VALERIE J. SHUTE & LISA A. TORREANO

Chapter 6

FORMATIVE EVALUATION OF AN AUTOMATED KNOWLEDGE ELICITATION AND ORGANIZATION TOOL

Abstract. This chapter serves three purposes. First, we briefly review knowledge representations to stress the implications of different knowledge types on instruction and assessment. Second, we describe a novel cognitive tool, DNA (Decompose, Network, Assess), designed to aid knowledge elicitation and organization for instruction – specifically geared to increase the efficiency of creating the domain model used within intelligent instructional systems. Third, we present an exploratory test of the tool's efficacy. Three statistical experts used DNA to explicate their knowledge related to measures of central tendency in statistics. DNA was able to effectively elicit relevant information, commensurate with a benchmark system, generating a starting curriculum upon which to build instruction, and did so in hours compared to months for conventional elicitation procedures.

1. INTRODUCTION

The face of teaching and training is changing—from classroom-based, teacher-led instruction to electronic learning (e-learning) with a focus on individual or small groups of students and their knowledge and skill acquisition. Along the same lines, e-learning is shifting from developing infrastructures and delivering information online to improving learning and performance (see Shute & Towle, in press, for more on this topic). One large obstacle in this envisioned path concerns obtaining relevant content that will underlie these new student-based systems to support learning and performance. In the best case, relevant content is derived from the results of cognitive or behavioral task analyses. The downside of these approaches relates to their exorbitant price tag—i.e., a very high cost in terms of both time and money, with no guarantees as to effectiveness.

The aim of knowledge-elicitation tools (KETs), in general, is to increase the efficiency of collecting and using content; but current KETs are limited in utility, typically focusing on just one specific purpose (e.g., eliciting variables and their relations for a student model) and one type of knowledge (e.g., conceptual) (e.g., Chipman, Shalin, & Schraagen, 2000). Yet, education and training courses are substantially richer in scope. Furthermore, there is a wide range of purposes for e-learning systems. Where do we start?

Murray, Ainsworth, & Blessing (eds.), Authoring Tools for Adv. Tech. Learning Env.

Anyone who has attempted to design effective instruction or training knows that it begins with sound curriculum. In all cases, whether instructing karate beginners, nuclear physicists, network administrators, or aircraft mechanics, what information to include in the curriculum and how to ensure learners' mastery of the material must be determined. Good teachers and trainers make these determinations intuitively; the computer's insight, however, must be programmed. Therefore, resolving and specifying these "what to teach" and "how to teach" issues is critically important in forthcoming computer- or Internet-based instructional systems. New tools are needed to aid elicitation and organization of knowledge and skills for both assessment and instructional purposes. Specifically, KETs need to be designed to facilitate the development of future e-learning courses.

To render such instructional systems intelligent—or *responsibly adaptive*—three components must be specified: (a) a domain model, (b) a student model, and (c) an instructor model (e.g., Lajoie & Derry, 1993; Polson & Richardson, 1988; Shute & Psotka, 1996; Sleeman & Brown, 1982). The domain model represents the material to be instructed. This includes domain-related elements of knowledge, as well as the associated structure or interdependencies of those elements. In essence, the domain model is a knowledge map of what is to be taught. The student model represents the student's knowledge and progress in relation to the knowledge map. Finally, the instructor model, also known as the "tutor," manages the course of instructional material based on discrepancies between the student and domain models. Thus, the instructor model determines how to ensure learner mastery by monitoring the student model in relation to the domain model and addressing discrepancies in a principled manner. In short, these three models jointly specify "what to teach and how to teach it."

There are three main aims of this chapter, which was originally published in a special issue of International Journal of Artificial Intelligence in Education (1999). First, we briefly overview knowledge representations, focusing on those that can support student and domain modeling across different types of knowledge and skill. Specifically, we describe three categories of knowledge: (a) declarative (what), (b) procedural (how), and (c) conceptual (why). Our contention is that each knowledge type, best captured by different representations (i.e., knowledge maps), implies slightly different instructional and assessment techniques. For instance, assessing a person's factual knowledge of some topic requires a different approach than assessing how well someone can actually execute a procedure. By attending to knowledge type distinctions, and their representations, we hope to be better able to specify the component models of adaptive instructional systems for a broad range of content.

Second, we describe a novel cognitive tool that has been designed to aid elicitation and organization of knowledge for both assessment and instructional purposes. Specifically, it was originally designed to facilitate the development of intelligent tutoring system (ITS) curricula, while maintaining sensitivity to the knowledge type distinctions we discuss in the representation section of the paper. The same tool can be used in adaptive e-learning systems. Our primary aim for this tool, embodied in a program called DNA (Decompose, Network, Assess), is to increase the efficiency of developing the domain model—often referred to as the

backbone of intelligent instructional systems (Anderson, 1988) and sometimes called a "proficiency model" in assessment circles (e.g., Almond, Steinberg, & Mislevy, in press; Mislevy, Almond, Yan, & Steinberg, 2000; Mislevy, Steinberg, & Almond, 1999; Mislevy, Steinberg, Almond, Haertel, & Penuel, 2001). The tool attempts to automate portions of the cognitive task analysis process, often viewed as a bottleneck in system development. We will summarize its interface and functionality, but refer the reader to a more detailed description of the program (Shute, Torreano, & Willis, 2000).

The third and primary purpose of this paper is to present the results of an exploratory test of the tool's efficacy, or design feasibility. We outline the results from an empirical validation of the tool that examined how efficiently and effectively DNA works in the extraction of knowledge elements related to statistics. Specifically, we used DNA with three statistical "experts" to explicate their knowledge related to measures of central tendency. (Note: These were not technically "experts" but volunteers who were quite knowledgeable in the area of statistics, thus we use the term "experts" for economy).

1.1 Knowledge Representation

A variety of knowledge representation schemes have been developed that can be used to support student (and domain) modeling across diverse types of knowledge and skill (e.g., Merrill, 1994; 2000). For instance, Merrill (1994) presents four types of knowledge: facts, concepts, procedures, and principles. We simplify the issue by describing three broad categories of knowledge, conjoining Merrill's first two types into our single knowledge type: (a) declarative (what), (b) procedural (how), and (c) conceptual (why). Each has implications for instruction and assessment.

Declarative knowledge is factual information – propositions in the form of relations between two or more bits of knowledge that are either true or false. A formal distinction is often made between declarative knowledge that is autobiographical (episodic), and that representing general world (semantic) knowledge. Episodic knowledge entails information about specific experiences or episodes (e.g., *I inadvertently chewed a chili pepper hidden in my entrée - and it was hot! My mouth burned for twenty minutes and I was unable to taste the rest of my dinner*). Semantic knowledge (i.e., the meaning of information) is not tied to particular events, but rather entails information that is independent of when it is experienced, such as category membership and properties (e.g., habañero, tabasco, and jalapeño are kinds of chili peppers – habañero being one of the hottest). Episodic knowledge is thought to precede and underlie semantic knowledge. For example, after the experience of biting a habañero, one would likely be able to recognize novel examples of the pepper as being members of the same category – and of being hot.

Declarative (specifically semantic) knowledge can be functionally represented as a network of nodes and links, often called a semantic network (originally coined by Collins & Quillian, 1969). Alternatively, it may assume the form of a Bayesian inference network (e.g., Mislevy, et al., 2000). Although initially developed as an

efficient means of storing information in a computer, semantic networks have been shown to be cognitively plausible by studies that reveal that the hypothesized organization of the network structure is predictive of human performance on a variety of tasks. For example, response time to verify category and property statements (e.g., "A habañero is a chili pepper" or "Chili peppers contain capsaicinoids)" as well as to answer questions (e.g., "Is a habañero a pepper?") are predicted by features of the structure. Some of these features include the number of hierarchical levels to be crossed and whether stored features must be retrieved. Collins and Loftus (1975) proposed more general semantic network models along with the concept of spreading activation. These more general models do not strictly entail hierarchical relations.

For adaptive or intelligent instruction in declarative domains, semantic networks have been used as student models by instantiating the network with the knowledge to be taught, and then tagging nodes as to whether the student has learned it or not. These networks are an economical way to represent large amounts of interrelated information, are easily inspected, and support mixed-initiative dialogs between user and tutoring system. They are considerably less effective, however, for representing procedural information (i.e., knowledge or skill related to doing things).

Procedural knowledge is the knowledge of how to do something, and *procedural skill* is the demonstrable capability of doing so. For example, one may know how to remove the skin of a chili before cooking by roasting, but not do it very well. Or one may know how to preserve chilies, and also be able to do so quite well. In the former case (skinning), one may be said to have procedural knowledge but not procedural skill. In the latter case (preserving), one would have both procedural knowledge and skill. While there may be some cases where it is possible to have skill and not knowledge (or at least be unable to articulate that knowledge, such as when knowledge has become automated), more commonly having the skill logically entails having the knowledge.

Current theories of knowledge representation hold that procedural knowledge/skill can be functionally represented using a rule-based formalism, often called a production system (Anderson, 1993). These rules, or productions, consist of two parts – an action to be taken and the conditions under which to do so. An example might be, "if the goal is to alleviate a burning mouth that results from chewing a chili pepper, then drink milk." Thus, production systems combine step-by-step procedures (actions) with propositions (conditions), described previously as being represented by semantic networks. Production systems have been shown to be cognitively plausible by studies showing that the hypothesized structure of the rule-base is predictive of the kinds of errors people make in solving problems.

For intelligent instruction in procedural domains, production systems have been used as student models in several ways. One way is to instantiate an expert (production) system with the knowledge/skill to be taught, and then teach the knowledge/skill to the student, keeping track of what is and is not learned by tagging productions appropriately (e.g., Anderson, 1987). In another approach, expertise is modeled through negation by matching student errors to previously identified common patterns of errors that are associated with incorrect productions, or procedural "bugs" (e.g., VanLehn, 1990). Production systems are a fine-grained way

to represent procedural knowledge or skill, are easily implemented in most programming languages, and support a variety of straightforward ways to automate instruction because they directly represent the performance steps to be taught. They are sub-optimal, however, for representing declarative information. Additionally, the level of feedback that is most easily obtained may be too elemental for efficient instruction. Finally, the "bug library" approach to teaching procedural knowledge/skill is limited in that it is not possible to anticipate all possible procedural errors that students might manifest, and procedural bugs tend to be transient before disappearing altogether.

Conceptual Knowledge supports qualitative reasoning and constitutes a specialized category of knowledge not well handled by either semantic networks or production systems alone. Conceptual knowledge stems from the organization, or structure, of one's knowledge of a domain and the intuitive theory developed from what one has experienced in order to explain *why* things are as they are. For example, reasoning about principles of electricity, complex weather systems, or even why chili peppers are hot seems to involve internalized mental models that contain both declarative information (e.g., knowledge about electrical components) and procedural information (e.g., knowledge about how electrical systems behave). Conceptual knowledge allows humans to reason about how a system will behave under changing input conditions, either accurately or inaccurately. Regarding misconceptions, students who think that electricity flows through wires analogous to water flowing through pipes, will make predictable errors in reasoning about electricity. Conceptual knowledge also allows humans to generalize domain-specific knowledge and apply it in novel situations. In the words of Friedrich Nietzche, "*He who has a why can endure any how*."

Conceptual knowledge can be functionally represented by mental models, which are representations that support imagined states of affairs reflecting one's understanding of a domain. Pragmatic reasoning schemas, reflecting a generalized form of a specific rule, may also be used to represent conceptual knowledge. In general, conceptual knowledge is built on declarative and procedural knowledge, and thus can be partially represented by semantic networks in that certain cognitive processes considered "conceptual" in nature—such as similarity comparisons or generalization across domains—could be predicted by these formalisms. Thus, semantic networks account, in part, for conceptual knowledge by providing organization, or the structural glue, for category membership and property/feature information.

These networks primarily describe storage structure of knowledge units and predict patterns of retrieval of information. Mental models, in contrast, apply to semantic representations of complex scenarios allowing for reasoning about situations. Consequently, one's conceptual knowledge may be faulty either because it is built on unsound declarative or procedural knowledge or, when based on a sound foundation, because the intuitive theory is inaccurate. For example, if unaware of capsaicinoid compounds found in chilies, one may erroneously deduce from experience that color or size is the cause of a chili's heat. Indeed, this theory may prevail even with the knowledge that capsaicinoids are contained in chilies if it is not understood how they affect nerve pain receptors (i.e., they release a molecule

that sits on the pain fiber of the nerve, thus sending a message of pain to the brain). Having a rich mental model of chilies and their compounds' biological effects may lead to hypotheses about medicinal uses of peppers, such as treating chronic pain or mouth sores (i.e., understanding causes of pain may suggest ways to prevent or manage pain via related chemical processes).

A variety of reasoning studies support the cognitive plausibility of mental models by showing that mental model theory can predict the types of errors that people are likely to make and can explain individual differences in reasoning capacity in that better reasoners create more complete models (Johnson-Laird, 1983; Minstrell, 2000). For purposes of adaptive instruction, certain kinds of qualitative reasoning can be modeled by matching the student's beliefs and predictions to the beliefs and predictions associated with mental models that have been previously identified as characteristic of various levels of understanding or expertise. It is possible to infer what conceptualization the student currently holds, and contrive a way to show the student situations in which the model is wrong, thus pushing the student toward a more accurate conceptualization. This "progression of mental models" approach (White & Frederiksen, 1987) or "failure-driven learning environments" (Schank, 1999) teach reasoning skills that are ideal for remediating misconceptions, but cannot easily address other kinds of declarative knowledge or procedural knowledge/skill.

Our interest in knowledge representations is that we would like to outline the parameters for deriving, representing, and utilizing valid knowledge and skill elements for automated instructional and/or assessment systems. For example, in an adaptive e-learning system, the design of instruction may best be driven by a clear understanding of the representational nature of the knowledge or skill to be taught or assessed, subsequently tailored to address specific knowledge/skill deficiencies per learner. One key to optimizing the predictive utility of an assessment instrument is a careful mapping between the knowledge and skill tapped by the instrument and the knowledge and skill required in the classroom or on the job. The knowledge representation and student modeling techniques being developed by the intelligent tutoring, e-learning, psychometric, and assessment communities provide the basis of a formal system for accomplishing that mapping.

Assessment of declarative knowledge is fairly ubiquitous, particularly in classroom environments. Furthermore, the most common formats for such assessments include multiple-choice and fill in the blank items. For these item types, predictive validity is often limited (i.e., successful solution of these types of items does not guarantee successful performance on tasks that require procedural skill). With an understanding of the task requirements, in conjunction with the underlying knowledge representation, we believe probes can be designed to assess not only declarative knowledge, but also procedural knowledge/skills and conceptual understanding. The exception is certain procedural skills (especially those requiring specialized motor skills), which are more challenging to assess without technologies that provide psychomotor fidelity.

Presenting various scenarios may be used to assess a learner's misconception(s) of some phenomenon. For example, the computer could provide a series of questions concerning DC circuits. This would be in the form of: "What would

happen if …" questions (e.g., *If you measure the current in each of the branches of a parallel net and sum those measurements, would the total be higher, lower, or equal to the current in the entire net?*). Solutions to these types of items would provide information about the presence and nature of the current conceptualization (pun intended) of the domain.

The program focused on in this chapter was originally designed to operate with a particular student modeling approach to obtain and manage the critical knowledge required by an intelligent instructional system. That is, DNA (Shute, et al., 2000) is a knowledge elicitation and organization tool that was designed to operate with SMART (Student Modeling Approach for Responsive Tutoring; Shute, 1995), a student modeling paradigm based on a series of regression equations diagnosing mastery at the element level (i.e., particular knowledge or skill). Furthermore, SMART is an instructor modeling paradigm that determines a pathway of instruction based on mastery diagnosis. Thus, DNA relates to the "what" to teach, while SMART addresses the "when" and "how" to teach it. Both programs divide the universe of learning outcomes into three types: basic (or declarative), procedural, and conceptual.

In general, SMART engages in the following activities: (a) calculates probabilistic mastery levels via a set of regression equations, (b) evaluates what a learner knows in relation to individual bits of knowledge and skill (curriculum elements), (c) tailors instruction and assessment for the learner by combining both micro-and macro-adaptive modeling techniques (see Shute, 1995), and, (d) adapts to both domain-specific knowledge/skills as well as general aptitudes.

More specifically, SMART consists of curriculum elements (CEs—units of instruction and assessment) that represent the complete set of knowledge and skill elements comprising the curriculum. These are arranged in an inheritance hierarchy. Each new piece of instruction introduces the next set of CE(s), which in turn are assessed while students solve problems in the tutor. Each question within a problem set posed by the tutor is associated with a specific CE, so blame assignment (and consequent remediation) is precise and timely.

A value, which represents the learner's probable mastery of the CE (p[CE]), is maintained for each CE. The program allows for continuous representation of the learner's probable mastery values, employing regression equations to compute new p[CE] values. SMART's specific regression equations are shown, below. Each of the four equations is linked with the level of assistance required by the learner in the solution of a problem involving one of more of the CEs. In other words, the equation invoked is tied to the actual number of hints (i.e., level of feedback) provided by the system to the learner, from no feedback (level 0) to most explicit (level 3).

$$\hat{Y}_{(\text{Level 0})} = 0.3026 + 1.4377X - 0.7207X^2$$
$$\hat{Y}_{(\text{Level 1})} = 0.3316 + 0.2946X + 1.1543X^2 - 0.9507X^3$$
$$\hat{Y}_{(\text{Level 2})} = -0.0117 + 0.566X + 0.3518X^2$$
$$\hat{Y}_{(\text{Level 3})} = 0.0071 - 0.6001X + 2.5574X^2 - 1.4676X^3$$

SMART is initialized based on pretest performance data, where each pretest item is scored, in real-time, from 0 to 1, with partial credit given where appropriate (note:

the pretest contains items assessing all CEs, but adaptive testing is also possible). This provides the potential for pre-assessed abilities (per CE) which influences tutor delivery. A learner is thus placed in the curriculum (relative to the CE hierarchy) and presented with the CE(s) having a p[CE] value below some pre-established mastery criterion (e.g., < .70). The hints given are progressively more explicit (ranging from level 1, vague, to 3, specific). Moreover, the feedback is specific to the particular problem, and sensitive to the number of retries. It is provided in response to erroneous inputs, not explicitly requested by the student.

SMART has been incorporated into an experiential learning environment called Stat Lady (Shute & Gluck, 1994) and has undergone a series of controlled evaluation studies where the main components (e.g., diagnostic updating routines, and mastery and remediation control structures) have been systematically evaluated. Two studies have been completed and are discussed in more detail in Shute (1995). Results show dramatic (i.e., 2.2 standard deviation) learning gains in the normal Stat Lady environment, and even greater improvement with SMART active. The efficacy of the program's diagnostic capabilities was shown to be quite accurate, accounting for 54% of the unique outcome variance on the basis of just the computed student model values, and 67% of the outcome variance when aptitude and pretest data entered the equation.

We believe that the SMART approach to student modeling, used in conjunction with DNA to obtain the CEs for instruction and assessment, can provide for a diagnostically valid system that can assist with both micro- and macroadaptation decisions (i.e., what to teach, as well as when and how to teach it). We now present an overview of DNA.

1.2 General Description of DNA

DNA (Decompose, Network, Assess) is a novel cognitive tool designed to help expedite, without sacrificing accuracy, the cognitive task analysis (CTA) phase of developing adaptive assessment or tutoring systems. In addition, our goal is to create a tool that is broadly applicable across domains. That is, our goal is for the tool to be able to help map out constituent knowledge and skill elements for a variety of potential domains. Specifically, DNA should be equally capable of analyzing task performance (e.g., how to interpret radar signals), as well as domains more conceptual in nature (e.g., understanding the factors that influence stock market fluctuations). For a more detailed description of the program, see Shute, et al. (2000).

In short, DNA is intended to Decompose a domain, Network the knowledge into comprehensive structures, and employ other experts in the given domain to Assess the validity, completeness, and reliability of the knowledge representations. The program embodies a semi-structured series of questions aimed at extracting and organizing knowledge structures from experts. These questions, in general, map on to the three main types of knowledge that DNA attempts to elicit—basic (AKA declarative or symbolic), procedural, and conceptual knowledge. These knowledge types make DNA compatible with SMART, described earlier.

1.3 Modules of DNA

There are four "modules" comprising DNA. In addition to the expert-centered modules—Decompose, Network, and Assess—there is a Customize module that is used by the person requiring curriculum elements and structures for training or assessment purposes. Each will be discussed in turn.

Customize. The Customize module allows the instructional designer to provide information about the domain that is to be analyzed, characteristics of the intended learner population, as well as a list of the goals for the training session, assessment, or instructional course. Additionally, by adjusting "what, how, and why" gauges, the instructional designer indicates what is desired from the expert's decomposition of the domain in terms of the intended relative instructional emphasis or flavor for the curriculum. For instance, the instructional designer may want experts to focus primarily on providing procedural knowledge (75%) for some training regime, with less basic (20%) and conceptual (5%) knowledge delineation. Altogether, the instructional designer's input is intended to guide experts in their task of conveying knowledge so that it will be suitable for the instructor's purposes. After obtaining all of this information from the instructional designer, the Customize module generates a brief introductory letter addressed to prospective experts and a set of floppy diskettes that contain all the necessary program files to execute DNA (note: the next version of DNA is intended to reside online). This letter may be printed, as is, or edited within the preferred word processing software. The introductory letter and diskettes are forwarded to one or more experts who will use DNA to delineate the curriculum. See Figure 1 for an example letter generated by the Customize module that requests the delineation of measures of central tendency in the field of statistics.

Decompose. The Decompose module does the bulk of the work in eliciting the subject-matter expert's explicit domain knowledge. This module functions as an interactive, semi-structured interview that is similar to the "What, How, Why" questioning procedure that has been shown in the past to successfully elicit knowledge from experts (e.g., Gordon, Schmierer, & Gill, 1993; Hyperknowledge, at http://www.hyperknowledge.com). In particular, each of these general questions has been transformed into a path of interrogation. The "what" path elicits basic knowledge, the "how" path focuses on procedural knowledge, and the "why" path is aimed at obtaining conceptual knowledge. These paths result in three different interfaces that attempt to obtain information corresponding to the different representations discussed earlier.

The first screen the expert sees upon opening the Decompose module appears with the items ("ultimate goals") that were articulated in the Customize letter, restated as questions. These comprise the general learning objectives that will be further fleshed out during the decomposition process. The first two queries (see Figure 2, below) relate to basic knowledge and thus would invoke the "what" path, upon selection. The third query requests procedural knowledge and relates to the "how" path. And the last question seeks to obtain more conceptual information from the expert—bringing up the "why" path when it is selected.

Figure 1. An example letter generated by the Customize module requesting delineation of measures of central tendency.

Dear [Insert Expert's Name Here],

We're writing today to get your help in designing a course teaching *measures of central tendency*. Before you begin working with the enclosed program, please sit down and think about the critical things that help you understand various measures of central tendency.

As you go through the enclosed program and respond to our questions, try to respond in terms of how you currently think about the particular domain. Please don't respond with your original knowledge of measures of central tendency; you have probably developed more complex ways of thinking about the domain since then.

The ultimate goals of the course are for our students to:

Identify the main measures of central tendency
Specify relevant formulas
Know how to compute or derive each measure
Understand the functional relationship(s) between each measure and different underlying distributions

How specific should you get? You can presume that our students will have the following knowledge and skills:

Basic math abilities (including algebra skills)
Familiarity with PCs (e.g., Windows 98, 2000, XP environments)
Basic reading skills

Therefore, you will not need to define knowledge or skills at a detailed level in relation to these elements.

When answering questions during the program, please adjust your responses to fit the following guidelines:

What box: 55%
How box: 35%
Why box: 10%

Thanks very much for your time.

Sincerely,
[Signature]

Figure 1. An example letter generated by the Customize module requesting delineation of measures of central tendency.

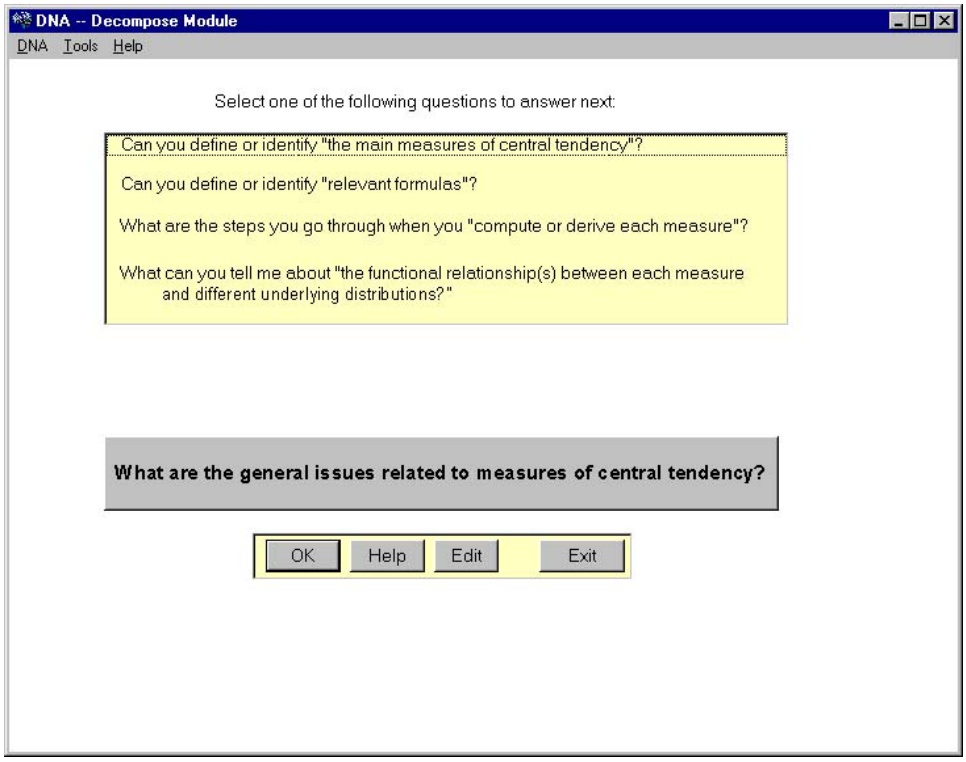

Figure 2. First screen of the Decompose Module

Given the domain of *measures of central tendency*, for example, suppose an expert proceeded down the basic knowledge (i.e., "what") path. The expert would select a question, then be guided through a series of questions that aim to elicit terms and definitions related to the curriculum element (CE) in focus. Further suppose that the expert began with the first question relating to defining the main measures of central tendency and elected to start with the Mean. At some point during the expert's definition of the Mean, the issue of distributions of data would arise, spawning a new line of questions (e.g., "Define or identify a normal distribution"). The expert can choose to decompose an area either breadth-first (e.g., specifying Mean, Median, and Mode as the three main types of central tendency). Alternatively, the expert may decompose in a depth-first manner—specifying the Mean, then within that context, discussing distributions, which may give rise to normal and skewed distributions, and so forth. In any case, responses are typed directly into a text box that can hold up to 16,000 characters. Multimedia files may be explicitly associated

(via the "Link This" button) with a curriculum element to further embellish it. See Figure 3, below, for an example of the kinds of things that may be associated with a particular CE as part of the "Link This" option.

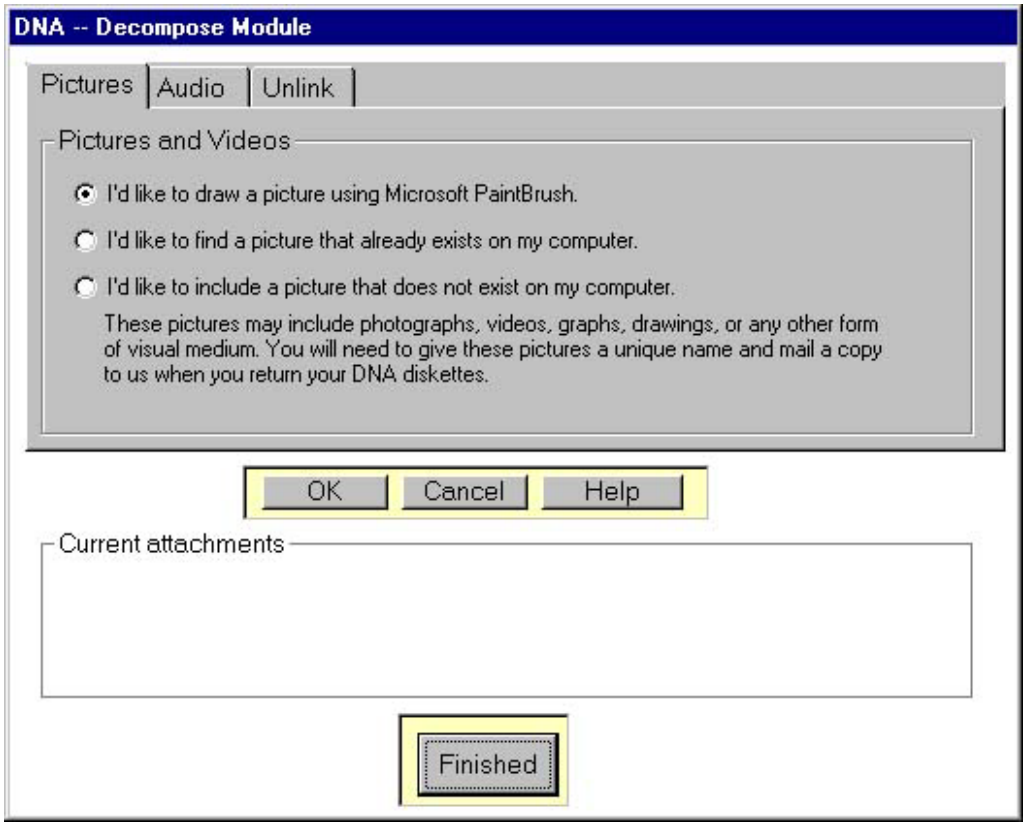

Figure 3. Screen from the "What" path of the Decompose Module in response to choosing the "Link This" option.

If a graphic were desired to supplement the definition, the expert would choose the "Link This" option, then either elect to draw a picture or associate the CE with an existing graphic. As shown in Figure 3, the expert chose to draw a picture.

Figure 4 shows an example of some output produced using the option to draw and label a "normal distribution." That file then becomes part of the particular CE record.

When decomposing procedural knowledge, the expert uses the "how" path, which presents a series of screens that allow the expert to construct procedures in the "step editor." In this process the expert delineates the steps of, and any conditional statements embodied within, the procedure. An expert's procedure for

finding the Median in a data set with an odd number of values might be represented as the following:

(1) Sort the data in the distribution
(2) Determine the midpoint: (N+1)/2
(3) Find the corresponding X value

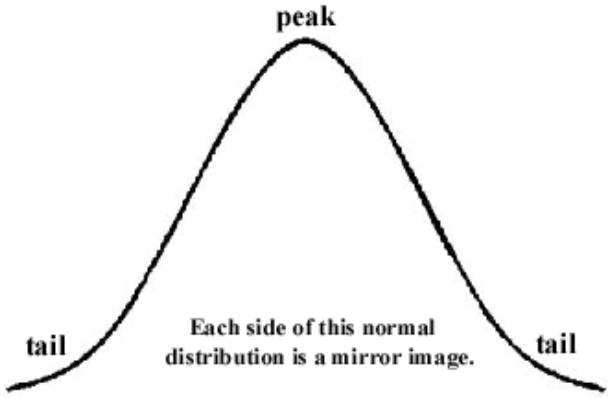

Figure 4. Example resulting from the option to draw and label a "normal distribution"

Any of the steps in a procedure could potentially be further decomposed into a sub-procedure. For instance, step 1 (sort the data in the distribution) may be broken into a sub-procedure detailing how to sort data in either ascending or descending order.

During the delineation of a procedure, the expert has a number of options to clarify and enrich the explanation of the task. The expert can add if-then statements, re-arrange steps, insert new ones, or delete any that are deemed unnecessary. In addition, and at any point, the expert may define terms that may otherwise be ambiguous to novices, thus providing additional basic knowledge. Figure 5 illustrates the step editor interface that shows one way an expert might summarize the steps underlying the computation of the Mean.

To decompose conceptual knowledge, the expert is guided through the "why" path, which is a series of questions that attempt to elicit as much information about complex concepts as possible. To illustrate using our example domain of measures of central tendency, suppose the expert chose to characterize the relationship between the Mean and its underlying distribution. The first question that DNA would present is: "What are the important issues that relate to the Mean and its underlying distribution?" This question is intended to obtain an initial listing of important elements associated with the Mean, such as: (a) "The Mean is affected by each value and its associated frequency within some distribution" and (b) "There are various types of distributions (e.g., normal, skewed, bimodal)."

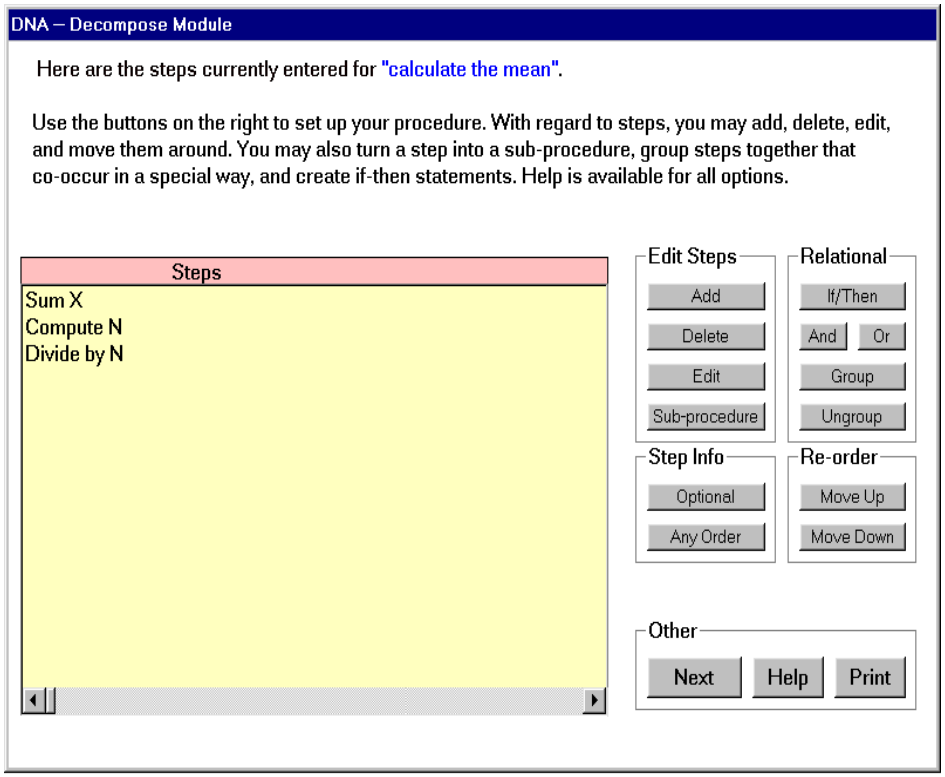

Figure 5. An example of the step-editor screen, from the procedural knowledge (PK) path, with the steps for calculating the Mean.

The second question in this path of inquiry is: "How are these elements functionally related?" This question is designed to elicit conceptual knowledge concerning how the important components (cited in the previous response) function together. A representative answer to this query might be, "The Mean is located in the center of a normal distribution, along with the Median and Mode. However, because the Mean is the only measure of central tendency affected by extreme scores, it will shift away from the center of skewed distributions and fall closer to the tail—where the extreme scores reside—than either the Median or Mode. For bimodal distributions, the Mean is located between the two humps."

The third question of the conceptual path asks: "Why is knowing about the relationship between the Mean and its underlying distribution important in understanding measures of central tendency?" This question attempts to link the current element being decomposed (Mean and its underlying distribution) to the general topic of instruction (measures of central tendency). Each of these questions, residing along this path, aims to provide a database of rich conceptual knowledge. A reasonable response would be: "Knowing the type of distribution of some data can

influence your decision as to which measure of central tendency to use. For a skewed distribution (e.g., salaries in a small business where most are in the lower range and one or two in the very high region), the Mean would not be as good a summary of the central tendency of these data as the Median. Rather, it would be artificially inflated."

Finally, the expert is asked to describe typical and atypical situations in which knowing or understanding the relationship(s) between the Mean and different underlying distributions, is useful. An exemplar response would be: "A typical situation related to understanding the relationship between the Mean and various distributions is if you need to determine which measure of central tendency you should use to summarize some data. An atypical situation involving use of this knowledge would be if you wanted to purposefully distort a conclusion. For instance, if you wanted to impress some friends about the average salary of the small business (described above), you could report the Mean, knowing that the more typical salary was far less."

A particular path (what, how, why) is completed when its series of questions has been answered and the expert clicks the "Finished" button to indicate that no elaboration or additional elements warrant explanation at that point. How does the expert know when the domain is finally decomposed? The instructional designer specified the "ultimate learning goals" of the curriculum in the letter generated by the Customize module. This indicates the starting point for the expert's decomposition of the domain. The stopping point is also indicated in the letter by the statements of knowledge and skill that the learner population is presumed to possess. For instance, in the letter shown in Figure 1, learners are presumed to have basic math skills, thus the expert need not decompose the curriculum below that point. That is, if the expert delineated the procedure of computing the Mean (sum all numbers and divide by the sample size), no additional steps would have to be decomposed relating to the arithmetic operations embodied by those steps. In addition, the stopping point occurs when the expert believes that sufficient information has been specified for each of the ultimate learning goals indicated in the Customize letter.

All information given by experts is stored in a MS Access database record of CEs. These CEs serve as the guidelines for developing assessment or instruction in the domain. Multiple fields are listed with each CE record, e.g., name, number, description, relationships to other CEs, learning objective tapped, format, and so on. By storing this type of information in each CE record, it was hoped that restructuring DNA's output into teachable curriculum units would be more easily accomplished compared to traditional cognitive task analysis interview methods. Figure 6 illustrates DNA's object model.

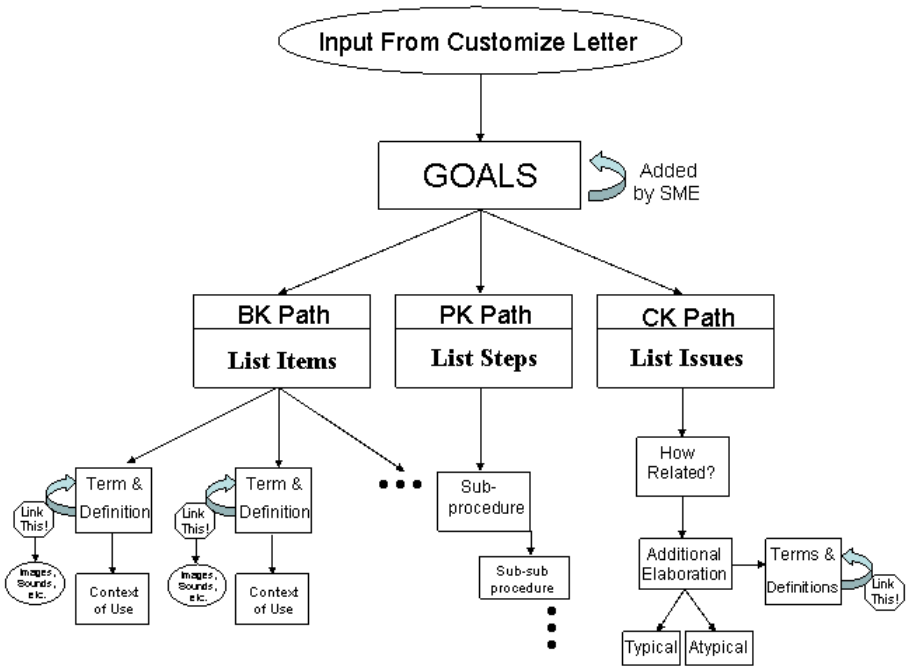

Figure 6. Object model representing DNA's Decompose Module.

Network. This module is currently under redevelopment. Ultimately, it is intended to transform CEs elicited during the Decompose module into graphical nodes that experts spatially arrange and link to form knowledge hierarchies, conceptual graphs, or production rules. Each node assumes the name of the CE and its contents that were defined during the Decompose module. To simplify viewing, only main-level CEs and their first-level "children" nodes appear upon the initial screen. "Pregnant" CEs are those that have elements embedded within them, such as sub-procedures within procedures. They appear in bold font. Any pregnant element can be unpacked to reveal its components by right clicking on the node and choosing the option "unpack."

To compose a meaningful hierarchy, nodes and their links are designed to differ along certain dimensions. Node shapes indicate the various knowledge types used by the SMART framework—rectangles reflect basic knowledge, ovals are procedural elements, and rounded rectangles denote conceptual knowledge. Links differ along three dimensions: type, direction, and strength of association. Some links are already in place when the subject matter expert (SME) arrives at the Network module. These come from information provided during the Decompose module (e.g., IF-THEN relationships from the step editor window). Other links must be created and labeled by the SME.

The first kind of link relationship is "type." This denotes the specific kind of relationship(s) between nodes (e.g., is a, causes, fixed serial order, child of). DNA's link types can relate to both semantic and procedural knowledge elements. Semantic links enable the SME to specify the relationships among curriculum elements, allowing for the conceptual structure of the domain to be specified. Procedural links enable the SME to specify the relationships among procedural steps and sub-steps, similar to a production-system representation. In addition to the semantic and procedural links available, there is a user-definable link that allows the SME to type in a label for a relationship not already defined.

The second link-label option is "directionality." This refers to the flow of control or causation between curriculum elements. Three options exist for this: uni-directional, bi-directional, and no direction. These relationships are established via arrowheads that are attached to the end of a line. For instance, the formula for the Mean ($\Sigma X/N$) could have arrows emanating from it to the individual elements comprising the formula (i.e., to each of Σ, X, and N and with a "parent of" label assigned to each node).

Finally, links can differ in terms of the "strength" of association. There are three values for this trait: weak, moderate, and strong. This indicates the degree to which the items are related. The information on strength is accomplished by varying the width of the link line (fine, medium, and bold).

This module is intended to be functionally similar to conceptual graph analysis (e.g., Gordon, et al., 1993) except that with DNA experts generate the conceptual graphs instead of the instructional designers. We believe that the use of a graphical representation will make relationships among knowledge units salient, which could also highlight missing knowledge components. Thus, we speculate that the network module of DNA will enable experts to recognize gaps in the knowledge and skills they provided in their decomposition. Moreover, they have a chance to readily correct inadequacies as they can return to the Decompose module and update the curriculum element record by adding or editing information.

After experts complete the Network module, data are stored on floppy diskettes and returned to the instructional designer who reviews the curriculum element record and conceptual graphs for any glaring omissions in content. If any omissions are present, the instructional designer can ask the expert to expand the inadequate curriculum elements to encompass, in full, the intended scope of instruction.

Assess. The final module, which is actually more a process, will be used to validate the CE records and conceptual graphs generated by experts. This will be accomplished by having other experts in the domain review the data and conceptual graphs generated by the first expert or group of experts. That is, multiple experts will be employed to review and edit one another's conceptual graphs as a method of validating externalized knowledge structures.

Before describing the results from the preliminary examination of DNA's efficacy, we first present DNA in the context of (a) the Field of AI and education, in general; followed by (b) its relation to other ITS authoring systems, specifically.

2. DNA'S NICHE IN THE FIELD OF AI AND EDUCATION

We are creating DNA in response to the charge that conventional cognitive task analysis (CTA) methods are often inefficient, laborious to translate into usable instruction, limited to procedural tasks, and difficult to use (see Shraagen, et al., 1997). These shortcomings identify areas to be improved in the intelligent tutoring, or adaptive e-learning system arenas. To help alleviate the impediment of the CTA process, we are attempting to address some of these limitations in the design of DNA.

Efficiency. Traditional CTA methods typically involve extensive interviews with experts, transcription of ensuing protocols, and organization of knowledge and skill units. This process normally requires many months of work and many person-hours to achieve. These traditional methods often employ knowledge engineers to interview and observe experts, others to transcribe sessions, and cognitive psychologists to summarize the data into a hierarchical representation. In contrast, DNA attempts to streamline the bulk of the interview, transcription, and organization process which is intended to significantly decrease both time and personnel resources required for the cognitive task analysis.

Instructional design framework. A common limitation of traditional CTA methods is that it is often difficult to translate the pages of interview transcriptions and conceptual graphs into a usable curriculum. DNA is designed such that its output is compatible with an existing instructional system framework (i.e., SMART; Shute, 1995), which should further enable efficient, adaptive systems development. In short, both the goal and the format of instruction are considered in the information sought. DNA's database record of CEs contains rich and useful information at the individual CE level – the unit base for structuring usable curriculum. CEs are classified according to knowledge types that are compatible with the SMART framework (i.e., basic, procedural, or conceptual knowledge). In addition, each CE includes a unique number, detailed description, and hierarchical information relating it to other CEs in the knowledge structure of the domain. The hierarchical structure represents dependency relations among knowledge elements that inform curriculum design.

In the near future, DNA will include expert-supplied embellishments such as: typical points of difficulty (impasses) in understanding the domain, good examples and counter-examples of concepts and procedures, along with more specific questions assessing conceptual and/or functional understanding of the particular domain or individual CE. All of this information is well suited for subsequently developing principled instruction.

An additional improvement to DNA, already begun, is to align it with the evidence-centered design (ECD) approach (e.g., Mislevy, et al., 1999). This is intended to enhance its capability to aid in the design of valid assessments. ECD involves: (1) defining the general proficiencies and particular claims to be made about the students (i.e., the knowledge, skills, and abilities to be measured), (2) establishing and delineating relevant evidence, per claim (i.e., student performance data demonstrating varying levels of mastery), and (3) determining the nature and format of tasks that will generate or elicit that evidence. Evidence, in the center, ties

the tasks directly back to the underlying claims and proficiencies. As part of the original program, DNA already elicits the first part of the ECD requirements (i.e., the claims). We plan to expand it to also elicit associated evidences, per CE. Incorporating a content model will ensure that the associated instruction/assessment is tied to the desired proficiencies. Conforming to current industry standards (e.g., IMS, SCORM) will allow these diagnostic assessments and instructional units to be recycled in many different learning environments.

Broad applicability. Another common limitation of traditional CTA methods is that many are only applicable to procedural domains. DNA's specific purpose is to support intelligent tutoring system or adaptive e-learning development, applicable across a range of domain topics, both procedural and conceptual in nature. It attempts to achieve this broad applicability by eliciting CEs ranging in knowledge types compatible with the SMART framework. In addition, this applicability is to be achieved via its underlying hybrid representational structure of knowledge and skill elements, functionally a cross between a semantic net and production system (see Shute, et al., 2000).

User-friendly. As indicated previously, traditional CTA methods often rely on several individuals trained in knowledge elicitation techniques. In contrast, DNA was designed to be usable by those without CTA expertise. The interface offers context-sensitive examples and the interview questions were written at a fifth-grade reading level. Thus any instructional designer who wants to develop a curriculum will be able to use this tool, with a variety of experts, to elicit knowledge.

We now examine the question of how DNA fits in the field of ITS authoring systems.

DNA and Other ITS Authoring Systems

Although DNA was designed with the goal of facilitating ITS development, it is not an ITS authoring tool, per sé. As mentioned, DNA serves the development of intelligent instructional systems by attempting to streamline the process of interviewing subject matter experts (i.e., individuals with expertise about the domain to be instructed). This interview process is laborious but necessary, as the author of a course (or the instructional designer) does not always have the expertise of the domain to be taught, nor does the SME necessarily have the inclination or ability to instruct. Foremost a cognitive task analysis tool, DNA uses a semi-structured automated dialogue to elicit and structure the elements of knowledge of a domain from the SME that will be used by the instructional designer to create instruction. More precisely, rather than helping to "author" or output instruction, DNA helps to produce the input of the "authoring." Thus, relative to other authoring tools, DNA is decidedly limited in scope to producing the elements that go into the content or domain model – it does not produce complete tutors.

As is the trend for authoring tools, DNA divorces the knowledge base from the instructional strategy (Murray, 1998). Adherence to this divorce is intrinsic in the fact that DNA's scope is limited to the content or domain model. In addition, although DNA was designed with SMART in mind, it is not committed to any

particular instructional strategy. In fact, the output of DNA would require a bit of transformation before being able to be used by SMART.

First, by producing a representation of the knowledge (not the instruction) that underpins a domain, DNA provides for designing at the pedagogical level, not the media level. The content or domain model in SMART, however, encapsulates the actual material to be presented to the learner during an instructional session. Thus, there would be some transition work (wording, arranging the graphics, etc.) to take DNA's output to the point of being a full domain model on which SMART can operate. The degree of effort required to transform DNA's output into an instructional format to be presented to the learner will vary, depending on the SME's direct responses and the goals of the tutor.

Second, while DNA's output identifies curriculum elements—the crux of the content—and implicitly represents the relationship among CEs (and will do so explicitly when the Network module is re-coded), this representation does not strictly specify an order of instruction. Rather, the output provides a map of the underlying knowledge domain, but the sequencing of instruction is left to the instructional designer's choice of instructional strategies. For example, the relationship between a procedure, its sub-procedures, and the conceptual knowledge supporting decisions and reasons for the process would be mapped out in the output. From that representation, the instructional designer could choose to take a part or whole task approach, or provide conceptual support for the procedure before or after requiring practice. Thus, additional specifications of sequencing DNA's output would be required before taking shape as a domain model for SMART, or other systems.

The tradeoff of DNA's output not being plug 'n play with SMART is that DNA's output is not restricted to use with SMART; it is not committed to a particular ordering of instruction, nor is it committed to a particular instructional theory. Thus, with a narrow focus towards the content or domain model, DNA's applicability is widened both in terms of content addressed and instructional strategies incorporated. Theoretically, with the aforementioned transitions, DNA's output should be usable with any number of intelligent instructional systems—whether the pedagogy is "learn by reading and thinking" or "learn by doing" simulation-basedsystems – a distinction made by Murray (1998).

Finally, given that DNA seeks out and identifies multiple knowledge and skills types (fact, process, concept), its output can be used with systems that are predicated on the belief that different knowledge/skill types should be instructed in different manners (e.g., Gagné, 1985; Merrill, 1994). Differentiation of knowledge types and their corresponding components, intra-relations, and inter-relations is done with DNA. However, facilitating the production in these types of tutors, the rules specifying feedback, hints, explanations, content as presented to learner, sequencing of instruction, and so forth is not done by DNA.

In general, our goal for DNA is that of decreasing the overall effort for making intelligent instructional or assessment systems by supporting the acquisition and organization of the knowledge base of the domain to be assessed and/or taught. Although DNA is an incomplete authoring tool prototype, initial evaluation supports the feasibility of the approach, as will be addressed in the next section. Further

research remains in order to determine the full effort required to transition the material DNA produces to a domain model functioning in a tutor. By focusing on the cognitive task analyses phase of development and not specifying the pedagogical model, authors require instructional design skill, as they are not guided in how to finalize the domain model to fit whichever instructional strategy they would like to follow. The person actually using the system, however, needs no special training or knowledge base, and can use the DNA system with a minimal learning curve.

We now turn our attention to a formative investigation testing the Decompose module of the DNA system. In general, the different evaluation issues relate to the efficiency of the system and the validity of the output.

3. PRELIMINARY DNA EVALUATION

3.1 Design

DNA promises a great deal in its potential to uncork the cognitive task analysis (CTA) bottleneck. However, because DNA has been designed to be broadly applicable across domains, it is an open-ended and flexible system. The downside of this design feature is that the system may sometimes fail to keep SMEs grounded in their explication of domain expertise. Therefore, before relative benefits of DNA can be assessed, the more fundamental issue of whether DNA's general design is functionally feasible must be determined. As a stand-alone program and with only minimal direction to the SME via an introductory letter, can DNA actually extract *any* knowledge that can serve as the basis for curriculum development?

In order to address this basic feasibility question, we tested the degree to which our SMEs' data agreed with a benchmark representation of a topic. Williams (1993) conducted a similar analysis using a production system representing cutting and pasting text with a word processor. We extended this evaluation technique beyond its previous use with a simple procedural task by using it with a more complex domain containing a variety of knowledge types (i.e., measures of central tendency). Thus, we used the curriculum from an existing tutor, the second descriptive statistics module (DS-2) from Stat Lady (Shute, Gawlick, & Lefort, 1996) that focuses on the topic "measures of central tendency," as the benchmark. The curriculum for this module of Stat Lady was derived from a traditional cognitive task analysis involving document analysis as well as interviews with two SMEs. Although no formal records were kept regarding development time, we estimated that the CEs in the Stat Lady curriculum required approximately five months to obtain, structure, and outline.

Using a domain that has already been decomposed and transposed into an effective curriculum provides a way to gauge DNA's potential efficiency and validity. The Stat Lady curriculum provides a benchmark as to (a) the time and cost of eliciting the curriculum elements of the domain (i.e., an efficiency measure), and (b) the qualitative characteristics of curriculum elements of the domain (i.e., exemplar elements that constitute a valid, effective curriculum).

The degree to which the knowledge elements derived by DNA from experts map onto the elements of the Stat Lady curriculum, already embodied in an existing tutor, will shed light on the potential effectiveness of DNA's output. If the obtained output is close to the "idealized" extant domain structure of a tutor that has already been shown to improve learning, we can infer that the output is valid–or that it could be the basis for developing an effective curriculum.

3.2 Participants

Three volunteer subject-matter experts participated in this preliminary study. While none were formally "statisticians," all had graduate degrees in psychology and a minimum of 10 years experience conducting statistical analyses. Further, all reported that they were quite familiar with the measures of central tendency. None had prior, formal interactions with Stat Lady.

To assess incoming levels of expertise, the SMEs completed a computer-based test of measures of central tendency that is typically used in conjunction with Stat Lady. The test assessed knowledge and skills related to all CEs contained within the Stat Lady curriculum (i.e., a total of 127 CEs). While no time limits were imposed, our experts required between 1-1.5 hours to complete the test. Scores ranged from 71.3% to 87.5% ($\underline{M}$ = 79.2, $\underline{SD}$ = 8). Following the test, each expert completed the Decompose portion of DNA.

Before the experts' sessions with the program, the authors of this chapter completed the Customize module of DNA to produce a letter, similar to the one shown in Figure 1, informing the experts of the curriculum goals for some hypothetical students to achieve. In addition, this letter informed the SMEs of the intended learner population's expected skills and abilities. This provided the SMEs with parameters for their decomposition of the domain. Experts interacted with DNA in individual sessions, during which at least one of the authors was present to answer only general questions.

3.3 Benchmark

The Stat Lady DS-2 (Shute, et al., 1996) database consists of 127 curriculum elements. However, of those, only a subset of 78 CEs served as the benchmark against the output of DNA's Decompose module. This benchmark was used as the basis for assessing completeness and validity of the SMEs output. Some Stat Lady CEs were not included in the benchmark because they were deemed as not applicable, for a variety of reasons, to our current purpose. For instance, most of the first 37 CEs of the tutor constitute a stand-alone review module extracted from the first descriptive statistics module of Stat Lady. The review module included CEs related to organizing data (e.g., sorting data, identifying the minimum or maximum value, etc.) and manipulating frequency distribution tables. Since most of these items were not our experts' focus, all but 7 of these CEs were excluded from analysis. The CEs from the review module that *were* judged as relevant to the experts' task, and therefore kept in the benchmark, include knowing: (a) definitions

for distribution, frequency distribution, and variable, (b) notations for variable, frequency, and sample size, and (c) the steps needed to create a frequency distribution.

Five additional CEs from the benchmark were removed because they were deemed as somewhat idiosyncratic to the Stat Lady tutor. That is, three conceptual knowledge (CK) elements were eliminated since they related to the instruction of measures of central tendency via analogy to a seesaw (not a standard practice, but helpful to learners, nonetheless). The remaining two CEs included basic knowledge (BK) elements that were concerned with identifying tutor-specific notation for the Median (i.e., Mdn) and the Mode (i.e., Mo). It is unrealistic to expect "experts" to outline these curriculum elements, given that these particular abbreviations for Median and Mode were specific to the Stat Lady curriculum, and not standard in Statistics instruction. Finally, an additional 14 Stat Lady CEs were excluded due to a subtle difference between procedural knowledge (PK) and procedural skill (PS). That is, Stat Lady CEs are coded either as BK, CK or PS elements depending upon how they are instructed and assessed. For instance, if a learner's knowledge of how to calculate the Mean was assessed by identifying steps of the procedure from a multiple-choice list, the element would be coded BK (which includes "knowledge of rules" or PK). In contrast, if the learner's knowledge was assessed by having them actually calculate the Mean of a set of data, the element would be coded PS. In short, coding of knowledge is context sensitive. In DNA, however, when decomposing their knowledge, experts describe procedures of a domain (i.e., knowledge of procedures – PK); they do not perform their "procedural skill" of these elements. Because elements coded as PS in the tutor context track the learner's ability (or success) in doing various tasks or procedures, they are not appropriate to serve as a comparison benchmark. Therefore all of Stat Lady's PS elements were excluded from the benchmark. Some of these PS elements that were removed include computing: the sum of values, N, cross products, midpoint, Mean, Median, and Mode (and doing so in a variety of contexts).

In general, Stat Lady's "measures of central tendency" curriculum concentrates on basic, procedural, and conceptual knowledge relating to the Mean, Median, and Mode. Basic elements include definitions, formulas, and notations for each measure and their components (e.g., sample size $N = \sum f$; Mean = $\sum X/N$; cross product = Xf). Procedural elements describe the steps of how to calculate each measure of central tendency from both data sets and frequency distribution tables. In addition, alternative methods for these calculations are detailed, where appropriate. For example, the curriculum includes differences in the procedure for calculating the Mean when all frequencies equal one ($f = 1$) and for when they do not (i.e., some $f > 1$). Conceptual elements emphasize understanding which central tendency measures are appropriate within different circumstances, and why.

For sufficient instruction of the domain, additional BK, PK, and CK elements cover various types of distributions (e.g., normal, flat, symmetric, bimodal, platykurtic, leptokurtic, mesokurtic, postively and negatively skewed), as well as issues of symmetry, kurtosis and skewness. To bolster concept integration, many elements highlight the relationship between the three measures of central tendency

and the underlying distribution. Specifically, they instruct and assess on each measure's location, and relationship(s) to one another, within different types of distributions. The sum of this information supports understanding of the guidelines for using each of the three measures.

In total, 78 CEs from the Stat Lady DS-2 curriculum remained as the final benchmark for analysis. The distribution of knowledge types in the benchmark was as follows: 74% BK elements, 18% PK elements, and 8% CK elements. This was not substantially different from the distribution of the original 127 CEs for the entire tutor (79% basic, 13% procedural, and 8% conceptual elements).

4. RESULTS AND DISCUSSION

The output from DNA exists in two forms: (a) a Microsoft Access database of CEs and (b) a graphical array of the hierarchical knowledge structure (future design). The focus of this DNA assessment was on the Decompose module therefore the CE databases were analyzed in this formative evaluation.

The analysis involved assessing the content of each SME's database relative to the benchmark described above. For each CE we assigned either a "1" to indicate that the SME included it in the decomposition, or a "0" to denote its absence. In some instances, we assigned partial credit if we judged that a portion of a CE was decomposed (e.g., .67 if 2 out of 3 steps of a procedure were listed). There were a couple of cases where a SME delineated a CE that was not present in the benchmark listing. Those instances were noted, but not included in the current analysis. For example, one expert delineated and defined "data" (i.e., *a set of observations about the world; within statistics, data commonly refers to a set of numbers that are collected or observed*). This CE was not in the original Stat Lady database as it was presumed to be part of incoming knowledge.

How well do the experts capture the benchmark curriculum? Our three SMEs' output captured 25%, 49% and 23% of the Stat Lady benchmark database. Furthermore, each required 285, 170, and 100 minutes to complete DNA, respectively. One expert (SME-2) was clearly more in line with Stat Lady than the others, producing the array of CEs most consistent with the benchmark in less than 3 hours of decomposition time.

When developing a curriculum for a domain, an instructional designer aggregates information from several sources. Likewise, we combined the outputs produced by all three experts, however we did not have to deal with the issue of potentially contradictory data from multiple SMEs in this case. This issue of aggregating data across SMEs, consistent and otherwise, will be examined further in an upcoming study by the first author. Specifically, the utility of a statistical approach called combinatorial data analysis (Hubert, Arabie, & Meulman, 2001) will be examined as a possible solution to combining potentially disparate data.

Table 1 presents the comparison between (a) the CEs elicited by DNA from our three SMEs combined, and (b) the CEs that compose the benchmark. The data in the table show the total count of CEs, overall and by knowledge type. Results showed that the distribution of knowledge types derived by DNA in our combined SME data

(71% BK, 23% PK, and 6% CK) is similar to the distribution seen in the benchmark data (74% BK, 18% PK, and 8% CK). This seems to suggest that DNA addresses the different knowledge types adequately.

	Total	*BK*	*PK*	*CK*
Combined SMEs' CE output	48	34	11	3
Stat Lady Benchmark CEs	78	58	14	6

Table 1. Comparison of curriculum elements (CEs) elicited from our experts by DNA to those of the Stat Lady benchmark, overall and by knowledge type (basic, procedural, and conceptual).

With regard to the SMEs' collective capture of the benchmark, results show that 62% (i.e., 48/78) of the Stat Lady CEs were delineated by at least one of our three experts. For this domain, DNA was relatively more successful at eliciting a match of the benchmark's *procedural* knowledge, capturing 11/14 (79%) of the benchmark, than at eliciting *basic* 34/58 (59%) or *conceptual* knowledge 3/6 (50%).

Which elements were extracted and which were not? Some benchmark CEs were reported by all of our experts, some by only a subset of the SMEs, while other elements were omitted completely. In the following paragraphs, we discuss the nature of the CEs produced by the decomposition and those omitted.

Results indicated that nine (i.e., 12%) benchmark CEs were outlined by all three experts (5 BK and 4 PK). These included definitions of the Mean, Median, and Mode. To illustrate, one of the SMEs outlined the definition of the Mean shown in Figure 7; the other definitions were comparable. Other CEs that were reported by all experts included the basic steps required to determine the values of each measure of central tendency. For instance, each expert delineated the steps to (a) calculate the Mean when $f = 1$, (b) determine the Median when N is odd and when it is even, and (c) identify the Mode. See Figure 5 in the General Description of DNA section for a SME's outline of the procedure to calculate the Mean. Finally, all SMEs conveyed that in a normal distribution, the three measures of central tendency have the same value.

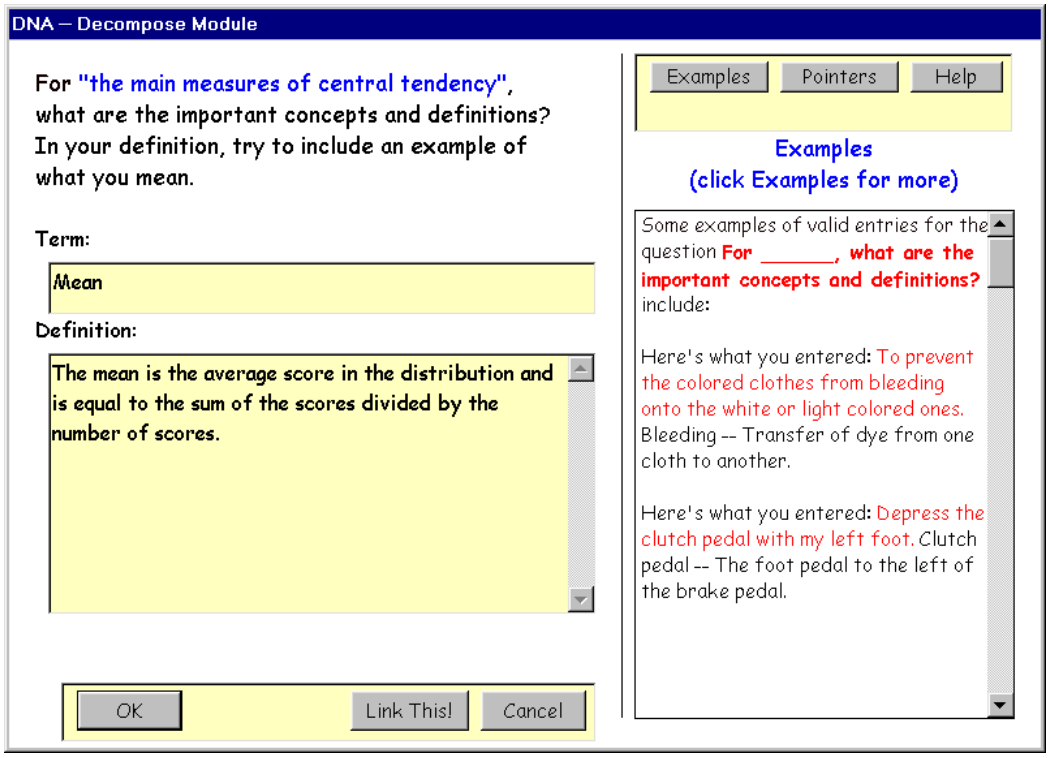

Figure 7. SME's output defining the Mean in the basic knowledge (BK) path.

Next, 39 (i.e., 50%) benchmark CEs were reported by a subset of our experts: 29 BK, 7 PK, and 3 CK (Note: SME-2, who individually matched the benchmark 49%, provided the bulk of these elements, while the other two experts contributed only 8 additional unique CEs). Some of these elements included: definitions of normal distribution, tail, and skewness, notations for sample size (N) and variable (X), and the formula for computing cross products ($X * f$). Other examples of elements reported by a subset of experts included the guidelines for using different measures of central tendency and the relationship among them within a skewed distribution (e.g., the Mode is used with categorical data; the Median is better for representing quantitative data within a skewed distribution). A number of CEs were reported that relate to distributions (e.g., normal, positively and negatively skewed) and their particular relationship(s) with the measures of central tendency. For example, the functional relationship of each of the three measures within a normal distribution was described, and the Mean was further discussed within skewed distributions. Figure 8 shows an excerpt of one SME's response to DNA's conceptual path query

regarding the important aspects of the relationship between a measure of central tendency and its underlying distribution.

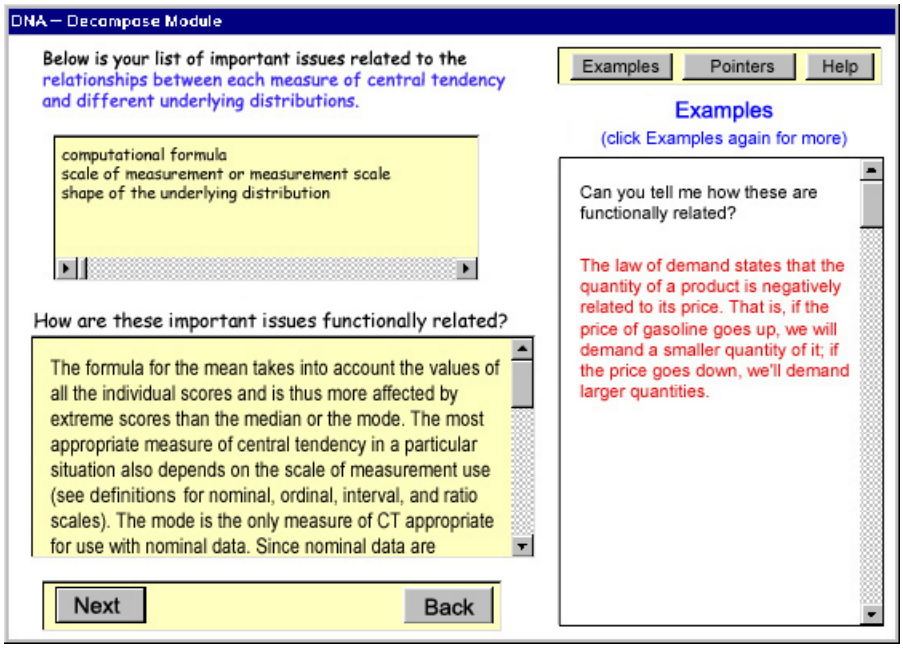

Figure 8. A response along the conceptual knowledge (CK) path representing an excerpt of a SME's output on the functional relationship between measures of central tendency and underlying distribution types. The vertical scrollbar indicates additional text.

Finally, there were 30 (i.e., 38%) benchmark elements omitted by all three experts (24 BK, 3 PK, 3 CK). Some of these omissions included low-level information that related to formulas (e.g., the sum of cross products, $\Sigma[X * f]$), definitions (e.g., variable), and notations (e.g., *f* for frequency). Thus, either (a) the stopping point for the decomposition of knowledge was not communicated well by the Customize letter, or (b) what was outlined as the goal of the tutor did not match exactly with the benchmark domain model. Additional omissions included the specification of alternative formulations (e.g., computing the Mean when $f > 1$, and use of midpoint formula as a step to determine the Median). Other elements not reported by any of our SMEs were some conceptually complex ones. For instance, no expert described the Median and Mode in relation to skewed or flat distributions. Finally, the last set of elements omitted were those fairly peripheral to the curriculum emphasized by the Customize module letter. Some of these included defining platykurtic, mesokurtic, and leptokurtic distributions.

In sum, the agreement between the aggregate and benchmark data showed that DNA was able to elicit 62% of the CEs present in an existing database, and was able to do so in a reasonable amount of time (i.e., approximately 9 hours, the total time

required by all 3 experts). In relation to simple counts of CE types, our SMEs produced more BK elements than either PK or CK ones. But in relation to the benchmark, our SMEs' output matched a greater percentage of PK elements than either BK or CK elements.

Limitations

How do we interpret these data? Why did we not see 100% overlap? We suspect there is an interaction of at least three factors contributing to this "less than perfect" capturing of the benchmark: (1) idiosyncrasy of the Stat Lady curriculum, (2) issues related to our specific "experts," and (3) inadequacies within the DNA program.

First, with regard to the Stat Lady curriculum, the elements selected for inclusion represent "measures of central tendency" as culled from extensive document analysis on the topic, as well as interviews with two subject-matter experts. Some of the items our current (DNA using) experts omitted included definitions of leptokurtic, platykurtic, and mesokurtic distributions, possibly deeming these items as not central to (or too esoteric for) the scope or goals of the decomposition. Further, other elements that were not articulated by our experts included very low-level CEs such as defining the summation notation (Σ). Thus, the Stat Lady curriculum contained many items that our experts may have considered to be only tangentially related to the decomposition task.

The second factor we suspect contributed to our failure to get perfect overlap involved the nature of our experts. As mentioned earlier, we solicited local participants who are not formally statisticians, but rather experimental psychologists who are familiar with statistics. Their knowledge structure of the field, while no doubt solid, may not have reflected the knowledge structures of true statisticians. This was further indicated by their test data; we suspect that statisticians would have scored in the 90^{th} percentile on that test. Recall that the mean pretest score from our group was 79%. Thus, it appears that the experts participating in the current evaluation had areas of deficient knowledge.

The third factor contributing to our obtained degree of overlap relates to possible shortcomings of the DNA program itself. The data from the present study made apparent several places where DNA could be enhanced. First, the data showed that our experts provided fewer CK elements than PK or BK elements in relation to the benchmark. This finding could be an indicator that the conceptual path in DNA was simply not effective in eliciting CK structures. We are currently addressing this problem by adding some follow-up questions at the end of important junctures within DNA. For instance, at the end of the entire Decompose module, the expert will be asked a series of thought-provoking questions designed to capture an overview of the domain/field (e.g., themes and principles). We believe that this information will further aid the instructional designer in generating curriculum and provide more illuminating conceptual knowledge to the curriculum. Some example global follow-up questions include: (a) What are some difficult areas you've encountered in the acquisition of [domain being decomposed]? (b) What has worked

for you in surmounting these obstacles? (c) Can you describe a good analogy that can help learners understand aspects of the domain?

In addition, we are planning to elicit more CK information in conjunction with explicated procedures. That is, at the conclusion of each "how path," the experts will be specifically probed to flesh out the procedures in terms of their underlying rationales. Thus, besides obtaining a listing of steps comprising some procedure, we also want to elicit the reasons *why* they chose to do it that particular way.

Another important revision to the Decompose module was motivated by our findings. That is, on occasion, our experts would input procedural specifications that were ambiguous (e.g., Do A or B and C). DNA is now becoming sensitive and responsive to instances of ambiguity. As a result, the new version of DNA will require the expert to specify "groupings" to render any potentially ambiguous procedure or statement more precise (e.g., Do (A or B) and C). Furthermore, DNA will also request that the expert think about alternative methods to accomplish the same goal. For example, if the expert has specified some conditional statements in a procedure, the Decompose module will probe for additional, logical antecedents and consequences (e.g., When A does not hold, should one still do B? Are there other conditions that can trigger B?).

In summary, these data provide preliminary information about the efficacy of DNA as a knowledge elicitation tool. That is, given limited direction via one introductory letter of expectations for the decomposition of the domain and minimal guidance in use of the DNA program, experts appear to be able to use the tool efficiently to explicate components of their knowledge structures. Moreover, the obtained data are, for the most part, consistent with an existing curriculum. Thus we are gaining confidence that our tool has potential value as an aid to ITS and adaptive e-learning system development. Rather than being discouraged that our overlap was "only 62%", we are encouraged that the results suggest the basic design of DNA is feasible.

5. SUMMARY AND CONCLUSIONS

This paper describes an ongoing effort to develop DNA, a knowledge-elicitation tool to be used by subject-matter experts across a variety of domains. We also describe an exploratory test of the effectiveness and efficiency of the program. Preliminary results show that DNA can produce reasonably valid and reliable data within an acceptable amount of time. This has a direct implication for streamlining the intelligent assessment and instructional system development process, often viewed as a major obstacle in developing adaptive instructional systems. In addition, given these data were obtained from individuals who are not "statisticians" suggests that DNA can be used by persons varying in levels of expertise. This also suggests a potential avenue for DNA as a research tool investigating knowledge structures of people with varying levels of competence in a domain, as well as changes in those structures over time.

There are several key features of DNA that, we believe, make this a viable alternative to current, costly knowledge-elicitation techniques. Because DNA

supports streamlining portions of the interview, transcription, and organization processes, it allowed us to obtain data simply by giving each expert the program along with a short letter explaining the goals of the curriculum. The program obviates the need for transcribing lengthy interviews. Additionally, experts are able to explicate and organize their knowledge within the same elicitation session, which translates into expected savings of time and money without sacrificing accuracy. This will be examined in future studies.

DNA's applicability is enhanced because it elicits, and then allows SMEs to represent graphically, a range of knowledge types. Specifically, the Decompose module focuses on eliciting *three* knowledge types: basic, procedural, and conceptual (what, how, and why). Additionally, the Network module will ultimately be able to produce a conceptual graph that incorporates information from the three types of representations mentioned earlier. The result is that the representational scheme enables DNA to obtain declarative, procedural, and conceptual information, promoting applicability across multiple topics. Heretofore, intelligent instructional systems have been built for single outcome types (e.g., production systems for procedural knowledge), thus varied knowledge types have been forced into a one-scheme-fits-all representation. In contrast, typical courses or curricula contain rich mixtures of knowledge types. For example, one can know the formula of the statistical Mean (BK) but not know how to compute it (PK), or one can be unfamiliar with the formula, but know how to compute it. In any case, it makes sense to be sensitive to representation differences when initially gathering curriculum elements for any course (during the CTA). That is what we are attempting to do with DNA via the three interfaces or paths reflecting the three main knowledge types.

Another design feature of DNA is its compatibility with an empirically validated instructional framework (i.e., SMART). SMART relies on information present in hierarchical-knowledge structures (e.g., parent/child relations) to manage instruction and remediation. DNA's Network module provides the SME with tools to create such a hierarchical knowledge structure. In addition, the Decompose module's what, how, and why questions map onto the instructional framework of basic, procedural, and conceptual knowledge types embodied by SMART, which relies on these knowledge types to provide differential instruction, remediation, and assessment. For instance, procedural knowledge is instructed within a problem-solving context, while conceptual knowledge may use analogies for instruction. Therefore, DNA's capacity to identify different knowledge types facilitates SMART's management of more customized instruction.

Our initial question underlying DNA's design feasibility concerned whether, indeed, DNA can extract comprehensive and reasonable knowledge from experts. Results from this preliminary evaluation are encouraging. In a relatively short amount of time and with minimal resource cost, the Decompose module of DNA was able to elicit 62% of the curriculum elements that are in place in an extant tutor. This suggests that the general approach implemented by DNA (with all of its limitations) works to produce valid data that could potentially serve as the basis for curriculum development. Future studies will examine DNA's efficiency relative to standard knowledge elicitation techniques. Additional questions we plan to explore

include, among others: (a) Can DNA be used to elicit knowledge across a broad range of domains? (b) Is it differentially effective in eliciting basic, procedural, and conceptual knowledge elements? and (c) Do differing levels of expertise result in data structures that vary in kind, rather than quantity? In short, future research and development will focus on identifying where we have and have not succeeded in our aim to expedite development of intelligent instructional systems.

6. REFERENCES

Almond, R., Steinberg, L. & Mislevy, R. (in press). A four-process architecture for assessment delivery, with connections to assessment design. *Journal of Technology, Learning, and Assessment*.

Anderson, J. R. (1987). Skill acquisition: Compilation of weak-method problem solutions. *Psychological Review*, *94*, 192-210.

Anderson, J. R. (1988). The expert module. In M. C. Polson & J. J. Richardson (Eds.), *Foundations of intelligent tutoring systems* (pp. 21-50). Hillsdale, NJ: Lawrence Erlbaum Associates.

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

Chipman, S., Shalin, V., & Schraagen, J. (Eds.) (2000). *Cognitive Task Analysis*. Hillsdale, NJ: Erlbaum Associates.

Collins, A. M., & Loftus, E. F. (1975). A spreading activation theory of semantic processing. *Psychological Review*, *82*, 407-428.

Collins, A. M., & Quillian, M. R. (1969). Retrieval time from semantic memory. *Journal of Verbal Learning and Verbal Behavior*, *8*, 240-248.

Gagné, R. (1985). *The Conditions of Learning and Theory of Instruction*. Holt, Rinehart, and Winston. New York.

Gordon, S. E., Schmierer, K. A., & Gill, R. T. (1993). Conceptual graph analysis: Knowledge acquisition for instructional system design. *Human Factors*, *35*, 459-481.

Hubert, L., Arabie, P. & Meulman, J. (2001). *Combinatorial data analysis: Optimization by dynamic programming* (Vol. 1). Philadelphia, PA: SIAM/ Monograph on Discrete mathematics and applications.

Johnson-Laird, P., N. (1983). *Mental models*. Cambridge, MA: Harvard University Press.

Lajoie, S. P., & Derry, S. J. (1993). *Computers as cognitive tools.* Hillsdale, NJ: Lawrence Erlbaum Associates.

Merrill, M. D. (1994). *Instructional Design Theory*. Englewood Cliffs: Educational Technology Publications.

Merrill, M. D. (2000). Knowledge objects and mental models. In D. A. Wiley (Ed.), *The Instructional Use of Learning Objects: Online Version*. Retrieved March 6, 2002, from the World Wide Web: *http://reusability.org/read/chapters/merrill.doc*

Minstrell, J. (2000). Student thinking and related instruction: Creating a facet-based learning environment. In J. Pellegrino, L. Jones, & K. Mitchell (Eds.) *Grading the Nation's Report Card: Research for the Evaluation of NAEP*. Committee on the Evaluation of NAEP, Board on Testing and Assessment. Washington, DC: National Academy Press.

Mislevy, R. J., Almond, R. G., Yan, D., & Steinberg, L. S. (2000, March). *Bayes nets in educational assessment: Where do the numbers come from?* (CSE Technical Report 518). Retrieved from *http://www.cse.ucla.edu/CRESST/Reports/TECH518.pdf*

Mislevy, R. J., Steinberg, L. S., & Almond, R. G. (1999, January). *On the roles of task model variables in assessment design* (CSE Technical Report 500). Retrieved from *http://www.cse.ucla.edu/CRESST/Reports/TECH500.pdf*

Mislevy, R. J., Steinberg, L. S., Almond, R. G., Haertel, G. D., & Penuel, W. R. (2001, February). *Leverage points for improving educational assessment* (CSE Technical Report 534). Retrieved from *http://www.cse.ucla.edu/CRESST/Reports/newTR534.pdf*

Murray, T. (1998). Authoring Knowledge Based Tutors: Tools for Content, Instructional Strategy, Student Model, and Interface Design. *Journal of the Learning Sciences* (Special Issue on Authoring Tools for Interactive Learning Environments), *7*, No.1, pp. 5-64.

Polson, M. C., & Richardson, J. J. (1988). *Foundations of Intelligent Tutoring Systems*. Hillsdale, NJ: Lawrence Erlbaum Associates.

Schank, R. (1999). *Dynamic memory revisited*. Cambridge, England: Cambridge University Press.
Shraagen, J. M. C., Chipman, S. E., Shute, V. J., Annett, J., Strub, M., Sheppard, C., Ruisseau, J. Y., & Graff, N. (1997). *State-of-the-art review of cognitive task analysis techniques*. Deliverable Report of RSG.27 on Cognitive Task Analysis NATO Defense Research Group (Panel 8/RSG.27). TNO Human Factors Research Institute Group: Information Processing.
Shute, V. J. (1995). SMART: Student modeling approach for responsive tutoring. *User Modeling and User-Adapted Interaction*, *5*, 1-44.
Shute, V. J., & Gluck, K. A. (1994). *Stat Lady: Descriptive Statistics Module*. [Unpublished computer program]. Brooks Air Force Base, TX: Armstrong Laboratory.
Shute, V. J., & Psotka, J. (1996). Intelligent tutoring systems: Past, present, and future. In D. Jonassen (Ed.), *Handbook of Research on Educational Communications and Technology*, (pp. 570-600). New York, NY: Macmillan.
Shute, V. J. & Towle, B. (in press). Adaptive e-Learning. Paper to appear in special issue of *Educational Psychologist*, in honor of Dr. Richard Snow.
Shute, V. J., Gawlick, L. A., & Lefort, N. K. (1996). Stat Lady (DS-2 module) [Unpublished computer program]. Brooks Air Force Base, TX: Armstrong Laboratory.
Shute, V. J., Torreano, L. A., and Willis, R. E. (1999). Exploratory test of an automated knowledge elicitation and organization tool. *International Journal of AI and Education*, *10*(3-4), 365-384.
Shute, V. J. & Torreano, L., & Willis, R. (2000). DNA: Towards an automated knowledge elicitation and organization tool. In S. P. Lajoie (Ed.) *Computers as Cognitive Tools, Volume 2*. Hillsdale, NJ: Lawrence Erlbaum Associates, pp. 309-335.
Sleeman, D. H., & Brown, J. S. (1982). *Intelligent tutoring systems.* London, England: Academic Press.
VanLehn, K. (1990). *Mind bugs: The origins of procedural misconceptions.* Cambridge, MA: MIT Press.
White, B. Y. & J. R. Frederiksen (1987). Qualitative models and intelligent learning environments. In R. Lawler & M. Yazdani (Eds.), *AI and education*, (pp. 281-305). Norwood, NJ: Ablex Publishing.
Williams, K. E. (1993). *The development of an automated cognitive task analysis and modeling process for intelligent tutoring system development*. Contract final report on N00014-97-J-5-1500. Manpower Personnel and Training Program, Office of Naval Research.

ACKNOWLEDGEMENTS

This work was done while that authors worked for Armstrong Laboratory Human Resources Directorate. The research was supported by the USAF Armstrong Laboratory, and also, in part, by a National Research Council (USAF) Research Associateship Award granted to the second author (1997-1999). We would like to acknowledge the invaluable contributions to this research (from concept design to development) by our colleague Ross Willis, and to thank Tom Murray and two anonymous reviewers for their sage comments on an earlier draft of this chapter. Finally, we thank Irv Katz for his assistance specifying DNA's object model.