Teaching Computation for Design

Richard NORMAN

Clemson University, United States of America

Abstract: Modeling the formal idea has long been problematic in schools. Renaissance tools, however inspired by the invention of perspective, usually result in two-dimensional sketches. Cardboard and small pieces of wood occasionally become the three-dimensional media of first visualization in studios; modeling on the computer is a newer idea. This paper examines two experiments, one where design fundamentals and solid modeling are introduced in a common studio, the other where instruction is removed from the studio environment and made an adjunct elective. In the first case the course is an introduction to both design and computation, an electronic investigation of Paul Klee's first principles: point, line, shape, form and space—adding, subtracting, rotating and multiplying objects using Beaux Arts principles to create the design. The result is architectural form-making that was not possible in the studio-past. The second case is a course which isolates computer instruction from the studio, making it a separate academic discipline. Fantasy projects then demonstrate computational principle, exploring pure form without burden of technical or social obligation that studio imposes; alternative methods are presented for introducing design computation to the architectural student.

1. Principles

Architectural ideas are born in the mind. Sure there are lists of conditions, searches and codes, but the fundamental idea—the creative thought—is a product of the mind, it is not computed. But soon computation can begin and for a first sketch the ability to model in three dimensions is paramount. This paper explores three-dimensional modeling as the designer's tool of choice. No pretense is made that modeling is a complete design process—no structural or programmatic issues are addressed in these courses and the many social issues so essential in architecture are put aside; modeling addresses only the form-giving aspects of design. Examined is how computer modeling can be taught, how the designer best learns this complex software. Two experiments are presented, one in a class where the first design studio is used to instruct in both design fundamentals and the techniques of solid modeling; the other where computer techniques are covered in a separate course, where instruction in modeling is removed from the studio environment to become an elective.

Just as drawing has its courses, so computer tools have become sufficiently complex that they deserve their own academic position. Architects today grow up with the tools of the renaissance—plan, section, and elevation; perspective, too, is a renaissance invention and is credited with changing design. Solid modeling is about to change design again, just as forcefully. Three-dimensional images have been usually an afterthought, but computation can change that, it can make the three dimensional drawing primary—the tool of choice for a first sketch. This paper explores how solid modeling should be integrated into the curricula of architectural schools.

Somewhere in the search to conceive each design a thought is born, an idea, or perhaps an intuition. Sure there are computer lists of requests—conditions, searches and codes; but the fundamental idea—the creative thought, is a product of the mind; it is not computed. Attempts to find the black box computer that can do this have not been successful.

But computation can soon begin as a part of the design process, and the ability to model in three dimensions becomes paramount. This paper will therefore pursue three-dimensional modeling as the designer's tool of choice. this is not a complete story, but an exploration of how the solid model can revolutionize the form giving aspects of design process; it is about the need for computerized solid modeling in the studio, and how it can be taught.

Architects have grown up with renaissance tools, with the pre-eminence of plan, section, and elevation as methods of conception. Perspective, also from the renaissance, is credited with changing design concepts. Solid modeling is about to change things again, just as forcefully. Solid modeling uses the computer to draw in a way that does not duplicate what has always been done by hand, this is their potential. The paper explores two methods for integrating solid modeling into the curricula of architectural schools.

2. Modeling in the basic design studio

To teach a basic design studio on a computer is to believe that computation can be learned along with design One must assume that the subjects need not be separated, that in the studio a computer is simply made the tool of choice. A series of projects are then devised for the dual purpose of introducing solid modeling and design principles simultaneously. The test case is a basic studio, a first course offered to non-architecture majors. Let's see how this works.

Historically there are two models for beginning a first design studio, one developed at the Ecole des Beaux-Arts, the other at the Bauhaus; architectural schools have subscribed to one method or the other. Beaux-Art teaching begins with the principles of composition, originally drawing from plaster casts. In principle student must become facile with the hand—with other people's compositions—before being allowed to "design". Under Beaux-Art teaching, only the proven student could go on take advanced courses—to design architecture. Drawing was a first principal learned by everyone at the school; architectural composition was learned and practiced in drawing classes, long before one ever designed so much as a room.

The Bauhaus disapproved of this policy, found it illogical—unstructured and contrary to function. Bauhaus students began by designing a building. The accuracy of simple linear orthographics appealed to this group for their simplicity and logic. Paul Klee's theories of the point as a prime generator of form became the method d'jour; thought was advanced from point to line, line to plane, plane to form, and form to space. In Bauhaus teaching this is the grammar of form, it begins with architecture. Plan and section drawing were preferred; perspective was applied at the end, a sort of finalé that was often hired out, only to show everyone what the building looked like. Practitioners at the Bauhaus would never admit to the notion of a three-dimensional "afterthought", but that is what developed from their teachings.

Computers can emulate either process, but really introduce a third way of designing as a unique contribution to the designer's bag of tools—the direct construction of a fully rendered three dimensional model. Careful construction of a Beaux Arts rendering is not needed when you can have instant, perfect perspective; those articulate ink washes learned in Beaux-Arts drawing classes are all replaced by computation. The precision of Bauhaus drawing is all there, but it is warmed with color and made understandable with automatic perspectives.

The new studio course begins with the theories of Paul Klee as design principle, it moves from compositions of simple points to more complex drawings. The intricacies of point and line are quickly submerged by the mesmerizing presentation of form and space that solid modeling allows. Follow now a series of projects, a sequence used to introduce these design fundamentals alongside those of solid manipulation on the computer—a first studio course.

The first project is about "point". A point is always an abstraction, never an object; only with difficulty does it become the subject of a studio project. Points can convey emotion, though they are difficult to objectify and impossible to function. If one begins with points and lines, then emotion must be the glue. This is not a bad beginning for discussion of computer principles. The first project is about "point". A point is always an abstraction, never an object; only with difficulty does it is desired—it can also begin a discussion of architects and of the emotional qualities of architecture. Emotion is present in a line drawing class when one faces a figure with a piece of charcoal—those qualities are harder to see in a building, still harder to find on the computer.

In this project "lines" are arranged to illustrate a design principles: rhythm, balance, or emphasis. It is very hard to draw a sensual line on the computer. One tries, talks about it and hopes that emotional qualities are not submerged by the technical discussion.

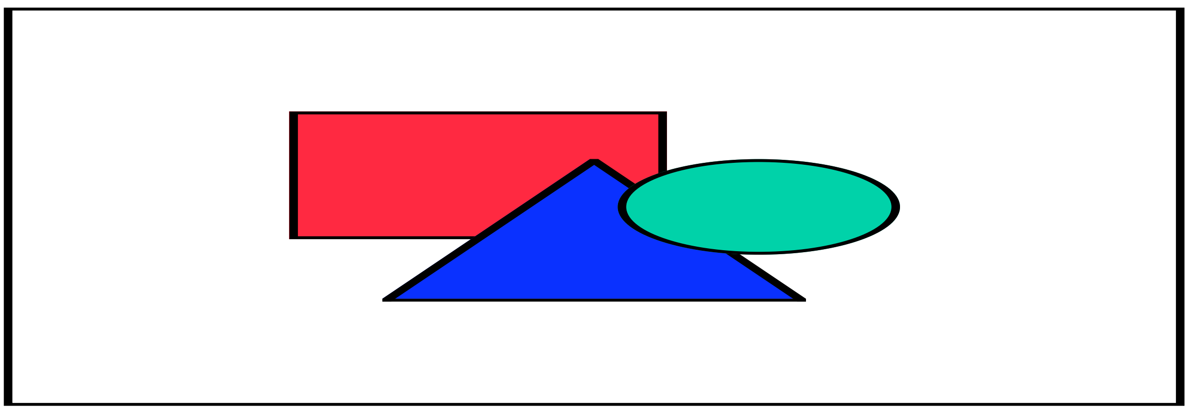


Figure 1. Line Compositions

Shape can be functioned; they easily imply plans when they are seen as rooms arranged across a sheet of paper. Freshmen like function as a form determinant, they find it more easily understood than form. Abstraction is a way of isolating form, reinforcing design principle and eliminating preconceptions.

Three-dimensional design begins in earnest with the introduction of "form" as a design element. A transition is made to three-dimensional modeling, an enormous step. What was once a collection of shapes on flat paper becomes a three-dimensional world of forms that one can climb upon and move about in. Complexity increases dramatically, it is no longer possible to explain everything from one point of view. There is an explosion in the class of architectural interest. On the computer everyone thinks they can draw; students become like children first discovering the pencil, or the crayon. In computer terms there is a whole world of new information to learn, a seemingly endless list of forms and the ways they can be changed—the exercise is computationally rich. Can it be also valuable as a design experience? This requires a persistent reference to principle as the exercises are developed. Studio focus is on the appreciation of form as an element of design.

The geometric primitives found in a computer modeler make excellent basic design material. If sufficient formal vocabulary has been established with the exercises in "point" and "line", then an abstract project can be understood in three dimensions, form can be understood in a way that would make the Bauhaus proud—no such abilities were within their grasp.

For a freshman the transition from form to space can be difficult because focus must shift from form to a concentration on the space between forms. In the studio-past talk has been of arranging space rather than form. On the computer boolean subtractions let you carve space more directly. Design can start with a given solid and from it a space can subtracted; like Michelangelo one searches for the form that is "hidden in the marble".

The assignment is a dense block of form that is given as the site for a sequence of spaces. The exercise makes a valid design point, forcing one to consider the space—its proportion, rhythm, balance—the entire design vocabulary. Now it is applied to space rather than form. On the computer the class is given simple tube figures—the kind simply created with today's modeler. By moving these people around, the eye is led sequentially through spaces, giving them a scale by their presence and a sequence because of the numbers on their back. Both "scale" and "time" are introduced as design elements. The exercise introduces the boolean operators for carving a form, and discusses methods for manipulating point of view in order to see the spaces that have been created. This is a successful merger of computer technology with design fundamentals.

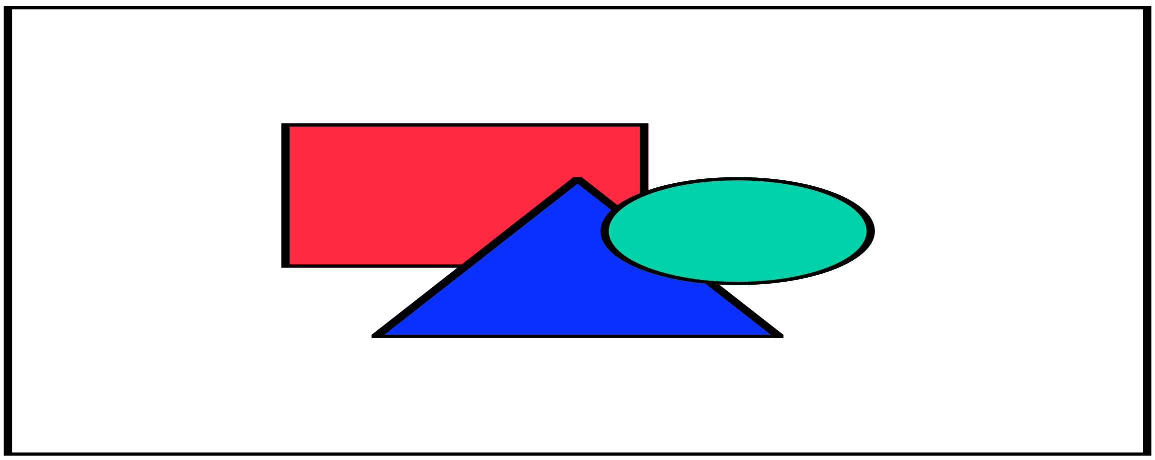


Figure 2. Form, Space, and Time

The bridge to paradise is a project in color. The assignment is simply to build a bridge between "good and evil". This is a way to discuss color with all its complications. By requiring alternative color schemes for projects, the student is made aware of "color" as a design element, the computer permits color manipulation in a way that was never possible in the studio.

This project introduces color as an element of design. The Bauhaus produced several theories of color behavior, but instruction in these principles has all but disappeared from design schools in recent years. The Ecole des Beaux-Arts did not formalize color study, though it was frequently discussed and carefully used. With computers the topic becomes again relevant; everything seems to come in color now, from copiers to the web.

The suggestion of "bridge" as a design topic excites many functional urges in the student. Form decisions can be easy, one can always put a slab from one side to the other and walk across, but the results are seldom that simple.

Color contrasts are required on the two banks of the gorge, for this is the bridge to paradise; it implies the opposition of bad and good. By changing the color selections for each bank the two sides differ, but things are complicated by introducing a "theme within a theme". Structuring a project explicitly about color can force consideration of the differences that color can make.

Function comes into consideration only in the final project of this course, a test of both budding design ability and computer prowess. How simple can an architectural project be made? A shelter is to be designed for a stay in an idealized environment—no bad weather, no bugs. There are no new skills involved in this exercise in either computers or design, this is the summation of a semester's learning, a first effort at shelter. Pre-conceptions come out of the woodwork, gadgets and the necessities of life, rather that the principles of design; but the results are usually a good summation of what is learned about design in the course. Students are made aware of design principle as a generator of form. Considerable dexterity is shown on the computer, it is possible now to assemble a group of primitives, cut and stretched to convey design intent. The new tool has been learned, adequately, to where it can be used to good purpose in future design efforts. They have also been made aware of design principles.

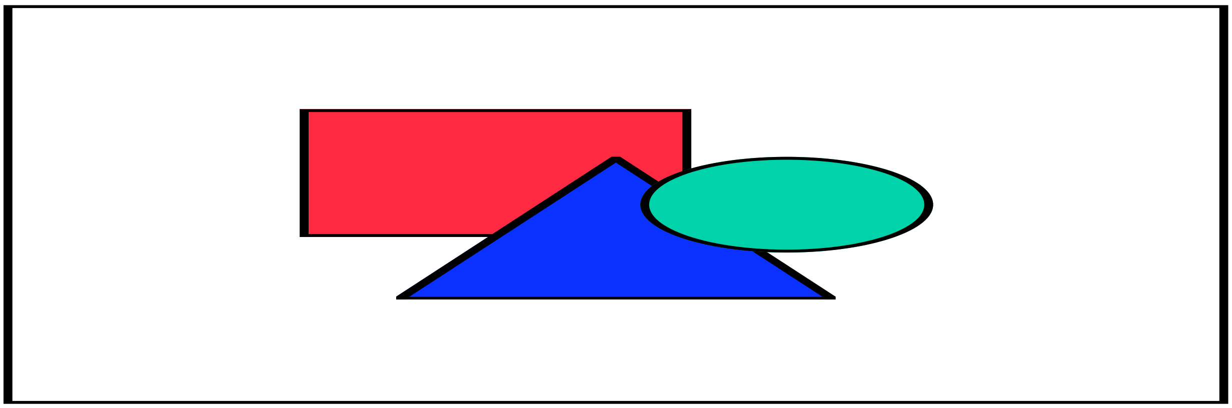
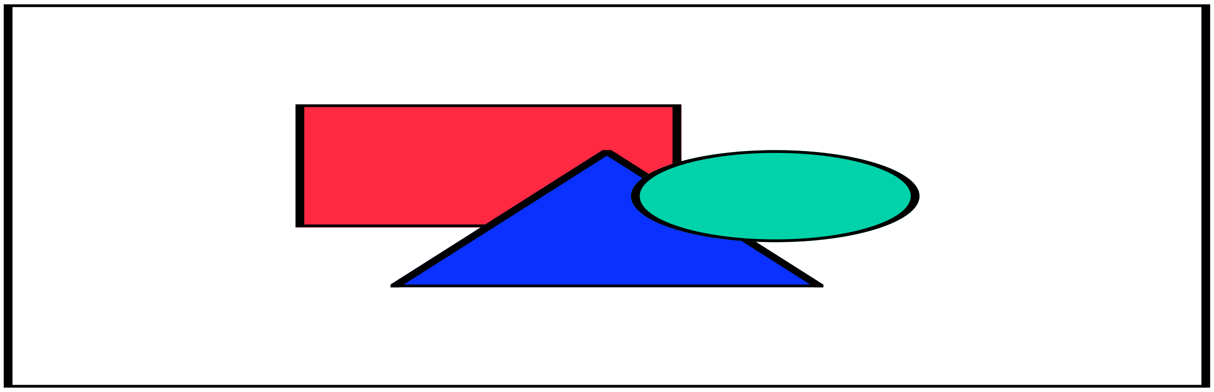


Figure 3. Beginning models

3. An Adjunct Course

Now let’s compare a second course in which all computer instruction is removed from the studio and offered as an elective subject. The resulting class drew students from all levels of instruction, bonded by a common interest in computer modeling. The course projects are separate from the studio. Generally a "fantasy" project is assigned, the sort of "non-brainer" that stimulates discussion about form without requiring the preparation that design research would demand. It is difficult to devise a project that encourages design exploration but does not tax design mentality. The course is designed to instruct in solid modeling in a relevant way for students with a wide diversity of design experience. It begins with a series of "fantasy projects" and ending with the application of what has been learned to each student’s own project.

This course opener involves the design of a chess set, modeling with a geometry that is common to both chess pieces and the solid modeler. It involves duplicating forms, moving them around, and changing their color—all fundamentals of any modeling system. A demonstration brings students to where they can technically combine the forms needed to assemble chess pieces. A theme is asked for, to avoid the clichés of the super mart chess set. There are only 6 playing pieces in a chess set, it seems to offer enough complication and design potential to intrigue everyone. Additionally we can usually be seen from the default viewpoint on a computer, leaving a few lessons for the next class.



The design of a simple shelter of is usually the next pursuit; shelter design seems common to all studios; several assignments are built around this theme. First forms are combined to create a space, then using boolean operations forms are subtracted from a solid. Implied always are the many ways of conceiving space.

Mature design students realize the potential of the ideal climate that is assigned—one needs no roof, no windows—only form for form's sake. This project usually requires multiple points of view, a new computer experience. It opens the door to further instruction in computer techniques. Designers find their own ways to complicate a problem, they are quick to discover that Photoshop images easily become a project background, that the web is full of photos to be used—the creative mind searches always for new ways to use software. It has been found that by removing the project from studio a greater experimentation in computer technique is encouraged.

Subtractive design should not be a new concept for anyone experienced in the precepts of a studio course—but the concept seems better illustrated on the computer than with pencil. As a designer one expects the result of a subtractive process to appear different from what is added. Boolean operations are introduced, along with the simple design admonition that what is cut should look cut.

Full appreciation for a site comes only through experience—to the freshman the world is usually flat. The assignment here is simply to design a gorge and then build a bridge across it. Methods of site construction are explored; like its cardboard predecessor the stepped model built on a computer can seem simple, but for an experienced designer the art of entering the information to form a desirable landscape is complex. It opens the door to more further computer procedures for modeling the surface of a form.

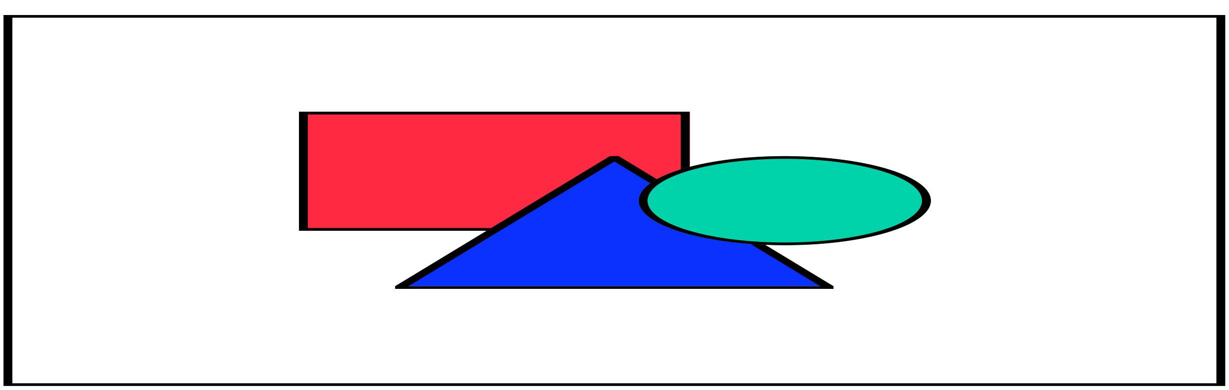


Figure 5. The bridge to paradise

The ambitious assignment here is to use a computer model for showing a complex form—a task not easily done without recourse to a plan. Many subjects will do, we were plotting an escape from prison. There are many ways to make a stair with modeling software, lots of opportunity for technical instruction. But as an essential part of either efficient exiting or gracious entry, the stair is far more than technology. Stairways glue the parts of a building together, they isolate zones, they can make an entrance commodious. This project explores the intricacies of stairs.

Until this exercise, each project had been discussed as a discrete scene, never as a sequence of spaces that were tied as continuous phenomena. The greater purpose is to show a building as a whole, to sense the entire scope of a complex project. Before computation it was possible to draw a plan, which together with ample sections and elevations would illustrate a complete work; no corner was too far away. But solid modelers are not intended for drawing plans. Further, it is a contradiction to design by building a model in the computer, only to stop this process and draw the plan in another way. If methods can be found to show complexity by entirely using the computer model, then the computer becomes a better tool. Escape from Sing Sing is a search for this capacity.

For some people animation of the scene is the way that a project can be shown holistically. Virtual reality is introduced into the design process; but often it is not logical to "stop and animate". We are looking at a tool for first design decisions. For some people alternative view selections help, choosing overhead views or take-apart models. For others, rendering can show the architecture more completely; wire frame models explain the whole while rendered models to explain the parts. A combination of these techniques can illustrate a concept more completely.

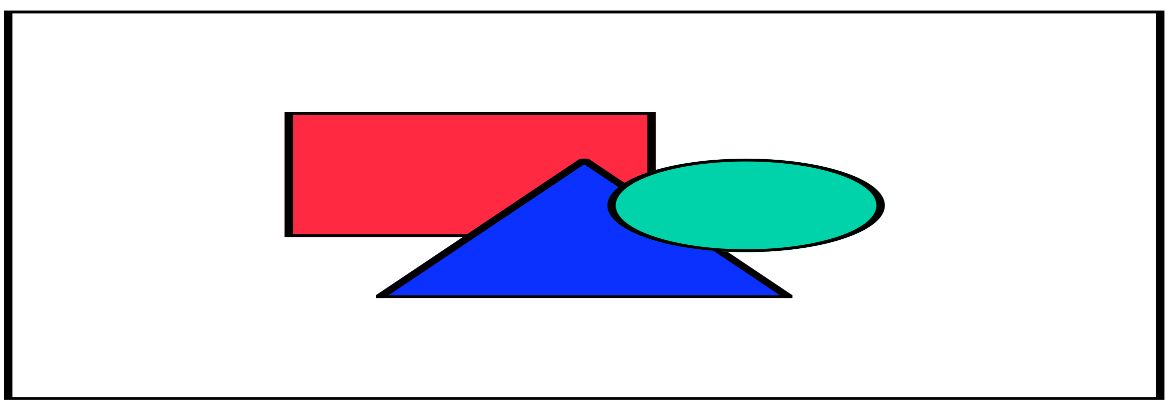


Figure 6. Escape from Sing Sing

Finally the hut becomes a house as the idea of home completes computer instruction in this course and ties the information back to the design studio. Covered here are many of the technical "tricks" needed to visualize a project that is drawn completely on the computer. Importing files and exporting what is created to other programs are both essential modeling skills. Imported files give context; seeing a site on the screen removes the design from designing on white paper. This final projects illustrate how computation is never one software package, but an attitude that reaches for all the world’s technology, bringing to the studio a whole new gamut of design experience.

The conclusion to the course is an effort to integrate modeling into the studio environment. Students complete a studio project of their choice, using the computer modeling techniques The results are as diverse as each students experience; These projects demonstrate how enriched modeling can make the studio experience at any level. In every case the projects undertaken are better understood, altered substantially by the use of this new tool.

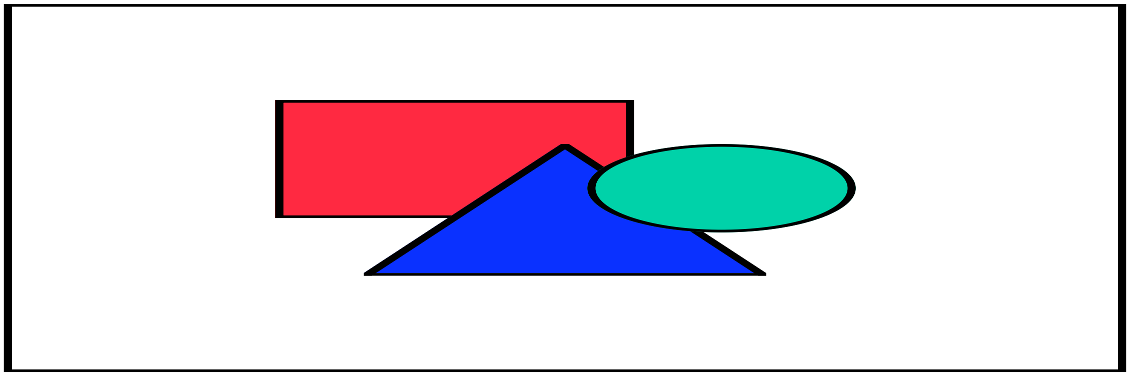


Figure 7. The house — studio projects

4. Conclusion

To learn the use of new tools efficiently and creatively is in many cases beyond the usual design class; today's studios already have more missions and objectives than a semester can cover—the skills of computation deserve a place of their own. Two courses were presented in this paper. One, a beginning studio, assumes that design and computation can be learned simultaneously; the other, an elective course, permits interested students to explore design computation apart from the studio. It is perhaps an unfair comparison, for advanced students are usually better designers than beginners. But as a process for learning, the separation of computer material from that of the studio allows a depth of study that is not possible when everything is studio-based. Students appear to apply the lessons of computation well to studio projects when those lessons are separated; solid modeling seems to deserve a place of its own.

Computer processes do create astounding graphics—but not automatically. As a design tool solid modeling is changing the way that forms are conceived; in many ways its importance parallels the discovery of perspective in that it provides a new way to visualize buildings. Both schools and offices are discovering that computation is inevitable because it provides a better way a see what is designed. The integration of computers into teaching deserves careful consideration; computation is changing design process; it will inevitably change buildings themselves.

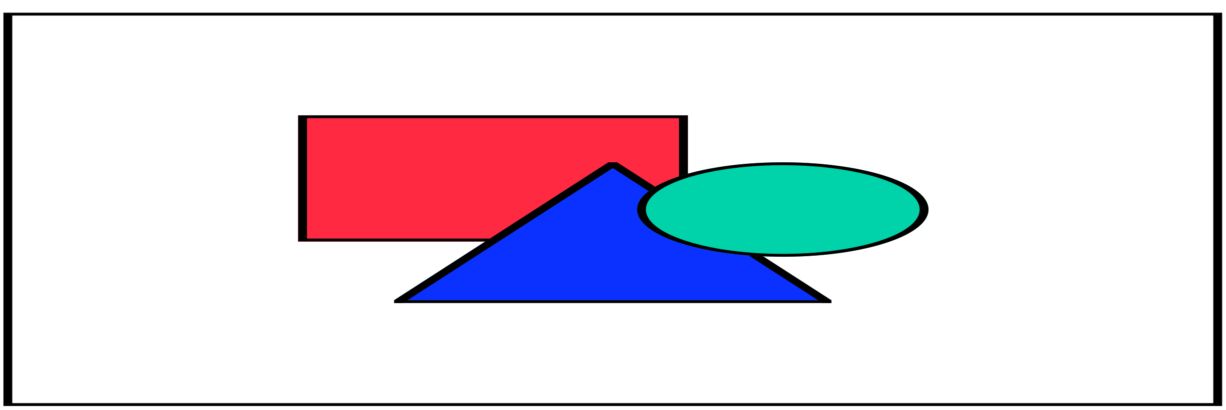


Figure 8. Studio projects

5. Acknowledgements

Illustrations included in the paper are the work of Clemson University students:

Figure 1. Ian Bannister, Tony Parker, Time Clark, Brett Krammer

Figure 2. Jason Childs, Tony Parker, Seth Kirschner

Figure 3. Ian Bannister, Billy Evett, Jason Holder

Figure 4. Brooke Taylor, Michael Hoenes

Figure 5. Wei Zheng, Scott Morris

Figure 6. Brooke Taylor, Steve Padgett

Figure 7. Todd Hoffman, Scott Morris

Figure 8. Rob Moehring, Yuan Chen