Integration of Life-Cycle Constraints in Design Activity

Tichkiewitch S., and Brissaud D.
Laboratoire Sols, Solides, Structures - I.N.P.G./U.J.F./C.N.R.S.
Domaine Universitaire – BP N°53 – 38041 GRENOBLE Cedex 9 - France

Abstract: The constraints concerning the whole life-cycle of a product must be integrated as soon as possible during the product design process, in order to decrease the final cost and to reduce the time to market. For this purpose, we propose a multi-actors and multi-views Cooperative Design Modeler, named CoDeMo, that allows the integration of different partners during the design activity. Our design methodology proposes the emerging of a product model from the specific constraints, results of the activity of the technologists, the manufacturers or the recycling actors. CoDeMo facilitates the dialogue between the diverse professions with the specification of the links existing among the views and by proposing a coordination system.

Key words: Design, Integration, Knowledge intensive CAD

1. INTEGRATED DESIGN CONTEXT

In order to reduce the time to market, the cost of a product, and, in the same time, to increase its quality and the capability for it to be recycled, designers have to use a new methodology for the design process, called integrated design (Tichkiewitch 1994) and based on concurrent engineering environment (Solhenius 1992). In such a context, a designer is not an individual people, but is one of the components of a design team composed of every actor concerned at any instant by the life of the product. In the team, we found persons as different as technologists in charge of the choices of the technological solutions, scientists in charge of the evaluations of the stresses with structure analysis tools, manufacturers, maintenance partners or recyclers in charge of the determination of the different processes.

Each of those actors must participate to the common work and give his own constraints as soon as possible during the design process. So he has to know what has been done by the previous actors, be able to see what has been already decided and what are the current functions and constraints on the required product. We propose that each actor is able to work on the same product data base, called the product model. The needs for a product model has been discussed in (Krause 1993). That product model is built with a lot of elements (components, links and relations) and a specific grammar in order to be always coherent (Tichkiewitch 1996). We remind in this first chapter the knowledge model and the data model that have to be used in order to solve this paradox.

The new Integrated Design Modeller CoDeMo, which is a multi-actors system described in (Tichkiewitch 1997-1), gives to each actor an access to the common product model. We evoke in the second chapter the main concepts used in CoDeMo.

It seams to be clear that it is not possible to ask some participants to work together without giving us some help in order to coordinate them. So, the third chapter is related to the coordination principles generally used, and specifically those of our system.

In the conclusion, we saw that complex problem may be solved with such an emerging design methodology.

1.1 The knowledge model

The notion of feature in product design has been studied for a long time and particularly in (Ovtcharova 1991), (Salomons 1994), (Taylor 1994) or (De Martino 1996). The domain features come from those definition and are more specific to an expert domain (Tichkiewitch 1997-2). The lector who wants more details on the feature concept can read those references.

A feature may be considered as an atom of knowledge relatively to a specific actor. In such a context, this feature is called vernacular feature. A lot of features which represent the knowledge of a specific trade is the property of this trade, and is available at any time during the design of a product. Each trade owns a feature library.

Some features are recognised and used by several actors of different trades. Those features have a meaning for each of those actors and they can be considered as multi-context features. They are called vehicular feature. The idea consists in considering those features communication objects where negotiation and compromise between actors must be discussed.

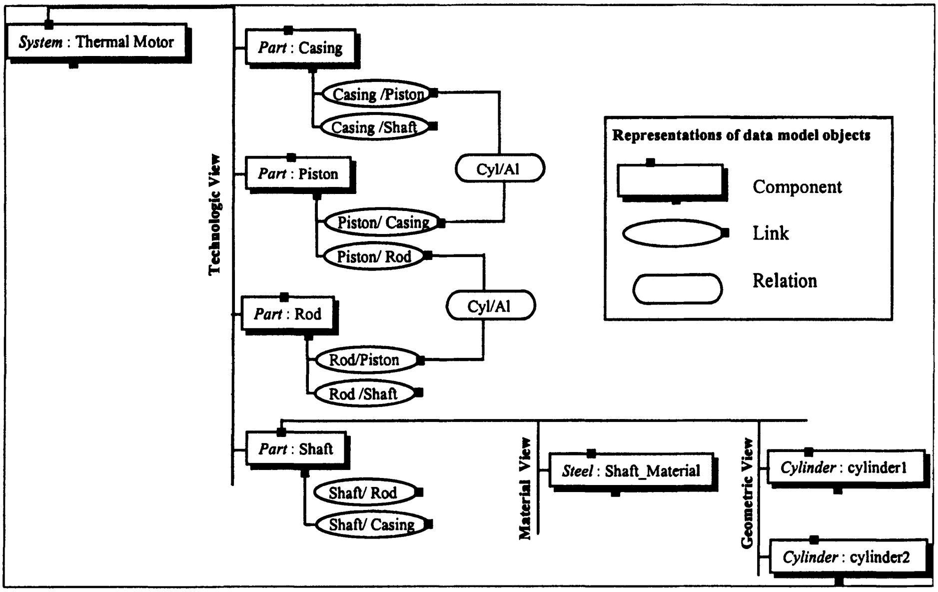

Figure 1. Partial graph of a thermal motor with multi-views decomposition

In (Tollenaere 1995), the definition of skin features and skeleton features are proposed. They are open features which can be used in a common view. The use of such open features facilitates the notion of integrated design in an industrial context (Lenau 1993), and gives the possibility to use common intermediate objects in a negotiation context between actors. Such open features are called universal features.

1.2 The data model

As we said in the introduction, the main lack in CAD system are the real tools for actors to cooperate and to share data during the design process. CoDeMo objective is not to integer knowledge in order to take automatic design decisions as it was with the development of expert systems. CoDeMo assists the integration and the co-operation of each participant who would be external or internal actor. It achieves thereby a formal network of co-operation based on the shared database, which will be manipulated by every design external actors and managed by the internal actor.

External actors use a graphic interface in order to have specific views (point of view) on the product. From these views and trough the formal network, they will modify or create new product data, which will be added to the shared database. In such case, new data are notifies by CoDeMo and therefore seen by every actors.

Internal actor manages the database structured with a data model proposed by MatraDataVision and based on Components, Links and Relations objects (Debarbouillé 1994). Chapa added to this model the multi-views concept (Tichkiewitch 96). This last amelioration takes into account the large kind of knowledge used during the design. It then allows the decomposition of the product in many groups of data related to specific trade knowledge. Figure 1 shows different views decomposition of a thermal motor using the component, links and relation data model and its basic representations.

2. THE COOPERATIVE DESIGN MODELLER

The Cooperative Design Modeller (CoDeMo) we are developing is built around a distributed data base management system, allowing access to different actors logged on many workstations. These actors are an initiator of a product design project, an internal actor and a lot of complementary external actors with specific trade's knowledge. They gradually build the product data base, a complex graph made of components, links and relations as described before in the product model. Now, have a look on the manner the product model base is built.

2.1 The initiator

In order to start a new product design process, an initiator has to run the design application and to give a name and a password to the project. The first task of the application is to create an internal task which acts as an internal actor. This task is running as long as an external actor (the initiator or an other actor) his working on the project. It can fall asleep when everybody log out the application, but will be wake up as soon as an actor enter the name's project. We see hereafter the goal of such internal task.

2.2 The internal actor

At any time of the design process, the actors work with a lot of features. Each feature is defined with a name and is always associated with the trade which is concerned with. Often the feature definition includes a geometrical implicit reference and some complements which are the specific trade's point of view. So, in order to make easier the work of the initiator or any external actor, a task of the internal actor is devolved to associate the initial feature to the corresponding implicit form feature, and this increasing the definition of the geometrical view.

A second task of the internal actor is devolved to keep the coherence between constraints, as a propagation system. We saw that it is possible to associate relations with the links of the components in order to add some constraints. These constraints can act on the values of parameters (a > b), on topological elements (a is parallel with b), or on any kind of specifications. So a formal propagation system is a useful tool in order to associate the different constraints and is running as the second task.

2.3 The other actors

Each actor who wants to intervene during the design process can at any time connect himself on the design application from any workstation, can select the name of the project he want to reach with the help of an unrolled menu and has to give the appropriate password. A first question asks him to choose the trade he want to represent, in order to specify his interface and to give him access to particular tools. We are developing today different trade's interfaces on technology, forging, tooling, structure analysis and recycling.

With the use of a specific filter in accordance with each trade, the new actor keeps free the capture of all the information on the product model he is concerned with. So the actor can react to this model in order to give more information or to refuse any of the presented elements. In the first case, he dynamically participates to the emergence of the product, adding a specific view by example to a decomposition of a component, or only doing a substitution of a relation with a well known sub-graph. In the second case, he has to address the trade at the origin of the bad elements in order to negotiate a compromise. For particular reasons, the actor can also query the data base to obtain some complementary information which would have been filtered.

In such a modeller, any trade is able to take into account the design process all long as it runs and with the same rights that anybody else.

2.4 A new rule for the internal actor

As we have actually defined our design modeller, the design application is a multi-user and multi-task application. So we have to introduce a new task in order to manage this application, and this new manager task is attributed to the internal actor.

Each time an actor prepares a decision, in order to increase the product model, he uses some data, extracted from the known elements of the graph, and adds some complementary specific knowledge. We ask this actor to mark each used data during the time of his reflection (if this actor has to run a specific software, such as finite element method, this period of reflection may be very long). As long as a data is marked by any user, the manager task has to survey the validity of this data. If any actor want to change it, the manager task report this change to the actors who have marked the data in order to obtain a permission to do it. If somebody refuses, the manager task asks the protagonists to settle the conflict and to found a compromise.

One of the goal of the internal actor being the propagation of the constraints between relations, it may arrive that the propagation is impossible and in such a case the system fails. In this case the manager task has also to inform the actor who gives the last relation that this relation is in conflict with the previous data and has to give him the origin of concerned constraints. We can found a good analyse of such problem in (Degirmenciyan, 1996).

2.5 An external multi-media layer

During the development of the design process, we saw that we can have some conflicts to settle between different actors. We are sure that a good compromise can only result of direct dialogue. From the confrontation between two different persons, each of them being with his personal context, can emerge new ideas and new solutions. We use here some results of the systemic theory which attributes to the all system more possibilities than the sum of the individual possibilities (Le Moigne, 1977). The emergence of solutions is one way in order to obtain innovation during design. This cannot be the result of any algorithms. So it is of a great importance to permit the different actors to be creative by the use of communication techniques.

Such an emerging solution is also the way to obtain a result for a complex problem, as this result is the image of the network that has been weaved by the different actors. More connections have been taken into account by the actors during design activity, more relations exist among the views and make complex the connective graph.

In the proposed design modeller, each user can use the multi-media interface of the workstation in order to call another user and start a dialogue. This multi-media interface is composed of a phone system allowing to call anybody of the connected actors, a video camera in order to have a visual contact during negotiation and to transmit the view of any object and also a common notice-board to exchange written or drawn information.

3. THE COORDINATION AMONG THE ACTORS

3.1 Two families of models for the coordination of the design process.

These correspond to the two approaches for the modelling of the product development process which are proposed in the literature: a now classical approach based on the modelling of design activities and their interactions, and a new approach, more pragmatic, based on the network of professionals involved in the design process, using the intermediary objects. Those two approaches are complementary: the predictive activity-based approach becomes more realistic when the reactive professional-based approach is also processed. Such models have been described in (Tichkiewitch 2000).

3.2 Typology of coordination

A typology of coordination methods of a product development process is also proposed in (Tichkiewitch 2000). It is based on four criteria related to the coordination activity. Eight coordination systems, extracted from the literature and our experience, have been analysed through those criteria : Information workflow (1) considers that an optimal solution of coordination exists. It is based on the analysis of the information workflow among the diverse design activities in order to propose a sequence of activities that minimises the lead time considering mastered risks accepted by the company. An example is given in (Eversheim 1998 & 1999). Negotiated objectives (2) considers that efficient activities need a clear objective in order to have a result accepted by the concerned professionals. Examples are given by Brissaud on design and process planning (Brissaud 1998) or Troussier on design and mechanical analysis (Troussier 1999). Milestones (3) is a classical approach in project management. Bender identifies the critical parameters to discuss in rendezvous to argue solutions (Bender 1998). Professional rules (4) is based on DFMA methods and leans on the early integration of downstream constraints (Boothroyd 1992). Mapped parameters or features (5) is concerned with the necessary coherency of parameters or features belonging to different professionals' views. The mapping between those parameters or features is generally not obvious and needs complex functions based on manufacturing know-how and hypotheses as shown in (Brissaud 1999) (Tichkiewitch 1999). Design space sharing (6) consists in sharing the global design space into several design sub-spaces explored by different professionals. Grabowski proposes planning work areas and design work areas (Grabowski 1998). Joint parameters and features (7) considers that the complexity of the design problem cannot be shared and must be managed by direct negotiation. Intermediary objects can support this negotiation. Data coherency (8) leans on the fact that many constraints can be expressed and then propagated to maintain the coherency at every decision making.

3.3 Criteria

Nature of coordination.

The coordination can be thought as an a priori plan or a negotiation space. In an a priori plan, an intelligent analysis of product development process has allowed to forecast some entities of coordination. Those entities must be respected and therefore ensure a correct processing of several professionals' works: existence of input information starting a specific activity to be processed (1), processing of formal rules including fundamental constraints due to downstream activities (4), propagating constraints ensuring the validity of the decision (8). When negotiation spaces have been foreseen, the coordination becomes effective by discussion among concerned professionals; it is not imposed a priori. The negotiation can be about local objectives whose sense must be jointly defined for a complementary and efficient work (2), decision makings while milestones (3), solutions or part of solutions where parameter or feature mapping must be discussed by experts (5) or that need a joint agreement and justification to be accepted by all the professionals (7). Some systems combine the two natures by an a priori plan (for example a priori sharing of the design space (6)), on which can be added some coordination spaces (the sharing can be put into question if necessary by negotiation).

Product- or process-related coordination.

The coordination can be based on design process features (1 to 3) or product features (4 to 8).

Degree of specification.

It consists of the conditions for the coordination system to trigger off and depends on the level of the product definition or the current state of the design process the coordination system applies on. The level or state must be predefined or opportunist. A predefined state can consist of predefined conditions of processing (1) or milestones predefined at crucial moments (3). Local objectives must only be negotiated when necessary; it does not depend on predefined states, it opportunistly triggers off (2). Most of the coordination systems can be activated by predefined definitions of the product generally depending on the degree of specification of parameters or features (4 to 8). The coordination system must be processed all along the design process in a CE framework and, of course, from several degrees of specification (for example at the earliest phases of design). In the system (7), it is clear that some parameters or features must be jointly defined at degrees of specification predefined while necessary. But generally, the need to negotiate a joint parameter or feature occurs without any preparation.

Indicators of running.

Was it a correct coordination? How to follow the coordination through? Indicators must be generally defined to control the coordination effects. They are more or less quantitative by information on cost and time (1) or on the number of conflicts (2 to 8). The nature of the conflicts to be settled should be deeply studied to propose indicators usable to control the design process.

3.4 Coordination in 3S Computer aided integrated design system

Our CoDeMo system, presented in chapter 2, particularly focus on integration of manufacturing and recycling into the design phase (Tichkiewitch 1999). Three types of coordination have been currently implemented in the system: mapped parameters or features (5), joint parameters or features (7) and data coherency (8).

3.4.1 Mapped parameters or features

A feature is a piece of knowledge defined relative to a professional trade and can therefore be used in the specific view. The attributes of features can be defined by other features of the same view or other views. For example, a feature "bore skin", which represents a functional surface in the framing view, can be characterised by a feature "cylindrical surface" which represents the associated geometrical surface. This feature belongs to the geometrical view. An attribute of the feature "cylindrical surface" is the feature "axis" itself belonging to the geometrical view.

A mechanism propagates information and the system makes automatically an instance of features "cylindrical surface" and "axis" as soon as the feature "bore skin" has been created. The ad hoc relationships keep coherent the three features. Those relationships should have been described beforehand in the module about information propagation. the behaviour is the same at each new instantiation of the concerned feature.

Assisting by such a mechanism, a professional can only think about his trade. He can avoid to give information at the border of his competence domain. The distribution of induce information is assured quickly to the other views. One also takes advantage of the flexibility of the feature concept to work in a fuzzy domain, in the sense of not completely defined. A feature "axis" is known even if the whole set of its director coefficients have not been given. the relationships make the coherency of information sure. A data of a feature could not be changed without envisaging the consequence of the change onto the other data.

Propagating information can concerned features on the product (previous case) or features on the manufacturing process (for example, an interior threading operation can induce a boring operation, then a tapping operation, which themselves induce the need of specific tools).

3.4.2 Joint features

The semantic of a feature can be partly common for two different trades. A feature can belong to two different views. It is a common language between the two professionals.

Let us see for example the feature "burr". A forger manufactures the burr by processing the forged part. A machiner deburrs the part by specific machining operations. There is no doubt that the burr geometry gives properties used by the two professionals. Some properties can therefore be specific for only one professional. For example, it is the case of the input pressure needed to perform the burr in forging. Choosing a joint parameter results of course from negotiation of compromise between the involved persons. The system of joint features concerns both the descriptions of the product or the process, but not simultaneously.

3.4.3 Data coherency

The coordination is here realised by a system of constraint propagation continuously testing the coherency of data by checking that given relationships do not lead to impossibility. The data coherency concerns essentially the product definition.

An incoherence detected by the system of constraints propagation is indicated to the professionals involved in the constraint satisfaction. A session of negotiation is therefore necessary to settle ambiguity or impossibility.

4. CONCLUSIONS

The CoDeMo system presented in this paper is a system running now in Valencia (Spain), Enschede (Netherlands), and is actually installed in Bucharest (Romania) and Bangkok (Thailand). Such an open design system give the ability to use the best competencies of each team of users in their specific fields (electric motors for Valencia, metal forming for Enschede, mecatronics for Bucharest, extrusion for Bangkok).

Running with such a system as a virtual laboratory, the design activity obtains the benefit of the different competences and permits the solving of complex problem.

It is the result of many PhD student's works (Belloy, Chapa Kazusky, Roucoules and actually Kazan) at the 3S laboratory. The structure of the modeler is done and each team is able to add their own features and their own specific software. Experiments in cooperative design activities have to validate the different options we have introduced in CoDeMo. As a tool as to be done in parallel with the organization needed to use it, we think that the next years may increase the knowledge introduced in the system and prove the perspicacity of the main choices.

REFERENCES

Bender B., Tegel O., Beitz W. (1998) Management of cooperation in product development, In Krause F-L., Heimann R., Raupach C. (eds), *New tools and workflows for product development, Fraunhofer IRB Verlag*

Boothroyd G., Alting L. (1992) Design for Assembly and Disassembly, *Annals of the CIRP*, vol.41/2

Brissaud D., Blondaz L. (1998) Feed-back to design from knowledge on process planning by indicators, *In Proceedings of the TMCE Tools and Methods for Concurrent Engineering conference, Manchester.*

Brissaud D., Paris H. (1999) Co-operation in design: indicators as supports for the discussion between product and process engineers, In Kals H., Van Houten F. (eds), *Integration of process knowledge into design support systems, Kluwer academic publishers.*

Degirmenciyan, I. (1996) Du paradigme 'Essai-erreur' au paradigme du choix dirigé. Application à la conception mécanique. Acteur unique, acteurs multiples, *Thèse de doctorat de l'Ecole Centrale de Paris*

Débarbouillé, G. (1994) Design data structure - version 1.0, *Rapport interne, Matra Datavision*

De Martino, T., Falcidieno, B., Giannini, F. (1996) Feature-based modelling as a solution for concurrent engineering design, In *Computer Integrated Manufacturing and Automation Technology*, CIMAT'96, Grenoble.

Eversheim W., Deuse J., Graessler R., Schulten I. (1998) Information based optimization of workflow, In design and process planning, in Krause F-L., Heimann R., Raupach C. (eds), *New tools and workflows for product development, Fraunhofer IRB Verlag*

Eversheim W., Schulten I. (1999) Optimal degree of parallelism and integration in design and process planning, In Kals H., Van Houten F. (eds), *Integration of process knowledge into design support systems*, Kluwer academic publishers.

Grabowski H., Spath D., Agostini A., Michelis A. (1998) Integrated product development using planning work areas and design work areas, In Krause F-L., Heimann R., Raupach C. (eds), *New tools and workflows for product development, Fraunhofer IRB Verlag*

Krause, F.L., Kimura, F., Kjellberg, T., Lu, S.C-Y. (1993). Product Modelling: a keynote paper, *Annals of the CIRP*, 42/2:696-706

Le Moigne, J.L. (1977) La théorie du système général. Théorie de la modélisation, *Presses Universitaires de France*

Lenau, T., Mu, L. (1993) Features in integrated modelling of products and their production, *Int. J. Computer Integrated Manufacturing*, Vol. 6, No 1&2

Ovtcharova, J., Habinger, S. (1991) Feature-based reasoning in product modeling, *In Fourth International IFIP TC5 Conference on Computer Application in Production and Engineering*, CAPE'91, Bordeaux, pp 379-387.

Salomons, O.W., Van Slooten, F., Jonker, H.G., Van Houten, F.J.A.M., Kals, H.J.J. (1994) Interactive feature definition, *In IFIP international conference*, Valenciennes, pp 181-204

Solhenius, G. (1992). Concurrent Engineering, *Annals of the CIRP*, Vol. 41/2.

Taylor, L.E., Henderson, M.R. (1994) The role of features and abstraction in mechanical design, In *6th International Conference on Design Theory and Methodology, ASME DE, Vol 88*, Mineapolis, Minnesota.

Tichkiewitch, S. (1994). From CAD-CAM to integrated design, *Int. J. of CADCAM and Computer Graphics*, Vol. 9, No. 5, pp 609-621.

Tichkiewitch, S. (1996). Specifications on integrated design methodology using a multi-view product model, In *Third Biennal Joint Conference on Engineering Systems Design & Analysis, Montpellier, July 1-4, 1996, ASME, pp 101-108*

Tichkiewitch, S., Chapa Kasusky, E. (1997-1). Methods and tools for integration and holonic design, *Int. J. of CADCAM and Computer Graphics*, Vol. 12, No. 6, 12/1997, pp 647-667

Tichkiewitch, S., Veron, M. (1997-2) Methodology and product model for integrated design using a multi-view system, *Annals of the CIRP*, Vol. *46/2:81-84*

Tichkiewitch S., Veron M., (1998) Integration of manufacturing processes in design, *Annals of CIRP*, Vol. 47/1

Tichkiewitch S., Brissaud D. (1999) Diverse aspects of tolerancing on an integrated design context, *Annals of CIRP, Vol. 48/1*

Tichkiewitch S., Brissaud D. (2000) Coordination between product and process definitions in a concurrent engineering environment, *Annals of CIRP, Vol. 49/1:75-79*

Tollenaere, M., Belloy, Ph., Tichkiewitch, S. (1995) A part description model for the preliminary design, In Soenen & Olling (eds) *: Advanced CAD/CAM systems, Chapman & Hall, pp 127-143*

Troussier N., Pourroy F., Tollenaere M., Trebucq B. (1999) Information structuring for use and reuse of mechanical analysis models in engineering design, *Journal of Intelligent Manufacturing*, Vol. 10/1.