# A Four-Process Architecture for Assessment Delivery, with Connections to Assessment Design

Russell G. Almond[1], Linda S. Steinberg[1], and Robert J. Mislevy[2]

Educational Testing Service, Princeton, New Jersey

[1] Educational Testing Service
[2] University of Maryland

June 2002

(revised October 3, 2002)

**Abstract**

Persistent elements and relationships underlie the design and delivery of educational assessments, despite their widely varying purposes, contexts, and data types. One starting point for analyzing these relationships is the assessment as experienced by the examinee: 'What kinds of questions are on the test?' 'Can I do them in any order?' 'Which ones did I get wrong?' 'What's my score?' These questions, asked by people of all ages and backgrounds, reveal an awareness that an assessment generally entails the selection and presentation of tasks, the scoring of responses, and the accumulation of these response evaluations into some kind of summary score. A four-process architecture is presented for the delivery of assessments: Activity Selection, Presentation, Response Processing, and Summary Scoring. The roles and the interactions among these processes, and how they arise from an assessment design model, are discussed. The ideas are illustrated with hypothetical examples. The complementary modular structures of the delivery processes and the design framework are seen to encourage coherence among assessment purpose, design, and delivery, as well as to promote efficiency through the reuse of design objects and delivery processes.

Key words: Adaptive testing, assessment design, modularity, test theory

## Introduction

Pressing educational concerns have driven the increased use of assessment, the need for improved assessment, and the need for new kinds of assessment. Advances in technology, cognitive science, and psychometrics have been rising to meet this challenge. As with any field experiencing such growth, the impact of these advances in the field of educational assessment is creating a proliferation of new assessment practices, materials, and processes that attempt to address a variety of purposes and stages in the life of a learner. As the field moves beyond standard forms and approaches to assessment, novel requirements will appear—Internet delivery, for example, simulation-based problem solving, or more complex open-ended formative assessments to guide learning and instruction—while demands for efficiency and validity remain.

Although there are many different reasons or purposes for assessment, the current drive to develop a variety of new forms of assessment may potentially lead to a large number of instruments and methodologies that operate independently of each other and cannot be adapted easily to meet different purposes. One way to overcome this potential inefficiency is to create a a common framework or design architecture that enables the delivery of operational assessments that can be easily adapted to meet multiple purposes. This is a tall order. The requirements for a college entrance exam seem quite different from those of an assessment to support learning embedded in an intelligent tutoring system, or from a large-scale survey of an educational achievement system. In addition to accommodating a range of purposes, this architecture must be sufficiently flexible to support a range of formats among assessment delivery and authoring systems—from the standard multiple-choice and essay-type items, which form the core of current practice, through portfolios and classroom-based activities, to the advanced constructed-response items and interactive tasks we envisage as the future of assessment.

To this end, a common architecture for assessment systems is described here. This architecture defines the structures that underlie the operation of assessment systems, where the roles, interactions, and information used in any given assessment depend on the purpose and context of that assessment. This architecture also enables components of the assessment system, such as the delivery mechanism (e.g., paper, text on a computer screen, text accompanied by sound, etc.), the scoring process, decision rules (e.g., performance categorization), and feedback,

to be adjusted to meet specific purposes. The architecture system presented here is based on the principles for designing and developing assessments contained in the evidence-centered assessment design (ECD) framework. While it is beyond the scope of this article to describe the full ECD framework in detail, the ECD framework describes a process that begins by defining the decisions to be made based upon an assessment and then works backwards to develop tasks, delivery mechanisms, scoring procedures, and feedback mechanisms that provide evidence that informs the pre-defined purposes.[1]

The sections that follow describe how the same conceptual framework, defined at the right level of generality, can be used to guide the design and delivery of assessments that look very different on the surface, and span purposes that range from selection to instructional support. We use the term *assessment* broadly to emphasize the range of purposes we want to think about within this framework. We include, for example, high-stakes entrance exams, lower-stakes placement and diagnostic tests, tutoring systems, and even surveys, which are not scored. Each purpose for which a product will be used defines particular requirements for the security of the tasks, the reliability of the results, the nature and timing of feedback, and the level of detail of the reported claims. The architecture described here is also intended to guide the design of a co-operative system of assessments that can use the same material for different purposes. For example, tasks retired from a high-stakes exam could be used in a diagnostic exam, or a practice test, or a tutoring system. The different level of reporting details that are needed for these uses would require different scoring models. It will be shown how the full set of design specifications produced by the ECD process provides this flexibility by separating the presentation of the task from the scoring of the task and the decisions made based on these scores. This ability to separate scoring from presentation and decision-making allows us to reuse tasks in different contexts and to meet the requirements of different assessment purposes.

Section 1 begins by describing the four-process delivery system architecture. Section 2 provides a brief overview of the key elements of the ECD design upon which the architecture system is based. Section 3 introduces an example of a Chinese writing assessment that illustrates how the design and delivery of tasks interact in fulfilling different assessment purposes and requirements. The example addresses purposes that include a simple selection test and a more extensive diagnostic setting (as well as multiple-choice and constructed response formats) which

illustrate the advantages of the modular design elements defined by this common architecture. Sections 4 and 5 discuss in greater detail how these design objects govern the roles and the interactions of the processes. Note that this article introduces and employs several terms that may be unfamiliar to the reader. For this reason, a glossary is provided to help the reader become more familiar with the ECD terminology. Although many of the terms refer to objects or processes in familiar assessments, the more general language illuminates structural analogues in assessments that can look quite different on the surface—structures which, once recognized, can be exploited to increase both operational efficiency and conceptual clarity.

## 1. The Assessment Cycle

This section lays out the four basic processes that are present in an assessment system, broadly conceived. After introducing the processes (Section 1.1), it describes the central repository for information needed to present tasks and evaluate the data they produce (the Task/Evidence Composite Library - Section 1.2), and the messages the processes need to pass from one to another to carry out their responsibilities (Section 1.3).

### 1.1 The Four Processes

Any assessment system must have (at least in some trivial form) four different processes. The first important thing to note in this general description is that any assessment--whether it's a high stakes medical school entrance examination, a collaborative high school biology assessment, an interchange between a teacher and a student on the subject of philosophy, or a parent accompanying a newly licensed child out on an interstate -- is carried out using the same set of four processes: Activity Selection, Presentation, Response Processing, and Summary Scoring.

- The *Activity Selection Process* is the process responsible for selecting and sequencing tasks (or items) from the Task/Evidence Composite Library. These could be tasks with any kind of assessment or instructional focus, or activities related to test administration.
- The *Presentation Process* is responsible for presenting the task to the participant. As necessary, it will retrieve materials necessary to the task from the task library. In particular, certain kinds of presentation material such as images, audio, or applets

may be external resources brought in with the presentation of the item. When the participant performs the task, the Presentation Process will capture their response as one or more Work Products. These Work Products are delivered to Response Processing for evaluation.

- *Response Processing* performs the first step in the scoring process: It identifies and evaluates the essential features of the response (Work Products) that provide evidence about the participant's current knowledge, skills, and abilities. These evaluations are recorded as a series of Observations that are passed to the next process.
- The *Summary Scoring Process* performs the second, or summary, stage in the scoring process: It uses the Obsrevations to update the Scoring Record. The Scoring Record represents our beliefs about the participant's knowledge, skills, and abilities based on evidence accumulated across tasks. As we will show, separating the Response Processing step from both Summary Scoring and Presentation is vital to an evidence-based focus in assessment design and supports reuse of the task in multiple contexts.

The assessment cycle is produced by the interaction of these four processes and involves two actors: the administrator and the participant.

- The *Administrator* is the person responsible for setting up and maintaining the assessment. The Administrator is responsible for starting the process and configuring various choices; for example, whether or not item-level feedback will be displayed during the assessment.
- The *Participant* is the person whose skills are being assessed. The participant interacts with the various tasks that the Presentation Process puts forward.

Figure 1 shows the four processes, the actors in the system, and the interaction among them.

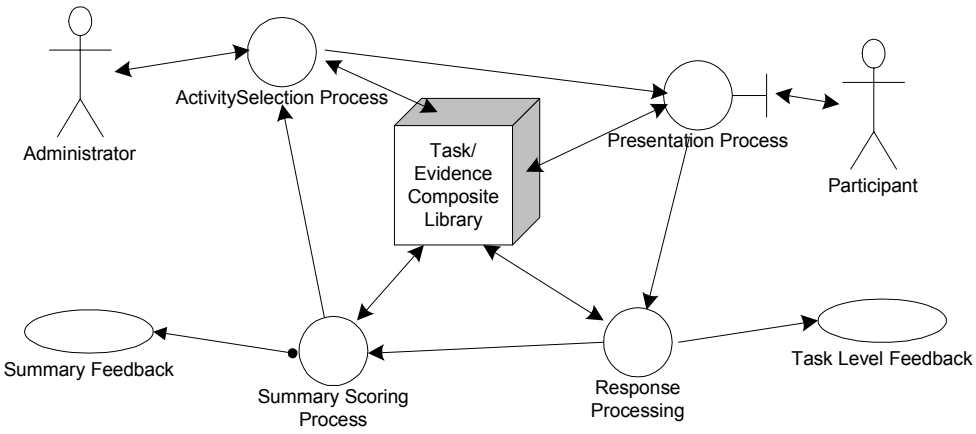


***Figure 1. The four principle processes in the assessment cycle***. The Activity Selection Process selects a task (tasks include items, sets of items, or other activities) and directs the Presentation Process to display it. When the participant has finished interacting with the item, the Presentation Process sends the results (a Work Product) to Response Processing. This process identifies essential Observations about the results and passes them to the Summary Scoring Process, which updates the Scoring Record, tracking our beliefs about the participant's knowledge. All four processes add information to the Results Database. The Activity Selection Process then makes a decision about what to do next, based on the current beliefs about the participant or other criteria.

The assessment cycle is neutral with respect to what knowledge, skill, or ability we are trying to assess (whether it is a body of facts or a set of complex cognitive or physical skills or abilities); and the claims we want to be able to make about examinees' proficiency. In addition, none of these processes assumes that it will happen using computers or humans, or whether it will run dynamically (real-time concurrent with examinees' engagement), or offline (at some other time). Finally, it is very convenient that we can talk about all kinds of assessment in terms of the same set of processes. However, the purpose of an assessment has a profound impact on what these processes actually turn out to be when implemented for a specific assessment. Therefore, the primary piece of information important to convey when we talk about assessment is: assessment for what purpose? In short, describing an assessment solely as "computer-delivered" or "multiple-choice" or "adaptive" or "multi-media" or "standardized" leaves us substantially uninformed about the true nature of the assessment.

This Four-Process Architecture, namely Activity Selection, Presentation, Response Processing, and Summary Scoring, can work in either a synchronous and or an asynchronous

mode. In the synchronous mode, the Activity Selection Process tells the Presentation Process to start a new task after Response Processing and Summary Scoring for the previous task are completed. In this case, the messages move around the system in cycles. In the asynchronous mode, once the Presentation Process is told to start a task or series of tasks, it generates a new Work Product whenever the participant finishes an appropriate stage of the task. Based on messages it may receive from any of the other processes, the Activity Selection Process decides whether to let the current activities continue, to send a message to the Presentation Process requesting a new activity, or to make inquiries of the Scoring Record for updated estimates of participant proficiency(ies) to use in its decision making.

Given the separation of Response Processing from Summary Scoring and flexible sequencing of the processes via messaging, this architecture facilitates the generation of two types of feedback: Task-level Feedback and Summary Feedback.

- *Task-Level Feedback* is an immediate response to the participant's actions in a particular task, *independent of evidence from other tasks.* As an example, Response Processing that performs diagnostic evaluation of participant work used in combination with related Activity Selection means that the system could immediately indicate the correct answer after the response was submitted, suggest an alternative approach, or explain the underlying principle of the task if misconceptions are evident. If desired, an additional Response Scoring Process can be used concurrently to evaluate the same response to produce observations that are accumulated by the Summary Scoring Process. Task-level feedback can be generated for real-time use during assessment or for reporting after assessment is complete.
- *Summary Feedback* reports the beliefs of an examinee based on evidence from multiple tasks, accumulated in the Scoring Record, concerning the participant's knowledge, skills, and abilities along the dimensions measured by the assessment. That is, it is feedback based on synthesized evidence from responses to any number of tasks. Summary feedback can be reported to the Administrator, the Participant, or other interested parties.

All four processes of the assessment cycle interact with the Task Evidence Composite Library.

As shown in Figure 1, the Task Evidence Composite Library forms the nucleus of the assessment cycle.

### 1.2 Task/Evidence Composite Library

The Task/Evidence Composite Library (Figure 2) is a database of task materials (or references to such materials) along with all the information necessary to select, present, and score the task. For each such task/evidence composite, the library stores information required by the four processes of the assessment cycle.

- Information required by the Activity Selection Process includes descriptive properties that are used to ensure content coverage, prevent overlap among tasks, or in some other way characterize tasks. This information is referred to as the Task Description Properties.
- Information required by the Presentation Process includes specific values of, or references to, task materials to be presented as well as other environmental variables that are used for presenting the task (e.g., font size, availability of tools, simulator settings). This information is referred to as Task Materials and Environment Variables.
- Information required by Response Processing includes specific data and algorithms (e.g., rubrics and solution data) that are used to extract and evaluate the salient characteristics of Work Products. This information is referred to as Evidence Rule Data.
- Information required by the Summary Scoring Process includes Weights of Evidence that are used in combination with observations from task responses to update a participant's Scoring Record— scoring weights, conditional probabilities, or parameters in a psychometric model. This information is referred to as Weight of Evidence Parameters.

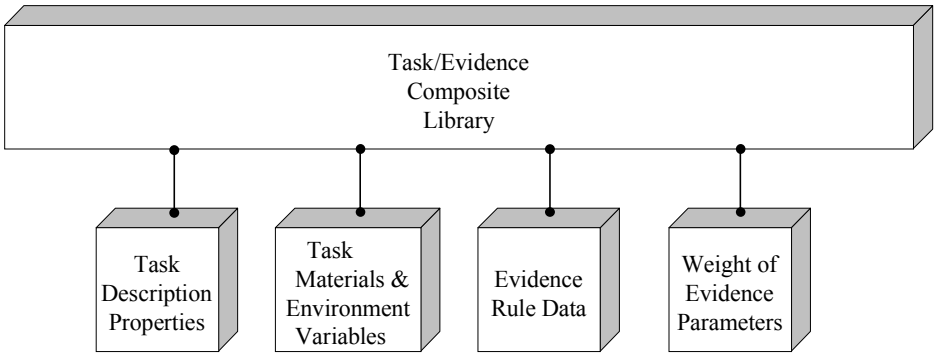


***Figure 2. The four kinds of information stored with each Task/Evidence Composite***. Task Description Properties are used by the Activity Selection Process; Task Materials & Environment Variables are used by the Presentation Process; Evidence Rule Data are used by Response Processing; Weight of Evidence Parameters are used by the Summary Scoring Process.

### 1.3 Communication Between the Processes

While the previous section described the four types of information that come from the Task/Evidence Composite Library needed by the four processes, this section elaborates on the data that flow around the assessment cycle between processes. Figure 3 builds on Figure 1 by including these data messages that flow around the assessment cycle.

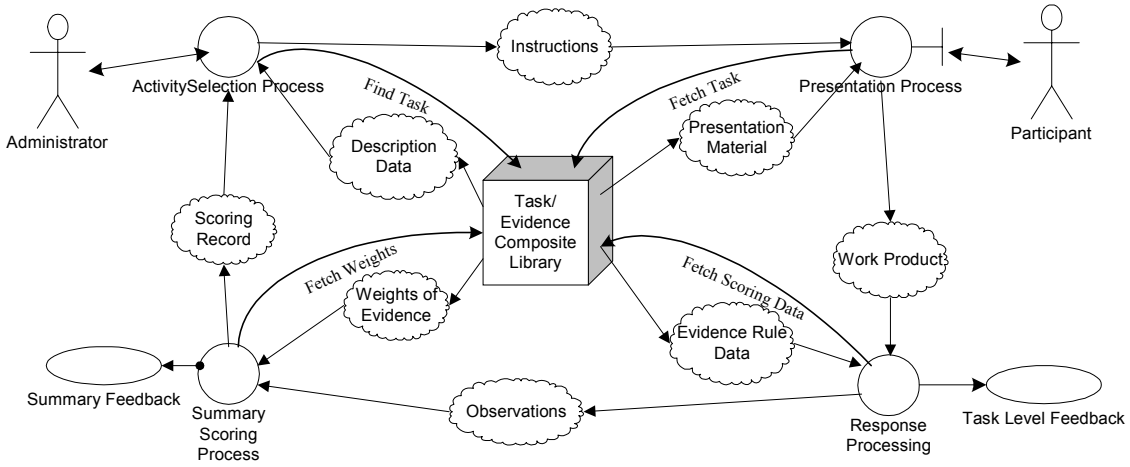


***Figure 3. Detailed view of the assessment cycle.*** In addition to the four processes, this figure includes the data objects taken from the Task/Evidence Composite Library and the flow of data around the assessment cycle.

The communications between processes include Instructions, Work Products,

Observations, and the Scoring Record.

- **Instructions** are commands sent by the Activity Selection Process to the Presentation Process. "Start Task *X*" is a common and important example. Other instructions include time-outs and administrative protocols. The most important part of the instructions is the identifier for the next task.
- **Work Products** are responses produced by the participant in the course of attempting to complete a task. They can be as simple as the selection made in a multiple-choice task, or as complex as a simulator activity trace or a collection of pieces of art work produced to meet the requirements of the Advanced Placement Studio Art Portfolio Assessment.
- **Observations** are variables that describe the quality of salient features of the Work Product. They may be as simple as the "correctness" of a response, or more complex as in the "artistry" of a performance. A Work Product may be evaluated for one or more observations depending on the purpose(s) of the assessment and nature of feedback for both Response Processing and Summary Scoring.
- The **Scoring Record** is the accumulated beliefs about the participants proficiencies across multiple tasks. These beliefs describe the current state of knowledge about the participant's knowledge, skills, and abilities.[1.]

All of this data taken together comprises the **Examinee Record**. This is a database that includes the tasks to which the participant has been exposed, the Work Products and Observables that were obtained, and statistics describing the final state of the Scoring Record.

## 2. Defining the Process Specifications for the Four Processes

As stated above, all assessment cycles contain four processes: Activity Selection, Presentation, Response Processing, and Summary Scoring. While a cycle is neutral to the purposes of assessment the specific components of each process are dependent on the purpose. The evidence-centered ECD framework provides a useful structure for informing the specifications for each process of the assessment cycle. As described in greater detail in section

4, the ECD Conceptual Assessment Framework (CAF) consists of six different types of models that specify the materials, capabilities, and other information that are needed by the four processes to deliver a particular assessment. Figure 4 shows the six models.

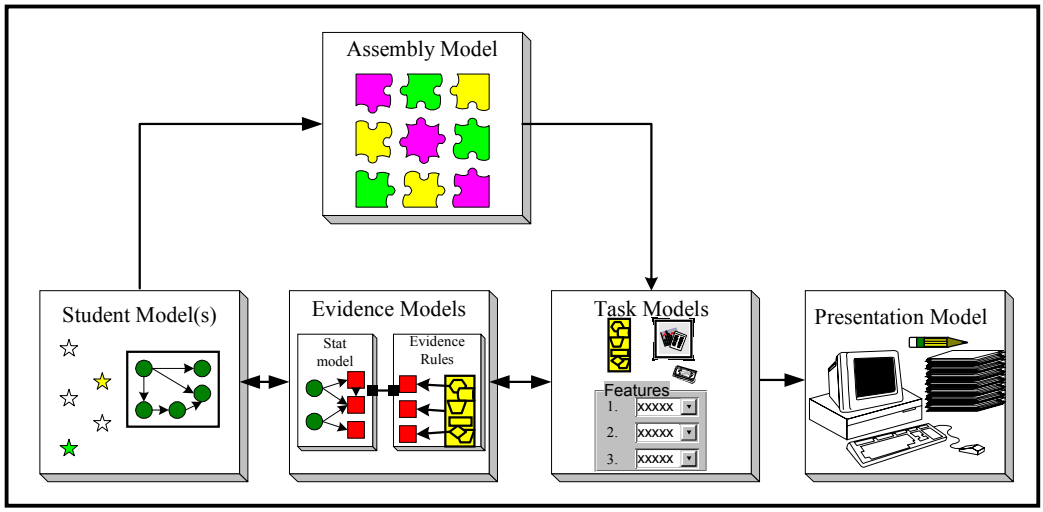


***Figure 4. The principle design objects of the ECD CAF***. These objects describe the requirements for the objects in the assessment delivery system.

The six models include the Student Model, the Task Model, the Evidence Model, the Assembly Model, the Presentation Model, and the Delivery Model.

- The **Student Model** represents the knowledge, skills, and abilities of a participant about which inferences will be made that lead to claims about the participant, and the attendant consequences--decisions about selection, placement, certification, instruction, task selection, and so on. The Student Model specifies the dependencies and statistical properties of relationships among these variables. These dependencies and statistical properties are dependent upon the types of claims about the participant we aim to make and the antecedent consequences. The Scoring Record describes our knowledge about the values of those variables for a specific participant at any given point in time.
- The **Task Model** is a generic description of a family of tasks. A Task Model contains (1) a list of variables that are used to describe key features of the tasks, such as their

content, difficulty, and conditions under which they are presented; (2) a collection of Presentation Material Specifications that describe the structure and format of material that will be presented to the participant as directions, stimulus, prompt, or instruction, and (3) a collection of Work Product Specifications that describe the structure and format of material that the task will return to Response Processing.

- **The Evidence Model** is a set of instructions for interpreting the response (Work Product) to a specific task. The Evidence Model contains two parts. The first is a series of Evidence Rules that describe how to identify and evaluate essential features of the Work Product. The second is a statistical model that tells how the Scoring Record should be updated given the observed features of the response.
- The **Assembly Model** is a set of instructions for assembling the assessment.
- The **Presentation Model** describes how a particular task is to be presented (or rendered) in a particular delivery environment. For example, tasks administered by computer and on paper would have different Presentation Models, even if their stimulus content, response format, and evidence rules were identical.
- The **Delivery Model** is a catchall for things that affect the entire assessment. It is a container for the other models and also contains information about administrative constraints, which do not fit elsewhere (e.g., security, recovery, etc.).

A typical assessment product would generally employ a single Student Model (related to the purpose) and a single Assembly Model (to control the selection of tasks). It could, however, have a number of Task Models and corresponding Evidence Models and Presentation Models to support them. The separation of design specifications into the various models is an important part of the flexibility of the Four-Process Architecture and allows the reuse of various assessment and assessment delivery elements.

## 3. Examples of the Use of the Four-Process Architecture for Two Different Purposes

Having described the four-process architecture for assessment delivery and its specific components, in this section we describe how the four-process architecture can be applied to develop assessments that meet two different types of purposes. Specifically, we focus on a high-

stakes selection test and a drill-and-practice tutoring system for Chinese character reading and writing. These examples, while relatively easy to describe even to people with no experience with East Asian languages, are singularly useful in helping us address a number of difficult design issues, including the impact of different purposes on assessment design and delivery as well as dealing with non-traditional types of data, including audio and pictures.

### 3.1 A High-Stakes Assessment

We will look first at an assessment system design for high-stakes selection testing (Figure 5).

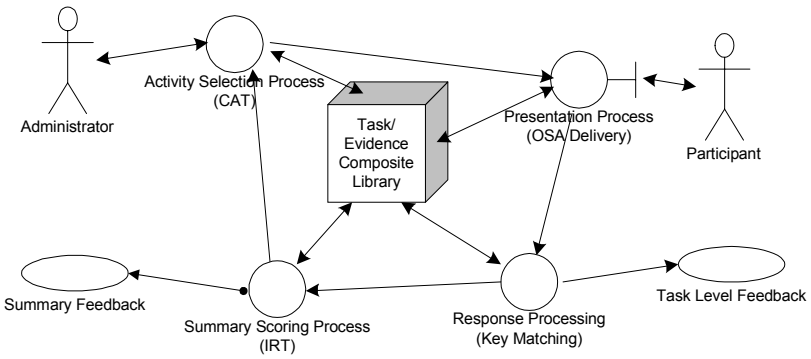


***Figure 5. Assessment cycle objects specialized to a high-stakes selection type assessment***.

All elements of an assessment's design flow from the purpose of the assessment -- in particular, the nature of the claims made as a result of a participant's engagement with the assessment, as well as the timing, nature, and granularity of feedback. For our high-stakes selection example, a single score (coupled with normative information) delivered at the end of the assessment will suffice. Because no task-specific diagnostic feedback is required, responses can be evaluated as either correct or incorrect. Task performance will not be scaffolded (i.e., supported with help), and all forms of the test span comparable content and difficulty. Our high-stakes selection example also contains typical operational constraints: it must be delivered to a large population of test-takers, with only a limited amount of time, covering a potentially large domain, and it must be scored inexpensively and quickly.

Working through the design process, we identify the salient claims, evidence, and tasks

for our purpose and blend these requirements with the constraints described above. The result is a set of models that represents the specifications for this assessment.

- What we want to measure is represented by the *Student Model for Overall Proficiency*. In this Student Model we have a single (continuous) Student Model Variable, which indicates the participant's overall level of mastery. (This can be supported by familiar unidimensional IRT-based statistical processes.) Task Model Variables can be used to help predict the parameters of each item. This model, the Student Model for Overall Proficiency, is used to accumulate information across tasks and is not capable of providing detailed task-level diagnostic feedback.
- Evidence to support inferences related to mastery is evaluated by the *Evidence Model for Correct/Incorrect Key Matching*. In this Evidence Model, a simple algorithm matches a selected response containing the desired evidence against a key to produce a Boolean value (representing 'Correct' or 'Incorrect'). Information from a single observable is used to update the student model variable.
- Two Task Models are employed: the *Phonetic Transcription Task Model* and the *Character Identification Task Model.* The *Phonetic Transcription Task Model* presents a picture of one or more characters and requested the participant to type a phonetic transcription. The resulting Work Product from this task is a string of characters that can be matched to a key. The *Character Identification Task Model* presents a speech clip to the participant giving both the character and an example of usage of the character. The participant is asked to select the correct character from a list of candidates. The Work Product is a logical identifier indicating the selection the participant made.

We cycle through the four assessment processes in the following manner:

1. We start with the Activity Selection Process. After taking care of the administrative requirements, its job is to select the *next task* (or item) from the Task/Evidence Composite Library. In doing this, it may examine the values of certain Task Model Variables, to ensure breadth of content or prevent task overlap (Almond & Mislevy, 1999). In an adaptive assessment, it also consults the current state of the Scoring Record (i.e., our current estimate of overall proficiency) to select a task that is

particularly informative in light of what we know about the participant's preceding responses (Berger & Veerkamp, 1996).

2. When the Activity Selection Process has selected a task, it sends an instruction to the Presentation Process. The Presentation Process uses the Task Model to determine what Presentation Material is expected for this task and what Work Products will be produced (in this case either a tag identifying the choice or a string giving the short response). It might also consult with Task Model Variables to set options for the presentation of the task (e.g., use of different screen resolutions).
3. The participant interacts with the Presentation Process to produce some kind of response, which in this case is just the choice or character string. This is stored in a Work Product, which is sent to Response Processing to start the scoring process.
4. Response Processing looks at the Evidence Rule Data to ascertain the "key," or correct answer, for this item. It then checks the Work Product against this data using the Evidence Rules to set the Observables to appropriate values. For operation with the Student Model for Overall Mastery, only the Observable "Correct" (with Boolean value) is relevant.
5. The Summary Scoring Process takes the Observable and uses it to update the Scoring Record. For the Overall Proficiency schema, the Student Model contains only a single variable, the IRT proficiency parameter $\theta$. The Weights of Evidence in this case are the IRT item parameters; for example, difficulty, discrimination, and guessing under the three-parameter logistic model. Summary Scoring is accomplished through successive products of the likelihood functions induced by each item response; from a Bayesian perspective, successive updating of the probability distribution that reflects current belief about the participant's $\theta$.
6. The Activity Selection Process can now select the next task, or decide to stop. In making this decision, it can use the updated distribution for $\theta$, either to select an item likely to be particularly informative about the participant based on what is known thus far, or to terminate testing because a predetermined level of accuracy has been achieved.

For this testing purpose, we can use mostly off-the-shelf components. The Activity

Selection Process can be an adaptive item selection algorithm or a linear one. The Summary Scoring Process is the standard IRT scoring process. The Presentation Process could be a standard computer-based client interface with a few customizations (e.g., support for Chinese fonts). One big difference from most current assessment delivery architectures is that we have separated the first scoring step (Response Processing) from the presentation of the task (Steps 3 and 4). This may not seem consequential because the example we have used in Step 4 is so simple: just comparing a tag or string. However, doing so gives us flexibility for using the tasks for other purposes.

Separating the stages has some important implications for modularity. None of these processes needs to be computer-based; some or all could be manual processes. The four processes can be implemented to best meet the needs of a particular assessment; thus we could exchange a pronunciation scoring process based on human raters with one based on computer speech recognition. Alternatively, we could exchange an English language-based presentation process with one in which directions were localized for a different region. Distinguishing the separate pieces conceptually maximizes the potential for re-use even if we ultimately decide to implement them in the same (human or computer) manner.

### 3.2 A Drill-and-Practice Tutoring System

To illustrate how components can be reused, we will look at a delivery system specialized for a drill-and-practice tutoring system (Figure 6).

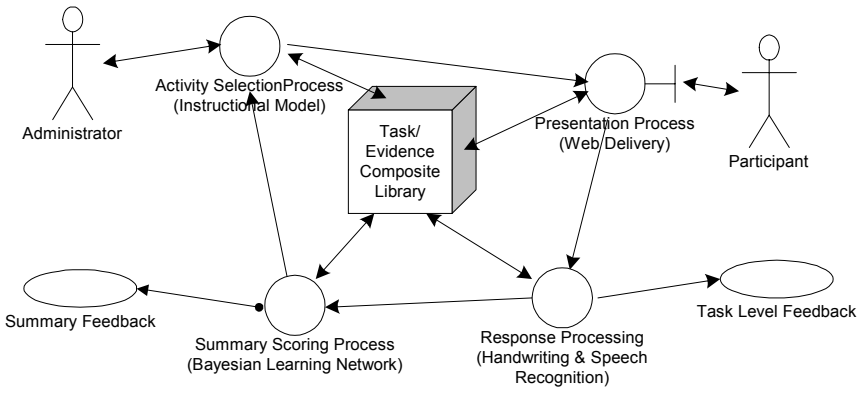


***Figure 6. The assessment cycle specialized for a tutoring system***. The new processes enable diagnostic task-based feedback and accumulation of evidence about multiple skills to support more targeted and informative summary feedback.

The design for our drill-and-practice example induces different requirements than our high-stakes example. In this instance, we need to be able to deliver across-task feedback on multiple aspects of proficiency, as well as task-specific diagnostic feedback. Therefore, responses will be scored as either correct or incorrect and evaluated diagnostically. The quality and timing of this feedback is of central importance. To support learning directly, various kinds of help can be provided as part of the task performance environment and the participant can benefit from being able to choose which tasks to do in what sequence. While the collection of tasks available to a participant needs to reflect a full range of content and difficulty, comparability among participants with respect to tasks across participants is not essential.

Again, working through the design process, we identify the salient claims, evidence, and tasks for the given purpose and blend these requirements with the constraints described above. The result is a set of models that represents the specifications for this assessment.

- What we want to measure is represented by a *Student Model for Diagnosis.* This model is based on some defined set of common speaking and listening skills (e.g., discrimination among tones, recognition of initial and terminal phonetic units, stroke order, recognition of common radical and phonetic components of a character), each which is represented by a Student Model Variable. Evidence about each of these skills across tasks can be accumulated and feedback on specific problems the participant exhibits can be reported. We can also use this information to assign more tasks that draw on the knowledge and skills with which the participant seems to be having the most trouble. This kind of feedback could be delivered as the participant works through the assessment, or it could be delivered at the end of the assessment.

  As an alternative, we could use a *Student Model for Lesson Groups*. This model is based on groupings of the characters into vocabulary sets. These groupings may be based on the lessons of a particular textbook, or may correspond to frequency of use. We assign one Student Model Variable for each vocabulary set. We construct each set with four possible values: mastered reading and writing; mastered reading, but not writing; mastered writing, but not reading; and mastered neither reading nor writing. Under this schema, we would want to provide task-based diagnostic feedback to

augment the summary scores.

- *Task Models* provide tasks designed to fulfill evidentiary requirements for diagnosis. Four task models could be utilized for this example, the *Phonetic Transcription Task Model* and the *Character Identification Task Model*, the *Reading Task Model*, and the *Writing Task Model.* The *Phonetic Transcription Task Model*, can be used as described earlier in high-stakes testing example. The *Character Identification Task Model*, can be reused if modified to include specifications of possible responses reflecting the variety of error patterns of interest. For the *Reading Task Model*, the *Phonetic Transcription Task Model* can be modified to request the participant to pronounce the character(s) aloud. Thus, the Work Product becomes a speech sample. For the Writing Task*,* the *Character Identification Task Model* can be modified to request the participant to draw the character. Thus, the Work Product becomes a picture of the character.

  For this drill-and-practice example, we could use support-related variables in our task models to author tasks that give the participant a prompt or help in the form of a phonetic pronunciation guide for the character, or allow the participant to request such help.
- *Evidence Models* appropriate to these student and task models require evaluation of Work Products for identification of specific types of problems as well as for correct/incorrect. For the former, specifications for answer keys (Evidence Rule Data) reflect requirements for diagnosis. Each Observable will update either a Lesson Group Student Model Variable or a Diagnosis Student Model Variable.

We cycle through the four assessment processes in the following manner:

1. We again begin with the Activity Selection Process. After administrative startup (including possibly loading a previously saved version of the Examinee Record), a task is selected based on the current state of the Scoring Record. Using the Student Model for Lesson Groups, for example, a task is selected from the first group of tasks not yet mastered.
2. The Activity Selection Process sends an instruction to the Presentation Process to start a particular task.

3. The Presentation Process fetches the presentation material from the Task/Evidence Composite Library. It presents the material to the participant, either by showing a picture or playing a sound. When the participant responds, the Presentation Process bundles the response into a Work Product and sends it to Response Processing. For the four kinds of tasks, the Work Products will consist of sound clips, pictures, character strings, and logical identifiers.
4. The Response Processing for the *Reading* and *Writing* tasks requires either human raters or speech and handwriting recognition software. There is more required of Response Processing for the Student Model for Diagnosis than for the Student Model for Overall Proficiency. A single observable with values "right" and "wrong" is no longer sufficient. If the participant is wrong, we want to know what kind of mistake was made: tone confusion, phoneme confusion, mistaking one character with a common radical for another, and so on. Response Processing for *Phonetic Transcription* and *Character Identification* tasks can continue to use key matching algorithms, but these algorithms must set Observables to values representing different diagnostic outcomes. In our Student Model for Lesson Group, tasks must be scored both as *Correct/Incorrect* and for diagnosis.
5. The Summary Scoring Process is more sophisticated as well. Not only must it determine *how much* beliefs about the participant's abilities should change as a result of our observations, but it must also indicate *which* variables in the Scoring Record are affected. With the Student Model for Lesson Groups, this is straightforward: Each task belongs to a Lesson Group, and we assume limited interaction among the groups. However, for the Student Model for Diagnosis, the presence of the knowledge, skills, and abilities we are trying to measure is often highly correlated (as is our knowledge about them). Therefore, an approach based on multivariate graphical models, a generalization of more familiar psychometric models, is used for this step (Almond & Mislevy, 1999; Mislevy, 1994; Mislevy & Gitomer, 1996). Finally, the Observables produced by Response Scoring for diagnostic task-based feedback in our Lesson Group version will not be accumulated by Summary Scoring, but instead sent to Activity Selection.

6. Finally, the Activity Selection Process chooses the next activity. Using a selection algorithm in combination with the Student Model for Lesson Groups, this decision is based on how many lessons we believe the participant has mastered, as well as whether speaking, reading, or both have been mastered. Selection based on the Student Model for Diagnosis would choose tasks focusing on identified trouble areas, and would have rules for how and when to shift focus based on the evolving state of the Scoring Record.

Although straightforward, this example raises numerous issues:

- *Multimedia.* We need to allow for both audio and pictures as both input and output of tasks. We must choose from among an array of potentially useful formats and fonts. Our Task Model must make it clear to the Activity Selection Process, the Presentation Process, and Response Processing what is expected.
- *Representational Issues.* We must choose how to represent a character drawing. We could use a static picture (e.g., a bitmap or compressed bitmap format) or describe the character by the series of strokes used to draw it. The former is easier to work with, but the latter is more closely aligned with the principles of Chinese calligraphy. This presents a trade-off between convenience and the quality of the evidence about certain aspects of writing.
- *Input Method.* There are several possibilities for inputting characters. These including drawing with a mouse, using a graphics tablet or light pen, or drawing with brush on paper and scanning the result into the computer.
- *Response Scoring Method.* Depending on the optimal granularity of the feedback, we may choose to explore character recognition programs that require the stroke-based representation of the character.
- *Localization.* For use in China, we may want the instructions to be in Chinese. For use in another country, we may want the instructions to be in the local language.
- *Reusability.* Although we limit this example to teaching reading and writing directly in Chinese, it is easy see how we could extend this tutoring system to include translation tasks. In addition, tasks of this sort could be embedded in a high-stakes

exam that offers placement out of a college course. Standards for interoperability would allow vendors of such placement exams to purchase and easily incorporate a special purpose Presentation Process for these tasks from a software company whose primary business was making software to support Chinese, Japanese, and Korean languages.

This example stretches the limits of the standard assessment design model, but it is not far-fetched. Many Chinese and Japanese computer-assisted instruction systems already incorporate at least some of this functionality. For example, the Wenlin program has a "flashcard" mode that uses a variation of the Writing Task. The PhonePass system for evaluating English language speaking skills is an examination that is similar to the Reading Task. Our example moves thinking beyond conventional multiple-choice type items, and toward extended constructed response tasks for which computer presentation provides a clear advantage over paper and pencil administration in terms of both multimedia and automatic scoring.

In the ECD assessment framework as it applies to both of our examples, assessments meant to fulfill different purposes are not expressed using different design objects, but rather by linking different instances of the same collection of generic objects. There is no such thing as an "Instructional Task Model." A Task Model is blind to purpose and presentation: It participates in fulfilling a specific purpose only when it is linked to a specific Evidence Model, as in the examples above. This means that a Task Model, and the tasks created from it, can be reused for multiple purposes and in multiple environments (within the constraints of its inputs, namely presentation materials, and its outputs, namely responses). The larger implication is that an assessment can be constructed from a series of smaller generic objects that are blind to purpose. The intended purpose of a product, whether selection or instruction, is fulfilled by linking models and processes in a way to meet the specified purpose.

Suppose we want to re-use some of the tasks from our Chinese tutoring system as part of a Chinese language placement examination. The new purpose would require a new Student Model—one with fewer variables, which can be measured more reliably. We need new Evidence Models in order to use these tasks with the new Student Model. However, as long as the Task Model is compatible with both Evidence Models, we do not need to redesign or re-author the tasks (except for perhaps adding some additional Evidence Rule Data). Similarly, switching between paper and computer administration is straightforward. We simply switch Presentation

Models and adjust the Delivery Model and Assembly Model. As long as all the tasks are compatible with both Presentation Models, we have no further difficulties. If some classes of tasks are inherently incompatible with one mode of delivery or the other, though, we may need to create new Task Models and adjust the Assembly Model to compensate. The Assembly Model is further restricted as adaptive testing is not compatible with paper-based tests.

## 4. How CAF Models Inform Different Aspects of Assessment Delivery Design

We can now look more closely at each of the models and illustrate how together they specify the delivery system design. (While these descriptions include models interactions with each of the four delivery system processes, a detailed summary of the characteristics of the four processes is provided in Appendix A.) We will address the specific functions of each of the CAF models in the following order: Task Models, Evidence Models, Student Models, Assembly Models, Presentation Models, and Delivery Models.

### 4.1 Task Models

A Task Model (Figure 7) is a formal description of a family of related tasks It is important to distinguish between *Task Models* and *tasks*. The *Task Model* is a set of specifications; the *task* is a specific instance of the kind of object that the Task Model describes. For example, the Task Model describes what kinds (and sizes) of objects to expect as input, while a particular task gives actual values or resource locators for particular instances of input that fit those descriptions. For instance, any of the Task Models in our examples would include variables describing properties of characters to be presented (e.g., possibilities for "number of strokes" may range from very few to very many, or possibilities for "idea represented" may range from abstract to concrete). Test developers use variables such as these to guide decision-making during the authoring of any particular task. The reader is referred to Mislevy, Steinberg, and Almond (2002) for more detailed discussion of the various roles that Task Models play in assessment design.

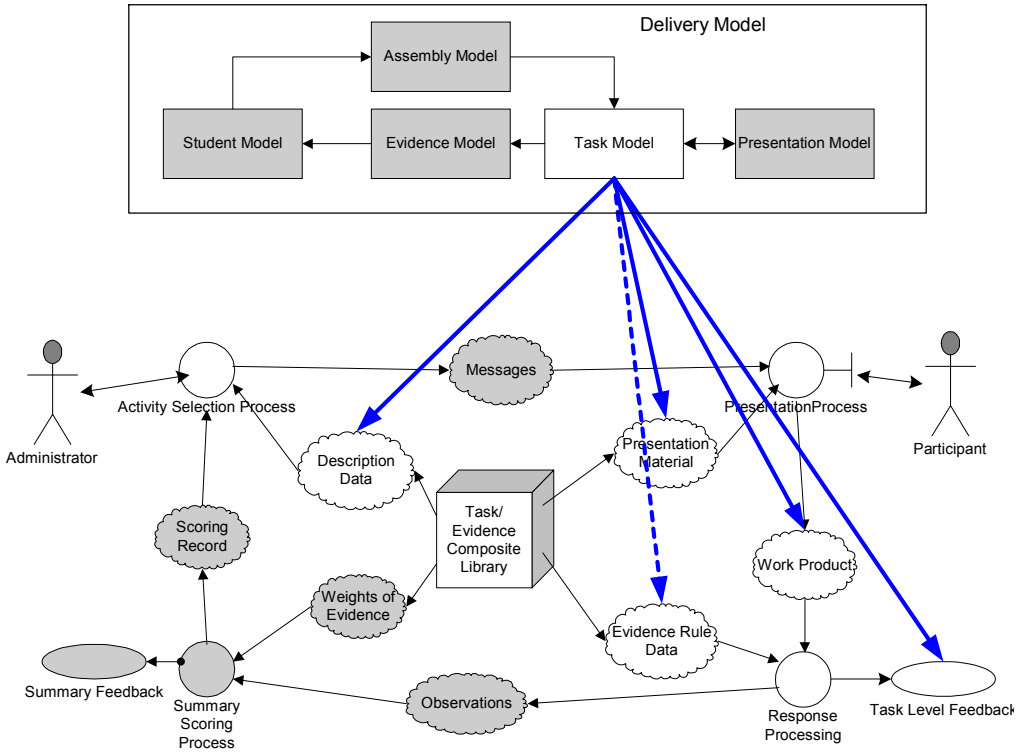


***Figure 7. The Task Model.*** The Task Model describes the Presentation Material, the Work Product, and the Description Data available for task selection within the Activity Selection Process. It also influences the contents of Evidence Rule Data.

The Task Model views the Presentation Process as a generalized interface for presenting material specified in any given Task Model. Similarly, as far as the Response Processing is concerned, a task produces a Work Product consisting of a collection of arbitrary types of objects. The Task Model also specifies the types and lengths (possibly variable) of these collections. This means that the protocols for communication between the Task/Evidence Composite Library and the Presentation Process are quite flexible: The properties of the content for a given class of tasks are specified in the Task Model that describes those tasks. The Task Model also specifies what the Presentation Process has to do to deliver and manage tasks of this class. This means that for a given Presentation Process to present and manage a class of tasks, it must meet the requirements specified in the Task Model. Defining these specifications during the design phase ensures that the people responsible for task creation and the people responsible for task delivery will share an understanding of what is possible, what is required, and how it must

be communicated.

Specifically, the Task Model consists of three pieces: a description of the "presentation material" to be presented to the user, a description of the Work Products that will be returned as a result of user interaction with the task, and a collection of Task Model Variables that describe properties of the content of presentation material and Work Products needed for authoring of specific tasks as well as those that can be used by the Task Selection and Presentation Processes.

We now examine in more detail the four specific task models in the context of our running example followed by a more detailed description of the three pieces of the task model: the presentation material, the Work Product, and the Task Model Variables. In particular, for this example, we need to support four Task Models: the Reading Task Model, the Phonetic Transcription Task Model, the Writing Task Model, and the Character Identification Task Model.

- We start with the Reading Task Model. In this kind of task, our presentation material is a representation of the character. This might be a bitmap picture, or a Unicode character ID, or a drawing on a flashcard. The Work Product will be an attempted pronunciation of the character. This could be a variety of sound file formats, or even a physical tape recording of the spoken response. The Task Model needs to be specific about which formats will be supported for both kinds of material. The Task Model may also indicate that we will store other types of supplementary material with tasks of the class. For example, we may want a frequency index for the character so we can tell whether it is common or rare. We may want a list of morphemes used in the character, or phonemes used in the pronunciation; these would be helpful for using the task in a diagnostic context. We might also want to have the pronunciation in applicable phonetic system as a supplementary piece of presentation material. This could be helpful if the task was used in an instructional mode, as well as to automatically identify homophones. Specific variables that provide information about the content and difficulty of tasks that can be generated from our Task Model would also be included.
- The Phonetic Transcription Task Model requires only a simple change from the Reading Task Model. For this kind of task instead of a speech sample, the expected

response would be a short string of characters that give the phonetic pronunciation. Employing a standard notation that uses numbers to indicate tones, a standard Roman alphabet keyboard could be used as the input device with minimal training for the participant. The presentation material and descriptors are similar to that of the Reading Task. The Work Product consists of a standard ASCII string, which can be readily matched to the standard pronunciation.

- In the Writing Task Model, the presentation material will consist of one of the pronunciations of the character, followed by an example of its usage in a common word or phrase; for example, "rén as used in rénlèi" (man as used in mankind). The response will be the written representation of the character. Here we have a critical choice to make. For the Work Product we could choose either a static picture of the final character or a stroke order representation of how it was drawn. We will discuss this choice in some detail below. Again, we need to be explicit about the supported representations for that material. As with each task model, specific variables that provide information about the content and difficulty of the tasks that can be generated from this Task Model will be included.
- For the Character Identification Task Model, like the Writing Task Model, the presentation material is a sound clip. The Work Product in this case is an indicator of whether the key or one of the distracters was chosen. However, we need be careful here and look ahead to consider the Evidence Models to ensure that the Task Models are designed to meet the needs of the Evidence Models. The Evidence Models for using this task with the Overall Proficiency and Lesson Group Student Models are straightforward. But for use with the Diagnostic Student Model, information about the key and distracter needs to be provided. The distracter a participant chooses will provide evidence about the various skills, such as tone discrimination, initial sound discrimination, and identifying radical and phonetic components of characters. Therefore, the distracters will need to be generated in such a way as to enable this feedback, and encoded as values of Task Model Variables.

### *4.1.1 Presentation Material*

As mentioned earlier, the task model consists of a description of the "presentation material" to be presented to the user. The Task Model must explain what is coming in sufficient detail so that the Presentation Process knows what to expect. Therefore, the Task Model must contain variables specifying options for how tasks might be presented or scaffolded. These might be specified values of the task object, or the task object might instead contain resource locators (instructions for how to fetch the required resources).

Let us first consider potential Reading Task Model requirements for the Presentation Process. Here we have two pieces of Presentation Material: (1) the picture of the character and (2) its phonetic rendering. For the first piece we need to specify two things: the size of the image and the supported formats for the image. One possibility is to specify a particular existing image format, or possibly to allow for several formats and thus force a conforming Presentation Process to support all of them. We similarly have several choices for transmitting the size of the picture. It is easy to fix the picture size and require task authors to make pictures of this size. An alternative would be to make the picture size a Task Model Variable, which the Presentation Process would read and adapt its layout to on the fly. A different way to specify the character picture is to simply give a reference to a position in a font; for example, the Unicode character ID of the character. This would shift the responsibility for rendering the character from the task author to the Presentation Process. The decision of bitmap vs. character/font format can hinge on the particular purpose of the intended assessment. The bitmaps, for example, may be necessary to test recognition of variant or calligraphic forms of characters.

The second piece of Presentation Material is the phonetic transcription, which may or may not be presented to the participant. For example, the Presentation Process could use it as a prompt, but only if the participant requests it as could be the case in our tutoring example. A phonetic transcription system may use Roman letters to represent the Chinese phonemes. The four tones can be represented in two ways: with accent marks or with a standard numbering system for tones. Therefore, we have two choices for representing phonetic information: (1) using ASCII letter and a number to indicate the tone, or (2) using an extended character set, which includes both letters and accents. The purpose of the assessment and the background of the intended users should influence the decision.

A task from the Writing Task Model similarly has two pieces of Presentation Material. The first is a speech sample; the second is again the phonetic transcription (to be used as an optional "help"). We would define the speech sample as having two parts: (1) the pronunciation of the character and (2) an example of usage. Again, we have a variety of sound formats from which to choose, and which we must specify in the Task Model.

Design trade-offs arise from the overwhelming number of sound formats that are currently available. If we loosen the Presentation Material specifications to include more supported formats, we reduce the work at authoring time in return for increasing the work the Presentation Process must do, either in supporting the required formats directly or translating them. We would be guided by those responsible for implementing the task authoring and Presentation processes. Since we do not need particularly high quality sound for this task, it seems reasonable to restrict our Presentation Material to one of a number of common formats that are supported by most multimedia-capable PCs.

### *4.1.2 Work Products*

The Task Model also specifies the Work Products that participants who interact with the tasks will produce. As with the Presentation Material, the Task Model contains only specifications. A specific participant interacting with the task in the presentation environment produces an actual Work Product. Many participants responding to the same task will create many such Work Products, each unique but all described in terms of the same Task-Model specifications. These Work Products are the objects the Delivery Process passes to the Response Processing to be evaluated.

Often a Work Product will go through several stages of "parsing" before its contents can finally be used as data to update the Student Model. Determining which of those parsing steps are the responsibility of the Presentation Process and which are the responsibility of the Response Processing is a difficult design decision. Choosing at the right point maximizes the potential for re-use.

To illustrate this point, let us consider a simple variation of the Character Identification Task Model. In a task from this model, the participant hears a sound clip and must choose which

of several characters written on the screen matches the pronounced word. Suppose we only report whether or not the participant selects the correct character. This Work Product might suffice for the purpose of assessing course mastery, but it does not capture information about the kind of mistake a participant makes, which might be useful for diagnostic purposes. Suppose, for example, the stimulus was rén, intending to elicit the correct response 人 (man). The response 任 (rèn, appoint) would indicate confusion of the tones, while the response 入 (rù, enter) would indicate confusion about stroke order. On the other hand, reporting the exact location of the mouse click would produce a Work Product whose interpretation depended on the exact screen layout. This is a level of detail that is not relevant for inference about the participant's understanding. Therefore, for this task, a reasonable design decision would be to have the Presentation Process determine from the low-level mouse-click event which alternative is selected and pass a logical identifier for the option to Response Processing. Depending on the purpose of the assessment, the responsibility of the Response Processing would be either identifying it as correct or incorrect by comparing it with a key (in the overall proficiency model), or indicating the choice as a value in an Observable Variable that provides evidence about the skills at a finer level of detail and/or can be used to provide task-based feedback.

Returning to our primary task, from the Reading Task Model, we merely need to choose a sound format in which to record the speech sample. We may also want to include a secondary flag that tells us whether the participant attempted to answer the prompt or skipped to the next question. Carrying out any natural language recognition at this stage would be premature, even if it were possible to do so here. Assigning the responsibility of evaluation to the Presentation Process and other tasks to the Response Processing by natural logic processing or human raters as would be described in the Evidence Model would better support re-use for different purposes. These purposes might need to inform different Student Models, and thus need to extract different observable variables from the performance.

A task from the Writing Task Model will produce a picture. Again, for similar reasons, we will assign any attempt to recognize the character to the responsibility of Response Processing, to be specified in the Evidence Model. However, even figuring out the correct form for the picture requires some thought. There are any number of bitmap formats we could choose from, but they all ignore an important part of the rules of Chinese handwriting: stroke order. For

example, the primary distinction between 人 (rén, man) and 入 (rù, enter) is the order of the strokes (the extra tail on 入 is added to the printed form to help this distinction). Therefore, it would seem preferable to create a format that consists of a collection of an arbitrary number of strokes, where each stroke would be some representation of the way the character would be stroked on the screen.

On the other hand, practical limitations may cause us to rethink this position. In particular, drawing Chinese characters with a mouse is a difficult task, and quite different from writing them on paper. (We invite the reader to try writing Roman letters with a freehand graphics tool for a comparison.) A graphics tablet is a more realistic choice of input device, but supplying a large number of test stations with graphics tablets might be too expensive. An alternative is to have the participant write the characters on paper—possibly with a brush—and later scan the responses into the computer. However, this would limit the form of the Work Product to a static picture. We could use either a scanned image or the paper drawing as the Work Product. In the latter case, the responsibility of the Presentation Process for capturing the Work Product and those responsibilities of Response Processing for evaluating it will be at least partially a human rather than a computer system.

This last discussion illustrates an important principle of assessment system design: The final form of the Work Product will be a compromise between the needs of the domain experts and the delivery system designers. The domain experts will know what will provide the most evidence about the participant's knowledge, skills, and abilities; the system designers will supply information about what alternatives are likely to cost. In a good design, these two factors are balanced. Again, it should be noted that use of ECD, including the Four Process Delivery Architecture, does *not* assume anything about the role of computers in an assessment. The fact is that using a comprehensive design methodology is essential for designing any assessment that challenges assumptions underlying common forms of assessment. This is most likely to occur when we want to assess for new purposes and/or want to be able to use more complex performance data for more complex inferences.

### *4.1.3 Task Model Variables*

Task Model Variables describe features of the task that are important for designing,

authoring, calibrating, selecting, executing, and scoring it (Mislevy, Steinberg, & Almond, 2002). These variables include features of the task that are important descriptors of the task itself—such as substance, interactivity, size, and complexity—or are descriptors of the task performance environment—such as tools, help, and scaffolding.

In the Chinese Character Assessment, many of the Task Model Variables concern the character to be read or written. The initial, medial, and final sound of the character and the correct tone constitute one group of Task Model Variables. A frequency of usage count for the character and the number of strokes used to create it (a measure of complexity) are two more potentially useful Task Model Variables. For the Lesson Group purpose, the lesson in which the character is introduced is another Task Model Variable.

Other Task Model Variables concern the presentation of the task. For example, in the Writing Task, whether or not the participant can erase and start over, or can request that the stimulus be repeated, are both Task Model Variables controlling the presentation of the task.

The value of a Task Model Variable for a particular task is set when that task is created. Specification Rules tie the values of the Task Model Variables to the Presentation Material for the task. These are either rules for selecting the values for the variables based on the stimulus, or rules for selecting stimuli that meet targets for various Task Model Variables.

Not all Task Model Variables are used in the assessment process as described here. Many play roles before the assessment is ever presented to the participant. Some Task Model Variables are meant to help task authors exercise the full range of activities that the participant must perform. Other Task Model Variables (for example, number of strokes or number of homophones) might be used to model the Weights of Evidence during statistical analysis, but not used during the operational assessment. (See Mislevy, Sheehan, and Wingersky, 1993, on using task variables to reduce pre-testing in the context of IRT.) Still other Task Model Variables, such as Lesson Group, might be useful for some purposes but not others, so they would be defined simply to make tasks easier to reuse.

As a task makes its way from development, through pre-testing and analysis, and into the operational assessment, many of these Task Model Variables are "integrated into" the Task or Evidence Model. In particular, Task Model Variables that are used only in the design or calibration part of the task life cycle may be irrelevant in the operational phase. Such variables

can be omitted from the Task/Evidence Composite, or they can be retained and ignored.

Three of the four assessment processes use Task Model Variables. First, the Activity Selection Process uses Task Model Variables as part of the task selection process to determine the mix and sequence of tasks to administer to a participant (Stocking & Swanson, 1993). For example, the Assembly Model might specify that the participant be presented with a minimum of five characters from very common characters, 10 from common characters, 10 from uncommon characters, and five from rare characters. The Activity Selection Process must check the tasks that it can administer so it will be able to meet this constraint.

Second, the Presentation Process consults Task Model Variables that provide presentation options. One example is the variable that determines whether the participant can request that the stimulus be repeated in the Writing Task. Another example is the variable that determines how long to wait before presenting a "hint" prompt, such as a phonetic rendering of the character.

Third, Response Processing can match Task Model Variables to features of the Work Product. An example would be the Character Identification Task, as used with the Diagnostic Feedback Student Model. The distracters of these multiple-choice items are specifically written to provide evidence about different targeted skills, so it will be necessary to indicate which skills are informed by which distracters. Generally this would be done in terms of Evidence Rule Data. However, because this information might also be used for task selection, we would include it in terms of Task Model Variables and might not care to repeat it as Evidence Rule Data. We therefore allow the Response Processing to access Task Model Variables from tasks in the Task/Evidence Composite Library, in order to increase flexibility.

### 4.2 Evidence Models

The Evidence Model (Figure 8) is the bridge between the Task Model, which describes the task, and the Student Model, which describes the framework for expressing what is known about the participant's state of knowledge. Thus, there has to be at least one Evidence Model for each Task Model in any assessment. However, we may need more than one Evidence Model per Task Model, even within the same assessment, if there is a substantial change in evidentiary focus. Assessments designed for different purposes may need to extract different information from a

Work Product or accumulate it in different ways. Our example assessments with their different purposes—Overall Proficiency, Lesson Groups, and Diagnostic—will each need their own sets of Evidence Models. For example, for tasks from the Reading Task Model we might want to characterize the participant's initial sound, the final sound, and the tone—not only whether each was correct, but also whether it identifies a class of error that suggests the benefit of particular practice exercises.

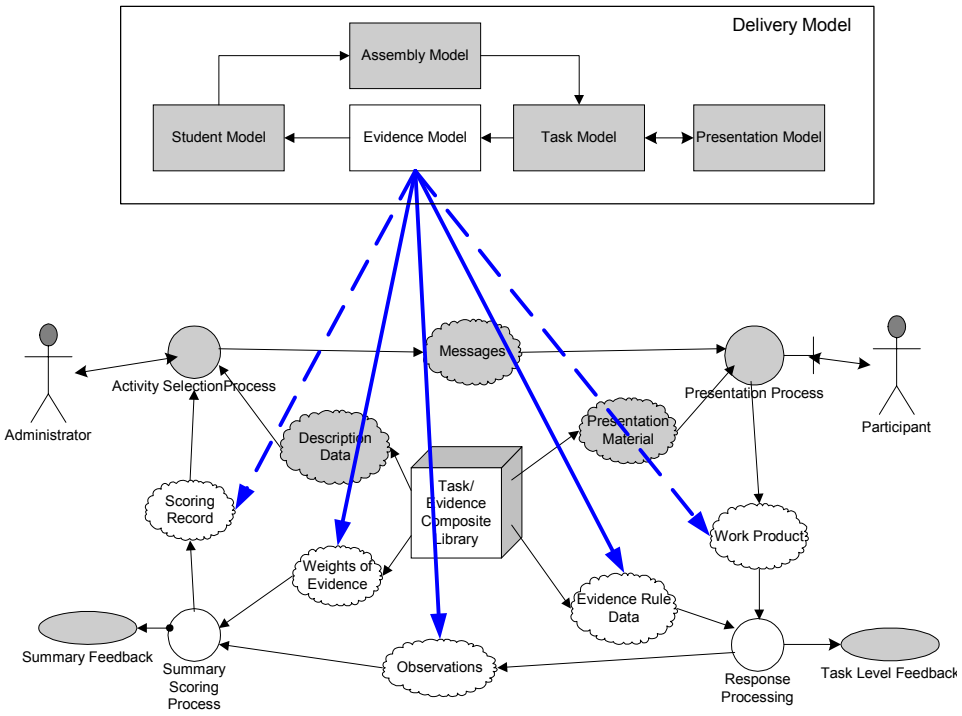


***Figure 8. The Evidence Model***. This model describes the observations that must be made as well as the data that are needed to make those observations and to update the Student Model Variables in Scoring Records in light of the observations (i.e., Weights of Evidence). An Evidence Model updates the Student Model based on data produced from a task written under a given Task Model. Therefore, the Evidence Model must agree with the Task Model as to the descriptions of expected Work Products found in the Task Model, and it must agree with the Student Model as to the descriptions of the knowledge, skill, and ability variables in the Scoring Record.

The Character Identification task can be used to illustrate the need for different Evidence Models both within and across assessments. When it is used for gauging overall proficiency, as in the Student Model for Overall Proficiency, the only Student Model Variable to update is a single overall proficiency measure, and the only observable variable that needs to be extracted is

whether a response is correct or incorrect. An Evidence Model is created to this end. This Evidence Model will not suffice for the diagnostic purpose, however. Although the Task Model and the Work Product are the same, a finer-grained Student Model must be maintained, and more detailed information must be extracted from the Work Product for the Diagnostic Student Model. Evidentiary focus is determined by Task Model Variables describing particular properties of the set of distracters that are offered. If distracters only differ by tone, then the evidentiary focus of the task will be on tonal identification. If the distracters only differ by initial sound, then the evidentiary focus will be on initial sound. We clearly need a more focused Evidence Model than for the Overall Proficiency Model. We may even want to use different Evidence Models for these two tasks, even though they were written according to the same Task Model. When implemented, tasks are stored in the Task/Evidence Composite Library with references to both their Task Models and Evidence Models so the proper models can be used at the time of the assessment.

The work that typically is called "scoring" actually proceeds in two stages, which are important to distinguish for purposes of both design and implementation. In the first stage, Response Processing instantiates (calculates values for) Observables, the key features of the Work Product that make up the body of evidence upon which participant scores will be based. In the second stage, the Summary Scoring Process updates the Scoring Record based on these observed values, integrating the information they contain about a participant in terms of variables defined in the Student Model. An Evidence Model contains information that is needed for both stages.

In particular, Evidence Models contain five kinds of information, all of which carry through from the earliest stages of assessment design to the operational product: Observable Variables, Evidence Rules, Evidence Rule Data, Student Model Variables, and Weights of Evidence.

1. **Observable Variables** indicate what we are looking for in the Work Product. They are the predefined characteristics of the task Work Product that will be evaluated. For our example with the Overall Proficiency Model, we need only determine whether the Work Product of each task conveys a correct or incorrect answer. In order to get information about the various misconceptions

a participant might have, the Diagnostic Model requires more Observable Variables—ones that correspond to the kinds of mistakes that the participant might make, such as tonal confusion, sound confusion, stroke order confusion, or substituting the character with a homophone or one that is similar in appearance.

2. **Evidence Rules** are the rubrics, algorithms, assignment functions, or other methods for evaluating the Work Product. They specify how values are assigned to Observable Variables, and thereby identify those pieces of evidence that can be gleaned from a given Work Product. Evidence Rules embody our argument about what is important for the purpose of our assessment about what participants say, do, or make; how we know it when we see it; and how we express that evaluation in terms of one or more variables. In computerized Response Processing, Evidence Rules are highly specified and embodied, for example, as computer code that operates on a computer file Work Product. Evidence Rules in Response Processing based on human judgment can be just as specific, but they could also be written to allow for a considerable degree of latitude—a generally stated rubric, for example, supplemented by examples of how it has been used to evaluate a variety of responses.[2]

   Generally speaking, there are two kinds of Evidence Rules: Parsing Rules and Evaluation Rules. Parsing Rules re-express the Work Product into a more convenient form. For example, a Parsing Rule might normalize the volume for the sound file that contains a participant's response in an Evidence Model used with a task from the Reading Task Model. Another Parsing Rule might separate the radical part of a drawn character from the "phonetic" (the structural base) for the Writing/Diagnostic Evidence Model. Evaluation Rules actually set the values of Observable Variables. For example, in the Writing/Diagnostic Model, if the participant had the correct number and kinds of strokes but had done them in the wrong order, this would result in setting the value of the Stroke*s* variable to "order confusion," in contrast to "missing

strokes," "extraneous strokes," or "correct strokes and order."

3. **Evidence Rule Data** provides specific information about elements that might be perceived in a Work Product that would cause particular Observable Variables to be set to certain values. A familiar simple example is the key, or correct option, for a multiple-choice item. In the less familiar example of Chinese Character Identification under the Diagnostic Model, the Evidence Rule Data tells us which alternative should map to what kind of mistake on the part of the participant (e.g., radical confusion, homophone confusion, tone confusion). In the Writing/Overall Proficiency Model, the Evidence Rule Data might be information about the gestures used in drawing the character with which the Work Product was to be compared. In this case, the mapping is much more straightforward; if the character meets the tolerance, the Observable Variable indicating correctness is "right," otherwise it is "wrong."
4. References to **Student Model Variables** tell us what each Observable Variable is evidence of (i.e., how the Scoring Record needs to be updated when this piece of evidence is absorbed). Each Observable Variable is linked to one or more Student Model Variables. For the Overall Proficiency purpose, there is only one Student Model Variable, so this part of the model is trivial. For the Lesson Group purpose, each Lesson Group has its own Student Model Variable; the Evidence Model points at the one appropriate for the task at hand. For the Diagnostic Model, this can be quite complex. A given task might draw on several different skills to various extents (Mislevy & Gitomer, 1996; Mislevy, Almond, Yan, & Steinberg, 1999).
5. **Weights of Evidence** inform us about the size and direction of the contribution an Observable Variable makes in updating our belief about the state of its Student Model parent(s). In our work (e.g., Mislevy & Gitomer, 1996; Mislevy, Almond, Yan, & Steinberg, 1999), we employ statistical models for the probabilities of possible responses as a function of a designated subset of Student Model Variables. The Weights of Evidence specify the conditional probabilities. Examples of Weights of Evidence parameters from

psychometrics include factor loadings, IRT item parameters, and true- and false-positive probabilities in latent-class models.

In the Overall Proficiency Evidence Models, for example, the Weights of Evidence correspond to the standard IRT item parameters, such as difficulty and discrimination; the item parameters, in conjunction with the IRT model, completely specify the conditional probabilities of possible item responses given any value of the single Student Model Variable representing overall proficiency, or $\theta$. In the Diagnostic Evidence Models, in which a given response may have multiple Student Model parents, the Weights of Evidence are again either conditional probabilities or parameters of functions that together imply conditional probabilities of response values given the values of Student Model Variables. They tell us not only about the overall difficulty of the item, but also about the relative importance of the various skills in correctly solving the task.

The Evidence Model plays a central role in determining the requirements for Response Processing and the Summary Scoring Process. The former uses Evidence Rules and Evidence Rule Data (as well as information about the structure of the Work Product from the Task Model) to determine how the Observables are set. This could be a human, computer, or human–computer system. The Summary Scoring Process updates the Scoring Record based on the value of the Observables, the referenced Student Model Variables, and the Weights of Evidence.

Finally, the Evidence Model also plays a role in the Activity Selection Process. In an adaptive assessment, the Activity Selection Process is charged with maximizing expected information, relative to its current target and subject to content and format constraints. Expected information is calculated with the current state of the Scoring Record and the Weights of Evidence for a given task.

### 4.3 Student Models

A Student Model[3] (Figure 9) is an explicit structured statement of the knowledge, skills, and abilities in terms of which we have chosen to characterize participants, and will seek to

measure for each participant (Mislevy, 1994; Mislevy, Almond, Yan, & Steinberg, 1999). The purpose of the assessment product guides the choice of these variables; that is, what information the people who use the assessment need for their purposes. A Student Model accumulates the evidence produced across multiple tasks, and synthesizes it in terms of belief about values of the Student Model Variables. These inferences do not provide details on the participant's performance on specific tasks but rather provides more general claims about what the participant knows, can do, or has done. From the perspective of educational measurement, Student Model Variables correspond to constructs. Claims, or the inferences we would like to make about participants with respect to these constructs, are an essential part of assessment design since they represent the manner in which the participant's state of proficiency (as captured in states of Student Model Variables) are interpreted to achieve the purpose of the assessment. Feedback to the participant or other users that is based on the belief about Student Model Variables is distinct from the task-level feedback, which is the responsibility of the Evidence Model.

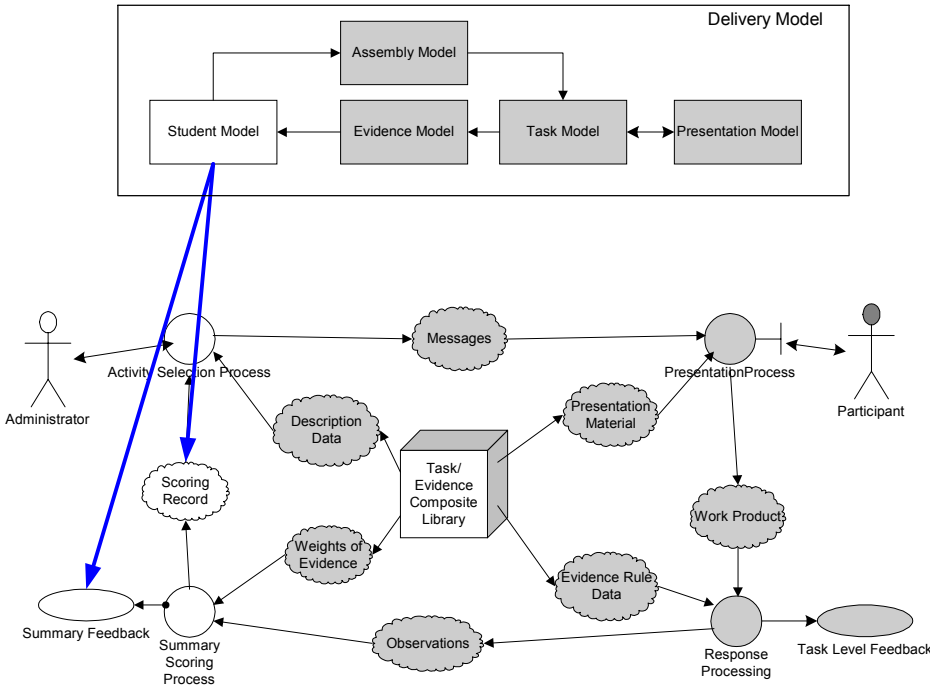


***Figure 9. The Student Model.*** The Student Model describes the Scoring Record and the information available for summary feedback.

A Student Model comprises three types of information: Student Model Variables, Model Type, and Reporting Rules.

- **Student Model Variables** correspond to aspects of proficiency (knowledge, skills, and abilities) the assessment is meant to measure. The Student Model defines not only these knowledge, skill, and ability variables, but also establishes stochastic relationships among them. These relationships could be due to prerequisition, shared knowledge requirements, or simply empirical correlations in the population of interest. Formally, the relationships are expressed in terms of a graphical model that indicates conditional probability distributions among the Student Model Variables.
- **Model Type** describes the mathematical form of the Student Model, such as univariate IRT, multivariate IRT, or discrete Bayesian Network. In the ECD approach, the model type is always a full probability model[4], so the updating is always done using Bayes rule. The particular algorithm used for updating depends on the mathematical form of the model. (As for how the updating is carried out and "where the numbers come from," see Almond and Mislevy, 1999, and Mislevy, Almond, Yan, and Steinberg, 1999.)
- **Reporting Rules** tell us how Student Model Variables should be combined, re-expressed, or sampled to produce scores and how those scores are to be interpreted (claims). Scores are generally functions of one or more Student Model Variables (or more formally, statistics of the distribution representing our state of knowledge about the variable), so there is a close relationship between variable choice in the Student Model and scoring. Reporting Rules can be as simple as reporting our best guess (the expected value) of a single Student Model Variable, or as complex as using the whole Student Model to predict performance on a market basket of pre-defined tasks with known Weights of Evidence (Mislevy, 2000). Other possibilities include probabilities of being above predetermined levels of proficiency or profiles of skill mastery, and draws from the posterior distributions of Student Model Variables.

The Scoring Record contains our beliefs about the values of the Student Model Variables

of a particular participant at any given time during the assessment. Specifically, it maintains a joint probability distribution describing our current beliefs about those variables. The Weights of Evidence (from the Evidence Model) provide a way of predicting the performance of a participant with a given state of the Student Model Variables on a given task. We update the information in the Scoring Record by inverting this prediction using Bayes Theorem, thereby incorporating the new evidence from the performance observed on the task. More technically, the Weights of Evidence associated with the observed performance on the task induce a likelihood function over the Student Model Variables, which is combined via Bayes Theorem with the distribution in the Scoring Record prior to the observation.

### 4.4 Assembly Models

The mission of the Assembly Model (Figure 10) is to provide the information that is needed to control the selection of tasks. For a non-adaptive assessment, the Assembly Model describes how to construct forms. For an adaptive assessment, the Assembly Model describes first how to construct the pool from which assessment forms will be constructed (the Task/EvidenceComposite Library), then how to construct each participant's particular assessment form from that pool in light of the unfolding pattern of responses.

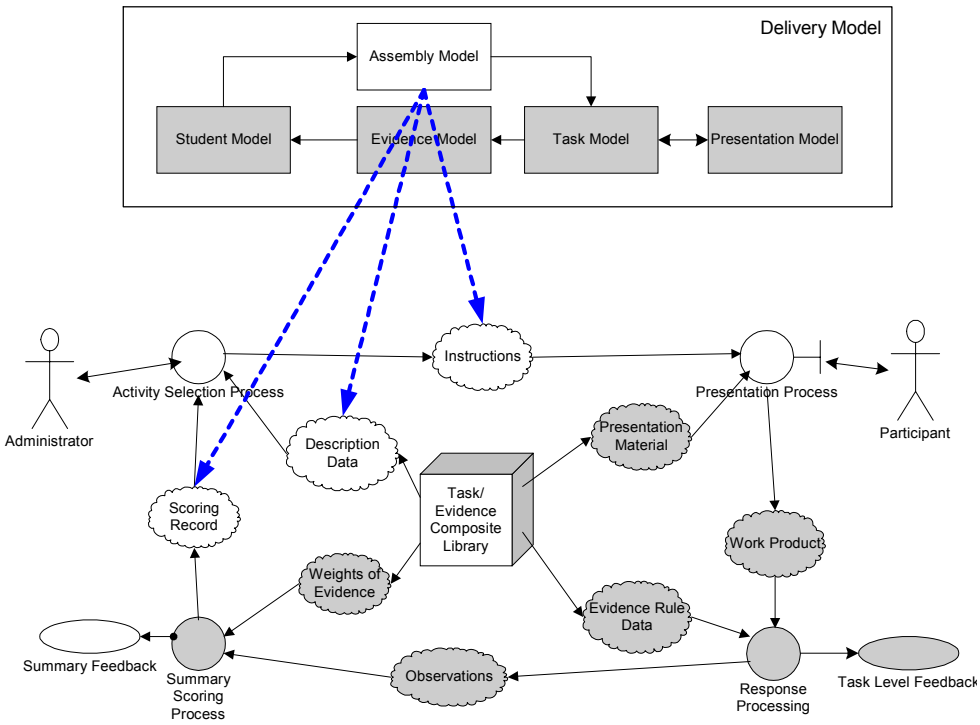


***Figure 10. The Assembly Model***. The Assembly Model describes the strategy used for selecting tasks. These strategies can consult the current Scoring Record and use descriptive data about the task. The Assembly Model also describes how instructions can be used to alter the presentation of tasks in the Presentation Process.

An Assembly Model contains the following types of information: Strategy, Targets, and Constraints.

1. **Strategy**. This is the overall method that will be used to select tasks. Examples include the following:
   - *Linear*: A set of items is selected, possibly according to constraints on information and content, and is available to be administered in the same sequence to any number of participants. Often items are selected long before administration.
   - *Random selection*: Items are selected at random as the participant proceeds through the assessment, possibly with content constraints but not with regard to measurement accuracy for that particular participant.

- *Linear on the fly*: At the beginning of each participant's assessment, a collection of tasks is selected from the pool, possibly in accordance with content constraints. They are presented in a predetermined order to that participant. Each participant can have a different custom-built assessment, but those assessments are not tailored response by response to provide optimal measurement.
- *Adaptive with a single target*: Items, or groups of items, are selected for an individual participant in light of the responses made, with the intent of maximizing information about the same single Student Model Variable (or the same single function of multiple variables). This is the strategy used in most applications of computerized adaptive testing (CAT) (Wainer et al., 1990).
- *Adaptive with multiple targets*: Items or groups of items are selected for an individual in light of previous responses as above, but now the Student Model has more than one variable and information is maximized for different ones (or different functions of them) as testing proceeds. These are the Targets discussed below.

2. **Targets**. Task selection strategies can focus on getting information about some particular aspect of proficiency measured in the Student Model. There are two main variations of this idea, alluded to in the preceding list of item selection strategies. In the first variation, adaptive testing with a single Target, activities are selected to maximize the value of information for the same targeted aspect until some threshold is reached (as in CAT). In the second variation, adaptive testing with multiple Targets, the state of a particular aspect of proficiency may trigger a change in task selection strategy (as in diagnostic testing).

   Consider the Writing Task. This task draws on two skills: listening (recognizing the sound clip) and writing (drawing the character). If a diagnostic system detected that a participant was having difficulty with this kind of task, we would know there was a deficiency in either listening or writing skills. To determine which was the problem, we would make the target listening and choose tasks that focused on listening. If we established that the participant had adequate listening ability, we

would shift the focus to writing skills.

Target specifications in the Assembly Model, then, include those Student Model Variables that are to be examined, and Target Rules that specify how a particular state for each of them controls task selection activities. Generally, the Activity Selection Process will try to select tasks that maximize the value of information for the current target given the other constraints.

3. **Constraints**. A task selection strategy must also respect constraints about particular task features such as specifics of content and format (Stocking & Swanson, 1993). These sorts of constraints are intended to ensure, for example, that the content domain of the assessment is adequately represented, that construct-irrelevant features are not inappropriately over-emphasized, and that evidence is acquired for the intended range of skill and knowledge (Almond & Mislevy, 1999). In our Chinese character example, constraints might specify how many tasks should be selected with common, as opposed to rare, characters.

The Activity Selection Process for a given product may require more than one Selection Strategy, each accompanied by appropriate Target and Constraint definitions, to fulfill its purpose. In our Chinese character example, overall assessment would require one selection strategy, but a different strategy would be required for performing more detailed diagnosis of areas of strength and weakness.

### 4.5 Presentation Model

The Presentation Model (Figure 11a) describes how tasks will look and feel in the delivery environment – how tasks are presented, how evidence is identified and accumulated, and how tasks are scheduled. For example, a paper and pencil (or brush and ink) presentation might require quite a different layout from an on-screen presentation. Separating this from the Task Model allows us to reuse the task in a variety of delivery environments. Although peripheral to the main evidentiary argument, it nevertheless can impact the evidentiary value of various tasks. For example, the use of brush and ink to draw characters has a different evidentiary impact than using a mouse. (Bridgeman, Lennon, and Jackenthal, 2001, describe the evidentiary impact of screen size on SAT-type tasks.)

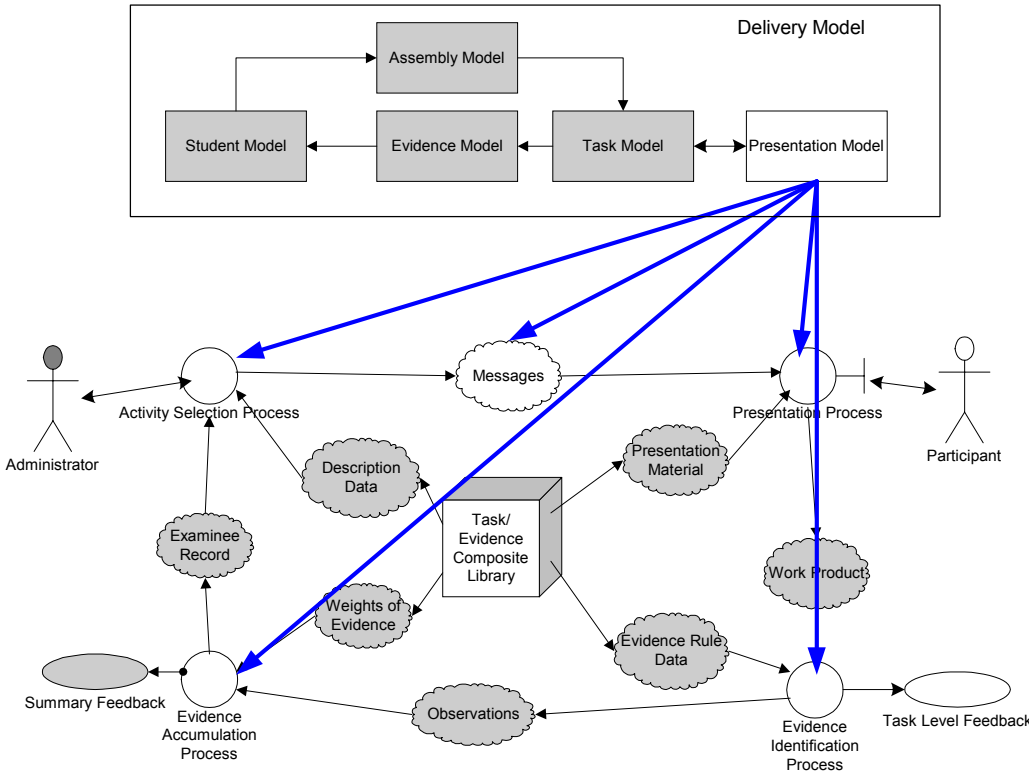


***Figure 11a. The Presentation Model.*** The Presentation Model defines the messages that link the processes and manage the flow of control through the system.

### 4.5 Delivery Model

The Delivery Model (Figure 11b) describes which other models will be used as well as other properties of the assessment that span all four processes. It provides specifications for the following objects:

1. The *platform* to be used to deliver the product, where platform is defined broadly to include human, computer, paper and pencil, etc.
2. The *format* in which objects in each of the various delivery system processes will be rendered or implemented
3. The *administrative requirements* related to security, demographic data collection, transmission and archiving of assessment data, backup and recovery, etc.
4. The *operational models* (Student, Evidence, Task, and Assembly Models) to be included

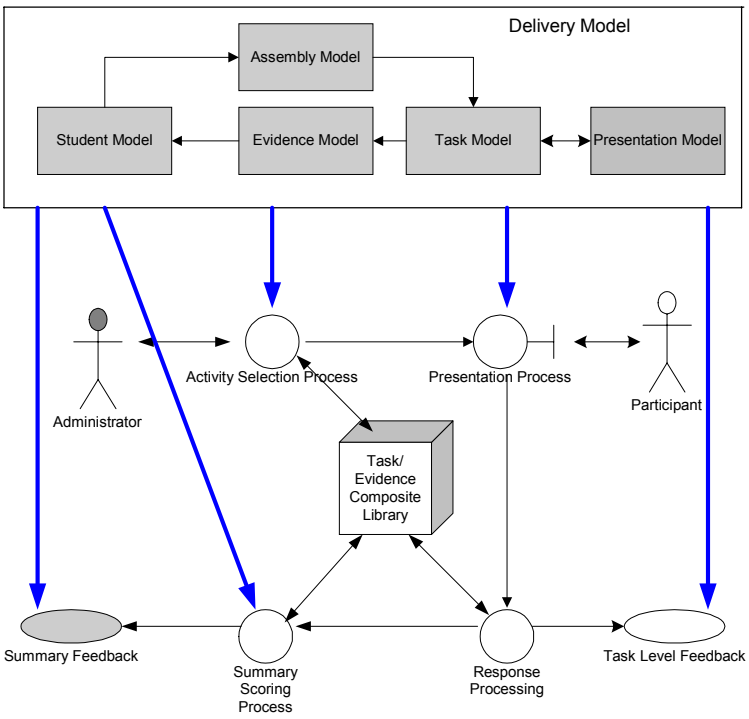


***Figure 11b. The Delivery Model.*** The Delivery Model provides specifications for the platform, the format, and the administrative requirements, as well as the operational Student, Evidence, Task, and Assembly Models.

We could consider two different Delivery Models for our Writing Task example. The first would have the participants write the character with brush and ink on a piece of paper. In this case, the platform would consist of paper, ink, brush, and a means of scanning the response. The second would have participants use workstations equipped with graphics tablets. The selection of models to be included in the operational product, the rendering of the material to be presented, and the administrative requirements would be tailored to each of these platforms. For example, at this point a choice would be made between the Evidence Models that use human raters and the ones that use machine character recognition in the Response Processing.

## Conclusion

Designing a complex assessment is hard work. There are problems of content, functionality, and communication. There are issues of psychology, statistical modeling, and fulfillment of purpose. To make the process efficient, we want to provide the designer with as much structure as we can without constraining design options unnecessarily. We want to

maximize opportunities to reuse assessment objects and processes. Ideally, assessments that serve different purposes should not be expressed in terms of different design objects, but rather as different linkages of instances from the same collection of generic objects.

To this end, this report has described a Four-Process Model for the operation of a generic assessment, and discussed the relationships between the functions and responsibilities of these processes and the objects in the ECD assessment design framework. The complementary modular structures of the design framework and the operational processes encourage the efficiency and reuse we need to make complex assessments practical.

## Acknowledgments

The authors would like to thank Marjorie Biddle, Paul Holland, and Janice Lukas all of whom put significant effort into helping us make the presentation of these ideas easier to understand. We would also like to thank the members of the IMS Global Consortium working group on Question and Test Interoperability, in particular, Richard Johnson, Andy Heath, Steve Lay, and Colin Smythe, for listening to early versions of these ideas and helping us think through the issues of how they play out in the types of assessments they are familiar with. We are happy that they have chosen to include some of these ideas in the information model for sharing information about assessments.

# APPENDIX A

## Delivery System Process Characteristics

### Presentation Processes

The primary purpose of the Presentation Process is to present the tasks to the participant and to return the participant's responses to the task as Work Products. Each different type of task (Task Model) makes demands about the types of material that must be presented (Presentation Material) and the type of responses that must be captured. Therefore, a large part of the description of a Presentation Process depends on which Task Models it will support. However, Task Models may be used in a number of delivery environments; for example, both computer and paper and pencil. Presentation Models describe details of the presentation that are specific to the presentation environment.

As mentioned earlier, Presentation Processes can operate in two different modes: synchronous and asynchronous. In the synchronous mode, the messages from the Activity Selection Process tell which task is next and what its Task Model is. When each task is complete, a Work Product is generated. This is the signal to the Activity Selection Process to pick the next task (although in an adaptive test, the Activity Selection Process may have to wait for item or section level response processing before choosing the next task).

In the asynchronous mode, the interaction is more complicated. In this case, the Presentation Process usually launches a complex task environment, such as a simulator. Interactions between the participant and the task environment lead the Presentation Process to generate Work Products as appropriate. These can be evaluated by Response Processing to produce values of Observable Variables, which are in turn used by the Summary Scoring Process to update the Student Model Variables in the Scoring Record. The Activity Selection Process monitors the state of the Scoring Record and sends messages to the Presentation Process as to when it should change modes; for example, to time out or to interrupt current work to present an instructional task.

For our example of a drill-and-practice system with a primary focus of instruction, we allow the Activity Selection Process to send both a "New Task" (which may be either a new assessment task or diagnostic feedback based on values of diagnostic Observables) and a "Give

Hint" message. When the "Give Hint" message arrives, the Presentation Process is instructed to afford the participant an opportunity to access a phonetic transcription of the character or word currently displayed.

The Presentation Process is responsible for the following tasks: locating and presenting different stimulus media, capturing user input data and creating Work Products, managing interface tools, dynamic screen layout, and messaging.

1. **Locating and presenting different stimulus media**. For tasks from the Reading and Phonetic Transcription Task Models, this means fetching and presenting the bitmap picture of the character. It may be further necessary to translate picture format or load appropriate fonts. For the Writing and Character Identification Tasks, this means presenting the proper sound file. Again, some format translation may be necessary. The Character Identification Task has the additional chore of displaying the characters for the key and the distracters. In all tasks, appropriate material must be fetched from a multimedia database or server. The Materials Specifications in Task Models lay out the specifications for these stimulus objects.
2. **Capturing user input data and creating Work Products**. The Presentation Process is responsible for capturing the participant response, bundling it into Work Products as specified by the Task Model, and executing whatever parsing is necessary to produce the defined Work Product. For the Character Identification Task, this means translating an input gesture into an indicator of which choice was selected. For tasks from the Phonetic Transcription Task Model, it means returning the participant's keystrokes as a string. For tasks from the Reading Task Model, it means converting the captured speech sample into a sound file with the appropriate format. For tasks from the Writing Task Model, it means producing a picture file of the appropriate format, either stroke order or bitmap. Depending on what the Task Model calls for, we may need to convert between one format and the other.

   Note, that if we use the Delivery Model that calls for the characters to be first drawn on paper then scanned into the computer, we need to provide appropriate tools and hardware. In this case, the Presentation Process is a combined human and computer system.

3. **Managing interface tools**. The Presentation Process provides tools for building the presentation interface. There are several kinds of tools:
    - *Primitives*, such as scrolling, buttons, and window manipulation. For example, tasks from the Character Identification Task Model will use a standardized set of selection gestures, and tasks from the Phonetic Transcription Task Model will use a text-input box. Tasks from the Writing and Character Identification Task Models both require a tool to play sound clips. For primitives, the process designer has a choice of whether to use a native toolkit look-and-feel (e.g., Windows, Motif, or MacOS) or to create a uniform look-and-feel across platforms.
    - *Task-specific desktop tools*, such as calculators and dictionaries. For a more complex task, the process might provide access to small applets that can aid the participant in performing the task. For example, in a task that calls for writing a few sentences about a topic or translating a paragraph, the Presentation Process could provide a Chinese-English dictionary. These tools are often re-usable across many tasks. Task Model Variables can instruct the Presentation Process as to whether these tools should be made available (which can both affect task difficulty and shift the focus of evidence, and must therefore be accounted for in Evidence Accumulation, as specified in the Evidence Model, as influences of Task Model Variables on Weights of Evidence).
    - *Task performance environments*, such as simulators and word processors. The most complex tasks will launch complex software that creates and manages internal elements of these environments. For example, a Writing Task could launch a Chinese calligraphy applet to handle user input.
4. **Dynamic screen layout**. In general, the Presentation Process is responsible for the layout of the information to be presented to the participant as part of a task. This allows the Presentation Process to adapt to the particular circumstances, such as an oversize font, small screen size, or pencil and paper. Information about layout comes specifically from the Presentation Model. (A more complicated situation arises when some aspect of delivery is known to have an important but construct-irrelevant

cognitive effect on the task; for example, reading comprehension items tend to be more difficult when presented on a computer screen than when presented on paper. In this case it is necessary to have a Task Model Variable that indicates mode of delivery not only for the Presentation Process, but also for the Summary Scoring Process, which must add a term to item difficulty parameters when instantiating Weights of Evidence for evidence accumulation.)

5. **Messaging.** Finally, the Presentation Process must be able to respond to any messages the Activity Selection Process passes to it. These are specified in the Assembly Model, and can include, for instance, "next task" and "timeout" messages. In our example, the Presentation Process responds to the timeout message by displaying a hint. The Work Product includes a flag to indicate whether the hint was given.

**Response Processing**

When the Presentation Process collects a participant response in the form of one or more Work Products, simple or complex, it passes them to Response Processing to begin the scoring cycle. As with all of the other processes, this could be a human process, a computer system, or some combination of both. For tasks from the Reading Task Model and Writing Task Model in our Chinese language example, we could choose to have character drawings or sound samples rated by humans or scored by machine. In any case, Response Processing is then responsible for notifying the Summary Scoring Process that a response has been made and that the salient characteristics have been distilled from it. The Summary Scoring Process, in turn, updates the Scoring Record, and based on these outcomes, it may pass any or all of them to the Activity Selection Process to guide the flow of the assessment.

Response Processing is responsible for implementing the part of the Evidence Model called the *Evidence Rules*. These are instructions for how to set the values of the Observables, based on the contents of the Work Product(s) and can be different for each Evidence Model. Therefore, an important part of the information about a particular task in the Task Library is which Evidence Model will be used to discern and evaluate the key features in the Work Product the participant creates—in short, "how to score it." The choice of Evidence Model for a task also

depends on the Student Model and hence the purpose of the assessment. The same task could be linked with different Evidence Models when used for different purposes, since different aspects of the Work Product may be important for those purposes, or they are summarized along different dimensions. The appropriate catenation of Task Model and Evidence Model describes the specific tasks in a particular assessment as elements of the Task/Evidence Composite Library.

Response Processing is responsible for the following operations: locating the relevant parts of the Work Product(s), analyzing the problem state, executing evidence rules, setting the values of Observables, and messaging.

1. **Locating the relevant parts of the Work Product(s).** Work Products may contain a large amount of irrelevant material. Response Processing must separate out those parts that will be used for local feedback or scoring. It may also need to translate the format of the information. Suppose we have captured a stroke order representation of the participant's attempt to draw a Chinese character, but a human rater must evaluate it. We may need to translate the abstract representation into a bitmap before we send it to the raters.
2. **Analyzing the problem state**. Response Processing is responsible for monitoring the problem state of the task being performed for purposes of scoring and providing task-based feedback.
3. **Executing Evidence Rules**. Once the relevant portions of the Work Product have been identified (and, if necessary, translated into the correct format), the real work of scoring begins. The Evidence Rules describe how to set the values of the Observables, based on the Work Product and other task-specific data. This is the Evidence Rule Data (which must be retrieved from the task library). As a simple example, the results of the Character Identification Task (a code indicating which alternative was selected) are compared to Evidence Rule Data, which tells which response was the key and which problems each incorrect alternative suggests. The Evidence Rule Data for the full Writing Task would describe the expected strokes and stroke order for the character.[5] Evidence Rules can use other Task Model Variables as well. For example, an Evidence Rule may need to consult the pronunciation of the

character in trying to decide whether a mistake was a phonetic confusion or a pictographic confusion. If the requirements of the assessment require human raters rather than machine scoring, then the Evidence Rules are expressed as a rubric for the human rater. As discussed in the following two operations, executing an Evidence Rule may result in setting the value of an observable, or it may trigger immediate task-based feedback.

4. **Setting the values of Observables**. Response Processing sets the values of the Observables and sends those values on to the Summary Scoring Process. The Evidence Model has specified the number and meaning of the Observables. For example, with the Overall Mastery Model, we can use a simple Evidence Model with the single Boolean observable: "IsCorrect." For use with the Diagnostic Model, we need several Observables that correspond to the various kinds of mistakes for which we want to provide feedback.
5. **Messaging**. In addition to, or instead of, evaluating Observables, Response Processing can use Evidence Rules and Evidence Rule Data to analyze Work Products and problem states to determine whether task-based feedback is required at a given point in time. If so, it creates the appropriate message for the Activity Selection Process. These triggers for task-based feedback differ from information stored in Observables, in that evidence in the Summary Scoring Process Observables is accumulated across tasks in terms of Student Model Variables, while task-based feedback is a strictly local use of the information.

**Summary Scoring Processes**

The evidence accumulation process is responsible for updating the Scoring Record from the observations made about the work product. The Scoring Record contains information about our current beliefs about the student's knowledge, skills, and abilities.[6] Because our beliefs are based on limited observations, we represent our uncertainty about those beliefs with probability distributions. In a probability-based system, the Evidence Model and Weights of Evidence for a particular task allow us to make predictions about how well the participant will perform on a new task. Using Bayesian statistical methods, we can turn these predictions around and use them to

update our beliefs about their knowledge, skills, and abilities (Mislevy, 1994; Mislevy & Gitomer, 1996). Any statistic of the Student Model can be reported as an outcome of the section or assessment level response processing.

Although this model of evidence accumulation is designed to allow the representation of even sophisticated psychometric models, it is flexible enough to represent many potential models, ranging in complexity from simple number right and percent correct scoring, to complex multivariate models. Here is how some common psychometric models fit into this framework.

- **Percent Correct**. The Scoring Record consists of two variables: number of tasks attempted and number of tasks for which the outcome was "Correct." Weights of evidence are all one. Statistics that can be reported are the total number of tasks attempted, the total score, and the percent correct.
- **Weighted Number Right**. The Scoring Record consists of two variables, the total weight of the tasks attempted, and the total weight of tasks for which the outcome was "Correct." The weight of evidence is the maximum possible score for each item. Note that under this model, partial credit can be given for parts of the item.
- **IRT Scaling (Bayesian Formulation).** The Scoring Record consists of the posterior distribution over the unobservable proficiency variable $\theta$. Before seeing any observations, the posterior distribution will be the prior distribution derived from the distribution of $\theta$ in the testing population, or a non-informative "vague" prior distribution. The Weights of Evidence are the IRT parameters for a particular item.[7] After observing each outcome, we update our knowledge about the student's proficiency to produce a posterior distribution over $\theta$. The statistics that can be reported as outcomes include the posterior mean, mode, and standard deviation. (The maximum likelihood formulation of IRT is slightly more complicated because the sufficient statistic is the vector of outcomes along with their IRT parameters of the items which were attempted.[8])
- **Graphical Models** (Bayesian Networks) (Almond & Mislevy, 1999). Here the Scoring Record is multivariate, with each variable representing a different aspect of proficiency. A graph or network is used to represent the structure of dependency among the variables. (In the special case where all Student Model Variables are

discrete, this is a Bayesian network.) The Scoring Record for a particular participant is a Graphical Model, which provides a probability distribution over the Student Model Variables given the evidence provided by those outcomes already observed. The Weights of Evidence are Graphical Model fragments that give the conditional distribution of the outcome variables for a particular task, given the states of one or more Student Model Variables. Using Bayes rule, these predictive models are inverted to provide revised beliefs about the various proficiency variables. The current expected beliefs about any of the Student Model Variables, or any function of the Student Model Variables, can be reported as a section or assessment level outcome from this model.

Exactly which mathematical machinery is appropriate for evidence accumulation depends on the purpose of the assessment. In our Chinese language proficiency example, we could use an IRT model for the Overall Proficiency Model, with right/wrong responses. Here the Weights of Evidence are the standard IRT item parameters (e.g., difficulty, discrimination, guessing), which tell us how likely participants at various proficiency levels are to answer the question correctly. For the Diagnostic Model with Student Model variables representing different aspects of knowledge and skill, we could use discrete Bayesian networks. In this case, the Weights of Evidence could be true- and false-positive probabilities in a multivariate latent class model (e.g., Haertel & Wiley, 1993). Alternatively, we could use a multivariate IRT model in which the parameters convey not only how difficult the task is, but also the relative importance of various skills in performing the task (Adams, Wilson, & Wu, 1997). Finally, if our purpose is primarily self-practice, simple percent correct may be sufficient.

The Summary Scoring Process is responsible for the following operations: absorbing evidence, processing/sampling of reporting variables, calculating value of information, and messaging.

1. **Absorbing evidence.** The Summary Scoring Process is responsible for updating the Scoring Record. In particular, it receives the values of the Observables and inverts the predictive Evidence Model to update belief about the participant's knowledge, skills, and abilities. The particular form of these beliefs is a probability distribution over the values of the Student Model variables that specify those aspects of knowledge, skills,

and abilities that the assessment has been designed to measure.

2. **Processing/sampling of reporting variables**. For both Score Reporting and Activity Selection, the Summary Scoring Process needs to respond to queries about the current state of the Scoring Record. In general, a "score" is any function of the cognitive variables in the Scoring Record. As our beliefs about the cognitive variables are expressed as a probability distribution, we can sample from that distribution and produce Monte Carlo estimates for any score as well as an indicator of its precision, such as a posterior standard deviation or the probability that the participant's true score is above a designated cut point.
3. **Calculating value of information**. How much information we might expect to gain if a participant attempts a given task depends on two things: (1) our current beliefs about the participant's knowledge, skills, and abilities, and (2) the Weights of Evidence that determine how difficult the task is for a person with any given level of knowledge, skill, and ability. For example, if we already know the participant does well on a certain type of task, we will not learn much by administering another easy one. Therefore, the Summary Scoring Process must be able to calculate value of information for a given task on demand. Expected information can be calculated for any particular variable in the Scoring Record, or any specified function of them. Much research on value of information has been carried out in the context of adaptive testing with univariate IRT models (Berger & Veerkamp, 1996). One example of analogous work in multivariate contexts is Madigan and Almond (1996).
4. **Messaging.** The Summary Scoring Process must respond to three kinds of messages: (1) messages from the Response Processing informing it about new observations; specifically, requests to absorb new evidence; (2) messages requesting score reports, to which it responds with status information about Scoring Record variables or score functions; and (3) messages from the Activity Selection Process requesting the value of information for a particular task given the current state of the Scoring Record.

### Activity Selection Processes

The most obvious function of the Activity Selection Process is picking the next task. This

includes both selection—deciding whether or not to present a given task—and sequencing—deciding the order in which to present selected task. But the Activity Selection Process has a number of additional important responsibilities. In an instructional system, it is responsible for monitoring the Scoring Record and changing focus among assessment, diagnostic assessment, and instructional modes of operation. In an asynchronous assessment, it is responsible for interrupting the Presentation Process when warranted by the instructional strategy. In almost all assessments, it is responsible for deciding where to start and when to stop.

The Activity Selection Process is responsible for the following operations: monitoring the state of the assessment, carrying out the assessment/instructional strategy, task selection, customizing the strategy, and messaging.

1. **Monitoring the state of the assessment**. The Activity Selection Process must poll or listen to automatic messages from the other processes in order to monitor the current participant state. If the Activity Selection is adaptive, it needs to monitor belief about the knowledge, skill, and ability variables as maintained in the Scoring Record. Even in a non-adaptive assessment, it will need to monitor information about task exposure in the Examinee Record. In a simulation-based assessment, it may need to monitor the state of the simulator as well.
2. **Carrying out the assessment/instructional strategy**. The Activity Selection Process is responsible for strategic decisions about the operation of the product. For the Overall Proficiency Model, the strategy is very simple: maximize information about overall proficiency. However, for multivariate Student Models, this strategy can be non-trivial. For the Lesson Group Model, the Activity Selection Process is responsible for making the decision about when to shift the focus from Lesson *n* to Lesson *n*+1. The strategy for the Diagnostic Model is even more complex. It may start with general assessments to see whether the participant can perform intrinsically valued tasks—usually integrated tasks that draw on several skills. If the participant shows signs of difficulty, it will shift to a more diagnostic focus and determine which of the requisite skills is weakest. Then, in response to specific problems with specific tasks, it may switch to an

instructional strategy. In this more instructional mode of operation, it needs to decide when to interrupt assessment activities with instructional activities; perhaps the participant is stuck or requests scaffolding.

3. **Task selection**. Given the current strategy, the Activity Selection Process picks the task that best serves the current purposes. Generally, it will pick a task that maximizes the value of information with respect to some Student Model Variable measuring some knowledge, skill, or ability. It chooses the task based on constraints about breadth of tasks (content constraints), constraints about task exposure, and constraints about content overlap. Generally speaking, these constraints are expressed as functions of Task Model Variables. Note that value of information generally depends on both the Weights of Evidence for a task and on current belief about the participant's knowledge, skills, and abilities. Therefore, the Activity Selection Process will usually request that the Summary Scoring Process calculate the value of information for a proposed task.
4. **Customizing the strategy**. The Activity Selection Process may provide administrative options for customizing the assessment strategy. This includes both strategies for accommodating participants with special needs[9], as well as customizing the assessment for a special purpose; for example, selecting which lessons or units will be presented, or making feedback available for learning purposes but unavailable for a final exam.
5. **Messaging**. The messaging requirements for the Activity Selection Process are the most complex, because it needs to monitor the state of the other systems in order to make strategic decisions: (1) It needs to respond to both system and participant driven requests from the Presentation Process; (2) It needs to monitor the acquisition of new Work Products, especially those that indicate that a task has been completed; (3) It needs to monitor the presentation of task-level feedback; (4) It needs to monitor changes to the Scoring Record, and base assessment and instructional decisions on those changes.

## References


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

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

Berger, M. P. F., & Veerkamp, W. J. J. (1996). A review of selection methods for optimal test design. In G. Engelhard, and M. Wilson (Eds.), *Objective measurement: Theory into practice (Vol. 3)*. Norwood, NJ: Ablex.

Bridgeman, B., Lennon, M. L., & Jackenthal, A. (2001). *Effects of screen size, screen resolution, and display rate on computer-based test performance* RR-01-23. Princeton, NJ: Educational Testing Service.

Haertel, E. H., & Wiley, D. E. (1993). Representations of ability structures: Implications for testing. In N. Frederiksen, R. J. Mislevy, & I. I. Bejar (Eds.), *Test theory for a new generation of tests* (pp. 359-384). Hillsdale, NJ: Erlbaum.

Hall, E. P., Rowe, A. L., Pokorny, R. A., & Boyer, B.S. (1996). *A field evaluation of two intelligent tutoring systems.* Brooks Air Force Base, TX: Armstrong Laboratory.

Hambleton, R. K. (1989). Principles and selected applications of item response theory. In R. L. Linn (Ed.), *Educational measurement* (3rd ed., pp. 147-200). New York: American Council on Education/Macmillan.

Madigan, D., & Almond, R. G. (1996). Test selection strategies for belief networks. In D. Fisher & H-J Lenz (Eds.), *Learning from data: AI and Statistics IV* (pp. 89-98). New York: Springer-Verlag.

Mislevy R.J. (1994). Evidence and inference in educational assessment. *Psychometrika, 5,* 439-483.

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

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

Mislevy, R. J., Sheehan, K. M., & Wingersky, M.S. (1993). How to equate tests with little or no data. *Journal of Educational Measurement, 30,* 55-78.

Mislevy, R. J., Steinberg, L. S., & Almond, R. G. (2002 ). On the roles of task model variables in

assessment design. In S. Irvine and P. Kyllonen (Eds.), *Item generation for test development*. Mahwah, NJ: Erlbaum. 97-128.

Mislevy, R. J., Steinberg, L. S., & Almond, R. G. (in press). On the structure of educational assessments. *Measurement: Interdisciplinary Research and Commentary.*

Mislevy, R. J., Steinberg, L. S., Almond, R. G., Haertel, G., & Penuel, W. (in press). Leverage points for improving educational assessment. In B. Means & G. Haertel (Eds.), *Evaluating the effects of technology in education.* Hillsdale, NJ: Erlbaum.

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

Mislevy, R. J., Steinberg, L. S., Breyer, F. J., Almond, R .G., & Johnson, L. (in press). Making sense of data from complex assessments. *Applied Measurement in Education.*

Norman, D. A. (1998). *The invisible computer.* Cambridge, MA: The MIT Press.

Steinberg, L., Mislevy, R. J, Almond, R. G., Baird, A., Cahallan, C., Chernick, H., et al.. (2000). *Using evidence-centered design methodology to design a standards-based learning assessment.* Research report, Educational Testing Service, Princeton, NJ.

Stocking, M. L., & Swanson, L. (1993). A method for severely constrained item selection in adaptive testing. *Applied Psychological Measurement, 17*, 277-292.

Wainer, H., Dorans, N.J., Flaugher, R., Green, B. F., Mislevy, R. J., Steinberg, L. S., et al. (1990). *Computerized adaptive testing: A primer.* Hillsdale, NJ: Lawrence Erlbaum Associates.

Wolf, D., Bixby, J., Glenn, J., and Gardner, H. (1991). To use their minds well: Investigating new forms of student assessment. In G. Grant (Ed.), *Review of Educational Research, Vol. 17* (pp. 31-74). Washington, DC: American Educational Research Association.

## Notes

[1] For more information on the ECD framework, see Mislevy, Steinberg, and Almond (in press) for an overview. Examples of projects in which the ideas are put to work include Almond and Mislevy (1999), Mislevy, Steinberg, Breyer, Almond, and Johnson (1999; in press), Mislevy and Gitomer (1996); Mislevy, Steinberg, and Almond (2002), and Steinberg et al. (2000).

[2] We may note in passing the educational importance of Evidence Rules and the value of making some version of them public—even having participants help create them in classroom applications. Learning the standards of good work is an essential element of learning what a domain is really about, as it is understood in a community of practice (Wolf et al., 1991)

[3] The term *Student Model* has been used by many authors for many things. In our case, it only captures state of knowledge about a participant's knowledge, skills, and abilities. It does not attempt to capture information about learning style or preferences for language and accessible content. For that reason, it might also be called a *Student Proficiency Model*.

[4] The Probability Model is required for the ECD approach, but not for the Four-Process Architecture. However, some other design methodology would be needed. Possibilities include Student Models based on predicate logic.

[5] We really only need to store an index to this data with the actual item. Further, in most Chinese character recognition systems, the character code of the expected character would be sufficient.

[6] The Examinee Record can also contain administrative information about the participant and assessment-related variables, such as tasks that have been presented so far, tasks the participant has seen in previous assessments, lessons that have been mastered, and so on.

[7] Taken together, the form of the IRT model and the item parameters give the conditional distributions of potential responses to a particular item, given $\theta$. The usual assumption in IRT is that responses to all items are conditionally independent given $\theta$.

[8] Under the Rasch IRT models, the sufficient statistics are total scores along with item parameters.

[9] Not all tasks can be adapted for participants with special needs. In those cases, substitutions must be made. For example, the writing task relies on sound output capabilities. A new kind of "Writing Pinyin" task (where the sound clip is replaced by a phonetic transcription) could be substituted; however, it would have a different evidentiary value, so new Evidence Models would be needed as well.

## Appendix A: Glossary of Four-Process Framework Terms

### A

**Activity Selection Process**

The Activity Selection Process is the part of the Assessment Cycle that selects a task or other activity for presentation to an examinee.

**Administrator**

The Administrator is the person responsible for setting up and maintaining the assessment. The Administrator is responsible for starting the process and configuring various choices; for example, whether or not item level feedback will be displayed during the assessment.

**Assembly Model**

The Assembly Model, one of a collection of six different types of models that comprise the Conceptual Assessment Framework (CAF), provides the information required to control the selection of tasks for the creation of an assessment.

**Assessment**

An Assessment is a system (computer, manual, or some combination of the these) that presents examinees, or participants, with work and evaluates the results. This includes high stakes examinations, diagnostic tests, and coached-practice systems, which include embedded assessment.

**Assessment Cycle**

The Assessment Cycle is comprised of four basic processes: Activity Selection, Presentation, Response Processing, and Summary Scoring. The Activity Selection Process selects a task or other activity for presentation to an examinee. The Presentation Process displays the task to the examinee and captures the results (or Work Products) when the examinee performs the task. Response Processing identifies the essential features of the response and records these as a series of Observations. The Summary Scoring Process updates the scoring based on the input it receives from Response Processing. This four-process architecture can work in either synchronous or asynchronous mode.

### BC

**Conceptual Assessment Framework (CAF)**

The Conceptual Assessment Framework builds specific models for use in a particular assessment product (taking into account the specific purposes and requirements of that product). The conceptual assessment framework consists of a collection of six different

types of models that define what objects are needed and how an assessment will function for a particular purpose. The models of the CAF are as follows: the Student Model, the Task Model, the Evidence Model, the Assembly Model, the Presentation Model, and the Delivery Model.

## D

**Delivery Model**

The Delivery Model, one of a collection of six different types of models that comprise the Conceptual Assessment Framework (CAF), describes which other models will be used, as well as other properties of the assessment that span all four processes, such as platform and security requirements.

## E

**Evaluation Rules**

Evaluation Rules are a type of Evidence Rules that set the values of Observable Variables.

**Evidence**

In educational assessment, Evidence is information or observations that allow inferences to be made about aspects of an examinee's proficiency (which are unobservable) from evaluations of observable behaviors in given performance situations.

**Evidence-Centered Assessment Design (ECD)**

Evidence-Centered Assessment Design (ECD) is an ETS-developed methodology for designing assessments that underscores the central role of evidentiary reasoning in assessment design. ECD is based on three premises: (1) an assessment must build around the important knowledge in the domain of interest, and an understanding of how that knowledge is acquired and used; (2) the chain of reasoning from what participants say and do in assessments to inferences about what they know, can do, or should do next, must be based on the principles of evidentiary reasoning; (3) purpose must be the driving force behind design decisions, which reflect constraints, resources, and conditions of use.

**Evidence Model**

The Evidence Model is a set of instructions for interpreting the output of a specific task. It is the bridge between the Task Model, which describes the task, and the Student Model, which describes the framework for expressing what is known about the examinee's state of knowledge. The Evidence Model generally has two parts: (1) A series of Evidence Rules which describe how to identify and characterize essential features of the Work Product; (2) A Statistical Model that tells how the scoring should be updated given the observed features of the response.

**Evidence Rules**

Evidence Rules are the rubrics, algorithms, assignment functions, or other methods for evaluating the response (Work Product). They specify how values are assigned to Observable Variables, and thereby identify those pieces of evidence that can be gleaned from a given response (Work Product).

**Evidence Rule Data**

Evidence Rule Data is data found within the Response Processing. It often takes the form of logical rules.

**Examinee**

See *Participant*.

**Examinee Record**

The Examinee Record is a record of tasks to which the participant is exposed, as well as the participant's Work Products, Observables, and Scoring Record.

## F

**Four Processes**

Any assessment must have four different logical processes. The four processes that comprise the Assessment Cycle include the following: (1) The Activity Selection Process: the system responsible for selecting a task from the task library; (2) The Presentation Process: the process responsible for presenting the task to the examinee; (3) Response Processing: the first step in the scoring process, which identifies the essential features of the response that provide evidence about the examinee's current knowledge, skills, and abilities; (4) The Summary Score Process: the second stage in the scoring process, which updates beliefs about the examinee's knowledge, skills, and abilities based on the evidence provided by the preceding process.

## GHIJKLM

**Instructions**

Instructions are commands sent by the Activity Selection Process to the Presentation Process.

**Model**

A Model is a design object in the CAF that provides requirements for one or more of the Four Processes, particularly for the data structures used by those processes (e.g., Tasks and Scoring Records). A Model describes variables, which appear in data structures used by the Four Processes, whose values are set in the course of authoring the tasks or

running the assessment.

## NO

### Observables

Observables are variables that are produced through the application of Evidence Rules to the task Work Product. Observables describe characteristics to be evaluated in the Work Product and/or may represent aggregations of other observables.

### Observation

An Observation is a specific value for an observable variable for a particular participant.

## P

### Parsing Rules

Parsing Rules are a type of Evidence Rules that re-express the Work Product into a more *convenient* form, where convenient is interpreted to mean the form of the Work Product required by the Evaluation Rules.

### Participant

A Participant is the person whose skills are being assessed. A Participant directly engages with the assessment for any of a variety of purposes (e.g., certification, tutoring, selection, drill and practice, etc.).

### Platform

Platform refers to method that will be used to deliver the presentation materials to the examinees. Platform is defined broadly to include human, computer, paper and pencil, etc.

### Presentation Material

Presentation Material is material that is presented to a participant as part of a task (including stimulus, rubric, prompt, possible options [multiple choice]).

### Presentation Process

The Presentation Process is the part of the Assessment Cycle that displays the task to the examinee and captures the results (or Work Products) when the examinee performs the task.

### Presentation Material Specification

Presentation Material Specifications are a collection of specifications that describe material that will be presented to the examinee as part of a stimulus, prompt, or instructional program.

## QR

### Reporting Rules

Reporting Rules describe how Student Model Variables should be combined or sampled to produce scores, and how those scores should be interpreted.

### Response

See *Work Product*.

### Response Processing

Response Processing is the part of the Assessment Cycle that identifies the essential features of the examinee's response and records these as a series of Observations. At one time referred to as the *Evidence Identification Process*, it emphasizes the key observations in the Work Product that provide evidence.

### Response Processing Data

See *Evidence Rule Data*.

## S

### Scoring Record

The Scoring Record is the portion of the Examinee Record that accumulates beliefs about Participant proficiencies across multiple tasks.

### Statistical Model

The Statistical Model is that part of the Evidence Model that explains how the scoring should be updated given the observed features of the response.

### Strategy

Strategy refers to the overall method that will be used to select tasks in the Assembly Model.

### Student Model

The Student Model is a collection of variables representing knowledge, skills, and abilities of an examinee about which inferences will be made. A Student Model is comprised of the following types of information: (1) Student Model Variables that correspond to aspects of proficiency the assessment is meant to measure; (2) Model Type that describes the mathematical form of the Student Model (e.g., univariate IRT, multivariate IRT, or discrete Bayesian Network); (3) Reporting Rules that explain how the Student Model Variables should be combined or sampled to produce scores.

**Summary Scoring Process**

The Summary Scoring Process is the part of the Assessment Cycle that updates the scoring based on the input it receives from Response Processing. At one time referred to as the *Evidence Accumulation Process*, the Summary Scoring Process plays an important role in accumulating evidence.

## T

**Task**

A Task is a unit of work requested from an examinee during the course of an assessment. In ECD, a task is a specific instance of a Task Model.

**Task/Evidence Composite Library**

The Task/Evidence Composite Library is a database of task objects along with all the information necessary to select and score them. For each such Task/Evidence Composite, the library stores (1) descriptive properties that are used to ensure content coverage and prevent overlap among tasks; (2) specific values of, or references to, Presentation Material and other environmental parameters that are used for delivering the task; (3) specific data that are used to extract the salient characteristics of Work Products; and (4) Weights of Evidence that are used to update the scoring from performances on this task, specifically, scoring weights, conditional probabilities, or parameters in a psychometric model.

**Task Models**

The Task Model is a generic description of a family of tasks that contains (1) a list of variables that are used to describe key features of the tasks, (2) a collection of Presentation Material Specifications that describe material that will be presented to the examinee as part of a stimulus, prompt, or instructional program, and (3) a collection of Work Product Specifications that describe the material that the task will return to the scoring process.

**Task Model Variables**

Task Model Variables describe features of the task that are important for designing, calibrating, selecting, executing, and scoring it. These variables describe features of the task that are important descriptors of the task itself, such as substance, interactivity, size, and complexity, or are descriptors of the task performance environment, such as tools, help, and scaffolding.

## UVWXYZ

### Weights of Evidence

Weights of Evidence are parameters that provide information about the size and direction of the contribution an Observable Variable makes in updating beliefs about the state of its Student Model parent(s). The Weights of Evidence provide a way of predicting the performance of an examinee with a given state of the Student Model variables on a given task.

### Work Product

A Work Product is the Examinee's response a task from a given task model. This could be expressed as a transcript of examinee actions, an artifact created by the examinee and/or other appropriate information. The Work Product provides an important bridge between the Task Model and the Evidence Model. In particular, work products are the input to the Evidence Rules.