Interactive Model-Driven Case Adaptation for Instructional Software Design

Benjamin Bell **Smadar Kedar** **Ray Bareiss**
The Institute for the Learning Sciences
Northwestern University
Evanston, IL 60201
{bell, kedar, bareiss}@ils.nwu.edu

Abstract

Research in case-based design has demonstrated some capability to retrieve relevant designs and to adapt them automatically to satisfy new design constraints. However, some domains are less amenable to automated adaptation, particularly when the cases are very complex and when relationships among the design components are difficult to express formally. The design of interactive learning environments is one such domain. We describe a case-based approach to instructional software design which utilizes *interactive, model-driven case adaptation*. Our model for computer-based instruction is Goal-Based Scenarios. We describe a tool, Goal-Based Scenario Builder, which supports interactive adaptation of instructional software using the model, and illustrate its use in adapting an example case of a successful instructional software program, Sickle Cell Counselor.

Introduction

Case-Based Design (CBD) is an application of Case-Based Reasoning which seeks to support the design of new artifacts by recalling and adapting relevant previous designs. Perhaps the key research issue in CBD is the adaptation of an artifact to fit new, somewhat different design goals. This paper discusses CBD in the context of designing interactive learning environments, in particular, Goal-Based Scenarios. We are developing a design tool that supports the user in adapting an existing Goal-Based Scenario (GBS) in order to create a new one, through an incremental and interactive collaboration between the tool and an instructional designer.

We believe interactive case adaptation to be appropriate for domains in which cases are likely to be quite complex and design knowledge difficult to codify. This approach is well-suited for two reasons: First, it does not require a CBD system to possess deep, operationalized knowledge of design goals and of which specific design choices best satisfy them. Instead, such design choices are made by the user, aided by an example design augmented with design rationale in purely textual form. Second, the user is able to observe the adaptation process incrementally as each adaptation is applied. This enables a user to make each design decision in a logical sequence, observing its effects on the emerging artifact, its consistency with prior design choices, and how it constrains later decisions.

In the remainder of this paper, we first introduce an example GBS to serve as a base case. We then discuss the abstract instructional model underlying the case which helps align the case to a new situation and to focus the adaptation. Next, we describe a case-based design tool to help users create GBSs via interactive, model-driven case adaptation. We conclude with a discussion of the issues currently being addressed in our on-going research on interactive case adaptation.

Sickle Cell Counselor: a prototype Goal-Based Scenario

Goal-Based Scenarios is a framework for creating interactive learning environments (Schank, in press). The central tenet underlying the GBS framework is that skills are best learned when embedded within an engaging task, a position shared with other situated approaches to instructional technology (*e.g.* Bransford et al., 1990; Brown, Collins, & Duguid, 1989; CTGV, 1990). The GBS framework is described in detail in (Schank, Fano, Bell, & Jona, in press). In this section we consider which features of a GBS might play important roles in subsequent adaptation, by looking at a successful, implemented GBS.

SICKLE CELL COUNSELOR (Bell, Bareiss, & Beckwith, in press) is an exploratory hypermedia system designed for the Museum of Science and Industry in Chicago. This program provides the museum visitor with a basic understanding of genetics, and in particular, of Sickle Cell Disease, by allowing him or her to play the role of a genetic counselor, assisting clients who are considering having children, but are worried about the risk of Sickle Cell Disease. The four main activities a user can engage in are performing tests in a simulated blood lab, calculating the risks faced by the clients' offspring using an animated Punnett Square, asking experts for information, and advising the clients about the results.

Because our proposed method for interactive case adaptation relies heavily on the quality of the base case to produce a quality target case, it is important to address whether Sickle Cell Counselor (SCC) is an effective Goal-Based Scenario. We performed three studies to evaluate SCC. An in-museum study of usage patterns (Bell, Bareiss, & Beckwith, 1993) provided strong indications that the program is engaging and usable. Findings from a cognitive gain study (Bell, Bareiss, & Beckwith, 1993) suggest that

SCC does teach factual information about Sickle Cell effectively. A third study (Bell, Bareiss, & Beckwith, in press) examined the effects of situating learning within the counseling task. Results from these evaluations indicate that the counseling task promotes learning, not only of factual information, but also of the conditions under which that information is appropriate to apply.

An abstract model of GBS design

The general GBS model underlying SCC specifies a simulation-based, learn-by-doing program consisting of six phases. In the **Problem** phase, students are introduced to the role they will play in the simulation, and the task to carry out in that role. In Sickle Cell Counselor, for example, clients seeking advice about Sickle Cell Disease appear on video, and users are then told that they will be acting in the role of Sickle Cell Counselors. The **Do** phase is where the principal activities associated with the task take place, which in the case of SCC includes performing an electrophoresis test to establish the clients' genotypes, and calculating the probable genotypes of their offspring. The **Decide** and **Communicate** phases are where students makes the decision posed by the task and then convey that decision to agents in the program. These phases may co-occur, as in the case of SCC where users simultaneously decide what to tell the clients and communicate that decision to the clients. The **Critique** phase provides students with feedback regarding their decision, and suggestions about how to proceed. The clients' reaction to the user's advice in SCC serves both of these functions. The **Final** phase offers some closure to the interaction by demonstrating the effects of students' final decision; in the case in SCC, the clients return "a year later" to talk about what actions they took and the ultimate outcomes.

The model guides the design of these phases, beginning with definition of the student's task within each phase. This task is defined first for each phase, but does not, by itself, complete the instructional design. A complete design must also define the guidance and critiquing provided by the program, the general domain knowledge available to the student, and permissible paths of navigation within the task environment. The model we have adopted, then, identifies six phases of a GBS and four aspects of the design which must be instantiated to define each.

The **Do** phase defined by the model engages the student in identifying what information is needed to reach a decision, locating or extracting that information, and synthesizing a decision from the data that was obtained. Since this phase is the most specialized and hence, the least likely to apply to all Goal-Based Scenarios, we define the subclass of GBSs to which it does apply (including SCC) to be *Investigation* GBSs. An Investigation GBS is structured around a decision which the student is asked to make by performing an investigation. A "correct" decision must rely on the knowledge derived from the investigation, and although there may not be a single, correct decision, there should be a generally correct process for making the decision (cf. Padilla, 1991; Simmons, 1991).

An Example of Interactive Case Adaptation

While an instructional designer could, in theory, instantiate the abstract design model directly to create an investigation GBS, we believe this would be extremely difficult in contrast to adapting a specific instance of that model. On the other hand, if the designer is confronted simply with an artifact such as SCC, he or she would lack sufficient guidance as to which aspects of the program are likely to be relevant and adaptable, and which are not. Our approach, therefore, combines a specific example (*e.g.* SCC) with an abstract model of GBSs to guide the designer's adaptation process.

The prototype tool we have developed, called GBS Builder, guides the process of interactive case adaptation using such a predefined model associated with its base case. Adherence to the model ensures that the approach to instruction embodied in the base case will closely resemble those of the target cases (although target cases may differ substantially in terms of surface features). In this section, we illustrate how the tool guides the dialog between designer and model via interactive case adaptation. In the example which follows, GBS Builder helps a designer who would like to teach basic concepts in molecular chemistry by creating a simulated *crime lab*, where forensic scientists test samples for their composition in order to uncover evidence relevant to a criminal investigation. The designer sees descriptions of various GBSs in the case library, and finds that Sickle Cell Counselor is most similar to the structure of the GBS he wants to build (this retrieval step is currently manual).

Step 1: Executive Summary

The first step in the interactive adaptation is a conceptual design step, called the *executive summary*, in which the designer specifies top-level parameters including the name of the GBS, descriptions of the phases, names of agents in the scenario, etc. The purpose of the executive summary is to help the designer assess the applicability of the model underlying SCC to the new GBS and to serve as a first step in formalizing and organizing his or her design. Although the textual report generated during this step plays a non-operational role in the new case, it helps guide later case adaptation because parameters specified in the executive summary can act to constrain choices at more detailed levels in the design. The example below begins after the designer has named the target GBS "Crime Lab", and is then presented with the phases of the model. When the designer selects a phase, he or she is provided with its general definition and with a description of its implementation in SCC, and is asked to supply a corresponding textual description for the same phase in Crime Lab. For example, selecting the first phase of the model, **Problem**, gives the following:

```
GBS Builder: The Problem phase
presents the role the student will be
playing in the simulation, and presents
```

the set of problems the student will be asked to solve. In SCC, the student plays the role of a genetic counselor. What is the role of the student in Crime Lab?

Designer: Forensic scientist

GBS Builder: In SCC, the problem(s) the student needs to solve are: Advising couples about the risks of their offspring inheriting Sickle Cell Disease. What problem(s) does the student solve in Crime Lab?

Designer: Advising detectives about evidence collected at a crime scene, based on results of testing the composition of substances provided by the detectives.

Once the top-level executive summary is completed, the designer begins specifying a more detailed summary of each phase. For example, when selecting the **Do** phase, the designer is given the subphases of the **Do** phase, and asked to design each in turn in correspondence with SCC. The dialog continues:

GBS Builder: The **Do** phase consists of four subphases: OBTAIN DATA, ANALYZE, GET RESULTS, INTERPRET. In OBTAIN DATA, the student selects samples to analyze which would help solve the problem. In SCC, a blood sample is taken from each individual. Define the OBTAIN DATA subphase for Crime Lab:

Designer: Take blood, liver, or hair samples from crime scene.

GBS Builder: In ANALYZE, the student analyzes that data using various instruments. In SCC the tests in ANALYZE are: microscope, electrophoresis. What test(s) comprise ANALYZE in Crime Lab?

Designer: Gas Chromatograph, Mass Spectrometer

GBS Builder: In GET RESULTS, the student obtains and reads the results of the analysis. In SCC, the student finds the hemoglobin genotype of each clients. What results will the student obtain in Crime Lab?

Designer: levels of toxic compounds present in the victim's body.

GBS Builder: In INTERPRET, the student interprets the results as they pertain to a the solution to the problem. In SCC, the student uses a Punnett Square to predict the genetic outcomes. What is INTERPRET in Crime Lab?

Designer: Determine toxicity of compound, and whether this is consistent with evidence taken from the crime scene using an LD50 chart.

Step 2: Model-Based Adaptation

The next step in the interactive case-adaptation is to operationalize the design to create a running program, guided by the implementation of SCC and by its underlying model. We illustrate this step in defining an example subphase of the **Do** phase. Figure 1 shows the tool displaying the template (center) associated with an intermediate step in the OBTAIN SAMPLE subphase. In the dialog window (left), the tool lists the conceptual decisions the designer will make about this subphase, and then helps the designer implement this subphase as a sequence of graphical interface actions performed in the investigation (similar to the actions performed in the base GBS). Note that the corresponding screen in Sickle Cell Counselor is available by clicking the small screen icon in the lower right.

Here is a sample interaction:

GBS Builder: In the OBTAIN SAMPLE stage, the student:
(1) SELECTs the source of the sample,
(2) EXTRACTs the sample, and
(3) STOREs the sample for analysis.

«Designer switches to viewing SCC»

In SCC, the student (1) selects blood samples from each client, (2) extracts the sample by drawing blood using a syringe, and (3) places each blood sample in a test tube. Now define the screens and their components for each of the steps in the OBTAIN SAMPLE subphase of Crime Lab.

Assume the designer specifies that the student in Crime Lab will select liver tissue as the source of the sample. The following interaction defines the next step, EXTRACT:

GBS Builder: During this step, the student prepares the specimen for lab tests. Choose a picture which represents an activity for preparing the sample in Crime Lab.

Designer: (chooses a graphic showing liver sample in solution).

When the tool displays the graphical primitives (*e.g.*, button labels) and their associated responses (*e.g.*, button actions), the designer may supply desired values for each screen element. In the case of images and video, the designer may specify either the appropriate graphic or video, or a "placeholder" (a label which is inserted into the interface and which is collected by the program when generating a To-Do list of resources missing from the new GBS). Figure 2 shows the screen which implements this subphase of the **Do** phase, after the designer has operationalized a new set of design parameters.

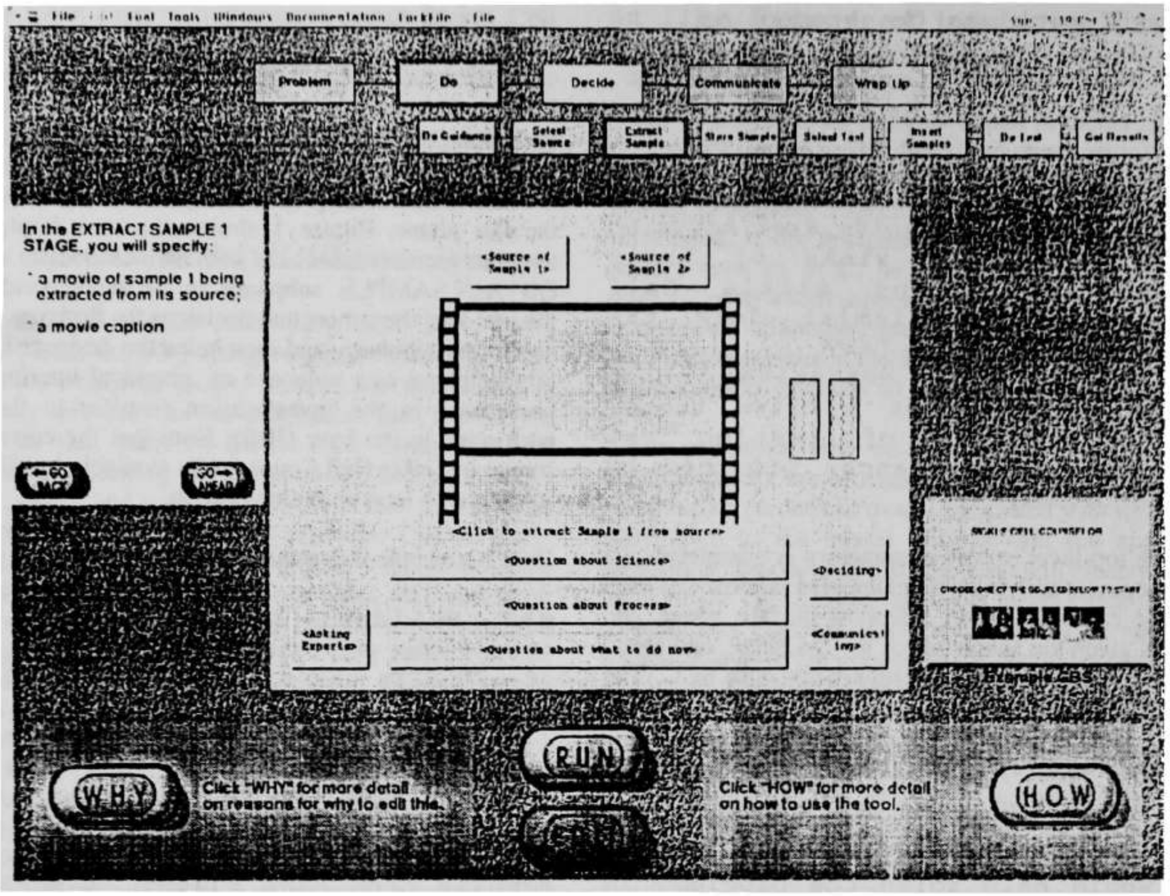

Figure 1: OBTAIN SAMPLE template before instantiation

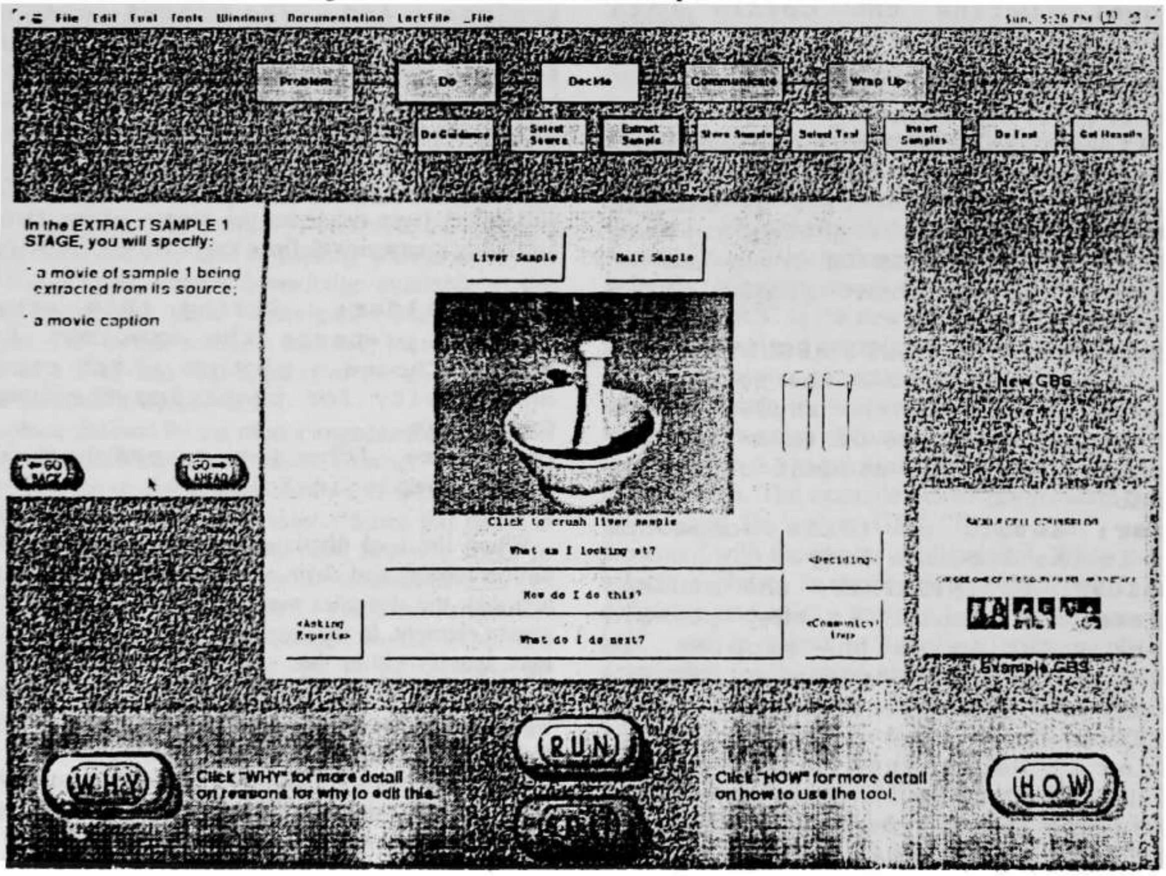

Figure 2: OBTAIN SAMPLE subphase in Crime Lab **Do** phase

Related Work

Case adaptation is defined (Kolodner, 1993) as altering a solution to an old case to fit a new case. Most research to date in the area of case adaptation has focused on methods for detecting failures in the old solution and repairing it, and on various domain-independent (*e.g.* Hinrichs & Kolodner, 1991) or domain-specific (*e.g.* Kass, 1990) repair strategies. Model-based adaptation has received some attention in the literature recently, *e.g.*, JULIANA (Shinn, 1988) for case-based reasoning using abstraction, CASEY (Koton, 1988) for case-based diagnosis, and KRITIK (Goel, 1991) for case-based design. For example, KRITIK designs new devices using past designs, guided by a model of the relationship between device structure and function. Adaptation is performed as design repair, isolating what makes the retrieved design inappropriate for the new device, and repairing it using both strategies and rich domain knowledge of the devices. Our approach differs from such knowledge-intensive approaches to adaptation in that it is the user, not the system, that determines how to adapt the solution. This obviates the need for fully representing domain knowledge which is an onerous task in a domain with design artifacts as complex as large software programs. Second, in our approach, the new case is not fully defined *a priori*, but is defined and designed incrementally during the interaction to fit the constraints of the model, so that there are no failures and repairs *per se*.

Little research on *interactive* case adaptation has been done. Some preliminary work has been described for dynamic modification of an airline schedule, which employs a case base of typical schedule failures and recoveries (Borse & Owens, 1992). This approach proposes that the user adapt suggested recoveries if they are not appropriate to the current failure. Similar work in the domain of radiation therapy allows the user to manually repair therapy plans, and then uses these repairs during replanning (Berger, 1993). The adaptation process itself can be interactive, with the system suggesting repairs and the users choosing among them or executing their own.

Ongoing Work and Conclusions

The approach we presented to Case-Based Design views case adaptation as an interactive dialog between the designer and a tool, with the process of adapting a base case mediated by an abstract model. We have argued, moreover, that such a dialog requires an abstract model to guide the adaptation. The particular model we presented supports interactive case adaptation of Investigation Goal-Based Scenarios, a specific class of interactive learning environments. The example interaction showed the current capabilities of the prototype tool we have developed.

Two open issues remain in the design and implementation of the GBS Builder. First, the tool needs to more fully support adaptation of the **Do** phase. The model specifies the various steps and substeps within this phase, but does not make explicit any rationale for organizing its elements (*e.g.* why blood samples are relevant, or what tests are needed to establish hemoglobin genotypes). We are working on characterizing a range of general problem-solving tasks and cataloging the functional artifacts associated with each one.

The second issue arises from our focus on case adaptation; we have yet to address case retrieval. Currently, the retrieval in GBS Builder is done manually. We see the need to interactively support this process, by identifying a set of features which are good predictors of case applicability, and by engaging a designer in a dialog to elicit the relevant features of the target case. Our ultimate goal is to create a tool which elicits a designer's objectives and then helps the designer to retrieve as well as adapt a similar, retrieved case. Our work on GBS Builder's interactive, model-driven case adaptation is a first step toward that goal.

Acknowledgments

We gratefully acknowledge the contribution of Roger Schank, who initially proposed the Goal Based Scenario Builder and who, along with Alex Kass and Chris Riesbeck, provided a wealth of helpful ideas. This project is a product of the contributions of: Erica Dubach, Steven Feist, Jarrett Knyal, Ann Kolb, Michael Korcuska, Ed Swiss, Barbara Thorne, and Mary Williamson. This research was funded, in part, by the Defense Advanced Research Projects Agency, monitored by the Office of Naval Research under contracts N00014-90-J-4117 and N00014-91-J4092. The Institute for the Learning Sciences was established in 1989 with the support of Andersen Consulting. The Institute receives additional support from Ameritech and North West Water, which are Institute Partners.

References

Bell, B.L., Bareiss, R., and Beckwith, R. (1993). The role of anchored instruction in the design of a hypermedia science museum exhibit. Paper presented at the annual meeting of the American Educational Research Association, Atlanta, GA.

Bell, B.L., Bareiss, R., and Beckwith, R. (in press). Sickle Cell Counselor: A Prototype Goal-Based Scenario for Instruction in a Museum Environment. *Journal of the Learning Sciences, 3*(4), forthcoming.

Berger, J. (1993). Knowledge Acquisition and Design Support in a Medical Domain. In *Proceedings of the AAAI-93 Workshop on Case-Based Reasoning.* Washington, D.C.

Borse, J.E. and Owens, C.C. (1992). Research in Progress on Knowledge-Intensive Methods for Irregular Operations Airline Scheduling. In *Proceeding of the AAAI-92 Spring Symposium*,. Stanford.

Bransford, J.D., Sherwood, R.D., Hasselbring, T.S., Kinzer, C.K., and Williams, S.M. (1990). Anchored instruction: why we need it and how technology can help. In R. Spiro and D. Nix (Eds.), *Cognition, Education, and Multimedia: Exploring Ideas in High Technology*. Lawrence Erlbaum: Hillsdale, N.J.

Brown, J.S., Collins, A., and Duguid, P. (1989). Situated cognition and the culture of learning. *Educational Researcher, 18*(1), 32–41.

Cognition and Technology Group at Vanderbilt (CTGV). (1990). Anchored Instruction and its relationship to Situated Cognition. *Educational Researcher, 19*(6), 2–10.

Goel, A. (1991). A Model-Based Approach to Case Adaptation. In *Proceedings of the Thirteenth Annual Conference of the Cognitive Science Society,* Chicago, IL.

Hammond, K.J. (1986). CHEF: A Model of Case-Based Planning. In *Proceedings of AAAI-86*, Philadelphia, PA.

Hinrichs, T.R. and Kolodner, J.L. (1991). The Role of Adaptation in Case-Based Design. In *Proceedings of the Case-Based Reasoning Workshop*, pp. 121-132, Washington, D.C.

Kass, A.M. (1989) *Developing Creative Hypotheses by Adapting Explanations.* PhD Thesis, Yale University. Issued as Technical Report #6, Institute for the Learning Sciences, Northwestern University, Evaston, IL.

Kolodner, J.L. (1993). Case-based reasoning. San Mateo, CA: Morgan Kaufmann.

Koton, P. (1988). Reasoning about Evidence in Causal Explanations. In *Proceedings of AAAI-88*, pp. 256 - 261, Minneapolis, MN.

Padilla, M.J. (1991). Science activities, process skills, and thinking. In S.M. Glynn, R.H. Yeany, and B.K. Britton (Eds.), *The Psychology of Learning Science.* Lawrence Erlbaum: Hillsdale, NJ.

Shinn, H.S. (1988). Abstractional Analogy: A model of analogical reasoning. In *Proceedings of the Case-Based Reasoning Workshop*, pp. 370–387, Clearwater Beach, FL.

Simmons, P.E. (1991). Learning Science in Software Microworlds. In S.M. Glynn, R.H. Yeany, and B.K. Britton (Eds.), *The Psychology of Learning Science.* Lawrence Erlbaum: Hillsdale, NJ.

Schank, R.C. (in press). Goal Based Scenarios. *The Journal of the Learning Sciences, 3*(4), forthcoming.

Schank, R.C., Fano, A., Bell, B.L. , and Jona, M.Y. (in press). The Design of Goal Based Scenarios. *The Journal of the Learning Sciences, 3*(4), forthcoming.