A Sample Assessment Using the Four Process Framework

CSE Technical Report 543

Russell Almond and Linda Steinberg,
Educational Testing Service

Robert Mislevy
CRESST/Educational Testing Service

May 2001

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

Project 3.2 Validity of Interpretations and Reporting Results—Evidence and Inference in Assessment
Robert J. Mislevy, Project Director, CRESST/Educational Testing Service

The work reported herein was supported in part under the Educational Research and Development Centers Program, PR/Award Number R305B60002, as administered by the Office of Educational Research and Improvement, U.S. Department of Education.

The findings and opinions expressed in this report do not reflect the positions or policies of the National Institute on Student Achievement, Curriculum, and Assessment, the Office of Educational Research and Improvement, or the U.S. Department of Education.

A SAMPLE ASSESSMENT USING THE FOUR PROCESS FRAMEWORK

Russell Almond, Linda Steinberg, and Robert Mislevy
CRESST/Educational Testing Service

Preface

The authors of this white paper constitute a research group within Educational Testing Service charged with developing methodologies for “reinventing assessment,” specifically assessment that supports learning. These new methodologies grow out of an evidence-centered approach to assessment design. Portions of this evidence-centered design approach are described in various publications given in the references. This white paper is essentially a response of our research group to an early draft of the Question and Test Interoperability specs from Instructional Management Systems. The early draft (Version 0.1) had done a good job of cataloging existing practice, but lacked a framework for future extensibility. This paper is an attempt to remedy this lack.

In the fall of 1999, our research group presented this framework to the QTI working group. After much discussion, the framework was adopted in principle, with substantial changes in the language to better reflect the common practice in the industry. Members of our research team continued to actively participate in the QTI working group to ensure that our approach to assessment design was among those represented in the final QTI specifications.

This version of our white paper has been edited to de-emphasize terminology related specifically to our work in evidence-centered design. A table at the end of the document shows the relationship between the two sets of terminology.

This document only represents the viewpoint of one contributing IMS member to the QTI working group (and for the most part mainly that of our research group within ETS). But because a sufficient number of these ideas found their way—either directly or indirectly—into the final information model or bindings, the working group thought that we should circulate this white paper with the supporting documentation for the project.

This paper has been edited *post hoc* to bring it more in line with the final specifications, but almost all of the original ideas remain. The terminology has been updated, the meaning of certain points clarified, and, where appropriate, links to the QTI information model or XML bindings have been added. We are pleased to offer this white paper to help you with your understanding of the Question and Test Interoperability information model and look forward to sharing tasks and processes with you in the future.

A SAMPLE ASSESSMENT USING THE FOUR PROCESS FRAMEWORK

Russell Almond, Linda Steinberg, and Robert Mislevy
CRESST/Educational Testing Service

Abstract

This paper describes a four-process model for the operation of a generic assessment: ACTIVITY SELECTION, PRESENTATION, RESPONSE PROCESSING (EVIDENCE IDENTIFICATION), and SUMMARY SCORING (EVIDENCE ACCUMULATION). It discusses the relationships between the functions and responsibilities of these processes and the objects in the IMS Question and Test Interoperability (QTI) information model. The ideas are illustrated with hypothetical examples that concern elementary Chinese language proficiency. The complementary modular structures of the design framework and the operational processes encourage efficiency in design and the reuse of objects and processes.

> Infrastructure is usually thought to be dull. Tedious. Few people wish to think about it until it is necessary, which is then often too late. Once established, it is expensive and often difficult to change. Moreover, infrastructures require standardization; they're too expensive and restrictive to allow multiple infrastructures to coexist, too important to society to allow the monetary interests of one company or industry to determine the underlying infrastructure for everyone.
>
> Probably the most important lesson for the development of information appliance industry is the importance of establishing an open, universal standard for exchanging information. If only we can establish world-wide standards for the sharing of information, then the particular infrastructure used within each appliance becomes irrelevant. Each appliance can use whatever best fits its needs. Each company can select whatever infrastructure makes most sense to its operations. Once the information exchange is standardized, nothing else matters.
>
> Donald Norman (1998, pp. 132-133)

The Instructional Management Systems (IMS) project attempts to bring together suppliers of educational material and processes for a variety of purposes and stages in the life of a learner. The challenge facing IMS is to create a framework that supports the delivery of operational assessments fulfilling this range of purposes. This is a tall order. The requirements for a college entrance exam seem quite different from those of assessment to support learning embedded in an Intelligent Tutoring System or from a large-scale survey of educational achievement. The IMS standard for interoperability among assessment delivery and authoring systems must support both the standard multiple-choice and essay-type items, which form the core subset of current practice, and provide sufficient flexibility to grow into the advanced constructed-response items and interactive tasks we envisage as the future of assessment.

To meet these challenges, the Question and Test Interoperability (QTI) specifications required a model of the testing process. One starting point is the general-purpose Evidence-Centered Design Framework that ETS employs for developing educational assessments for a variety of purposes (Almond & Mislevy, 1999; Hall, Rowe, Pokorny, & Boyer, 1996; Mislevy, 1994; Mislevy, Almond, Yan, & Steinberg, 1999; Mislevy & Gitomer, 1996; Mislevy, Steinberg, & Almond, in press). This paper presents a framework for assessment delivery that is fully compatible with the ETS framework. It was adopted by the working group as one of the starting points for the QTI interoperability information model and bindings. The four processes described below are thus the complementary *processes* that are meant to work with the *data structures* defined in QTI.

The objective here is to show how the same conceptual model, defined at the right level of generality, can be used to describe the design objects and delivery processes in assessments that look very different on the surface, and span purposes that range from selection to instructional support. All of the functionality we describe applies to familiar item types and univariate "overall proficiency" scoring models as special cases. The issue is how easy it is to reuse functionality across contexts. As we define requirements for assessment processes, we will work to avoid a trap that comes from studying only familiar assessments: grouping together functionalities that don't need to be separate for these assessments, but which if separated would allow for more flexible recombinations. This capability would make it easier to create new kinds of assessments. It would also make it easier to re-use existing components for new purposes, and to develop new components that would be compatible with one another and with existing components. We are not proposing a change in the essential functionality of an assessment delivery system, but an arrangement of the pieces into functional objects that maximize the potential for reuse in different contexts.

We use the term "assessment" instead of the more specific "test" to emphasize the broad range of assessments we want to think about within the same framework. We want to include high-stakes entrance exams, lower stakes placement and diagnostic tests, tutoring systems to support learning, and even surveys, which may not produce any scores at all for individual students. 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. But we would want to be able to design a cooperative system of assessments that could use the same material for different purposes. Tasks retired from a high-stakes exam could be used in a diagnostic exam, for example, or a practice test or tutoring system. However, the different levels of reporting details that are needed in these uses would require different scoring models. The four process framework provides this flexibility by separating the presentation of the task—described in the task model[1] —from the scoring of the task—described in the evidence model. This ability to separate scoring from presentation provides allows us to reuse tasks in different contexts and to meet the requirements of different assessment purposes.

1. A Simple Example

As a running example, we will consider a Chinese character reading / writing tutor. This example, while relatively simple to describe even to people with little experience with East Asian languages, still forces us to work with nontraditional kinds of data, including audio and pictures. It presents a number of difficult design issues to explore.

Our simple assessment system will contain two kinds of tasks: reading tasks and writing tasks:

Reading Task—The examinee is presented with a picture of one (or more) *hanzi* characters and is requested to pronounce the characters aloud in *putonghua* (People's Republic of China's standard for Mandarin Chinese). The result (*response*) of this task is a speech sample which must be "scored" for accuracy.

Writing Task[2]—The examinee is provided with a speech clip giving both the character and an example of usage of the character. The examinee is asked

[1] At ETS the terms Task and Evidence Model have very specific and formal meanings related to our proprietary design process. However, in this paper we intend to use them informally as models for task design and presentation and response scoring and feedback respectively. The IMS bindings are designed to have places where Items and Sections can be hooked into proprietary models for tasks and scoring by many vendors through the labeling of tags.

[2] A prototype of this task rendered in XML is available as part of the IMS QTI distribution kit. (The applet which actually collects the character drawing, however, is only hypothetical.) The UTF-8 version of this XML should be readable by most computers although the actual text of the instructions and prompt will be visible only on systems with the appropriate Chinese language fonts installed.

to draw the character. The result (*response)* is a picture of the character the examinee drew.

For both of these tasks, we can choose to give the examinee a prompt or help in the form of a phonetic pronunciation guide for the character, or allow the examinee to request such help. For example, the *pinyin* system uses Roman characters to indicate the pronunciation, with accent marks to indicate the tone.

In order to illustrate how the framework works with more usual multiple-choice / short-answer-type items, we introduce two variants on these tasks:

Reading Pinyin Task—The examinee is presented with a picture of one or more *hanzi* characters and is requested to type a phonetic transcription in *pinyin* (People's Republic of China's standard for Romanization system). The result (*response*) of this task is a string of (Roman) characters that can be matched to the key.

Character Identification Task—The examinee is provided with a speech clip giving both the character and an example of usage of the character. The examinee is asked to select the correct character from a list of candidates. The result (*response)* is a logical identifier indicating the selection the candidate made.

The scoring model is intimately tied to the purpose of the assessment. Even within the general purpose of a practice system, we still need to make choices about the granularity of the feedback. In particular, if we want to present detailed diagnostic feedback that addresses student behavior across tasks (for example, confusing tones or initial sounds across tasks) we need student-model variables that will accumulate information across those kinds of observations. Furthermore, our student model must able to provide the information necessary to choose the next tasks if adaptive task selection best suits our purpose. In this paper we will look at three summary scoring models:

Lesson Groups—In this model the characters are grouped into vocabulary sets. Perhaps these follow the lessons of a particular textbook, or they correspond to frequency of use. We assign one student-model variable for each vocabulary set. We assume that it has four levels: *mastered reading and writing; mastered reading but not writing; mastered writing, but not reading;* and *mastered neither reading nor writing.* Under this schema, we do not plan to report any feedback except right / wrong on the tasks and overall level of mastery by vocabulary set (student-model variable).

Diagnostic Feedback—In this model we plan to report targeted feedback on specific problems the student exhibits, partly so we can assign more tasks that draw on the knowledge and skills on which the student seems to be having the most trouble. This kind of feedback could be delivered dynamically as the student works through the task, or it could be delivered at the end of the task. The types of variables an expert instructor might use could include ones related to common speaking and listening problems (e.g., confusing tones, confusing similar initial and terminal phonetic units) and variables related to common writing problems (e.g., stroke order problems, recognizing common radical and phonetic components of a character).

Overall Mastery—In this model we have a single (continuous) variable which indicates the overall level of mastery. We use item response theory (IRT; Hambleton, 1989) scaling to score the various tasks. Task feature variables can be used to help predict the parameters of each item. This model is not capable of giving detailed feedback, but could be used in a "final mastery test" mode. Further, using this model to accumulate information *across* tasks does not preclude giving "local" feedback *within* tasks.

Although straightforward, this example generates numerous issues:

1. **Multimedia.** We need to allow for both audio and pictures as both input and output of a task. We must choose from among a confusing babble of formats and fonts that are potentially useful for our tasks. Our task model must make clear to the presentation, task selection, and scoring processes what is expected.

2. **Representational Issues**. Even more basic is the choice for how we represent a character drawing. We could either 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. We will encounter a tradeoff between convenience and quality of evidence of knowledge about certain aspects of writing.

3. **Input Method.** There are several possibilities for inputting characters. These include drawing with a mouse, using a graphics tablet or light pen, or drawing with brush and paper and scanning the result into the computer.

4. **Task (Item) Scoring Method.** For both the Reading and Writing Tasks we can try to have the work product scored by human raters or parsed by some sort of automatic speech or character recognition program. Many CJK (Chinese, Japanese, and Korean) character recognition programs require the stroke-based representation of characters.

5. **Localization.** For use in China, we would like the instructions to be in Chinese; for use in another country, we may want the instructions in the language used there. There are other features we might also want to tailor; for example, the use of traditional form characters (used in Taiwan and Hong Kong) vs. simplified form characters (used in the People's Republic of China) or the *bopomofo* phonetic characters in place of *pinyin*. In addition, it

would easy to reuse these kinds of tasks in tutoring systems for Japanese, Korean, or Cantonese languages.

6. **Reusability.** Although we limit this example to teaching reading and writing directly in Chinese, it is easy to see how we could extend this tutoring system to include translation tasks. We can also see how we might want embed tasks of this sort in a high-stakes exam offering placement out of a college course. An interesting illustration of the value of interoperability would be a vendor of such placement exams purchasing a special-purpose Presentation Process for these tasks from a software company whose primary business was making CJK software.

This example stretches the limits of the existing model, but it is not farfetched. 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 does a variation of the Writing Task (it works from a *pinyin* prompt rather than from speech samples). The PhonePass system for English language evaluation is a deployed high-stakes 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 automated scoring.

2. 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 2.1), it describes the central repository for information needed to present tasks and evaluate the data they produce (the Task/Evidence Composite Library; Section 2.2), and the messages the processes need to pass from one to another to carry out their responsibilities (Section 2.3). Each of the processes is described more fully in Section 3.

2.1 The Four Processes

Any assessment system must have (at least in some trivial form) four different processes. Figure 1 shows the four processes and the interaction between them.

First, we describe the actors in the system:

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.

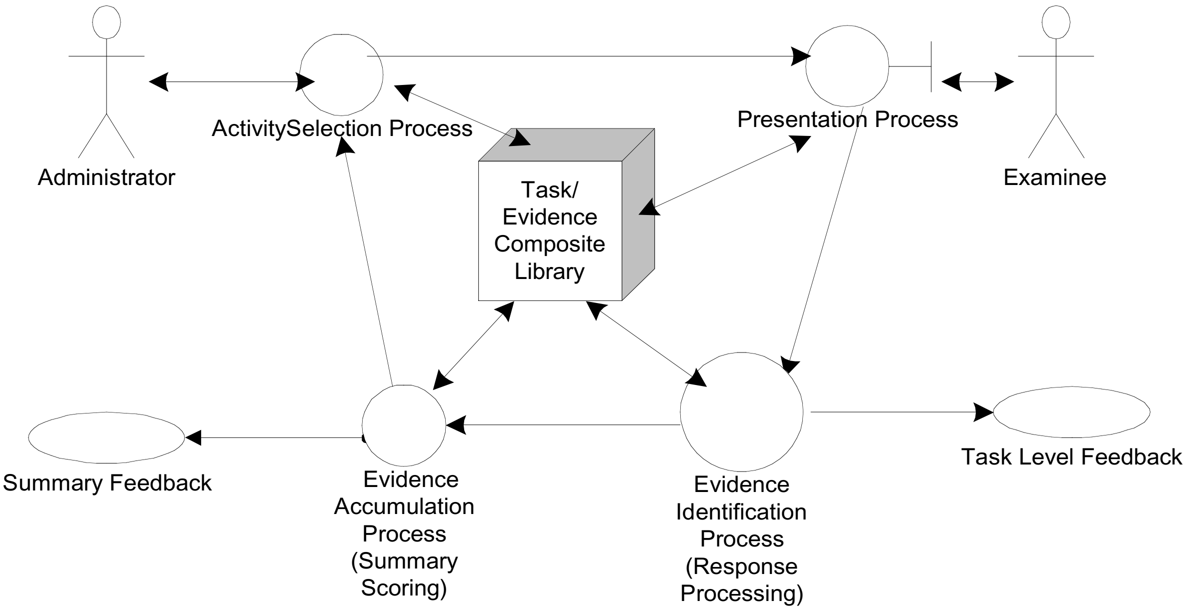
Administrator
ActivitySelection Process
Presentation Process
Examinee
Task/ Evidence Composite Library
Summary Feedback
Evidence Accumulation Process (Summary Scoring)
Evidence Identification Process (Response Processing)
Task Level Feedback

Figure 1. This figure shows the four principle processes in the assessment cycle. The Activity Selection Process selects a task (or other activity) and instructs the Presentation Process to display it. When the examinee has finished interacting with the item, the Presentation Process sends the results (a collection of responses) to the Evidence Identification Process for item-level response processing. This process identifies key outcomes of the task and passes them to the Evidence Accumulation Process (section- or assessment-level scoring), which updates the Examinee Record. The Activity Selection Process then makes a decision about what to do next based on whatever criteria are appropriate, including, for example, tasks already completed or current beliefs about the examinee.

The *Examinee* is the person whose skills are being assessed. The examinee interacts with the various tasks (items) the Presentation Process puts forward.

The assessment cycle is produced by the interaction of these four processes:

The *Activity Selection Process* is the system responsible for selecting and sequencing tasks (or items) from the task library. These could be tasks with an assessment focus or an instructional focus, or activities related to test administration. The task selection process may use any of several different selection algorithms and may consult the current Examinee Record (especially in an adaptive system) to decide when to stop, whether to present a task with instructional or assessment focus, or which kind of task to present next to maximize information about the examinee. Precisely how Activity Selection will work has not been standardized in Version 1.0 of the IMS QTI bindings. Tags have been reserved for <selection> and <sequencing> in the Section and Assessment objects. Also, presumably item-level <meta-data> will be used in complex selection algorithms.

The *Presentation Process* is responsible for presenting the task to the examinee. As necessary it will take details about the task from the task library. (In particular, certain kinds of presentation material such as images, audio or applets may be represented as external resources to be brought in with the presentation of the item.) When the examinee performs the task, the Presentation Process will capture one or more responses (results) from the examinee. These are delivered to the Evidence Identification Process for item-level response processing. The presentation of the task is governed by a task model, which describes what kinds of material must be presented as well as what kinds of responses are expected to be produced. In the IMS QTI binding, this information is specified through the part of the item description in the <presentation> tags.

The *Evidence Identification Process* performs the first step (Item Level Response Processing) in the scoring process: It identifies the essential features of the response that provide evidence about the examinee’s current knowledge, skills, and abilities. These are recorded as a series of outcomes that are passed to the next process. In the IMS QTI binding, this information is specified through the <responseprocessing> tag in the item.

The *Evidence Accumulation Process* performs the second, or summary, stage (Section or Assessment Level Response Processing) in the scoring process: it updates our beliefs about the examinee’s knowledge, skills, and abilities based on this evidence. As we will show below, separating the evidence identification step from both the evidence accumulation and the task presentation is vital to supporting reuse of the task in multiple contexts. How to store data for evidence accumulation has not been standardized in Version 1.0 of the QTI bindings; however, space has been reserved for Section and Assessment Level <response processing>.

The terms “Evidence Identification” and “Evidence Accumulation” were not formally adopted by the IMS working group for these stages of processing. (They are called “Item Level Response Processing” and “Section / Assessment Level Response Processing” respectively.) The names used here follow more closely with ETS’s Evidence-Centered Design methodology, to emphasize that the student-model variables by means of which we synthesize information across tasks may be quite different in kind, number, and nature than the variables we extract from any given task performance. For example, a response to the Writing Task that had the wrong number of strokes would provide evidence that the candidate probably did not recognize the character, which in turn would provide evidence of a lower state on some aspect of Chinese language proficiency; hence we would lower the candidate's score on one or more variables that characterize that aspect of proficiency. We call the first stage of processing (identifying that the response has the wrong number of strokes) evidence identification, and the second stage (updating our beliefs in candidate proficiency as reflected in the score) evidence accumulation. In this paper we have retained the evidence-centered terminology because (a) it is slightly less verbose, (b) it ties in better with our body of published work on this topic, and (c) it provides some clarity in how to “score” advanced examples like the Writing Task.

This four process system can work in either a synchronous or an asynchronous mode. In the synchronous mode, the Activity Selection Process tells the Presentation Process to start a new task after processing the results of the previous task. 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 examinee finishes an appropriate stage of the task. The Activity Selection Process is informed of the change in state of the Examinee Record and decides whether to let the current activities continue or to send a message to the Presentation Process requesting a new activity.

The system is capable of generating two kinds of feedback:

Task-Level Feedback is an immediate response to the examinee actions in a particular task, *independent of evidence from other tasks*. For example, 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.

Summary Feedback is a report about our *accumulated belief based on evidence from multiple tasks*, about the examinee’s knowledge, skills, and abilities along the dimensions measured by the assessment. This can be reported to the Examinee, the Administrator, or other interested parties.

The IMS QTI binding provides tags for both item-level and section-level feedback. Also the standard item-level response processing (Evidence Identification) binding allows feedback to be triggered by certain patterns in the response.

2.2 Task/Evidence Library

The Task/Evidence Composite Library (Figure 2) 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 (a) descriptive properties that are used to ensure content coverage and prevent overlap among tasks; (b) specific values of, or references to, Presentation Material and other environmental parameters that are used for delivering the task; (c) specific data that are used to extract the salient

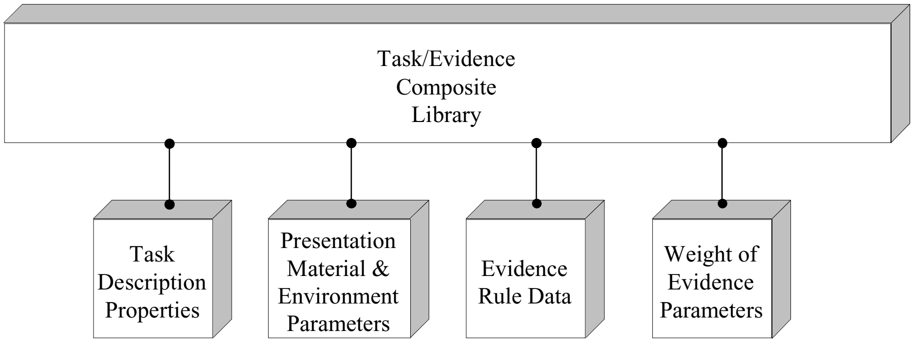

Figure 2. This figure shows the four kinds of information stored with each task / evidence composite.

characteristics of Work Products; and (d) Weights of Evidence that are used to update the Examinee Record from performances on this task—specifically, scoring weights, conditional probabilities, or parameters in a psychometric model.

Exactly what information is stored in each of these four categories will vary from task to task. The task model and evidence model for a particular task govern what data are stored with that task. The task model describes the presentation material this task displays, as well as task model variables that describe the task to the Activity Selection Process and to control options in the Presentation Process. The evidence (scoring[3]) model describes the data necessary for scoring and the parameters used when updating the Examinee Record (the weights of evidence).

If a task is used in two different contexts (hence for two different purposes, using two different student models), it will need different evidence models in those contexts. For use in a particular assessment, the task data must be joined with the evidence rule data and weights of evidence for that task using the evidence model appropriate to the particular task and student models. For this reason, we refer to the entries in the library as *task/evidence composites.*

[3] The term *Evidence Model* is an artifact of the way we think about assessment design in ETS: that each task must provide "evidence" about some aspect of proficiency to be assessed. You can think of the evidence model as a scoring model; however, the scoring model is really the evidence and student models taken together.

2.3 Delivery System Message Objects

Figure 3 shows a more detailed picture of what happens during an assessment. The central ring of Figure 1 has been enlarged to show the data objects that flow around the assessment cycle.

The previous section described the four types of information that come from the Task/Evidence Composite Library. The objects that constitute communications between processes are as follows:

Messages are commands sent by the Activity Selection Process to the Presentation Process. "Start task *X*" is a common and important example. Other messages include time-outs and administrative protocols. From the viewpoint of the QTI schema, the most important part of this message is the identifier attribute of the <item>.

Responses are data objects produced by the examinee in the course of attempting a task. They can be as simple as which selection was 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 a studio art portfolio assessment. In the *Reading Task,* the response is a sound clip, and in the *Writing Task*, the response is a picture of the character. In the QTI bindings, responses are defined by various <response_*xxx*> tags within the <presentation> block.

Outcomes are variables that describe features of the response. A simple and familiar example is whether the examinee got a task right or wrong.

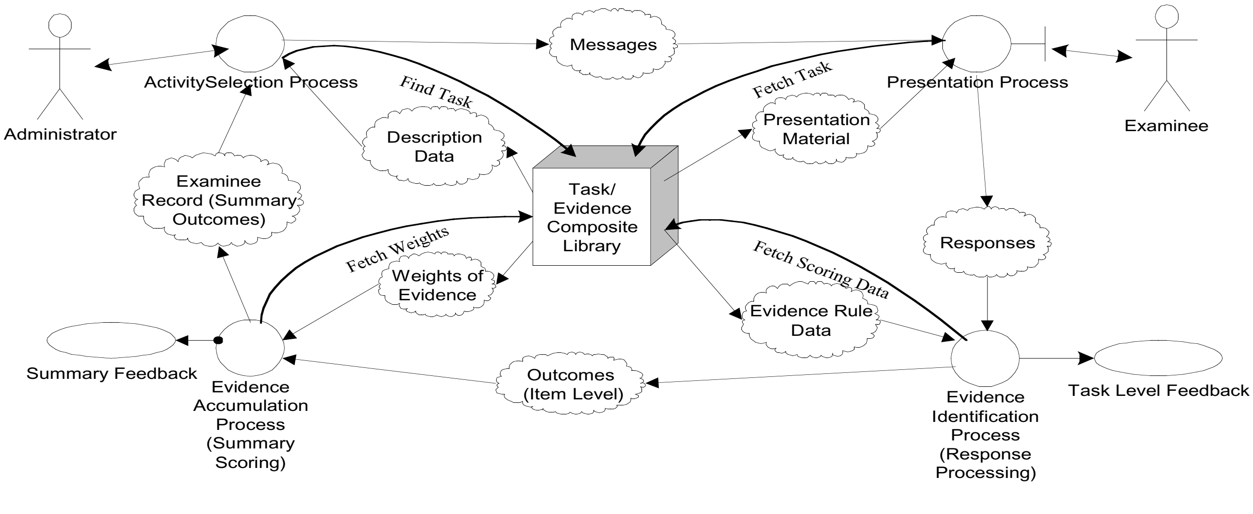

Figure 3. This is a more detailed view of the assessment cycle. Here we expand the picture to show the data objects taken from the Task/Evidence Library and passed around the cycle.

However, for diagnostic purposes, we will often use more complex observations. For example, in the *Reading Task* we might want to characterize the examinee's initial sound, the final sound, and the tone—not only as to whether each was correct, but whether it suggested a class of error that suggests the benefit of particular practice exercises. In the QTI bindings, (item-level) outcomes are defined within the <outcomes> block of the <responseprocessing> block of the <item>.

The *Examinee Record* is a collection of section- and assessment-level outcome[4] variables. These can include variables describing our current state of knowledge about the examinee's knowledge, skills and abilities[5] and variables that record which tasks the examinee has been exposed to, as well as administrative information about the examinee. In the QTI bindings, the Examinee Record is defined through the <outcomes> block in the <section> and <assessment>. The term Examinee Record recognizes that this information may play a role outside of the assessment delivery system. (Within IMS a separate working group is studying issues related to learner profiles.)

3. Examples for Two Different Purposes

Having described the major objects, we can now go back and look at the assessment cycle in two different contexts. Section 3.1 looks a typical high-stakes placement examination; Section 3.2 works through a diagnostically focused tutoring system in the same framework.

3.1 Example 1: High-Stakes Assessment

First, we will look at an assessment system design for high-stakes placement testing shown in Figure 4. Here we are using the *Overall Mastery* student model (which can be supported by the existing IRT-based statistical processes; see Wainer et al., 1990) and the *Reading Pinyin* and *Character Identification* task models. Such an assessment can be accommodated by many existing assessment delivery systems (such as Educational Testing Service's OSA) with limited modification.

[4] Note that the outcomes produced by Evidence Identification Processes are distinct from the outcome variables (updated by the synthesizing of observed outcomes across tasks) in the Examinee Record and may be completely different in nature.

[5] The variables reflecting current knowledge about the examinee may be statistically dependent. For example, in an adaptive test based on Bayesian updating under a multivariate IRT (MIRT) model, the Examinee Record can contain the joint posterior distribution for the student-model variables at any given point in the test. The Activity Selection Process can use this data, along with the MIRT item parameters for items in the Task/Evidence Composite Library that have not been presented yet, to chose an item that maximizes expected information about the examinee in light of her responses so far.

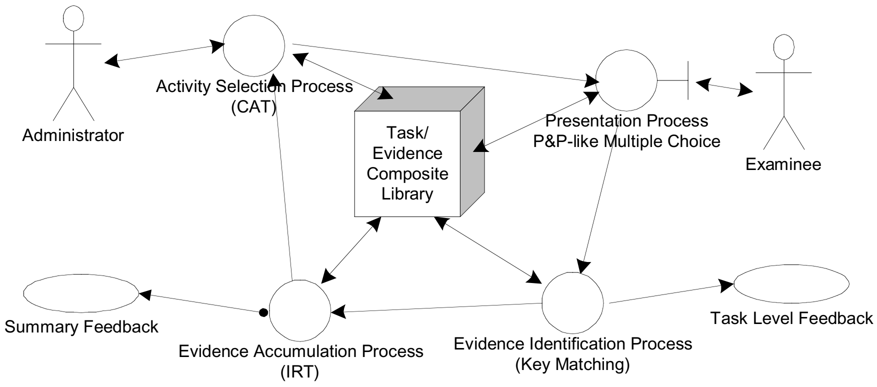

Figure 4. This shows the delivery system processes specialized to a high-stakes placement-type exam. All of these pieces exist (albeit with different names) in current computer-based testing, or CBT, delivery systems.

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 look at the values of certain task meta-data variables—for example, to ensure breadth of content or prevent task overlap (Almond & Mislevy, 1999). In an adaptive test, it also consults the current state of the *Examinee Record* (specifically, at our current estimate of overall proficiency) to select a task that is particularly informative in light of what we know about the examinee's preceding responses (Berger & Veerkamp, 1996).

2. When the *Activity Selection Process* has selected a task, it sends a *message* to the *Presentation Process.* The Presentation Process uses the task model to determine what *presentation material* (<material> embedded in the <presentation> block of the item) is expected for this task and what *responses* will be produced (in this case, either a logical identifier for the selection or a string giving the short response). It might also check flags to set options for the presentation of the task (e.g., use of prompt text in a tutorial mode.)

3. The examinee interacts with the *Presentation Process* to produce some kind of response (in this case just the choice or short answer). This is stored in *response variables,* which are sent to the *Evidence Identification Process* to start the scoring process. The response variables are defined by tags starting with <response> within the <presentation> block of the item.

4. The *Evidence Identification Process* looks at the e*vidence rule data* to ascertain the "key" for the item. It then checks the *responses* against these data using the rules of evidence (either given declarations within the <responseprocessing> block of the item, or by internal logic of the scoring

process) to set the *outcomes* to appropriate values. For operation with the *Overall Mastery* model, only the outcome “isCorrect” (with Boolean value) is relevant.

5. The *Evidence Accumulation Process* takes that outcome and uses it to update the *Examinee Record*. For the *Overall Mastery* model this consists of a single outcome variable θ. The *weights of evidence* in this case are the IRT-scaling parameters (difficulty, discrimination, guessing).

6. The *Activity Selection Process* can now select the next task (or decide to stop). In making this decision, it can use the value and the precision of the updated estimate of proficiency

For this testing purpose, we can mostly use off-the-shelf components. The *Activity Selection Process* is an item selection algorithm from the familiar CAT process. The *Evidence Accumulation Process* is just the standard IRT scoring process. The *Presentation Process* could be an off-the-shelf browser with a few customizations (e.g., support for Chinese fonts). One big difference is that we have separated the first scoring step (the *Evidence Identification Process*) from presentation of the task. (i.e., steps 3 and 4 above.) This may not seem consequential, because the step 4 is so simple in this example (just comparing a tag or string). However, doing so gives the system quite a bit of flexibility for use in other contexts.

Separating the stages has important implications for modularity. None of these processes needs to be computer-based; some or all could be manual processes. We are free to assemble the four processes in a way that best meets the needs of a particular assessment. Thus we could swap out a pronunciation scoring process based on human raters and replace it with one based on computer speech recognition. Alternatively, we could swap out an English language-based Presentation Process and replace it 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) process.

3.2 Example 2: Drill and Practice Tutoring Assessment

To illustrate how components can be reused, we look at hypothetical delivery system processes specialized for use with a drill and practice tutoring system (Figure 5). Here we use either the *Lesson Groups* or the *Diagnostic Feedback* model, and include all four of the proposed task models.

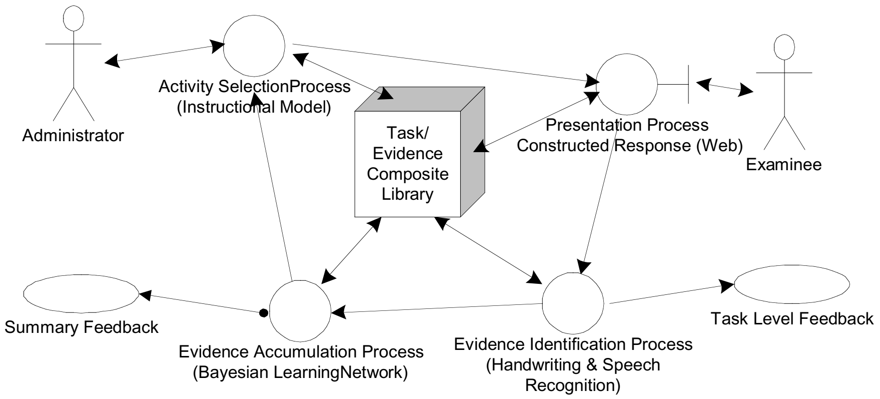

Figure 5. The delivery system processes specialized for a tutoring system. The new processes give more detailed kinds of feedback and accumulate evidence about particular skills to be used diagnostically.

1. We again start with the *Activity Selection Process*. After administrative startup (including possibly loading a previously saved version of the Examinee Record), it selects a task based on the current state of the Examinee Record. Under the *Lesson Groups* model, for example, it would select a task from the first group because those tasks have yet to be mastered.

2. The *Activity Selection Process* sends a message to the *Presentation Process* to start a particular task. If the examinee does not complete that task within a specified time frame, it might send an additional message to provide prompting text.

3. The *Presentation Process* fetches the presentation material from the *Task/Evidence Composite Library*. It presents it to the examinee (as a picture or playing a sound; these are the external resources referenced with URI attributes of <matimage> and <matsound> tags). When the examinee is finished, it bundles the responses and sends them to the *Evidence Identification Process*. Note that for the *Reading Task*, the response will be a sound clip, and for the *Writing Task*, the response will be a picture. (These response types are not standard QTI response types; however, they are easily permitted under the extension facility.)

4. The *Evidence Identification Process* for the *Reading Task* and the *Writing Task* requires either human raters or speech and handwriting recognition. Evidence Identification needs to do more work for the *Diagnostic Feedback* student model than for the *Overall Proficiency* student model. A single observable with values “right” and “wrong” is no longer sufficient. If the student is wrong, we want to know what kind of mistake it was: tone confusion, phoneme confusion, mistaking one character with a common radical for another, and so on. In this case, we need a new *Evidence*

Identification Process even for the *Reading Pinyin* and the *Character Identification* tasks. Under the *Lesson Groups* student model, though, the familiar right/wrong pattern matching routine we used under the *Overall Proficiency* model is still appropriate. For this reason, the QTI spec supports multiple <responseprocessing> blocks for a single item. For example, the *Reading Task* and *Writing Task* might both require different data for human raters and machine scoring. For the *Character Identification* task we will want two different response processing models: one to work with the *Diagnostic Feedback* student model and one to work with the *Overall Proficiency* model.

5. The *Evidence Accumulation Process* must be more sophisticated too. Not only must it determine *how much* our beliefs about the student's abilities should change as a result of our observations, but it must also indicate *which ones* of the variables in the Examinee Record are affected. With the *Lesson Group* model, this is straightforward: Each task belongs to a *Lesson Group,* and we assume limited interaction among the groups. (This can be easily implemented with the <section> structure in the QTI information model.) However, for the *Diagnostic Feedback* model, the presence of the knowledge, skills, and abilities we are trying to measure is often highly correlated (as is our knowledge about them). Therefore, Almond and Mislevy (1999) recommend an approach based on multivariate graphical models, a generalization of more familiar psychometric models, for this step (see also Mislevy, 1994; Mislevy & Gitomer, 1996).

6. Finally, the *Activity Selection Process* would choose the next activity. For the *Lesson Group* model, this decision would be based on how many lessons the examinee is believed to have mastered, as well as whether she has mastered speaking, reading, or both. The *Diagnostic Feedback* model would select tasks to focus on identified trouble areas, as well as having rules for how and when to shift focus on the basis of the evolving state of the Examinee Record.

In our view, assessments meant to fulfill different purposes are not expressed using different objects, but rather by linking different instances of the same collection of generic objects. There is no such thing as an "Instructional Task" as opposed to a "Diagnostic Task" or a "Selection Task." A task model is blind to purpose and to presentation—it participates in fulfilling a specific purpose only when it is linked to a specific evidence model (set of scoring rules), as in the example above. This means a task model 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 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 appropriate instances of the same objects, as specialized for that purpose.

4. Delivery System Process Characteristics

The data objects for the four processes in this delivery model have a straightforward correspondence to data objects in the QTI information model and tags in the XML binding of that model. For the most part, the data for the Presentation Process are found within the <presentation> block of the <item>. Data for the Evidence Identification Process are found within the <responseprocessing> block of the <item>. Although the information model and bindings for activity selection and sequencing and for evidence accumulation are not complete in Version 1.0, placeholders show where they will eventually go (and allow early adopters to experiment with proprietary implementations). The evidence accumulation data go into the <responseprocessing block> of a <section> or <assessment>. Although the representation for those data has yet to be defined, the most important aspect—which variables are reported into the Examinee Record—is defined through the <outcomes> block of the Section and Assessment level response processing. Activity selection data come in through the <sequencing> and <selection> blocks of the <section> and <assessment>, as well as the <precondition> and <postcondition> blocks of the <item> and <section>.

This section explores the purposes and abstract requirements of each of the four processes more fully.

4.1 Presentation Processes

The primary purpose of the Presentation Process is to present the tasks to the examinee and return the examinee’s work in response variables. Each different kind of task (task model) makes demands about the kinds 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 is which task models it will support. In the case of the presentation material, the requirements for a particular task are given in two places: in the “audiotype” attribute of the <mataudio> tag or the “imagetype” of the <matimage> tag, and in the <qmd_renderingtype> within the meta-data for the <item> or <assessment> The response type requirement is also presented in two places: in the <qmd_responstype> tags in the meta-data for the <item> or <assessment> and in the specific <response_*xxx*> tags. Note that two of our examples, the *Reading Task* and the *Writing Task,* require nonstandard types of responses—audio and picture respectively. These types are easily accommodated through the use extensions, although not all systems are likely to interoperate with examples with such advanced requirements. However, the <item> (or <section> or <assessment>) meta-data should help us match the requirements of the item to a corresponding Presentation Process.

Items and tasks from any task model may be used in a number of presentation environments—for example, both computer and paper-and-pencil. Implementers of Presentation Processes describe the presentation-specific details of how tasks are realized (rendered) in their environment.[6]

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 should be next. When the task is complete, it generates a response from the student. 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. The Presentation Process generates new responses at appropriate stages of the examinee's work. The Activity Selection Process monitors the current state of the Examinee Record and sends messages to the Presentation Process as to when it should change modes; for example, to time out, interrupt current work to present an instructional task, and so on.

For our example of the drill system, with its primary focus on instruction, we will allow the Activity Selection Process to send both a "New Task" and a "Give Hint" message. When the "Give Hint" message arrives, the Presentation Process is instructed to afford the examinee an opportunity to access to a phonetic transcription of the character or word that is currently displayed.

The Presentation Process is responsible for the following tasks:

1. *Locating and presenting different input media.* For the Reading and Reading Pinyin tasks, 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,

[6] The IMS Information Model describes item content separately from its rendering in a particular Presentation Process.

appropriate material needs to be fetched from a multimedia database or server.

2. *Capturing user input data and creating work products.* The Presentation Process is responsible for taking the examinee work and bundling it into responses as specified by the task model, executing whatever processing is necessary to produce the defined response types. For the Character Identification task, this means translating an input gesture into an indication of which choice was selected. For the Reading Pinyin task, this means returning the examinee's keystrokes as a string. For the full Reading Task, this means turning the captured speech sample into a sound file of the appropriate format. For the Writing Task, 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.

   An alternative to computer-based presentation for *Writing Task* is to have the examinee write the character on a piece of paper—with a brush!—and scan the paper for subsequent electronic or on-line human rater scoring. In this case, the Presentation Process is both a Human and a Computer system. The "Presentation Process" must provide the appropriate tools and hardware.

3. *Interface tools.* The Presentation Process provides tools for building the presentation interface. There are several kinds of tools.

   a. Primitives, such as scrolling, buttons, and window manipulation. For example, the Character Identification task will use a standardized set of selection gestures; the Reading Pinyin task will use a text-input box. The Writing and Character Identification tasks 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 generate a uniform cross-platform look and feel.

   b. 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 examinee 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 whether these tools should be made available (which, by the way, can both affect task difficulty and shift the focus of evidence).

   c. Task performance environments, such as simulators and word processors. The most complex tasks will launch software that will create and manage internal elements of these environments. For example, the Writing Task could launch a Chinese calligraphy applet to handle user input.

4. *Presentation layout.* In general, the Presentation Process is responsible for the layout of the information to be presented to the examinee as part of a task. This allows the Presentation Process to adapt to the particular circumstances (e.g., large font, small screen size, or paper-and-pencil testing). Information about layout comes specifically from the Presentation Process unless it is known to have an important cognitive effect on the task.

5. *Messaging.* Finally, the Presentation Process must be able to respond to any messages the Activity Selection Process passes to it. These can include, for example, "next task" and "timeout" messages. In our example, the Presentation Process responds to the timeout message by displaying a hint. The response should include a flag to indicate whether or not a hint was given.

4.2 Evidence Identification Processes

When the Presentation Process collects the student responses, it passes them on to the Evidence Identification Process to begin the scoring cycle. Like all of the other processes, this could be a human process, a computer system, or a combination of both. For the *Reading Task* and *Writing Task* in our Chinese language example, we could choose to have character drawings and sound samples rated by humans or scored by machine. In any case, the Evidence Identification Process is responsible for notifying the Evidence Accumulation Process of the *outcomes* of that scoring process. These outcomes are formally defined in the <responseprocessing> block of the <item>. The Evidence Accumulation process updates the Examinee Record based on these outcomes, and it may pass any or all of them along to the Activity Selection Process to guide the flow of the assessment.

The Evidence Identification Process is responsible for implementing the part of the evidence model we call the "rules of evidence." These are instructions for how to set values of the observable outcomes based on the contents of the response variable. These will generally be different for each evidence model. Thus, an important part of the information about the task in the task library is which evidence rules will be used for item-level response processing. The choice of evidence model for a task depends on the student model and hence the purpose of the assessment. The same item could link up with different response processing modules intended for use with different purposes; the example "mchc_ir_103_prespext.xml" provided with the QTI distribution kit shows an example of this. Furthermore, different rules of evidence may require different kinds of data, even if the goals are the same. For example, a human rater will want a scoring rubric and some examples, whereas computer pattern recognition software will require the set of parameters learned from training the software. The example "fibs_ir_102_prespext.xml" provided with the QTI distribution kit shows how to set up <responseprocessing> blocks for both human and machine scoring.

The Evidence Identification Process is responsible for the following operations:

1. *Locating the relevant part(s) of the response.* Complex constructed responses may contain a large amount of irrelevant material. The Evidence Identification Process must separate out those parts that will be used in the task-level feedback and/or scoring stages. It may also be responsible for translating the format of the information. Suppose we have captured a stroke order representation of the examinee's attempt to draw a Chinese character, but it must be evaluated by a human rater. We will need to translate the abstract representation into a bitmap before we send it to the raters.

2. *Executing evidence rules.* Once the relevant portions of the response 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 outcomes based on the content or pattern of responses and other task-specific data (evidence rule data). As a simple example, the response of the Character Identification Task (a logical identifier indicating which alternative was selected) is compared to evidence rule data, which tell which response was the key and which problems each distracter indicates. The evidence rule data for the full *Writing Task* would describe the expected strokes and stroke order for the character.[7] Execution of evidence rules may result in setting the value of outcome variables or triggering task-based feedback.

3. *Creating outcomes.* The Evidence Identification Process establishes the values of the observable outcome variables, which it sends on to the Evidence Accumulation Process. The evidence model controls the number and meaning of the outcome variables. For example, with the *Overall Mastery* model, we can use a simple evidence model with the single Boolean observable: "isCorrect." For the *Diagnostic Feedback* model, we need several outcomes corresponding to the various possible kinds of mistakes for which we want to provide feedback.

4. *Triggering feedback.* The Evidence Identification Process is also responsible for monitoring the responses for purposes of providing task-based feedback. The default scoring rule model of QTI provides a tag to indicate

[7] We actually need to store only 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.

that a block of item-specific feedback should be triggered in response to a particular pattern of responses. Outcome variables can also be passed along to feedback selection rules in the Activity Selection Process (for adaptive selection and sequencing of tasks).

4.3 Evidence Accumulation Processes

The Evidence Accumulation Process is responsible for updating the Examinee Record from the observations made about the work product. The Examinee Record contains information about our current beliefs about the student's knowledge, skills, and abilities.[8] As our beliefs are based on limited observations, many psychometricians represent their 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 examinee will perform on a new task. Using Bayesian statistical methods, we can turn these predictions around and use them to update our beliefs about the 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 student model 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. This Evidence Accumulation Process is supported in the Version 1.0 QTI spec by using the default "SCORE" integer outcome with values of 1 for correct and 0 for incorrect.

Weighted Number Right. The student model 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. This Evidence Accumulation Process is supported by (a) declaring a maximum value for the default "SCORE" outcome and (b) using

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

the maximum scoring weight for the correct response. Note that under this model, partial credit can be given for parts of the item.

IRT Scaling (Bayesian Formulation). The student model consists of the posterior distribution over the unobservable proficiency variable θ. Before seeing any observations, the posterior distribution will be the prior distribution derived from the distribution of θ in the testing population, or a non-informative "vague" prior distribution. The weights of evidence are the IRT parameters for a particular item.[9] After observing each outcome, we update our knowledge about the student's proficiency to produce a posterior distribution over θ. 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 the IRT parameters of the items that were attempted.[10])

Graphical Models (Bayesian Networks; Almond & Mislevy, 1999). Here the student model 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. The graphical model 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 examinees at various proficiency levels are to answer the question correctly. For the *Diagnostic Feedback* 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

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

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

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).

The Evidence Accumulation Process is responsible for the following operations:

1. *Absorbing evidence.* The Evidence Accumulation Process is responsible for updating the Examinee Record. This includes both the administrative part of the record (i.e., which items have been presented) and the cognitive part of the record (i.e., which knowledges, skills, and abilities characterize the examinee). In particular, it takes the outcomes from each task and updates the student-model variables based on the weights of evidence for that task.

2. *Processing/sampling of reporting variables.* For both score reporting and activity selection, the Evidence Accumulation Process needs to respond to queries about the current state of the Examinee Record. In general, a “score” is any function of the variables in the Examinee Record. The statistics that are reported from a student model for a section or assessment are declared in the <outcomes> block of the <responseprocessing> block of the <section> or <item> structure.

3. *Calculating Value of Information.* How much information we expect to gain about an examinee attempting a given task depends on two things: (a) our current beliefs about the examinee’s knowledge, skills, and abilities, and (b) the weights of evidence that determine how difficult the task is for a person with a given level of knowledge, skill, and ability. For example, if we already know the student does well on a certain kind of task, not much information can be gained from similar tasks. Thus, the Evidence Accumulation Process must be able to calculate value-of-information for a given task on demand. 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 Evidence Accumulation Process must respond to three kinds of messages: (a) Messages from the Evidence Identification Process informing it about new observations (i.e., requests to absorb new evidence); (b) messages requesting score reports, to which it responds with status information (statistics) about Examinee Record variables or score functions; (c) messages from the Activity Selection Process requesting the value of information for a particular task given the current state of the Examinee Record.

4.4 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 which order 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 Examinee 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.

The Activity Selection Process is responsible for the following operations:

1. *Monitoring the state of the assessment*. The Activity Selection Process must poll, or listen to automatic messages from, the other processes to monitor the current examinee state. If the activity selection is adaptive, it will need to monitor our knowledge about the knowledge, skill, and ability variables in the Examinee Record. Even in a non-adaptive test, it will need to monitor information about task exposure in the Examinee Record. In a simulator-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 nontrivial. For the *Lesson Group* model, the Activity Selection Process is responsible for making the decision about when to shift the focus of attention from Lesson *n* to Lesson *n*+1. The strategy for the *Diagnostic Feedback* model is even more complex (Madigan & Almond, 1996, describe issues with maximizing weight of evidence in multivariate testing). It may start with general assessments that see whether the student can perform intrinsically valued tasks—usually integrated tasks drawing on several skills. When the student shows evidence of difficulty, it will shift to a diagnostic focus and determine whether particular requisite skills are weak. Then, in response to specific problems with specific tasks, it may switch to an instructional strategy. In this instructional mode of operation, it needs to decide when to interrupt assessment activities with instructional activities; perhaps the student is stuck, or requests scaffolding.

3. *Selection and sequencing tasks*. 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 a student-model variable, measuring some knowledge, skill, or ability. It chooses the task subject to constraints about breadth of tasks (content constraints), constraints about task exposure, and constraints about content overlap. Generally speaking, these constraints are expressed in terms of task description variables (item meta-data). Note that value of information generally depends both on the weights of evidence for a task and on current knowledge about the student's knowledge, skills, and abilities. Therefore, in an adaptive assessment, the Activity Selection

Process sends requests to the Evidence Accumulation Process to calculate the value of information for a proposed task.

4. *Customizing the strategy*. The Activity Selection Process provides administrative options for customizing the assessment strategy. This includes both strategies for accommodating examinees with special needs and customizing the assessment for special purposes—for example, selecting which lessons or units will be presented, or making feedback available for learning purposes but unavailable for the final exam.

5. *Messaging*. The messaging requirements for this process are the most complex, because it needs to monitor the state of the other processes in order to make strategic decisions. In particular, it needs to respond to both system- and examinee-driven requests from the Presentation Process. It needs to monitor the acquisition of new work products, especially those that indicate that a task has been completed. It needs to monitor the presentation of task-level feedback. It needs to monitor changes to the Examinee Record, and base assessment and instructional decisions on those changes. In cases where the scoring is done off-line, it will need to make strategic and tactical decisions based on the previous state of the Examinee Record.

5. Implementing This Framework in the QTI Information Model

This four process framework provides a flexible, logical structure in which a wide variety of assessment products can be implemented. In any given implementation, these processes may be grouped (for example, the Presentation and Evidence Identification processes might be bundled so that both could be done on the client side of a client-server architecture). However, keeping them separate in the logical model makes it easier to reassemble them later with a new process (for example, exchanging human raters for machine scoring). This framework should both handle the current state of best practice and scale to future assessments with complex task, scoring, and interactivity requirements.

The adoption of this framework by IMS should provide two marketplaces for assessment components. The first is a marketplace for assessments, sections, and items (as defined in the QTI Version 1.0 and Version 2.0 information models). The second is a marketplace for the four processes themselves as software (or human-computer system) components. Although these are only implicitly defined in the current version, this version does provide the definitions for the data structures that provide the point of contact between the four processes.

Version 1.0 of the QTI information model provides definitions for the critical data structures that form the interfaces among the four processes. The *responses* from the Presentation Process and the *outcomes* from the Evidence Identification Process are required parts of the item data structure. The *Examinee Record* will consist largely of *outcome* variables defined in the Assessment and Section level response processing (even though the rest of those processes are largely undefined). If a bank of items is stored in a format compatible with the QTI information model (any format that can be mapped onto the QTI bindings), then it should be possible to separate the data about the item into the four pieces called for in the task/evidence composite library, each piece going to a different one of the four processes.

Each institution that implements the QTI specifications will have its own models for the kinds of tasks and evidentiary (scoring) algorithms they commonly use (even if these models are never formally stated). In general, these task and evidence models will be more detailed than the QTI framework, providing syntactic and semantic constraints that would not be appropriate in the more general-purpose framework. Each institution will need to determine how to best implement its task and evidence models in the interoperability framework, as well as how to best use the labeling structure to link between its models and the QTI model.

In order to facilitate this implementation process, we have provided two resources: Table 1, which shows the correspondence between the names we used in this paper and those used in the QTI specs, and Table 2, which lists advanced examples that illustrate how the ideas presented in this paper play out in actual items. Most of these advanced examples are available in an annotated slideshow format where the XML code is placed in the slide and information about the design philosophy is placed in the speaker's notes. We hope these materials will help readers match their own ideas in assessment design with those presented here and in the QTI specifications.

Table 1

Correspondence Between the Names of the Objects as Defined in Educational Testing Service's Framework for Assessment Design and the Names Adopted by the IMS Question and Test Interoperability Working Group

ETS term	IMS QTI term	XML binding	Notes
Presentation Process	Presentation	*Not bound*	Process—not bound in QTI bindings.
Presentation Material	Material	<item> <presentation> <material>	Material that is part of the presentation block of the item.
Work Product	Response	<item> <presentation> <response_xxx>	Can be multiple response tags. *xxx* is replaced with standard QTI type or proprietary extension.
Evidence Identification Process	Item Level Response Processing	*Not bound*	Evidence identification emphasizes the role of the response in making judgements about the candidate's ability.
Evidence Rule Data	Response Processing Data	<item> <responseprocessing>	Data found within the item response processing block are evidence rule data. Often this takes the form of logical rules.
Observables	(Item Level) Outcomes	<item> <responseprocessing> <outcomes>	Name "observables" is meant to evoke that these outcomes are the key observations on which we will base claims about the student.
Evidence Accumulation Process	Section / Assessment Level Response Processing	*Not bound*	Name "Evidence Accumulation" is meant to evoke the combination of outcomes from many tasks.
Weights of Evidence	Scoring Weights	*Version 2.0*	In Version 1.0 these can be stored within proprietary extensions in either the item, section, or assessment response processing block.
Examinee Record	(Section / Assessment Level) Outcomes	<section> | <assessment> <responseprocessing> <outcomes>	The Examinee Record typically will be the section / assessment-level outcomes plus administrative information about an examinee. A different IMS working group on Profiles is investigating similar issues.
Activity Selection Process	Item / Section Selection and Sequencing	*Not bound*	This will be taken up in Version 2.0 of the QTI spec.
Task Description Properties	Meta-data, and Pre-Post Conditions	*Version 2.0*	A number of placeholders for selection and sequencing data have been left in the Version 1.0 specs.
Messages (to Presentation Process)	*unnamed*	<item ident=""> and <displayfeedback>	Messaging capability of Presentation Process is not specified by QTI. Presumably most processes will support the ability to display an item with a given ID. Many will support the ability to display general and item specific feedback.

Table 2

Pointers to Various Examples in the QTI Distribution Kit That Were Developed to Illustrate or Test Some of the Ideas in This Paper

Name	Description	Interesting features
mchc_ir_103_prespext.xml	Multiple-choice question with culminating and diagnostic scoring options.	Multiple response processing modules for interoperation with multiple evidence identification and accumulation processes. Use of labels to indicate task model structure.
fibs_ir_102_prespext.xml	Essay question with human and computer response processing.	Multiple response processing modules for different Evidence Identification Processes. Also, additional module for use with diagnostic evidence accumulation. Shows how to use proprietary response processing modules.
mchc_ir_102.xml	Reading testlet with three questions.	Shows one approach to multiple items that share presentation material. Also shows how to do text hotspots and provides multiple response processing models.
mchc_ir_007.xml	Chinese Sentence Completion 1: Multiple Choice	A simple sentence completion item using multiple choice. Presentation material uses non-ASCII text encoding.
fibs_ir_008.xml?	Chinese Sentence Completion 2: String Response	A simple sentence completion item using string response. Both presentation material and response use non-ASCII text encoding.
ext_ir_009.xml?	Chinese Sentence Completion 3: Character Drawing	This is essentially the *Writing Task* described in this paper. Response is a proprietary format from a fictitious applet. Response processing is done by human or computer raters.
fibs_ir_101b.xml	Sentence completion with two responses.	This example shows how to take tabular and/or pseudo-code representations for response processing and implement them with the QTI bindings for the default response processing method.

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, & M. Wilson (Eds.), *Objective measurement: Theory into practice (Vol. 3).* Norwood, NJ: Ablex.

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: Lawrence Erlbaum Associates.

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, 59,* 439-483.

Mislevy, R. J., Almond, R. G., Yan, D., & Steinberg, L. S. (1999). Bayes nets in educational assessment: Where do the numbers come from? In K. B. Laskey & H. Prade (Eds.), *Proceedings of the Fifteenth Conference on Uncertainty in Artificial Intelligence* (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., Steinberg, L. S., & Almond, R. G. (in press). On the roles of task model variables in assessment design. In S. Irvine & P. Kyllonen (Eds.), *Generating items for cognitive tests: Theory and practice.* Hillsdale, NJ: Lawrence Erlbaum Associates.

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

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