# BIM applications in Design Studio

## *An integrative approach Developing Student Skills with Computer Modeling*

*Dr. Stan G. Guidera RA, PhD*
*Bowling Green State University*

**Abstract**

*This paper proposes a reductionist approach to the integration of software used in professional practice with course activities associated with design studio. This proscriptive strategy emphasizes the use of task-specific software features to support specific aspects of design project activities and learning outcomes. The rationale of this strategy is discussed followed by case studies that review the application of a reductionist approach to building information modeling software in a third year studio and a design foundation course.*

## Introduction

Digital design tools have enabled architects to embrace the exploration of the form and space unbounded by issues of constructability. The ability to create "hyper-architecture" has had a profound effect on the zeitgeist of 21$^{st}$ century architectural education. In contrast, practitioners have utilized computation as a means of increasing productivity and have been critical of architectural education for producing graduates that are perceived to lack technical knowledge as well as skills with computing (Mitgang 1999, 1997). With the growing role of Building Information Modeling (BIM) in practice, architectural education will be viewed as a resource for skill development with this technology. Consequently, educators will need to adopt methods to develop BIM-related skills in ways that contribute to the overall learning objectives of the curriculum while providing a foundation for skills needed in practice. This paper proposes that, through a reductionist approach, complex digital design tools such as building information modelers can be integrated into curricula in a manner that supports learning outcomes associated with design studio while fostering the development of computing skills that are increasingly central to architectural practice.

## Computing and design: CAD or CADD?

The first-generation commercial CAD software systems that emerged from the research initiatives at MIT and Cambridge in the mid to late 1960s were primarily intended to automate repetitive drafting chores (Computer Aided Drafting). By the 1990s high performance 3D CAD applications such as CATIA and Pro/Engineer had evolved to provide highly integrated 3D/2D functionality. These applications, widely adopted in the automotive and aviation industries, introduced design as well as drafting functionality (CADD) (CAD). However, unlike engineering-based disciplines, the productivity gains associated with 3D CAD did not materialize in architecture. This was due in part to the systematic way in which these disciplines "cast a design problem" in a form that would render it amenable to such methods (Liddament 1999). In contrast, architectural design problems are more "ill-structured" and involve less definable parameters and objectives (Goel 1995), and are therefore less amenable to a computational solution. The inherent precision of CAD has led to the perception that it is inappropriate for the representations required in early stages of the design process, which are characterized by intentionally ambiguous representations such as sketches and freehand vignettes. Due to the precision of its representations, CAD has "accentuated the divide between explicit and implicit information" (Koutamanis 2000) and is inherently unambiguous in that its representations are explicit rather than implicit (Chastain, Kalay, & Peri 2002). Additionally, the underlying organization of CAD software is perceived to have an implicit influence on its operation, leading to the premise that it would influence design exploration and that *a priori* solutions to design problems are inherently inevitable. This leads to the conclusion that, among novice designers, design explorations would be influenced by the extent to which a software application facilitated the constructability of forms and objects, leading to the use of forms that were easiest to create and modify.

Productivity gains were also limited by resistance to takes steps to exploit technology. The move to mass computing in architectural practice was, in part, customer driven as clients in the late 1980s and early 1990s began to require drawings in digital format (Andia 2002). However, even though CAD production became a baseline requirement among most clients (Green 2002), most architects chose to implement CAD in a way that did not alter existing practices. Information was drawn twice, first manually by an architect and second by a CAD operator (Andia 2002). Automation of pre-existing processes reflected the common approach among industries to attempt to utilize new tools in the same way they used their old ones (Fallon 1998). As in manual drafting, the use of lines and shapes produced representations of architectural elements such as walls and doors that were "generally incapable of corresponding to architectural elements in an appropriate manner" (Johnson 1998).

### Object Oriented Programming

This paradigm shifted with the introduction of design applications based on Object Oriented Programming (OOP) languages. Object oriented programming is computational ontology; a data model that represents a domain and is used to reason about the objects in that domain and the relations between them. Programmers define a data structure, and then define the operations that can be applied to that data structure. Therefore, the data structure is defined as an object that includes both data and functions with specified relationships between the object and other objects. Benefits include reduction of complexity, the ability to re-use or combine objects, and easy modification and extension of individual components without requiring re-coding (Kilkelly 12). Consequently, architectural objects relate "intelligently" (Ethier & Ethier 2000) and function more intuitively: floors "know" they are to be horizontal and the components of a window "know" that they are part of a wall (Mondero 2000). The result is an application that, despite its power, is easier and more intuitive to use.

### Building Information Modeling

Parametric object-based design tools have become standard in architectural CAD applications and the ability to utilize parametric control of geometry and dimensional relationships has become an expectation (Roberts 2004). While the design flexibility and productivity afforded by parametric modeling positively impacts the role of computing in architectural design, the adoption of parametric computer modeling does not in itself dramatically alter the design process. However, this is not the case with the current generation of OOP-based architectural CAD applications that utilize Building Information Modeling (BIM). Building Modelers expand the conceptualization of computer modeling from an assembly of 3D geometry to that of a repository of project information and introduce a fully model-centric design process. BIM applications integrate 3D-2D operations, and 3D-2D model-based technology is linked with information. Lower-level tasks such as drafting, view coordination, document generation, and schedule creation are automated, suggesting that computation for architectural practice may have matured to a level at which the benefits may now be on par with those in other disciplines. Davis (2003) stated that "BIM is what we all expected computer-aided design (CAD) to be," adding that "we finally can harness the power of computers in a real value-added design process that doesn't just mimic drafting."

According to Roberts (2004) "the growing adoption of the building information modeling (BIM) methodology has raised practitioners' expectations and overall awareness of the value of tools and technologies that offer more informed design experiences." As with CAD, the adoption of BIM is likely to be driven by client requirements as well. Additionally, governmental agencies and other large-scale clients are already developing and implementing mandates requiring the use of BIM in project delivery. It is this context that will require architectural educators to consider the role of BIM technology in both design and technical education. However, consideration of techniques for integrating BIM applications and design studies has potential for developing learning activities that are aligned with the goals and outcomes of design studio while providing students with a foundation for utilizing these tools in practice.

## Integrating BIM in Design Studio

The tradition of the studio as the core of architectural design education has been well documented. Therefore, using studio activities as a mechanism to develop skills with building information modelers has greater potential studio outcomes with the needs of practicing professionals. Three very general approaches to relating to computing and its role in design studio can be identified. The first two approaches have been referred to as the "exclusionary" computing studio, and the "tangential" computing studio. In the exclusionary studio computer and CAD use is either proscribed or, if allowed, is not supported. In the second model, the "tangential studio," content related to computing is taught in discrete course or courses separate from studio. These applications are permitted to be used to produce drawings and images typically for final presentations. However, minimal formal studio time is allocated to skill development with computation. Both approaches are aligned with the premise that studio may be an inappropriate venue for learning to use computers as a design tool.

A third "digital studio" approach is organized around the use of digital design, a process in which design decisions are made on screen rather than on paper. This approach would be characterized by the use of 3D computer models at all stages of the design process, and must be distinguished from virtual studio. In contrast with other approaches, the digital studio approach is reliant on studio instructors having a high level of expertise with the software applications. At some institutions this limitation has been addressed through team-teaching in which the design studio instructor is paired with a CAD instructor.

There are curricular as well as pedagogical issues associated with the "digital studio." In order for skills with CAD and computer modeling to be developed to the level at which they can be employed as design tools, extensive studio time must be allocated to learning to use software. Therefore, other outcomes intended to be met by a studio course may be compromised. This may again require a separate course for developing competencies with higher level CAD and computer-modeling skills.

### Task-specific computation: Object Modeling Studio Approach

An alternate approach to integrating computation takes as its point of departure the premise that 3D modeling applications may be more appropriately utilized as a design a tool that considers CAD software not as a holistic design environment and representation tool, but rather as an interface that can be "decomposed" to exploit specific features that are clearly aligned with outcomes. This reductionist approach is analogous in its conceptual underpinning to that of Liang and O'Grady's (1998) DwO (Design with Objects) design process. They suggested that the emphasis of object oriented design is decomposition and representation. Decomposition refers to the breakdown of the functions of a product so that the lowest level of the function structure consists exclusively of functions that cannot be sub-divided further," and that "it is decomposition that differentiates object oriented design from conventional, structured design."

This approach is arguably counter-intuitive to the software designer's efforts to expand power and functionality of their software in order to address the full range of issues and phases associated with design, an approach clearly aligned more with the demands of the profession than those of architectural education. Task specific computation approaches the cognitive development of computation skills in a manner that is at least to some extent parallel with design skill development with an emphasis on fundamental concepts and problems.

This approach is also based on the premise that the strengths of CAD can be exploited by identifying a task or tasks for which the software is most suited and requiring it to be used for that task. However, the BIM knowledge and skill set required is also more focused since competencies with all the features of the software is not necessary to complete the task. Both students, and studio instructors, need only develop competencies with the specific commands associated with the task.

This reductionist strategy for introducing Object-based computer modeling in studio is based on a proscriptive rather than inclusive approach to the use of the software. Assignments are structured to systematically limit the use of software features. The utilization of features is tied directly to specific learning outcomes of the studio project. The advantages of this approach include the following:

*Emphasis on Traditional Studio:* Since the required role of the computer is supportive rather than central, the emphasis of the studio remains on design and aesthetic explorations. The use of task-specific CAD requirements does not mandate changes in studio activity time-frames or sequences, or prevent or limit the use of other media requirements or activities such as physical models, etc.

*Time allocation efficiencies:* Since the skill set for the task-specific requirement is narrowly focused, the time allocated to providing software-specific instruction does not adversely impact the amount of class time that is devoted to primary studio activities.

*Media Independence:* The task-specific CAD requirement does not restrict students from using digital representations and media for other project elements. Students are free to pursue other graphic techniques that either they or the studio instructor wish to pursue.

*BIM skill development is incremental:* Fundamental skills associated with the use of BIM is introduced incrementally and developed in parallel with fundamental design skills. Little & Cardenas (2001) proposed that in addition to being sufficiently complex to permit an evolving design space and allowing for multiple acceptable solutions, design studio problems should also benefit from the use of design tools. In this scenario, students are provided an opportunity to expand CAD skills and to explore the use of CAD as a design tool within a studio context. For many students it could be a first opportunity to use CAD for 3D modeling tasks and as a tool for creative problem solving.

## Integrating BIM in Design Studio: Case studies

As noted previously, successful implementation of BIM applications in design studios is reliant on the decomposition of the software operation in order to facilitate the use into tasks linked to specific studio outcomes. Therefore, the design of the studio project must include a systematic process for analyzing the software operations and connecting the use of technology to the studio goals.

The following third-year studio project and two first-year studio exercises utilized Autodesk's Architectural Desktop. However, while it was necessary to cover software-specific features, the emphasis on parametric objects and styles would make the assignment adaptable to utilizing other BIM.

Each example followed a similar sequence:

- Learning objectives that used computation to support course outcomes were established
- Software features were assessed in order to determine what operations would be most conducive to promoting the intended knowledge and skill development and the operational parameters were defined
- Define the relationship of the task to project product (project development and presentation requirements)
- Define the in-class task-specific Training Requirements and Content

## Project One: Structural Integration

This assignment was incorporated into a half-semester third-year studio project for the design of a mid-rise apartment complex with a parking garage and atrium. Submission requirements included a physical model along with conventional orthographic drawings. The project was structured to be media independent; students were given the choice to produce representations either manually or with digital media. All students had completed an introductory statics course as well as an introductory 2D CAD course. Therefore, the assignment assumed students were at least somewhat familiar with both basic CAD operations and structural concepts.

### Structural Integration Project Outcomes:

The design studio activities focused on the design of moderately complex structures with learning outcomes emphasizing programming and adjacency analysis, the relationship of building envelope and structure, and the relationship between built form and site. Computer modeling activities used to support these outcomes required the students to create a computer model of the structural system of their design proposal in order to encourage students to integrate knowledge from introductory structures courses into the design studio. However, the studio focused on a conceptual approach to the structural proposal emphasizing the use of a "visual approach" to structural design in order to foster a qualitative interpretation of structure which can result in the principles of structures becoming an unconscious part of the students' way of thinking: (Abdelmawla, Elnimeiri, and Krawczyck 2002).

The objectives associated with the software task were to:

- Provide students with experience with employing computer modeling concepts in a design studio environment
- Provide students with tangible experience synthesizing theoretical issues with technical knowledge
- Require students to demonstrate a fundamental understanding of the aesthetic as well as functional implications of a steel frame structural system for their design proposal

### Selection of Software features and Task Operational Parameters

A key benefit associated with the use of Structural Component modeling tools in Architectural Desktop was the availability of generic structural components as well as geometry based on shapes listed in the American Institute of Steel Construction catalog, which introduced a more tangible "real-world" component to the task. Structural members can be positioned in 3D space using feature-based parameters such as justification and elevation settings as well as with standard CAD drawing techniques (move, copy, object snap) enabling users with basic CAD skills to create structural

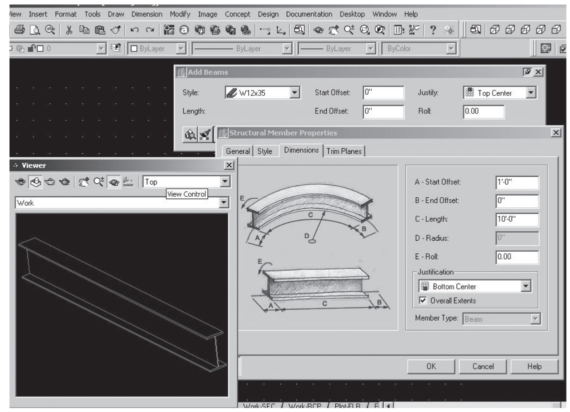

*Figure 1. Structural Parameters for Steel Beams.*

assemblies. The structural elements are style based and therefore easily interchangeable; a beam or column shape can be "swapped out" for a different shape and retain any parameter settings. Lengths, starting and ending offsets (extensions), and trim and intersection properties could be edited parametrically (Figure 1).

Software use was limited to column objects, beam objects, and slabs. The initial modeling task employed generic columns and beams styles (components with fixed sections but parametric lengths and features). Once a structural system was developed students were permitted to use styles from the AISC library to develop more realistic representations which would expose students to object-interchangeability and style-based component modeling, both fundamental concepts underlying the operation of OOP and BIM applications. Representations were to be limited to isometric views. No materials were permitted.

### Project output

Students were not required to use digital media in any component of the project development other than the task-specific course activities. Assignments called for physical models as well as 2D orthographic documentation at each phase of the project. While final project requirements were also media independent, students were required to include 3D representations of their structural models in their final presentations.

### Task-specific Training Requirements and Content

Since all students in the course possessed at least basic CAD knowledge and experience, course instruction was focused in four basic areas:

- Translation of 2D CAD navigation skills to the 3D CAD environment and the introduction of 3D specific navigation techniques, and use of pre-set isometric views
- ADT specific content including ADT object palates and basic style-based modeling concepts
- ADT column and beam assembly, structural member modification techniques, and the use of the ADT structural style catalog
- Introductory rendering, including file out-put, lighting, and perspective view set-up

### The Project Sequence

The studio project was divided into a three-phase sequence: "conceptualization - structural exploration - re-conceptualization." During the four weeks allocated to the first phase, students developed a proposal using physical massing models and orthographic drawings. The conceptualization phase explored the project in a more traditional studio context. References to structural concepts were intentionally loose and not given a high priority in the review processes. While the design proposals were to indicate a "logical structural system," no drawings of the structural system were required. Following the conceptualization phase critique, time in four class meetings was allocated to providing students with training for the command set needed to complete the 3D structural assignment. Each session was either followed or preceded by desk critiques. During this phase the BIM application was used to create a model of a steel-frame structural system for the proposal. Developing the structure as discrete design iteration required students to represent the structure as a "stand-alone" three-dimensional composition and to consider the aesthetics of the structural system.

In the last phase, students were required to reassess and refine their proposed design solutions and integrate the structural system and the design proposal. This phase required a synthesis of issues raised in both Phase One and Two and culminated in a formal presentation to faculty and external critics for assessment.

### Discussion and Assessment of outcomes

The initial structural explorations revealed several common characteristics among the proposals:

- Most initial explorations were limited to steel frame structural systems using standard columns and beams placed on a grid closely aligned with their plans. In many cases, these elements were simply substituted for bearing walls; structural elements were primarily placed at locations of both exterior and interior walls, indicating a lack of awareness of structure as an architectural element or an understanding of the potential for the building envelope to be independent of the structure.

- Sizes of structural elements used were largely independent of location, indicating limited acknowledgement of structural loads. Additionally, the visual characteristics of the size of structural members were not addressed. For example, the selection of structural shapes and sizes for tall columns were often "visually" too thin even if structurally they may potentially have been acceptable. There was some experimentation with substituting AISC members rather than generic members. (Figure 2)
- All proposals were at least to some extent incomplete. In several cases where structural grids did not or could not be aligned with the phase one design, the structural model was left unresolved indicating a lack of understanding of how to proceed with a structural system for that condition.

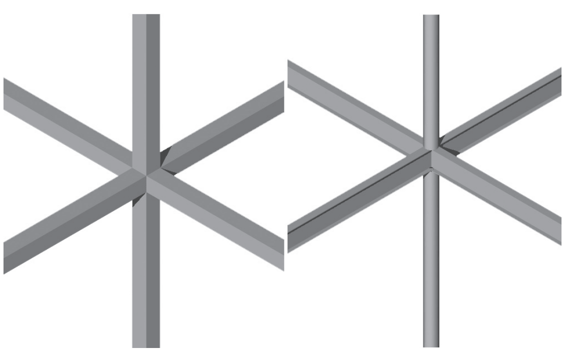

*Figure 2. Generic structural components (left), revised with AISC components (right).*

These characteristics were attributed to a lack of applied structural knowledge as well as lack of experience with using the software. While for many students this was their initial experience at working with 3D CAD, there was little difficulty among even the least experienced students in understanding 3D concepts. Students commented that their experience using drawing aids such as object snaps in 2D was easily transferred to 3D.

As the project developed, a greater consistency between the structural model and the design proposal emerged. There was a noticeable increase in explorations of structural aesthetics. Several students had begun experimenting with trusses and bracing elements (Figure 3) which were selected for their architectural effect rather than for structural considerations.

Additionally, at this stage some proposals began to use free standing or exterior structural components as design elements and reflected an increasing awareness of the influence of structural conditions on member size.

Models reflected adjustments that were influenced by revisions to their project, particularly in terms of building elevations. For example, the structure and/or building were modified to coordinate with curtain walls and at locations where structure was exposed. In most proposals the relationship between building envelope and structure was less restrictive, reflecting experimentation with cantilevers and

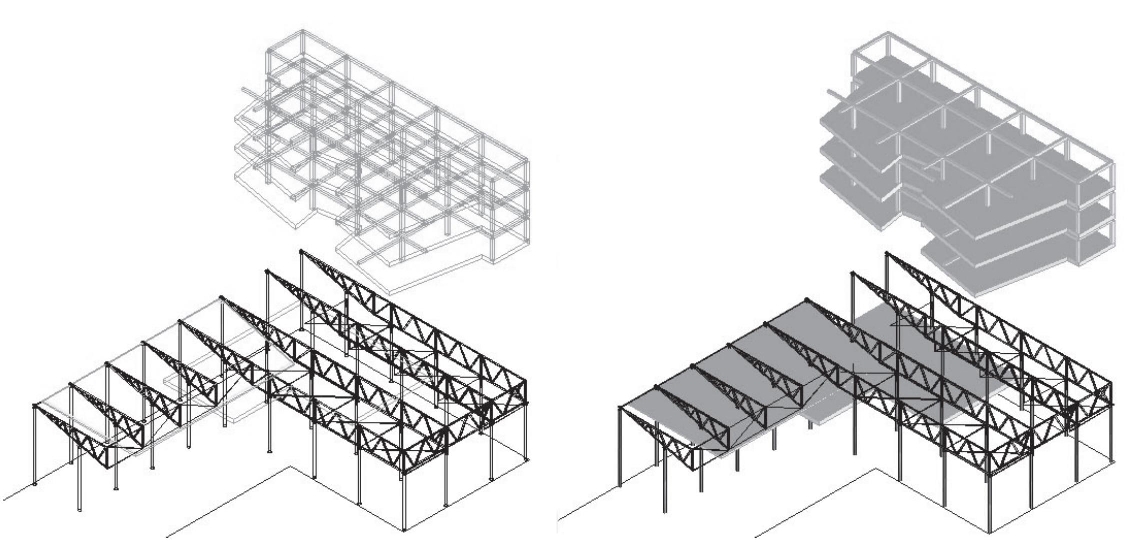

*Figure 3. Structural explorations using trusses and slabs.*

curtain walls.

Phase Three proved to be challenging as the students attempted to resolve inconsistencies between their project refinements and modifications to their structural system. Students varied in their success in producing visually and intuitively correct structural proposals in which structural member sizes and placement corresponded in some way to loads and spans. However, though some students found they needed to allocate much of their studio time to redesign rather than refinement, several students produced projects that were highly developed architecturally with structural systems that were functional and in some cases highly detailed, thus exceeding expectations (Figure 4).

### Project Two: Introductory Design Studio Exercises

Two exercises for what would be considered design foundations courses were developed to integrate computing using a BIM application. Students had completed a design representation and design foundation coursework. The first project involved a predefined column grid structure which established the project envelope. The emphasis of this project was on the use of vertical and horizontal planes to define interior and exterior space as elements independent of the column grid system. The second project was developed as a hypothetical "end building" of a block of row-structures. This

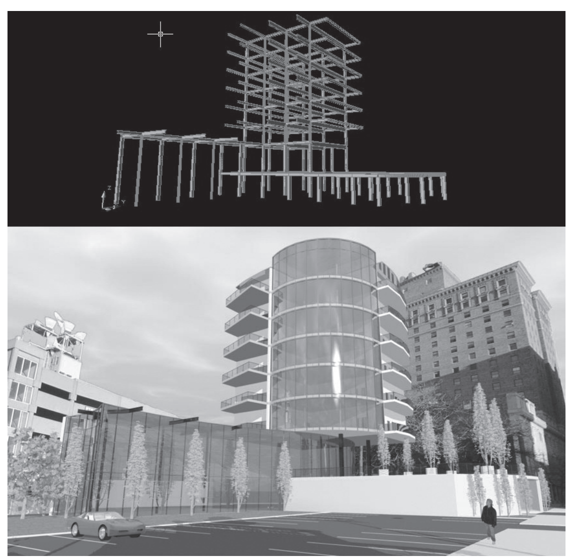

*Figure 4. Perspective Rendering (Final Submission).*

scenario provided the parameters for the building envelope and a project that had an exposed wall at three of four sides. As with the third-year project, these projects were organized with three phases: the first phase using physical modeling and orthographic drawings, a second phase that was exclusively computer based, and a third and final design phase intended as a synthesis of the work and critiques related to the first two phases. Both foundation exercises incorporated a "physical" limit on the project building envelope and emphasized the use of planes in the definition of space. The total area of the horizontal planes and the plane elevations were pre-defined, however, students were free to allocate the horizontal plane area as part of their design proposals. Both design problems also required students to incorporate vertical volumes that integrated horizontal planes.

### Learning Objectives:

The primary learning objectives associated with the project were to:

- Develop understanding of the integration of vertical and horizontal space
- Develop ability to conceptualize enclosure in relation to the use of vertical and horizontal planes
- Develop understanding of the implications of openings as a device for creating thresholds and transitions

The learning objectives associated with the software task were:

- To provide students with an alternative way to explore the integration of volumes
- To exploit computer modeling as a means for students to produce alternative proposals
- To use computer modeling to enhance student's comprehension of the experiential implications of their design proposals

### Selection of Software features and operational parameters

Since solid modeling objects would require a competence with a much wider range of commands, ADT wall objects and slab objects were selected as the primary geometric components. The wide variety of door styles available in ADT was viewed as a potential distraction for novice designers who may be tempted to use the highly detailed and easily inserted pre-defined assemblies rather than focus on the issues related to conceptualizing an opening as a void in a solid and/or as a transition/threshold between spaces and volumes. Therefore, the use of doors and windows was proscribed; openings within walls were to be created with a generic parametric opening available in the standard design palate. To accommodate vertical movement, the students were required to incorporate one stair element, but the geometry was limited to the generic style. Additional features such as railings were not permitted.

A "startup" file would be provided to students in the studio. The file included pre-modeled geometry that would provide the "project context"; a three dimensional column grid and a ground plane for the first project, and a building mass and project area ground plane for the second. This geometry was created with conventional solid modeling and formatted to be un-editable.

The students' limited CAD experience also lead to a decision to utilize predefined object and style-based modeling components in the selected BIM application to simplify the computation variables and to facilitate students' emphasis on the issues. The default design palate was customized to eliminate some of the object-types the students would not be incorporating in their designs. The file was also pre-configured with two view-ports, a plan view and a 3D isometric view, with instructions that the view-port configuration must be retained over the duration of the project. The preset parameters in the startup file were:

- Generic slab style with thickness pre-set at 12"
- Generic wall style with width pre-set at 8"
- Wall Transparencies: In addition to the generic wall style, a second wall style was pre-set in the startup file with a preset material that was semi transparent.
- A pre-set light source was set to facilitate rendering at later stages of the project.

The wall and slab dimensions were identified as program requirements, and therefore were to be reflected in all physical modeling and manual orthographic representations. Walls and slabs were not to be angled or sloped vertically.

In the computer model, students were not permitted to use materials or add lights. Only one camera was allowed in each project. The students were permitted to change the direction of the camera, but were prohibited from modifying other parameters in order to reduce the operational knowledge associated with setting up their perspectives.

## Project output

Students were only required to use the software for the task-specific course activities. Project requirements included physical models as well as 2D orthographic documentation. While final project requirements were also media independent, students were required to include 3D representations of their models, including one that used a perspective view that communicated a sense of the interior space that was created.

## Task-specific Training Requirements and Content

Students participating in these projects had little or no CAD experience, requiring course time to be allocated to specific aspects of basic CAD operations. The use of a BIM application enabled the students to learn these basics by drawing with walls rather than lines and other abstract 2D geometry. Additionally, the use of object properties to position and adjust objects in 3D space reduced the number of CAD commands and techniques needed to be covered. Basic commands identified as essential for the tasks included zooming and panning, use of the preset 3D views, object snaps, and 2D and 3D coordinate entry. The use of modification commands was limited to erase, move, offset, and copy.

Software-specific features determined

necessary for the assignment were:

- Use of the object (design) palate
- Use of the properties palate to change and set object parameters (such as wall justification, height, and elevation above the ground plane
- Use of the properties palate to exchange styles
- Use of camera and camera position modification dialogue box.
- Basic rendering (for purposes of perspective rendering).

## Project Sequence

The initial tasks required students to build physical models and manually produce plan and section drawings. The physical model was to use components dimensionally equivalent to those pre-defined in the digital context. After initial critiques, students refined their proposals using computer models. The starting point for the computer model was the proposal presented in the initial critique. This provided a physical object used as a reference in developing the computer model.

## Discussion and assessment of outcomes

The project parameters, specifically the limitation that the objects could not be rotated or sloped, proved conducive to students developing competencies with the necessary CAD operations since it allowed the in-class "training" to leave out many commands that would have been necessary otherwise. The initial trials of these projects found that by emphasizing the use of object snaps and ortho-drawing modes, students were able to create wall components and modify them using object handles or grips as well as with manually entered dimensions and coordinates in the object properties palate. Working in 3D was also simplified by pre-setting the files with two viewports and emphasizing the use of the preset views.

In the case of the wall components, the property palate was used to change wall objects styles as intended. The object property palates were also utilized reposition objects in both horizontal and vertical directions. Although this was less efficient in terms of productivity, it was assumed that the students found that using the property palate interface was less confusing than manipulating objects on screen with the mouse or typing command line coordinates. All students were able to complete the technical parameters associated with the project (Figure 5).

It was anticipated that the rendering activities would prove to be the most problematic aspect of the assignment for the students, particularly in placing and navigating the camera. However, all

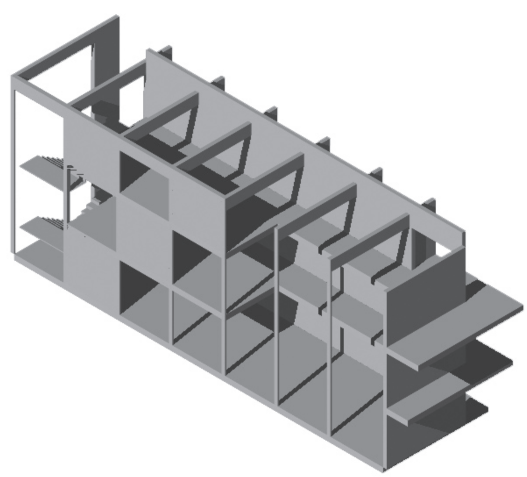

*Figure 5. Design Foundations course computer model: Isometric rendering.*

students were able to place and modify the cameras.

The greatest difficulty with the project requirements was related to modeling the stairs. All students had difficulty configuring the stairs to be properly positioned and sized relative to the geometry of the rest of the model. This was attributed to two factors. First, many students were not familiar with the architectural terminology associated with the stair objects, making it difficult for them to know the appropriate parameters to adjust to appropriately manipulate the object as intended. Secondly, the number of variables they were required to adjust was higher, which also affected student's abilities to place and configure the ADT stair objects.

## Conclusions

It is the intention of this paper to illustrate that computation, specifically computer modeling using Building Information Modeling software, can be effectively introduced at early stages of the curriculum through the use of an exclusive and prescriptive approach to software features and commands. While this strategy does not expose students to many of the advanced analysis, documentation, and representational features that BIM applications have developed to serve the computational demands of the architectural profession, it does develop an understanding of the conceptual underpinnings of object-based modeling, thus providing a foundation for the use of more advanced applications of BIM later in the design curriculum as well as in the profession.

## References


Abdelmawla, S., M Elnimeiri, and R. Krawczyck. (2000). Structural gizmos. In *The ACADIA Proceedings: Eternity, Infinity, and Virtuality.* Washington D.C.: 115-121

Andia, A. (2001). Integrating Digital Design and Architecture During the Past Three Decades. *Seventh International Conference on Virtual Systems and Multimedia*. Berkely, CA: Virtual Systems and Multimedia.

CAD Software History, 1960s – 2004. Available at http://www.cadazz.com/cad-software-history.htm.

Chastain, T., Y. Kalay, and C. Peri. (2002). Square Peg in a Round Hole or Horesless Carriage? Reflections on the Use of Computing in Architecture. *Automation in Construction*. 11(2): 237–248.

Davis, Dianne. (2003). Available online at *http://www.aia.org/tap_a_0903bim*

Ethier, S. and C. Ethier. (2000). *Instant AutoCAD architectural desktop.* New York: Prentice Hall.

Fallon, K. (1998). Early Computer Graphics Developments in the Architecture, Engineering, and Construction Industry. *IEEE Annals of the history of computing.* 20 (2): 20-29.

Green, K. (2002). How clients choose architects. In *The architect's handbook of professional practice: Student edition.* ed. J. Demkin, 19–22. New York: John Wiley & Sons.

Goel, V. (1995). *Sketches of thought.* Cambridge, MA: MIT Press

Johnson, S. (1998). What's in a representation, why do we care, and what does it mean? Examining the Evidence from Psychology. In

*Automation in Construction.* 8(1): 15-24.
Kilkelly, M.(2000). Off the Page: Object Oriented Construction Drawings. In *The ACADIA Conference Proceedings: Eternity, Infinity, and Virtuality.* Washington DC: 147 - 151.
Koutamanis, A. (2000). Digital Architectural Visualization. In *Automation in Construction.* 9(4): 347-360
Laseau, P. (2002). *Architectural Representation Handbook.* New York: McGraw-Hill.
Leglise, M.(1995). Art Under Constraint - Preserving the Creative Dimension in Computer Aided Design. In *Languages of Design*. 3(1): 27-52.
Liang, Wen-Yau and Peter O'Grady., Peter (1998). Design with objects: an approach to object-oriented design. *Computer-Aided Design.* 30(12): 943-956.
Liddament, T. (1999). The Computationalist Paradigm in Design Research. In *Design Studies.* 20(1): 41-56.
Little, P. and M. Cardenas. (2001). Use of 'studio' methods in the introductory engineering design curriculum. In *Journal of Engineering Education* 7:309-318.
Madrazo, L. (1998). Computers and Architectural Design: Going Beyond the Tool. In *The ACADIA Proceedings: Digital Design Studios: Do Computers Make a Difference?.* Quebec, CA: 44-57.
Mitgang, L. (1999). Back to school: architects sound off on 10 critical issues facing architectural education. In *Architectural Record*. September, 1999:112-114
Mitgang, L. (1997). Saving the soul of architectural education: Four critical challenges face today's architecture schools. In *Architectural Record.* May, 1997: 124-130
Mondero, J. (2000). Parametric Design: A Review and Some Experiences. In *Automation in Construction.* 9(4): 369-377.