DOCUMENT RESUME

ED 474 275 TM 034 816

AUTHOR	Bauer, Malcolm; Williamson, David M.; Steinberg, Linda S.; Mislevy, Robert J.; Behrens, John T.
TITLE	How To Create Complex Measurement Models: A Case Study of Principled Assessment Design.
PUB DATE	2001-04-00
NOTE	26p.
PUB TYPE	Reports - Research (143)
EDRS PRICE	EDRS Price MF01/PC02 Plus Postage.
DESCRIPTORS	Case Studies; *College Students; Computer Networks; *Computer Simulation; Design; Educational Principles; Engineers; Higher Education; Problem Solving; *Student Evaluation; Troubleshooting

ABSTRACT

In computer-based simulations, students must bring a wide range of relevant knowledge, skills, and abilities to bear jointly as they solve meaningful problems in a learning domain. To function effectively as an assessment, a simulation system must additionally be able to evoke and interpret observable evidence about targeted knowledge in a manner that is principled, defensible, and suited to the purpose at hand (e.g., licensure, achievement testing, coached practice). This study focused on the grounding for a simulation-based assessment of design and troubleshooting in the domain of computer networks. The application was designed as a prototype for assessing these skills in an instructional program, as interim practice tests and as chapter or end-of-course assessments. An evidence-centered assessment design framework was used to guide the work. An important part of this work is a cognitive task analysis, designed to tap the knowledge network engineers and students use when they design and troubleshoot networks and elicit behaviors that manifest this knowledge. There were 24 participants at 3 levels of ability in this early phase of the research. After summarizing the results of the analysis, the paper discusses implications for designing psychometric models, automated scoring algorithms, and task frameworks, and for the capabilities required for the simulation environment itself. (Contains 6 figures and 24 references.) (Author/SLD)

How To Create Complex Measurement Models:
A Case Study of Principled Assessment Design[*]

Malcolm Bauer, David M. Williamson, Linda S. Steinberg
Educational Testing Service
Robert J. Mislevy
University of Maryland
John T. Behrens
Cisco Systems

April 4, 2001

ABSTRACT

In computer-based simulations, students must bring a wide range of relevant knowledge, skills, and abilities to bear jointly as they solve meaningful problems in a learning domain. To function effectively as an assessment, a simulation system must additionally be able to evoke and interpret observable evidence about targeted knowledge in a manner that is principled, defensible, and suited to the purpose at hand (e.g., licensure, achievement testing, coached practice). This presentation concerns the grounding for a simulation-based assessment of design and troubleshooting in the domain of computer networks..The application is a prototype for assessing these skills in an instructional program, as interim practice tests and as chapter or end-of-course assessments. An evidence-centered assessment design (ECD) framework was used to guide the work. An important part of this work is a cognitive task analysis, designed to (a) tap the knowledge network engineers and students use when they design and troubleshoot networks, and (b) elicit behaviors that manifest this knowledge. After summarizing the results of the analysis, we discuss implications for designing psychometric models, automated scoring algorithms, and task frameworks, and for the capabilities required for the simulation environment itself.

---

[*] This paper is based on research conducted jointly by the Cisco Learning Institute (CLI) and Educational Testing Service (ETS). The principal investigators of the project are John Behrens, Cisco Systems, and Malcolm Bauer, ETS. Additional members of the research team whose contributions we gratefully acknowledge include, from Cisco, Tara Jennings, Sarah Demark, Ken Stanley, Bill Frey, Michael Faron; from ETS, Peggy Redman; and from Unicon, Perry Reinhert. We would like to express our gratitude to the 27 students who volunteered to work through the cases of the cognitive task analysis, and their instructors,

Mike Hogan, Dara West, K Kirkendall, and Tenette Petelinkar. We especially wish to acknowledge and thank the Cisco Learning Institute for funding the research.

INTRODUCTION

This article summarizes research by the Cisco Learning Institute (CLI) and Educational Testing Service (ETS) that lays the foundation for simulation-based assessment of network design and troubleshooting.

Advances in technology make it possible to design complex and realistic simulation environments, in which students must draw upon a wide range of relevant knowledge, skills, and abilities as they solve meaningful problems in a learning domain. But a good simulation system is not the same as a good assessment system (Melnick, 1996). To function effectively as an assessment, a system must also be able to evoke and interpret observable evidence about targeted knowledge in a way that is principled, defensible, and suited to the purpose at hand (e.g., licensure, achievement testing, coached practice). It is not an effective strategy to design a simulation system and interesting problems, and only then ask "How do you score it?" The foundation for sound assessment should be laid at the start, and guide all the design decisions throughout the development process—tasks, scoring, psychometrics, simulators—so that the many elements come together to best serve the purpose of the assessment.

An important part of this work was a cognitive task analysis, designed to (a) tap the knowledge network engineers and students use when they design and troubleshoot networks, and (b) elicit behaviors that manifest this knowledge. After summarizing the results of the analysis, we discuss implications for psychometric models, automated scoring algorithms, and task frameworks, and for the capabilities required for the simulation environment itself.

BACKGROUND

The context for our research and development is the Cisco Networking Academy Program (CNAP). The program is designed to help high schools, community colleges, and vocational schools teach students the fundamentals of computer networking. CNAP is a complete, four-semester program on the principles and practice of designing, building, and maintaining networks capable of supporting national and global organizations. Students are taught directly by teachers as well as through on-line curricular materials and activities. Assessment can likewise occur through class-based activities and on-line testing. Because this program involves instructional delivery, data collection, and data maintenance via the World Wide Web, new opportunities and challenges for educational research are emerging. Approximately 70,000 individual students in 60 countries participate in this educational program, resulting in the collection of up to 10,000 test results per day from the on-line assessment system. This 24-hour a day operation both facilitates educational research by allowing efficient large-scale data collection and also creates substantial logistical challenges.

The motivation for this project is to improve online assessment in CNAP. The assessment component that is currently available on-line to all the local academies consists of multiple-choice items that focus mainly on declarative knowledge. Dennis Frezzo, CNAP lead curriculum designer, warns CNA instructors not to depend on these assessments alone to evaluate their students:

> [The current] online assessments ... are limited checks for understanding that will help the students get ready for that [certification] exam. But to produce students who can make real networks run, your assessment must be MUCH broader and deeper than any online assessment. ... Recall a primary goal of the program -- designing, installing, and maintaining networks. Quite frankly, the Assessment Server tests and [certification] test do not adequately test the complex problem-solving and manual set of skills required to maintain actual school networks. That is why the Instructor's Guide and Training model emphasize project-based, hands-on, lab-based, troubleshooting, "authentic", journal-

> and-portfolio-based assessments -- making cables, configuring routers and switches, building networks, wiring schools, all graded by rubrics.[1]

At present, then, some of the most important aspects of CNA knowledge are only assessed locally, with comparatively much less structure and guidance from the curriculum designers and subject-matter experts at Cisco. Cisco's research has shown that students' opportunities to integrate their knowledge and bring it to bear on applied problems vary considerably from one local academy to another as a consequence. CLI and ETS are now exploring ways in which the centrally developed curriculum and assessment resources that are made available to the local academies can be extended to provide practice, feedback, and evaluation in these important skill areas. In particular, the technology for creating simulated, realistic networking environments opens the door to exercising the *cognitive,* if not physical, aspects of network design, troubleshooting, maintenance, and implementation in on-line assessment.

Within this context, the goal of our project is to build a prototype that will assess students' abilities and provide targeted feedback as students perform authentic tasks that are of educational value within a network simulation environment. CLI would like to provide feedback to teachers and students on students' knowledge of networking, their ability to solve networking problems, their ability to carry out procedures, and their misconceptions. Being able to make claims about this complex cluster of knowledge, skills, and abilities requires an assessment design process that will support an appropriately complex statistical model. As this is written, the researchers have completed the cognitive task analysis and begun to sketch assessment design elements. We report here developments thus far and describe the steps toward implementation that are now being taken.

THE EVIDENCE-CENTERED ASSESSMENT DESIGN FRAMEWORK

A simulation-based assessment must elicit behavior that bears evidence about key skills and knowledge, and it must additionally provide principled interpretations of that evidence in terms that suit the purpose of the assessment. Figure 1 sketches the basic structures of an evidence-centered approach to assessment design (Almond, Mislevy, & Steinberg, in press). Working out these variables and models and their interrelationships is a way to answer a series of questions that Sam Messick posed (1994) that get at the very heart of assessment design:

- "What complex of knowledge, skills, or other attribute should be assessed?" A given assessment is meant to support inferences for some purpose, such as a licensing decision, diagnostic feedback, guidance for further instruction, or some combination. Student-model variables describe characteristics of examinees—knowledge, skills, and abilities, which we will call *knowledge* collectively for short—upon which these inferences are to be based. The student model expresses the assessor's knowledge about an examinee's values on these variables.
- "What behaviors or performances should reveal those constructs?" An evidence model expresses how what is observed in a given task constitutes evidence about student-model variables. Observable variables describe features of specific task performances.
- "What tasks or situations should elicit those behaviors?" Task-model variables describe features of situations that will be used to elicit performance. A task model provides a framework for characterizing and for constructing situations with which a candidate will interact to provide evidence about targeted aspects of knowledge.

[[Figure 1 here—three ECD basic models]]

---

[1] Dennis Frezzo, "Assessment pains," 5/2/2000. Article in the CNAP on-line website for the instructor community.

Assessors use the student model to accumulate and represent belief about the targeted aspects of knowledge, expressed as probability distributions for student-model variables uses task models (Almond & Mislevy, 1999). They use evidence models to identify from what the examinee says or does what can provide evidence about that knowledge (Steinberg & Gitomer, 1996), and to express in a psychometric model how the evidence depends on the student-model variables (Mislevy, 1994). They use task models to design situations that can evoke required evidence (Almond, Mislevy, & Steinberg, in press). For examples of how this approach was used to develop a design rationale for simulation-based problem-solving in the domain of dental hygiene, the interested reader is referred to Mislevy, Steinberg, Breyer, Almond, and Johnson (1999; in press).

PROCEDURES

Claims and Evidence across Semesters

The work described in this presentation is part of a larger project that aims to upgrade CNAP assessment in several ways, across all four semesters. We address the portion of the project that concerns building a simulation-based prototype assessment for network design and troubleshooting only, with a focus on Semester 3. The first two semesters provide a great deal of concepts and terminology, and by Semester 3 students have enough background to begin working on some realistic problems, relatively simple at first, but increasing in complexity over the semester. The intention is for students to work on case problems throughout the semester with actual Cisco networking equipment, although as noted above the quality of the instruction and evaluation can vary from one local academy to another.

As part of the larger project, CLI and ETS developed an organized list of knowledge and skills that instructors would want a successful graduate of the CNA course to have ("claims" one would make about a successful learner). This list covered all four semesters. Sources of information included the current curriculum objectives and assessments, but also skills and knowledge not found there explicitly. An important further source of information was the set of hands-on lab exercises mentioned above. Several rounds of examination and feedback by CNA instructors and subject-matter experts (SMEs) further refined and extended the list of claims.

The next stage of work involved eliciting from instructors and SMEs the kinds of behaviors that provide evidence of the knowledge and skills listed in the claims. This involved specifying the characteristics of situations in which a student would employ the targeted knowledge and skill, and the ways he or she would act to display that knowledge. The equipment and the representational systems, and the beginning and end conditions of each, were particularly important. This work set the stage for the cognitive task analysis, which would provide the level of detail needed to define the elements of the assessment design.

The Cognitive Task Analysis

A traditional job analysis focuses on valued tasks in a domain, in terms of how often people must perform them and how important they are. A cognitive task analysis, in contrast, focuses on the knowledge people use to carry out those tasks. A cognitive task analysis in a given domain seeks to shed light on (a) essential features of the situations; (b) internal representations of situations; (c) the relationship between problem-solving behavior and internal representation; (d) how the problems are solved; and (e) what makes problems hard (Newell & Simon, 1972). With creating assessment structures as the ultimate objective, we adapted cognitive task analysis methods from the expertise literature (Ericcson & Smith, 1991) to capture and to analyze the performance of CNA students at different levels of expertise, under standard conditions, across a range of tasks.

We designed the cognitive task analysis to flesh out the assessment structures described above with the specifics of network design and troubleshooting, for the primary purpose of low-stakes (learning) assessment with supplementary feedback. As mentioned above, the set of claims we previously developed covers knowledge, skills, and abilities across all four semesters. The prototype assessment will focus on a portion of the third semester curriculum, although it will include proficiencies that develop throughout all four semesters. In consultation with CLI subject matter experts (SMEs) and assessment designers, we decided upon the smaller set of claims appropriate for the prototype. These cover 3 broad skill areas: Network troubleshooting, network configuration, and VLAN design (VLAN stands for virtual local area network).

Materials

For each of the three content areas we constructed scenarios at 3 levels of difficulty (easy, average difficulty, and difficult) for the target population of 3rd semester network academy students, creating a total of 9 scenarios. In each case, students would solve the problems using actual Cisco networking equipment which had been prepared to specified initial conditions:

- *Network troubleshooting*. The student is introduced to an existing network, with specified properties and meant to perform in predetermined ways. User reports of certain failures are provided. It is the student's task to determine the fault(s) and fix them. Figure 2 is a sample scenario.
- *Network configuration*. The student is presented user requirements and constraints for designing a local or wide area network, with specified equipment available for the job. The student designs a network in a standard representational form, then implements the network with the actual equipment. Figure 3 is a sample scenario, and Figure 4 is an acceptable solution.
- *VLAN design*. Similar to the configuration task described above, except that the requirements include the use of a virtual local area network.

[[Figure 2 here—box containing setup for easy troubleshooting scenario]]

[[Figure 3 here— box containing setup for difficult design scenario]]

[[Figure 4 here— solution diagram for difficult design scenario]]

Participants

We recruited 24 participants at 3 levels of ability (8 lower, 8 average, 8 high), as defined in terms of CNA Semester 3 students. (Note that this choice differs from traditional CTA analysis, which examines differences between novices and acknowledged domain experts. This difference, interesting though it is, was not directly pertinent to designing an assessment for Semester 3 CNA students.) Estimates of participant ability were provided by the local field instructor and corroborated with a pretest. Participants were selected from community colleges and high schools in North Carolina, Georgia, and Montana.

Method

Each participant took a pretest and solved 4 scenarios, 1 pair from each of two of the three content areas. Participants were asked to think-aloud as they solved the scenarios. After participants completed the set of four scenarios, they were given a chance to recall the rationale for their problem solving steps for each scenario using a structured retrospective protocol in which we reviewed their solutions with them. We assigned participants to particular orderings of scenarios to control for sequential learning effects and difficulty order effects to the extent possible. We chose to administer items with a difficulty level consistent with estimates (from teachers) of student ability under the assumption that we would get more useful information than if the problem difficulty levels were equally distributed across the different levels of students. We also believed we needed to learn more about performance on the design and troubleshooting scenarios than the implementation scenarios, which mostly involve following fairly standard procedures. Therefore, in general, more participants solved troubleshooting and design scenarios than implementation scenarios. (7 pairs each of troubleshooting and design, and 2 pairs for implementation, for each of the 3 participant ability levels).

Analysis

The data obtained from subjects solutions consisted of transcripts of their talk-aloud solutions, log files of actions they took on the workstation from which they configured or tested network configurations, and diagrams and calculated they produced as they solved the problems. This material was examined and discussed by a team of ten researchers, instructors, and subject matter experts from Cisco and ETS. While each solution to each scenario was unique, the researchers sought to identify recurring patterns they could use to describe sequences of actions across scenarios and across levels of expertise. These patterns would be the starting point for defining re-usable observed variables in the evidence models.

CTA Results

Troubleshooting Problems

By examining the student protocols in discussion with SMEs, we divided student troubleshooting performance into four major categories of actions:

Actions associated with gathering information about the router configuration. These include using commands that provide information about the state of a router such as all the variants of the "show" command, and commands that provide information about network connectivity ("ping" and "telnet").

Actions associated with making changes to a router configuration to fix faults. These include commands to add a clock signal, set a router protocol, setting IP address on interfaces, etc.

Actions associated with testing network behavior after fixes. There is overlap here with 1) above – but more of a focus on the commands to test network connectivity ("ping" and "telnet") rather than on the commands that provide information about a router configuration.

Actions associated with getting information about commands. These include all uses of the help system (i.e. use of "?")

In this way, we were able to analyze troubleshooting in terms of the frequency, character, and appropriateness of actions of each of these types.

We found three patterns of solutions in troubleshooting protocols. A given student did not necessarily exhibit the same pattern on all problems, and the patterns were correlated with but not identical to the levels of expertise at which they had been designated by their instructors. That is, the following are descriptions of *behaviors*, not of *students*. It will be seen that the characteristics that differentiated the patterns could be summarized by means of four categories:

1) Efficiency of procedures
2) Sequence of procedures
3) Correctness of procedures
4) Correctness of outcome

The typical patterns in troubleshooting solutions were the following:

> **Pattern A.** The student followed unsystematic inefficient troubleshooting procedures, rarely found and fixed faults, and used the help system extensively to guide their command use.
>
> *Efficiency of procedure*
>
> These solutions lack direction in their troubleshooting; that is, efficiency of procedure is low. Characteristic patterns of actions of this type were repeated use of help (?) command; repeated "show running-configuration" commands or other information gathering commands, especially

looping through different routers repeatedly asking for information without actually fixing anything; and gathering information on aspects of the network that don't need to be examined in light of the information students have obtained thus far and a reasonable understanding of the curriculum. For example, the problem statement of the T3 scenario makes it clear that the fault lies with an access control list, but some students look elsewhere, for example, with repeated show interface commands.

These solutions show inefficiency in troubleshooting as higher use of the help system, a higher volume of actions in general, and a larger number of syntax errors.

Sequence of Procedures

We also found, in these solutions, problems with the sequencing of actions. Specifically, we saw students often failed to gather information about the state of the network before making a change to the configuration, or test the network after making a change.

Correctness of procedure

As the efficiency and correctness of sequence was low in these solutions, their correctness of procedure also tended to be low.

Correctness of outcome

In solutions following Pattern A, students would fix things that were not broken (i.e. make unnecessary changes to the configuration of the router, e.g. apply a clocking signal to an interface even though one was present in the original configuration). Also in these solutions the students would fail to notice or fix all faults that were present.

Pattern B. The student found and fixed faults correctly and efficiently.

Efficiency of procedure

In these solutions, students were very directed in their troubleshooting. There was little use of help, information gathering was targeted primarily in problem areas, and there were few syntax errors.

Sequence of procedure

The sequencing of actions was appropriate. In these solutions the student would usually test the network after making a change in the configuration, and gather information about the state of the network before making a change.

Correctness of procedure

The procedures carried out in these solutions were appropriate.

Correctness of outcome

Because we expect these students to address only faults that are real, and to fix them correctly, we expect the highest level of correctness of outcome.

Pattern C. Student follows a standardized troubleshooting procedure.

Pattern C solutions are intermediate between Patterns A and B. They follow a more set procedure and usually take more steps to isolate each problem than a Pattern A solution. The student attempted to be systematic in their testing, but flailed sometimes. They correctly fixed some of the problems, but also fixed faults that were not actually present.

Efficiency of procedure

These solutions exhibited moderately directed troubleshooting. They often exhibited a strategy called serial elimination, starting at one end of the network and working through to the other end of the network rather than capitalizing on an understanding of the problem information thus far to carry out a more efficient strategy such as space splitting. There was some use of help, mostly information gathering in the problem area but also some outside it, some syntax errors, and overall a higher volume of actions than Pattern B solutions. There were occasional lapses into undirected sequences of actions, similar to Pattern A solutions, but for only portions of the problem.

Sequence of procedure

There were some problems with sequencing of actions, midway between Pattern A and Pattern B solutions.

Correctness of procedure

Because we expect these students to be moderately efficient and generally follow correct sequences, they will have a medium correctness of procedure.

Correctness of outcome

We expect these students to fix some but not all faults correctly, and to occasionally fix things that are not broken. We expect a medium level of correctness of outcome.

Design Problems

Based on the examination of CTA results, several patterns of performance emerged which distinguish varying degrees of quality in Design solutions. The basic distinction was the *Correctness of Outcome*, which focuses on whether the resulting network design is functional, correct, and complete.

In professional practice, there are two aspects to Correctness of Outcome: the Functionality of Design, which is a measure of the extent to which the network serves its intended purpose, and the Efficiency of Design, which considers aspects which affect network performance such as the number of components used, the cost of components, and the maintenance and performance implications of the selected components. Both aspects are important in professional practice, but the vast majority of professional designs are functional; they differ mainly as to their efficiency. In contrast, Efficiency is not a major factor discriminating among the student-generated designs in the relatively simpler problems addressed in Semester 3; for the most part, they either satisfy the requirements or they don't. Therefore, only the Functionality of Design has been considered in analyzing the student design solutions.

Functionality of Design can be decomposed into two parts, concerning Core Requirements and Peripheral Requirements. The Core Requirements represent the critical elements of the function of the network design, which that must be in place for the network to meet even the rudimentary intent of the network function (for example, in a VLAN design a student must have indicated the right broadcast domains). The Peripheral Requirements are elements that are required for network performance, but which are not central to the purpose of the network or the intent of the task (for example, having an appropriate number of signal repeaters to account for distance).

In these terms, then, we distinguished three patterns of performance among the solutions to the design problems in the CTA:

Pattern A. Designs that result in a network that meets all the core operational requirements and many of the peripheral operational requirements.

Correctness of Outcome (with respect to functionality of design)

The student created a design that possesses all, or very nearly all, of the core functionality called for in the problem statement. In addition, the student design also exhibits many of the peripheral functionality requirements either explicitly or implicitly requested in the problem.

Pattern B. Designs that resulted in a network that meets all, or very nearly all, of the core operational requirements.

Correctness of Outcome (with respect to functionality of design)

These solutions met all, or very nearly all, of the core operational requirements. However, there was little or no representation of the important peripheral features and operation of the network.

Pattern C. Designs that result in a network that do not meet the core operational requirements.

Correctness of Outcome (with respect to functionality of design)

These solutions produced designs that do not meet the core operational requirements. Given the failure to meet the core operational requirements, the degree of success in designing for the peripheral elements of the network is moot.

Implementation

In our review of the CTA results and discussion with SMEs, we established that the same overall framework of observations that was appropriate for troubleshooting is also appropriate for implementation. Both the way in which students perform the configuration (correctness of procedure), and the quality of the resulting configuration (correctness of outcome) are important high-level behavioral characteristics. The same types of actions (gathering information, making changes or adding to the configuration, testing, and getting help) apply to implementation as well. Compared to troubleshooting, however, students need to focus more on the configuration changes than on gathering information and testing as.

Correctness of Procedure has the same two components, Efficiency and Correctness of Sequence. Efficiency is largely the same as in troubleshooting, focusing on the overall number of actions, use of help, and correct syntax of IOS commands. There are some optimal patterns of behaviors relating to sequence of procedure, but they are less constrained than in troubleshooting; often, the order in which commands are used does not matter in configuration. It does matter in some cases, though--as examples, (1) the order of commands in constructing an access control list (ACL), and (2) the sequencing of actions that are used to configure and test connectivity (it is necessary to set up the routing protocol and interfaces on the routers before testing for connectivity).

Correctness of outcome consists of getting the configuration right – however, some components of the configuration are more difficult than others. For example ACLs are particularly difficult. In addition, there are some aspects of the configuration, such as setting passwords that students should know to do even if the scenario only mentioned higher-level requirements (security issues, in the case of passwords). The categories of behavior below reflect these differences.

Pattern A. Solutions in which students experienced substantial difficulties in configuring network devices.

Efficiency of procedure

Students relied extensively on help to find the correct commands. There were many syntax errors, and far more information was gathered than was actually needed.

Sequence of procedure

Students producing these solutions either tested the network configuration at inappropriate times or not at all. They worked on access control lists before verifying that their initial configuration is working properly.

Correctness of outcome

These protocols had many "sins of omission"; i.e., they left out important parts of the configuration. For example, when enabling a router protocol, they did not include a network (address) command and left an interface unconfigured or left out one of the addresses. Sometimes they also incorrectly configured components. For example, for the same router protocol command, they would put in the wrong network address. These types of errors can occur on many aspects of the configuration, including:

a) Naming the router
b) Setting passwords
c) Host tables
d) Descriptors on interfaces, and router tables
e) Defining protocols and static routes
f) Enabling web browser capabilities

There were often problems with the access control lists. A solution of this type may have ACL applied to the wrong interface, the wrong wild card mask set, or the statements in the wrong order.

Pattern B. Straightforward and correct solutions.

Efficiency of procedure

Protocols for these solutions showed rare use of help to find the correct commands. There were few syntax errors, and information was gathered judiciously. As a result, the total number of actions was low.

Correctness of Sequence

In these solutions, students tested the network configuration at appropriate times. They verified that their initial configuration was working properly before working on access control lists.

Correctness of outcome

Configurations were completely or almost completely correct, including ACLs.

Pattern C. Solutions obtained with some difficulty in configuring network devices.

Efficiency of procedure

These solutions showed the use of help sometimes to find the correct commands. They showed some syntax errors, but the use of "show" command variant to gather information was usually appropriate. Sometimes certain configuration commands were left out initially, and the students had to go back into router configuration mode to set the parts previously left out. As a result, the overall number of actions fell between those seen for Pattern A and Pattern B solutions to the same problem.

Correctness of Sequence

These solutions did show tests of the network configuration, but not optimal ones; a solution might show too little or too much testing. In virtually all cases, however, these solutions showed verification that the initial configuration was working properly before work on access control lists began.

Correctness of outcome

Configurations under this pattern were mostly correct, but they may have some of the errors typical of pattern A solutions--access control lists, in particular.

Network Diagram Construction

For all three areas, students were asked to sketch out a diagram of the network described in the text of each scenario before beginning the actual design, implementation, or troubleshooting. While this diagram construction task was done in the context of a larger scenario, performance was similar across content areas, and so we group all the results here. We focus only on the characteristics of the diagrams students constructed because technology constraints (described in the section below on Implications for the Presentation Process) prevent us from capturing information about the sequence of steps students use to construct the diagrams in the prototype.

Overall, the diagrams students constructed for troubleshooting varied by level of detail and accuracy. Many students were able to represent the network at the right level of abstraction: essential devices, connections between devices, and appropriate labels. Other students, however, failed to draw network connections, and /or devices. Some students put unnecessary detail into their drawing by placing devices in buildings and focusing on irrelevant physical characteristics. We describe the patterns of behavior for a single observable feature, Correctness of Outcome.

Pattern A. Essential features of the network are left out.

Correctness of outcome:

In this pattern, the network is missing key aspects such as network devices, necessary connections between devices or addressing and other key labels; for example, failure to connect two routers that must have a physical connection between them. Diagrams may also show additional incorrect elements such at the wrong devices (e.g. a hub instead of a switch) or include incorrect connections or labels (e.g. wrong IP address). The level of detail in the diagram of non-relevant characteristics may be too high (e.g. irrelevant room details).

Pattern B. All essential characteristics are present.

Correctness of outcome:

In this pattern, the key features of the network (devices, connections, address labels) are mostly present; however other irrelevant details are included as well.

Pattern C. Network represented correctly, at the correct level of abstraction.

Correctness of outcome:

In this pattern, all key features of the network (devices, connections, address labels) are present, and little, if any, irrelevant detail is included.

We may summarize these patterns by saying that Pattern A diagrams are ineffective; Pattern B diagrams are effective but not efficient; Pattern C diagrams are both effective and efficient, that is, efficacious. Efficacious diagrams are a common feature of expert representations (Kindfield, 1999).

THE STUDENT MODEL, EVIDENCE MODELS, AND TASKS

The Cognitive Tasks Analysis (CTA) was conducted to guide the construction of the Student Model, Evidence Models, and tasks for the networking prototype. The following sections discuss implications of the CTA and the preceding analyses of the domain for these assessment design elements.

Implications for the Student Model

The initial draft for the Student Model, shown as Figure 5, represents the constellation of knowledge, skill and abilities that are important for success as a student of Cisco's networking academy. The Student Model is developed to meet the need of supporting claims about students taking the examination and is developed on the basis of the results of the cognitive task analysis and expert judgment. The variables represented in the Student Model were identified and structured to reflect the dependencies of knowledge and skill in the domain.

[[Figure 5—draft of student model]]

The Student Model is composed of a number of variables representing aspects of knowledge, skill and ability. The Domain Disciplinary Knowledge variable represents the declarative knowledge of network components and operation. As declarative aspects of knowledge about networks, this represents the type of knowledge typically assessed through traditional multiple-choice questions addressing the recall of specific elements of network knowledge. There are a number of elements of declarative knowledge that are part of the Domain Disciplinary Knowledge, including the OSI network model, addressing schemes, hardware components of a network, media, IOS, protocols and security.

The Network Proficiency variable represents the procedural knowledge necessary to conduct a variety of tasks critical to successful network operations. These network operations include the tasks of Implementing (configuring) a network, Troubleshooting, Designing, Operating, and Planning a network (only Troubleshooting, Implementation and Design tasks were utilized in the CTA and will be used in the prototype assessment). As each of these network activities requires some declarative knowledge in order to conduct the procedures required to perform these tasks, there is a modeled relationship between the declarative knowledge represented in Domain Disciplinary Knowledge and the procedural knowledge required for Network Proficiency.

The Network Modeling variable is the ability of the student to represent a network structure that may facilitate their Network Proficiency in various types of tasks. The ability to produce a model of a network requires Domain Disciplinary Knowledge, which is therefore represented as a prerequisite of Network Modeling ability. Since the ability to produce a model of the network in question is thought to facilitate such tasks as troubleshooting and design, there is a modeled relationship between this ability and the Network Proficiency ability.

Implications for Evidence Models

The tasks utilized in the CTA were developed with the intent of providing opportunity for subjects to provide evidence of their level of ability and understanding in specific areas of network skill. As such, tasks were developed to address the three primary areas of emphasis in Network Proficiency (the procedural ability to conduct important procedures on a network) represented in the Student Model: Design, Implementation, and Troubleshooting. To ensure that the full range of ability may be demonstrated in the CTA, three tasks were developed for each area of emphasis; Design, Implementation, and Troubleshooting, with each task targeting a different degree of ability (high, moderate, and low) in these areas. By utilizing these tasks in the CTA the results yielded specific examples of evidence supporting the claims being made as a result of the assessment. From the CTA, we have observable examples of the types of errors made by students at various levels of expertise, as well as the elements they successfully implement. This evidence informs our development of the Evidence Model, which relates students' actions to the states of the Student Model variables. In addition, the results of the CTA suggest refinements that can be made to the Tasks to maximize their tendency to make appropriate distinctions among students to fulfill the reporting requirements and ensure that the evidential requirements for the claims can be fully met.

The Evidence Model is the vehicle by which the raw data from the student's solution is transformed into evidence about the specific claims we wish to make, and absorbed into the Student Model to update our understanding of the student knowledge, skills and abilities. On the basis of results of the CTA, the statistical portion of the Evidence Model is constructed by positing student model variables to be "parents" of observables which are meant to bear evidence about their (inherently unobservable) values.

An outline of the Evidence Model observable variables that are used to update the Student Model variables for Design, Implementation and Troubleshooting, and Network Modeling is provided in Figure 5. The italicized, composite features will be included in Bayes nets fragments as observable variables. Their values will be summaries of the lower level, non-italicized features listed below them, along the lines of Clauser et al. (1995). Each of the italicized observables will also update student-model variables for Disciplinary Knowledge student-model variables, as they are required in a given scenario. For each of these generically defined features, an algorithm will be written to parse the student's work product to identify, evaluate, and summarize the quality of the work product in that aspect. All of the observables from a given scenario will be modeled as conditionally dependent, in the manner described in Mislevy, Steinberg, Breyer, Almond, and Johnson (in press).

[[Figure 6 about here: Lists of observables informing SM variables]]

Reporting and Feedback Requirements

The prototype will provide information on student performance and abilities in two ways. The prototype will provide:

- reports on student abilities based upon a student model
- task level feedback on student performance

Reporting based upon the student model

The prototype will report on each of the proficiencies that the simulation is designed to address and are described above. The estimates of these proficiencies will come from a student model that will be updated as evidence is accumulated across tasks. Each of the proficiencies will be distinguished at five levels: utter novice, 1st semester, 2nd semester, 3rd semester, and 4th semester of the CCNA curriculum. For each proficiency variable, the student will be described in terms of probabilities of having the networking ability typical of students at a particular level (utter novice through 4th semester). These levels are defined and will be reported in more detail in terms of the claims associated with the CTA (see below). While reports of student ability can be generated at the completion of any number of tasks, the higher the number of tasks completed, the more accurate the estimates. These reports can be used by students and teachers to target gaps in students' knowledge and abilities and focus their learning activities.

Feedback at the task level

The observables described above for troubleshooting, implementation, and design are evidence from student performance that will be collected at the end of each task. Students should benefit from receiving feedback about characteristics their performance on each task (in addition to the reports on their overall abilities described above). This task level feedback will let students know what difficulties and specific misconceptions they had with that particular task, and enable them to pay attention to key areas of their performance on the next task they attempt.

We noted in the previous section that more detailed aspects of behavioral features are combined to produce the higher-level observable used to update the student model. These features, and variations of them, will be the basis of task-level feedback. For example, students will receive feedback for troubleshooting problems on the following aspects of their solutions:

- The efficiency of their performance, in terms of the volume of actions, their use of help, and their degree of difficulty using IOS commands.
- The degree to which they followed reasonable troubleshooting sequences of gathering information to identify faults, fixing faults, and testing the network.
- The faults they correctly identified, faults they missed, and parts of the configuration where they mistakenly identified faults that did not exist.
- Misconceptions and errors they had that are task specific.

IMPLICATIONS FOR TASKS

The findings of the CTA have several implications for the design of tasks to be used in the prototype. By noting the potential areas of improvement in tasks to refine the nature of the evidence, the assessment can be honed to a better tool for updating the Student Model variables of primary interest.

One aspect of the Student Model with implications for the tasks used in assessment is Domain Disciplinary Knowledge. This proficiency correlates with the Network Proficiency student model variable. As our estimate of students' Network Proficiency increases, so does our estimate of their Domain Disciplinary Knowledge. There is, however, an ambiguity in the current model. If students encounter difficulties with tasks, we cannot be certain whether those difficulties are due to deficiencies in Domain Disciplinary Knowledge or a lack of Network Proficiency. To disambiguate estimates of these proficiencies, we could obtain direct of Domain Disciplinary Knowledge through the inclusion of knowledge-based questions in the assessment (e.g. multiple-choice).

Similarly, the Evidence Model variable of Error Identification in troubleshooting tasks relies on the evidence that a student has taken action to successfully remedy an existing network error. However, the possibility exists that a student has sufficient skills to identify an error in a network but insufficient skill to actually intervene and correct the deficiency. The current evidence identification process does not attempt to disambiguate these two possible factors (error identification and error correction) contributing to the Error Identification variable. It would be possible to introduce aspects of the task that would attempt to disambiguate these elements by either (a) interrupting the natural course of student action at periodic intervals to explicitly ask the student what they currently believe the network faults to be or (b) asking the student to complete a questionnaire after the completion of the exercise asking what faults were present in the network.

Another finding from the CTA with implications for task design in the prototype is that many students spent an inordinate amount of time on the network diagrams at the expense of time devoted to the actual task of interest. A related concern is that by directing the student to construct a network diagram the task may be providing 'scaffolding' that may alter the naturalistic manner in which a student addresses a task. Since the Student Model variable of Network Modeling is a correlate of the primary variable of interest (Network Proficiency) rather than a main focus for inference and reporting, the tasks may be altered so that the diagram portion of tasks are optional segments in the prototype (for the Troubleshooting and Implementation tasks). This will still permit direct evidence for the Network Modeling but will allow greater flexibility in use of the prototype (e.g. skipping from task to task during a demonstration). Furthermore, the use of a preexisting program such as ConfigMaker may restrict the types of evidence we can collect, as compared the pencil and paper data from the CTA. Specifically, by using the preexisting software the student will have limited opportunity to provide excessive and/or irrelevant detail in the diagram (such as diagramming all of the buildings housing network components, including the location of the component in the building) that was observed during the CTA.

IMPLICATIONS FOR THE PRESENTATION PROCESS

In the prototype, the presentation process will be responsible for all direct interactions with the students, including presenting tasks, providing tools for students' use, and collecting user responses. This includes simulation capabilities, but it is far more than that: The presentation process controls the interaction of the examinee with the simulator, including what is presented and when, what actions the examinee can take in what sequences, and what work products from this interaction are captured.

The student model, evidence model, and tasks that we have sketched here have important implications for the presentation process, which take the form of requirements and constraints. We describe the requirements and constraints in two groups. The first group focuses on presentation implications for implementation and troubleshooting, while the second group focuses on presentation implications for design and diagram construction.

Implementation and Troubleshooting

Implementation and troubleshooting require a network simulator with which students can interact. At the beginning of each scenario, the simulator must be able to automatically load the configuration file associated with the scenario. It must provide an IOS command window from which students can interact with the network and allow students to perform all actions required by the scenarios. It must save all interactions with the IOS in a log file that can be accessed by other software objects in the prototype.

Design and Diagram Construction

Design and diagram construction require the use of software that allows students to construct network diagrams. It must allow for the creation of all components that are in the solutions to the scenarios. For example, if the program does not allow for the delineation of broadcast domains, we will not be able to use the VLAN scenarios. The program must produce representations of the network in which the structure of the network (devices, connections, and labels) are easily parsable by other software.

Because it is unlikely that existing programs will allow students to generate work products that will enable us to collect all needed evidence, we will have to have students answer supplemental questions and create additional work products. Network simulator and diagramming software must therefore be able to be embedded within a larger task context in which we can collect this supplemental evidence.

NEXT STEPS

As this paper is written, work is proceeding along three fronts:

- *Implementing the processes for presentation, task-level scoring and feedback, and integration of evidence across tasks via a Bayes net.* This work will take place within the four-process model for assessment delivery systems described in Almond, Steinberg, and Mislevy (in press).
- *Creating task models for troubleshooting and design tasks, then implementing modifications of the CTA tasks for presentation in the simulation environment.*
- *Creating the statistical structure for the student- and evidence models.* This process will be similar to the one described in greater detail for the scoring engine for the dental hygiene project mentioned in the introduction to this paper (Mislevy, Steinberg, Breyer, Almond, & Johnson, in press). The conditional probabilities will be initially set by expert opinion.

When these steps are completed and the results are integrated, field trials will be carried out with CNA students from across the country. We plan to first learn how to improve the interface and task presentation. When the configuration settles down and data begin to accumulate, the conditional probabilities used in the Bayes net will be refined by the estimation procedures along the lines of those described in Mislevy, Almond, Yan, and Steinberg (1999).

A FINAL COMMENT

"We live in an age when we are still more adept at gathering, transmitting, storing, and retrieving information than we are at putting this information to use in drawing conclusions from it," assert statistician Jay Kadane and evidence scholar David Schum (1996, p. xiv). This is surely the case with the complex assessment now appearing, whether in portfolios, extended projects, and computer-based simulations—all offering the promise of evidence about skills and knowledge beyond that captured in traditional assessments, yet each lying beyond the reach of traditional assessment practices.

Tasks in standardized tests are encapsulated and observations are spare mainly because historically, we could not handle more. Computers and simulation capabilities shatter this barrier—only to reveal a new one: just how to make sense of "rich and realistic data." A principled approach to this problem requires insights and methods from several disparate fields, including technology, cognitive psychology, educational measurement, and the subject domain in question—a daunting challenge. This paper has attempted to outline and illustrate just such an approach, developed in a replicable way, using a meaningful example.

REFERENCES

Almond, R.G., & Mislevy, R.J. (1999). Graphical models and computerized adaptive testing. *Applied Psychological Measurement, 23,* 223-237.

Almond, R.G., Steinberg, L.S., & Mislevy, R.J. (in press). A sample assessment using the four process framework. *CSE Technical Report.* Los Angeles: The National Center for Research on Evaluation, Standards, Student Testing (CRESST), Center for Studies in Education, UCLA.

Baxter, G. P., Elder, A. D., & Glaser, R., (1996). Knowledge-based cognition and performance assessment in the science classroom. *Educational Psychologist, 31* (2), 133-140.

Clauser, B.E., Subhiyah, R., Nungester, R.J., Ripkey, D., Clyman, S.G., & McKinley, D. (1995). Scoring a performance-based assessment by modeling the judgments of experts. *Journal of Educational Measurement, 32,* 397-415.

Collis, J.M., Tapsfield, P.G.C., Irvine, S.H., Dann, P.L. & Wright, D. (1995) The British Army Recruit Battery goes operational: From theory to practice in computer-based testing using item-generation techniques. International Journal of Selection and Assessment,3(2)

Dennis, I., Collis, J. & Dann, P. (1995). Extending the scope of item generation to tests of educational attainment. Proceedings of the International Military Testing Association, Toronto, October, 1995.

Ericsson, K. A., & Smith, J., (1991). Prospects and limits of the empirical study of expertise: An introduction. In K.A. Ericsson & J. Smith (Eds.), *Toward a general theory of expertise: Prospects and limits.* Cambridge: Cambridge University Press.

Gitomer, D.H., Steinberg, L.S., & Mislevy, R.J. (1995). Diagnostic assessment of trouble-shooting skill in an intelligent tutoring system. In P. Nichols, S. Chipman, & R. Brennan (Eds.), Cognitively diagnostic assessment (pp. 73-101). Hillsdale, NJ: Erlbaum.

Glaser, R., & Chi, M.T.H. (1988). Overview. In M.T.H. Chi, R. Glaser, & M.J. Farr (Eds.), *The nature of expertise* (pp. xv-xxxvi). Hillsdale, NJ: Lawrence Erlbaum Associates, Publishers.

Kadane, J.B., & Schum, D.A. (1996). *A probabilistic analysis of the Sacco and Vanzetti evidence.* New York: Wiley.

Kindfield, A.C.H. (1999). Generating and using diagrams to learn and reason about biological processes. *Journal of the Structure of Learning and Intelligent Systems, 14(2),* 81-124.

Melnick, D. (1996). The experience of the National Board of Medical Examiners. In E.L. Mancall, P.G. Vashook, & J.L. Dockery (Eds.), *Computer-based examinations for board certification* (pp. 111-120). Evanston, IL: American Board of Medical Specialties.

Messick, S. (1992). The interplay of evidence and consequences in the validation of performance assessments. *Educational Researcher, 23*(2), 13-23.

Mislevy, R.J., (1994). Evidence and inference in educational assessment. *Psychometrika, 59*, 439-483.

Mislevy, R.J., Almond, R.G., Yan, D., & Steinberg, L.S. (1999). Bayes nets in educational assessment: Where do the numbers come from? In K. B. Laskey & H. Prade (Eds.), *Proceedings of the Fifteenth Conference on Uncertainty in Artificial Intelligence* (437-446). San Francisco: Morgan Kaufmann Publishers, Inc.

Mislevy, R.J., & Gitomer, D.H. (1996). The role of probability-based inference in an intelligent tutoring system. *User-Modeling and User-Adapted Interaction, 5,* 253-282.

Mislevy, R.J., Steinberg, L.S., & Almond, R.G. (in press). On the roles of task model variables in assessment design. In S. Irvine & P. Kyllonen (Eds.), *Generating items for cognitive tests: Theory and practice.* Hillsdale, NJ: Erlbaum. Available online as UCLA Center for Studies in Education Research Report #500 http://www.cse.ucla.edu/CRESST/Reports/TECH500.pdf.

Mislevy, R.J., Steinberg, L.S., Breyer, F.J., Almond, R.G., & Johnson, L. (1999). A cognitive task analysis, with implications for designing a simulation-based assessment system. *Computers and Human Behavior, 15,* 335-374.

Mislevy, R.J., Steinberg, L.S., Breyer, F.J., Almond, R.G., & Johnson, L. (in press). Making sense of data from complex assessments. *Applied Measurement in Education.* Newell, A., & Simon, H.A. (1972). *Human problem solving.* Englewood Cliffs, NJ: Prentice-Hall.

O'Neil, H.F., Allred, K., & Dennis, R.A. (1997). Validation of a computer simulation for assessment of interpersonal skills. In H.F. O'Neil (Ed.), *Workforce readiness: Competencies and assessment* (pp. 229-254). Mahwah, NJ: Erlbaum.

Schum, D.A., (1994). *The evidential foundations of probabilistic reasoning.* New York: Wiley.

Steinberg, L.S., & Gitomer, D.H., (1993). Cognitive task analysis and interface design in a technical troubleshooting domain. *Knowledge-Based Systems,6,* 249-257.

Steinberg, L.S., & Gitomer, D.G. (1996). Intelligent tutoring and assessment built on an understanding of a technical problem-solving task. *Instructional Science, 24,* 223-258.

Wiley, D.E., & Haertel, E.H. (1996). Extended assessment tasks: Purposes, definitions, scoring, and accuracy. In M.B. Kane & R. Mitchell (Eds.), *Implementing performance assessments: Promises, problems, and challenges.* Mahwah, NJ: Erlbaum.

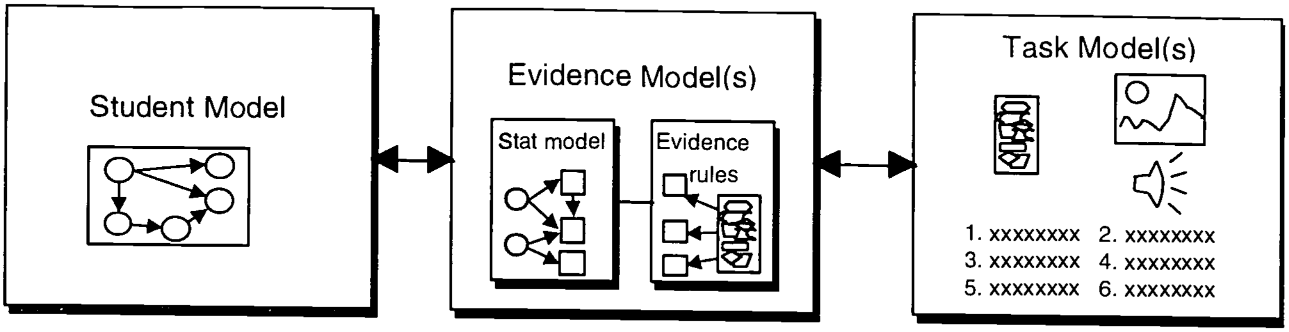

Figure 1

The Three Central Models of an Evidence-Centered Assessment Design Framework

A 3 router network is working properly. The names of the routers are listed in parenthesis. All devices attached to each router have connectivity to each other. A technician arrives to verify network health and has access to all 3 routers. After the technician leaves, all users who are attached to the first router report that they have no connectivity to devices attached to routers 2 and 3. All users attached to routers 2 and 3 are found to have connectivity to all devices except those attached to router 1.

Information about the network:

Router 1 (Idaho)
E0 192.5.5.1
E1 206.7.6.1
S0 201.100.11.1 (DCE)

Router 2 (Florida)
E0 219.17.200.1
S0 204.204.7.1 (DCE)
S1 201.100.11.2

Router 3 (Jersey)
E0 223.6.151.1
S1 204.204.7.2

All networks have a default subnet mask and use RIP.

Instructions: Problems can exist at any of the lower 3 layers of the OSI model. Before you begin troubleshooting, please create a diagram of the network. Label each interface with IP and indicate where clocking signal is located. Label all the components of your diagram. When your diagram is complete, mark the area(s) on your diagram that you think are likely to contain the source of the problem. Use your troubleshooting skills to determine the cause of the problem. When you begin to troubleshoot, answer question 2: What is the first thing you will do? Why?

Figure 2

An Easy Troubleshooting Task

Difficult Design Scenario (LAN)

A small university has decided to implement a campus wide network. The university consists of 3 buildings: Kenan, Trask, and Dobo. Each building should provide access for faculty and students on different VLANs. The university has obtained a license of *185.125.10.0.* Connection to the internet is provided from Dobo Hall.

Design a LAN with the following characteristics:

1. Subnet the license in a manner to provide connectivity for the present 50 students and 50 faculty users on each subnet in each building with an *allowance for 100% growth in users and 100% growth in buildings.* Perform the subnetting in a manner to provide for the *maximum number of subnets.*
2. The faculty and students should be on *separate VLANs.*
3. *The faculty should have access to the student VLAN but the students should not have access to the faculty VLAN.*
4. Use the first available subnets to for IP assignment. The faculty VLAN should receive the lower IP on E0.
5. Assign the router interfaces the first available IP in each subnet.
6. Assign each workstation an appropriate IP, subnet mask, and default gateway.

On the drawing it is necessary to provide only 2 faculty and 2 student workstations in each building.

[Core requirements italicized above, but not for CTA subjects]

Figure 3

A Difficult Design Task

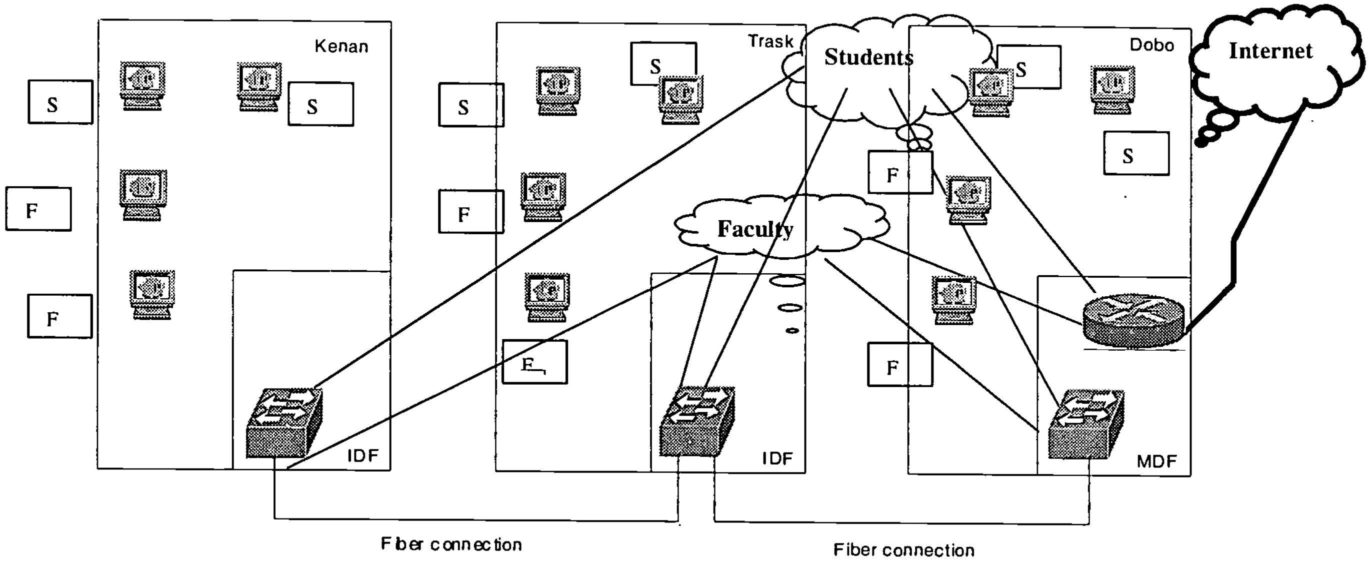

Figure 4

A Solution to the Difficult Design Task

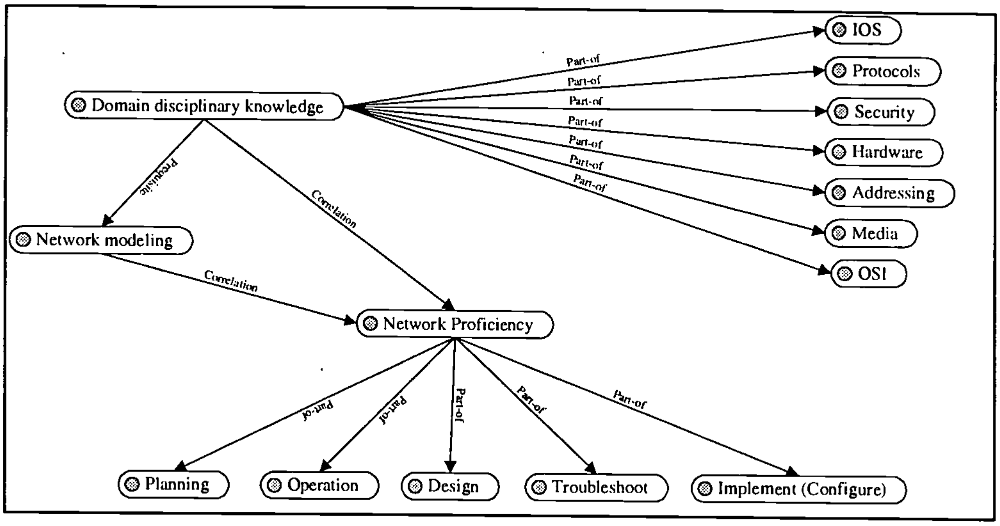

Figure 5

Student Model for Prototype Networking Assessment

Design	**Implementation**	**Troubleshooting**	**Network Modeling**
Correctness of Outcome Functionality of Design Core requirements Peripheral requirements	*Correctness of Outcome* *Correctness of Procedure* Efficiency of Procedure Help usage IOS syntax Volume of actions Procedural Sequence	*Correctness of Outcome* Error Identification Error Over-Identification *Correctness of Procedure* Efficiency of Procedure Help usage IOS syntax Volume of actions Procedural Sequence Sequence of actions Sequence of targets	*Correctness of outcome* Necessary components Extraneous components

Figure 6

Observable Variables

U.S. Department of Education
Office of Educational Research and Improvement (OERI)
National Library of Education (NLE)
Educational Resources Information Center (ERIC)

REPRODUCTION RELEASE

(Specific Document)

TM034816

I. DOCUMENT IDENTIFICATION:

Title: How to Create Complex Measurement Models: A Case Study of Principled Assessment Design

Author(s): Malcolm Bauer; David M. Williamson; Linda S. Steinberg; Robert J. Mislevy; John T. Behrens

Corporate Source:
ETS
Rosedale Road
Princeton, NJ 08541

Publication Date: April, 2001

II. REPRODUCTION RELEASE:

In order to disseminate as widely as possible timely and significant materials of interest to the educational community, documents announced in the monthly abstract journal of the ERIC system, *Resources in Education* (RIE), are usually made available to users in microfiche, reproduced paper copy, and electronic media, and sold through the ERIC Document Reproduction Service (EDRS). Credit is given to the source of each document, and, if reproduction release is granted, one of the following notices is affixed to the document.

If permission is granted to reproduce and disseminate the identified document, please CHECK ONE of the following three options and sign at the bottom of the page.

The sample sticker shown below will be affixed to all Level 1 documents	The sample sticker shown below will be affixed to all Level 2A documents	The sample sticker shown below will be affixed to all Level 2B documents
PERMISSION TO REPRODUCE AND DISSEMINATE THIS MATERIAL HAS BEEN GRANTED BY Sample TO THE EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC) 1	PERMISSION TO REPRODUCE AND DISSEMINATE THIS MATERIAL IN MICROFICHE, AND IN ELECTRONIC MEDIA FOR ERIC COLLECTION SUBSCRIBERS ONLY, HAS BEEN GRANTED BY Sample TO THE EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC) 2A	PERMISSION TO REPRODUCE AND DISSEMINATE THIS MATERIAL IN MICROFICHE ONLY HAS BEEN GRANTED BY Sample TO THE EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC) 2B
Level 1 ☑	Level 2A ☐	Level 2B ☐
Check here for Level 1 release, permitting reproduction and dissemination in microfiche or other ERIC archival media (e.g., electronic) *and* paper copy.	Check here for Level 2A release, permitting reproduction and dissemination in microfiche and in electronic media for ERIC archival collection subscribers only	Check here for Level 2B release, permitting reproduction and dissemination in microfiche only

Documents will be processed as indicated provided reproduction quality permits.
If permission to reproduce is granted, but no box is checked, documents will be processed at Level 1.

I hereby grant to the Educational Resources Information Center (ERIC) nonexclusive permission to reproduce and disseminate this document as indicated above. Reproduction from the ERIC microfiche or electronic media by persons other than ERIC employees and its system contractors requires permission from the copyright holder. Exception is made for non-profit reproduction by libraries and other service agencies to satisfy information needs of educators in response to discrete inquiries.

Sign here, → please

Signature:	Printed Name/Position/Title: Research Scientist	
Organization/Address: ETS Rosedale Road Princeton, NJ 08541	Telephone: 609-734-1303	FAX: 609-734-5880
	E-Mail Address: dmwilliamson@ets.org	Date: 3/19/03

(Over)

III. DOCUMENT AVAILABILITY INFORMATION (FROM NON-ERIC SOURCE):

If permission to reproduce is not granted to ERIC, *or*, if you wish ERIC to cite the availability of the document from another source, please provide the following information regarding the availability of the document. (ERIC will not announce a document unless it is publicly available, and a dependable source can be specified. Contributors should also be aware that ERIC selection criteria are significantly more stringent for documents that cannot be made available through EDRS.)

Publisher/Distributor:

Address:

Price:

IV. REFERRAL OF ERIC TO COPYRIGHT/REPRODUCTION RIGHTS HOLDER:

If the right to grant this reproduction release is held by someone other than the addressee, please provide the appropriate name and address:

Name:

Address:

V. WHERE TO SEND THIS FORM:

Send this form to the following ERIC Clearinghouse:

ERIC CLEARINGHOUSE ON ASSESSMENT AND EVALUATION
UNIVERSITY OF MARYLAND
1129 SHRIVER LAB
COLLEGE PARK, MD 20742-5701
ATTN: ACQUISITIONS

However, if solicited by the ERIC Facility, or if making an unsolicited contribution to ERIC, return this form (and the document being contributed) to:

ERIC Processing and Reference Facility
4483-A Forbes Boulevard
Lanham, Maryland 20706

Telephone: 301-552-4200
Toll Free: 800-799-3742
FAX: 301-552-4700
e-mail: ericfac@inet.ed.gov
WWW: http://ericfacility.org

-088 (Rev. 2/2001)