IMPLICATIONS OF EXPERT SYSTEMS, DATA MANAGEMENT AND DATA COMMUNICATION FOR ARCHITECTURAL EDUCATION

Ir. Geert T.A Smeltzer
Eindhoven University of Technology, CALIBRE

SUMMARY

The availability of computersystems for processing, managing and communicating data and expertise, does not mean that the results of these processes, will improve automatically. It also shows to much optimism if you expect that the use of process management and communication tools will enlarge the possibilities of the processes themselves.

First of all we will have to face the limitations and we will have to accept them, at the cost of the traditional or ordinary architectural education. Mainly we will have to settle for less and worse design results in the beginning of the use of the new tools. In later stages, however, we will have to be able to deal with higher design quality for more aspects at the same time, in stead of the average quality, for mainly only the visual aspect.

To meet the limitations of tools like computersystems, we will have to limit and structure the data and expertise, until we will have reached an absolute minimum quantity and a maximum quality of data and expertise. In fact we should strive, at first, for an “implosion” of data and expertise. Then, by adding more and more expertise and necessary data of the same quality, we can control an information explosion.

PREFACE

After the era of the Industrial Revolution, the Electronic Revolution gave birth to Electronic Data Processing (EDP). At first the contribution of EDP was aimed at an irnprovement of the quality of data (data processing). At present the contribution of EDP is mainly an increase of quantities of data (data production). This conference uses even the reference to an Information Explosion.

From the architectural point of view, the demand for multi criteria design evaluations is getting stronger. Multi criteria evaluation asks for more, more specific and related data and expertise about a design. A more intensive iteration between evaluation and improvement asks for data and expertise on more different levels of detail.

This data and expertise should, as much as possible, be stored in general available data and rules bases. Specific project data may still be stored in project data and rules bases.

In fact we can determine that multi criteria evaluation of designs is the fuse of the EDP bomb, which can cause an information explosion.

The title of this paper, translated to a question that has to be answered, could have been:

"How should we deal with data management data communication and expert systems to be able to control this Information Explosion"

POSSIBILITIES OF EDP

The possibilities of EDP, concerning data management and data communication, limit themselves to computer support of data (processing), management and communication.

For real automation a lot of expertise about design methods and techniques has still to be developed (processed) manually. Only when this expertise exists explicitly we can update the computer support to automation.

Expert systems can then play a very important role when we think of automatic data (processing), management and communication.

The wish to use the possibilities and benefits of computer supported data management and communication and of expert systems supported automation, together with the wish to control the information explosion, implies that we will have to be able to deal with the limitations of EDP and with the limitations of explicit expertise of design methods and techniques.

IMPLICATIONS FOR ARCHITECTURAL EDUCATION

The implications of the possibilities and limitations of data management and communication and of expert. systems for architectural education are just as versatile as demanding. In short it means that if there will be no structural changes in curricula and of teaching staff, the implications can be disastrous.

What kind of changes do we have to consider?

Multi disciplinary education

The possibility of multi criteria evaluation asks for multi disciplinary education. The architect should be able to interpret the different aspects, like structural environmental cost and visual aspects, of the computers output. He should be able to identify a week spot in his design and he should be able to make several proposals for improvement

Teachers should be able to teach the iterative process of evaluating and improving a design. Therefor they should be able to teach things like forward and backward reasoning to help students to predict the expected result of their design decisions and to help them to decide about possible design changes.

A design project, were student can use computer supported multi criteria evaluation, should have a special brief, dealing with all the possible aspects in a complete and consistent way.

Learning curve and expertise ceiling

The limitations of computer support and automation have a delaying influence on the productivity of architectural education

In the old fashion way, a student could improve and enlarge his knowledge very fast in a relative short period of time. On the longer term this tendency curves slowly until the student reaches a ceiling. The height of this ceiling will also be determined by the students talents, but mainly by the general available expertise in architecture. Initiatives to break through this ceiling mostly have little strength and glance off (figure 2.).

The limitations of computer support and automation force education to use less extensive and less complex examples and problems.

Because of the extra time that the use of computersystems cost in the beginning, students will only be able to produce less drawings and lists. The esthetic aspects will also have to settle for less attention in the beginning.

These things cause a learning curve that seems not to be able to get a real lift off. The learning process will be laborious and frustrating. But later on...

Later on the curve will take an unpredictable turn when the possibilities of computersystems can be used properly.

Because of a "steeper" approach of the expertise ceiling a breakthrough will stand a better change (figure 3.).

The figure suggests that the increase of expertise could be endless. Of cause that is difficult to imagine, but why not?

Data models for data management and communication

To build the necessary confidence in multi disciplinary computersystems, these systems should meet a number of conditions. In short they come down to the conditions that the results of multi criteria evaluations must be consistent and that the output to others can always be complete.

Therefor the source of the output and the algorithms should meet minimal the same conditions of completeness and consistency.

To reach this an approved data structure is needed. Data can be structured alphabetically, by example, but that is no real logic.

It shows more sens to structure the data, describing an object like a building, in a kind of an "intelligent geometrical budding model"

The algorithms should be able to extract the relevant data out of the data model and should be able to store the results in the model again.

For the development of such a model it is of very high importance, that the model will not contain to much information. The model builders should try to identify the greatest common measure of the data for the different design disciplines that will use the model.

Except from the statement that a data model should have as little as possible data, it is also of great importance that that data is added to the model as late as possible in the design process. Although editing data is a lot easier, using a computersystem, it is still a waste of time and you will run a great risk to get inconsistencies in your model if you change your data to often.

For the data communication it is also of importance that the model is a kind of greatest common measure of the data formats that are used for the data exchange between different computersystems, like IGES, STEP, DXF.

Education for model and system development

To let the relationship between architecture and computer support and automation be as meaningful and constructive as possible we need very good trained "diplomats".

These diplomats should have the disposal of expertise of methods and techniques that are of importance to architecture and to computing. They do not have to be architects or programmers; rather not even.

Besides their expertise of architecture and computing they should have experience in the actual development of computersystems.

In opposite to the role of the computer user, the role of the diplomat is not available to every one. Only a few students with a clear analytical ability will be capable and needed.

These students should not only be trained in the development of computersystems, they should also be trained in making architectural expertise explicit (knowledge engineering).

Only for these students expert systems are specially important.

However, all student should do a project in which they have to deal with rationalizing, systematizing and or normalizing a more or less specific aspect of architecture. They should then have the opportunity to discover the specific problems that will arise and to discover their talents.

Tools for tools

In a growing number of cases of computing in architecture, users and developers make use of general available and, often, easy to use drawing, calculation and or data base programs.

These programs. however, are not only general available, they are also only general applicable. You can draw a line with them, but, in most cases, it will not be possible to draw a "wall". Another example is the possibility to draw a circle; however, who is going te build circular walls?

In short not everything of a general program is needed and not everything that is needed will be present.

Fortunately, a still growing number of programs offer the possibility to add on to the program in a relatively easy way. This can be done with a composition of commands, so called "macro" commands, or, for more complex features, with specially programmed new subroutines.

Besides making additions to a program it is, in many cases, just as important to be able to leave unimportant things out. The main reason for this is not only an improvement of the performance of your computersystem it also improves the comprehensibility of the program and makes a shorter period for training possible. (figure 4.)

After a training in the use of Electronic Data Processing systems and after applying them in design projects there should be time available, in the curriculum, for a training in the adjustment of suitable programs.

Humanmatic and automatic processes

An architect first makes a sketch of an idea. Then he assigns meanings, measurers and relationships, among other data. After that he, or someone else, can produce a first drawing of his idea.

An architect is forced to determine meanings, measures and relationships as exact as possible before he can start to work on a computer supported design system.

Therefor, humanmatic sketching always precedes computer supported or automatic drawing.

For making sketches a lot of sketch or paint programs exist. Although they are a long way away from usability in multi disciplinary design problems, they mainly have the same problem of availability, when you need them, as other systems. They are never there when you get your most bright ideas.

Because of the fact that you can not automate everything and that you can not automate nothing anymore, it is of very great importance to determine a sensible separation and conversion between humanmatic and automatic processes.

Teaching teachers

It will be obvious that, if you want to take architectural education seriously and you want to teach (computer supported) data (processing) management and communication and expert systems (for automaton in architecture), that you will need professional computing and automation experts. Of course these experts have to be trained also and of course it is difficult to add more teaching staff, but retraining of architectural teachers is not the way to get the expertise that is needed for efficient computing education in house.

We will have to be realistic and we will have to accept the fact that replacing retiring and or superfluous architectural teachers is the best solution in the interest of the new student generations.

Education for practice

The possession of a computersystem is mainly accounted for from the expectation that, by using such a system the costs for the conduct of business and the time needed for projects will decrease.

In architectural education, however, the expectation, that the use of computersystems will improve the quality of designs, is the main reason

Because of here pragmatism practice will keep focussing on productivity aspects of computing. Education will therefor also have to give the necessary attention to productive use of computersysterns besides the use for quality improvement.

Productive use of computersystems can be reached by skins in using these systems. It can, however, also be achieved by using a minimum quantity of data and expertise, of a maximum quality, for the different. processes in a design process.

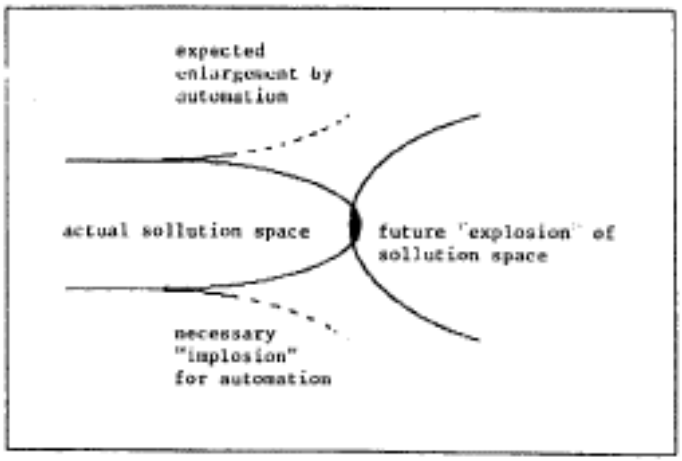

Figure 1.

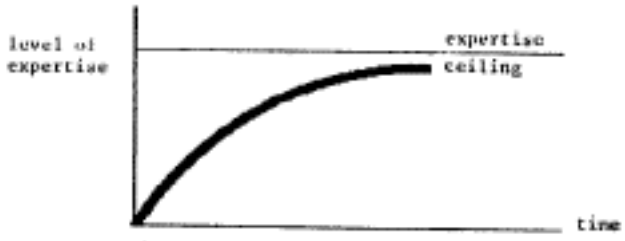

Figure 2: actual learning curve

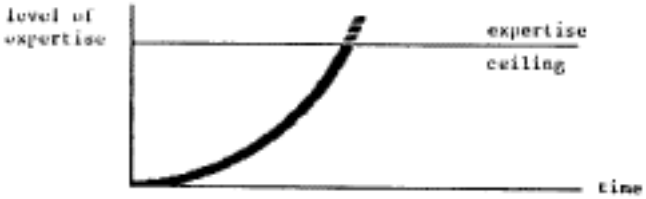

Figure 3: new learning curve

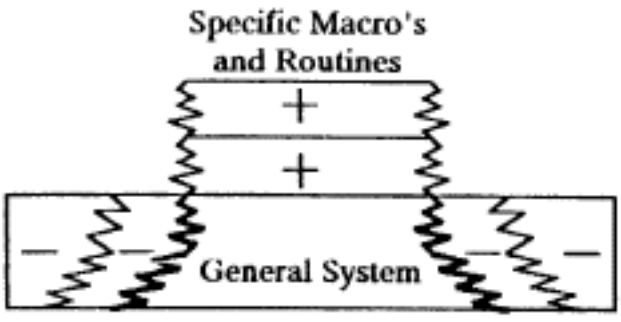

Figure 4