Marko Djukić [1)]
Milan Erić [1)]
Miladin Stefanović [1)]
Branko Tadić [1)]

1) Faculty of Mechanical Engineering, Univerisity of Kragujevac, Serbia,
mail: djuka84yu@gmail.com ,{ericm, miladin, btadic}@kg.ac.rs

PLANNING INFORMATION SYSTEM ARCHITECTURE – KEY PROBLEMS AND POSSIBLE SOLUTIONS

Abstract: *We live in an information era, where most employees, use Information Systems, dealing exclusively with processing information and the work of the same in all forms. Information systems occupy a major place in the management of all activities of the business system. In terms of technology, today has a large number of modern tools for the design of IS and DB. However, in addition to, a large percentage of projects related to development and/or implementation of information systems exceed the budget, break down the terms or does not satisfy customer requirements. This paper analyzes possible causes and possible solutions to this problem.*
Keywords: *Information system architecture, Conceptual modelling*

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’’ [2].

There is a high tally of failures in IS development projects, even though they are a focal point of industrial concern. According to, over 60% of IS projects represent a failure in terms of exceeding the budget and the deadlines and also in terms of unsatisfied requirements. The methods adopted to implement IS development projects, which are designed to avoid such deviations, are structured as generic time segments, the main phases of which are as follows:

- the preliminary study;
- detailed design;
- realisation;
- implementation;
- assessment.

The last four phases are organised according to project characterisation criteria, most of which are quantified on the basis of the IS architecture defined in the preliminary study. Basing our approach on the work of [4] and adopting the systemic angle, we propose the following definition for the notion of IS architecture. 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).

The IS architecture is, therefore, a map [5] which is absolutely necessary, during the preliminary study, to locate the technologies and organisational modifications which will be deployed during the development of the IS, in relation to the large number of existing solutions in these two fields. Consequently, calling the ‘‘preliminary’’ architecture into question during the detailed design phase, for example, implies not only rerunning this phase - at least in part - but also reconfiguring all the downstream phases. Indeed, the realisation and implementation phases are, respectively, linked to the technologies and organisational modifications.

Exceeding the deadlines in IS projects can therefore be attributed, among others, to an inaccurate definition of the IS architecture during the preliminary study phase. A bad definition can lead to mistakes in the quantification of the project's characterisation criteria. This leads to wrong choices being made when organising the downstream phases of the development project. The chain of events is represented in the Ishikawa diagram in Figure 1.

Thus, to reduce the time wasted by calling into question and redefining the architecture, means to guarantee the reliability of IS architectures determined during the preliminary study must be made readily available.

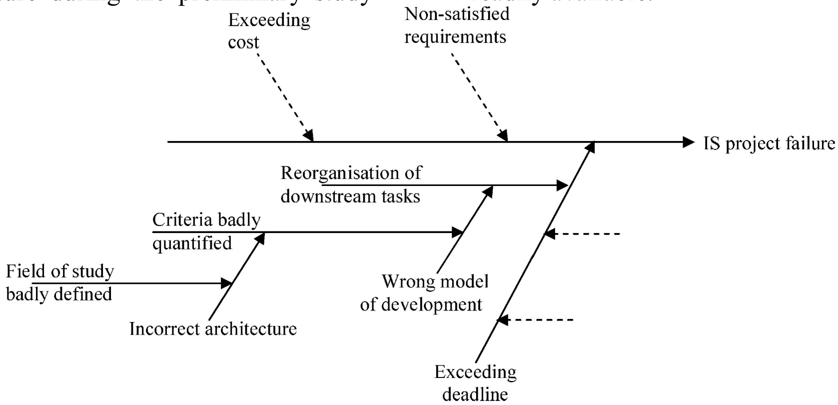

Figure 1. ‘‘Partial’’ Ishikawa's diagram of IS project failure

Within this context, the aim of this article is to suggest a reliable approach to working out an architecture, i.e. where nothing is called into question during the downstream phases of the IS development process.

Next part is dedicated to the study of approaches of possible used to design an IS architecture. The majority of these approaches are built up around an a priori choice of target technologies, and a minority adopt a different attitude. For the first group, the limited field of study has an impact on the quality of the ‘‘preliminary’’ architecture. For the second group, the link with technological components poses a problem. Therefore to be reliable design of IS architecture must combine an overall analysis of the IS and its reason for existing while guiding the choice of technological components.

2. POSSIBLE APPROACHES IN ARCHITECTURE DESIGN

Working out the IS architecture is a major problem in IS development, and can be approached in two different ways (cf. Figure. 2):

- approaches based on an a priori choice of the target technology;
- approaches not based on an a priori choice of target technology.

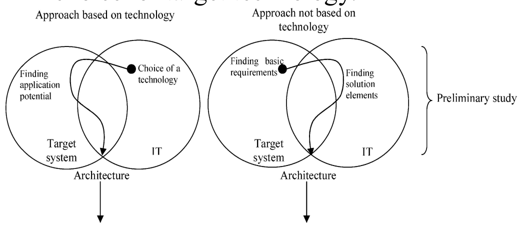

Figure 2. Two types of approach to building IS architectures

Schematically, the first class of approaches consists in characterising the technological components which can be implemented and in defining the minimum process required to identify their application potential within a target system. Implicitly, such approaches consider that the implementation of technological components constitutes the main improvement to the system. Calling the existing system into question is not the main objective here. Consequently, during the detailed design phase, certain developments to the system linked with introducing the technology may turn out to be problematic. In this category, the following approaches can be observed:

- proposing an architecture based on knowledge of the pair target technology/application conditions;
- proposing an architecture for IS based on explicit workflow models.

The second class of approaches focuses on the basic system requirements, but difficulties are still encountered when establishing the link with the technological components. The following approaches fall into this category:

- The architecture is designed from the results of a requirements engineering phase.
- The architecture can be built up by confronting the life cycle of the IS with other sub-systems within the company which are in the IS environment. An analysis framework must help determine the components which will be implemented following this confrontation.

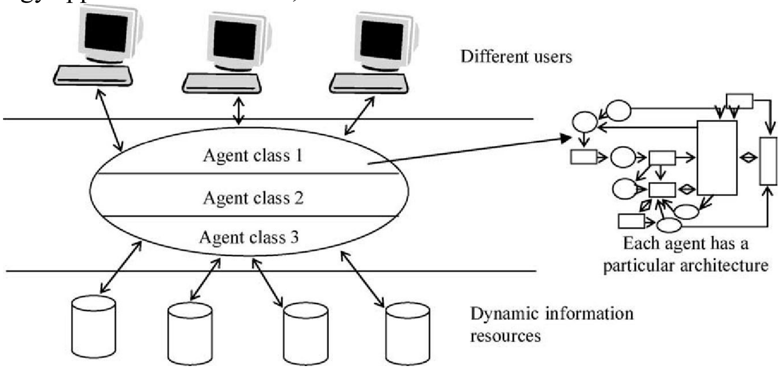

Figure 3. A particular agent architecture adapted from [3]

Research of this type is plentiful for agent-based IS, where defining the architecture amounts to selecting the number and type of agents to be implemented, their mode of cooperation and their field of action within the IS. In [7], for example, there is a proposal for an agent-based architecture within the context of open environment computing systems such as the Internet or large-scale intranets, in order to help users locate and retrieveinformation from a wide range of heterogeneous information resources (cf. Figure 3).

According to the objectives underlying the IS, it is possible to choose from among the many architectures proposed.

3. INFORMATION MODELING FOR MANUFACTURING SYSTEMS

With the development of the global economic market, it has become a major challenge for most manufacturing enterprises around the world to optimize their production strategies because the demand for better quality products with shorter lead-time and lower life-cycle cost forces them to be innovative in the product realization task, to make leaner their management and manufacturing operations, and to increase the efficiency of their day-today business processes. In order to meet these requirements, many enterprises face a critical need for advanced system-engineering tools and methods, because existing and/or new systems must be formally modeled, analyzed, specified and prototyped as part of their engineering and re-engineering processes. As the foundation for these engineering activities, however, what they need most is an advanced manufacturing philosophy for their production management and organization.

As a type of production and management strategy developed for about two decades or so, computer-integrated manufacturing (CIM) has been widely accepted by both industry and academia as the new generation of manufacturing technology. As the ultimate aim of the CIM approach, enterprise integration strives to achieve a proactive and aware enterprise which is able to act in a real-time adaptive mode, responsive to customer needs in a global way, and to be resilient to the changes in the technological, economic and social environments.

Observed manufacturing system (OMS). Observed system (OS) at present consists of more than 10 pieces of computer numerical control (CNC) equipment with a wide range in degree of automation, machining functions and material-handling manners.

It has been increasingly recognized that information integration is the first and also the most important problem to be solved during the system-integration procedure, because the information flow, or data flow, is the unique medium which ties together all the operational functions and activities that are completed by different components of the system, including hardware and software. A well-designed and properly implemented information system is one that can provide multiple information services throughout the manufacturing system at different levels to all system components in the most effective and efficient manner possible. Effective here means that the information provided has to be complete, accurate and on time, while efficient means that the manner by which information is provided is resource conservative and cost effective.

To meet these strategic requirements for an information system of an integrated-manufacturing system, it is very important to have an appropriate methodology to analyze, organize and construct the system-wide information model in such a way that, hopefully from it, the information system could be derived or mapped out. In the remainder of this paper, the system configuration and information flow structure in OS will first be briefly analyzed, based on which the information-modeling strategy will be carefully examined and presented. Then the IDEFlx methodology will be used to model the information system in accordance with the system requirements and information-modeling strategy.

It is known from the R&D plan and the function design for OS that the prospective OS system will be composed of three flexible manufacturing system (FMS)-type manufacturing cells, two distributed numerical control (DNC)-type manufactur-ing cells, and an automatic guided vehicle (AGV) and robot-based material-handling system.

As compared with the hierarchical control structure of a CIM system proposed by the National Bureau of Standards of the U.S.A., it is found from the system requirements and structure analyses that the prospective OS is a typical shop-floor manufacturing system. OS, as shown in Fig. 4, corresponds to the lower four levels in the CIM structure. In order to have a thorough understanding of the characteristics of information flow in OS and to construct a feasible information model for it, it is very important to collect all the information requirements in every respect at each level of the OS system.

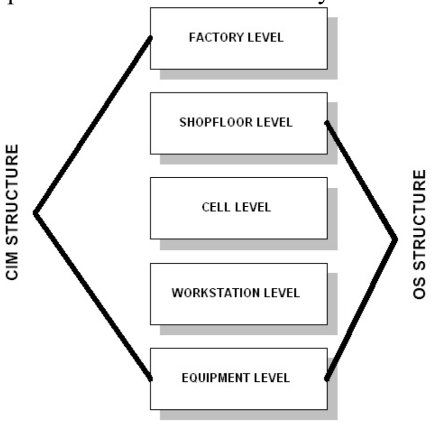

Figure 4. CIM structure of OS

Based on the information-requirements analysis, the function design for OS and by reference to the hierarchical control structure of the CIM, the information flow in OS is abstracted graphically, as shown in Fig. 5.

The objective for modeling all the possible information for OS is to develop a consistent, shareable, complete and non-redundant information system which should be able to serve all users and applications, both inside and outside the system. To make the constructed data model more compatible in a CIM environment, more advanced in technology and more conformable to the system requirements, it is very important to determine an appropriate data-modeling strategy, because it will make a notable impact on the effectiveness and reliability of the modeled information system.

The information flow of OS, as shown in Fig. 5, is organized hierarchically and analyzed conceptually in accordance with the hierarchical control structure of the CIM. The information model and the information system for OS can also be constructed hierarchically to conform to its functional structure. In this way, the data input and output mechanisms and data relationships must be established and defined separately at different levels. What must be pointed out, however, is that a great deal of information in the manufacturing system, e.g. design data files of product models and NC programs for part machining and material handling, is characterized by transmissibility throughout the system. These data with transmissibility are supposed and required to be shared by many different functions at different levels in the system. The degrees of data sharing are directly proportional to the length of the data-transmitting path. To meet this practical requirement during the system-operation process, it is a very important issue to guarantee the consistency and completeness of the data, as well as the shareability of those data with transmissibility at different locations.

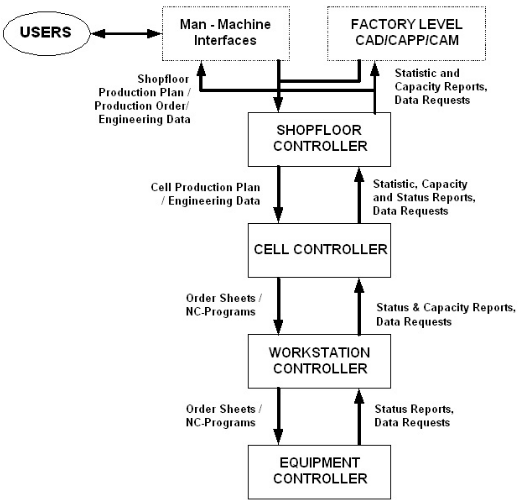

Figure 5. Information-flow structure in OS.

Due to the separation of data linkages among different levels, it can be naturally inferred that the hierarchically constructed information model for OS may result in data redundancy, data inconsistency and data incompleteness in its further-developed information system.

Considered from the viewpoint of system configuration, OS is simply composed of both hardware and software. Hardware implies the physical configuration of the system, which consists of the tangible objects such as equipment and computers. Software, on the other hand, implies those intangible system components that are responsible for implementing system-operation control and monitoring functions and activities. The combination of system hardware and software conceptually forms different system levels of the hierarchical control structure as shown in Fig. 4.

As the first step in organizing the information resources effectively and completely from the system point of view, the information system of OS can be modeled from the following three entry points as shown in Fig. 6:

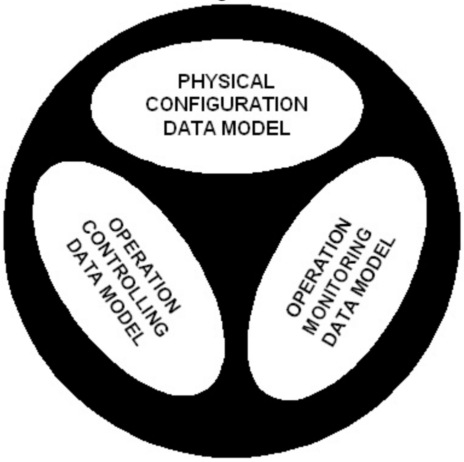

Figure 6. Data-modeling strategy for OS

Physical-configuration-oriented model

To implement the system-operation control and monitoring strategy, a data model is needed to provide all relevant system applications and users with required information about the system physical configuration. This model can be organized and constructed based on the following three categories.

Equipment category: comprises all the objects that can operate or move initiatively under the control of NC programs and implement the concrete operational functions, e.g. part machining or material handling, during the manufacturing process. In the OS system, this model is composed of such objects as machine tools, a coordinate-measuring machine (CMM), an automated guided vehicle (AGV), a robot, a washing machine, a drying machine, computers, a part-loading/unloading station, and a tool-loading/ unloading station. This categorized model will be able to provide relevant users and applications with basic information such as equipment identifiers, name and type, as well as further detailed information such as the operational parameters and specifications of the specific equipment.

Resource category: involves those physical objects that are unable to provide initiative services to system users or application, but meanwhile are imperatively required for the system operation. Objects modeled into this category are cutting tools, fixtures, pallets, buffers, material and personnel who are required passively at different working positions and treated as a sort of resource. Information such as resource identifier, name and their detailed parameters are very important to the manufacturing process planning and scheduling departments.

System category: to identify properly the information flow in the OS system and strengthen the system-wide operation control and management, the components that constitute the hierarchical structure of OS system must be modeled into this category. This model will comprise the logically existing components such as shop-floor, cell and workstation, and physically existing components such as central tool base and ware house. The modeling of these system-related components will facilitate the planning, scheduling, control and monitoring functions, and activities inside and outside the OS system.

Operation-controlling-oriented model

According to the operational requirements for OS, it should be capable not only of being integrated and operating compatibly in a CIM environment, but also of operating independently within its self-contained environment. So it is necessary for OS to be able to accept production orders, shop-floor production plans and engineering data either from the factory level or directly from system users by means of the man-machine interfaces as shown in Fig. 5.

The data model should contain all the objects that are necessary for controlling and scheduling the system operation, which involve production order, shop-floor production plan, task-allocation plan, cell-production plan, manufacturing-process plan, operation, NC-program, job group, job, schedule, order sheet, resource requirement, and supplying plans that are related to such objects as personnel, cutting tool, fixtures and material, etc.

Operation-monitoring-oriented model

In a manufacturing system, all the operation-related decision-making activities such as planning, scheduling, communicating and order dispatching, are carried out and triggered according to the accurate and timely data reports provided by the monitoring system composed of a large number of sensors and data processors, including hardware and software, throughout the system, which is the critical mechanism for guaranteeing the operational reliability of every component throughout the system. Objects to be modeled into this data model involve the following three categories of report.

Statistic reports: usually the resource-utilization statistics provide the upper levels with decision making references for the allocation of production tasks and the generation of production plans for the lower levels. Statistical information may be generated at the shop-floor, cell and workstation levels, respectively, in the OS system.

Capacity reports: usually consist of capacity and capability figures, are also the reference information for the upper levels to generate and/or decompose production plans for the lower levels. Capacity reports will be generated from cell, workstation and equipment levels, respectively.

Status reports: are the most important information for guaranteeing the reliability of system operation and will reflect the current states of such objects as equipment, workstation, cell, part and other resource objects. The status reports will provide the required current status data to all relevant components and activities for implementing their real-time controlling, adjusting and decision-making functions, such as dynamic scheduling and power switching-off.

As a particular object served by all activities and other objects in the OS system, the part may appear in all of these data models mentioned above because there exist many interrelationships between the part and the other objects in these models.

Once the data-modeling strategy for the information system of OS has been determined, it is very important to have an appropriate data-modeling methodology and tool to organize the information into a form that should be technically understood and easily transformed into a practical information system through further development and implementation. To meet the R & D requirements for OS, it is strongly recommended that the information system of OS be modeled into one with open-system architecture, which may possess the openness in both time and space domains to meet the specific requirements for the OS system, such as compatibility in a CIM environment, operation independence and flexibility, and system expandability and reconfigurability. There should be no doubt that a powerful database-management system (DBMS) will be the unique candidate to meet the requirements mentioned above about an information system for the OS system, because it can provide the system-wide data with consistent, reliable and shareable storage, efficient and concurrent access, convenient and effective manipulation. Since the logical data structure of a DBMS, whether it is network, hierarchical or relational, is usually system-oriented and may have more or less restrictions on its data definition, a so-called conceptual schema is suggested to separate the physical storage of the data (internal schema) from their practical applications (external schema) as shown in Fig. 7. A conceptual schema is considered as an integrated data definition that is independent of both particular applications and physical data storage and access manners. The main purpose of a conceptual schema is to provide a conformable and neutral definition about the modeled objects and interrelationships among them, which can effectively guarantee data consistency and completeness, and reduce data redundancy.

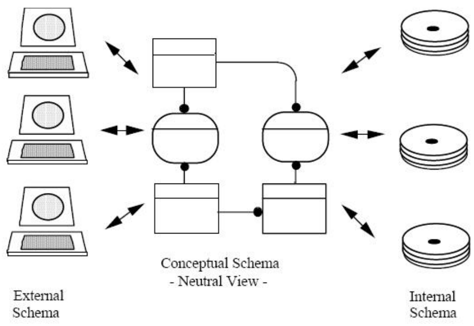

Figure 7. Three-schemata concept

The requirement to define modeled data conceptually brings about the development of semantic data-modeling techniques which can be used to define and/or determine the meanings of the modeled objects and interrelationships between them. A semantic data model is an abstracted data definition which specifies how these symbols stored in the data model are related to the real world. A semantic data model can be used to:

- organize and plan the data resource;
- construct shareable databases;
- evaluate some commercial software packages;
- integrate currently available databases.

It presented a set of IDEF methodologies to cover various aspects involved in the design of manufacturing systems.IDEF consists of several approaches: IDEFO allows a functional model to be built, IDEF1 an information model and IDEF2 a dynamic model. Recently, IDEF3 and IDEF4 were also developed to deal with the enterprise scenario and object-oriented modeling. IDEFlx methodology is an extended version of the IDEF1 after it was practically applied in the manufacturing industry and research institutions. As an effective approach, IDEFlx has been widely accepted by industry to model the information view of a manufacturing system, and also by the OS system because of the following characteristics:

- IDEF1x can support the development of a conceptual schema because its grammar can assure the semantic structure required by conceptual schema development, and a complete IDEFlx model may possess the expected data consistency, extensibility and transformability.
- IDEFlx is a kind of relevant language that has simple and conformable structure for different semantic ideas.

The semantics and grammar of the IDEFlx are powerful and robust, and also easily commanded by users.

With its strong expressive capability, rich semantics and simple development procedure, IDEFlx is very easy for lecturing and exchanging ideas among different working groups.

Formalism of the IDEFlx methodology. The basic formalism of an IDEFlx data model is graphically described, as shown in Fig. 8, with the following specifications:

- *Boxes:* represent the relevant objects which may contain some information contents and will be defined as entities in the model.
- *Connection lines:* represent data relationships between any two objects by means of text description.
- *Names inside the boxes:* represent the characteristics of an object which will be defined as attributes of an entity.

To configure an appropriate data model with the IDEFlx approach, the following procedure is usually recommended:

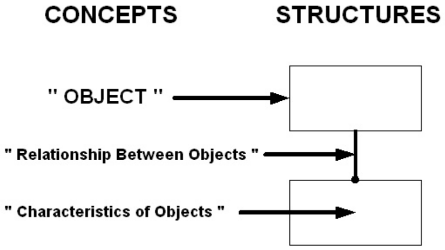

Figure 8. The formalism of the IDEFlx approach.

- analyze information thoroughly about the modeled system and/or aspects;
- define entities for the data model;
- examine and define the relationships between entities;
- define the key attribute(s);
- define attributes; and
- improve and modify the data model.

	SHOPFLOOR/1	CELL/2	WAREHOUSE/3	CENTRAL TOOL BASE/4	WORKSTATION/5	EQUIPMENT/6
SHOPFLOOR/1		●	●	●		
CELL/2	●				●	
WAREHOUSE/3	●					
CENTRAL TOOL BASE/4	●					
WORKSTATION/5		●				●
EQUIPMENT/6					●	

Figure 9. Data relationship matrix

Based on the information analysis and the information modeling strategy mentioned above, the construction of data models could be started from the second stage.

According to the modeling formalism, syntax and grammar of the IDEFlx methodology and the information-modeling strategy, three semantic data models are constructed for the OS system as shown in Figs 9, which correspond, respectively, to the system-physical-configuration-oriented model, system-operation-controlling-oriented model and system-operation-monitoring-oriented model. All the defined entities and their key attributes, data relationships, can be identified, respectively, from them. As for the other attributes of every entity to be described further, they will be determined at the detailed design stage in the system life cycle.

4. CONCLUSIONS

Information integration is the first, and also the most important, problem to be solved in the manufacturing system and/or enterprise integration procedure. This paper deals with the information modeling issue of OMS, an integrated system with flexible manufacturing and material handling capabilities. Based on the brief analyses of the system configuration and information flow structure of OMS, it presents a data modeling strategy for organizing the information resources in OMS from three such respects as system physical configuration, system operation control and system operation monitoring, respectively. Since the data modeling strategy organizes the information resource based on the characteristics of the information content and from the system point of view, rather than strictly following the physical controlling structure of the system, its implementation in future information systems will promisingly improve consistency and completeness of data, and reduce data redundancy.

With the characteristics of easy to follow, expressive graphics, rich semantics and a fairly straightforward modeling procedure, the IDEFlx methodology can be effectively used to model the system wide information of OMS into three semantic data models in accordance with the modeling strategy.

From the system implementation point of view, the IDEFlx based data models (conceptual schemata) can be easily transformed into the internal schemata and the external schemata can be extracted correspondingly, because they contain many details of further developed information systems and are related directly to the practical applications.

.

REFERENCES:

[1] T. Clancy, Chaos, The Standish Group International, Inc., 2003.

[2] N. Khomenko, D. Kucharavy, OTSM-TRIZ problem solving process: solutions and their classification, in: Etria World Conference—TRIZ Future 2002, Strasbourg, France, (2002).

[3] E. Shakshuki, H. Ghenniwa, M. Kamel, An architecture for cooperative information systems, Knowledge-Based Systems 16 (1) (2003) 17–27.

[4] Virginie Goepp *, Franc¸ois Kiefer, Franc¸ois Geiskopf: Design of information system architectures using a key-problem framework INSA de Strasbourg, LICIA, 24 Bld de la Victoire, 67084 Strasbourg Cedex, France Received 10 December 2004; accepted 12 September 2005.

[5] C. Merloa,b,*, Ph. Girardb: Information system modelling for engineering design co-ordination. aLIPSI/ESTIA, Technopole Izarbel, Bidart 64210, France bLAP/GRAI, UMR 5131, Universite´ Bordeaux I, 351 Cours de la Libe´ration, Talence 33405, France, (2004).

[6] Jian Zhang, Bing Chuah, Edmund Cheung and Zi Deng: Information Modeling For Manufacturing Systems: A Case Study. Department of Manufacturing Engineering, Nanjing University of Science and Technology, Nanjing 210016, People's Republic of China and Department of Manufacturing Engineering, City University of Hong Kong, Hong Kong, (2006).

Acknowledgment: Research presented in this paper was supported by Ministry of Science and Technological Development of Republic of Serbia, Grant III-44010, Title: Intelligent Systems for Software Product Development and Business Support based on Models.