THE JOURNAL OF THE LEARNING SCIENCES, *13*(3), 337–386

A Scaffolding Design Framework for Software to Support Science Inquiry

Chris Quintana
School of Education
University of Michigan

Brian J. Reiser
School of Education and Social Policy
Northwestern University

Elizabeth A. Davis, Joseph Krajcik, and Eric Fretz
School of Education
University of Michigan

Ravit Golan Duncan, Eleni Kyza, and Daniel Edelson
School of Education and Social Policy
Northwestern University

Elliot Soloway
College of Engineering, School of Education, and School of Information
University of Michigan

The notion of scaffolding learners to help them succeed in solving problems otherwise too difficult for them is an important idea that has extended into the design of scaffolded software tools for learners. However, although there is a growing body of work on scaffolded tools, scaffold design, and the impact of scaffolding, the field has not yet converged on a common theoretical framework that defines rationales and approaches to guide the design of scaffolded tools. In this article, we present a scaffolding design framework addressing scaffolded software tools for science inquiry. De-

Correspondence and requests for reprints should be sent to Chris Quintana, School of Education, University of Michigan, 610 East University Avenue, 1360D SEB, Ann Arbor, MI 48109. E-mail: quintana@umich.edu

veloped through iterative cycles of inductive and theory-based analysis, the framework synthesizes the work of prior design efforts, theoretical arguments, and empirical work in a set of guidelines that are organized around science inquiry practices and the challenges learners face in those practices. The framework can provide a basis for developing a theory of pedagogical support and a mechanism to describe successful scaffolding approaches. It can also guide design, not in a prescriptive manner but by providing designers with heuristics and examples of possible ways to address the challenges learners face.

Recent educational approaches emphasize more ambitious environments for learning in which learners engage in extended inquiry to develop knowledge and skills in the context of investigating meaningful problems (Blumenfeld, Fishman, Krajcik, Marx, & Soloway, 2000; Bransford, Brown, & Cocking, 2000; Linn, 2000). These learning contexts consist of more authentic, challenging, and open-ended problems and thus require significant disciplinary knowledge and metacognitive skills. Consequently, the need for pedagogical support tailored to the demands of these more ambitious learning tasks has been an emerging focus of design and empirical research efforts. An important construct used in design and theory is the notion of scaffolding learners. *Scaffolding* has been traditionally defined as the process by which a teacher or more knowledgeable peer provides assistance that enables learners to succeed in problems that would otherwise be too difficult (Wood, Bruner, & Ross, 1976; see also Palincsar, 1998; Stone, 1998). For example, a teacher may provide strategic guidance, help learners set appropriate goals, or perform difficult parts of a task.

This notion of scaffolding can productively frame empirical research, theory building, and design. The idea of scaffolding has been adopted in research on technological supports for learning, which have become increasingly important in pedagogical designs. Designers have argued that software tools can support learners by providing needed structure for difficult tasks (Bell & Davis, 2000; Collins & Brown, 1988; Guzdial, 1994; Jackson, Stratford, Krajcik, & Soloway, 1994; Toth, Suthers, & Lesgold, 2002). For example, Guzdial (1994) outlined one of the first descriptions of "software-realized scaffolding" by suggesting three roles software could play to provide scaffolding: communicating processes to learners, coaching learners with hints and reminders about their work, and eliciting articulation from learners to encourage reflection. Soloway, Guzdial, and Hay (1994) introduced the idea of a "learner-centered design" approach to consider software-realized scaffolding with respect to the needs of learners. Soloway et al. argued that software designers need to consider scaffolding for the tasks learners perform and for the tools and interfaces learners use. Although support provided through technology differs from that provided by human teachers or peers, the common idea is that the task is modified (in this case by a tool with particular characteristics) in ways that make it more tractable and productive for learners.

THE NEED FOR A SCAFFOLDING DESIGN FRAMEWORK

Researchers have explored the design of software tools to scaffold learners across a range of domains, and this work has yielded rich examples of design approaches and research studies. There is a growing body of work detailing scaffolded tools, scaffolding design strategies, and the impact of scaffolded tools on learners. Researchers have developed theoretical frameworks to guide their designs such as Linn's scaffolded knowledge integration framework, which stresses integrating scientific understanding with prior commonsense knowledge (Linn, Davis, & Eylon, 2004; Linn & Hsi, 2000), and Scardamalia and Bereiter's (1991) intentional learning framework, which encourages learners to articulate their understandings through structured discourse.

Despite these individual successes, accumulation of both theory and craft knowledge about scaffolding design has been difficult. The field has not converged on a common framework that defines a system of theoretical rationales and design principles to guide the design and empirical investigation of scaffolded tools. Typically, each research group proposes a theoretical motivation and a particular set of design principles underlying their scaffolding design, making it difficult to synthesize claims and results across contexts. Furthermore, design arguments for scaffolding range across different levels of specificity from, for example, arguments about the utility of graphical representations in interfaces to more general claims for the importance of externalizing planning processes.

We argue that advances in the field require an empirically grounded consensus about successful scaffolding methods. This requires a common theoretical framework to define and evaluate scaffolding approaches for software tools. Researchers need a common theoretical vocabulary that allows them to characterize and test the generality of claims about scaffolding, for example, to determine whether two different tools implement the same theoretical scaffolding claim. Principles must go beyond specific software implementations or a history of tools from a particular research project and instead synthesize the craft and theoretical understanding of the field to guide new developments in software, empirical research, and theory.

Our goal is to explore the characteristics of such a framework for software scaffolding. We present a scaffolding design framework that synthesizes the work of prior design efforts, theoretical arguments, and empirical work to develop a systematic set of guidelines and strategies grounded on what we understand about learning. We build on current proposals of general principles such as the scaffolded knowledge integration framework (Linn, Davis, et al., 2004; Linn & Hsi, 2000), principles of learner-centered design (Quintana, Soloway, & Krajcik, 2003), problem-based inquiry (Kolodner et al., 2003), and others.[1]

THEORETICAL FOUNDATIONS

The warrants for elements in a design framework need to be more than recognition of commonalities across different design approaches. Design guidelines must be defensible in terms of what researchers understand about learning and instruction. In the scaffolding design framework presented here, we ground the guidelines on complementary theoretical analyses of the nature of learning, the obstacles learners face, and the nature of pedagogical support.

First, we consider cognitive apprenticeship. The instructional situations we are targeting fall loosely under this approach in which students become increasingly accomplished problem-solvers given guidance from mentors through coaching, task structuring, and hints (Bruner, 1996; Collins, Brown, & Newman, 1989b). Cognitive apprenticeship provides a model of how performance of complex tasks can be distributed, with others helping to minimize obstacles and compensate for limitations by providing assistance at opportune moments. Second, we consider cognitive models of learning by doing (e.g., Anderson, 1983; VanLehn, 1989) to explore the nature of expertise in a discipline and the difficulties learners face in working on rich open-ended problems. Third, we consider the perspectives of social constructivism and situated cognition, which provide an account of socially situated tasks and describe how learning a discipline involves social interaction and discourse dimensions (e.g., Lave & Wenger, 1991; Vygotsky, 1978).

Scaffolding can help learners accomplish tasks within their zone of proximal development (Vygotsky, 1978) by providing the assistance learners need to accomplish tasks more complex than they could do alone in a way such that they can still learn from that experience. Scaffolding in software tools can be characterized in terms of the differences the scaffolding creates in comparison to some presumably more difficult reference version of the task (Sherin, Reiser, & Edelson, this issue)—it can, in fact, transform tasks for learners. We build on earlier characterizations of software that incorporates scaffolding (e.g., Guzdial, 1994; Soloway et al., 1994) but emphasizes the transformative nature of scaffolding rather than a more feature-oriented perspective. In our characterization, the scaffolding may not be separable from the tool itself. Furthermore, by acknowledging the interplay among teachers, students, software, curriculum, and other elements of a classroom (Salomon, 1996), we intend to promote the notion that although scaffolding may

---

[1]The goals of this effort are complementary with those of the Center for Innovative Learning Technologies (2002) *Design Principles for Educational Software* project. That project is focused on developing a community resource, an infrastructure to pull together design ideas from learning environment researchers. It is developing a networked database to which researchers can contribute their design rationales, principles, and examples. Hence, the approach takes an inclusive approach rather than focusing on synthesis. In contrast, our aim is to distill the various proposals from the literature, find equivalences, ground the design guidelines theoretically, and develop a consistent level for the articulation of principles.

be provided in part through software, learners' use of it is mediated by these many other elements. Thus, instead of using the term *software-realized scaffolding,* we refer to scaffolding (or "scaffolding approaches") within "software tools" (or "scaffolded tools"); we intend to de-emphasize the notion that scaffolding is a layer of supportive features that lies on top of software and that acts on learners directly and straightforwardly. Our analyses focus on how software tools in particular can provide scaffolding by transforming tasks in ways that lead to greater success and opportunities to learn.

DEVELOPING THE SCAFFOLDING DESIGN FRAMEWORK

In exploring how software can scaffold learners, we have developed our scaffolding design framework within the domain of science inquiry learning. Consistent with general notions in the literature, we define *inquiry* as the process of posing questions and investigating them with empirical data, either through direct manipulation of variables via experiments or by constructing comparisons using existing data sets. We focus on this domain because it provides a rich corpus of literature on the nature of the learning and obstacles in the discipline. Furthermore, it is representative of ambitious learning, as the problems of managing investigations, monitoring progress, testing hypotheses, and constructing explanations are general to many disciplines.

Given this domain, we organize our scaffolding design framework around three constituent processes for inquiry synthesized from current descriptions of scientific reasoning and related to other models of scaffolding (e.g., Hannafin, Land, & Oliver, 1999): *sense making,* which involves the basic operations of testing hypotheses and interpreting data; *process management,* which involves the strategic decisions involved in controlling the inquiry process; and *articulation and reflection,* which is the process of constructing, evaluating, and articulating what has been learned. These processes entail tasks that are cognitively complex and are often implemented in social activity such as discussion, negotiation, and consensus building. We expand on each of these processes in motivating each component of the scaffolding design framework.

Method

Our goal is a theoretical framework that can be investigated empirically to identify effective approaches for scaffolding learners. Identifying features from different systems that represent a similar design idea with different interface implementations and disciplinary content yields useful candidates for analysis. However, a theoretical analysis has to be grounded on the way the tool transforms tasks for learners; therefore, a framework requires more than an inductive analysis, and its principles cannot be specific to interface features (such as prompts or process maps). Thus, we employ a theory-driven approach, working also with observations from an inductive analysis, to develop a system of scaffolding guidelines and examples. In this theory-driven, principled analysis, we analyze why particular effects of scaffolding are needed and evaluate candidate exemplars with respect to their relevance to the need.

A principled analysis of the way tools can assist learners must begin with a description of the tasks being supported (here, science inquiry tasks). Furthermore, the principled analysis requires identifying the obstacles learners face in performing given tasks to focus the scaffolding design (Quintana, Krajcik, & Soloway, 2001; Reiser, this issue). Although many of the obstacles and design solutions we identify seem candidates for more general scaffolding guidelines, our claims in this article are limited to supporting scientific inquiry.[2]

Our analysis attempts to synthesize current thinking in the field about the nature of inquiry practices. The analysis draws on characterizations of inquiry learning in science (Krajcik, Berger, & Czerniak, 2002; Minstrell & Van Zee, 2000; Reiser et al., 2001; White & Frederiksen, 1998), characterizations of the more general nature of scientific practice (Latour, 1990; Lemke, 1990), and core principles of learning environments (Bransford et al., 2000). The analysis also draws on existing design frameworks for instruction such as descriptions of project-based science (Blumenfeld et al., 1991; Kolodner et al., 2003; Polman, 2000; Ruopp, Gal, Drayton, & Pfister, 1993), "learning-for-use" (Edelson, 2001), intentional learning (Scardamalia & Bereiter, 1992), scaffolded knowledge integration (Linn, Davis, et al., 2004; Linn & Hsi, 2000), and communities of learners (Bielaczyc & Collins, 1999; Brown & Campione, 1994). Furthermore, we reviewed empirical studies of the specific obstacles learners face in scientific inquiry framed around our analysis of science inquiry itself and used the obstacles to help us identify relevant scaffolding guidelines.

This theory-driven approach includes three main phases:

1. Characterizing the cognitive tasks, social interactions, tools, and artifacts that constitute the scientific practices in which learners are engaged.
2. Characterizing the aspects of these practices in which learners encounter obstacles.
3. Characterizing scaffolding guidelines that specify ways that tools can alter the task to address the obstacles by helping make tasks more tractable and productive for learners.

---

[2]Note that although our focus is on science inquiry, we did consider nonscience software in our analyses if the software presented a key scaffolding idea we wanted to explain during our inductive analysis or a good example of a solution to a problem we identified in our theory-driven analysis.

The result of this process was an initial framework we used as a structure to encode patterns of scaffolding approaches that emerged from an inductive analysis of current software tools. We then refined the framework through repeated cycles of theory-driven and inductive analyses.

Our inductive process involved reviewing examples of scaffolding approaches. We initially considered examples drawn from tools that were part of our own learning environment research.[3] We evaluated these examples in terms of their connections to the obstacles we identified in our theory-driven analysis. As we continued our analyses, we broadened the set of tools to include examples from a range of learning environments in the literature. As we identified multiple examples that were candidates of a common approach, we developed a description of the strategy the examples represented and encoded that strategy as one way to implement one of the general scaffolding guidelines.

For example, in our analysis, we found that one obstacle learners face involves difficulty in keeping track of plans and monitoring progress (Klahr, 2000; Schauble, Glaser, Raghavan, & Reiner, 1991). In our review of software tools, we identified several systems with features designed to support learners by providing ways to help them keep track of where they were in an overall plan. These included the inquiry maps of the Web-Based Inquiry Science Environment (WISE; Linn & Slotta, 2000) and the process maps of Symphony (Quintana, Eng, Carra, Wu, & Soloway, 1999). We characterized these as a scaffolding strategy of providing task decompositions and encoded that strategy within a more general guideline (identified in the theory-driven analysis) that highlighted the need to provide learners with structure for complex tasks.

SCAFFOLDING DESIGN FRAMEWORK

The combination of the theory-driven approach and the inductive process in the methodology we just described led us to a scaffolding design framework that includes several elements. The theory-driven approach helped us identify three elements. First, the *task model* includes the constituents of inquiry identified from the literature—three interactive processes of sense making, process management, and reflection and articulation. The second element includes *obstacles* learners face in each constituent. We identified obstacles based on predictions from theories of learning as well as results of empirical studies of learning through inquiry. The third element of the framework is the set of scaffolding guidelines. A *scaffolding guideline* specifies a way in which tools modify the task to help learners overcome

[3]Our initial set of tools for analysis included Animal Landlord, Artemis™, eChem, ExplanationConstructor, Galápagos Finches, Knowledge Integration Environment (KIE), Model-It™, Progress Portfolio, Symphony, and WorldWatcher.

obstacles (e.g., "provide structure for complex tasks and functionality"; see Guideline 4). The inductive process also helped us refine the evolving framework and identify two more elements. We identified *scaffolding strategies,* or specific types of implementation approaches that can achieve a given guideline (e.g., "describe complex tasks by using ordered and unordered task decompositions"; see scaffolding Strategy 4b). The final element of the framework is *examples* of features within software tools that exemplify each scaffolding strategy. Table 1 summarizes the framework's scaffolding guidelines and scaffolding strategies we identified.

SCAFFOLDING SENSE MAKING

We begin our description of the framework by considering how to scaffold sense making. First, we present a brief characterization of sense making in the literature and then describe the sense-making challenges learners face. We then present three guidelines for supporting sense making and discuss strategies and examples for each guideline.

Nature of Sense Making

Sense making refers to the basic operations of science inquiry such as generating hypotheses, designing comparisons, collecting observations, analyzing data, and constructing interpretations. Sense-making operations must connect reasoning about a phenomenon to a process for testing a conjecture and from the empirical data generated in that testing back to the implications for the phenomenon (Klahr & Dunbar, 1988). Some sense-making operations are general to experimental scientific investigation such as "vary one thing at a time" (Klahr, 2000), whereas others are specific to a particular discipline (Knorr-Cetina, 1996; Reiser et al., 2001). For example, learners need to translate conjectures about a phenomenon into a hypothesis that can be operationalized into a systematic comparison (e.g., testing whether increased carbon emissions are associated with increases in temperature). They need to see how different areas of science manipulate data representations in different ways to look for patterns (e.g., comparing two histograms to look for a difference in populations). They need to identify relevant variables from the description of a situation (e.g., identifying objects and forces in a physics problem). Finally, learners need to interpret data in light of predictions to reason about the implications of the data for a hypothesis (e.g., comparing predictions about the effect of temperature on a chemical reaction to observations of that reaction to develop a new hypothesis).

Sense making involves formal scientific representations, which are very powerful tools in that they encode much shared expertise (Latour, 1990). Learners must learn how to represent what is known or understood about a situation by making di-

TABLE 1
Summary of the Scaffolding Design Framework

Scaffolding Guidelines	*Scaffolding Strategies*
Science inquiry component: Sense making	
Guideline 1: Use representations and language that bridge learners' understanding	1a: Provide visual conceptual organizers to give access to functionality
	1b: Use descriptions of complex concepts that build on learners' intuitive ideas
	1c: Embed expert guidance to help learners use and apply science content
Guideline 2: Organize tools and artifacts around the semantics of the discipline	2a: Make disciplinary strategies explicit in learners' interactions with the tool
	2b: Make disciplinary strategies explicit in the artifacts learners create
Guideline 3: Use representations that learners can inspect in different ways to reveal important properties of underlying data	3a: Provide representations that can be inspected to reveal underlying properties of data
	3b: Enable learners to inspect multiple views of the same object or data
	3c: Give learners "malleable representations" that allow them to directly manipulate representations
Science inquiry component: Process management	
Guideline 4: Provide structure for complex tasks and functionality	4a: Restrict a complex task by setting useful boundaries for learners
	4b: Describe complex tasks by using ordered and unordered task decompositions
	4c: Constrain the space of activities by using functional modes
Guideline 5: Embed expert guidance about scientific practices	5a: Embed expert guidance to clarify characteristics of scientific practices
	5b: Embed expert guidance to indicate the rationales for scientific practices
Guideline 6: Automatically handle nonsalient, routine tasks	6a: Automate nonsalient portions of tasks to reduce cognitive demands
	6b: Facilitate the organization of work products
	6c: Facilitate navigation among tools and activities
Science inquiry component: Articulation and reflection	
Guideline 7: Facilitate ongoing articulation and reflection during the investigation	7a: Provide reminders and guidance to facilitate productive planning
	7b: Provide reminders and guidance to facilitate productive monitoring
	7c: Provide reminders and guidance to facilitate articulation during sense-making
	7d: Highlight epistemic features of scientific practices and products

agrams and symbolic notations (e.g., chemical structures, equations). As learners reason about a situation, they must modify representations to encode new inferences (e.g., drawing vectors, attaching numerical values). Learners can further progress toward constructing empirical tests, which can require encoding numerical data (e.g., by constructing tables or graphs). Scientific reasoning requires coordinating these representations (Greeno, 1989; Lehrer & Schauble, 2002) and fluidly mapping between them to develop empirical tests and draw inferences.

Obstacles Learners Face in Sense Making

In the hands of experts, formal representations support scientific work, enabling the detection of patterns and the testing of ideas. Yet what is easy for experienced practitioners can be overwhelming for learners. Experts can see meaningful patterns in problem-solving situations that may not be apparent to novices (Chase & Simon, 1974; Chi, Feltovich, & Glaser, 1981; Larkin, McDermott, Simon, & Simon, 1980; VanLehn, 1989). Consequently, learners are more likely to be distracted by similarities that are only superficial.

Therefore, when considering sense making, several related challenges for learners emerge. First, there is a gap between the way learners intuitively think about a phenomenon and the formalisms used to represent it in expert practice (Reif & Larkin, 1991; Sherin, 2001). Thus, learners need support to map from their understandings to disciplinary formalisms (Edelson, Gordin, & Pea, 1999). Second, noticing what is important about scientific situations requires substantial conceptual domain-specific knowledge, which learners may lack. Third, the expert strategies needed to guide sense making may not be made explicit in traditional instruction (Collins & Brown, 1988; Collins, Brown, & Newman, 1989a; Merrill, Reiser, Ranney, & Trafton, 1992), so learners need support in acquiring them.

We have identified three scaffolding guidelines that help learners overcome these obstacles. They describe how tools can help learners connect their intuitions and situational understanding with manipulation of scientific formalisms.

Scaffolding Guideline 1: Use Representations and Language That Bridge Learners' Understanding

Developing expertise in a discipline requires building the domain-specific knowledge needed to guide the sense-making process and to work with disciplinary formalisms. Learning requires continually accessing and building on prior knowledge, so it is critical that new expert practices are connected with learners' prior conceptions and with their ways of thinking about ideas in the discipline (e.g., Clement, 1993).

Tools can support learners by using representations that connect with learners' intuitions and also map onto expert practice. The representations employed in a tool can shape how people conceive a task (Norman, 1991). In this way, the tool's structure provides this type of bridging scaffold, helping learners make the connection between their own ways of thinking about problems and the concepts and formalisms used in more expert practice.

We identified three strategies for implementing this guideline:

1a. Provide visual conceptual organizers to give access to functionality.
1b. Use descriptions of complex concepts that build on learners' intuitive ideas.
1c. Embed expert guidance to help learners use and apply science content.

Providing visual conceptual organizers to give access to functionality is one way of using representations that bridge learners' understanding. Scaffolding Strategy 1a states that software should help learners access and interact with the software functionality in a way that allows them to think about the deeper concepts and structure of disciplinary relations and not get caught up in surface details. Examples here use various types of visual representations to organize access to functionality, such as conceptual diagrams or representations of visual scenes (see Table 2). For example, WorldWatcher (Edelson et al., 1999), a scientific visualization system, allows learners to pursue investigations and explore data in atmospheric sciences. The program uses an energy balance diagram to help students understand what factors are relevant to investigate and reason about what data to consider next in light of this conceptual organization (Figure 1).

In addition, different disciplinary representations and language can be used to anchor constructs and terminology in learners' prior understanding and everyday experiences as recommended by scaffolding Strategy 1b. The general notion of grounding learner understanding by helping learners access familiar ideas on which more formal concepts can be built is a widely used strategy in instruction (Linn, Bell, & Davis, 2004; Strike & Posner, 1985). Doing so can support learners in understanding disciplinary representations and terminology by helping them reframe their intuitive ideas in terms of expert practice (C. Smith & Unger, 1997). For example, Model-It™,[4] a modeling tool that enables learners to represent and explore causal networks that encode relations between variables (Metcalf, Krajcik, & Soloway, 2000; Stratford, Krajcik, & Soloway, 1998), replaces quantitative expressions with more intuitive qualitative language when students are building relationships between variables in their model (see Figure 2). Other examples shown in Table 2 employ language or visual representations to help learners build on their informal prior understandings.

---

[4]Model-It™ is available from the GoKnow Web site: http://www.goknow.com

TABLE 2
Software Examples of Guideline 1: Use Representations and Language That Bridge Learners' Understanding

Software	*Description*
Scaffolding Strategy 1a: Provide visual conceptual organizers to give access to functionality	
WorldWatcher	WorldWatcher (Edelson, Gordin, & Pea, 1999) uses an energy balance diagram to help students understand what factors are relevant to investigate and reason about what data to consider next
Astronomy Village	Astronomy Village, a software environment for space science, uses visual scenes from laboratories to organize access to data from satellites and solar system probes (Dimitrov, McGee, & Howard, 2002)
Scaffolding Strategy 1b: Use descriptions of complex concepts that build on learners' intuitive ideas	
Model-It™	Model-It™ (Metcalf, Krajcik, & Soloway, 2000; Stratford, Krajcik, & Soloway, 1998) replaces quantitative expressions with qualitative language when students are building relations between variables in a model
ThinkerTools	ThinkerTools conveys the notion of acceleration to students by having moving objects in a simulation leave a visual trace of equally timed marks (White, 1984)
BioKIDS CyberTracker	The CyberTracker software uses "taxonomic common sense" to allow students to categorize animals with accurate, but understandable, intuitive classification schemes and language rather than traditional biological classification schemes (Parr, Jones, & Songer, 2002)
Scaffolding Strategy 1c: Embed expert guidance to help learners use and apply science content	
KIE and WISE	The "Mildred" guide in KIE gives students content hints in the form of questions to think about or thought experiments to do; the WISE learning environment (Linn & Slotta, 2000) provides similar hints
Knowledge Mediator Framework	The Knowledge Mediator Framework (Jacobson, Sugimoto, & Archodidou, 1996) presents annotated examples that include "expert commentaries" explaining how a scientific construct is applied in the example
Why2–Atlas	The Why2–Atlas system for teaching qualitative physics features a coach that tries to identify and address different student misconceptions about physics by engaging in a dialog with the student, essentially modeling to the student how an expert might reason about different physics concepts (VanLehn et al., 2002)

Note. KIE = Knowledge Integration Environment; WISE = Web-Based Inquiry Science Environment.

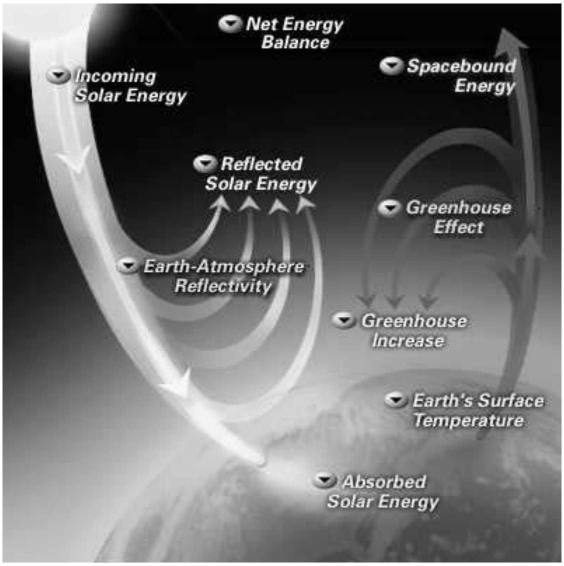

FIGURE 1 The energy balance diagram in WorldWatcher provides a conceptual organizer of the constituent factors under investigation.

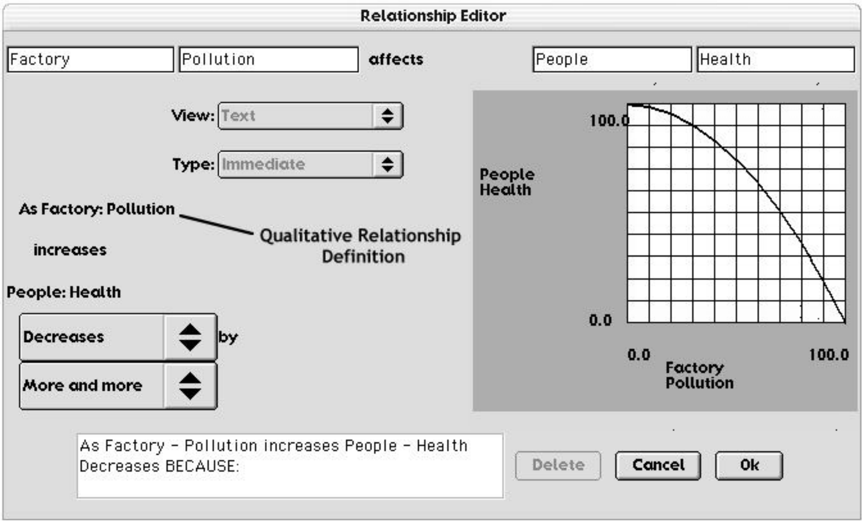

FIGURE 2 The relationship editor in Model-It enables students to specify relations between variables in more intuitive terms.

Furthermore, as noted previously, learners lack the background knowledge that experts can apply to a science investigation. Therefore, another scaffolding strategy (Strategy 1c) that helps bridge between intuitive and expert disciplinary thinking is to embed expert guidance to help learners make connections to phenomena they already understand while pushing them to interpret new phenomena. Hints providing expert guidance have been a core approach in intelligent tutoring systems, which are designed to provide those hints in appropriate problem-solving contexts (Anderson, Corbett, Koedinger, & Pelletier, 1995). For example, a hint might suggest which part of a diagram might be most productive to focus on during work on a geometry or physics problem. Another approach for providing expert guidance is to embed examples that show how experts think about the scientific constructs in a problem. Table 2 provides some examples.

We illustrate this guideline more fully by expanding on an example specific to scaffolding Strategy 1c about embedding expert guidance. Students use KIE to complete complex science projects focused on using scientific evidence and claims to develop critiques, arguments, and designs (Bell & Davis, 2000). Because it is difficult for learners to critique, debate, and design the way experts would, KIE supports students with content hints in the "Mildred" guide (Figure 3). The guide presents questions and thought experiments that students can consider to help

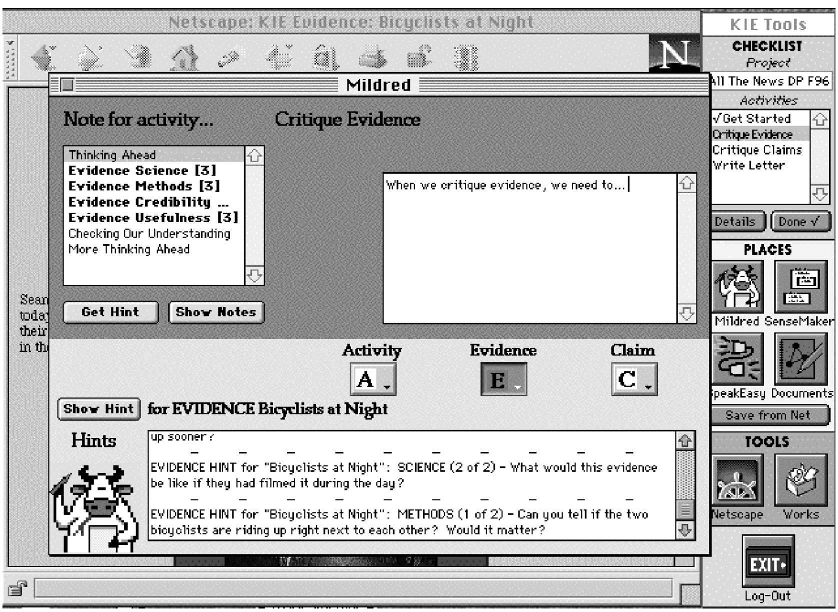

FIGURE 3 The Knowledge Integration Environment (KIE) "Mildred" guide contains different types of hints to help learners understand content. Mildred also contains "Thinking Ahead" prompts to foster productive planning.

them apply scientific constructs to their investigation. For example, a hint could ask students to consider what might happen if an experiment about energy conversion were conducted during the day rather than at night. These hints are modeled on questions an experienced teacher would ask of students working on the project. The WISE learning environment (Linn & Slotta, 2000) provides similar hints to help move learners toward more sophisticated thinking about science content.

In KIE and the other examples illustrating this guideline, we see how tools help compensate for the "bootstrapping problem" of needing rich conceptual knowledge to guide learning. Tools can transform tasks to help compensate for the limited conceptual knowledge learners possess while at the same time helping them build more formal knowledge for future use. Such scaffolding provides opportunities for learners to successfully solve problems, an important aspect of learning by doing (e.g., Anderson, 1983). At the same time, from a social constructivist perspective, such scaffolding is useful because it supports learners in guided practice needed to appropriate these disciplinary practices.

Scaffolding Guideline 2: Organize Tools and Artifacts Around the Semantics of the Discipline

The previous guideline discussed strategies for linking new practices to learners' prior understanding. Here we discuss a complementary guideline addressing the obstacles arising from the need for learners to acquire discipline-specific ways of approaching problems. Because expert practice relies on specific background knowledge that learners lack, learners need support to implement general notions of science inquiry in specific disciplinary contexts (Reiser et al., 2001; Schauble, Glaser, et al., 1991).

Guidelines 1 and 2 both exploit the role of tools in helping shape learners' conceptions of tasks. However, where Guideline 1 refers to using representations that can be productively understood from the learners' perspective, Guideline 2 focuses on the other side of the gap, helping bring disciplinary ways of thinking closer to learners by making such thinking more visible in tool interactions. Such support helps learners overcome limitations in their disciplinary knowledge by making disciplinary semantics and strategies more explicit in the tools they use and the artifacts they construct.

We identified two strategies for implementing this guideline:

2a. Make disciplinary strategies explicit in learners' interactions with the tool.
2b. Make disciplinary strategies explicit in the artifacts learners create.

Making disciplinary strategies explicit in tools (Strategy 2a) can help learners think about the steps they need to take in their work, building on the more general idea of making thinking visible (Collins, 1996; Collins & Brown, 1988; Linn, Davis, et al., 2004). This type of guidance goes beyond decomposing the task into its constituents (which we discuss in Guideline 4). Rather, in this strategy, designers structure the software interface to make explicit the various disciplinary strategies learners may use.

For example, Galápagos Finches (Reiser et al., 2001; Tabak, 1999; Tabak, Smith, Sandoval, & Agganis, 1996) uses a conceptually organized data query allowing access to relevant data choices as students construct queries about ecosystem populations (Figure 4). The data query requires students to communicate their desired query in terms of disciplinary strategies such as making longitudinal (i.e., “seasons”) versus cross-sectional (i.e., “subgroups”) comparisons. Students need to consider their goal in terms of these types of comparisons, so the salient expert strategies students need to appropriate are made visible. Similarly, students must identify what they want to compare in terms of strategic distinctions such as, for example, examining how a variable is distributed or how a variable is related to another variable. This forces them to focus on the type of comparison they wish to construct and not on the superficial aspects of the comparison.

Tools may also provide support by structuring the artifacts learners create to encode their thinking in ways that highlight important disciplinary ideas or distinctions (Strategy 2b). For example, the Animal Landlord (B. K. Smith & Reiser, 1998) provides a structured environment in which students can record their analyses of animal behavior, encouraging students to distinguish between their observations and their interpretations, a critical distinction in such analyses (see Figure 5).

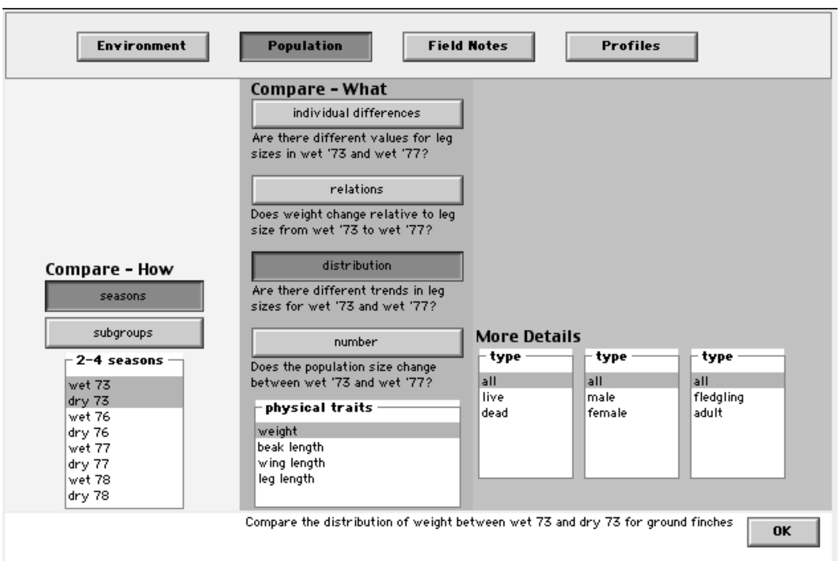

FIGURE 4 Galápagos Finches allows learners to construct a query by making strategic decisions about the comparison.

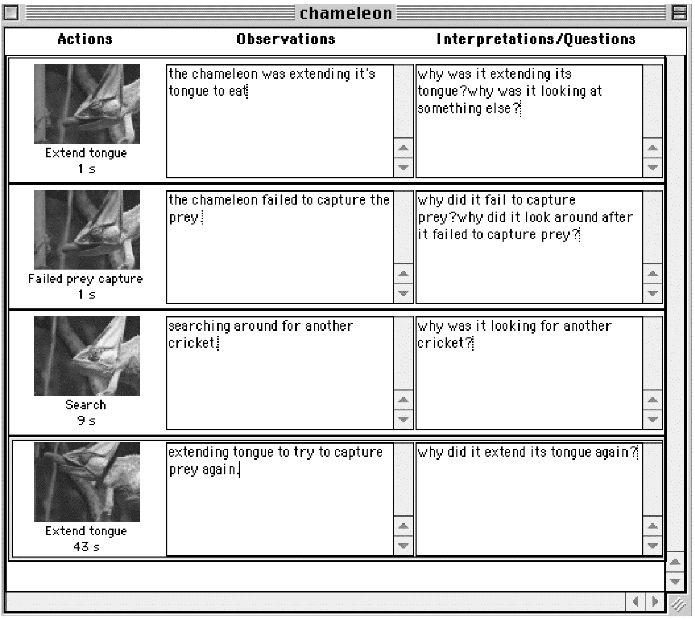

FIGURE 5 The Animal Landlord helps learners record their analyses of animal behavior into an artifact that encodes observations and interpretations for each step of a behavior, creating an annotated storyboard.

Table 3 describes other software examples that make disciplinary strategies explicit through tool options and the artifacts students create.

Reflecting disciplinary strategies in tools and artifacts should help learners engage in sense-making practices in several ways. First, this support can help learners build the strategic knowledge needed to participate in a scientific practice by focusing learners' attention on making strategic decisions they might otherwise avoid (Reiser, this issue). Second, this approach focuses learners on the language of the scientific practice, assisting them in thinking about and talking about their ideas using the discourse norms of the discipline. Learners can engage in practices that are more tractable for them than the expert practice but that share some key elements with the expert practice, allowing them to act as legitimate peripheral participants in the practice (Lave & Wenger, 1991).

Scaffolding Guideline 3: Use Representations That Learners Can Inspect in Different Ways to Reveal Important Properties of Underlying Data

Guideline 3 continues our focus on limitations in learners' conceptual knowledge about the discipline. Here we discuss ways to address obstacles learners face in deal-

TABLE 3
Software Examples of Guideline 2: Organize Tools and Artifacts Around the Semantics of the Discipline

Software	*Description*
Scaffolding Strategy 2a: Make disciplinary strategies explicit in learners' interactions with the tool	
Galápagos Finches	Galápagos Finches (Reiser et al., 2001; Tabak, 1999; Tabak, Smith, Sandoval, & Agganis, 1996) uses a conceptually organized data query allowing access to relevant data choices as students construct queries about populations in an ecosystem
Scaffolding Strategy 2b: Make disciplinary strategies explicit in the artifacts learners create	
Animal Landlord	The Animal Landlord environment lets students analyze animal behavior using digital video tools and a storyboard representation showing student-selected video along with an associated description for each frame (B. K. Smith & Reiser, 1998); this representation makes behavioral analysis explicit to students, requiring them to find and decompose complex behaviors into constituents, categorize the constituents, and use them to convey a story
Geometry Tutor	The Geometry Tutor provides facilities for students to construct a Proof Graph; the notation in the artifact makes explicit that proofs can and should combine forward and backward reasoning and consist of multiple paths from givens to goal (Anderson, Boyle, & Yost, 1986)

ing with the representations of a phenomenon they need to understand and manipulate when making sense of that phenomenon. Access to scientific phenomena is typically mediated through the creation and understanding of representations such as tables, graphs, equations, and diagrams. However, these representations impose additional challenges for learners. Guideline 3 addresses these challenges by recommending inspectable representations to simplify the process of mapping between representations and the aspects of phenomena they encode and help learners manipulate and explore representations in different ways.

We identified three scaffolding strategies for implementing this guideline:

3a. Provide representations that can be inspected to reveal underlying properties of data.
3b. Enable learners to inspect multiple views of the same object or data.

3c. Give learners "malleable representations" that allow them to directly manipulate representations.

Scaffolding Strategy 3a can be considered to be the base level of this guideline recommending that learners be able to inspect representations to see important characteristics of data or a phenomenon such as patterns in data or values of different variables. Table 4 describes some examples of software that provide inspectable representations for learners. Note that in some systems, this strategy overlaps with the earlier scaffolding Strategy 1b of using descriptions that connect with learners' intuitions. The earlier strategy suggests the type of description or visual representation to use. In this strategy, we add the additional feature that the representation can be inspected to help learners see patterns about the inner workings of the phenomenon. For example, the Density Learning Environment uses an intuitive graphical "dots per box" representation in simulations to convey a model of density as how tightly packed the dot "masses" are in each space so learners can inspect objects to reveal mass, volume, and density relations (Snir, Smith, & Grosslight, 1995).

Representations can go even further when the task requires it. Scientific reasoning involves accessing different representations that highlight different aspects of or ways of reasoning about a phenomenon. Scaffolding Strategy 3b involves providing learners with multiple views of the same object or data, giving them a means to coordinate different kinds of intuitive and formal representations. For example, one may need to graph the same data in different ways or move fluidly between aggregate displays and individual data records. Visualizations can be provided side by side to support simultaneous inspection (Linn, Bell, et al., 2004). Table 4 describes examples in which software gives learners multiple inspectable views of the same object.

Scaffolding Strategy 3c also builds on the strategy of providing inspectable representations, but rather than simply providing multiple representations to inspect, this strategy considers malleable representations. Here, learners can directly manipulate one aspect of a representation (such as a value in a simulation input equation) and get immediate feedback about the representation (such as a corresponding change in the simulation). This can help make abstract concepts more understandable (Linn, Bell, et al., 2004) such as how changing one variable like dissolved oxygen can affect an outcome like water quality (i.e., the abstract concept of cause and effect; Metcalf et al., 2000). Table 4 describes various software examples in which students can directly manipulate representations and see their effects.

To more fully illustrate this guideline, we elaborate on an example of how one of its strategies—Strategy 3b—plays out in software designed to support chemistry learning. In chemistry, different representations of molecular structure, such as ball and stick models, wireframe models, and space-filling models, emphasize different molecular properties such as the number of and angles between bonds or the relative sizes of different atoms in a molecule. The software eChem provides all

TABLE 4
Software Examples of Guideline 3: Use Representations That Learners Can Inspect in Different Ways to Reveal Important Properties of Underlying Data

Software	*Description*
Scaffolding Strategy 3a: Provide representations that can be inspected to reveal underlying properties of data	
Density Learning Environment	The Density Learning Environment uses an intuitive graphical "dots per box" representation in simulations to convey a model of density as how tightly packed the dot "masses" are in each space so learners can inspect objects to reveal mass, volume, and density relations (Snir, Smith, & Grosslight, 1995)
STEAMER	STEAMER allows students to inspect the inner workings of a dynamic model such as various components of a steam propulsion system model (Hollan, Hutchins, & Weitzman, 1984)
Scaffolding Strategy 3b: Enable learners to inspect multiple views of the same object or data	
eChem	eChem includes a range of visualizations providing different views of the molecule that learners build so they can automatically generate correspondences between various molecular representations (Wu, Krajcik, & Soloway, 2002)
TableTop™[a]	TableTop™ allows students to view data sets focusing on aggregate or individual data by examining an aggregate data display and zooming in on individual data elements to inspect that data record (Hancock, Kaput, & Goldsmith, 1992)
WorldWatcher	WorldWatcher links the same location on two different maps that represent two different variables measured on the same spatial display (e.g., temperature of the same geographical area at two different months; Edelson, Gordon, Pea, 1999)
Scaffolding Strategy 3c: Give learners "malleable representations" that allow them to directly manipulate representations	
SimCalc	SimCalc lets students change a mathematical equation by directly manipulating the phenomenon being represented in a simulation, or by manipulating the mathematical expression and seeing its effects in the simulation (Roschelle, Kaput, & Stroup, 2000)

Body in Motion	Body in Motion translates learners' own motion into graphs through computer-based motion detectors, thereby linking learners' first-hand experience of the phenomenon (their own motion) with a symbolic expression (Nemirovsky, Tierney, & Wright, 1998)
GenScope	GenScope lets students manipulate linked genetics representations in which manipulating the allele representation of a gene causes a corresponding change in the DNA sequence representation as well as changes in the trait (e.g., wings, color) at the phenotype level (Horwitz, 1996)

[a]TableTop™ is available from the Sunburst Web site (http://www.sunburst.com).

these different views of molecules (Wu, Krajcik, & Soloway, 2002). The system enables students to automatically generate correspondences between various molecular representations (Figure 6). Similarly, 4M Chem Environment (Kozma, Russell, Jones, Marx, & Davis, 1996) helps students link chemical notations to

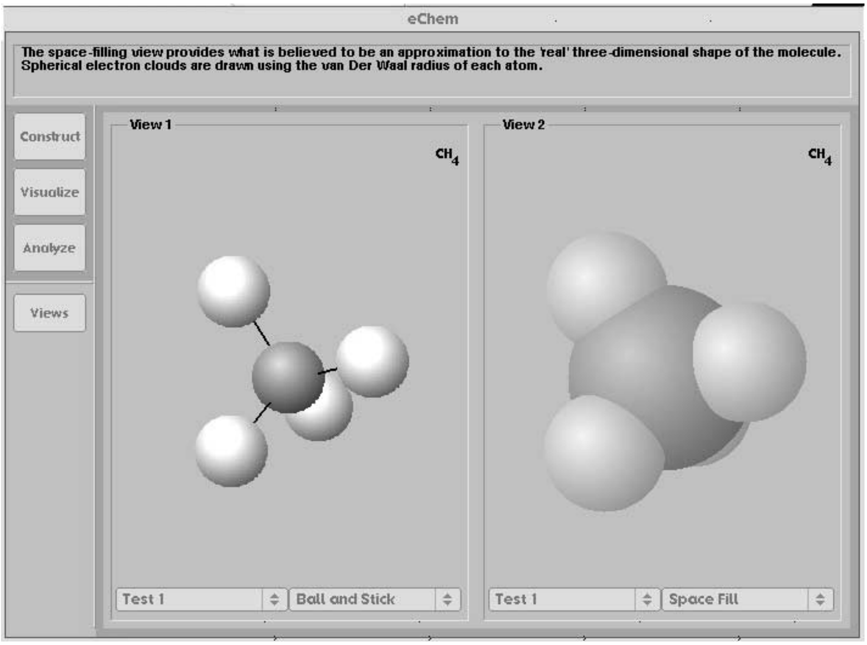

FIGURE 6 eChem provides linked views of multiple representations of the same chemical structure: the chemical formula, ball and stick models, and space fill models (representing spatial relations between constituents).

molecular level animations, graphs, and videos of the phenomena to make connections between the formalisms of the discipline and the phenomena.

The eChem and 4M Chem Environment software and the examples in Table 4 vary in the nature of the connection between representations and phenomena, but the core idea involves tools that help learners make the links between representations and their meaning. Such explicit links can encourage learners to process the results of their actions more deeply, going beyond superficial understandings of the representations. These links can also help learners be more productive in their scientific investigations by enabling them to more easily test ideas and evoke more understandable results. At the same time, sufficient practice may help learners acquire fluency with the more sophisticated formalizations of the discipline, such as force vectors or mathematical equations, so that they require less and less support over time to decode these formalisms. Tools like these, then, help address one of the most critical sense-making challenges for learners: the gap between the ways learners intuitively think about a phenomenon and how it is represented by experts.

SCAFFOLDING PROCESS MANAGEMENT

Sense making involves the core processes in scientific inquiry, but learners need to manage these processes, making decisions about what steps to take next in their investigations. In this section, we consider how to scaffold the challenges in managing the science inquiry process.

The Nature of Process Management

Classic models of problem solving contain both basic operations and a set of control processes (e.g., Anderson, 1983). Our characterization of scientific inquiry includes the process management mechanisms that direct the knowledge and strategies needed to control and steer the investigation itself such as implementing an investigation plan and keeping track of hypotheses and results.

Process management is particularly critical given the ill-structured nature of inquiry. A science investigation is ill-structured because it lacks a definitively prescribed manner for how the problem should be tackled (M. Davis, Hawley, McMullan, & Spilka, 1997) and because one cannot always define in advance the exact process to find a solution (Newell & Simon, 1972; Simon, 1973). There is a large range of activities to perform in an investigation, and one must constantly take stock of previous work to select and perform activities that may take the investigation a step closer to completion (Krajcik et al., 1998; National Research Council, 1996; Quintana et al., 1999; Reiser et al., 2001).

Obstacles Learners Face in Process Management

Managing the science inquiry process thus results in several related challenges for learners (Quintana et al., 1999). For example, learners lack the knowledge experts have about the activities that constitute inquiry and the procedures for performing those activities (Anderson, 1983; Bransford et al., 2000; Springmeyer, Blattner, & Max, 1992), so they may not know what actions are most relevant.

Second, learners lack the strategic knowledge needed to select activities and coordinate the inquiry (Bransford et al., 2000). Indeed, in complex domains, the strategic knowledge managing the basic operations of the discipline constitute the core of what is learned, typically through decades of experience (Anderson, 1983; Newell & Simon, 1972). Without such expertise, learners can be overwhelmed by the complexity of options available, making it difficult to direct their investigations, see what steps are relevant and productive, and make effective activity decisions.

A third related challenge is that learners can be distracted by the less important managerial "chores" that need to be performed throughout an investigation. Management tasks can require a significant amount of time and effort (Knapp, 1994). Whereas experts find such management tasks annoying, learners who are trying to maintain a foothold on more significant activities can find them detrimental. Therefore, learners need support to help them automatically handle routine tasks that are not as salient to the learning goals themselves.

Our scaffolding guidelines for process management help learners by describing activity spaces in ways that structure tasks, specifying when and how to perform different activities in a science investigation, and supporting the routine activities that can divert cognitive focus away from the salient aspects of the task.

Scaffolding Guideline 4: Provide Structure for Complex Tasks and Functionality

Guideline 4 suggests that tools should structure learners' tasks and tool functionality should be structured to support learners in seeing what steps are possible, relevant, and productive. Specifically, this guideline looks at how software tools can constrain or describe tasks in ways that make them more accessible to learners. The strategies associated with this guideline help learners by limiting the scope of the activity space within which learners work. This is similar to how apprentices are given parts of an authentic task rather than being expected to work on the entire task at once (Lave & Wenger, 1991).

We identified three scaffolding strategies for implementing this guideline:

- 4a. Restrict a complex task by setting useful boundaries for learners.
- 4b. Describe complex tasks by using ordered and unordered task decompositions.
- 4c. Constrain the space of activities by using functional modes.

The first strategy (Strategy 4a) for structuring tasks involves restricting the richness and complexity of learner activities to a level in which learners can focus on the most important parts of their tasks (Collins, 1996). In general, designers should reduce the complexity of examples, visualizations, or models to make the science more accessible to learners (Linn, Bell, et al., 2004; Linn & Hsi, 2000). For example, students can be impeded by having to track down and use the large scientific data sets that expert scientists would use in an investigation. Software can effectively restrict the data collection task by providing learners with an authentic but more manageable data set to explore. For example, WorldWatcher (Edelson et al., 1999) contains authentic and manageable preselected data that learners can use to focus on important aspects of climate investigations. Table 5 describes some software examples that implement this scaffolding strategy.

A second strategy (Strategy 4b) for structuring tasks is to decompose a complex task into its constituents following the notion that novices can be guided by making activity spaces explicit in an interface (Favorin & Kuutti, 1996). Two approaches to decomposing tasks include using unordered and ordered task decompositions. *Unordered* task decompositions involve displaying an unconnected space of activ-

TABLE 5
Software Examples of Guideline 4: Provide Structure for Complex Tasks and Functionality

Software	*Description*
Scaffolding Strategy 4a: Restrict a complex task by setting useful boundaries for learners	
WorldWatcher	WorldWatcher (Edelson, Gordon, & Pea, 1999) contains preselected data (e.g., specific sets of climate data) that are relevant and manageable in size so learners can access and examine them, but rich enough to be educationally useful
Media Fusion	Media Fusion can be seeded with video clip libraries (e.g., clips of global warming experts) so students can create digital video linked to data analysis tools to convey their understanding of some domain (Bellamy, 1996)
Scaffolding Strategy 4b: Describe complex tasks by using ordered and unordered task decompositions	
Symphony	The main inquiry process map in Symphony's planning workspace decomposes an investigation into manageable components by visually describing the space of possible inquiry activities (Quintana, Eng, Carra, Wu, & Soloway, 1999)

KIE/WISE	KIE uses an ordered checklist of inquiry tasks to show learners the sequential parts of an activity (Bell, Davis, & Linn, 1995); WISE (Linn & Slotta, 2000) changed the analogous interface feature to instead involve unordered task decompositions, allowing more fluid movement among tasks
Personal Assistants for Learning (PALs)	PALs help students interpret and apply important physics concepts and principles, (e.g., Newton's mechanics) with an explicit visual representation of qualitative reasoning processes so students can see the process they should be following (Reif & Scott, 1999)
Emile	Emile uses pull down "design stage" menus that describe the different steps involved in the design stages of a programming project (Guzdial, 1994)
Scaffolding Strategy 4c: Constrain the space of activities by using functional modes	
Model-It™	Model-It™ contains three functional modes: the "plan mode" only allows model planning tasks and tools, the "build mode" only allows model building tasks and tools, and the "test mode" allows only model testing tasks and tools (Metcalf et al., 2000)
Galápagos Finches	Galápagos Finches has different areas corresponding to different aspects of learners' investigation: areas about the environment, the populations they are studying in that environment, field notes, and so forth (Reiser et al., 2001; Tabak, Smith, Sandoval, & Reiser, 1996)
Training Wheels	Training Wheels organized and constrained the user interface to only allow certain functional paths through the software by identifying and disallowing functionality that new users were not ready for or could be confused by (Carroll & Carrithers, 1984)

Note. KIE = Knowledge Integration Environment; WISE = Web-Based Inquiry Science Environment.

ity possibilities so learners can see the space of possible activities without explicitly being given specific activity sequences or selections. *Ordered* task decompositions involve using representations that can be categorized generally as classification diagrams, which describe processes and their constituents (Kress & van Leeuwen, 1996) to display the steps and sequence needed to perform an activity. KIE, for example, uses a checklist of inquiry activities to show learners the sequential parts of an activity (Figure 7). Students complete one activity before moving on to the next (Bell, Davis, & Linn, 1995). In subsequent work on WISE (Linn & Slotta, 2000), the interface supports unordered task decompositions, allowing more fluid movement among tasks. Another unordered task decomposition for inquiry is the process map in Symphony (Figure 8). Table 5 describes other examples of software using task representations for learners.

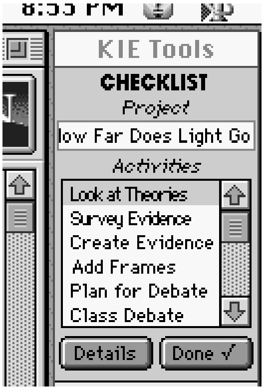

FIGURE 7 The Knowledge Integration Environment (KIE) uses an activity checklist to structure the different activities involved in science work.

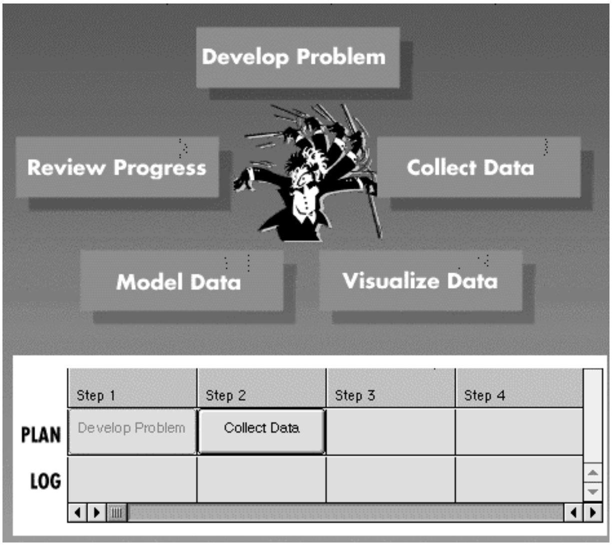

FIGURE 8 The Symphony planning workspace uses a process map to visually describe the space of possible inquiry activities.

Structuring a complex set of tasks can also be supported through scaffolding Strategy 4c, which states that designers can constrain the space of activities by incorporating functional modes that offer only certain relevant subsets of software functionality. This is somewhat similar to Strategy 4b. However, whereas the task decomposition Strategy 4b deals with the visual description of task sequences or activity spaces on the software interface, the functional modes Strategy 4c is concerned with what software tools are provided and when. Tools with a more constrained and streamlined approach help learners make progress in a new domain (Rosson, Carroll, & Bellamy, 1990). Constraining activity spaces with functional modes is a way to prevent learners from facing an overwhelming situation in which all possible tools and functions are available. Table 5 describes some examples of software that use functional modes to constrain the space of activities for learners.

We can use Model-It to illustrate the guideline of providing structure for tasks and functionality by looking at the two scaffolding strategies about task decompositions (Strategy 4b) and functional modes (Strategy 4c). The implementation of Strategy 4b in Model-It is an activity palette to display the high-level tasks involved in system modeling. The palette consists of three options representing the three key components of modeling—plan, build, and test (Metcalf et al., 2000). This serves as an ordered task decomposition; the palette makes the activity structure visible to learners and only allows learners to progress to the next step after they have completed each phase. The implementation of Strategy 4c in Model-It involves using three functional modes accessible from that palette: From each mode (i.e., plan, build, and test), only the relevant tools associated with that phase of activity are accessible, simplifying the choices to relevant options and preventing the potentially overwhelming situation in which all modeling tools are available at all times.

In Model-It and the other examples illustrating Guideline 4, one sees how software tools can help learners make decisions to drive and coordinate their inquiry. Specifically, some of the tools constrain what learners can access or engage in at a given time (or at all) to limit their activity space. Other examples show how tools can describe the tasks learners are to do in ways that are more structured and thus more manageable. As in a person-to-person cognitive apprenticeship, this aspect of scaffolding provides learners with manageable tasks on which they can work.

Scaffolding Guideline 5: Embed Expert Guidance About Scientific Practices

The sense-making scaffolding guidelines described earlier focused on how tools can help learners understand and develop knowledge to engage in the substance of scientific practices. Guideline 4, our first process management guideline, emphasized how software tools can describe or constrain activity spaces to make tasks more tractable for learners. Now, Guideline 5 provides another approach for increasing the tractability of tasks to help learners manage the processes entailed in the scientific practices.

Experts engaging in inquiry may see clear paths and strategies. Learners, however, rely on less elaborated and sophisticated understandings of the practice and thus encounter obstacles in understanding the specifics of performing scientific practices. Guideline 5 recommends providing access to expert knowledge about scientific practices (e.g., explaining, observing, and inferring) so learners can understand both how and why they should embark on a particular task and how to strategically steer their investigation. Expert knowledge can be made available to learners in tools that parallel the guidance provided in a more traditional, person-to-person cognitive apprenticeship. This can help learners understand the nature and rationale for scientific practices.

We identified two strategies for implementing this guideline:

5a. Embed expert guidance to clarify characteristics of scientific practices.
5b. Embed expert guidance to indicate the rationales for scientific practices.

First, embedding contextualized expert guidance can help learners develop competence in the scientific practices they are trying to learn (Bransford et al., 2000). Scaffolding Strategy 5a then recommends embedding expert guidance about the characteristics of scientific practices to help learners understand these practices and identify appropriate steps to take as they engage in them. Such expert guidance can be embedded in software with hints or prompts that elaborate and clarify different aspects of scientific practices such as providing criteria or specific goals learners need to understand to guide the investigation. Table 6 describes software that provides expert guidance through hints, guiding questions, and examples learners can study to understand why experts made the decisions they made; one example, from KIE, is further elaborated later.

A related point is that learners also need to understand the reasons behind the different scientific practices they are performing, which may be difficult for them (Collins, 1996). Therefore, another scaffolding strategy for providing expert knowledge—Strategy 5b—recommends conveying to learners the rationales for performing different scientific practices. Table 6 provides some examples of this strategy.

Elaborating on an example from KIE illustrates both strategies. The Mildred guide in KIE provides hints in the form of information or explanations (Bell & Davis, 2000). For example, Mildred provides guidance about the process of critiquing evidence, suggesting that students should pay attention to the credibility of sources and other criteria. This helps them engage in the scientific practice of critique by giving them heuristics for critiquing (an aspect of scaffolding Strategy 5a). Furthermore, KIE addresses scaffolding Strategy 5b with a "Why to do it?" button that describes rationales for the different learning activities (E. A. Davis & Bell, 2001).

TABLE 6
Software Examples of Guideline 5: Embed Expert Guidance About Scientific Practices

Software	*Description*
Scaffolding Strategy 5a: Embed expert guidance to clarify characteristics of scientific practices	
KIE and WISE	The "Mildred" guide in KIE provides process hints in the form of information or explanations (e.g., explaining the term *critique* if students are critiquing evidence) by suggesting that they pay attention to the credibility of sources and other criteria (Bell & Davis, 2000)
ExplanationConstructor	The ExplanationConstructor contains explanation guides that help learners understand the specific constituents an explanation needs to contain to address questions in the target discipline (Sandoval, 2003)
Sherlock	Sherlock (Lesgold, Lajoie, Bunzo, & Eggan, 1992) is a coached practice environment in which students learn to troubleshoot complex electronics equipment by studying an expert solution to a problem they have solved and contrasting their problem-solving moves with those of the expert
Scaffolding Strategy 5b: Embed expert guidance to indicate the rationales for scientific practices	
KIE and WISE	KIE contains a "Why to do it?" button that describes rationales for the different learning activities; although most students do not use this form of guidance (E. A. Davis & Bell, 2001), it may be important to have available
Symphony	Symphony embeds rollover activity rationale guides describing the different activities shown in the main process map as students plan their investigations (Quintana, Eng, Carra, Wu, & Soloway, 1999; Quintana, Krajcik, & Soloway, 2002)
Goal-based scenario (GBS) software	A GBS is a highly contextualized simulated system consisting of different video-based coaches, experts, and critics for a range of domains (e.g., disease diagnosis, art history, and criticism) to interject commentary about the course of action learners took in the simulation (Schank & Cleary, 1995)

Note. KIE = Knowledge Integration Environment; WISE = Web-Based Inquiry Science Environment.

It is important to emphasize the difference between Guideline 5, which focuses on providing expert guidance about engaging in scientific practices, and the sense-making guidelines, which are focused on helping learners implement the operations of sense making such as examining and drawing inferences from data. The examples here for Guideline 5 illustrate software tools that can help learners make strategic decisions about how to engage in their scientific investigation by telling them both what the important scientific practices are and why they are important, as a mentor might tell an apprentice.

Scaffolding Guideline 6: Automatically Handle Nonsalient, Routine Tasks

Whereas the previous two process management guidelines focused on structuring and embedding expert guidance about scientific practices, Guideline 6 provides further process management support by reducing the cognitive load learners need to bear as they engage in scientific inquiry. Engaging in complex practices requires concentration on salient activities to reach an optimal state of deep cognitive focus (Csikszentmihalyi, 1991). Such a focused state is important for learning, but to reach such a state, it is especially important to minimize distractions and disruptions that can interfere with the sense of deep engagement in the work at hand (Miyata & Norman, 1986).

Because potential disruptions for learners can arise from having to deal with management and navigational tasks, Guideline 6 recommends automatically handling such nonsalient, routine tasks. This approach builds on prior conceptualizations of technology as minimizing the overhead for complex work (e.g., arguments for calculators in mathematics learning) and as cognitive tools that offload nonproductive work, thereby reducing the load on memory and cognitive resources (Anderson, Boyle, & Reiser, 1985; Anderson et al., 1995).

We identified three strategies for implementing this guideline:

- 6a. Automate nonsalient portions of tasks to reduce cognitive demands.
- 6b. Facilitate the organization of work products.
- 6c. Facilitate navigation among tools and activities.

One strategy (Strategy 6a) to reduce the distraction for learners is to have software perform routine tasks, such as calculating results or drawing graphs, that might prohibit them from focusing on important tasks more relevant to their learning goals. Table 7 describes examples of such task automation.

Other examples of distracting tasks involve the managerial aspects of complex practices such as managing artifacts like data sets, models, plans, and notes. Scaffolding Strategy 6b suggests facilitating artifact organization with a buffer layer between learners and the computer file system so learners can easily save, orga-

TABLE 7
Software Examples of Guideline 6: Automatically Handle Nonsalient, Routine tasks

Software	*Description*
Scaffolding Strategy 6a: Automate nonsalient portions of tasks to reduce cognitive demands	
Galápagos Finches	Galápagos Finches automatically generates graphs for student-specified data so students can focus on the more important work of determining appropriate comparisons for testing their hypotheses (Reiser et al., 2001)
Model-It™	Model-It™ automatically calculates variable values given student-defined relationships during a system model run so students can focus on understanding the overall behavior of their model rather than diverting their attention to the variable calculations (Metcalf, Krajcik, & Soloway, 2000)
Geometer's Sketchpad™[a]	Geometer's Sketchpad™ automatically generates different geometrical structures for students so they can focus on examining and learning different properties of these structures (Jackiw, 1995)
Scaffolding Strategy 6b: Facilitate the organization of work products	
Galápagos Finches	Galápagos Finches has a data log that acts as a repository where students can easily store different graphs and data sets they create during their investigation
TableTop™	As students analyze data, TableTop™ allows them to capture and organize plots in a slideshow-like artifact where they can annotate and present data (Hancock, Kaput, & Goldsmith, 1992)
Scaffolding Strategy 6c: Facilitate navigation among tools and activities	
Symphony	Symphony uses tabbed activity workspaces that allow students to quickly open different workspaces for different activities and quickly navigate between them (Quintana, Eng, Carra, & Soloway, 1999; Quintana, Krajcik, & Soloway, 2002)
Smithtown	Smithtown, a discovery environment for learning about economics, gives students palettes to easily switch between different tools and activities for collecting, organizing, and viewing data as they develop economic models (Shute, Glaser, & Raghavan, 1989)

[a]Geometer's Sketchpad™ is available from the Key Curriculum Press Web site (http://www.keypress.com).

nize, and find artifacts. Table 7 describes several examples of software features that facilitate the organization of work products.

Besides dealing with the distractions involved with artifact organization, learners also can be distracted by moving between the different software tools they may use in an investigation. Scaffolding Strategy 6c suggests facilitating seamless navigation among tools and activities to avoid the distraction of having to move to the underlying operating system to launch or shift between tools. Table 7 describes some examples.

We can illustrate Guideline 6 more fully with an example from the Galápagos Finches software (Reiser et al., 2001; Tabak, Smith, Sandoval, & Reiser, 1996). First, Galápagos Finches automatically draws graphs—the less important part of the task—for student-specified data so students can focus on the more important task of using the graphs to determine appropriate comparisons for testing their hypotheses (scaffolding Strategy 6a). Galápagos Finches also has a data log that acts as a repository where learners can easily store different graphs and data sets they create during their investigation (see Figure 9). The data log maintains the different products in the order learners saved them, which helps learners easily organize and find their work products (scaffolding Strategy 6b).

The idea in Guideline 6 of automatically handling nonsalient, routine tasks represents perhaps the most straightforward of the three process management guidelines. Yet questions about what aspects of tasks should be assisted by technology are often quite contentious among educators. Educators might argue that students will not learn to graph if the software generates graphs for them, or they will not learn algorithms for long division if they use calculators. Sherin et al. (this issue) discuss the need for being explicit about learning goals in identifying scaffolding. Arguments for scaffolding entail assumptions about goals—in this context, for

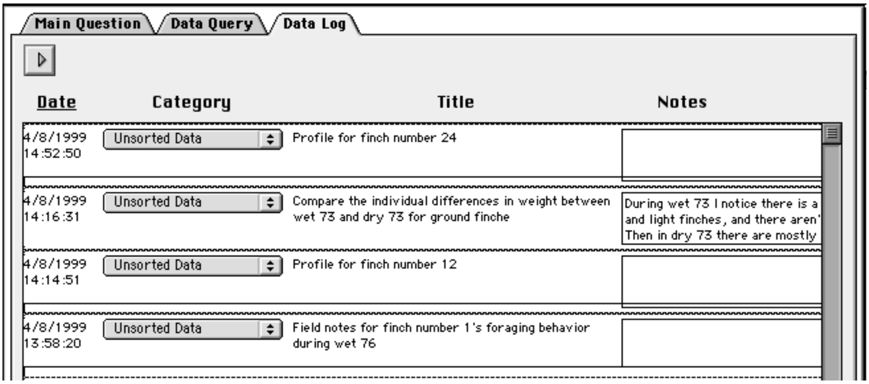

FIGURE 9 The Data Log in Galápagos Finches automatically saves all graph queries associated with the query specification and timestamp.

these learners, is learning the mechanics of graphing the learning objective? Or is graphing a means to an end with the real goal being to learn data analysis skills? Eliminating aspects of tasks that are less central to the given learning goals can reduce learners' cognitive load and allow them to focus on the more central learning tasks. The tools we use to illustrate this guideline and its associated strategies, then, provide examples of some of the ways these less salient tasks can be minimized or eliminated, freeing learners to focus on tasks that directly address learning objectives.

SCAFFOLDING ARTICULATION AND REFLECTION

We have delineated the scaffolding guidelines and strategies for sense making and process management. The third constituent of scientific inquiry includes the complementary processes of articulation and reflection. We combine articulation and reflection because they are mutually supportive processes that are difficult to disentangle.

The Nature of Articulation and Reflection

The articulation and reflection processes support process management and sense making as well as the collaboration needed to make inquiry effective. A critical aspect of inquiry involves constructing and articulating an argument; this in turn involves reviewing, reflecting on, and evaluating results; synthesizing explanations; and deciding where the weaknesses and strengths are in one's thinking (Collins & Brown, 1988; E. A. Davis, 2004; E. A. Davis & Linn, 2000; Loh et al., 2001).

Reflection and articulation play a key role throughout many instructional approaches. For example, Linn's scaffolded knowledge integration framework (Linn, 1995; Linn, Davis, et al., 2004; Linn & Hsi, 2000) highlights the importance of making thinking visible and promoting autonomous lifelong learning, and reflection is key in these. Engaging in reflective self-assessment can help students improve their understanding of content and inquiry (White & Frederiksen, 1998). Reflective processes help learners proceed with an investigation process, detect when past decisions should be reconsidered, identify dead ends, and remember to address goals (Schoenfeld, 1987). Furthermore, reflecting and articulating intentionally and publicly can promote knowledge building at the individual and community level (Scardamalia & Bereiter, 1991).

Obstacles Learners Face in Articulation and Reflection

Learners, however, face several challenges in articulating and reflecting productively. First, learners often do not realize that they should articulate their ideas (Linn

& Songer, 1991; Loh et al., 2001; Scardamalia & Bereiter, 1991; van Zee & Minstrell, 1997). In fact, learners sometimes interpret opportunities for articulation and reflection as merely being blanks to fill in (E. A. Davis & Linn, 2000; Schauble, Glaser, Duschl, Schulze, & John, 1995). Furthermore, learners often do not know how to reflect productively (E. A. Davis, 2003a; Palincsar & Brown, 1984); thus, they need support to identify good ways to reflect on and articulate their ideas.

A second related challenge is that learners may focus on achieving quick outcomes (Schauble, Klopfer, & Raghavan, 1991). Learners working collaboratively do not necessarily identify or reconcile mismatches in group members' ideas unless they are required to reach consensus (Cohen, 1994; Webb, 1983) or commit explicitly (Bell, 1998; Golan, Kyza, Reiser, & Edelson, 2001; Reiser, this issue).

Third, learners have difficulty in planning and monitoring their investigations. They forge ahead without considering alternatives or ramifications of their decisions, get bogged down in logistical details of their work (Schauble, Glaser, et al., 1991), and focus on superficial measures of progress (Lan, 1996; Palincsar & Brown, 1984; Tien, Rickey, & Stacy, 1999; White & Frederiksen, 1998). Learners may develop illusions of competence that preclude them from identifying weaknesses in their knowledge (E. A. Davis, 2003a). Studies have shown that students who do not appropriately plan their work and monitor their understanding tend to not perform as well as students who do (Chi, Bassok, Lewis, Reimann, & Glaser, 1989; Flower & Hayes, 1980; Recker & Pirolli, 1995). Thus, learners need support for articulating and reflecting as they plan and monitor their investigations (Bielaczyc, Pirolli, & Brown, 1995; Linn & Songer, 1991).

A fourth challenge for learners in articulating and reflecting stems from the fact that the form of the articulated epistemic products of science is critical (Collins & Ferguson, 1993). For example, claims need to be supported with evidence, and arguments need to be warranted (Toulmin, 1958/1964). Descriptions should include observations but exclude inferences. Explanations should refine or expand on ideas or infer consequences (Chi & Bassok, 1989), and explanatory arguments should explore multiple hypotheses, present coherent assertions, provide evidence, and justify connections between claims and evidence (Sandoval, 2003). However, science learners have trouble with all of these practices. For example, when learners describe objects and phenomena, they may not notice important details or they may confuse description and explanation (Bell, 1997; Driver, Leach, Millar, & Scott, 1996; Gallas, 1995; Songer & Linn, 1991). When they discuss causality, learners may omit justifications or reasons (e.g., Bell, 1997; Kuhn, 1993; Sandoval & Reiser, 2004).

Scaffolding Guideline 7: Facilitate Ongoing Articulation and Reflection During the Investigation

We have identified one scaffolding guideline to address these challenges and support learners with reflection and articulation. It involves support that encourages students to articulate and reflect on their ideas in ways that are scientifically productive. We identified four strategies for implementing this guideline:

7a. Provide reminders and guidance to facilitate productive planning.
7b. Provide reminders and guidance to facilitate productive monitoring.
7c. Provide reminders and guidance to facilitate articulation during sense making.
7d. Highlight epistemic features of scientific practices and products.

Scaffolding Strategies 7a and 7b involve providing reminders and guidance to promote productive planning and monitoring. Linn, Bell, et al. (2004) presented several design principles associated with promoting articulation and reflection for planning and monitoring such as including prompts that remind learners of the aspects of the project they have yet to complete so they can monitor their progress and plan accordingly. Our examples supporting planning and monitoring (see Table 8) range from fairly implicit reminders or guidance, such as the ability to store records of important findings (e.g., Progress Portfolio; Loh et al., 1998), to more explicit support, such as directed prompts for planning or monitoring (e.g., KIE; E. A. Davis & Linn, 2000).

When learners articulate their ideas, they also need help to make sense of the science they are learning. As discussed earlier, learners often do not know to—or simply neglect to—articulate their ideas, or they need help to do so productively. Scaffolding Strategy 7c addresses these challenges by providing guidance—including simple reminders—for articulating ideas to promote sense making. Typically, the examples here involve providing a mechanism for recording questions, findings, or ideas during the investigation (see Table 8). For example, the Progress Portfolio provides mechanisms for learners to document important findings from their investigation (an image from a Web browser, simulation, or database) and to record their thinking on the artifact created (see Figure 10).

In addition to support for planning and monitoring, learners also need guidance and support to understand the epistemic characteristics of the products they need to create (e.g., explanations, descriptions, interpretations) and the practices they use to construct these products. This support can help them engage in the discourse of science (Lemke, 1990) and make sense of science concepts. An important idea in design theories is making these epistemic assumptions, which are often tacit in instruction, explicit in the tools and artifacts students use (Linn, Bell, et al., 2004; Reiser et al., 2001; Sandoval, 2003). The examples of scaffolding Strategy 7d in Table 8 focus on understanding scientific argumentation (Bell & Linn, 2000; Sandoval, 2003; Sandoval & Reiser, 2004), the process of coordinating theory and evidence, considerations of the utility and relevance of scientific evidence (Linn, Bell, et al., 2004), and distinguishing between observations and inferences.

TABLE 8
Software Examples of Guideline 7: Facilitate Ongoing Articulation and Reflection During the Investigation

Software	*Description*
Scaffolding Strategy 7a: Provide reminders and guidance to facilitate productive planning	
ExplanationConstructor	The ExplanationConstructor asks students to generate questions that they use to guide their own investigation (Sandoval, 2003)
Symphony	Symphony includes a planning workspace where students see a high-level inquiry process map showing possible activities, and a planning grid where students can drag and place activities to incrementally develop an investigation plan (Quintana, 2001; Quintana et al., 2002)
Scaffolding Strategy 7b: Provide reminders and guidance to facilitate productive monitoring	
KIE and WISE	KIE and WISE can use "Checking Our Understanding" prompts as well as other more generic reflection prompts to encourage students to monitor their progress and their understanding (E. A. Davis, 2003a; E. A. Davis & Linn, 2000)
Geometry Tutor	The Geometry Tutor provides a visual representation of learners' reasoning chains as they work on geometry proofs to help learners keep track of what goals have been met and what gaps remain (Anderson, Bayle, & Yost, 1986)
Scaffolding Strategy 7c: Provide reminders and guidance to facilitate articulation during sense making	
Progress Portfolio	The Progress Portfolio contains prompted text fields and a sticky note tool to enable students to annotate their findings, thus supporting sense making by encouraging them to annotate the artifacts they create with their understanding or interpretation (without providing guidance to direct the sense making; Loh et al., 1998)
CSILE/Knowledge Forum	Knowledge Forum promotes sense making by giving students different opportunities to articulate their ideas with contributions to a communal database of notes that support community knowledge growth (Scardamalia & Bereiter, 1991)

SMILE	SMILE has a "Pin-Up Tool" with prompted text fields that remind learners to justify their features for a design project and connect them to scientific principles (Kolodner, Owensby, & Guzdial, 2004)
Scaffolding Strategy 7d: Highlight epistemic features of scientific practices and products	
ExplanationConstructor	Guiding questions in the ExplanationConstructor make explicit the important components of scientific explanations for the particular discipline (Sandoval & Reiser, 2004)
Animal Landlord	Animal Landlord uses a video frame capture tool and ethogram menu that requires differentiation between observation and inference to help students distinguish between observations and inferences in their scientific descriptions (B. K. Smith & Reiser, 1998)

Note. KIE = Knowledge Integration Environment; WISE = Web-Based Inquiry Science Environment; CSILE = Computer-Supported Intentional Learning Environment; SMILE = Supportive Multiuser Interactive Learning Environment.

We expand on an example of this final strategy and this guideline more generally. Through their use of ExplanationConstructor, students are able to productively articulate and reflect on their ideas (Sandoval & Reiser, 2004). First, the ExplanationConstructor asks students to generate questions and subquestions that they use to guide their own investigation (Sandoval, 2003). This helps them plan (scaffolding Strategy 7a). Furthermore, guiding questions in the ExplanationConstructor make explicit the important components of scientific explanations for the particular discipline, and the structure of the artifact students create makes explicit the core epistemic features of an explanatory argument, addressing scaffolding Strategy 7d (see Figure 11). Students create an explanation that contains embedded evidence (attached to each claim) and is created in a window that contains the relevant guiding questions for the discipline. Explanations are connected to questions students articulate, and these questions can contain two or more alternative explanations, suggesting the importance of discriminating alternative hypotheses.

The examples we have selected to illustrate Guideline 7, including the previous ExplanationConstructor example, address the obstacles learners face in reflecting on and articulating their ideas productively. The tools variously remind learners to plan investigations, monitor progress, or make sense of ideas. They provide guidance about how to do so productively. Finally, they help them do so in ways that are true to the demands of the discipline. By supplementing the learners' own metacognition and by helping learners use epistemically appropriate practices and

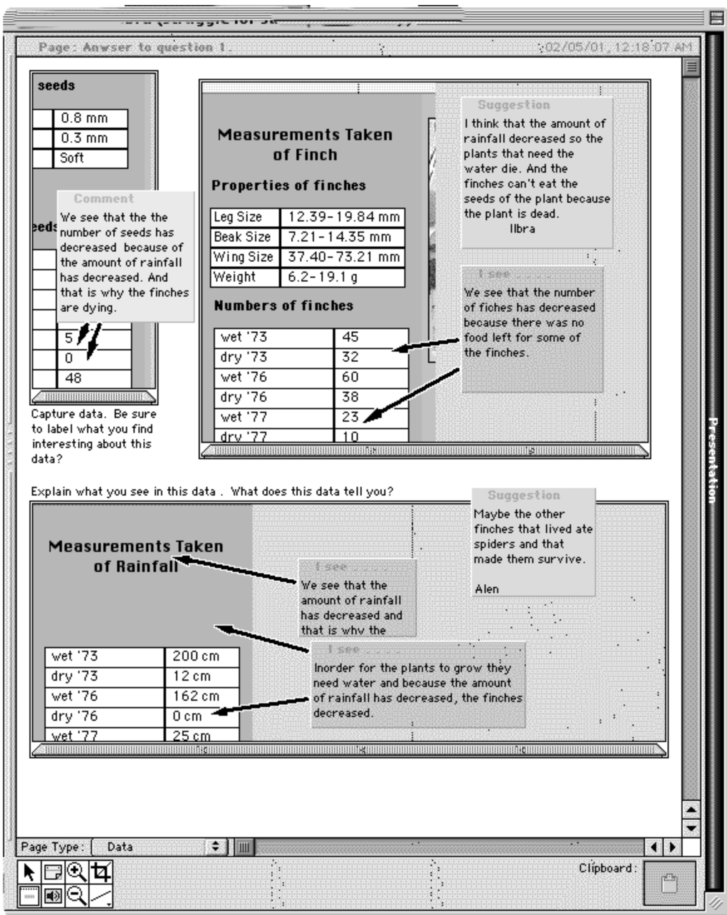

FIGURE 10 The Progress Portfolio enables students to create a record of their thinking attached to the artifacts they create in an investigation. Here the students have captured data and use sticky notes to articulate their interpretations.

products, the scaffolding approaches we have outlined here help learners engage in more productive cognitive activities.

DISCUSSION

Our goal in this article was to argue the need for a scaffolding design framework to provide a common theoretical framework that defines rationales and approaches for how software tools can scaffold learners. We provide an initial proposal for

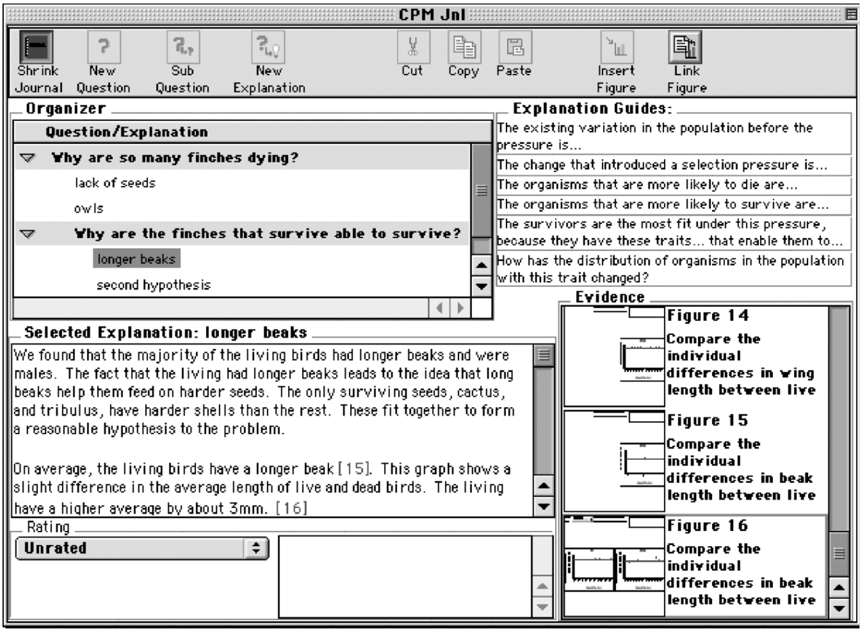

FIGURE 11 ExplanationConstructor contains guiding questions that provide discipline-specific support on the necessary constituents of an explanation from a target theoretical framework.

such a framework in the domain of science inquiry to provide a theory of pedagogical support and a mechanism that describes successful scaffolding approaches with respect to the obstacles learners face in science inquiry. In this section, we discuss the utility of the framework and consider its scope.

The Utility of the Framework

With this framework, we intend to synthesize the wisdom drawn from the research literature and the field of practice to characterize principled approaches for supporting learners using software as they engage in scientific inquiry. If this framework is successful, it will be useful for the field along several dimensions.

First, the framework provides a basis to develop an integrated theory of pedagogical support for complex learning with software. Our perspective is that scaffolding researchers and designers need to synthesize theoretical claims across different design research efforts. Abstracting from the specifics of design ideas and the experiences with their enactment is a critical aspect of design research (Edelson, 2002). A common framework in which to characterize and motivate design is one mechanism that would allow design research to be more of a community knowledge-building enterprise.

Second, the framework provides a mechanism to guide the construction of an empirical knowledge base about successful design approaches for scaffolding. Developing general principles, abstracted from particular implementations and particular research projects but grounded in identified learning obstacles, will allow researchers to frame hypotheses about what pedagogical approaches are effective in supporting learners.

We consider the need for and potential utility of such a framework through an example. A basic research goal in a set of studies might be an investigation of whether prompting students during their investigations could support learning. With the scaffolding design framework, one could probe such a research question at a deeper level by explicitly uncovering important distinctions. For example, one distinction is that prompts for reflection are intended to promote quite different kinds of outcomes than prompts for explanations, and indeed, different prompts can work to produce different outcomes (E. A. Davis & Linn, 2000; E. A. Davis, 2003a). Application of the scaffolding design framework can lead to a more detailed set of empirical predictions, providing a common language to distinguish between approaches and a mechanism for making more nuanced analyses. Our framework characterizes the distinctions that we expect to matter both in the intent and the expected outcomes of scaffolding approaches. By tying interface elements such as prompts to a substantive analysis of the nature of the task at hand (i.e., reflection tasks vs. explanation tasks), we improve the utility of the analysis.

A third use of the framework is for guiding design. Designers of educational software can review existing or candidate designs and identify scaffolding approaches that may be appropriate to incorporate. The framework is intended to provide guidance in the form of design heuristics, not specific prescriptions. It provides guidance about what learning challenges are important to address and provides candidates, through the guidelines and strategies, for approaches to address those challenges. However, the framework cannot tell a designer which specific strategy to select for which context. Specific design decisions must be based on the kinds of task and obstacle analyses we have described here.

Scope and Comprehensiveness of the Framework

We have illustrated the utility of the framework, and next we address its scope. A first question to consider is the completeness of the analysis. We have synthesized a broad array of theoretical and empirical literature on science learning and the obstacles that arise for learners engaging in science practices to develop a framework that organizes software scaffolding guidelines around those science tasks and obstacles. We have considered a broad range of more than two dozen software tools, many of which are represented in the examples in this article. We present this framework, however, as an initial proposal to facilitate research on software scaffolds in science. Its comprehensiveness needs to be tested, leading to extensions and adaptations of the framework to encompass a broader diversity of design ideas.

While attempting to be thorough, we have also bounded our analyses. First, we have focused on the core processes of inquiry, mainly from the perspective of working with data and constructing explanations. In these aspects of the problem solving, we suggest the analysis of inquiry in the framework represents the central themes of a sense-making view of inquiry in the literature. Correspondingly, we suggest the scaffolding guidelines are relatively comprehensive in responding to the obstacles that have been identified in prior research and in characterizing the software scaffolding design approaches represented in the literature. However, we chose to focus on what we viewed as the central and most discipline-specific of scientific practice (i.e., sense making being supported by process management, articulation, and reflection), and we have not dealt in our framework with some of the broader issues of scientific practice in which these core processes are embedded. For example, one could focus on the problems inherent in collaboration between learners, or in communication between teachers and students, or in whole-class presentation and discussion. Certainly, some of our guidelines and strategies do touch on collaboration when the collaboration is around scientific practices such as jointly constructed explanations. However, the framework does not include a systematic analysis of general approaches for supporting collaborative work. Many excellent examples of scaffolds for collaboration exist, such as those in the "launcher" projects in "Learning by Design" (Holbrook & Kolodner, 2000; Kolodner et al., 2003) or Hoadley's (1999) socially relevant representations. These, however, are beyond the scope of this article.

Another aspect of practice not addressed relates to scaffolding students' data collection. Although we see this as an important aspect of sense making, most of the empirical work that has been done in science software has focused on what happens once the data has already been collected (with a few notable exceptions, such as Linn & Songer, 1991). Thus, our framework may not adequately address scaffolding approaches for data collection.

Furthermore, although we argue that our scaffolding guidelines represent a synthesis of current approaches, the scaffolding strategies addressing those guidelines are not intended to be comprehensive. As designed responses to learning challenges, we would expect future design work to uncover new types of scaffolding strategies that can achieve these guidelines. The software strategies presented in this framework are meant to capture and situate current design approaches in a coherent theoretical framework rather than be a comprehensive set of possible strategies.

A final way we have bounded our investigations is that we have considered only scaffolding embedded within software. Other kinds of scaffolding are critical—for example, teachers, peers, and curriculum materials all play extremely important roles in the ways students learn science. Tabak (this issue) provides an extensive discussion of the synergies that exist between software scaffolding and other delivery mechanisms.

Another question to pose concerns the generality of our framework. As we have noted, we limit our claims to supporting science inquiry with software tools. Our analyses proceeded with an analysis of the challenges of this particular discipline, and our examples were primarily drawn from tools that support science or from related efforts to support mathematics. Some of the principles in the framework may be argued to have generality beyond the discipline of science. Testing the generality of the scaffolding design framework is an important ongoing research agenda—as is testing the generality of any design principle or set of design principles.

In closing, we have argued the need for a synthesis of both craft knowledge, drawn from design efforts, and theoretical accounts of scaffolding, drawn from design and empirical research. We can look back to the initial descriptions of software-realized scaffolding (Guzdial, 1994; Soloway et al., 1994), which were instrumental in defining how software can instantiate scaffolding approaches to support learners and characterizing the construct of scaffolding with respect to software. Our work here expands on that view of scaffolding as well as the perspectives presented elsewhere in this issue (Reiser, this issue; Sherin et al., this issue; Tabak, this issue) to articulate a set of scaffolding guidelines and strategies gleaned from theory and practice. We have presented an initial scaffolding design framework constructed in a rich terrain of learning opportunities and challenges. The framework has been successful in accounting for a broad range of software types and styles of interaction. It presents candidate principles to account for how software tools can transform tasks within science inquiry to make them more tractable and productive for learning. We hope that the framework can serve as a springboard for the research community to explore a range of scaffolding issues and approaches.

ACKNOWLEDGMENTS

An earlier version of this article was presented at the October 2002 International Conference of the Learning Sciences. This research was conducted as part of the KDI/ASSESS project and the Center for Learning Technologies in Urban Schools, supported, in part, by National Science Foundations Grants REC–9980055 and REC–972038, respectively. The opinions expressed herein are those of the authors and not necessarily those of the National Science Foundation. All of the authors were involved in conceptualizing the work presented here. The writing was done by Chris Quintana, Brian J. Reiser, and Elizabeth A. Davis, with significant input from Joseph Krajcik.

We thank all of the researchers involved with the ASSESS project for their work and feedback, especially Laura Barden-Gabbei, Barbara Crawford, Andrew

Lumpe, Cathleen Norris, and Carla Zembal-Saul. We also thank Janet Kolodner, Naomi Miyake, and our anonymous reviewers for their comments and suggestions. Finally, we thank all the members of our research groups for their helpful comments on the article.

REFERENCES

Anderson, J. R. (1983). *The architecture of cognition.* Cambridge, MA: Harvard University Press.

Anderson, J. R., Boyle, C. F., & Reiser, B. J. (1985). Intelligent tutoring systems. *Science, 228,* 456–462.

Anderson, J. R., Boyle, C. F., & Yost, G. (1986). The geometry tutor. *Journal of Mathematical Behavior, 5,* 5–19.

Anderson, J. R., Corbett, A. T., Koedinger, K. R., & Pelletier, R. (1995). Cognitive tutors: Lessons learned. *Journal of the Learning Sciences, 4,* 167–207.

Bell, P. (1997). Using argument representations to make thinking visible. In R. Hall, N. Miyake, & N. Enyedy (Eds.), *Proceedings of CSCL '97: The Second International Conference on Computer Support for Collaborative Learning* (pp. 10–19). Toronto, Ontario, Canada: University of Toronto Press.

Bell, P. (1998). *Supporting conceptual change in the science classroom through technology-enhanced instruction involving evidence and argument.* Unpublished doctoral dissertation, University of California at Berkeley.

Bell, P., & Davis, E. A. (2000). Designing Mildred: Scaffolding students' reflection and argumentation using a cognitive software guide. In B. Fishman & S. O'Connor-Divelbiss (Eds.), *Proceedings of ICLS 2000: International Conference for the Learning Sciences* (pp. 142–149). Mahwah, NJ: Lawrence Erlbaum Associates, Inc.

Bell, P., Davis, E. A., & Linn, M. C. (1995). The knowledge integration environment: Theory and design. In J. L. Schnase & E. L. Cunnius (Eds.), *Proceedings of the CSCL Conference '95* (pp. 14–21). Mahwah, NJ: Lawrence Erlbaum Associates, Inc.

Bell, P., & Linn, M. C. (2000). Scientific arguments as learning artifacts: Designing for learning from the web with KIE. *International Journal of Science Education, 22,* 797–817.

Bellamy, R. K. E. (1996). Designing educational technology: Computer-mediated change. In B. A. Nardi (Ed.), *Context and consciousness: Activity theory and human–computer interaction* (pp. 123–146). Cambridge, MA: MIT Press.

Bielaczyc, K., & Collins, A. (1999). Learning communities in classrooms: A reconceptualization of educational practice. In C. M. Reigeluth (Ed.), *Instructional-design theories and models: A new paradigm of instructional theory* (pp. 269–292). Mahwah, NJ: Lawrence Erlbaum Associates, Inc.

Bielaczyc, K., Pirolli, P. L., & Brown, A. L. (1995). Training in self-explanation and self-regulation strategies: Investigating the effects of knowledge acquisition activities on problem solving. *Cognition and Instruction, 13,* 221–252.

Blumenfeld, P. C., Fishman, B., Krajcik, J., Marx, R. W., & Soloway, E. (2000). Creating usable innovations in systemic reform: Scaling up technology-embedded project-based science in urban schools. *Educational Psychologist, 35,* 149–164.

Blumenfeld, P. C., Soloway, E., Marx, R. W., Krajcik, J. S., Guzdial, M., & Palincsar, A. (1991). Motivating project-based learning: Sustaining the doing, supporting the learning. *Educational Psychologist, 26,* 369–398.

Bransford, J. D., Brown, A. L., & Cocking, R. R. (Eds.). (2000). *How people learn: Brain, mind, experience, and school* (expanded ed.). Washington, DC: National Academy Press.

Brown, A. L., & Campione, J. C. (1994). Guided discovery in a community of learners. In K. McGilly (Ed.), *Classroom lessons: Integrating cognitive theory and classroom practice* (pp. 229–270). Cambridge, MA: MIT Press/Bradford Books.

Bruner, J. (1996). *The culture of education.* Cambridge, MA: Harvard University Press.

Carroll, J. M., & Carrithers, C. (1984, August). Training wheels in a user interface. *Communications of the ACM, 27,* 800–806.

Center for Innovative Learning Technologies. (2002). *Design principles for educational software.* Retrieved April, 2002, from http://wise.berkeley.edu/design

Chase, W. G., & Simon, H. A. (1974). Perception in chess. *Cognitive Psychology, 4,* 55–81.

Chi, M., & Bassok, M. (1989). Learning from examples via self-explanations. In L. B. Resnick (Ed.), *Knowing, learning, and instruction: Essays in honor of Robert Glaser* (pp. 251–282). Hillsdale, NJ: Lawrence Erlbaum Associates, Inc.

Chi, M., Bassok, M., Lewis, M. W., Reimann, P., & Glaser, R. (1989). Self-explanations: How students study and use examples in learning to solve problems. *Cognitive Science, 13,* 145–182.

Chi, M., Feltovich, P., & Glaser, R. (1981). Categorization and representation of physics problems by experts and novices. *Cognitive Science, 5,* 121–152.

Clement, J. (1993). Using bridging analogies and anchoring intuitions to deal with students' preconceptions in physics. *Journal of Research in Science Teaching, 30,* 1241–1257.

Cohen, E. G. (1994). Restructuring the classroom: Conditions for productive small groups. *Review of Educational Research, 64,* 1–35.

Collins, A. (1996). Design issues for learning environments. In S. Vosniadou, E. D. Corte, R. Glaser, & H. Mandl (Eds.), *International perspectives on the design of technology-supported learning environments* (pp. 347–362). Mahwah, NJ: Lawrence Erlbaum Associates, Inc.

Collins, A., & Brown, J. S. (1988). The computer as a tool for learning through reflection. In H. Mandl & A. M. Lesgold (Eds.), *Learning issues for intelligent tutoring systems* (pp. 1–18). Chicago: Springer-Verlag.

Collins, A., Brown, J. S., & Newman, S. E. (1989a). Cognitive apprenticeship: Teaching the craft of reading, writing, and mathematics. In L. B. Resnick (Ed.), *Cognition and instruction: Issues and agendas* (pp. 453–494). Hillsdale, NJ: Lawrence Erlbaum Associates, Inc.

Collins, A., Brown, J. S., & Newman, S. E. (1989b). Cognitive apprenticeship: Teaching the crafts of reading, writing, and mathematics. In L. B. Resnick (Ed.), *Knowing, learning, and instruction: Essays in honor of Robert Glaser* (pp. 453–494). Hillsdale, NJ: Lawrence Erlbaum Associates, Inc.

Collins, A., & Ferguson, W. (1993). Epistemic forms and epistemic games: Structures and strategies to guide inquiry. *Educational Psychologist, 28,* 25–42.

Csikszentmihalyi, M. (1991). *Flow: The psychology of optimal experience.* New York: HarperPerennial.

Davis, E. A. (2003a). Prompting middle school science students for productive reflection: Generic and directed prompts. *The Journal of the Learning Sciences, 12,* 91–142.

Davis, E. A. (2003b). Untangling dimensions of students' beliefs about scientific knowledge and science learning. *International Journal of Science Education, 25,* 439–468.

Davis, E. A. (2004). Creating critique projects. In M. C. Linn, E. A. Davis, & P. Bell (Eds.), *Internet environments for science education* (pp. 89–113). Mahwah, NJ: Lawrence Erlbaum Associates, Inc.

Davis, E. A., & Bell, P. (2001, April). *Design principles for scaffolding students' reflection and argumentation in science.* Paper presented at the American Educational Research Association conference, Seattle, WA.

Davis, E. A., & Linn, M. C. (2000). Scaffolding students' knowledge integration: Prompts for reflection in KIE. *International Journal of Science Education, 22,* 819–837.

Davis, M., Hawley, P., McMullan, B., & Spilka, G. (1997). *Design as a catalyst for learning.* Alexandria, VA: Association for Supervision and Curriculum Development.

Dimitrov, D. M., McGee, S., & Howard, B. C. (2002). Changes in students' science ability produced by multimedia learning environments: Application of the linear logistic model for change. *School Science and Mathematics, 102,* 15–24.

Driver, R., Leach, J., Millar, R., & Scott, P. (1996). *Young people's images of science.* Buckingham, England: Open University Press.

Edelson, D. C. (2001). Learning-for-use: A framework for integrating content and process learning in the design of inquiry activities. *Journal of Research in Science Teaching, 38,* 355–385.

Edelson, D. C. (2002). Design research: What we learn when we engage in design. *The Journal of the Learning Sciences, 11,* 105–121.

Edelson, D. C., Gordin, D., & Pea, R. (1999). Addressing the challenges of inquiry-based learning through technology and curriculum design. *The Journal of the Learning Sciences, 8,* 391–450.

Favorin, M., & Kuutti, K. (1996). Supporting learning at work by making work activities visible through information technology. *Machine-Mediated Learning, 5,* 109–118.

Flower, L. S., & Hayes, J. R. (1980). The dynamics of composing: Making plans and juggling constraints. In L. W. Gregg & E. R. Steinberg (Eds.), *Cognitive processes in writing* (pp. 31–49). Hillsdale, NJ: Lawrence Erlbaum Associates, Inc.

Gallas, K. (1995). *Talking their way into science.* New York: Teachers College Press.

Golan, R., Kyza, E. A., Reiser, B. J., & Edelson, D. C. (2001, March). *Structuring the task of behavioral analysis with software scaffolds.* Paper presented at the annual meeting of the National Association of Research in Science Teaching, St. Louis, MO.

Greeno, J. G. (1989). Situations, mental models, and generative knowledge. In D. Klahr & K. Kotovsky (Eds.), *Complex information processing: The impact of Herbert A. Simon* (pp. 285–318). Hillsdale, NJ: Lawrence Erlbaum Associates, Inc.

Guzdial, M. (1994). Software-realized scaffolding to facilitate programming for science learning. *Interactive Learning Environments, 4,* 1–44.

Hancock, C., Kaput, J. J., & Goldsmith, L. T. (1992). Authentic inquiry with data: Critical barriers to classroom implementation. *Educational Psychologist, 27,* 337–364.

Hannafin, M. J., Land, S., & Oliver, K. (1999). Open learning environments: Foundations, methods, and models. In C. M. Reigeluth (Ed.), *Instructional design theories and models* (Vol. 2, pp. 115–140). Mahwah, NJ: Lawrence Erlbaum Associates, Inc.

Hoadley, C. M. (1999). *Scaffolding scientific discussion using socially relevant representations in networked multimedia.* Unpublished doctoral dissertation, University of California at Berkeley.

Holbrook, J., & Kolodner, J. L. (2000). Scaffolding the development of an inquiry-based (science) classroom. In B. Fishman & S. O'Connor-Divelbiss (Eds.), *Proceedings of ICLS 2000: International Conference of the Learning Sciences* (pp. 221–227). Mahwah, NJ: Lawrence Erlbaum Associates, Inc.

Hollan, J. D., Hutchins, E. L., & Weitzman, L. (1984). STEAMER: An interactive inspectable simulation-based training system. *The AI Magazine, 5*(2), 15–27.

Horwitz, P. (1996). Linking models to data: Hypermodels for science education. *The High School Journal, 79,* 148–156.

Jackiw, N. (1995). The Geometer's Sketchpad™ [Computer software]. Emeryville, CA: Key Curriculum Press.

Jackson, S. L., Stratford, S. J., Krajcik, J., & Soloway, E. (1994). Making dynamic modeling accessible to precollege science students. *Interactive Learning Environments, 4,* 233–257.

Jacobson, M. J., Sugimoto, A., & Archodidou, A. (1996). Evolution, hypermedia learning environments, and conceptual change: A preliminary report. In D. C. Edelson & E. A. Domeshek (Eds.), *Proceedings of ICLS 1996: International Conference on the Learning Sciences* (pp. 151–158). Charlottesville, VA: Association for the Advancement of Computing in Education.

Klahr, D. (2000). *Exploring science: The cognition and development of discovery processes.* Cambridge, MA: MIT Press.

Klahr, D., & Dunbar, K. (1988). Dual space search during scientific reasoning. *Cognitive Science, 12,* 1–48.

Knapp, L. K. (1994). *A task analysis approach to the visualization of geographic data.* Unpublished doctoral dissertation, University of Colorado at Boulder.

Knorr-Cetina, K. (1996). The care of the self and blind variation: An ethnography of the empirical in two sciences. In P. Galison & D. J. Stump (Eds.), *The disunity of science: Boundaries, contexts, and power* (pp. 287–310). Stanford: Stanford University Press.

Kolodner, J. L., Camp, P. J., Crismond, D., Fasse, B., Gray, J., Holbrook, J., et al. (2003). Problem-based learning meets case-based reasoning in the middle-school science classroom: Putting learning by design into practice. *The Journal of the Learning Sciences, 12,* 495–548.

Kolodner, J. L., Owensby, J. N., & Guzdial, M. (2004). Case-based learning aids. In D. H. Jonassen (Ed.), *Handbook of research on educational communications and technology* (2nd ed., pp. 829–862). Mahwah, NJ: Lawrence Erlbaum Associates, Inc.

Kozma, R. B., Russell, J., Jones, T., Marx, N., & Davis, J. (1996). The use of multiple, linked representations to facilitate science understanding. In S. Vosniadou, E. DeCorte, & H. Mandel (Eds.), *International perspective on the psychological foundations of technology-based learning environments* (pp. 41–60). Mahwah, NJ: Lawrence Erlbaum Associates, Inc.

Krajcik, J., Berger, C. F., & Czerniak, C. M. (2002). *Teaching science in elementary and middle school classrooms: A project-based approach* (2nd ed.). New York: McGraw-Hill.

Krajcik, J., Blumenfeld, P., Marx, R., Bass, K. M., Fredericks, J., & Soloway, E. (1998). Middle school students' initial attempts at inquiry in project-based science classrooms. *The Journal of the Learning Sciences, 7,* 313–350.

Kress, G., & van Leeuwen, T. (1996). *Reading images: The grammar of visual design.* London, England: Routledge.

Kuhn, D. (1993). Science as argument: Implications for teaching and learning scientific thinking. *Science Education, 77,* 319–337.

Lan, W. (1996). The effects of self-monitoring on students' course performance, use of learning strategies, attitude, self-judgment ability, and knowledge representation. *Journal of Experimental Education, 64,* 101–115.

Larkin, J. H., McDermott, J., Simon, D., & Simon, H. A. (1980). Models of competence in solving physics problems. *Cognitive Science, 4,* 317–345.

Latour, B. (1990). Drawing things together. In M. Lynch & S. Woolgar (Eds.), *Representation in scientific practice* (pp. 19–68). Cambridge, MA: MIT Press.

Lave, J., & Wenger, E. (1991). *Situated learning: Legitimate peripheral participation.* Cambridge, England: Cambridge University Press.

Lehrer, R., & Schauble, L. (Eds.). (2002). *Investigating real data in the classroom: Expanding children's understanding of math and science.* New York: Teachers College Press.

Lemke, J. L. (1990). *Talking science : Language, learning, and values.* Norwood, NJ: Ablex.

Lesgold, A., Lajoie, S., Bunzo, M., & Eggan, G. (1992). SHERLOCK: A coached practice environment for an electronics troubleshooting job. In J. H. Larkin & R. W. Chabay (Eds.), *Computer assisted instruction and intelligent tutoring systems: Shared goals and complementary approaches* (pp. 201–238). Hillsdale, NJ: Lawrence Erlbaum Associates, Inc.

Linn, M. C. (1995). Designing computer learning environments for engineering and computer science: The scaffolded knowledge integration framework. *Journal of Science Education and Technology, 4,* 103–126.

Linn, M. C. (2000). Designing the Knowledge Integration Environment. *International Journal of Science Education, 22,* 781–796.

Linn, M. C., Bell, P., & Davis, E. A. (2004). Specific design principles: Elaborating the scaffolded knowledge integration framework. In M. C. Linn, E. A. Davis, & P. Bell (Eds.), *Internet environments for science education* (pp. 315–339). Mahwah, NJ: Lawrence Erlbaum Associates, Inc.

Linn, M. C., Davis, E. A., & Eylon, B.-S. (2004). The scaffolded knowledge integration framework for instruction. In M. C. Linn, E. A. Davis, & P. Bell (Eds.), *Internet environments for science education* (pp. 47–72). Mahwah, NJ: Lawrence Erlbaum Associates, Inc.

Linn, M. C., & Hsi, S. (2000). *Computers, teachers, peers: Science learning partners.* Mahwah, NJ: Lawrence Erlbaum Associates, Inc.

Linn, M. C., & Slotta, J. D. (2000). WISE science. *Educational Leadership, 58*(2), 29–32.

Linn, M. C., & Songer, N. B. (1991). Teaching thermodynamics to middle school students: What are appropriate cognitive demands? *Journal of Research in Science Teaching, 28,* 885–918.

Loh, B., Radinsky, J., Russell, E., Gomez, L. M., Reiser, B. J., & Edelson, D. C. (1998). The Progress Portfolio: Designing reflective tools for a classroom context. In C. Karat, A. Lund, J. Coutaz, & J. Karat (Eds.), *Proceedings of CHI 98 Conference on Human Factors in Computing Systems* (pp. 627–634). Reading, MA: Addison-Wesley.

Loh, B., Reiser, B. J., Radinsky, J., Edelson, D. C., Gomez, L. M., & Marshall, S. (2001). Developing reflective inquiry practices: A case study of software, the teacher, and students. In K. Crowley, C. D. Schunn, & T. Okada (Eds.), *Designing for science: Implications from everyday, classroom, and professional settings* (pp. 279–323). Mahwah, NJ: Lawrence Erlbaum Associates, Inc.

Merrill, D. C., Reiser, B. J., Ranney, M., & Trafton, J. G. (1992). Effective tutoring techniques: A comparison of human tutors and intelligent tutoring systems. *The Journal of the Learning Sciences, 2,* 277–306.

Metcalf, S. J., Krajcik, J., & Soloway, E. (2000). Model-It: A design retrospective. In M. J. Jacobson & R. B. Kozma (Eds.), *Innovations in science and mathematics education* (pp. 77–115). Mahwah, NJ: Lawrence Erlbaum Associates, Inc.

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

Miyata, Y., & Norman, D. A. (1986). Psychological issues in support of multiple activities. In D. A. Norman & S. W. Draper (Eds.), *User centered system design* (pp. 265–284). Hillsdale, NJ: Lawrence Erlbaum Associates, Inc.

National Research Council. (1996). *National science education standards.* Washington, DC: Author.

Nemirovsky, R., Tierney, C., & Wright, T. (1998). Body motion and graphing. *Cognition and Instruction, 16,* 119–172.

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

Norman, D. A. (1991). Cognitive artifacts. In J. M. Carroll (Ed.), *Designing interaction: Psychology at the human–computer interface* (pp. 17–38). New York: Cambridge University Press.

Palincsar, A. S. (1998). Keeping the metaphor of scaffolding fresh—A response to C. Addison Stone's "The metaphor of scaffolding: Its utility for the field of learning disabilities." *Journal of Learning Disabilities, 31,* 370–373.

Palincsar, A. S., & Brown, A. L. (1984). Reciprocal teaching of comprehension-fostering and comprehension-monitoring activities. *Cognition and Instruction, 1,* 117–175.

Parr, C. S., Jones, T., & Songer, N. B. (2002). CyberTracker in BioKIDS: Customization of a PDA-based scientific data collection application for inquiry learning. In P. Bell, R. Stevens, & T. Satwicz (Eds.), *Keeping learning complex: The Proceedings of the Fifth International Conference of the Learning Sciences.*

Polman, J. L. (2000). *Designing project-based science: Connecting learners through guided inquiry.* New York: Teachers College Press.

Quintana, C. (2001). *Symphony: A case study for exploring and describing design methods and guidelines for learner-centered design.* Unpublished doctoral dissertation, University of Michigan, Ann Arbor.

Quintana, C., Eng, J., Carra, A., Wu, H., & Soloway, E. (1999). Symphony: A case study in extending learner-centered design through process-space analysis. *Proceedings of CHI 99 Conference on Human Factors in Computing Systems* (pp. 473–480). Reading, MA: Addison-Wesley.

Quintana, C., Krajcik, J., & Soloway, E. (2001). Exploring a description and methodology for learner-centered design. In W. Heineke & L. Blasi (Eds.), *Methods of evaluating educational technology* (Vol. 1, pp. 125–146). Greenwich, CT: Information Age Publishing.

Quintana, C., Krajcik, J., & Soloway, E. (2002, April). *Scaffolding design guidelines for learner-centered software environments.* Paper presented at the annual meeting of the American Educational Research Association, New Orleans, LA.

Quintana, C., Reiser, B., Davis, E. A., Krajcik, J., Golan, R., Kyza, E., et al. (2002). Evolving a scaffolding design framework for designing educational software. In P. Bell, R. Stevens, & T. Satwicz (Eds.), *Keeping learning complex: The Proceedings of the Fifth International Conference of the Learning Sciences.*

Quintana, C., Soloway, E., & Krajcik, J. (2003). Issues and approaches for developing learner-centered technology. In M. Zelkowitz (Ed.), *Advances in computers* (Vol. 57, pp. 271–321). New York: Academic.
Recker, M., & Pirolli, P. (1995). Modeling individual differences in students' learning strategies. *The Journal of the Learning Sciences, 4,* 1–38.
Reif, F., & Larkin, J. H. (1991). Cognition in scientific and everyday domains: Comparison and learning implications. *Journal of Research in Science Teaching, 28,* 733–760.
Reif, F., & Scott, L. A. (1999). Teaching scientific thinking skills: Students and computers coaching each other. *American Journal of Physics, 67,* 819–831.
Reiser, B. J., Tabak, I., Sandoval, W. A., Smith, B. K., Steinmuller, F., & Leone, A. J. (2001). BGuILE: Strategic and conceptual scaffolds for scientific inquiry in biology classrooms. In S. M. Carver & D. Klahr (Eds.), *Cognition and instruction: Twenty-five years of progress* (pp. 263–305). Mahwah, NJ: Lawrence Erlbaum Associates, Inc.
Roschelle, J., Kaput, J. J., & Stroup, W. (2000). SimCalc: Accelerating students' engagement with the mathematics of change. In M. J. Jacobson & R. Kozma (Eds.), *Innovations in science and mathematics education: Advanced designs for technologies of learning* (pp. 47–76). Mahwah, NJ: Lawrence Erlbaum Associates, Inc.
Rosson, M. B., Carroll, J. M., & Bellamy, R. K. E. (1990). Smalltalk scaffolding: A case study of minimalist instruction. In J. C. Chew & J. Whiteside (Eds.), *Proceedings of CHI 90 Conference on Human Factors in Computing Systems* (pp. 423–430). New York: ACM.
Ruopp, R., Gal, S., Drayton, B., & Pfister, M. (1993). *LabNet: Toward a community of practice.* Hillsdale, NJ: Lawrence Erlbaum Associates, Inc.
Salomon, G. (1996). Studying novel learning environments as patterns of change. In S. Vosniadou, E. DeCorte, R. Glaser, & H. Mandl (Eds.), *International perspectives on the design of technology-supported learning environments* (pp. 363–377). Mahwah, NJ: Lawrence Erlbaum Associates, Inc.
Sandoval, W. A. (2003). Students' understanding of causal explanation and natural selection in a technology-supported inquiry curriculum. *The Journal of the Learning Sciences, 12,* 5–51.
Sandoval, W. A., & Reiser, B. J. (2004). Explanation-driven inquiry: Integrating conceptual and epistemic supports for scientific inquiry. *Science Education, 88,* 345–372.
Scardamalia, M., & Bereiter, C. (1991). Higher levels of agency for children in knowledge building: A challenge for the design of new knowledge media. *The Journal of the Learning Sciences, 1,* 37–68.
Scardamalia, M., & Bereiter, C. (1992). An architecture for collaborative knowledge building. In E. De Corte, M. C. Linn, H. Mandl, & L. Verschaffel (Eds.), *Computer-based learning environments and problem solving* (pp. 41–66). Berlin: Springer-Verlag.
Schank, R. C., & Cleary, C. (1995). *Engines for education.* Hillsdale, NJ: Lawrence Erlbaum Associates, Inc.
Schauble, L., Glaser, R., Duschl, R., Schulze, S., & John, J. (1995). Students' understanding of the objectives and procedures of experimentation in the science classroom. *The Journal of the Learning Sciences, 4,* 131–166.
Schauble, L., Glaser, R., Raghavan, K., & Reiner, M. (1991). Causal models and experimentation strategies in scientific reasoning. *The Journal of the Learning Sciences, 1,* 201–238.
Schauble, L., Klopfer, L., & Raghavan, K. (1991). Students' transition from an engineering model to a science model of experimentation. *Journal of Research in Science Teaching, 28,* 859–882.
Schoenfeld, A. H. (1987). What's all the fuss about metacognition? In A. H. Schoenfeld (Ed.), *Cognitive science and mathematics education* (pp. 189–215). Hillsdale, NJ: Lawrence Erlbaum Associates, Inc.
Sherin, B. (2001). How students understand physics equations. *Cognition and Instruction, 19,* 479–541.
Shute, V., Glaser, R., & Raghavan, K. (1989). Inference and discovery in an exploratory laboratory. In P. L. Ackerman, R. J. Sternberg, & R. Glaser (Eds.), *Learning and individual differences: advances in theory and research* (pp. 279–326). New York: Freeman.
Simon, H. A. (1973). The structure of ill-structured problems. *Artificial Intelligence, 4,* 181–202.

Smith, B. K., & Reiser, B. J. (1998). National Geographic unplugged: Classroom-centered design of interactive nature films. In C. Karat, A. Lund, J. Coutaz, & J. Karat (Eds.), *Proceedings of CHI 98 Conference on Human Factors in Computing Systems* (pp. 424–431). Reading, MA: Addison-Wesley.

Smith, C., & Unger, C. (1997). What's in dots-per-box? Conceptual bootstrapping with stripped-down visual analogs. *The Journal of the Learning Sciences, 6,* 143–181.

Snir, J., Smith, C., & Grosslight, L. (1995). Conceptually enhanced simulations: A computer tool for science teaching. In D. N. Perkins, J. L. Schwartz, M. M. West, & M. S. Wiske (Eds.), *Software goes to school: Teaching for understanding with new technologies* (pp. 106–129). New York: Oxford University Press.

Soloway, E., Guzdial, M., & Hay, K. E. (1994). Learner-centered design: The challenge for HCI in the 21st century. *Interactions, 1,* 36–48.

Songer, N. B., & Linn, M. C. (1991). How do students' views of science influence knowledge integration? *Journal of Research in Science Teaching, 28,* 761–784.

Springmeyer, R. R., Blattner, M. M., & Max, N. L. (1992). A characterization of the scientific data analysis process. In A. E. Kaufman & G. M. Nielson (Eds.), *Proceedings of IEEE Visualization '92* (pp. 235–242). Los Alamitos, CA: IEEE Computer Society Press.

Stone, C. A. (1998). The metaphor of scaffolding: Its utility for the field of learning disabilities. *Journal of Learning Disabilities, 31,* 344–364.

Stratford, S. J., Krajcik, J., & Soloway, E. (1998). Secondary students' dynamic modeling processes: Analyzing, reasoning about, synthesizing, and testing models of stream ecosystems. *Journal of Science Education and Technology, 7,* 215–234.

Strike, K. A., & Posner, G. J. (1985). A conceptual change view of learning and understanding. In L. H. T. West & A. L. Pines (Eds.), *Cognitive structure and conceptual change* (pp. 211–231). Orlando, FL: Academic.

Tabak, I. (1999). *Unraveling the development of scientific literacy: Domain-specific inquiry support in a system of cognitive and social interactions.* Unpublished doctoral dissertation, Northwestern University, Evanston, IL.

Tabak, I., Smith, B. K., Sandoval, W. A., & Agganis, A. (1996, April). *BGuILE: Supporting inquiry in a learning environment for biology.* Paper presented at the 1996 annual meeting of the American Educational Research Association, New York.

Tabak, I., Smith, B. K., Sandoval, W. A., & Reiser, B. J. (1996). Combining general and domain-specific strategic support for biological inquiry. In C. Frasson, G. Gauthier, & A. Lesgold (Eds.), *Intelligent tutoring systems: Third International Conference, ITS '96* (pp. 288–296). Montreal, Quebec, Canada: Springer-Verlag.

Tien, L., Rickey, D., & Stacy, A. (1999). The MORE cycle: Guiding students' thinking in the laboratory. *Journal of College Science Teaching, 18*(5), 318–324.

Toth, E. E., Suthers, D. D., & Lesgold, A. M. (2002). "Mapping to know": The effects of representational guidance and reflective assessment on scientific inquiry. *Science Education, 86,* 264–286.

Toulmin, S. E. (1964). *The uses of argument.* Cambridge, England: Cambridge University Press. (Original work published 1958)

VanLehn, K. (1989). Problem solving and cognitive skill acquisition. In M. Posner (Ed.), *Foundations of cognitive science* (pp. 527–580). Cambridge, MA: MIT Press.

VanLehn, K., Jordan, P. W., Rose, C. P., Bhembe, D., Boettner, M., Gaydos, A., et al. (2002). The architecture of Why2-Atlas: A coach for qualitative physics essay writing. In S. A. Cerri, G. Gouardènes, & F. Paraguaçu (Eds.), *Intelligent Tutoring Systems: Proceedings of the 6th International Conference on Intelligent Tutoring Systems* (pp. 158–167). Berlin: Springer.

van Zee, E., & Minstrell, J. (1997). Reflective discourse: Developing shared understandings in a physics classroom. *International Journal of Science Education, 19,* 209–228.

Vygotsky, L. S. (1978). *Mind in society: The development of higher psychological processes.* Cambridge, MA: Harvard University Press.

Webb, N. M. (1983). Predicting learning from student interaction: Defining the interaction variables. *Educational Psychologist, 18,* 33–41.
White, B. (1984). Designing computer activities to help physics students understand Newton's laws of motion. *Cognition and Instruction, 1,* 69–108.
White, B., & Frederiksen, J. (1998). Inquiry, modeling, and metacognition: Making science accessible to all students. *Cognition and Instruction, 16,* 3–118.
Wood, D., Bruner, J., & Ross, G. (1976). The role of tutoring in problem solving. *Journal of Child Psychology and Psychiatry and Allied Disciplines, 17,* 89–100.
Wu, H., Krajcik, J., & Soloway, E. (2002). Promoting conceptual understanding of chemical representations: Students' use of a visualization tool in the classroom. *Journal of Research in Science Teaching, 38,* 821–842.