METAMODELS OF INFORMATION TECHNOLOGY BEST PRACTICES FRAMEWORKS

Arthur Nunes Ferreira Neto
João Souza Neto
Catholic University of Brasilia, DF, Brazil

_____

ABSTRACT

This article deals with the generation and application of ontological metamodels of frameworks of best practices in IT. The ontological metamodels represent the logical structures and fundamental semantics of framework models and constitute adequate tools for the analysis, adaptation, comparison and integration of the frameworks of best practices in IT. The MetaFrame methodology for the construction of the metamodels, founded on the discipline of the conceptual metamodelling and on the extended Entity/Relationship methodology is described herein, as well as the metamodels of the best practices for the outsourcing of IT, the eSCM-SP v2.01 (eSourcing Capability Model for Service Providers) and the eSCM-CL v1.1 (eSourcing Capability Model for Client Organizations), constructed according to the MetaFrame methodology.

Keywords: Framework; Best Practices in IT; Metamodels; eSCM.

1. INTRODUCTION

According to the IT Governance Institute (2005), "the survival and success of an organization in light of the new globalized market, where time and distance are suppressed, are under the effective management of information and related technology". In light of this context, in which IT (Information Technology) takes a decisive role within organizations, the models or frameworks of best practices in IT have emerged in the last two decades. These frameworks are the enterprise's and academia's response to the challenges of management and governance of IT functioning as instruments for the promotion and alignment between the processes of IT and the strategic objectives of the organization.

_____

Arthur Nunes Ferreira Neto, Master in Information Technology and Knowledge Management – MGCGI/UCB, Catholic University of Brasilia, Campus Avançado, SGAN 916 Asa Norte - Modulo B - Sala A111 - CEP: 70.790-160 Brasília – DF, Brasil - Telefone: (61) 3338-6534 - E-mail: arthurnetobsb@gmail.com.

João Souza Neto, Doctor of Science in Electrical Engineering, University of Brasilía – UNB, Professor at Catholic University of Brasilía, on the Master's degree Program in Information Technology and Knowledge Management, Campus Avançado, SGAN 916 Asa Norte - Modulo B - Sala A121 - CEP: 70.790-160 Brasília – DF, Brasil - Telefone: (61) 3448-6534 - E-mail: joaon@ucb.br.

According to Johannsen and Goeken (2007), the frameworks for best practices in IT "describe organizational objectives, processes and aspects of the management and control of IT". Among the main frameworks of IT best practices currently used in organizations, we have the eSCM (eSourcing Capability Model), the CobiT (Control Objectives for Information and Related Technology), CMMI (Capability Maturity Model Integration), PMBoK (Project Management Body of Knowledge) and ITIL (Information Technology Infrastructure Library).

The effective implementation of an IT best practices framework is a complex activity that demands planning and that normally brings significant changes in the organization and in its processes. The challenge then arises to understand, in depth, the structure of the framework so that a preliminary study of its adoption on the processes of the organization can be done.

Besides this, it is noted that the adoption of just one of these IT best practices frameworks may not be sufficient for a particular organization. Despite the different focuses and the conceptual and structural differences, the IT best practices frameworks are not, in principle, incompatible, and can be used concomitantly to promote the improvement of the organization's information technology management. However, one of the challenges currently faced in IT management is how to analyze, adapt, compare and integrate the different frameworks of IT best practices.

It is understood, consequently, that the first step towards solving these problems is by understanding the logical structures and generating semantics of the IT best practices frameworks. This can be achieved by generating ontological metamodels of these frameworks.

The metamodels of the ontological type were identified and defined in the studies by Atkinson and Kühne (2003a and 2003b) and will be considered in the theoretical reference of this article. We highlight that the creation of a domain ontology for the IT best practices frameworks is not part of the present work, but it will be the object of further studies.

Among the main approaches used, up to now, to make the analysis and the comparison of the IT best practices frameworks, we have high level classifications, based on diverse criteria of comparison and the high level, detailed mapping, of the functions and processes between frameworks (ITGI, 2006, 2008).

Meanwhile, just the application of these two approaches does not contribute significantly to the solution of the problem of comparison and integration of the frameworks. The high level classifications, based on comparison criteria, are not detailed enough to detect the correspondences or incoherencies between different areas of the IT best practices frameworks. At the other extreme, the detailed mapping of the functions and processes of the frameworks possess a high level of detail, but little information to understand the conceptual and logical structures are available, which are important for planning and effective integration.

In an effort to fill this gap, the MetaFrame methodology, which joins procedures, strategies and instructions for the creation of ontological-type metamodels for IT best practices frameworks is presented in this article. An example of the application of the MetaFrame methodology is also presented: the generating of the metamodelof the eSCM-SP v2.01 01 (Hyderet all, 2006) and the eSCM-CL v1.1 (Hefley e Loesche, 2006).

1. THEORETICAL REFERENCE

1.1.   Ontological Metamodels

The managing of elements of an organization increasingly makes use of models, tools, and environments of modeling even more complex. For Karagiannis (2002), the state of the art in the area of organizational modeling is based on metamodels.

One can begin a literal analysis of what metamodel means by the prefix "meta". In Greek, "meta" means "that which is beyond", "that which encompasses", "that which supersedes", "that which transcends". We use the prefix "meta" when a certain operation is completed twice. For example, a dialogue about how to conduct a dialogue is a metadialogue. Summarizing, "meta" is put before some **f** operations, so that it is characterized as being completed twice. Instead of writing   **ff**, as in modelmodel, meta**f** is used, that is, metamodel. For another application subsequent to the operation, another prefix, "meta", is added, for example, metametamodel.

For Kühne (2005) author of a semantic formalization for the metamodels, a model can be thought of as a projection, which implies that something (the original) is projected and that part of the information is lost during the projection, in the activity called abstraction. The part that is retained depends on the proposition to which the model will be used. The author uses the definition of a model as an abstraction of a system (real or based on a language), permitting predictions or inferences to be made.

According to the open consortium of the OMG (Object Management Group), responsible for the MDA (Model-Driven Architecture, 2003) and UML (Unified Modeling Language, 2004) specifications, a model is an instance of a metamodel, which implies that a metamodel is a model of another model.

An important contribution to the studies on the subject of this article was provided by Atkinson and Kühne (2003a and 2003b), which identifies two dimensions of metamodelling, giving rise to two distinct forms of instancing of objects of the metamodel (linguistic and ontological). One dimension is related to the definition of the language and makes use of linguistic instantiation, used, for example, in MDA architecture, the basis of UML language. Another dimension is about the definition of the domain or type of object and uses the ontological instancing used in the creation of the metamodels of frameworks of best practices in IT in this study. Both forms occur simultaneously and serve to precisely locate an element of the model in the linguistic-ontological space.

In figure 1, the OMG_MDA architecture with four layers of abstraction (M_0 toM_3) is used, also followed by UML2.0 and MOF 2.0 linguistic modeling standards. We have the visualization of a linguistic metamodel with four horizontal layers that begin by M_0, denoting the lowest level, and M_3, the highest level of abstraction. At the same time, we have the visualization of the ontological metamodel, represented by different areas separated by a dashed line in the vertical division at the M_1 level. By explaining the two metadimensions, Figure 1 also illustrates the relationship between the elements of the model and the real world. The dog and the lamp (mental concept) of the M_0level are the elements of the real world to be modeled. The real Lassie is "represented" by the object Lassie and not by an 'instance of' Collie. The abstraction level M_1 contains the first level of abstraction of an object in the real world, together with the type of which the object is an ontological instantiation. The Lassie object (O_0)

is an ontological instantiation of the type Collie (O_1). From M_1 each level is a model expressed in the language defined at the higher level. In M_2, the Lassie object is a linguistic instantiation of the Object type, which, in M_3 is a linguistic instance of the Class type.
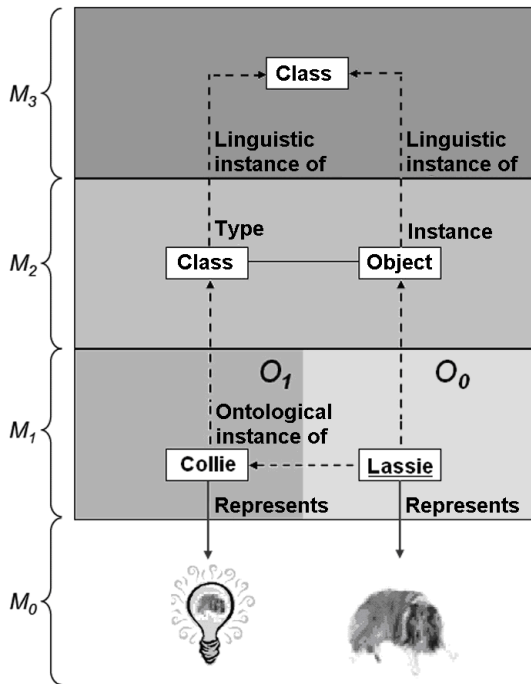

Figure 1: The Linguistic Metamodel (Adapted from Atkinson and Kühne, 2003b)

The ontological metamodels use the 'instance of' relationship to relate the concepts with their types or metatypes. In Figure 2, we extend the ontological levels rotating Figure 1 to the right, and adding level O_2. In this way, the ontological metalevels are arranged horizontally. For Atkinson and Kühne (2003b), the two points of view are equally valid and useful.
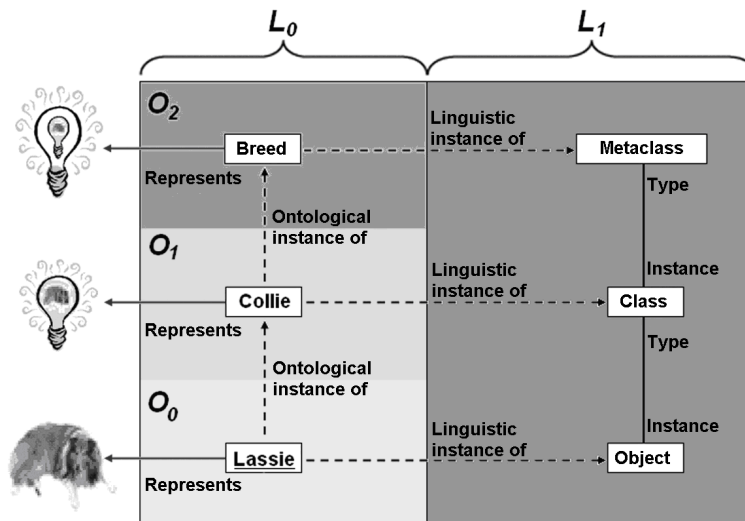

Figure 2: The Ontological Metamodel (Adapted from Atkinson and 2003b).

The utility of using metaconcepts has been recognized for a long time. For example, using meta concepts with races and species brings enormous advantages. Figure 3 shows one of the most mature ontological metamodels, the biological taxonomy for human beings. One notes that the Lassie object is an ontological instance of the Collie type or class, which is at the O_1 level or level of the model. The Collie type, on the other hand, is an ontological instance of the Race type which is in O_2, or at the metamodel level. One can see, from the UML notation, that the Collie type is also a specialization of the CanisFamiliares type (domestic dog) at the same ontological level O_1. The presence of the other metaconcepts like Species, Genus etc. is perceived in O_2.
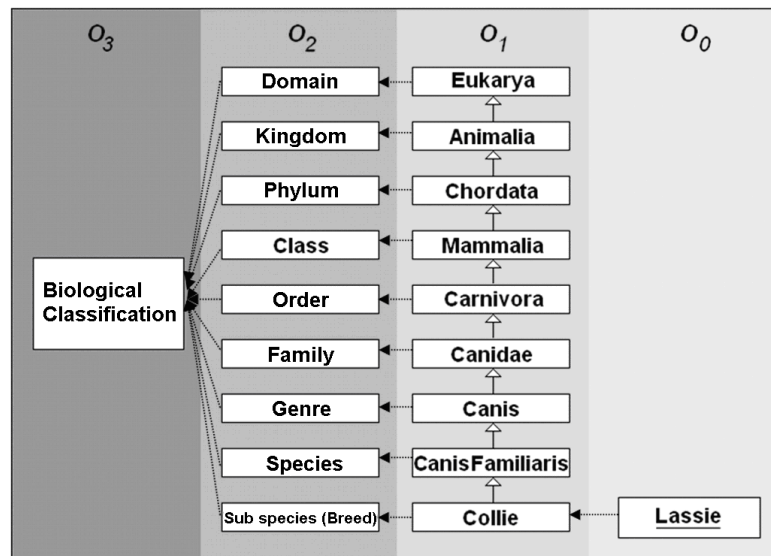

Figure 3: Biological Classification (Adapted from Atkinson and Kühne, 2003b).

From a linguistic or grammatical perspective, Lassie is a noun or an object, or rather noun and object are linguistic classifiers of Lassie. From a semantic or ontological point of view, the word Lassie can be understood as a type of dog or animal film character. We understand, however, that 'type of dog' and 'animal film character' are ontological classifiers for Lassie.

The first type of classification refers to the form and the second to the content of the element. These two dimensions of classification can be expressed graphically. In the visual models, the linguistic metamodels refer to the classification of the elements of the model with relation to its form (Object, Class, Association, Attribute). The ontological metamodels, on the other hand, refer to the classification of the content of the elements of the model (Collie, Race etc.)

According to Atkinson and Kuhne (2003b), despite the validity and utility of the ontological metamodels of types, for the tool builders and members of the standardizing consortia, such as the OMG, the metamodel term refers typically just to the metamodel of the linguistic type. Meanwhile, from the perspective of the user of the language, the hierarchy of types formed by ontological levels is much more relevant. In other words, the ontological metamodels are metamodels for the users focused on the content and the linguistic metamodels are a standard of metamodels focused on forms.

Researcher Strahringer (1996) studied how the level hierarchies of the models are built and coined the term 'metaization principle', to designate an operation that is repeatedly applied from a level to another, or rather, the primary mechanism of abstraction to structure the objects in levels of hierarchy. Kühne's analysis (2006) is similar to Strahringer's (1996), however making use of a different distribution of the elements for the levels and a different terminology. The MetaFrame methodology, presented in this article, utilized the metaization principle to verify and inform users how the metamodel components of IT best practices frameworks were built.

The metaization principle most used in information systems is the linguistic metamodeling. For example, the syntax of languages of modeling is at the M_2 level, as the well known methodology E/R (Entity/Relationship) by Chen (1976), used to represent part of the objects in the real world (M_0) at the level of an E/R (M_1) model, where only the components of the language (types, entity, relationship types, attributes etc.) can be used. Starting with this principle, an M_2 level structures the representation of the objects at the M_0 level in the M_1 level. In the ontological metamodeling, metatypes at the M_x level are defined, which describe the existing concepts at the M_{x-1} level.

2.2 Metamodel Principles and Instructions

The traditional focus of quality evaluation is on the final product; however, the defects of the final product often have roots in the initial planning and conception phases. This suggests that greater efficiency and effectiveness would be reached if efforts were made to evaluate the quality of the conceptual models. For Moody (2005), the current state of the evaluation practice of the conceptual models quality possesses more art characteristics =than engineering characteristics. For the conceptual modeling to progress from art to engineering, quality standards need to be defined, agreed on and used in practice.

Schütte (1998) is one of the authors who contribute to the work of this research through the modeling instructions contained in GoM (Guidelines of Modelling). The GoM is a framework for the development and evaluation of conceptual models composed of six general principals, described as follows:

1.   Construction adequacy Principle: a consensus must exist among specialists and users on what type of construction of a model is adequate for the problem and its proposal.

2.   Language adequacy Principle: the language used to create the metamodel fulfills its proposal. This principle refers to the completeness of and the consistency between the model and the metamodel. This means that the model should not possess any symbol or item that has not been specified in the metamodel.

3.   Economic efficiency Principle: this principle formulates economic restrictions on the task of modeling. The costs of development of a model should not surpass the gains of its use.

4.   Clarity Principle: this principle deals with the comprehensibility and expressivity of the model. Within the objectives of clarity, are the hierarchical decompositions, the formatting (arrangement of the elements) of the model and the filtering of information. Criteria and objectives of the quality of the graphic formatting of a model were defined by Tamassia (1988).

5.   Systematic conception Principle: this principle deals with the consistency of the construction between the models and it is also important for the integration of the models.

6.   Comparability Principle: this principle deals with the semantic comparison between two models according to their correspondence or similarity. This is one of the most important principles in a metamodelling environment. Metamodels are frequently used to compare and integrate models.

Goeken (2009) proposes the use of the principles defined by Schütte (1998) to also evaluate the metamodels. The author adds three new specific instructions to evaluate the quality of the metamodels:

Instruction 1: a metamodel reveals its metaization principle. It is important for the user of the metamodel to know which rules were used to construct the metamodel levels.

Instruction 2: a metamodel should posses a clear mapping between the universe of discourse and the words and symbols that name and describe them. Questions should not exist among users about the meaning of concepts in the metamodel.

Instruction 3: a metamodel must have rich semantic connections. The relationships between the metamodel components must be relevant and described in an expressive way.

The metamodels created from the MetaFramemethodology are verified as to the principles and instructions described.

2.3 Aplications of the Metamodels

The ontological metamodels can be applied to complete the analysis, adaptation, comparison and integration of the IT best practices frameworks. Once the components of the metamodels are extracted, the frameworks can be examined and analyzed to know the characteristics of their structure. This analysis contributes to the evaluation of the framework and in helping the implantation and adaptation within the organization.

Other possibilities of the application of the metamodels of the IT best practices frameworks are the comparison and integration with different frameworks. Using the same methodology for construction or, according to Strahinger (1996), the same metaization principle, the representation of the metamodels allows the comparison between the frameworks at a high or abstract level. This process of comparison is an important step in the integration of the frameworks. The integration of the metamodels can guide the integration of the frameworks at a concrete or low level.

However, despite the advantages of the comparison and integration with the use of the metamodels, when models with different concepts are compared, difficulties caused by the differences in language, such as synonyms and homonyms, arise. To resolve these problems, the solution can come from the research of data bases about the comparison and integration of schemes.

For Zaniolo (1982), "the mapping between different models is a formidable problem to be resolved". The author points out that the more promising approach consists of the use of a metamodel, at the conceptual scheme level.

According to Heuser (1998), "the description of a model is called, in the data base terminology, the scheme of the data base". One can consider the scheme as a textual representation of the model or metamodel and that the diagram or model of the E/R methodology is a graphic representation. The schemes, as textual representation of the model, have their own language, where the syntax is given for a particular grammar. A formal description of a scheme for the extensive E/R methodology, which is used in this research, is given by Engels et all (1992).

Various authors research the comparison and integration of models/metamodels; among these, the work of Batini et all (1986), Spaccapietra (1992), Teorey (1999), Conrad (2002), Rizopoulos (2005), Magnani et all (2005), Kurpjuweit (2007) e Karagiannis (2008) is notable.

According to Karagiannis (2008), the integration of models that were created from different metamodels can be approached through the mapping on the metamodel layer or meta^2 layer. The metamodel will act as a translator between the models that were instantiated by their metamodels. For this author, the use of metamodels is the most adequate way to integrate models; however the approaches, up to now, have not been capable of completing semantic integration and interoperability, which deal with the use of explicit semantic descriptions, very frequently provided in the ontological form.

Karagiannis (2008) cites the Gartner group, which informs that more than 40% of the expenses of the IT company are spent on problems of integration (be they syntactic, structural or semantic). Of these expenses, around 60 to 80% of the work force dedicated to resolving integration problems is spent on reconciling the semantic heterogeneity (be it in data bases, information systems etc.) The author reminds that, whatever the integration problem is, it has to be represented adequately. Diagramatic languages like UML and the E/R methodology, aligned with the metamodel concepts, are capable of expressing the syntactic, structural and semantic aspects in question.

Karagiannis (2008) describes the process of lifting or ontology anchoring as the essence of the semantic integration of models. For the author, lifting is what is called the ontological metamodeling, which is not limited by the meta^1 layer, but can be applied to the meta^2 layer and beyond. The author observes that when two models—whose forms were created by different metamodels—are integrated, their semantically related components will have to be found. According to the author, the combination of metamodels and ontologies bring an excellent way to resolve the task of integration and interoperability, achieving all the syntactic, structural and semantic heterogeneity.

2.4 Extensive E/R Methodology

The Entity Relationship E/R methodology, proposed by Chen (1976), was developed for the creation of conceptual and semantic models. The metamodels constructed with the MetaFrame methodology, presented in this study, follow the concepts and the notation of an extension of the E/R methodology, formalized by Engel et all (1992), with the objective of improving metamodel expressiveness. Figure 4 presents the main components and their notation, according to the authors cited above.
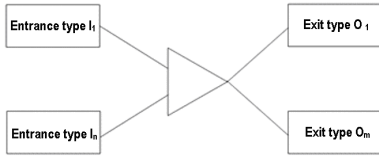
Object	Visual Representation
Entity type: classes of objects (concrete or abstract) with the same properties, which have their own meaning and which can be distinctly identified.	**Name**
Attribute: describe the properties of characteristics of an entity type. They receive a value of a particular type of data.	**Name: Type**
Relationship type: facts, the occurrences that link two or more entity type objects. The type, relationship, is a common grouping form of various entity types.	**Name**
Cardinals: integrity restrictions. Each entity type participant that is a relationship type can be restricted by the minimum and maximum cardinal numbers (min. max.) An entity type can participate in at least minimum, min., and maximum, max., relationships with another entity type. The symbol o "*" is used in place of the numbers to denote an indeterminate number of associations.	$(min._1, max._1)$ **Name** $(min._2, max._2)$
Constructor type: classification of the entity types. Starting with one or more entity types, entry, one or more types of exit can be obtained. They are used to represent a generalization, specialization, and partitioning of entity types.	Entrance type I_1 ... Exit type O_1 / Entrance type I_n ... Exit type O_m

Figure 4: Components and notation of the extended E/R methodology. Engels et all (1992).

Depending on the quantity and complexity of the objects (entity types, relationship types, attributes, constructor types), the use of a modeling strategy is important to help in the organization and development of the work of finding and defining the metamodel components. One modeling strategy for the extended E/R methodology is a sequence of steps that repeat themselves, producing small transformations of the initial model in the final model. The choice of the strategy for the construction of the model is influenced by the main source of information of the modeling process.

In the literature, there are four types of basic modeling strategies (Top-Down, Bottom-Up, Inside-Out or Middle-Out and Mixed) however, there is no consensus among the authors on which of these is the best technique. They use the work of Heuser (1998) and Atzeniet (1999) to describe these strategies. In the Top-Down strategy, an initial model is created in which the most abstract concepts ('from above') are represented first. Afterwards, intermediary models are created gradually through the refinement of the concepts into more specific concepts.

The Bottom-Up strategy (from below to above) is the inverse of the Top-Down (from above to below), consisting in starting from the most elementary and detailed concepts to construct more abstract and complex concepts. The Inside-Out (from inside

to out) or Middle-Out (from the middle out) strategy consists in considering the most important, or central concepts (inside), and gradually adding peripheral concepts related to them (out). The Mixed strategy is a combination of the other strategies.

None of the modeling strategies presented is universally accepted. The authors prescribe the use of a certain strategy or a combination of them starting with the specific information. Figure 5 shows some sources of information and recommendations on strategies use.

Information Source	Top-Down	Bottom-up	Inside-Out	Mixed
Personal information (interviews).	✔		✔	
Descriptions of data from authorized systems (reverse engineering).		✔		
Documentation, (reports, forms, etc.), of non-authorized systems.	✔	✔		✔
Unstructured data documents, books, etc.	✔		✔	
Semi-structured data in the form of manuals, guides, etc.	✔	✔	✔	✔
Data without descriptions	✔		✔	

Figura 5: Modelling strategies by source of information. Source: the authors.

The complexity of the model depends on the types of sources of information and quantity of types of entities to be represented. However, in more complex models, with more than 20 types of entities, various concomitant strategies are normally used. In these cases, a high level model is divided so that each partition can be modeled separately.

2.5 The best practices frameworks for IT eSCMP and eSCM-CL

The modelseSCM-SP 2.01 (ITSqc/CMU, 2006) andeSCM-CL 1.1 (ITSqc/CMU, 2006) were created by ITqsc (Information Technology Services Qualification Center) from the Carnegie Mellon University. The two models address 26 questions critical to the success of IT outsourcing, from the point of view of the service provider and the client organization. These critical questions are the result of literature reviews and interviews with IT service providers and clients.

The eSCM-SP and eSCM-C models are broad ranging sets of the best practices, developed, exclusively, for the management of operations of outsourcing of IT services. Their objectives are to offer guidance to IT clients and service providers to evaluate and improve the capacity of the outsourcing cycle organization. The eSCM-SP also offers a standard so that service providers differentiate themselves from their competitors.

The structure of the models is composed of three dimensions: Outsourcing life cycle, Areas of Capacity and Levels of Capacity, while the life cycle of sourcing is divided into four phases in the eSCM-SP model (Continuum, Initiation, Delivery and Closure) and five in the eSCM-CL model, with the addition of the Analysis phase before hiring services. Phases group the practices that occur over a certain outsourcing life cycle.

The eSCM-SP 2.01 possesses 84 best practices of outsourcing, grouped in ten areas of capacity (personnel management, knowledge, performance, relationships, threats, technology, contracting, transference of service, planning and service implementation and service delivery) and the eSCM-CL model has 95 best practices grouped in 17 capacity areas (outsourcing strategy management, governance, relationship, value, organizational changes, people, knowledge, technology, threats, outsourcing opportunities, outsourcing approach, planning, evaluation of service providers, outsourcing agreements, transference of service, source of service and conclusion of outsourcing).

The levels of ability show the path of evolution of the service provider and client capacity. They are divided into five levels in the two models: 1-Providing services/completing the outsourcing, 2-consistently attending the requirements/consistently managing the outsourcing, 3-managing organizational performance of outsourcing, 4-proactively setting values and 5-maintaining excellence.

The eSCM-SP and eSCM-CL are third generation best practice models, or rather, were projected to be articulated with other models. In this way, they complete models of best practices with COBIT. ITIL, and CMMI, among others, that don nor address in a comprehensive way all of the critical questions regarding IT outsourcing. For this reason, the eSCM-SP and the SCM-CL were selected as the research theme of this article, which intends to create a metamodel of these frameworks with the use of the MetaFrame methodology.

2. METHODOLOGY

For the development of this research the hypothesis that ontological metamodels, presented in 2.1 of the theoretical reference, facilitate the analysis of IT best practices frameworks was considered, based on the restructuring of a higher level abstraction, its components and a rich logical structure and semantics of its relationships.

In order to prove the hypothesis of this study, a collection of data, depuration, organization, analysis and presentation of data for the creation of the metamodels was made. As sources of information, official guides of IT best practices frameworks were used as a source, shown in 2.5 of the theoretical reference. The process of data collection of the official documents is similar to the technique of gathering data for systems analysis for the modeling of information systems. The Extended Entity/Relationship methodology, by Engels et all (1992), was used and the conceptual modeling strategies, presented in 2.4 of the theoretical reference, for the organization, analysis and representation of the data in the following types: entity, relationship, attribute and constructor. The final purpose of this data gathering was to elaborate the conceptual metamodeling framework.

All of the procedures described above are included in the methodology created in this research study, called MetaFrame, which describes a detailed process of creation and of verification of the quality of the metamodels of IT best practices. The objective of the MetaFrame methodology is to guarantee the quality of the metamodel and create useful products, such as dictionaries of the metamodel data, to be used in the

applications of metamodels, for example, in the comparison and integration of frameworks.

2.1 The MetaFrame Methodology

The aim of the methodology, called MetaFrame, presented in this article, is to create a metamodel framework of IT best practices based on the collecting and analyzing of data contained in the official guides of the IT best practices framework. The methodology has an iterative process of construction of the components of the metamodel, using modeling techniques and documentation of information systems, determining the verification of the results based on quality criteria.

The metamodel documentation, generated by the MetaFrame methodology, is important to analyze, adapt, compare and integrate IT frameworks, as they contain a dictionary of data with the definitions of the components represented.

Phase 1 of the methodology Metaframe encompasses the preparation of the study. In this phase, objectives are defined, professionals selected and their roles assigned, training and the distribution of support materials for the participants are performed. Phase 2 is the execution phase, where the collection of data and the iterative processing of the construction and documentation of the metamodel using modeling techniques are performed. Phase 3 verifies the quality of the metamodel according to the principles and instructions presented in 2.2 and also the correction and updating of the documentation generated by the methodology. A summary of the methodology is presented in Figure 6.
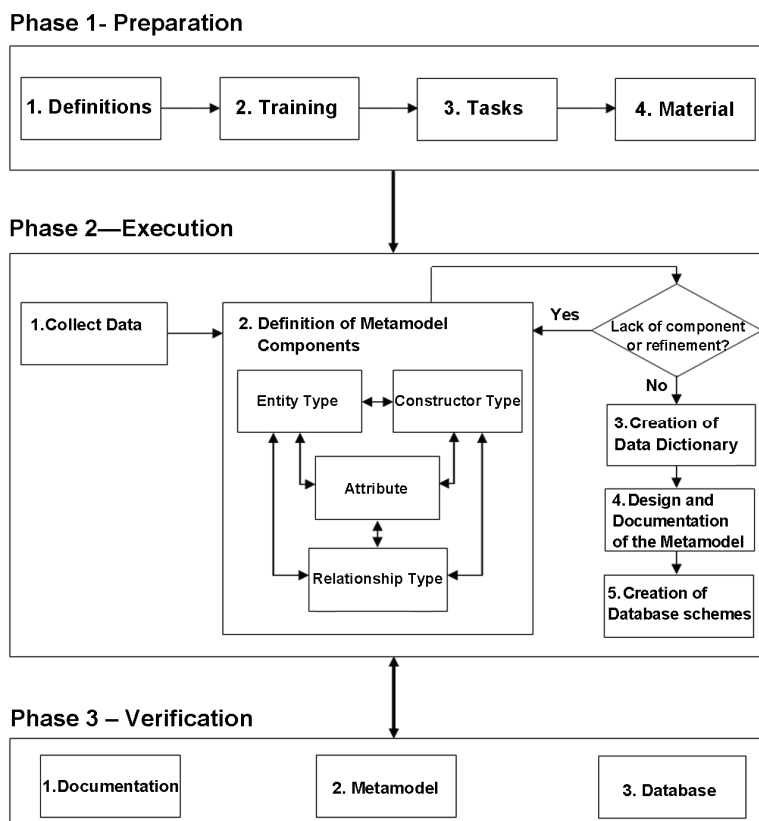

Figure 6: Metaframe Methodology for the creation of IT metamodel frameworks.

As an example, some steps and procedures of Phase 1 of the MetaFrame methodology are described in the following illustration:

Table 1. Some steps and procedures of Phase 1 of the Meta Frame methodology .

Phase 1—Preparation: it defines the objectives of the study; preparation and allocation of the resources; task planning; material distribution.
Step 1- Definitions: it defines the framework and the objectives of the study; selection of the professionals and definition of their roles; selection of the official guides.
Procedure 1- Framework: it defines the framework of IT best practices that will be used in the study and their characteristics (name, work area, version, version date, etc.).
Procedure 2—Objectives: it defines the objectives of the study to create a metamodel (aide the installation, analysis, customization, comparison, integration or fusion of the model, creation of application system, etc.).
Procedure 3—Participants: it selects the participants (systems analysts, business analysts, framework specialists, etc.) .
Procedure 4—Roles: it defines the participants' roles or functions. The suggested roles are: systems and business analysts, documenter, framework specialist, data administrator, etc.
Procedure 5—Official Guidelines: it select the official framework guidelines and register the bibliographic information (title, authors, year of publication, number of chapters, number of pages, etc.) on the bibliographic data form of the official guidelines.
Step 2—Training: Training of participating professionals in the necessary tasks of constructing a metamodel framework.
Procedure 1—E/RMethodology: Training of participating professionals in the E/R methodology extended from Engels et al (1992).
Procedure 2—Metamodeling: Training participating professionals in the modeling strategies, concepts of ontological metamodeling and in the metamodel quality criteria presented in sections 2.1 and 2.2.
Procedure 3—MetaFrame: Training participating professional in the MetaFrame methodology.
Procedure 4- IT framework: Training participating professionals in the IT framework, which will be the 'metamodel'. This training may be a course, lecture, or a written text, etc.

The next table displays some of the forms and models of the MetaFrame methodoloy, such as the formula for the collection of component candidates, in Figure 7, the forms of the official guides and metamodel components, in Figure 8, the model of verification of the metalevel of the components, in Figure 9, and the model of verification of the quality of the metamodel, in Figure 10.

Figure 7: Form for collecting the component candidates.

Metamodels of information technology best practices frameworks

Guide Name:	eSCM-SP v2.01: Model Overview
Guide number:	1
Authors:	Elaine B. Hyder, Keith M. Heston, Mark C. Paulk.
Editor:	Carnegie Mellon University, 2006
Volumes:	1
Chapters:	8
Pages:	104

Name of Relationship:	contents / support, documentation and implementation			
Number of Relationship:	1			
Description:	By standard, a practice of the SCM-SP and SCM-CL models are divided into three activities called main activities. The activities contain the necessary tasks to support, document and implement the practices of the model.			
Entity 1	**Relationship**		**Cardinal**	**Entity 2**
Practice	contains		3..3	Activity
Activity	supports, implements and documents		1..1	Practice
References:	Guidelines 2, chapter 2, pages 13-14			
	Guidelines 4, chapter 2, page 15			
Attribute	**Key**	**Type**	**Size**	**Description**
No attribute				

Figure 8: Form for the official guides and components of the metamodel.

Levels of the ontological metamodel:

Level O_2	Metamodel	Entity name: Represents a classification for the concepts of level O1
Level O_1	Model	Concept of the best practices model: represents an instance, example or element of the Entity defined in O2
Level O_0	Real World	Real world object: Represents the materialization or implementation of the concept defined in O1 of the best practices model.

1. Entity Type 1

Level O_2	Metamodel	Practice
Level O_1	Model	ppl01 – encourage innovation
Level O_0	Real World	Practice of the model implemented by an organizational process.

Figure 9: Model of verification of the metalevel components.

Verification of the Metamodel Quality
1. Adaptation and Construction Principle
2. Language Adaptation Principle
3. Economic Efficiency Principle
4. Clarity Principle
5. Systematic Conception Principle
6. Comparability Principle
7. A metamodel should reveal its metazation principle
8. A metamodel should have a clear map of the concepts and its meanings in the scope of the metamodel.
9. A metamodel should have semantically rich connections.

Figure 10: Model of the verification of quality of the metamodel.

With the finalization of the verification phase of the MetaFrame methodology, the results or products will be ready to be published within the organization or externally. The metamodel and the explanatory summary should be released together so that the users will have no questions regarding the components represented. Once the products of the methodology are ready, these can be used in the applications defined in the objectives of Phase 1, Step 1, Procedure 2.

3. RESULTS AND DISCUSSIONS

3.1 The Metamodel of the eSCM-SP v2.01 and eSCM-CL v1.1

Figure 11 represents the metamodel created, applying the MetaFrame methodology, of the eSCM-SP v2.01 and eSCM-CL v1.1. frameworks. Despite both frameworks having practically the same ontological metamodel, they were built separately. After the building process, and with the finding of the almost identical metamodels, they were integrated in one single ontological metamodel, with the representation of the maximum cardinality through two numbers. For example 84 and 95 represent the number of practices in the eSCM-SP and SCM-CL models respectively.
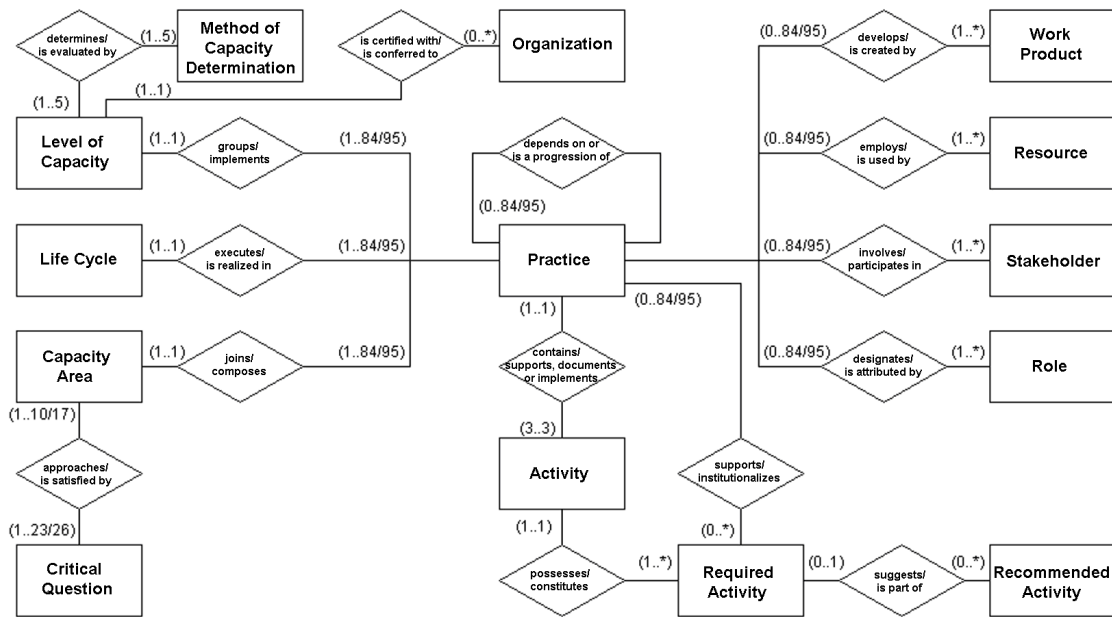
Figure 11: Metamodel of the eSCM-SP 2.01 and eSCM-CL v1.1, MetaFrame methodology. Source: the authors.

The metalevel of each one of the entity type of the metamodel was verified through the form of verification of the level of the metamodel. For example, the Practice entity type, which is in O_2, classifes the "pp101" concept at the O_1 level of the model. In the real world, in O_0, "pp101" represents the document of the policy of the organization for the incentive to innovate, required by the practice of the model.

It was verified that the metamodel serves all of the principles and quality instructions through the verification form of metamodel quality. For example, the Comparability Principle, related to the semantic comparison between models according to their correspondences or similarities, is addressed, as the metamodels created through the MetaFrame methodology follow the same construction mechanism and are comparable due to the fact that they present in their documentation a dictionary of data of the metamodel, which allows an effective comparison of the exposed concepts.

The explanatory summary, previewed by the MetaFrame methodology in procedure 2 of step 4 of the second phase, intends to interpret the metamodel in a clear way for the user. The definitions here presented were selected from the official guides of the frameworks.

In the eSCM-SP and eSCM-CL models, the central entity is a **Practice.** A **Practice** for these models corresponds to a set of actions that must be completed, by the provider/client of the IT services, so that the outsourcing relationship is successful. The eSCM-SP model has 84 practices and the eSCM-CL has 95 practices. A **Practice** can depend on or be the progression of another **Practice.** In the dependence relationship, a **Practice** depends on the realization of another to be initiated. In the relationship of progression, a **Practice** is an advanced or deeper development of another Practice, on an inferior capacity level. Each **Practice** contains, exactly, 3 main activities. The **Activity** entity represents the main activities of the models, which support, document and

implement each **Practice**. An **Activity** has one or more required activities. The **Required Activity** entity represents the necessary steps for the constitution of the **Activity**. Some practices (support practices) support the institutionalization of the activities required by other **Practices.** A **Required Activity** can suggest recommended activities. The **Recommended Activity** represents the procedures suggested for the organization to perform, though they are not mandatory for the certification in the models.

A **Practice** develops one or more work products. The **Work Product** entity represents whatever type of documentation, tools or software created by **Practice**, that is, it is a result of the **Practice**. A **Practice** employs one or more resources. The **Resource** entity represents all of the people, financial resources, implicit and explicit knowledge, infrastructure, systems, networks etc. A **Practice** involves the stakeholders that participate in the realization of the practice. The entity **Stakeholder** represents the staff, clients, final users, partners, suppliers, merchants and all of the people affected by the practices. A **Practice** designs one or more roles. The entity **Paper** represents the accountabilities, authorities or responsibilities attributed to a certain **Practice**.

The organizations are certified with just one level of capacity that can be from 1 to 5. The **Organizaton** entity represents the supplying organizations or IT service clients. A level of capacity is given to these organizations after the Complete Evaluation for Certification, which is the only method of determination of capacity for granting certification. The entity, **Capacity Determination Method** then determines the **Level of Capacity** of the organization.

Each **Practice** implements just one of the capacity levels. The **Capacity Level** entity groups none or at most, 84 and 95 practices of the eSCM-SP and eSCM-CL models, respectively. At capacity level 1, the organization cannot have any practices of the model implemented. However, to receive a capacity level of 2 in the eSCM-SP model, for example, the organization must have, at least, 48 practices implemented.

The practices of the models are completed in just one IT outsourcing life cycle. In the eSCM-SP model, this life cycle is composed of 4 phases and in the eSCM-CL model, which includes an analysis phase, there are 5 phases. The **Life Cycle** entity executes, at minimum, one, and at maximum, 84 and 95 practices of the eSCM-SP and eSCM-CL models, respectively. These values are theoretical, since, in practice, the life cycle executes more than one and less than the total practices of the models.

In both models, the practices compose groupings called capacity areas. There are ten capacity areas in the eSCM-SP model and 17 capacity areas in the eSCM-CL model. The **Capacity Area** entity joins theoretical values from, at minimum, one and, at maximum, 84 and 95 practices of the models eSCM-SP and eSCM-CL respectively. Each **Capacity Area** deals with, at least, one critical question of IT outsourcing. The **Critical Question** entity represents the critical questions of IT outsourcing that are served by one or more capacity areas of the models. There are 23 questions in the eSCM-SP model and three additional questions in the eSCM-CL.

4. CONCLUSION AND FUTURE RESEARCH

The objective of this article was to present the MetaFrame methodology for creating metamodels of IT best practices frameworks and to exemplify the application of this methodology with the metamodels of the s eSCM-SP and eSCM-CL framework. The approach of creating ontological metamodels can contribute significantly to the analysis of the frameworks through the entities and their relationships. Other possibilities of applications that emerge from the analysis of the metamodels and merely cited in this article, deal with the adaptation, comparison and integration of the frameworks.

The present research had certain limitations, such as the definition of a scope circumscribed by the identification and definition of ontological metamodels, without dealing with the creation of a domain ontology for the IT best practices frameworks. Another relevant limitation is the metamodeling of two similar frameworks, eSCM-SP and eSCM-CL. In the future, the application of the Metaframe methodology to other IT best practices frameworks is aimed at .

In future studies, the metamodels developed by the MetaFrame methodology will be used in diverse applications. The metamodels can be used to analyze the general structure of the framework as well as its scope, completeness and coherence in relation to its binding objectives. The metamodels can, also, offer methodological support for the adaptation or customization of the framework on the processes and structures of an organization. For example, the metamodel can suggest the adaptation or implementation of a new process or practice within the framework by exhibiting the entity types and relationship types associated.

Future research intends to study the comparison of frameworks of IT best practices through the metamodels, which can be very useful to analyze eventual complementary functionality. For example, one can observe, through the metamodel and the documentation generated by the Metaframe methodology that the ITIL does not offer metrics or other components of control in the same extension as COBIT. In this case, the dictionary of the metamodel generated by the MetaFrame methodology would be a prerequisite to the comparison of the structures of two or more frameworks and for dealing with the question of the synonymous and homonymous concepts.

It is intended to research the possibility of applying the IT best practices framework metamodels to the solution of the integration problem of frameworks. The term integration is here used when one wants to maintain the characteristics of each framework, but at the same time, wishes to create a common area among them. After the processes of analysis and comparisons of the metamodels are performed, connections between the components of the frameworks can be found through an integration metamodel. Entity types, such as Processes, Activities, Resources and Products are present in many of the frameworks of IT best practices, with similar meanings and attributes. Other components, despite having different names, have the same meaning and can also be integrated.

REFERENCES

Atkinson, C., and Kühne, T. (2003a), **Model-Driven Development: A Metamodeling Foundation**, IEEE Software, vol. 20,no. 5, pp. 36-41.

Atkinson, C., and Kühne, T. (2003b), **Calling a Spade a Spadein the MDA Infrastructure**, International Workshop Metamodeling for MDA, York.

Atzeni, P., Ceri, S., Paraboschi, S., TORLONE,R. (1999), **Database Systems Concepts**, Languages and Architectures. McGraw-Hill.

Batini,C., Lenzerini, M., Navathe, S.B. (1986), **A Comparative analysis of methodologies for database schema integration**. In ACM Computing Surveys 18, 4, pp. 323-364.

Chen, Peter, P.S. (1976), **The Entity-Relationship Model: Towards a Unified View of Data**, ACM Transaction on Database Systems, vol. 1, n°1, pp. 9-36.

Conrad S. (2002), **Schemaintegration - Integrationskonflikte, Lösungsansätze, aktuelle Herausforderungen**. Informatik - Forschung und Entwicklung, Springer-Verlag, Vol. 17, No. 3, pp. 101-111.

Engels, G., Gogolla, M.,Hohenstein, U., Hulsmann, K. (1992), **Conceptual Modelling of Database Applications Using an Extended ER Model**. North Holland, Amsterdam, pp. 157-204.

GoekenM., Alter S. (2009), **Towards Conceptual Metamodeling of IT Governance Frameworks Approach - Use - Benefits**, hicss, 42nd Hawaii International Conference on System Sciences, pp.1-10.

Hyder,E. B., Heston, K.M., Paulk,M. C. (2006). **The eSourcing Capability Model for Service Providers** (eSCM-SP).V 2.01.Pittsburgh: ITSqc. Disponívelem<htttp://itsqc.cmu.edu/downloads>.Acesso em 10 de Jun de 2009.

Hefley, W. E., Loesche, E.A. (2006). **The eSourcing Capability Model for Client Organizations** (eSCM-CL).V 1.1.Pittsburgh: ITSqc. Disponívelem<htttp://itsqc.cmu.edu/downloads>.Acesso em 10 de Jun de 2009.

Heuser, C. A. (1998), **Projeto de Banco de Dados**, 6ª edição. ISBN: 979-85-7780-382-8. Editora Bookman.

ITGovernance Institute. **COBIT 4.1** (2005).Disponível em: http://www.isaca.org/Template.cfm?Section=COBIT6&Template=/TaggedPage/TaggedPageDisplay.cfm&TPLID=55&ContentID=7981. Acessoem: 14 Jun 2009.

ITGovernance Institute: COBIT Mapping: Overview of International IT Guidance (2006), 2nd Edition, ISBN 1-933284-31-5.

IT Governance Institute COBIT® MAPPING: MAPPING OF ITIL V3 WITH COBIT® 4.1 (2008), ISBN 1-933284-31-5.

Karagiannis, D. and Höfferer, P. (2008), **Metamodeling as an integration concept**.Software and Data Technologies.Publisher Springer Berlin Heidelberg (2008), pp.37-50.

Karagiannis, D., and Kühn, H. (2002), **Metamodeling Platforms**. In A. Min Tjoa, & G. Quirchmayer (Eds.), Lecture Notes in Computer Science: Vol. 2455. Proceedings of the Third International Conference EC-Web, Springer, pp. 451-464.

Kühne, T. (2006), **Matters of (Meta-) Modelling**, In Journal on Software and Systems Modeling, Volume 5, Number 4, pp. 369-385.

Kühne, T. (2005), **What is a model**? In: Seminar 04101 Language Engineering for Model-Driven Software Development, Dagstuhl Seminar Proceedings InternationalesBegegnungs.

Kurpjuweit, S., and Winter, R. (2007). **Viewpoint based Meta Model Engineering**. EMISA, pp 143-161, In Manfred Reichert, Stefan Strecker, Klaus Turowski (Eds.): EnterpriseModelling and Information Systems Architectures – Concepts and Applications, Proceedings of the 2nd International Workshop on EnterpriseModelling and Information Systems Architectures).

Johannsen, W., Goeken, M. (2007). Referenzmodelle für ITGovernance. dpunkt. verlag GmbH, Heidelberg.

Magnani, M., RizopoulosN., Mcbrien, P., Montesi, D. (2005). **Schema Integration based on Uncertain Semantic Mappings**, Conference or Workshop Paper, 24th International Conference on Conceptual Modeling (ER05), Klagenfurt, Austria Lecture Notes in Computer Science, Volume 3716, pp.31-46.

Moody, D.L. (2005). Theoretical **and practical issues in evaluating the quality of conceptual models**: current state and future directions, In Data & Knowledge Engineering 55, pp. 243-276.

OMG (2003).MDA Guide Version 1.0.1 Version 1.0.1, OMG document omg/03-06-01.

OMG (2004).**UML-Unified Modeling Language Infrastructure Specification**, Version 2.0, Version 2.0, OMG document ptc/03-09-15.

Rizopoulos, N., McBrien, P. (2005). **A general approach to the generation of conceptual model transformations**.In: Proc. CAiSE. LNCS, Springer-Verlag.

Schütte, R., Rotthowe, T. (1998). **The Guidelines of Modeling- an approach to enhance the quality in information models**. In Ling, Ram, Lee (Eds.) Conceptual Modeling – ER 98.Singapore, 16.-19.11.98, pp. 240-254.

Spaccapietra, S., Parent, C., Dupont, Y. (1992). **Model Independent Assertions for Integration of Heterogeneous Schemas**.VLDB Journal 1, 1, pp. 81.126.

Strahringer, S. (1996). **Metamodellierung als Instrument des Methodenvergleichs**, Shaker Verlag, Aachen.

Teorey, T.J. (1999). **Database Modeling and Design**, 3rd Edition, University of Michigan, Lecture Notes.

Zaniolo, C. (1982). **A Formal Approach to the Definition and the Design of Conceptual Schemata for Database Systems**. ACM Transactions on Database Systems, Vol. 7, No. 1, Pages 024-059.