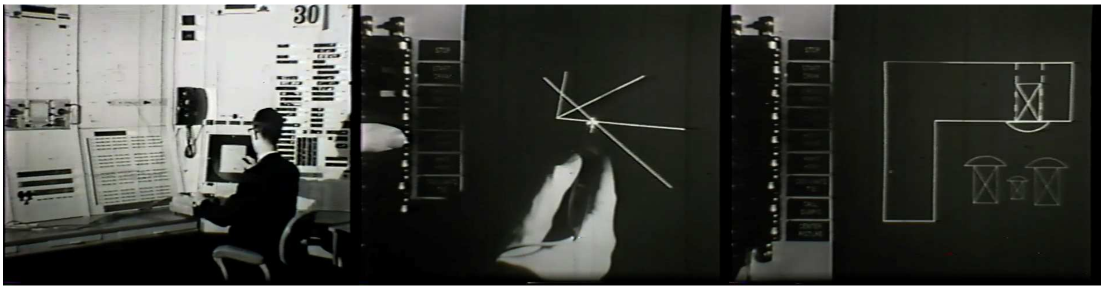
A Perspective on Computer Aided Design after Four Decades

We offer a summary view of competing theories that have guided and inspired the development of computer aided design over the past forty years with attention to how they support design processes. We identify eight distinct schools of thought. We then speculate on what a collective view of these schools posits with respect of the next generation of tools. This perspective reflects our view as teachers and researchers at institutions with different curricula, a first-hand role in the development of computer aided design technology, and specialization in design theory and methods.



Sketchpad, 1963, Ivan Sutherland (left), Constraints (middle) and Instances (right)

First generation computer aided design tools were intended to support a design process rather than provide for pre-conceived and ready-made solutions. Sketchpad, the 1963 landmark system developed by Ivan Sutherland at MIT, demonstrated that a design process could be viewed as setting and relaxing geometrical constraints, building up a kit of object instances, and allowing for parametrical control over their formation (Sutherland, 1963). Some current generation tools, due to the desire to automate design and documentation, may focus constraints, instances and their variation on conventional architectural types, such as “wall types”, “window types” and other pre-defined objects. These systems appear biased towards commonly recognized building products. Still, their intention is to give the designer a free hand in the layout, scale, and relationships between a set of parts.

On the one hand, designers value flexibility in exploring new forms. On the other hand, they also begin to constrain objects as a sketch transforms into one invested with specifically constructed materials and purposes. The state of the art is imperfect in addressing these sometimes opposite demands. For example, some tools focus on geometry only and offer a great range of formal explorations. Other more automated computer aided design systems offer design lifecycle features that can quickly expedite assemblies of three-dimensional building models based upon more conventional components (e.g., doors, windows, walls) where the geometrical configurations, material attributes, and part-to-part relationships (e.g., windows placed walls) are anticipated in the software, and where construction documents and material take-off lists are more easily generated.

Today it’s difficult to have both a capacity for open ended geometrical modeling and highly automated assembly of architectural products in the same software. The requirements for software to automatically assemble conventional architectural parts seem to impose a set of geometry limits. Can we train a computer aided design system, as first proposed by Negroponte, to become more adept at delivering upon our individual architectural intentions and open formulation of objects, and perhaps partner with us in uncovering our design process (Negroponte, 1970)? The range of approaches and opinions varies widely.

Admitting to oversimplification and overlap, our paper briefly describes the following schools of thought: (1) design by constraints, instantiation and parametric variation, (2) shape grammars, (3) frame based design methods, (4) object oriented design, (5) generative systems, (6) top-down design, (7) knowledge based design systems, and (8) design and cognition. We find potential common ground in some observations:

- Design process in architecture is intimately associated with geometrical modeling.
- Geometrical models need to be mutable with less predefined representations.
- Variation in design does not proceed at the whimsy of the designer but is informed and can be controlled via a set of established relations between objects and invented rules.
- A design proceeds from ambiguous and loosely defined relationships among objects to deliberately associated relationships.

This yields some common conclusions about the next generation of tools:

- Geometrical modeling processes should at times re-engage the histories of how individual objects have been created.
- Knowledge based approaches or rule-based system must not be fully deterministic.
- The development of smarter design tools will continue to beckon research and theory well into the future, as each generation discovers new paradigms to more tightly connect design activity with the acts of making and building.

Given our present view of the state of the art in design tools, what direction might development of technology take? The perspective of the paper is limited. We attempt to organize a pattern of achievements over the history of computer aided design. We argue that the next generation of design tools should emphasize diversity of specialized approaches rather than comprehensive technology solutions (a caveat relative to BIM).

We note the observations of Martin Woolley who claimed that in the post-IT era tools must be developed ad hoc as per need, where the designer is the tool builder (unlike today). In this respect the tool is not selected, but created as per need and contingent on the type of design task, the stage it is in, and adapted to circumstances. This may have more implications for interfaces than for the actual output of the particular tool, for databases that are fluid, and for tools that offer an open architecture, rather than those that make automated construction simpler and easier for all. The growing acceptance of computer technology in our academic discipline and the design profession motivates putting the present state of the art in perspective.

KEY WORDS: Design Theory and Methods, Constraints, Parametric Variation, Instantiation

References

Negroponte, Nicholas, The Architecture Machine, MIT Press, 1970.

Negroponte, Nicholas, Reflections on Computer Aids to Design and Architecture By Nicholas, Petrocelli/Charter, 1975

Schon, Donald, The Reflective Practitioner: How Professionals Think in Action, Basic Books, 1983

Eastman, Chuck, *Fundamental Problems in the Development of Computer Based Architectural Models*, Proceedings of the Computers & Design Research Symposium, M.I.T., August 4, 1986.

Sutherland, Ivan, SKETCHPAD: A Man-Machine Graphical Communication System, proceedings of the AFIPS Spring Joint Computer Conference, Detroit, Michigan, May 21-23, 1963, pp. 329-346.

Woolley, M. (2004). The thoughtful mark maker, in Goldschmidt, G. & Porter, W.L. (eds.), *Design Representation*, London: Springer Verlag, 185-201.