

Approaches to the integration of CAAD education in the electronic era: two value systems

Alexander Koutamanis

In recent years the democratization of information and communication technologies (ICT) has become the greatest influence on the structure of CAAD education. While the content of the CAAD courses simply had to absorb the new technological possibilities, the structure of the courses and in particular their relationship to the rest of the curriculum has become the subject of speculation and experimentation. Integration of CAAD education in an architectural curriculum occurs either by (a) placing emphasis on designing in CAAD courses, or by (b) integrating computing in design courses. Both approaches respond to the democratization of ICT by making design computing widely available and acceptable. Further improvement is possible if the student becomes the carrier of integration. This is based on the long-term amplification of two value systems. The first refers to personal cognition: rather than rewarding a student with the teacher's approval, educational goals should be translated into individual skills and knowledge. The second system addresses the values of the peer group: such groups support learning by comparison to other individuals and emerging communal characteristics, either as a result of competition or for reasons of assimilation.

Keywords: Education, democracy, personal cognition.

Introduction: the new CAAD specialist

Any uncertainty concerning the role and structure of CAAD education is probably symptomatic of a general ambivalence in architectural education with respect to the choice of either the *qualificational* or the *organizational* approach to vocational skill acquisition (Maurice, et al., 1986). The *qualificational* approach entails job description and recruitment on the basis of specific vocational qualifications. The *organizational* approach implies that vocational skills are taught on the job. These approaches correspond to two options in academic education:

1. Teach vocationally relevant skills in response to requirements from practice but also taking into account future developments. In this case, practice has the tendency to adapt to these skills. This results into a relative uniformity and a higher mobility in practice. One consequence of that is that the performance of the skilled academic does not vary significantly in different contexts (Müller and Shavit, 1998).
2. Leave the learning of such skills to practice. Therefore, skills tend to be firm-specific. This has a negative influence on mobility in practice: change of working environment is counterbalanced by sometimes lengthy,

arduous and expensive training and acclimatization.

To a certain degree, the ambivalence concerning the qualificational and organizational approaches is due to historical and cultural reasons. For example, Germany and The Netherlands are closer to the qualificational approach. The educational system of these countries makes use of specific training (including apprenticeship) in the framework of general education in order to produce a high degree of specialization. The organizational approach is characteristic of e.g. France and the U.S.A., even though French education is far less general than American (Müller and Shavit, 1998).

Another reason for ambivalence is that occupation-specific vocational education results into few transferable skills. Graduates are forced to focus on just the specific occupations for which they have been trained. Employers may appreciate this, as it saves them expensive on-the-job training. A prerequisite is that vocational specificity meets their own requirements. In an occupation as individualistic as architecture, it is often assumed that each practice has an own, at least partially unique *modus operandi*. Consequently, specific training may be rejected on the grounds that e.g. a firm uses a different drafting system to the one employed in training.

Yet another reason for ambivalence is that architectural schools appear uncertain as to the character of the education they want to provide. In the area of CAAD most schools concentrate on what is required in practice today, i.e. CAAD literacy and modelling / visualization skills. At the same time, aspirations go beyond the function of assistant architect associated with these skills. For instance, virtual environments are seen as an extension of the architectural domain. Therefore, many schools are proud if they can extend their curriculum to or if their graduates find employment in high-profile areas such as digital animation or interface design. Equally frequent is acknowledgement and accommodation of emerging niches in the architectural domain, which

sooner than later will become part of the core of design and construction automation. Digital document management, advanced visualization, rapid prototyping, communication and collaborative work require a different, advanced form of CAAD training oriented more towards the methodical component of CAAD.

The initial appeal of graduates versed in CAAD to prospective employers lies in their drawing and modelling skills. Design documentation and presentation is predominantly electronic because of its sheer modernity, but also for reasons of efficiency like the compactness and transformability of digital drawings. Computer visualization, from rendered still images to interactive walkthroughs, is also commonplace. Many of these are combined into multimedia and Web presentations, sometimes geared to more complex forms of communication. Equally important for specialized offices are skills in other forms of information processing, e.g. for project and facility management, especially if they can be linked to the visual electronic documentation. This means that the new CAAD specialist normally devotes most of his working time making drawings, models and calculations on the computer. Graduates often see this as part of the training they require in order to be integrated in practice. And when it extends beyond the training stage, they may even accept it as a logical consequence of computerization, namely that thanks to the efficiency of the computer the designer can almost dispense with the services of supporting staff (Mitossi and Koutamanis, 1996).

There is, however, another role a new CAAD specialist may assume, that of initiator and coordinator of ICT integration in design projects. This leads to more crucial tasks than drawing and modelling, such as testing and choosing new systems, coordination of computing activities and sometimes even developing new instruments ranging from databases of components to programs which facilitate or automate specific procedures. Unfortunately, there are several obstacles to fulfilling this role. The first is the graduate's limited knowledge and experience of how

architecture works in practice. Another problem relates to the lack of knowledge of the history and theory of CAAD. Quite often what fascinates practice is intriguing ideas that have been attacked and dismissed in CAAD research (Maver, 1995). The graduate's understanding of these is often sketchy and cannot provide persuasive arguments for abandoning these ideas or for adopting better alternatives. Finally, another obstacle is that academic education can be detached from the real-life problems of architecture and may provide an incomplete picture of the relevance of computational methods and techniques.

Integration of CAAD in the architectural curriculum

The turn to the methodical component of CAAD also relates to the democratization of ICT. The proliferation of affordable powerful computers and the popularization of computer applications have had two main effects. The first is the explosion in the number of available techniques which CAAD courses must absorb. This, however, also involves expansion to a much larger spectrum of architectural subjects. Despite its traditional generalist attitude, it is doubtful whether CAAD is capable of covering adequately so many subjects. The second effect is that techniques previously used only by a computing elite have become accessible to most architectural specializations. As a result, there is a growth of computing knowledge outside and independently from CAAD. The consequence of both effects is that CAAD can no longer function in isolation. In recent years, the structure of CAAD courses and in particular their relationship to the rest of the architectural curriculum has become the subject of endless speculation and wide experimentation, while retaining its three basic learning areas:

- *Computer literacy.* This is a prerequisite to and not an integral part of CAAD. It is currently necessary because large numbers of

university students do not have prior practical experience in computing. This necessity will disappear in a number of years, as future students will have a basic knowledge of computing from school and home. Already a fair level of computer literacy is achieved through educational computer applications quite distinct from CAAD, from text processing to electronic library searches.

- *CAAD literacy* is the next step to basic acquaintance with the computer and has three main aspects:
 - *Drawing and modelling with the computer:* Learning how to employ the computer for the production of visual design representations is commonly seen as the core of CAAD teaching. The grammatical and syntactic correlation of such representations with designing has been the most consistent feature of CAAD.
 - *Advanced technologies:* Despite the democratization of ICT, there are still many computational technologies that have yet to find their way into practice, mostly for reasons of cost and availability. CAAD teaching often pays attention to such technologies, normally in the framework of advanced courses.
 - *History and theory of computational design:* Acquaintance with the development and the underlying principles of current techniques and approaches used to represent a major part of CAAD teaching. In recent years there seems to be less time for such matters, probably due to the rapid proliferation of new technologies which steer attention to the future rather than the past.
- *Integration of design and computing:* A significant change in architectural education is the lessening reluctance to integrate computing in design teaching. The

proliferation of computers in practice has led to a growing presence of the computer in the studio and design courses. At the same time, CAAD courses have been displacing attention from technical subjects to the theory and practice of architectural design. Drawing and modelling techniques are used to rationalize intuitive forms or, reversely, to de-rationalize architectural form by means of e.g. geometric transformations, with analogue and digital representations used interchangeably (Burry, 1999). These combined representations are frequently augmented with analysis and conceptualisation in explorations of how design issues can build on computational technologies (Brady, 1999, Wood and Chambers, 1999). Communication and collaboration between geographically dispersed design teams through the Internet is another subject still primarily linked to CAAD (Kosco, et al., 1999).

Integration in design courses leads to two distinct possibilities concerning the role of CAAD teachers (Koutamanis, 1996). The first is that CAAD specialists become the computer technicians of architectural education, focusing on technology rather than design. Design teachers provide the problem and the methods for its resolution and CAAD specialists guide the student in the use of suitable techniques. This occurs mostly at the very basic level of drawing and modelling, in a way similar to the current position of CAAD in practice.

The second possibility is that CAAD teachers become design teachers who give either (a) CAAD courses that place emphasis on designing, or (b) design courses that involve and integrate computing. The first option has been extensively explored in traditional CAAD teaching, with variable success. The generally prescriptive or technical character of CAAD courses, as well as the brevity of the average CAAD curriculum are important limitations which restrict such courses to an episodic structure

characterized by the superimposition of design issues on a collection of computational techniques (Koutamanis, 1999). The second type of courses offers more time and scope but frequently results into the suppression of the methodical dimension of CAAD, as it may conflict with conventional design approaches.

Integration and the student

Both types of courses represent a step forward in that they respond to the democratization of ICT by making design computing similarly widely available and acceptable. However, only highly focused courses can claim a contribution to the improvement of design quality and performance. Further improvement is possible by shifting from institutional integration of teaching areas to the integration of the subject matter. The essential difference lies in that the student rather than the teacher becomes the carrier of integration. The way to achieve this is by the long-term amplification of two value systems that complement and partially replace the apprenticeship model of architectural teaching.

The first value system refers to personal cognition: rather than rewarding a student with the teacher's approval, we should create a framework that translates educational goals to individual skills and knowledge. In this sense, a successful learning environment is one that supports recognition of relevance and adequacy for both computer handling skills and design decisions. Such an environment facilitates learning not by instruction but through selection. Success of a decision or action should reinforce the cognitive or even movement patterns that are active when the decision was taken or the action performed, i.e. by an *ex post facto* selection from an existing repertoire (Edelman, 1992).

Selective systems for learning in areas such as CAAD probably represent a major educational activity of the future. As successive generations of students start training with increasingly higher levels of computer literacy and affinity with ICT forms of

interaction, explicit instructions of the type traditionally used in skill acquisition and in prescriptive design thinking are incapable of addressing the subtleties of various operations. The development of environments that stimulate extensive yet targeted exploration of possibilities and support identification of success will provide the means for an evolution of CAAD into a truly integral part of designing.

The second value system is based on making CAAD methods and techniques part of the target group's common properties. It has been recently proposed that a child's peer group exerts a primary influence on the child's personality development, often more important than parental nurture (Harris, 1998). The arguments for the peer thesis also bear on CAAD education. This becomes obvious when we substitute parents with teachers, children with students, personality development with learning and the nurturing of offspring with the apprenticeship model of design education:

- Students tend to adopt the computer jargon popular among their fellow students rather than the "correct" terminology put forward by the teachers, similarly to children who end up with the language and accent of their peers rather than these of their parents.
- Subjects not covered by the teachers but popular with students are picked up by younger students, in the same way that children learn from their slightly elder peers.
- Students switch to different justifications and terminologies for the presentation and defence of the same design, depending on the teacher. Similarly to code-switching in bilingual children, students temporarily adopt the teacher's viewpoint, unless the teacher fails to fit in.

Transfer of such ideas to integrated CAAD education suggests that in order to make various elements of CAAD part and parcel of distinct architectural specializations, we should stimulate

independent development of these elements in their new contexts among students rather than teachers. Experience teaches that students frequently learn despite their teachers. Conflicts between what is taught and what actually interests the students form no obstacle to the integrated development of CAAD in an application area. Students may acquiesce to the teachers' power but only temporarily, i.e. as long as they depend on the teachers for grades, acceptance etc. Opportunities to follow their own preferences are generally available and eagerly followed in academic education. An additional incentive is the possibility to compete with and outperform their teachers.

Unobtrusive dissemination of CAAD to other areas relies on four main factors:

1. *The democratization of information and communication technologies:* the growing popularization, affordability and accessibility of computing run independently from the development of CAAD. As a matter of fact, it is increasingly obvious that CAAD has been overtaken in terms of influence on practice and students.
2. *The correlation between the technical and methodical components of CAAD:* integration of technical skills in other areas depends on their appropriate presentation in a framework that matches traditional design media, representations and approaches. A coherent CAAD theory with grounding in design practice is essential for the understanding of possibilities and limitations in CAAD instrumentation.
3. *The compatibility of CAAD with the solutions to emerging design problems:* while in the past CAAD has derived justifications from architectural history and general design approaches, future acceptance is more dependent on roles for computational methods and techniques in specific design problems. Identification of emerging problems

and investigation of CAAD contributions to their resolution are probably the highest priorities in current CAAD research.

4. *The possibility of transition from CAAD to other specializations*: integration of CAAD in architectural specializations is not feasible in the short term without the transition of CAAD teachers and researchers to specific application areas. This presupposes that a CAAD specialist combines knowledge of design computing and of a concrete architectural specialization.

The difference between unobtrusive integration and a single, integral CAAD specialization is that design computing becomes available in many environments. Different contexts mean different opportunities: activities discouraged in one environment may be tolerated or even encouraged in another. Obviously consistency across these environments supports stability in learning. However, as CAAD is not the central theme of any environment, peer support and competition do not antagonize the purpose of the environment. Instead, the peer group complements the learning environment with additional aspects, including CAAD. The ostensibly peripheral character of these aspects encourages extensive interactive experimentation, which in turn supports learning by selection rather than instruction.

References

- M. Maurice, F. Sellier, and J.-J. Silvestre, *The social foundations of industrial power: a comparison of France and Germany*. (MIT Press, Cambridge, Massachusetts, 1986).
- W. Müller and Y. Shavit, *The institutional embeddedness of the stratification process. A comparative study of qualifications and occupations in thirteen countries*. in: Y. Shavit and W. Müller, ed., *From School to work. A comparative study of educational qualifications*

and occupational destinations. (Clarendon Press, Oxford, 1998).

- V. Mitossi and A. Koutamanis, *Preparing for action: responding to requirements from practice*. in: A. Ekholm, S. Fridqvist, and J. af Klercker, ed., *Education for practice*. (ECAADE, Lund, 1996).
- T. W. Maver, *CAAD's seven deadly sins*. in: M. Tan and R. Teh, ed., *The global design studio*. (Centre for Advanced Studies in Architecture, National University of Singapore, Singapore, 1995).
- M. Burry, *Handcraft and machine metaphysics*. in: H. Neuckermans and B. Geebelen, ed., *Computers in design studio teaching*. (KU Leuven, Leuven, 1999).
- D. A. Brady, *Premise and process: the pedagogical implications of computing in design*. in: H. Neuckermans and B. Geebelen, ed., *Computers in design studio teaching*. (KU Leuven, Leuven, 1999).
- J. B. Wood and T. Chambers, *Value added learning: the CADET experience*. in: H. Neuckermans and B. Geebelen, ed., *Computers in design studio teaching*. (KU Leuven, Leuven, 1999).
- I. Kosco, J. Tucny, and A. Dobson, *Around the table on the Net: Intranet and Internet design studios and teaching*. in: H. Neuckermans and B. Geebelen, ed., *Computers in design studio teaching*. (KU Leuven, Leuven, 1999).
- A. Koutamanis, *CAAD teaching in the electronic era*. in: A. Ekholm, S. Fridqvist, and J. af Klercker, ed., *Education for practice*. (ECAADE, Lund, 1996).
- A. Koutamanis, *Designing with the computer: the influence of design practice and research*. in: H. Neuckermans and B. Geebelen, ed., *Computers in design studio teaching*. (KU Leuven, Leuven, 1999).
- G. M. Edelman, *Bright air, brilliant fire. On the matter of mind*. (Basic Books, New York, 1992).
- J. R. Harris, *The nurture assumption*. (Free Press, New York, 1998).

Alexander Koutamanis
Faculty of Architecture
Delft University of
Technology
Netherlands
a.koutamanis@bk.tudelft.nl