# A Reference-Scheme for Describing Organizations

**Andreas Winter**[1]

January 1996

**Abstract**

A reference-scheme is presented, which can be used for integrated modeling of organizations. This model is also suited for comparing different approaches of organization-modeling and for building tools supporting organization-modeling in an multi-paradigmatic manner.

## 1 Introduction

Graphical languages such as dataflow diagrams, entity-relationship diagrams or organization charts are used to describe various aspects of organizations. These languages have their origins in the field of software-engineering and organizational theory. Looking at organizations regarding Grochlas sociotechnical systems [Grochla, 1978, p. 10] these languages describe

- the ***rules***, which have to be followed to reach the aim of the organization,
- the ***tasks*** including processes and objects, which have to be mastered by the organization,
- the ***tools***, used to solve the tasks, and
- the ***people*** working in the organization and their relationships.

The *rules*, given by laws can be described by predicates. A lot of graphical or textual languages like task-lists, task decompositions, data-flow-diagrams, nets, control-flow-diagrams, Nassi-Shneiderman-diagrams, state-charts, etc. are used to describe *procedural aspects of tasks*. *Objects* which are worked at can be described by use of entity-relationship-diagrams or object diagrams. These process and object-related languages are also used to describe more software-technical aspects of *tools* used in the organization. To describe the *personal interdependences* diagrams like organization charts are given. Figure 1 shows some examples of languages concerning procedural and object related aspects of hospital information systems.

Using one diagrammatic language seperately shows only some small view to the organization. To describe organizations as a whole, an integrated approch combining these languages has to be choosen.

Models and reference models of large organizations e.g. hospital information systems or adminstrative authorities are also build by using these diagramming techniques. Parts of models are often developed by different goups. All working groups may use their favourite modeling languages varying in different dialects (cp. [Martin/McClure, 1985]). To build the total model these partial models have to be integrated.

Consequently an integrated ***approach*** covering all relevant aspects of organizations and all the known languages as completely as possible is desired. To achieve this, the concepts described by each of these languages have to be represented in the same way in a ***common formalism*** into an ***integrated scheme***. A specific organization, represented as an *instance* to this *scheme* can be viewed by using any desired graphical notation included in the scheme. This leads to an ***integrated multi paradigmatic description*** of organizations.

Literature about business-process-modeling, workflow-modeling or civil-service-modeling gives examples of integrated structures representing organizations. Bußler and Jablonski model workflows through *functional*, *behavioural*, *informational*, *structural* and *operational aspects* [Bußler/Jablonski, 1996]. The ARIS-Architecture (e.g. [Scheer, 1992], [Scheer, 1994]) uses the concepts *process*, *event*, *state*, *processor*, *organizational unit* and *resources*. The SOM-Approach (e.g. [Ferstl *et al.*, 1994], [Ferstl/Sinz, 1996]) models business processes with the concepts *transaction*, *object*, *task*, *event* and their dependencies. A scheme modeling civil service organizations consisting of *processes*, *organizational units* and *documents* is presented by Engel [Engel, 1995]. The concepts *participant*, *role*, *activity*, *object*, *organizational unit* and *tool* build the structure shown by Gruhn [Gruhn, 1994]. These approaches use different concepts and different graphical notations for organization modeling. They also emphasise different aspects of organizations. A ***comparison of these approaches*** in a neutral manner can only be done by using a ***common reference-model***.

Modeling large organizations requires ***tools*** supporting investigation, documentation, presentation and analysis of these structures. To build these tools, ***efficient data structures*** representing all relevant organizational aspects are wanted. This is given by an ***integrated scheme*** or its subschemes.

[1] Andreas Winter, University of Koblenz-Landau, Institute for Software-Technology, Rheinau 1, D-56075 Koblenz, Germany, winter@informatik.uni-koblenz.de

The phd-thesis will show

- how such a ***scheme*** can be developed,
- how it can be used for ***describing organizations***,
- how it can be applied for ***comparing different modeling-approaches***, and
- how it can be used for ***building integrated tools*** supporting organization modeling.

# 2 Building the scheme

A general formal modeling method is needed in order to build a scheme representing different graphical notations uniformly. This is given by the approach of ***graph based modeling*** [Ebert/Franzke, 1995].

Objects and their relationships are represented by *typed*, *attributed* and *ordered directed graphs* (***TGraphs***). Objects are represented by exactly one vertex and the relationships between them are represented by edges. Similar objects or relationships can be subsumed under vertex- resp. edge-types. Further information can be annotated to vertices and edges by attributes.

TGraphs representing structurally similar information are modeled as *instances* of the same ***graph class***. Graph classes describe the *schemes*, concrete graphs corresponds to. Graph classes leeds to *formal models* which allow the integration of submodels, to *expressive diagrams* which allow discussions and comparisons of concepts and to *efficient data structures*, which allow implementing efficient tools. A variant of extended entity-relationship-diagrams including predicative annotations is used to define these schemes [Carstensen *et al.*, 1994].

Graphical languages from the field of organization modeling or from software-engineering are used for modeling different aspects of organizations. Organizations are described from

- the ***task-view*** which looks at tasks and their decomposition,
- the ***structural-view*** which looks at organizational units, persons and their interdependencies,
- the ***process-view*** which looks at single tasks, and
- the ***object-view*** which looks at the objects (material, data etc.) used in executing the tasks.

To describe the *task-view*, languages following the task-decomposition-paradigm (e.g. task-lists, decomposition-diagrams) are used. The *structural view* is described by languages of the structure-paradigm (e.g. organization-charts). The languages presenting the *process-view* follow four paradigms. The net-paradigm includes Petri-nets and networks. Languages like flow charts, decision-tables or Nassi-Shneiderman-diagrams belong to the control-flow-paradigm. In the state-transition-paradigm techniques like state-transition-diagrams or State-Charts to describe reactive aspects can be found. Flows of material or data are modeled by languages of the data-flow-paradigm e.g. data-flow-diagrams or SADT-diagrams. The object-view is described by languages of the object-relationship-paradigm which includes entity-relationship-diagrams or data-dictionaries. A closer look to these paradigms is given in [Ebert/Engels, 1993].

In the ***scheme for describing organizations*** each paradigm is modeled representing the relevant concepts and their relationships as well as their integration [Winter/Ebert, 1996].

Languages overlapping the sketched organizational views are represented in this scheme too. Information used for those representations (e.g. function-chains or mechanism annotated data-flow-diagrams) can be found following the respective relationships.

The following sections will sketch the approach of describing organizations, of comparing modeling-approaches, and of building tools supporting organization modeling based on this scheme.

# 3 Scheme-based description of organizations

A graph satisfying the scheme represents a concrete organization. Graphical or textual notations of this organization can be found by extracting the relevant information from this data-structure and translate it into concrete languages.

A graph corresponding to the scheme is shown in the middle of figure 1. This graph represents a small part of a hospital information system from the object and from the process-view including interdependencies between both views.

The process-view can be represented by languages following the dataflow-paradigm. The light gray part of the graph shows the relevant subgraph which delivers information for concrete notations with dataflow-diagrams (upper left corner) or SADT-diagrams (upper right corner). Analogously the dark grey subgraph models the object-view, which can be drawn as entity-relationship-diagram (lower left corner) or as simple data-dictionary entries (lower right corner).

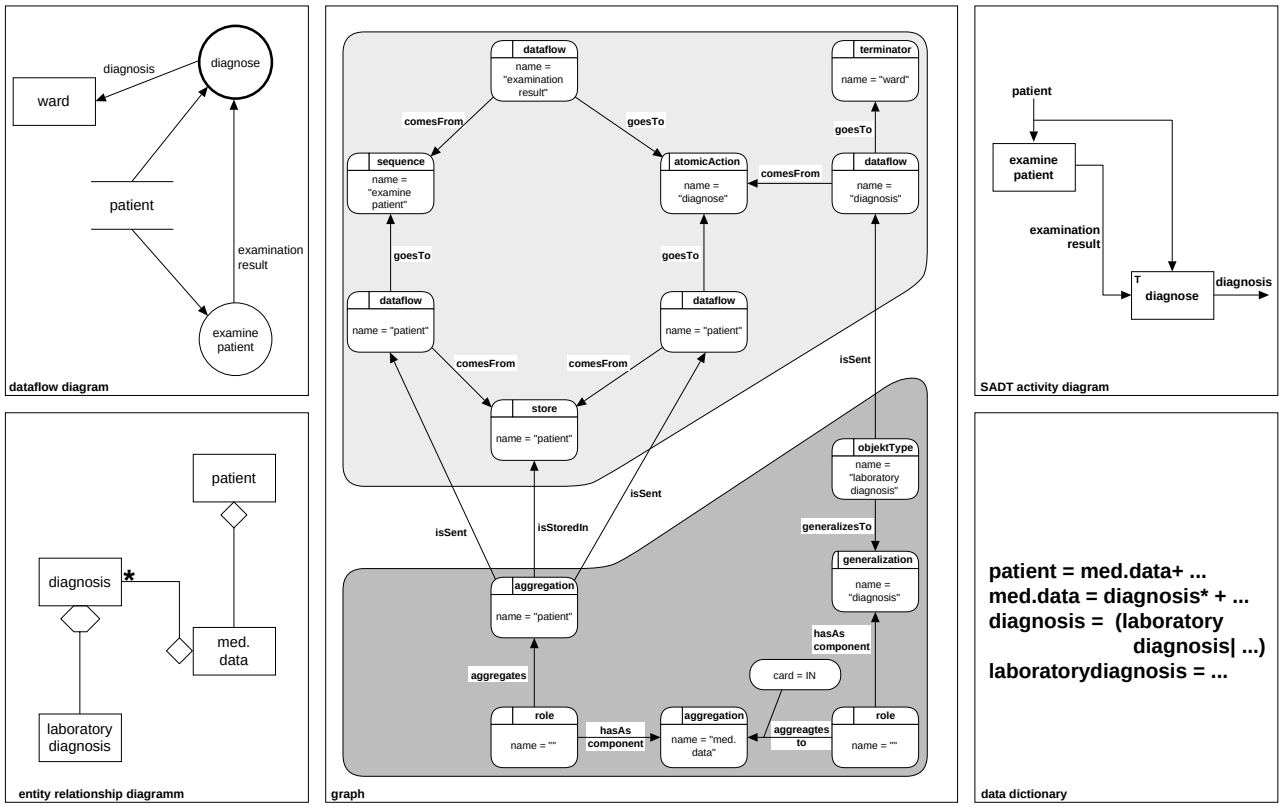


Figure 1: Graph and concrete notations

# 4 Scheme-based comparison of modeling approaches

Following the claim representing most of the known organization modeling languages, the scheme developed in section 2 delivers a ***reference-scheme***. This gives the base for comparing the schemes used in other organization modeling approaches in an neutral manner.

The organizational concepts and their interdependencies building the foundations of other organization modeling approaches as sketched in section 1 can be modeled with the same techniques used for building the reference-scheme. By this way, the reference-scheme and the schemes belonging to the other approaches are represented in the same calculus.

Since the known approaches only emphasize single aspects of organizations using few concete diagraming techniques their schemes will result in subschemes of the reference-scheme.

# 5 Scheme-based tools supporting organization modeling

By use of the graph-based semantics defined for the entity-relationship-dialect a ***data-structure*** for building organization modeling tools is given by the scheme.

So the the scheme for describing organizations gives the conceptual part of a tool in a declarative way. Building a tool supporting organization modeling this has to be extended by functional aspects and the concrete syntaxes for the wanted graphical and textual languages.

With these information a ***tool supporting investigation, documentation, presentation and analysis of organizations*** can be built with the KOGGE-approach [Ebert, 1995], which gives a generator for graphical design environments from declarative descriptions of design models.

A tool supporting organizational modeling for evaluating software-products in the field of concretes application is under construction. This tool bases on a subscheme of the reference-scheme supporting organization charts, data-flow-diagrams, SADT-diagrams, task-lists, task decompositions, entity-relationship-diagrams and data-dictionaries.

# 6 Conclusion and Outlock

An integrated reference-scheme for descibing organizations is presented. This scheme can be used for multi-paradigmatic description of organizations, as a reference for comparing organization modeling approaches and as a specification of a data-structure for modeling and analyzing tools.

Further work will be concerned with the completion of the reference-scheme and its extension to not yet modeled languages. Further subschemes for selected organization modeling approaches will be derived. In parallel a prototype-tool to support organization modeling in the field of software-evaluation is being developed.

# References


[Bußler/Jablonski, 1996] **C. Bußler, S. Jablonski**. Die Architekur des modularen Workflow-Management-Systems MOBILE. *in J. Becker, G. Vossen (Hrsg.): Geschäftsprozeßmodellierung und Workflows, Thomson, Bonn*, S. 369–388, 1996.

[Carstensen *et al.*, 1994] **M. Carstensen, J. Ebert, A. Winter**. Deklarative Beschreibung von Graphsprachen (Erweiterte Kurzfassung). Friedemann Simon (Hrsg.): Tagungsband zum Workshop „Deklarative Programmierung und Spezifikation“, der GI-Fachgruppe 2.1.4 Alternative Konzepte für Sprachen und Rechner, 9.-11. Mai 1994, Bad Honnef, Nr. 9412, Institut für Informatik und praktische Mathematik, Universität Kiel, Kiel, September 1994.

[Ebert/Engels, 1993] **J. Ebert, G. Engels**. Design Representation. In *J. J. Marciniak (eds.): Encyclopedia of Software Engineering*, S. 382–394. John Wiley & Sons, New York, 1993.

[Ebert/Franzke, 1995] **J. Ebert, A. Franzke**. A Declarative Approach to Graph Based Modeling. *in: E. Mayr, G. Schmidt, G. Tinhofer (Eds.)* Graphtheoretic Concepts in Computer Science *Springer, Berlin, LNCS 903*, S. 38–50, 1995.

[Ebert, 1995] **J. Ebert**. KOGGE: A Generator for Graphical Design Environments. to appear, 1995.

[Engel, 1995] **A. Engel**. Vorgangsbearbeitung im Informationsverbund. *in: Huber-Wäschle et. al.: GISI 95, Herausforderungen eines globalen Informationsverbund für die Informatik, Berlin, 1995*, S. 118–126, September 1995.

[Ferstl/Sinz, 1996] **O. K. Ferstl, E. J. Sinz**. Geschäftsprozeßmodellierung im Rahmen des Semantischen Objektmodells. *in J. Becker, G. Vossen (Hrsg.): Geschäftsprozeßmodellierung und Workflows, Thomson, Bonn*, S. 47–61, 1996.

[Ferstl *et al.*, 1994] **O. K. Ferstl, E. J. Sinz, M. Amberg, U. Hagemann, C. Malischewski**. Tool-Based Business Process Modelling using the SOM Approach. Bamberger Beiträge zur Wirtschaftsinformatik 19, Universität of Bamberg, Business Informatics, Bamberg, 1994.

[Grochla, 1978] **E. Grochla**. *Einführung in die Organisationstheorie*. C.E. Poeschel, Stuttgart, 1978.

[Gruhn, 1994] **V. Gruhn**. Interpersonal Process Support Systems — Purpose and Architecture. submitted to COMPSAC'95, March 1994.

[Martin/McClure, 1985] **J. Martin, C. McClure**. *Diagramming Techniques for Analysts and Programmers*. Prentice Hall, Englewood Cliffs, 1985.

[Scheer, 1992] **A.-W. Scheer**. *Architecture of Integrated Information Systems: Foundations of Enterprise Modeling*. Springer, Berlin, 1992.

[Scheer, 1994] **A.-W. Scheer**. *Wirtschaftsinformatik, Referenzmodell für industrielle Geschäftsprozesse*. Springer, Berlin, 5. Auflage, 1994.

[Winter/Ebert, 1996] **A. Winter, J. Ebert**. Ein Referenz-Schema zur Organisationsbeschreibung. *in J. Becker, G. Vossen (Hrsg.): Geschäftsprozeßmodellierung und Workflows, Thomson, Bonn*, S. 101–123, 1996.