Effective systematic design of instruction for teaching technology can be achieved by starting with end goals and working backward to building assessments, content, and context. This is so that instruction functions as a bridge from what learners know and can do, to what they need to know and do, to achieve the goals of instruction.

This approach takes the technology teacher through the process of designing technology instruction, from cognitive and performance goals, through decoding goals into cognitive and sensorimotor tasks, to building instruction. The target product is technology instruction that functions systematically as a bridge from where learners begin, to where they need to go. Learner motivation, communication, assessment, and feedback are implemented as systematic, integrative, and iterative components of the whole design process.

Technology educators face special challenges in a content area of tremendous complexity and fast-paced change, with a broad range of learners’ prior knowledge and experience. Thus, technology education is a field that can benefit significantly from the systematic application of principles and methods of instructional design (Alessi & Trollip, 2000; CTGV, 1996).

Instructional design is a process useful for conceptualizing instruction to account for its complexity and context. Both formal and informal models and
methods of instructional design are used by teachers and designers (Clark & Estes, 1998; Reigeluth, 1999; Zook, 2001). One central principle of the field is that design of instruction begins at the end. That is, effective planning and development begin with the target goals of instruction (what is to be achieved) and work backward to build a bridge from what learners know to what they need to know (Dick, Carey, & Carey, 2001; Smith & Ragan, 1999).

This “start at the end” methodology can be expressed as a series of steps such as the following:

1. Know and articulate goals;
2. Decode goals into cognitive and performance subgoals and tasks;
3. Build assessments from end goals and subtasks (where learners are going);
4. Analyze learners’ relevant prior knowledge, skills and experience;
5. Build instructional tasks, content, communication, and context as a bridge from prior knowledge, skills, and experience to goals;
6. Build in learner needs (affect, motivation, preferences) at all phases.

There are many and varied types of technology instruction, instructional environments, and target goals. I will illustrate these steps and principles using basic website design. Learners from elementary school to graduate and professional schools are learning to design websites, and their needs can be addressed using the same practical approach.

Step 1: Know the Nature of Goals and Articulate Them

In general, goals of instruction fall into two broad categories, cognitive (knowledge) and performance (task) goals. Cognitive goals identify what learners should know or understand at the conclusion of an instructional unit, and performance goals identify what they should be able to do (Smith & Ragan, 1999; Mayer, 1999). The two are integrally linked, since perfor-
Association for the Advancement of Computing In Education, 12(3)

Performance goals function as demonstrations of learners’ cognitive goal achievement (Anderson, 1993; Bransford, Brown, & Cocking, 1999).

In website design, for instance, if students understand the conceptual relationships between linked pages and know the technical process of creating links in an html editor, they can demonstrate it by making a working link from one document to another within a website. However, the performance goal is dependent on achievement of both cognitive goals; (a) understanding the concepts and (b) knowing the technological process. The student who understands the relationship but not the linking process can’t demonstrate the knowledge by this method. One who knows the technical process but not the conceptual relationship might “clunk out” a link, but not understand, despite apparent “success” on the test. Neither learner has achieved the goal of effectively learning both the knowledge and skill.

**Step 2: Decode the Goals into Cognitive and Performance Subgoals and Tasks**

Decoding goals and tasks occurs at two general levels, domain (general or macro level) and goals/tasks (micro level). Within any domain or skill area there are generalizable characteristics of skill and tasks (Mayer, 1999; Sternberg & Wagner, 1994). Understanding these domain attributes enables teachers to view the cognitive features of skills globally. Cognitive demands of the domain become “givens” for specific goal decoding.

In the case of web design, the instructor should consider domain-level cognitive demands both for general use of technology-based software, and the specific sub-domain of web design itself. Beyond these general areas, the instructor moves to decoding specific goals and tasks for the instructional unit.

Technology skills for software users share some generalizable cognitive characteristics. They tend to:

1. Be comprised of actions dependent on visual cues (e.g., dialog boxes, menus, paths);
2. Demand skills that are sensorimotor in nature (e.g., mouse clicks, keystrokes);
3. Require complex responses to stimuli, represented as recall + action (recall is cognitive response, and action is motor response) and sequence of actions is critical to success.

In more concrete terms, a software user generally sees a dialog box or prompt and has to respond with an action that is dependent on remembering some related information about how the interface or system works. Even this admittedly simplified degree of analysis reveals some domain-level skills prerequisite to the mastery of most specific tasks.

The knowledge and skills required to develop facility in doing the type of tasks described above include at least:

1. Familiarity with the interface (e.g., where to find various features on the screen);

2. Understanding sequencing of actions (e.g., what to do in what order);

3. Fluency with the system and feedback (e.g., the ability to see and understand what indicates success, failure, or that another action is required);

4. Flexible transfer of skill to novel tasks (e.g., see essential elements despite distractors).

For most systems and uses, flexibly transferable skill is what divides novices from experts (Bransford et al., 1999; Brown, Collins, & Duguid, 1989; Druckman & Bjork, 1994). It also makes the difference between failure and success. Each time it is presented, a prompt may look a little different or appear in a different context. An expert will see the essential elements and understand how to respond.

Besides the generalizable cognitive demands of any technological tool, web design requires specialized planning and reasoning. First, learners need to develop long-term planning skills that enable site structure and organization. Second, they need a clear understanding of the relationships between document types (e.g., when graphics and media are embedded in documents and when they are linked to documents). These relationships have important implications for site design and management.
Beyond the global cognitive and performance goals of the skill area (e.g., technology & website design), each individual goal should be decoded into its component tasks and subtasks, to be sure that the teacher has made a comprehensive cognitive and performance task analysis (Morrison, Ross & Kemp, 2004). This process, the task analysis, is the key to designing instruction. Decoding domain skill and instructional goals into cognitive and performance subgoals and tasks enables teachers to plan well and anticipate students’ possible misunderstandings and special needs. As a result they can teach and coach better.

For making a working link from one document to another within a website, the performance goal might be articulated like this: “Students will successfully build an index page with text and at least one graphic and link it to another document in the site, using the html editorial software.” For this goal, the nonnegotiable elements of assessment (with their requisite subtasks and companion tasks) would be:

1. “build index page” (create, save, open, close, replace, & name documents);
2. “with text” (word processing basics, formatting, editing);
3. “and at least one graphic” (open graphic, file recognition, or digitize/scan);
4. “link it” (make link doc-to-doc, replace doc, preview in browser & test link);
5. “another document” (html or different document type?);
6. “with html editor” (familiarity with interface, know basic functions).

**Step 3: Build Assessment From Goals and Subtasks**

Having articulated and decoded goals, the teacher can craft an assessment consistent with them. Keywords of each goal identify the nonnegotiable elements of an assessment, and other possible elements of the assessment should be made negotiable so learners can exercise choice and preferences.
An assessment of the learners’ ability to demonstrate these skills by performing the task of building and linking documents within a site would focus on *nonnegotiable* elements. Negotiable elements would include the parts of the task not specified in the goal this assessment is designed to measure (e.g., topic, length, and content of the text; graphic type); link type (text, graphic, hotspot?); embellishments to the page (challenge for creative & skilled students); any font types and sizes (or were there separate goals to cover best practices for these?).

Any negotiable element of one goal can generate separate goals and sub-goals. For instance, when the teacher decodes the phrase “with graphic” from this goal, he or she realizes the need to decide about access to eligible graphics. The teacher could:

1. Provide a folder full of graphics for learners to choose from;
2. Have students download preformatted graphics from the Web;
3. Teach students to digitize graphics as a prior activity and use them here.

There are tradeoffs for each of these options. Having decoded the goals, the teacher can view options systemically and design the instructional environment as appropriate.

Designing an assessment from this decoded goal involves crafting an assignment that reflects the measures of competence on each key term in the goal. Essentially this means assigning the goal as task with its relevant limitations and choices as indicated by the analysis of nonnegotiable and negotiable elements and relevant resources identified.

Designing the assessment includes also designing a scoring method (i.e., criteria, rubric, range, and scale) that reflect the task and its reasonable, relevant measures of competence (Frisbie, 2000; Nitko, 1996). The scoring component of the assessment should assign value to all of the key elements of the goal-based task (nonnegotiables) but not assign score value to elements of choice (negotiables), so that there is no “hidden agenda” in the assessment (Nitko; Pressley, 1995). If there is “extra credit” for optional tasks, then that should be explicitly stated in the assessment instructions (Frisbie, 2000).
For example, the assignment for the create-and-link goal might read something like this:

**Create a simple website**

Create an index page (homepage) for your website. Include some text and at least one graphic image (a picture or piece of clip art). Then create a second page for the site. Make a link from the homepage to the other page you created. Test the link to make sure that it works.

Make both pages as html documents in [web design software]. Make your link from any one line of text on the homepage to the file of the second page. Your graphic may be any (gif or jpeg) image from [file name or source]. [Add any options here, such as what else pages may contain & any “extra credit” features.]

This assignment is worth [x] points, based on the following criteria . . . [fill in specific assessment criteria].

Including specific parameters and criteria help learners understand the purpose of the assessment, reduce repetitive “what if” and “can I” questions, and free both teacher and learners from suspicion of subjective judgment, so they can focus on the important work of achieving goals and interacting in meaningful ways (Frisbie, 1999; Nitko, 1996).

**Step 4: Analyze Learners’ Relevant Prior Knowledge, Skills, and Experience**

By now the teacher’s goals are clearly articulated and decoded, assessments are aligned with goals, and requisite tasks have been analyzed. The teacher knows where students are going, and now must figure out where they are beginning (Morrison, Ross & Kemp, 2004). Technology instructors can use learner analysis of prior knowledge, skill and experience to create a profile of the learner group.

Teachers should try to assess all three components (knowledge, skill, and experience) of any learner group, to develop the richest understanding of learners’ needs and preferences. The instructor can use the tasks reflected in the goal set to frame questions. The most basic questions, for each subtask are:
Association for the Advancement of Computing In Education, 12(3)

1. Do these learners know this (do they have the knowledge required)?

2. Can these learners do this (do they have the skills required)?

3. What kind and how much experience have these learners had in this skill or task area? (this question identifies experiences to build knowledge & skills on)

For each question with regard to each subtask, the teacher should consider the range of learners in the class or group, by trying to answer these question in ways that describe: the most able or most skilled learners, the least able or least skilled learners, and the middle ability or typical learners. The sum of these answers creates a profile of the learner group that instructional designers call a learner analysis.

From this information, the teacher can decide at what level and pace to design the instruction (e.g., teach to the middle and offer challenges to more skilled and individually remediate less skilled; or teach to the less skilled and let more skilled have free time when done).

Step 5: Build Instructional Bridge From Prior Knowledge, Skills and Experience To Goals

At this point the technology instructor knows: the components of tasks required to achieve goals, what learners know, what they lack, and the range of learners’ abilities and needs. The core of designing instruction is to build a bridge from what is known to what must be known—bridging the gap. For each lack of knowledge and skills, the teacher needs to design:

1. Content Access (a way for learners to receive, develop, or discover the necessary information);

2. Practice Opportunities (events that enable learners to interact with the content in meaningful ways);

3. Progressive Assessment (opportunities to test their abilities against the standard of expertise or competence);
4. Feedback (individualized and group information about how they are developing toward the goals of instruction).

Conceptual understanding is a key to long-term retention and transfer (Druckman & Bjork, 1991, 1994; Greeno, Collins, & Resnick, 1996). Conceptual understanding can be facilitated by providing or helping learners develop concept models, mental models that identify the structural and functional relationships between elements within the system (Hardré, 2001). For effective concept knowledge of a technology system, learners need an understanding of at least:

1. How the system is structured (essential components);

2. What it does (essential functions);

3. How tasks are organized (essential features).

Rushing to keystrokes and mouse clicks without building concept understanding can result in sets of disjunct and faulty cognitive linkages (misunderstandings of the system) (Bransford et al., 1999; Hardré, 2001; Moore, 1999). Disjuncts produce confusion, disorientation, and frustration (Druckman & Bjork, 1991, 1994). Faulty linkages produce misunderstandings that lead to further misunderstandings, which are more difficult to remediate than initial lack of understanding (Anderson, 1993; Pressley, 1995). Taking time to build concept understanding for technological system users saves overall instructional time and resources, and can reduce frustration for both teachers and learners (Alessi & Trollip, 2000; Hardré, 2001).

Concept understanding should be built incrementally, generally in one of two ways (depending on the system): simple-to-complex components, or global-to-local elements (CTGV, 1996). In application to a technology system, these two approaches are nearly identical in practice. Using the example of website design, Figure 1 shows a simple concept view of the system, identifying only its most global components without detail.
Figure 1. Global Model of the website design and publishing system

It takes only a few minutes to be certain that learners have a general mental picture of the relationship between their desktop computers and the Worldwide Web (or any larger networked system). This general picture may seem so clear as to be obvious, but recalling the familiar and obvious anchor learners’ encoding, retention, and retrieval of the new information that is more complex and less familiar. From this clear and simple picture, the teacher can build concept understanding of the functional relationships between systems and sites the learners will create. These functional understandings serve as rationales for the way learners need to design websites so that they are useful to and usable by others who access the Web. Figure 2 shows a more complex model illustrating functional and design components of sites posted to the Web.
Figure 2. Detailed Model of the website design and publishing system

Visual details such as line directions and thickness, spatial relationships, and line and text color are cognitively important in a concept model. A prominent text color could be used for the desktop site files and the arrows indicating their locations, to unify them in learners’ minds. Different colors could be used for lines designating external-to-site links and internal links, to discriminate between them. The upload arrow is thicker than access arrows to emphasize moving the whole site content up to the server, while access arrows are dashed to signify access to website information rather than actual movement of content. The upload arrow points to the general web-space as that is the locational target of the upload function. The browser access arrows point directly to the index.html page in each site, since that is the site entry point for browsers. External link lines point to other internal documents, as links can go anywhere in a site for which the designer has the precise universal resource locator (URL).

The teacher can prepare a concept model in a presentation software (e.g., PowerPoint©, Acrobat©) and display it interactively or incrementally, or print it to an overhead. It can be replicated in a paper or electronic document for learners to review as needed. A static model can be prepared in
as little as 5-10 minutes in a word processing system’s draw tools or with a graphics toolkit. Presenting concept takes just a few minutes of preparation and instructional time, and the investment can improve learners’ conceptual understanding, contribute to ongoing learning, reduce confusion and frustration, and enhance development of skill-based competence as well as long-term retention and transfer.

When teachers present new skills or troubleshoot tasks, they can use the concept model to develop reference points (e.g., “We’re here in the process, doing this part, and next we go here...”). Troubleshooting tasks by helping learners see how what they’re doing relates to what they just did and what they will do next helps them chart their own progress and make rich cognitive connections that serve them for retention, recall, and transfer of knowledge and skills.

**Step 6: Build in Learner Needs for Affect, Motivation and Preferences At All Phases**

Affect and learning are deeply interconnected (O’Rorke & Ortony, 1994). All learners have affective and motivational needs and preferences about the way they learn and interact (Deci, 1996; Reeve, 1996). These needs and preferences have major implications for how will they learn and can use new skills (Pintrich & Schunk, 1996; Ryan & Deci, 2000). Some of the most powerful needs of learners arise from their perceptions of themselves in relation to the skills or tasks:

1. **Perceived Access** (e.g., “I don’t have this software at home and I need extra time to practice.” Will the teacher offer a workshop or open worktime in class?)

2. **Perceived Domain Ability** (e.g., “I can’t do anything on computers” or “I already know everything about computers.” Both are counterproductive perceptions.)

3. **Perceived Developing Competence** (e.g., “I don’t know how I am doing.” Feedback can tell the student, “You have skill b, now work on skill c.”)

4. **Perceived Safety** (vs. social risk e.g., “I might look stupid if I ask a
question or don’t get it right away”)

5. Perceived Value (i.e., “How will I benefit from learning this skill?)

6. Perceived Utility (i.e., “How will I use this skill outside of class?)

7. Perceived Relevance (i.e., “How does this affect me and the things that matter to me?”)

8. Aspirations (i.e., “Why does this matter to me in my future? I want to be x; how will this help me?”)

The first four perceptions drive learners’ motivation to engage, to invest time energy in learning and doing tasks. The second four perceptions drive motivation to persist, to stay engaged when challenged or frustrated, to overcome obstacles (Bandura, 1997; Reeve, 1996).

To enhance learners’ perceived access, ability, competence, and safety, the teacher can design and maintain a learning environment that allows learners to work at their own pace, to ask questions in nonpublic ways, to receive regular competence feedback, and to experience incremental successes (Keller, 2000; Pintrich & Schunk, 1996; Reeve, 1996). To promote learners’ high perceived value, utility, relevance, and links to aspirations, the teacher can provide frequent, relevant rationales and illustrations of how skills and tasks are used and what people gain from doing them well (Keller; Pintrich & Schunk; Reeve, 1996).

One caveat on perceptions is to GO SLOWLY when troubleshooting learners’ problems in the system. People who are facile with a technological tool notoriously say thing like “just click here”; a menu appears, a dialog box flashes on the screen and disappears before learners can absorb what it said and what the teacher did in response. If teachers move too rapidly through solutions, then learners never see the path of that solution, and they will need help with the same problem again. Going slowly and explaining the steps in problem-solving with learners enables them to develop more independent skills, to solve the problem themselves the next time, and maybe even to help their peers.
A FEW CLOSING WORDS

The more teachers know about how people learn, the better we can teach. The better teachers understand the nature of what we are teaching, the more clearly and effectively we can communicate it to our students in meaningful ways. These principles are at the core of instructional design. With a systematic, organized approach to analyzing content and understanding learners’ needs, teachers can step back from the familiar work of teaching and perhaps gain a fresh perspective on their learners and what they are learning. Once instruction is planned, it should be flexibly responsive to the developing abilities, and the changing needs and perceptions of learners (Bransford et al., 1999; Bieleczyc & Collins, 1999).

There are many models for designing instruction and many methods of creating effective instruction for teaching technology skills. Systematic instructional design is offered here as one process for considering and designing the whole of instruction so that, lasting, meaningful, learning occurs.

References


Moore, J. (1999). *Discussions in cognition and instruction course*, University of Iowa.


