COBEM 99
XV Congresso Brasileiro de Engenharia Mecânica
15th Brazilian Congress of Mechanical Engineering
22 - 26 de Novembro de 1999 / November 22 - 26, 1999 Águas de Lindóia, São Paulo,

DEVELOPMENT OF A MANAGER SYSTEM FOR CALCULATION OF MACHINE ELEMENTS FOR MECHANICAL PROJECTS

SILVEIRA, Zilda de Castro
LIRANI, João
CARVALHO, Jonas de
Universidade de São Paulo, Escola de Engenharia de São Carlos,
Departamento de Engenharia Mecânica.

Summary. *With a market more and more demanding in terms of efficiency and costs, one has the need to integrated all the possible information of a company, helping in this way to guarantee the quality of products and services. Moreover, nowadays there are plenty of resources to systematize the repetitive and tedious tasks, such as calculating a shaft or a pair of gears. By using them, the engineer will hold all the data he/she needs and, at the same time, will have more available time for more creative tasks. With this focus, this work presents the development of a manager system for calculations of machine elements for mechanical projects. It was devised under the context of EDM–Engineering Data Management. The system is constituted of a manager program, a database and modules of calculation of machine elements. The manager program controls the flow of calculations in the project through existence and consistency rules among the mechanical elements, and also provides the availability of the input and output data. Those rules are fired through consultations to the data stored into a relational database. As consistency rule one can cite, for example, the one stating that the lubricant used in the gears calculation be the same as that used in the calculation of the bearings. The database was designed by using a logical model based on ME-R, taylored to the assembly of a specific mechanical system : a gearbox. The calculations are performed by independent computational modules which use traditional methods of elements of machines calculation (NIEMANN, 1971).*

Keywords: *PDM; machine elements; integration.*

INTRODUCTION

The technological progress that occurred in the last 30 years promoted deep changes in the market laws. The globalization process increased the consumers demand levels, forcing the companies to restructure their organization. The product development cycle tends to be much smaller, resulting in products and services with increased quality and reduced costs. In order to achieve these objectives, manufacturing companies have several approach to their dispodal, which onr of them enabling new technologies and philosophies.

In the end of the 80’s, simultaneous engineering has been proposed, so that tasks would not be performed sequentially, but in a parallel way. It was observed in the practice that this approach contributed to the reduction of the product development cycle (CHIUSOLI, 1996). However, for implementing the simultaneity concept, companies should readapt technical functions and invest in training multidisciplinary task forces. In this context, there are several computer tools and systems which are available to help the reduction of the time to market and to promote a continuous flow of information inside of the company.

Until very recently, CAD was essentially a geometric modeller that represented assemblies and parts as bi - and tri-dimensional objects in the form of a combination of computational structures such as wireframes, surfaces and solids. The designer would develop them by applying boolean operation. Several visualization problems and limitation of constructions of more complex parts were observed. Nowadays new concepts for the generation of complex geometries were presented, among them one can cite feature based modelling, parametric modelling and more interactive graphic interfaces (GUI's) based on objects.

Computer Aided Engineering (CAE) also appear as another aid to the design environment. It enables the analysis of a mathematical model of the part through the Finite Elements Method, for instance. This latter is a method of numerically solving differential equations which represent problems of solid mechanics so that, for example, stress and strain gradients; distribution of temperatures, pressures and velocities etc., can be obtained. Computer programmes based on traditional analytical calculation methods of mechanical components, exemplified by the system proposed in this work, is considered as part of CAE.

Still, one of the main obstacles to be overcome is the integration of the abundant data originated in CAD/CAE/CAM and CIM systems. In other words, it is still a challenge to organize, standardize, store and make these data available for general use inside the company : geometries of the project, engineering drawings, spreadsheets, part production data, assembly diagrams, specifications of the product, numerical control programmmes, analyses of results, bill of materials and engineering alterations, etc. The PDM (Product Data Management) system is a technologycal tool that provides such an structure (CIMData, 1997). Those data are manipulated by database management systems. As the integration involves conectivity tools for different areas, PDM should store and make available with integrity all the different information of the company among the engineering group as well as vendors and customers. In this sense PDM promotes an environment of simultaneous engineering and it collaborates with the real processes of development of the product, resulting in a flow of more homogeneous work, with less mistakes and redundancies. Figure 1 illustrates the general idea of the integration process inside the company.

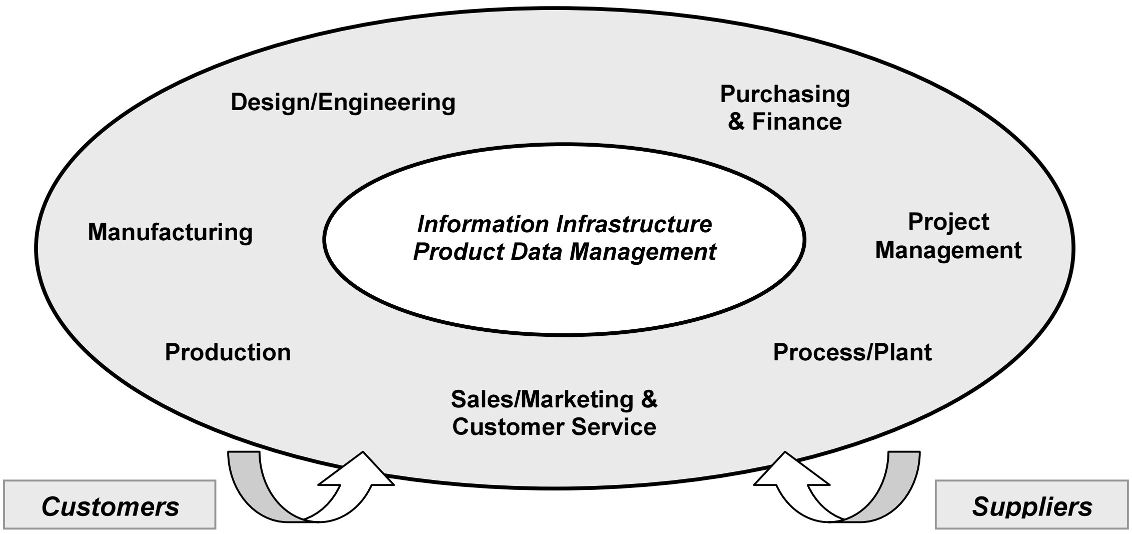

FIGURE 1 – PDM and the integration process.

However it has been felt that PDM treats the technologycal data in a not differentiated way. Therefore a subsystem is needed to bridge this gap. An specific Manager System to coordinate and control the calculation in the design process.

This work presents a manager system for calculation of machine elements for mechanical projects (*CMM – Calculation Methods Manager*). It comprises modules of calculation of different machine elements, a database to store, standardize and dispatch the data and a master managing module that contains the logic of the design and works by firing predetermined rules.

The structure of the proposed CMM

For many years, due to the limitations of the limitations of computational hardware and software, the calculation of machine elements of a project was made manually, aided by a series of consultations to books, manuals, catalogs and abaci. Those elements already present well accepted, systematic methodologies of calculations, which, by the other hand, are repetitive and error prone. Moreover, the time spent in calculation during the traditional design cycle is characterized by being split into sequential stages developed by the same designer, making it difficult to share the task with other engineers.

In order to try to overcome these limitations, the following needs were considered during the definition of the structure of the proposed system:

- Sistematization of the calculation procedures of the main machine elements, so that they could be computacionally efficient. This systematization should result into the basic logic of the design, making it possible to stablish design rules;
- Division of the CMM into two forms of operation: standalone operation of the calculation modules and integrated operation.
- Choice of an appropriate programming language which allowed modularity, portability and easy of mantenance;
- Friendly graphic interface, based on visual objects.
- The results should be readly transferred into CAD systems and transformed into drawings by using parametric technology.
- Choice of a database server with open structure.

Very early during the implementation stage one had to choose a real mechanical system to be studied. Gearboxes were the choice, due to the fact of being a well-known system, composed by the main machine elements. Figure 2 presents a general vision of the CMM, showing the three main components: calculation modules, database and system manager.

Modules of calculation

The building blocks of the proposed system are the computer programmes developed in modules which calculate the various machine elements based on classical literature: NIEMANN (1971) and SHIGLEY (1984), references contained in catalogs, abaci and technical standards DIN and ABNT. The following modules were developed and implemented :

- <u>Cylindrical gears</u>: the algorithm implemented in this module allows calculation of straight, helical and bi-helical gears. The user should supply some basic data as: input power, rpm, number of teeth, module, type of lubricant and materials.

With these data an algorithm is executed based on NIEMANN(1971), DUDLEY, DIN. It supplies the main dimensions and safety coefficients for bending of the tooth (SG), surface pressure (SB) and clogging (SF);

- Shafts: this module follows a hybrid method of calculation based on NIEMANN (1971), SHIGLEY (1984) and HANCHEN (1960). The method uses a preliminary sketch of the shaft. On this sketch the location of bearings and acting forces are given. Special attention is given to form variations that cause stress concentration such as stepped construction, keyways, holes, etc. Once the data are supplied, the module performs the static and fatigue calculations.

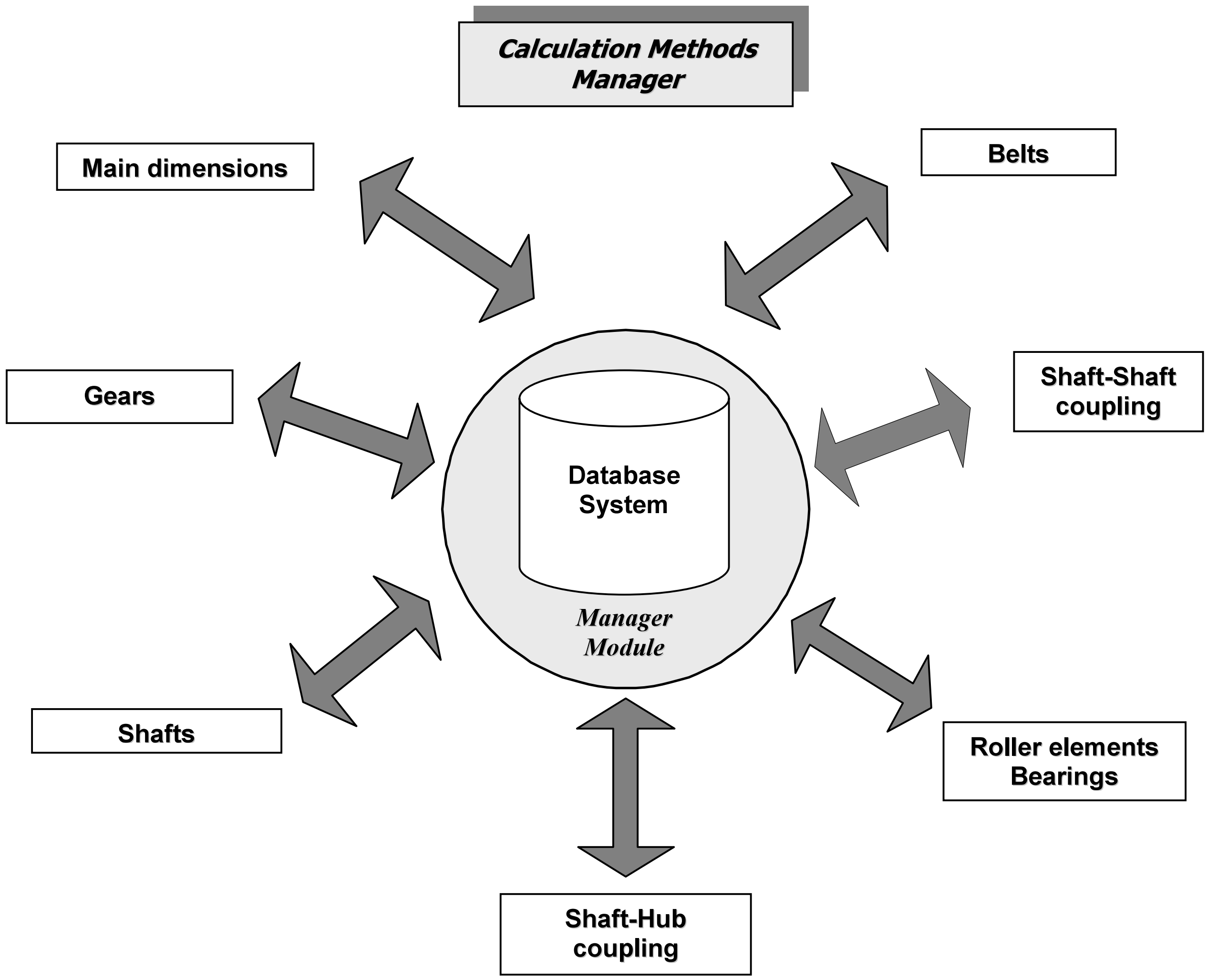

FIGURE 2 – System overview.

- Bearings : instead of the traditional direct approach of calculation which implies in several reassumptions and recalculations, this module makes use of an indirect approach. In other words, by using the design parameters such as bearing type, speed, acting loads and expected life, a minimum load carrying capacity is obtained allowing a good result with one iteration. Basic formulae and abaci are based on SKF(KK) used thoroughly in the industry.

- Shaft-hub couplings: this module has algorithms for calculation of couplings using square keys and using for traverse interference. The methodologies are found in NIEMANN (1971) and DIN 6885.
- Shaft-shaft couplings: this particular module possesses routines for flanged type coupling, plane coupling (Hirth) and split hub coupling, again using NIEMANN (1971)
- Belts: the same source provides the basis for the routines for calculation of V” belts.

The output data coming from the calculation modules are presented to the user either through INPRISE's C++ BUILDER graphical user interface, i.e., graphical windows, or by standard written reports. This is evidently not sufficient. Therefore a CAD interface has been implemented using routines written in AutoLISP to visualize the component in AutoCAD. In this way, interactivity with the user is greatly enhanced. Figure 3 presents the the typical flow of information inside a module and the respective computational tools.

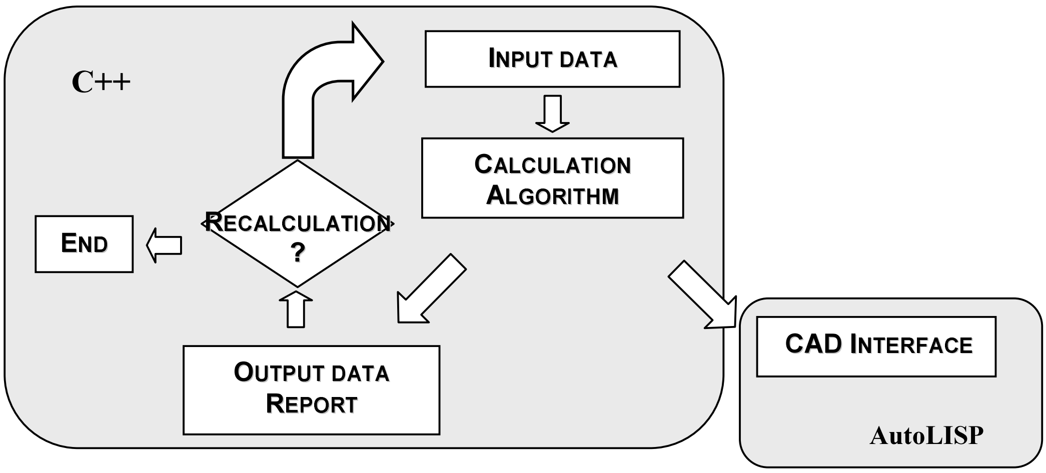

FIGURA 3 – Typical flow of information inside a module.

Modelling and implementing the database

Once the calculation modules are designed and implemented in a standalone fashion, the second stage of this project is the modelling and implementation of the database which is going to promote the integration of the system. The correct specification of a Database Management System System brings several advantages: contributes to data formats standardization; enforces the use of electronic documentation; allows flexible alterations in the structure of the system; prompts consistent and up-to-date data; centralizes the data and hence reduces the time of consultation and applications development. The choice was Borland´s INTERBASE, since it connects perfectly with the C++ platform used in the modules. This system presents a relational structure, in which the machine elements are associated to interrelated tables. The relational structure brings another bunch of advantages such as flexibility in the modelling, ease of data communication and implementation, well stablished database services and use of SQL language.

The database phase was divided into three stages:

- Design: during this stage, the requirements and needs of the system were determined. After, the data were modelled through the conceptual model ME-R (Model Entity-Relationship). The types of entities, its relationships and attributes were then defined and obtained. These information allows an outline of the database. They provide an insight of the problems as well, which is very useful in understanding and determining the design rules to be used inside the CMM.
- Implementation: it consisted of the construction of the structure defined by the ME-R study of the previous stage. Here the application programs were developed, tables were implemented, the construction of the identification keys of the elements (domains) were defined by the position of the elements in the gearbox.
- Tests and maintenance: in this stage the system is tested through an initial load of the tables with a coeherent set of values for the data. The correct operation of the routines, the integrity of the database structure and the consistency of the data are verified. The maintenance stage requires a larger period for execution of tests, adaptations and eventual changes and therefore is still being carried out.

Implementing the calculation methods manager

A manager program is a general purpose software that facilitates the access process to the database for several applications, allowing the user to control the data of the system. The Calculation Methods Manage*r* (CMM) is characterized by the structured treatment given to the modules, since it uses rules and supplies a logical control of the project. This allows the modules to be called in an integrated form.

The CMM program has three main objectives:

- to permit an orderly input of checked, properly formatted data that constitute a different archive for each different project;
- to manage and control the development of the project through predetermined rules. These rules are fired through the Borland database engine, in a very similar manner to the ones used in artificial intelligence;
- to allow the user to access input and/or output data supplied or obtained during the calculations.

The Figure 4 presents the integrated use of the modules and the connections among the CMM components.

A very important step during the design of the CMM was the determinations of the rules. To start with very simple ones have been chosen for ease of implementation sake. More complicated ones can be implemented once it has been proved that the easy ones work properly. Also rules were divide into two categories:
a) existence rules : the ones that checks whether an element or data already exists. This state allows subsequent elements to be calculated. For instance, one has to calculate the gears of the third shaft prior to calculate the shaft itself and/or the respective bearings. The calculation of the gears will feed the database with forces and dimensions that are going to needed during the shaft and/or bearing calculation.
b) consistency rules : two or more data are checked to avoid desobeying an engineering concept. This is the case when one examines whether any diameter of a shaft interferes with any outer diameter of surrounding gears.

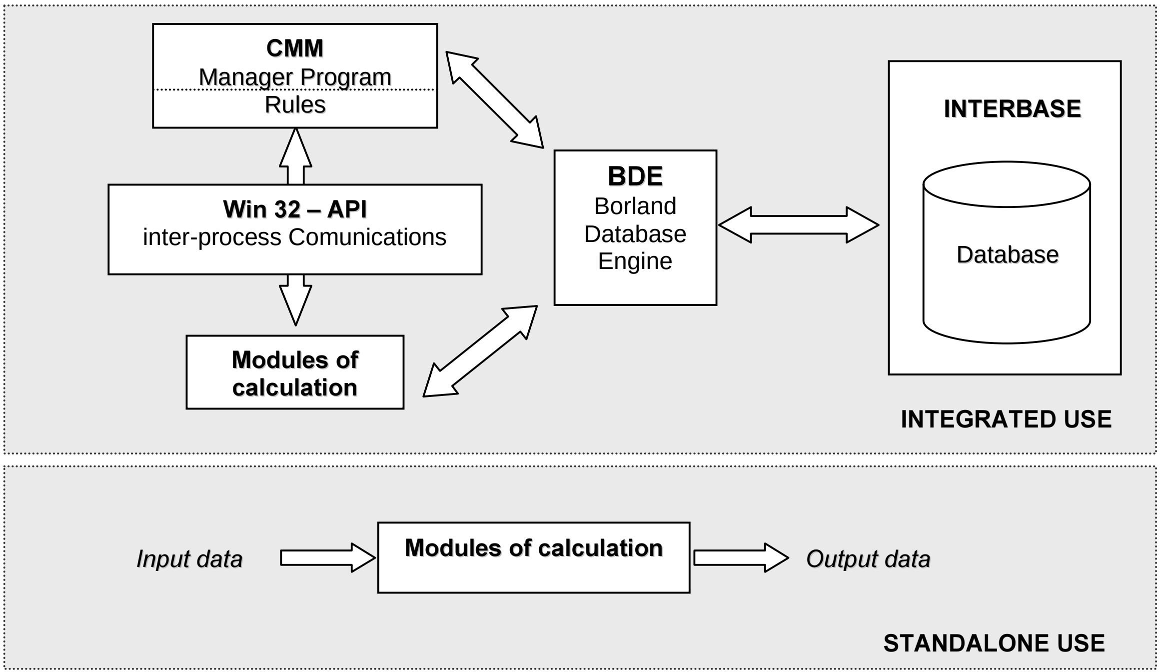

FIGURE 4 – Connections diagram among the CMM components.

When the user is running a module and is not satisfied with the output data, she/he can run it again by changing the input data and this causes no problem whatsoever (see Fig 3). Imagine, on the other hand, that a certain component has been already calculated and its input and output data have been fed into the database and, more than that, these results have been spread into subsequent componenets. If for some reason - and this frequently happens in real life - this component has to be recalculated, the whole chain of machine elements has to be done again. However, this has to be done orderly, otherwise the process will be lost. The entry points for recalculation and their relationship with the logic of the modules are presented in Figure 5 .

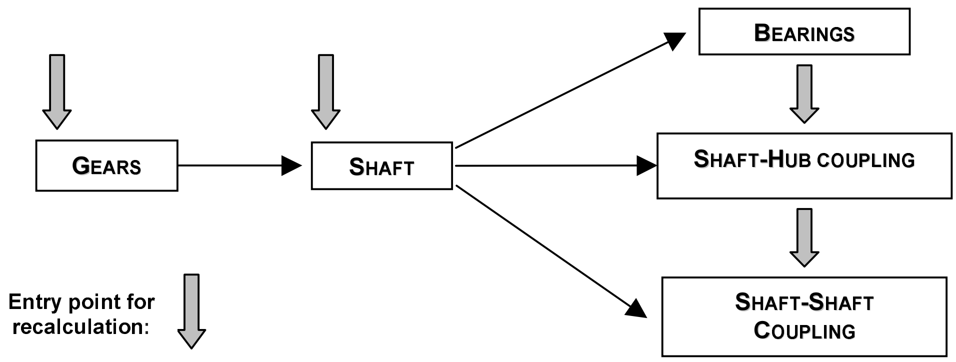

FIGURE 5 - Calculation and recalculation flow.

Recalculation is a special case where the existence rules prove to be very important. In order to deal with such a situation, three states of the modules have been defined: in-process, complete and recalculation. The transition among these three states is shown in Figure 6.

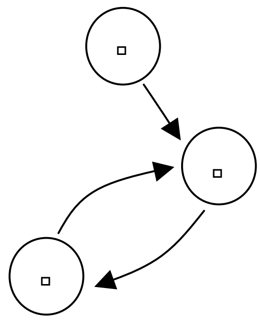

FIGURE 6 - States of transition among processes.

After developing these concepts it is possible to implement rules for recalculation and for authorizing the storage of data into the database. Firstly, the existence of the proper tables and access key are guaranteed. Secondly the module is declared "in-process". After adequate iterations, the user is satisfied with results and before leaving the module declares that it is ready, which implies that the state should be turned to "complete". This fires the authorizationm to store the data into the database.

The re-calculation algorithm for the implementation of first part of this rule is presented below.

```
BEGIN
Get current pair_of_gears
        current pair_of_gears  in_Process
        For each shaft  of current pair_of_gears
                        Get  shaft
                        Shaft in Recalculation
                        Shaft  presents    Calculation_Authorization = 0

           IF  Shaft = "input shaft"
                        Get SHAFT-SHAFT COUPLING of "input shaft"
                        SHAFT-SHAFT COUPLING  of "input shaft" in Recalculation
                        SHAFT-SHAFT COUPLING of "input shaft" presents
                                                    Calculation_Authorization = 0
                        Get SHAFT-HUB COUPLING of "input shaft"
                        SHAFT-HUB COUPLING  of "input shaft" in Recalculation
                        SHAFT-HUB COUPLING of "input shaft" presents
                                                    Calculation_Authorization = 0
          ELSE
           IF  Shaft = "output shaft"
                        Get SHAFT-HUB COUPLING of "output shaft"
                        SHAFT-HUB COUPLING  of "output shaft" in  Recalculation
                        SHAFT-HUB COUPLING of "output shaft" presents
                                                    Calculation_Authorization = 0
           ELSE  Shaft  belongs to intermediate reduction
                        Get SHAFT-HUB COUPLING of Shaft
                        SHAFT-HUB COUPLING  of  Shaft in Recalculation
                        SHAFT-HUB COUPLING of  Shaft presents
                                                    Calculation_Authorization = 0
```

Get SHAFT-HUB COUPLING of Shaft
SHAFT-HUB COUPLING of Shaft *in Recalculation*
SHAFT-HUB COUPLING do Shaft *presents*
Calculation_Authorization = 0

Get Rol RE of Shaft
Left Bearing of Shaft *in Recalculation*
Left Bearing of Shaft *presents* *Calculation_Authorization = 0*

Get Rol RD do Shaft
Right bearing of Shaft *in Recalculation*
Right bearing of Shaft *presents* *Calculation_Authorization = 0*

END
Call module for the calculation of gears
END

The continuation of the implementation of this rule, as well as the implementation of another ones, follow this same pattern. Insofar four similar basic rules have been implemented. More sophisticated ones may be created according to the methodology given above. Also, at this stage, once the database is implemented and stores consistent data, one can develop typical database services by using SQL language. Some of these rules and services are suggested below:

- Check if the oil used during all the bearings calculations is the same one and if it is the same oil used during the calculation of all pairs of gears (rule);
- Estimate the heigh and width of the gearbox by consulting and adequate operation on the relevant dimensions of the components (service) and;
- Prepare a report presenting all types of gears being manufactured inside the company seeking standardtization (service).

Other calculation modules can also be added e.g.: brakes, hydrodynamic bearings, conical gears, chains, etc. Even modules for very specific components such as the one for calculation of the cabin of a lift can be developed. If this is done, the idea of the CMM presented in this paper may be replicated for other products in place of the gearbox. The example of the computer aided calculation of a complete lift is a good one.

Conclusions

The traditional methods for the calculation of machine elements which are already known to be reliable, proved to be very adequate to be transformed into computacional tools .

When developing the graphical user interface, C++ Builder adopted for the coding of the modules, presented very good results in terms of interactivity with the user, communication to the database and portability. The graphic interfaces developed for the modules are friendly and could be used as teaching tools to be used in machine design courses. However excruciating effort has to be used to transform algorithms typically procedural into object oriented codes.

The modules in standalone operation reduced greatly the time as compared to manual calculation, although no precise quantification of this reduction has been carried out.

By implementing the database into the CMM, one gets the centralization and the integration of the data obtained during the calculation stage of individual machine elements. This promotes a simultaneous engineering environment which is very useful in optmizing machine design.

The authors wish to thank CNPq and FAPESP for the financing of this research project.

Bibliography

ALCOFORADO, F., 1997, **Globalização.** São Paulo, Nobel. p.14-30.

CHIUSOLI, R., 1996, **Engenharia simultânea: estudo de casos na indústria brasileira de autopeças.** Dissertação (Mestrado). 147p. São Carlos, SP. Universidade Federal de São Carlos.

CIMData 1997, **Product data management: the definiton**. September. http://www.cimdata.com.

CIMData 1997, **PDM vs EDM. Is there a real difference?** June. http://www.std.com/Newbury/CIMdata/pages/pdmvedm.htm.

DATE, C.J., 1986, **Introdução a sistemas de banco de dados**. 3a.ed. Rio de Janeiro, Campus. p.26-89; p. 76-82; p.93-193.

DHANDE S. G.; GUPTA, A. ,1984, Computer-aided interactive graphical design of multispeed gearboxes. **Journal of mechanisms, transmissions and automation in design.** p.106-163.

FERNANDES, O. C. 1997, **Apostila de elementos de máquinas: introdução ao projeto de engrenagens cilíndricas.** Departamento de Engenharia Mecânica, EESC, São Carlos.

GASCOIGNE, B.,1998, **PDM: the essential technology for concurrent engineering**. p.1-9. http://www.pdmic.com/articles/artetfce.html.

LIRANI, J., 1995, **Apostila de elementos de máquinas.** EESC, São Carlos.

NIEMANN. G., 1971, **Elementos de máquinas 1, 2 e 3**. Editora Edgard Blücher. São Paulo.

KORTH, H.F. SILBERSCHATZ, A., 1989, **Sistemas de bancos de dados.** McGrawHill, São Paulo. p.1 - 48; p.50 -115; p.187 - 239.

PRESSMAN, R., 1996, **Engenharia de software.** São Paulo, McGrawHill. p. 3-35;p. 177-229; p. 231-240; p. 276-279; p. 317-352; p. 415-431; p. 521-533; p. 601-631; p. 675-684; p. 694-702.

ROZENFELD, H.; OLIVEIRA. C.B.M.; OMOKAWA, R., 1997, Development of conceitual engineering enviroment. **International Journal of Computer Integrated Manufacturing.**

SILVEIRA, Z. de C., 1999, **Desenvolvimento de um sistema computacional de auxílio ao cálculo e desenho de elementos de máquinas.** São Carlos, 1999. 133p. Dissertação (Mestrado). Escola de Engenharia de São Carlos, Universidade de São Paulo.

SPOTTS, M.F; SHOUP,T.E., 1998, **Design of Machine Elements**. 7.ed. [s.l], Prentice Hall.