Pedagogical Framework for Developing Thinking Skills Using Smart Learning Environments

Sahana Murthy, Sridhar Iyer, and Madhuri Mavinkurve

Contents

S. Murthy (✉) • M. Mavinkurve
IDP in Educational Technology, Indian Institute of Technology Bombay, Mumbai, India
e-mail: sahanamurthy@iitb.ac.in

S. Iyer
Department of Computer Science & Engineering, Indian Institute of Technology Bombay, Mumbai, India

J. M. Spector et al. (eds.), *Learning, Design, and Technology*,
DOI 10.1007/978-3-319-17727-4_14-1

Abstract

Students need to develop thinking skills in addition to content knowledge. Many thinking skills in engineering and science are pan-domain in nature, such as system design, algorithmic thinking, creation and revision of scientific models, problem posing, and so on. Emerging smart learning environments have high potential in developing learners' thinking skills. While there exist teaching-learning strategies for various thinking skills as well as learning environments that promote the learning of thinking skills, there is a need for a pedagogical framework that helps researchers to design effective smart learning environments targeting thinking skills. This chapter describes one such framework, called the TELoTS framework, which was developed using a design-based research methodology. The framework is based on theoretical foundations of the nature of thinking skills, learning and transfer of thinking skills, and design principles for complex learning in technology-enhanced learning environments. The framework prescribes a set of actions to be followed by the designer of a smart learning environment and offers guidelines on implementing the actions. The framework was applied to design a smart learning environment to develop thinking skills in engineering system design. Evaluation studies showed that the smart learning environment designed using the TELoTS framework led to improved student learning of the thinking skill.

Keywords

Thinking skills • Pedagogical framework • System design • Technology-enhanced learning

Introduction

An important goal of education is that students develop thinking skills in addition to content knowledge. There is no uniquely accepted definition of thinking skills, but overall, they are regarded as abilities and processes that human beings apply for sensemaking, reasoning, and problem-solving (Lipman, 2003). Thinking skills in engineering and science include system design, problem posing, estimation, algorithmic thinking, creation and revision of scientific models, data representation and analysis, and so on. These thinking skills are pan-domain in nature, that is, they share common characteristics that have applicability across domains.

Researchers have identified and characterized pan-domain thinking skills in a variety of ways, such as transdisciplinary habits of mind (Mishra, Koehler, & Henriksen, 2011), twenty-first-century skills (Pellegrino & Hilton, 2012), critical thinking skills (Facione, 1990), science process skills (Padilla, 1990), computational thinking skills (ISTE, 2014), and so on. Professional bodies have listed student outcomes (ABET, 2014) which include a number of pan-domain thinking skills. Regardless of the education discipline, development of thinking skills has been shown to be crucial for students' success in the twenty-first-century workplace (NAS, 2014). While the importance of thinking skills has been well established, its teaching and learning are complex. Learners do not automatically develop thinking skills while learning content. Practice of routine application, such as learning to solve well-structured problems, does not transfer to solving ill-structured problems (Jonassen, Strobel, & Lee, 2006). Hence, it is important to address the teaching and learning of thinking skills explicitly, in addition to content.

There have been several efforts at teaching thinking skills at various educational levels and in various domains. The field of inquiry learning in science (Minstrell & Van Zee, 2000) consists of targeted efforts toward student learning of thinking skills such as investigating questions with empirical data, testing hypotheses, manipulating variables in an experiment, and so on. Educational researchers have designed curricula aimed at promoting scientific abilities, such as ISLE (Etkina & Van Heuvelen, 2007). The affordances of modern information and communication technologies have been used to design learning environments not only for domain knowledge and conceptual understanding but also for thinking skills. Such technology-enhanced learning environments provide opportunity for formulating and testing hypotheses via virtual experimentation, allow systematic exploration of what-if scenarios in simulations, engage learners in argumentation, afford multiple external representations to manipulate, and so on. Examples of technology-enhanced learning environments which promote learning of thinking skills include Model-It (Jackson, Krajcik, & Soloway, 2000), WISE (Linn, Clark, & Slotta, 2003), and Co-Lab (van Joolingen, de Jong, Lazonder, Savelsbergh, & Manlove, 2005) for science inquiry and modeling skills, Belvedere (Suthers et al., 2001) for scientific argumentation, gIBIS (Conklin & Begeman, 1988) for decision-finding processes for "wicked problems," and many others.

More recently, researchers have focused on smart learning environments (SLEs). SLEs enrich a physical learning environment with technologies that add productive learning functions; are context aware and adaptive; provide opportunities for planning, reflection, and practice; and provide learners evolving feedback to monitor and assess their progress (Koper 2014; Scardamalia, & Bereiter, 2014; Spector, 2014). These affordances in SLEs are valuable for the learning of thinking skills. In terms of design approaches to SLEs and other technology-enhanced learning environments, there are numerous detailed instances supported by empirical studies. Most such examples propose and use a set of design principles underlying their learning environments; however, translating these principles and applying them to the design of a new learning environment is difficult. Hence, a pedagogical framework for designing SLEs for thinking skills would be useful.

Approach and Scope of TELoTS Framework

There exist a few broad theoretical models for problem-solving and inquiry, for example, cognitive apprenticeship (Collins, 2006) and “first principles of instruction” (Merrill, 2002). A relevant and influential framework for instructional design (ID) is the *Ten Steps to Complex Learning* (van Merriënboer & Kirschner, 2012), which provides ID blueprints for a course or curriculum involving complex learning, defined as “the integration of knowledge, skills and attitudes; coordinating qualitatively different constituent skills; and often transferring what was learned in school or training to daily life and work” (pp. 2).

In this chapter, we describe a pedagogical framework which draws on elements from the above models and suggests an ID approach for designing SLEs for learners’ development of thinking skills. Since SLEs for thinking skills need to fulfill common requirements such as identifying potentially useful ideas, exploring what-if scenarios, making decisions, integrating knowledge, and evaluating progress, it would be useful to have a pedagogical framework that helps researchers design effective learning environments specifically targeting thinking skills. Another reason for focusing on a pedagogical framework is that a large body of research in SLEs is led by technology experts such as computer scientists, whose expertise is essential to the successful creation of SLEs. However, the primary focus of such work is often the technological features and functionalities of the SLE system, whereas the pedagogical features are considered as an additional “layer” to be included at a later stage. Such an approach may be detrimental to the goal of learning. Our approach is a framework that maintains the focus on the learning goals, i.e., developing thinking skills. It provides pedagogical design guidelines toward facilitating these goals and suggests technological affordances of an SLE that support this pedagogy.

Our framework is called the TELoTS framework – Technology-Enhanced Learning of Thinking Skills. The framework was developed using a design-based research approach. It is based on the theoretical foundations of characterizing the nature of thinking skills, learning and transferring thinking skills, and designing principles for complex learning in technology-enhanced learning environments. The TELoTS framework provides the why, what, and how of various steps of designing an SLE for thinking skills. It prescribes a set of broad actions for the designer of the SLE to follow and one possible way of implementing the actions.

The intended primary user of the TELoTS framework is a researcher who has some familiarity with interactive learning technologies and with the learning sciences. A typical such user would be a graduate student intending to design an SLE for thinking skills. A secondary user is a technology expert who wishes to develop a pedagogically sound SLE. The framework will guide the user in conceptualizing the SLE, creating the broad ID, designing the learning activities, making decisions for the choice of technologies, and directing the work of the technical implementation team.

The scope of the TELoTS framework in this chapter is thinking skills related to system design in engineering. We focus on system design thinking because it is one of the important pan-domain skills across engineering disciplines (Dym, Agogino, Eris, Frey, & Leifer, 2005). The need for defining such a scope for the framework is that while thinking skills themselves may be pan-domain, applying them productively in a given context may require knowledge of concepts and techniques specific to that domain and context. Our approach is to first develop and evaluate the framework within the stated scope and then propose extensions of its applicability.

Organization of This Chapter

This chapter is organized as follows: we begin by describing the theoretical foundations of the TELoTS framework, in which we discuss the ways researchers have characterized and assessed thinking skills, findings from learning sciences research on productive supports for complex learning, and various issues in the transfer of thinking skills. We briefly discuss the development methodology and evolution of the TELoTS framework. The bulk of the chapter is focused toward the actions and guidelines prescribed by the TELoTS framework. We then illustrate one example of application of the framework to the design of an SLE for structure open problem thinking skill, a key initial thinking skill required in system design thinking. We show empirical results of student learning of structure open problem thinking skill from the SLE. Based on the design and results, we conjecture a learning trajectory, that is, we discuss possible mechanisms of how student learning of thinking skills might have occurred. Finally, we describe how different categories of users can leverage different parts of the TELoTS framework for their own research and development.

Theoretical Foundations

The actions and guidelines of the TELoTS pedagogical framework are grounded in the theoretical bases of (i) how researchers characterize thinking skills and assess learners' development of these thinking skills; (ii) how to design productive learning supports using technology for complex, higher-level learning; and (iii) how researchers understand the transfer of pan-domain thinking skills, each of which are reviewed below.

Characterizing and Assessing Thinking Skills

In order to develop thinking skills among learners, one needs to be able to define what the thinking skill means and analyze how learners have progressively developed the thinking skill. One way to do so is a competency-based approach that provides a means to operationalize the abstract understanding of the thinking skill. Competencies are an attempt to capture various aspects of the process of application of a thinking skill. A competency approach has the advantage that it can be used to define measurable outcomes to recognize the development of learners' thinking skills. Researchers have used such an approach to characterize various thinking skills such as engineering design thinking (Crain, Davis, Calkins, & Gentili, 1995), problem-solving (Woods et al., 1997), critical thinking (Facione, 1990), and scientific abilities such as devising and testing hypotheses, designing experiments, and evaluating reasoning (Etkina et al., 2006).

An important consideration for assessment of thinking skills is that learners be given the opportunity to demonstrate the application of their knowledge and skill in "worthy intellectual tasks" (Wiggins, 1990). Such an approach to assessment is different from that of administering a test for specific skills and measuring the gains. To move toward "authentic assessment" (Wiggins), the context of assessment can involve solving open-ended, ill-structured problems that relate to actual problems solved by professionals in the real world. The assessment process is aided by the competencies, which provide the measurable outcomes related to students' application of the thinking skills in such open-ended problem contexts.

A recommended method of assessing open problems is by the use of assessment rubrics. Rubrics are descriptive rating scales which provide performance criteria for the target skill learners are expected to achieve (Mertler, 2001). To assess students' development of thinking skills, the rubric categories are based on the thinking skill competencies. Rubrics are known for their capacity to enhance deep learning among students by providing rich, detailed, clear, and specific feedback to students about their performance (Arter & McTighe, 2001). Rubrics provide formative feedback not only at the target level of performance but also at all intermediate levels, thereby helping students assess their own efforts relative to the target criteria. Assessment instruments such as rubrics go beyond the function of evaluation and can be used to promote students' learning especially if they are used for self- and peer assessment (Black & Wiliam, 1998).

Designing Productive Learning Supports

Learning environments for thinking skills should allow learners to construct, articulate, represent, and share ideas, provide opportunities to experiment, engage them in sensemaking and reasoning processes, and offer timely and adaptive formative feedback (Spector, 2014). SLEs can provide technology affordances for the above requirements in the form of interactive simulations, pedagogical agents, visual conceptual organizers, multiple representations of concepts and data, question prompts with automated feedback, etc. SLEs can personalize these learning supports based on learners' immediate needs. Another effective condition for learning is collaboration among peers, which has support from social constructivism theories (Vygotsky, 1978) and empirical studies alike (Johnson & Johnson, 1999). SLEs can harness computer-supported collaborative learning strategies (Dillenbourg, Järvelä, & Fischer, 2009) to promote peer learning and social interactions during the learning and problem-solving process.

To learn and apply thinking skills, students need to work on problems which are open-ended and authentic. This could pose a significant challenge, and learners need appropriate scaffolding to help them succeed in such complex tasks. The inclusion of scaffolds has been recommended to promote learning of not only conceptual and procedural knowledge but also to guide reasoning, sensemaking, and thinking skills such as conducting experimental investigations (Reiser, 2004). Since the learning of thinking skills may involve both discipline-specific knowledge as well as generalized structural knowledge (Ge & Land, 2004), scaffolds should be provided for both. A useful framework to design scaffolding for inquiry learning using software tools (Quintana et al., 2004) recommends that scaffolds be provided for sensemaking, process management, articulation, and reflection.

A highly recommended design component in an SLE for learning of complex cognitive tasks is metacognitive supports (Ge & Land, 2004; Reiser, 2004). For performing such tasks, learners – especially novices – need to consciously abstract the required strategies from the learning context and mindfully apply them to the new context (Perkins & Salomon, 1992). Metacognitive supports help the learner plan, monitor, and evaluate strategies needed to develop thinking skills (Ge & Land, 2004).

Transfer of Thinking Skills

A basic premise of designing interventions for the development of thinking skills is to investigate the potential of their transfer to new contexts and topics and (possibly) to new domains. Traditionally, transfer has been considered as "direct application" in which the learner independently applies knowledge and skills acquired in one situation into another. This approach has been criticized because of its narrow criteria for successful transfer as well for its view of knowledge as a static entity. A more current theoretical approach to understanding transfer is "preparation for future learning (PFL)" (Bransford & Schwartz, 1999), which focuses on students' abilities to use knowledge learned in one context to learn (or relearn) in new contexts, with new resources. According to PFL, the new context is not isolated and can involve supports that help the learner perform the task in the new situation. PFL recommends that assessments of learning should go beyond sequestered problem-solving by providing opportunities for new learning and focusing on students' abilities to learn in new contexts (Schwartz & Martin, 2004).

Key features of learning environments recommended to promote transfer are that they need to support constructive learning processes, enhance students' self-regulation, and consciously encourage students to use their knowledge and skills (De Corte, 2003). Learners' attention should be focused on recognizing similar patterns and strategies across multiple and diverse instances (Gentner, Loewenstein, & Thompson, 2003). Engaging students in reflection on their implementation of strategies is recommended for successful transfer (Catrambone & Holyoak, 1989).

Development and Evolution of Framework

Our original goal was to develop an SLE for thinking skills. In that process, we found a few examples and design guidelines for SLEs. Although these had been empirically validated for their context, applying them directly to our context (specifically to the learning goals of thinking skills) proved to be difficult. Hence, we evolved a pedagogical framework for thinking skills along with the design of an SLE.

Our methodological approach followed a design-based research process (DBR) (Reeves, 2006). This is a systematic design-oriented approach to research, allowing for iterative cycles of problem analysis, prototype design, field implementation, evaluation, and refinement. DBR studies in education have dual goals – empirically engineering solutions that target a teaching-learning problem and developing theories of domain-specific learning processes and the means to support the learning (Cobb, Confrey, Lehrer, & Schauble, 2003). In recent years, DBR has been frequently used in the design of technology-enhanced learning environments.

We carried out three cycles of DBR, each cycle consisting of four stages – problem analysis, solution development, evaluation, and reflection – as shown in Fig. 1a. The specific activity in a given stage was different for each cycle. In the first cycle, the problem analysis stage dominated and dealt with characterizing the

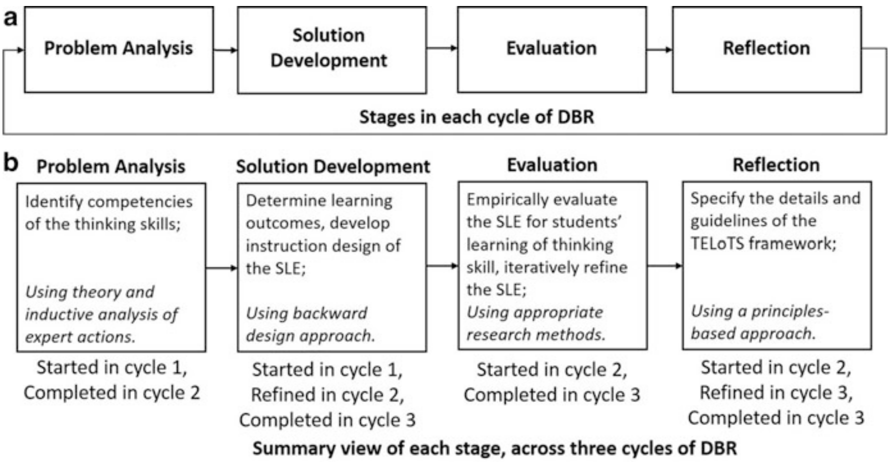

Fig. 1 DBR stages in creating the TELoTS framework

thinking skills. Problem analysis was carried out by a combination of a theory-driven approach and an inductive analysis of the actions of practicing domain experts to identify the underlying steps and processes involved in the development of the thinking skill. In this cycle, the solution development stage also began but focused predominantly on identifying the requirements of the SLE.

In the second cycle, the solution development stage dominated and dealt with detailed design and development of the SLE. Solution development was carried out using a backward design approach (Wiggins & McTighe, 2005) to determine the learning outcomes related to the thinking skill competencies, decide assessment measures, and develop the instructional design of the SLE. In the second cycle, the evaluation stage began but focused predominantly on identifying refinements to the SLE. The reflection stage of the second cycle gave rise to the abstraction of the broad steps of the TELoTS framework.

In the third cycle, the evaluation stage dominated and dealt with empirical evaluations of the SLE for students' learning of the thinking skill and iterative refinement of the SLE. The reflection stage of the third cycle gave rise to the detailed steps of the TELoTS framework. For ease of comprehension, Fig. 1b depicts a summary view of the main actions in each stage, merged across all three cycles.

The TELoTS Framework

We use Garrett's model of the *Elements of User Experience* as the structure for building the TELoTS framework (Garrett, 2011). The goal of our TELoTS framework is to help designers of smart learning environments build an interactive system with a learner-centric focus, i.e., the learning experience of the user is the key. Garrett's work describes an information architecture consisting of five planes to address different aspects of a system in order to achieve the desired level of user experience. While Garrett's model was originally proposed in the context of designing a website, it was later applied more broadly to designing for user experience with complex products or systems (see *Introduction to the Second Edition*, *pp xiii* in Garrett, 2011).

The five planes and the *Elements* are summarized below in Table 1 (going from abstract to concrete). On each plane, the designer needs to take decisions at the level of detail required at that plane. The decisions at each plane are dependent on those made in the previous plane (going top to bottom) and influence the choices available in the further planes.

The TELoTS framework uses Garrett's model as a structure to prescribe a set of actions and guidelines to design an SLE for thinking skills. The broad categories of actions are deciding the thinking skill, topic, and problem-solving context, characterizing the thinking skill, designing the learning strategy, and creating the technology system. These actions are typically sequential, but occasionally, the SLE designer may need to loop back and revise previous actions. This is in line with Garrett's model in which actions in one plane may sometimes lead to a reevaluation of decisions made in a previous plane.

Table 1 Model for the *Elements of User Experience* (Garrett, 2011)

	Plane	Aspects addressed in the plane	The *Elements* Decisions and actions taken in the plane
Abstract concepts ↑	Strategy	What do the designers want to get out of the system? What do the users want?	Goals of the product (e.g., interactive system) Needs of the user
	Scope	Transform strategy to requirements: What features will the system need to include?	Features, functions, services, facilities, content
	Structure	Give shape to scope: How will the pieces of the system fit together and behave?	Categories, hierarchy, scenarios, storyboards, workflows, use case
	Skeleton	Make structure concrete: What components will enable people to use the system?	Layouts, placements, interfaces, widgets, controls, task flows
↓ *Concrete details*	Surface	Bring everything together visually: What will the finished system look like?	Text, images, links, look

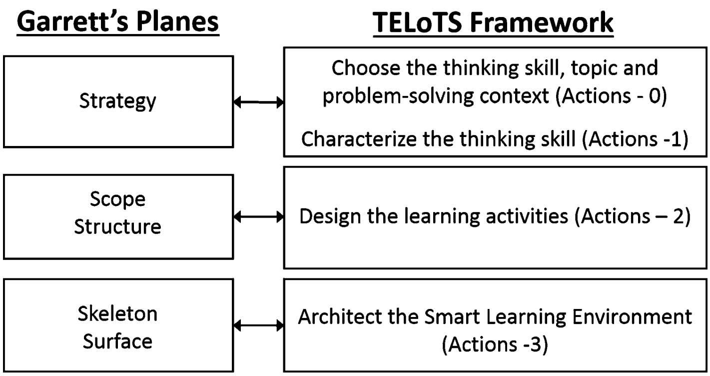

Fig. 2 Mapping of Garrett's planes to TELoTS framework

The mapping between Garrett's model and the TELoTS framework is shown in Fig. 2 below.

Figure 3 shows the details of the TELoTS framework – the actions (0, 1, 2, 3) and guidelines (a, b, c, d) to be considered for each action.

A key requirement in the development of learners' thinking skills is that learning needs to take place at multiple levels. At a granular level, the learner needs to acquire various competencies of the thinking skill and develop expertise in the individual competencies. At the same time, the learner needs to be able to understand and successfully apply the integrated thinking skill. The SLE in turn needs to contain

The TELoTS Framework

0. Choose the thinking skill, topic and problem-solving context

1. Characterize the thinking skill

a. Identify the competencies of the chosen thinking skill
b. Create learning outcomes
c. Consider assessment measures

2. Design the learning activities

a. Analyze expert actions and learner needs
b. Decide instructional strategies and supports
c. Identify technology features to realize the instructional strategies
d. Create a sequenced set of learning activities

3. Architect the components and interfaces of the SLE

Fig. 3 Actions and guidelines of the TELoTS framework

elements that facilitate learning at each level, as well as promote the back-and-forth interactions between the granular and integrated levels. Thus, the SLE designer needs to take a holistic approach to design which "deals with complexity without losing sight of the separate elements and the interconnections between those elements" (van Merriënboer & Kirschner, 2012, pp. 5). To address these simultaneous perspectives, each action of the TELoTS framework provides guidelines at both the granular (i.e., competency) level as well as the integrated (i.e., entire thinking skill) level.

In the next few subsections, we describe the details of the TELoTS framework. We suggest implementation strategies to achieve each action in Fig. 3 by applying the guidelines. While the framework suggests one way of implementing the "how" of the actions and guidelines, the SLE designer can choose alternate appropriate implementations if required.

Actions-0: Choose the Thinking Skill, Topic, and Problem-Solving Context

One assumption made in the framework is that the broad thinking skill has been chosen by the SLE designer. While the nature of a thinking skill may be pan-domain, its learning and application occur in the context of a topic within a domain. Thus, an important action in designing an SLE for thinking skills is to determine an appropriate topic to develop the thinking skill. Some researchers have stated that the acquisition of a thinking skill and the domain knowledge (content) in which it is acquired go hand in hand and may not be clearly separable (e.g., argumentation skill in Scheuer et al. 2010). This may be valid, and we do not imply an artificial separation between the thinking skill and domain knowledge. Instead, our premise is that students often do not recognize the common pattern of application of a thinking skill. Hence, it is important that the SLE *explicitly and primarily* focuses on the development of the thinking skill within the chosen topic(s).

The TELoTS framework provides various options for an SLE designer to decide the thinking skill and topic:

(i) The SLE designer may be an expert in a given topic and may decide that deep learning of that topic requires a particular thinking skill. For example, an electrical engineering faculty member would argue that the learning of analog electronic circuits needs students to develop design thinking skill.
or
(ii) The SLE designer may be an expert in a particular topic and may need to identify which thinking skills can be developed through that topic.
or
(iii) The SLE designer may be interested in student learning of a particular thinking skill and may need to decide an appropriate topic.

Decisions regarding the topic and thinking skill have to be made cooperatively. There may be some back-and-forth iteration to decide a suitable thinking-skill-topic combination till *the key properties needed for the manifestation of the thinking skill are identified*. It is also up to the SLE designer whether to choose all topics (or subtopics) for a thinking skill from the same domain or whether to find relevant topics from multiple domains.

At this point, the SLE designer also needs to select or create a set of problems from a real-world context in which the thinking skill needs to be applied. These problems are representative of the complex thinking skill. Desirable characteristics of such problems are that they are rich and contextualized and admit multiple solutions or solution paths. They lie closer to the ill-structured end of the continuum of the problem types described by Jonassen (1997). These problems should require the learner to apply all aspects of the thinking skill so that they develop an integrated set of knowledge and skills.

Given that the TELoTS framework recommends problems that are open-ended and from a real-life context, another decision point that may come up is the amount of domain knowledge that can be assumed. For this, the SLE designer may assume that the required domain knowledge is present among the intended learners and choose a suitable set of learners accordingly, or they may provide the required domain concepts within the SLE itself, which the learner can access as needed.

Actions-1: Characterize the Thinking Skill

Implementing the current step of characterization of the thinking skill involves devising an operational definition of the thinking skill via competencies (*Guideline 1a*), defining the expected outcomes for learning (*Guideline 1b*) and deciding the assessment measures (*Guideline 1c*). While it may seem unusual to consider assessment measures before an instructional strategy is designed or the technological features are decided, this approach is consistent with the *backward design* approach (Wiggins & McTighe, 2005). Backward design has been recommended as an alternative to traditional curriculum design since it identifies a clear understanding of the "destination" of the educational process as well as how to get there. In the backward design approach, the desired results are first identified (in this case, outcomes based on thinking skill competencies), then assessment measures, strategies, and instruments are defined to provide evidence for the above results, and finally learning activities are designed to achieve the desired results.

Guideline 1a: Identify Competencies of the Chosen Thinking Skill

The competency-based approach provides a means to operationalize the thinking skill and further provides a basis for assessment via learning outcomes. The purpose of the competencies is not to categorize or compartmentalize learning domains. On the other hand, it provides a tractable approach for a researcher to capture various aspects of the complex thinking skill.

To identify competencies of the chosen thinking skill, the framework recommends a combination of a literature-driven approach and an inductive analysis of experts' actions as they demonstrate various competencies related to the thinking skill (e.g., when experts solve a problem requiring the application of the thinking skill). In the case of some thinking skills such as critical thinking (Facione, 1990), a theoretical framework is available for what competencies constitute the thinking skill. Analysis of literature may be sufficient to identify the competencies since researchers may already have done the task of breaking down a thinking skill into sub-skills and operational competencies. However, in the case of other thinking skills, such literature may only be partially complete or even incomplete. In such cases, the TELoTS framework suggests an empirical inductive analysis of experts' actions to identify relevant competencies. The experts are chosen based on their expertise of application of the thinking skill in the domain of interest. It is possible that the SLE designers themselves may be one of the experts in a domain in which they wish to develop learners' thinking skill.

The broad steps to conduct the empirical analysis to identify the competencies for the thinking skills are:

(i) Identify a context of application, such as an open problem where the application of the thinking skill is required.

(ii) Ask experts to solve the problem, and write a detailed solution including various options considered, justifications for choices, and so on.
(iii) Code expert solutions to identify the specific performance actions which relate to the application of the thinking skill. The unit of analysis can be chosen to be the key “steps” of the solution (as determined by the expert solution). The codes can be based on categories like “what action did the expert take at this step,” or “what cognitive process did the expert follow to apply the thinking skill at this step.”
(iv) If more details are required for a solution step, or if the reasoning behind a particular step is not clear, conduct a follow-up interview of the expert to capture these details. Recode if necessary.
(v) Group all performance actions under a similar code, and label it as a competency of the thinking skill.

Steps iii–v above provide only broad guidelines for coding and categorization of experts’ actions. The reader can refer to techniques such as cognitive ethnography (Williams, 2006) for a detailed description. It is advisable to conduct the above analysis on multiple problems relevant to the thinking skill. It is also desirable to choose more than one expert. Both these recommendations will help strengthen the validity of the results.

Guideline 1b: Create Learning Outcomes

The competencies of a thinking skill typically do not make any reference to a topic, as they are expected to hold across a range of topics (within the stated scope). However, in a later step, learning activities in the SLE need to be designed within a topic. Hence, a bridge is needed between the topic-independent thinking skill competencies and the topic-contextualized learning activity for applying that thinking skill. This bridge is provided by the learning outcomes, which operationalize the competencies of the thinking skill into the chosen topic.

The guidelines to write learning outcomes for thinking skill competencies are similar to those provided by various taxonomies. The most well known of these, Bloom’s taxonomy (Anderson, Krathwohl, & Bloom, 2001), suggests the use of an action verb to describe a learner’s performance outcome. The TELoTS framework does not map the learning outcomes and action verbs to specific cognitive levels (as in Bloom’s taxonomy). Instead, it maps the learning outcomes to the competencies of the thinking skills.

A learning outcome contains aspects of both the thinking skill competency and the topic, i.e., it is written by “applying” the competency to the topic. One way to write learning outcomes is (i) write the solution to an open problem from the chosen domain for which the thinking skill is needed; (ii) from the solution, identify the steps to achieve each competency of the thinking skill; and (iii) use action verbs to write specific learning outcomes for each step.

While it is pragmatic to break down an abstract thinking skill into its constituent competencies and corresponding learning outcomes, such an approach could lead to the problem of the compartmentalization and fragmentation (van Merriënboer & Kirschner, 2012). That is, students may develop individual competencies and achieve corresponding learning outcomes, but they may not be able to integrate various elements to apply the complex thinking skill in the domain context. To address these issues, the TELoTS framework prescribes an integrated learning outcome common for all thinking skills, in addition to the learning outcomes written using the above steps:

> *Students should be able to reflect on the specific learning activity carried out in the domain context, and identify how it relates to the abstract thinking skill being addressed in the activity.*

The implication of including this learning outcome is that the SLE designer will need to consider assessment measures and learning activities corresponding to this outcome (described in the future sections).

Guideline 1c: Consider Assessment Measures, Strategies, and Instruments

An advantage of choosing a competency approach for characterizing thinking skills is that it provides a basis for assessment via measureable learning outcomes. One point to keep in mind while assessing thinking skills is that it involves not merely a final product evaluation but also the identification of learners' cognitive processes as they apply the thinking skill in a new context. To address these conditions for assessment, the TELoTS framework recommends the use of descriptive assessment rubrics based on thinking skill competencies. Assessment rubrics contain several items or categories related to the product or process being analyzed, in this case students' understanding and application of thinking skills. The rubrics are applied to learners' performance on open-ended problems requiring the application of the thinking skill.

Several detailed guidelines are available for drafting, validating, and implementing assessment rubrics (Mertler, 2001; Moskal, 2000). Below is a summary of the key steps and points to be noted while creating rubrics, so that they are effective as assessment instruments for thinking skills:

(i) The TELoTS framework recommends that analytic rubrics be used in which each thinking skill competency forms an item (or category), along with a holistic judgment built in (Moskal, 2000), i.e., at least one item of the rubrics should correspond to the prescribed integrated-level learning outcome (Guideline 1b).

(ii) For each item, rubrics should provide rich and detailed descriptions of performance levels, not just a graded scale (go beyond simplistic descriptions such as excellent, good, fair, and poor).
(iii) Rubrics should provide descriptions of the target level of performance as well as all intermediate levels so that students can understand the target concept or skill they are expected to achieve and the criteria to achieve that skill.
(iv) The scale and number of performance levels for an item can be decided by the SLE designer, but typical rubrics contain three to seven performance levels.
(v) Once the rubrics are drafted, they should be tested for validity and inter-rater reliability using standard methods (Moskal and Leydens (2000) contains a discussion of validity and reliability specific to rubrics).
(vi) A good practice for the use of rubrics is that students should have access to them during their learning process; hence, the TELoTS framework recommends the use of rubrics in a formative manner.

Actions-2: Design the Learning Activities

After having chosen the problem context for learning the thinking skill and decided learning outcomes, the SLE designer has to design specific learning activities that help the learner solve the problem and achieve the outcomes. To design the learning activities, the SLE designer must understand experts' practices and productive actions as they apply the thinking skill in the problem context, as well as learners' needs and challenges in doing so (*Guideline 2a*). Both these analyses inform the choice of instructional strategies which form the basis of the learning activities. These analyses also suggest where and what supports should be incorporated (*Guideline 2b*). To realize these instructional strategies and supports within the SLE, the designer needs to identify requirements of the technology (*Guideline 2c*). Finally, the SLE designer's role is to integrate the above aspects of content, pedagogy, and technology and create a sequenced set of activities through which the learner interacts with the SLE (*Guideline 2d*). In the following subsections, we elaborate on the above guidelines and discuss some techniques on how to implement the guideline when the focus is on the learning and application of thinking skills.

Guideline 2a: Analyze Expert Actions and Learner Needs

Solving of complex tasks have cognitive requirements such as domain-specific content knowledge and organized knowledge structures as well as metacognitive requirements, that is, knowledge and regulation of one's own cognition (Ge & Land, 2004). Metacognition encompasses the goal setting, planning, monitoring, control, and regulation of one's own cognitive processes (Veenman, 2012). It has been found that when experts solve ill-structured problems, especially when their domain knowledge is limited, such metacognitive processes play an important role. Thus, for learners to be able to solve complex problems, metacognitive supports should be included in the learning environment (Ge & Land, 2004). The SLE needs to be designed such that the instructional strategies and learning activities in it trigger the required metacognitive processes as students work on a complex problem requiring the application of the thinking skill.

To identify the underlying metacognitive processes needed in the chosen problem context, the SLE designer may first review related work to locate if such processes have been identified for similar thinking skills and problem contexts. If this is unavailable or insufficient, the TELoTS framework suggests conducting a cognitive task analysis of experts. There are a variety of techniques to carry out cognitive task analysis (Clark, Feldon, van Merriënboer, Yates, & Early, 2008; Jonassen, Tessmer, & Hannum, 1999). Below is an overview of the broad steps applied for the goals of the TELoTS framework:

(i) Choose a context of application of the thinking skill. This can be the same as the chosen problem context (Guideline 0). Ask experts to write its detailed solution (similar to the steps for identifying thinking skill competencies in Guideline 1a).

(ii) Since the goal of the task analysis is to recognize the metacognitive processes that experts use when they solve the problem, ask the expert to write down not just the solution but also their decision points, how they made the decision, what alternatives they considered, their reasoning for various choices, and so on.

(iii) Group the experts' solution steps that fall under common learning outcomes for the thinking skill.

(iv) Analyze the solution steps and code it to focus on the underlying metacognitive processes of the expert. Metacognitive processes involve knowledge about one's cognition such as one's capacities and limitations, knowledge about when to use various acquired cognitive strategies, planning solution approaches and setting goals, reflection and monitoring of one's cognitive efforts toward the solution, frequent evaluation of the methods used and the results obtained, and reconsidering and refining of plans and goals (Jacobs & Paris, 1987; Pressley & McCormick, 1987). The following are indicators to recognize metacognitive actions of experts (adapted from Kinnebrew, Segedy, & Biswas, 2014):
   - The knowledge and control exhibited over their thinking and performance activities
   - Awareness of their own thinking and conceptions
   - Active monitoring of their cognitive processes
   - An attempt to control and regulate their cognitive processes to support learning
   - The application of heuristics or strategies for developing their own approach to solving problems

v. Group the identified metacognitive processes, and map them to their respective learning outcomes of the thinking skills.

In addition to identifying the underlying metacognitive processes in experts' practice, another dimension of analysis is to identify the needs of learners. This analysis can help the SLE designer to focus the design and identify where supports are needed. In the process of learning and applying thinking skills to solve a complex problem in an SLE, learners face challenges at several levels. They need to master domain knowledge, domain-specific process skills and strategies, general problem-solving heuristics, metacognitive processes, discourse practices (such as expressing a hypothesis or conducting a pros and cons analysis), and social interaction practices (such as constructing scientific arguments to convince peers) (Reiser, 2004). Many of these challenges have been documented and categorized; for example, see challenges related to sensemaking, process management, and articulation in science inquiry learning (Quintana et al., 2004). If needed, the SLE designer can conduct a need analysis study of the intended learners by giving them the open problem to solve and identifying what and where supports are required.

Such an analysis can go beyond identifying learners' cognitive needs. Learners' affective states, interests, motivation, and familiarity with the thinking skill affect students' learning (D'Mello & Graesser, 2012) and hence may be considered in the design of the learning environment. At this point, detailed recommendations for these considerations are out of scope of the TELoTS framework. SLE designers who may wish to consider interest, motivation, and other affective needs can refer to a special issue on interest in *Learning and Instruction*, *2002* (see, e.g., the introduction by Boekaerts & Boscolo, 2002).

Guideline 2b: Decide Instructional Strategies and Supports

The main purpose of identifying the metacognitive processes underlying the application of the thinking skills is to decide the instructional strategies that trigger these metacognitive processes in learners. These instructional strategies will form the basis of the learning activities to be designed. To map the identified metacognitive processes to instructional strategies, the SLE designer will need to synthesize learning sciences literature. While specific instructional strategies will depend on the expert and learner analysis (Guideline 2a), some instructional strategies are commonly recommended for a variety of thinking skills. These include formative assessment, interpreting and creating multiple representations for a concept, opportunity for variable manipulation and experimentation, and learner evaluation of results. If applicable, the SLE designer can use these strategies.

Scaffolding. In addition to instructional strategies, the SLE needs to incorporate supports, which provide the important function of *scaffolding* complex learning, i.e., the software tools in the SLE modify learners' tasks so that they can accomplish the tasks which would otherwise be out of their reach. Scaffolding serves two key purposes – (i) it provides performance support, i.e., the scaffold structures the tasks to make them more tractable for learners; and (ii) it provides learning support, i.e., the scaffolds problematize the tasks to make the problem-solving experience more productive for learning (Reiser, 2004).

For learning of thinking skills, both purposes are important, but the latter (learning supports) are particularly recommended. By problematizing the task, the scaffolds in the SLE point students toward aspects that require attention (such as making a decision at a given point), leading them to grapple with key issues in applying the thinking skill. This in fact makes the problem more challenging in the short term but may aid transfer as the learners are stimulated to devote effort to the key issue and reflect on the experience. The SLE designer needs to keep in mind that the two purposes of structuring the task and problematizing the task may sometimes be at odds with each other.

To identify productive scaffolds, the SLE designer can:

i. Use the analysis of learners' obstacles and challenges (from Guideline 2a) to decide what scaffolds are needed for structuring and simplifying the open-ended complex problem. Examples of such scaffolds are procedural prompts, scoping complex tasks, task decompositions, visual concept organizers, etc. (Ge & Land, 2004; Quintana et al., 2004).
ii. Use analysis of experts' productive actions (Guideline 2a) to identify what types of scaffolds can focus learners' attention on the metacognitive processes that trigger thinking skills. Such scaffolds can include tools for learners to seek and acquire information, providing and allowing manipulation of multiple representations of the same object or data, making problem-solving strategies explicit in learners' interactions with the SLE, providing reminders and guidance to facilitate planning, and monitoring of learners' tasks (Quintana et al., 2004).
iii. Incorporate elaboration and reflection prompts (Ge & Land, 2004), which prompt learners to articulate their reasoning and explicitly evaluate their efforts (e.g.,"We use this approach because . . .," "Is the qualitative model you developed useful to . . .," "Are the units in your equation correct . . ."). This category of scaffolds is necessary for learners to achieve the integrated learning outcome (described in Guideline 1b), i.e., for learners to abstract the thinking skill being addressed in the learning activity.

Overall pedagogical strategies. The SLE designer also has to consider and decide overall pedagogical strategies related to collaboration, personalization, adaptation, analytics, and so on. When implemented appropriately, each of these broad pedagogical strategies has been reported to be beneficial to learners. The powerful technologies in SLEs can provide various opportunities for these pedagogical strategies (Spector, 2014).

Guideline 2c: Identify Technology Features to Realize the Instructional Strategies

Once the instructional strategies, scaffolds, and personalization approaches are decided, they need to be realized in the SLE by identifying appropriate technological features. Current SLEs provide a variety of affordances such as location and context awareness for personalization, adaptive technologies for recognizing and responding to learners' needs, learning analytic tools for immediate feedback, interactive simulations and augmented situations for deep conceptual understanding, virtual worlds for immersive learning, facilitation of dialogs for collaborative learning, and so on. Spector (Spector 2014) contains a discussion of necessary, desirable, and likely characteristics of an SLE. Also see other numerous articles in the *Smart Learning Environments* journal http://slejournal.springeropen.com/ for examples of technological affordances of SLEs.)

The TELoTS framework is based on the premise that it is first important to identify the need for the smart technology feature and consider what the technology can afford toward the development of learners' thinking skills. Thus, the TELoTS framework recommends the following broad steps:

(i) Consider each instructional strategy and scaffold from the perspective of functions and requirements for technology features.
(ii) A given instructional strategy or scaffold may have several aspects or components, each of which can be achieved by one or a combination of technologies. Inspect each component of the chosen instructional strategies and scaffolds (from Guideline 2b), and decide which aspect can be implemented by which technology feature.
(iii) Make choices of technologies which can support all the chosen features. More than one choice may be possible in which case the SLE designer can use other considerations such as novelty of the technology, their own expertise with a technology, and feasibility for their context.

Guideline 2d: Create a Sequenced Set of Learning Activities

At this stage, the SLE designer has to create complete individual learning activities for a given problem context and then sequence the learning activities. For a chosen thinking skill, the designer may need to devise multiple problems in a variety of contexts and sequence them in the SLE.

Recommendations for learning activities. The TELoTS framework recommends the following characteristics for effective learning of thinking skills:

- *Learner centric*. The learning activities should largely be learner centric: they should promote learners to seek and acquire information (either from the SLE itself or by searching other resources such as the WWW), apply that information to solve a task, pose and answer questions, act on feedback, and self-assess their solution approaches.
- *Mapping to thinking skill competencies*. The learning activities for a given problem context should together address all the competencies of the thinking skill.
- *Variability*. Sufficient learning activities should be created to provide practice for each thinking skill competency; at the same time, the activities must be different from each other so that learners can abstract the thinking skills. Such variability of

practice on "all dimensions that also differ in the real world" has been recommended to promote transfer (van Merriënboer & Kirschner, 2012).

- *Address integrated thinking skill*. The goal of the SLE is that the learner develops the integrated thinking skill (and not merely demonstrate its competencies). Hence, the SLE should contain not only individual learning activities separately mapped to each competency or learning outcome but also learning activities that address the integrated learning outcome (described in Guideline 1b). The TELoTS framework recommends that:
  - Explicit reflection questions be provided after every (or every few) learning activity that prompt learners to articulate which thinking skill competencies were addressed in that activity.
  - For a chosen thinking skill, multiple open problems be provided in a variety of contexts that require the application of the thinking skill. These problems may be sequenced using a more to less scaffolded approach so that learners do more independent application and reflection in the subsequent problems.
  - A summary *synthesis and reflection activity* be provided toward the end of each open problem in the sequence, wherein learners articulate which all thinking skill competencies were used in solving the open problem and how they may be useful to solve the next problem in the sequence (which requires the application of the same thinking skills but may be in a different topic). This activity is crucial for being able to transfer the thinking skill.

Overview of steps to create learning activities. The SLE designer is expected to "wear the hat" of a teacher and relate the domain context (part of the chosen open problem context from Actions-0), instructional strategies and scaffolds (Guideline 2b), and the technological requirements (Guideline 2c). The instructional strategies and scaffolds provide the pedagogical format of the learning activity. The identified technological features determine the "look and feel" of the activity and provide details of what actions that learner should take during the activity.

An overview of the steps to design a learning activity is:

(i) Decide the specific content from the chosen open problem context for the learning activity at hand.
(ii) Decide the back-and-forth interactions of the learner with the system, given the broad steps of the instructional strategy and the features of the chosen technology.
(iii) Integrate (i) and (ii). Create individual learning activities, and sequence them following the recommendations in the previous bullets. In addition:
  (a) One technique that can be used at this stage is *conjecture mapping* (Sandoval, 2014), which provides an SLE designer "a means of specifying theoretically salient features of a learning environment design and mapping out how they are predicted to work together to produce desired outcomes" (pp. 3).
  (b) The decision of specific learning activity and sequence may also arise from the experience of teaching in the domain, as well as from the expertise with the technology affordances.

iv. Choose new problem contexts that require the application of the thinking skill, following guidelines from Actions-0. Repeat the steps in Actions-1 and Actions-2 as required. All actions may not be necessary if the new problem context requires the application of the same thinking skill.
v. Do mock testing of the learning activities with a few learners. A preliminary form of the technology may be built for this. The goal of this step is not to test the entire system but to test if the specific learning activities are effective in meeting its intended learning outcomes.

Actions-3: Architect the Components and Interfaces of the SLE

Having designed and sequenced the learning activities and supports (Actions-2), the next set of actions is to design the architecture of the SLE. This architecture serves as the blueprint to inform the implementation of the SLE. The SLE designer needs to decide the components, interfaces, and other software aspects that will enable learners to use the SLE. At this stage, the SLE designer may need to work closely with a software developer. Since implementation details are likely to be platform specific, the TELoTS framework provides only broad guidelines for architecting the components and interfaces of the SLE. For each guideline, we provide additional readings that the SLE designer can refer to if a more detailed discussion is needed. The guidelines to architect the SLE are as follows:

- Identify the *functional requirements* of the SLE, i.e., what the SLE should do, based on the technology features identified earlier (Guideline 2c).
- Identify the *nonfunctional requirements* of the SLE, i.e., how the SLE should work, based on the sequence of learning activities decided earlier (Guideline 2d). For example, if the learning activities require tracking the progress of individual learners, the SLE needs to have modules that implement the corresponding logging mechanisms.
- Provision for *adaptation* and *personalization*. These involve presenting the learning material according to students' learning needs, as indicated by their responses to questions and tasks. The SLE could adjust for the pace of learning, or the approach of learning, thereby providing for different instructional paths for different students. This may require implementation of features such as student models, expert models, logging and tracking of interactions, and adaptation logic in the SLE. Adaptation and personalization strategies in mobile learning are discussed by Kinshuk, Chang, Graf, and Yang (2010). An editorial (Kinshuk, 2012) on personalized learning introduces a special issue of *ETR&D* journal on various approaches to personalization and personalized learning environments.
- Design the parts of the SLE that are critical to determine whether the main learning outcomes of the SLE are likely to be met. This is important because user testing may indicate need for modifications which are desirable to discover early in the development cycle. Hence, one important nonfunctional requirement to consider is *extensibility*, i.e., adding of new functionality or modification of

existing functionality if needed. Extensibility enables "rapid prototyping" and "user testing" of the SLE with students, at multiple points during the development process. Another reason to provide for *extensibility* is because the technologies underlying the SLE may evolve over a period of time through release of new/modified libraries, and it may be necessary to upgrade the SLE to use the new versions of the technology.

- Design the user interfaces (UI) and test them for *usability*, i.e., ease of use of the learner. This is because (i) lack of usability is the first barrier to usage and adoption of the SLE by students and teachers, and (ii) it is often hard to modify the UI post-facto, without significant implementation overheads. Moreover, usability of the SLE may in turn have an impact on the learning. Hence, it is worthwhile to design the UI of the SLE, create a "storyboard" of each learning sequence, test it with users, iteratively refine the UI, and then go on to implement the SLE.
- Follow standard best practices and coding conventions during implementation (programming) of the system. Two desirable nonfunctional requirements for the software developer to keep in mind are *maintainability*, i.e., ease of finding and fixing "bugs," and *portability*, i.e., ensuring that the system works on different hardware platforms.

  (Detailed guidelines for SLE implementation are beyond the scope of the TELoTS framework; the reader can refer to a standard software engineering book such as Pressmann (2005)).

Example of Applying TELoTS Framework

We now illustrate the application of the actions and guidelines of the TELoTS framework to the design of an SLE for structure open problem thinking skill in engineering system design.

Applying Actions-0: Choosing the Thinking Skill, Topic, and Problem Context

An important thinking skill in engineering is system design thinking. Educators (Sheppard, Macatangay, Colby, & Sullivan, 2009), professional organizations (ABET, 2014), and numerous other reports (NAE, 2005) have emphasized that graduating students should be able to design effective solutions for given needs. System design thinking is a complex cognitive process that results into an open-ended creative task (Dym et al., 2005). It requires cognitive processes of both analysis and synthesis about a system. The designer not only performs a problem-solving activity but also applies a systematic and thoughtful process involving exploration and decision-making (Sheppard & Jenison, 1997). A key and initial thinking skill in engineering system design is structure open problem thinking skill.

A substantial part of design activity is devoted to structuring and formulating the problem (Cross, 2007).

The chosen domain was analog electronic circuits, which is taught as a foundational course in undergraduate engineering programs in most universities. Design of circuits is a key topic in analog electronics. To effectively learn structure open problem thinking skill in system design, the topic should allow multiple structures for the same open design problem. Each structure should be the result of the specifications or parameters in the problem and the way they are related to each other. These features are present in the design of analog electronic circuits. An exemplar in this domain is amplifier design, which was chosen as the problem context. Within it, several problems were chosen so that students learn design thinking in the context of audio frequency as well as power amplifiers, use active devices such as bipolar junction transistors (BJT), field-effect transistors (FET), and OPAMPs. One such problem is shown below, which is used in further analysis in this section:

> *A weak signal of strength 1 mV is recorded using a recorder which needs minimum 1 V signal to start recording. The frequency range of the signal is 100 Hz to 100 KHz. Design a circuit to record the signal accurately.*

Applying Actions-1: Characterizing the Thinking Skill

Identifying the Competencies of Structure Open Problem Thinking Skill (Guideline 1a)

While an analysis of literature yields a number of related ideas that define structure open problem thinking skill, the approach taken by different researchers varies, and the nomenclature used is different. For example, structure open problem has been referred to as "problem definition" (Davis et al., 1997), which involves "for a given problem situation, prepare a goal statement with specific technical and nontechnical, measurable, criteria to be satisfied in a successful design solution." It has been considered to be a design ability (Sheppard & Jenison, 1997) to "define and formulate an open-ended and/or under defined problem, including specifications," and as a design activity (Aurisicchio, Ahmed, & Wallace, 2007) in which the system designer "frames the problem with broader view and connects different issues to create chunks."

In order to identify the specific competencies constituting structure open problem thinking skill, we conducted an empirical inductive analysis of experts' solutions to an open system design problem. Five experts were identified: they were engineering college faculty members with 10+ years of experience in teaching design courses in analog circuits. Each expert was given open design problems in the topic of amplifier design to structure. Their solutions were then broken down into design steps and coded for the performance action taken by the expert in each design step. Codes were compared and refined till consistency and saturation were reached. Table 2 shows the

Table 2 Identifying competencies of a thinking skill, via analysis of experts' solutions

Steps	Application of step	Example
Identify a context of application of the thinking skill in the chosen topic	Consider, for example, a problem which needs the application of the thinking skill	A weak signal of strength 1 mV is recorded using a recorder which needs minimum 1 V signal to start recording. The frequency range of the signal is 100 Hz to 100 KHz. Design a circuit to record the signal accurately
Ask experts to write the solution to the given problem	**For given problem**, expert needs to	**Transcription of part of an expert's solution, with categorization of design steps** (labeled as S1, S2...)
	Write detailed solution which will contain design steps	
	Give justification or reasoning based on content	S1: In the open problem, input voltage is V_{in}=1 mV, and output voltage is V_{out}=1 V. So calculate voltage gain of amplifier =1 V/1 mV=1,000
Choose an individual design step as the unit of analysis	Along with experts' help, the researcher needs to identify the design steps in the expert solution	S2: Gain=1,000 indicates high gain, so which circuits can provide such high gain? If we select BJT amplifier, we can get gain maximum of 100; it is possible to cascade other stage to get required gain. Possible circuits are two-stage BJT amplifier or BJT–FET combination
		S3: In open problem, another specification mentioned is frequency range as 100 Hz to 100 KHz. Hence, required bandwidth=100 KHz-100 Hz ~ 100 KHz
		S4: If we increase gain by cascading, there is reduction in bandwidth since gain is inversely proportional to bandwidth, but required bandwidth is possible with above choices
		S5: The circuit should not overload the recorder as well as should not get overloaded from input signal generator. This gives idea of impedance matching, i.e., high input impedance and low output impedance
		S6: If both active devices are BJT, it gives low to moderate

(*continued*)

Table 2 (continued)

Steps	Application of step	Example
		input impedance. So select active device with high input impedance, i.e., FET. Two-stage FET–BJT combination can be suggested with FET at input stage
Code each step based on the action taken by the expert	Decide the action in each design step	In step *S1*, the expert identifies the gain value using given data. Gain is a specification of amplifier
	Write a code for each action taken	This step is coded as "identification of specification"
		Repeat for S2, S3, . . .
Group all design steps involving a common code	Go through all the steps of solution and identify code categories	Four code categories emerged:
		Identification of specifications
		Use specifications to decide structure
		Identify sequence of design steps
		Write structured problem
	Group all design steps coded under common category	For code of "identification of specifications," steps S1, S3, and S5 are grouped together
	Reanalyze design steps under a single category, to check possibility of further categorization	When we checked all the steps above, we found that each step leads to "identification of specification" code, and no further categorization was possible

steps of the inductive analysis of experts' solutions and coding process, along with an example.

The following four competencies for structure open problem thinking skill emerged as a result of the analysis:

- SOP1 – Identify relevant specifications (or parameters)
- SOP2 – Use specifications to decide structure
- SOP3 – Sequence design steps
- SOP4 – Write structured design statement

Creating Learning Outcomes (Guideline 1b)

Table 3 shows the learning outcomes for structure open problem competencies and applied to the problem chosen in Actions-0 (also shown in Row 1 of Table 2, Row1).

Table 3 Learning outcomes for structure open problem competencies

Competency	Expected learning outcome	Applying learning outcomes for a problem in amplifier design
SOP1: Identify specifications	Students should be able to identify relevant visible and hidden specifications/parameters/ requirements from open problem	Students should be able to identify gain and bandwidth as the key visible specifications and input impedance as hidden specification in design of amplifier
	Students should be able to determine the values of the identified specifications and interpret them	Students should be able to calculate gain of given system and determine if it is "high" or "medium" or "low"
SOP2: Use specifications	Students should be able to apply all the relevant specifications to take decisions to structure problem	Students should be able to decide number of stages in the amplifier circuit based on gain
	Students should be able to decide the interconnections of the system based on the identified specifications	Students should be able to determine relation between gain and bandwidth for amplifier
SOP3: Decide design steps	Students should be able to identify all decision steps to structure the problem	Students should be able to decide design steps such as gain-bandwidth calculations and which active devices to be used.
	Students should be able to decide sequence of decision steps to structure problem	Students should be able to decide sequence of steps, like calculation first and stage identification second
SOP4: Write structured statement	Students should be able to write problem statement by systematically integrating specifications, decision steps, devices, structures, etc.	Students should be able to write statement as "design multistage amplifier with FET–BJT combination with specific gain value/ bandwidth

Considering Assessment Measures and Instruments (Guideline 1c)

To assess if learners have acquired structure open problem thinking skill in system design, they were given an open design problem and asked to structure it. Their detailed solution was then scored using rubrics. The criteria of assessment are the competencies identified by applying Guideline 1a. A four-point scale was chosen for the performance levels, ranging from "target performance" to the lower levels of "need improvement," "inadequate," and "missing." Table 4 shows the rubrics to assess the competencies of structure open problem thinking skill.

Applying Actions-2: Design the Learning Activities

Analyzing Experts' Actions (Guideline 2a)

Five experts from analog electronic circuit domain were asked to write solutions to an open design problem in amplifier. Experts' solutions to these design problems were analyzed for their actions to achieve the learning outcomes, and the underlying

Table 4 Assessment rubrics for structure open problem competencies

	Performance level			
Competency	Target performance	Need improvement	Inadequate	Missing
SOP1: is able to identify relevant specifications in detail from the given open problem	All relevant visible and hidden specifications are identified in detail and interpreted accurately. No irrelevant specifications are identified	An attempt is made to identify specifications. Most are identified but hidden ones are missing, or a few need more interpretation	An attempt is made, but most specifications that are identified are wrong, irrelevant, or incomplete	No attempt is made to extract specifications
SOP2: is able to structure the open problem using specifications	All specifications are used to take decisions to structure problem. All interconnections of the system are identified based on given and identified specifications	An attempt is made to use specifications, but a few minor specifications are not used for deciding the structure	An attempt is made to use specifications, but they are wrongly applied, or some required specifications are not applied	No attempt is made to use specification or identify structure
SOP3: is able to sequence the design steps based on specifications	All major and minor design steps are identified and sequenced correctly based on specifications	Most design steps are identified and sequenced correctly. Minor steps are missing or not sequenced correctly	Design steps are not sequenced at all or not based on specifications	No attempt is made to write design steps
SOP4: is able to write structured problem statement	Problem statement is written clearly including details of devices, structures, and design steps	Problem statement is written clearly, but few minor details are missing	Problem statement is not written clearly, but scattered attempts are seen	No attempt is made to write a structured problem statement

metacognitive processes were inferred. Table 5 shows the implementation of the process.

We did not carry out learners' needs analysis in this example.

Deciding Instructional Strategies and Supports (Guideline 2b)

The metacognitive processes identified from experts' design solutions to attain structure open problem thinking skill were decision-making, concept integration, and synthesis.

Table 5 Steps to decide instructional strategies to develop structure open problem thinking skill competencies

Steps to decide instructional strategy	Application of step
Identify a context of application of the thinking skill	(*This step is similar to the one in Table 2 but repeated for clarity*) A weak signal of strength 1 mV is recorded using a recorder which needs minimum 1 V signal to start recording. The frequency range of the signal is 100 Hz to 100 KHz. Design a circuit to record the signal accurately
Ask experts to write detailed solution to the given problem	**Transcription of part of an expert's solution**, **along with categorization of design steps**: (*This step is similar to the one in Table 2 but repeated for clarity*)
	S1: In the open problem, input voltage is V_{in}=1 mV, and output voltage is V_{out}=1 V. So calculate voltage gain of amplifier =1 V/1 mV=1,000
	S2: Gain=1,000 indicates high gain, so which circuits can provide such high gain? If we select BJT amplifier, we can get maximum gain of 100; it is possible to cascade another stage to get required gain. Possible circuits are two-stage BJT amplifier or BJT–FET combination
	S3: In open problem, another specification mentioned is frequency range as 100 Hz to 100KHz. Hence, required bandwidth=100 KHz to 100 Hz ~ 100 KHz
	S4: If we increase gain by cascading, there is reduction in bandwidth since gain is inversely proportional to bandwidth, but required bandwidth is possible with above choices
	S5: The circuit should not overload the recorder and should not get overloaded from input signal generator. This gives idea of impedance matching, i.e., high input impedance and low output impedance
Group steps which contain similar learning outcome	Learning outcome SOP1 – students should be able to identify relevant specifications
	S1: In the open problem, input voltage is specified as V_{in}=1 mV, and output voltage Vout =1 V is given. The first step is this specification indicates that increased amplitude leads to voltage amplification. *Decide specifications based on concepts*
Code the solution to identify metacognitive actions of experts	*S2*: Calculate gain of the amplifier as voltage gain=1 V/1 mV=1,000. The first specification of voltage gain of the amplifier is 1,000. *Concept association for specifications*
	S5: The circuit should not overload the recorder as well as should not get overloaded from input signal generator. This gives idea of impedance matching, i.e., high input impedance and low output impedance. *Decide specifications based on concepts*

(*continued*)

Table 5 (continued)

Steps to decide instructional strategy	Application of step
Note: Here we consider the following indicators:	Learning outcome SOP2 – students should be able to interpret specifications
Awareness of one's own thinking and conceptions	*S2*: Gain=1,000, indicates high gain. *Concept linkage to decisions*
	S4: Bandwidth =100 KHz is medium bandwidth. *Concept linkage to decisions*
An attempt to control and regulate one's cognitive processes	Learning outcome SOP3 – students should be able to apply all the relevant specifications to take decisions to structure problem
	S2: *Gain=1,000 indicates high gain*, *which circuits can provide such high gain*? *If we select BJT amplifier, we can get gain maximum up to 100. Conceptual linkages for decisions*
	It is possible to cascade other stage to get required gain. Two-stage BJT amplifier can be designed. Decide circuits based on concepts
	S4: If we increase gain by cascading, there is reduction in bandwidth. *Decide connection between specifications based on concepts*
	S4: But required bandwidth is possible with two-stage BJT amplifier. *Decide circuits configuration*
Group codes related to common metacognitive processes	*Decision-making*
	Decide specifications based on concepts
	Decide circuits based on concepts
	Decide circuit configuration
	Concept integration
	Concept application for specifications
	Concept linkage to decisions
	Synthesis
	Synthesis of all above tasks which involves recalling of concepts, deciding the structures, applying information, and integrating process

- **Decision-making**. Decision-making process is defined as generating possible options for a given situation and then evaluating options based on set of information. For decision-making, students need to think of many options based on set of information and evaluate them based on domain knowledge expertise. Decision-making can be triggered using series of reasoning questions targeted toward making a decision as well as providing options for selection. Addition of self-regulation mechanism is known to work as a catalyst in decision-making process. One way to implement self-regulation is via formative assessment questions (Aurisicchio et al., 2007; Gresch, Hasselhorn & Bögeholz, 2013).
- **Concept integration**. Learners are expected to recall appropriate concept, identify interrelationship between various concepts, and connect relevant concepts. It

Table 6 Instructional strategies for the underlying metacognitive processes of structure open problem thinking skill

Metacognitive process	Requirements of the instructional strategy	Instructional strategies
Decision-making	Planning, monitoring, and evaluation	Formative assessment question at each decision-making step, in which students are asked to make decision along with reasoning
		Possible choices for decisions provided as scaffolds to guide students toward making informed decision
	Self-regulation	Constructive feedback provided for each student response to aid them in making appropriate decisions, as well as explanations for why certain decisions are inappropriate for the problem
		Self-assessment activities with guidance
Concept integration	Knowledge integration	Question prompts related to association of concepts
		Opportunity for experimentation involving relationship between concepts
	Information visualization	Activity to interpret multiple representations of a given concept
	Reflection	Question prompts related to monitoring and evaluation, with feedback
Synthesis	System thinking	Summary statements that help students think in terms of the system as a whole
		Activities for decision-making, information integration, multiple representations, and opportunity for experimentation (similar to above) to synthesize knowledge about the entire system
		Self-assessment activities with guidance

also requires knowledge of multiple representations with visual thinking. Concept integration process shows similarity with knowledge integration process for inquiry learning for which opportunity for experimentation is a recommended strategy for knowledge integration Chen, Hong, Sung, & Chang, 2011).

- **Synthesis**. Synthesis involves thinking in terms of the system as a whole and needs decision-making, information integration, multiple representation, as well as opportunity for experimentation. There is a need to provide an opportunity to converge thinking process by putting all concepts and decisions together. In addition to the recommended strategies for decision-making and concept integration, supportive summary statements are recommended to develop synthesis.

Table 6 summarizes the instructional strategies for the above metacognitive processes.

In addition, a number of scaffolds such as question prompts and self-assessment prompts were provided, which are discussed in the examples in the upcoming subsection "Creation and Sequencing of Learning Activities."

Identifying Technology Features (Guideline 2c)

The main technology requirement for the above instructional strategies is that there is a reciprocal interaction between the learner and the SLE. When the learner interacts with the SLE and performs the learning activities, it provides customized feedback to guide the learner.

In addition, opportunity for experimentation is provided as variable manipulation simulations. The learner decides and chooses the inputs, and the SLE provides dynamic visual feedback based on the inputs.

Creating and Sequencing Learning Activities (Guideline 2d)

We refer to the learning activities as *learning dialogs*, to indicate the two-way reciprocal process of interaction and response between the learner and the SLE. We illustrate the process creating learning dialogs, by connecting the instructional strategy, technology features, and content, with four examples, including an overall synthesis and reflection activity.

Example 1. Learning Dialog: Decision-Making Task Question

Consider the competency SOP1 – identify specifications and its corresponding learning outcome in amplifier design. “Students should be able to identify the main visible and hidden specification in open problem.” When applied to the chosen problem (Actions-0), the relevant specifications turned out to be gain and bandwidth (visible specifications) and input impedance (hidden specification). These specifications are required to be able to decide the structure of the circuit (e.g., number of stages) in the next step of solving the problem.

For a learner to be able to make this decision, we provided formative assessment questions at each decision-making step, in which students are asked to make a decision along with reasoning. We provided possible choices for decisions. Upon making a choice, students are provided feedback for that choice, to guide them through their reasoning and point them toward a productive decision. The technological requirements for this activity are that it should provide various choices with customized feedback for each choice.

A learning activity that implements the above is a “decision-making task question” (DMTQ). A DMTQ is a conceptual multiple choice question in which each choice is a plausible decision related to that question. For each choice, explanatory feedback and further actions are designed to guide the learner toward productive decisions. Figure 4 shows an example of a DMTQ learning activity for the SOP1 competency.

The question asks students to identify which is relevant specification from given set of specifications. The feedback contains:

(i) Explanation related to reasoning for why the chosen decision leads to poor design.
(ii) Guidelines which can lead students to a productive decision (but not to tell them a specific correct decision).

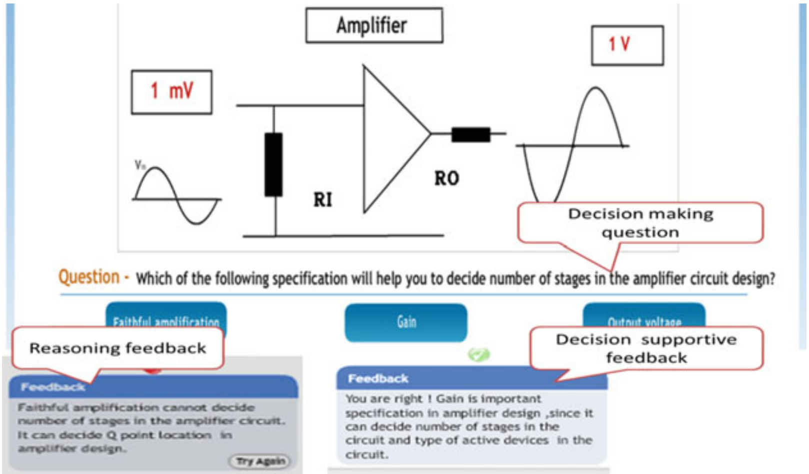

Fig. 4 Decision-making task question for learning outcome of SOP1

(iii) In case of choice of productive decision, the feedback explains what makes the selected decision productive.

Example 2. Learning Dialog: Self-Assessment

A powerful form of implementing formative assessment is self-assessment (Black & Wiliam, 1998). One way to realize self-assessment in an SLE is via the thinking skill rubrics developed in Guideline 1c (e.g., see Table 4). Rubrics can be used to provide feedback to students so that they can monitor their own learning process with respect to the learning goals. At the same time, the rubrics focus students' attention on the important tasks needed to be done for solving the complex open problem at hand, i.e., these can be considered as scaffolds that problematize the task. Figure 5 shows a screenshot of self-assessment rubrics for structure open problem competency included in a DMTQ activity.

Example 3. Learning Dialog: Simulative Manipulation

One strategy to trigger concept integration is by using guided experimentation. We designed *simulative manipulations* as a learning dialog to provide experimentation opportunity to students. We created simulative manipulation using guided activity principle. In simulative manipulation, students are allowed to select different parameters of design, and changes are shown as graphs or wave forms based on various input values. Figure 6 shows an example of a simulative manipulation learning dialog.

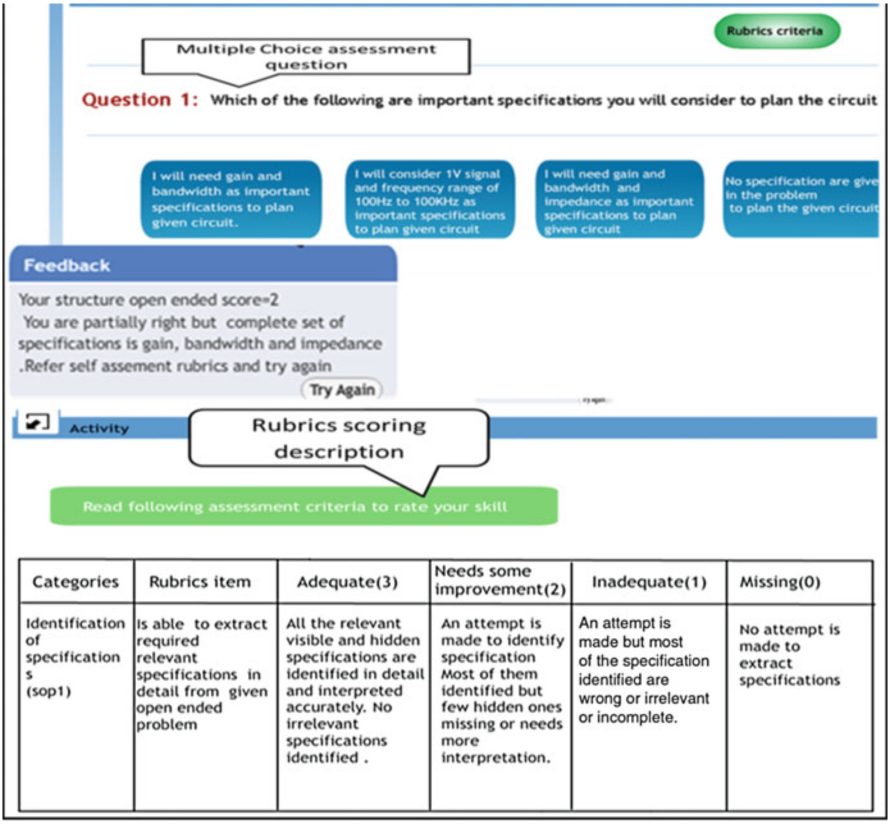

Fig. 5 Self-assessment rubrics added in a DMTQ activity

Example of synthesis and reflection activity. Students are asked to write structured problem statement for the given unstructured problem. They are prompted to systematically integrate the specifications, devices, structures, etc. on which they worked in the previous learning activities (such as DMTQ, simulative manipulation, etc.). Students are provided capsule recommendations, which are key summary statements that act as scaffolds in the design process. Examples of such statements are “increase in the number of amplifier stages increases overall gain of the amplifier,” or “product of gain and bandwidth for a given amplifier system is constant.” The look and feel of capsule recommendations is up to the SLE designer. Principles of personalization and adaptation can be used to create pedagogical agents that provide the capsule recommendations at appropriate times.

Table 7 summarizes the learning activities mapped to the learning outcomes of structure open problem competency for problems in the topic of amplifier design.

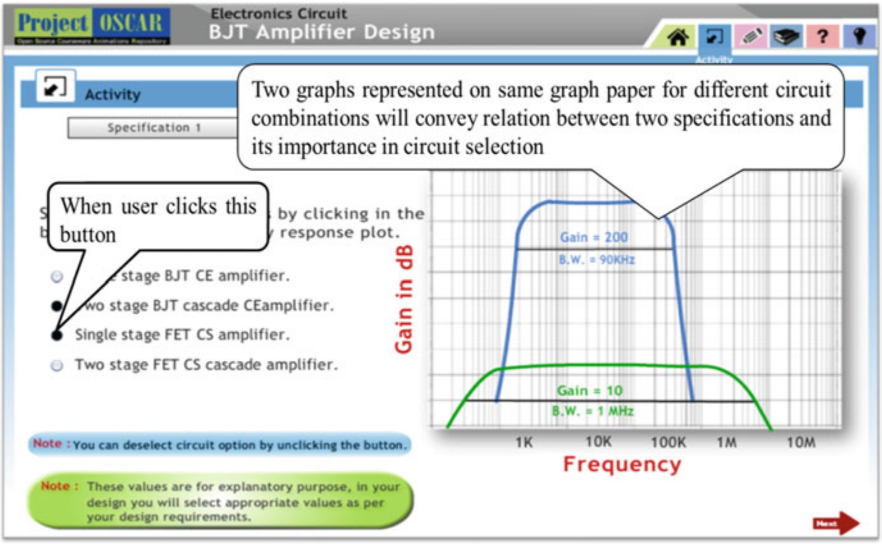

Fig. 6 Simulative manipulation learning dialog

Table 7 Learning activities for structure open problem thinking skill

Competencies of structure open problem thinking skill	Learning outcomes (from Table 3)	Learning activities
SOP1 – Identify specifications	Students should be able to identify relevant visible and hidden specifications/parameters/ requirements from open problem	Concept clarification question
		Decision-making task question
	Students should be able to determine the values of the identified specifications and interpret them	Self-controlled animation
SOP2 – Use specifications	Students should be able to apply all the relevant specifications to take decisions to structure problem	Decision-making task question
	Students should be able to decide the interconnections of the system based on the identified specifications	Simulative manipulation
		Decision-making task question
SOP 3 – Sequence design steps	Students should be able to identify all decision steps to structure the problem	Concept clarification question
	Students should be able to decide sequence of decision steps to structure problem	Decision-making task question
SOP 4 – Write structured problem statement	Students should be able to write problem statement by systematically integrating specifications, decision steps, devices, structures, etc.	Information agents
		Capsule recommendations

Evaluation of TELoTS Framework

In the previous section, we illustrated the application of the TELoTS framework to structure open problem thinking skill in the topic of amplifier design in analog electronics. We designed learning activities as per the TELoTS framework and developed an SLE based on it called TELE-EDesC. We now summarize the results of two experimental studies investigating students' acquisition of structure open problem thinking skill while learning with TELE-EDesC compared to a control group (Study 1) and transfer of the thinking skill competencies to a new topic in a new problem context (Study 2). We also briefly describe the results of an interaction analysis and interviews of students who worked with the learning activities of TELE-EDesC (Study 3).

Study 1: Acquisition of Structure Open Problem Competencies

The research question for this study was: does the process of engaging in TELE-EDesC learning activities affect students' acquisition of structure open problem design competencies?

Learning activities were developed for TELE-EDesC SLE in various subtopics of analog electronics (DC circuit design, audio amplifier, power amplifier, OP-AMP, etc.), which are typically taught in undergraduate engineering programs. The participants in the study were second year undergraduate students from various engineering colleges in urban areas in India. Students were divided into two groups based on randomized assignment (N _total = 293, N_exp = 146, N_control = 147). The two groups were analyzed to be equivalent based on an independent sample *t*-test of differences in students' previous semester's marks in analog electronic course (no statistically significant difference at $p > 0.05$ level). Further, all students from both groups were familiar with the topic in TELE-EDesC, as they had learned it in the theory course on the same topic in the previous semester. However, they were not exposed to engineering design in this topic. All students were familiar with ICT-based learning materials such as interactive simulations. However, they were mostly used to learning from lectures and by doing homework problems and were not used to self-learning.

During the learning phase, both groups learned in self-study mode using their respective learning materials. Experimental group students learned with the activities in TELE-EDesC, in which they went through the process of structuring an open design problem. Control group students learned with material in the same topic but in the format of *informative visualizations*, which was in the format of slides containing text, diagrams, and animations but without the learning activities based on the TELoTS framework. Both groups were given 40 min to work with the material. In the testing phase, students in both groups took a competency acquisition test which involved a paper and pencil task of structuring a different open problem in the same topic.

Students' open responses were scored using the structure open problem competency rubrics (Table 4), which had been validated for content and construct validity and inter-rater reliability (Cohen's kappa = 0.61). The statistical significance of score difference between two groups was analyzed using Mann–Whitney *U*-test. The results showed that students who learned from TELE-EDesC had higher scores on all four competencies, and the difference in scores was statistically significant at the $p < 0.01$ level, leading us to conclude that the learning activities in TELE-EDesC helped students acquire structure open problem thinking skill. (Some aspects of this study have been reported in Mavinkurve & Murthy, 2012.)

Study 2: Transfer of Structure Open Problem Competencies

The research question investigated in this study was: do students who learn with TELE-EDesC transfer the design competencies to a new situation?

In this study (reported in detail in Mavinkurve & Murthy, 2015), a subset of students from the experimental group Study 1, i.e., students who learned with TELE-EDesC, were given new learning material in the form of slides with diagrams and explanation of decision steps (i.e., *not* as TELE-EDesC learning activities). Students were not familiar with design of circuits in this topic. Students studied the material for 30 min. Then they were given a test in which they had to structure an open problem in the new topic. Their responses were scored on the structure open competency rubrics as before. Students' scores in the new topic were maintained at the same high levels as the ones in the TELE-EDesC topic (Study 1), for competencies SOP1, SOP2, and SOP3. However, for SOP4 – write structured problem statement – their scores reduced compared to the scores on the TELE-EDesC topic.

What is important to note is that students were not trained in TELE-EDesC learning activities in the new topic on which their structure open problem competency was tested. They learned the content of the new topic in a "traditional" manner using slides and diagrams, without specific learning activities targeting the thinking skill. Yet students were able to score equally well on the new topic for some thinking skill competencies. Hence, the conclusion is that students transferred the structure open problem competencies they had acquired from the first topic.

Study 3: Interaction Analysis

This study involved a qualitative interaction analysis of students' behaviors as they interacted with various TELE-EDesC learning activities (Mavinkurve & Murthy, 2013). A subset of students who worked with TELE-EDesC learning activities (i.e., experimental group students in Study 1) were chosen for this study. Screen-capture logs of the interactions were transcribed, coded, and analyzed. Each action of the student was coded, for example, "read information," "manipulated all/some variables in simulative manipulation," "chose incorrect decision in DMTQ," "read feedback," "acted on suggestion in feedback," and so on. The data were analyzed on how much time each student spent on various learning activities, how frequently they visited it, and what they did in each visit. The time spent on an activity and the number of revisits indicate the emphasis a student places on different learning activities.

The results indicated that students who scored high on the structure open problem competency rubrics had spent most of their time on decision-making via DMTQ activities (32 % of their time) and experimentation via simulative manipulation activities (17 %). Students who scored low were found to spend the largest fraction of their time reading information (34 %) but spent only 8 % of their time on variable manipulation. The TELoTS framework recommends activities like DMTQ and simulative manipulation to trigger metacognitive process underlying thinking skills. The results support the correlation between the time spent on such activities and the acquisition of the thinking skill.

This is further supported by the students' responses to semi-structured interview questions, in which they were asked about their opinions on the various aspects of TELE-EDesC. Students were encouraged to reflect on why they perceived TELE-EDesC to be helpful or challenging. Below we report some quotes from students who scored high in the structure open problem rubrics:

> *When I studied the learning material I understood that I need to ask questions to decide which is the appropriate circuit* [. . .] *also I have to think what all circuits can satisfy given data. I also need to apply my knowledge to select appropriate circuit.*

The following quotes specifically relate to the self-assessment activity using structure open problem rubrics (similar to the one in Fig. 5):

> *If I know where I go wrong and I also know how I should correct myself I can design the system. Rubrics showed me how to reach there.*
>
> *After I read the assessment rubrics, I again studied the material and re-attempted the questions and read feedback again.*

Students' perceptions showed that during their interaction with TELE-EDesC, they went beyond content and focused on the goals to be set, the questions to be asked, the decisions to be made, and the monitoring of their actions. We infer that the learning activities in TELE-EDesC triggered the necessary metacognitive processes to develop the thinking skill.

How Different Users Should Use the TELoTS Framework

In the previous sections, we have described the actions and guidelines of the TELoTS framework for designing an SLE to develop thinking skills and shown a detailed example of designing an SLE for structure open problem thinking skill in the topic of amplifier design in analog electronics. In this section, we suggest how

Table 8 Summary view of the TELoTS framework

Actions and guidelines of TELoTS framework	Key points and recommendations for applying the TELoTS framework
0: Choose thinking skill, topic, and problem context	Choose thinking skill and appropriate topic to develop the thinking skill
	Iterate back and forth to decide a suitable thinking skill topic combination till *the key properties needed for the manifestation of the thinking skill are identified*
	Choose problems from a real-world context, such that they are rich and open-ended and admit multiple solution approaches
1: Characterize thinking skill	
1a: Identify competencies of thinking skill	Survey literature to find if the chosen thinking skill has been defined and characterized. Synthesize multiple characterizations if necessary
	If literature is unavailable or incomplete, conduct inductive analysis of experts' solutions as they solve a problem that requires application of the thinking skill
1b: Create learning outcomes	Apply standard guidelines for writing learning outcomes, such as use of action verbs
	Write learning outcomes corresponding to the identified competencies of the thinking skill. A learning outcome must contain elements from both topic (domain) and thinking skill competencies
	Include at least one learning outcome addressing the entire thinking skill at the integrated level
1c: Consider assessment measures	Create descriptive rubrics
	Use rubrics for formative and summative purposes
	Use a combination of analytic and holistic rubrics, i.e., rubrics should contain items for granular competencies and integrated thinking skill
2: Design learning activities	
2a: Analyze expert actions and learner needs	Do a cognitive task analysis of experts to identify underlying metacognitive processes as they apply the thinking skill to solve the problem
	Consider learners' cognitive and affective needs to inform the design of scaffolds
2b: Decide instructional strategies and supports	Choose instructional strategies so that they trigger required metacognitive processes in learners
	Some recommended strategies are multiple representations, variable manipulation and experimentation, personalization, formative assessment, and self-evaluation
	Include scaffolds to provide performance support (e.g., simplify the task) as well as learning support (e.g., problematize the task)
	Include elaboration and reflection prompts

(*continued*)

Table 8 (continued)

Actions and guidelines of TELoTS framework	Key points and recommendations for applying the TELoTS framework
2c: Identify technology features	Analyze instructional strategies and scaffolds from the perspective of functions and requirements for technology features
	Decide overall technologies which can support above features
2d: Create sequenced set of learning activities	Create learner-centric activities as far as possible
	Provide sufficient practice for each thinking skill competency; also, provide variability in the practice
	Sequence the learning activities so that they proceed from more to less scaffolded
	Provide a *synthesis and reflection activity* toward the end of each open problem, wherein learners articulate what all thinking skill competencies they used in solving the problem and how they may be useful to solve the next problem
3: Architect the components and interfaces of the SLE	Identify functional requirements based on identified technology features such as adaptation and provision for them
	Identify nonfunctional requirements such as extensibility
	Design the parts of the SLE that are critical to determine whether the main learning outcomes are likely to be met, and do user testing
	Design user interfaces and test for usability

different users can use the TELoTS framework for their goals. We first summarize the overall "what" and one way of implementing the "how" of the TELoTS framework. This is shown in Table 8, which is applicable for all users.

- If you want to develop an SLE for structure open problem thinking skill for a different topic:
  - The section "Example of Applying TELoTS Framework" is relevant, since it focuses on structure open problem thinking skill.
  - You can directly use Tables 2, 3, 4, 5, 6, and 7 in this chapter. Table 7 is especially applicable.
  - You can follow the details of creating the learning activities from Guideline 2d. The examples of *learning dialogs* (Figs. 4, 5, and 6) can be used as a template and applied to your topic. You will have to create the content of each learning activity according to your chosen topic and problem context.
  - You may change the technology if required.
- If you want to develop an SLE for a different thinking skill within engineering system design:
  - We have provided the details for Actions-1 – characterization of the thinking skill. The key thinking skills in engineering system design and their

Table 9 Thinking skills and competencies related to engineering system design

Thinking skill	Description	Competencies
Structure open problem (SOP)	Identification and formulation of problem for given specifications	SOP1 – Identify specifications from given open-ended problem
		SOP2 – Decide structure based on specifications
		SOP3 – Implement design steps sequentially
		SOP4 – Write problem statement in structured manner
Multiple representation (MR)	Constructing various valid representations while designing product and also maintaining consistency between different representations	MR1 – Construct valid representations for given problem
		MR2 – Maintain consistency between the representations
		MR3 – Apply representations to solve problem
Information gathering (IG)	Identifying relevant sources of information and using them accurately to gather relevant information	IG1 – Decide all relevant sources of information
		IG2 – Use sources to extract relevant information
Divergent thinking (DIV)	Thinking for different relevant possible solutions based on specifications, principles, and pros and cons analysis. Suggesting different solutions as well as different methods of solving the problem while considering constraints	DIV1 – Write multiple solution ideas for given problem
		DIV2 – Suggest multiple solutions based on specifications/ constraints
		DIV3 – Analyze multiple solutions based on pros and cons
		DIV4 – Analyze solutions using different problem-solving methods
Convergent thinking (CONV)	Selecting accurate solutions based on principles and constraints, justifying selected solutions, and making suitable and valid assumptions. Using formulae accurately and working out overall solution in proper steps	CONV1 – Select appropriate solution based on pros and cons analysis
		CONV2 – Select solution based on principles
		CONV3 – Justify chosen solution
		CONV4 – Evaluate solution based on constraints
		CONV5 – Write assumptions for solving the problem
		CONV6 – Justify assumptions
		CONV7 – Write complete solution using appropriate mathematical formulae

constituent competencies are shown in Table 9. The assessment rubrics for each of these competencies are available at www.et.iitb.ac.in/resources.

  - Once you choose one of the thinking skills in Table 9 as the focus of your SLE, decide an appropriate topic and problem context (Actions-0).
  - You can then apply the TELoTS framework from Actions-2 onward. You may find that some metacognitive processes in Table 6 apply, in which case you can use the corresponding instructional strategies and learning activities.
- If you want to develop an SLE for a different thinking skill not part of engineering system design, you will have to start at Actions-0, implement all the actions and guidelines of the TELoTS framework, and generate your own results from the application of the steps.

Discussion and Conclusion

We conclude this chapter with a discussion on how the design of an SLE based on the TELoTS framework may lead to learning of thinking skills and the strengths and limitations of the TELoTS framework.

Learning Trajectory in the TELoTS Framework

A difficult issue in the learning of complex concepts and skills is that the learner has to develop expertise in the constituent components of the complex concept or skill, as well as be able to understand and apply it at an integrated level. Paying attention only to the former may help the learner develop isolated understanding of the constituent components but not a holistic understanding of the entire concept or skill. Some ID models address this by gradually increasing the number and difficulty level of the constituent components of the complex concept or skill. However, this approach too will likely not work when the complex concept or skill is characterized by interactions between its components, such as in the case of thinking skills. On the other hand, only focusing on the integrated concept or skill might leave the learner daunted and without any footholds to navigate the complexity.

The TELoTS framework addresses this dual need by focusing the learners' efforts at both the constituent or granular level and the integrated level. The recommended design of the SLE is such that the learner goes back and forth between these levels and at each instance reflects on how the learning at each level is connected to that at the other. Figure 7 below shows a schematic diagram of the interactions of the learner with the SLE actions and activities.

Our conjecture of the learning trajectory in an SLE based on the TELoTS framework is as follows:

- Learners begin by encountering the topic at the integrated level, in the form of an open problem from a real-world context in which the thinking skill needs to be applied (Actions-0).
- They then move to the granular level, where they work on learning activities. Each learning activity focuses on a limited set of competencies for the thinking skill. Learners get practice in a number of learning activities for a given

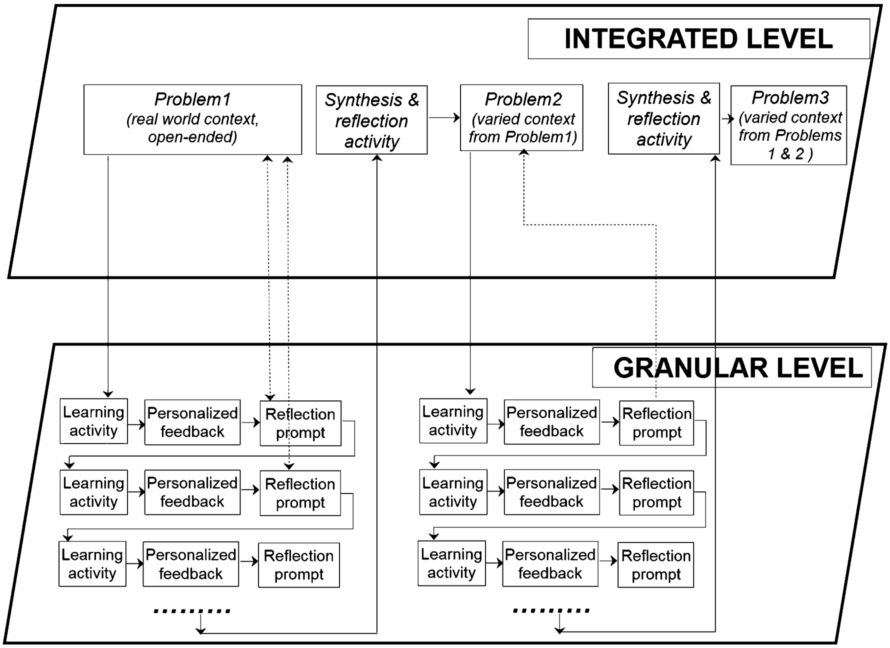

Fig. 7 Interactions of the learner with SLE at granular and integrated levels

competency, in varied contexts. For example, they may work on decision-making tasks in a multiple subtopics relevant to the open problem.

- Each learning activity is accompanied by personalized feedback based on the learners' actions, and reflection prompts in which the learner reflects on which competencies were applied, and how the learning activity helps in solving the open problem. Thus, there is a brief "visit" to the integrated level. While the learner spends a lot of time in practice at the granular level, there are repeated back-and-forth visits between the levels at this stage.
- Toward the end of a problem, the learner explicitly reflects on which competencies were required to solve the problem, and they may be useful to address the next problem in the SLE. The next problem is sequenced such that it requires the application of similar thinking skills as in the previous problem but possibly in a varied context. This overall reflection activity is crucial for potential of transfer, i.e., for the learner to be able to inductively abstract the thinking skills and concepts required to solve a problem in a new topic.
- Recommendations for the assessment too involve targeting both the granular competency levels (analytic rubrics items) and the integrated level of application of the thinking skill (solve open problem requiring the thinking skill in a new context).

Strengths and Limitations

One of the key strengths of the TELoTS framework is that it attempts to address both the granular competencies of a thinking skill as well as in its integrated sense. The framework does so by providing the learner frequent back-and-forth experiences of applying the thinking skill at both levels. Secondly, since the focus of the SLE is thinking skills, the TELoTS framework prescribes an explicit learning outcome which focuses learners' attention on the abstractions of the thinking skill from a given context, so that they can apply it in other contexts. Corresponding to this learning outcome, the TELoTS framework prescribes assessment measures and learning activities to be included in the SLE.

Currently, one limitation of the TELoTS framework is its scope of application. Some actions of the framework, such as the characterization of the thinking skill, have been applied and evaluated for different thinking skills relevant to engineering system design. Other actions, such as identifying productive instructional strategies, have only been implemented and evaluated in specific thinking skills within engineering system design. As an illustrative example of the latter, this chapter described the application of the TELoTS framework to structure open problem thinking skill. It has also been applied to other thinking skills such as multiple representation (Mavinkurve & Murthy, 2015). Another limitation related to scope is the choice of topic and problem. Within a given domain and topic, the application of the TELoTS framework for a thinking skill may occur in one manner. We have yet to determine exactly how the application of the framework may differ for a different domain. For example, if an SLE designer wishes to teach structure open problem thinking skill in a new topic, say computer networking, to what extent would the learning dialogs be similar to the ones created for the topic of amplifier design? This needs to be rigorously tested.

Overall, the TELoTS framework provides SLE designers one starting point to conceptualize and design learning environments that explicitly address thinking skills, by systematically and effectively making use of the affordances of current technologies.

Acknowledgments We thank Aditi Kothiyal, M. Sasikumar, Mrinal Patwardhan, Lakshmi Ganesh, Soumya Narayana, Jayakrishnan M. and Shitanshu Mishra for suggestions, discussions on various parts of this article and help with the references.

References

ABET (2014). *Criteria for accrediting Engineering Programs* http://www.abet.org/eac-criteria-2014-2015/

Anderson, L. W., Krathwohl, D. R., & Bloom, B. S. (2001). *A taxonomy for learning, teaching, and assessing: A revision of Bloom's taxonomy of educational objectives*. Boston: Allyn & Bacon.

Arter, J., & McTighe, J. (2001). *Scoring rubrics in the classroom: Using performance criteria for assessing and improving student performance*. Thousand Oaks, CA: Corwin Press.

Aurisicchio, M., Ahmed, S., & Wallace, K. M. (2007). Improving design understanding by analyzing questions. In ASME *International Design Engineering Technical Conferences and Computers and Information in Engineering Conference* (pp. 307–317).

Black, P., & Wiliam, D. (1998). *Inside the black box: Raising standards through classroom assessment*. London: Granada Learning.

Boekaerts, M., & Boscolo, P. (2002). Interest in learning, learning to be interested. *Learning and Instruction, 4*(12), 375–382.

Bransford, J. D., & Schwartz, D. L. (1999). Rethinking transfer: A simple proposal with multiple implications. *Review of Research in Education, 24,* 61–100.

Catrambone, R., & Holyoak, K. J. (1989). Overcoming contextual limitations on problem-solving transfer. *Journal of Experimental Psychology. Learning, Memory, and Cognition, 15*(6), 1147.

Chen, Y. L., Hong, Y. R., Sung, Y. T., & Chang, K. E. (2011). Efficacy of simulation-based learning of electronics using visualization and manipulation. *Journal of Educational Technology & Society, 14*(2), 269–277.

Clark, R. E., Feldon, D., van Merriënboer, J. J., Yates, K., & Early, S. (2008). Cognitive task analysis. *Handbook of Research on Educational Communications and Technology, 3,* 577–593.

Cobb, P., Confrey, J., Lehrer, R., & Schauble, L. (2003). Design experiments in educational research. *Educational Researcher, 32*(1), 9–13.

Collins, A. (2006). Cognitive apprenticeship. In R. K. Sawyer (Ed.), *The Cambridge handbook of the learning sciences* (pp. 47–60). Cambridge University Press.

Conklin, J., & Begeman, M. L. (1988). *gIBIS: A hypertext tool for exploratory policy discussion* (pp. 140–152). New York: ACM Press. Proceedings of the ACM Conference on Computer-supported Cooperative Work (CSCW ’88).

Crain, R.W., Davis, D.C., Calkins, D.E., & Gentili, K. (1995). Establishing engineering design competencies for freshman/sophomore students. In *Proceedings of Frontiers in Education Conference,* Vol. 2, 4d2-1.

Cross, N. (2007). From a design science to a design discipline: Understanding designerly ways of knowing and thinking. In R. Michel (Ed.), *Design research now* (pp. 41–54). Basel : Birkhäuser.

D’Mello, S., & Graesser, A. (2012). Dynamics of affective states during complex learning. *Learning and Instruction, 22*(2), 145–157.

Davis, D.C., Crain, R.W., Trevisan, M.S., Calkins, D.E., & Gentili, K.L. (1997). Categories and levels for defining engineering design program outcomes. In *Proceedings of 1997 Annual Meeting of the American Society for Engineering Education.*

De Corte, E. (2003). Transfer as the productive use of acquired knowledge, skills, and motivations. *Current Directions in Psychological Science, 12*(4), 142–146.

Dillenbourg, P., Järvelä, S., & Fischer, F. (2009). The evolution of research on computer-supported collaborative learning: From design to orchestration. In N. Balacheff, S. Ludvigsen, T. Jong, A. Lazonder, & S. Barnes (Eds.), *Technology-enhanced learning* (pp. 3–19). New York: Springer.

Dym, C. L., Agogino, A. M., Eris, O., Frey, D. D., & Leifer, L. J. (2005). Engineering design thinking, teaching, and learning. *Journal of Engineering Education, 94*(1), 103–120.

Etkina, E., & Van Heuvelen, A. (2007). Investigative science learning environment – A science process approach to learning physics. In E. F. Redish & P. Cooney (Eds.), *PER-based reforms in calculus-based physics*. College Park, MD: AAPT.

Etkina, E., Van Heuvelen, A., White-Brahmia, S., Brookes, D. T., Gentile, M., Murthy, S., . . . Warren, A. (2006). Scientific abilities and their assessment. *Physical Review special topics-physics education research*, *2*(2), 020103.

Facione, P. A. (1990). Critical thinking: A statement of expert consensus for purposes of educational assessment and instruction. The Delphi Report: Research findings and recommendations prepared for the American Philosophical Association. *ERIC ED, 315,* 423.

Garrett, J. J. (2011). *The elements of user experience: User-centered design for the web and beyond* (2nd ed.). Berkeley, CA: Pearson Education.

Ge, X., & Land, S. M. (2004). A conceptual framework for scaffolding III-structured problem-solving processes using question prompts and peer interactions. *Educational Technology R&D, 52*(2), 5–22.

Gentner, D., Loewenstein, J., & Thompson, L. (2003). Learning and transfer: A general role for analogical encoding. *Journal of Educational Psychology, 95*(2), 393.

Gresch, H., Hasselhorn, M., & Bögeholz, S. (2013). Training in Decision-making Strategies: An approach to enhance students' competence to deal with socio-scientific issues. *International Journal of Science Education, 35*(15), 2587–2607.

ISTE (2014). International Society for Technology in Education. Computational thinking for all, retrieved from http://www.iste.org/learn/computational-thinking, Oct. 2015.

Jackson, S., Krajcik, J., & Soloway, E. (2000). Model-It™: A design retrospective. In M. Jacobson & R. Kozma (Eds.), *Advanced designs for the technologies of learning: Innovations in science and mathematics education*. Hillsdale, NJ: Erlbaum.

Jacobs, J. E., & Paris, S. G. (1987). Children's metacognition about reading: Issues in definition, measurement, and instruction. *Educational Psychologist, 22,* 255–278.

Johnson, D. W., & Johnson, R. T. (1999). *Learning together and alone: Cooperative, competitive, and individualistic learning* (5th ed.). Boston: Allyn & Bacon.

Jonassen, D. H. (1997). Instructional design models for well-structured and ill-structured problem-solving learning outcomes. *Educational Technology R&D, 45*(1), 65–94.

Jonassen, D., Strobel, J., & Lee, C. B. (2006). Everyday problem solving in engineering: Lessons for engineering educators. *Journal of Engineering Education, 95*(2), 139–151.

Jonassen, D. H., Tessmer, M., & Hannum, W. H. (1999). *Task analysis methods for instructional design*. Mahwah, NJ: Lawrence Erlbaum Associates.

Kinnebrew, J. S., Segedy, J. R., & Biswas, G. (2014). Analyzing the temporal evolution of students' behaviors in open-ended learning environments. *Metacognition Learning, 9,* 187–215.

Kinshuk (2012). Guest editorial: Personalized learning. *Educational Technology R&D, 60*(4), 561–562.

Kinshuk, C. M., Graf, S., & Yang, G. (2010). Adaptivity and personalization in mobile learning. *Technology, Instruction, Cognition and Learning, 8*(2), 163–174.

Koper, R. (2014). Conditions for effective smart learning environments. *Smart Learning Environments*, *1*(5). doi:10.1186/s40561-014-0005-4.

Linn, M. C., Clark, D., & Slotta, J. D. (2003). WISE design for knowledge integration. *Science Education, 87*(4), 517–538.

Lipman, M. (2003). *Thinking in education*. New York: Cambridge University Press.

Mavinkurve, M. & Murthy, S (2012). Interactive visualizations to teach design skills. In *Proceedings of the 20th International Conference on Computers in Education, ICCE 2012,* Singapore.

Mavinkurve, M. & Murthy, S. (2013). Comparing self-learning behavior of low and high scorers with EDIV. In *Proceedings of the 21th International Conference on Computers in Education*, Bali, Indonesia.

Mavinkurve, M. & Murthy, S. (2015). Development of engineering design competencies using TELE-EDesC: Do the competencies transfer? In *Proceedings of the 15th IEEE International Conference on Advanced Learning Technologies,* Athens, Greece.

Merrill, M. D. (2002). First principles of instructional design. *Educational Technology Research and Development, 50,* 43–59.

Mertler, C.A. (2001). Designing scoring rubrics for your classroom. *Practical Assessment, Research & Evaluation*, 7(25).

Minstrell, J., & Van Zee, E. (Eds.). (2000). *Inquiring into inquiry learning and teaching in science*. Washington, DC: American Association for the Advancement of Science.

Mishra, P., Koehler, M. J., & Henriksen, D. (2011). The seven trans-disciplinary habits of mind: Extending the TPACK framework towards 21st century learning. *Educational Technology, 51* (2), 22–28.

Moskal, B.M. (2000). Scoring rubrics: What, when and how? *Practical Assessment, Research & Evaluation*, 7(3). Retrieved Oct. 2015 from http://pareonline.net/getvn.asp?v=7&n=3.

Moskal, B. M., & Leydens, J. A. (2000). Scoring rubric development: Validity and reliability. *Practical Assessment, Research & Evaluation, 7*(10). Retrieved April 2016 from http://PAREonline.net/getvn.asp?v=7&n=1.

NAE (2005). Educating the Engineer of 2020: Adapting Engineering Education to the New Century, Committee on the Engineer of 2020, Phase II, Committee on Engineering Education, National Academy of Engineering of the National Academies, Washington DC.

NAS (2014). *Preparing for 21st century, the education imperative*. Retrieved from http://www.nas.edu/21st/education, Oct. 2015.

Padilla, M.J. (1990). The science process skills. Research matters – to the science teacher, No. 9004.

Pellegrino, J., & Hilton, M. L. (2012). In National Research Council (Ed.), *Education for life and work: Developing transferable knowledge and skills in the 21st century*. Washington, DC: The National Academies Press.

Perkins, D. N., & Salomon, G. (1992). Transfer of learning. International encyclopedia of education, 2.

Pressley, M., & McCormick, C. B. (1987). *Advanced educational psychology for educators, researchers, and policy makers*. New York: HarperCollins.

Pressman, R. S. (2005). *Software engineering: A practitioner's approach*. New York: Palgrave Macmillan.

Quintana, C., Reiser, B.J., Davis, E. A., Krajcik, J., Fretz, E., Duncan R. G., . . . Soloway, E. (2004) A scaffolding design framework for software to support science inquiry. *Journal of the Learning Sciences*, 13(3), 337-386.

Reeves, T. (2006). Design research from the technology perspective. In J. Van den Akker, K. Gravemeijer, S. McKenney, & N. Nieveen (Eds.), *Educational design research*. London: Routledge.

Reiser, B. J. (2004). Scaffolding complex learning: The mechanisms of structuring and problematizing student work. *Journal of the Learning Sciences, 13,* 273–304.

Sandoval, W. (2014). Conjecture mapping: An approach to systematic educational design research. *Journal of the Learning Sciences, 23*(1), 18–36.

Scardamalia, M., & Bereiter, C. (2014). Smart technology for self-organizing processes. *Smart Learning Environments*, 1(1).

Scheuer, O., Loll, F., Pinkwart, N., & McLaren, B. M. (2010). Computer-supported argumentation: A review of the state of the art. *International Journal of Computer-Supported Collaborative Learning*, *5*(1), 43–102.

Schwartz, D. L., & Martin, T. (2004). Inventing to prepare for future learning: The hidden efficiency of encouraging original student production in statistics instruction. *Cognition and Instruction, 22*(2), 129–184.

Sheppard, S., & Jennison, R. (1997). Freshman engineering design experiences and organizational framework. *International Journal of Engineering Education, 13,* 190–197.

Sheppard, S., Macatangay, K., Colby, A., & Sullivan, W. M. (2009). *Educating Engineers: Designing for the future of the field*. San Francisco: Jossey-Bass.

Spector, J. M. (2014). Conceptualizing the emerging field of smart learning environments. *Smart Learning Environments, 1*(1), 1–10.

Suthers, D. D., Connelly, J., Lesgold, A., Paolucci, M., Toth, E. E., Toth, J., & Weiner, A. (2001). Representational and advisory guidance for students learning scientific inquiry. In K. D. Forbus & P. J. Feltovich (Eds.), *Smart machines in education: The coming revolution in educational technology* (pp. 7–35). Menlo Park, CA: AAAI/MIT Press.

van Joolingen, W. R., de Jong, T., Lazonder, A. W., Savelsbergh, E. R., & Manlove, S. (2005). Co-Lab: Research and development of an online learning environment for collaborative scientific discovery learning. *Computers in Human Behavior, 21*(4), 671–688.

van Merriënboer, J. J. G., & Kirschner, P. A. (2012). *Ten steps to complex learning: A systematic approach to four component instructional design* (2nd ed.). New York/London: Routledge.

Veenman, M. (2012). Metacognition in science education: Definitions, constituents, and their intricate relation with cognition. In *Metacognition in science education* (pp. 21–36). Netherlands: Springer.

Vygotsky, L. S. (1978). In M. Cole, V. John-Steiner, S. Scribner, & E. Souberman (Eds.), *Mind in society: The development of higher psychological processes*. Cambridge, MA: Harvard University Press.

Wiggins, G. (1990). The case for authentic assessment. The case for authentic assessment. *Practical Assessment, Research & Evaluation*, 2(2). Retrieved Oct. 2015 from http://pareonline.net/getvn.asp?v=2&n=2

Wiggins, G. P., & McTighe, J. (2005). *Understanding by design*. Alexandria, VA: Association for Supervision and Curriculum Development.

Williams, R. F. (2006). Using cognitive ethnography to study instruction. In *Proc. of the 7th International Conference on Learning Sciences* (pp. 838–844). International Society of the Learning Sciences.

Woods, D. R., Hrymak, A. N., Marshall, R. R., Wood, P. E., Crowe, C. M., Hoffman, T. W., … Bouchard, C. G. (1997). Developing problem solving skills: The McMaster problem solving program. *Journal of Engineering Education*, 86(2), 75–91.

Sahana Murthy is an associate professor in the Inter-disciplinary Program in Educational Technology at IIT Bombay since 2009. Prior to that, she was a lecturer at the Experimental Study Group in MIT from 2006 to 2009, during which she implemented and evaluated innovative teaching methods. She worked as a postdoctoral researcher in Physics Education Research at MIT (2005–2006) and Rutgers University (2004–2005), USA. She got her PhD in Physics from Rutgers University in 2004. Her current research interests in educational technology are in students' development of thinking skills through technology-enhanced learning environments. She has conducted large-scale blended-mode training programs on ET4ET – Educational Technology for Engineering Teachers – and Research Methods in Educational Technology, via the "Train 10000 Teachers (T10KT)" program under the Indian government's National Mission on Education through ICT.

Sridhar Iyer is a professor in the Dept. of Computer Science and Engineering at IIT Bombay. His current research interests are in the field of educational technology. This includes technology-enhanced learning environments for thinking skills, pedagogies for effective use of educational technologies, development of ICT tools for educational applications, and computer science education research. Prior to Educational Technology, he has worked in wireless networking protocols and mobile applications. Sridhar Iyer received his BTech, MTech, and PhD from the Dept. of Computer Science and Engineering at IIT Bombay. More information about him is available from his Web page: ▶ www.cse.iitb.ac.in/~sri

Madhuri Mavinkurve has a B.E. in Electronics Engg. from Shivaji University and M.E. in Electronics from Mumbai University. She recently submitted her PhD thesis in Educational Technology at IIT Bombay. She is currently an associate professor in the Electronics and Telecommunication Department at Thakur College of Engineering, Mumbai, and has over 25 years teaching experience. Her research includes the development of technology-enhanced learning environment to teach engineering design thinking skill. She has organized and taught several faculty professional development workshops on research-based instructional strategies, including Educational Technology for Engineering Teachers for 3,000 college instructors, via the Teach 10,000 Teachers project under the Indian government's National Mission in Education through ICT.