# 8

# Distributed infrastructure and applications modelling: DIAMond

*J.B.M. Goossenaerts[1], A.T.M. Aerts[2], F.A. Mouws[1], R.J. van den Berg[3]*

*[1] Eindhoven University of Technology, Department of Technology Management, Section Information and Technology, P.O.Box 513, Paviljoen, 5600 MB Eindhoven, The Netherlands*
*[2] Eindhoven University of Technology, Department of Mathematics and Computing Science, Section Information Systems, P.O.Box 513, 5600 MB Eindhoven, The Netherlands*
*[3]Baan Institute, P.O. Box 231, 3880 AE Putten, The Netherlands*

**Abstract**

Business processes are frequently becoming distributed processes. This is due to the emergence of global markets and global manufacturing made possible by improved communications and support from ERP systems.
Optimization of business processes requires tuning of the process and application layers, and of the process and infrastructure layers. This requires insight into the consequences of decisions at one layer for another layer. Automated tools for analysis are expected to play an important role in the support of decision making during ERP implementations for distributed businesses.
The present paper covers the DIAMond project, its method, cases studied and intermediate results.

**Keywords**

Modelling of distributed processes, applications and infrastructure, analysis tools, performance indicators.

J. J. Mills et al. (eds.), *Information Infrastructure Systems for Manufacturing II*

# 1 INTRODUCTION

The 1990's mark the first decade when companies around the world have to start thinking globally to remain competitive. Time and distance have been rapidly shrinking with the advent of faster communication, transportation, and financial flows. Companies that are able to exploit these developments can put themselves in a position of decisive competitive advantage. In particular, the increased technical opportunities can be exploited to efficiently stretch the realm of integrated logistic control beyond the borders of the individual company and support so-called extended or even virtual enterprises (Kotler, 1991). But to realize this, more is needed than the state-of-the-art in infrastructure, per se, can offer. In particular, the information systems that support integrated logistic control, such as Enterprise Resource Planning software, should be inherently suited for a distributed environment.

Implementation of ERP systems in globally distributed companies shows that this type of information system is still too much designed with geographically centralized companies in mind. For ERP suppliers it is a challenge to gear their products more towards a distributed context. More options concerning distribution in space and time should be available. But a higher degree of flexibility in distribution of the ERP system also implies that more sophisticated support is needed to select at least a satisfactory *allocation* from a nearly infinite number of choices, during the implementation. Which allocation is chosen depends on the business requirements and the constraints concerning the application and the infrastructure in that particular situation. The objectives and constraints at these three levels are strongly interrelated (see Figure 1) which make the implementation far from straightforward.

The complexity of allocation selection described above led to the start of the DIAMond (Distributed Infrastructure and Applications Modelling) research project. The DIAMond project is supported by the Dutch Ministry of Economic Affairs, and is carried out by the Eindhoven University of Technology in co-operation with Baan Research & Development, Philips Consumer Communications, Baan Institute and Origin International B.V.

The goal of the DIAMond project is the development of methods, techniques and tools by which an enterprise can be modelled, such that insight can be gained into the performance of a multi-site ERP system and such that an optimal allocation of resources can take place.

Some of the questions are:

- Which entities are relevant in a model of a distributed application and infrastructure and what are their relevant attributes?
- What constitutes the performance of an allocation and how can it be quantified?
- What relations exist between allocations and their performance?
- What properties does a modeling syntax need to have to be able to describe the entities mentioned above and their distribution?

In its approach, the DIAMond project addresses methodology by looking at how the reference architectures of ENV 40003 and ARIS can be used to model distributed businesses, and how they can serve the expression of model alternatives, and the assessment of performance characteristics. An interesting aspect is that settings can be done at different modeling layers of the reference architectures as distinguished in Figure 1 and that performance characteristics will appear on the other layers. The methodology is then applied to a case to identify performance parameters and indicators and mechanisms to calculate with them.

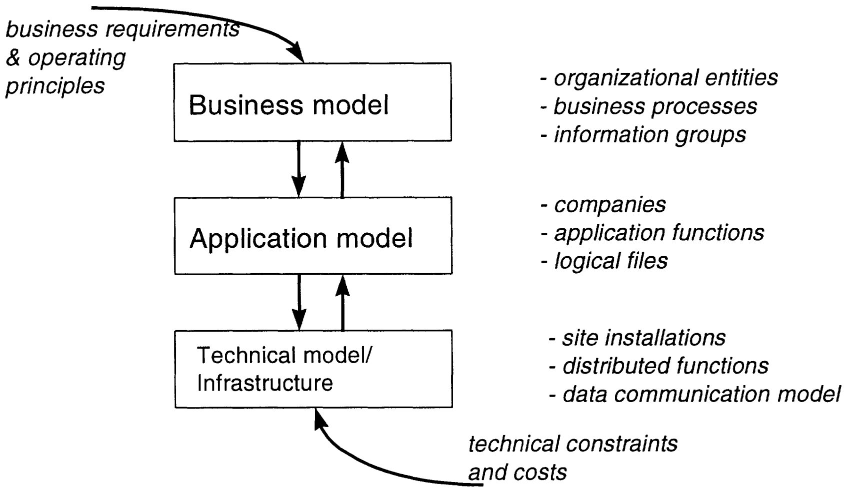


Figure 1 Layers between business requirements and technical constraints.

The DIAMond approach differs in focus from that taken in projects such as PRODNET II, VEGA and NIIIP, which are also concerned with distributed infrastructures. The emphasis in DIAMond lies on the modeling of distributed infrastructures and the applications using them in support of design and development, and not on building or developing such an infrastructure and making it available. For instance, the objective of the NIIIP (National Industrial Information Infrastructure Protocols) project is to foster the development of technology enabling industrial virtual enterprises and the adoption of STEP in such enterprises. The NIIIP goal is to develop, demonstrate and transfer into wide spread use the technology to enable industrial virtual enterprises. The PRODNET (Production Planning and Management in an Extended Enterprise) effort is aimed at the design and development of an open platform and adequate IT protocols and mechanisms to support virtual enterprises, in particular those involving small and medium size enterprises. VEGA (Virtual Enterprises using Groupware Tools and distributed Architectures) is aimed at creating a software architecture that will support the exchange of project and product information in a virtual enterprise. The

DIAMond project aims at providing primitives (and the methodology to use them) for modeling an (extended, virtual or regular) enterprise, and its business processes. The resulting enterprise model is capable of expressing particular performance aspects for various distributed designs for a given business process and this way supports a well-founded choice (with respect to infrastructure and applications) for its implementation in a distributed environment. The issue here is not how to provide a common platform or infrastructure, but how to allocate the various system components (hardware and software) to get suitable IT support for the business process.

In the remainder of this paper we will first discuss a case to provide some context for the distribution issues we wish to model. Then we give some background on the methodology used. This provides the necessary knowledge for the discussion of the questions cited above.

## 2 THE EM CASE

In this section the case of an electronics manufacturer (EM) will be introduced. In the next sections specific elements from this case will return for the purpose of illustration.

A large manufacturer of mobile telephones has a Central Sales Office for its European market (CSO/Europe). The manufacturer controls several warehouses around Europe and has three production sites on as many continents. The production of mobile phones takes place in two phases. In the *manufacturing phase* transceivers are produced that are suitable for all types of phones. The second phase is the *distribution phase* in which the phone is assembled. In this phase the transceiver is put in a case with a certain color, imprinting, logo and so on. After the distribution phase the products go to one of the warehouses. Diversity of the end products is high, with more than 400 models in the production range.

The CSO accepts orders from customers and from salesmen or sales offices. When an order comes in, the nearest warehouse is checked to see if the requested product is in stock there. When this is not the case, the CSO checks the production sites by means of an Available to Promise (ATP)-check, starting with the production sites that are near the customer. The Available to Promise check indicates which part of the planned production is still not reserved for specific orders. The ATP-check results in a delivery date that the CSO sends to the customer or salesmen/sales office. Depending on the delivery date the customer or salesman will place the order or not. When this happens, the ATP-number is adjusted: there is less available to promise. Naturally it is undesirable that a second customer or sales person performs an ATP-check before the first one places his order, based on the same ATP-number. That would easily lead to over sales. On the other hand, it is not wise to reserve already planned production based on an initial check only.

The sales volume of mobile phones for this company requires millions of ATP checks a year. On average an ATP-check has to be performed every three seconds. Each ATP-check generates a lot of communication. The problem that arises is that several ATP-checks are made at the same production site at the same time. To prevent this, unrealistic demands have to be made to the LAN, to the WAN, to processing times, processing capacities, etc. The current state of affairs is that the order acceptance process is too slow.

## 3 METHODOLOGY

The background for our approach is based on the analogy between the model layers in the ENV 40003 (CEN/Cenelec, 1990; Amice, 1993) framework (requirements, design and implementation) and the three layers linking business requirements, operating principles, and technical constraints and costs (Figure 1). The three model levels aid the enterprise modeler in structuring the modeling work (ENV 12204, CEN/Cenelec, 1995):

**Requirements Model Level.** At this level, one identifies the business requirements of the enterprise, using a business terminology, in terms of enterprise operations, but without reference to implementation options or decisions. Characteristic aspects of the business such as the global structure of its production process (which is split in two phases in the EM case) are given. The model is expressed in such a way that it can be used as a starting point for a consistent stepwise derivation of a design level model, and such that it can be processed for transformation and analysis of conformance and other criteria.

**Design Model Level.** At this level one specifies how the enterprise operations are to be performed, that is the actions and processes that are to be performed to achieve the requirements. An example is the ordering process in the EM case. It reflects the design phase of the specifications of the enterprise model, which meet the requirements of the enterprise. Constructs at this level model elements that are the concern of the design modeler and system designer from both a business and IT viewpoint. At the design model level constructs are derived from those of the requirements model level. They are then enriched with attributes that reflect general IT interfaces, such that the model can be used as a starting point for a consistent stepwise derivation of an implementation level model, and such that it can be processed for transformation and analysis of conformance and other criteria.

**Implementation Model Level.** Here one describes the means of execution -- by selecting the real Information Technology and Manufacturing Technology Components such as human resources, machines and programs -- and/or rules to be used in executing the enterprise operations as described at the Requirements Model Level. It reflects the implementation phase of the enterprise model, which meets the requirements of the enterprise. At this level the communication volume and response times and such come into the picture.

With respect to the dimensions of the generation of views, we feel more comfortable with those proposed in the ARIS approach (Scheer, 1994) (see figure 2). The views that ARIS uses to reduce the complexity of enterprise modeling differ from those in ENV 40003 (organization, resource, function, and information). The ARIS views are **function view** (formed by the functions to be performed and their relationships), **data view** (formed by *conditions* (customer status, article status) and *events* (customer order received, completion notice received)), and **organization view** (formed by the structure and relationships between users and organizational units). The elements of these views are then linked to each other in the **control view**, which integrates the design results. The three views together with the links between them form a business model. The function, data and organization view present a view of the business model at the meta- (type) level. In the control view, the consideration of instances (occurrences) can be included as well.

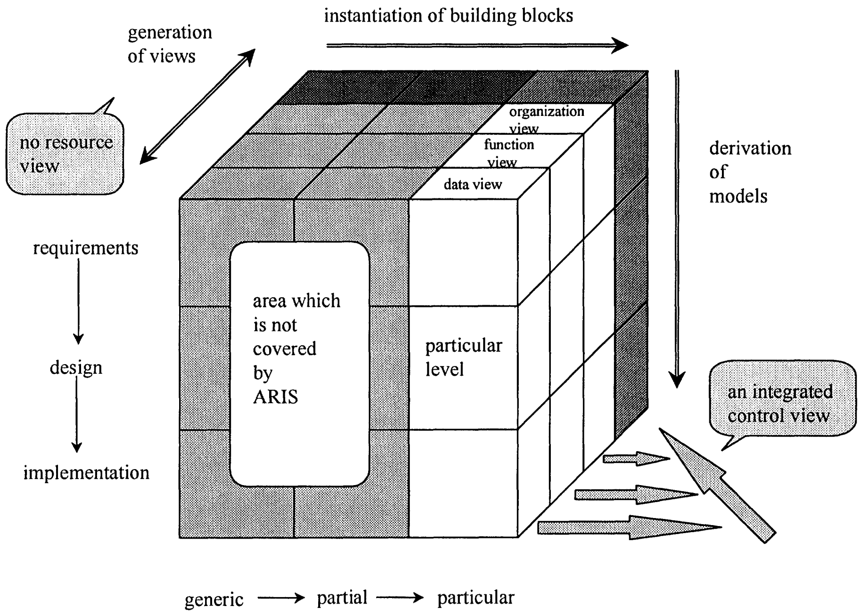


Figure 2 ENV 40 003 and ARIS.

Baan's Dynamic Enterprise Modeler also uses these views, but in addition utilizes a third dimension of the ENV 40003 framework, i.e. the concept of reference models (along the axis from generic, over partial to particular models).

## 4 DISTRIBUTION IN SPACE AND TIME

The views and the links between them are given a role in addressing distribution. Traditionally, models at the requirements (Figure 2) or business model level (Figure 1) only focus on types irrespective of distribution choices. Distribution would only be considered to matter for the design and implementation description. As businesses have become distributed themselves it is important to lift the visibility of distribution to the requirements/business model level.

Distribution in space is concerned with the allocation of data types, function types and organization unit types to sites.

Distribution in time is concerned with the degree of synchronization required for functions. Synchronous execution is natural in a centralized system. Asynchronous execution is natural in a distributed system.

*Distribution and Organization View*

In the modeling case of a single site, the organization view will discern different units. For the case of a multiple site business process, we can use the concept of organization unit to represent the sites (or their units). A site can also represent one of the companies in the distributed enterprise. For instance, in the EM case, there are sites for General Management, Development, Central Sales Office and Manufacturing. The CSO/Europe communicates with supporting sales offices and the customer services in the different European countries. Each of the organization units may consist of additional units.

*Distribution and Data View*

In the case of a single site business model, the data view will discern different data objects (types) that are used during business processes. For a multiple site business process model, we must allocate the data types in the data view to the different organizational units that were discerned in the organization view. One must determine which data should be managed at General Management, at Development, CSO, and Manufacturing. Within each of these units data may be allocated to sub-units.

Given the data view and the organization view of a business model, the following indicators convey information about the degree of distribution:

Data distribution: the number of unit/site-types where data-types have to be retrieved or to be updated by the processes in the business model;

Local Types: for each unit/site-type, the (relative) number of data types required by the processes that are located on the site.

Type distribution: for each data type, the number of site-types it has been allocated to.

*Distribution and Function View*

In the case of a single site model, the function view will discern the different activity types that make up the processes. For a multiple site business process, we must allocate the activities (types in the function view, executions in the control view) to the different organizational units that were discerned in the organization view.

Given the function and organization view of a business model (set of processes), we propose the following indicators:

Activity distribution: for each activity type, the number of unit/site types where the activity can be executed

Activity density: for each site type, the relative number of activity types that are allocated to it.

In the EM case, for example, salesmen and supporting sales offices are two site types involved in order entry, order sending, order acceptance (activity distribution is 2), CSO is the only type performing an ATP-check (activity distribution: 1). In another allocation performing the ATP-check might be a task of the salesmen and supporting sales offices, resulting in an activity distribution of 2.

Within the function view, activity-types can be linked to one-another by time-related demands such as synchronization or short response times. An example is the ATP-check and delivery date calculation for an order inquiry.

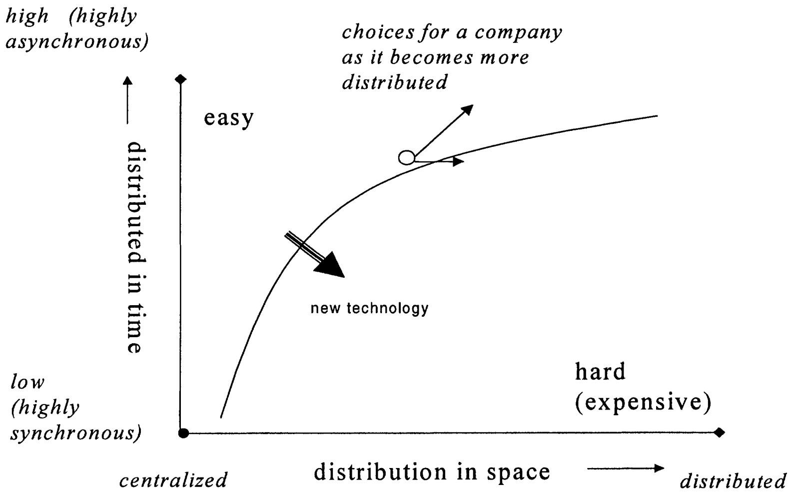


Figure 3 Simple heuristic guidelines for allocation.

Dependencies between parameters can be identified. Reasoning within a primitive view is relatively simple. Alternate data and function allocations can be characterized within the three primitive views. At the level of primitive views of the business model simple heuristic rules, such as those represented in Figure 3, could guide allocations. Figure 3 indicates the choices a company has to make when it becomes more distributed in space (for example by opening new subsidiaries). For example, it may invest in technology to keep the same level of synchronicity (horizontal arrow), or not do anything special and put up with a higher distribution in time in the form of more communication delays(diagonal arrow). Figure 3 also suggests that, as new technology becomes available, the border between what is easy (cheap) to realize, and what is hard (expensive) will move in favor of allowing less distribution in time (more synchronous operation) or more distribution in space. In-depth calculation of performance indicators depends on the control view and the further derivation of models in design and implementation level.

## 5 PERFORMANCE OF AN ALLOCATION AND CONTROL VIEW

In the control view, a reconstruction in space and time (following the decomposition of the process according to the three primitive views) of the business process takes place, in simulation (test) or in interflow (life/runtime) mode. Occurrences of sites and organization units are created, occurrences of data types (are stored in tables of databases and) participate in activity instances that make up the workflow of the business processes, either single site or multi-site.

Activities follow the allocation of data to sites. Activities and information groups are allocated to the most suitable sites, with interface protocols and replication policies for other sites that need the data or activity.

In the control view, detail and complexity is added, giving rise at the business level to indicators such as:

- frequency of activities;
- number of sites;
- number of simultaneous users (average, peak, expectations);
- response time: time required for activities and remote access;
- currency of information: the interval within which information is refreshed.

Alternate system designs can probably largely be characterized in the combination of the three primitive views. Allocating in the EM case the ATP-check to the sales offices, rather than to the CSO will result in a lower frequency of the activity per site. It will require on the other hand a larger effort to keep currency at an acceptable level: each office will have to propagate the sales it has closed to the others. How much exactly will depend on the choices made at the design and implementation levels. Ultimately, the performance of an allocation depends on implementation indicators, calculated from the available capacities for processing, storage, and communication. In the control view these choices are synthesized (in an upward fashion, integrating decisions made at implementation and design level, with the choices made higher up (in relation to the business process). For instance, the processing power needed for an ATP-check can be characterized in terms of the amount of CPU time it costs to extract the information from a local database and transform it into the number of items available at a certain date.

At the requirements level other aggregate indicators also play a role such as:

- cost of operation and of ownership;
- resource requirements;
- availability: the percentage of time activities can be performed;

since decisions are usually based on an optimum involving multiple criteria.

Indicators can be used in two ways. They can be used to reflect an actual situation, or they can be used to formulate a requirement for which an acceptable (in the multi-criteria sense) allocation has to be found.

Analysts or consultants who model or redesign a business process in a distributed environment in preparation of the implementation of an ERP system often will use modeling tools to support them. Especially in the case of complex software systems such as ERP software, good tools support is essential in gaining insight into the advantages and disadvantages of the different alternatives. This applies at all three layers (see Figure 1) and requires the collaboration of specialists for each layer. A business process designer will need to know if a certain set-up of a business process can be supported by the application and infrastructure, and must be able to verify this on the basis of a set of relevant indicators. Vice versa, an IT specialist will need to know the constraints arising from the business process and ERP software to tune the infrastructure. They will need to work together to construct a good compromise. An integrated tool supporting these activities will need to bring together the various aspects and thus must be founded on a sound, sufficiently expressive formalism.

# 6 TOOLS

In the previous section we discussed the impact of distribution on the four views and the three layers in the ARIS framework. One conclusion that can be drawn at this point is that distribution is a dimension of the model that is orthogonal to the four views already included. It should be made explicit at all three layers.

To support a methodology for the development of distributed information systems, such as ERP systems, a set of integrated tools is required that allows the construction of the various ARIS views at all three levels. It should also support the expression of a mapping between related concepts in the views at each level and between the models at successive levels. Additionally, the tool set should enable the systems architect to express the distribution aspects of the system at the three modeling levels and then use the distribution models as a starting point to calculate the various performance indicators at each level.

In this project we analyzed three formalisms for specifying information systems: ExSpect, LOTOS and SDL, and compared them with respect to their ability to express distribution aspects. ExSpect is based on hierarchical, colored Petri Nets (Hee, 1994), LOTOS on process algebra, and SDL is based on state transition diagrams (Turner, 1993). All three formalisms are designed to model dynamic, distributed information systems, and form the basis of a tools set to support the modeler, but none of them allows for the explicit expression of such entities as the physical location of a site, or the distance between sites. We therefore examined how these formalisms could be extended to incorporate such entities.

Various approaches to extending formalisms can be taken. One can extend formalisms by adding new syntactic constructs to them, thereby enhancing their expressive power to the required level. This approach implies that a new version of the existing tool set has to be produced which can handle the new extensions. Although this approach is a fundamental one, and allows for a rigorous design and validation of the required constructs, it requires a major effort, both in terms of research and development, and we did not take it within the scope of this project. Instead we studied how the existing formalisms can be used to suit our purpose.

Another approach is to annotate the existing formalism, such as was done recently in (Faltin, 1997). When the annotations are embedded in the standard formalism, for example in the form of comments, a co-specification of regular (functional, data) aspects and distribution aspects is obtained. The existing tools can still be used for the regular part of the specification. A separate tool has to be constructed that extracts the annotations and constructs a distribution view on the model from them. The advantage of this approach is that all specifications can be kept in one place, which makes maintenance easier. Another advantage of the annotation mechanism is that it can be used to include a specification of the performance characteristics such as capacities. These annotations then could be extracted and fed into a simulation tool or an analytical model to calculate performance indicators.

Since one of our goals is to be able to allocate activities to sites and then evaluate allocation alternatives with respect to some performance indicators, formalism extension through annotation appears to be an attractive option. Preliminary results from our analysis of ExSpect, LOTOS and SDL indicate that these formalisms, in principle, can handle annotations. The tools for these formalisms, however, also support a simulation mode. Instead of building a tool to feed the annotations into a separate simulation tool, a preprocessor approach could be taken. The preprocessor would take the annotated specification, express annotations in terms of the construction primitives for the simulation and this way generate an extended simulation model. The question here is to provide a suitable mapping from the annotations to the construction primitives. Such a mapping can be based on a number of standard constructions that can be instantiated for the purpose of simulation by setting parameters. Both ExSpect and SDL provide mechanisms to pass both data and function types as parameter. It therefore turned out that only a small number of construction primitives were needed to support the specification of the distribution and performance aspects we are interested in. We choose to pursue this alternative and are looking at a number of standard constructions. These constructions are used to model a number of cases, such as the EM case, to test their usefulness.

Using the standard constructions directly, one can run simulations to evaluate performance of alternatives for distributed ERP system (for example for the two business cases cited above). Simulation is a useful approach for evaluating such alternatives in considerable detail, but it requires both time (runs have to made for each setting of the parameters) and skill to set up the simulation and evaluate its results (requiring a sound insight into statistics). Therefore we also considered the mathematical modeling approach for evaluating distribution alternatives. Preliminary results indicate that, although analytical models are less flexible and usually are less detailed, these models can be used, e.g., at the requirements level to get a quick, order of magnitude impression of the relative merits of various alternatives. This would provide support for a quick reduction of the number of alternatives to a short list of candidates that have to be looked at in detail. Also in this case, the annotations have to be extracted from the specification and mapped to a number of standard constructions.

## 7 CONCLUSION AND FURTHER WORK

In this paper we looked at a number of questions related to the modeling of distributed enterprises to gain insight in the performance aspects of a multi-site ERP system and the way these are influenced by the allocation of resources. It appears that distribution has an impact on all levels of the ARIS framework and should be considered as an independent view on these levels. Just as the specifications at successive levels are related to each other (a specification at one level is derived from the one directly above it) as are also the distribution aspects. The relevant entities in a distributed system therefore are specializations of those in a centralized system, with added attributes such as location.

Performance aspects of allocations have been analyzed in line with the ENV 40 003 and ARIS architectures. This has resulted in an early taxonomy of indicators and a coarse view of the performance critical aspects of allocations, and their computational relations with performance.

We also looked at the properties a syntax for modeling distribution needs to have, and we have proposed two viable alternative approaches .

At present we are looking in more detail at the relations between the various levels and the way various choices influence each other. We are still experimenting with various choices for performance indicators using our standard constructions to gain some insight into their effectiveness for signaling potential performance bottlenecks. This should provide a sound basis for the next step: the incorporation of distribution aspects in a business modeling tool.

## 8 REFERENCES

AMICE ESPRIT Consortium, editor (1993) CIMOSA: Open System Architecture for CIM. Springer Verlag, Berlin, 2nd, rev. and ext. edition, 1993.

CEN/CENELEC (1995) ENV 12 204: Computer Integrated Manufacturing Systems Architecture: Constructs for Enterprise Modeling. CEN/CENELEC, Brussels, December 1995.

CEN/CENELEC/IT/WG ARC (1990) *ENV 40 003: Computer integrated manufacturing systems architecture -- framework for enterprise modeling.* European pre-standard, CEN/CENELEC, Rue de Stassart 36, 1050 Brussels, Belgium, 1990.

Faltin, N., L. Lambert, A. Mitschele-Thiel, F. Slomka (1997) An annotational extension of message sequence charts to support performance engineering. Proceedings of the Eighth SDL Forum, Evry, France, Elsevier Publishing , 1997.

Hee, K.M. van (1994) *Information Systems Engineering*, Cambridge University Press 1994, and references therein.

Kotler, Ph. (1991) *Marketing management, analysis, planning, implementation, and control*, seventh edition, the Prentice-Hall series on Marketing, Prentice-Hall International 1991.

NIIIP (National Industrial Information Infrastucture Protocols) home page: http://www.mel.nist.gov/msid/projs/97projs/niiip2.html

PRODNET II (Production Planning and. Management in an Extended Enterprise) ESPRIT project number 22.647. http://cupido.uninova.pt/~prodnet/

Scheer, A.-W. (1994) *Business Process Engineering -- Reference Models for Industrial Enterprises*. Springer Verlag, Berlin, 1994.

Turner, K.-J. (1993) Using Formal Specification Techniques - An Introduction to Estelle, LOTOS and SDL, University of Stirling, 1993.

VEGA (Virtual Enterprises using Groupware tools and distributed Architecture) ESPRIT project nr. 20.408. http://cic.cstb.fr/ILC/ecprojec/vega/HOME.htm

# 9 BIOGRAPHY

Dr. Jan Goossenaerts is assistant professor at the Eindhoven University of Technology. He holds degrees in mathematics, philosophy and computer science from the University of Leuven. In 1991 he received his Ph.D. degree from the same university with a dissertation on the design of an enterprise modeling language. Prior to joining the University of Eindhoven he was visiting researcher at the University of Tokyo (1992-93) and at the United Nations University, International Institute of Software Technology in Macau (1994). His research interests include the architecture of an information infrastructure for manufacturing, knowledge systematization for manufacturing, product life cycles and extended enterprises, and the application of virtual manufacturing.

Dr. A.T.M. Aerts received his Ph.D. from the University of Nijmegen in the Netherlands in 1979. He joined the Department of Mathematics and Computing Science of the Eindhoven University of Technology in 1985 as an assistant professor. His research interests include methods, tools and techniques for the specification, design and analysis of information systems, and for integration of and collaboration between information systems. Recently he has been working on topics in the area of supporting collaboration on the World Wide Web and on mobile agent systems. Homepage: http://wwwis.win.tue.n./~wsinatma

Frans Mouws holds an M.Sc. in Industrial Engineering and Management Science from Eindhoven University of Technology. His research interests are concerned with performance indicators for information systems.

Roel van den Berg holds an M.Sc. in Industrial Engineering and Management Science cum laude from Eindhoven University of Technology. He worked as a researcher for the Netherlands Organisation for Scientific Research (section for Computing Science) for four years before he joined Baan Institute, where he is employed as a research program manager since February 1997. He will defend his dissertation "Rigour with Relevance in Information Management; a two-dimensional perspective on integration of information systems" later this year.