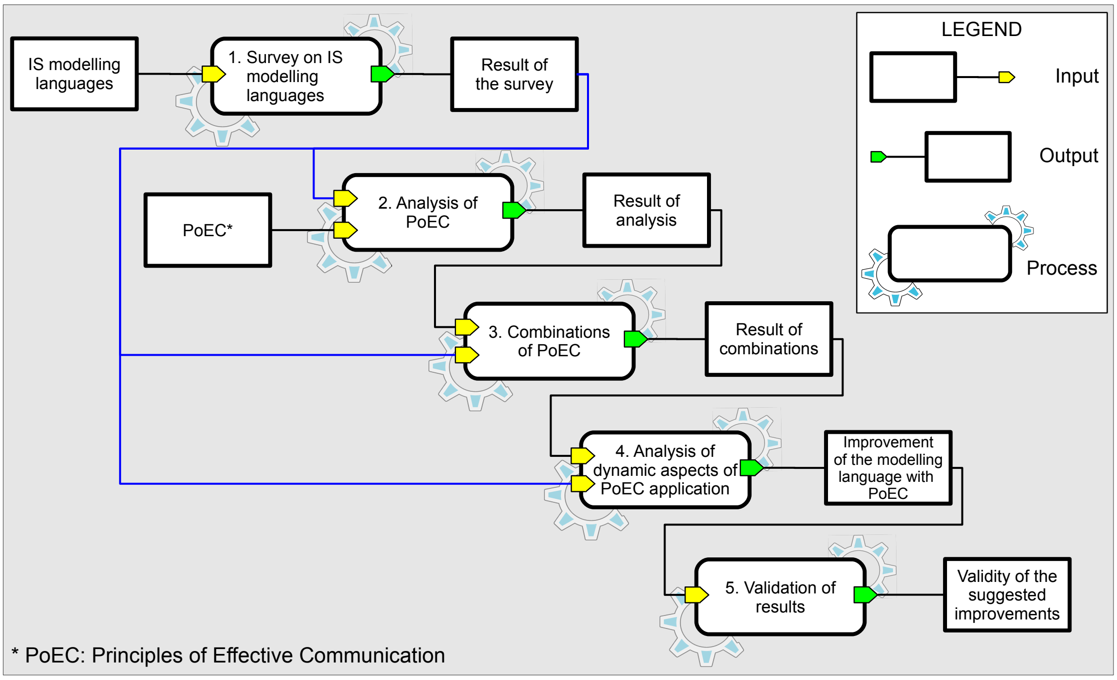

Figure 1: Illustration of our research method

using and extending the Graphical Symbolic Language (GraSyLa) [Eng00] [EH99]. GraSyLa is the declarative language used to define the concrete syntax of modelling languages in the MetaDONE tool [EH07]. MetaDONE is a so-called metaCASE tool, that is, a tool used to generated CASE (computer-aided software engineering) tools. It will be used in the validation step (step 5 in Figure 1 of our method).

3 Principles of Effective Communication

The principles of effective communication are presented in [Moo06b]. Recently they have been applied to analyse IS modelling languages like ER in [Moo02], KAOS in [MH07] and Tropos in [Bou08].

In this paper, out of the nine principles, we will only briefly recall modularity and cognitive integration. The *cognitive information integration* principle focuses on the mental mechanisms that the human mind uses to integrate information received from multiple sources (like e.g., sub-diagrams) to have a complete understanding of the overall content of the model. *Modularity (decomposition)* aims to reduce the complexity resulting from the size of diagrams and the number of visual elements that they contain. In this regard, the human mind has two main limitations. Firstly, it has a perceptual limit that concerns the ability to discriminate between diagram elements. Secondly, it is bounded by cognitive limitations: as initially described in [Mil56], on average, the human mind can deal with *seven plus or minus two* elements at the same time. Modularity is a common *divide-and-conquer* approach [Bro87] [Moo03] to reducing complexity. It is used in a wide range of disciplines: for example, in software programming, complex programs are modularised [Par02]; in cartography, complex maps are segmented [RMM+95]. In IS modelling, large diagrams are decomposed into sub-diagrams in order to reduce the number of interrelated elements. Modularity results in improvements of the diagram layout [Moo03] and helps to discriminate diagram visual elements. Decomposition strategies/algorithms are not discussed in this position paper because we use the set of diagrams resulting from those strategies as input for our new technique (described in Section 4). This aspect should be investigated in further research.

The principles of modularity and cognitive integration are closely interrelated: if we consider a large diagram as the representation of an overall model and if we decompose this large diagram into sub-diagrams as recommended by the modularity principle, the number of information sources (sub-diagrams) is then increased. To combine this information into a coherent mental representation of the model, the human mind needs to cognitively integrate information from each sub-diagram. The strong relationship between modularity and cognitive integration motivates our choice to combine them together.

4 Positioning Map

Here, we illustrate step 3 of our research method (see Figure 1) by combining the principles of modularity and cognitive integration. As discussed above, both these principles can lead to thwarting effects. On the one hand, modularity attempts to limit the cognitive overload sensation by creating new sub-diagrams from a complex one. On the other hand, cognitive integration speaks about summarising model information and sources of information (diagrams) to be understandable by humans. For example, to improve modularity, one can use a navigation map to decompose a model into different diagrams. A navigation map is defined as "a representation of the entire system of diagrams and the navigation paths between them. It corresponds to the concept of longshot in Human-Computer Interaction" [Moo03]. To deal with cognitive integration, one can apply a locator map to track the working area inside the overall diagram. A locator map is "a device used in cartography to show how a map fits into a larger region" [Moo03]. The locator map technique is implemented in some API (e.g., *satellite view* in Netbeans Visual Library[1]) and tools (e.g., *diagram overview* in Visual Paradigm IDE[2]). Figure 2 A provides an illustration. We see a navigation map indicating that the overall model is decomposed into diagrams 1 and 2. Diagram 2 is further decomposed into diagrams 3 and 4. The active diagram, currently being edited, is diagram 3. In the working area, we see a part of diagram 3 corresponding to the highlighted locator rectangle in the locator map. However, the use of these techniques as suggested in [Moo06b] and illustrated in Figure 2 A does not show how both principles can be applied together. It requires a cognitive effort from the user to see the link between locator map and navigation map. So the question remains: can we use both principles (modularity and cognitive integration) and their supporting techniques (e.g. navigation map and locator map) together?

In order to answer this question, we propose a *positioning map*, illustrated in Figure 2 B. In Figure 3, we describe how a positioning map can be constructed. It can be built in five steps:

1 http://graph.netbeans.org/
2 http://www.visual-paradigm.com

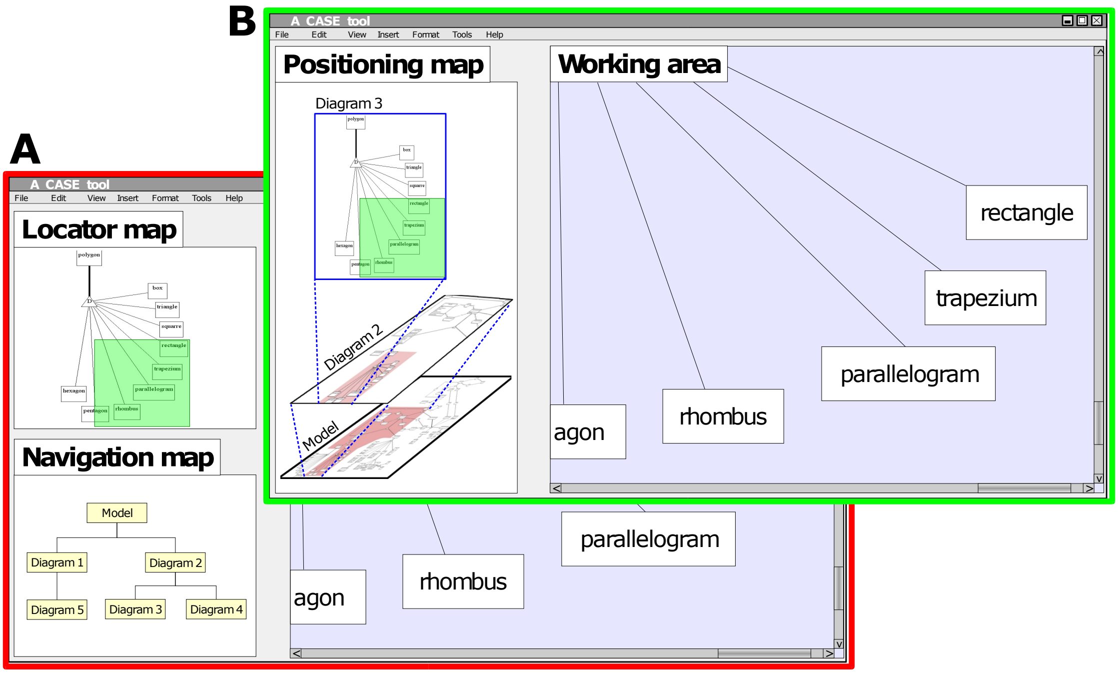

Figure 2: (A) a locator map and a navigation map and (B) a positioning map

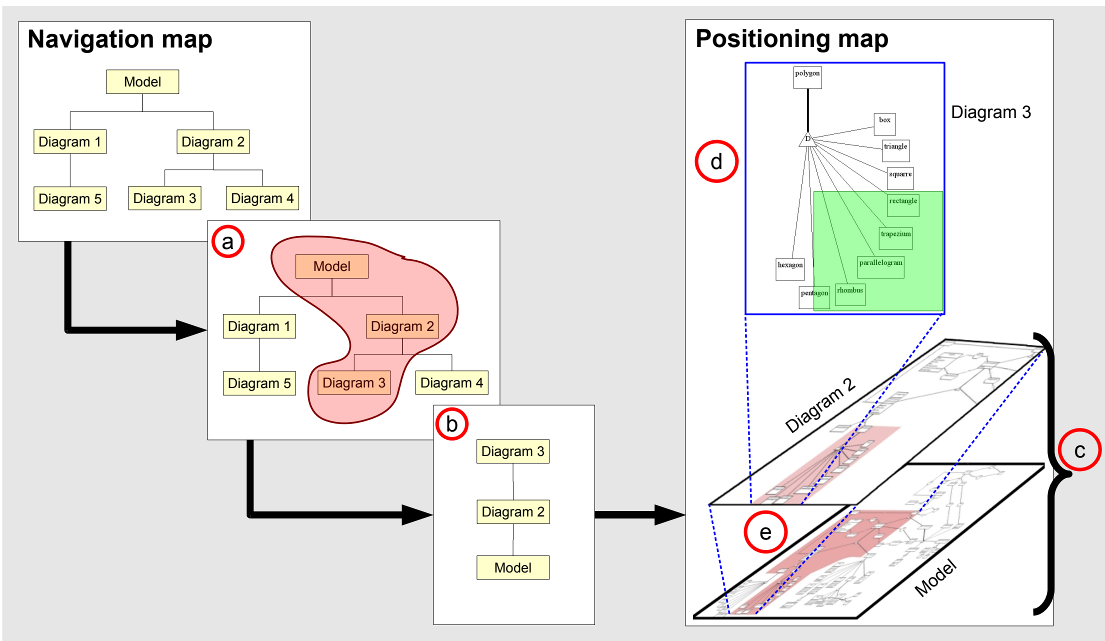

Figure 3: Building a positioning map

Table 1: Comparison navigation map + locator map *vs.* positioning map

Navigation Map + Locator Map	**Positioning Map**
Locator map: information related to cognitive integration.	The same information is available since the locator map is used as the top level node.
Navigation map: – the navigation map only shows the names of sub-diagrams and their decomposition relationships.	– the full model decomposition is displayed; – only information on the path from the root model until the top level node is shown; – relationships between hierarchy nodes are defined at groups-of-elements granularity; – extra information: a) how elements of a diagram are grouped and modularised into a sub-diagram; b) how a sub-diagram fits into its parent diagram.

(a) we need to select the active nodes in the root models. In our case, active nodes are diagrams "Model", "Diagram 2" and "Diagram 3"; (b) we reverse the hierarchy of the navigation map; (c) we replace each node – except the top level node – with a miniature of the diagram it is related to; (d) in the top level diagram (here "Diagram 3"), we display the information provided in the locator map as illustrated in Figure 2 A; (e) we need to highlight the information in the regions of each lower level diagram where the sub-diagram appears.

5 Discussion

Did we manage to combine together the two principles of modularity and cognitive integration? Table 1 answers this question by comparing the contextual information conveyed by the navigation map + the locator map with the information of the positioning map. The information appears not to be equivalent. But, the aim of combining the modularity and cognitive integration principles is primarily to relieve the effort required for understanding the represented information. Showing in the positioning map how sub-diagrams fit into their parent diagram seems to reduce this effort. Moreover, we think that presenting the whole decomposition hierarchy is not required. Thus, omitted information in the positioning map (*wrt* navigation map) is not expected to hamper understandability of the overall model.

Also note that we illustrated how to build the positioning map from an existing locator map and an existing navigation map. However, the positioning map can also be constructed from scratch by decomposing the model into different diagrams (like b and d in Figure 3) without creating navigation and locator maps initially.

6 Conclusion

In this paper we investigated how it is possible to combine two principles (model decomposition and cognitive information integration) and their respective techniques (navigation and locator maps) for effective communication [Moo06b]. The combination results in a new technique called *positioning map*. We believe that the positioning map is a better means than locator + navigation maps to decompose models and to integrate information cognitively. However, our proposal needs to be validated empirically. Our future work, in the scope of the positioning map, includes an implementation of this (and other PoEC) technique(s) in MetaDONE (using GraSyLa) and testing its (their) validity.

Acknowledgements: This work was supported in part by the Interuniversity Attraction Poles Programme - Belgian State - Belgian Science Policy (MoVES).

Bibliography

[BG03] A. Blackwell, T. Green. Notational systems – The Cognitive Dimensions of Notations Framework. In Carrol (ed.), *HCI Models, Theories, and Frameworks: Toward a Multidisciplinary Science*. Pp. 103–133. Elsevier, 2003.

[Bou08] Q. Boucher. Visually Effective TROPOS Models. Master's thesis, University of Namur, 2008.

[Bro87] F. P. J. Brooks. No Silver Bullet Essence and Accidents of Software Engineering. *Computer*, pp. 10–19, 1987.

[EH99] V. Englebert, J.-L. Hainaut. GRASYLA: Modelling Case Tools GUIs in Metacases. In Vanderdonckt and Puerta (eds.), *Proc. of the 3rd Conference on Computer-Aided Design of User Interfaces. (CADUI'99)*. Pp. 217–230. Kluwer Academic Publishers, 1999.

[EH07] V. Englebert, P. Heymans. Towards More Extensible MetaCASE Tools. In Krogstie et al. (eds.), *Proc. of Advanced Information Systems Engineering, 19th International Conference (CAiSE '07)*. Pp. 454–468. Springer, 2007.

[Eng00] V. Englebert. *A Smart Meta-CASE: Towards an Integrated Solution*. PhD thesis, The University of Namur, May 2000.

[HR04] D. Harel, B. Rumpe. *Computer*. Chapter Meaningful Modeling: What's the Semantics of "Semantics", pp. 64–72. IEEE Computer Society Press, 2004.

[KLM+97] G. Kiczales, j. Lamping, A. Mendhekar, C. Maeda, C. Videira Lopes, J.-M. Loingtier, J. Irwin. Aspect-Oriented Programming. In *Proc. of the European Conference on Object-Oriented Programming (ECOOP 1997)*. P. 220. Springer, Jun 1997.

[Kro01] J. Krogstie. Using a Semiotic Framework to Evaluate UML for the Development of Models of High Quality. *Unified Modeling Language: Systems Analysis, Design and Development Issues*, pp. 89–106, 2001.

[MH07] R. Matulevičius, P. Heymans. Visually Effective Goal Models Using KAOS. In Hainaut et al. (eds.), *Proc. of Advances in Conceptual Modeling - Foundations and Applications (RIGiM '07)*. Lecture Notes in Computer Science 4802, pp. 265–275. Springer, 2007.

[Mil56] G. Miller. The Magical Number Seven, Plus or Minus Two: Some Limtis on Our Capacity for Processing Information. *Psycological Review*, pp. 81–97, 1956.

[Moo02] D. L. Moody. Complexity effects on end user understanding of data models: An experimental comparison of large data model representation methods. In *Proc. of the 10th European Conference on Information Systems, Information Systems and the Future of the Digital Economy (ECIS '02)*. 2002.

[Moo03] D. L. Moody. Dealing with Complexity in Information Systems Modeling: Development and Empirical Validation of a Method for Representing Large Data Models. In *Proc. of the International Conference on Information Systems (ICIS 2003)*. Pp. 207–221. Association for Information Systems, December 2003.

[Moo06a] D. L. Moody. Dealing with "Map Shock": A Systematic Approach for Managing Complexity in Requirements Modelling. *in Requirements Engineering: Foundation for Software Quality (REFSQ'06)*, 2006.

[Moo06b] D. L. Moody. What Makes a Good Diagram? Improving the Cognitive Effectiveness of Diagrams in IS Development. In *Proc. of the 15th int.conf. in Information Systems Development (ISD 2006)*. 2006.

[NH98] N. H. Narayanan, R. Hübscher. Visual Language Theory: Towards a Human Computer Interaction Perspective. *Visual language theory*, pp. 87–128, 1998.

[Par02] D. Parnas. On the Criteria to be Used in Decomposing Systems into Modules. *Software pioneers: contributions to software engineering*, pp. 411–427, 2002.

[RMM$^+$95] A. H. Robinson, J. L. Morrison, P. C. Muehrcke, A. J. Kimerling, S. C. Guptill. *Elements of Cartography, 6th Edition*. John WILEY and Sons LTD, April 1995.

[Ros78] E. Rosch. Principles of Categorization. In Rosch and Lloyd (eds.), *Cognition and Categorization*. Pp. 27–48. Lawrence Erlbaum, 1978.