19
CALinCAD: Computer-Aided Learning in CAAD

Rob van Zutphen

Department of Architecture and Building Science
Eindhoven University of Technology

Calibre, Eindhoven University of Technology, ABACUS, University of Strathclyde, and LEMA, University of Liege, investigate whether it is possible to teach the architectural design process, using different CAAD techniques in a more integrated way. The research is funded by the EC in the European Comet project.

The architectural design process is a complex, multidisciplinary process in which we have to deal with the many regulations, standards, and procedures of different participants and companies. There are also constraints in time and costs. This complexity makes teaching CAAD difficult. The design process itself is still not completely understood. Several models of the design process have been developed; most of them, including the "Analysis-Synthesis-Evaluation" model by Mayer that we used in our work, are based on or influenced by the model developed by Asimov in his "Introduction to Design."

Partitioning is often used to cope with problems of such large scale. But when we focus the attention of students on one aspect of the design problem at a time, as is done in many schools of architecture and universities, the highly complex, holistic nature of the design seems to get scant attention.

Our aim is to bring the different disciplines together into a workable whole, using the powerful and flexible possibilities of modern computer aids. The first goal will be to develop an open framework with only a few design constraints. Then new parts will be attached, making the model ever more mature and realistic. The openness of the model allows this attachment to be made by people from other departments or other universities and schools of architecture.
CALinCAD Project
CALinCAD stands for Computer-Aided Learning in Computer-Aided Design. It is a computer-aided learning system intended to inform and excite students in schools of architecture and those in architectural practice about computer-aided architectural design. Our philosophy is that the urgent need for education and training in CAAD is best met by deploying it. Our goal is not to write a commercial program but to develop courseware for education and training.

CALinCAD involves integrating CAAD, CAL, and system design and development into one system (figure 1). This demands knowledge not only about CAAD, but also about teaching methods and software engineering. All three universities working on this project have a broad range of experience in using computer technology in the building environment and in architectural design. Contacts with educationalists within Eindhoven University provide the working group with the necessary information in the area of education.

Computer-Aided Learning
Computer systems can serve not only as sophisticated tools in architectural design, but also as powerful learning aids. Some important advantages to using computer systems in education are:

- Students can work at their own pace, selecting their own learning path through the program.
- Computers are good for students' confidence (computers don't laugh when errors are made).
- Computers are time efficient for teachers.
- In general computers offer all the advantages of the individual teaching style.

Figure 1 CALinCAD
It is possible in CALinCAD to record the student's actions, resulting in a history of the learning process. This can be used to improve the courseware, to assess the student, or just as a demonstration (playback).

There are five major types of computer-based instruction programs. They are: Tutorials, Drills, Simulations, Games, and Tests

Simulation of the design suited our needs best (figure 2). Simulations typically have three major advantages over the other instruction types: they enhance motivation, have better transfer of learning, and are more efficient. In general a simulation is a simplified imitation of the real world, in our case the complex design. On a highly interactive base students learn to solve design problems in a integrative way, to make evaluations, and to make appropriate decisions. They learn about architectural procedures, come to understand and to use the characteristic computer aids like drawing, calculating, data-management. The idea is that students are not only highly motivated by simulations, but also learn by interacting with them in a manner similar to the way they would react in real situations.

User Interface

Whether the computer is used as a tool in architectural design or as a tool in teaching, the success or failure of the program depends to a great extent on the design and implementation of the user interface. If, for example, you don't get the attention of the student, he/she will not be motivated, resulting in an ineffective education.
Well-designed screen layouts, fast responses, graphics, and a wide range of help functions are all involved in designing a good user interface. In general there are four major components: the human factor, the task to be performed, the environment, and the machine. Much research is still needed in this area.

**Software Engineering**

The program CALinCAD as included in this paper is a prototype used to gather information about how students and teachers respond to this way of teaching design. We are using SDM, System Development Method, which has been developed by some large companies and software houses in the Netherlands. SDM describes the process of software development from the first idea through maintenance, dividing it into six phases. They are:

- Definition study (GOAL)
- Functional specifications (WHAT)
- Technical specifications (HOW/WITH)
- Implementation
- Testing
- Distribution and maintenance

At present, the project is at the end of the second phase. We developed a prototype version of the computer program and used it in our curriculum to define the different functional specifications.

SDM has in fact some similarities to the architectural design process, which is not surprising if you look at both processes as being a production line with a final product, e.g. a building or a computer program. Both processes are partitioned to make them controllable and workable. The main difference is the higher level of uncertainty in architectural design.

**CAAD**

As mentioned above, we used the "Analysis-Synthesis-Evaluation" model as a suitable theoretical base for CAAD. First, a search for information relevant to solve the design problem takes place. Using this information, a design is created. During the evaluation phase the design is tested against the design parameters.

If the design does not fulfill these parameters, the designer must decide whether to accept it or to adjust the parameters.

Each time we go through the phases of analysis, synthesis, and evaluation, the design gets more detailed. The process goes from draft design to preliminary design to final design. As a tool for the designer, CAD techniques can be used in a responsible way only after the design process is very well described in a more formal matter. The decomposition of design can lead to several distinct, defined processes in which the mutual relations and information flows are set. The descriptions of the
subprocesses, their relations, and information together comprise a so-called information system.

The Computer Program
To get a clear understanding of the way computer systems can be used to support decision-making in design, we first developed a prototype of the computer program CALinCAD (figure 3). We used this prototype to define the functional specifications of computer-aided learning in CAAD. The prototype has a small database and limited constraints.

The program consists of three modules (figure 4). The most important one is the Design module. In this module the student has to design a single-room school building. There are some design objectives, such as the required floor area, the type of wall elements to use, the building and running costs, the annual energy consumption, and the daylight factor.

The second part is the Application module, which is divided into several submodules that each focus on a particular design objective. Implemented on the PC are the submodules "heat transfer" and "thermal comfort." Other submodules, "layout" and "structure," run on other computer systems.

The third module of the prototype, the Protocol module, ought to give a first introduction to several parts of computer systems and CAD techniques (D for drawing). Lack of time made it necessary to skip most of the planned submodules. Only a small paint package and a part of the visualisation methods have been implemented. We did, however, get the required feedback through the use of other existing software.

![Figure 3 Concept of CALinCAD](image-url)
The Design Module

The student first sizes the room and then specifies the four walls (figure 5). He/she can choose from a limited number of prefab elements of different size and type. These are solid elements, window elements, and door elements, each with their specific features such as cost and U-value. There are also a number of general features—such as energy cost per kWh, the interest, and the inside and outside temperatures—stored in the database.

As the design is finished, the computer calculates all the required results for the different targets which are then available for evaluation by the designer (figures 6, 7). Each design target is characterized by its required input data, a calculation method, and the output data. The output data from one process can be the input for one or more other processes. For instance, the area of glass is an input parameter necessary not only for calculation of the daylight factor but also of the cost and the energy needs. Globally this means that when a certain parameter such as the area of glass is changed (figure 8), other results will also change because of the relationship between the different parameters. The CALinCAD program offers the student the possibility of exploring quickly and simply the consequences of the action he or she takes by changing the different parameters. In this way he/she can concentrate on the design while the computer takes care of the calculations. The design process can be seen as a decision-making process in which modern computer techniques can be of great help and support, though the final decision will always be made by the designer.
Figure 5 Design of walls, using prefab elements

Figure 6 Example of calculation results-Energy

Figure 7 Example of calculation results-Costs
The Application Module
The Application module is divided into submodules that give the designer a closer view of the different aspects of building design. Each module deals with a single aspect, giving the student an opportunity to learn interactively the sensitivity for the different parameters involved. For instance, he/she can learn about the heat transmission through a wall element whose construction he/she designed by first defining the number of layers and then selecting from different basic materials such as concrete, bricks, or wood (figure 9).

The computer calculates the amount of energy transmitted through the wall. By changing the thickness of, for instance, the insulation material, the user gets a good feeling for the relations between the energy transmission and the type of material. Other factors that influence the calculation of the heat transfer, such as inside and outside temperature, have a default value that can always be changed. The advantage of using default values is the ability to keep the necessary input for the calculation to a minimum. In this way results for evaluation are quick at hand. The use of default values is used not only in the application modules but throughout the whole program.

Other examples of submodules are: a module for a beam built from different materials, which is tested on its stiffness and strength with different forces, and a module in which we calculate the degree of use of space in, for instance, a bathroom with different sanitary appliances. The placement, size, and number of appliances are chosen by the user. In another submodule students explore the thermal comfort (PMV-Predicted Mean Vote) within a room (Figure 10). The degree of thermal comfort depends on six parameters: the room temperature, the mean radiant temperature, the air velocity, the humidity, the human activity (sitting, walking, learning, etc.), and the clothes the occupants wear. Every parameter can be changed. The mean radiant temperature depends on the construction of the walls. It is important to realize that first- and second-year students do not know each value for the different parameters. As much as possible, we let students enter parameters by name and not by value. For example, the thermal comfort calculation requires the thermal clothing resistance. It must be given in CLO units. The student does not need to know about CLO values, but can just select the menu "clothing" and make a choice from options like "Summer," "Winter," and "Business clothes."

In the Design module we have used the computer system mainly for decision-supported designing, but in the application module the computer gives the students information about the design aspects. The aim is that each design parameter in the Design module can be explained or explored in the Application module.
Figure 8 After user changes design parameters, the calculation results are immediately available for evaluation.

Figure 9 An example of an Application submodule. Here students explore the thermal transmission through a multilayer wall.

Figure 10 Another example of an Application submodule. Here the calculation of the thermal comfort.
Future Developments
Students experienced the single-room building in this prototype as one of the greatest limitations of the program. The project team will use a multiroom, single-story building model in the new program. Another limitation was the constrained set of wall elements. In the future there will be not only prefab construction elements but also other common construction methods for walls as well as for the roof and foundation. There will also be more design targets and parameters, but the team has not yet decided what these will be.

These changes are intended to make the model of the building more realistic. Our aim is to use the computer more and more as an intelligent design assistant and perhaps in the (far?) future as a more intelligent teacher (figure 11).

All the design objectives in the present prototype were of a quantitative nature, which is why the implementation has been done with a third-generation computer language (Pascal) using an algorithmic approach. In reality, however, architectural design involves many qualitative aspects which need a more heuristic approach. Decisions will then be based not only on numbers but also on rules.

The heuristic techniques can also be used for computerized evaluation methods, making it possible, for example, to compare an old design with a new one. The first step toward an intelligent assistant would be to leave the computer to do a first evaluation and let it make a suggestion for improving the design. Thinking of today’s chess computers will give you an idea of what we have in mind.

Although the project team is still in the second phase (SDM), research has already been done about how to program the new functional specifications. Object-oriented programming seems to be very promising (SmallTalk or C++). We are also looking at the possibility of using existing software like Autocad in a multitasking environment (SUN 386i).