On the Roles of External Knowledge Representations in Assessment Design

CSE Report 722

Robert J. Mislevy,[1] John T. Behrens,[2] Randy E. Bennett,[3] Sarah F. Demark,[2] Dennis C. Frezzo,[2] Roy Levy,[4] Daniel H. Robinson,[5] Daisy Wise Rutstein,[1] Valerie J. Shute,[6] Ken Stanley,[2] and Fielding I. Winters[1]

[1]University of Maryland, College Park; [2]Cisco Systems, Inc.; [3]Educational Testing Service; [4]Arizona State University; [5]University of Texas at Austin; [6]Florida State University

July 2007

National Center for Research on Evaluation,
Standards, and Student Testing (CRESST)
Center for the Study of Evaluation (CSE)
Graduate School of Education & Information Studies
University of California, Los Angeles
GSE&IS Building, Box 951522
Los Angeles, CA 90095-1522
(310) 206-1532

Project 3.6 Study Group Activity on Cognitive Validity: Strand 1–Cognitively Based Models and Assessment Design
Robert J. Mislevy, Project Director, National Center for Research on Evaluation, Standards, and Student Testing (CRESST)/University of Maryland

The work reported herein was supported under the Educational Research and Development Centers Program, PR/Award Number R305B960002-01, as administered by the Institute of Education Sciences, U.S. Department of Education.

ON THE ROLES OF EXTERNAL KNOWLEDGE REPRESENTATIONS IN ASSESSMENT DESIGN

Robert J. Mislevy,[1] John T. Behrens,[2] Randy E. Bennett,[3] Sarah F. Demark,[2] Dennis C. Frezzo,[2] Roy Levy,[4] Daniel H. Robinson,[5] Daisy Wise Rutstein,[1] Valerie J. Shute,[6] Ken Stanley,[2] and Fielding I. Winters[1]

Abstract

People use external knowledge representations (EKRs) to identify, depict, transform, store, share, and archive information. Learning how to work with EKRs is central to becoming proficient in virtually every discipline. As such, EKRs play central roles in curriculum, instruction, and assessment. Five key roles of EKRs in educational assessment are described:

1. An assessment is itself an EKR, which makes explicit the knowledge that is valued, ways it is used, and standards of good work.
2. The analysis of any domain in which learning is to be assessed must include the identification and analysis of the EKRs in that domain.
3. Assessment tasks can be structured around the knowledge, relationships, and uses of domain EKRs.
4. "Design EKRs" can be created to organize knowledge about a domain in forms that support the design of assessment.
5. EKRs from the discipline of assessment design can guide and structure the domain analyses noted in (2), task construction (3), and the creation and use of design EKRs noted in (4).

The third and fourth roles are discussed and illustrated in greater detail, through the perspective of an "evidence-centered" assessment design framework that reflects the fifth role. Connections with automated task construction and scoring are highlighted. Ideas are illustrated with two examples: "generate examples" tasks and simulation-based tasks for assessing computer network design and troubleshooting skills.

1.0 Introduction and Overview

Knowledge representation is a central theme in cognitive psychology. Internal knowledge representation refers to the way that information about the world is represented in our brains, and as such lies at the center of learning, interacting, and

[1]University of Maryland, College Park; [2]Cisco Systems, Inc.; [3]Educational Testing Service; [4]Arizona State University; [5]University of Texas at Austin; [6]Florida State University.

problem solving of all kinds. This report concerns external forms of knowledge representation. An external knowledge representation (EKR), or inscription (Lehrer & Schauble, 2002), is a physical or conceptual structure that depicts entities and relationships in some domain, in a way that can be shared among different individuals or by the same individual at different points in time. EKRs are human inventions that overcome obstacles to human information processing with respect to limited working memory, faulty long-term memory over time and in volume, coordinating the actions of many individuals, and idiosyncratic ways of thinking about some phenomenon of common interest. Examples of EKRs include maps, lists, graphs, wiring diagrams, bus schedules, musical notation, mathematical formulas, object models for business systems, and the 7-layer OSI model for computer networks.

This report considers the roles of knowledge representations in educational assessment, with an eye toward making the activities of assessment design more explicit, more valid, and more efficient. The following section provides a brief review of important features of EKRs. Five roles of EKRs in assessment are then outlined. We note how EKRs connect expertise in a domain with learning and assessment in that domain, and hence shape both instructional design and assessment design. We then further develop and illustrate two of these roles, namely the design of assessment tasks around domain EKRs and the creation of special EKRs that help the assessment designer accomplish this. We place this discussion in the context of evidence-centered assessment design (ECD; Mislevy, Steinberg, & Almond, 2003) to take advantage of EKRs emerging from that work.

The ideas presented here are illustrated with examples from three assessment projects. The first is a relatively simple example based on Butterfield, Nielsen, Tangen, and Richardson (1985) concerning inductive reasoning tasks. This example is interleaved throughout the report. Three additional, more complex examples are discussed in greater detail in their own sections in the latter part of the report. They concern "generating examples" and "arithmetic expressions" task types developed at Educational Testing Service (Bennett et al., 1999; Bennett, Morley, & Quardt, 2000; Katz, Lipps, & Trafton, 2002), and Cisco Systems' computer network simulation (CNS) assessments of design and troubleshooting (Behrens, Mislevy, Bauer, Williamson, & Levy, 2004; Frezzo & Stanley, 2005; Williamson, Bauer, Steinberg, Mislevy, & Behrens, 2004).

2.0 Knowledge Representations in Assessment

EKRs play a central role in human cognition, as a means of identifying, expressing, communicating, and utilizing information in social spheres. Generally speaking, EKRs are a vehicle for discourse, used either by a single individual (mediated cognition) or among individuals (distributed cognition), at one point in time or across multiple points in time. They concern entities, relationships, and processes in some domain, and their organizational form is used to create, gather, store, transform, and use information more easily than would be accomplished without them. Markman's (1999, pp. 5-8) definition of a knowledge representation has four components:

- A represented world: The domain that the representations are about. The representation world may be the world outside the cognitive system or some other set of representations inside the system. That is, one set of representations can be about another set of representations.
- A representing world: The domain that contains the representations. (The terms represented world and representing world come from a classic paper by Palmer, 1978.)
- Representing rules: The representing world is related to the represented world through a set of rules that map elements of the represented world to elements of the representing world.
- A process that uses the representation: It makes no sense to talk about representations in the absence of processes. The combination of the first three components (a represented world, a representing world, and a set of representing rules) creates merely the potential for representation. Only when there is also a process that uses the representation does the system actually represent, and the capabilities of a system are defined only when there is both a representation and a process.

Some EKRs, such as mathematical notation and computer languages, gain their power through symbol manipulation. Once information has been encoded in the required form, operations can be carried out on the symbols to transform or combine the information in ways that would be difficult or impossible for a human to do unaided. A quotation from Whitehead (1958) is apropos:

> By relieving the brain of all unnecessary work, a good notation sets it free to concentrate on more advanced problems, and, in effect, increases the mental power of the race. . . . Civilization advances by extending the number of important operations which we can perform without thinking about them. (pp. 39, 42)

Other EKRs, such as graphs and maps, encode information in ways that capitalize on humans' strengths in recognizing patterns and interpreting spatial relationships (see, for example, Lewandowsky & Behrens, 1999, on statistical graphs and maps):

> The greatest possibilities of visual display lie in vividness and inescapability of the intended message. A visual display can stop your mental flow in its tracks, and make you think. A visual display can force you to notice what you never expected to see. . . . One should see the intended at once; one should not even have to wait for it to appear. (Tukey, 1990, p. 328)

Many EKRs use both symbolic and perceptual representation in varying mixtures (e.g., Tufte, 1990). A table exploits spatial arrangement to communicate the relevance of the organizing concepts of rows and columns for the subject of each cell (Mosenthal & Kirsch, 1989).

2.1 Properties of Knowledge Representations

Several properties of EKRs are relevant to their roles in assessment. One of the most important is that an EKR does not attempt to include everything in the represented world, only certain entities and relationships. It highlights those entities and relationships, and facilitates thinking about them, talking about them, and working with them. This is the *ontology* of the EKR. Unrepresented aspects of the represented world are considered irrelevant. The velocity of a falling body is represented by $v_0 + g\,t$, whether the body is a cannonball or a feather, whether it is falling in Austin or Tokyo. The breadth of applicability of EKRs can be a strength. It is also a potential weakness in application when what is omitted from the mapping is important in the real-world situation, as when the velocity of a falling feather is lower because of air resistance. While carrying out reasoning within the representing world is important in learning to use EKRs, it is just as important to learn when to apply them gainfully and how to recognize a potentially hazardous misfit (a central topic in statistics, for example; e.g., Belsley, Kuh, & Welch, 1980, on diagnostics in regression analysis).

In addition to focusing on only certain aspects of situations in the represented world, EKRs are optimized for certain uses regarding those aspects. A domain of any complexity typically has many EKRs, each tuned to different relationships and purposes. For example, matrix algebra, path diagrams, and computer code input are all used to represent structural equation models (SEMs) in psychometrics (Figure 1). The matrix equations admit to symbol-manipulation procedures for taking

$$\begin{pmatrix} y_1 \\ y_2 \\ y_3 \end{pmatrix} = \begin{pmatrix} \lambda_1 \\ \lambda_2 \\ \lambda_3 \end{pmatrix} \eta + \begin{pmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \varepsilon_3 \end{pmatrix}$$

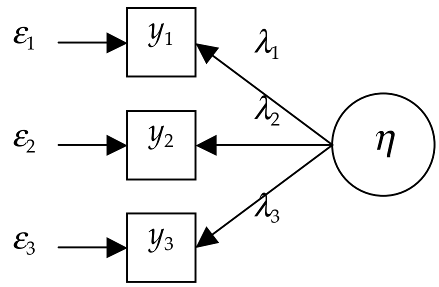


Figure 1. Matrix algebra and path diagram representations of a factor analysis model.

derivatives, which support algorithms for finding the values of the variables that fit the data best—finding maxima of multivariable likelihood functions is not something people do well in their heads. But graphical representations have advantages at the model-building stage, because the qualitative relationships among variables are immediately apparent and rapidly specified. Computer programs such as EQS (Bentler, 1995) allow the user to specify a model by working with a graphical interface, then generate code automatically to estimate the parameters with algorithms derived under the algebraic representation. We will see in the CNS example of Section 5 how algorithmic conversions from one knowledge form to another provides advantages in computer-based assessment systems for domain analysis, task authoring, task presentation, interaction with the examinee, and automated scoring.

This attunement of EKRs to different uses accounts for the use of multiple EKRs in a given domain (Ainsworth, 1999). Multiple EKRs also occur when the complexities of real world situations lend themselves to modeling at different levels or from different perspectives. In transmission genetics, for example, there are EKRs for expressing relationships at the levels of species, individuals, cells, and molecules. While each EKR highlights entities and relationships at a certain level of analysis, relationships and constraints can cross levels and representational forms as well. The similarities of elements' chemical properties in a column of Mendeleev's periodic table correspond to similarities in electron shell diagrams. Translating information from one form to another is often a target of learning in content domains, as the process of solving a problem may take the form of a sequence of transformations within and between models, mediated by operations carried out with EKRs.

One can speak of EKRs at various levels of generality. For example, the elements and representational capabilities of Cartesian graphs and their attendant elements can be addressed at a general level, to understand the kinds of relationships that can be used to represent among variables in any domain. Certain knowledge associated with graphs can thus be learned and used (and assessed) across domains. Scatterplots in statistics and acceleration graphs in physics are both special cases of Cartesian graphs that can be studied in their own right, as patterns in graphs correspond to more specialized representations such as acceleration formulas, which are in turn grounded in the generative principles of that particular domain. More focused use of EKRs, as to both form and substance, entails additional connections with relationships in the represented world, more instances of connections across EKRs in that world, and more powerful capabilities for more specific operations and inferences.

EKRs hold value because people can do things with them. Well-chosen EKRs can incorporate subtle and hard-won insights into a form that can be applied mechanically. Fifty years ago, an economist could win a Nobel prize for generating and solving from first principles the kinds of systems of linear equations that EQS users can apply today without knowing either calculus or matrix algebra. It is an advantage of an EKR that a user can exploit the results of deep principles without knowing them explicitly. To enjoy these benefits, however, the user must become attuned to ways the EKR offers to display or transform information—its affordances, to use Gibson's (1966) term. The problem of designing EKRs to best communicate information and affordances receives both practical and academic attention in fields such as graphics (e.g., Pinker, 1990) and human-computer interfaces (e.g., Card, Moran, & Newell, 1983). This research is prompted in part by the fact that EKRs that can be expressed in symbolic form support multiple views and automated transformations. For example, CNS works back and forth between perceptual EKRs for presenting and capturing information from examinees and symbolic EKRs for evaluating their work and transforming information from one form to another.

How do EKRs facilitate work? By focusing on recurrent patterns at a level above the particulars of any problem, EKRs facilitate analogies across problems and domains. They make it easier to acquire and structure information. They coordinate work in projects so large or complex that no one can know all the details of all of its facets. In such cases, EKRs such as Gantt charts and object models help people understand their roles and connect their work with that of others. They provide a common language for people to express information and work with it in ways that tacitly incorporate experience from other times and other people. The form of an EKR can indicate when information is missing. For example, representing text information in a matrix graphic organizer rather than text makes missing information more salient (Figure 2). EKRs such as blueprints, agendas, schedules, and to-do lists are significant in planning, as they organize and indicate what information is needed, how it is to be acted upon, and what a solution will look like. Collins and Fergusen (1993) emphasize how people create new knowledge by using

Moths and Butterflies (text)

A moth has two sets of wings. It folds the wings down over its body when it rests. The moth has feathery antennae and spins a fuzzy cocoon. The moth goes through four stages of development.

A butterfly also goes through four stages of development and has two sets of wings. Its antennae, however, are long and thin with knobs at the ends. When a butterfly rests, its wings are straight up like outstretched hands.

Moths and Butterflies (matrix organizer)

	Moths	**Butterflies**
Wings	Two sets	Two sets
Rest	Wings over body	Wings outstretched
Antennae	Feathery	Long, thin, with knobs
Cocoon	Fuzzy	
Development	Four stages	Four stages

Figure 2. What information is presented about moths but not about butterflies? The missing element is easier to see from the matrix organizer than in the text.

such forms by referring to them as "epistemic forms," and the ways that people learn to use them as "epistemic games." These advantages turn out to be a fair summary of the arguments for the evidence-centered assessment design framework (Mislevy, Steinberg, et al., 2003; Mislevy & Riconscente, 2006) discussed in Section 3.

2.2 Roles of Knowledge Representations in Assessment

Looking at educational assessment through the lens of EKRs reveals their presence throughout the enterprise, at different stages, at different levels, and with different purposes. The following sections discuss five key roles that EKRs play in assessment:

1. An assessment is in itself an EKR, which makes explicit the knowledge that is valued, ways it is used, and standards of good work.
2. The analysis of any domain in which learning is to be assessed must include the identification and analysis of the EKRs in that domain (i.e., the "domain EKRs").
3. Assessment tasks can be structured around the knowledge, relationships, and uses of domain EKRs.
4. "Design EKRs" can be created to organize knowledge about a domain (including its domain EKRs) in forms that support the design of instruction and assessment.
5. EKRs from the disciplines of instructional design and assessment design can guide and structure the domain analyses noted in (2), the task construction noted in (3), and the creation and use of design EKRs noted in (4).

2.2.1 Assessments Are Themselves Knowledge Representations

The analogy of assessment to measurement is vital to its conduct, but it is not sufficient. A student taking an assessment is engaged in a form of a socially construed discourse (Gitomer & Steinberg, 1999) no less than a teenager playing a video game or a taxpayer completing an IRS 1040 form. This observation holds implications for assessment designers and students alike. Designers must always be aware that an assessment constitutes the most direct statement of the knowledge and skills that are valued, in effect if not in intention. The process of constructing an assessment, done thoughtfully, elicits an understanding of the knowledge that is targeted, the actions of students that provide evidence about it, and the circumstances under which that knowledge should be brought to bear in order to achieve certain kinds of outcomes (Wiggins, 1998). An assessment is an EKR that communicates the targets of learning and the standards of performances to all stakeholders, and its construction serves educative purposes before the first examinee ever sees it.

In order to perform well in an assessment, students must not only have become facile with the targeted knowledge and skills, but they must also be able to work with them in the forms and under the conditions that characterize the assessment situation. That is, the students must be attuned to the affordances of the assessment as a form of EKR. The more complex an assessment is, in terms of the embedded EKRs students will interact with and the standards by which EKRs that students produce will be evaluated, the more important it is to ensure this attunement has taken place before the assessment begins. For students attempting to solve an interactive chemistry investigation with an unfamiliar computer interface, the interface can present more difficulties than the chemistry. Similarly, students cannot "explain" a solution to a mathematics problem until they understand the nature, the forms, and the expectations of exposition that are required to produce a "satisfactory explanation."

2.2.2 Identifying the Knowledge Representations of a Domain

Becoming an expert in a domain is a process of learning about the nature of knowledge in the domain, including terms, principles, patterns, and exemplars, and the nature of interaction among those who participate in that domain (Ericsson, 1996). That is, the kinds of knowledge highlighted under both an acquisition metaphor and a participation metaphor (Sfrad, 1998) are required. Knowledge representations play central roles in both. EKRs embody the important ideas and relationships in a domain, organize them so that they are the vehicle for doing work in the domain, define the language by which people acquire and communicate information in that domain, and coordinate the interactions of people as they work toward common ends. It is not much of an understatement to say that learning in a domain is learning to use the EKRs of the domain—the domain EKRs, as we will call them.

No analysis of a learning domain can be complete without an investigation of the EKRs that are used in the domain and the situations in which they are used. Learning materials such as textbooks and exemplars are a natural place to begin, but the selection of EKRs used in instruction can be biased toward "academic" EKRs. Additional EKRs used in practical work, perhaps informal or embedded in tools, are also part of the targeted domain, and learning how and when to use them is part of the targeted learning.

2.2.3 Structuring Tasks Around Domain Knowledge Representations

Assessment is reasoning about what students know, can do, or have accomplished more broadly, from evidence in the form of a relative handful of particular things they say, do, or make in particular situations. The situations in which the student is to act are defined in large part through knowledge representations. The various EKRs that constitute an assessment task provide information about a situation to the student, suggest the nature of the problem, suggest the terms in which the problem is to be approached, offer clues as to the nature of a solution and the criteria of evaluation, and provide affordances for getting started. This is as true of open-ended performances or portfolios as it is of objective tests consisting of multiple-choice items. Furthermore, what the student says, does, or makes in response—the work products—are typically structured in terms of the knowledge representations of the domain as well.

Research on expertise reveals increasing expertise in the use of domain EKRs as proficiency increases, in ways that hold implications for designing tasks and evaluating performances. As a first example, Kindfield's (1999) study of experts' and novices' use of diagrams to reason through genetics problems revealed an interesting reversal: Novices' drawings were often more complete and better proportioned than experts', but what distinguished experts' diagrams was that only the salient features tended to be shown, and the relationships important to the problem at hand were rendered with whatever accuracy was needed to solve the problem. That is, the experts' diagrams were more efficacious than those of the novices. As a second example, Cameron et al. (2000) found increasing proficiency in dental hygienists at increasing levels of experience with respect to their use of EKRs such as radiographs, hard and soft tissue charts, and probing depth charts. Early stages of learning were marked by the ability to identify and interpret key features on a given single representation. Expert hygienists were distinguished from recently licensed hygienists by a superior ability to integrate information across multiple representations of different types, effectively constructing a model of a patient about whom all the representations were different, yet coherent, views of the same person.

A central idea for assessment design, and a central topic of the CNS example later in this report, is that a systematic analysis of the EKRs in a domain—what they are, their features, and how people use them—is a foundation for principled generation of assessment tasks. An understanding of the entities and relationships of each EKR and the relationships among them is developed in conjunction with an understanding of the kinds of reasoning or actions one wants students to carry out using the EKRs. The outcomes of this analysis lay the groundwork for schemas of tasks that focus on valued work in the domain in explicit ways, and exist at some level of generality above particular tasks. The level of generality of the EKRs and the resulting schemas depends on the intended use, with the usual understanding that broad applicability of general forms trades off against the power of specific forms. These task construction schemas can themselves be expressed in terms of EKRs—item forms, as discussed in Hively, Patterson, and Page (1968), for example, item shells as discussed by Haladyna and Shindoll (1989), or task models, as discussed in Mislevy, Steinberg, et al. (2003).

At this point, we introduce an example from Butterfield et al. (1985) concerning theory-based generation of letter series tasks, a measure of inductive reasoning (Thurstone & Thurstone, 1941). Here are two examples based on the Primary Mental Abilities test battery (Thurstone & Thurstone, 1962):

Fill in the next letters in the series:

C D C D C D __ __ __

A T B A T A A T B A T __ __ __

This knowledge representation is an example of an *item type*—a particular kind of EKR used in assessment to present information to an examinee and set expectations for a response. This particular EKR consists of a series of symbols, read from left to right, arranged according to a pattern, or rule, that both explains the appearance of the symbols that are depicted and sets expectations for the symbols that would come next. The student's task is to determine the rule and make predictions. The blanks are affordances—the natural place to write the symbols that extend the pattern if you understand what the EKR is about, but mysteries if you don't. Although these items require no specialized content knowledge other than the alphabet, they reflect the kind of reasoning required in more complex inductive problems that do require more substantive knowledge, such as scientific inquiry. Because this is the representational form that the student works with, it is the domain EKR in our first assessment example.

2.2.4 Representations for Designing Assessments in Given Domains

Advantages can be gained when the characteristics of the EKRs can themselves be represented in higher-level EKRs that are devised to serve the purposes of assessment design. We may call these design EKRs. Design EKRs are related to domain EKRs, but they are tuned to the purpose of generating domain EKRs to be used in tasks. They describe salient features of task situations, in ways that both imply domain representations and indicate the kinds of reasoning and knowledge the student will need to call upon. We shall see that the same representations can provide information to EKRs used in other stages of assessment design and delivery, such as task selection and psychometric modeling (Bejar, 2002; Embretson, 1998).

Butterfield et al. (1985) created a design EKR for the domain of letter series tasks described in the previous section. Letter series tasks had been used at least as far back as Thurstone's research in 1941 in both practical applications and psychometric research. Task generation was idiosyncratic, however, and systematic examinations of both the structure of tasks and how people solve them were lacking (Butterfield et al., 1985). Simon and Kotovsky (1963) devised a symbol system to describe such tasks after they have been written, and their analysis is Butterfield et al.'s starting point for an EKR that supports automated task generation in this domain. An abbreviated version and a few examples of the design EKR for letter series rules convey the key ideas: Letter series tasks are composed of one or more strings of letters. Within a string, special relationships hold for moving from one letter to the next, such as identity (I), next letter (N), and back a letter (B). A rule is expressed by the relationships of letters within a string, and the strings' relationships to one another. The rule underlying the series CDCDCD is denoted by I1 I2, instantiated with C and D as the initial values of the first and second strings. The same rule instantiated with R and T as the initial values yields RTRTRT. The series MABMBCMCD is expressed as I1 I2 N2, with initial values M and A.

This design EKR for expressing rules is obviously distinct from letter series tasks themselves, but they are related in ways that serve the purposes of the assessment designer. A rule expressed in the design EKR grammar and initial string values suffices to produce a letter series task. Operations can be defined on rules expressed in the grammar of the EKR to address issues of form, such as when two rules produce identical series. Other operations on rules address psychological issues such as memory load, as a function of calculable properties such as "Counts = # moving strings * (period – # adjacent identity relations)." Related operations can be used to address psychometric issues such as task difficulty (as in Embretson, 1998). The design EKR for letter series tasks, therefore, has pragmatic connections to the task authoring, psychological argument, and measurement modeling layers of the assessment enterprise.

An early example of generative design EKRs is Hively et al.'s (1968) idea of "item forms" for generating whole number arithmetic items, two of which appear as Figure 3. Another example appears in Bormuth's (1970) work on generating "wh" questions from text. The EKR is a syntactic representation of one or more propositions, which is amenable to symbolic transformations that yield questions that can be used to assess basic comprehension. Both of these examples provided EKRs that enabled an assessment designer to map the structures and content of domain EKRs (arithmetic items and English text) into more abstract EKRs that support transformations into tasks. The "generating examples" and CNS examples in Sections 4 and 5 illustrate more recent work, in which the capability of computers to carry out symbol manipulation is exploited more fully in the automated construction of tasks.

Descriptive Title	Sample Item	General Form	Generation Rules
Basic fact; Minuend > 10	13 <u>-6</u>	A <u>-B</u>	1. A=1a ; B=b 2. (a<b) ε U 3. {H, V}
Borrow across zero	403 <u>-138</u>	A <u>-B</u>	1. # digits = {3,4} 2. $A=a_1a_2\ldots$; $B=b_1b_2\ldots$ 3. $(a_1>b_1)$, $(a_3<b_3)$, $(a_4 \geq b_4)$, ε U_0 4. b_2 ε U_0 5. $a_2 = 0$ 6. P{{1,2,3},{4}}

Capital letters represent numerals, lower case represent digits.
x ε { -- } means chose x with replacement from the set.
U = {1,2,…,9}; U_0={0,1,…,9}.

Figure 3. Two "item forms" from Hively, Patterson, and Page (1968).

2.2.5 Knowledge Representations in the Discipline of Assessment Design

As long as assessment has been practiced, EKRs have been developed to aid designers. Familiar examples include the aforementioned item types and item forms, test specifications (see Davidson & Lynch, 2001, for a recent in-depth discussion), and content-by-process matrices often based on Bloom's (1956) taxonomy of educational objectives. These EKRs are used to help designers generate items and assemble test forms. EKRs used in the analysis of test data are also familiar, from the symbolic representations used in psychometric models to innovative displays used to summarize patterns in performance for students and their teachers. Schemas for rubrics to evaluate open-ended task performances are also widely used, allowing an assessor (such as a classroom teacher) to adapt a tested evaluation procedure to locally customized tasks; a number of tools are available in interactive formats on the Internet. Wiggins (1998) offers designers of performance assessment a number of templates and flowcharts, all with an eye toward connecting what is assessed with goals of instruction.

Designing assessments of any complexity involves considerations at many levels: substantively grounded evidentiary arguments, design of operational elements such as tasks and scoring models, implementing the design in terms of specific tasks, and all the operational activities involved in actually carrying out the assessment. No single EKR can encompass all of this work; multiple, coordinated, representations are required. Developing frameworks for assessment design, complete with a conceptual rationale and multiple supporting EKRs, has been a focus of research in the assessment community in recent years (e.g., Embretson, 1998; Luecht, 2002, Wilson, 2005). The next section discusses one of these approaches in greater detail.

3.0 A Closer Look at EKRs and Assessment Design

Evidence-Centered Assessment Design (ECD) is a process of assessment design that involves gathering, organizing, and transforming information in a variety of representational forms, within the framework of a clearly articulated assessment argument. Under the ECD framework, EKRs are integral at every step in the process of developing and using an assessment. This section starts with a brief overview of ECD, then, through this perspective, discusses and provides examples of EKRs in assessment design.

3.1 A Brief Overview of Evidence-Centered Assessment Design

Central ideas in ECD are the assessment argument, layers of the assessment, and the role of EKRs in designing and implementing assessments. Messick (1994) concisely lays out the key aspects of an assessment argument by asking "what complex of knowledge, skills, or other attributes should be assessed. Next, what behaviors or performances should reveal those constructs, and what tasks or situations should elicit those behaviors?" (p. 16). All of the many terms, concepts, representations, and structures are aimed at constructing a coherent assessment argument and building machinery to implement it.

Adapting a "layers" metaphor from architecture and software engineering, ECD organizes the design process in terms of the following layers: domain analysis, domain modeling, conceptual assessment framework, assessment implementation, and assessment delivery (Mislevy & Riconscente, 2006). The fundamental work in assessment design can be viewed as creating, transforming, and using information in the form of EKRs within and between these layers. Table 1 summarizes these layers in terms of their roles, key entities (e.g., concepts and building blocks), and the EKRs that assist in achieving each layer's purpose. The layering suggests a sequential design process, but cycles of iteration and refinement across layers are the norm.

The first layer in the process of designing an assessment is the *domain analysis*. It lays the foundation for later layers by defining the knowledge, skills, and abilities (KSAs) that assessment users want to make inferences about, the student behaviors they can base their inferences on, and the situations that will elicit those behaviors. To this end, domain analysis entails gathering information about how people acquire, construct, represent, use, and communicate knowledge within the domain of interest. A critical part of domain analysis includes identification of EKRs important to the domain, as expertise in a domain necessarily includes knowledge of and understanding about how and when to use the EKRs in that domain. As such, domain EKRs represent knowledge, symbols, and procedures about which test takers can be assessed. Moreover, looking ahead to design layers, domain EKRs will also provide ways in which information about the domain can be represented to the test taker at the level of the assessment interface. In this way, an understanding of the EKR, as well as how to use it, is being assessed (Gitomer & Steinberg, 1999).

Table 1

Layers of Evidence-Centered Design

Layer	Role	Key entities	Selected external knowledge representations
Domain analysis	Gather substantive information about the domain of interest that has direct implications for assessment: how knowledge is constructed, acquired, used, and communicated.	Domain concepts, terminology, tools, knowledge representations, analyses, situations of use, patterns of interaction.	Content standards, concept maps (e.g., Atlas of Science Literacy, American Association for the Advancement of Science, 2001). Representational forms and symbol systems of domain of interest, e.g., maps, algebraic notation, computer interfaces.
Domain modeling	Express assessment argument in narrative form based on information from domain analysis.	Knowledge, skills and abilities; characteristic and variable task features, potential work products and observations.	Assessment argument diagrams, design patterns, content-by-process matrices.
Conceptual assessment framework	Express assessment argument in structures and specifications for tasks and tests, evaluation procedures, measurement models.	Student, evidence, and task models; student model, observable, and task model variables; rubrics; measurement models; test assembly specifications.	Test specifications; algebraic and graphical EKRs of measurement models; task template; item generation models; generic rubrics; automated scoring code.
Assessment implementation	Implement assessment, including presentation-ready tasks, scoring guides or automated evaluation procedures, and calibrated measurement models.	Task materials (including all materials, tools, affordances); pilot test data for honing evaluation procedures and fitting measurement models.	Coded algorithms to render tasks, interact with examinees, evaluate work products; tasks as displayed; IMS/QTI representation of materials; ASCII files of parameters.
Assessment delivery	Coordinate interactions of students and tasks: task-level and test-level scoring; reporting.	Tasks as presented; work products as created; scores as evaluated.	Renderings of materials; numerical and graphical score summaries; IMS/QTI results files.

The next layer in the ECD process is *domain modeling*. At the domain analysis layer, EKRs within the domain of assessment design come into play in the form of assessment argument diagrams (Bachman, 2003, Mislevy, 2003; see Figure 4 for the basic structure, adapted from Toulmin, 1958), content-by-process matrices, and the design patterns that will be discussed in more depth in the next section. Using these EKRs, domain modeling structures the outcomes of domain analysis in a form that reflects the structure of an assessment argument, in order to ground the more technical student, evidence, and task models that are required in the subsequent *conceptual assessment framework* (CAF) layer.

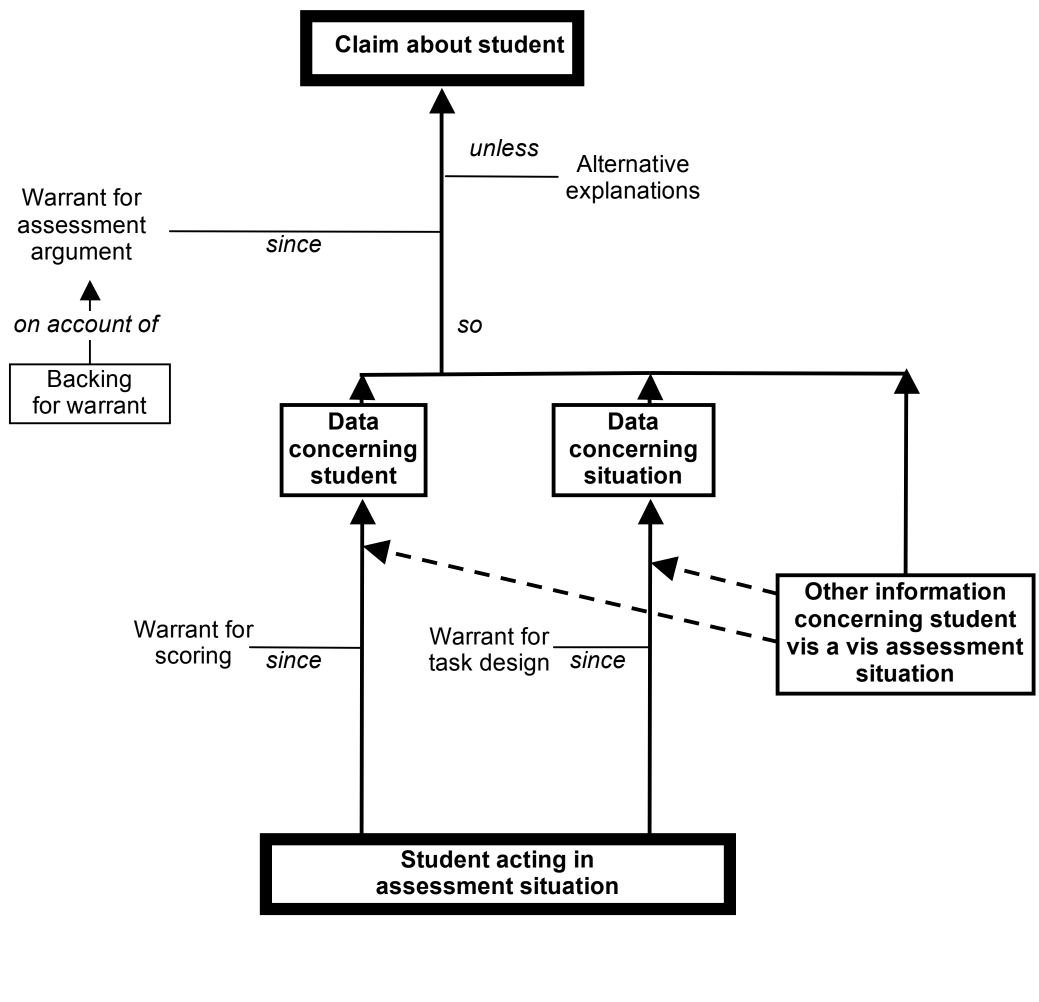


Figure 4. An assessment argument diagram (after Mislevy, 2006).

The conceptual assessment framework (CAF) concerns the technical specifications for the nuts and bolts of assessments, that is, the materials and processes that embody assessments. The central models in the CAF are the student model, evidence model, and task model (Figure 5). In addition, the assembly model governs how tasks are assembled into tests, a presentation model indicates the requirements for interaction with a student (e.g., simulator requirements), and the delivery model specifies requirements for the operational setting. An assessment argument laid out in narrative form at the domain-modeling layer is here expressed in terms of specifications for tasks, measurement models, scoring methods, and delivery requirements. Details about task features, measurement-model parameters, stimulus material specifications, and the like are expressed in terms of EKRs and data structures we will have a bit more to say about later in this section, which guide their implementation and ensure their coordination.

With information from the models in the CAF, delivery of an assessment from an ECD perspective is defined by a four-process architecture that includes the *activity selection process*, the *presentation process*, *response processing*, and the *summary scoring process* (Figure 6). The following brief outline is not sufficient to understand the roles and interplay of the processes; the interested reader is referred to Almond, Sternberg, and Mislevy (2002). What is important for this presentation is that every message that passes from one process to another is embedded in some EKR. It has

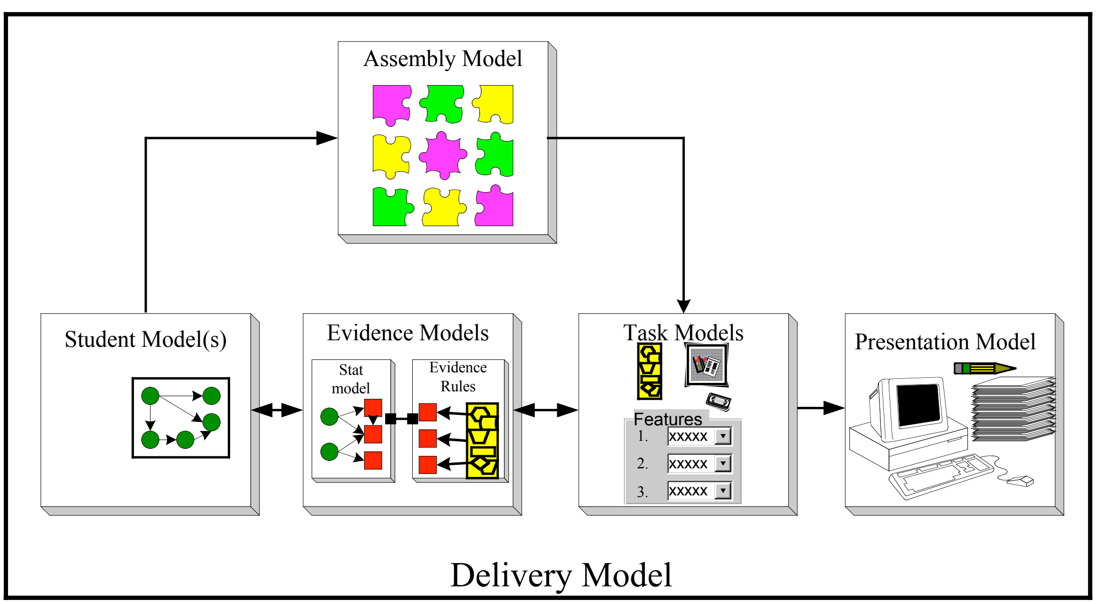


Figure 5. The central models of the conceptual assessment framework (CAF).

been produced by the sender, be it a human or computer, and provided in a form that the receiver, again a human or a computer, can use to carry out some other function essential to the operation of the assessment. Following sections will provide examples.

Starting in the upper left corner of Figure 6, the *activity selection process* selects a task (tasks include items, sets of items, or other activities) and directs the *presentation process* for display to the examinee. When the examinee has finished interacting with the item, the results (a *work product*) are sent to *response processing*. Information from the *task model* defined in the CAF provides the basis for *the presentation process* and *work product* specifications. From information outlined in the *evaluation model* of the CAF, the *response process* identifies essential *observations* about the results and passes them to the *summary scoring process*, which updates the *scoring record* about the examinee. The *scoring record* describes knowledge about the student-model variables articulated in the *student model* of the CAF. All four processes add information to the *results database*. The *activity selection process* again makes a decision about what to do next, based on the current scoring record of the participant or other criteria.

Similar to the process of object modeling in system design, the ECD structure ensures that the underlying logic of complex assessment activities is explicit, rather than being implicit throughout the process. That is, ECD structures are epistemic forms to organize and coordinate the work of designing and delivering assessments. They are meant to ensure continuity and clarity of the assessment argument at each layer and phase of the process.

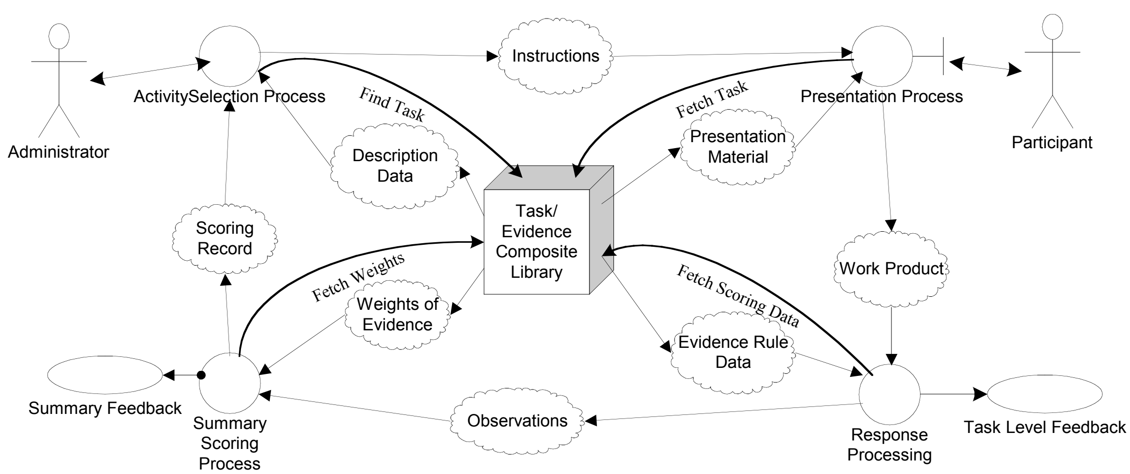


Figure 6. The four principal processes in the assessment cycle.

3.2 Domain Reasoning, Knowledge Representations, and Task Design

The ECD process affirms the idea that analysis of the EKRs central to a given domain sets the stage for the generation of assessment tasks in that domain. Essential to this idea is the connection between a given domain EKR itself and reasoning and the way people use it in the domain. This is critical because the knowledge needed to use a domain EKR in a particular circumstance is what we wish to draw inferences about. Identifying and articulating the relationship between using specific EKRs in particular situations and the type of knowledge this elicits is an important link in the assessment design process. Identification of these relationships during the domain analysis process sets up the construction of arguments in domain modeling, which in turn sets up the creation of schemas for designing tasks.

Butterfield et al.'s (1985) letter series example provides an example of the interplay between EKRs and knowledge. In this example, the EKR, a pattern of letters, provides a way for assessment users to reason about the underlying pattern. In essence, this EKR allows for assessment of the inductive reasoning ability of the test taker; the EKR itself becomes a tool for assessing this knowledge.

Checklists and behavioral inventories are examples EKRs that have long been used to ground licensure and certification tests. As epistemic forms, they provide structure to the job analyst's task of identifying the nature and frequency of tasks professionals carry out, from which assessment tasks will be devised.

More recent work in cognitive task analysis addresses the nature, organization, and use of knowledge employed (Schraagen, Chipman, & Shalin, 2000). This allows for distinctions between different types of knowledge and skills that one may want to evoke from an examinee, including declarative, procedural, or strategic knowledge, which may all be associated with one particular domain EKR. The information is collected during the domain analysis phase of the assessment design. For example, Shute, Torreano, and Willis's (2000) automated knowledge elicitation tool DNA (for Decompose, Network, Assess) provides structured, user-friendly Web forms to elicit domain experts' input on declarative, procedural, and conceptual knowledge requirements of common tasks in the domain. The results of such an analysis can be expressed in EKRs that are tuned to support the next step in the design process, namely domain modeling.

In addition to the argument schema shown in Figure 4, another EKR that has been developed for work in the domain modeling layer is the *design pattern* (Mislevy, Hamel, et al., 2003). Design patterns encapsulate knowledge about ways to address assessment challenges that recur across domains or within particular domains, organized in categories that connect to elements of an assessment argument on the one hand, and point ahead toward the more technical elements of the CAF. For example, Table 2 shows selected portions of a design pattern for *problem-solving in finite systems*, a valued skill in both everyday life (why won't this door close?) and in domains such as aircraft repair, computer programming, and the troubleshooting of computer networks addressed by the Cisco's Network Simulator tasks (see Section 5). A design pattern for this particular skill can be utilized across domains because it capitalizes on similar patterns in each. Within any given domain, multiple design patterns can be used to target the knowledge, skills, and abilities of interest—such as building a teamwork task around troubleshooting, working-in-groups, and self-monitoring design patterns.

The design pattern structure can be used to address the type of proficiencies that people employ when using domain EKRs. For example, in model- based reasoning an initial model, usually expressed in the form of an EKR, is created and iteratively revised as it is tested in real world situations (Stewart & Hafner, 1994). The Architectural Registry Examination (ARE) utilizes this type of reasoning with a CAD system that has examinees produce a domain EKR in the form of a site plan. At each step in this iterative process, examinees react to and modify their design based on their previous designs and remaining constraints for the design (Katz, 1994). The steps examinees take in this process become a critical aspect of assessing their level of expertise in architectural design.

Thus, the design pattern EKR serves first as an epistemic form to synthesize experience and analysis of classes of valued work in ways that will support assessment design. It is then a source of information for the task author creating such specific tasks or task models for a specific context. It provides grounding for the validity of tasks created in this manner by making explicit the link between the features, requirements, and evaluation procedures of a task and knowledge and skills that are valued in the domain.

Table 2

Portions of a Design Pattern for Problem Solving in Finite Systems

Summary	Students are presented a problem of determining the state of a system, and methods for gathering information about its state. No available diagnostic procedure is definitive; each rules in some possibilities and rules out others.
Rationale	Integrated knowledge structures, characteristic of effective problem solvers, are displayed in the ability to represent a problem, select and execute goal directed strategies, monitor and adjust performance, and offer complete, coherent explanations. In particular, problem solving to determine the state of a finite system with a set of tests requires an understanding of the procedures that can be applied to rule sets of states in or out, being able to interpret the results of the tests, synthesizing their information to determine what states are still possible after a series of tests, and being able to choose a next test that will effectively narrow the search space.
Focal knowledge, skills, and abilities	Ability to apply knowledge of system and component functioning to solve a problem. Ability to generate and elaborate explanations of task-relevant concepts. Ability to build a mental model or representation of a problem to guide solution. Ability to devise and manage problem-solving procedure.
Additional knowledge, skills, and abilities	Domain knowledge. Capability to carry out tests. Ability to coordinated problem solving with others (if required).
Characteristic task features	Statement of problem provides system, initial conditions, and set of test procedures. System with imperfectly known state (e.g., fault, unknown components). There is a finite (though possibly large) space of possibilities of the system state. Each test procedure rules some aspects of system state in and others out.
Variable task features	Level and nature of content knowledge required to solve problem. Degree of domain familiarity required. What is the fault(s)? Fault simple, compound, intermittent? Complexity of system to troubleshoot. Degree of scaffolding or prompting. Individual work, with a partner, or as a member of a group? Number of diagnostic procedures to choose from. Redundant diagnostic procedures? Overlapping diagnostic procedures?
Potential observable variables	Correctness of solution. Quality of evidence to support conclusions. Quality of explanation of task-specific concepts. Adequacy of problem representation or problem-solving plan. Appropriateness of solution strategies. Frequency and flexibility of self-monitoring. Efficiency of solution. Accuracy of deductions at each step.

Table 2 (continued)

Potential work products	Written or verbal description / identification of where the problem is or what the solution is to the problem.
	Illustration of problem solution and / or written justification for "Here's how I know."
	Verbal or written description of anticipated problem-solving approach.
	Verbal or written explanation of task-specific concepts.
	Log or observation of student actions.
	Observation data / log file / think aloud protocols during solution.
	Indication of possibilities are ruled in or out by a given test procedure.
	Indication of which possibilities are ruled in or out by all test procedures given thus far, at any point during the solution.

3.3 Knowledge Representations for Creating, Presenting, and Scoring Tasks

Once the evidentiary argument has been defined at the domain analysis and domain modeling layers, the next layers focus attention on structuring and generating actual tasks. These are the layers of the CAF, in which student, evidence and task models are articulated, and of implementation, which includes task generation. This section notes some of the roles that EKRs play in these processes.

3.3.1 Task Creation

In domain analysis the designer identifies situations in which practitioners in a domain use the KSAs of interest, and on this basis in domain modeling they outline in design patterns paradigmatic situations to elicit those KSAs (recall Table 1). In the CAF, more detailed *task models* are created. A task model is a design EKR that structures the authoring of the actual tasks that will be presented to the student. It describes the environment in which students will act to provide the data necessary to make inferences about KSAs, including the domain EKRs that will be used to provide information to the examinees, serve as work spaces and tools for them, and in which they will express the products and processes of their work. The values of the *task model variables* identified in a task model provide specifications such as the form of the work product, the materials necessary, and other features of the setting, all of which are grounded in the original assessment argument and play a variety of roles in task construction, presentation, scoring, and interpretation of results (Mislevy, Steinberg, & Almond, 2002).

Figure 7 shows a schematic diagram of the relationship between the task model variables (on the right-hand side) and the assessment implementation and delivery

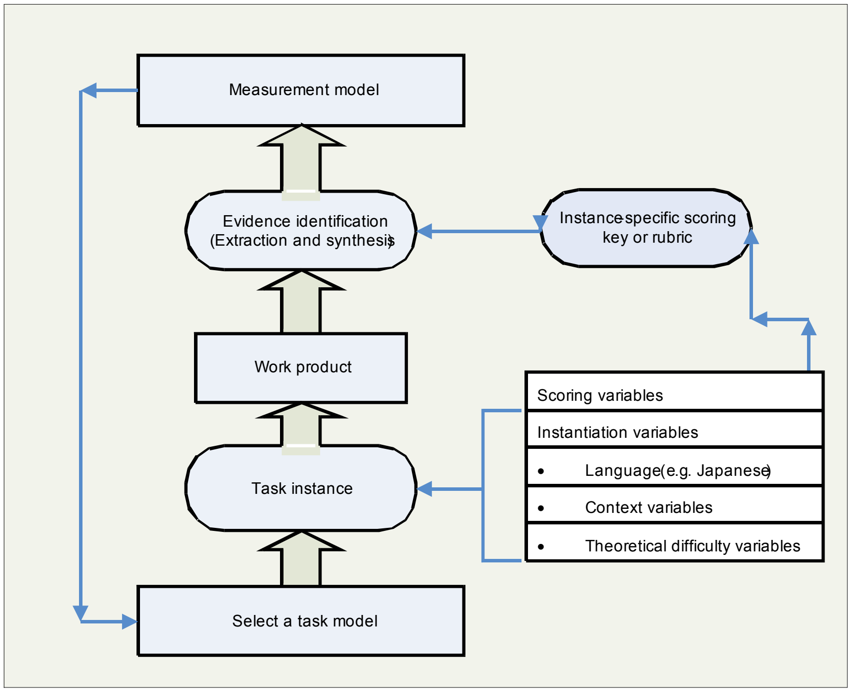
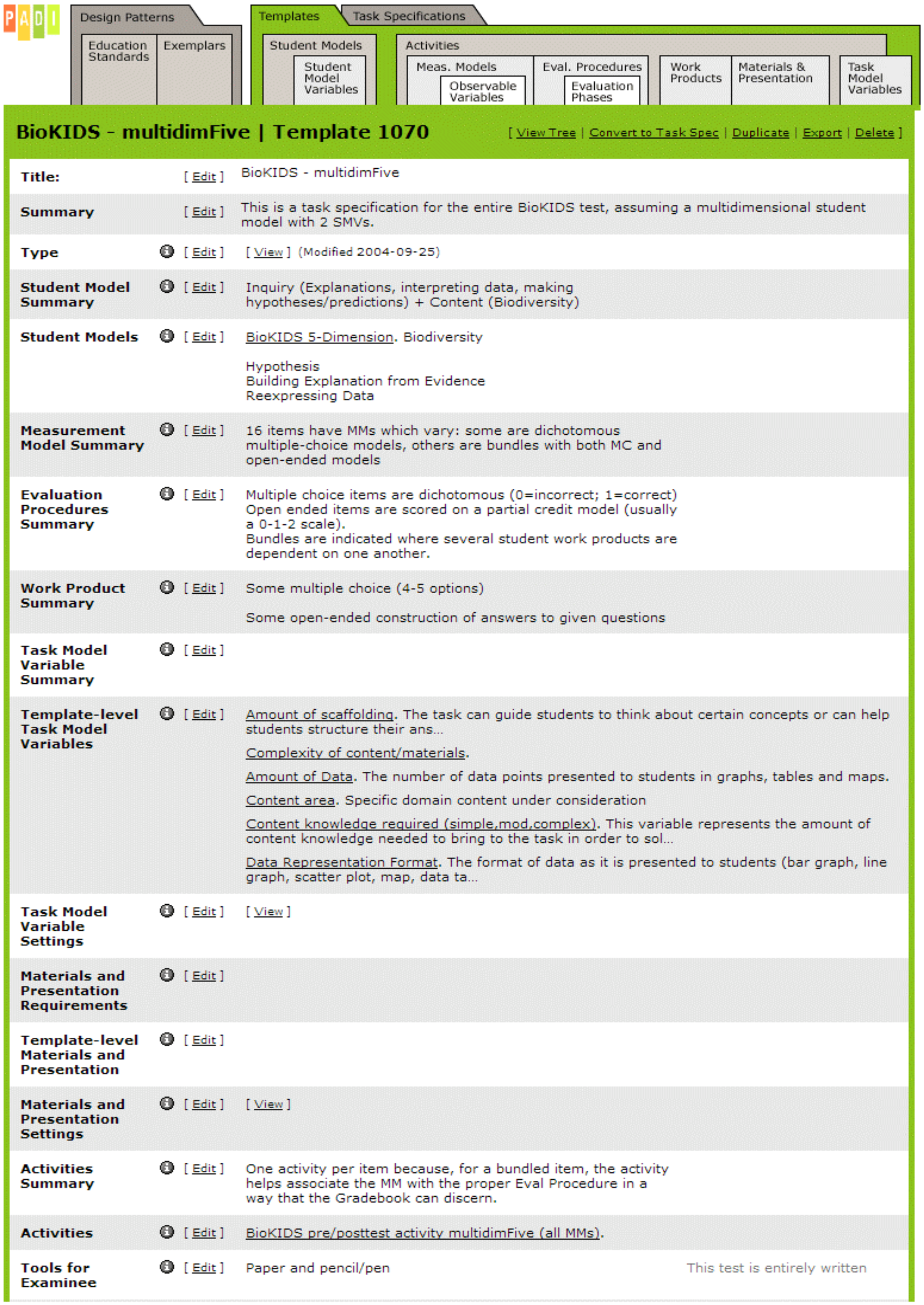


Figure 7. Schematic showing roles of task model variables.

process. The task model variables, which in this example include the language in which the task will be presented, inform the task design as well as the evidence portion of the process. As described in Mislevy et al. (2002), these attributes in the task model EKR provide information for EKRs used in task authoring, task selection, automated scoring, psychometric modeling, and score reporting.

A task model, then, is a design EKR that includes details about how the information the tasks elicit is related to other components of the assessment. The task model also explicates what particular features are necessary to include and which are variable, or optional. This general idea has been embodied in a variety of particular forms. For illustration we use here the *task template* (Riconscente, Mislevy, & Hamel, 2005) developed in the Principled Assessment Designs for Inquiry (PADI) project to describe task models more specifically. Task authors can use the template as a blueprint to create actual tasks that are grounded in the original assessment argument, without having to reconstruct this reasoning. As an example, Figure 8 shows an example of a PADI task template for BioKIDS, a project that helps students learn science inquiry (Gotwals & Songer, 2006). As can be seen in this example, the template lays out the student and measurement models in conjunction with the task model. Further, the template articulates particular materials, activities, and tools



PADI

Design Patterns | Education Standards | Exemplars

Templates | Task Specifications

Student Models | Student Model Variables

Activities | Meas. Models | Observable Variables | Eval. Procedures | Evaluation Phases | Work Products | Materials & Presentation | Task Model Variables

BioKIDS - multidimFive | Template 1070 [View Tree | Convert to Task Spec | Duplicate | Export | Delete]

Title:	[Edit]	BioKIDS - multidimFive
Summary	[Edit]	This is a task specification for the entire BioKIDS test, assuming a multidimensional student model with 2 SMVs.
Type	[Edit]	[View] (Modified 2004-09-25)
Student Model Summary	[Edit]	Inquiry (Explanations, interpreting data, making hypotheses/predictions) + Content (Biodiversity)
Student Models	[Edit]	BioKIDS 5-Dimension. Biodiversity Hypothesis Building Explanation from Evidence Reexpressing Data
Measurement Model Summary	[Edit]	16 items have MMs which vary: some are dichotomous multiple-choice models, others are bundles with both MC and open-ended models
Evaluation Procedures Summary	[Edit]	Multiple choice items are dichotomous (0=incorrect; 1=correct) Open ended items are scored on a partial credit model (usually a 0-1-2 scale). Bundles are indicated where several student work products are dependent on one another.
Work Product Summary	[Edit]	Some multiple choice (4-5 options) Some open-ended construction of answers to given questions
Task Model Variable Summary	[Edit]	
Template-level Task Model Variables	[Edit]	Amount of scaffolding. The task can guide students to think about certain concepts or can help students structure their ans... Complexity of content/materials. Amount of Data. The number of data points presented to students in graphs, tables and maps. Content area. Specific domain content under consideration Content knowledge required (simple,mod,complex). This variable represents the amount of content knowledge needed to bring to the task in order to sol... Data Representation Format. The format of data as it is presented to students (bar graph, line graph, scatter plot, map, data ta...
Task Model Variable Settings	[Edit]	[View]
Materials and Presentation Requirements	[Edit]	
Template-level Materials and Presentation	[Edit]	
Materials and Presentation Settings	[Edit]	[View]
Activities Summary	[Edit]	One activity per item because, for a bundled item, the activity helps associate the MM with the proper Eval Procedure in a way that the Gradebook can discern.
Activities	[Edit]	BioKIDS pre/posttest activity multidimFive (all MMs).
Tools for Examinee	[Edit]	Paper and pencil/pen This test is entirely written

Figure 8. A BioKIDS template in PADI design system.

associated with the task template. In this way, the task template is connected to the chain of reasoning that occurs at the domain analysis and domain modeling layers.

Another advantage of task models as design EKRs, beyond ensured instantiation of the assessment argument at the task level, is their potential for guiding reusability and adaptability of tasks to different forms or assessments. Hively et al.'s (1968) item forms (see Section 2.2.3) provide an early example of this type of design EKR. Item forms and item models provide item-level templates that can be adapted to a number of different assessments through changes in task features. Such templates allow the assessment designer the flexibility of adapting particular items types or tasks without losing the connection to the original assessment argument. This provides both efficiency and validity in task creation. Continuing with the example from the Butterfield et al. (1985) letter-series task introduced in Section 2, one can imagine using an item form approach whereby particular features of the letter-series task change (e.g., letters, pattern) to create distinct items assessing the same reasoning.

Two examples of programs that can facilitate task authoring using the idea of item templates are Mathematics Test Creation Assistant (TCA; Singley & Bennett, 2002) and the Free-Response Authoring, Delivery, and Scoring System (FRADSS; Katz, 1995). Both these tools allow for creation of multiple items from particular item models or item objects that are at a more general level of abstraction. Like PADI task templates, item forms and models support efficiency in their potential for reusability, as well as validity in their connection to the assessment argument laid out in the domain analysis phase.

EKRs play an important role in the decisions that are made about the environment around the task. For example, choice of the format (e.g., paper and pencil or computer-based; multiple choice or diagram with essay) and the materials (e.g., physical manipulatives) will all be shaped by the EKRs critical to the domain, as identified in the domain analysis phase and carried through to the task template. This aspect of task authoring is discussed in further depth in the next section on task presentation.

3.3.2 Task Presentation

There are several ways in which EKRs are important for task presentation. First, the tasks themselves can be considered EKRs. They are designed, based on the assessment argument, to be EKRs that examinees must respond or react to in some manner, producing a work product that will be subsequently evaluated. Most often, a task employs important domain EKRs to achieve this. Mathematics tasks use diagrams and mathematical notation, social studies tasks use maps and graphs, and music tests use musical notation. The Cisco Network Simulator (CNS), described in Section 5.0, utilizes symbols of network systems to assess examinees' understanding of network troubleshooting. Thus, the presentation of the tasks in this environment necessarily includes EKR symbols, formats, and manipulations that the test taker must be able to understand and use.

An example of a task as the examinee experiences it (in contrast to the task object EKR, in IMS/QTI xml form that the presentation process uses to render this view) is depicted in Figure 9. This screen shot is of a task from the Full Option Science System (FOSS) project (Delta Education, Nashua, NH; www.lawrencehallofscience.org/foss/index.html), in which science phenomena are simulated in a computer environment. For this particular example, examinees are asked to interact with the symbols on the screen to simulate electrical circuits. A number of domain EKRs are present in this example, such as the battery and switch.

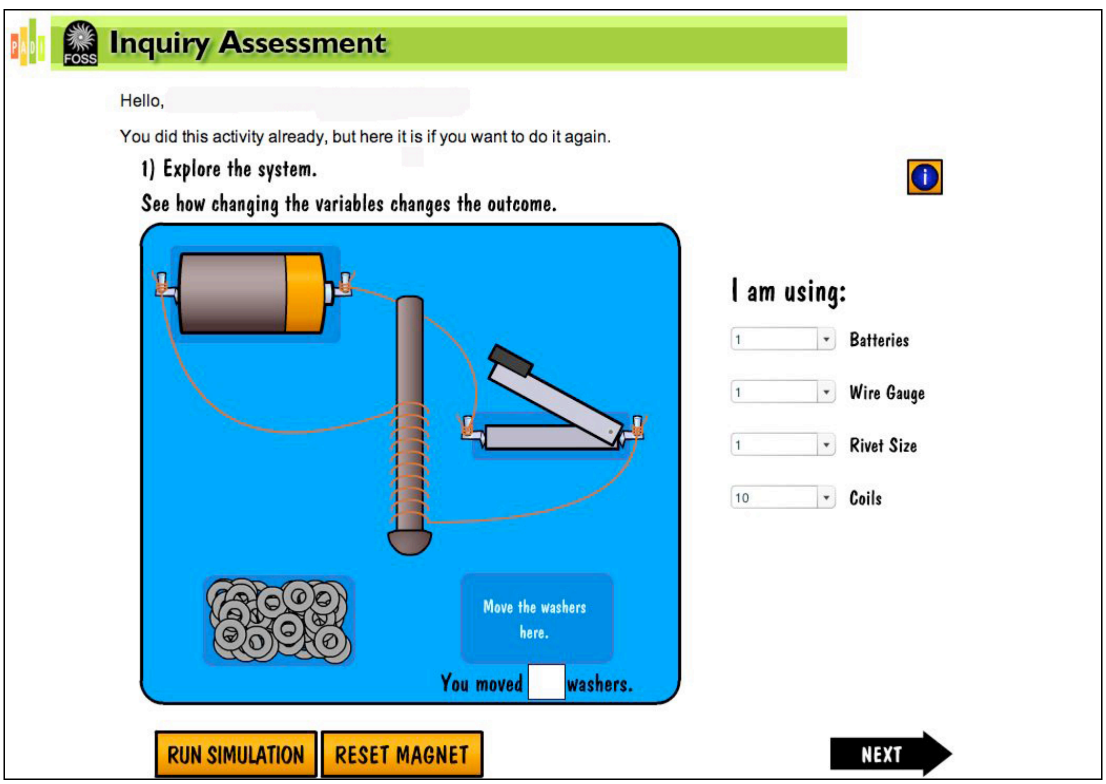


Figure 9. Prompt from FOSS/ASK simulation.

Decisions regarding what stimulus materials, resources, and levels of scaffolding will be provided to examinees are all described in the task model. These decisions are often impacted by the type of work product that is derived for a particular task. With the FOSS example, the work products produced for this item are similar in form to many others produced from the tasks created with the same template.

Just as specifications for particular tasks are articulated in the task model, the presentation model provides specifications for rendering the task in a particular environment. For example, presentation models for a computer-based assessment will be different from a presentation model for a paper-based test, even though the two may have identical task and evidence models. This flexibility is yet another example of the way in which the ECD approach enables adaptability and reusability of tasks.

Finally, design EKRs also play a role in facilitating presentation of tasks across the various aspects of assessment delivery. For example, the IMS Question and Test Interoperability (QTI) specification (IMS Global Learning Consortium, Inc., Lake Mary, FL) is an assessment EKR that allows for interchange of information between authoring tools, item banks, test construction systems and assessment delivery systems. In this way, the QTI aids in creating and presenting tasks more efficiently.

3.3.3 Task Scoring

Articulation of the student model entails identification of the student-model variables. Each student-model variable corresponds to some aspect of knowledge, skill, ability, or proficiency, presumed to drive probabilities of observable responses. Psychometric models such as classical test theory, item response theory, and cognitive-diagnosis models use probability-based methods to ground inferences about students. These probability models can be expressed in terms of EKRs called Bayes net fragments. From the perspective of ECD, the student model and the *measurement submodel* of the evidence model are EKRs that support probability-based reasoning about examinees based on evaluations of their performances. Structured around recurring evidentiary themes, Bayes net fragments can be fit together to flexibly account for different problems and different kinds of data (Conati, Gertner, & VanLehn, 2002; Mislevy, 2006; Rupp, 2002). Being able to assemble probability models in light of purposes and evolving conditions, as in simulation-based assessment, is an example of what engineers call "knowledge-based model construction" (Breese, Goldman, & Wellman, 1994). Its implementation depends on developing EKRs that encode key features of situations to guide the assembly of the Bayes net EKRs.

The *evaluative submodel* of the evidence model involves identifying and evaluating features of the examinee work product in terms of values for the observable variables that are used by the measurement submodel to update the values of student model variables. We have discussed how what examinees say, do, or make to provide evidence in assessments is often expressed in terms of domain EKRs, which examinees create, complete, transform, or interrelate—this as central to proficiency in the domain of interest. Students produce these response EKRs in their interactions with the Presentation process. They constitute the message passed to the Evidence Evaluation process.

What is important here from the perspective of representation is that the form of the work product, as an EKR, can be tuned to identifying and evaluating the features that convey evidence about the examinee's proficiencies. The work product EKR must capture traces of the cognitive processes that produced it, no matter whether the evaluation is carried out by humans or automatically (Messick, 1994). Taking advantage of developments in technology to evaluate performances requires attention not just to the form of the work product EKR and the procedures to be carried out, but to virtually every link in the chain of reasoning that comprises the assessment argument (Bennett & Bejar, 1998). To this end, Williamson, Mislevy, and Bejar's (2006) edited volume, *Automated Scoring of Complex Tasks in Computer Based Testing,* contains chapters describing various methodologies for automated scoring of EKRs from performance assessments from the perspective of ECD. Section 5 discusses automated scoring procedures used in CNS tasks, which adapt ideas from both the rule-based algorithms for scoring the log of patient management problems in the National Board of Medical Examiners' *Primum* assessment (Margolis & Clauser, 2006) and the natural language processing techniques used in automated scoring of essays (Deane, 2006).

The EKR of multiple-choice response format revolutionized testing when it was introduced in the early decades of the 20th century because it virtually eliminated judgment in evaluation, and then again in the middle of the century when machine-based scoring of multiple-choice items made standardized testing economical at vastly larger scales. Current work focuses on the use of more ecologically valid EKRs as work products, that is, examinees' performance in directly constructing, completing, or transforming domain EKRs. To accomplish this objective economically requires EKRs that in one view the examinee can interact with, but in another view support both customizable automated evaluation procedures and flexible re-use across assessment domains and purposes. Scalise and Gilford (2006) have provided a useful taxonomy of response EKRs that meet these goals, for use in computer-based testing.

4.0 Generating Examples and Mathematical Expression Tasks

This section looks more closely at two innovative task families for use in large-scale testing, through the lens of knowledge representations, namely Mathematical Expressions and Generating Examples tasks.

Bennett, Steffen, Singley, Morley, and Jacquemin (1997) developed the Mathematical Expression (ME) response type that allows presentation of any item for which the answer is a rational symbolic expression. It was created primarily to present mathematical modeling problems such as the following:

> A normal line to a curve at a point is a line perpendicular to the tangent line at the point. The equation of the normal line to the curve $y = 2x^2$ at the point (1,2) is given by _______.

Such questions typically describe a situation in one representational form (verbal), which the examinee must then translate to a symbolic form more suitable for mathematical modeling. Translating between alternative representations is key to success in any technical field. In fact, in most applied fields—mathematics, engineering, architecture, and computer programming are good examples—a key activity is to translate the verbally stated requirements of a client to the representational forms of the field because it is those representational forms that can be more effectively and efficiently operated on to satisfy client requirements (Larkin & Simon, 1987). This notion of translating verbal into more graphic or pictorial EKRs is also consistent with research demonstrating the advantages of having students construct graphic organizers or concept maps from text (e.g., Lambiotte, Dansereau, Cross, & Reynolds, 1989; Robinson, 1998).

In addition to using the ability to translate between EKRs as the object of measurement, how this response type uses EKRs in scoring is of interest. One of the attractions of ME items is that they have no single correct answer. Rather, there may be many—perhaps an infinite number—of correct answers because there are numerous ways to express the same mathematical relationship. For ME, examinee responses always share the same basic EKR—a mathematical expression. However, correct responses will almost certainly vary in their surface features. Thus, the scoring challenge is one of *mathematical paraphrase*. For example, in field trials, the following were among the correct responses examinees produced for the previous problem:

–1/4x + 9/4	(–1*x + 9)/4	–1/4 * x + (9/4)
1/4 * (9 – x)	–x/4 + 9/4	(–x + 9)/4
–.25x + 2.25	(9 – x)/4	2 – 1/4 * (x – 1)

To score answers automatically, each response is compared against a *key expression*, where that key expression can be any paraphrase of a correct answer. The comparison is done by substituting values in the examinee's expression, evaluating it, substituting the same values in the key expression, evaluating it, and subtracting one expression from the other. If the result is repeatedly zero (i.e., across many different substitutions), the examinee response is considered to be correct. ME scoring works, then, by manipulating EKRs. It does nothing more than compare the examinee's EKR to an EKR expressed in the same symbol system that may differ in its surface configuration but, if the response is right, not in semantics.

Bennett et al. (1999) also developed the Generating Examples (GE) response type where problems present constraints and ask examinees to pose one or more instances that meet those constraints but do not present enough information to determine the answer uniquely. GE questions thus relax the problem structure but unlike Simon's (1978) "ill-structured" problems, GE items give enough information to determine whether a posed solution is a member of the universe of correct responses. And, unlike ME, this universe is not composed of only paraphrases but includes quantitatively different responses.

The following is a sample item:

> If n and m are positive integers and $11n–7m=1$, what are two different possible sets of values for n and m?

The GE item class overlaps with the ME class. That is, we can pose GE items for which the answer is an expression. That expression may take many quantitatively different forms and each of those forms may, in turn, have many paraphrases. Neither the paraphrases nor the quantitatively different forms may be completely specifiable in advance.

The GE response type can also accommodate other representational forms including numbers, letter patterns, graphs, or geometric figures (see Bennett, Morley, & Quardt, 2000; Bennett, Morley, Quardt, & Rock, 2000). From the perspective of EKRs, GE can be used to pose a problem in one representational form (e.g., verbal) and collect a response in another (e.g., symbolic, numeric, figural). But in contrast to ME, GE scores responses using an EKR that differs from the examinee's production. This EKR is an executable key—computer code that tests each examinee response against the constraints expressed in the item stem. Thus, the executable key is nothing more than an alternative EKR of the problem statement, optimized for use by a computer.

For the sample item, the executable key would essentially check each response to see if it:

- Contained two pairs of values,
- Had a second pair different from the first,
- Had each member of each pair be a positive integer,
- Returned for the first pair a true result when its values are substituted for n and m in the equation, $11n–7m=1$, and
- Returned for the second pair a true result when its values are substituted for n and m in the equation, $11n–7m=1$.

For this question, then, there are multiple EKRs in play. The examinee works with verbal and symbolic representations in translating the problem, and then with symbolic and numerical ones in formulating a response. The scoring works with the numerical response and its own logical representation to process that response.

5.0 Cisco Network Simulator (CNS) Performance Assessments

The Cisco Networking Academy Program (CNAP; http://cisco.netacad.net) is a public-private partnership that teaches apprentice-level design, installation, and troubleshooting of computer networks in more than 50,000 locations ("academies") throughout the world. Since its inception, CNAP has employed hands-on, instructor administered performance (skills) examinations. When well-administered, these exams constitute a "gold standard" for assessing proficiency in the program. With more than 10,000 instructors and little local control, however, their reliability and validity can vary substantially from one site to another. The Web-based Cisco Network Simulator (CNS) provides all academies with high-quality, simulation-based performance assessment to complement local hands-on exams (Frezzo & Stanley, 2005). The CNS tasks discussed below grew out of research the NetPass project (Behrens et al., 2004, Williamson et al., 2004), which produced the initial versions of the presentation process and automated scoring procedures. This section considers the roles of EKRs in the development and use of CNS tasks.

5.1 The CNS Assessment as a Knowledge Representation

The CNS assessment is itself a knowledge representation, which coordinates information about the curriculum and instruction that occurs in the Cisco Networking Academy Program, expert-novice studies on design and troubleshooting (Williamson et al., 2004), and research on assessment design in order to provide evidence about student proficiency at the end of the program. Figure 10 shows the Web page that the students taking the CNS exam see as they work. This page contains a title, instructions for submitting the assessment, a timer, and tabs that link to key domain EKRs that will be discussed further in the next section. The affordances that appear on the Web page were designed to mirror other tools students have used, including real networking devices.

The "assessment as knowledge representation" of CNS is of paramount importance in CNAP. The widely varying quality of skills assessments across thousands of academies meant that instructional goals and performance expectations were not being clearly communicated to instructors and students. CNS was seen as a cost-effective method to provide this information widely, and to provide students with opportunities to work through the cognitive aspects of design, configurations, and troubleshooting with CNS learning tasks as well as summative exams.

5.2 Domain Analysis of EKRs in Networking

An important aspect of any domain is the EKRs that people use to represent and communicate information and to solve problems. From an assessment standpoint, learners are expected to show proficiency related to understanding and using domain EKRs. These EKRs are used to represent and communicate

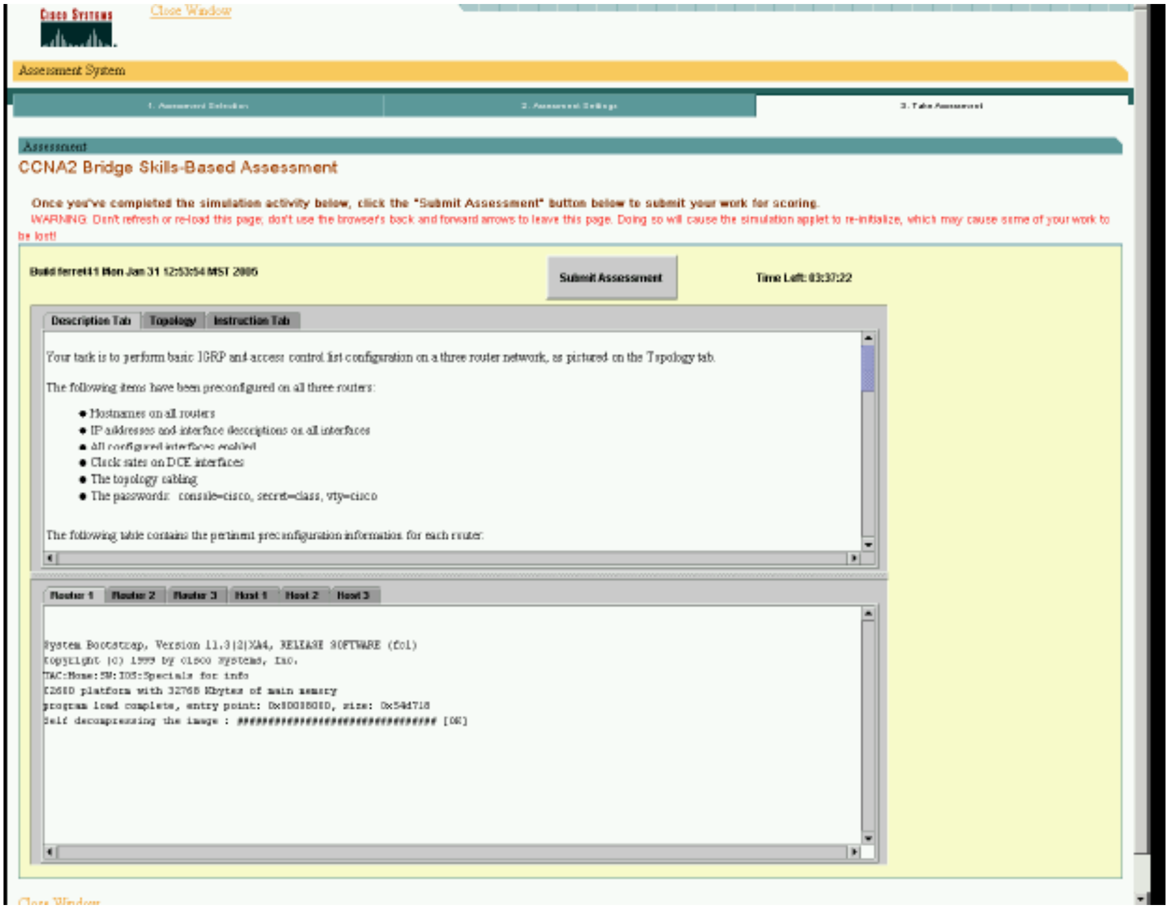


Figure 10. The CNS assessment itself acts as an EKR.

information about the domain within the assessment itself, to present information to the student, and by the student, to carry out work and express solutions. Subject matter experts analyzed CNAP curriculum materials to survey the EKRs used in instructional materials and in real-world problems at the targeted level of skills. They found usage of both general purpose EKRs, such as tables and graphs populated with networking information, and EKRs that were particular to the domain.

One example of a critical domain EKR in the domain of computer networking, and thus for CNS, is the logical topology representation. The logical topology is an abstracted map of the networking device nodes and the interconnections between those nodes (Frezzo & Stanley, 2005). Figure 11 shows an example of a logical topology, with icons representing PCs and icons representing routers. Two other domain EKRs are shown at the bottom of this figure: the command line interface

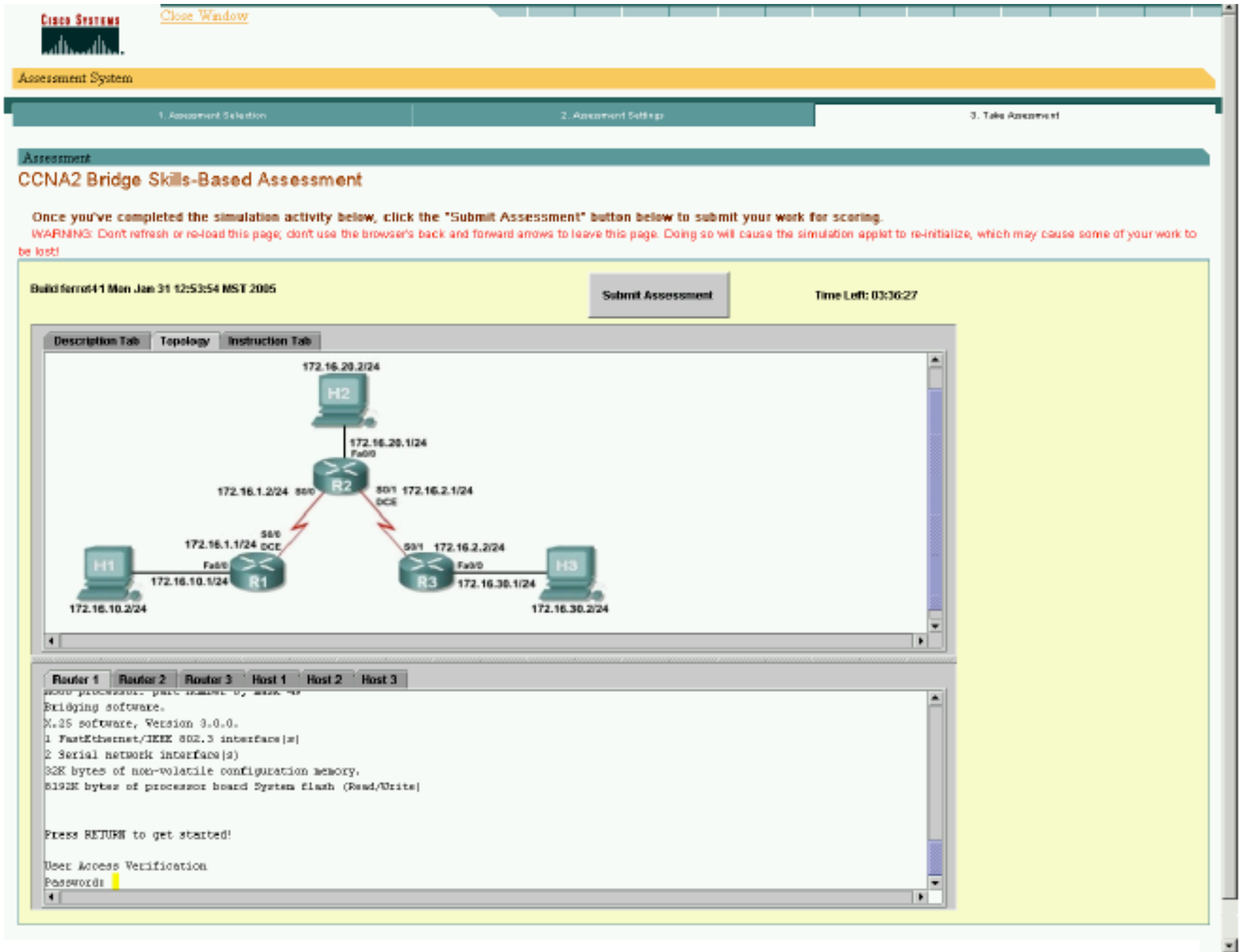


Figure 11. Two key domain EKRs, the Logical Topology (top) and the Internetwork Operating System (IOS) Command Line Interface (CLI) (bottom).

(CLI), which allows students to interact with the virtual routers, and Cisco's Internetwork Operating System (IOS), which is the control and programming language for networking the switches and routers in the logical topology EKR. As an aspect of knowledge about the domain, students are expected to be able to understand each type of EKR and how they interact in a given network—that is, what each representation tells one about the network and what it does not, where the representations share information in different forms and must therefore be consistent, and how each representation supports different aspects of reasoning about the network when troubleshooting.

5.3 Structuring Tasks Around Domain EKRs

EKRs play a central role in assessment in determining the context in which students will provide evidence of their knowledge, skills, and abilities, which, as discussed above includes knowledge and proficiencies with domain EKRs. CNS network configuration tasks illustrate the interactions between a student and the delivery system in the presentation, creation, and transformation of EKRs.

The initial presentation of the problem to the student takes the form of domain EKRs, in the form of verbal descriptions using networking terminology and concepts (Figure 10), a logical topology diagram (upper window in Figure 11; we will have more to say about the EKR that underlies the these diagrams in Sections 5.3 and 5.4), and a command line interface for configuring the devices in the network (lower window in Figure 11). The student uses the CLI to configure the network devices by means of the IOS control language, which is a symbol-system EKR through which humans and network devices communicate with other devices. We note in passing the fidelity of the CNS configuration tasks to real-world device configuration: The CNS environment uses the same IOS language and the same CLI interface as when configuring real devices remotely from a terminal, and the simulator provides the same messages back as real devices would.

As the student proceeds, two new EKRs are created and others are transformed. The transformed EKRs are the representations of the devices inside the simulator. These are symbol-system EKRs as well, representing the state of each hypothetical network device in a form the simulation program can use, to compute device responses to communicate back to the student or to modify the behavior of other devices. These are not EKRs of the learning domain, and they are not visible to the student.

The EKRs that are created are called the running configuration and the log file. The running configuration file for a router is the result of using the CLI to issue commands to change the active configuration of the router and its traffic control behavior. Figure 12 shows an example. Running configuration files are of great importance in the networking domain and serve as the key work product in CNS configuration tasks. As a work product, a running configuration file indicates the final status of the network when a student completes the problem. The log file additionally captures all of the commands a student issues during the course of the work and the responses from the network.

Running configuration files and log files are domain EKRs, produced by examinees as they interact with a (simulated) network system using the IOS symbol-system they are learning for just this purpose. As work products, they are assessment EKRs that can be operated on by the evidence identification process of

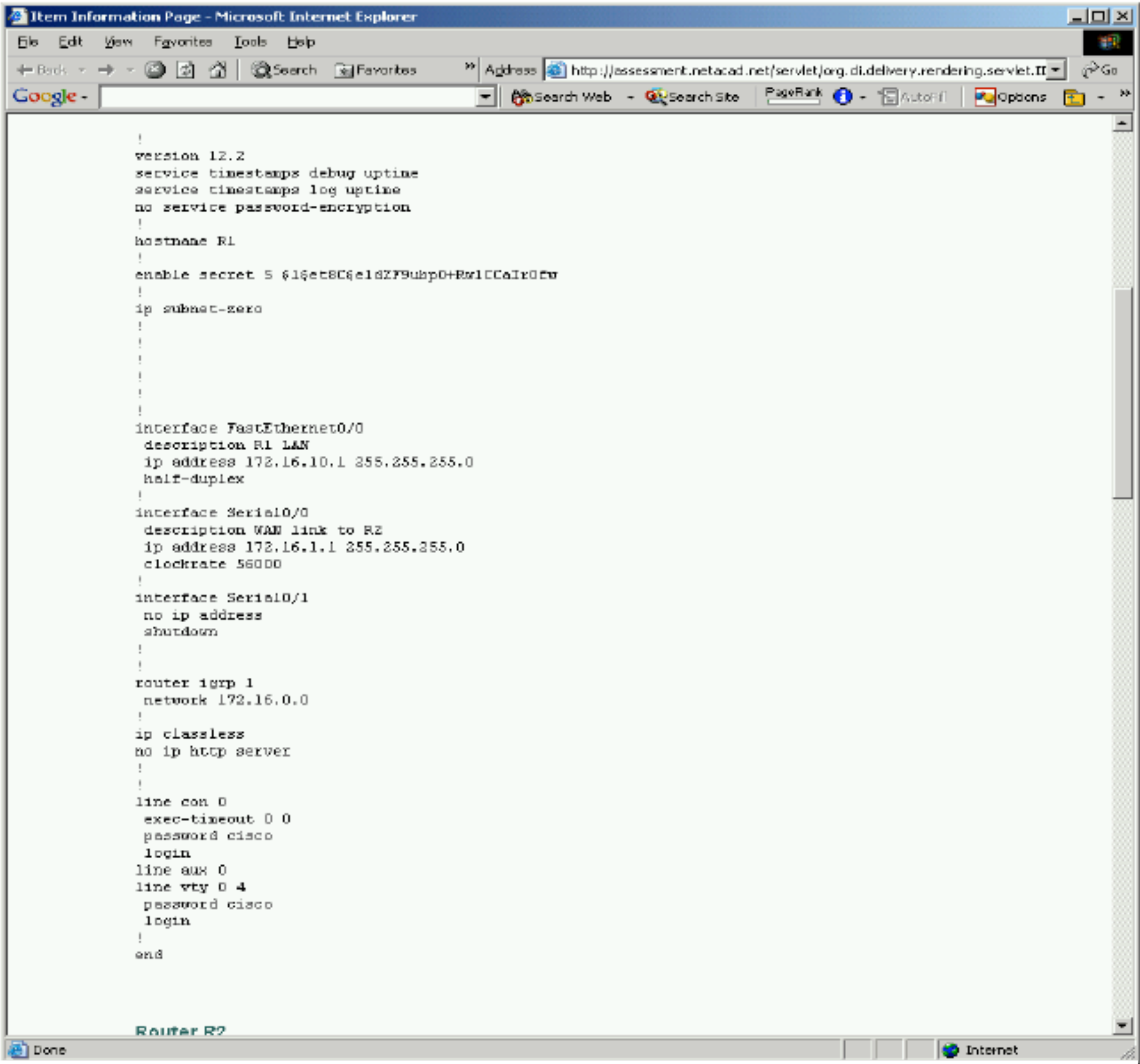


Figure 12. Router running configuration file serves as student work product.

the CNS delivery system in order to identify and evaluate evidence about student proficiency. The interplay between humans—students and instructors—and the CNS system continues in the automated scoring and reported processes discussed in Section 5.5.

5.4 Using Design EKRs to Support Task Creation

Another way in which EKRs played a crucial role in the development of the CNS is though design EKRs called design patterns, noted in Section 3. In the case of

CNS, design patterns were used to create multiple forms to ensure exam security. Design patterns that are of interest to the CNS are those related to network design, implementation, and troubleshooting tasks (Wise, 2005). More focused design patterns were developed from the Problem-Solving in a Finite System design pattern presented in Section 3, which incorporated the specialized domain knowledge and context of troubleshooting computer networks.

Tasks shells are another EKR used in CNS. CNS task shells are built around the specification of stimulus domain EKRs, key aspects of their contents in terms of task model variables, and targeted EKRs in terms of work products. Figure 13 is an example of the part of a task shell that test developers use to create instances from a family of simple network design tasks.

1. *Setting sentence*: A(n) **setting** is [create something that is a typical activity for this setting].
2. *Building size sentence*: The **setting** is **buildingLength** long.
3. *Network type sentence*: The **setting** has been asked to install a(n) **EthernetStandard** network for this [the typical activity for this setting created above].
4. *Subgroup 1 specification*: The **subgroup1** connections require a bandwidth of **bandwidthForASubgroup1**.
5. *Subgroup 2 specification*: The **subgroup2** connections require a bandwidth of **bandwidthForASubgroup2**.
6. *Subgroup 3 specification*: The **subgroup3** connections require a bandwidth of **bandwidthForASubgroup3**.
7. *Force closets sentence?*: No networking equipment can be stored in the **Subgroups123** area.
8. *Location of POP sentence*: The link to the internet is located **locationOfExternalConnection(POP)**.

Figure 13. Shell for CNS Design Task Problem Statement. Bold phrases are variables.

5.5 Using EKRs to Create Tasks and Manage Assessment Systems

CNS has revolutionized assessment, and in turn teaching and learning, in the Cisco Networking Academy Program by virtue of making high-fidelity simulations of the cognitive aspects of the domain available at low cost throughout the program over the Internet. Many EKRs, some mentioned above, are used in the design, implementation, and delivery of the tasks. In this section we point to two particular ways that EKRs are used in task design and delivery, namely task authoring and automated scoring. These leverage points concern the way that assessment designs can use technology to more efficiently create the domain EKRs examinees interact with, and capture and evaluate the EKRs they produce as work products.

As noted in Section 3, task shells like CNS's are not a new idea. They are an EKR that has been used for decades to synthesize knowledge in learning domains and knowledge about assessment, in order to improve both efficiency and validity. As also noted in Section 3, what is new is the expression of task shells in computer-based forms that facilitate the work of test developers by allowing them to work with interfaces that create task specification EKRs, and automated or semi-automated procedures that work on these forms to generate the EKRs used in assessment delivery. Figure 14, for example, shows a screen from a CNS task authoring tool in which a test developer selects stimulus and work product EKRs for troubleshooting tasks. Having specified that a topology diagram will be present in a task, a test developer then specifies and configures a network that meets the conditions indicated in the task model variables, using an interface similar the one that a student uses in a design task. The output of this interaction is another EKR, an XML file whose format can be used by the presentation process to display the topology diagram and by the simulator to create the network and govern its behavior.

CNS uses automated scoring procedures, which consist of computer programs that scan for salient features of the EKRs produced by students' interactions with the presentation process, namely configuration files, log files, and network topology XML files. The scoring rules for the running configuration in configuration tasks, for example, produce values for graded response observable variables for accuracy of the routing protocol, whether access control lists (ACLs) are assigned to appropriate devices, and the correctness of the ACL rules. Log files contain more information, about strategy use and efficiency, for example, but present greater scoring challenges because they can vary considerably from one student to another. The

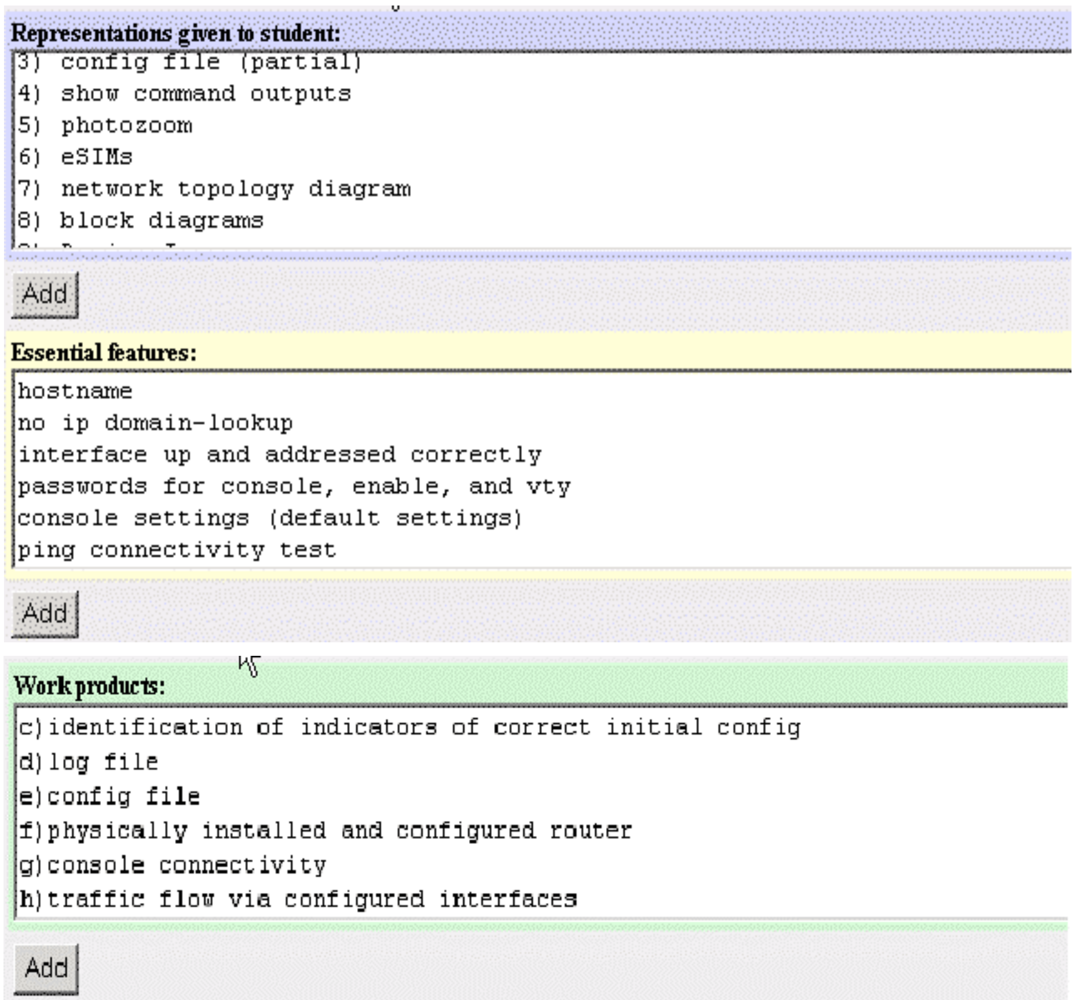


Figure 14. Screen from CNS task authoring interface.

NetPass prototype used logical rules to identify the presence or absence of key features of the interaction, systematicity of steps, and number and seriousness of errors (Williamson et al., 2004). Clauser et al. (1997) described this style of automated scoring for interactive problem solving in simulated patient management problems at the National Board of Medical Examiners. Viewing the interaction between an engineer and a network as a conversation carried out in the IOS language, DeMark and Behrens (2004) have taken a statistical language processing approach to analyzing the log files, with promising results in classifying learners along a novice-to-expert curriculum.

The resulting observable variables are an EKR that is sent the reporting process to produce the student score report. A computer program thus transforms information in the form of machine-readable EKRs containing values of observables into an EKR that summarizes results on this task for human students and instructors. The reporting process creates an accompanying EKR called an item-information page (IIP; Figure 15), which details by item how the student responded and the scoring rubric that was applied.

EKRs play roles in managing and coordinating the various aspects of building an assessment. For CNS, aspects of the curriculum, instruction, and assessment are intertwined around the domain and design EKRs. Many actors, including learners, instructors, subject matter experts, programmers, psychometricians, and automated delivery processes use the EKRs embodied in the assessment to interact and communicate with each other. Several benefits have accrued from explicating and exploiting the roles of EKRs in assessment design (DeMark, West, & Behrens, 2005).

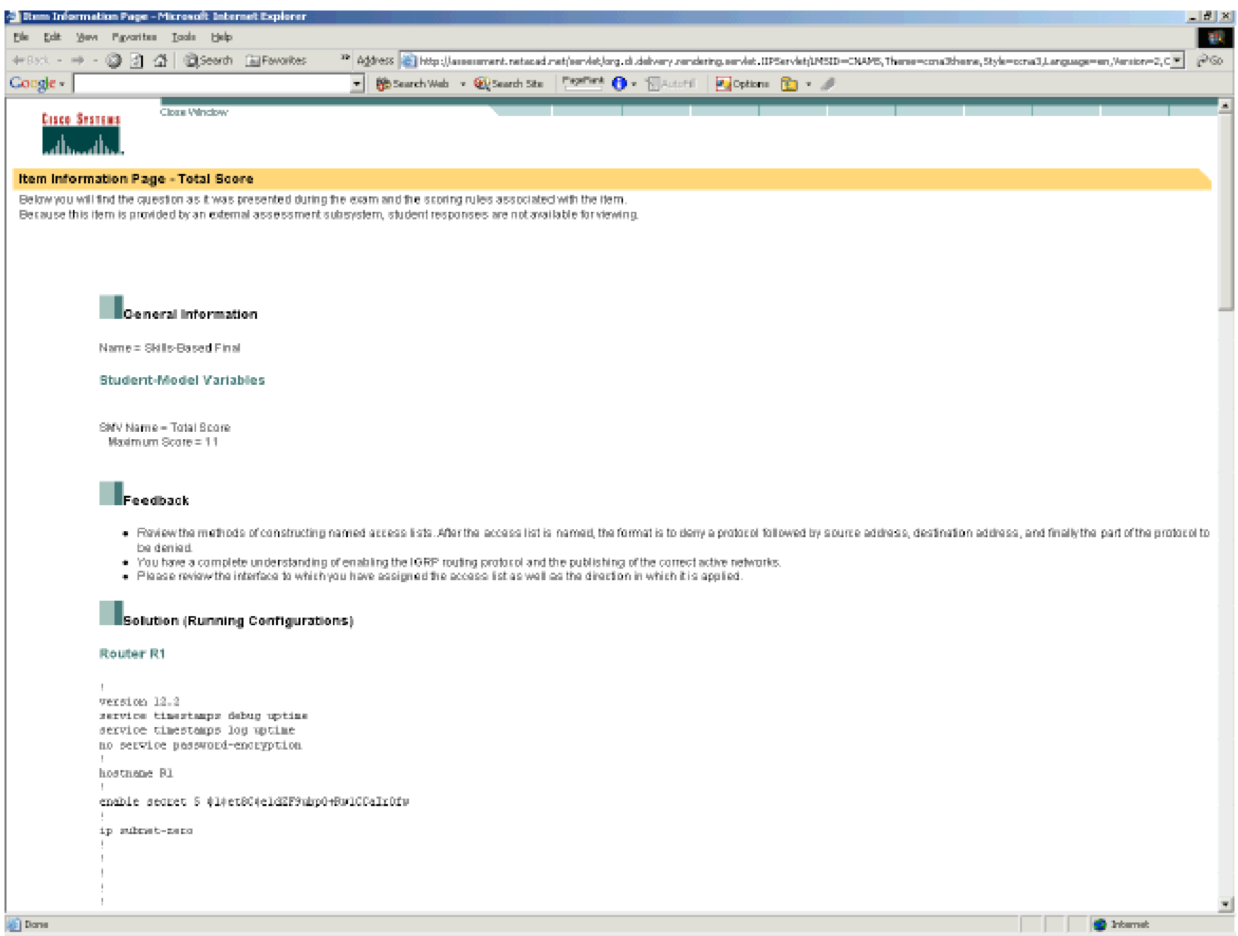


Figure 15. Item information page including student model variables, feedback, and work product.

These include improving alignment among curriculum, assessment, and instruction; providing efficiency and scalability in task and test construction; and grounding the defensibility of tasks in high-stakes tests.

In more recent work, assessment designers have extended these ideas to more local customization for instructors for learning exercises and formative assessment. A dynamic software environment called Packet Tracer allows instructors to create tasks and students to be able to use and manipulate the multiple EKRs it contains. Figure 16 shows an example with multiple interactive EKRs, including the logical topology and command line interface. The central development team used design patterns for network design, configuration, and troubleshooting to create sample tasks and a help system to assist instructors in using Packet Tracer effectively.

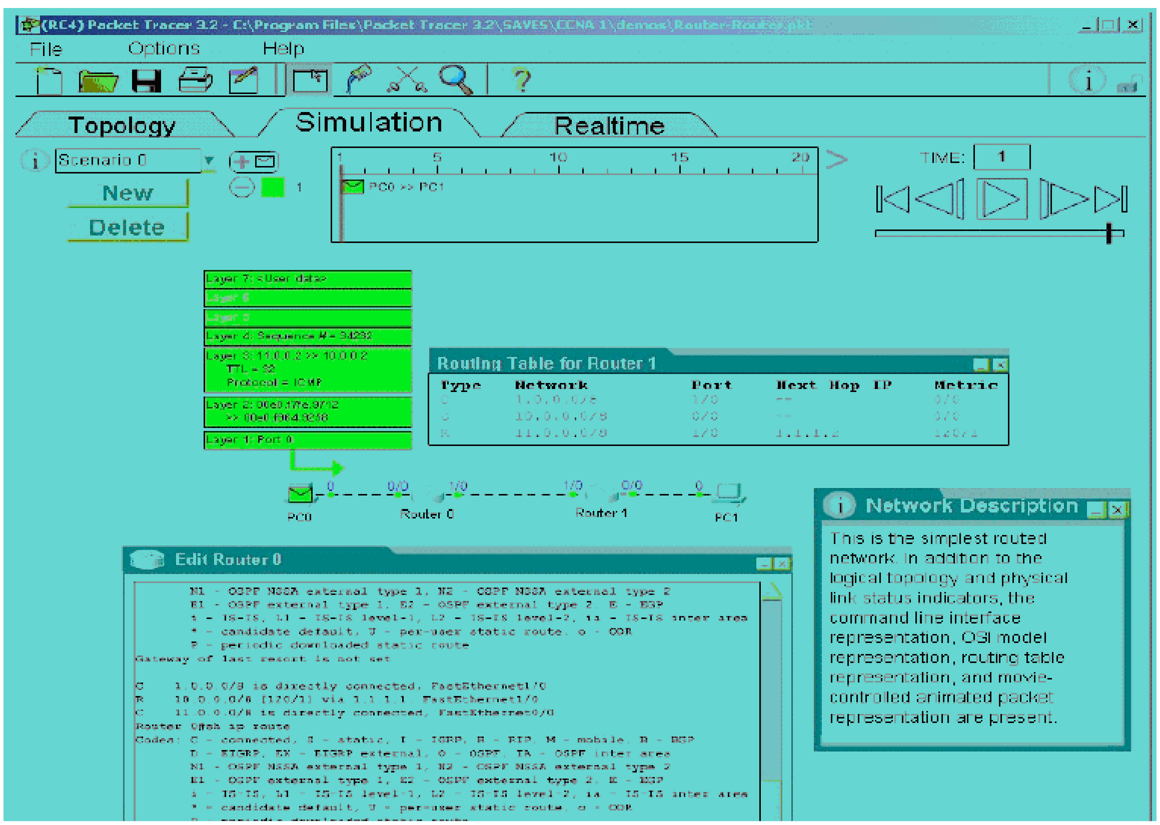


Figure 16. Packet Tracer's multiple interactive knowledge representations, including Logical Topology, IOS CLI, OSI model view, router state table, and animated "Packet Movie" mode.

6.0 Conclusion

These are exciting times in assessment, in light of rapid developments in fields that are fundamental to the conception, design, and use of educational tests. These include statistics, measurement models, technology, cognitive psychology, and learning domains. The challenge is how to put new insights to work to improve assessment. Knowledge representation plays a central role in this endeavor. Two primary ways external knowledge representations (EKRs) play a role in assessment can be described as domain EKRs and design EKRs.

Domain EKRs are representations that are used to express ideas and carry out work in domains. They concern the *what* of assessment. Insights from the cognitive, situative, and sociocultural perspectives in psychology help us understand the roles of EKRs in the development of competence, of expertise. They are critical for understanding the domain, hence pivotal points in learning and in assessment. Learning to think in their terms is a target of learning; they are used in assessment to help define the environments students work in, serve as vehicles for carrying out the work, and as they are produced, constitute work products for evaluation.

Making assessment design more efficient requires greater understanding of the assessment enterprise. Recent work on "assessment engineering" (e.g., Luecht, 2002, Mislevy, Steinberg, et al., 2003) aims to not only make the underlying principles explicit, but also to embed them in design EKRs that help assessment professionals structure, and at times automate, their work (Mislevy & Haertel, 2006). Assessment design EKRs facilitate communication between different levels of the assessment design and provide capacity for reusing assessment ideas and task components. Insights and new understandings are necessary to improve assessment; it is through EKRs that they will be put into practice.

References

Ainsworth, S. E. (1999). A functional taxonomy of multiple representations. *Computers and Education, 33*, 131-152.

Almond, R. G., Steinberg, L. S., & Mislevy, R. J. (2002). Enhancing the design and delivery of assessment systems: A four-process architecture. *Journal of Technology, Learning, and Assessment, 1*(5). Retrieved 28 June 2007 from http://www.bc.edu/research/intasc/jtla/journal/v1n5.shtml

American Association for the Advancement of Science. (2001). *Atlas of science literacy*. Washington DC: American Association for the Advancement of Science; National Science Teachers Association.

Bachman, L. F. (2003). Building and supporting a case for test use. *Language Assessment Quarterly, 2*, 1-34.

Behrens, J. T., Mislevy, R. J., Bauer, M., Williamson, D. M., & Levy, R. (2004). Introduction to evidence centered design and lessons learned from its application in a global e-learning program. *The International Journal of Testing, 4*, 295-301.

Bejar, I. I. (2002). Generative testing: From conception to implementation. In S. H. Irvine & P. C. Kyllonen (Eds.), *Item generation for test development* (pp. 199-217). Hillsdale, NJ: Lawrence Erlbaum Associates.

Belsley, D. A., Kuh, E., & Welch, R. E. (1980). *Regression diagnostics: Identifying influential data and source of collinearity.* New York: John Wiley.

Bennett, R. E., & Bejar, I. I. (1998). Validity and automated scoring: It's not only the scoring. *Educational Measurement: Issues and Practice, 17*(4), 9-17.

Bennett, R. E., Morley, M., & Quardt, D. (2000). Three response types for broadening the conception of mathematical problem solving in computerized tests. *Applied Psychological Measurement, 24*, 294-309.

Bennett, R. E., Morley, M., Quardt, D., & Rock, D. (2000). Graphical modeling: A new response type for measuring the qualitative component of mathematical reasoning. *Applied Measurement in Education, 13,* 303-322.

Bennett, R. E., Morley, M., Quardt, D., Singley, M. K., Katz, I. R., & Nhouyvanisvong, A. (1999). Generating examples: A new response type for measuring quantitative reasoning. *Journal of Educational Measurement, 36,* 233-252.

Bennett, R. E., Steffen, M., Singley, M. K., Morley, M., & Jacquemin, D. (1997). Evaluating an automatically scorable, open-ended response type for measuring mathematical reasoning in computer-adaptive tests. *Journal of Educational Measurement, 34*, 163-177.

Bentler, P. M. (1995). *EQS: Structural equation modeling software.* Encino, CA: Multivariate Software, Inc.

Bloom, B. S. (Ed.). (1956). *Taxonomy of educational objectives: The classification of educational goals. Handbook I, cognitive domain.* New York: Longman.

Bormuth, J. R. (1970). *On the theory of achievement test items.* Chicago: University of Chicago Press.

Breese, J. S., Goldman, R. P., & Wellman, M. P. (1994). Introduction to the special section on knowledge-based construction of probabilistic and decision models. *IEEE Transactions on Systems, Man, and Cybernetics, 24,* 1577-1579.

Butterfield, E. C., Nielsen, D., Tangen, K. L., & Richardson, M. B. (1985). Theoretically based psychometric measures of inductive reasoning. In S. E. Embretson (Ed.), *Test design: Developments in psychology and psychometrics* (pp. 77-148). New York: Academic Press.

Cameron, C. A., Beemsterboer, P. L., Johnson, L. A., Mislevy, R. J., Steinberg, L. S., & Breyer, F. J. (2000). A cognitive task analysis for dental hygiene. *Journal of Dental Education, 64,* 333-351.

Card, S. K., Moran, T. P., & Newell, A. (1983). *The psychology of human-computer interaction.* Hillsdale, NJ: Lawrence Erlbaum Associates.

Clauser, B. E., Ross, L. P., Clyman, S. G., Rose, K. M., Margolis, M. J., Nungester, R. J., et al. (1997). Development of a scoring algorithm to replace expert rating for scoring a complex performance-based assessment. *Applied Measurement in Education, 10,* 345-358.

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

Conati, C., Gertner, A., & VanLehn, K. (2002). Using Bayesian networks to manage uncertainly in student modeling. *User Modeling and User-Adapted Interaction, 12,* 371-417.

Davidson, F., & Lynch, B. K. 2001: *Testcraft: A teacher's guide to writing and using language test specifications.* New Haven, CT: Yale University Press.

Deane, P. (2006). Strategies for evidence identification through linguistic assessment of textual responses. In D. M. Williamson, R. J. Mislevy, & I. I. Bejar (Eds.), *Automated scoring of complex tasks in computer based testing* (pp. 313-371). Mahwah, NJ: Lawrence Erlbaum Associates.

DeMark, S. F., & Behrens, J. T. (2004). Using statistical natural language processing for understanding complex responses to free-response tasks. *International Journal of Testing, 4,* 371-390.

DeMark, S. F., West, P. A., & Behrens, J. T. (2005, April). *Explorations in domain analysis and task model specification sensitive to underlying knowledge representations.* Presented at the annual meeting of the American Education Research Association, San Francisco, CA.

Embretson, S. E. (1998). A cognitive design system approach to generating valid tests: Application to abstract reasoning. *Psychological Methods, 3,* 380-396.

Ericsson, K. A. (1996). *The road to excellence: The acquisition of expert performances, sports, and games.* Mahwah, NJ: Lawrence Erlbaum Associates.

Frezzo, D. C., & Stanley, K. (2005, April). Knowledge representations driving the design of computerized performance assessments in a complex simulated environment. In R. M. Mislevy (Chair), *Knowledge representation in assessment.* Symposium presented at the annual meeting of the American Educational Research Association, Montreal, Canada.

Gibson, J. J. (1966). *The senses considered as perceptual systems.* Boston: Houghton Mifflin.

Gitomer, D. H., & Steinberg, L. S. (1999). Representational issues in assessment design. In I. E. Sigel (Ed.), *Development of mental representation* (pp. 351-370). Hillsdale, NJ: Lawrence Erlbaum Associates.

Gotwals, A., & Songer, N. (2006). *Cognitive predictions: BioKIDS implementation of the PADI assessment system* (PADI Tech. Rep. 10). Menlo Park, CA: SRI International.

Haladyna, T. M., & Shindoll, R. R. (1989). Shells: A method for writing effective multiple-choice test items. *Evaluation and the Health Professions, 12,* 97-104.

Hively, W., Patterson, H. L., & Page, S. H. (1968). A "universe-defined" system of arithmetic achievement tests. *Journal of Educational Measurement, 5,* 275-290.

Katz, I. R. (1994). Coping with the complexity of design: Avoiding conflicts and prioritizing constraints. In A. Ram, N. Nersessian, & M. Recker (Eds.), *Proceedings of the Sixteenth Annual Meeting of the Cognitive Science Society* (pp. 485-489). Mahwah, NJ: Lawrence Erlbaum Associates.

Katz, I. R. (1995). FRADS: A system for facilitating rapid prototyping by end users. In Y. Anzai & K. Ogawa (Eds.), *Proceedings of the Sixth Annual International Conference on Human-Computer Interaction* (pp. 53-58). Amsterdam: Elsevier Science Publishers.

Katz, I. R., Lipps, A. W., & Trafton, J. G. (2002). *Factors affecting difficulty in the generating examples item type* (ETS Res. Rep. RR-02-07). Princeton, NJ: Educational Testing Service.

Kindfield, A. C. H. (1999). Generating and using diagrams to learn and reason about biological processes. *Journal of Structural Learning and Intelligent Systems, 14,* 81-124.

Lambiotte, J. G., Dansereau, D. F., Cross, D. R., & Reynolds, S. B. (1989). Multirelational semantic maps. *Educational Psychology Review, 1,* 331-367.

Larkin, J. H., & Simon, H. A. (1987). Why a diagram is (sometimes) worth ten thousand words. *Cognitive Science, 11,* 65-99.

Lehrer, R., & Schauble, L. (2002). Symbolic communication in mathematics and science: Co-constituting inscription and thought. In E. D. Amsel & J. Byrnes (Eds.), *Language, literacy, and cognitive development: The development and consequences of symbolic communication.* (pp. 167-192). Mahwah, NJ: Lawrence Erlbaum Associates.

Lewandowsky, S., & Behrens, J. T. (1999). Statistical graphs and maps. In F. T. Durso, R. S. Nickerson, R. W. Schvaneveldt, S. T. Dumais, D. S. Lindsay, & M. T. H. Chi (Eds.), *Handbook of applied cognition* (pp. 513-549). Chichester, UK: Wiley.

Luecht, R. M. (2002, April). *From design to delivery: Engineering the mass production of complex performance assessments.* Paper presented at the annual meeting of the National Council on Measurement in Education, New Orleans, LA.

Margolis, M. J., & Clauser, B. E. (2006). A regression-based procedure for automated scoring of a complex medical performance assessment. In D. M. Williamson, R. J. Mislevy, & I. I. Bejar (Eds.), *Automated scoring of complex tasks in computer based testing* (pp. 132-167). Mahwah, NJ: Lawrence Erlbaum Associates.

Markman, A. B. (1999). *Knowledge representation.* Mahwah, NJ: Lawrence Erlbaum Associates.

Messick, S. (1994). The interplay of evidence and consequences in the validation of performance assessments. *Educational Researcher, 23*(2), 13-23.

Mislevy, R. J. (2003). Substance and structure in assessment arguments. *Law, Probability, and Risk, 2,* 237-258.

Mislevy, R. J. (2006). Cognitive psychology and educational assessment. In R. L. Brennan (Ed.), *Educational measurement* (4th ed., pp. 257-305). Westport, CT: American Council on Education/Praeger Publishers.

Mislevy, R. J., & Haertel, G. D. (2006). Implications of evidence-centered design for educational testing. *Educational Measurement: Issues and Practice*, *25*(4), 6-20.

Mislevy, R., Hamel, L., Fried, R., G., Gaffney, T., Haertel, G., Hafter, A., et al. (2003). *Design patterns for assessing science inquiry* (PADI Tech. Rep. 1). Menlo Park, CA: SRI International.

Mislevy, R. J., & Riconscente, M. M. (2006). Evidence-centered assessment design: Layers, structures, and terminology. In S. Downing & T. Haladyna (Eds.), *Handbook of test development* (pp. 61-90). Mahwah, NJ: Lawrence Erlbaum Associates.

Mislevy, R. J., Steinberg, L. S., & Almond, R. G. (2002). On the roles of task model variables in assessment design. In S. Irvine & P. Kyllonen (Eds.), *Item generation for test development* (pp. 97-128). Mahwah, NJ: Lawrence Erlbaum Associates.

Mislevy, R. J., Steinberg, L. S., & Almond, R. G. (2003). On the structure of educational assessments. *Measurement: Interdisciplinary Research and Perspectives, 1*, 3-67.

Mosenthal, P., & Kirsch, I. (1989) Understanding documents: Intersecting lists. *Journal of Reading, 33,* 210-213.

Palmer, S. E. (1978). Fundamental aspects of cognitive representation. In E. Rosch & B. B. Lloyd (Eds.), *Cognition and categorization* (pp. 259-303). Hillsdale, NJ: Lawrence Erlbaum Associates.

Pinker, S. (1990). A theory of graph comprehension. In R. Freedle (Ed.), *Artificial intelligence and the future of testing* (pp. 73-126). Hillsdale, NJ: Lawrence Erlbaum Associates.

Riconscente, M. M., Mislevy, R. J., & Hamel, L. (2005). *An introduction to PADI task templates.* (PADI Tech. Rep. 3). Menlo Park, CA: SRI International.

Robinson, D. H. (1998). Graphic organizers as aids to text learning. *Reading Research and Instruction, 37*, 85-105.

Rupp, A. A. (2002). Feature selection for choosing and assembling measurement models: a building-block-based organization. *International Journal of Testing, 2,* 311-360.

Scalise, K., & Gifford, B. (2006). Computer-based assessment in e-learning: A framework for constructing "Intermediate Constraint" questions and tasks for technology platforms. *Journal of Technology, Learning, and Assessment, 4*(6) [online journal]. Retrieved 28 June 2007 from http://escholarship.bc.edu/jtla/vol4/6

Schraagen, J. M., Chipman, S. F., & Shalin, V. J. (2000). *Cognitive task analysis.* Mahwah, NJ: Lawrence Erlbaum Associates.

Sfard, A. (1998). On two metaphors for learning and the dangers of choosing just one. *Educational Researcher 27,* 4-13.

Shute, V. J., Torreano, L. A., & Willis, R. E. (2000). DNA: Toward an automated knowledge elicitation and organization tool. In S. Lajoie (Ed.), *Computers as cognitive tools: No more walls, II* (pp. 309-335). Mahwah, NJ: Lawrence Erlbaum Associates.

Simon, H. A. (1978). Information-processing theory of human problem solving. In W. K. Estes (Ed.), *Handbook of learning and cognitive processes. Vol. 5: Human information processing* (pp. 271-295). Hillsdale, NJ: Lawrence Erlbaum Associates.

Simon, H. A., & Kotovsky, K. (1963). Human acquisition of concepts for sequential patterns. *Psychological Review, 70,* 534-546.

Singley, M. K., & Bennett, R. E. (2002). Item generation and beyond: Applications of schema theory to mathematics assessment. In S. Irvine & P. Kyllonen (Eds.), *Item generation for test development* (pp. 361-384). Hillsdale, NJ: Lawrence Erlbaum Associates.

Stewart, J., & Hafner, R. (1994). Research on problem solving: Genetics. In D. Gabel (Ed.), *Handbook of research on science teaching and learning* (pp. 284-300). New York: Macmillan.

Thurstone, L. L., & Thurstone, T. G. (1941). Factorial studies of intelligence. *Psychometric Monographs, No. 2.*

Thurstone, L. L., & Thurstone, T. G. (1962). *Primary mental abilities* (Rev. ed.). Chicago: Science Research Associates.

Toulmin, S. E. (1958). *The uses of argument.* Cambridge: Cambridge University Press.

Tufte, E. (1990). *Envisioning information.* Cheshire, CT: Graphics Press.

Tukey, J. W. (1990). Data-based graphics: Visual display in the decades to come. *Statistical Science, 5,* 327-339.

Whitehead, A. N. (1958). *An introduction to mathematics* (Galaxy Book GB 18). New York: Oxford University Press.

Wiggins, G. P. (1998). *Educative assessment: Designing assessments to inform and improve student performance.* San Francisco: Jossey-Bass.

Williamson, D. M., Bauer, M., Steinberg, L. S., Mislevy, R. J., & Behrens, J. T. (2004). Design rationale for a complex performance assessment. *The International Journal of Testing, 4,* 303-332.

Williamson, D. M., Mislevy, R. J., & Bejar, I. I. (Eds.). (2006). *Automated scoring of complex tasks in computer-based testing.* Mahwah, NJ: Lawrence Erlbaum Associates.

Wilson, M. R. (2005). *Constructing measures: An item response modeling approach.* Mahwah, NJ: Lawrence Erlbaum Associates.

Wise, D. (2005, April). *Design patterns for assessing troubleshooting in computer networks.* Presented at the annual meeting of the American Education Research Association, San Francisco, CA.