

INTEGRATION AND COLLABORATION BASED ON GRAPHICAL TECHNIQUES EDITORIAL

Editors: Hartmut Ehrig^{*)}, Michael Goedicke^{)}, Julia Padberg^{*)}**

^{*)}Dept. of Computer Science, Technical University, Berlin, Germany

^{**)}Dept. of Mathematics and Computer Science, University of Essen, Essen, Germany

Graphic and diagrammatic techniques have always been very popular for defining and communicating software system properties. Engineering in general has a whole body of drawing techniques to design complex systems. In software engineering research and practice tried to emulate these drawing techniques and early approaches like ER and SA diagrams¹⁾ are still in use today. Recently object orientated analysis and design has lead to numerous similar graphic notations where we now observe UML (Unified Modelling Language) as a kind of industrial standard being established.

Despite their popularity within the software and requirements engineering communities such diagrammatic description techniques are also disqualified by regarding them at most semi formal or even up to interpretation by the reader. This feature - regarded by some as a virtue - make these obviously useful techniques not applicable in the context of critical work like flight deck management systems, automatic rail road crossing or medical applications. The current lively discussion wrt. a formal basis for UML shows need and the difficulties arising in such a problem area.

However, looking into the detail of the actual problem it is revealed that there is a small number of problems with graphic / diagrammatic description techniques:

- there are many different graphic techniques - as with formal textual descriptions - suitable for different needs and it is necessary to combine them in a structured way
- graphic techniques have to be scalable i.e. it must be possible to define a hierarchy of interacting diagrams
- graphic techniques must be precise or even formal on the one side and easily adaptable to a new

1. The acronym ER stands for Entity Relationship diagrams – a diagrammatic representation scheme used often to describe database schemata – and SA stands for System Analysis diagrams which are used to formulate hierarchies of interconnected processes.

context on the other side

In this issue of the SDPS Journal a number of interesting contributions have been collected addressing the aforementioned needs in graphic specification techniques. These contributions are extended versions of papers presented at the 3rd International Conference on Design and Process Technology held in Berlin in July 1998. For all contributions applies that they have a formal mathematical semantics i.e. they try hard to provide a sound basis for system analysis and synthesis. Also all contributions focus on the practical aspect of their work thus avoiding to go into too much detail wrt their mathematical foundations.

In particular the contribution by Ehrig, Padberg, Heckel, Schnieder and Jansen „Cooperability in Train Control Systems Specification Scenarios Using Open Nets“ the problem is addressed to tackle the numerous aspects of a complex system by formally defining scenarios which are then combined to form a complete and consistent specification of that system. They apply their techniques to the well known and challenging problem of the rail road crossing problem. Various results regarding the cooperability and ways to combine different graphic notations are described.

The contribution by Wirsing and Kosiuczenko „Towards the Integration of a Graphical and a Formal Specification Method: Message Sequence Charts and Timed Maude“ the problem domain is addressed to make a given graphic representation scheme precise. Here the popular graphic representation scheme Message Sequence Chart (MSC), which is part of UML and SDL as well, are defined precisely by giving a translation into a formal logic including the important time aspect.

As indicated above graphic description techniques are often not very flexible in adopting a new context. For example, if one chooses Petri Nets it is necessary to pick one out of many. There is not the one and only Petri Net Notation. In the contribution „Flexible Integration of Petri Net Based Process Description with User-Specific Data Description“ by Gruhn and Lembke an approach is presented to address the problem to make Petri Nets flexible wrt the way the data aspect is represented. This work leads to a better understanding what a choice of a given Petri Net approach in combination with a data representation scheme implies for the system analysis and synthesis capabilities of the given combination.

Rule based approaches to system description are always attractive since they provide a compact way to define system properties. The other two contributions of this issue deal with rule based approaches for graphic description techniques in the realm of the design of man machine interfaces. In the contribution „Hierarchical Specification of Graphical User Interfaces using a graph grammar approach“ by Goedicke, Tröpfner and Enders-Sucrow the idea is pursued to define a graphical user interface (GUI) using the idea of graph transformation rules and visual elements of the GUI- domain representing widgets like buttons, windows, scrollbars. In particular the need arises for describing complex GUIs to provide hierarchic descriptions. Also the formal basis in form of graph transformation systems (graph grammars) is discussed as well.

The contribution „Describing a Continuous Collaborative Specification Process of Human- Computer Interaction by Graph Rewriting“ by Enders-Sucrow discusses a formal approach to man machine interface design using the information resources approach developed at the University at York. The problem addressed in the contribution here is to provide a formal basis for the information resources approach. It is shown that the graph rewriting as underlying formal basis provides the results necessary to define important notions for the process of defining complex man machine interfaces. In the case study discussed here important aspects of the flight management system are defined and refined across a number of levels of detail.

In summary the various contributions in this issue show that graphical and diagrammatic techniques can be equipped with a sound formal basis and that all the nice properties at the informal level these representation scheme offer can be represented at the formal level as well. Thus we hope that this work offers another step to develop and integrate graphical techniques in such a way that the engineer can trustfully apply such techniques also for critical applications.