Between Design and Digital: Bridging the Gaps in Architectural Education

Shelby Elizabeth Doyle, Iowa State University
Nicholas Shane Senske, Iowa State University

Available at: https://works.bepress.com/nicholas-senske/22/
Between Design and Digital: Bridging the Gaps in Architectural Education

Shelby Doyle and Nick Senske.
Iowa State University.

ABSTRACT This article brings accepted educational research into the discussion of digital design’s relationship to architecture and architectural education. Digital technologies, such as computational design and digital fabrication, have transformed the design and construction of contemporary architecture. However, a lack of educational theory among instructors and widespread belief in pedagogical myths, such as the ‘digital native,’ have made it difficult for architecture schools to establish teaching methods for effectively integrating technology. In response to this situation, the authors present two proposals that attempt to address this issue at both the tactical (instructional methods) and strategic (curricular) levels. Respectively, the first proposal describes the teaching of soft skills for digital design and the second uses Bloom’s Taxonomy as a method of developing learning objectives for digital design instruction. These proposals represent two examples of how educators can bridge the gaps that commonly exist between design teaching and technology teaching.

KEYWORDS digital, design, pedagogy, technology, education

Introduction

Digital technologies, such as computational design and digital fabrication, have transformed the design and construction of contemporary architecture. However, the understanding of how to teach digital technology as an essential design skill has not kept pace with these rapid changes, as evidenced by uneven educational distribution of digital technology. Design education and digital technology education continue to be seen as separate loci of learning, with pedagogical gaps preventing meaningful alignment. In the interest of helping to bridge these gaps, this article presents a pedagogical agenda for digital design in architectural education by debunking myths such as the
‘digital native’ and applying proven educational research to the pursuit of digital design.

**Between Design and Digital**

Over the past three decades, Computer-Aided Drafting and Design (CADD) technologies have become commonplace in architectural practice as tools of efficiency and production. For these very reasons the introduction of CADD in early architectural curricula has been fraught with anxieties along a continuum: from the undoing of creativity through positivist and reductionist logic to a firm belief that these technologies will revolutionize the way architects practice and think about design. The presence of these anxieties and biases often leads to gaps in architectural pedagogy, as digital tools are misunderstood and misappropriated by students and teachers alike.

Digital design is a term in common use, however, its definition is unclear. On one hand, there is very little architectural work today which does not use the computer in some capacity, and yet there are also designs which consciously engage in digital formalisms and processes. The latter is digital in aesthetic, but the former could still be considered digital by method. The very existence of the category of digital design is problematic because it implies two cultural silos in architecture: those who are digital and those who are not. These two cultures are part of the gap between design and digital that exists in education.

For the purposes of this article, *digital design* refers to the use of the computer and computer-driven tools (such as CNC machines, robots, etc.) when one designs architecture. The key is not what is designed, but rather whether the computer is employed, or not, as a tool in architectural work. This article interprets digital design as a broad and teachable skillset that should be available to all students, rather than niche specialization. This is the foundation of an inclusive theory for education that bridges between design and the digital.

**The Myth of the Digital Native**

The introduction of the computer in architecture changes both what and how architects design. It introduces both new capabilities and new sources of bias and error.

Therefore, it is necessary for architectural education to address and teach specific ways of designing with the computer – not how to use software or operate machines, but how to *design digitally*.

Unfortunately, the distinction between using and designing with a computer and the need to explicitly teach digital design are not commonly held beliefs among architectural educators. Some assume that students can and should teach themselves software outside of the classroom while classroom instruction should be devoted to teaching design. The myth that students, as ‘digital natives,’ who are self-regulating and superior to their instructors when it comes to learning and using technology is harmful and untrue.

The term digital native derives from a series of articles written by the technologist Marc Prensky during the early 2000s. Prensky describes the generation of young people born since 1980 as “digital natives” due to what he perceives as an innate confidence and intuitive ability in using new technologies such as the internet, videogames, mobile telephones and ‘all the other toys and tools of the digital age.’ Enrique Dans counters Prensky’s claims: ‘Simply being born into the internet age does not endow one with special powers. Learning how to use technology properly requires learning and training, regardless of one’s age.’ Although scholars have similarly debunked the claims of Prensky and others, the myth of the digital native, and the failure to recognize the importance of high-quality technology instruction, persists.

While students are often considered digital natives, they tend to be digital orphans, who are lacking in any behavioral models to copy or criteria for understanding digital tools. Beyond basic fluency, architectural instructors are uniquely positioned to model substantive content creation and healthy critical thinking about these technologies. By perpetuating the myth of the digital native, architectural education is missing the opportunity to establish strong pedagogical foundations from which future designs and digital advancements will emerge.

The what, when, and why of architectural education is intrinsically relevant to the discipline and practice of architecture. Digital media and technology are inseparable from contemporary design today and therefore must
be more articulately discussed as an educational agenda. The way digital design is taught (and not taught) has ramifications upon the discipline and practice of architecture: how buildings are designed, documented, and disseminated. Therefore, teaching methods should not be dismissed as pedagogic issues but rather discussed as essential questions of the discipline.

The authors present two proposals that attempt to address these issues at both the tactical (instructional methods) and strategic (curricular) levels. Respectively, the first proposal describes the teaching of soft skills for digital design and the second uses Bloom’s Taxonomy as a method of developing learning objectives for digital design instruction. These proposals represent two examples of how educators can bridge the gaps that commonly exist between design teaching and technology teaching.

Soft Skills and Fostering Learning Habits

An important step in advancing the discipline of digital design is to define digital skills which are learnable rather than innate. One way to encourage more critical thinking about technology is to teach generalizable skills that apply across a range of technologies and to train students to recognize when to apply them. This stands in direct contrast to other forms of digital technology instruction which tend to emphasize skills with a specific piece of software or machine.

Computer use in architecture is often discussed and taught as a series of technical or ‘hard (as in absolute)’ skills. In contrast, ‘soft’ skills are related to emotional intelligence, attitudes, habits, and interpersonal relationships. An example of a soft skill is resourcefulness: being inclined and able to find alternate solutions to a problem, rather than giving up or deferring responsibility. In this manner, soft skills influence the ways that an individual applies technical skills to achieve goals, such as a design. Learning soft skills has been related to improved employment outlook and better job performance. Professions such as business and information services have cited employees’ lack of soft skills as one of the primary reasons why projects fail. Thus, for students, developing soft skills is equally as important, if not more important, than learning technical skills. This is because soft skills can be reapplied to changing technology, whereas hard skills may fall away as technology changes. It is the re-application of these skills that makes them relevant beyond the beginning education of architects and deeply important to the discipline.

The influential Boyer report on architectural education concluded that: ‘[A]rchitectural education is really about fostering the learning habits needed for the discovery, integration, application, and sharing of knowledge over a lifetime.’ Soft skills are the learning habits Boyer references and as such must be taught rather than assumed to be pre-existing skills. This also extends to those soft skills which relate to digital design in architecture.

While technology has rapidly become more accessible to more people, its benefits are not evenly shared. Architectural education must recognize that university students are not comprehensively or consistently trained in digital technologies when they arrive on campus. This is exacerbated when less privileged students arrive less digitally skilled than students from economically privileged backgrounds. By not addressing these inequalities, architecture schools end up perpetuating disparities through education.

Figure 1: Knowing how to operate a smartphone does not necessarily make one an effective computer user. Photograph by authors.
Digital Soft Skills

While traditional soft skills such as conscientiousness and empathy are helpful for architects, digital soft skills – introduced in this section – apply specifically to the tools and processes used in digital design. Digital soft skills support students as they are learning digital design and, later, help students apply technical skills successfully and with sophistication and to adapt to a rapidly changing technologic landscape.

Digital soft skills differ from traditional soft skills because they take into account the particular challenges of computing and digital machinery. The special attributes of digital tools that make them powerful: symbolic logic, abstraction, and automation, can invite cognitive biases when designers operate those tools simplistically, at face-value (i.e. using a computer like a cell phone, a pencil, or a typewriter). To best leverage the unique characteristics of the digital, architects must adapt their thinking, expectations, and habits, as analogue inclinations can interfere with working effectively with digital tools. Even those who work with digital tools frequently need to learn digital soft skills, as they may have developed bad habits and misconceptions over time. Merely using digital tools is not enough to cultivate a mindfulness of the limitations and opportunities of the medium and one’s responses to it.

To cite an example: digital tools are often ‘black boxes’ with complex layers of interrelated procedures that make it difficult for users to be aware of what they are doing and how their software operates. Users expect simple cause-and-effect relationships between their operations and the results on a screen, when the reality is that many “hidden” processes are at work and can affect the outcome of an interaction.

Students who lack the digital soft skills to understand and respond to seemingly opaque technologies often have a poor attitude when faced with computer problems and may spend their time in unproductive ways trying to ‘hack’ solutions to technical problems. This affects not only the quality of their final designs, but their outlook on technology in general.

Figure 2: The top diagram demonstrates a curriculum where design (architecture) and hard skills (technology) are typically taught in parallel. The bottom diagram demonstrates that soft skills (learning habits) create the bridge between design (architecture) and hard skills (technology). Over time these skills become mutually supportive. Diagram by authors.
Digital soft skills are similar to traditional soft skills in the way they affect how students apply technical skills. They are the bridge across the gap that often exists between design skills and technical (hard) skills like digital methodologies. However, very little time, if any, is given in architectural curricula to the explicit cultivation of digital soft skills.

**Samples of Digital Soft Skills**

The following list is a representative sample of digital soft skills which support digital pedagogy: communication skills, adaptability, time management, and digital hygiene.

1. **Communications Skills**

   Communicating clearly with others is a critical set of soft skills for architects, particularly when using digital tools. For instance, many students have never been explicitly taught how to ask a question via email: to provide necessary information and files upfront, anticipate follow-up questions, and to communicate their expectations for resolution. This is important not only professionally, but especially when trying to learn or fix something like a new piece of software.

2. **Adaptability**

   Adaptability is resiliency in response to imperfect tools and a field constantly in change. Digital designers should work with the understanding that failures are to be expected, while being empowered to seek alternatives. They must also update their skills and abilities often while remaining critical users of technology.

3. **Collaboration** - The ability to work with others digitally, particularly at a distance. Some examples of this include establishing and maintaining representational standards and version control for files.

**Authorship** - This is the ability to understand digital intellectual property and to distinguish between resourcefulness and plagiarism. This notion of authorship becomes increasingly important when the line between programmer and designer is blurred by the use of digital tools. Of particular note is the downloading of code or Grasshopper definitions which are then deployed as design generators.

**Support** - Architects should be able to seek, locate, and pursue support for software and technical issues, many of which might exceed the abilities of the instructor or the support offered by an academic institution. These skills include asking fellow students, contacting the software maker directly, and using the Internet as a resource.

**Autodidacticism** – The ability and inclination to teach oneself (quickly) is a valuable skill for designers. This includes planning and scheduling regular time to learn and a recognition of common concepts and methods.
shared between tools, which can make learning more efficient.

**Conversion** – An effective strategy for error recovery is knowing how to share data between several types of files and programs. It is important to also note that many computer programs are able to convert various file formats and often have similar procedures.

### 3. Time Management

Digital design projects in architecture are often complex, involving many different programs and machines, as well as human team members. Some of these elements can be hands-off (such as rendering) or very hands-on (supervising CNC fabrication). Part of completing them successfully is knowing the workflows involved and having a sense of their coordination and time requirements.

**Estimation** - There is a common misconception that technology makes design faster and easier. It takes experience and skill to determine the full amount of time needed to complete a digital task or processes (e.g. milling, printing, rendering).

**Sourcing** - The ability to identify the most effective tool and process for the development of the idea and in relation to the time available for production. This requires understanding the different elements of digital production such as the difference between a raster and a vector.

**Preparation** - Plan for contingencies and alternatives. Assume some things will inevitably not go as expected and know the options available.
Scheduling - Develop internal deadlines, realistic calendars, and skills for planning and implementing a multi-step process. For instance: development of a digital file for fabrication, then fabrication, then post-production.

4. Digital hygiene

Digital hygiene refers to the good habits of caring for equipment, computer hardware and software as well as preventing and recovering from errors.

Organization - Maintain files in a structure which is both navigable and searchable by users.

Backups - Create a backup routine that is an embedded part of the digital process (cloud, physical media, & storage). This also includes knowledge and use of software auto-backup and recovery. Keep at least one physical backup off-site.

Clean-up – Regularly sort, store, and purge project files to manage storage and make important files easier to locate.

Teaching Digital Soft Skills

Successful students may demonstrate behaviours and habits similar to those presented in the last section. Therefore, it is often assumed that soft skills are character traits rather than teachable attributes. However, this is not representative of how the majority of students approach digital design. The very notion of ‘soft skills’ implies that these behaviours and habits can be taught to students who need them.\(^\text{14}\)

Another common argument is that soft skills are best learned in the workplace. While the workplace presents a professional context, it does not offer the same opportunities for focused learning as design school. Moreover, one of the reasons for learning soft skills is to make one more competitive in finding employment. Students should have a sense of how these skills translate into practice before they enter the market or transition into related fields where soft skills are still applicable.

Supporting a new habit which a student does not create themselves requires helping them understand its meaningfulness. It can be easy to dismiss soft skills out of hand because they might seem to be obvious or less interesting than learning technical skills. However, soft skills precede the development of technical skills. For this reason, it is important for the instructor to communicate why new strategies and habits support design development.\(^\text{15}\)

Investment begins by identifying the soft skills in question and explaining to students the value of the skills within design and production workflows. To be most effective, those values should be immediate and goal-oriented. Although it is true that developing soft skills can help a student get a job in the future, explaining to a student (for example) how organizing their files saves them time and reduces errors on their current project is less abstract and applies to their current situation. To give context to this time-saving practice, file structures should be discussed as integral to design processes rather than tangential. Helping students understand the gaps in their present abilities and how learning soft skills can help close those gaps and improve designs is the first step toward effective habituation.

At the same time, teaching soft skills is most effective when integrated with hard skills teaching and preferably in the context of a design project.\(^\text{16}\) An instructor can introduce soft skills where they naturally occur within design and production processes. For example, using an error that students commonly encounter to teach search, problem-solving, and communications skills. Relevant material like this helps focus student attention while a legitimate context helps them retain and access what they have learned later.

Figure 6: Example of a backup protocol. Soft skills enable students to feel confident that computers will fail and that they are empowered to seek alternatives. Photograph by authors.
Demonstrations can be more effective when they are supported by teaching materials that help organize knowledge for students. A simple check-list, for example, can help students remember how to organize a digital group project. Once students have mastered the soft skills involved, the student will not need the scaffolding provided by the list. However, if the student makes a mistake or needs to refresh their learning later, the list provides a useful reference and a prompt for activating digital soft skills. Externalizing implicit practices and helping students focus on relevant information and methods improves the effectiveness of soft skills teaching and integrate these skills into the design process.

Delivering soft skills in class benefits from a coaching approach. Because the goal is to change student attitudes over time, rather than delivering information or procedures, a ‘one and done’ demonstration is not an appropriate teaching style. With coaching, the instructor discusses the advantages of a skill (creating investment), then models the behaviour while explaining to the student what they are doing and why. This last step is important because students need to understand when to apply a skill as much as they need to know the technical operations involved.

Next, students demonstrate the skill and receive feedback from the instructor on their performance. This is followed by more practice and feedback over time and in concert with other skills to approximate holistic design activities. The goal of coaching is to cultivate not just practice but deliberate practice over time – making the student aware of their own actions and motivating retention and refinement. This creates deep and lasting learning: the ‘learning habits’ championed by Boyer.

Formative assessment techniques, which encourage personal reflection, timely feedback, and student response are useful support for the ‘coaching.’ Many courses emphasize the final artefact and never look at the files involved. The studio or classroom offers the opportunity to do just that. Reviewing files is critical so the instructor can observe attributes such as organization, efficiency, and other procedural nuances.

Lastly, in order to properly cultivate habits, soft skills should be reinforced in the studio and lab even when they are not being formally taught. Instructors should be mindful and consistent in their own habits, demonstrating modelled behaviours in their personal actions. For example, an instructor’s demonstration files should be well-organized to set a good example for the students. Student interactions should also emphasize consistent behaviour. If a student asks for help with a tool, for instance, the instructor should evaluate how the student asks questions and replay the scenario with them while making explicit the strategies involved. Learning should be embedded in the classroom experience and design process. It must be an iterative and continuous practice, not merely an exercise.

**Pedagogical Alignment and the Value of Digital Design**

Although digital soft skills bridge an important gap in how students learn digital design, effective teaching must also provide a framework for how students approach learning and apply their skills. Digital design instruction in higher education is often misunderstood as merely teaching technical skills. However, in order to be used successfully, those technical skills must be supported by knowledge and sound judgement.

The issue facing architectural education is that the goals of learning digital design are often too superficial and ill-defined. Merely acquiring skills is not enough. By reflecting upon the nature of learning itself, architecture education can realign its pedagogy and understand the value of digital design. Towards this end, one of the most significant advances made by educational research in the past 20 years has been to redefine the goals of learning. Decades ago, before the development of contemporary learning theories, schools emphasized developing core skills such as reading and memorizing information such as dates and facts in a history class. The implicit assumption was that this level of learning was sufficient for students to write reports, solve problems, and produce other sophisticated applications of literacy. However, while many students could demonstrate ability at, for instance, providing the correct solution for a specific type of word problem, educational researchers found that students rarely understood what they had learned, nor could they easily apply their skills and strategies to new contexts. The students knew their lessons by rote and adapted to succeed at their...
instructor’s tests, but they had a superficial understanding of the material. Today, although educational models and expectations have evolved, digital technology is often relegated to this type of learning.

While skills and facts remain important to learn, the goal of education today has been restated: to provide students with a foundation of deep learning and the intellectual tools to ask and address meaningful questions. In contrast to superficial learning of facts and procedures, deep learning entails knowledge of the underlying principles, domain structure, and strategies to activate skills and knowledge and apply them flexibly in a variety of conditions—particularly conditions which are different from the ones where learning originally occurred, such as the translation of design thinking from an academic to professional context. Deep learning is what most instructors would recognize as productive and transferable learning yet few courses actually achieve this new standard. Architectural studios are examples of a deep learning environment.

However, in contrast to architectural studios, the current state of digital design instruction in architecture tends to follow an educational model which does not support deep learning. Presently, much of what students learn is by rote: sequences of commands and procedures intended to produce reliable results. While students can operate software and other tools with what appears to be great fluency, a majority do not have a deep understanding of computing or digital media principles. As a result, their work tends to be inefficient and derivative. Like the school teachers in the earlier example, digital design instructors emphasize core skills for using digital tools and then expect students to apply them towards design projects. This is the reason a learning gap exists: first, students do not learn the tools with significant guidance to develop depth and rigor; second, they are not taught explicit strategies for applying digital methods to design tasks. Students often fail to develop an understanding of digital design methods because the pedagogy is not aligned with the goal of deep learning. This leads to a frequently cited criticism of digital design: work which is repetitive or un inventive because students are grappling with technology rather than controlling it. In this case technology controls the learner, not vice versa.

Here we assume that such a goal is recognized in the first place. Learning digital tools is often seen—by students and faculty alike—as mere technical skill ing rather than a way of thinking about design. The organization governing professional architectural accreditation (NAAB) in American schools uses a set of learning criteria that specify Ability and Understanding. However, this set of criteria does not address digital design with any specificity. There is no agreement upon the value or content of a digital design education, and so student abilities can vary widely from school to school, and within academic units.

The educational model of the design studio is unique in its approach because it has many elements which contribute to the production of deep learning, such as opportunities for synthetic learning, active learning, complex problem solving, and self-reflection and critique. This is precisely the kind of approach that would benefit digital design education. Unfortunately, the architectural design studio is often seen as one type of learning, while digital design, which is thought of as mere technology, is seen as another. This disconnection is due to a misunderstanding about digital design due to a lack of clearly-defined and shared pedagogical goals. The present situation in education has come about because the implied goal of digital design education is mere tool operation (which does not require deep learning) when the expected outcome should be increased agency and sophistication of design ability. One way to address the problem of pedagogical misalignment is to develop learning objectives for digital design. Learning objectives have the benefit of being a structured, well-understood, and research-based approach to curricular development. This method informs clarity and represents an explicit way to connect the goal of deep learning with pedagogical execution.

Bloom’s Taxonomy

A useful tool for developing better learning objectives and thereby digital education is Bloom’s taxonomy. The taxonomy is a hierarchical framework intended to help instructors coordinate their planning and assessment using a common language. It represents the process of learning from acquiring simpler to more sophisticated thinking skills. The general idea of Bloom’s taxonomy is that lower levels of cognition
Figure 7: Bloom’s taxonomy was first introduced in 1956 and since then has seen widespread use in instructional design. A revised version was issued in 2001, which changed the levels from nouns into active verbs, added the knowledge dimension, and placed creation (synthesis) at the top of the hierarchy of cognitive process. More recently, Churches created a ‘digital’ version of Bloom’s taxonomy that updates many its application to computing activities. Diagram by authors.
support higher levels. For instance, one must understand the difference between different methods of digitally constructing a 3D surface (comprehending) before choosing which kind of 3D surface to use (applying).

In its revised form, Bloom’s taxonomy lists six levels of cognitive processes:

1. **Knowing**: memorization and factual recall.

2. **Comprehending**: understanding the meaning of facts and information.

3. **Applying**: selection and correct use of facts, rules, or ideas.

4. **Analyzing**: breaking down information into component parts.

5. **Evaluating**: judging or forming an opinion about the information.

6. **Creating**: combination of facts, ideas, or information to make a new whole.

A more recent addition to the discussion of the taxonomy is the inclusion of types of knowledge. Anderson and Krathwohl addressed criticisms of the taxonomy by recognizing that not all knowledge is equal in complexity and that knowledge tends to be developed from concrete (facts and concepts) to abstract (procedural) and finally to knowledge of one’s own cognition (metacognitive). In concert with cognitive processes, the knowledge dimension of the revised taxonomy enables a more nuanced discussion of learning objectives. For instance, under the newer version, the taxonomy does not progress and stop with creating, but also includes reflective feedback loops between making and learning. Many of these processes can already be found within the studio teaching model.

Bloom’s taxonomy is not a prescription for every course to follow. However, it is used here to move forward a conversation about how to structure the digital education of architects. For example, one could design a course with at least one learning objective from each cognitive level. Depending upon the skills required, some levels may need additional objectives. Students with different abilities may be able to begin learning at higher levels. The value of the taxonomy is not that it represents exactly how learning works or that it tells instructors how to teach, but rather in how it helps to organize and align pedagogical thinking.

Educational frameworks like Bloom’s taxonomy are not in common use in architectural education. The reason for this is unclear. However, for those developing or revising architectural curricula, having access to a set of learning objectives that uses a shared taxonomy can enable a dialog within the discipline. Furthermore, because Bloom’s taxonomy is recognized outside of architecture, it can help produce interdisciplinary connections with other educational researchers.

Bloom’s taxonomy helps support the goal of developing deep understanding in digital design instruction. One way it accomplishes this is by establishing the basic cognitive processes involved in learning to design thoughtfully. To see these processes organized and consider them with respect to digital design is to shed light on what is often an opaque practice. The taxonomy makes it clear that one does not just use or not use various tools, but one must understand them, choose from them, and evaluate those choices as part of a design process. In this manner, an advantage of learning objectives developed through Bloom’s taxonomy is that they promote student outcomes of greater complexity and iterative feedback loops. For example, without the proper outcomes articulated, a student might submit what appears to be an original design, but was merely applying a procedure. Bloom’s taxonomy makes it clear that creation depends as much on understanding one’s decisions (the ‘why’) as knowing the correct commands (the ‘how’ – which is often students’ focus). For can help clarify that the goal of digital design instruction is not only to learn how to use digital tools, but to apply them towards better designs and more sophisticated design thinking.

With regards to teaching methodology, the clarity of learning objectives derived from Bloom’s taxonomy can help motivate qualities of student performance which are often lacking in digital design courses, such as innovative solutions and well-crafted, thoughtful representation. As mentioned in the previous section, many learning objectives are not specific enough, sufficiently measurable, or
targeted to student’s learning level. Bloom’s taxonomy can help ensure that students are practicing the skills that they should be learning in their activities and at an appropriate level of cognition. This enables the pedagogical gap between learning digital methods and creating designs to be filled with deliberate (or mindful) practice.

Deliberate practice is another element which is often missing in digital design education. It is a recognized process through which individuals train themselves to high levels of performance. Like digital soft skills, some students do this intuitively, but it can also be taught. Research has shown that learning complex skills is most effective when students engage with tasks that are appropriately challenging, with clear performance goals and feedback, and sufficiently frequent opportunities for practice. The difference between merely making and deliberate practice is that a student monitors their progress towards a specific goal and changes their performance in response to feedback. Learning objectives assist students in deliberate practice by creating specific and appropriate performance goals which they can use to monitor their progress. This guidance directly supports the development of abilities on the highest (metacognitive) level of the taxonomy, which are crucial for sophisticated work and achieving transfer of skills and knowledge to other domains. Thus, the notion of deliberate practice stands in contrast to the disengaged ways that many students learn and use digital tools, which is often oriented towards production for its own sake rather than for quality or thoughtfulness. Introducing deliberate practice is one way for schools to motivate deep understanding and to bring craft back into discussions about digital representation.

Towards Learning Objectives for Digital Design

Learning objectives, supported by Bloom’s Taxonomy, provide valuable insight into the how and why of teaching. They promote a common language of naming educational concepts which define and enrich the discipline of digital design. The idea of a learning objective is straightforward, but often misapplied or misunderstood as bureaucratic syllabus fodder. A learning objective is a specific statement which describes what a student will know (knowledge) and be able to do (skills) as a result of engaging in a learning activity. A learning objective must have three parts: a measurable verb associated with the intended cognitive process, the necessary condition (if any) under which the performance is to occur, and the criteria for measuring acceptable performance (this is often implied). A simplistic example of a learning objective that fits this pattern is: ‘Given a set of contours the student will be able to generate a topographic model.’ The condition is having a set of contours and the implied measurement is an acceptable model. Learning objectives are focused solely on student outcomes and do not specify methods or other expectations for the teacher. They are not an attempt to create uniform classroom procedures or hinder instructor creativity through standardization. Learning objectives are useful because they help instructors with course planning and the creation of content. Furthermore, the explicitness of properly-constructed learning objectives establishes a basis for student assessment as well as the evaluation of teaching and curricula. A primary challenge of digital architecture evaluation is the lack of criteria and therefore a lack of agreed upon traits for which to evaluate whether digitally produced code, drawings or images are successful.

In this manner, learning objectives support better learning and provide a common framework for schools to organize their efforts at improving education. For this reason, many North American universities, such as Vanderbilt and Carnegie Mellon, have standardized their syllabus policies to address learning objectives. The use of learning objectives may seem obvious or unnecessary if one is only considering their use in one’s own syllabi, but in terms of disciplinary alignment, digital design instruction could benefit from the additional clarity offered.

The real issue is not that learning objectives do not exist for digital design courses, but rather that they are not often used correctly, in response to the findings of educational research. First, many stated learning objectives do not take into account the learning process for developing complex skills and thinking. As mentioned earlier, traditional digital design pedagogy tends to emphasize learning through design tasks. The tacit learning objective of most activities, ostensibly, is to design something via digital methods. However, this
does not acknowledge the steps involved to prepare students for design, such as learning about the tools, practicing methods, comparing and selecting methods, etc. These skills and knowledge are implied by the goal of designing. However, by not stating this explicitly and assuming that students will learn on their own, the instructor might neglect teaching and assessing the constituent skills and knowledge that students need.

When developing learning objectives, it is important for digital design instructors to acknowledge how learning occurs as a developmental process. Creativity and autonomy, abilities exercised in design work, are higher order thinking skills. Higher order thinking is dependent upon requisite technical skills and other cognitive resources. As such, these activities may not be beneficial learning experiences for beginner and intermediate students. Research shows the importance of matching learning objectives to student level. Novices benefit from direct guidance in basic skills and knowledge, while objectives for advanced students should emphasize synthesis and independence.

Second, many learning objectives for digital design instruction conflate activities and goals with learning outcomes. A goal is a statement of the overall intended outcome of a learning activity or course. Learning objectives are specific achievements which contribute to the goal. For example, a course description that says ‘students will be exposed to digital fabrication technologies’ has presented a goal, but not stated a specific, measurable outcome. Likewise, a statement like ‘students will fabricate a small-scale physical model’ describes an activity, but does not provide enough information to discern what students are supposed to learn from the activity or what determines successful learning from the activity. A learning objective that addresses these issues would be: ‘students will use GIS data to generate a small-scale physical model using appropriate digital fabrication techniques.’ This objective presents a condition (GIS data), an outcome (the model), and assessment criteria (are the techniques appropriate? is the model correct?). Understanding the learning objective helps define the cognitive skill level of the activity and the appropriate assessment. For instance, if the objective was to learn about computing concepts, issuing a quiz with questions about procedures would not be a helpful measurement. To facilitate effective instruction, goals, activities, and learning objectives must be aligned with one another.

**Discussion of Learning Objectives**

The challenge of making claims about design pedagogy interventions, such as soft skills, is proving their effectiveness. In educational research the difficulties of empirical measurement in traditional subjects like math and reading are well-known, but the challenges of demonstrating the impact of an intervention upon design outcomes – which are not easily measured or quantified – make this task even more burdensome and its conclusions unreliable. For this reason, there is no accepted model for proving the effectiveness of design pedagogy. What is more important and perhaps easier to ‘prove’ is that well-articulated digital soft skills create a framework and a platform where technology can be used expansively and in unique ways rather than reductively and repetitively. The value of digital soft skills is to suggest a replicable model which remains relevant and useful for students as technology changes.

With regards to learning objectives, their greatest value is not necessarily what they add to a syllabus, but rather how they prompt a larger conversation about educational and professional values and standards – a conversation which is necessary for establishing a discipline and then moving it forward. Creating learning objectives for digital design in architecture exposes many implicit assumptions about what faculty believe about learning and the role of computing in the studio. At the same time, discussing learning objectives is a provocation towards architecture schools to consider digital design as more than merely learning to operate tools and software (activities which are not themselves valid learning objectives) and to instead connect these practices to design thinking and the development of architectural designs.

Bloom’s taxonomy assists in framing a more constructive discussion about learning to design digitally by offering a structure of cognitive accomplishments for students. This helps move architectural educators away from frameworks derived from intuitions about pedagogy and towards established theories and research into educational psychology and
learning cognition. Instead of teaching and learning digital skills and knowledge through a hierarchy of the tool’s features or increasing complexity, Bloom’s taxonomy foregrounds processes of remembering, thinking, and judgment. These objectives are more closely aligned with deeper understanding and integrative mastery. This type of learning is precisely the antidote to the kind of superficial engagement one often finds in architecture schools that prompts negativity towards the use of computing in design. In the future, educational theory in architecture could benefit from a more specific taxonomic model, but at the moment, Bloom’s taxonomy offers a useful framework for envisioning and discussing alternatives to traditional digital design instruction.

The purpose of reflecting upon learning objectives for digital design in architecture is not to produce a definitive list of what students ought to learn. Learning objectives are written for specific curricula, student needs, and faculty interests. They are useful because they provide a clear definition of expected outcomes and which becomes a point of dialogue. In order to evaluate something, it first must be named. Through evaluation and discussion, a discipline develops. When Bloom created the learning taxonomy, this was the goal. Not to explain or lay claim to how students must learn, but to provide a shared structure so educators could compare their approaches. In a similar manner, creating and sharing learning objectives for digital design instruction can produce a more organized dialogue about how to align the use of digital tools with the core values of architectural education and the development of the discipline itself. In doing so, it may be possible to move away from strictly hierarchical models (such as Bloom) and into more networked or non-linear theories of learning. Nevertheless, the development of a more coherent set of evaluation criteria in digital education will increase the rigor of conversations about the future of digital design in architecture.

**Conclusion**

While digital design skills are critical for 21st century designers, architectural education must also recognize and deliver more than technical proficiency. Working creatively and effectively with computers, digital fabrication machines, and other devices requires a new set of workflows and adaptations to professional behaviours. Traditional learning habits and educational goals have not been updated in response to these changes in technology. We propose that research-based educational methods, such as soft skills and learning objectives for digital design, can help schools to bridge these gaps.

The proposals and critical reflections shared in this article are intended to support the explicit cultivation of knowledge, abilities, attitudes, and habits tied to technology and architectural education. Contemporary architectural pedagogy and curricula must address the difference between teaching students to merely use technology and teaching students to design with technology.
REFERENCES


16 Barbara Y. White, and John R. Frederiksen, “Inquiry, Modelling, And Metacognition: Making science accessible to all students.” Cognition and Instruction, 16(1), (1998), 113-118.


