INFORMATICA, 1995, Vol. 6, No. 4, 457–482

BRIDGING ENTERPRISE MODELLING TO EUROMETHOD

Remigijus GUSTAS

Information System Department
Kaunas University of Technology
Studentų St. 50, 3028 Kaunas, Lithuania

Abstract. Enterprise goals justify and explain the presence of information system constraints, activities and actors. One of the main problem in the area of enterprise modelling is to bridge the gap between the description of goal as early requirement statement and the precise specification of related to this goal limited area of activity of which a stakeholder is concerned. Various kinds of entities, relationships and rules in the specification of several information system views exist for some reason, because they express the needs and rationales of requirements at the pragmatic level. On the other hand, goals justify and explain the presence of requirement components. Description of goals in terms of static and dynamic constraints is of interest, because it allows interpretation of goal descriptions as a driving force in the process of information system modelling at the organisational level.

The main focus of EUROMETHOD is on the contractual level starting with the call for tender, proceeding through the signing of a contract, eventually entering into the actual production of a set of deliverables. EUROMETHOD is focusing on the dynamic contractual relationships between customer and supplier. The basis of a contract consist of the specification of a problematic situation, and some general intentions and constraints for the description of a final state. Although the desired constraints and intentions of the information system could be formulated in the form of objectives, this part by EUROMETHOD is not guided. The aim of this paper is to introduce a unifying framework for modelling of enterprise goals in terms of semantic descriptions of information system views. Such a framework provide basis for better understanding of contractual customer–supplier relationships within several worlds of enterprise modelling.

Key words: enterprise modelling, information system adaptation, goal modelling, semantic descriptions, EUROMETHOD

1. Introduction. A commonly expressed view is that customer organisations are unable to define their business needs or requirements clearly. This could be described within the following situations (Euromethod, 1994a):

1) The requirements specified by the customers are too fuzzy or vague. This makes it impossible for supplier to identify what the solution to the customer's needs might be.

2) The customer presents a solution, which very often does not encapsulate the real requirements, and interested suppliers are asked to implement it.

The problems outlined above lead to a frustration of customer organisations in their goals of obtaining the information system required to meet business needs. Poor requirement elicitation inevitably result in the provision of the wrong information system to the customer.

Enterprise modelling (F3 Consortium, 1994) could facilitate transformation of fuzzy and ill defined requirements into a validated semantic specification of information system at the organisational level. According to Davis and Olson (1985), two levels of information system requirements are necessary. Organisational level requirements specify the enterprise structure, and boundaries within which individual decisions are to be made. Application level requirements determine information processing needs for specific application. Requirements at the enterprise level are dealing mostly with what the user needs, but not finding a set of software components that can be assembled to meet those needs. Besides its main concern to improve the acquisition and validation of requirements at the organisational level, enterprise modelling also pays attention to the explicit modelling of the requirements engineering process, and reuse of earlier requirements.

EUROMETHOD project (1994a) was established to facilitate mutual understanding between customers and suppliers within a market for development of information systems. It provides the means to better define requirements in the contract and to better plan and manage the project. EUROMETHOD supports the understanding, planning and management of the contractual relationships between a customer and supplier. The summary of this fundamental view to two levels customer–supplier relationships is depicted in Fig. 1.

A distinction is made between to levels of customer–supplier relations:

1) the relationships at a contractual level, and

2) the relationships at a project level.

The main focus of EUROMETHOD is on the contractual level starting with the call for tender, proceeding through the signing of a contract, eventually

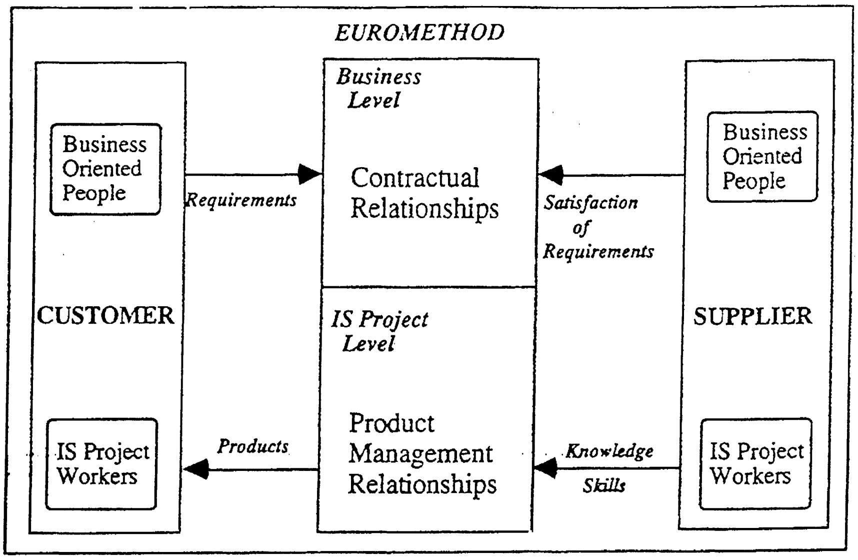

Fig. 1. Two levels of a customer–supplier relationships.

entering into the actual production of a set of deliverables, and finally ending when the contract is terminated (Euromethod, 1994a). The main focus of enterprise modelling is on transfer of knowledge between people with different backgrounds: customers and suppliers. Furthermore, F3 project (F3 Consortium, 1994) investigates situation when a part of contractual knowledge is not even initially present, as customers develop an understanding of their needs during the development process, and the knowledge must be obtained from several people, with different – even contradictory – views and interests. These aspects make interaction between results of enterprise modelling and EUROMETHOD a very important issue.

This document examines the relationships between enterprise modelling and EUROMETHOD. It is organised as follows: in Section 2 we discuss briefly an interplay between contractual customer and supplier relationships, and objectives modelling. The third Section presents an interaction of EUROMETHOD views and enterprise sub-models. We will in Section 4 describe contents of business level views, and address some important issues of bridging business modelling knowledge of F3 and EUROMETHOD projects. In Section 5 we discuss related semantic and pragmatic categories. The deeper understanding of relationship between objectives and states is presented in Section 6. Finally, we summarise the work in the concluding section.

2. Interplay of contractual relationships and objectives modelling. EUROMETHOD applies to any information system adaptation that can be characterised by an initial state and a final state. As a consequence, EUROMETHOD can be reapplied at various stages during the life of the same information system. The contractual arrangements during the life cycle could be as follows (Euromethod, 1994a):

1) Maintenance.
   Initial state: Problem description.
   Final state: New installed version of the information system.
2) Change study.
   Initial state: Problem description.
   Final state: Information system change study.
3) System description.
   Initial state: System description.
   Final state: Detailed business design.
4) System construction and installation.
   Initial state: Detailed business design.
   Final state: New installed version of the information system (subsystem).
5) Maintenance.
   Initial state: Global design.
   Final state: New installed version of the information system.

The life of an information system will typically contain several information system adaptations regulated by contracts. When any such adaptation contract is established, the knowledge on the information system are in a certain state, called the initial state of the adaptation. The desired state of the information system and related products which express the knowledge on the information system at the termination of the contract is called the final state of the adaptation. Euromethod provides the concepts and guidelines to determine both initial and final states of information system adaptations in a flexible way, in order to better suit for a specific situation.

In F3 project, the pragmatics of change is captured by the objectives model in terms of goals, problems, opportunities and pragmatic links among them. A problem describes a state which is not desirable. This state can be specified in terms of semantic links at the concept, activities and actors model (F3 Consortium, 1994). It is very important that the problem can not exist without stating the goal (Gustas *et al.*, 1995). If a customer has no predefined goal then the initial and final states does not make sense. A goal describes a final state. The interplay among objectives model, and initial and final states of an information system within the contractual customer–supplier relationships is presented in Fig. 2.

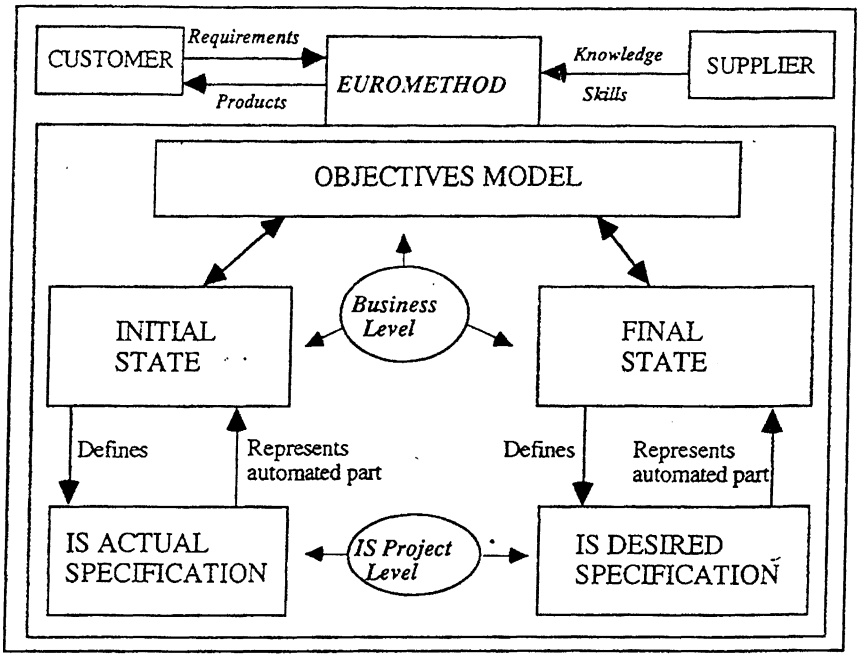

Fig. 2. Contractual customer–supplier relationships.

The objectives model (F3 Consortium, 1994) describes the reason, or motivation, for activities actors and concepts. Activities, actors, concepts and some semantic links among them, specify together either the initial, or final state of the customer–supplier relationships. Both the description of an actual state and the specification of objectives of the customer, let the supplier to derive the description of a final state. The semantic description of an actual state and objectives corresponds to the enterprise model in F3.

The concept of enterprise here should be interpreted in a very wide sense (Bubenko, 1993). It could actually mean description of qualitative aspects of business of the whole enterprise or some small part of it. In the context of information system development, the concept of enterprise rather denote a limited area of activity of the organisation which is concerned by the customer and supplier. The rest of this section describes the different "worlds" of enterprise model.

The enterprise submodels are based each on a number of pragmatic and semantic dependencies which are typical for that submodel. Each submodel concentrates on specific aspect of the enterprise. The basic components of enterprise model are depicted in Fig. 3.

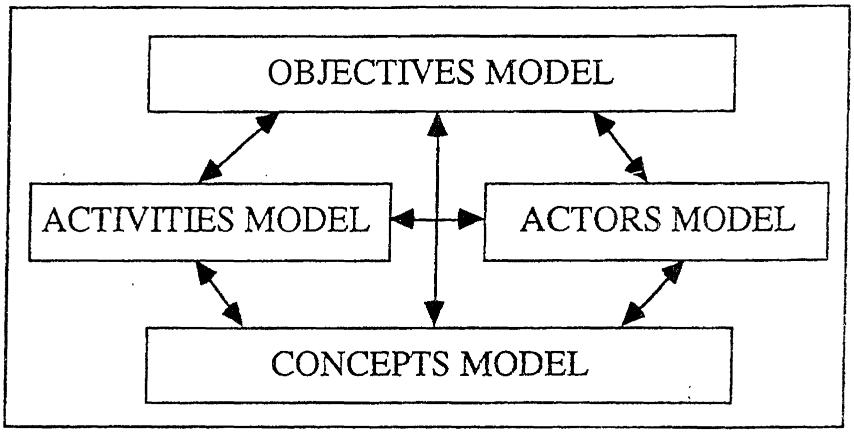

Fig. 3. Enterprise model "Worlds".

There are four general enterprise modelling components which can be used for the description of business: Objectives submodel, Activities submodel, Actors submodel, Concepts submodel. Notice that enterprise modelling doesn't mean specification of the information system at the same time. Models of business are one thing, while models of information systems are something different. The interaction among states of customer supplier relationships and enterprise submodels is presented in Fig. 4.

Each enterprise sub-model focuses on the specific business modelling aspect. The objectives are motivating and driving the development of other parts of enterprise model. Sometimes, enterprise components at a final state can be

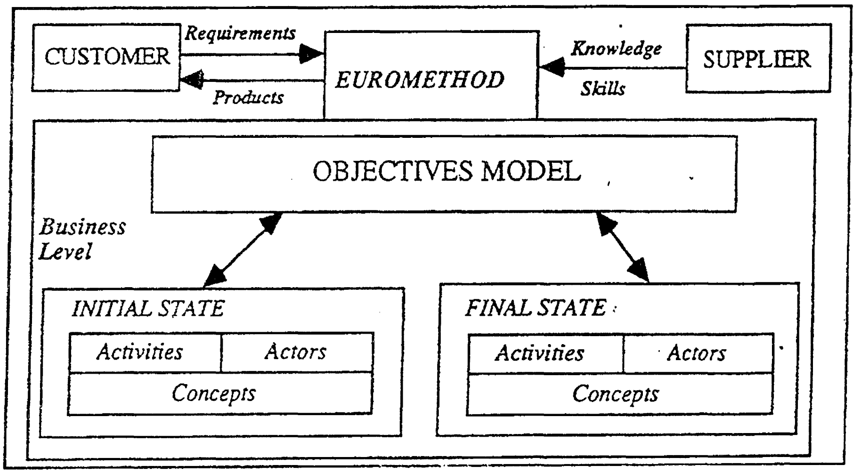

Fig. 4. Contractual customer–supplier relationships in terms of enterprise models.

easily stated before goals are well-understood. Elaboration of the objectives sub-model is then done in bottom-up way, by asking the reason for their existence (Bubenko, 1993). This part of enterprise model addresses the "why" perspective. The activities sub-model describes the "how" perspective and contains organisational activities, i.e., actions or tasks of an enterprise. Activities are able to make changes to states of affairs, which are expressible in terms of concepts. The actor sub-model addresses the "who" perspective. It defines the strategic dependencies between actors in achieving of their goals by means of message passing and triggering actions at the same time. The semantics of each message is expressible in terms of concepts. The concepts sub-model defines static constraints among concepts. It addresses the "what" we are talking about.

3. Enterprise and information system views. The type of knowledge about the information system that is contained in a specification usually is determined by the view it represents. EUROMETHOD project defines six such views. Each of them is characterised by a purpose and a set of information system properties (Euromethod, 1994b). The views define the concepts and terminology which are used to characterise the knowledge about the information system of a target domain. They capture all the knowledge relevant for the development of an information system. The six views are classified into information system views and computer system views. The information system views define the representation of the information resource (concepts in F3 project), the actors using and producing it, their processes (activities in F3 project) and the relationships among them. The computer system views define the representation of the structure and functionality of the computer system.

The information system (IS) views are the following (Euromethod, 1994b) (see Fig. 5):

- The business information view defines the information system properties which characterise the knowledge about the information resource.
- The business process view defines the information system properties which characterise the processes and their use of the information resource, independently from the actors of the information system.
- The work practice view defines the information system properties which characterise the knowledge about the actors, their location, their tasks and the subset of the information resource they use.

The information system and computer system views at the business level and information system project level are depicted in Fig. 5.

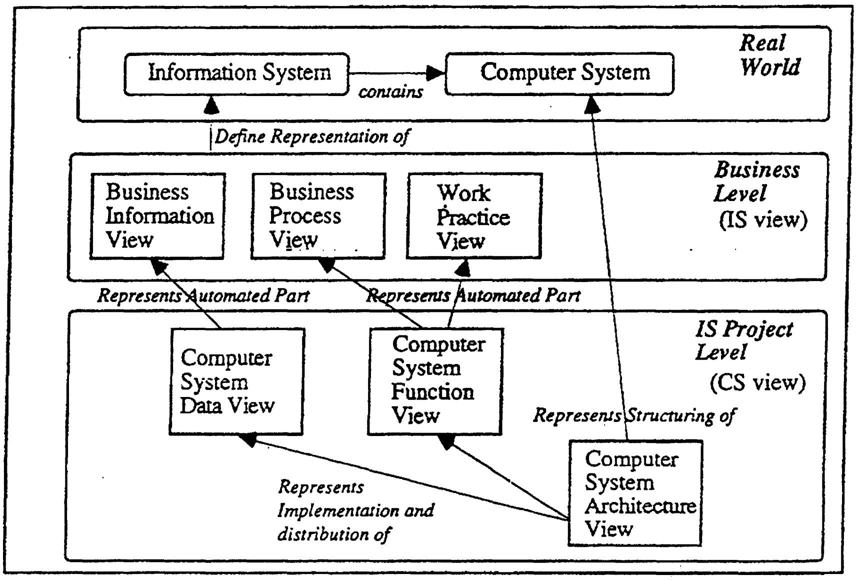

Fig. 5. EUROMETHOD. The information system and computer system views.

The knowledge about how the tasks of actors are automated and how the computer system is structured is described by the computer system view. The computer system (CS) views are the following (Euromethod, 1994b):

- The computer system data view defines the information system properties that characterise the knowledge about the data retained in the computer system and which thus represents the automated part of the information resource.
- The computer system function view defines the information system properties that characterise the knowledge about the functions performed by the computer system and which thus represents the automated part of the tasks performed by actors.
- The computer system architecture view defines the information system properties related to the structure of and processing units in the computer system.

Some of the actors in the target domain can be supported or replaced by a computer system. This means, some or all of their tasks are partially or fully performed by a computer system. This can be described in the work practice view which defines the information system properties characterising the knowledge about the selection of the activities and the concepts which constitute a subset of the information resource (see Fig. 6).

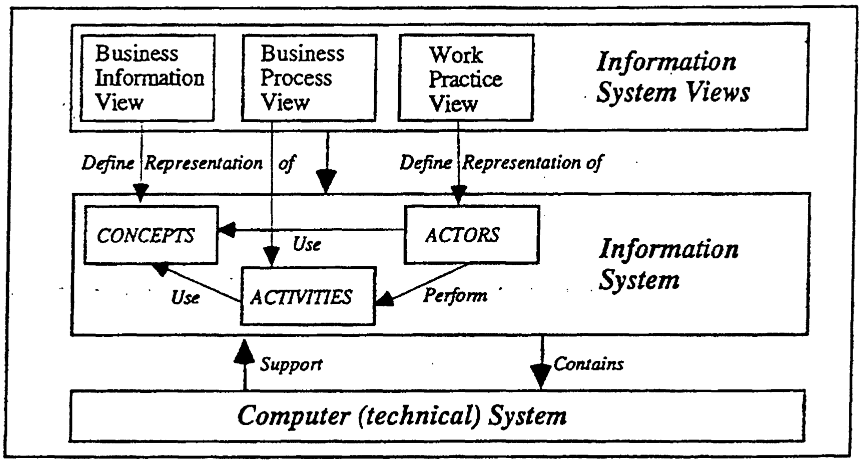

Fig. 6. Enterprise and information system views.

Information system adaptation addresses procurement and development of information system with a broad view on the products and processes. On the product side, information system is considered as organisational, human and technical enterprise elements. Organisational description primarily has to do with enterprise activities (see Fig. 6). Enterprise human resources consist of enterprise actors. Information system contain information resources described by enterprise concepts. These resources are used in the processes performed by various actors within the organisation. Several kinds of information technologies are developed to support the involved actors and to automate parts of the processes involved.

Considerably less attention in the area of requirements engineering has been given to the activities that precede the formulation of initial requirements. Earlier system development activities include those that consider how the intended system would meet new goals embedded in the larger organisational environment (Lundberg, 1995). This would enable of understanding of "how" and "why" (Sowa and Zachman, 1992) requirements came about. This earlier phase of the requirements analysis is just as important, if not more important than the second – refinement of initial requirements. Poor understanding of an enterprise is a primary cause of the system development project failure. A systematic framework is needed to help suppliers to understand what users want and to help users understand what technical systems can do. Because of the misunderstanding among customers and suppliers (Euromethod, 1994c), many systems that are technically sound have failed to address the real needs. As a consequence, the scope of EUROMETHOD is wider than software development: it is concerned with improving the enterprise system as such in all aspects.

Enterprise modelling aims at capturing not only activities performed during the information system development process, but also why these activities are performed and in which context. On the process side, information system adaptation addresses a variety of enterprise changes including their modification and computerisation to fulfil the changing needs of the organisation.

4. Bridging business modelling "Worlds" of F3 and information system views of EUROMETHOD. The information system view define the information resource, the actors using and producing it, their processes and relationships among them (Euromethod, 1994b). EUROMETHOD distinguishes the following information system views:

- The business information view. It defines the information system properties which characterise the knowledge about the information resource.
- The business process view. It defines the information system properties which characterise the processes and their use of the information resource, independently from the actors of the information system.
- The work practice view. It defines the information system properties which characterise the knowledge about the actors, their location, their tasks and the subset of the information resource they use.

The schema of interplay between information system views and enterprise sub-models in F3 is presented in Fig. 7.

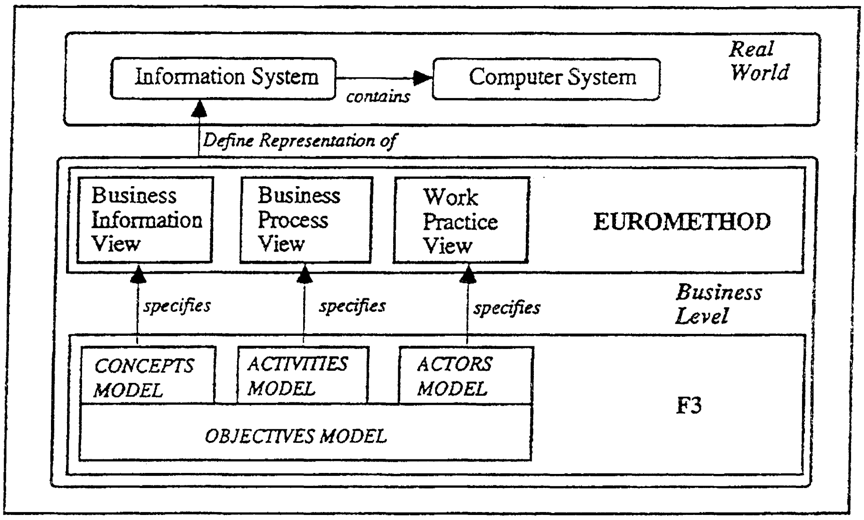

Fig. 7. Interplay among EUROMETHOD views and enterprise submodels (F3).

In F3 project (F3 Consortium, 1994), Gustas *et al.* (1995), it is proposed to structure a requirements specification of an information system in an number of "sub-models" (Fig. 3).

- *The concept sub-model.* The concept sub-model is used to define the "ontology" of the "universe of discourse" that concerns us, i.e., the set of object types, relationships, and object properties of the application we are talking about. In this sub-model we also further refine business rules of the objectives model into static as well as dynamic rules for the concept sub-model states as well as for permissible state changes. For developing a functional information systems specification, the concept sub-model will define about what "things and relationships" information must be represented in the information system, and what should be the rules, implemented in the processes of the information system, according to which the information system should behave. The concept sub-model will, moreover, serve as a dictionary of user and customer defined concepts, and conceptual structures to be used to strictly express other parts of a requirements specification.

- *The actors sub-model.* This sub-model is used to discuss and define the set of actors of the studied activity (individuals, groups, job–roles/positions, organisational units, machines, etc.), and their inter-relationships, such as part-of, reports-to, etc. This model includes links to the other sub-models, e.g., who is the "stake–holder" of a particular goal, who is responsible for managing an activity according to a particular goal, who is the author of a non-functional requirement, etc.
- *The activities sub-model.* In this part of a requirements specification, the particular organisational activity (in a wide sense), existing, to be modified, or to be developed, is defined and described from the point of view of activities, tasks, processes, and the information and material flows between them. Describing, for instance, human–computer interaction is part of this sub-model. Clearly, components of this sub-model are motivated by components of the *objectives sub-model*, they are performed using or referring to components of the *concept sub-model*, and they are typically carried out by components (resources) of the *actors sub-model*.
- *The objectives sub-model.* In this part of a requirements specification we describe, in a structured way, the "why" component of a requirements document. *Goals* and *business rules* for a particular enterprise activity (or set of activities), existing, to be modified, or to be designed, will be stated, and their relationships analysed. Other component types of this sub-model are *problems, causes* (of problems), *opportunities*, and *development actions* (Gustas *et al.*, 1995). While goals and rules are the "important" components of this sub-model, the other types of components help and support the detection as well as formulation of goals

and rules. They are hence considered as important text components in order to document and explain why certain rules and goals have been formulated. The objectives model is a graph with the above types of components as nodes, connected by relationships of type "motivates" or "influences". The motivates relationship is here seen as a refinement relationship (e.g., a goal is refined in a number of sub-goals). The influences relationship is a relationship indicating positive or negative influences between objectives model components.

In the following, we describe briefly the contents of Information system views in relation with F3 enterprise modelling components (Fig. 7).

The purpose of the business information view is to represent the information resource of the target domain, independently of how the information resource is implemented and used by actual working practices. It encompasses the information system properties that capture knowledge related to information items that are relevant to concepts in F3, i.e.:

- Concepts and static semantic dependencies among concepts. The static dependencies among concepts are represented by relationships (aggregation and generalisation relationships) and constraints (equality, inclusion, cardinalities, etc.).
- Concept evolution and the rules for every concept evolution. This includes the events that trigger these changes, the dynamic dependencies between changes and the events that are generated by these changes.

The purpose of the business process view is to represent the processes in the target domain, independently of how these are performed by actors in the actual working practices. It encompasses the information system properties characterising the knowledge about:

- activities in the target domain,
- relations among activities,
- their triggering events and conditions,
- their use of the information resource (relationships with concept model).

The information system properties defined in the business process view are as follows (Euromethod, 1994b):

- Business processes: identification of the process in the target domain and it relation with goals.
- Triggering business events.

- Triggering conditions.
- Generated business events.
- Business process decomposition: the decomposition of business processes into sub-processes. Sub-processes may also be decomposed.
- Dynamic dependencies: the dynamic dependencies between sub-processes of the business process.
- Business rules: the business rules governing the business process.
- Information use and generation: the concepts used and generated by the business process.

The business information view and the business process view together encompass the properties that define a view of an information system that is independent of any actors (human or computer) performing the processes and using the information resource. The purpose of the work practice view is to represent the actors, and the way in which they perform the processes in the target domain and make use of the information resource. The category of task is used in the work practice view. A task is a part of a process performed by the actor.

The work practice view encompasses those information system properties which characterise knowledge regarding Euromethod (1994b):

- the actors of the information system under consideration together with the organisational structure they work in the physical locations they work at,
- the task by which they achieve the business goals.

The information system properties defined in the work practice view are:

- Actors.
- Locations.
- Organisational structure: the organisational relationships between the actors. This includes the aggregation of actors into organisational units (organisational units are considered as actors too).
- Communications: The flow dependencies between the actors.
- Access needs and rights of actors: the information access needs and necessary rights for each actor, i.e., the external views of the actors at the concept model.
- Tasks. Identification of the tasks in the information system and their goals. Each task could have a description of:

1) The rules or procedures governing the task.

2) The way of performing the task: manual, computerised or interactive. An interactive task is partly performed by human actors and partly by the computer system. The details of the computerisation are defined with the task decomposition.

3) The decomposition of tasks into sub-tasks. Sub-tasks may be considered as tasks and described in the same way. Sub-tasks may also be decomposed. An interactive task can be decomposed into manual and computerised sub-tasks. A computerised task is usually not decomposed (it may be decomposed in the computer function view).

- Information use and generation by the task.
- The dynamic dependencies with other tasks and among the sub-tasks.
- Task triggering organisational events.
- Task generated organisational events.
- Task triggering conditions.
- The actor (performer) performing the task.
- The actor (manager) controlling the task.
- Resources: the concepts necessary for performing the task.
- Static dependencies of human–computer interface: the concepts that has to be passed through the human–computer interface (this includes the description of screen layouts, paper reports) and the organisational events which are generated by a manual task and trigger a computerised task or which are generated by a computerised task and trigger a task performed by a human actor.
- Behavioural dependencies of human–computer interface: triggering conditions for the human–computer interface. The behaviour of the human–computer interface also depends on the dynamic dependencies among computerised and manual tasks.

The ultimate objective of the presented analysis of information system views is to introduce principles for modelling of pragmatic dependencies on a basis of different classes of semantic descriptions. The interplay among goals and information system views, creates facilities for unambiguous understanding of contractual relationships. Therefore, semantic representations of an enterprise can be regarded as a basis for the investigation of conflicts and inconsistencies among goals of several stakeholders.

5. Related pragmatic and semantic categories of information system views. A goal hierarchy could be formed of interconnected goals of different levels of abstraction ranging from higher level business objectives to lower level operational goals (Wangler, 1993). The structural decomposition and refinement of such pragmatic categories may be understood in correspondence to the hierarchical decomposition of processes performed in structured analysis (Yourdon, 1989), and combination with object-oriented analysis and design methods (Rumbaugh *et al.*, 1991). The goals are broken dawn into lower level business functions each of which is intended to perform some specific task. The decomposition goes until a level is reached where the goal function is sufficiently well understood, i.e., it can be specified in terms of semantic constraints. The goal decomposition hierarchy is illustrated in Fig. 8.

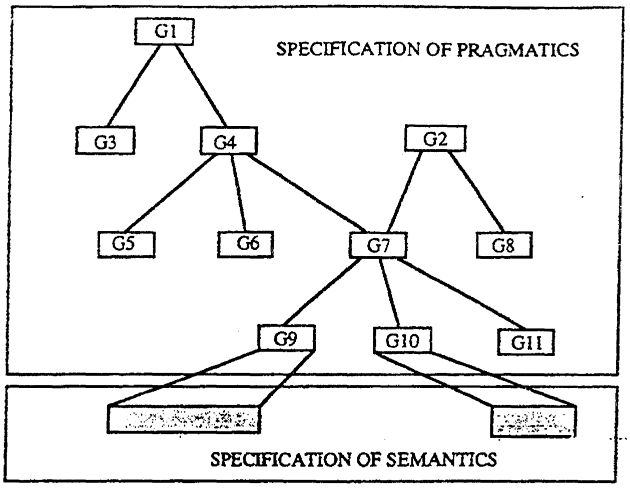

Fig. 8. The goal decomposition hierarchy.

Several goal decomposition hierarchies can be interpreted as the reasons for the specification of several enterprise or information system semantic level dependencies. At the lower level of a goal hierarchy, the goals are much easier to formalise than the more vague higher level business objectives. The reason is quite simple – the lower level goals consist of lesser number of dependencies and constraints.

Modelling of information system pragmatics typically involves such notions as goals, objectives, problems, constraints, etc. Goals are often described as desired situations that should be reached or striven for. The goals express wishes and desires concerning the properties of the product under development. A situation in this case can be described in terms of semantic constraints (static and dynamic dependencies) (Gustas, 1995). Problems are described as actual situations that are not desirable. They denote restrictions we have to suffer or try to avoid. Some of dependencies among semantic constraints and related pragmatic categories are presented in Fig. 9.

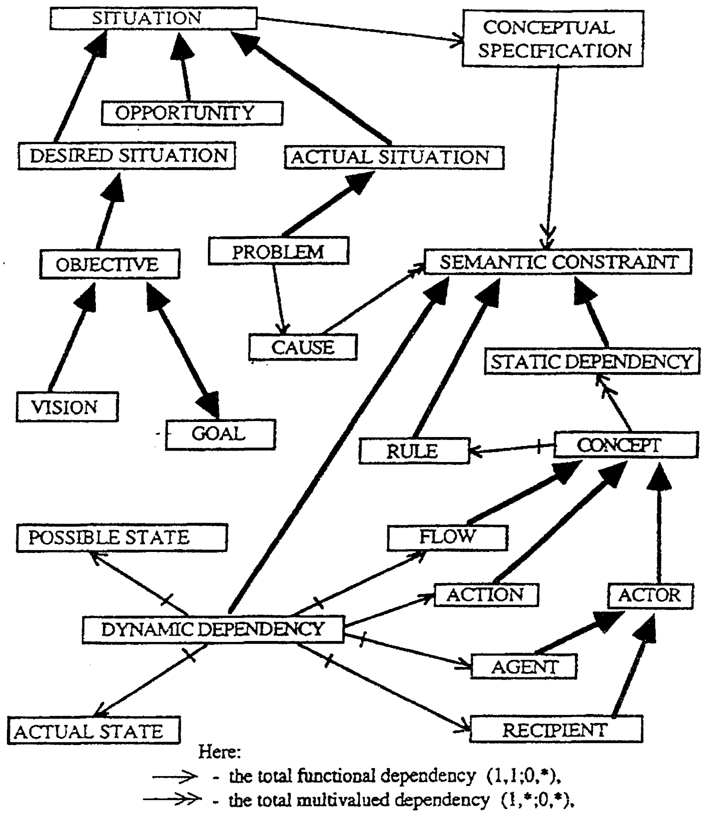

Fig. 9. Meta-model of related semantic and pragmatic categories.

In the following, we discuss briefly some of the notions (Gustas *et al.*, 1995) appearing in the goal modelling:

- Intentional notions such as goal, objective, vision, etc. express wishes and desires concerning the properties of the information system under development.
- Restrictive notions such as problems, perturbations describe not desirable properties of the information system. Problems denote restrictions we have to suffer or try to avoid.
- Development action take us from one situation to another, for instance from the problematic situation to the situation specified by goal statement.
- Indicative situations such as opportunities express statements that may be referred to improve an already existing situation.
- Notions of semantic constraints such as specification of causes of a problem, specification of goal, policy, strategy, etc. They are normally used to for denotation of rules of business. Business rules and constraints are usually specified for pragmatic categories the lower level in order to satisfice (Simon, 1984) the higher level goals.

The descriptions of goals, problems or opportunities are expressed as natural language statements, while specification of these pragmatic categories is performed in terms of static and dynamic representations of an information system. Conceptual models constitute the basis for the specification of pragmatic requirements. Modelling primitives applied for specification of semantics of an information system are usually based on Entity – Relationship like (ER) notations (Rumbaugh *et al.*, 1991). Although these notations are much more readable and understandable for human beings with respect of formalisms of traditional logic, there is no uniform theoretical basis or even agreement whether the concept should be specified as the entity, or as relationship, attribute, action, actor, etc. Usually, the same name of concept can be interpreted in terms of several roles, but just in different contexts of representation.

One of the main difficulties of ER representations are that they require to distinguish sharply among meta-categories, i.e., the designer is enforced to make decision whether the concept will be interpreted as an entity, or as a relationship, or as an action, etc. Strict interpretation of categories can cause different specifications of the same objective, even if the same modelling constructs are used. Because of different interpretations of the same concept, the designer is able to define the same objective in terms of several conceptual schemes.

So, ER kind of notation is representation dependent. Lack of relativeness of interpretation of basic categories of conceptual odels, injure the main principle of conceptualisation (Griethuisen, 1982).

Actually, every concept can be represented in the role of several semantic categories. The interpretation of a concept is relative and it is dependent on the set of specified semantic relationships. The relativeness of semantic categories is illustrated in Fig. 10.

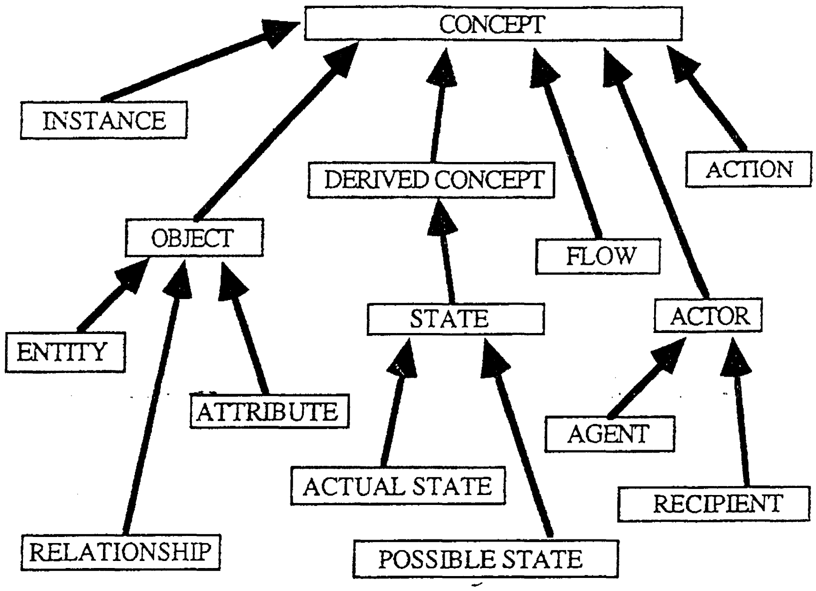

Fig. 10. Related semantic categories.

Dependencies among concepts (Gustas, 1995) constitute the basis for derivation of several semantic categories and constraints exploited in different Entity–Relationship like models. In principle, any kind of construct can be used for bridging goals and semantic descriptions of information systems. If any set of constraints, using the same modelling technique, can be represented differently, then it is dependent on the representation dimension. Different representations of the same universe of discourse give rise for discrepancies – structural differences of semantically equivalent representations. Lack of flexible interpretation of semantic categories can be a reason of complications in the process of integration of semantic representations caused by several goals. It should be noted, that the most primitive conceptual dependencies (Gustas, 1995) are independent of the representation dimension.

The pragmatic descriptions of goals, problems and opportunities are expressed by natural language statements, while the semantic descriptions of these pragmatic categories should be specified in terms of the most primitive static and dynamic dependencies. The notions of information system specification such as semantic dependencies (Gustas, 1995) (semantic relations (Storey, 1993)) are used for description of constraints and rules of business. Business rules and constraints are usually specified for pragmatic categories of the lower level in order to satisfice (Simon, 1984) the higher level goals.

6. Towards a deeper understanding of objectives modelling in terms of initial and final states. The ultimate goal of objectives modelling is to identify, to document and to analyse the business in terms of pragmatic level entities such as goals, problems, causes of problems, opportunities, and pragmatic relationships. The objectives model addresses the "why" perspective. It is motivating and driving the development of other enterprise sub-models. So, one of the main difficulties, as everywhere in the area of requirements modelling, is unambiguous description of pragmatic level entities on the basis of modelling artefacts of concepts, actors and activities sub-models. It can be done by defining pragmatic relationships in terms of information system views.

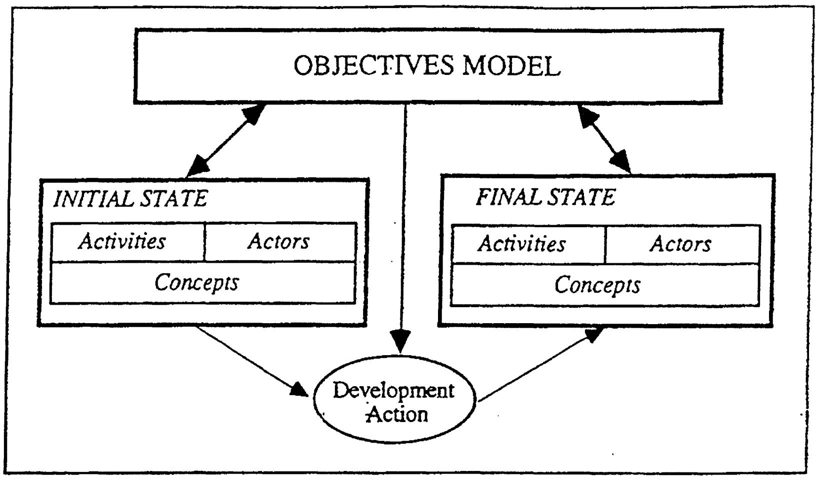

Fig. 11. Relation between objectives model and states of IS.

Objectives sub-model components specify the pragmatic aspects of an information system. Those aspects can be expressed on the basis of two specifications: the specification of actual state and the specification of final state. The initial state defines the actual static and dynamic descriptions of the information system at business level, and the final state does it for the desired description. The explanation of the relation between these two situations is depicted in Fig. 11.

The development action describes activities that have to be performed in order to achieve a predefined set of goals. The activity forms a set of development tasks that have to be carried out (Bubenko, 1993). Let us consider the special case when the specification has to be developed for one particular goal of the enterprise. In this case, an actual situation may be specified by the problem statement, while the semantics of a desired situation can be refereed by a goal statement. Thereby, the development action takes us from the problematic to the expected situation, specified by the goal. This case is illustrated in Fig. 12.

Fig. 12. Relation between problem and goal.

The opposite of a goal is a problem. *A problem describes a situation which is not desirable.* This situation can be specified by set of static and dynamic constraints. *A specification of a problem* is defined as a semantic difference between a set of semantic constraints of the actual situation and a set of constraints of the desired situation. So, the semantics of a problem can be regarded as part of the specification of product at the actual situation. It is very important that *the problem can not exist without stating the goal.* If the designer has no predefined goal then the problem does not make sense. A goal describes a *desirable situation. Specification of a goal* is unambiguously defined as part of the semantic description of a desired situation which is deduced as a semantic difference between a set of constraints of the desired situation and the set of constraints of the actual situation. So, according to this understanding, the specification of goal is the opposite side of specification of problem. At the same time, the specification satisfice the goal or the problem and is regarded as the *extension* of this pragmatic element.

Any two goals can be *contradictory*, i.e., one goal can influence negatively or hinder to the achievement of the other goal. It means that interpretations of the notion of a goal and a problem are relative and dependent on actor objectives. The same description could be interpreted as a goal for one actor, and as a problem for another actor. To specify contradictions between several goals of agents, the negative influence dependency (F3 Consortium, 1994) is introduced. An influence from A to B ($A \cdots\cdots > B$) dependency means that situation denoted by B is affected by situation A. The negative influence pragmatic relationship ($\overset{\cdots\cdots}{-}>$) from one goal to another indicates that the first hinders to the achievement of the second. The positive influence dependency ($\overset{\cdots\cdots}{+}>$) would mean, that the achievement of some goal, automatically contributes to the achievement of the other.

The *pragmatic dependency of specialisation* of A by B is denoted $A \longrightarrow\!\!< B$ if and only if the specification of B is a subset of the specification of the pragmatic element A. The "motivation" dependency (F3 Consortium, 1994) is usually used instead of specialisation to refine a goal. The "cause" dependency can be used in the case of specialisation of a problem. So, the consistency of decomposition of complex pragmatic entities to simple ones can be verified as far as their structures are specified by semantic constraints. Graphical notation of types of the pragmatic relationships are presented in Fig. 13. Here is the inheritance dependency.

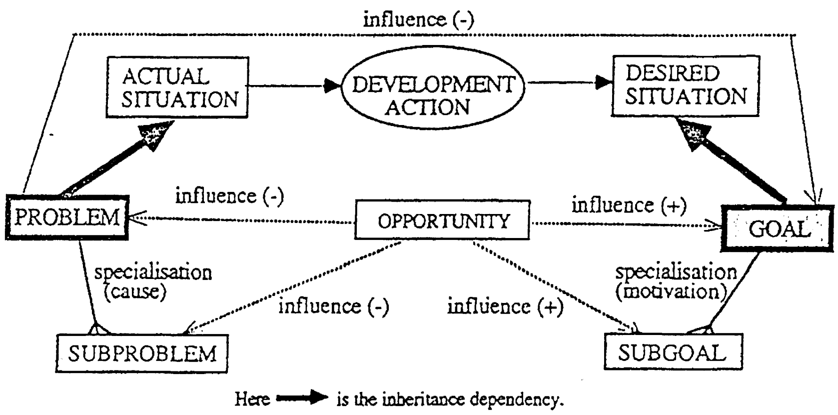

Fig. 13. Types of pragmatic dependencies.

Using the above described primitives, we can now explain notions of an opportunity and a cause of a problem introduced in the area of enterprise modelling (F3 Consortium, 1994). *Opportunities* express situations that may be used to improve actual situations. It should be noted that the notion of an opportunity is relative and dependent on the methodology. Sometimes, opportunities are used for description achievable situations not regarded as final, i.e., situations that we may want to take advantage of. If so, they could be interpreted as intermediate goals. Opportunities can be specified using representations of the semantic level.

A cause of a problem can have different interpretations as well. One of the most simple definitions of a cause could be based on the notion of a situation which is the reason for a problem to exist. Causes are usually used for specification the explanations of problems and regarded as extensions of problems. Every specification of a sub-problem is a cause for a problem, and thereby, a sub-problem causes a problem. The cause of the problem could be formed by the union of the causes of sub-problems.

It should be noted, that this earlier phase of the objective modelling is just as important, if not more important than the second – refinement of initial requirements. Poor understanding of an application domain (enterprise) is a primary cause of the system development project failure. The proposed systematic framework is needed to help suppliers to understand what users want and to help users understand what technical systems can do. Because of the misunderstanding among customers and suppliers (Euromethod, 1994b), many systems that are technically sound have failed to address the real needs.

7. Concluding remarks. Nowadays, a great variety of commercially marketed information system development methods and CASE tools exist. However, these products address mainly the middle and late stages of the systems development life–cycle. Practically none of them address the early, business objectives analysis, and the problem moving from the vague to the formal specification of requirements. Existing methods are not adequate for explicit capturing, and representing business objectives and problems, to be subsequently used to drive the later information system development phases (Bubenko, 1995). Links between business models and information system specifications are not maintained. This makes it impossible to reason about changes required in several information system "worlds", as a consequence of changes of the enterprise objectives.

One of the possible sources of complexity in the case of goal modelling is the inflexibility of the semantic level representations. Traditionally enterprise modelling techniques are based on the entity notations provided by several links (F3 Consortium, 1994). Links are established to capture semantic detail about the dependencies among resources, activities, actors, etc., and other enterprise modelling elements. The problem here is that all these notations are not provided by capabilities of integratability of representations defined in the contexts of different goals.

The main focus of Enterprise modelling is on transfer of knowledge between people with different backgrounds: customers and suppliers. Furthermore, the area of goal modelling is dealing with the situation when a part of contractual knowledge is not even initially present, as customers develop an understanding of their needs during the modelling process, and the knowledge must be obtained from several people, with different – even contradictory – views and interests (F3 Consortium, 1994). These aspects make interaction between the area of enterprise modelling and EUROMETHOD project as a very important issue.

An interplay between Enterprise modelling and EUROMETHOD addresses procurement and development of information system with a broad view: both the goals and information system views are considered. Semantic descriptions are defined as the organisational (activities), human (actors) and information resource (concepts) sub-models. Enterprise resources are used in the activities performed by various actors within the organisation. Thereby, a goal driven information system adaptation technology could be extremely fruitful for organisation re-engineering to support the involved actors. As a consequence, the scope of the presented framework is broader than traditional enterprise modelling: it is concerned with improving the information system in all aspects according to goals of several stakeholders.

The main focus of EUROMETHOD is on the contractual level starting with the call for tender, proceeding through the signing of a contract, eventually entering into the actual production of a set of deliverables, and finally ending when the contract is terminated (Euromethod, 1994a). EUROMETHOD is focusing on the dynamic contractual relationships between customer and supplier. The basis of a contract consist of specification of a problematic situation and some general intentions, constraints for the description of a final state. Although the desired constraints and intentions of the information system could be easily specified in the form of objectives, this part by EUROMETHOD is not guided. In this paper, we have presented the framework for understanding of interaction between goal models and semantic descriptions of information system views.

Acknowledgements. The presented investigations have been accomplished by author during his work at Swedish Institute for Systems Development. This activity was partially funded by European Union Information Technology Programme (ESPRIT III) Basic Research Action 6353 entitled NATURE which is concerned with Novel Approaches to Theories Underlying Requirements Engineering. The participation of Swedish Institute for Systems Development in this project was sponsored by NUTEK, Swedish National Board for Industrial and Technical Developments.

REFERENCES

Brachman, R. (1979). On the Epistemological Status of Semantic Networks. *Associative Networks,* Academic Press, 3–50.

Bubenko, J. A. (1993). Extending the scope of information modelling. *Fourth International Workshop on the Deductive Approach to Information Systems and Databases,* Polytechnical University of Catalonia, 73–97.

Bubenko, J. A. (1995). Challenges in requirements engineering. *Proc. of the Second International Symposium on Requirements Engineering,* York, England.

Davis, G.B., and M. Olson (1985). *Management Information Systems.* McGraw Hill, New York.

*Eurometho*d. *Overview.* (1994a).

Euromethod. Concept Manual, Deliverable Model. (1994b).

Euromethod. Method Bridging Guide. (1994c).

F3 Consortium. "F3 Reference Manual (Esprit III Project 6612)". (1994). SISU, Kista, Sweden.

Van Griethuisen, J.J. (ed.) (1982). *Concepts and Terminology for the Conceptual Schema and Information Base,* Report ISO TC97/SC5/WG5, Vol. 695.

Gustas, R., J. Bubenko jr. and B. Wangler (1995). Goal driven enterprise modelling: bridging pragmatic and semantic descriptions of information systems. *Proc. of the European–Japanese Seminar on Information Modelling and Knowledge Bases,* Sapporo, Japan .

Gustas, R. (1995). A basis for integration within enterprise modelling. *Second Int. Conference on Concurrent Engineering: Reasearch and Applications,* Washington, DC Area.

Lundberg, B.G. (ed.) (1995). *Proc. of the Second Scandinavian Research Seminar on Information and Decision Networks.* Vaxjo University.

Rumbaugh, J., M. Blaha, W. Premerlani, F. Eddy and W. Lorensen (1991). *Object–Oriented Modelling and Design.* Prentice Hall Int.

Simon, H.A. (1984). *The Sciences of the Artificial.* MIT Press.

Sowa, J.F., and J. A. Zachman (1992). Extending and formalizing the framework for information systems architecture. *IBM Systems Journal,* **31**(3), 590–616.

Storey, V.C. (1993). Understanding semantic relationships. *VLDB Journal,* **2**, 455–487.

Yourdon, E. (1989). *Modern Structured Analysis.* Prentice-Hall, Englewood Cliffs, N.J.

Wangler, B. (1993). *Contributions to Functional Requirements Modelling.* PhD theses, Dept. of Computer and Systems Sciencies, Royal Institute of Technology, Stockholm, Sweden.

Received November 1995

R. Gustas holds a diploma in Systems engineering and Informatics from the Kaunas University of Technology (KUT). He is a Doctor of Technical Sciences in Information Systems (KUT, 1986). Since 1991 R. Gustas has been working in the capacity of docent at the Information Systems Department, KTU, Lithuania. His research interests lie in conceptual modelling, specification semantics and pragmatic aspects of information systems, knowledge representation, view integration, formal aspects of enterprise modelling, goal driven information systems development, logical basis of computer aided information systems engineering.

ORGANIZACINIŲ MODELIŲ IR EUROMETHOD INTEGRAVIMAS

Remigijus GUSTAS

Daugelis mokslo darbų poreikių inžinerijos srityje akcentuoja tokių metodų sukūrimo būtinybę, kurie sudarytų galimybę aprašyti informacijos sistemas jau ankstyvose projektavimo fazėse. Dabartiniu metu egzistuoja visa eilė informacijos sistemų projektavimo metodų ir CASE programinių priemonių. Tačiau labai gaila, kad šie komerciniai produktai daugiausia skirti paskutinėms sistemos projektavimo ciklo fazėms. Praktiškai, nei vienas iš jų nepadeda formalizuoti ankstyvų informacinių poreikių, kurie formuluojami tikslų ir problemų pavidale, t.y. nepadeda išspręsti neraiškių ir formalių poreikių surišimo problemos. Straipsnyje yra parodoma, kaip minėta problema galėtų būti išspręsta, integruojant organizacijų aprašymo metodus su EUROMETHOD projekto pasiūlymais.