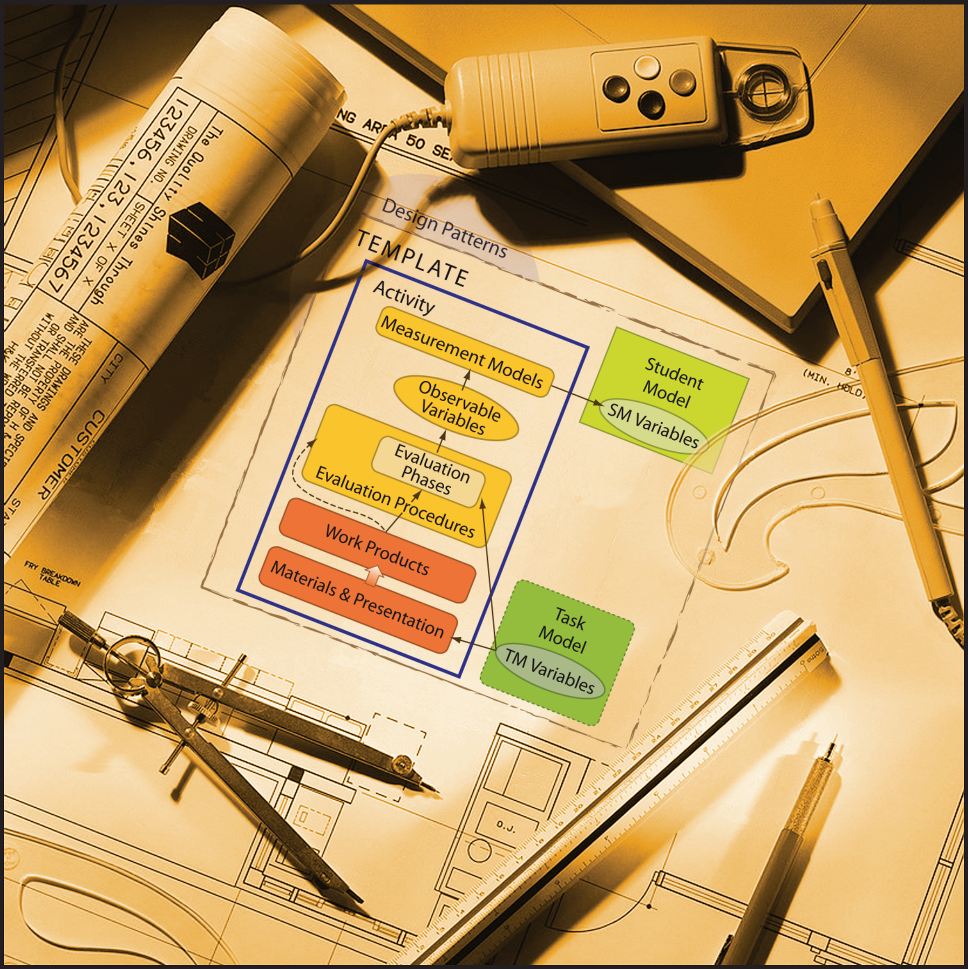


PADI Technical Report 7 | November 2005

Constructing PADI Measurement Models for the BEAR Scoring Engine

PADI | Principled Assessment Designs for Inquiry

Cathleen A. Kennedy, University of California, Berkeley

Report Series Published by SRI International

SRI International
Center for Technology in Learning
333 Ravenswood Avenue
Menlo Park, CA 94025-3493
650.859.2000
http://padi.sri.com

PRINCIPLED ASSESSMENT DESIGNS FOR INQUIRY

TECHNICAL REPORT 7

Constructing PADI Measurement Models for the BEAR Scoring Engine

Prepared by:
Cathleen A. Kennedy, University of California, Berkeley

Acknowledgments

This material is based on work supported by the National Science Foundation under grant REC-0129331 (PADI Implementation Grant). I also would like to thank those who provided thoughtful feedback to help make this report more accessible to readers with differing backgrounds in science education, assessment, and measurement: Han Bao, Karen Draney, Geneva Haertel, Larry Hamel, Andrea Hayes, Bob Mislevy, Mark Wilson, and Futoshi Yumoto.

Disclaimer

Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author and do not necessarily reflect the views of the National Science Foundation.

CONTENTS

FIGURES

TABLES

ABSTRACT

Representing complex science inquiry tasks for item response modeling (IRM) presents a number of challenges for the assessment designer. Typically, such tasks provide evidence of multiple aspects of learning and involve sequential or interdependent responses. The BEAR Scoring Engine is introduced as a software tool to compute proficiency estimates for such tasks. In addition, reusable data structures containing measurement model specifications are presented as a technique to enhance internal assessment coherence, improve the interpretability of statistical analyses of student responses, and speed the process of developing new assessment tasks that are consistent with articulated learning goals. This report begins by defining an assessment system as comprising task design and delivery components. The Principled Assessment Designs for Inquiry (PADI) design system is then introduced and positioned within the assessment system framework. Next, we describe the role of measurement models in operationalizing the manner in which inferences are drawn from observations and interpreted in an assessment system. Connections among the task design, student work products, evaluation, and inferential reasoning are highlighted, and the BEAR Scoring Engine is presented as one example of how multidimensional item response modeling can be integrated with the PADI design system. Finally, several examples of assessment tasks common to science inquiry are developed to illustrate how they would be implemented with this particular scoring engine.

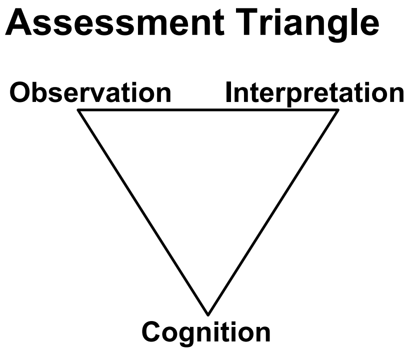
1.0 Introduction

Advances in science education, cognitive science, educational measurement, and computer technologies have matured to the point that powerful tools are emerging to support the development of high-quality assessments in science inquiry. In 2001, the National Research Council (NRC) Committee on the Foundations of Assessment published *Knowing What Students Know: The Science and Design of Educational Assessment* to integrate developments in our understanding of human learning with innovations in assessment practice. The report concludes:

Every assessment, regardless of its purpose, rests on three pillars: a model of how students represent knowledge and develop competence in the subject domain, tasks or situations that allow one to observe students' performance, and an interpretation method for drawing inferences from the performance evidence thus obtained. (NRC, 2001, p. 2)

The NRC assessment triangle, shown in Figure 1, is a model of the essential connections and dependencies present in a coherent and useful assessment system. Meaningful connections among the three vertices—cognition, observation, and interpretation—are deemed essential for assessment to have a positive impact on learning. Thus, assessment activities (the *observation* vertex) must be aligned with the knowledge and cognitive processes (the *cognition* vertex) one wishes to affect through the instructional process, and the scoring and interpretation of student work (the *interpretation* vertex) must reflect sound measures of the same knowledge and cognitive processes.

Figure 1. NRC assessment triangle.



One example of the need for more and better science assessment comes from the *No Child Left Behind Act of 2001* (NCLB). As the deadline approaches for putting state science assessments into place for the 2007-08 academic year, assessment developers are under increasing pressure to produce high-quality tests that conform to state and national science standards. Increasingly, these standards include science inquiry skills, as well as expectations for content knowledge (e.g., Connecticut DOE, 2005; NRC, 1996; Wisconsin DPI, 2005). Educators are called to develop not only new large-scale assessments but also new classroom assessments that can be used diagnostically so that steps can be taken to improve student outcomes. As the U.S. Department of Education puts it:

> States are to develop rigorous academic standards . . . and those standards should drive the curriculum, which, in turn, must drive instruction. Annual statewide assessments will be aligned with the curriculum to provide an external, independent measure of what is going on in the classroom, as well as an early indicator showing when a student needs extra help. (U. S. Department of Education, n.d.)

A key challenge in educational measurement, unlike in measuring height or weight, is making inferences about cognitive processes, such as knowledge, that are not directly observable and to do so from a limited set of observations. Efforts to develop science assessments that reveal complex understandings of science that go beyond recitation of isolated science facts can be resource intensive. Comprehensive and coherent sets of assessment items must be developed, student responses elicited from these items must be evaluated in a consistent manner, and then this evidence must be interpreted to inform classroom teaching and support individual student learning. A complicating factor faced by assessment developers is differences in science standards from state to state and at the national level. In addition, many standards are broadly defined, do not identify specific competencies linked to the types of activities that students should engage in to produce evidence of those competencies, and do not guide how evaluation of student work should proceed to provide useful inferences about competence and learning needs. Faced with these challenges, assessment developers need support in designing new high-quality assessments.

To address this need, the NSF-funded Principled Assessment Designs for Inquiry (PADI) project is developing technologies to facilitate the design and development of assessment tasks that are consistent with the model of high-quality assessment advanced by the NRC. In particular, PADI researchers have developed a software application to assist developers in designing and building assessment tasks from reusable components. The practice of reusing components helps ensure consistency throughout an assessment system and also speeds the development process. The system takes advantage of advances in educational measurement by anticipating the need for multidimensional item response modeling (IRM) to draw inferences from the evidence generated from student responses. The use of multidimensional IRM can enhance the interpretability of assessment evidence by relating it to multiple learning goals.[1] It also can improve the reliability and validity of comparisons made over time and among student groups, particularly when students do not complete the same assessment tasks, through the use of consistent scaling at the task level (Rasch, 1960; Wright, 1993).

The assessment design framework developed in the PADI project is based on the evidence-centered assessment design (ECD) model developed by Almond, Steinberg, and Mislevy (2002). Key features of the PADI design system include *design patterns*, *task templates*, and design tools. *Design patterns* capture assessment arguments describing how the alignment of cognitive objectives for performance, observations, and interpretation are operationalized for a specific (narrow or broad) domain of knowledge. *Task templates* lay out the components of an assessment task and specify the chain of reasoning from gathering evidence to drawing inferences. Design tools assist developers in designing and constructing assessment tasks that will produce interpretable results. The framework and tools developed for this project could be

[1] By "interpretability" we mean facilitating the alignment of an assessment measure, such as proficiency, with an assessment purpose or goal, such as improvement in a particular domain of knowledge or a specific level of competence. The measure that is produced from an assessment should make sense in the context of why the assessment was administered.

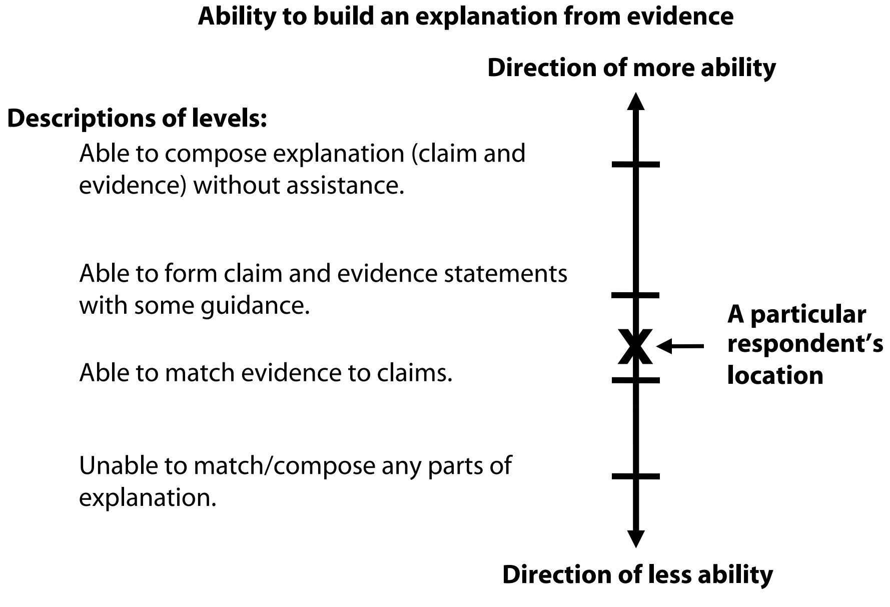
applied to assessment in any subject area, although the focus of the current grant is on the assessment of science inquiry.

This report begins by defining an assessment system as comprising task design and delivery components. The PADI design system is then introduced and positioned within the assessment system framework. Next, we describe the role of measurement models in operationalizing the manner in which inferences are drawn from observations and interpreted in an assessment system. Connections among the task design, student work products, evaluation, and inferential reasoning are highlighted, and the BEAR Scoring Engine is introduced as one example of how multidimensional IRM can be integrated with the PADI design system. Finally, several examples of assessment tasks common to science inquiry are developed to illustrate how they would be implemented with this particular scoring engine.

2.0 Assessment Design and the Four-Process Model

An assessment comprises a series of tasks that are administered to a respondent to elicit evidence about particular aspects of his or her knowledge, skill, or ability. These targeted cognitive processes are referred to as Student Model Variables, and the collection of variables for a given assessment purpose is referred to as a Student Model. A Student Model Variable can be represented as a continuum from having less of the knowledge, skill, or ability to having more of it, and although a particular assessment may target a narrow range on the continuum, the Student Model Variable itself is theoretically without bounds. Examples of Student Model Variables in the domain of science inquiry include “ability to build an explanation from evidence,” “creating hypotheses and predictions,” and “interpreting data.” Figure 2 is a graphical representation of the “ability to build an explanation from evidence” Student Model Variable showing descriptions of qualitatively different levels of ability. When we speak of measuring, we mean identifying the location of a particular respondent at some point on the Student Model Variable continuum (shown as an X in Figure 2). Aligning all items and respondents on the same continuum enables valid and reliable comparisons among respondents at a specific point in time, and for a given respondent at different time points (Embretson, 1996; Wright, 1968, 1977).

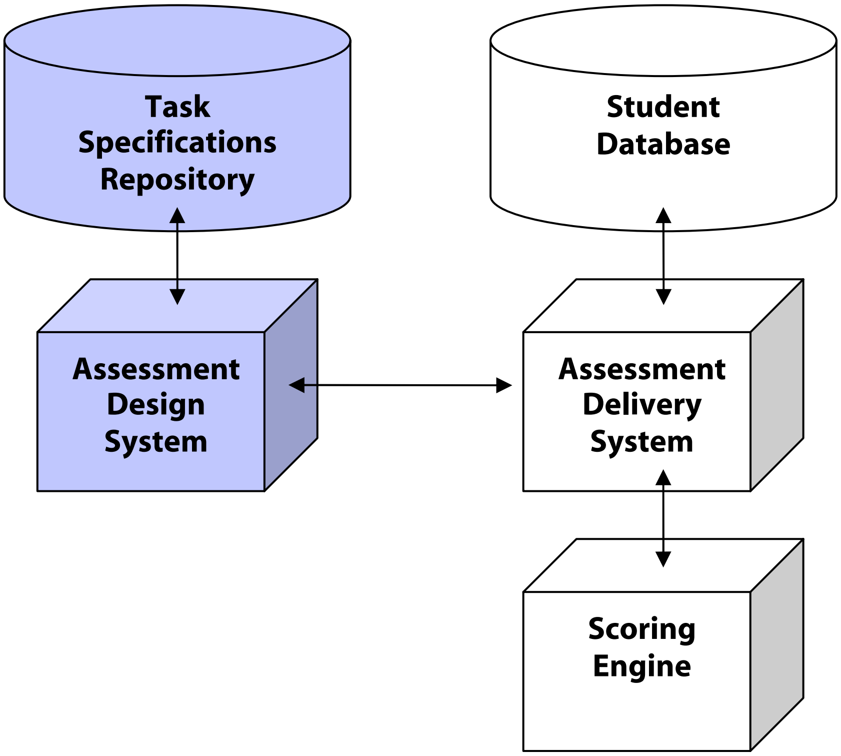
Figure 2. Example of qualitatively different levels on the “ability to build an explanation from evidence” Student Model Variable. The measure for a particular respondent at a particular time is shown as an X on the continuum.



Ability to build an explanation from evidence
Direction of more ability
Descriptions of levels:
Able to compose explanation (claim and evidence) without assistance.
Able to form claim and evidence statements with some guidance.
Able to match evidence to claims.
Unable to match/compose any parts of explanation.
X
A particular respondent’s location
Direction of less ability

The PADI project encourages a principled approach to assessing proficiency with a detailed model of how assessments are related to the specific competencies one is interested in measuring. As illustrated in Figure 3, an *assessment design system* manages the principled design and representation of assessment *task specifications*. An *assessment delivery system* is also needed to instantiate assessment tasks, deliver them to students, gather and evaluate student work, compute the analytics to arrive at estimates of student proficiency, and report back to teachers, students, and other interested parties. Note that the delivery system may access previously designed *task specifications* through the design system, as shown in the figure, or may keep a local copy of the *task specifications* and/or instantiated tasks and access them directly. The delivery system is also responsible for maintaining the longitudinal database of student response data and proficiency estimates. A *scoring engine* is used by the *assessment delivery system* to produce estimates of student proficiencies in the domains of interest from response data gathered during assessment delivery. A computerized assessment system, composed of integrated design and delivery modules, can facilitate the construction of high-quality assessments. This is accomplished by maintaining the connections among the cognition, observation, and interpretation vertices of the NRC assessment triangle.

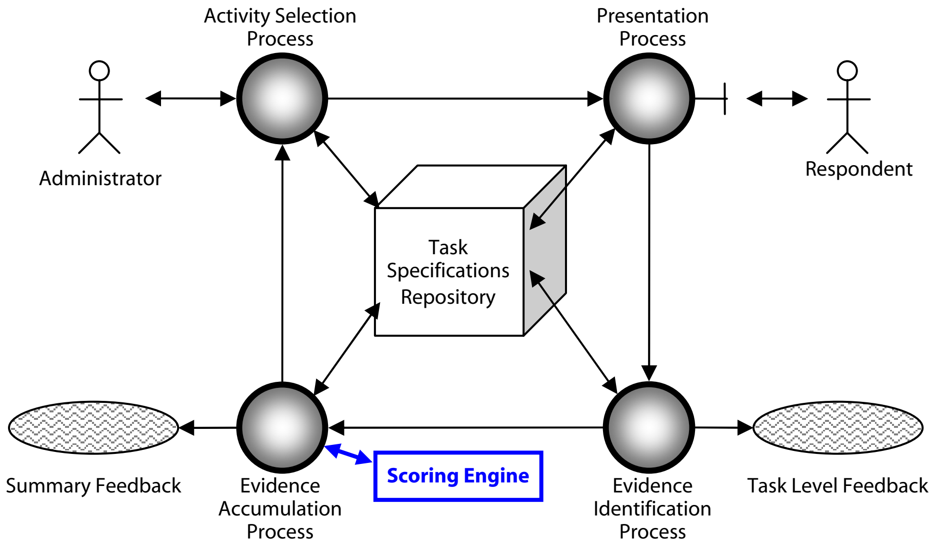
Figure 3. Relationship of an assessment design system, delivery system, and scoring engine in an integrated assessment application. Shaded components constitute the PADI design system.



An *assessment delivery system*, whether computerized or manual, comprises four interrelated processes, as described in the Four Process Model developed by Almond et al., (2002): (1) assessment tasks are selected for delivery to the respondent, (2) the tasks are rendered and presented to the respondent and respondent work products are collected, (3) the Work Products are evaluated and categorized into evidence associated with the targeted Student Model Variables, and (4) the evidence is used to draw inferences about the Student Models of individual respondents. In an integrated assessment system, both the design and delivery modules access the same repository of assessment *task specifications*. These *task specifications* define how tasks are to be generated and rendered to respondents, how work products are to be gathered and evaluated, and how inferences are to be drawn about respondents' knowledge, skills, or abilities.

A *scoring engine* is used to implement the interpretation model applied in the inferential process (step 4). This Measurement Model, as we call it here, defines the way evidence is used to produce estimates of each respondent's locations on the Student Model Variables at the time of participating in the assessment. As shown in Figure 4, the *assessment delivery system* evaluates student work (in the *Evidence Identification Process*) prior to calling the *scoring engine* to produce proficiency estimates. The evaluated response data and associated Measurement Models for each assessment task (accessed from the *task specifications repository*) are then sent to the *scoring engine*, and the *scoring engine* computes and returns proficiency estimates for each respondent. The *assessment delivery system* then produces *summary feedback* or may use intermediate proficiency estimates as input into the selection process for the next task.

Figure 4. Four-process assessment delivery architecture highlighting location of Scoring Engine interface.[1]


Activity Selection Process
Presentation Process
Administrator
Respondent
Task Specifications Repository
Summary Feedback
Evidence Accumulation Process
Scoring Engine
Evidence Identification Process
Task Level Feedback

[1] *Adapted from Mislevy, Almond, and Lukas (2004).*

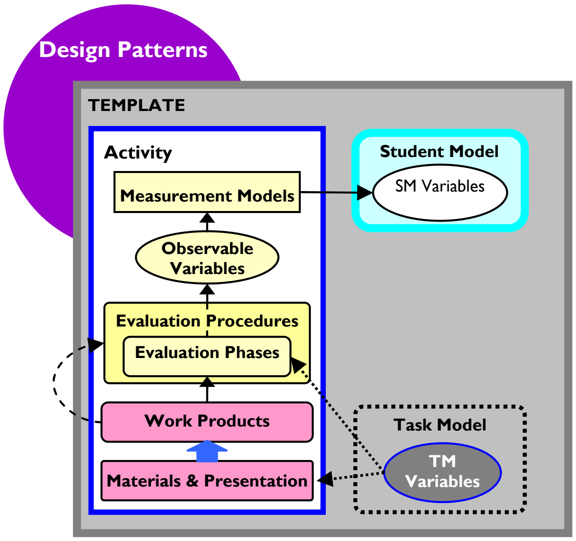
Note that development of an *assessment delivery system* is beyond the scope of the PADI project, but understanding the interfaces between the delivery system and the other components is central to the principled design approach.

3.0 The PADI Design System

The PADI design system consists of an *assessment design system* and a *task specifications repository*, as illustrated in Figure 3. The *assessment design system* manages the design and representation of assessment *task specifications*. It is a software application comprising a series of object models constituting a framework that can be used to represent the interrelated components of assessment: (1) a theory of how students develop targeted knowledge, skills, and abilities; (2) designs of *task templates* and tasks that would allow one to observe students exercising those proficiencies; and (3) evaluation and interpretation methodologies that define the manner in which the observations are associated with the proficiencies to be measured.

Design pattern objects are used to represent a rationale for assessing a particular aspect of science inquiry, such as designing and conducting scientific investigations. They tie curricular learning goals and standards to a description of how student work products are connected to inferences about the student knowledge, skills, and abilities one wishes to advance in the curriculum. *Template* objects are used to represent *task specifications* that conform to the assessment objectives of one or more *design patterns*. As implied in Figure 5, the linkage between a *template* and its guiding *design patterns* is the mechanism that ensures consistency between learning goals and assessment tasks.

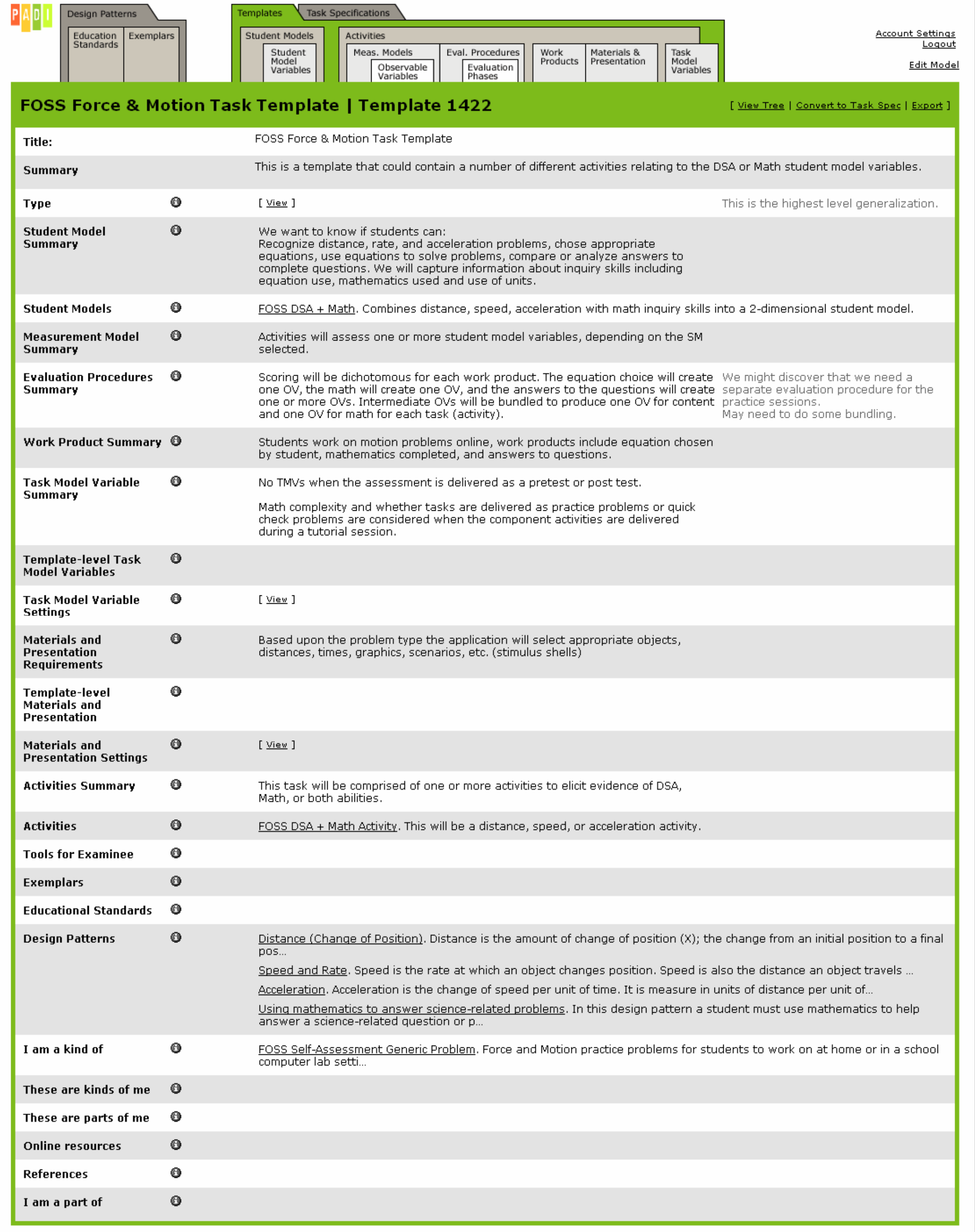
Figure 5. PADI design system template object components and associations.



The PADI design system manages the representation of *design patterns*, *task templates,* and related objects and relationships to support the design of assessment tasks. Figure 5 shows the logical components of the system. *Design patterns* are prominent, providing in narrative form the assessment argument that provides the foundation for the development of *templates*. The components of a *template* allow an assessment designer to specify what an assessment task will look like to a student, how the student work generated from the task will be evaluated, and how that evaluation will be used to draw inferences about the student proficiencies of interest. A *template* contains a Student Model object to define the proficiencies of interest and one or more Activity objects to define how information will be elicited from students as evidence of those proficiencies. A *template* may also contain one or more Task Model Variables to define the assessment environment. In the context of an Activity, information about the appearance of assessment tasks when rendered to students is contained in the Work Products and Materials and Presentation components, while information about how student Work Products are evaluated and transformed into observations is contained in the Evaluation Procedures. An Evaluation Procedure may include multiple Evaluation Phases. The observations, called Observable Variables, are associated with statistical rules defined in Measurement Models. These rules are used to draw inferences about the assessment measures of interest from the evidence contained in the Observable Variables. These design system components help ensure the consistency of an assessment, from the learning goals articulated in the *design patterns* to the evaluation of the Work Products to drawing inferences about student proficiencies.

Figure 6 shows a *template* for a science assessment task that is to be delivered in an interactive online system. The “FOSS Force & Motion Task” *template* contains all the components identified in Figure 5. In Figure 6, the labels along the left margin are called the *template* attributes, and the entries to the right are the attribute values. In this report, we focus on three components of a *task template*: the Student Models, Evaluation Procedures, and Measurement Models. These are the components that operationalize the chain of reasoning from gathering evidence to drawing inferences about what students know and can do.

Figure 6. Excerpt from the "FOSS Force & Motion Task" template.



FOSS Force & Motion Task Template | Template 1422

[View Tree | Convert to Task Spec | Export]

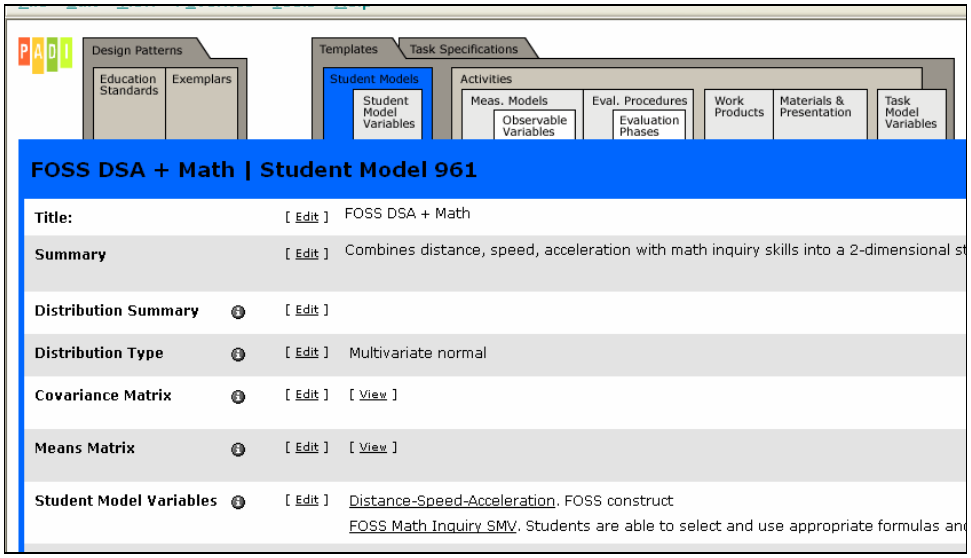
Title:	FOSS Force & Motion Task Template	
Summary	This is a template that could contain a number of different activities relating to the DSA or Math student model variables.	
Type	[View]	This is the highest level generalization.
Student Model Summary	We want to know if students can: Recognize distance, rate, and acceleration problems, chose appropriate equations, use equations to solve problems, compare or analyze answers to complete questions. We will capture information about inquiry skills including equation use, mathematics used and use of units.	
Student Models	FOSS DSA + Math. Combines distance, speed, acceleration with math inquiry skills into a 2-dimensional student model.	
Measurement Model Summary	Activities will assess one or more student model variables, depending on the SM selected.	
Evaluation Procedures Summary	Scoring will be dichotomous for each work product. The equation choice will create one OV, the math will create one OV, and the answers to the questions will create one or more OVs. Intermediate OVs will be bundled to produce one OV for content and one OV for math for each task (activity).	We might discover that we need a separate evaluation procedure for the practice sessions. May need to do some bundling.
Work Product Summary	Students work on motion problems online, work products include equation chosen by student, mathematics completed, and answers to questions.	
Task Model Variable Summary	No TMVs when the assessment is delivered as a pretest or post test. Math complexity and whether tasks are delivered as practice problems or quick check problems are considered when the component activities are delivered during a tutorial session.	
Template-level Task Model Variables		
Task Model Variable Settings	[View]	
Materials and Presentation Requirements	Based upon the problem type the application will select appropriate objects, distances, times, graphics, scenarios, etc. (stimulus shells)	
Template-level Materials and Presentation		
Materials and Presentation Settings	[View]	
Activities Summary	This task will be comprised of one or more activities to elicit evidence of DSA, Math, or both abilities.	
Activities	FOSS DSA + Math Activity. This will be a distance, speed, or acceleration activity.	
Tools for Examinee		
Exemplars		
Educational Standards		
Design Patterns	Distance (Change of Position). Distance is the amount of change of position (X); the change from an initial position to a final pos... Speed and Rate. Speed is the rate at which an object changes position. Speed is also the distance an object travels ... Acceleration. Acceleration is the change of speed per unit of time. It is measure in units of distance per unit of... Using mathematics to answer science-related problems. In this design pattern a student must use mathematics to help answer a science-related question or p...	
I am a kind of	FOSS Self-Assessment Generic Problem. Force and Motion practice problems for students to work on at home or in a school computer lab setti...	
These are kinds of me		
These are parts of me		
Online resources		
References		
I am a part of		

3.1 Student Model

In Figure 6, the Student Models attribute is defined as "FOSS DSA + Math," a multivariate model of aspects of physics knowledge the curriculum targets. "DSA" represents the physics content area of distance, speed, and acceleration treated as a single cognitive element, and "Math" represents general knowledge of mathematics applied to solving science problems. An assessment designer typically determines these knowledge areas from examining the Knowledge, Skills, and Abilities attribute of the associated *design patterns* (Mislevy, Hamel, Fried, Gaffney, Haertel, Hafter, et al., 2003). The *design patterns* relevant to this *template* are also shown in Figure 6. They include “Distance (Change of Position),” “Speed and Rate,” “Acceleration,” and “Using mathematics to answer science-related problems.”

The presence of the “FOSS DSA + Math” Student Model in the *template* requires all the observations associated with the *template* to provide evidence of either DSA content knowledge or mathematics knowledge. Observations are associated with a *template* through the Activity objects contained by the *template*. Figure 7 shows the “FOSS DSA + Math” Student Model components. Note that the Distribution Type attribute describes the population distribution for the Student Model—in this case, a multivariate normal distribution. The Covariance Matrix and Means Matrix attributes, shown in Figure 8, define the values to be used in the population distribution function to estimate EAP (expected a-posteriori) student proficiencies (expected a-posteriori and maximum likelihood proficiency estimates are described in Section 4.0 The BEAR Scoring Engine). The specific values in the covariance and means matrices may not be known at the time of *template* development; they may be determined during a calibration study of pilot data, or they may be set to default values and updated as assessment data are gathered by the *assessment delivery system*.

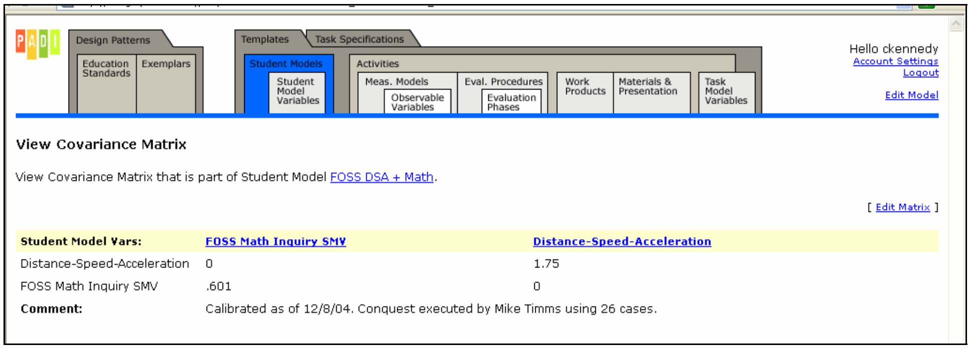
Figure 7. Excerpt of the Student Model “FOSS DSA + Math” containing two Student Model Variables, “Distance-Speed-Acceleration” and “FOSS Math Inquiry.” Student Models are contained in template objects, and their Student Model Variables are referenced in Measurement Model objects.



FOSS DSA + Math | Student Model 961

Title:	[Edit]	FOSS DSA + Math	
Summary	[Edit]	Combines distance, speed, acceleration with math inquiry skills into a 2-dimensional st	
Distribution Summary	[Edit]		
Distribution Type	[Edit]	Multivariate normal	
Covariance Matrix	[Edit]	[View]	
Means Matrix	[Edit]	[View]	
Student Model Variables	[Edit]	Distance-Speed-Acceleration. FOSS construct FOSS Math Inquiry SMV. Students are able to select and use appropriate formulas an	

Figure 8. Examples of Covariance Matrix and Means Matrix attributes of the "FOSS DSA + Math" Student Model.

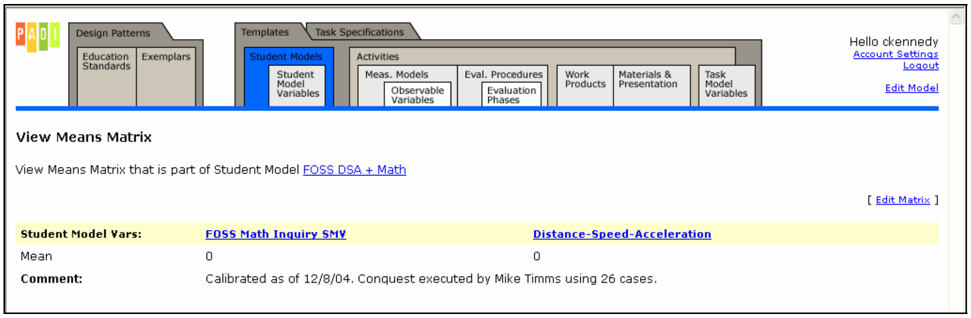


View Covariance Matrix

View Covariance Matrix that is part of Student Model FOSS DSA + Math.

[Edit Matrix]

Student Model Vars:	**FOSS Math Inquiry SMV**	**Distance-Speed-Acceleration**
Distance-Speed-Acceleration	0	1.75
FOSS Math Inquiry SMV	.601	0
Comment:	Calibrated as of 12/8/04. Conquest executed by Mike Timms using 26 cases.	



View Means Matrix

View Means Matrix that is part of Student Model FOSS DSA + Math

[Edit Matrix]

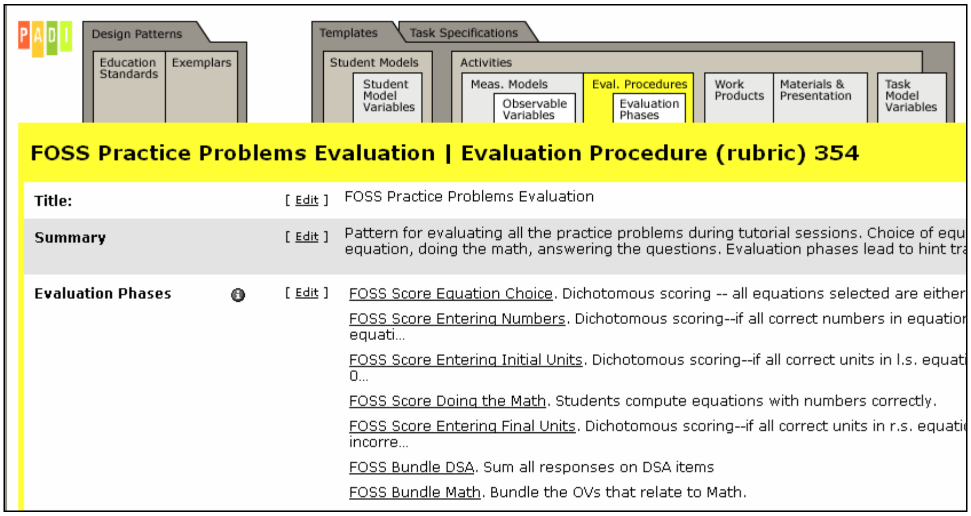
Student Model Vars:	**FOSS Math Inquiry SMV**	**Distance-Speed-Acceleration**
Mean	0	0
Comment:	Calibrated as of 12/8/04. Conquest executed by Mike Timms using 26 cases.	

3.2 *Evaluation Procedures*

We also see in Figure 6 a summary of the Evaluation Procedures that are to be applied to transform student work into the Observable Variables that constitute the evidence we need to draw inferences about student competencies. Evaluation Procedures are elaborated in more detail in Activity objects. The "FOSS Force & Motion Task" *template* contains a generic activity that can be customized for a specific problem to be presented to a student. This activity, named "FOSS DSA + Math," is shown in the Activities attribute in Figure 6. The Evaluation Procedure shown in Figure 9 describes the steps, or Evaluation Phases, that will be taken to evaluate student work from this type of Activity. The Evaluation Phases indicate that the equation selected by the student is evaluated first, followed by the values the student entered into the equation, the units entered into the equation, the mathematical calculation, and then the units entered into the final response. Note that each of these Evaluation Phases results in individual scores[2] of 0 or 1 for an Observable Variable. After these individual scores are determined, the scores associated with the DSA variable are bundled together into a comprehensive single score in the "FOSS Bundle DSA" Evaluation Phase, and the scores associated with the mathematics variable are bundled in the "FOSS Bundle Math" Evaluation Phase. Bundling scores is a process used to combine conditionally dependent responses into a single composite score. This step is required for certain item response models to meet the assumption of conditional independence among the Observable Variables used to produce proficiency estimates (Hoskens & deBoeck, 1997; Wang, Wilson, & Cheng, 2000; Wilson & Adams, 1995).

[2] The term "score" has several meanings in assessment. The usage here and following is that a score on a performance is an evaluation of that performance, expressed as a value of an observable variable.

Figure 9. Example of the "FOSS Practice Problem Evaluation" Evaluation Procedure containing several Evaluation Phases. Evaluation Procedures are contained in Activity objects and link Work Products to Observable Variables.

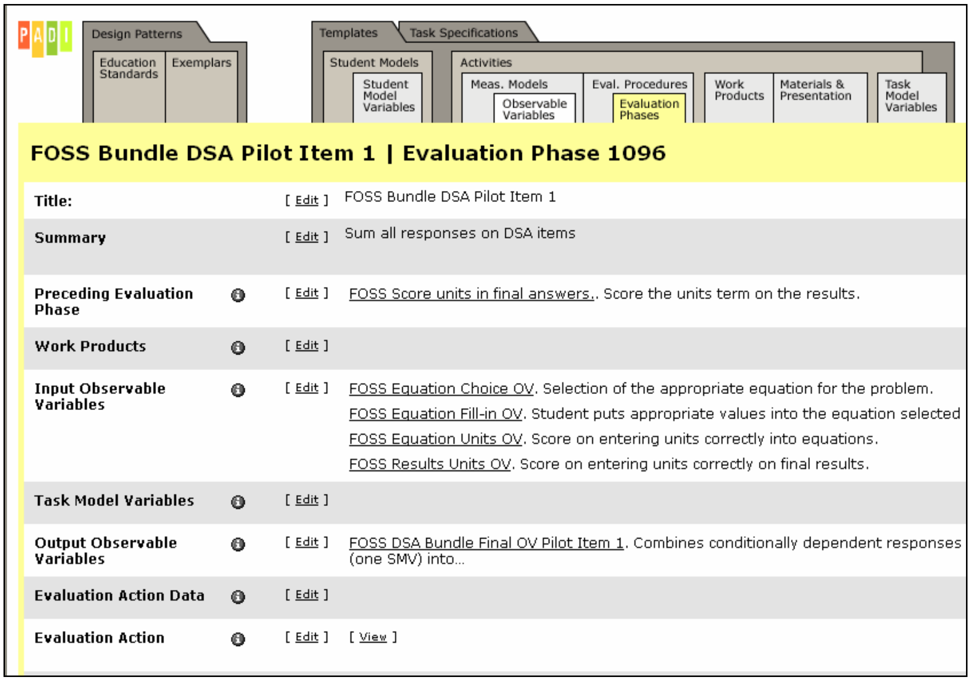


FOSS Practice Problems Evaluation | Evaluation Procedure (rubric) 354

Title:	[Edit]	FOSS Practice Problems Evaluation
Summary	[Edit]	Pattern for evaluating all the practice problems during tutorial sessions. Choice of equ equation, doing the math, answering the questions. Evaluation phases lead to hint tra
Evaluation Phases	[Edit]	FOSS Score Equation Choice. Dichotomous scoring -- all equations selected are either FOSS Score Entering Numbers. Dichotomous scoring--if all correct numbers in equation equati... FOSS Score Entering Initial Units. Dichotomous scoring--if all correct units in l.s. equati 0... FOSS Score Doing the Math. Students compute equations with numbers correctly. FOSS Score Entering Final Units. Dichotomous scoring--if all correct units in r.s. equati incorre... FOSS Bundle DSA. Sum all responses on DSA items FOSS Bundle Math. Bundle the OVs that relate to Math.

Figure 10 shows more detail as to how the DSA scores are to be bundled. Four Input Observable Variable values are examined to determine the value of a single Output Observable Variable. Only the value of the Output Observable Variable will ultimately be used to estimate student proficiencies. Figure 11 shows the Translation Chart for Bundling of the Evaluation Phase shown in Figure 10 (this screen is obtained by selecting the View option of the Evaluation Action attribute of the Evaluation Phase). It contains the rules to be followed in determining the value of the Output Observable Variable. The first four columns show possible values for each of the Input Observable Variables, and the last column shows the value to be assigned to the Output Observable Variable for that row. For example, a student who gets the equation correct (i.e., a score of 1 on the Equation Choice Observable Variable) but enters an incorrect value somewhere in the equations (i.e., a score of 0 on the Equation Fill-in Observable Variable) will receive a score of 2 on the Output Observable Variable, regardless of whether the units were entered correctly or incorrectly in the equation and in the final result. Bundling rules are usually determined by substantive experts who examine student responses to rank each of the possible response patterns. An ordered partition model (Wilson, 1992) in which multiple patterns result in the same score value is frequently implemented by assessment designers. Bundling approaches are described in more detail in Section 5.0, PADI Measurement Model Examples.

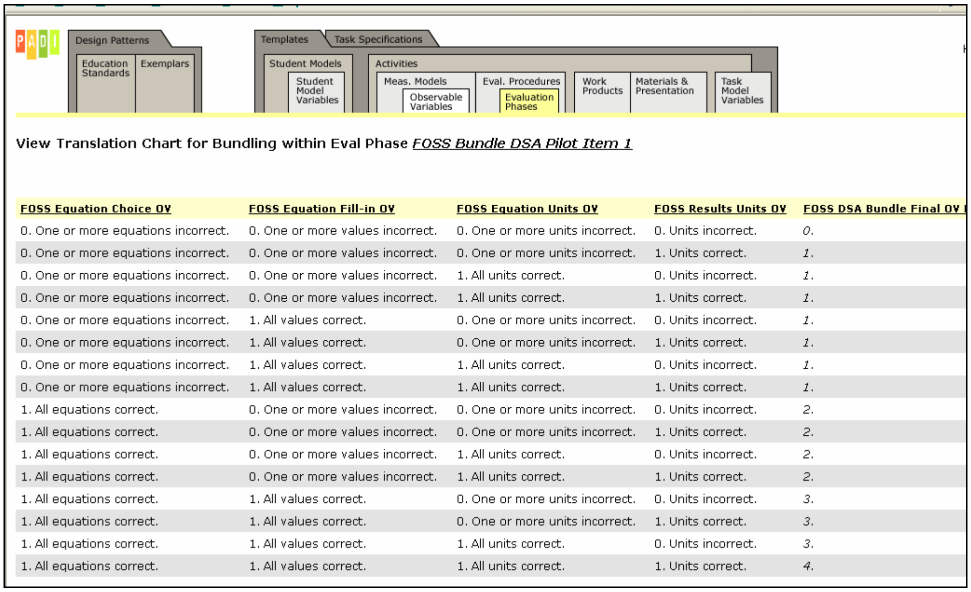
Figure 10. Example of the “FOSS Bundle DSA Pilot Item 1” Evaluation Phase, which bundles several Observable Variables into a final Observable Variable to be used in proficiency estimation.



FOSS Bundle DSA Pilot Item 1 | Evaluation Phase 1096

Title:	[Edit]	FOSS Bundle DSA Pilot Item 1
Summary	[Edit]	Sum all responses on DSA items
Preceding Evaluation Phase	[Edit]	FOSS Score units in final answers.. Score the units term on the results.
Work Products	[Edit]	
Input Observable Variables	[Edit]	FOSS Equation Choice OV. Selection of the appropriate equation for the problem. FOSS Equation Fill-in OV. Student puts appropriate values into the equation selected FOSS Equation Units OV. Score on entering units correctly into equations. FOSS Results Units OV. Score on entering units correctly on final results.
Task Model Variables	[Edit]	
Output Observable Variables	[Edit]	FOSS DSA Bundle Final OV Pilot Item 1. Combines conditionally dependent responses (one SMV) into...
Evaluation Action Data	[Edit]	
Evaluation Action	[Edit]	[View]

Figure 11. Example of how Observable Variables are bundled in the “FOSS Bundle DSA Pilot Item 1” Activity.



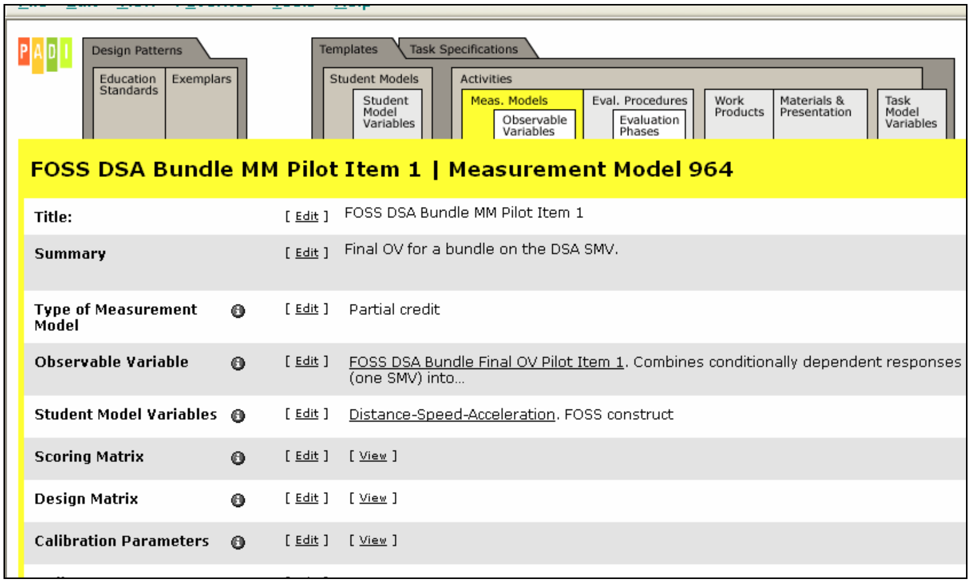
View Translation Chart for Bundling within Eval Phase *FOSS Bundle DSA Pilot Item 1*

FOSS Equation Choice OV	FOSS Equation Fill-in OV	FOSS Equation Units OV	FOSS Results Units OV	FOSS DSA Bundle Final OV
0. One or more equations incorrect.	0. One or more values incorrect.	0. One or more units incorrect.	0. Units incorrect.	*0.*
0. One or more equations incorrect.	0. One or more values incorrect.	0. One or more units incorrect.	1. Units correct.	*1.*
0. One or more equations incorrect.	0. One or more values incorrect.	1. All units correct.	0. Units incorrect.	*1.*
0. One or more equations incorrect.	0. One or more values incorrect.	1. All units correct.	1. Units correct.	*1.*
0. One or more equations incorrect.	1. All values correct.	0. One or more units incorrect.	0. Units incorrect.	*1.*
0. One or more equations incorrect.	1. All values correct.	0. One or more units incorrect.	1. Units correct.	*1.*
0. One or more equations incorrect.	1. All values correct.	1. All units correct.	0. Units incorrect.	*1.*
0. One or more equations incorrect.	1. All values correct.	1. All units correct.	1. Units correct.	*1.*
1. All equations correct.	0. One or more values incorrect.	0. One or more units incorrect.	0. Units incorrect.	*2.*
1. All equations correct.	0. One or more values incorrect.	0. One or more units incorrect.	1. Units correct.	*2.*
1. All equations correct.	0. One or more values incorrect.	1. All units correct.	0. Units incorrect.	*2.*
1. All equations correct.	0. One or more values incorrect.	1. All units correct.	1. Units correct.	*2.*
1. All equations correct.	1. All values correct.	0. One or more units incorrect.	0. Units incorrect.	*3.*
1. All equations correct.	1. All values correct.	0. One or more units incorrect.	1. Units correct.	*3.*
1. All equations correct.	1. All values correct.	1. All units correct.	0. Units incorrect.	*3.*
1. All equations correct.	1. All values correct.	1. All units correct.	1. Units correct.	*4.*

3.3 Measurement Models

The Measurement Model summary shown in Figure 6 provides a narrative of how the observations associated with the *template* will be used as evidence in a statistical model to draw inferences about the targeted proficiencies of students. Individual Measurement Models, one for each Observable Variable, are defined in the Activities. Figure 12 provides an example of one Measurement Model for an Activity. Note that this Measurement Model is associated with the “FOSS DSA Bundle Final OV Pilot Item 1” Output Observable Variable from the “FOSS Bundle DSA” Evaluation Phase shown in Figure 10. This Measurement Model defines the Observable Variable as a partial credit, or polytomous, score. It also indicates that the Observable Variable provides evidence about the “Distance-Speed-Acceleration” Student Model Variable only. The Scoring Matrix, Design Matrix, and Calibration Parameters attributes complete the Measurement Model specification to completely define how inferences about the Student Model are to be ascertained from this Observable Variable. The Scoring Matrix, shown in Figure 13, relates responses in particular levels of the Observable Variable to scores on the “Distance-Speed-Acceleration” Student Model Variable. The Design Matrix, shown in Figure 14, relates responses in particular levels of the Observable Variable to the relevant item parameters (i.e., item difficulties and step difficulties). The Calibration Parameters, shown in Figure 15, are the values to be used in computing item response probabilities when estimating student proficiencies. Calibration Parameters are usually generated from a sample of student responses to the activity or from expectations of relative item difficulty and anticipated student performance. As was the case for the covariance and means matrices for the Student Model, the specific values of the Calibration Parameters may not be known at the time of *template* development.

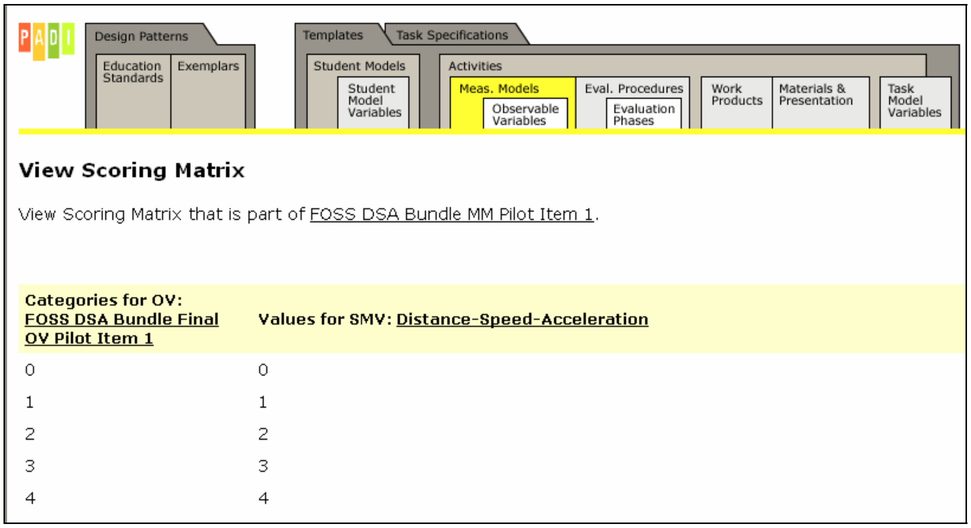
Figure 12. Example of a Measurement Model object with its associated Observable Variable object. Measurement Model objects are contained in Activity objects.



FOSS DSA Bundle MM Pilot Item 1 | Measurement Model 964

Title:	[Edit]	FOSS DSA Bundle MM Pilot Item 1
Summary	[Edit]	Final OV for a bundle on the DSA SMV.
Type of Measurement Model	[Edit]	Partial credit
Observable Variable	[Edit]	FOSS DSA Bundle Final OV Pilot Item 1. Combines conditionally dependent responses (one SMV) into...
Student Model Variables	[Edit]	Distance-Speed-Acceleration. FOSS construct
Scoring Matrix	[Edit]	[View]
Design Matrix	[Edit]	[View]
Calibration Parameters	[Edit]	[View]

Figure 13. Scoring Matrix for the “FOSS DSA Bundle MM Pilot Item 1” Measurement Model.

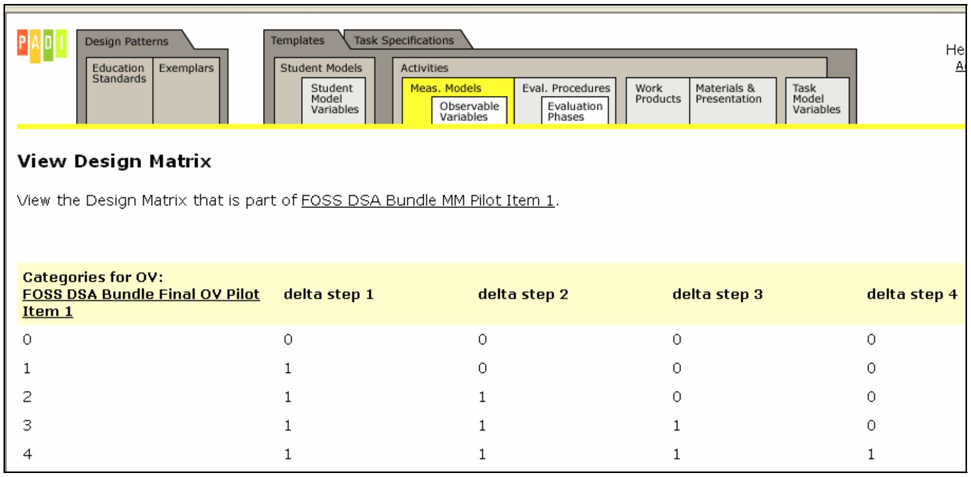


View Scoring Matrix

View Scoring Matrix that is part of FOSS DSA Bundle MM Pilot Item 1.

Categories for OV: FOSS DSA Bundle Final OV Pilot Item 1	Values for SMV: Distance-Speed-Acceleration
0	0
1	1
2	2
3	3
4	4

Figure 14. Design Matrix for the “FOSS DSA Bundle MM Pilot Item 1” Measurement Model.

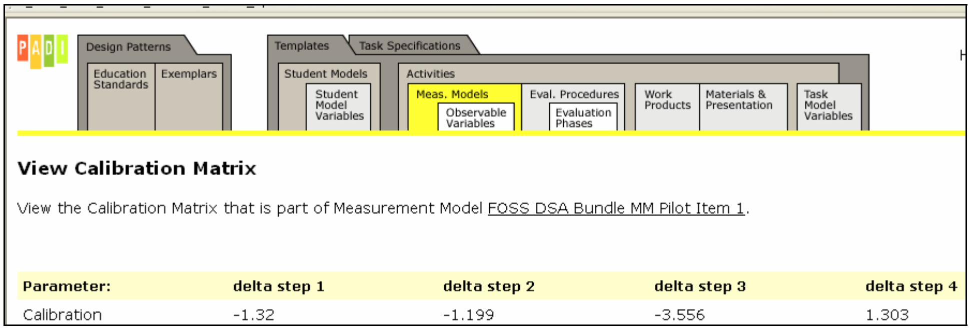


View Design Matrix

View the Design Matrix that is part of FOSS DSA Bundle MM Pilot Item 1.

Categories for OV: FOSS DSA Bundle Final OV Pilot Item 1	delta step 1	delta step 2	delta step 3	delta step 4
0	0	0	0	0
1	1	0	0	0
2	1	1	0	0
3	1	1	1	0
4	1	1	1	1

Figure 15. Calibration Parameters from the “FOSS DSA Bundle MM Pilot Item 1” Measurement Model.



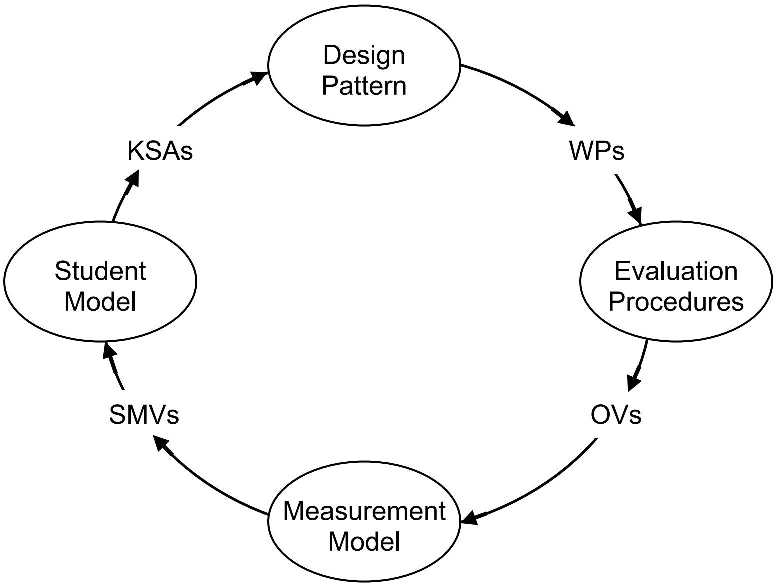
View Calibration Matrix

View the Calibration Matrix that is part of Measurement Model FOSS DSA Bundle MM Pilot Item 1.

Parameter:	delta step 1	delta step 2	delta step 3	delta step 4
Calibration	-1.32	-1.199	-3.556	1.303

The Student Model, Evaluation Procedures, and Measurement Models, through their contained attributes, provide the linkages needed to maintain consistency throughout an assessment system. The PADI design system components that operationalize these linkages are shown graphically in Figure 16. *Design patterns* inspire the Work Products (WPs) that will be gathered from students. Evaluation Procedures transform the content of student Work Products into Observable Variables (OVs), which provide the evidence from which inferences are drawn about what students know and can do. The details of how those inferences are to be drawn are contained in Measurement Models, which connect the Observable Variables to the Student Model Variables (SMVs). And those Student Model Variables are directly related to the knowledge, skills, and abilities (KSAs) targeted by the originating *design pattern*.

Figure 16. Chain of reasoning from student Work Products to Observable Variables to Student Model Variables to assessment objectives as articulated in a design pattern, as implemented in the PADI design system.



As explained earlier, an *assessment delivery system* manages the actual delivery of assessment tasks to students and the gathering and evaluation of student response data. The delivery system may include the analytics for drawing inferences from the data to produce estimates of student proficiencies on the measures of interest. Part of the PADI project, however, is development of multidimensional item response modeling software that can be called by an *assessment delivery system* to produce these estimates.

4.0 The BEAR Scoring Engine

The BEAR Scoring Engine uses the Multidimensional Random Coefficients Multinomial Logit (MRCML) model (Adams, Wilson, & Wang, 1997), which provides a generalized solution for a family of multidimensional, polytomous, Rasch-based models to produce inferences about student proficiencies. The model is flexible in that it can fit assessments with a wide range of item types and gives the designer control of how parameters are defined and applied at the category level on each item. Assessment developers specify the model by defining a prior multivariate distribution, scoring and design matrices, and item parameters. These components, which typically are defined in *task specifications* generated by the PADI design system, are sent to the Scoring Engine, along with the evaluated student response data, in XML (Extensible Markup Language) documents. The *assessment delivery system* accesses the Scoring Engine through a URL (uniform resource locator) address. The Scoring Engine applies the values from the XML documents to the proficiency algorithm, computes student proficiency estimates and covariance data, and returns updated information to the requesting application in another XML document. Excerpts from the input and output XML documents are illustrated in Appendix A.

The Scoring Engine estimates student proficiencies by using two methods: expected a-posteriori (EAP) and maximum likelihood (ML) estimation. The EAP is a Bayesian estimation procedure using information from both the respondents' scores (i.e., values of observable variables) and the distribution of the respondents, whereas the ML approach uses only the respondents' scores. As described earlier, a PADI Student Model describes the prior distribution by defining a means matrix (actually, a one-row matrix, so we may also think of it as a vector) and a covariance matrix across all the Student Model Variables. Table 1 provides examples of means vectors and covariance matrices for unidimensional and multidimensional models. For a unidimensional model, the means vector contains a single value and the variance-covariance matrix contains only the variance cell. For a multidimensional model, a mean is entered for each dimension and the complete variance-covariance matrix is entered.

Table 1. Examples of unidimensional and multidimensional means vectors and covariance matrices.

	Unidimensional	Multidimensional
Means Vector	$\begin{array}{c} \text{SMV}_1 \\ \begin{bmatrix} 0.652 \end{bmatrix} \end{array}$	$\begin{array}{cc} \text{SMV}_1 & \text{SMV}_2 \\ \multicolumn{2}{c}{\begin{bmatrix} 0.542 & 0.865 \end{bmatrix}} \end{array}$
Covariance Matrix	$\begin{array}{cc} & \text{SMV}_1 \\ \text{SMV}_1 & \begin{bmatrix} 0.954 \end{bmatrix} \end{array}$	$\begin{array}{cc} & \begin{array}{cc} \text{SMV}_1 & \text{SMV}_2 \end{array} \\ \begin{array}{c} \text{SMV}_1 \\ \text{SMV}_2 \end{array} & \begin{bmatrix} 1.260 & 0.783 \\ 0.783 & 0.812 \end{bmatrix} \end{array}$

The *assessment delivery system* can request either EAP or ML estimates and can also specify a number of other parameters that the Scoring Engine uses in executing the estimation procedure, such as the integration method, the number of nodes, and convergence criteria.

Through MRCML modeling and the PADI design system structure, the Scoring Engine accommodates assessments that measure multiple aspects of proficiency and that can be defined at the category level. In the PADI environment, we consider each Observable Variable separately, so Measurement Models also are defined at the Observable Variable, or item, level. The following section, PADI Measurement Model Examples, elaborates on a number of Measurement Models that can be implemented with this system, including between- and within-item multidimensionality and bundling examples.

5.0 PADI Measurement Model Examples

Designing a coherent assessment—that is, one that reliably measures a specific set of proficiencies—is a complex process. Items must elicit responses that, when evaluated, produce evidence that can be used to draw inferences about the proficiencies of interest. The data must fit the statistical model and conform to model assumptions. In the case of multidimensional IRM, the standard assumptions include unidimensionality of each Student Model Variable, monotonicity over the variable, and local independence of the items. Unidimensionality refers to the degree to which items measure the same Student Model Variable. Monotonicity refers to the situation in which persons with more of the Student Model Variable have greater probabilities of responding at higher score levels on the items than do persons with less of the Student Model Variable. Local independence means that a person's response on one item does not influence his or her responses on any other items. These assumptions are usually tested and confirmed during the calibration phase of task development.

The MRCML Measurement Model specified in the PADI system describes response probability equations by defining a Scoring Matrix to associate items with Student Model Variables, a Design Matrix to associate items with item parameters, and calibrated item parameters. These probabilities are used to determine the likelihood of responses to items for persons with specific abilities. Using this information, we can infer an ability from response data. The general MRCML formulation for the probability of a response pattern, **x**, is

$$P(\mathbf{x};\xi \mid \boldsymbol{\theta}) = \frac{\exp\left[\mathbf{x}'(\mathbf{B}\boldsymbol{\theta} - \mathbf{A}\xi)\right]}{\sum\limits_{\mathbf{z}\in\Omega} \exp\left[\mathbf{z}'(\mathbf{B}\boldsymbol{\theta} - \mathbf{A}\xi)\right]}$$

where θ is the vector of Student Model Variables, ξ is the vector of calibrated item parameters, and Ω is the set of all possible response patterns (Adams, Wilson & Wang, 1997). We use **z** to denote a pattern coming from the full set of response patterns while **x** denotes the one of interest (**z′** and **x′** are transpositions of **z** and **x**). The response pattern, **x**, is comprised of vectors for each item with one element in the vector for each item category, $\mathbf{x} = \{\mathbf{x}_{1,} \mathbf{x}_2, ..., \mathbf{x}_I\} = \{x_{11}, x_{12}, ..., x_{1m1}, x_{21}, x_{22}, ..., x_{2m2}, ..., x_{ImI}\}$ for *mi* = number of categories for item *i*, and *I* = number of items. Note that in this formulation the item parameters are considered known and conditioned on θ. The Scoring Matrix, **B**, is used to construct the θ component of the probability equations, and the Design Matrix, **A**, is used to construct the ξ component. Specific probability equations generated from the Scoring and Design Matrices are shown in the following set of examples. Background information about item response modeling is presented in Appendix B.

When an assessment is intended to measure multiple Student Model Variables, individual items may measure a single Student Model Variable or multiple variables. We refer to the case in which each item provides evidence about a single variable as *between-item* multidimensionality and the case in which a single item provides evidence about multiple variables as *within-item* multidimensionality. In the PADI design system, a case of between-item multidimensionality occurs when the Student Model contains multiple Student Model Variables but each Observable Variable maps to only one of them. When an Observable Variable maps to more than one Student Model Variable, we have a case of within-item multidimensionality.

In the sections that follow, we first provide examples of IRM models, both unidimensional and multidimensional, that could be used to describe a single Observable Variable within the PADI system when the assumptions of IRT are met. We then show how PADI and the Scoring Engine can be used to model Observable Variables that are not conditionally independent through a process called “bundling.” The final section shows how modeling Observable Variables within PADI and the Scoring Engine compares with approaches used to model full tests consisting of a collection of conditionally independent Observable Variables.

5.1 Modeling One Observable Variable

These examples describe the modeling of assessment tasks in which individual responses to multiple items are conditionally independent.

5.1.1 Unidimensional Dichotomous Model

This model is useful for representing responses that are either correct or incorrect and that measure only one Student Model Variable. Examples include making a selection from a list, responding to a true/false or multiple-choice question, and fill-in-the-blank items that have a single correct response.

Scoring Matrix (one Student Model Variable, so one column):

$$\begin{array}{ll} & \text{SMV}_1 \\ \begin{array}{l}\text{Category 1}\\[4pt] \text{Category 2}\end{array} & \begin{bmatrix}0\\1\end{bmatrix}\end{array}$$

Design Matrix (one observable variable, so one column):

$$\begin{array}{ll} & \delta_1 \\ \begin{array}{l}\text{Category 1}\\[4pt] \text{Category 2}\end{array} & \begin{bmatrix}0\\1\end{bmatrix}\end{array}$$

In addition to the Scoring and Design Matrices, the Scoring Engine requires calibrated item parameters to compute response probabilities. These are provided as vectors in which the number of elements is equal to the number of columns in the Design Matrix. In this example, we have one column in the Design Matrix and one element in the calibrated parameters vector.[3]

Calibrated Parameters Vector:

$$\begin{array}{ll} & \delta_1 \\ \text{Item difficulty} & \begin{bmatrix}1.05\end{bmatrix}\end{array}$$

These matrices are transformed by the Scoring Engine into the following probability equations:

[3] The calibrated parameter vector values in these examples are hypothetical.

$$P(x=0)=\frac{1}{1+\exp(\theta-\delta_1)}$$ and

$$P(x=1)=\frac{\exp(\theta-\delta_1)}{1+\exp(\theta-\delta_1)}$$, where δ_1 = 1.05.

5.1.2 Unidimensional Partial Credit Model

This model is used to represent responses that can be scored at more than two levels. A scoring rubric is usually required to describe what a score at each level means relative to the Student Model Variable being measured. Essay questions are typically scored with this approach, with scores ranging from 0 to 10, for example.

The Scoring and Design Matrices below represent an Observable Variable with four categories. In this Scoring Matrix, a response in the third category is represented by a score of 2. Note that the response data sent to the Scoring Engine indicate which *category* the response is in, using integral values beginning at 0. Thus, a response in the second category is sent to the Scoring Engine as the value 1. The response categories are always positive integers. For simple models, such as that shown below, it is quite common for the response category value to be the same as the score value. However, it is permissible for the Scoring Matrix to include negative or fractional values.

Scoring Matrix (one SMV, so one column):

$$\begin{array}{ll} & \text{SMV}_1 \\ \text{Category 1} & \begin{bmatrix} 0 \\ 1 \\ 2 \\ 3 \end{bmatrix} \\ \text{Category 2} & \\ \text{Category 3} & \\ \text{Category 4} & \end{array}$$

Design Matrix (four categories means three steps, so three columns):

$$\begin{array}{l|ccc} & \delta_1 & \delta_2 & \delta_3 \\ \text{Category 1} & 0 & 0 & 0 \\ \text{Category 2} & 1 & 0 & 0 \\ \text{Category 3} & 1 & 1 & 0 \\ \text{Category 4} & 1 & 1 & 1 \end{array}$$

In this Design Matrix, the difficulty of achieving a response in the third category is computed from the difficulty of advancing from the first category to the second category (the first column) and the difficulty of advancing from the second category to the third category (the second column). That is, the difficulty of achieving a response in the third category is conditioned on being able to earn the lower scores also. This approach requires scores to be hierarchically ordered such that each score represents a higher level of proficiency than the score before. Just as for Scoring Matrices, entries in the Design Matrix may also be negative or fractional values.

Calibrated Parameters Vector:

$$\begin{array}{ccc} \delta_1 & \delta_2 & \delta_3 \\ \left[1.35\right. & 0.25 & \left.0.86\right] \end{array}$$

Note that the number of elements in the calibrated parameters vector is equal to the number of columns in the Design Matrix.

These matrices are transformed by the Scoring Engine into the following probability equations, where $\delta_{11} = 1.35$, $\delta_{12} = 0.25$, and $\delta_{13} = 0.86$:

$$P(x_i = 0) = \frac{1}{\sum_{k=0}^{3} \exp \sum_{j=0}^{k} (\theta - \delta_{ij})};$$

$$P(x_i = 1) = \frac{\exp \sum_{j=0}^{1} (\theta - \delta_{ij})}{\sum_{k=0}^{3} \exp \sum_{j=0}^{k} (\theta - \delta_{ij})} = \frac{\exp(\theta - \delta_{i1})}{\sum_{k=0}^{3} \exp \sum_{j=0}^{k} (\theta - \delta_{ij})};$$

$$P(x_i = 2) = \frac{\exp \sum_{j=0}^{2} (\theta - \delta_{ij})}{\sum_{k=0}^{3} \exp \sum_{j=0}^{k} (\theta - \delta_{ij})} = \frac{\exp(\theta - \delta_{i1} + \theta - \delta_{i2})}{\sum_{k=0}^{3} \exp \sum_{j=0}^{k} (\theta - \delta_{ij})} = \frac{\exp(2\theta - (\delta_{i1} + \delta_{i2}))}{\sum_{k=0}^{3} \exp \sum_{j=0}^{k} (\theta - \delta_{ij})}; \text{ and}$$

$$P(x_i = 3) = \frac{\exp \sum_{j=0}^{3} (\theta - \delta_{ij})}{\sum_{k=0}^{3} \exp \sum_{j=0}^{k} (\theta - \delta_{ij})} = \frac{\exp(3\theta - (\delta_{i1} + \delta_{i2} + \delta_{i3}))}{\sum_{k=0}^{3} \exp \sum_{j=0}^{k} (\theta - \delta_{ij})}.$$

Note the conventions $\exp(0) \equiv 1$ and $\sum_{j=0}^{0} (\theta - \delta_{ij}) \equiv 0$; and that

$\sum_{k=0}^{m} \exp \sum_{j=0}^{k} (\theta - \delta_{ij})$ is the sum of the numerators for all categories.

5.1.3 Unidimensional Rating Scale Model.

This is similar to the unidimensional partial credit model except that (1) the scoring rubric must be the same for all Observable Variables on the assessment, and (2) the step difficulties are parameterized differently. Rating scale models are often used for questionnaires and surveys. The following Scoring Matrix could be used for a rating scale Observable Variable with five categories. In the example below, a response in the second category is represented by a score of 1.

Scoring Matrix (one SMV, so one column):

$$\begin{array}{ll} & \text{SMV}_1 \\ \text{Category 1} & \left[\begin{matrix} 0 \\ 1 \\ 2 \\ 3 \\ 4 \end{matrix}\right. \\ \text{Category 2} & \\ \text{Category 3} & \\ \text{Category 4} & \\ \text{Category 5} & \end{array}$$

The Design Matrix could be constructed in the same manner as for the partial credit model. By convention, however, we parameterize the item difficulties differently in the rating scale model, as $(\delta + \tau)$ values, so we construct the Design Matrix differently also.

Preliminary RS Design Matrix (average difficulty, δ_i, and four tau parameters, $\tau_{1,}$ $\tau_{2,}$ τ_3, and $\tau_{4,}$ so five columns):

$$\begin{array}{l|ccccc|} & \delta & \tau_1 & \tau_2 & \tau_3 & \tau_4 \\ \text{Category 1} & 0 & 0 & 0 & 0 & 0 \\ \text{Category 2} & 1 & 1 & 0 & 0 & 0 \\ \text{Category 3} & 2 & 1 & 1 & 0 & 0 \\ \text{Category 4} & 3 & 1 & 1 & 1 & 0 \\ \text{Category 5} & 4 & 1 & 1 & 1 & 1 \end{array}$$

With this Design Matrix, the difficulty of achieving a response in the third category is computed from the average difficulty of the Observable Variable (the first column), the deviation from the average difficulty to get a response in the second category (the second column), and the deviation from the average difficulty to get a response in the third category (the third column). These tau parameters have a different interpretation, and are calibrated differently, from the step parameters in the partial credit model, so the formulation of the Design Matrix looks different from that for the partial credit model.

Note that the total difficulty of getting a response in the third category is:

	average item difficulty + difficulty in going from a category 1 response to a category 2 response	$(\delta_i + \tau_{i1})$
+	average item difficulty + difficulty in going from a category 2 response to a category 3 response	$(\delta_i + \tau_{i2})$
=	2*(average item difficulty)	$(2\delta_i$
	+ difficulty in going from a category 1 response to a category 2 response	$+ \tau_{i1}$
	+ difficulty in going from a category 2 response to a category 3 response	$+ \tau_{i2})$

In MRCML terms, the formulation is denoted as $2\delta_i+\tau_{i1}+\tau_{i2}$. The coefficients 2, 1, and 1 are captured in the Design Matrix row denoted as "Category 3."

Since the sum of all the tau parameters is 0, the total difficulty of getting a response in the fifth category is $4\delta_i + \Sigma\tau = 4\delta_i$; accordingly, we simplify the Design Matrix by setting the tau parameters in the last row to 0. Thus, we do not have to estimate τ_4, and we need only the first four columns of the Design Matrix.

Final RS Design Matrix (average OV difficulty, δ_i, and three tau parameters, τ_1, τ_2, and τ_3, so four columns):

$$\begin{array}{l c} & \begin{array}{cccc} \delta & \tau_1 & \tau_2 & \tau_3 \end{array} \\ \begin{array}{l} \text{Category 1} \\ \text{Category 2} \\ \text{Category 3} \\ \text{Category 4} \\ \text{Category 5} \end{array} & \begin{bmatrix} 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 2 & 1 & 1 & 0 \\ 3 & 1 & 1 & 1 \\ 4 & 0 & 0 & 0 \end{bmatrix} \end{array}$$

Calibrated Parameters Vector:

$$\begin{array}{c} \begin{array}{cccc} \delta & \tau_1 & \tau_2 & \tau_3 \end{array} \\ \begin{bmatrix} -1.35 & 0.26 & 1.03 & 0.64 \end{bmatrix} \end{array}$$

These matrices are transformed by the Scoring Engine into the following probability equations, where $\delta_i = -1.35$, $\tau_1 = 0.26$, $\tau_2 = 1.03$, and $\tau_3 = 0.64$:

$$P(x_i = 0) = \frac{1}{\sum_{k=0}^{4} \exp \sum_{j=0}^{k} (\theta - (\delta_i + \tau_j))};$$

$$P(x_i = 1) = \frac{\exp \sum_{j=0}^{1} (\theta - (\delta_i + \tau_j))}{\sum_{k=0}^{4} \exp \sum_{j=0}^{k} (\theta - (\delta_i + \tau_j))} = \frac{\exp(\theta - (\delta_i + \tau_1))}{\sum_{k=0}^{4} \exp \sum_{j=0}^{k} (\theta - (\delta_i + \tau_j))};$$

$$P(x_i = 2) = \frac{\exp \sum_{j=0}^{2} (\theta - (\delta_i + \tau_j))}{\sum_{k=0}^{4} \exp \sum_{j=0}^{k} (\theta - (\delta_i + \tau_j))} = \frac{\exp(2\theta - (2\delta_i + \tau_1 + \tau_2))}{\sum_{k=0}^{4} \exp \sum_{j=0}^{k} (\theta - \delta_{ij})};$$

$$P(x_i = 3) = \frac{\exp \sum_{j=0}^{3} (\theta - (\delta_i + \tau_j))}{\sum_{k=0}^{4} \exp \sum_{j=0}^{k} (\theta - (\delta_i + \tau_j))} = \frac{\exp(3\theta - (3\delta_i + \tau_1 + \tau_2 + \tau_3))}{\sum_{k=0}^{4} \exp \sum_{j=0}^{k} (\theta - (\delta_i + \tau_j))}; \text{and}$$

$$P(x_i = 4) = \frac{\exp \sum_{j=0}^{4} (\theta - (\delta_i + \tau_j))}{\sum_{k=0}^{4} \exp \sum_{j=0}^{k} (\theta - (\delta_i + \tau_j))} = \frac{\exp(4\theta - (4\delta_i + \tau_1 + \tau_2 + \tau_3 + \tau_4))}{\sum_{k=0}^{4} \exp \sum_{j=0}^{k} (\theta - (\delta_i + \tau_j))}.$$

Note the conventions $\exp(0) \equiv 1$ and $\sum_{\mathrm{j}=0}^{0} (\theta - (\delta_i + \tau_j)) \equiv 0$; and that

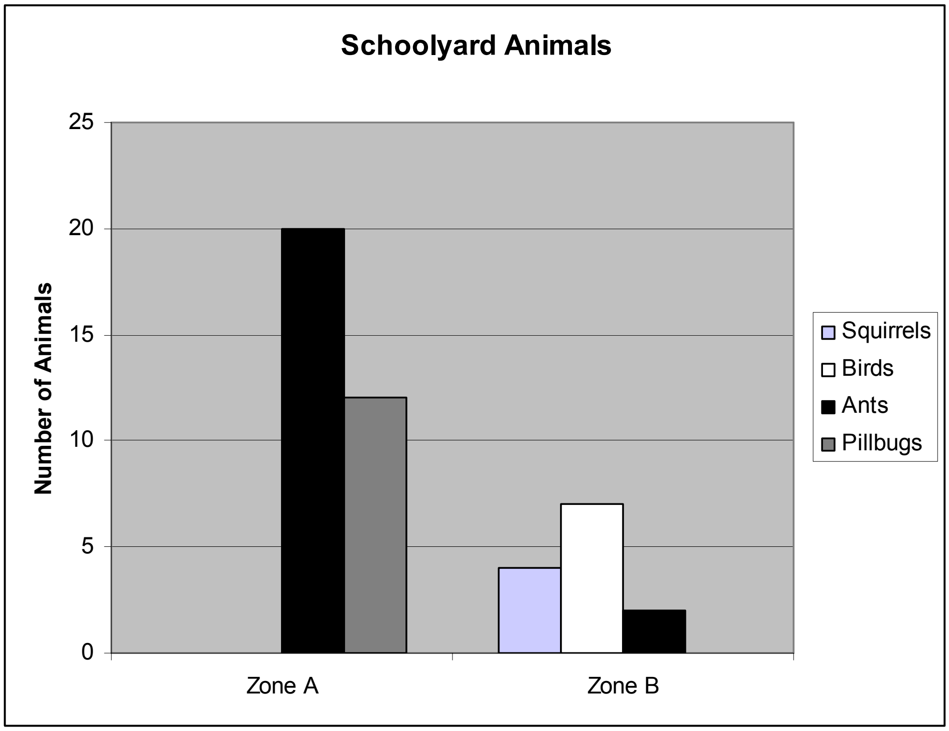
$\sum_{k=0}^{m} \exp \sum_{j=0}^{k} (\theta - (\delta_i + \tau_j))$ is the sum of the numerators for all categories.

5.1.4 Within-Item Multidimensional Partial Credit Model.

This model is used to represent a single Observable Variable that is associated with more than one Student Model Variable. For example, a single response to an open-ended problem may provide evidence of a student's content knowledge and his or her ability to formulate an explanation. One way to evaluate this type of response is to produce two scores for the Observable Variable, one for the *content knowledge* SMV and one for the *building explanations* SMV. An example of this type of assessment task from the BioKIDS curriculum (<http://www.biokids.umich.edu/>) is shown in Figure 17.

Figure 17. Example of within-item multidimensionality from BioKIDS Item 5.[1]

5. Using the graph below, predict which zone most likely has a tree in it and give one reason to support your prediction.



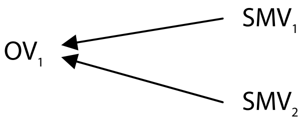
I think that zone _____________ has a tree in it because

[1] *For information on the BioKIDS project, see http://www.biokids.umich.edu/*

In this example, the selection of the zone is considered an indicator of content knowledge (in this case, biodiversity) and the explanation is an indicator of knowledge about building an explanation. A single Observable Variable provides evidence of the student's location on both Student Model Variables.

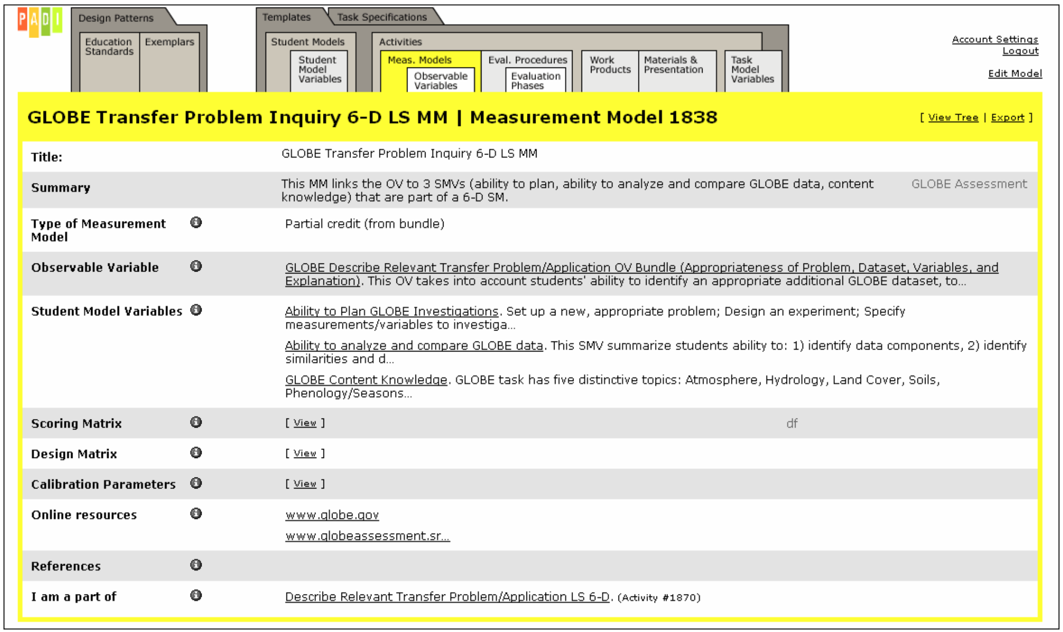
Each Student Model Variable may have a different number of categories. For example, *content knowledge* may have two categories (correct and incorrect) and *building explanations* may have three categories, resulting in six unique combinations of responses on the task overall.

The first category of the overall task represents the situation in which the student has a response in the first category on the first Student Model Variable and a response in the first category on the second Student Model Variable. We construct the complete set of overall task categories by building permutations of the combinations of responses on the two Student Model Variables. For proficiency estimation purposes, we do not consider the initial response categories again; only the overall response category information is sent to the Scoring Engine.



The Measurement Model for the task shown in Figure 17 is shown in Figure 18. Note that the Measurement Model contains one Observable Variable, *BioKIDS pre/post item 5*, and two Student Model Variables, *Biodiversity content* and *BioKIDS overall inquiry*.

Figure 18. Measurement Model from the PADI design system for BioKIDS Item 5.



GLOBE Transfer Problem Inquiry 6-D LS MM | Measurement Model 1838 [View Tree | Export]

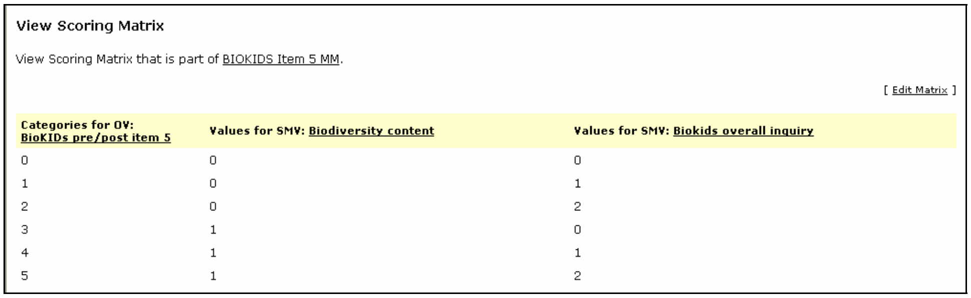
Title:	GLOBE Transfer Problem Inquiry 6-D LS MM	
Summary	This MM links the OV to 3 SMVs (ability to plan, ability to analyze and compare GLOBE data, content knowledge) that are part of a 6-D SM.	GLOBE Assessment
Type of Measurement Model	Partial credit (from bundle)	
Observable Variable	GLOBE Describe Relevant Transfer Problem/Application OV Bundle (Appropriateness of Problem, Dataset, Variables, and Explanation). This OV takes into account students' ability to identify an appropriate additional GLOBE dataset, to...	
Student Model Variables	Ability to Plan GLOBE Investigations. Set up a new, appropriate problem; Design an experiment; Specify measurements/variables to investiga... Ability to analyze and compare GLOBE data. This SMV summarize students ability to: 1) identify data components, 2) identify similarities and d... GLOBE Content Knowledge. GLOBE task has five distinctive topics: Atmosphere, Hydrology, Land Cover, Soils, Phenology/Seasons...	
Scoring Matrix	[View]	df
Design Matrix	[View]	
Calibration Parameters	[View]	
Online resources	www.globe.gov www.globeassessment.sr...	
References		
I am a part of	Describe Relevant Transfer Problem/Application LS 6-D. (Activity #1870)	

Scoring Matrix:

$$\begin{array}{lcc}
 & \text{SMV}_1 & \text{SMV}_2 \\
\text{Category 1 (0,0)} & 0 & 0 \\
\text{Category 2 (0,1)} & 0 & 1 \\
\text{Category 3 (0,2)} & 0 & 2 \\
\text{Category 4 (1,0)} & 1 & 0 \\
\text{Category 5 (1,1)} & 1 & 1 \\
\text{Category 6 (1,2)} & 1 & 2
\end{array}$$

The Scoring Matrix for the Measurement Model shown in Figure 18 is shown in Figure 19.[4]

Figure 19. PADI design system Scoring Matrix for BioKIDS Item 5.



View Scoring Matrix

View Scoring Matrix that is part of BIOKIDS Item 5 MM.

[Edit Matrix]

Categories for OV: BioKIDs pre/post item 5	Values for SMV: Biodiversity content	Values for SMV: Biokids overall inquiry
0	0	0
1	0	1
2	0	2
3	1	0
4	1	1
5	1	2

There are a number of options for generating Design Matrices for this example. The simplest is to assume the saturated model, shown below.

Saturated Design Matrix:

$$\begin{array}{lccccc}
 & \delta_1 & \delta_2 & \delta_3 & \delta_4 & \delta_5 \\
\text{Category 1 (0,0)} & 0 & 0 & 0 & 0 & 0 \\
\text{Category 2 (0,1)} & 1 & 0 & 0 & 0 & 0 \\
\text{Category 3 (0,2)} & 0 & 1 & 0 & 0 & 0 \\
\text{Category 4 (1,0)} & 0 & 0 & 1 & 0 & 0 \\
\text{Category 5 (1,1)} & 0 & 0 & 0 & 1 & 0 \\
\text{Category 6 (1,2)} & 0 & 0 & 0 & 0 & 1
\end{array}$$

[4] Note that the PADI design system automatically numbers categories beginning with 0, rather than 1.

These matrices are transformed by the Scoring Engine into the following probability equations:

$$P(x_i = 0) = \frac{1}{\sum_{k=0}^{5}\exp\sum_{j=0}^{k}((\theta_{1j} + \theta_{2j}) - \delta_{ij})};$$

$$P(x_i = 1) = \frac{\exp(\theta_2 - \delta_{i1})}{\sum_{k=0}^{5}\exp\sum_{j=0}^{k}((\theta_{1j} + \theta_{2j}) - \delta_{ij})};$$

$$P(x_i = 2) = \frac{\exp(2\theta_2 - \delta_{i2})}{\sum_{k=0}^{5}\exp\sum_{j=0}^{k}((\theta_{1j} + \theta_{2j}) - \delta_{ij})};$$

$$P(x_i = 3) = \frac{\exp(\theta_1 - \delta_{i3})}{\sum_{k=0}^{5}\exp\sum_{j=0}^{k}((\theta_{1j} + \theta_{2j}) - \delta_{ij})};$$

$$P(x_i = 4) = \frac{\exp(\theta_1 + \theta_2 - \delta_{i4})}{\sum_{k=0}^{5}\exp\sum_{j=0}^{k}((\theta_{1j} + \theta_{2j}) - \delta_{ij})};$$

$$P(x_i = 5) = \frac{\exp(\theta_1 + 2\theta_2 - \delta_{i5})}{\sum_{k=0}^{5}\exp\sum_{j=0}^{k}((\theta_{1j} + \theta_{2j}) - \delta_{ij})}$$

Note the conventions $\exp(0) \equiv 1$ and $\sum_{j=0}^{0}((\theta_{1j} + \theta_{2j}) - \delta_{ij}) \equiv 0;$ and that $\sum_{k=0}^{m}\exp\sum_{j=0}^{k}((\theta_{1j} + \theta_{2j}) - \delta_{ij})$ is the sum of the numerators for all categories.

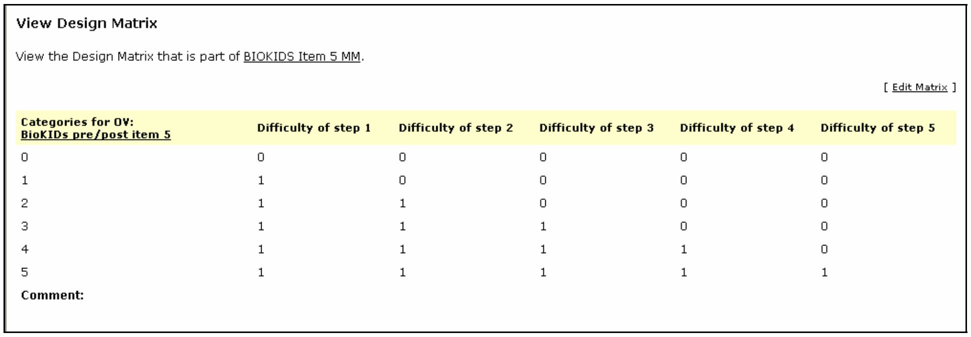
Another straightforward approach for generating the Design Matrix is to assume no interaction effects between the difficulty of the task and the Student Model Variables and treat the Observable Variable as a normal partial credit model with three steps, one for each "total score" possibility. The Design Matrix implemented for the Measurement Model shown in Figure 19 is shown in Figure 20.[5]

[5] A six-category item has five step parameters, one for each transition between score levels.

Partial Credit Design Matrix:

$$\begin{array}{lccc}
 & \delta_1 & \delta_2 & \delta_3 \\
\text{Category 1 (0,0)} & 0 & 0 & 0 \\
\text{Category 2 (0,1)} & 1 & 0 & 0 \\
\text{Category 3 (0,2)} & 1 & 1 & 0 \\
\text{Category 4 (1,0)} & 1 & 0 & 0 \\
\text{Category 5 (1,1)} & 1 & 1 & 0 \\
\text{Category 6 (1,2)} & 1 & 1 & 1
\end{array}$$

Figure 20. PADI design system Design Matrix for BioKIDS Item 5.



View Design Matrix

View the Design Matrix that is part of BIOKIDS Item 5 MM.

[Edit Matrix]

Categories for OV: BioKIDs pre/post item 5	Difficulty of step 1	Difficulty of step 2	Difficulty of step 3	Difficulty of step 4	Difficulty of step 5
0	0	0	0	0	0
1	1	0	0	0	0
2	1	1	0	0	0
3	1	1	1	0	0
4	1	1	1	1	0
5	1	1	1	1	1

Comment:

Another approach is to create parameters associated with the Student Model Variables. For example, a response in the second category may be associated with the difficulty of achieving a response at step 1 on the second Student Model Variable for the aggregate item, denoted $\delta_{D2,1}$ in the Design Matrix below. In this case, the Design Matrix parameters simply reflect the combined difficulty of getting the two response categories, one for each Student Model Variable. For example, to achieve an overall response in the third category, the respondent needs enough ability to achieve at the third category level on the second Student Model Variable ($\delta_{D2,1} + \delta_{D2,2}$), but no incremental ability for the first Student Model Variable is required.

Design Matrix for Parameters Associated with Student Model Variables:

$$\begin{array}{lccc}
 & \delta_{D1.1} & \delta_{D2.1} & \delta_{D2.2} \\
\text{Category 1 (0,0)} & 0 & 0 & 0 \\
\text{Category 2 (0,1)} & 0 & 1 & 0 \\
\text{Category 3 (0,2)} & 0 & 1 & 1 \\
\text{Category 4 (1,0)} & 1 & 0 & 0 \\
\text{Category 5 (1,1)} & 1 & 1 & 0 \\
\text{Category 6 (1,2)} & 1 & 1 & 1
\end{array}$$

In some cases, the Design Matrix may need to change to reflect a more complex conceptualization of item difficulties that includes interaction effects among multiple Student

Model Variables. For example, the first item parameter may represent the difficulty of the first Student Model Variable, conditioned on a response in the first category on the second Student Model Variable. The second parameter may represent the difficulty of getting a response in the second category on the second Student Model Variable, conditioned on a response in the first category on the first Student Model Variable.

Determining whether parameters are dependent or independent usually requires empirical analysis. An MRCML analysis can be useful in determining which model provides the best fit to the data. Clearly, the manner in which items are calibrated must be reflected in the Scoring and Design Matrices when proficiency estimates are subsequently requested of the Scoring Engine.

Similarly, the selection of between-item or within-item multidimensionality also should be confirmed empirically. Although task designers may have a hypothesis about how various Student Model Variables work together and whether responses are conditionally dependent or independent, an analysis of alternative models may provide additional information that leads to new insights about the processes involved in performance of the task.

5.2 Modeling Observable Variables That Are Not Conditionally Independent

These examples describe the modeling of assessment tasks in which individual responses are considered dependent. A bundling procedure is implemented prior to generating the Measurement Models and Observable Variables that will be sent to the Scoring Engine.

5.2.1 A Simple Bundling Example

When a single prompt leads to multiple Work Products and responses from students, it is likely that the responses have some conditional dependencies. For example, if a prompt asks students to compute the average distance traveled by three objects and the intermediate responses giving the distance traveled for each object are scored, then the final response depends to some extent on the intermediate responses.

If we use only the final response to compute proficiency estimates, conditional dependence is not an issue; however, if we wish to capture more of the information available about student thinking, we will want to retain the information from the intermediate responses, and the conditional dependencies must be modeled in some way.

An *item bundle* can be used to model dependencies between items. The bundling is implemented prior to sending the data to the Scoring Engine. First, individual item responses are evaluated, and then a procedure for combining the intermediate item responses into a new, aggregated (bundled) response is implemented. Only the final bundled response is transmitted to the Scoring Engine and used in estimating proficiencies.

In the PADI design system, bundling is implemented in the Evaluation Phases during scoring of the Observable Variables. First, individual Observable Variables are evaluated; then the procedure for combining Observable Variables into a new Observable Variable is implemented, resulting in a single “bundled” Observable Variable. As in the within-item multidimensional case above, the intermediate Observable Variables are not sent to the Scoring Engine; only the final bundled Observable Variable is used in estimating proficiencies.

For the simple unidimensional case, consider three dichotomous Observable Variables in the bundle, with each mapping to the same Student Model Variable. One can use a complete model with all possible score combinations mapping to a unique final response category, or one can use a reduced model if some of the possible response categories are not needed or if it makes sense to collapse some categories.

The bundle, rather than individual Observable Variables, maps to the Scoring Matrix and the Design Matrix. In this example, the bundle has eight response patterns (the number of representations of three observable variables with two categories each), represented by eight response categories. We refer to the case in which each pattern is associated with a unique score as an “ordered bundle.”

Ordered Scoring Matrix:

$$\begin{array}{l|c|}
 & \text{SMV}_1 \\
\text{Category 1 (0,0,0)} & 0 \\
\text{Category 2 (0,0,1)} & 1 \\
\text{Category 3 (0,1,0)} & 2 \\
\text{Category 4 (0,1,1)} & 3 \\
\text{Category 5 (1,0,0)} & 4 \\
\text{Category 6 (1,0,1)} & 5 \\
\text{Category 7 (1,1,0)} & 6 \\
\text{Category 8 (1,1,1)} & 7
\end{array}$$

In this Scoring Matrix, a response pattern consisting of a response in the first category of item 1, a response in the second category of item 2, and a response in the first category of item 3 (i.e., incorrect responses on items 1 and 3 and a correct response on item 2) would be associated with the score of 2.

This item bundle can be treated like a partial credit item, and construction of the Design Matrix would follow from the example in Section 5.1.2. The matrix would have seven columns, one for each step.

A *partially ordered* Scoring Matrix (i.e., we can differentiate among bundle *sum scores* of 0, 1, 2, or 3, but not among bundle *categories* 2, 3, and 4 or *categories* 5, 6, and 7) is shown below:

Partially Ordered Scoring Matrix:

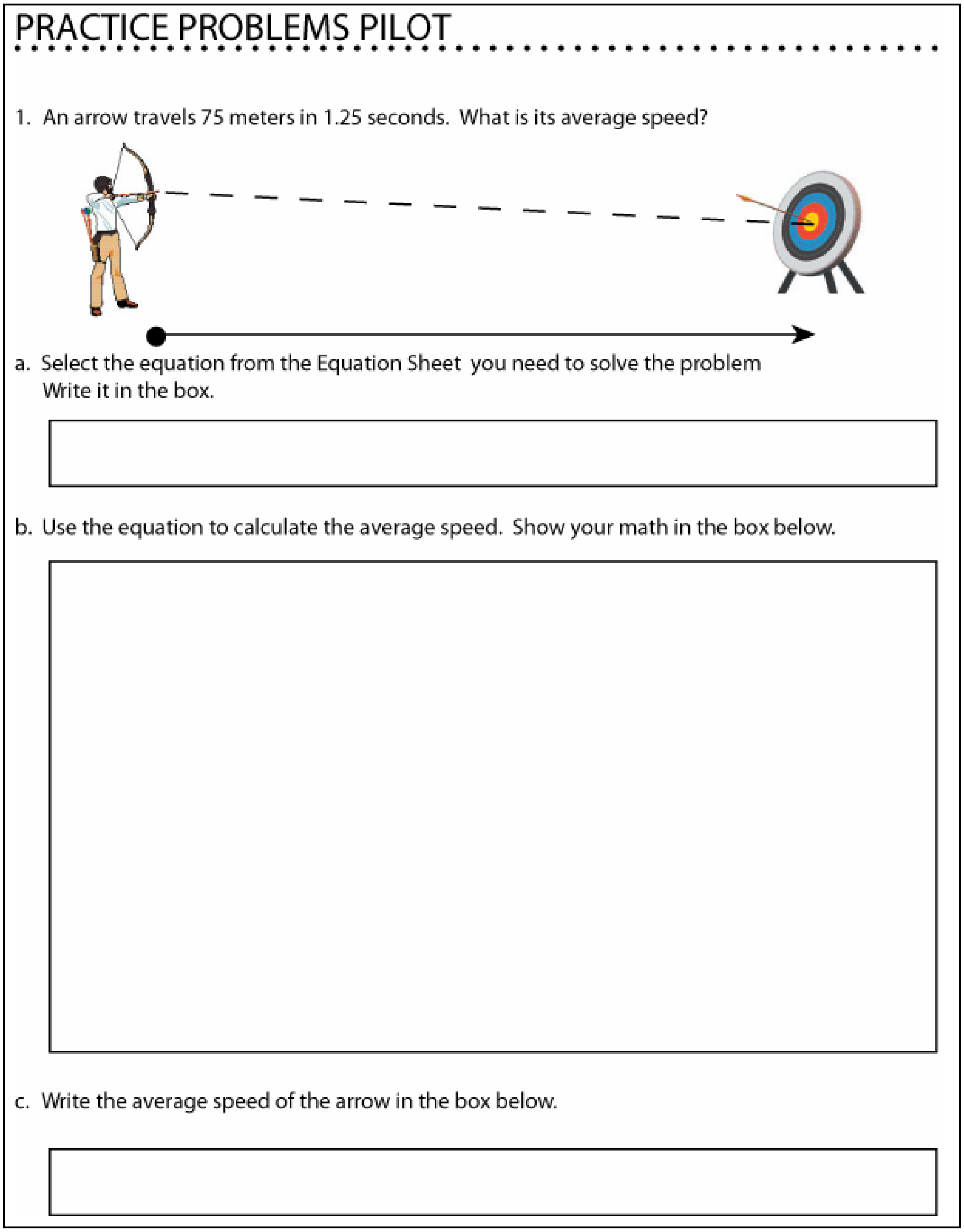
$$\begin{array}{lc} & \text{SMV}_1 \\ \begin{array}{l} \text{Category 1 (0,0,0)} \\ \text{Category 2 (0,0,1)} \\ \text{Category 3 (0,1,0)} \\ \text{Category 4 (0,1,1)} \\ \text{Category 5 (1,0,0)} \\ \text{Category 6 (1,0,1)} \\ \text{Category 7 (1,1,0)} \\ \text{Category 8 (1,1,1)} \end{array} & \begin{bmatrix} 0 \\ 1 \\ 1 \\ 1 \\ 2 \\ 2 \\ 2 \\ 3 \end{bmatrix} \end{array}$$

This is another type of partial credit model, and the Design Matrix again would follow from the example in Section 5.1.2. This Design Matrix could have three columns, one for each score category. Alternatively, one could design a saturated Design Matrix with a parameter for each response category, resulting in a matrix with seven columns.

5.2.2 Between-Item Multidimensional Bundle

In some cases, individual responses are conditionally dependent and are also indicators of different Student Model Variables. For example, in an interactive assessment of physics knowledge, students are required to select an appropriate equation for solving a speed problem (OV_1), place the correct values into the equation (OV_2), and then compute the average speed (OV_3). An example of this type of problem is shown in Figure 21.

Figure 21. A between-item multidimensional bundle.[1]



PRACTICE PROBLEMS PILOT

1. An arrow travels 75 meters in 1.25 seconds. What is its average speed?

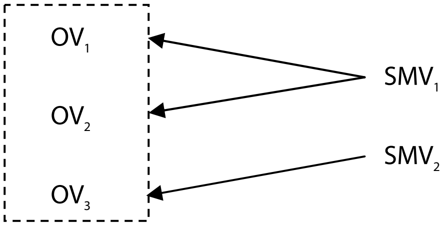
a. Select the equation from the Equation Sheet you need to solve the problem Write it in the box.

b. Use the equation to calculate the average speed. Show your math in the box below.

c. Write the average speed of the arrow in the box below.

[1] *From "FOSS Middle School Course Force and Motion Practice Problems" [computer software], developed at the Lawrence Hall of Science. Copyright 2004 by the Regents of the University of California. Reprinted with permission of the author.*

Clearly, the three responses are conditionally dependent because selecting the wrong equation will usually lead to the wrong final answer, as will selecting the wrong values for the variables in the equation. However, selecting the equation and choosing the correct values for the variables provide evidence about the student's knowledge of physics, while solving the equation provides evidence of mathematical ability. In this example, Observable Variables 1 and 2 are indicators of SMV_1 (physics), and Observable Variable 3 is an indicator of SMV_2 (mathematics).



First, the three responses are evaluated individually as correct or incorrect or as a response in the first category or a response in the second category. Then, the appropriate bundle category is determined from the pattern of responses on the three items. Note that this example is similar to the example in Section 5.1.4, but here the items are treated as conditionally dependent.

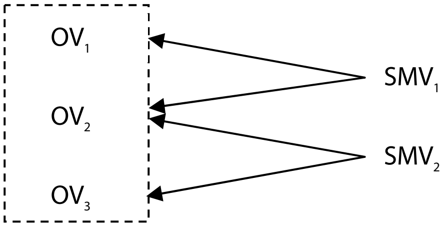
Scoring Matrix:

$$\begin{array}{lcl}
 & \begin{array}{cc} \text{SMV}_1 & \text{SMV}_2 \end{array} & \\
\begin{array}{l} \text{Category 1 (0,0,0)} \\ \text{Category 2 (0,0,1)} \\ \text{Category 3 (0,1,0)} \\ \text{Category 4 (0,1,1)} \\ \text{Category 5 (1,0,0)} \\ \text{Category 6 (1,0,1)} \\ \text{Category 7 (1,1,0)} \\ \text{Category 8 (1,1,1)} \end{array} &
\begin{bmatrix} 0 & 0 \\ 0 & 1 \\ 1 & 0 \\ 1 & 1 \\ 1 & 0 \\ 1 & 1 \\ 2 & 0 \\ 2 & 1 \end{bmatrix} &
\begin{array}{l} \\ \text{from OV 3 only} \\ \text{from OV 2 only} \\ \text{from OVs 2 and 3} \\ \text{from OV 1 only} \\ \text{from OVs 1 and 3} \\ \text{from OVs 1 and 2} \\ \text{from OVs 1, 2 and 3} \end{array}
\end{array}$$

The Design Matrix would follow any of the forms suggested in the example from this section.

5.2.3 Within-Item Multidimensional Bundle

If, instead of associating each Observable Variable with one Student Model Variable, we were to associate one Observable Variable with multiple Student Model Variables in the example from Section 5.2.2, we would need to construct a within-item multidimensional bundle. For example, we may believe that selecting the correct values to place into the equation (from the example from Section 5.2.2) requires both physics knowledge and mathematical ability. In that case, Observable Variable 1 is an indicator of the physics Student Model Variable, Observable Variable 3 is an indicator of the mathematics Student Model Variable, and Observable Variable 2 is an indicator of both physics and mathematics.



Scoring Matrix:

$$\begin{array}{l|cc|l}
 & \text{SMV}_1 & \text{SMV}_2 & \\
\text{Category 1 (0,0,0)} & 0 & 0 & \\
\text{Category 2 (0,0,1)} & 0 & 1 & \text{from OV 3 only} \\
\text{Category 3 (0,1,0)} & 1 & 1 & \text{from OV 2 only} \\
\text{Category 4 (0,1,1)} & 1 & 2 & \text{from OVs 2 and 3} \\
\text{Category 5 (1,0,0)} & 1 & 0 & \text{from OV 1 only} \\
\text{Category 6 (1,0,1)} & 1 & 1 & \text{from OV 1 and 3} \\
\text{Category 7 (1,1,0)} & 2 & 1 & \text{from OVs 1 and 2} \\
\text{Category 8 (1,1,1)} & 2 & 2 & \text{from all OVs}
\end{array}$$

The associated Design Matrix would also need to capture any interaction effects between the two dimensions (as in the example from Section 5.1.4).

5.3 Modeling a Complete Assessment

The MRCML literature generally refers to Scoring and Design Matrices for an entire assessment. The BEAR Scoring Engine, on the other hand, expects Measurement Models to be constructed at the Observable Variable level. This approach encourages reuse of components with similar measurement features. The Scoring Engine constructs a complete assessment Measurement Model from these individual Observable Variable models. The following examples of assessment-oriented matrices are shown to assist the reader in differentiating the approach used by the Scoring Engine from that used by assessment-oriented MRCML programs, such as ConQuest (Wu, Adams, & Wilson, 2005) and GradeMap (Kennedy, Wilson, & Draney, 2005).

5.3.1 Unidimensional Dichotomous Model

In the case of an assessment with 10 dichotomous Observable Variables, the associated matrices would have the form:

Assessment Scoring Matrix:

$$\begin{array}{l|c|}
 & SMV_1 \\
\text{OV 1, category 1} & 0 \\
\text{OV 1, category 2} & 1 \\
\text{OV 2, category 1} & 0 \\
\text{OV 2, category 2} & 1 \\
\text{OV 3, category 1} & 0 \\
\text{OV 3, category 2} & 1 \\
\text{OV 4, category 1} & 0 \\
\text{OV 4, category 2} & 1 \\
\text{OV 5, category 1} & 0 \\
\text{OV 5, category 2} & 1 \\
\text{OV 6, category 1} & 0 \\
\text{OV 6, category 2} & 1 \\
\text{OV 7, category 1} & 0 \\
\text{OV 7, category 2} & 1 \\
\text{OV 8, category 1} & 0 \\
\text{OV 8, category 2} & 1 \\
\text{OV 9, category 1} & 0 \\
\text{OV 9, category 2} & 1 \\
\text{OV 10, category 1} & 0 \\
\text{OV 10, category 2} & 1
\end{array}$$

Assessment Design Matrix:

$$\begin{array}{l|cccccccccc|}
 & \delta_1 & \delta_2 & \delta_3 & \delta_4 & \delta_5 & \delta_6 & \delta_7 & \delta_8 & \delta_9 & \delta_{10} \\
\text{OV 1, category 1} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\text{OV 1, category 2} & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\text{OV 2, category 1} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\text{OV 2, category 2} & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\text{OV 3, category 1} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\text{OV 3, category 2} & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\text{OV 4, category 1} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\text{OV 4, category 2} & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
\text{OV 5, category 1} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\text{OV 5, category 2} & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
\text{OV 6, category 1} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\text{OV 6, category 2} & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
\text{OV 7, category 1} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\text{OV 7, category 2} & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
\text{OV 8, category 1} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\text{OV 8, category 2} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
\text{OV 9, category 1} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\text{OV 9, category 2} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
\text{OV 10, category 1} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\text{OV 10, category 2} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1
\end{array}$$

5.3.2 Unidimensional Partial Credit Model

For an assessment with five Observable Variables in which OVs 1 through 3 have five categories and OVs 4 and 5 have three categories, the matrices would take the form:

Assessment Scoring Matrix:

$$\begin{array}{l|c|}
 & \text{SMV}_1 \\
\text{OV 1, category 1} & 0 \\
\text{OV 1, category 2} & 1 \\
\text{OV 1, category 3} & 2 \\
\text{OV 1, category 4} & 3 \\
\text{OV 1, category 5} & 4 \\
\text{OV 2, category 1} & 0 \\
\text{OV 2, category 2} & 1 \\
\text{OV 2, category 3} & 2 \\
\text{OV 2, category 4} & 3 \\
\text{OV 2, category 5} & 4 \\
\text{OV 3, category 1} & 0 \\
\text{OV 3, category 2} & 1 \\
\text{OV 3, category 3} & 2 \\
\text{OV 3, category 4} & 3 \\
\text{OV 3, category 5} & 4 \\
\text{OV 4, category 1} & 0 \\
\text{OV 4, category 2} & 1 \\
\text{OV 4, category 3} & 2 \\
\text{OV 5, category 1} & 0 \\
\text{OV 5, category 2} & 1 \\
\text{OV 5, category 3} & 2
\end{array}$$

Assessment Design Matrix:

$$\begin{array}{l|cccccccccccccccc|}
 & \delta_{11} & \delta_{12} & \delta_{13} & \delta_{14} & \delta_{21} & \delta_{22} & \delta_{23} & \delta_{24} & \delta_{31} & \delta_{32} & \delta_{33} & \delta_{34} & \delta_{41} & \delta_{42} & \delta_{51} & \delta_{52} \\
\text{OV 1, category 1} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\text{OV 1, category 2} & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\text{OV 1, category 3} & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\text{OV 1, category 4} & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\text{OV 1, category 5} & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\text{OV 2, category 1} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\text{OV 2, category 2} & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\text{OV 2, category 3} & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\text{OV 2, category 4} & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\text{OV 2, category 5} & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\text{OV 3, category 1} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\text{OV 3, category 2} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\text{OV 3, category 3} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
\text{OV 3, category 4} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\
\text{OV 3, category 5} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\
\text{OV 4, category 1} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\text{OV 4, category 2} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
\text{OV 4, category 3} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\
\text{OV 5, category 1} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\text{OV 5, category 2} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
\text{OV 5, category 3} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1
\end{array}$$

5.3.3 Unidimensional Rating Scale Model

The rating scale model is considered a special case of the partial credit model. All Observable Variables of the rating scale type for a particular assessment must use the same parameter estimates for the steps. The matrices below are appropriate for an assessment with five four-category Observable Variables.

Assessment Scoring Matrix:

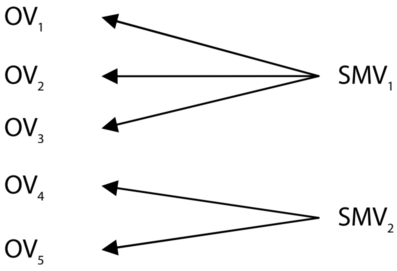
$$\begin{array}{l|c|}
 & \text{SMV}_1 \\
\text{OV 1, category 1} & 0 \\
\text{OV 1, category 2} & 1 \\
\text{OV 1, category 3} & 2 \\
\text{OV 1, category 4} & 3 \\
\text{OV 2, category 1} & 0 \\
\text{OV 2, category 2} & 1 \\
\text{OV 2, category 3} & 2 \\
\text{OV 2, category 4} & 3 \\
\text{OV 3, category 1} & 0 \\
\text{OV 3, category 2} & 1 \\
\text{OV 3, category 3} & 2 \\
\text{OV 3, category 4} & 3 \\
\text{OV 4, category 1} & 0 \\
\text{OV 4, category 2} & 1 \\
\text{OV 4, category 3} & 2 \\
\text{OV 4, category 4} & 3 \\
\text{OV 5, category 1} & 0 \\
\text{OV 5, category 2} & 1 \\
\text{OV 5, category 3} & 2 \\
\text{OV 5, category 4} & 3
\end{array}$$

Assessment Design Matrix:

$$\begin{array}{l|ccccccc|}
 & \delta_1 & \delta_2 & \delta_3 & \delta_4 & \delta_5 & \tau_1 & \tau_2 \\
\text{OV 1, category 1} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\text{OV 1, category 2} & 1 & 0 & 0 & 0 & 0 & 1 & 0 \\
\text{OV 1, category 3} & 2 & 0 & 0 & 0 & 0 & 1 & 1 \\
\text{OV 1, category 4} & 3 & 0 & 0 & 0 & 0 & 0 & 0 \\
\text{OV 2, category 1} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\text{OV 2, category 2} & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\
\text{OV 2, category 3} & 0 & 2 & 0 & 0 & 0 & 1 & 1 \\
\text{OV 2, category 4} & 0 & 3 & 0 & 0 & 0 & 0 & 0 \\
\text{OV 3, category 1} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\text{OV 3, category 2} & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\
\text{OV 3, category 3} & 0 & 0 & 2 & 0 & 0 & 1 & 1 \\
\text{OV 3, category 4} & 0 & 0 & 3 & 0 & 0 & 0 & 0 \\
\text{OV 4, category 1} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\text{OV 4, category 2} & 0 & 0 & 0 & 1 & 0 & 1 & 0 \\
\text{OV 4, category 3} & 0 & 0 & 0 & 2 & 0 & 1 & 1 \\
\text{OV 4, category 4} & 0 & 0 & 0 & 3 & 0 & 0 & 0 \\
\text{OV 5, category 1} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\text{OV 5, category 2} & 0 & 0 & 0 & 0 & 1 & 1 & 0 \\
\text{OV 5, category 3} & 0 & 0 & 0 & 0 & 2 & 1 & 1 \\
\text{OV 5, category 4} & 0 & 0 & 0 & 0 & 3 & 0 & 0
\end{array}$$

5.3.4 Between-Item Multidimensional Model

For an assessment with five Observable Variables in which OVs 1 through 3 are indicators of the first Student Model Variable and have five categories and OVs 4 and 5 are indicators of the second Student Model Variable and have three categories, the assessment matrices would take the form:



Assessment Scoring Matrix:

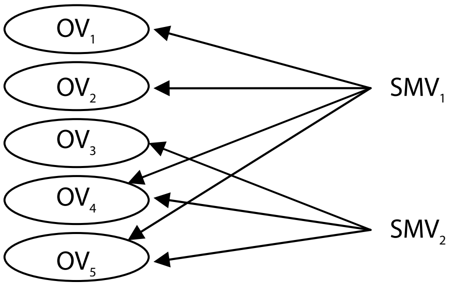
$$\begin{array}{l|cc}
 & \multicolumn{2}{c}{D_1} \\
\text{OV 1, category 1} & 0 & 0 \\
\text{OV 1, category 2} & 1 & 0 \\
\text{OV 1, category 3} & 2 & 0 \\
\text{OV 1, category 4} & 3 & 0 \\
\text{OV 1, category 5} & 4 & 0 \\
\text{OV 2, category 1} & 0 & 0 \\
\text{OV 2, category 2} & 1 & 0 \\
\text{OV 2, category 3} & 2 & 0 \\
\text{OV 2, category 4} & 3 & 0 \\
\text{OV 2, category 5} & 4 & 0 \\
\text{OV 3, category 1} & 0 & 0 \\
\text{OV 3, category 2} & 1 & 0 \\
\text{OV 3, category 3} & 2 & 0 \\
\text{OV 3, category 4} & 3 & 0 \\
\text{OV 3, category 5} & 4 & 0 \\
\text{OV 4, category 1} & 0 & 0 \\
\text{OV 4, category 2} & 0 & 1 \\
\text{OV 4, category 3} & 0 & 2 \\
\text{OV 5, category 1} & 0 & 0 \\
\text{OV 5, category 2} & 0 & 1 \\
\text{OV 5, category 3} & 0 & 2
\end{array}$$

Assessment Design Matrix:

$$\begin{array}{l|cccccccccccccccc}
 & \delta_{11} & \delta_{12} & \delta_{13} & \delta_{14} & \delta_{21} & \delta_{22} & \delta_{23} & \delta_{24} & \delta_{31} & \delta_{32} & \delta_{33} & \delta_{34} & \delta_{41} & \delta_{42} & \delta_{51} & \delta_{52} \\
\text{OV 1, category 1} & 0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0 \\
\text{OV 1, category 2} & 1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0 \\
\text{OV 1, category 3} & 1&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0 \\
\text{OV 1, category 4} & 1&1&1&0&0&0&0&0&0&0&0&0&0&0&0&0 \\
\text{OV 1, category 5} & 1&1&1&1&0&0&0&0&0&0&0&0&0&0&0&0 \\
\text{OV 2, category 1} & 0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0 \\
\text{OV 2, category 2} & 0&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0 \\
\text{OV 2, category 3} & 0&0&0&0&1&1&0&0&0&0&0&0&0&0&0&0 \\
\text{OV 2, category 4} & 0&0&0&0&1&1&1&0&0&0&0&0&0&0&0&0 \\
\text{OV 2, category 5} & 0&0&0&0&1&1&1&1&0&0&0&0&0&0&0&0 \\
\text{OV 3, category 1} & 0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0 \\
\text{OV 3, category 2} & 0&0&0&0&0&0&0&0&1&0&0&0&0&0&0&0 \\
\text{OV 3, category 3} & 0&0&0&0&0&0&0&0&1&1&0&0&0&0&0&0 \\
\text{OV 3, category 4} & 0&0&0&0&0&0&0&0&1&1&1&0&0&0&0&0 \\
\text{OV 3, category 5} & 0&0&0&0&0&0&0&0&1&1&1&1&0&0&0&0 \\
\text{OV 4, category 1} & 0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0 \\
\text{OV 4, category 2} & 0&0&0&0&0&0&0&0&0&0&0&0&1&0&0&0 \\
\text{OV 4, category 3} & 0&0&0&0&0&0&0&0&0&0&0&0&1&1&0&0 \\
\text{OV 5, category 1} & 0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0 \\
\text{OV 5, category 2} & 0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&0 \\
\text{OV 5, category 3} & 0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&1
\end{array}$$

5.3.5 Within-Item Multidimensional Partial Credit Model

For an assessment with five Observable Variables in which OVs 1 and 2 are indicators of the first Student Model Variable with five categories, OV 3 is an indicator of the second Student Model Variable with three categories, and OVs 4 and 5 are indicators of both Student Model Variables with three categories, the assessment matrices would take the form:



Assess. Scoring Matrix:

$$
\begin{array}{l|cc|}
 & SMV_1 & SMV_2 \\
\text{OV 1, category 1} & 0 & 0 \\
\text{OV 1, category 2} & 1 & 0 \\
\text{OV 1, category 3} & 2 & 0 \\
\text{OV 1, category 4} & 3 & 0 \\
\text{OV 1, category 5} & 4 & 0 \\
\text{OV 2, category 1} & 0 & 0 \\
\text{OV 2, category 2} & 1 & 0 \\
\text{OV 2, category 3} & 2 & 0 \\
\text{OV 2, category 4} & 3 & 0 \\
\text{OV 2, category 5} & 4 & 0 \\
\text{OV 3, category 1} & 0 & 0 \\
\text{OV 3, category 2} & 0 & 1 \\
\text{OV 3, category 3} & 0 & 2 \\
\text{OV 3, category 4} & 0 & 3 \\
\text{OV 3, category 5} & 0 & 4 \\
\text{OV 4, category 1} & 0 & 0 \\
\text{OV 4, category 2} & 1 & 1 \\
\text{OV 4, category 3} & 2 & 2 \\
\text{OV 5, category 1} & 0 & 0 \\
\text{OV 5, category 2} & 1 & 1 \\
\text{OV 5, category 3} & 2 & 2
\end{array}
$$

Assessment Design Matrix:

$$
\begin{array}{l|cccccccccccccccc|}
 & \delta_{11} & \delta_{12} & \delta_{13} & \delta_{14} & \delta_{21} & \delta_{22} & \delta_{23} & \delta_{24} & \delta_{31} & \delta_{32} & \delta_{33} & \delta_{34} & \delta_{41} & \delta_{42} & \delta_{51} & \delta_{52} \\
\text{OV 1, category 1} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\text{OV 1, category 2} & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\text{OV 1, category 3} & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\text{OV 1, category 4} & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\text{OV 1, category 5} & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\text{OV 2, category 1} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\text{OV 2, category 2} & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\text{OV 2, category 3} & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\text{OV 2, category 4} & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\text{OV 2, category 5} & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\text{OV 3, category 1} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\text{OV 3, category 2} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\text{OV 3, category 3} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
\text{OV 3, category 4} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\
\text{OV 3, category 5} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\
\text{OV 4, category 1} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\text{OV 4, category 2} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
\text{OV 4, category 3} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\
\text{OV 5, category 1} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\text{OV 5, category 2} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
\text{OV 5, category 3} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1
\end{array}
$$

6.0 Conclusions

These examples show how a number of assessment tasks, ranging from simple true/false questions to complex problems involving a series of constructed responses, could be modeled with the PADI design system. The data structures of the design system, particularly the Student Model, Evaluation Procedure, and Measurement Model objects, are used to represent the components that contribute to the design of a coherent assessment system. In particular, student Work Products are designed to elicit evidence about the student measures of interest as defined in a Student Model, Evaluation Procedures define how student work is to be evaluated and stored in Observable Variables, and Measurement Models specify how inferences about the student measures are to be drawn from the Observable Variables. The BEAR Scoring Engine implements a Rasch-based multidimensional item response model to arrive at proficiency estimates by using the set of Scoring Matrices, Design Matrices, and Calibration Parameters contained in PADI design system Measurement Models of the Observable Variables associated with an assessment.

Assessment developers can improve the interpretability and consistency of assessment measures by reusing PADI design system components in multiple tasks within an assessment system. Once the inferential structures of tasks are defined, developers can specify presentation details to generate a large number of assessment items. Different students can then be given different assessment tasks (or they can be given the same tasks) to produce comparable proficiency estimates. In addition, these measures can be used in a formative feedback loop and for longitudinal analyses of student change without the consistency problems associated with more traditional classroom testing environments, in which the tests change with the curriculum or from one grade to another. The PADI design system, paired with the BEAR Scoring Engine, brings advances in assessment and measurement research to an assessment developer’s toolbox.

References

Adams, R., Wilson, M., & Wang, W. (1997). The multidimensional random coefficients multinomial logit model. *Applied Psychological Measurement, 21*(1), 1-23.

Almond, R. G., Steinberg, L. S., & Mislevy, R. J. (2002). A four-process architecture for assessment delivery with connections to assessment design. *Journal of Technology, Learning, and Assessment, 1*(5), 1-64.

Connecticut State Department of Education. (2005). *Core science curriculum framework: Matrix of K-10 concept development*. Retrieved 3/17/05 from http://www.state.ct.us/sde/dtl/curriculum/science/framework/Matrix2005.doc

Embretson, S. E. (1996). The new rules of measurement. *Psychological Assessment, 8*(4), 341-349.

Full Option Science System (2004). *Self-assessment system v1.0*. [Computer software]. Berkeley, CA: Lawrence Hall of Science.

Hoskens, M., & De Boeck, P. (1997). A parametric model for local dependence among test items. *Psychological Methods, 2*, 261-277.

Kennedy, C. A., Wilson, M., & Draney, K. (2005). *GradeMap v4.2* [computer program]. Berkeley, CA: University of California, Berkeley Evaluation and Assessment Research Center.

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

National Research Council. (2001). *Knowing what students know: The science and design of educational assessment*. Committee on the Foundations of Assessment. J. W. Pellegrino, N. Chudowsky, & R. Glaser, (Eds.). Washington, DC: National Academy Press.

National Research Council. (1996). *National Science Education Standards*. Committee on Science Education Standards and Assessment. Washington, DC: National Academy Press.

Rasch, G. (1960). *Probabilistic models for some intelligence and attainment tests.* Copenhagen: Danish Institute for Educational Research.

U.S. Department of Education. (n.d.). *NCLB: Stronger accountability—testing for results*. Retrieved January 10, 2005, from http://www.ed.gov/nclb/accountability/ayp/testingforresults.html

Wang, W., Wilson, M., & Cheng, Y. (2000). *Local dependence between latent traits when common stimuli are used*. Paper presented at the International Objective Measurement Workshop, New Orleans, LA.

Wilson, M. (1992). The ordered partition model: An extension of the partial credit model. *Applied Psychological Measurement, 16*(3), 309-325.

Wilson, M., & Adams R. J. (1995). Rasch models for item bundles. *Psychometrika*, *60*, 181-198.

Wisconsin Department of Public Instruction. (2005). *Wisconsin Model Academic Standards: Science introduction.* Retrieved March 17, 2005 from http://www.dpi.state.wi.us/standards/sciintro.html

Wright, B. D. (1993). Equitable test equating. *Rasch Measurement Transactions, 7(2)*, 298-299.

Wright, B. D. (1968). Sample-free test calibration and person measurement. In *Proceedings of the 1967 Invitational Conference on Testing* (pp. 85-101). Princeton, NJ: Educational Testing Service.

Wright, B. D. (1977). Solving measurement problems with the Rasch model. *Journal of Educational Measurement*, *14*, 97-116.

Wu, M., Adams, R., & Wilson, M. (2005). *ACER ConQuest* [computer program]. Hawthorn, Australia: Australian Council for Educational Research.

Appendix A—Input and Output XML Documents for the BEAR Scoring Engine

A.1 Input Documents

Input to the BEAR Scoring Engine consists of two XML documents: a model document and a student results document. The student results document is IMS/QTI compliant, while the model document is designed specifically to transmit the information needed to construct the MRCML algorithms used by the Scoring Engine. Complete definitions of the schemas are available on the Scoring Engine Web page at http://bearcenter.berkeley.edu/padi/. In this appendix, we cover the basic elements of the schemas as they relate to the MRCML information required by the BEAR Scoring Engine.

A.1.1 Model Document

Essential Student Model information is stored in the following XML elements of the model document, shown in Figure A-1:

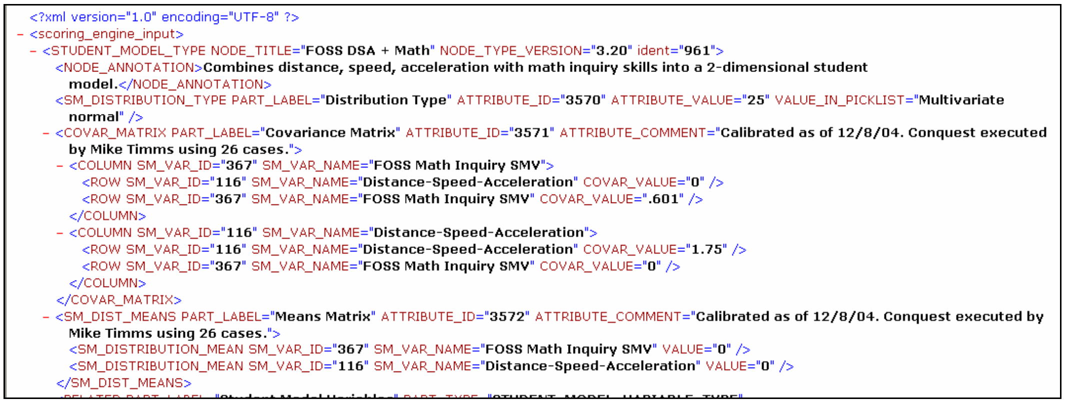
<SM_DISTRIBUTION_TYPE>

<COVAR_MATRIX>

<SM_DIST_MEAN>

The distribution type specified in the figure is "Multivariate normal," with two Student Model Variables indicated, "Distance-Speed-Acceleration" and "FOSS Math Inquiry SMV." The population model consists of a covariance matrix, $\begin{bmatrix} .601 & 0 \\ 0 & 1.75 \end{bmatrix}$, and a means matrix, $\begin{bmatrix} 0 & 0 \end{bmatrix}$.

Figure A-1. Student Model information in the XML model file.



```
  <?xml version="1.0" encoding="UTF-8" ?>
- <scoring_engine_input>
  - <STUDENT_MODEL_TYPE NODE_TITLE="FOSS DSA + Math" NODE_TYPE_VERSION="3.20" ident="961">
      <NODE_ANNOTATION>Combines distance, speed, acceleration with math inquiry skills into a 2-dimensional student
        model.</NODE_ANNOTATION>
      <SM_DISTRIBUTION_TYPE PART_LABEL="Distribution Type" ATTRIBUTE_ID="3570" ATTRIBUTE_VALUE="25" VALUE_IN_PICKLIST="Multivariate
        normal" />
    - <COVAR_MATRIX PART_LABEL="Covariance Matrix" ATTRIBUTE_ID="3571" ATTRIBUTE_COMMENT="Calibrated as of 12/8/04. Conquest executed
        by Mike Timms using 26 cases.">
      - <COLUMN SM_VAR_ID="367" SM_VAR_NAME="FOSS Math Inquiry SMV">
          <ROW SM_VAR_ID="116" SM_VAR_NAME="Distance-Speed-Acceleration" COVAR_VALUE="0" />
          <ROW SM_VAR_ID="367" SM_VAR_NAME="FOSS Math Inquiry SMV" COVAR_VALUE=".601" />
        </COLUMN>
      - <COLUMN SM_VAR_ID="116" SM_VAR_NAME="Distance-Speed-Acceleration">
          <ROW SM_VAR_ID="116" SM_VAR_NAME="Distance-Speed-Acceleration" COVAR_VALUE="1.75" />
          <ROW SM_VAR_ID="367" SM_VAR_NAME="FOSS Math Inquiry SMV" COVAR_VALUE="0" />
        </COLUMN>
      </COVAR_MATRIX>
    - <SM_DIST_MEANS PART_LABEL="Means Matrix" ATTRIBUTE_ID="3572" ATTRIBUTE_COMMENT="Calibrated as of 12/8/04. Conquest executed by
        Mike Timms using 26 cases.">
        <SM_DISTRIBUTION_MEAN SM_VAR_ID="367" SM_VAR_NAME="FOSS Math Inquiry SMV" VALUE="0" />
        <SM_DISTRIBUTION_MEAN SM_VAR_ID="116" SM_VAR_NAME="Distance-Speed-Acceleration" VALUE="0" />
      </SM_DIST_MEANS>
      <RELATED PART_LABEL="Student Model Variables" PART_TYPE="STUDENT_MODEL_VARIABLE_TYPE"
```

Measurement Model information is organized by scorable units, or Observable Variables, with exactly one Measurement Model for each Observable Variable. The collection of Measurement Modes received by the Scoring Engine is assembled into an assessment Measurement Model. Each OV Measurement Model must contain the following XML elements of the model document:

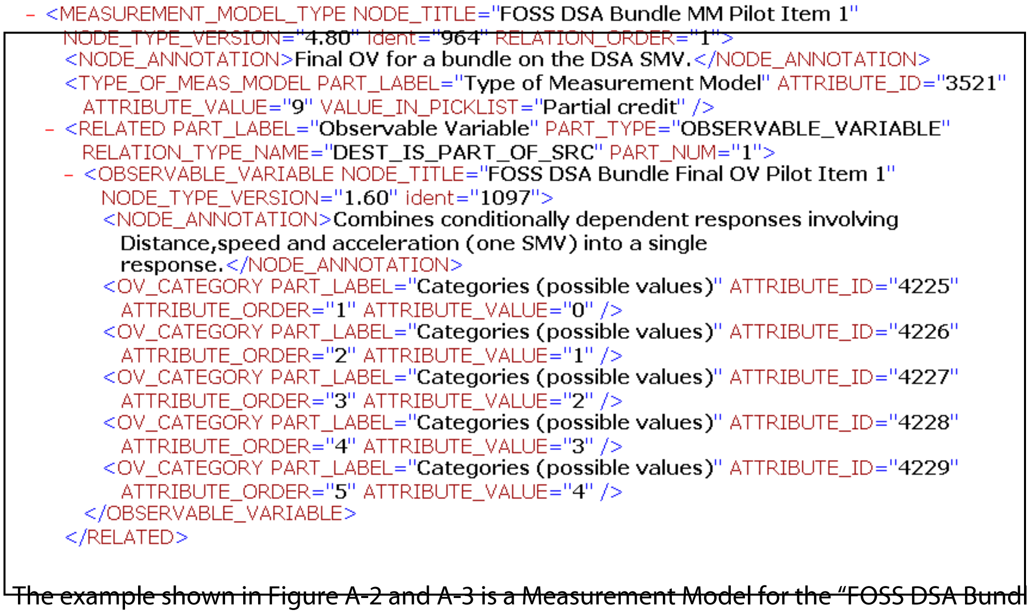
<OBSERVABLE_VARIABLE>

<SCORING_MATRIX>

<DESIGN_MATRIX>

<CALIBRATION_PARAMETERS>

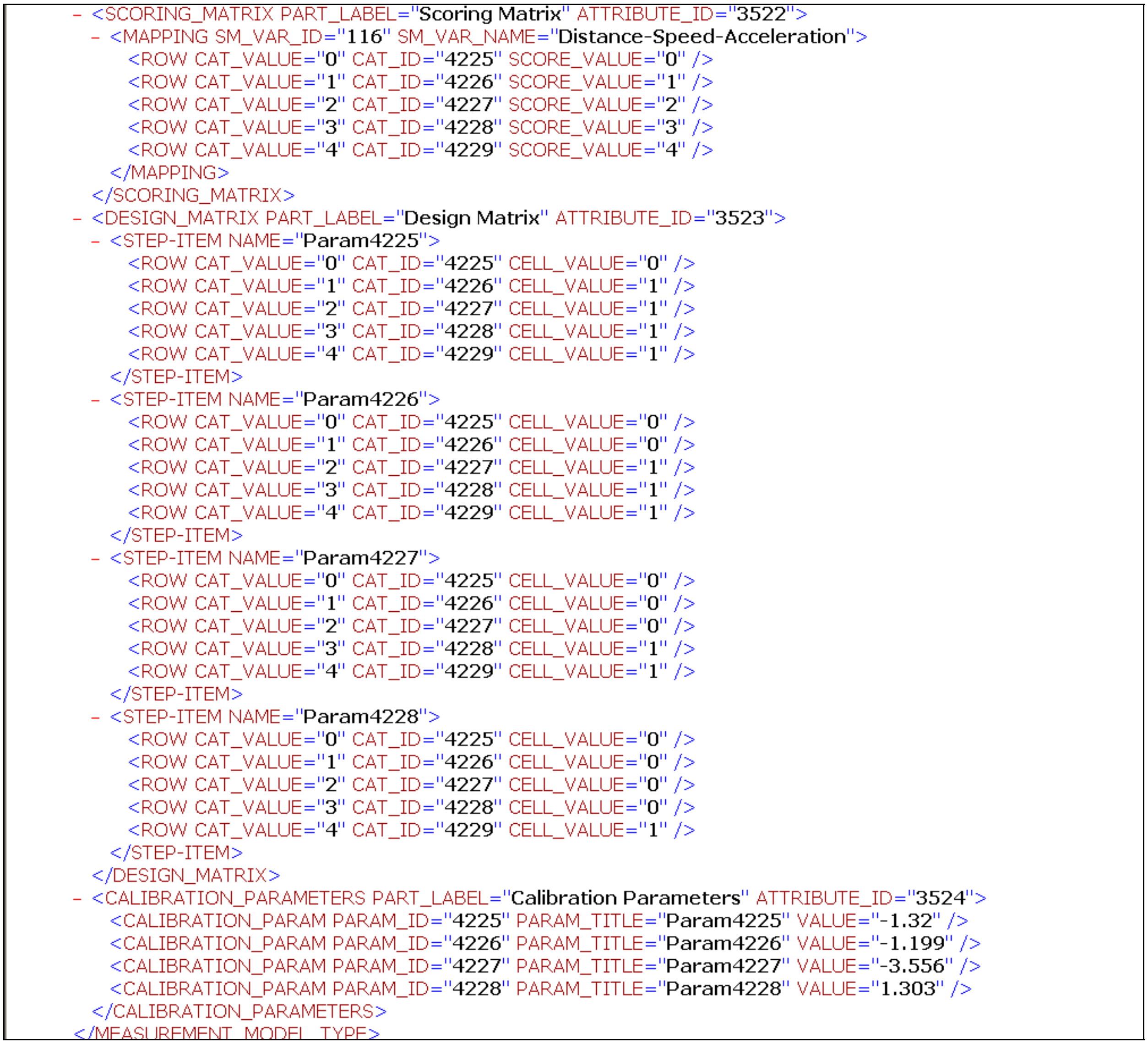
Figure A-2. Measurement Model part of XML model specification file showing the Observable Variable elements for the "FOSS DSA MM Pilot Item 1" Observable Variable.



```
- <MEASUREMENT_MODEL_TYPE NODE_TITLE="FOSS DSA Bundle MM Pilot Item 1"
    NODE_TYPE_VERSION="4.80" ident="964" RELATION_ORDER="1">
    <NODE_ANNOTATION>Final OV for a bundle on the DSA SMV.</NODE_ANNOTATION>
    <TYPE_OF_MEAS_MODEL PART_LABEL="Type of Measurement Model" ATTRIBUTE_ID="3521"
      ATTRIBUTE_VALUE="9" VALUE_IN_PICKLIST="Partial credit" />
  - <RELATED PART_LABEL="Observable Variable" PART_TYPE="OBSERVABLE_VARIABLE"
      RELATION_TYPE_NAME="DEST_IS_PART_OF_SRC" PART_NUM="1">
    - <OBSERVABLE_VARIABLE NODE_TITLE="FOSS DSA Bundle Final OV Pilot Item 1"
        NODE_TYPE_VERSION="1.60" ident="1097">
        <NODE_ANNOTATION>Combines conditionally dependent responses involving
          Distance,speed and acceleration (one SMV) into a single
          response.</NODE_ANNOTATION>
        <OV_CATEGORY PART_LABEL="Categories (possible values)" ATTRIBUTE_ID="4225"
          ATTRIBUTE_ORDER="1" ATTRIBUTE_VALUE="0" />
        <OV_CATEGORY PART_LABEL="Categories (possible values)" ATTRIBUTE_ID="4226"
          ATTRIBUTE_ORDER="2" ATTRIBUTE_VALUE="1" />
        <OV_CATEGORY PART_LABEL="Categories (possible values)" ATTRIBUTE_ID="4227"
          ATTRIBUTE_ORDER="3" ATTRIBUTE_VALUE="2" />
        <OV_CATEGORY PART_LABEL="Categories (possible values)" ATTRIBUTE_ID="4228"
          ATTRIBUTE_ORDER="4" ATTRIBUTE_VALUE="3" />
        <OV_CATEGORY PART_LABEL="Categories (possible values)" ATTRIBUTE_ID="4229"
          ATTRIBUTE_ORDER="5" ATTRIBUTE_VALUE="4" />
      </OBSERVABLE_VARIABLE>
    </RELATED>
```

The example shown in Figure A-2 and A-3 is a Measurement Model for the "FOSS DSA Bundle MM Pilot Item 1" Observable Variable. We note in Figure A-2 that the Observable Variable has five response categories. In Figure A-3 we see that the scoring matrix has a single column and is associated with the "Distance-Speed-Acceleration" Student Model Variable. The Design Matrix contains a <STEP–ITEM> element for each of the four step parameters (recall that the number of step parameters is one fewer than the number of categories). All four Calibration Parameter values are shown for the Measurement Model.

Figure A-3. Measurement Model part of XML model specification file showing the Scoring Matrix, Design Matrix, and Calibration Parameter elements for the “FOSS DSA MM Pilot Item 1” Observable Variable.



```
- <SCORING_MATRIX PART_LABEL="Scoring Matrix" ATTRIBUTE_ID="3522">
  - <MAPPING SM_VAR_ID="116" SM_VAR_NAME="Distance-Speed-Acceleration">
      <ROW CAT_VALUE="0" CAT_ID="4225" SCORE_VALUE="0" />
      <ROW CAT_VALUE="1" CAT_ID="4226" SCORE_VALUE="1" />
      <ROW CAT_VALUE="2" CAT_ID="4227" SCORE_VALUE="2" />
      <ROW CAT_VALUE="3" CAT_ID="4228" SCORE_VALUE="3" />
      <ROW CAT_VALUE="4" CAT_ID="4229" SCORE_VALUE="4" />
    </MAPPING>
  </SCORING_MATRIX>
- <DESIGN_MATRIX PART_LABEL="Design Matrix" ATTRIBUTE_ID="3523">
  - <STEP-ITEM NAME="Param4225">
      <ROW CAT_VALUE="0" CAT_ID="4225" CELL_VALUE="0" />
      <ROW CAT_VALUE="1" CAT_ID="4226" CELL_VALUE="1" />
      <ROW CAT_VALUE="2" CAT_ID="4227" CELL_VALUE="1" />
      <ROW CAT_VALUE="3" CAT_ID="4228" CELL_VALUE="1" />
      <ROW CAT_VALUE="4" CAT_ID="4229" CELL_VALUE="1" />
    </STEP-ITEM>
  - <STEP-ITEM NAME="Param4226">
      <ROW CAT_VALUE="0" CAT_ID="4225" CELL_VALUE="0" />
      <ROW CAT_VALUE="1" CAT_ID="4226" CELL_VALUE="0" />
      <ROW CAT_VALUE="2" CAT_ID="4227" CELL_VALUE="1" />
      <ROW CAT_VALUE="3" CAT_ID="4228" CELL_VALUE="1" />
      <ROW CAT_VALUE="4" CAT_ID="4229" CELL_VALUE="1" />
    </STEP-ITEM>
  - <STEP-ITEM NAME="Param4227">
      <ROW CAT_VALUE="0" CAT_ID="4225" CELL_VALUE="0" />
      <ROW CAT_VALUE="1" CAT_ID="4226" CELL_VALUE="0" />
      <ROW CAT_VALUE="2" CAT_ID="4227" CELL_VALUE="0" />
      <ROW CAT_VALUE="3" CAT_ID="4228" CELL_VALUE="1" />
      <ROW CAT_VALUE="4" CAT_ID="4229" CELL_VALUE="1" />
    </STEP-ITEM>
  - <STEP-ITEM NAME="Param4228">
      <ROW CAT_VALUE="0" CAT_ID="4225" CELL_VALUE="0" />
      <ROW CAT_VALUE="1" CAT_ID="4226" CELL_VALUE="0" />
      <ROW CAT_VALUE="2" CAT_ID="4227" CELL_VALUE="0" />
      <ROW CAT_VALUE="3" CAT_ID="4228" CELL_VALUE="0" />
      <ROW CAT_VALUE="4" CAT_ID="4229" CELL_VALUE="1" />
    </STEP-ITEM>
  </DESIGN_MATRIX>
- <CALIBRATION_PARAMETERS PART_LABEL="Calibration Parameters" ATTRIBUTE_ID="3524">
    <CALIBRATION_PARAM PARAM_ID="4225" PARAM_TITLE="Param4225" VALUE="-1.32" />
    <CALIBRATION_PARAM PARAM_ID="4226" PARAM_TITLE="Param4226" VALUE="-1.199" />
    <CALIBRATION_PARAM PARAM_ID="4227" PARAM_TITLE="Param4227" VALUE="-3.556" />
    <CALIBRATION_PARAM PARAM_ID="4228" PARAM_TITLE="Param4228" VALUE="1.303" />
  </CALIBRATION_PARAMETERS>
</MEASUREMENT_MODEL_TYPE>
```

A.1.2 Student Results Document

Student results data are organized by student, with one or more scores (Observable Variable values) per student. Essential results data are stored in the following XML elements of the Results document:

<name>

<item_result>

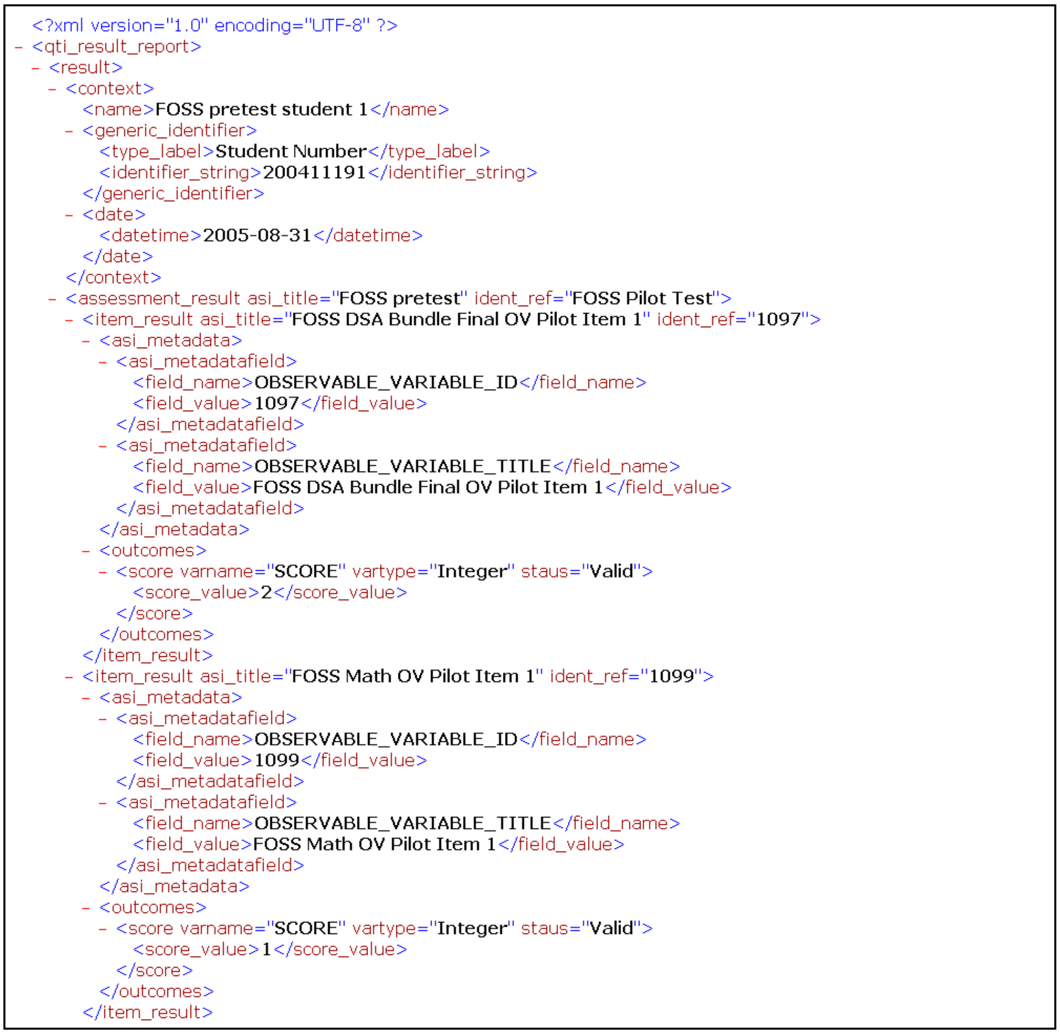
<score>

The connection from the <score> value in the Results XML document to the OBSERVABLE_VARIABLE measurement model fragment in the Model XML document (refer to Figure A-2) is made via <field_name >observable_variable_id <field_value> values in the Results XML document associated with corresponding <OBSERVABLE_VARIABLE> IDENT values in the Model XML document. In the excerpt shown in Figure A-4, the student name is “FOSS pretest student 1” and this student has a score of 2 on the “FOSS DSA Bundle Final OV Pilot Item 1” Observable Variable.

A.2 Output Document

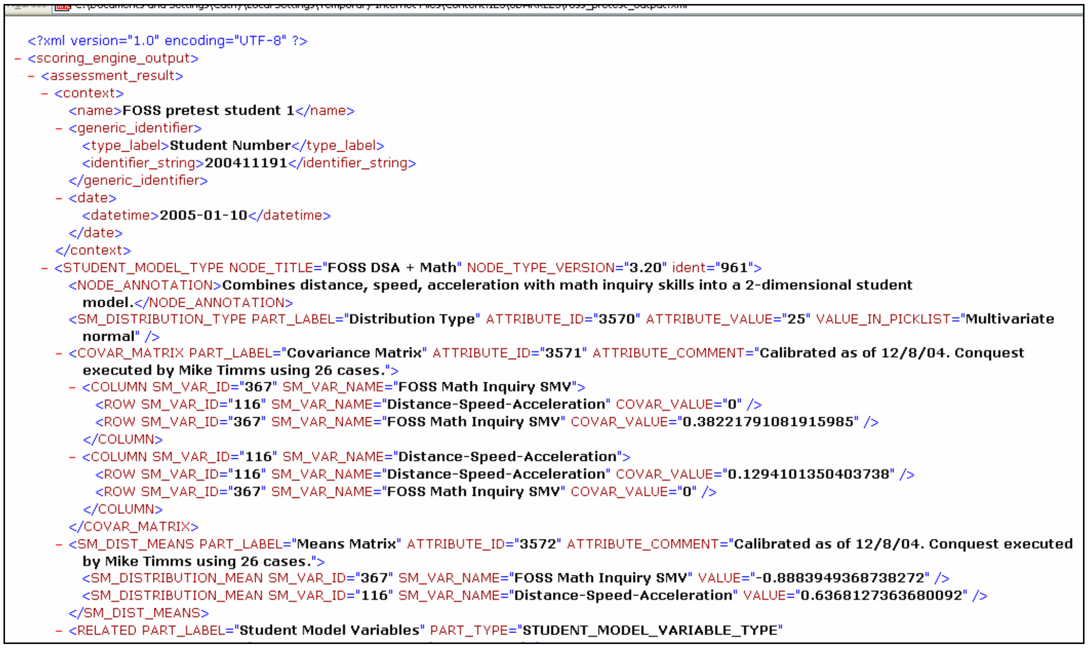
Proficiency estimates and standard errors are returned in another Student Results XML document, which is identical to the input document with values entered in two of the fields. As shown in Figure A-4, the file contains an <assessment_result> section for each student. The estimated proficiency on each Student Model Variable is found in the <SM_DIST_MEANS> element, while the standard error is found in the <COVAR_MATRIX> tag. In this case, the student, “FOSS pretest student 1” had a proficiency estimate of 0.637 on the “Distance-Speed-Acceleration” Student Model Variable and of .888 on the “Math Inquiry” Student Model Variable. Standard errors of the estimates were .129 and .382, respectively.

Figure A-4. Student results input file.



```
  <?xml version="1.0" encoding="UTF-8" ?>
- <qti_result_report>
  - <result>
    - <context>
        <name>FOSS pretest student 1</name>
      - <generic_identifier>
          <type_label>Student Number</type_label>
          <identifier_string>200411191</identifier_string>
        </generic_identifier>
      - <date>
          <datetime>2005-08-31</datetime>
        </date>
      </context>
    - <assessment_result asi_title="FOSS pretest" ident_ref="FOSS Pilot Test">
      - <item_result asi_title="FOSS DSA Bundle Final OV Pilot Item 1" ident_ref="1097">
        - <asi_metadata>
          - <asi_metadatafield>
              <field_name>OBSERVABLE_VARIABLE_ID</field_name>
              <field_value>1097</field_value>
            </asi_metadatafield>
          - <asi_metadatafield>
              <field_name>OBSERVABLE_VARIABLE_TITLE</field_name>
              <field_value>FOSS DSA Bundle Final OV Pilot Item 1</field_value>
            </asi_metadatafield>
          </asi_metadata>
        - <outcomes>
          - <score varname="SCORE" vartype="Integer" staus="Valid">
              <score_value>2</score_value>
            </score>
          </outcomes>
        </item_result>
      - <item_result asi_title="FOSS Math OV Pilot Item 1" ident_ref="1099">
        - <asi_metadata>
          - <asi_metadatafield>
              <field_name>OBSERVABLE_VARIABLE_ID</field_name>
              <field_value>1099</field_value>
            </asi_metadatafield>
          - <asi_metadatafield>
              <field_name>OBSERVABLE_VARIABLE_TITLE</field_name>
              <field_value>FOSS Math OV Pilot Item 1</field_value>
            </asi_metadatafield>
          </asi_metadata>
        - <outcomes>
          - <score varname="SCORE" vartype="Integer" staus="Valid">
              <score_value>1</score_value>
            </score>
          </outcomes>
        </item_result>
```

Figure A-5. Output student results file.



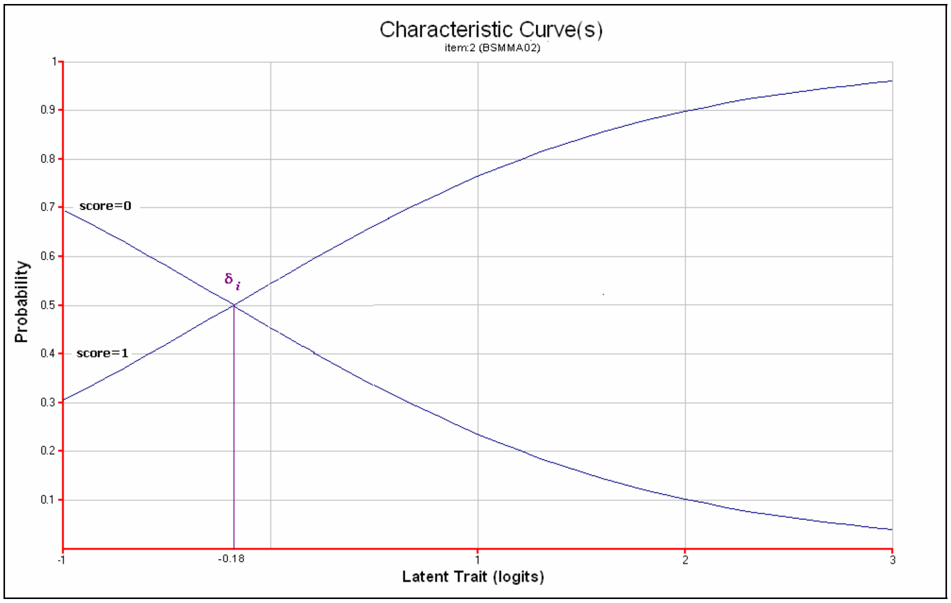
```
  <?xml version="1.0" encoding="UTF-8" ?>
- <scoring_engine_output>
  - <assessment_result>
    - <context>
        <name>FOSS pretest student 1</name>
      - <generic_identifier>
          <type_label>Student Number</type_label>
          <identifier_string>200411191</identifier_string>
        </generic_identifier>
      - <date>
          <datetime>2005-01-10</datetime>
        </date>
      </context>
    - <STUDENT_MODEL_TYPE NODE_TITLE="FOSS DSA + Math" NODE_TYPE_VERSION="3.20" ident="961">
        <NODE_ANNOTATION>Combines distance, speed, acceleration with math inquiry skills into a 2-dimensional student
          model.</NODE_ANNOTATION>
        <SM_DISTRIBUTION_TYPE PART_LABEL="Distribution Type" ATTRIBUTE_ID="3570" ATTRIBUTE_VALUE="25" VALUE_IN_PICKLIST="Multivariate
          normal" />
      - <COVAR_MATRIX PART_LABEL="Covariance Matrix" ATTRIBUTE_ID="3571" ATTRIBUTE_COMMENT="Calibrated as of 12/8/04. Conquest
          executed by Mike Timms using 26 cases.">
        - <COLUMN SM_VAR_ID="367" SM_VAR_NAME="FOSS Math Inquiry SMV">
            <ROW SM_VAR_ID="116" SM_VAR_NAME="Distance-Speed-Acceleration" COVAR_VALUE="0" />
            <ROW SM_VAR_ID="367" SM_VAR_NAME="FOSS Math Inquiry SMV" COVAR_VALUE="0.38221791081915985" />
          </COLUMN>
        - <COLUMN SM_VAR_ID="116" SM_VAR_NAME="Distance-Speed-Acceleration">
            <ROW SM_VAR_ID="116" SM_VAR_NAME="Distance-Speed-Acceleration" COVAR_VALUE="0.1294101350403738" />
            <ROW SM_VAR_ID="367" SM_VAR_NAME="FOSS Math Inquiry SMV" COVAR_VALUE="0" />
          </COLUMN>
        </COVAR_MATRIX>
      - <SM_DIST_MEANS PART_LABEL="Means Matrix" ATTRIBUTE_ID="3572" ATTRIBUTE_COMMENT="Calibrated as of 12/8/04. Conquest executed
          by Mike Timms using 26 cases.">
          <SM_DISTRIBUTION_MEAN SM_VAR_ID="367" SM_VAR_NAME="FOSS Math Inquiry SMV" VALUE="-0.8883949368738272" />
          <SM_DISTRIBUTION_MEAN SM_VAR_ID="116" SM_VAR_NAME="Distance-Speed-Acceleration" VALUE="0.6368127363680092" />
        </SM_DIST_MEANS>
      - <RELATED PART_LABEL="Student Model Variables" PART_TYPE="STUDENT_MODEL_VARIABLE_TYPE"
```

Appendix B—Background Information about Item Response Modeling (IRM)

B.1 Item Response Probability

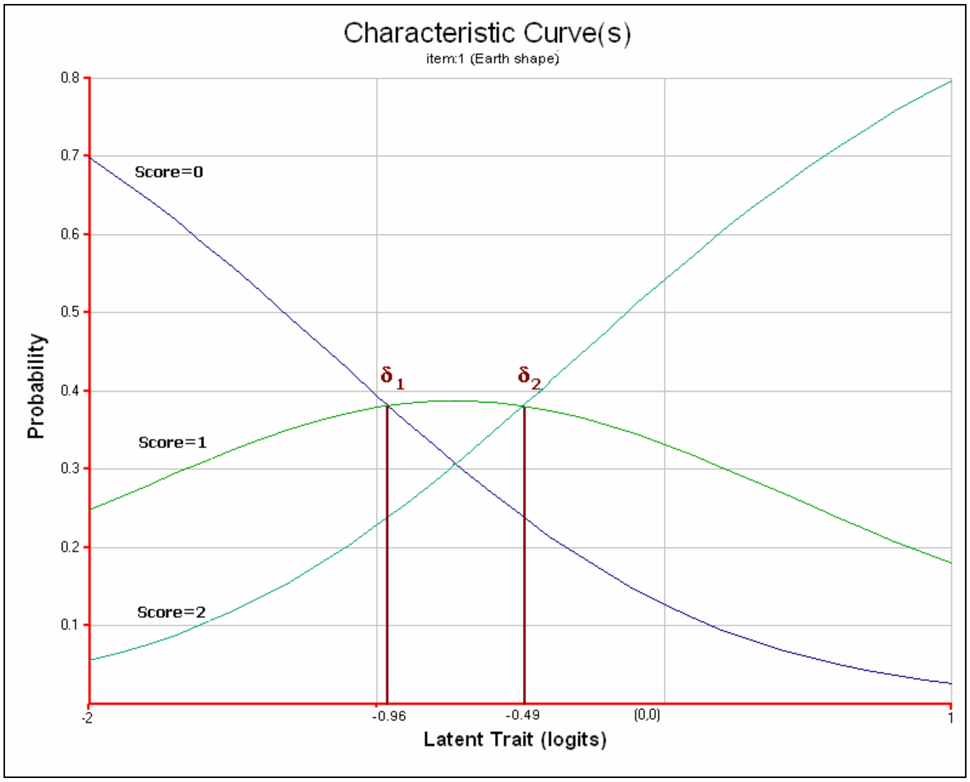
When an item response has only two possible values, correct or incorrect, the item difficulty is an expression of how much ability a person needs to give a correct answer. By convention, we describe the item difficulty as the ability level at which the student is equally likely to give a correct or incorrect response (that is, both probabilities are 0.5). In Figure B-1, for example, the item difficulty is -0.18; this is the point on the Latent Trait (i.e., ability) axis at which the probability curves for a correct and an incorrect response intersect.

Figure B-1. Item characteristic curves for a dichotomous (two-category) item.



When items have more than two possible outcomes, we need more information than the item difficulty. We need to know how much more of the latent trait is needed to achieve each possible score on the item. The partial credit case is an extension of the dichotomous case; moving from one category to another implies a dichotomous choice between two levels. For example, consider an item with three categories, scored 0, 1, or 2. The difficulty for step 1, denoted as δ_{i1}, is located at the point where, if one is considering just categories 0 and 1, one is equally likely to get the item partially correct (where x = 1) or incorrect (where x = 0). Note in Figure B-2 that this is where the curve for getting a score of 0 intersects with the curve for getting a score of 1. Subsequent steps in difficulty are interpreted in much the same way. The second step difficulty, δ_{i2}, is the ability required to have equal probabilities of getting a score of 2 or a score of 1 on the item. As shown in Figure B-2, each category has its own probability curve.

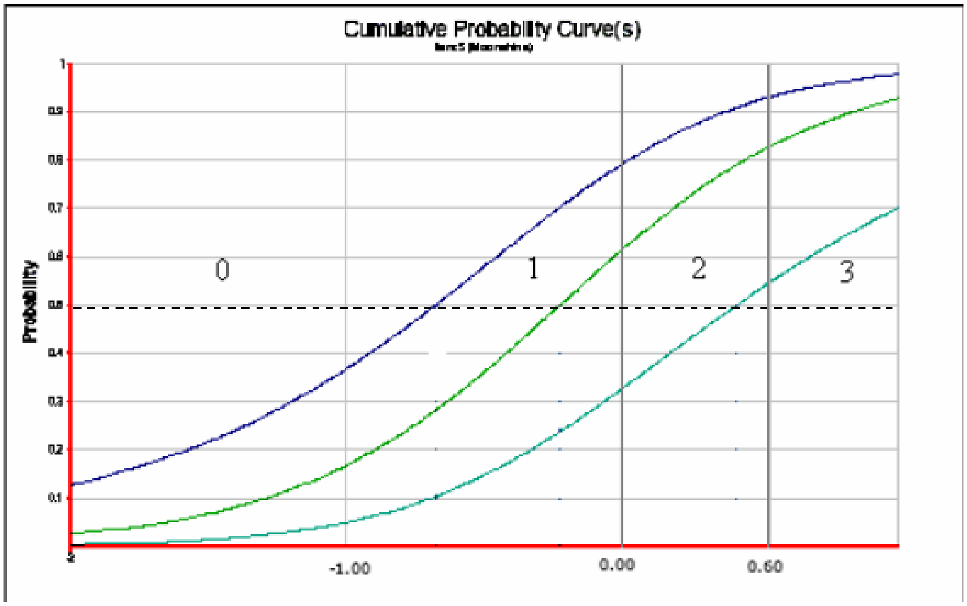
Figure B-2. Category probability curves and δ_{ij} values for a three-category polytomous item.



The location at which a person has a 50% probability of achieving a score in that category *or higher* is referred to as the Thurstonian threshold.[6] These locations can be identified on cumulative probability plots at the points where the curves intersect with the probability = 0.5 line, as shown in Figure B-3. These values tend to be more interpretable than δ_{ij} values because they identify ability levels where individuals are most likely to achieve specific scores. Figure B-3 shows an item for which a person with an ability located at 0.6 is more likely to achieve a score of 3 than a lower score, while a person with an ability located at 0 is more likely to achieve a score of 2 or 3 than a score of 1 or 0 (the curve for a score of 0 is not displayed in Figure B-3). This can be determined by examining the vertical lines at logit values of 0.00 and 0.60. For example, at a logit value of 0.60 the vertical line intersects the probability = 0.5 line in the area where the most probably score is 3. At a logit value of 0.00 the vertical line intersects the probability = 0.5 line in the area where the most probably score is 2 or higher.

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

Figure B-3. Cumulative probability curves and Thurstonian thresholds for a four-category polytomous item.



B.2 Measurement Models

To compute the probability of attaining a score of 1 rather than 0 on item *i,* given an item difficulty parameter of δ_i and a specific student proficiency of θ (in the unidimensional case), we use a Rasch formulation[7] in the form:

$$P(x_i = 1 \mid \theta, \delta_i) = \frac{P(x=1)}{P(x=0)+P(x=1)} = \frac{\exp(\theta - \delta_i)}{1+\exp(\theta - \delta_i)} \qquad (1)$$

For this dichotomous case, we have two probability equations:

$$P(x=0) = \frac{1}{1+\exp(\theta - \delta_1)} \text{; and}$$

$$P(x=1) = \frac{\exp(\theta - \delta_1)}{1+\exp(\theta - \delta_1)}.$$

For the polytomous case (i.e., the partial credit model[8]) the following equation shows the probability that a person with ability θ will respond in category *c* rather than in any other category on item *i*, given item difficulty parameters $\xi_i = (\delta_{i1}, \delta_{i2}, \ldots \delta_{im})$.

[7] Rasch, G. (1980). *Probabilistic models for some intelligence and attainment tests*. Chicago, IL: University of Chicago Press. (Original work published in 1960)

[8] Wright, B. D., & Masters, G. (1981). *The measurement of knowledge and attitude (Research Memorandum 30)*. Chicago, IL: University of Chicago, Department of Education, Statistical Laboratory.

$$P(x_i = c \mid \theta, \xi_i) = \frac{\exp \sum_{j=0}^{c} (\theta - \delta_{ij})}{\sum_{k=0}^{m} \exp \sum_{j=0}^{k} (\theta - \delta_{ij})}, \qquad (2)$$

where *m* is the number of steps (number of categories-1) for the item.

Thus, for a three-category item (with two steps),

$$P(x=0) = \frac{1}{\sum_{k=0}^{2} \exp \sum_{j=0}^{k} (\theta - \delta_{ij})};$$

$$P(x=1) = \frac{\exp \sum_{j=0}^{1} (\theta - \delta_{ij})}{\sum_{k=0}^{2} \exp \sum_{j=0}^{k} (\theta - \delta_{ij})} = \frac{\exp(\theta - \delta_{i1})}{\sum_{k=0}^{2} \exp \sum_{j=0}^{k} (\theta - \delta_{ij})}; \text{ and}$$

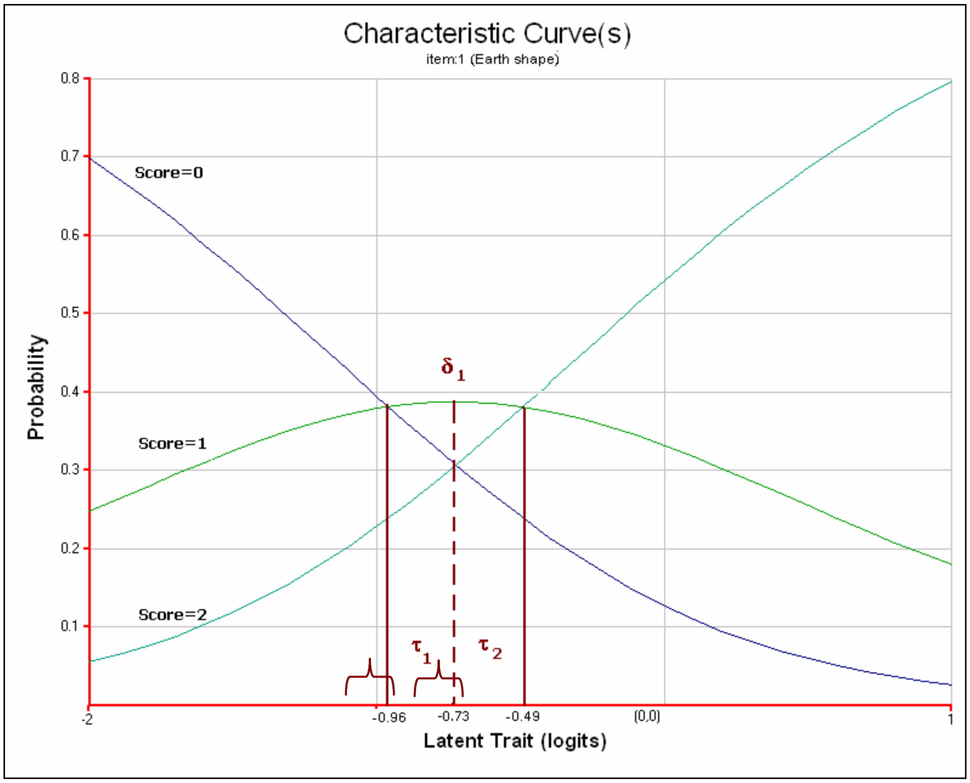
$$P(x=2) = \frac{\exp \sum_{j=0}^{2} (\theta - \delta_{ij})}{\sum_{k=0}^{2} \exp \sum_{j=0}^{k} (\theta - \delta_{ij})} = \frac{\exp(\theta - \delta_{i1} + \theta - \delta_{i2})}{\sum_{k=0}^{2} \exp \sum_{j=0}^{k} (\theta - \delta_{ij})} = \frac{\exp(2\theta - (\delta_{i1} + \delta_{i2}))}{\sum_{k=0}^{2} \exp \sum_{j=0}^{k} (\theta - \delta_{ij})}.$$

Note the conventions $\exp(0) \equiv 1$ and $\sum_{\text{j}=0}^{0} (\theta - \delta_{ij}) \equiv 0$; and that

$\sum_{k=0}^{m} \exp \sum_{j=0}^{k} (\theta - \delta_{ij})$ is the sum of the numerators for all categories: $1 + \exp(\theta - \delta_{i1}) + \exp(2\theta - (\delta_{i1} + \delta_{i2}))$.

When using the partial credit model, we generally parameterize the difficulty of achieving a score of *j* on item *i* and represent it with δ_{ij}. That is, δ_{ij} is the ability level required to expect an equal chance of responding in category *j* or in category *j*-1 on item *i*. Alternatively, we might think of the average of the δ_{ij} values as an overall item difficulty and the step difficulties as each step's deviation from the average. In looking at item difficulties in this way, we are saying that each δ_{ij} is a composite of $\delta_i + \tau_{ij}$, where τ_{ij} is the deviation from the average item difficulty for item *i* at step *j*. Note that in this case the last tau parameter (τ) is equal to the negative sum of the others so that the sum of all the tau parameters equals zero, $\tau_{im} = -\sum_{k=1}^{m-1} \tau_{ik}$. A graphical representation of this alternative formulation ($\delta_i + \tau_{ij}$) for an item with two steps (and therefore three categories) is shown in Figure B-4.

Figure B-4. δ_i**,** $\tau_{1,}$ **and** τ_2 **representations for the polytomous case with three categories.**



The rating scale model is a special case of the partial credit model in which the tau parameters for step *j* are the same for every item. That is, τ_{11}=τ_{21}=τ_{31}... and τ_{12}=τ_{22}=τ_{32}.... In this formulation, our measurement model becomes

$$P(x_i = c \mid \xi_i, \theta\ ) = \frac{\exp \sum_{j=0}^{c} [\theta - (\delta_i + \tau_j)]}{\sum_{k=0}^{m} \exp \sum_{j=0}^{k} [\theta - (\delta_i + \tau_j)]}, \quad (3)$$

where $\xi_I = (\delta_i, \tau_1, \tau_2, \ldots, \tau_{m-1})$. Again, the final step value, τ_m, is not estimated because it is constrained to make the sum of all the steps equal to zero.

The different parameterization techniques of the step difficulties for partial credit models and the item difficulties and tau parameters for rating scale models is an important distinction in representing the probability equations in PADI design system Measurement Models. If a rating scale model is to be used, all items that map to the same Student Model Variable must use the same tau parameters. These parameterization options are discussed in more detail in the *PADI Measurement Model Examples* section of this report.

The probability of a particular response pattern occurring is the continued product of the probabilities of the individual responses on an instrument when the items are conditionally independent. When the items are not conditionally independent, item bundles[9] can be constructed during the evaluation phases. The random coefficients multinomial logit (RCML) model[10] formulates the conditional probability of a response pattern, **x**, as

$$P(\mathbf{X}=\mathbf{x}\mid\theta)=\frac{\exp(\mathbf{x}'(\mathbf{b}\theta+\mathbf{A}\xi))}{\sum_{\mathbf{z}=\Omega}\exp(\mathbf{z}'(\mathbf{b}\theta+\mathbf{A}\xi))}, \quad (4)$$

where θ is person proficiency, **b** is the vector of response scores, **A** is the Design Matrix, ξ is the vector of item parameters with $\xi = (\delta_{11}, \delta_{12}, \ldots, \delta_{1m1}, \delta_{21}, \delta_{22}, \ldots, \delta_{lml})$, and **Ω** is the set of all possible response patterns.

[9] Wilson, M., & Adams R. J. (1995). Rasch models for item bundles. *Psychometrika*, *60*, 181-198.
Hoskens, M., & De Boeck, P. (1997). A parameteric model for local dependence among test items. *Psychological Methods, 2*, 261-277.

[10] Adams, R. J., & Wilson, M. R. (1996). Formulating the Rasch model as a mixed coefficients multinomial logit. In G. Engelhard & M. Wilson (Eds.), *Objective measurement: Theory into practice* (Vol. 3). Norwood, NJ: Ablex.

Sponsor

The National Science Foundation, Grant REC-0129331

Prime Grantee

SRI International. *Center for Technology in Learning*

Subgrantees

University of Maryland

University of California, Berkeley. *Berkeley Evaluation & Assessment Research (BEAR) Center* and *The Full Option Science System (FOSS)*

University of Michigan. *BioKIDs*