EDUCATIONAL PSYCHOLOGIST, *38*(1), 5–13

Taking the Load Off a Learner's Mind: Instructional Design for Complex Learning

Jeroen J. G. van Merriënboer, Paul A. Kirschner, and Liesbeth Kester
Educational Technology Expertise Center
Open University of The Netherlands, Heerlen

Complex learning aims at the integration of knowledge, skills, and attitudes; the coordination of qualitatively different constituent skills; and the transfer of what is learned to daily life or work settings. Recent instructional theories stress authentic learning tasks as the driving force for learning; but due to the complexity of those tasks, learning may be hampered by the limited processing capacity of the human mind. In this article we present a framework for scaffolding practice and just-in-time information presentation, aiming to control cognitive load effectively. We briefly describe a design model for complex learning consistent with cognitive load theory. Theoretical and practical implications of the presented framework are discussed.

Recent instructional theories tend to focus on authentic learning tasks that are based on real-life tasks as the driving force for learning (Merrill, 2002; Reigeluth, 1999a; van Merriënboer & Kirschner, 2001). The general assumption is that such tasks help learners to integrate the knowledge, skills, and attitudes necessary for effective task performance; give them the opportunity to learn to coordinate constituent skills that make up complex task performance; and eventually enable them to transfer what is learned to their daily life or work settings. This focus on authentic, whole tasks can be found in practical educational approaches, such as project-based education, the case method, problem-based learning, and competency-based learning; and in theoretical models, such as Collins, Brown, and Newman's (1989) theory of cognitive apprenticeship learning, Jonassen's (1999) theory of constructive learning environments, Nelson's (1999) theory of collaborative problem solving, and Schank, Berman, and MacPerson's (1999) theory of goal-based scenario.

A severe risk of all of these approaches is that learners have difficulties learning because they are overwhelmed by the task complexity. The aim of this article is to discuss managing cognitive load when rich learning tasks are used in education. First, methods for scaffolding whole-task practice are discussed, including simple-to-complex sequencing of learning tasks and the use of alternative tasks, such as worked-out examples and completion tasks. Second, methods for just-in-time information presentation are discussed, including timely presentation of information to support practice on learning tasks and the direct, step-by-step presentation of procedural information. Third, we briefly sketch an instructional design model for complex learning fully consistent with cognitive load theory (CLT). We conclude that CLT offers useful guidelines for decreasing intrinsic and extraneous cognitive load, so that sufficient processing capacity is left for genuine learning.

SCAFFOLDING WHOLE-TASK PRACTICE

Scaffolds, according to their original meaning within educational psychology, include all devices or strategies that support students' learning (Rosenshine & Meister, 1992). In both cognitive apprenticeship learning and our framework, *scaffolding* explicitly pertains to a combination of performance support and fading. Initially, the support enables a learner to achieve a goal or action not achievable without that support. When the learner achieves the desired goal, support gradually diminishes until it is no longer needed. Because excessive or insufficient support can hamper the learning process, it is critical to determine the right type and amount of support and to fade at the appropriate time and rate. Many types of support share the common characteristic that they do not direct the learner, as one can do when teaching an algorithm (i.e., procedure support), but rather guide the learner during his or her work on complex learning tasks (i.e., problem-solving sup-

Requests for reprints should be sent to Jeroen J. G. van Merriënboer, Open University of The Netherlands, Educational Technology Expertise Center, P.O. Box 2960, NL–6401 DL Heerlen, The Netherlands. E-mail: jeroen.vanmerrienboer@ou.nl port). Coaching by providing hints, prompts, and feedback; modeling the use of cognitive strategies by thinking aloud; presenting cue cards, checklists, and process worksheets; asking leading questions; and giving part of a solution are all examples of such support.

CLT emphasizes the need to integrate support for novice learners with the task environment fully; otherwise, split-attention effects increase extraneous cognitive load because learners have to integrate information mentally from the task environment with the given support (Chandler & Sweller, 1992; Sweller & Chandler, 1994). This effect may explain the common finding that learners who need the most support are least inclined to use nonintegrated support systems, such as manuals or job aids (Carroll & Rosson, 1987). When novice learners encounter problems while working on a learning task, the last thing they are inclined to do is further increase their already high-cognitive load by processing and mentally integrating additional information from a support system. They act as typical computer users encountering software problems: When all else fails, consult the documentation. Instead, CLT stresses that support for novice learners must be fully embedded in the learning tasks or task environment to be effective. Performance constraints such as training wheels on children's bikes are a good example of this (Carroll & Carrithers, 1984; Leutner, 2000; van Merriënboer, 2000). Such embedded support is clearly more effective than nonintegrated support (i.e., the parent running alongside and shouting "keep your handlebars straight!").

The next sections discuss two complementary approaches for scaffolding whole-task practice through fully embedded support. The first approach identifies simple-to-complex versions of the whole task to decrease *intrinsic cognitive load;* novice learners start to practice on the simplest version of the whole task encountered by experts in the real world and progress toward increasingly more complex versions. The second approach uses learning tasks that decrease *extraneous cognitive load* for novice learners; for instance, they may start to work on worked-out examples, then complete increasingly larger parts of incomplete, given solutions, and finally work on conventional tasks.

Simple-to-Complex Sequencing

It is clearly impossible to use highly complex learning tasks from the start of a course or training program because this would yield excessive cognitive load for the learners, with negative effects on learning, performance, and motivation (Sweller, van Merriënboer, & Paas, 1998). The common solution is to let learners start their work on relatively simple learning tasks and progress toward more complex tasks. Instructional theories developed in the 1960s and 1970s (see Reigeluth, 1983) typically advocated the use of part-task approaches to avoid early exposure to tasks that are too complex. Complex performances are broken down into simpler parts that are trained separately or, in a part-whole approach, are gradually combined into whole-task performance. It is not until the end of the training program that learners have the opportunity to practice the whole task. This sequencing strategy fits well with instructional design theories driven by separate instructional objectives or learning goals. Each objective corresponds with one of the part tasks, and developing practice for all instructional objectives naturally results in a part-task sequence.

Part-task approaches to sequencing are highly effective to prevent cognitive overload because the load associated with a part of the task is lower than the load associated with the whole task. But Naylor and Briggs (1963) already indicated in the early 1960s that they are not very suitable for learning complex tasks requiring much coordination between their distinct parts or constituent skills. Since then, there has been accumulating evidence that both part-task approaches to sequencing and instructional design models driven by separate objectives do not work well for complex performances that require the integration of skills, knowledge, and attitudes and the extensive coordination of constituent skills in new problem situations (see van Merriënboer, 1997, for a complete discussion). Such instruction does not effectively deal with integrated objectives and is too piecemeal and fragmented to allow for transfer to new problem situations. For this reason, current instructional theories increasingly focus on whole-task instead of part-task approaches to sequencing.

Whole-task approaches attend to the coordination and integration of constituent skills from the very beginning, and they stress that learners quickly develop a holistic vision of the whole task that is gradually embellished during the training. This approach to skills development is akin to the "global before local skills" principle used in cognitive apprenticeship (Collins et al., p. 485) and Reigeluth's (1999b) elaboration theory. For instance, elaboration theory describes a *simplifying conditions approach* to sequencing. All conditions that simplify the performance of the task are identified, and instruction starts with the most simple but authentic case that a professional might encounter in the real world. A simple illustration might be given for the task "searching for literature," for which simplifying conditions pertain to (a) clearness of the concept definitions within or between domains (ranging from clear to unclear); (b) number of articles that are written about the topic (ranging from small to large); (c) number of domains in which relevant articles have been published and, hence, the number of databases that need to be searched (ranging from one familiar database to many databases that are relevant for the topic of interest); (d) type of search (ranging from a search on titles and key words to abstracts and full text); and (e) number of search terms and Boolean operators used (ranging from a few search terms to many search terms that are interconnected with Boolean operators). Given these conditions, simple versions of the whole task confront learners with situations in which the search is performed in a domain in which the concepts are clearly defined, on titles and keywords in one particular database, with only few search terms and yielding a limited number of relevant articles. The most complex versions confront learners with situations in which concept definitions within or between domains are unclear and in which full-text searches have to be performed in several relevant databases and with many search terms interconnected by Boolean operators to limit the otherwise large number of relevant articles. Versions of the task with an intermediate complexity can be constructed by varying one or more of the simplifying conditions.

Whole-task approaches to sequencing stress that there are categories of learning tasks, each one representing a version of the task with a particular complexity. White and Frederiksen (1990) based simple-to-complex versions of a whole task on a progression of mental models; they called the categories *problem sets* and noted that the learning tasks within a higher order problem set require a more elaborated mental model for their performance. Scandura (1983) based simple-to-complex versions of a task on increasingly more detailed cognitive strategies; he called the categories *equivalence classes* and noted that the learning tasks within a following equivalence class require a more detailed cognitive strategy for their performance. Van Merriënboer (1997) introduced the broader concept of *task classes* to define simple-to-complex categories of learning tasks. Learning tasks within a particular task class are equivalent in the sense that the tasks can be performed on the basis of the same body of generalized knowledge (e.g., mental models, cognitive strategies, or other cognitive schemas). A more complex task class requires more knowledge or more embellished knowledge for effective performance than the preceding, simpler task classes.

Learning tasks within a simpler task class are associated with a lower intrinsic cognitive load than learning tasks within a more complex task class (Bannert, 2002). For a simpler task class, underlying schemas do not yet contain the elements that are needed when the simplifying conditions are relaxed. Less elements and interactions between elements need to be processed simultaneously in working memory when performing the tasks in simpler task classes; for more complex task classes, the number of elements and interactions between elements increases (cf. Pollock, Chandler, & Sweller, 2002). Each new task class contains learning tasks that are in the zone of proximal development of the learners (Vygotsky, 1934/1987). When learners start to work on a new task class, it is typically essential to lower extraneous cognitive load through the introduction of adequate support, as explained in the next section.

Types of Learning Tasks

Learning tasks are often equated with conventional problems. Such tasks confront the learner with a given state and a set of criteria for an acceptable goal state. The learner needs to be engaged in some kind of problem-solving process, which can be described as the tentative application of mental operations to generate a solution, that is, a sequence of operators that enables the transition from the given state to an acceptable goal state (Newell & Simon, 1972). There is overwhelming evidence that such conventional tasks are exceptionally expensive in terms of working memory capacity. They cause high-extraneous cognitive load due to the use of weak problem-solving methods (e.g., means–ends analysis), which bear little relation to schema construction processes that are concerned with learning to recognize problem states and their associated actions (Sweller, 1988). For novice learners, learning and performing conventional tasks are different and incompatible processes. Researchers used CLT to devise alternative formats for learning tasks to alter learner activities in a manner that reduces the extraneous cognitive load caused by conventional problem solving and to encourage schema construction processes.

Learning tasks that take the form of *worked-out examples* confront learners not only with a given state and a desired goal state but also with an example solution. Studying those examples as a substitute for performing conventional problem-solving tasks may be beneficial, because it focuses attention on problem states and associated solution steps and so enables learners to induce generalized solutions or schemas. As the use of capacity-demanding, weak problem-solving methods is eliminated and nothing else has to be attended to, extraneous cognitive load is low. Indeed, many experiments supported the counterintuitive prediction that studying worked-out examples facilitates schema construction and transfer performance more than actually solving the equivalent problems does (van Gerven, Paas, van Merriënboer, & Schmidt, 2002; also see Sweller et al., 1998, for an overview).

A disadvantage of worked-out examples is that they do not force learners to study them carefully (Renkl, Stark, Gruber, & Mandl, 1998). Learners may only briefly look at the worked-out examples and only consult them when they have difficulties in performing their tasks. Consulting worked-out examples simultaneously to performing a task, however, requires that both the worked-out example and the task to be simultaneously processed in working memory, resulting in a high-extraneous cognitive load. As an alternative, van Merriënboer and Krammer (1987) and van Merriënboer and Paas (1989) suggested the use of *completion tasks* that present a given state, a goal state, and a partial solution to the learners who must complete the solution. Completion tasks combine the strong points of worked-out examples and conventional learning tasks. Like conventional learning tasks, they directly encourage learners to be active because learners have to complete the solution, which is only possible by the careful study of the partial example provided in the completion task. And like worked-out examples, they decrease extraneous cognitive load (van Merriënboer, Schuurman, de Croock, & Paas, 2002). Several experiments have reported the superiority of completion tasks over conventional tasks (e.g., Paas, 1992; Stark, 1999; van Merriënboer, 1990; van Merriënboer & de Croock, 1992).

Table 1 provides illustrations of a conventional task, a worked-out example, and a completion task for searching for relevant research literature. In addition, examples of a goal-free task and a reverse task are shown. Goal-free tasks present the learner with a highly aspecific goal, which eliminates the need for means–ends search and so decreases extraneous cognitive load (Paas, Camp, & Rikers, 2001; also see Sweller et al., 1998, for an overview). Reverse tasks present the learner with both a goal and a solution, and they ask for which given situations the solution may be helpful to reach the goal state (Halff, 1993). Different types of learning tasks thus range from tasks with high built-in support (worked-out examples), via tasks with an intermediate level of support (completion, goal-free, and reverse tasks), to conventional tasks without support.

The fading principle suggests giving sizable early support for learning tasks and no support for the final tasks in the same task class. This can be easily realized if one starts with worked-out examples or completion assignments with almost complete solutions that must be studied, continues with completion assignments for which larger and larger parts of the solution must be completed, and ends with conventional problems. Several studies showed positive effects on learning for this so-called *completion strategy* (Renkl, Maier, Atkinson, & Staley, 2002; Stark, 1999; van Merriënboer & de Croock, 1992). Fading is also important because the positive effects found for worked-out examples may reverse as learners acquire more expertise: If the necessary schemas for solving a problem are already available, analyzing a redundant worked-out example and integrating it with previously acquired schemas in working memory may impose higher cognitive load than problem solving (see Kalyuga, Ayres, Chandler, & Sweller, 2003, for an overview of the expertise reversal effect).

In summary, we discussed two approaches for scaffolding whole-task practice. First, categories of simple-to-complex versions of the whole task are differentiated into task classes. Simpler task classes are associated with a lower intrinsic cognitive load than more complex ones, and each new task class contains learning tasks that are in the learners' proximal zone of development. Second, particular types of learning tasks such as worked-out examples, goal-free problems, or completion tasks are associated with a lower extraneous cognitive load than conventional problem solving. Within one task class, cognitive load may be further controlled by starting the task class with worked-out examples, progressing with low-load tasks, and ending with conventional problems.

JUST-IN-TIME INFORMATION PRESENTATION

In the previous section we discussed approaches for scaffolding learners' whole-task practice on complex learning tasks. In addition to such scaffolds, learners need relevant task-specific information to perform the learning tasks and to learn from them. If, for example, students are confronted with a learning task requiring them to find relevant research literature for the first time, they need information on how to approach this task (e.g., select one or more appropriate databases, formulate a search query, perform the search using appropriate search tools, and select relevant results), how typical bibliographical databases are organized (e.g., with separate fields for titles, abstracts, authors, keywords, etc.), and which procedures are useful for operating search programs or composing search queries with Boolean operators. Without such information it will be very difficult, if not impossible, to carry out the learning task(s). This necessary information must be active in working memory when performing the task to guide or influence behavior. Simply stated, there are two ways of reaching this goal. One way is to present necessary information before the learners start working on the learning task or series of tasks. They study this information prior to be-

TABLE 1
Examples of Different Types of Learning Tasks for the Searching for Relevant Research Literature Task

Learning Task	*Given*	*Goal*	*Solution*	*Task Description*
Conventional task	+	+	Find	Learners receive a research question and the goal to produce a list with a limited number of relevant articles. They must formulate the search query, perform the search, and make a selection of articles.
Worked-out example	+	+	+	Learners receive a research question, a list with articles, and a search query used to produce the list of articles. They must evaluate the quality of the search query and the list of articles.
Completion task	+	+	Complete	Learners receive a research question, the goal to produce a list with a limited number of relevant articles, and an incomplete search query. They must complete the search query, perform the search, and make a selection of articles.
Goal-free task	+	Define	Find	Learners receive a research question and a highly aspecific goal, for instance, to come up with as many search queries as possible that might be relevant to the research question. They must formulate those search queries.
Reverse task	Predict	+	+	Learners receive a list with articles and a search query used to produce the list of articles. They must predict possible research questions for which the list of articles and search query are relevant.

ginning so that it is encoded in schemas in their long-term memory and subsequently activated in working memory when needed to perform the task. The other way is to present the necessary information precisely when the learners need it during task performance. They do not learn it beforehand, but the external information is directly activated in working memory when it is necessary to perform the learning task.

CLT does not yield an unequivocal answer to the question of which of the two ways is best. For information with a high-intrinsic complexity, it seems advisable to present the information before the learners start on the learning tasks. Because learners have little cognitive capacity left for additional processing while working on the tasks, the simultaneous processing of intrinsically complex information can easily lead to cognitive overload. If the information is studied beforehand, a cognitive schema may be constructed in long-term memory that can subsequently be activated in working memory during task performance. Low-complexity information, however, may better be presented precisely when learners need it during their work on the learning tasks. Because of its low complexity, there is no or little risk of cognitive overload.

In the next sections we argue that the complexity of to-be-presented information and its optimal timing in instruction are closely related to the nature of different task aspects. The intrinsic complexity of information supporting the performance of variable, nonrecurrent task aspects is typically higher than that for information specifying the performance of highly consistent, recurrent task aspects. In our theoretical framework, the first is called *supportive information* and is best presented before equivalent learning tasks; it pertains to the knowledge necessary for problem solving and reasoning. The second is called *procedural information* and is best presented precisely when it is needed during task performance; it pertains to the how-to instructions for the application of rules (Kester, Kirschner, van Merriënboer, & Baumer, 2001).

Supportive Information

For expert task performers, some task aspects require reasoning and problem solving, whereas other aspects are performed as routines. These are called variable and consistent task components (Fisk & Gallini, 1989) or nonrecurrent and recurrent task aspects (van Merriënboer, 1997), respectively. Experts can effectively perform nonrecurrent task aspects because they have cognitive schemas available to reason about the domain and guide their problem solving. An experienced researcher searching for relevant research literature can, for example, reason about the effectiveness of different queries thanks to a well-developed mental model of the organization of bibliographical databases; and he or she can effectively approach new search tasks thanks to the availability of cognitive strategies for translating research questions into relevant search terms. Such mental models and cognitive strategies are examples of complex schemas with a general or abstract nature. They allow for multiple use of the same general knowledge for performing different tasks. Teachers often call this supportive information the *theory*.

How does one help novice learners construct mental representations such as mental models and cognitive strategies? Because the information describing mental models (e.g., how bibliographical databases are organized) and cognitive strategies (e.g., phases and rules of thumb for translating research questions into relevant search terms) typically has a high-intrinsic complexity, it is preferable not to present it to learners while they are working on the learning tasks. Simultaneously performing the tasks and studying the information would almost certainly cause cognitive overload (Kester et al., 2001; Marcus, Cooper, & Sweller, 1996). Instead, supportive information is best presented before learners start working on the learning tasks. Then, a cognitive schema may be constructed in long-term memory that can subsequently be activated in working memory during task performance. Retrieving the already constructed schema is expected to be less cognitively demanding than activating the externally presented complex information in working memory during task performance.

It is critical that helpful cognitive schemas are constructed in a process of *elaboration* whereby nonarbitrary relations are established between new information elements and the learner's prior knowledge. This allows for structural understanding and ensures that the schemas provide a bridge between what learners already know and what they need to know to perform the learning tasks. Furthermore, it allows for chunking so that a newly constructed schema can be treated as one information item (Gobet et al., 2001; Miller, 1956) and is thus easily retrievable from long-term memory and accessible in working memory during subsequent work on the learning tasks. While working on those tasks, the schemas guide the learners in performing the task. At the same time, learners mindfully abstract away from the concrete experiences offered by the learning tasks and thus reconstruct, modify, or embellish the existing schemas to make them more in agreement with their concrete experiences. This process of *induction* is important for constructing new knowledge and, especially, for adapting existing schemas to make them more appropriate to the given experiences (Holland, Holyoak, Nisbett, & Thagard, 1986).

For our purpose, it is important to note that the presentation of supportive information followed by elaboration and induction does not yield an algorithmic description of how to perform particular task aspects, but rather it yields a general heuristic knowledge that may be helpful for performing particular nonrecurrent aspects of the learning tasks. It does not guarantee that the problem will be solved. Therefore, it makes no sense to couple the presentation of supportive information to one particular learning task. Instead, it should be connected to a task class—a category of equivalent learning tasks that share the same body of underlying knowledge (e.g., schemas such as mental models and cognitive strategies). The just-in-time presentation of supportive information entails presenting this information just before the task class for which it is relevant using instructional methods that ensure that the information is elaborated. This, in fact, reflects a traditional educational perspective: First study for understanding, then practice for application.

This does not preclude the interchange of presentation and practice. The supportive information for each subsequent task class is an addition to or an embellishment of previous information—allowing learners to do things they could not do before. A continuous mix of presentation and practice arises wherein presentations of supportive information for a task class alternate with practice sessions for learning tasks in the same class. Supportive information is, thus, best presented explicitly just before the task class for which it is relevant. Typically, it is kept available for the learners while working on the learning tasks within this task class. For subsequent task classes, only new (additional) supportive information should be presented—extending the set of interacting elements (Pollock et al., 2002). Repeating supportive information from previous task classes may even be harmful. Because it is redundant with what the learners already know, it may increase extraneous cognitive load because learners have to determine that the presented information is actually identical with their prior knowledge (Kalyuga, Chandler, & Sweller, 1998).

Procedural Information

In contrast to supportive information, procedural information pertains to consistent task components or recurrent task aspects that are performed as routines by experts. Experts can perform recurrent task aspects effectively because they have at their disposal more or less automated schemas that associte particular characteristics of the problem situation (i.e., conditions) to particular actions. Experts may even reach a level of performance in which they perform routines automatically, without conscious control or the need to invest mental effort. An experienced researcher searching for relevant literature can, for example, operate a familiar search program without consciously paying attention to it; particular low-level goals automatically yield particular key presses or menu choices. Such automated schemas connecting particular conditions to particular actions are also called *rules* or *productions* (Anderson, 1993, 1996; Newell, 1991). They are highly domain specific and allow only for the same use of the same knowledge for performing recurrent aspects of tasks.

How does one help novice learners automate schemas for recurrent task aspects? The procedural information prescribing the performance of recurrent task aspects, and possibly the facts and concepts prerequisite to correct performance of the procedures, typically exhibits low-intrinsic complexity. Furthermore, automated schemas are constructed as a direct result of practice in a learning process known as *knowledge compilation*, whereby the information that is active in working memory is embedded in highly domain-specific representations, followed by *strengthening*, whereby schemas accumulate strength each time they are successfully applied (Anderson, 1993, 1996). Elaboration of the procedural information beforehand has no added value whatsoever; therefore, it is preferably presented precisely when learners need it. This is what we see when teachers give step-by-step or how-to instructions to their learners during practice, acting as an assistant looking over your shoulder.

CLT not only indicates that procedural information is best presented when learners need it, but it also raises two related design issues. First, presenting procedural information precisely when it is needed to perform particular actions prevents temporal split-attention effects. Information presentation separated in time must be integrated, which results in a higher extraneous cognitive load (Mayer & Sims, 1994). Second, presenting procedural information so that it is fully integrated with the task environment prevents spatial split-attention effects. Such effects may arise when multiple sources of information must be mentally integrated in order to follow procedural instructions and simultaneously manipulate the task environment. Integrating the multiple sources of information by, for instance, using balloon help features or spoken text for procedural information may reduce extraneous cognitive load (Mayer & Moreno, 2002). If temporal or spatial split-attention effects are not fully prevented, the presentation of procedural information before the learning tasks may be more effective than its presentation during practice.

For our current purpose it is also important to note that procedural information presentation and subsequent knowledge compilation yield an algorithmic description of how to perform recurrent aspects of the learning task. Applying the automated schemas warrants that these aspects are successfully performed. Therefore, procedural information is best connected to the first learning task for which the recurrent aspect is relevant. This reflects a perspective that is popular in business training: Practice for application and only study when needed (just-in-time learning; Romiszowski, 1997). For subsequent learning tasks, procedural information is quickly faded as the learners gain more expertise. This principle of fading is consistent with the idea that when learners have enough expertise, procedural information will become redundant and should thus be removed to decrease extraneous cognitive load (Kalyuga et al.,1998).

In summary, we made a distinction between supportive and procedural information. Supportive information may be helpful in performing the nonrecurrent aspects of learning tasks. It is best presented before a class of equivalent learning tasks, and it is critical that the learners elaborate on it so that it can be easily retrieved from long-term memory when necessary for the learning tasks. Procedural information specifies the correct performance of recurrent aspects of learning tasks. It is best presented precisely when learners need it during their work on learning tasks, and it is critical to prevent split-attention effects when presenting this information.

A DESIGN MODEL FOR COMPLEX LEARNING

Over the last decade, researchers (van Merriënboer, 1997; van Merriënboer, Jelsma, & Paas, 1992; van Merriënboer, Clark, & de Croock, 2002) developed an instructional design model for complex learning consistent with CLT. This four-component instructional design model (referred to hereafter as the 4C/ID model) presupposes that well-designed learning environments for complex learning always consist of four components: (a) learning tasks, (b) supportive information, (c) procedural information, and (d) part-task practice. In Figure 1a, learning tasks are represented as circles and organized in an ordered sequence of task classes (dotted lines) that represent simple-to-complex versions of the whole task. These learning tasks will typically be performed by the learners in a simulated or real task environment and, ideally, confront them with all constituent skills that make up the whole complex skill. Furthermore, and as indicated by the shaded areas of the circles, each task class starts with one or more learning tasks with a high level of embedded support (e.g., worked-out examples), continues with learning tasks with a lower level of support (e.g., completion, goal-free, or reverse tasks), and ends with conventional tasks without support. This is repeated for each subsequent task class, yielding a saw-tooth pattern of support throughout the whole training program.

The second and third components are depicted in Figure 1b and pertain to supportive and procedural information, repectively. The supportive information is presented in a just-in-time fashion to work on the nonrecurrent aspects of learning tasks within the same task class (indicated by the light gray, L-shaped figures). The learning tasks within the same task class are equivalent in the sense that they may be performed on the basis of the same general knowledge (i.e., mental models and cognitive strategies that allow one to perform a version of the task with a particular complexity). For each subsequent task class, additional supportive information is presented to enable the learners to perform the more complex version of the whole task. The procedural information, in contrast, is presented just in time to perform the consistent, recurrent aspects of the learning tasks (indicated by the dark gray rectangles with the black vertical arrows). It preferably takes the form of direct, step-by-step, or how-to instruction and is quickly faded away for subsequent learning tasks.

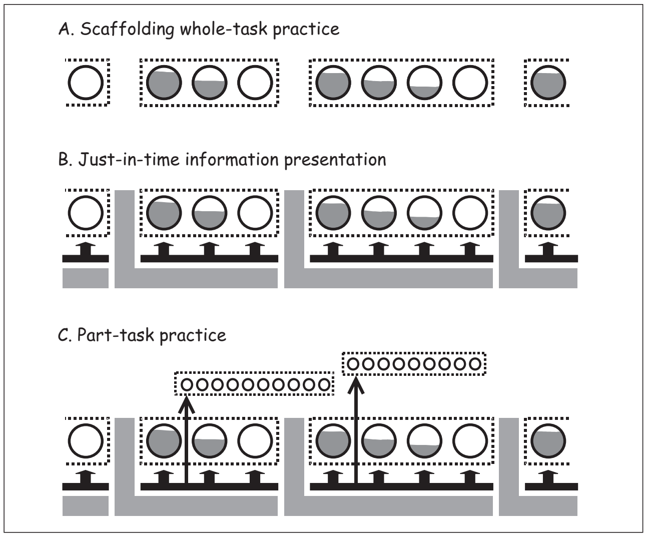
A. Scaffolding whole-task practice
B. Just-in-time information presentation
C. Part-task practice

FIGURE 1 Schematic representation of a training blueprint for complex learning that is fully consistent with CLT.

The fourth component, part-task practice, is indicated in Figure 1c. This component has not yet been substantiated in our theoretical framework but is related to the shift from a part-task to a whole-task paradigm. The whole-task approach that is characteristic of our framework and other recent instructional theories implies that recurrent aspects of performance are not trained separately but only practiced in the context of whole learning tasks. In general, an overreliance on part-task practice is not helpful to complex learning. But if a very high level of automaticity is desired for particular recurrent aspects, the learning tasks may not provide enough practice to reach this level because the responsible learning process, strengthening, requires large amounts of not available repetition. For those aspects, additional part-task practice may be provided—such as children drilling multiplication tables or musicians practicing musical scales. According to the 4C/ID model, additional part-task practice starts only after the learners have been introduced to the recurrent aspects in the context of the learning tasks, so that part-task practice takes place in a fruitful cognitive context that allows learners to identify the activities that are required to integrate the recurrent aspects in the whole task (Carlson, Khoo, & Elliot, 1990).

DISCUSSION

This article started from the observation that recent instructional theories tend to focus on real-life tasks as the driving force for learning. Such tasks are typically associated with a very high- cognitive load, which makes it more important than ever to take the limited human-processing capacity into account. We discussed strategies for scaffolding whole-task practice, including methods to sequence simple-to-complex classes of equivalent tasks (task classes) and methods to follow from worked-out examples to conventional tasks, and strategies for just-in-time information presentation, including methods to link supportive information to task classes and methods to present procedural information precisely when it is needed during task performance. We focused on guidelines and methods that were both deduced from CLT and empirically studied. This resulted in an overview that is still incomplete. For instance, one suggestion from CLT would be to develop learning tasks that not only provide support through manipulating given states, goal states, and solutions (see Table 1) but also through supporting the process needed to reach a solution (i.e., process-oriented instruction). For novice learners, studying the modeling of an expert who is solving a nontrivial problem or solving a problem with performance constraints that require them to satisfactorily complete one problem-solving phase before they may enter the next phase are examples of methods that may decrease extraneous cognitive load and improve learning. Such approaches were not discussed because we are not aware of empirical research relating process support to cognitive load.

Another major limitation of the presented framework is its focus on decreasing intrinsic and extraneous load. It should be perfectly clear that this is not a goal in itself but merely a means to prevent cognitive overload and (equally important) free processing resources that can be devoted to learning processes, such as schema construction and automation. To ensure that freed processing resources are actually devoted to learning, the explicit use of instructional methods that increase germane cognitive load may be required. Some of these methods have been studied in the context of CLT. For example, it is essential that the learning tasks within the same task class show a high variability on all dimensions that also vary in the real world to reach transfer of learning (Paas & van Merriënboer, 1994; Quilici & Mayer, 1996; van Merriënboer, Schuurman, et al., 2002). As another example, asking questions about worked-out examples or completion tasks is important to ensure that learners elaborate them (see Stark, Mandl, Gruber, & Renkl, 2002).

To conclude, a final remark should be made on the relation between CLT and the sketched 4C/ID model. Although CLT has a cognitive process view of learning and the 4C/ID model has a design point of view, both theories are fully consistent with each other. This is not to say that a strong theory of cognitive load alone is sufficient to develop a useful instructional design model for complex learning. For instance, the 4C/ID model relies on several specific learning theories (in particular, Anderson's 1993, 1996, Adaptive Control Theory) to prescribe instructional methods for each of its four components. But limited working memory is no doubt the most central aspect of human cognitive architecture. There are many factors that an instructional designer must consider, but the cognitive load imposed by instructional designs should be the preeminent consideration when determining design structures.

REFERENCES

Anderson, J. R. (1993). *Rules of the mind.* Hillsdale, NJ: Lawrence Erlbaum Associates, Inc.

Anderson, J. R. (1996). ACT: A simple theory of complex cognition. *American Psychologist, 51,* 355–365.

Bannert, M. (2002). Managing cognitive load—Recent trends in cognitive load theory. *Learning and Instruction, 12,* 139–146.

Carlson, R. A., Khoo, H., & Elliot, R. G. (1990). Component practice and exposure to a problem-solving context. *Human Factors, 32,* 267–286.

Carroll, J. M., & Carrithers, C. (1984). Blocking learner error states in a training-wheels system. *Human Factors, 26,* 377–389.

Carroll, J. M., & Rosson, M. B. (1987). The paradox of the active user. In J. M. Carroll (Ed.), *Interfacing thought: Cognitive aspects of human–computer interaction* (pp. 80–111). Cambridge, MA: MIT Press.

Chandler, P., & Sweller, J. (1992). The split-attention effect as a factor in the design of instruction. *British Journal of Educational Psychology, 62,* 233–246.

Collins, A., Brown, J. S., & Newman, S. E. (1989). Cognitive apprenticeship: Teaching the craft of reading, writing, and mathematics. In L. B. Resnick (Ed.), *Knowing, learning, and instruction: Essays in honor of Robert Glaser* (pp. 453–493). Hillsdale, NJ: Lawrence Erlbaum Associates, Inc.

Fisk, A. D., & Gallini, J. K. (1989). Training consistent components of tasks: Developing an instructional system based on automatic-controlled processing principles. *Human Factors, 31,* 453–463.

Gobet, F., Lane, P. C. R., Croker, S., Cheng, P. C. H., Jones, G., Oliver, I., & Pine, J. M. (2001). Chunking mechanisms in human learning. *Trends in Cognitive Science, 5,* 236–243.

Halff, H. M. (1993). Supporting scenario- and simulation-based instruction: Issues from the maintenance domain. In J. M. Spector, M. C. Polson, & D. J. Muraida (Eds.), *Automating instructional design: Concepts and issues* (pp. 231–248). Englewood Cliffs, NJ: Educational Technology Publications.

Holland, J. H., Holyoak, K. J., Nisbett, R. E., & Thagard, P. R. (1986). *Induction: Processes of inference, learning and discovery*. Cambridge, MA: MIT Press.

Jonassen, D. H. (1999). Designing constructivist learning environments. In C. M. Reigeluth (Ed.), *Instructional design theories and models: A new paradigm of instructional theory* (Vol. 2, pp. 371–396). Mahwah, NJ: Lawrence Erlbaum Associates, Inc.

Kalyuga, S., Ayres, P., Chandler, P., & Sweller, J. (2003). The expertise reversal effect. *Educational Psychologist, 38,* XX–XX.

Kalyuga, S., Chandler, P., & Sweller J. (1998). Levels of expertise and instructional design. *Human Factors, 40,* 1–17.

Kester, L., Kirschner, P. A., van Merriënboer, J. J. G., & Baumer, A. (2001). Just-in-time information presentation and the acquisition of complex cognitive skills. *Computers in Human Behavior, 17,* 373–391.

Leutner, D. (2000). Double-fading support: A training approach to complex software systems. *Journal of Computer-Assisted Learning, 16,* 347–357.

Marcus, N., Cooper, M., & Sweller, J. (1996). Understanding instructions. *Journal of Educational Psychology, 88,* 49–63.

Mayer, R. E., & Moreno, R. (2002). Aids to computer-based multimedia learning. *Learning and Instruction, 12,* 107–119.

Mayer, R. E., & Sims, V. K. (1994). For whom is a picture worth a thousand words? Extensions of a dual-coding theory of multimedia learning. *Journal of Educational Psychology, 86,* 389–401.

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

Miller, G. A. (1956). The magical number seven plus or minus two. *Psychological Review, 63,* 81–96.

Naylor, J. C., & Briggs, G. E. (1963). Effects of task complexity and task organization on the relative efficiency of part and whole training methods. *Journal of Experimental Psychology, 65,* 217–224.

Nelson, L. M. (1999). Collaborative problem solving. In C. M. Reigeluth (Ed.), *Instructional design theories and models: A new paradigm of instructional theory* (Vol. 2, pp. 241–267). Mahwah, NJ: Lawrence Erlbaum Associates, Inc.

Newell, A. (1991). *Unified theories of cognition.* Cambridge, MA: Harvard University Press.

Newell, A., & Simon, H. A. (1972). *Human problem solving.* Englewood Cliffs, NJ: Prentice Hall.

Paas, F. G. W. C. (1992). Training strategies for attaining transfer of problem-solving skill in statistics: A cognitive load approach. *Journal of Educational Psychology, 84,* 429–434.

Paas, F. G. W. C., Camp, G., & Rikers, R. (2001). Instructional compensation for age-related cognitive declines: Effects of goal specificity in maze learning. *Journal of Educational Psychology, 93,* 181–186.

Paas, F. G. W. C., & van Merriënboer, J. J. G. (1994). Variability of worked examples and transfer of geometrical problem solving skills: A cognitive load approach. *Journal of Educational Psychology, 86,* 122–133.

Pollock, E., Chandler, P., & Sweller, J. (2002). Assimilating complex information. *Learning and Instruction, 12,* 61–86.

Quilici, J. L., & Mayer, R. E. (1996). The role of examples in how students learn to categorize statistics word problems. *Journal of Educational Psychology, 88,* 144–161.

Reigeluth, C. M. (Ed.). (1983). *Instructional design theories and models*. Hillsdale, NJ: Lawrence Erlbaum Associates, Inc.

Reigeluth, C. M. (Ed.). (1999a). *Instructional design theories and models: A new paradigm of instructional theory* (Vol. 2). Mahwah, NJ: Lawrence Erlbaum Associates, Inc.

Reigeluth, C. M. (1999b). The elaboration theory: Guidance for scope and sequence decisions. In C. M. Reigeluth (Ed.), *Instructional design theories and models: A new paradigm of instructional theory* (Vol. 2, pp. 425–453). Mahwah, NJ: Lawrence Erlbaum Associates, Inc.

Renkl, A., Maier, U. H., Atkinson, R. K., & Staley, R. (2002). From example study to problem solving: Smooth transitions help learning. *Journal of Experimental Education, 70,* 293–315.

Renkl, A., Stark, R., Gruber, H., & Mandl, H. (1998). Learning from worked-out examples: The effects of example variability and elicited self-explanations. *Contemporary Educational Psychology, 23,* 90–108.

Romiszowski, A. (1997). Web-based distance learning and teaching: Revolutionary invention or reaction to necessity? In B. Khan (Ed.), *Web-based instruction* (pp. 25–37). Englewood Cliffs, NJ: Educational Technology Publications.

Rosenshine, B. V., & Meister, C. (1992). The use of scaffolds for teaching less structured cognitive tasks. *Educational Leadership, 49*(7), 26–33.

Scandura, J. M. (1983). Instructional strategies based on the structural learning theory. In C. M. Reigeluth (Ed.), *Instructional design theories and models* (pp. 213–246). Hillsdale, NJ: Lawrence Erlbaum Associates, Inc.

Schank, R. C., Berman, T. R., & MacPerson, K. A. (1999). Learning by doing. In C. M. Reigeluth (Ed.), *Instructional design theories and models: A new paradigm of instructional theory* (Vol. 2, pp. 161–181). Mahwah, NJ: Lawrence Erlbaum Associates, Inc.

Stark, R. (1999). *Lernen mit Lösungsbeisplielen: Einfluss unvollständiger Lösungsbeispiele auf Beispielelaboration, Motivation and Lernerfolg* [Learning by worked-out examples: The impact of completion tasks on example elaboration, motivation and learning outcomes]. Bern, Switzerland: Huber.

Stark, R., Mandl, H., Gruber, H., & Renkl, A. (2002). Conditions and effects of example elaboration. *Learning and Instruction, 12,* 39–60.

Sweller, J. (1988). Cognitive load during problem solving: Effects on learning. *Cognitive Science, 12,* 257–285.

Sweller, J., & Chandler, P. (1994). Why some material is difficult to learn. *Cognition and Instruction, 12,* 185–233.

Sweller, J., van Merriënboer, J. J. G., & Paas, F. G. W. C. (1998). Cognitive architecture and instructional design. *Educational Psychology Review, 10,* 251–296.

Van Gerven, P. W. M., Paas, F. G. W. C., van Merriënboer, J. J. G., & Schmidt, H. G. (2002). Cognitive load theory and aging: Effects of worked examples on training efficiency. *Learning and Instruction, 12,* 87–105.

van Merriënboer, J. J. G. (1990). Strategies for programming instruction in high school: Program completion vs. program generation. *Journal of Educational Computing Research, 6,* 265–287.

van Merriënboer, J. J. G. (1997). *Training complex cognitive skills: A four-component instructional design model for technical training.* Englewood Cliffs, NJ: Educational Technology Publications.

van Merriënboer, J. J. G. (2000). The end of software training? *Journal of Computer Assisted Learning, 16,* 366–375.

van Merriënboer, J. J. G., Clark, R. E., & de Croock, M. B. M. (2002). Blueprints for complex learning: The 4C/ID*-model. *Educational Technology, Research and Development, 50*(2), 39–64.

van Merriënboer, J. J. G., & de Croock, M. B. M. (1992). Strategies for computer-based programming instruction: Program completion vs. program generation. *Journal of Educational Computing Research, 8,* 365–394.

van Merriënboer, J. J. G., Jelsma, O., & Paas, F. G. W. C. (1992). Training for reflective expertise: A four-component instructional design model for training complex cognitive skills. *Educational Technology, Research and Development, 40*(2), 23–43.

van Merriënboer, J. J. G., & Kirschner, P. A. (2001). Three worlds of instructional design: State of the art and future directions. *Instructional Science, 29,* 429–441.

van Merriënboer, J. J. G., & Krammer, H. P. M. (1987). Instructional strategies and tactics for the design of introductory computer programming courses in high school. *Instructional Science, 16,* 251–285.

van Merriënboer, J. J. G., & Paas, F. G. W. C. (1989). Automation and schema acquisition in learning elementary computer programming: Implications for the design of practice. *Computers in Human Behavior, 6,* 273–289.

van Merriënboer, J. J. G., Schuurman, J. G., de Croock, M. B. M., & Paas, F. G. W. C. (2002). Redirecting learners' attention during training: Effects on cognitive load, transfer test performance and training efficiency. *Learning and Instruction, 12,* 11–37.

Vygotsky, L. S. (1987). Thinking and speech. In R. W. Rieber & A. S. Carton (Eds.), *The collected works of L. S. Vygotsky, Vol. 1: Problems of general psychology* (N. Minick, Trans.; pp. 39–285). New York: Plenum. (Original work published 1934)

White, B. Y., & Frederiksen, J. R. (1990). Causal model progressions as a foundation for intelligent learning environments. *Artificial Intelligence, 42,* 99–157.