Małgorzata ROGALA

Institute of Combustion Engines and Foundations of Machine Construction
Poznań Technical University, Poznań, Poland

USING A BLACKBOARD ARCHITECTURE FOR MANAGING A CAD SYSTEM DURING THE DESIGN OF SPRINGS

Summary. In this work the usefulness of blackboard architecture in CAD systems development is shown. Types of management of CAD systems actions are classified. The general structure of CAD software of blackboard architecture is characterized. It is shown, by the example of disc springs designing, that blackboard architecture application will make it possible to obtain the demanded usable features of CAD systems.

1. Introduction

The ever greater development of CAD systems and their greater and greater complexity bring about the increase of the role of man - computer dialogue in these systems. It is a pivotal factor of the user's evaluation of the system. The development of the dialogue tends towards augmenting its "intelligence" and flexibility. This requires an ever more sophisticated software handling dialogue.

Tools are available for the construction of dialogue forms software, e. g. of all menus and forms. However they are not suitable for the development of software of dialogue managing because they can be used only for describing a syntax of man - computer communication. Furthermore, the dialogue managing module checks if a user's utterance is correct, transforms it into a deep structure and sends to the supervising module. The dialogue module also converts and transmits to the user the results of operations performed by the system .

In many cases dialogue development requires working out a new inner organization of CAD system and new methods of their software development.

A new organization of the CAD software based on the idea of blackboard architecture is proposed. Blackboard architecture has so far been used in expert systems [3] and task planning systems [2]. This architecture is presented on the example of the CAD system for disc springs.

The starting point for this paper is characterization of the internal organization of CAD systems presented in p. 2. Systems are distinguished which work according to a fixed graph which represents the course of task solving and systems of an undetermined sequence of operations performed during task solving. On the example of implemented software aiding disc spring design, drawbacks of the former of organizations mentioned are pointed out. It is emphasized that the required features of the dialogue often demand the application of the second organization of an undetermined course of solving a design task. Blackboard architecture is characterized as an example of such an organization of a system in p. 3. of the paper. CAD system structure of blackboard architecture is presented . It is shown on the example of spring design that this approach will make it possible to obtain usable features of the software that are of essential importance.

2. Management of CAD systems work

The sophisticated man - computer dialogue and complex functions of the CAD system always lead to the increase of the number of modules dedicated to performing different tasks in the system. Thus, an essential role is played by systematic methods of creating its software, so that its developing by team of specialists of different fields, e. g. data base and computer graphics is possible.

CAD software design usually consists of the following stages:

- distinguishing of types of operations performed in the process of design task solving; among these operations there are dialogue ones that require the user's direct involvement and non - dialogue ones executed totally by the CAD system;
- defining of the necessary information which the CAD system should contain in order to aid task solving and classifying this information; the distinguished classes of information from now on will be called objects;
- selecting of the data structures for the representation of information classified;
- programming of procedures performing operations distinguished;
- determining of the sequence in which operations will be invoked during performing a given task.

A great variety of management methods is a characteristic feature of the CAD software. In the simplest programs, in which there does not exist a managing module, the procedures are invoked one by one. Complex systems use methods of artificial intelligence for performing inferences on which operation should be executed in the next state of the system according to the given strategy of task solving.

Generally speaking, two classes of systems may be distinguished with respect to the way their work is managed. These are:

a) systems of the fixed course of task solving - calls of procedures performing design operations are inserted in the system code; the structure of connections among procedures is a directed graph in which cycles may occur; managing consists in the selection of a subgraph to which belong the initial node and one of the end nodes;
b) systems of an undetermined course of task solving - procedure connections are not fixed; management depends on the values of the system states; the initial and end states are distinguished; the managing module selects upon the current values of states which design operation is to be performed and the activation of what procedures it demands; system states are transformed from the initial to final ones due to the execution of design operation.

Both ways of management of CAD systems work are schematically presented in Fig. 1. As it is shown in Fig. 1. the undetermined course of CAD system work is conducted on two levels. At the superordinated level the managing module performs its specific operations in order to determine which design operation is to be executed in the next step and invokes it. The design operations are placed on the lower, subordinate level and act on the demand of the managing module. The level of management is not practically distinguished in the system of the fixed course of task solving.

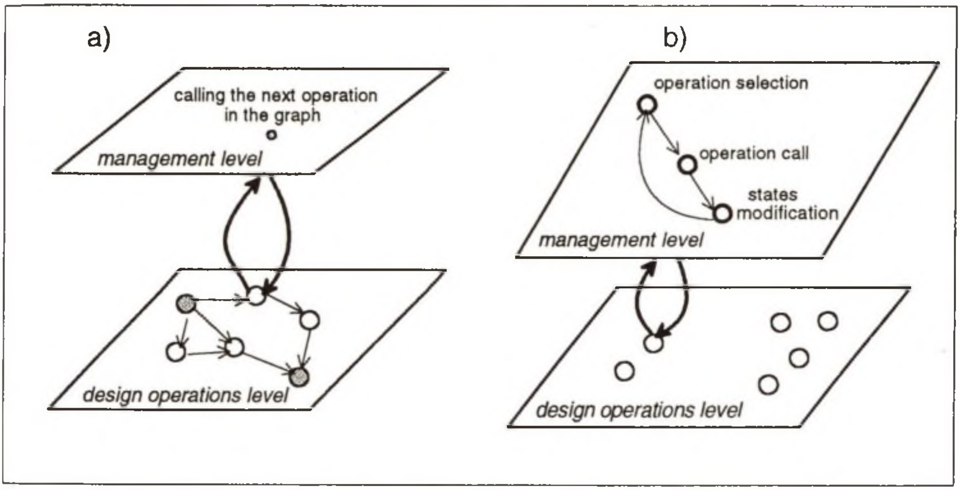

Fig. 1. CAD systems organization: a) of a fixed course of task solving,
b) of an undetermined course of task solving

The fixed course of task solving is a feature of the CAD system dedicated for disc springs [1]. It makes it possible to perform various calculations of these springs. It encompasses checking and constructing calculations at static and dynamic loads. Both single and assembly springs may be designed by its means. It takes advantage of the catalogues contained in the data base concerning different types of springs: standard, extra thin, with ground off edges. All in all the system accepts more then forty, previously determined in the system, variants of design task [1]. The system is used both in industry and for teaching purposes. A number of drawbacks of this system have been found. Below are some more important ones:

- the user is not informed at the start point of all variants of the design task; its variants are dissipated in the structure of the multilevel menu; it makes difficult to understand the system by the user and to create its model;
- the imposed sequence of performing design operations restricts a quick and easy modification of tasks;
- the implicit structure of the software makes it that even small changes in the man - computer interface or in the calculation process require reprogramming in many procedures and changing their connections.

It was decided to redesign the system of disc springs calculations to remove the disadvantages observed. The undetermined system work was assumed in the new version of the system. The blackboard architecture was used in the software development. Its application makes possible:

- to define a design task according to the user's scenario and not imposed by the computer system; to determine dynamically and to modify data which form the design task definition;
- to control the course of task solving without restrictions imposed by the system - i. e. to cancel operations, suspend them to resume them after other activities have been performed; to backtrack.

The blackboard architecture is characterized in the next point.

3. Blackboard architecture in the CAD systems

3.1. Software structure

Dividing knowledge to be represented in the computer system, into chunks that describe homogeneous subjects, is the very idea of blackboard architecture. Every such chunk is independently processed by different modules - sources of knowledge. The fundamental elements of blackboard software are blackboard, sources of knowledge and managing module [2,4].

Blackboard is a complex dynamic data structure describing the current state of computation in the system. Complex data structures (e. g. objects, frames) organized into hierarchical structure of inheritance and membership could be written in the blackboard.

Knowledge sources are independent modules which transform data written in the blackboard structures. Co-operation among sources of knowledge is only possible through the blackboard. Sources check information written into the blackboard and then communicate that they are ready to take up processing (when the data from the blackboard satisfy their conditions of activation). Sources have the knowledge indispensable for task solving. Each source has its own mechanisms of inference and of computation of information in the blackboard.

The managing module controls the work of the whole system. It determines which operation of which knowledge source will be performed next.

Table 1. mentions those objects which will be written in the blackboard and characterizes the contents of knowledge sources.

3.2. Blackboard architecture of the CAD system for disc springs

The new method of a disc spring description was worked out during redesigning of the CAD system for disc springs. Depending on it the data structures for representing disc springs in the CAD system were selected. The main features of disc springs description are as follows:

1. objects describing a disc spring (single or assembled) form a tree structure; a given node of the tree is in relation "is-an-element-of" with its antecedent;

2. different sets of design parameters describe a disc spring according to the mathematical model, constraints and criteria chosen by the user; it means that objects subordinate to the DISC SPRING object have different sets of attributes;
3. task solutions have references to the definition of a task because solutions of many different tasks and information of many tasks in progress are stored in the system;
4. a calculation process is assembled of different procedures according to the selected type of calculations (checking or constructing), the mathematical model or spring construction.

Table 1. Contents of blackboard and knowledge sources in the "blackboard CAD system"

Blackboard	Knowledge source
Knowledge on the designed object	
• definition of an ASSEMBLY object; definitions of subordinate objects; • instances of objects	• typical solutions (inst. of an ASSEMBLY object solutions chosen by the user and placed in the data base); • calculating procedures forming the mathemat. model; • mechanisms of calculating process configuring
Knowledge on the design task	
• definition of a TASK DEFINITION object and subordinate objects; • state of a design task; • history of task solving; • parameters of varianting programs and of optimizing programs	• varianting and optimizing programs, • rules of program selection according to the task conditions ; • typical values of program parameters; • definitions of tasks chosen by the user and placed in the data base
Knowledge on the dialogue	
• state of the dialogue; • characteristics of the user; • history of the dialogue	• dialogue actions; • procedures managing the dialogue techniques; • rules of dialogue technique selection suitable for the user expectations; • description of princip. of communication with a system

It is shown in Tab. 2. that mechanisms of blackboard architecture make it possible to describe disc springs in the demanded way. The features of the disc spring description are mentioned in the first column of the table. The possibilities of the disc spring description in the blackboard CAD system and the CAD system of traditional structure are compared in next two columns. It is possible to describe a disc spring in both types of systems. But a very complex and implicit structure of records or arrays would be created in the system of traditional structure, especially because of quite large (more then twenty) number of classes of objects used for the description of a disc spring and a design task. The possibility of dynamic assembling of calculation process shows the superiority of a blackboard architecture. In the traditional structure of a CAD system a large, implicit network of connections among procedures and compound instructions of their selection would be created. In the blackboard architecture, independent knowledge sources assemble calculation process. The managing module does not supervise this activity. It makes possible to simplify its construction. The similar, to blackboard architecture, possibilities of knowledge description, offer object structures in programming languages like C++ or Turbo Pascal. But they demand different methods of software development and a CAD system created in such a manner is not longer "traditional".

Table 2. The possibilities of knowledge description in the CAD systems: blackboard and traditional

Features of disc spring description	Blackboard architecture	Traditional structure
• Data structures form an "is-an-element" hierarchy	Object structures could form such a hierarchy	Records of records or arrays of arrays
• Objects subordinate in the hierarchy are described with different sets of attributes	Possible in object structures	Records with variants
• References to objects of different classes are placed in the description of a given object	Defined as attributes of a given object	Defined as additional fields in records or additional elements in arrays
• Calculation procedures are selected according to the set of attributes and their values	The set of procedures and the sequence of their execution inferred by the knowledge source	Information on calculation procedures placed in data structures; gives a complex management of procedure activation
• Instances of objects are deleted on the user's demand	Performed by primary instruction of object programming	Should be programmed
• Instances of objects are stored in the data base on the user's demand	Performed by the proper knowledge source	Needs the co-operation with the relational data base

The main features of of the process of task solving are determined in the design of the blackboard CAD system for disc springs. They are as follows:

1. the system stores many tasks in progress and resumes them on the user's demand; thus, states related to each task in progress has to be defined in the system;
2. the system makes possible backtracking to the previous stages of task solving, i. e. it reconstructs the previous task states;
3. it is possible to cancel design operation and reconstruct the state previous to the operation call, to suspend the operation to resume it after other activities have been performed.

In Tab. 3. the possibilities of a blackboard CAD system and a traditional one to manage the task solving process possessing the mentioned above features are compared. It is shown in this table that a CAD system which permits the user to control the process of task solving without system restrictions, practically demands undetermined management of its work. Attempts of control description by means of the fixed graph lead to the combinatorial explosion, to defining complex structure of such a great size that the system can not handle it effectively.

Blackboard architecture allows to embed a new category of knowledge i. e. information on the variants of design task solving into the CAD system for disc springs. In the traditional system, the tasks descriptions were dissipated in the multilevel menu structure and the control structure. Furthermore, blackboard architecture makes possible a new organization of user - computer co-operation. The course of co-operation with the traditional system and the blackboard one are presented in Fig. 2. The main difference between these two forms of co-operation is that in the traditional system the user does not know and can not foresee what the next action of the system will be, but in the blackboard system he is provided with information on all variants of design tasks at the starting point of co-operation. From the graph of variants the user selects a subgraph representing one variant or some of them he wants to calculate. He defines the data specific for a current task variant indicating (in any order) the node in the subgraph and entering the values of design parameters connected with it. The help is available to the user at any moment of co-operation. He can obtain information on design parameters, their typical values, examples of task definition. The new idea in the user - computer co-operation is presenting to the user the plan of task solving and indicating it the current state of calculations at the plan.

Table 3. Possibilities of description of management structures in CAD systems: blackboard and traditional

Features of disc spring design process	Blackboard architecture	Traditional software structure
• Using of an object which describe the design task state	Defining of an object class which describe states	Defining of states as records or arrays
• Reconstructing of the previous states of design task	Reading of object instances from the knowledge source data base	Loading of records or arrays stored in the data base
• Canceling and suspending of operations	Searching for the identifier of the instance of the STATE object in the history of task solving and reading the instance from the data base	It is practically impossible to describe the control graph expressing possibilities of canceling and suspending of operations

4. Conclusions

The blackboard architecture application brings profits to the user of a CAD system as well as to the software developer. The main advantages for the user are the unconstrained order of operation execution and significant assistance of the system in making design decisions. The advantages for the programmer consist in imposing systematic way of software development. Blackboard architecture makes possible to develop software hierarchically, by teams of different specialists. Separating independent knowledge sources in the blackboard architecture permits to add to the software new categories of information and methods of its processing without intefering with other modules of the system. In this way the software can be expanded and its can be usability increased. It has to be noticed that the blackboard architecture is a new idea and there are few computer systems using it. That is why, the construtcion of the prototype software of the CAD system demands solving of many programming problems.

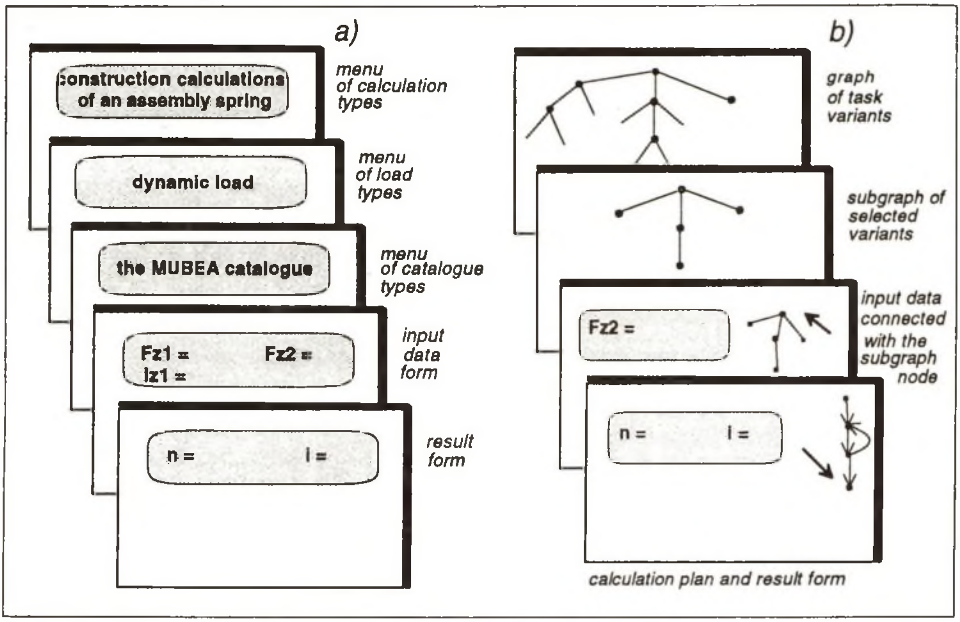

Fig. 2. The course of user - computer co-operation in the CAD systems: a) the traditional one, b) the blackboard one

REFERENCES

[1] Branowski B., Józefowicz W., Rogala M.: Interactive modelling of calculating - optimising tasks, Proceedings of the IX[th] Conference "CAD, CAM, CIM", Warsaw Technical University, Warsaw 1993.

[2] Hayes - Roth B.: A Blackboard Architecture for Control, Artificial Intelligence vol. 26 1985, pp.251 - 321.

[3] Michalski A.: A Language for Description of Blackboard Architecture of an Expert System, Proceeding of the Conference on Expert Systems, Wrocław Technical University, Wrocław 1993, pp. 406 - 414, (in Polish)

[4] Silverman B., Chang J., Feggos K.: Blackboard System Generator (BSG): An Alternative Distributed Problem - Solving Paradigm, IEEE Transactions on Systems, Man and Cybernetics, vol. 19 No. 2 1989, pp 334 - 355.

Revised by: Jan Kaźmierczak