

DESIGN FOR CONSTRUCTION (DFC) IN THE BUILDING AND CONSTRUCTION INDUSTRIES

G.T. Luiten and F.P. Tolman
Delft University of Technology, Ballast Nedam, TNO-Building Institute
PO Box 49 2600 AA Delft The Netherlands
August 1990

Abstract

This paper discusses the application of conceptual modelling techniques for the development of concepts for Computer Integrated Construction (CIC) in the Building and Construction industries. An extension of the General AEC Reference Model for the planning and production stages is proposed. As an example it demonstrates how the extended model can be used for the integration of design and construction of prefabricated concrete beams.

1 Introduction

For many years the Mechanical Industries have researched CIM application like: Design for Manufacturing (DfM) and Design for Assembly (DfA). Currently the mainstream of R&D in this area is devoted to Feature technology. The Building and Construction industries lack behind in the application of these state of the art technologies.

In the Netherlands we started a research project on Design for Construction (DfC) with three participants: Ballast Nedam (contractor), TNO-Building Institute (research) and Delft University of Technology. We define DfC as the integration of the design and construction process and all the processes in between. Our goal is to demonstrate DfC for prefabricated concrete structures. Two product types are being considered: a family of viaducts and a class of high rise office buildings. Currently, for practical reasons, the research concentrates on the design and fabrication of prefabricated elements both used in office buildings and viaducts.

The first step in integrating processes like design and construction is to analyse these processes in the current (AS-IS) situation. Here is determined which processes and sub-processes play a role and which information flows between the processes.

Using computers for the integration of building processes requires all the relevant entities and relations to be modelled in a conceptual model. A first version of such a Building Project Model (BPM) is in development, using concepts of the General AEC Reference Model (GARM), now part of the draft proposal of the ISO-standard STEP (STandard for the Exchange of Product model data) [1]. The current version of the GARM concentrates on the requirement and design stages of the product life cycle. For later processes like planning and production, and for the modelling of resources, new conceptual modelling constructs have to be found. A first attempt will be presented and illustrated. Also a first idea is given how the BPM can be used in a Computer Integrated Construction process (CIC) for the integration of design and construction knowledge and information.

2 AS-IS situation

The current situation of the building process for prefabricated concrete elements at a factory of Ballast Nedam is described with IDEF-0 (or SADT) diagrams. The main properties of the IDEF-0 methodology are processes or activities, modelled with boxes, and information flows, modelled with arrows. Information flows are classified in: input, output, control and support. Processes can be decomposed in sub-processes, e.g. the process with code A1 decomposes in A11, A12 etc. For a detailed explanation of the IDEF-0 method see [2].

The top-level process 'Realise prefabricated element' is decomposed in two sub-processes: 'Control realisation of prefabricated element' (A1) and 'Produce prefabricated element' (A2). The control consists of all the administrative jobs, like design, calculation, planning, control of the production etc. The control is connected with the production by working orders to and registration reports with results from the production units (IDEF-0 diagram figure 1). In the current situation much of the processes are sup-

ported with computer applications, but the communication between these processes is still traditional with technical drawings and paperwork.

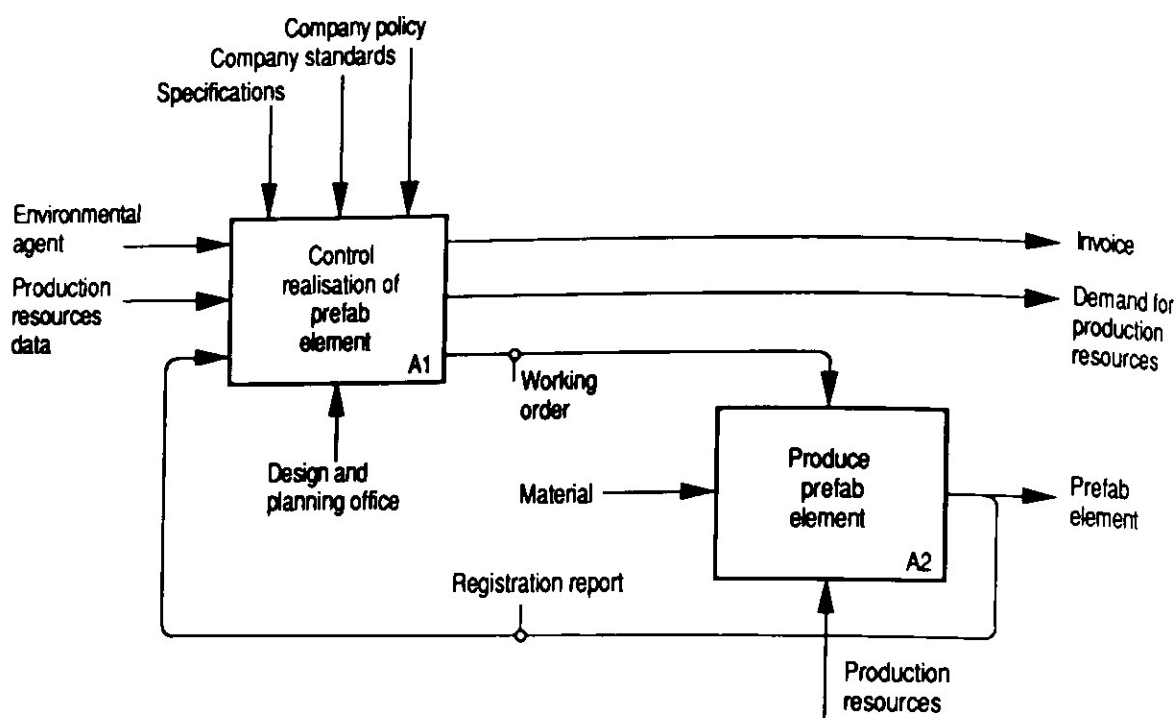


figure 1 IDEF-0 diagram of 'Realise prefab element'

3 Building Project Model

To support the processes, as modelled in IDEF-0, we need a model, a Building Project Model (BPM). During a building project there are three main groups of entities: the Product, the Activities and the Resources. The information about these entities can be modelled in respectively a Product Definition Unit (PDU), an Activity Definition Unit (ADU) and a Resource Definition Unit (RDU).

Examples of PDU's are: the product itself, parts of the product, assemblies and features. The choice for one of these categories depends on your point of view: a PDU can be the ending product for one party (e.g. a supplier) and only a part or a feature for another party (e.g. a contractor). For an ADU you can think of all the processes during the project: management, design, planning and production processes. RDU's are resources used by the ADU's, like manpower, equipment and raw materials. These resources can also be grouped in a company, a factory or a department.

There is quite some analogy with the IDEF-0 diagrams: the ADU's are the processes in the boxes, the PDU's are the input and the output arrows and the RDU are, in general, the support arrows.

For a PDU the main characteristics are 'shape' and 'material'. Other characteristics like 'strength' and 'weight' can be derived from the main characteristics. For an ADU the main characteristic is 'time', e.g. 'starting time', 'ending time' and 'duration'. For a RDU all the characteristics have something to do with 'money', e.g. 'application costs', 'acquisition costs' and 'remainder value'.

The modelling of a PDU is the major research object in ISO/STEP. One of the results is the General AEC Reference Model (GARM), which is part of a draft proposal. In the GARM concepts are worked out for PDU, that are also suitable for ADU and RDU. The most important concepts are illustrated by the example of the PDU 'Prefab concrete element' in chapter 4. To reduce the size of the model, common properties for PDU, ADU and RDU are modelled in a new entity called Manufacturing Definition Unit (MDU).

The relations between the Product (PDU), Activities (ADU) and Resources (RDU) can be graphically modelled in NIAM. For a short explanation of this graphical modelling language see [3] and [4]. The NIAM diagram of the Building Project Model can be found in figure 2. The PDU, ADU and RDU are subtypes of the MDU. In the MDU the identical properties are modelled. The state of the

PDU and the RDU are taken into account because they can be effected by the ADU. In general an ADU is preceded by a PDU state and succeeded by a new PDU state. An ADU always uses one or more RDU's. It is possible that this ADU also changes the state of the RDU. Some ADU's only effect the state of a RDU and not of a PDU. An example of the use of this model is given in chapter 5.

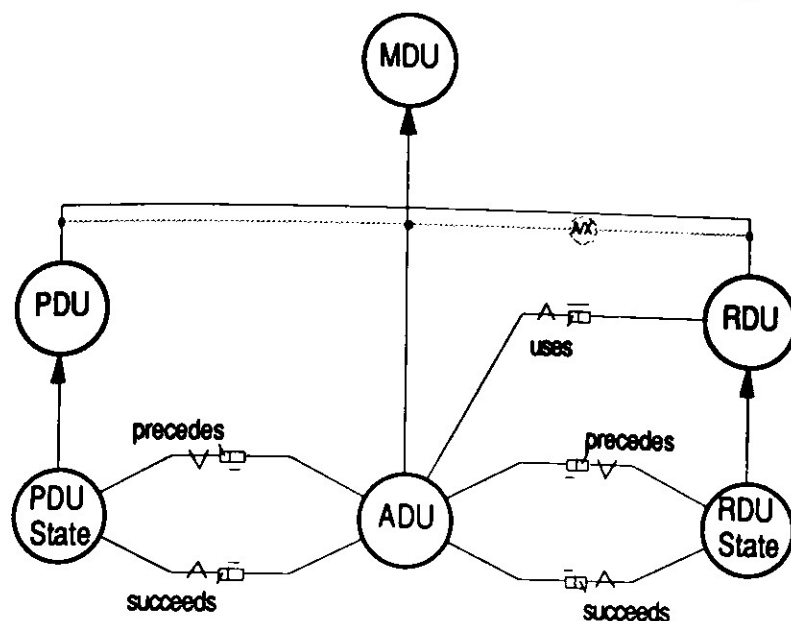


figure 2 NIAM diagram of the Building Project Model (BPM)

With this model all the entities of the building project are connected. Now we are able to say something about characteristics of entities that cannot be derived from its main characteristics. E.g. when you want to know, when a PDU is finished, you look at the 'ending time' of the ADU that precedes the state 'As Built' of the PDU. When you want to know the costs of an ADU you look at the total costs of the RDU's it uses and for the costs of the realisation of a PDU you look at the resources of the preceding ADU's.

4 Product Definition Unit according to GARM

The basic entity in the GARM is called PDU (Product Definition Unit). The major construct for a PDU is the division in life cycle stages. A PDU appears in different stages during its life cycle: As Required, As Designed, As Planned, As Built, As Used etc. Only the first two life cycle stages are worked out. A PDU in the As Required stage is called Functional Unit (FU) and in the As Designed stage Technical Solution (TS). These two stages are used for decomposition during the design of a PDU. The requirements of a FU are fulfilled by the properties of a TS. This TS can be decomposed into requirements of a lower level. These requirements are collected by new FU's. This decomposition mechanism is called 'FU-TS-decomposition' [3]. The FU's of a decomposition can be related to each other. For PDU's these relations are modelled with meta-topology [5]. For ADU's this can be done with network planning [6] and for RDU's with resource planning.

Another concept is the inheritance of properties of resembling classes of objects. You can compare this with the sub-/supertype relation in data definition languages. This concept is used to reduce the size (and the complexity) of the model. For data reduction and data re-use the concept of data-inheritance is used.

As an example of a PDU according to GARM we developed a model that describes all the properties of a family of prefabricated concrete beams. Such a model is called a producttype model. During the design process a product model for a specific beam is generated by choosing those properties from the producttype model that are needed to fulfil the specific requirements of that beam.

The kernel of our producttype model is the FU-TS-decomposition. As an example the first level is worked out here. It starts with the FU 'Beam'. In this FU requirements like required length, maximum height, extreme loads etc. are collected. These requirements originate from the specifications of the client, from computations for the viaduct, or office building, and from general regulations. The FU 'Beam' can be fulfilled by a lot of beams made of steel, in-situ concrete, prefabricated concrete

etc. Because of our choice for a prefab element factory only the TS 'Prefab concrete beam' is worked out for this project.

The 'Prefab concrete beam' will now be divided into sub-functions. Here a problem arises. The model will be used by different parties that all have a different view on the subject; e.g. a financial, a construction or a planning view. All these aspect views have to be supported without emphasising or neglecting one of them. This means the model has to be as neutral as possible, but you can't be totally theoretical without being too far away from the mental world of the users in practice. We decided to follow the design process as much as possible and to divide the beam in the FU's that are successively designed: a 'Body', a 'Pre-stressed element', 'Reinforcement field's and 'Cast in concrete object's. The NIAM diagram of the FU-TS-decomposition of a 'Beam' is presented in figure 3.

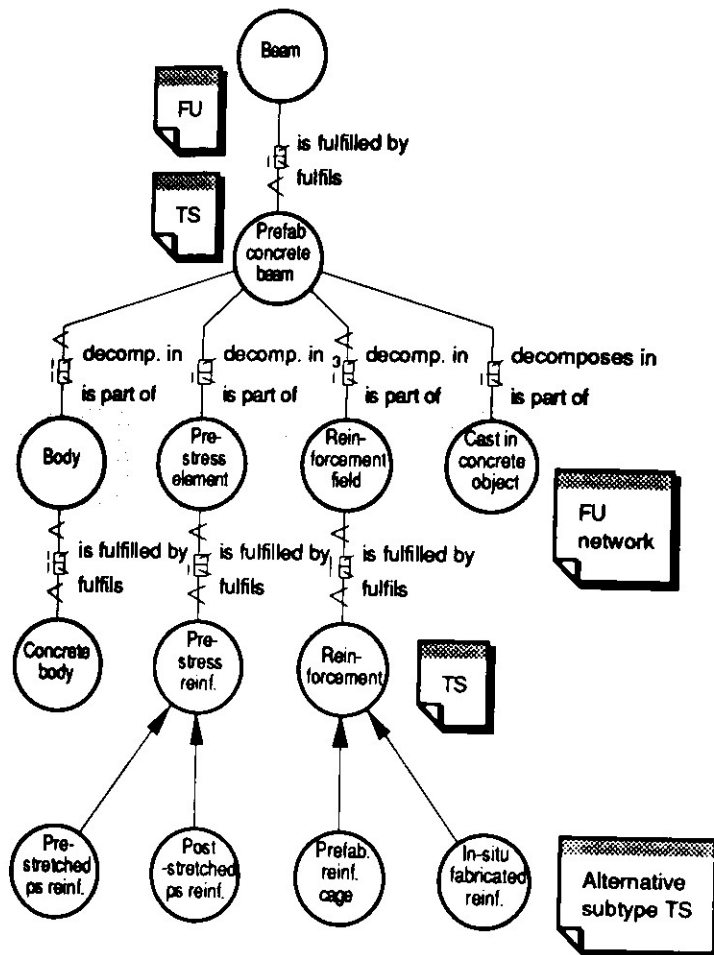


figure 3 NIAM diagram of the FU-TS-decomposition of a Prefab concrete beam

The FU 'Body' can be fulfilled by a TS 'Concrete body' with a standard shape (using a standard mould) or a new shape. The FU 'Pre-stressed element' can be fulfilled by the TS 'Pre-stressed reinforcement' and the FU 'Reinforcement field' can be fulfilled by the TS 'Reinforcement'. The other FU is not worked out completely yet. For the last two TS's you see the class-inheritance concept. The designer can respectively choose between 'Pre-stretched pre-stressed reinforcement' and/or 'Post-stretched pre-stressed reinforcement' and between 'Prefab reinforcement' and/or 'In-situ fabricated reinforcement'.

When a TS is worked out for a specific case, i.e. the parameters are set and the choices for decomposition are made, it can be (parametrically) stored in libraries. In this way it can be reused for other cases. You can see the stored TS's, including its decomposition tree, as features during the design process. You can always have that TS worked out by a specialist. E.g. this product type model of a beam is a feature for a designer of viaducts: he only sees it as a simple line to which he attaches some requirements. He has a specialist, the producer, work out the details.

We started the implementation of this model, using concepts as described in [3]. We were able to model a specific beam in Express and by using ProMod to generate a STEP-file, a quantity list and nice pictures. We started with the implementation of the producttype model in the object-oriented language Eiffel [7]. With this language it is not only possible to describe objects with attributes, but also to use multiple inheritance, to make user interfaces and to generate databases. For using the producttype model during the design process we needed an extended version of the GARM. That is why we developed the Building Project Model (BPM). The BPM only exists on conceptual level yet. We plan to implement this model in Eiffel and to use (parts of) ProMod for things like STEP-files, graphical facilities and quantity lists.

5 Example of Building Project Model

As an example of the use of the BPM in practice the production process of a prefabricated concrete beam is worked out. The beam is very simple: it only consists of a concrete body and reinforcement. The reinforcement is prefabricated by a subcontractor. All the other activities are done in the prefabricated element factory of Ballast Nedam. For the concrete body a mould is needed. In the factory a mould is placed in the mould, the concrete can be cast. When the prefabricated reinforcement is taken out of the mould and it is finished.

In figure 4 this process is modelled in a NIAM diagram using the concepts of the BPM. At the bottom level of the diagram you see the PDU 'Prefabricated concrete beam' that consists of the PDU's 'Reinforcement' and 'Concrete body' on the second level. On the fourth level the Activities are modelled in five ADU's and on the top level the Resources in the RDU's 'Subcontractor', 'Prefabricated element factory' and 'Mould'. Notice that the 'Mould' is part of the 'Prefabricated element factory'. On the third and fifth level you see the states of the PDU's and the RDU's before and after an Activity.

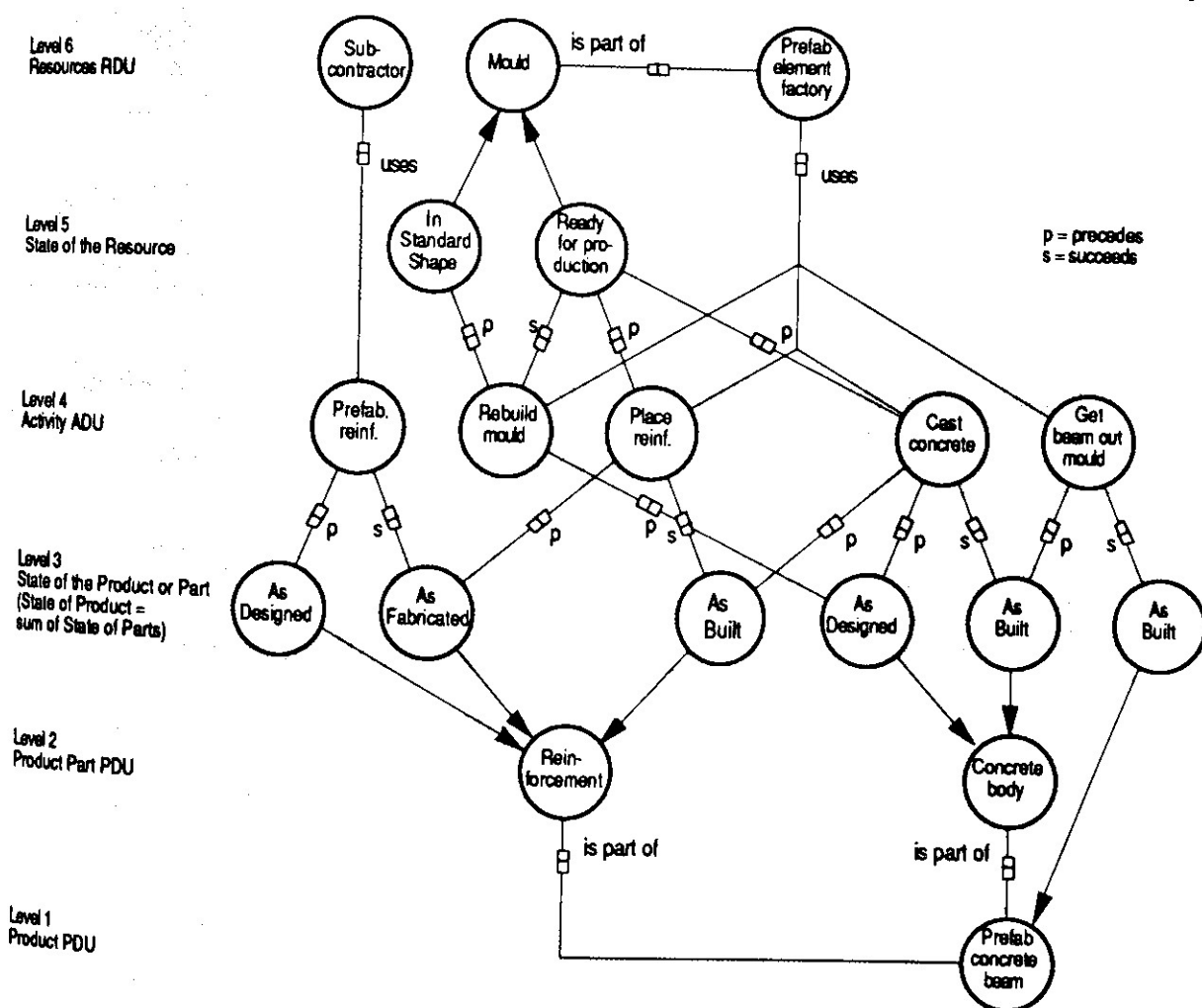


figure 4 NIAM diagram of the production process of a prefabricated concrete beam

The ADU 'Prefabricate reinforcement' changes the state of the 'Reinforcement' from 'As Designed' to 'As Fabricated'. It uses the RDU 'Subcontractor'. All the other ADU's use the RDU 'Prefab element factory'. First the mould is rebuilt in ADU 'Rebuild mould'. The state of the 'Mould' is changed from 'In standard shape' to 'Ready for production'. This ADU can only take place if the 'Concrete body' is in its 'As Designed' state, at least for the shape aspects. The next ADU, 'Place reinforcement', can only be performed if the 'Mould' and the 'Reinforcement' are in the right state. It results in the 'As Built' state of the 'Reinforcement'. Now the ADU 'Cast concrete' takes place. It starts with the 'As Designed' state of the 'Concrete body' and it results in its 'As Built' state. Requirements for this ADU are: 'Mould' in 'Ready for production' state, 'Reinforcement' in 'As Built' state and the 'Concrete body' in 'As Designed' state (now also the material aspects have to be designed). The last ADU 'Get beam out mould' starts if the 'Concrete body' is in its 'As Built' state. At the end the 'Prefab concrete beam' is in its 'As Built' state and the production process is finished.

6 Design for Construction

Before we can start the integration of design and construction with the developed BPM, models for activities and resources have to be worked out, similar to the product model. Also relations between the three entities have to be worked out on a detailed level. From that point on the integration consists of three steps.

The first step in the integration of design and construction is an automation of the information flow from the design process to the planning and production process. Many characteristics of the ADU's and RDU's are directly related to characteristics of the PDU. E.g. the first global estimations for the duration of Activities and the costs of Resources in a prefab element factory are linearly related to the volume of the concrete body and the weight of the reinforcement.

During the building process, data is stored in a neutral file according to the STEP format. Computer applications can derive their input from such a STEP file. When the application is finished, it stores the relevant output in the STEP file, so it can be used by the following applications.

Automated information transfer is more reliable, faster and more flexible than human supported information transfer. It is a precondition for flexible production automation, because it is the only way to control such a complex information flow. For a prefab concrete element factory, where every element is 'custom made', it is a must.

The second step of DfC is to give the responsibility of design decisions to the most qualified persons. E.g. a client only gives the requirements for a prefab element, the supplier designs it, so it fits best to his production process. This means better quality and lower costs. An other example in a prefab element company is the blending of the concrete body. The design department only says that there must be a blending with some requirements, the planning office determines how the blending will look like, depending on the shape of the moulds just before production.

The third step of DfC, and the most difficult one, is to use planning and production knowledge during the design process. When standard activities are attached to product parts and standard resources to activities, you can have a first impression for production time and costs during the design process on a very short notice. Planning and production knowledge can be used in an expert system for the design of an element and after the design the production can be simulated on a computer. In this way (financial and time) consequences of design decisions can be predicted.

7 Conclusions

The suggested top level for a Building Project Model (BPM), using concepts of the GARM, is very promising. It is still conceptual and on a rather global level. Before we can say more about the future of this BPM it has to be worked out on a more detailed level and implemented in some prototype version. Currently we are working on the implementation of already worked out parts of the BPM.

For the decomposition of a PDU during its design period we chose to follow the design process as much as possible, because such a decomposition supports all the aspect views without being too far away from the mental world of the users in practice.

Implementation of the BPM asks for a flexible, object-oriented language with a lot of possibilities. We believe that the language Eiffel can fulfil those demands.

Acknowledgment

The Building Project Model could not originate out of nothing. Wim Gielingh gave us a good starting point with his enthusiastic discussions or lectures. He is working on an integrated data reference model for the mechanical industries in an Esprit-CIM project, called IMPPACT.

References

- [1] Gielingh W.F., *General AEC Reference model (GARM)*. Proceedings CIB W74 + W78, October 1988, Lund, Sweden
- [2] Softech Inc., *Integrated Computer-Aided Manufacturing (ICAM), Architecture part II, Volume IV - Function Modelling Manual (IDEF-0)*. June 1981, Waltham, MA, USA
- [3] Tolman F.P., Kuiper P. and Luiten G.T., *Product Modelling at Work*. Proceedings CIB W74 + W78, September 1990, Tokyo, Japan
- [4] Turner J.A., *A systems approach to the conceptual modelling of buildings*. Proceedings CIB W74 + W78, October 1988, Lund, Sweden
- [5] Willems P.H., *A Meta-Topology for Product Modelling*. Proceedings CIB W74 + W78, October 1988, Lund, Sweden
- [6] Krom R.K. and Tolman F.P., *The Role of Standardized Product models in Construction Robotics*. Proceedings CIB W74 + W78, September 1990, Tokyo, Japan
- [7] Meyer B., *Object-oriented Software Construction*. Prentice Hall, 1988, New York, London, Toronto, Sydney, Tokyo