KEY-PROBLEM AND GOAL DRIVEN REQUIREMENTS ENGINEERING
Which Complementarities for Manufacturing Information Systems?

Virginie Goepp, François Kiefer
INSA de Strasbourg
24, Boulevard de la Victoire
67084 – Strasbourg Cedex, France

Keywords: Manufacturing information system, requirement analysis, problem formulation, dialectical analysis.

Abstract: The development of manufacturing information systems involves various stakeholders, who are not specialists for information systems. Therefore the stakes of the methods for such projects are to provide models which are understandable for all people involved, and conceptual enough to support the alignment between business, information system and manufacturing strategies of the company. The use of problem based models, stemmed from dialectical approaches, is efficient for the understand ability and a coarse strategic analysis, but it is limited through the project size. At the opposite, goal driven requirements engineering approaches enable to tackle large projects and detailed strategic analysis, but they are limited because of the difficulty to deal with the fuzzy concept of a goal. So, it would be interesting to gain from these two approaches. This paper first presents a problem driven approach for manufacturing information systems. It consists in a key-problem framework and a set of steps to exploit it. The assumption made is to base requirement elicitation on the problems encountered by the stakeholders. Then its matching with goal driven requirements engineering is shown and the complementarities between these two approaches are drawn and further discussed.

1 INTRODUCTION

Today, information systems (IS) occupy a prime position in our organisations. Indeed "Information that is timely, relevant and easy to access is a cornerstone of modern organisations. All organisations, whether in the private or the public sector, have IS to help them manage their activities. These IS are key to the success – and often survival- of many organisations" (General Direction III of European Commission 1996).

Thus, the development of the IS in particular for manufacturing companies is a crucial endeavour. Indeed, manufacturing systems are generally supported by computers and its peripherals, which ensure its facilities integration like, for example, in the Computer Integrated Manufacturing environment (Nagalingam and Lin 1999). This specific kind of IS, called "global information infrastructure" in (Chalmeta et al. 2001), should: carry out efficient information processing offering the correct information at appropriate time; allow for the co-operation between the enterprise's subsystems and its external elements; cover up the heterogeneity of physical resources and information applications and be able to respond to the changes in the enterprise's way of functioning and the evolution of support technologies (Mayer and Painter 1991).

In other words, to reach these objectives the main stakes of the development of such systems, in particular in the upstream phases, would be:

- To provide efficient means to support the construction of a comprehensive but although conceptual view of the system to be developed. This is essential to help developers understand what users want and to help users understand what technical systems can do in a context where technical systems increase in diversity and complexity (Yu 1997).
- To ensure the alignment between business, information system and manufacturing strategies. This alignment is acknowledged for the future cooperation between these sub-systems (Croteau and Bergeron 2001).

Goepp V. and Kiefer F. (2006).
KEY-PROBLEM AND GOAL DRIVEN REQUIREMENTS ENGINEERING - Which Complementarities for Manufacturing Information Systems?.
In *Proceedings of the Eighth International Conference on Enterprise Information Systems - ISAS*, pages 102-109
DOI: 10.5220/0002438901020109

However, such projects involve various stakeholders, from the workshop manager to the operators. Each has different background, skills, knowledge, perceptions and is generally not a specialist for the IS but only a user. Therefore the models and tools to tackle these stakes must be understood and shared easily by all of the actors involved. Experiences, stemmed from the enterprise modelling field, where the actors involved are also various, shows in (Chen et al. 1997) that discussions around the concept of problem are efficient to involve users during upstream phases of development and design.

So it is proposed, for the manufacturing information systems, to base the requirements engineering process on the concept of problem. The emphasis is the "whys" underlying the requirements (Yu and Mylopoulos 1994) but through a problem view and the related alternative IS architectures. According to (Sowa and Zachmann 1992) and adopting the systemic angle, the IS architecture is composed of the definition of the **components** of the IS, a description of their **interconnections** (both "logical" and "physical" within a network, for instance) and finally, their **interaction in time** (system dynamics) (Goepp and Kiefer 2005). The architecture concept is used as a shared and negotiated model of the target system. Its definition is built around the problem formulation.

However to be efficient the problem driven approach has to be guided. Therefore it is proposed to support the process with dialectical analysis tools. These tools deal with the formulation and solving of problems in the form of contradictions. The study of the contradictions enable to foresee changes and follow evolution in the future. The relevance of dialectical analysis in the IS field has been demonstrated in (Bjerknes 1992; Bratteteig and Ogrim 1994).

Our work is based on OTSM-TRIZ (Khomenko and Kucharavy 2002) because, unlike other dialectical approaches, OTSM-TRIZ provides effective ways to formulate and deal with the solving of contradictions. OTSM-TRIZ is used to build a key-problem framework, which is the basis for the problem driven requirements engineering. These set of generic contradictions are completed with a set of step supporting the requirements engineering process with as a result the definition of an architecture of the target IS.

Our practical experiences of this approach in (Goepp and Kiefer 2003b), for example, show its efficiency to support conceptual work and communication between the various stakeholders. The required communication and an in depth analysis of the domain are reached. However, this approach is only applicable to medium-scale projects where the project leader is capable of determining the state of a contradiction for the whole field of study. Moreover the recommended strategic analysis remains coarse.

Taking into account these drawbacks and looking for further improvements, it seems that goal driven requirements engineering should be useful. Indeed, goals have been recognized to be an essential component involved in the requirements engineering process (Potts 1997). These kind of approaches have proved, among other, to be an effective way to elicit requirements (Dubois et al. 1998) (Kaindl 2000) (Lamsweerde 2001) (Potts et al. 1994) (Rolland et al. 1998b) and to support a systematic exploration of design choices (Lamsweerde 2001) (Potts et al. 1994) (Hui et al. 2003) (Rolland et al. 1999a). Despite these contributions it is also acknowledged in (Anton and Potts 1998) (Haumer et al. 1998) and (Lamsweerde et al. 1995) that is not so easy to deal with goals. It is, for example, difficult for domain experts to deal with the fuzzy and abstract concept of a goal (Rolland et al. 1997).

Considering these different elements it would be relevant to gain from problem and goal based approaches for the requirements engineering process of manufacturing IS. This article proposes to lead a reflection on the complementarities between these two kind of approaches. It focuses on drawing potential research perspectives enabling to couple problem and goal driven approaches. Therefore the problem driven approach is presented progressively. Firstly, in the second section the key-problem framework and its role in the requirements engineering process is pinpointed. Then the stepping to exploit these framework is exposed in the third section. Based on this presentation, the fourth section makes a mapping between problem and goal driven requirements engineering and emphasizes the complementarities between these two approaches. These complementarities are analysed and further discussed in order to propose potential work directions. This analysis is split up into two opposite but complementary directions: from problem based approach to goal based one and vice versa.

2 KEY-PROBLEMS AS BASIS FOR PROBLEM DRIVEN REQUIREMENTS ENGINEERING

The proposed approach is based on generic models of problems in the form of a so called key-problem framework. It is built through a dialectical analysis using the OTSM-TRIZ problem formulation process in the form of contradictions. The framework consists in a set of three key-problems or evolution contradictions. It is combined with some basic problem solving tools of OTSM-TRIZ to guide the requirements elicitation, negotiation and specification around the definition of an architecture of the target manufacturing IS.

2.1 Formulation "Process" of "Evolution" Contradictions

The proposed dialectic analysis is based on OTSM-TRIZ (Khomenko and Kucharavy 2002), which is a meta-method facilitating the problem-solving process. As in most dialectic approaches, it is based on the contradiction concept but, unlike other approaches, OTSM-TRIZ suggests processes and tools for formulating and solving contradictions. Different contradiction classes are defined depending on the analysis point of view. Because we focus on the requirements engineering process, an overall analysis of the whole manufacturing IS domain is required. Therefore, the most general angle, i.e. the "evolution" contradiction, is chosen. It points to the overall contradiction of a particular family of systems – manufacturing IS in our case. To formulate evolution contradictions we should (cf. Figure 1):

- Describe the class of systems
- Identify functions to be fulfilled by this class of systems
- Identify performance parameters of each function. The evolution contradictions are contradictions between two performance parameters belonging to the same function. They are expressed through a characteristic element.

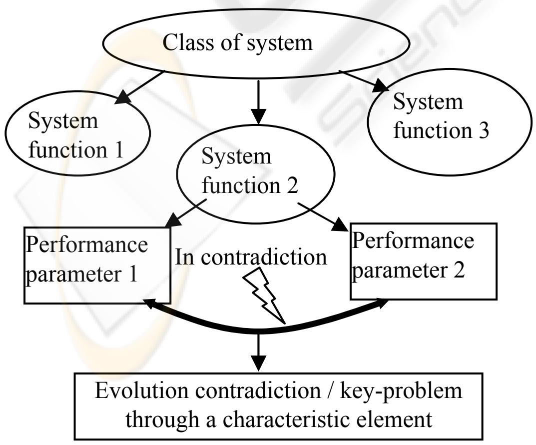

Figure 1: Formulation process of key-problems.

Table 1: Links between semiotic features, generic functions and characteristic elements of the key-problem framework (Goepp and Kiefer 2004a).

Semiotic features		Generic functions	Characteristic Element
Social effect	Per-forma	frame	Decision-making freedom
Pragmatics / semantics	In-forma	adapt, process	Specificity
Formal semantics syntactic	Forma	gather, store	Amount

2.2 Application to the Information System Family of Systems

This process has been applied for manufacturing IS and fully detailed in (Goepp and Kiefer 2003a). Even if OTSM-TRIZ details the way to formulate evolution contradictions, the knowledge required to perform these steps has to stem from the application domain. Therefore to be relevant they have to include an in-depth analysis of the basic requirements of the IS. So, building this framework is based on specific knowledge of the IS field, i.e. the role of IS within organisations to define performance parameters, and the semiotic analysis framework for IS engineering as proposed in (Stamper et al. 2000) to define generic functions. Each semiotic feature is associated to a generic function of manufacturing IS and finally to a characteristic element. Each characteristic element represents a class of contradiction (cf. Table 1).

At last, the outcome is a key-problem framework, representing the generic set of problems to be solved on a macro level, during the IS development. It contains three evolution contradictions which respectively relate to the amount of information, the degree of specificity of information and their decision-making freedom.

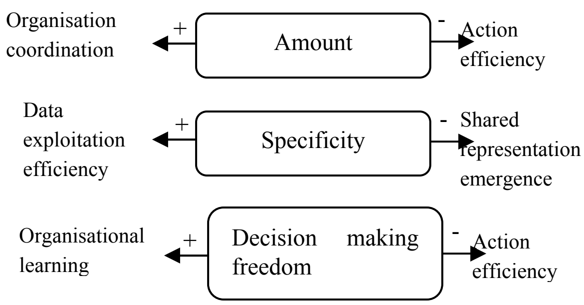

Figure 2: Overview of the key-problem framework.

The three contradictions are formulated as follow (cf. Figure 2):

The amount of information made available to each person must be increased to enhance coordination and the improvement of the organisation, but it should not be increased because too much information harms the efficiency of the action.

The degree of specificity of the information available to each person must be increased so that the data is exploited efficiently, but this must not be done because it harms the coordination and improvement of the organisation (making it more difficult for shared representations to emerge).

Freedom to make decisions must be reduced in actions in order to limit "non quality" and be efficient in the action, but this must not be done because people need a certain autonomy which is essential to implement organisational learning.

3 FROM THE KEY-PROBLEM FRAMEWORK TO AN EXPLOITATION STEPPING

3.1 Overview of the Steps

In view of how it is constructed, the key-problem framework provides a set of "fundamental" concepts to make an overall analysis of the what the system should be, while enabling the assessment of the design choices which kind of technical components, for example. It enables the evolution of specific manufacturing IS to be analysed from three different, but complementary, angles. In other words, the requirements engineering process shall be guided by acknowledged contradictions for a particular case. A contradiction is considered to be acknowledged when it is felt by the users of the IS under study. In view of the genericity of the contradictions formulated and in order to be efficient, the use of the framework has to be guided. This section presents a sequence of steps to exploit the key-problem framework in order to lead requirement specification by defining a target architecture of the system. The architecture of the system by describing the components and related objectives of the system, their time and spatial interactions provides of model easy to share by the various stakeholders.These set of steps are based on other tools and principles of OTSM-TRIZ, which are dedicated to contradiction solving.

The proposed set of steps and the principles are the following :

- Finding acknowledged contradictions or reformulation of the key-problems using partial solution and Ideal Final Result principle
- Determining the "extreme" architectures or solving/integration of the contradictions using the intensification principle
- Moving from "extreme architectures" to a targeted architecture or interpretation of the intensified architectures using the "multi-screen" view tool.

Figure 3 shows an overview of the proposed approach. The three steps and the results of each step are presented. The principles used at each step and the role of the key-framework within this process are specified in italic.

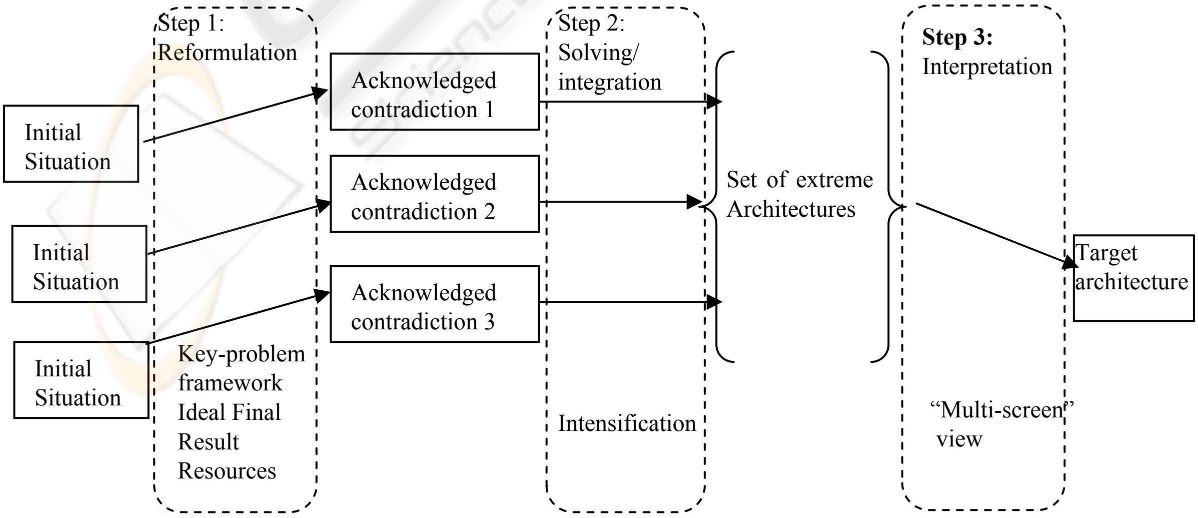

Figure 3 : Overview of problem driven requirements engineering process (Goepp and Kiefer 2004b).

3.2 Finding Acknowledged Contradictions

The first stage consists in determining the acknowledged contradictions for the IS under study. Identifying the "individual" and "collective" roles of the IS under study enables the three evolution contradictions to be reformulated so that they can be assessed by the IS users. Interviews are conducted to carry out an appraisal of the framework. The various contradictions stem from gaps between general knowledge in a given field (here the key-problem framework) and goals in a particular situation. In the boarder of OTSM-TRIZ, understanding opposition between general knowledge and specific conditions enable to "converge" efficiently on a solution. It is the "convergence" principle.

During this reformulation phase partial solutions, solving partially the formulated contradictions can be expressed. To support the partial solution finding it is proposed to use the notions of Ideal Final Result (I.F.R) and resources. The I.F.R. notion consists in defining all situations in terms of the ideal. This definition help formalise the solution-seeking direction while disregarding all restrictions to achieving it. The notion of resources during problem solving consists in fostering solutions which call on resources that are directly available in the environment. For technical systems, when the problem cannot be solved easily, new resources, which preferably cost nothing or very little, are introduced into the system. As a guide to refine this preliminary requirement elicitation and negotiate them we propose to use the notion of extreme architecture.

3.3 Determining "Extreme" Architectures

This phase is based on the intensification principle. It means we can imagine the harmful effect of the problem in an exaggerated form, even approaching the absurd. The essence of the problem is then outlined.

An extreme architecture is an architecture corresponding to a combination of the basic intensification of the acknowledged contradictions. For each acknowledged contradiction, two basic intensifications can be envisaged: one focusing on the collective aspect of the IS under study and the other on the individual aspect. Thus, for the contradiction relating to the amount of information, focusing on the individual aspect means reducing the amount of information stored. For the same contradiction, focusing on the collective aspect means increasing the amount of information stored.

When a single contradiction is acknowledged, intensification on the individual and collective levels means the definition of two extreme architectures. When there are two or three acknowledged contradictions, these basic intensifications must be combined and lead to four or eight extreme architectures. These phase enables to define a "reduced" set of alternative "extreme" architectures of the system to develop. This set will support the refinement and negotiation of the requirements during the third and last phase of the proposed approach.

3.4 Moving From Extreme Architectures to One "Target" Architecture

The architectures defined in the previous phase are not real architectures, but only asymptotical architectures to real situations. They intensifie the number and type of components which must be implemented.

Therefore, sometimes the absurd nature of certain extreme architectures is highlighted. In a particular study having set aside these architectures, the remaining architectures are made to converge towards a target architecture.

To do this, we use the "multi-screen" view to relocate the system under study both on a time scale (past, present, future) and on a systemic scale (sub-system, system, super-system). This graph offers (cf. Figure 4) a structuring support to carry out strategic alignment and alignment with the evolutions as defined in works on IS alignment.

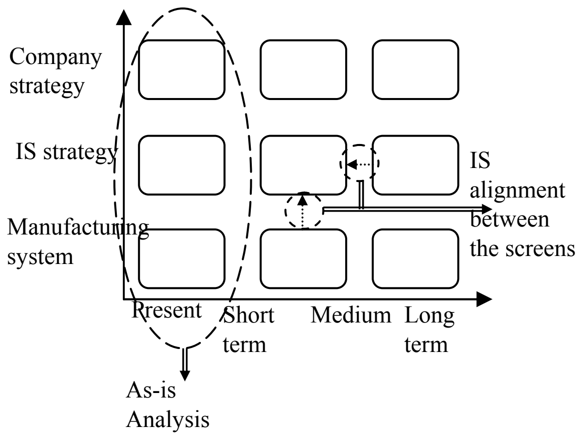

Figure 4: "Multi-screen" view tool.

For manufacturing IS, it is proposed to study the company strategy as super-system, the manufacturing IS strategy as system and the manufacturing strategy as sub-system for the following time scales: present, short/medium term and long term. The three "screens" concerning the present can be filled thanks to the as-is analysis. Then the work consists in analysing vertical links between the "screens" to ensure a coherent IS strategic alignment (between company, manufacturing IS and manufacturing system strategies), and horizontal links between the "screens" to ensure IS alignment with evolutions.

4 RESEARCH PERSPECTIVES TO COUPLE PROBLEM DRIVEN AND GOAL DRIVEN APPROACHES

The problem driven approach proposes an original view on requirements engineering for manufacturing IS. The use of problem based models, stemmed initially from the technical design field is relevant and efficient to assist the communication ability between various stakeholders. This concept is easy to share and understand by all of the people involved whereas it is difficult to deal with the fuzzy concept of a goal often used in requirements engineering. The scope of this section is to propose new research directions by showing the links and complementarities between the approach based on problem models and "classical" goal driven requirements engineering. First it is proposed to make a matching between problem and goal driven process in order to emphasize the differences and similarities between them. Then, in order to gain from advantages of both approaches their complementarities are studied into two opposite but complementary directions: from problem based approach to goal based one and vice versa.

4.1 Mapping from Problem Driven Approach to Goal Driven Approach

The main assumption of the problem driven approach, exposed in sections 2 and 3, consists in basing elicitation, negotiation, validation and specification of requirements on the formulation of the problems encountered by the users. The formulation of these problems for IS manufacturing is supported by the key-problem framework. This one combined with other tools and principles of OTSM-TRIZ enables to build progressively an consensual view of the system to be developed.

Indeed, during the reformulation phase, the Ideal Final Result and resources notions are useful to pinpoint quickly the solution seeking direction. During this phase, the expression of the reasons of the contradiction existence can be assimilated to the identification of goal elements. The partial solutions discovered during this phase are part of scenarios, which are usually combined with goal identification to improve goal modelling (Rolland et al. 1998b).

Similarly the "extreme" architectures are a limited set of scenarios. Indeed, according to (Rolland et al. 1998a), scenarios are classified according to their *intention*, *content* and their level of *abstraction*. The IS architecture appears in this classification at the intention level "exploratory", at the content level "objects of the real world" and at the abstraction level "type of objects" (Goepp and Kiefer 2004b). The combination of acknowledged contradictions and intensification principle is here interesting because only few but relevant scenarios are studied. These are relevant because built around the key-problems, which pinpoint the aspects to be treated carefully during the IS development.

During the last phase the "extreme" architectures are studied with the "multi-screen" view in order to converge to the target architecture. The target architecture is the base for the requirement specification of the system under study. This step is supported by the "multi-screen" view tool. The analysis of the potential links between the "screens" enables to refine the goals initially identified by taking the organizational environment into account. A coarse strategic alignment between the company, the information and manufacturing system strategies is provided. Requirement negotiation is supported by the visual feature of the "multi-screen" view.

The use of the concept of a problem as alternative to the goal one is interesting, however some drawbacks remain. Indeed, the approach is only applicable efficiently on small/medium scale projects, where the limited scope enables to determine the state of the contradictions "acknowledged" or not. The strategic analysis is not enough detailed and could be improved. In the following sub-sections potential complementarities are outlined and studied.

4.2 Mapping from Problem Driven Approach to Goal Driven Approach

In this sub-section we lead a reflection about how goal based approaches could improve the problem based approach. As shown previously the use of the concept of problem is interesting and relevant in the context of manufacturing IS. However the conceptual work with these models could be improved if combined with goal identification. This implies to analyse and formalize in detail the links between problem formulation and goal identification. Indeed, the study around the key-problem framework enables to discover goals however the completeness is not formally checked. Therefore, the problem dimension has to be linked with the goal taxonomies functional versus non functional goals or soft and hard…

Moreover the refinement strategy recommended in many goal driven approaches like in (Anton and Potts 1998) could complete the problem based approach. For example, the mechanisms proposed in (Rolland et al. 1998b) to discover goals at lower level of abstraction could be transposed to refine the problem breakdown. This breakdown would improve the detail level of the analysis, which remains coarse.

Last but not least goal modelling techniques like the goal/strategy map or map for short exposed in (Rolland et al. 1999b) could be used to describe more formally the architecture model by associating to each architecture the corresponding goal/strategy map in a standard formalism. The map formalism could also be used to reduce the gap between the acknowledged contradictions and the related alternative architectures.

In this sub-section we lead the reflection in the opposite direction in order to propose improvement possibilities for goal based approaches through concepts used in the problem based approach. The goal/problem coupling already evoked previously could be relevant to improve the practical use of the concept of a goal. Indeed, for example, the impacts between considered strategies and related key-problems could enable to make the goal concept more concrete for the various stakeholders. If the strategies and goals are illustrated through the related problems, these abstract concepts become more comprehensive.

Moreover, the intrinsic feature of a key-problem could be exploited to manage conflicts between goals. This management is a problem often discussed in the literature for example in (Darimont et al. 1998). Indeed, a key-problem is a contradiction and contains therefore intrinsically a conflict. By checking the consistency between the key-problems and the alternatives related to conflicting situations the goal conflict management could be improved. In this way solving directions for the conflicts could be highlighted systematically.

Concerning the scenario identification aspect, the intensification principle and the "multi-screen" view could be used advisedly to complete existing scenario identification approaches. Indeed, the intensification principle enables to emphasize quickly and efficiently the relevance of the identified scenarios. The "multi-screen" view gives a visual support for building a shared view of the scenario alternatives. This is essential to help managing various stakeholders with various concerns.

ACKNOWLEDGMENTS

We would like to thank the team of the CRI (Centre de Recherche en Informatique) – Panthéon La Sorbonne (France) particulary Colette Rolland and Camille Salinesi for their help by, among other, providing us bibliographic material.

REFERENCES

Anton, A. I., and Potts, C. "The use of goals to surface requirements for evolving systems." *International Conference on Software Engineering (ICSE'98)*, Kyoto, Japan, 157-166.

Bjerknes, G. (1992). "Dialectical Reflection in Information Systems Development." *Scandinavian Journal of Information Systems*, 4, 55-78.

Bratteteig, T., and Ogrim, L. "Soft dialectics - Structured Handling of Problem Situation in System Development." *Second European Conference on Information Systems*, Nijenrode University, Breukelen.

Camponovo, G. and Y. Pigneur (2004). "Information Systems alignment in uncertain environments". *IFIP International Conference on Decision Support System DSS'2004*, Prato, Tuscany.

Chalmeta, R., Campos, C., and Grangel, R. (2001). "Reference architectures for enterprise integration." *Journal of Systems and Software*, 57(3), 175-191.

Chen, D., Vallespir, B., and Doumeingts, G. (1997). "GRAI integrated methodology and its mapping onto generic enterprise reference architecture and methodology." *Computers in Industry*, 33(2-3), 387-394.

Croteau, A.-M., and Bergeron, F. (2001). "An information technology trilogy: business strategy, technological deployment and organizational performance." *The Journal of Strategic Information Systems*, 10(2), 77-99.

Darimont, R., Lamsweerde, A., and Letier, E. (1998). "Managing conflicts in goal-driven requirements engineering." *IEEE Transactions on Software Engineering*, 24(11), 908-926.

Dubois, E., Yu, E., and Petit, M. "From early to late formal requirements: a process control case-study." *9th International Workshop on Software Specification and Design - IWSSD'98*, 34-42.

General Direction III of European Commission. (1996). "Euromethod, User Book Version 1." Euromethod Project, France, Paris.

Goepp, V., and Kiefer, F. "Towards a definition of the key-problems in information system evolution: Formulating problems to better address information system projects." *3rd International Conference on Enterprise Information Systems - ICEIS'03*, Angers, France, 586-590.

Goepp, V., and Kiefer, F. (2003b). "Une démarche de conception de système d'information par l'identification des problèmes clés : application au suivi de pièces en atelier de production mécanique." *La Cible*(99).

Goepp, V., and Kiefer, F. (2004a). "Definition of a key-problem framework for information systems in CIM environments." *In review, submitted to International Journal of Computer Integrated Manufacturing*.

Goepp, V., and Kiefer, F. (2004b). "Design of information system architectures using a key problem framework." *Accepted for publication in Computers in Industry*.

Goepp, V., and Kiefer, F. "Information System Design and Integrated Enterprise Modelling through Key-Problems." *16th IFAC World Congress - Praha 2005*, Praha, Czech Republic.

Haumer, P., Pohl, K., and Weidenhaupt, K. (1998). "Requirements Elicitation and Validation with Real World Scenes." *IEEE Software*, 24(12), 1036-1058.

Hui, B., Liaskos, S., and Mylopoulos, J. "Requirements analysis for customizable software: a goals-skills-preferences framework." *IEEE Conference on Requirements Engineering*, Monterey Bay, USA, 117-126.

Kaindl, H. (2000). "A design process based on a model combining scenarios with goals and functions." *IEEE Transactions on Systems, Man, and Cybernetics*, 30(5), 537-551.

Khomenko, N., and Kucharavy, D. "OTSM-TRIZ Problem solving process: Solutions and their classification." *Etria World Conference - TRIZ Future 2002*, Strasbourg, France.

Lamsweerde, A. "Goal oriented requirements engineering : a guided tour." *International Joint Conference on Requirements Engineering RE'01*, Toronto, 249-263.

Lamsweerde, A., Dairmont, R., and Massonet, P. "Goal-directed elaboration of requirements for meeting schedulers : Problems and lessons learnt." *2nd IEEE International Symposium on Requirements Engineering (RE'95)*, York, England, 194-203.

Mayer, R. J., and Painter, M. K. "Roadmap for enterprise integration." *Autofact'91 Conference*, Chicago, USA, 7.1-7.26.

Nagalingam, S. V., and Lin, G. C. I. (1999). "Latest developments in CIM." *Robotics and Computer-Integrated Manufacturing*, 15(6), 423-430.

Potts, C. "Fitness for use : the system quality that matters most." *Third International Workshop on Requirements Engineering : Foundations of Software Quality REFSQ'97*, Bracelona, Spain, 15-27.

Potts, C., Takahashi, K., and Anton, A. I. (1994). "Inquiry-based requirements analysis." *IEEE Software*, 11(2), 21-32.

Rolland, C., Ben Achour, C., Cauvet, C., Ralyté, J., Sutcliffe, A., Maiden, N., Jarke, M., Haumer, P., Pohl, K., Dubois, E., and Heymans, P. (1998a). "A Proposal for a Scenario Classification Framework." *Requirements Engineering Journal*, 3(1), 23-47.

Rolland, C., Grosz, G., and Kla, R. "Experience with goal-scenario coupling in Requirements Engineering." *4th IEEE International Symposium on Requirements Engineering*, Limerik, Ireland, 74-81.

Rolland, C., Nurcan, S., and Grosz, G. "Guiding the participative design process." *Association for Information Systems Americas Conference*, Indianapolis, Indiana, 922-924.

Rolland, C., Prakash, N., and Benjamen, N. (1999b). "A multi-model view of process modelling." *Requirements Engineering Journal*, 4(3), 169-187.

Rolland, C., Souveyet, C., and Ben Achour, C. (1998b). "Guiding Goal Modelling using Scenarios." *IEEE Transactions on Software Engineering*, 24(12), 1055-1078.

Sowa, J., and Zachmann, J. (1992). "Extending and formalizing the framework for information systems architecture." *IBM Systems Journal*, 31(3), 590-616.

Stamper, R., Liu, K., Hafkamp, M., and Ades, Y. (2000). "Understanding the roles of signs and norms in organizations-a semiotic approach to information systems design." *Behaviour and Information Technology*, 19(1), 15-27.

Yu, E. "Towards Modelling and Reasoning Support for Early-Phase Requirements Engineering." *3rd IEEE International Symposium on Requirements Engineering (RE'97)*, Washington D.C., USA, 226-235.

Yu, E., and Mylopoulos, J. "Understanding Why in Requirements Engineering - with an Example." *Workshop on System Requirements: Analysis, Management, and Exploitation*, Schloss Dagstuhl, Saarland, Germany.