(Learning from Experience)? Promises, Problems and Side-effects of Case-Based Reasoning in Architectural Design

Ann Heylighen and Herman Neuckermans

Learning from experience is the essence of Case-Based Reasoning (CBR). Because architects are said to learn design by experience, CBR seemed to hold great promises for their field, which inspired, in the 1990s, the development of various Case-Based Design (CBD) tools. Learning from the experience of developing and using these tools is the objective of this paper. On the one hand, the original expectations seem far from being accomplished today. Reasons for this limited success can be found at three different levels: the cognitive model underlying CBR, the implementation of this model into concrete CBD tools, and the context in which these tools are to be used. On the other hand, CBR research seems to have caused some interesting side effects, such as an increased interest in creativity and copyright, and a re-discovery of the key role that cases play in architectural design.

1. Introduction

Computer-Aided Architectural Design (CAAD) has gone through many generations and philosophical perspectives. One of them, the focus of this paper, is the application of Case-Based Reasoning (CBR) to design, in short Case-Based Design (CBD). CBR is a relatively young theory and technology within the field of Artificial Intelligence (AI) based on an alternative view of human reasoning. Rather than linking abstract pieces of knowledge (e.g. rules or models), reasoning is seen as remembering one or a small set of concrete instances and basing decisions on comparisons between the new situation and the old instance [1]. This cognitive model was inspired by Roger Schank's Dynamic Memory Theory [2] and in turn inspired AI researchers to model people's raw memory. By someone's raw memory we mean the huge collection of individual experiences that the person has had or has heard about. Unlike the generalised knowledge that can be distilled out of it, raw memory contains these experiences in their original form, including all details that make the experience individual. The main part of a CBR system, then, consists of a case base: a relevant set of specific cases stored as complete patterns of experiences. Given a new problem, the Case-Based Reasoner retrieves the most relevant case in memory and subsequently adapts the corresponding solution to the current situation.

2. Promises

Because architects, and designers in general, are said to learn design by experience, the blend of CBR and CAAD seemed to hold great promises for the field of architecture in the early and mid 1990s.

First of all, CBD would bypass the knowledge acquisition bottleneck that Rule-Based and Model-Based approaches face to enrich CAAD tools with design knowledge. Whereas architects may have trouble citing the rules they apply or developing models of the objects they work with, they usually have no problems telling stories (which can be represented as cases) about previous design experiences.

Secondly, since generalizing always implies the danger of losing some important information, general rules, models or types are necessarily impoverished versions of the original experience they have been derived from. Analyses are not guaranteed to be correct and the resulting generalizations never perfect. Once knowledge is compiled into general principles, it remains compiled forever. In contrast, concrete cases, which contain experiences in their original, detailed form, can always be re-analyzed, re-interpreted and re-understood.

Given the fact that design problems are difficult to decompose, a third advantage is that "cases can provide the glue that hold a solution together" [1]. Buildings have to fulfill a whole series of different, sometimes even contradictory, requirements. It is quite impossible, though, to design a

building by decomposing it into parts according to these different constraints, solving each part separately and recomposing them into an overall solution. For every decision an architect takes is likely to have implications that cut across multiple aspects. Enlarging a window, for instance, may result in more light and a better view, but at the same time cause more heat loss and greater problems of privacy [3]. The architect's task is to integrate these different aspects into one coherent and meaningful design. Achieving integration in a Rule-Based or Model-Based approach is extremely difficult because there are few general principles that hold over all aspects. Cases, on the other hand, provide a natural solution to this integration problem: they themselves are integrated design solutions to particular contexts, including trade-offs among the several requirements a building needs to fulfill.

Fourthly, CBD has the potential to improve efficiency in architectural design. Many architects design every building as if it were the first of its kind, coolly ignoring the design experiences of their learned predecessors. Making use of previous cases during the design process could help architectural design out of its current inefficiency. The underlying assumption here is that adapting a previous design solution to a new situation takes less effort than generating a design from scratch.

A final promise is that CBD tools can extend architects' memory by storing design experiences they may not have thought of personally. The current problem may lie somewhat outside their proper area of specialization, they simply might have a bad day, or perhaps they are students and thus relatively novice in the field. Whereas novices approach a design by trial and error, experienced architects rapidly reduce the abundance of possible design solutions to a manageable handful. The difference lies in a repertory of heuristics that is accumulated through many years of design experience [4]. Prior to this experience, CBD tools can provide students with a substitute for the experience they are lacking so far.

Inspired by these and other prospects, a whole range of CBD tools have been, and are being, developed in the field of architecture, for example CADRE [5] and SEED [6] for building design, FABEL for technical buildings with complex installations [7, 8], Archie and Archie-II for courthouse design [9], and IDIOM for apartment floor layouts [10]. Several of them, like for example PRECEDENTS [11, 12], EDAT [13] and DYNAMO [14], are explicitly education-oriented. To learn from the experience of developing and using these tools is the objective of this paper (for an overview and detailed comparison of these CBD tools, see [15]).

3. Problems

At first sight, the expectations of the CBD pioneers appear far from being accomplished today. Despite numerous attempts by researchers to develop reliable CBD tools, a convincing breakthrough of CBD in architectural

practice and education has yet to emerge. Reasons for this rather limited success can be found at three different levels.

3.1. The cognitive model underlying CBD

The first level concerns the model of cognition in which CBD is historically rooted. Although this model may provide a plausible explanation of how designers acquire and apply design knowledge, within the field of architecture it raises some specific difficulties.

First of all, there is some confusion about the leading actor in this cognitive model: *design experience*. In general, design experience can be interpreted in multiple ways [16]. Strictly speaking, having experience in design means having designed yourself. In a wider sense, observing an architect's design process can be seen as a source of experience too. Finally, architects acquire experience by exposure to designs made by others, either in real life or, more often these days, through pictures in magazines, books, lectures or exhibitions. CBD researchers, however, tend to focus exclusively on this latter interpretation of design experience, the Galathea team being a notable exception to the rule [17]. Design cases, that is to say patterns of design experience, are usually interpreted as (fragments of) design products, generally, but not necessarily, at the final stage of the design process. Although design products definitely contribute to the acquisition of design experience, they cannot reveal the constantly changing conditions that actually structure the process of designing [18]. Indeed, "conflicting demands from within the client organization, the remoteness of the user, difficulties with the bearing capacity of the soil, an unsympathetic planning authority, changing circumstances during the design period, restricted or inflexible methods of financing scheme… and many more difficulties remain inscrutable to all but the most perceptible and insightful of architectural critics" [3]. Dealing with such changing and conflicting conditions requires a form of knowledge that is embedded within the very act of designing [19] and thus escapes the static form of a design product. It is obvious then that if architects learn design by experience, this learning does not only involve exposure to designs, but also active engagement in designing.

It would be misleading, however, to depict CBD researchers as completely unaware of the difference between product and process. Evidence of such awareness can be found in the idea of 'deep' cases [5], design products augmented with traces of the design and building process, such as options, decisions, justifications, assessments and revisions. In PRECEDENTS, for instance, cases contain descriptions of conceptual points, highlighting those aspects that were particularly decisive in the early stage of the design process [11, 12]. Archie-II supplements cases with stories from different stakeholders – people who carried out, use or maintain the building – telling how the design turned out [9]. Nevertheless, many projects share CADRE's preference for cases that are as shallow as possible, for the obvious reason that deep

cases are not there for the taking [5]. The FABEL team, for instance, seems rather pessimistic about getting richer cases and advises other CBD researchers to identify cases they might get in a realistic setting [8].

Another element of confusion in the cognitive model of CBD is the retrieval of a relevant case. What makes a previous design experience relevant to a new design situation in architecture is far from clear. Underlying most CBD research is the assumption that relevance equals similarity, in other words that the most relevant case is the one having the most features in common with the new design. In FABEL, these features range from functional components and visual appearance over occurrence of certain layout patterns to structure or topology [8]. Although this interpretation of relevance might apply to other design disciplines, its applicability to architectural design is highly questionable. In a series of interviews with architects we conducted [20], few interviewees confirmed the assumption of CBD researchers. Nor did they manage to formulate exactly why they call on a particular case during design. Probably the most honest answer we got was: "often simply because I *feel like* designing something like it." This feeling is not necessarily prompted by striking similarities between previous and current design situation. On the contrary, the more significant the difference, the greater the challenge.

3.2. The implementation of CBD tools

A second level, which poses serious problems, is that of implementation. Even if the model of cognition in CBD would perfectly match an architect's "designerly way" of thinking [21], few tools manage to draw the full consequences of this model, which often leads to an oversimplification of CBD and/or architectural design.

The problem starts with the representation of cases, i.e. records of design experience in a computer memory. A major advantage of using cases (as opposed to abstract rules or models) in design is that cases are concrete, full of detail, vivid and open to interpretation. However, the price of having to represent cases may be that this very richness is lost, simply because it refuses to be encoded in a digital form. Indeed, "representing design cases requires an *abstraction* of the experience into a symbolic form that the reasoner (either computer or human) can manipulate" [22, italics added]. No wonder CBD researchers prefer interpreting cases as products rather than processes. A major implementation question the Galathea team faces is precisely finding an appropriate format to describe and compare different states during design [17]. Yet, even in case of design products (i.e. building designs) digital representation can be characterized as a movement toward abstraction [23]. As long as human designers use this representation, there may be little cause for alarm. Indeed, what an architect derives from what is shown on the screen is much richer than what the computer only knows as 0s and 1s [24]. Through years of experience designers develop a

mapping between representation and reality, allowing them to form an idea of what the real building looks like. Yet, when cases are manipulated by a computer, as happens in CADRE, FABEL, IDIOM and SEED, the problem becomes more serious. Computers are not aware of a reality other than a design's digital representation. For them, this representation is the design. Thus, the representation of design cases reduces architecture to a set of related abstract, geometric shapes, instead of a meaningful form.

Another major advantage of cases that is seriously challenged by implementation is the coherence between the different aspects architects must take into account. Just like a case's richness, this integration is often lost in CBD systems. Some of them concentrate entirely on one single aspect of architectural design. CADRE and IDIOM, for instance, have focused on geometry or, more precisely, on the geometrical constraints on floor layouts. In these systems, integration is not a question, as other issues are simply not taken into consideration. A totally different approach, adopted in FABEL and SEED, is to develop different modules according to different aspects of architectural design (e.g. construction, topology, functional subsystem). Precisely because integration of such diverse aspects proved to be a non-trivial task, FABEL's modules – significantly called 'specialists' – are rather independent. Some are even equipped with a separate case base and adaptation method [8]. As long as this independence exists, perhaps such a tool should be called a collection of Rule- or Model-Based systems rather than a Case-Based one.

3.3. CBD in design education

Unlike the previous levels, where CBD's (in-)effectiveness has been discussed in relative isolation from the wider context in which it is used, level three tries to adopt a broader perspective. This perspective includes how CBD is integrated in design education, and how its use is presented by teachers, viewed by students and supported by the school. All these factors can influence the effectiveness of learning outcomes, even though they may not relate directly to CBD itself [25].

In general, design teachers are rarely burning with enthusiasm to introduce CAAD tools in design education [26]. In case of CBD tools, this enthusiasm is even harder to find, as these tools would considerably increase the danger of students blindly copying previous projects – a phenomenon known as 'design fixation' [27]. Judging from our experience with DYNAMO, student opinions are more divided. Some find cases an interesting source of inspiration, others share their teachers' concern that being original becomes more difficult. As one student testifies, "I know that everything has already been done. There are just some things that I'm not yet aware of" [14]. In our opinion, the present emphasis on originality and uniqueness in design education wrongfully excludes cases as treasures of knowledge from informing design in the studio [28]. Moreover, "originality

consists more in thinking for yourself than in thinking differently from others, and continuation of a living tradition, or an intelligent deployment of acquired knowledge are not the same as blind mimicry" [29].

Finally, the impact of the physical environment in which design education takes place should not be underestimated. Students do not always have easy access to computers, and this often prevents CBD tools from functioning the way they were originally supposed to. As long as students must leave their drawing board – or even the design studio – in order to access a CBD system, the majority of the students simply will not use it.

In our opinion, merely providing students with their own personal computer will not do. If CBD is to really support designers, cases should be accessible from within an integrated CAAD environment that supports sketching, modeling and testing, from the very beginning of the design process. This scenario still lies largely in the future, yet significant steps have already been taken in this direction [30].

4. Side effects

Although CBD may not have lived up to the early expectations, research in this field seems to have caused some interesting side effects. First of all, it has (re-)drawn attention to issues like creativity and copyright, often giving rise to interesting and lively debate.

As already mentioned, CBD is often criticized and dismissed as not being creative. Such criticism reveals as much about the critics' (mis-)conception of creativity as about CBD itself. Creativity here is viewed as 'creatio ex nihilo,' i.e. creating something out of nothing, a view researchers in creative behavior apparently do not share. According to Arthur Koestler, "the creative act is not an action of creation in the sense of the Old Testament. It does not create something out of nothing; it uncovers, selects, reshuffles, combines, synthesizes already existing facts, ideas, faculties, skills" [31]. Others describe creativity as "the easy recombination of ideas in the preconscious" [32] or "putting the elements of one's experiences into new combinations" [33]. What these attempts to define creative behavior demonstrate, is that the fact of starting from an existing design is no reason to accuse CBD of inhibiting creativity.

A second issue raised by CBD research is that of authorship in architecture. In principle, architectural designs are protected by the law of copyright. When complete designs become available in CBD systems, the danger of architects copying their colleagues' masterpieces would increase considerably. Back in 1834, Gottfried Semper said exactly the same about transparent tracing paper: "it would certainly hamper contemporary architectural design practice, and contribute to the then prevailing lack of ideas by facilitating copying of architectural elements of all times" [34]. More than one and a half century later, transparent tracing paper is still widely available, yet relatively few cases of plagiary in architecture have been

reported. On the one hand, the jurisdiction is not very inclined to give protection to architectural designs, presumably because of the important role technical means and their constraints play in these creations, and because their (at least partially) utilitarian character [35]. On the other hand, most architects seem to sense to what extent they can 'copy' a colleague's design without breaking the law on copyright [24]. Moreover, as Tzonis and White remark, copyright is far from being intended to prevent further use or development of a design. "The point of intellectual property is to encourage invention by rewarding the inventor, not by restraining those who learn from him" [29].

Interesting to mention in this respect, is Phase(x), a design strategy that leaves behind the single-authorship design process [36]. Students are introduced into the principles of CAAD by means of a design assignment divided in five phases. After each phase, students are not simply encouraged to look at each other's work, they must select the project of another student and further develop it in the next phase. Because the final results are design projects with shared authorship, there is much less 'negative' competition between individual designers. Students are able and willing to develop the best solutions, instead of continuing on their own, sometimes weaker, solutions. Although the applicability of Phase(x) seems limited to architectural education, its developers could imagine working in practice under similar conditions, thus realizing a breakthrough of productivity and quality in architectural design.

Apart from raising issues and stimulating discussion, another side-effect of CBD research may be the recent re-discovery of the key-role concrete cases play both inside and outside the field of CAAD. One example is the development of design support for daylighting in architecture. Although a growing number of testing tools to evaluate a building's daylighting performance have been developed, architects often continue to design in the 'traditional way', i.e. with reference to previous cases. Therefore, some researchers have decided to change track and provide architects with updated case studies of buildings embodying effective daylighting concepts, rather than with sophisticated testing methods [37].

Finally, we end with a project that lies outside the field of CAAD, yet successfully adopts a Case-Based approach to architectural education. The project, significantly called 'Little brother,' is a first year design assignment at the architecture school of Newcastle upon Tyne [38]. First, each student is introduced to an architect and his architecture. After an analytical phase in which students try to uncover and understand the architect's design philosophy, they are asked to design a small, single cell, studio, to be built within the garden of a house from his repertoire. Students have access to the building itself and to copies of the original working drawings. According to Diana Leitch, the choice of local architects for this project holds the double attraction of a direct link between student and architect, and of

course, a wealth of extra tutorial expertise. In fact, 'Little brother' combines different sources of design experience in a unique way: a 'living' architect, who can be inquired about his intentions during design and 'Big brother,' a concrete product devised by this process. The surprisingly high quality of the student designs, which inspired the organization of an exhibition, clearly indicates the value of this unique combination. Strictly speaking, 'Little brother' has nothing to do with CBD. However, it clearly demonstrates that a Case-Based approach definitely can contribute to architectural design education, be it without the support of computer technology.

5. The future

Having discussed the problems and side-effects of almost two decades of CBD research, what about the future? Is CBD a valuable path to follow, or should CAAD turn down another road? Although current CBD tools do not fulfill the early expectations, we should be careful not to throw out the baby with the bath-water. Projects like 'Little brother' encourage us not to brush the whole Case-Based paradigm aside. Of course, it is tempting to conclude that to develop more successful CBD tools, further research is needed. Yet, perhaps, something else is needed this time. A major reason for the so far limited success turns out to be confusion – confusion on the side of CBD researchers, over the meaning of design experience in architecture; on the side of architects, teachers and students, over what CBD is, and is not. As long as CBD researchers keep gathering at conferences to discuss new implementation methods, (student-)architects will probably quietly ignore their work. If CBD tools are to really support architectural design, what may be needed is a better communication and co-operation between on the one hand the practice and education of architectural design and on the other hand the world of research. For who better to tell researchers how cases can support design than architects?

Acknowledgements

Ann Heylighen is a Postdoctoral Fellow the Fund for Scientific Research (FWO) Flanders. This article is largely based on work undertaken as a Research Assistant of the FWO Flanders. The authors would like to thank Benjamin Geebelen for his critique and comments on earlier versions of this text.

References

1. Kolodner, J., *Case-Based Reasoning*, Morgan Kaufmann, San Mateo, 1993.
2. Schank, R.C., Dynamic Memory, *A Theory of Reminding and Learning in Computers and People*, Cambridge University Press, Cambridge, 1982.
3. Lawson, B., *How Designers Think*, Butterworth Architecture, London, 1990.
4. E. Rechtin, *Systems Architecture, Creating and Building Complex Systems*, Prentice Hall, London, 1991.

5. Hua, K. Faltings, B. and Smith, I., CADRE: Case-Based Geometric Design, *Artificial Intelligence in Engineering*, 1996, Vol. 10, pp. 171-183.
6. Flemming, U., Aygen, Z., Coyne, R. and Snyder, J., Case-Based Design in a Software Environment That Supports the Early phases in Building Design, in: Maher, M.L. and Pu, P., eds., *Issues and applications of CBR in Design*, Lawrence Erlbaum, London, 1997, pp. 61-85.
7. Schmidt-Belz, B. and Hovestadt, L., Scenario for an Integrated Design Support for Architects, *Design Studies*, 1996, Vol. 17, pp, 489-509.
8. Voss, A., Case Design Specialists in Fabel, in: Maher, M.L. and Pu, P., eds., *Issues and applications of CBR in Design*, Lawrence Erlbaum, London, 1997, pp. 301-335.
9. Domeshek, E.A. and Kolodner, J.L., Using the Points of Large Cases, *Artificial Intelligence for Engineering Design, Analysis and Manufacturing*, 1993, Vol. 7, No. 2, pp. 87-96.
10. Smith, I., Stalker, R. and Lottaz, C., Creating Design Objects From Cases for Interactive Spatial Composition, in: Gero, J.S. and Sudweeks, F., eds., *Artificial Intelligence in Design '96*, Kluwer Academic, Dordrecht, 1996, pp. 97-116.
11. Oxman, R. and Oxman, R., Case-Based Design: Cognitive Models for Case Libraries, in: Carara, G. and Kalay, Y.E., eds., *Knowledge-Based Computer-Aided Architectural Design*, Elsevier, Amsterdam, 1994, pp. 45-68.
12. Oxman, R. and Oxman, R., Remembrance of Things in the Past: Design Precedents in Libraries, in: Tzonis, A. and White, I., eds., *Automation Based Creative Design, Research and Perspectives*, Elsevier Science, Amsterdam, 1994, pp. 55-68.
13. Akin, Ö., Cumming, M., Shealey, M. and Tuncer, B., An Electronic Design Assistance Tool for Case-Based Representation of Designs, *Automation in Construction*, 1997, Vol. 6, pp. 265-274.
14. Heylighen, A. and Neuckermans, H., DYNAMO: A Dynamic Architectural Memory On-line, *Journal of Educational Technology & Society*, 2000, Vol. 3, No. 2, pp. 86-95.
15. Heylighen, A. and Neuckermans, H., A Case Base of Case-Based Design Tools for architecture, *Computer-Aided Design*, 2001, Vol. 33, No. 14, pp. 1111-1122.
16. Heylighen, A., Exposure to Experience, On the Role of Experience in Architectural Design Education, in: Scotford, M., Mabardi, J.-F. and Schneider, R., eds., *Research in Design Education*, Herber Center for Design Excellence, College of Architecture and Environmental Design, Raleigh, 1998, pp. 148-151.
17. Arlati, E., Bottelli, V. and Fogh, C., Applying CBR to the Teaching of Architectural Design, in: Klercker, J.A. , Eckholm, A. and Fridqvist, S., eds., *Education for Practice*, eCAADe, Lund, 1996, pp. 41-50.
18. Brown, J.S. and Duguid, P., Organizational Learning and Communities-Of-Practice: Toward a Unified View of Working, Learning, and Innovation, in: Cohen, M.D. and Sproull, L.S., eds., *Organizational learning*, Sage, London, 1996, pp. 58-82.
19. Schön, D., *The Design Studio: An Exploration of its Traditions and Potential*, RIBA Publications, London, 1985.
20. Heylighen, A. and Neuckermans, H., Are Architects Natural Case-Based Designers? Experts speaking, *Design Journal*, 2002, Vol. 5, No. 2, pp. 8-22.
21. Cross, N., Designerly Ways of Knowing, *Design Studies*, 1982, Vol. 3, pp. 221-227.
22. Maher, M.L. and de Silva Garza, A.G., AI in *Design: Case-Based Reasoning in Design*, IEEE Expert, 1997, Vol. 12, No. 2.
23. Matthews, D., and Temple, S., A Pedagogy of Interdependent Technologies: An Experimental Studio for Synthesizing Digital and Mechanical Processes, in: Neuckermans, H. and Geebelen, B., eds., *Computers in Design Studio Teaching, Transactions on Architectural Education*, No. 03, K.U.Leuven, Leuven, 1999, pp. 99-106.

24. Maher, M.L., Case-Based Reasoning And Design, in: Achten, H., Dijkstra, J. and van Leeuwen, J.P., eds., *Design Systems Report 2,* T.U. Eindhoven, Eindhoven, 1998.

25. Gunn, C., Isolation or Integration, in: Harvey, J., *Evaluation Cookbook*, Heriot-Watt University, Edinburgh, 1998, pp. 18-20.

26. Neuckermans, H. and Geebelen, B., eds., Computers in Design Studio Teaching, *Transactions on architectural education*, No. 03, K.U. Leuven, Leuven, 1999.

27. Christiaans, H. and van Andel, J., The Effects of Examples on the Use of Knowledge in a Student Design Activity: The Case of the 'Flying Dutchman', *Design Studies*, 1993, Vol. 14, No. 1, pp. 58-74.

28. Press, J., Soul Searching: Reflections from the Ivory Tower, *Journal of Architectural Education*, 1998, Vol. 51, No. 4, pp. 233-242.

29. Tzonis, A. and White, I., Introduction, in: Tzonis, A. and White, I., eds., *Automation Based Creative Design, Research and Perspectives*, Elsevier Science, Amsterdam, 1994, pp. 1-40.

30. Hendricx, A., Geebelen, B., Geeraerts, B. and Neuckermans, H., A Methodological Approach to Object Modeling in the Architectural Design Process, in: Timmermans, H., ed., *4th Design & Decision Support Systems Conference Proceedings*, T.U. Eindhoven, Eindhoven, 1998 [CD-Rom].

31. Koestler, A., *The Act of Creation*, Arcana, London, 1989.

32. Barron, F., Creativity, in: *Encyclopedia Britannica 6*, W. Baton, London, 1969, pp. 711-712.

33. MacKinnon, D.W., Creativity, A Multi-Faceted Phenomenon, in: Roslansky, J.D., ed., *Creativity: A Discussion at the Nobel Conference*, North Holland, Amsterdam, 1970, pp. 29.

34. Semper, G., Vorläufige Bemerkungen Über Bemalte Architektur und Plastik Bei den Alten, in: *Kleine Schriften, hrsg. von Hans und Manfred Semper*, Kunstwissenschaftliche Studientexte 7, Mäander, Mittenwald, 1979, pp. 215-258.

35. Ballon, G.L., *Recht voor ingenieurs, Een inleiding*, Mys & Breesch, Gent, 1995.

36. Kolarevic, B., Schmitt, G., Hirschberg U. and Kurmann, D., Virtual Design Studio: Multiplying Time, in: *Computerised Craftsmanship*, Ecole d'Architecture de Paris Val de Marne, Paris, 1998, pp. 23-130.

37. Baker, N., Fanchiotti, A. and Steemers, K., eds., *Daylighting in Architecture, A European Reference Book*, James & James, Brussels, 1993.

38. Leitch, D., Little brother, in: S. Chardonnet, ed., *Writings in Architectural Education*, EAAE, Louvain-la-Neuve, 1995, pp. 8-15.

Ann Heylighen and Herman Neuckermans, Katholieke Universiteit Leuven, B-3001 Leuven, Belgium

ann.heylighen@asro.kuleuven.ac.be
herman.neuckermans@asro.kuleuven.acbe