

Requirements Analysis

Smarter Balanced Assessment Consortium
Test Delivery System

Test Authoring
Level II Requirements

Contents

[Revision History - Revisions based on additional requirements gathering.](#)

[Revision Description](#)

[Author/Modifier](#)

[Date](#)

[Terms and Definitions](#)

[Component Description](#)

[Functional Requirements from SBAC-11 Architecture Report](#)

[Functional Requirements from SBAC-11 Architecture Report](#)

[Data Utilized](#)

[Test Specifications and Test Packages](#)

[Test Packaging Function](#)

[6.3 CAT Simulator Interface](#)

[6.4 Test Domain Element Hierarchy](#)

[Domain Element Sequence Workflow](#)

*

[Roles and Permissions](#)

[Deployment](#)

[Client Components](#)

[Assumptions](#)

[Requirements](#)

[User Interface Summary](#)[User Interface Screen Mockups](#)[Q&A Section This section is for screenshots and questions during development](#)

Revision History - Revisions based on additional requirements gathering.

<i>Revision Description</i>	<i>Author/Modifier</i>	<i>Date</i>
Final Draft (Originally submitted Final Draft located on KT)	Paul Krumrei	31 October 2013
<p>Draft 3</p> <ul style="list-style-type: none"> · Added Copy Test screen 07-3. · Reworked section 6 to describe test specifications, test packages and test simulation. · Reworked Section 7 Workflow to conform to RFP-07 requirement 45 · Added a requirement to include a three-character short name in the subject metadata. · Added Test Family to new/edit Test dialog (this needs to be emitted as a property of the testspecification element). · Added storage of simulation information and attachments. · Added storage of psychometric information and attachments. · Changed item selection algorithm configuration description and mockups · Changed computation rule configuration description and mockups · Added requirements and descriptions of conversion tables, and added conversion 	David Lopez de Quintana	31 October 2013

table loading in Scoring Rules		
Transformed into Google Document	Paul Krumrei	Nov 2013
<ul style="list-style-type: none"> - Added screen shots from David - Q&A Section Added - Duplicate Names Added - Authoring application creating permission - Roles and SSO features 	Paul Krumrei	Dec 2013
<ul style="list-style-type: none"> - Core Standards for the SBAC-ELA - DTD Pool Property information added - Item Groups - Validation Rules Determined - Implicit Associations Determined - Explicit Associations Determined - Info Enemies enforced by adaptive segments added including enemy types Determined - Master and Segment Blueprint Constraints Determined - Various Stages of rules determined - Assessment name uniqueness with Tenants - Display Parent Nodes when Search blueprint with Standard Key - Item Selection Algorithms determined - Added total quantity of items in Master Pool - Added total quantity of items in segment pool - Added total quantity of items in typical test - Added total quantity of items in each standard 	Paul Krumrei	Jan 2014
<ul style="list-style-type: none"> - Added Additional information on Implicit associations - Added Additional information on Explicit associates and the UI - UI Changes determined (screen shots to be updated) 	Paul Krumrei	Feb 2014

- Added Clarification on Pool Properties - Added Clarification on Performance and Measures - Added Forms, Form Partitions, Adding Items to Form Partitions - Added Pyschometrics information - Added Upload of Simulation results - Added Reporting Measures update - Added Permissions and Mapping UI to Backend Services - Added Additional Pool Property and segment information		
Removed old DTD and example XML artifacts table	Ryan Marinello	Feb 28,2014
Added new DTD example XML artifacts table with requirements and req' no.	Ryan Marinello	Feb 28,2014

Terms and Definitions

Term	Definition
APIP	Accessible Portable Item Profile. A technical standard that focuses on accessibility in assessment items.
Asset	Multimedia (digital text, images, audio and video) associated with an item
Blueprint	The design for a test. The test blueprint indicates the number of test questions or points related to each competency on the test and the relative emphasis placed on each competency.
Field Test	Test made up of test items intended to develop and calibrate new assessments.
Grade Level	Grade or standard the item is designed for
Item	A question on a test. It may be composed of several parts including audio and visual files.
Item Author	A person or agency responsible for item creation.
Item Bank	A systems application that manages the storage and retrieval of assessment items, tracks item versioning and lineage, and provides a robust search and query capability that allows searching on all types of metadata.
Item Status	A term used to indicate how an item can be used. Some of the valid values are: <ul style="list-style-type: none"> - Pilot Test - Field Test - Operational
JSON	JavaScript Object Notation, a lightweight, text based, open standard designed for human-readable data interchange.
Pilot Test	A trial series of new or modified items given to a select group of students.
Retired Item	An item that will no longer be used for an assessment.
Rubric	a guide listing specific criteria for grading or scoring academic papers, projects, or tests
Runtime exception	An unanticipated condition for which no error handling exists but normal processing cannot continue
Source Item Bank	The Item Bank from which an item is pushed to a Test Item Bank
Test	An instrument or procedure that proposes a sequence of tasks to which a student is to respond. The results are then used to form measures to define the relative value of the trait to which the test refers.
XML	Extensible Markup Language, a set of rules for encoding documents in machine-readable form.

Component Description

The Test Authoring component is responsible for the creation, management and workflow of test specification data. It uses the Test Spec Bank component to store and retrieve test specification data and queries the Test Item Bank for item data based on item metadata for assignment to adaptive item pools and test forms.

Test Packager functionality is downstream of Test Authoring, and uses test specifications that have advanced to a given level in the workflow to be combined with item data from the Test Item Bank to create test packages for various purposes including registration, simulation, scoring, reporting and administration.

The Test Authoring and Test Packager components reference the same components and must be aware of the same test specification workflow. If they were separate components, there would be a large amount of shared interface and workflow code. For this reason, the two components can be merged and Test Packaging can be considered to a function of Test Authoring.

The Core Standards component provides standards publications as a source of standards information by Test Authoring for the creation of blueprints and other uses.

The following diagram illustrates the relationship between these components.

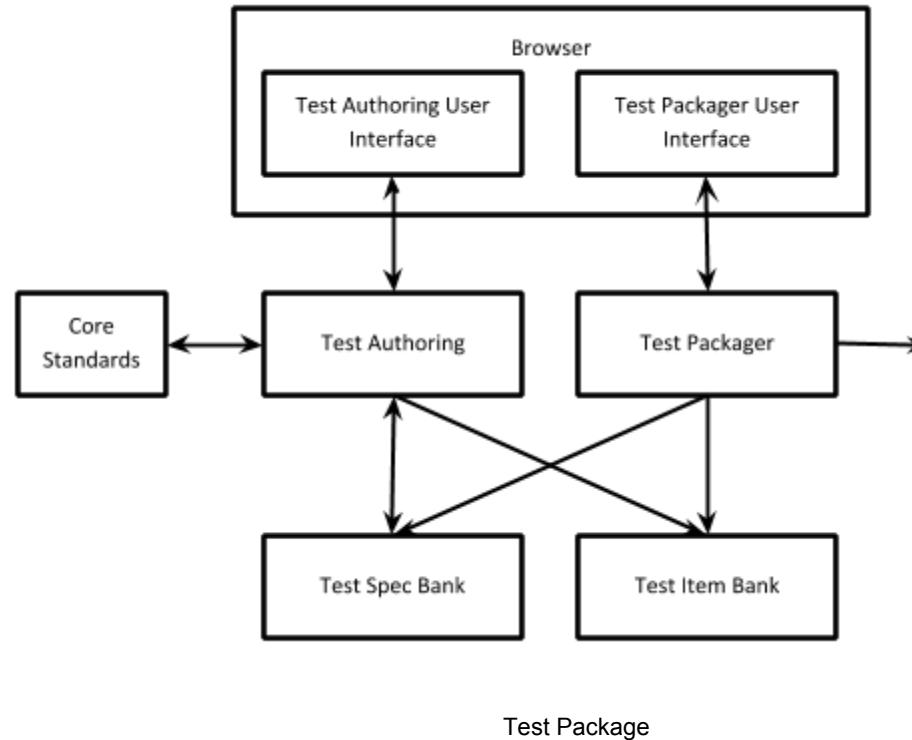


Figure 1. Test Authoring and Related Components

Functional Requirements from SBAC-11 Architecture Report

The SBAC-11 Architecture Report provides only a definition of the function of Test Authoring:

This component is a graphical interface used for creating test blueprints and specifications and manage the workflow. It will interact with the Test Item Bank component and the Test Spec Bank component.

Data Utilized

	Data Utilized	Additional Information
1.	Item Data	Item data is contained in the Test Item Bank which stores items that are ready for operational or field test use on an assessment. Test Authoring performs queries on the Test Item Bank for items that meet various search criteria for allocation to an adaptive item pool or a fixed test form.
2.	Test Specification Data	Test Specification data stores various domain entities such as tests, blueprints and forms. Item data and test specifications together form the basis for test packages that can be transferred and loaded into a test delivery system for online test administration to students.
3.	Standards Publication	Test Authoring uses standards publications provided by the Core Standards component as a source for standards-related data, primarily for blueprint creation. Every test requires a standards publication

Table 2. Data Utilized

Test Specifications and Test Packages

Test Package is the combination of a test specification in XML format that conforms to a test specification DTD, and optionally (depending on the purpose of the test package) item metadata, or item content, metadata and item assets. Not all types of test package require item content, metadata or assets, only those intended for simulation and administration.

Test Specification is a blanket term that is used to refer to a hierarchy of domain elements that are created and managed by the Test Authoring component. The following diagram visually illustrates the domain elements managed by Test Authoring. Domain elements managed by the Test Authoring application do not have to be stored at rest in the same XML format defined for test specifications. The

domain elements must be formatted into XMLs conforming to a DTD only when the Test Packaging function is used to create a test package for a given test.

The following table illustrates the different purposes for test packages.

Test Package Purpose	Contains	Description
Registration	Test Specifications	Registration packages intended for the Test Registration system. They are an abridged form of a full item package, and contain only minimal data that is needed for establishing an available test in Test Registration.
Simulation	Test Specifications and Item Metadata	A Simulation package is intended for the CAT simulator. Packages for this purpose contain all the test specifications that are needed for an Administration package, but only provide item metadata that is used by an adaptive algorithm to do item selection. Actual item content and item assets such graphics and multimedia files are not required for simulation purposes.
Administration	Test Specifications, Item Metadata, Item Content, Item Assets	An Administration package has the same test specifications as a package intended for simulation, but includes all of the item content and assets that are required for administration to students that the simulation package does not provide.
Scoring	Test Specifications	A Scoring package contains all of the test specifications that are required for configuration of a test scoring engine. It does not contain any item information beyond the item pool references in the test specifications.
Reporting	Test Specifications	A Scoring package contains all of the test specifications that are required for configuration of a reporting engine. If the reports do not require item or form distribution reports, they do not require test specifications beyond the master blueprint, reporting measures and performance levels.
Complete	Test Specifications	A Complete package is a superset of all test specifications and item information. If sufficient information is present for a Complete package then sufficient information is available for all other purposes.

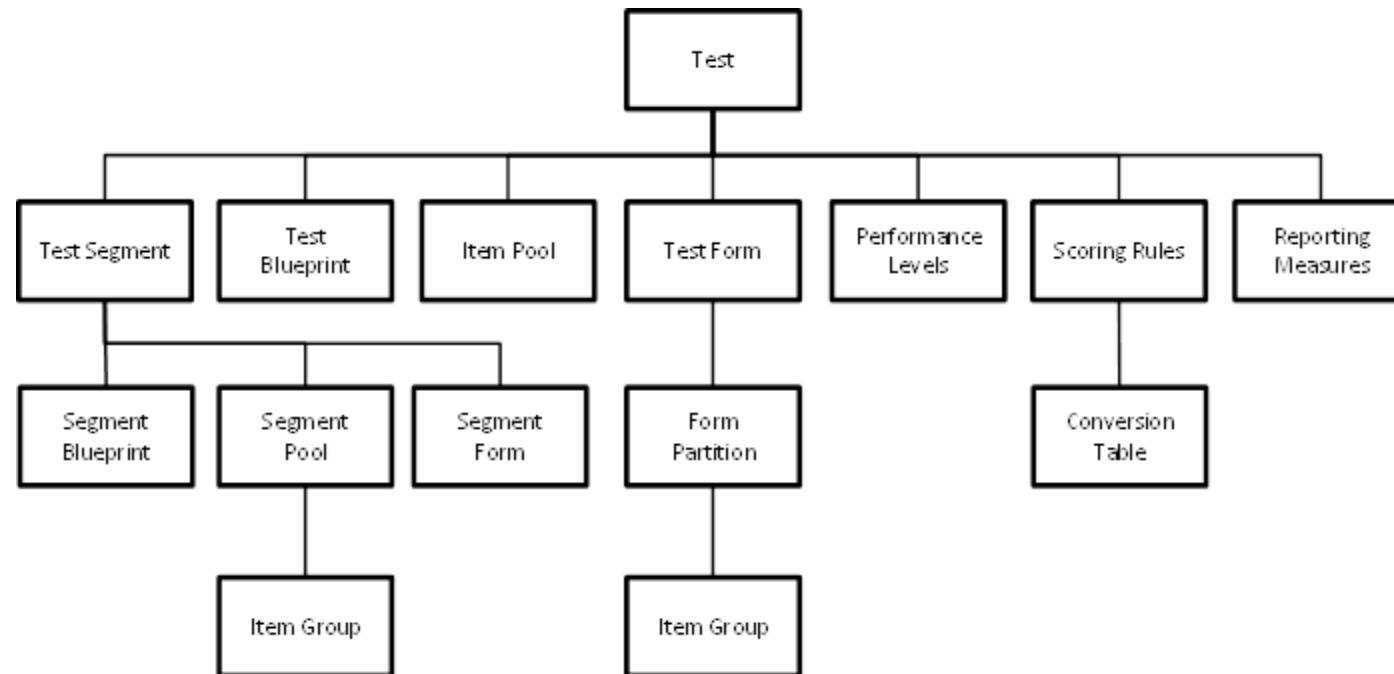


Figure 2. Test Authoring Domain Elements Hierarchy

The following are the DTD and example XML artifacts accompanying this document.

Req. Number	Requirement	Test Package Artifact	Description
RFP 7	Test Authoring Component will meet requirements detailed in the SBAC IT Systems Architecture (Appendix C)		Outlined in the RFP, this is the header of the main requirement for Test Packaging.
PRTIB 1	Create XML file containing data required for the DTD.	testpackage.dtd	A formal specification of the XML elements that represent a test. XML files that conform to this DTD are suitable the following purposes: 1. Administration: an XML representation of a test that

			<p>contains all elements necessary to administer the test to a student.</p> <p>2. An Administration XML packaged with item metadata is suitable for adaptive test simulation</p> <p>3. An Administration XML packaged with full item content, metadata and item assets can be used to configure a Test Delivery system for administration of a test to students</p> <p>4. Scoring: an XML representation of a test that contains sufficient elements for configuration of a test scoring system with custom rules for scale scoring this test</p> <p>5. Reporting: an XML representation of a test that contains sufficient elements for configuration of a test reporting system for reporting of the scale scores for a test</p> <p>6. Complete: A superset of all the test elements separately required in all of the test representations above</p>
PRTIB 2	Create XML file containing data required for the DTD.	DE_ALG_registration.xml	An abridged test specification that is used to configure the Test Registration component. No item data is packaged in this format.
PRTIB 2	Create XML file containing data required for the DTD.	DE_PT_ALG_admin.xml	A test specification that is used to perform adaptive simulations or to configure a test delivery system. A simulation XML must be packaged with the item metadata, and an XML intended for a test delivery component requires full item metadata, content and all item assets such as graphics and multimedia files.
PRTIB 3	Create XML file containing data required for the DTD.	DE_read4_scoring.xml	A test specification that is used to configure a test scoring system. Only item references and scoring information is packaged in this format.
PRTIB 4	Create XML file containing data required for the DTD.	DE_read4_reporting.xml	A test specification that is used to configure a reporting system. No item data is packaged in this format.

PRTIB 5	Create XML file containing data required for the DTD.	DE_ALG_complete.xml	A test specification that is a superset of all the elements described above. The four test specification types above contain some common elements and some unique elements. The Complete package contains a superset of all elements.
PRTIB 6	A Test Packaging function must be included with the Test Authoring application to create Test Packages for various purposes. In order to create a test package explicitly using the Test Packaging user interface, the test and its sub-elements must have been fully approved and be in the Approved or Published workflow level described in section 7.2 Test Approval Workflow		

Test data at rest in the Test Authoring system are not necessarily stored in this form, but when a test package is created, the test elements must be exported as XML files that conform to this DTD.

Test Packaging Function

A Test Packaging function must be included with the Test Authoring application to create Test Packages for various purposes. In order to create a test package explicitly using the Test Packaging user interface, the test and its sub-elements must have been fully approved and be in the Approved or Published workflow level described in section 7.2 Test Approval Workflow.

6.3 CAT Simulator Interface

A REST interface is required in the Test Authoring application so that the CAT Simulator can extract test packages for simulation. These test packages must be at least in the Validated workflow level described in section 7.2 Test Approval Workflow. It is not necessary for tests to be fully approved in this workflow to be eligible for simulation because simulation is a pre-production step that is required to tune an

adaptive test to ensure that it is capable of meeting the test blueprint. The test must be validated first to ensure that the domain elements are correct and consistent prior to attempting simulation.

The following diagram illustrates the sequence of test simulation.

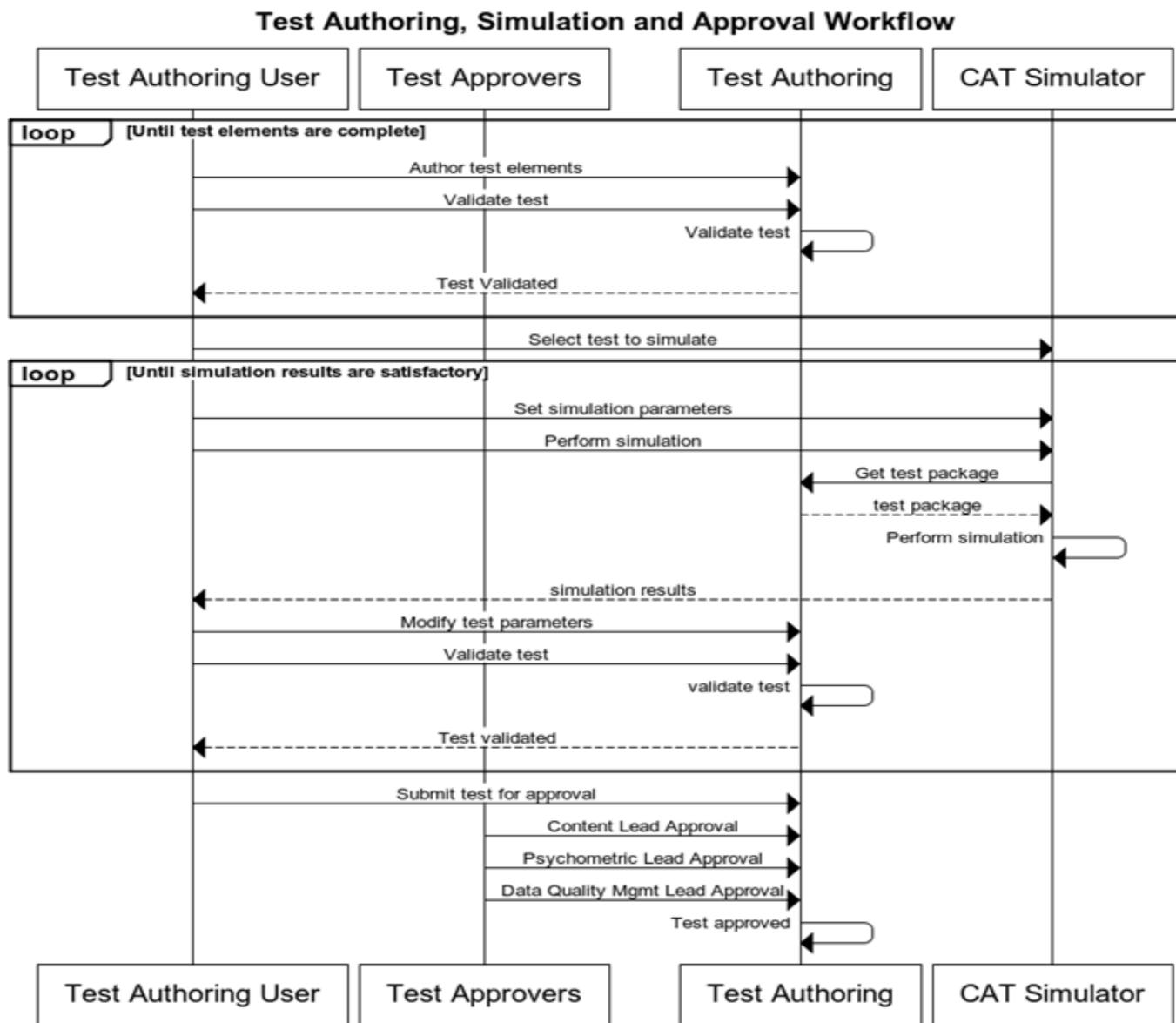


Figure 3 Adaptive Test Authoring, Simulation and Approval Workflow

In this workflow, the user works with the Test Authoring user interface to create and refine the individual test elements, using validation as a tool to understand what test element is incomplete or inconsistent with other elements.

If this is an adaptive test, the user will want to simulate it. Starting with a validated test, the user sets up the CAT Simulator to perform simulations on the test. The CAT Simulator should only have visibility to validated tests in Test Authoring so that the test elements are assured to be complete and consistent. The CAT Simulator uses a REST interface on Test Authoring to obtain a test package for simulation. It performs simulations and reports back to the user. The user makes the necessary corrections in Test Authoring, validates the test, and repeats the process until the test meets the required blueprint parameters.

Once simulations are complete, the test is submitted for approval and approved according to the workflow specified in section 7.2 Test Approval Workflow.

The Test Authoring application also provides an interface for the CAT Simulator to store simulation results. This interface creates a Simulation record stored with the test that stores a fixed number of predefined fields, but also allows the attachment of arbitrary artifacts to the record. These artifacts may be any file type including text, spreadsheet or pdf files.

6.4 Test Domain Element Hierarchy

The following diagram visually illustrates the hierarchy of domain elements managed by Test Authoring.

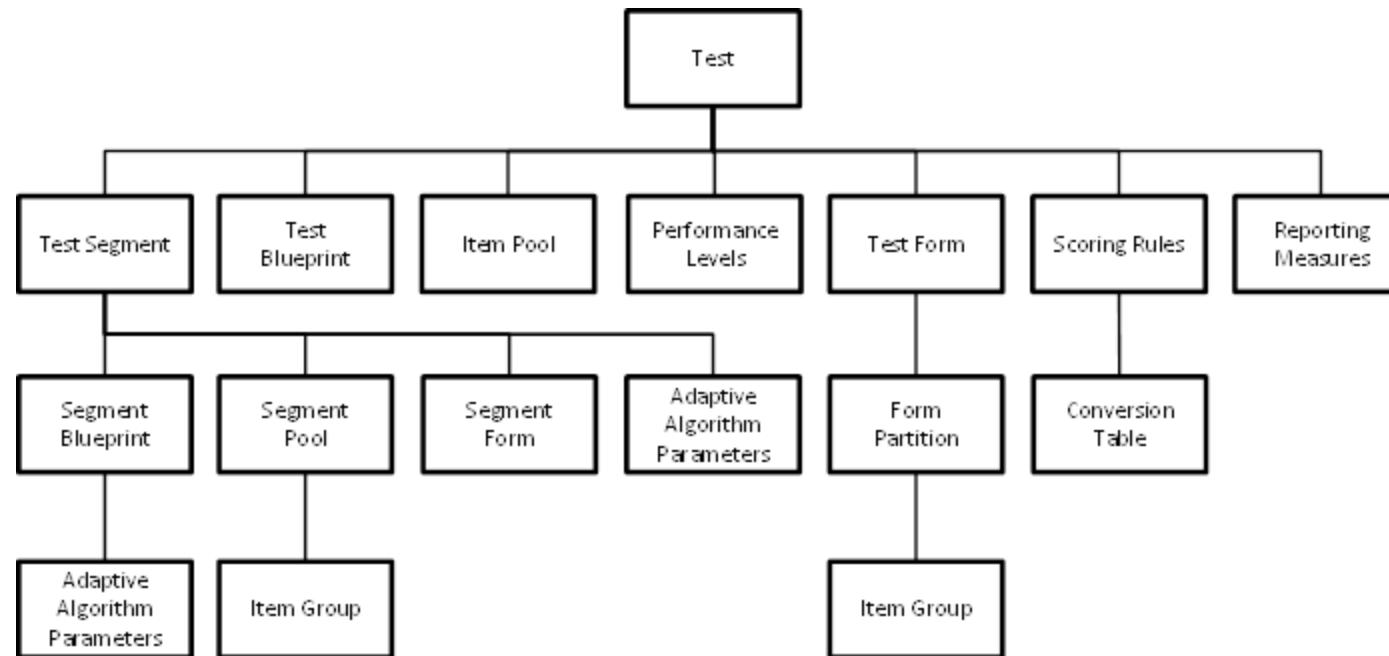


Figure 4. Test Authoring Domain Elements Hierarchy

The following table summarizes the test specification domain elements and provides a description of each.

Element	Parent Element	Description
Test	None, highest level element	The Test is the highest level element in Test Authoring. Tests are created by test authors on behalf of a Publisher which is a tenant in Test Authoring.
Test Segment	Test	All tests are segmented, and a test must contain at least one test segment. The use of tools such as calculators are managed on a per-segment basis. Segments may be fixed form or adaptive, and tests that contain fixed form segments must contain at least one form with a form partition per segment. Every test segment must choose a particular item selection algorithm and provide its configuration parameters. Item selection algorithms are provided to Test Authoring as configuration data, and when configured is offered to the user in a dynamic user interface for selection and configuration.

Level II Requirements: Test Authoring

Adaptive Algorithm Parameters	Test Segment, Segment Blueprints	Adaptive algorithms require tuning parameters for configuration. There are two types of adaptive algorithm parameters: parameters that configure the algorithm globally, and parameters that are configured for each row of a blueprint that corresponds to a content standard. The global configuration parameters are set for each Test Segment, where an adaptive algorithm can be selected for the segment. The Segment Blueprint parameters are defined in the Segment Blueprint screen where adaptive algorithm parameters are set for each row of the Segment Blueprint. Both types of parameters are defined by the specific adaptive algorithm selected for the test segment.
Test Blueprint	Test	Every test must contain a master blueprint. The blueprint is managed using standards information that is provided by a standards publication from the Core Standards component that is associated to a test. Adaptive tests use the test blueprint as a guide to item selection based on the information in the blueprint. Fixed-form tests are also required to have blueprints, primarily for documentation purposes. However, item selection is performed based on the information in the test form and blueprint information is not honored.
Item Pool	Test	Item pools capture the list of items, passages and enemies that are available for presentation on a test. Every test must have a master item pool whether the test is adaptive or segmented. Master Searching for items to be included in an item pool is conducted using the blueprint as a guide. The Test Item Bank is searched for items that correspond to standards on a blueprint, and search criteria is provided to refine item searches for the entire test or for a given standard.
Test Form	Test	Test Forms are the master element for grouping and sequencing items for a test that contains fixed form segments. At a minimum, there must be at least one test form if a test contains a fixed form segment, but in practice, multiple test forms are created for summative tests. A test form is primarily a container element for form partitions, and each test form must have at least one form partition for every fixed form test segment. This means that if a test form is needed at all due to the presence of a fixed form segment, it must contain a minimum of one form partition.
Performance Levels	Test	Performance levels provide a list of levels that refer to a blueprint identifier such as a test, segment, content standard or SOCK. Each performance level in the list has a level identifier and scaled lo and scaled hi values.
Scoring Rules	Test	Scoring rules are created for configuration of a scoring engine for a given assessment. They are composed of a series of computation rules that utilize an individual scoring method performed in a particular sequence. The list of scoring methods and their related parameters are provided to Test Authoring in a configuration file. Each scoring method must correspond to a scoring method programmed into the test scoring engine. The configured scoring methods are provided to the user by means of a dynamic user interface where the user selects from the list of configured scoring methods, and aligns the selected scoring method with a previously created blueprint elements. The dynamic user interface provides for entry of the configured parameters to each scoring rules if any.
Reporting Measures	Test	Reporting measures are a list of individual measures that are aligned to blueprint elements such as a test, segment, content standard or a SOCK. They are provided as a means of configuring a reporting system with a decomposition composed of specific elements along which an overall test score will be scored.
Segment Blueprint	Test Segment	Segment blueprints are a subset of the master test blueprint for a given segment. Every test segment must have a master blueprint. The user first creates a master blueprint, then uses the same screen to partition the master blueprint across test segments.
Segment Pool	Test Segment	Segment pools are a subset of the master test item pools for a given segment. A segment pool is required if the corresponding segment is selected as adaptive. The user first creates a master blueprint, then uses the same screen to partition the item pool for the corresponding adaptive segment.
Segment Form	Test Segment	Segment forms are simply references to specific form partitions that correspond to a given test segment. Segment forms are only required if the segment in question is a fixed form segment. If a test segment is a fixed form test, then a form partition must be selected that corresponds to the test segment for every test form that is part of the test.
Form Partition	Test Form	A form partition is a subset of a test form. There must be one form partition for every test segment. Unlike master test blueprints and test item pools, the master test form does not contain groupings of item references and their sequence. Rather, this functionality is delegated to the form partition

		level.
Item Group	Segment Pool Form Partition	An item group is the building block of segment pools and form partitions. They provide a means of grouping items together and passages together that are selected and administered as an atomic unit. Item groups may or may not be administered in their entirety due to factors such as item inactivation, item filtering due to examinee accommodations, and adaptive algorithm pruning.
Conversion Table	Scoring Rules	<p>A conversion table is a set of value transformations that are aligned with a specific blueprint element. A conversion table is a set of value transformations that are aligned with a specific blueprint element. They belong to specific computation rules that require them. Conversion tables consist of comma-separated text files that are loaded by navigation with a file open dialog from the computation rule. This is done this way because conversion tables can be quite lengthy, so data entry into a user interface is an error-prone process. A computation rule can be configured to require no conversion table, a value conversion table only or value and standard error conversion tables. For computation rules that require both value and standard error conversion tables, Test authoring will ensure that the tables have the same length.</p> <p>Conversion tables consist of two columns: invalue and outvalue. Test Authoring will ensure that the headings exist and are spelled exactly the same way. Invalues are integers and outvalues can be arbitrary strings.</p>

Table 5. Test Specification Domain Elements

Domain Element Sequence Workflow

The following diagram illustrates the required sequence of domain element creation. Certain elements are dependent on others, and this diagram illustrates the dependencies.

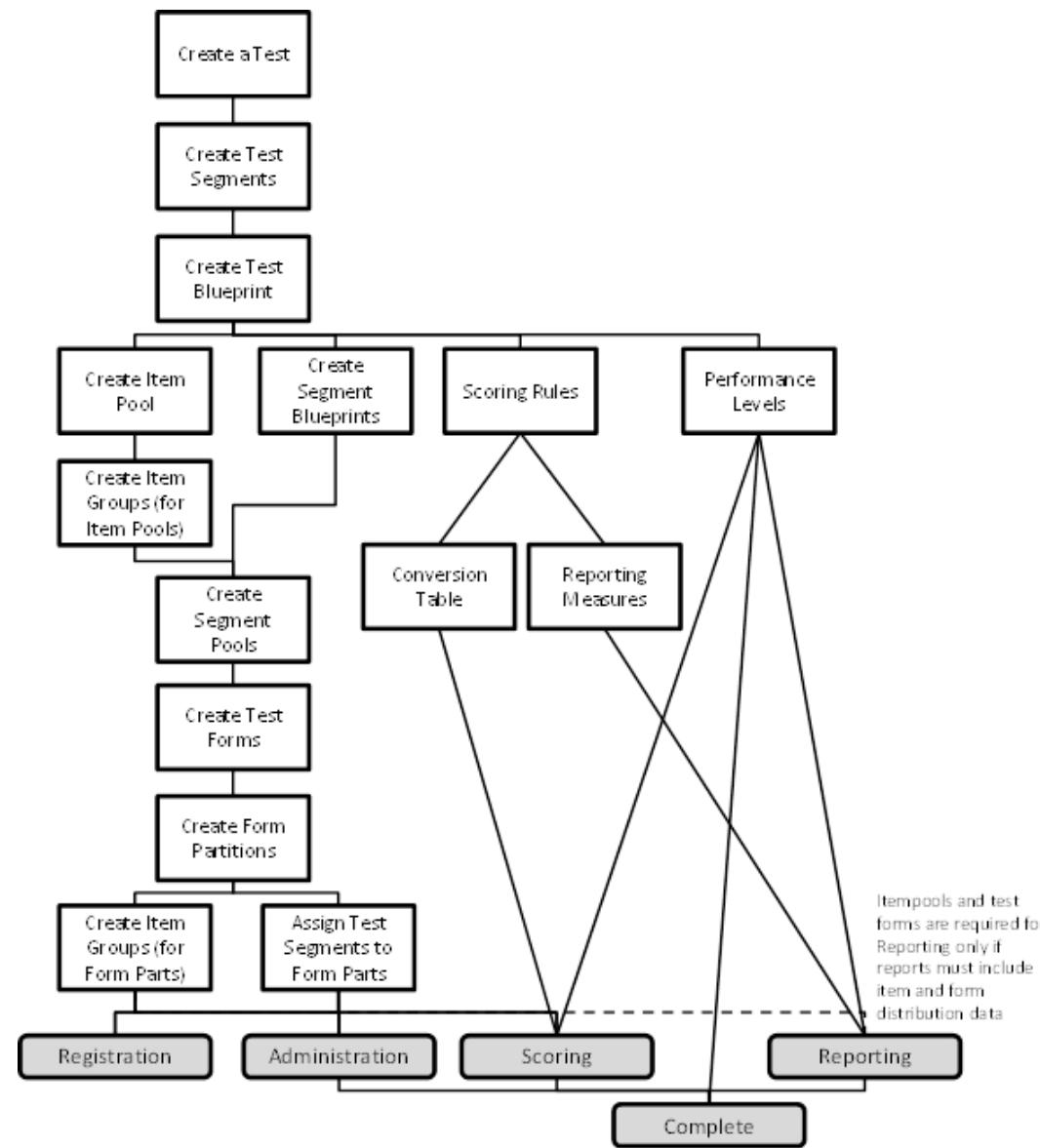


Figure 5 Element Workflow

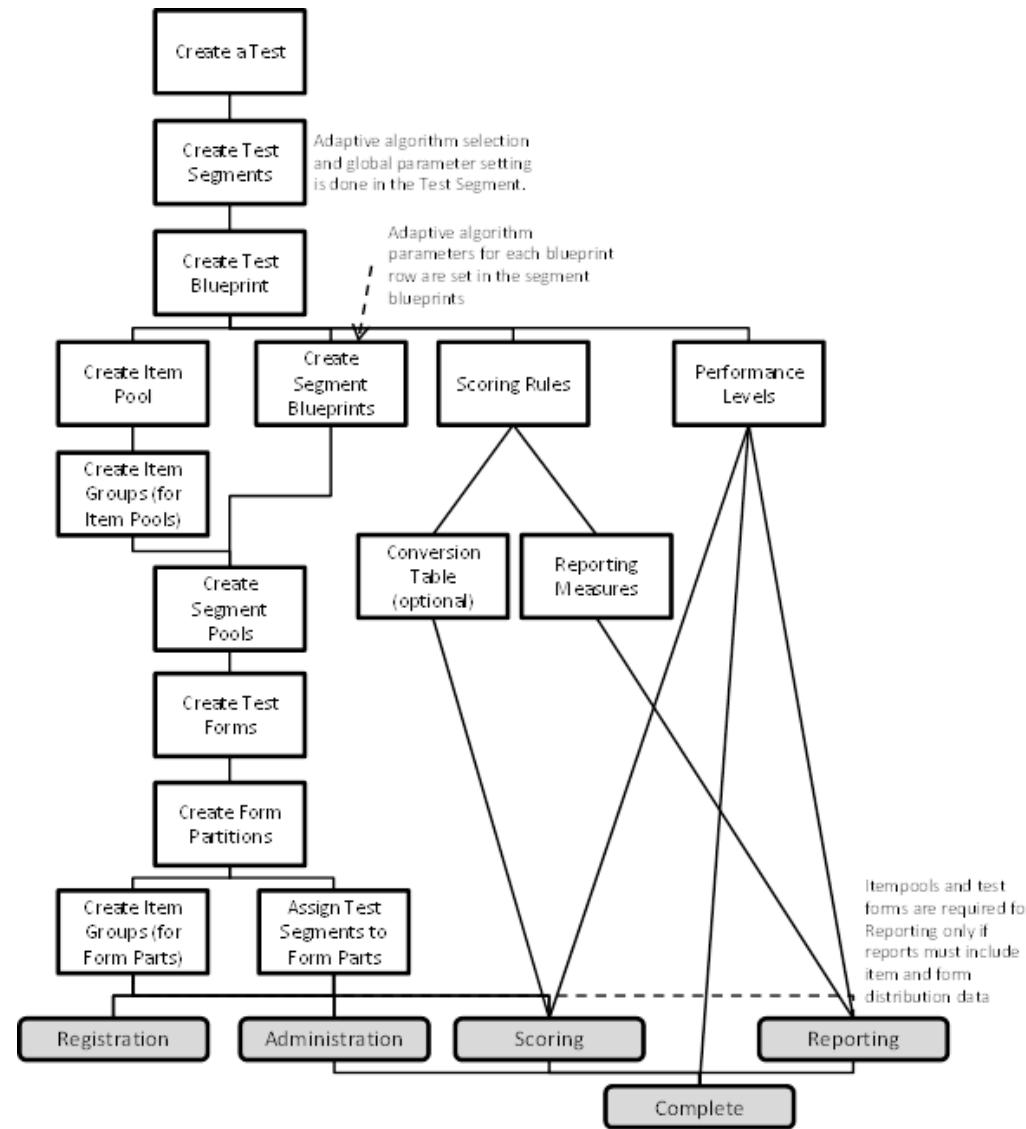


Figure 6. Domain Element Sequence Workflow

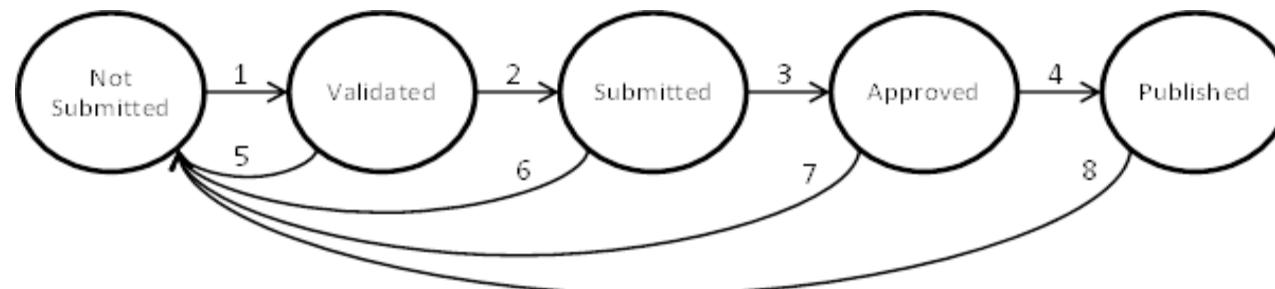


Figure 7 Approval of tests is performed according to a series of states and transitions as indicated below

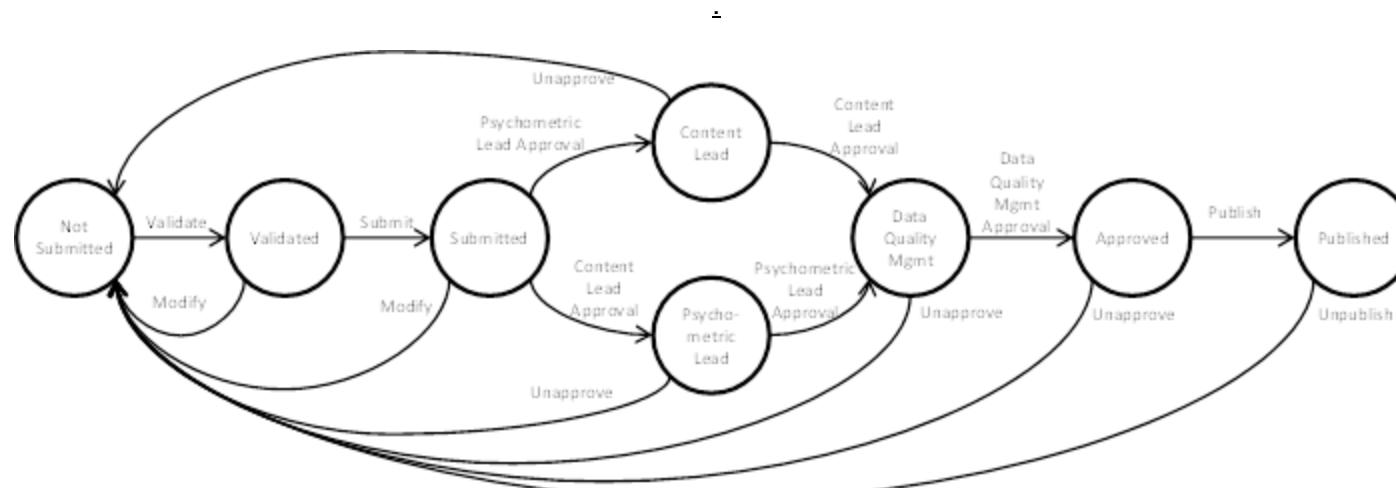


Figure 8. Test Approval State Transition Diagram

The following table details the states indicated in this diagram.

State	Description	Business Rules
Not Submitted	Initial state	Deletion, edits or additions are allowed with no change of state
Validated	Test has been validated and deemed ready for submission for approval for one of the five test specification	Deletion, edits or additions are allowed, but the test will revert automatically to Not Submitted if modified in this state. Some limited edits will not cause the state to revert to Not Submitted. These specific allowable edits will be discovered and documented in the design process.

	purposes:	
Submitted	A test has been submitted for approval for one of the five test specification purposes.	SDeletion, edits or additions are allowed, but the test will revert automatically to Not Submitted if modified in this state. Some limited edits will not cause the state to revert to Not Submitted. These specific allowable edits will be discovered and documented in the design proc
Approved-Content Lead	A test has been approved for one of the five test specification purposes. Once a test is submitted for approval, it can be approved by either the Content Lead or the Psychometric Lead. First approval by the Psychometric Lead places the test in the Content Lead state indicating it is awaiting approval by the Content Lead	Test specifications that have been approved cannot be modified in any substantial way, and are essentially locked down. The test must be manually transitioned back to the Not Submitted state using the Unapprove transition for modification in case a serious error is discovered in a published test package. This state is considered a staging state for publishing of the test.
Psychometric Lead	First approval by the Content Lead places the test in the Psychometric Lead state indicating it is awaiting approval by the Psychometric Lead.	Same as above.
Data Quality Management Lead	Approval by both the Content Lead and the Psychometric Lead places the test in the Data Quality Management state indicating that it is awaiting approval by the Data Quality Management lead.	Same as above.
Approved	A test has been approved by the Content, Psychometric and Data Quality Management Leads.	Same as above. This state is considered a staging state for publishing of the test.
Published	A test specification has been published for a given purpose. The act of publishing is defined as using a test specification in a package and loading it into one of the downstream systems.	Test specifications that have been published cannot be modified in any substantial way, and are essentially locked down. The test must be manually transitioned back to the Not Submitted state for modification in case a serious error is discovered in a published test package.

Table . Test Approval Workflow States

Transition	Source State	Destination State	Business Rules	Authorized Role

Level II Requirements: Test Authoring

Validate	Not Submitted	Validated	<ul style="list-style-type: none"> · The Test authoring application will perform all validation and consistency checks to ensure that all the necessary elements are in place, and are consistent within and between each other · A test must be validated before it can be submitted for approval. Successful validation transitions a test into the Validated state. A test will stay in the Not Submitted state upon unsuccessful validation. · Error and warning messages will be displayed to the user if the validation is unsuccessful that will help the user to make necessary corrections to the test specification elements. · A test can also be submitted for validation at any time with no intent to move it further along in the workflow. For example, a user can perform a validation to determine if there are any missing form partitions. · Validation of a test is not considered to be a high privilege activity in Test Authoring. It provides a convenient means of asking the system to tell the user what remains to be done to complete the test specifications 	Any Test Authoring role
Submit	Validated	Submitted	<ul style="list-style-type: none"> · Any test author role can submit a test for approval 	Any Test Authoring role
Psychometric Lead Approval	Submitted	Content Lead	<ul style="list-style-type: none"> · Only the Psychometric Lead role can perform the Psychometric Lead approval transition 	Psychometric Lead
	Psychometric Lead	Data Quality Management		
Content Lead Approval	Submitted	Psychometric Lead	<ul style="list-style-type: none"> · Only the Content Lead role can perform the Content Lead approval transition 	Content Lead
	Content Lead	Data Quality Management		

Level II Requirements: Test Authoring

Data Quality Management Approval	Data Quality Management	Approved	<ul style="list-style-type: none"> Only the Data Quality Management role can perform the Data Quality Management approval transition 	Data Quality Management Lead
Publish	Approved	Published	<ul style="list-style-type: none"> Publishing a test requires a Test Publisher role as publishing affects external systems 	Test Publisher
Modify	Validated	Not Submitted	<ul style="list-style-type: none"> A test in the Validated state can be modified at any time, so modifying the test by changing, adding or deleting any test specification element will automatically cause the test to be transitioned back to Not Submitted. A test that has been submitted for approval and subsequently modified will have to be validated and resubmitted for approval 	Any Test Authoring role
	Submitted	Not Submitted		
Unapprove	Approved	Not Submitted	<ul style="list-style-type: none"> A state in the Approved state is locked down and cannot be modified in preparation for publishing, so an authorized user must explicitly transition the test Not Submitted via the Unapprove transition. 	Psychometric Lead, Content Lead or Data Quality Management Lead
Unpublish	Publish	Not Submitted	<ul style="list-style-type: none"> A state in the Published state is locked down and cannot be modified, so an authorized user must explicitly transition the test Not Submitted via the Unpublish transition. Unpublishing a test will not change the status of any previously pushed test packages to external systems. The test will have to be transitioned through the above states and a new version must be pushed again to any external systems 	Test Publisher

Table 6. Test Approval Workflow Transitions

Tests must be validated in order to be approved. The validation step will ensure that all the test specification elements are consistent with

themselves and across each other, and that all the necessary test elements are present to satisfy the requirements of creating an Administration and a Registration package. This means that a test can be created without any scoring rules, reporting levels or performance measures since these are not required for an Administration package. However, any test that is created without these elements will not be suitable for creating Reporting, Scoring or Complete packages.

If a test that is approved or published without scoring rules, reporting levels or performance measures, the test will have to be unapproved or unpublished in order to add these elements, then reapproved or republished. If scoring rules, reporting levels or performance measures are present prior to validation, these elements will also be validated for consistency and correctness.

Roles and Permissions

Roles and permissions are managed in the following way by the Smarter Balanced Shared Services.

- Permissions are defined by component functions, and are an atomic unit of access control. Users that have access to a certain component permission are authorized by the component to access the feature or function. Although the Test Authoring permissions are defined by the Test Authoring system and code, the list of permissions for Test Authoring is captured by the Permissions component.
- Roles are not defined by components and are instead managed by the Permissions component.
- Permissions stores a mapping between roles and component permissions.
- Test Authoring is responsible for determining from the Permissions component the roles that matter to Test Authoring, and how each of those roles map to established permissions in Test Authoring
- When a user is authenticated, Test Authoring determines which of the user's roles are applicable, and from the role, the features that this individual user is authorized to access.

The result of this is that the primary task of Test Authoring is to specify on a set of permissions that protects each key feature of the application. These permissions can be mapped to a recommended set of roles, and the roles are in turn negotiated with Smarter Balanced. By judicious design of individual permissions, other individuals downstream are able to fine-tune the user roles and decide what roles are able to access what permission.

The Test Authoring permissions will grant access to screens that correspond to the test specification entities separately for CRUD operations. This means that user roles with:

- Create/Delete permissions will be able to create new tests,
- Read permissions is able to view test data but not create, modify or delete, and

- Update is able to modify existing tests , but not create or delete them.

Delete is combined with Create because they are considered to be equal but higher level activities than update, which is in turn higher than read. Read implies read only. The User Interface Summary matrix below will include a column for permissions for each screen.

Deployment

The Test Spec Bank is deployed as part of an Assessment Creation and Management group. At a minimum, the Test Authoring, Test Spec Bank, Test Item Bank and Test Authoring components must be deployed together. Item Authoring and Item bank can be optionally deployed with other remaining Assessment Creation and Management components.

Client Components

The Test Authoring component provides the user interface for the interactive creation and editing of test specifications. It utilizes the interfaces exposed by the Test Spec Bank for direct access to the persistent store to create, query, store and retrieve test specifications and workflow information.

Assumptions

	Assumption	Comments

Requirements are numbered according to the following convention:

- 1) RFP.## - a requirement from the RFP
- 2) RADTIB.## - a requirement from the detailed requirements from RFP-11 or the Architecture document
- 3) PRTIB.## - a requirement from the Proposal

Requirements

Source.I D	Requirement	Category	Priority	Comments
RFP 6	Test Authoring Component meets requirements detailed in the SBAC IT Systems Architecture (Appendix C)	Data Format		

Level II Requirements: Test Authoring

RADTIB 1.0	The Test authoring application will perform all validation and consistency checks to ensure that all the necessary elements are in place, and are consistent within and between each other	Validate		
RADTIB 1.1	A test must be validated before it can be submitted for approval. Successful validation transitions a test into the Validated state. A test will stay in the Not Submitted state upon unsuccessful validation	Validate		A test can also be submitted for validation at any time with no intent to move it further along in the workflow. For example, a user can perform a validation to determine if there are any missing form partitions.
RADTIB 1.2	Error and warning messages will be displayed to the user if the validation is unsuccessful that will help the user to make necessary corrections to the test specification elements.	Validate		Validation of a test is not considered to be a high privilege activity in Test Authoring. It provides a convenient means of asking the system to tell the user what remains to be done to complete the test specifications
RADTIB 1.3	A test in the Validated state can be modified at any time, so modifying the test by changing, adding or deleting any test specification element will automatically cause the test to be transitioned back to Not Submitted.	Validate		
RADTIB 2.0	Submission of a validated test for approval requires higher level user privileges than validation.	Submit		
RADTIB 2.1	Approval of a test requires higher level user privileges than submission for approval.	Submit		

Level II Requirements: Test Authoring

RADTIB 2.2	The user that approves the test must be a different user than the one that submitted the test regardless of the level of privilege	Submit		
RADTIB 2.3	Publishing a test requires a high level of privilege as publishing affects external systems	Submit		
RADTIB 2.4	A test that has been submitted for approval and subsequently will have to be validated and resubmitted for approval by an authorized user	Submit		
RADTIB 3.0	A state in the Approved state is locked down and cannot be modified in preparation for publishing, so an authorized user must explicitly transition the test Not Submitted via the Unapprove transition.	Unapprove		
RADTIB 4.0	A state in the Published state is locked down and cannot be modified, so an authorized user must explicitly transition the test Not Submitted via the Unpublish transition.	Unpublish		
RADTIB 4.1	Unpublishing a test will not change the status of any previously pushed test packages to external systems. The test will have to be transitioned through the above states and a new version must be pushed again to any external systems	Unpublish		

RADTIB 5.0	Create a version of the item selection algorithm that is provided as major.minor version string. It is acceptable to have multiple item selection algorithms with the same name so long as the versions are different. This is done to accommodate new versions of a item selection algorithm and maintain compatibility with older versions for an existing test.	Algorithm		Item selection algorithms in Test Authoring are composed of an item selection algorithm name and a variable collection of parameters. New item selection algorithms are added to Test Authoring by means of external configuration.
RADTIB 5.1	label	string		A string that describes the name of the item selection algorithm. Labels are offered to the user in a dropdown for selection.
RADTIB 5.2	version	string		v
RADTIB 5.3	number of parameters	int		Item selection algorithms have a variable number of parameters, and this field supplies the expected number of parameters
RADTIB 5.4	identifier	UUID		A UUID used to match the item selection algorithm in Test Authoring with the corresponding module in Test Delivery
RADTIB 6.0	Item selection algorithms have parameters, but this table only indicates the number of parameters Create The following parts of an item selection rule parameter.			

RADTIB 6.1	label	string		This is the name of the parameter. This will be used for labels for data entry fields in the configurable user interface
RADTIB 6.2	type	scalar blueprint		The item selection parameter may be a scalar configuration value or may be a value that is required for each blueprint element,
RADTIB 6.3	position	int		This is the sequence of presentation of the item selection rule parameter in the configurable user interface
RADTIB 6.4	type	Integer float boolean		The parameter can be a scalar or a dictionary. The allowable values of scalars are integer, float and string. The parameter value can also be a dictionary. Dictionaries are not of predetermined length, and allow the user to add an arbitrary number of rows. This means that the user interfaces have to be sufficiently flexible to allow for entry of new dictionary rows which consist of an index value and a parameter value. The user interface should validate the type on data entry.
RADTIB 6.5	default	<value>		The default value that the parameter should be given when first selected by the user in the user interface
RADTIB 6.6	minimum	<value>		The minimum value the parameter should have (optional), this is validated by the user interface
RADTIB 6.7	maximum			The maximum value the parameter should have (optional), this is validated by the user interface

RADTIB 6.8	identifier	UUID		A UUID used to match the item selection algorithm parameter in Test Authoring with the corresponding parameter in Test Delivery
RADTIB 7.0	Create Scoring rules in Test Authoring that are composed of a sequential set of computation rules.			
RADTIB 7.1	name	string		A string that describes the name of the computation rule. The list of computation rule names is offered to the user in a dropdown for selection. The combination of the name and the version should be unique.
RADTIB 7.2	version	string		The version of the computation rule is provided in as major.minor version string. It is acceptable to have multiple computation rules with the same name so long as the versions are different. This is done to accommodate new versions of a computation rule and maintain compatibility with older versions for an existing test.
RADTIB 7.3	number of parameters	int		Computation rules have a variable number of parameters, and this field supplies the expected number of parameters
RADTIB 7.4	identifier	UUID		A UUID used to match the scoring rule in Test Authoring with the corresponding module in Test Delivery
RADTIB 8.0	Create computation rule parameters.			
RADTIB 8.1	name	string		This is the name of the parameter. This will be used for labels for data entry fields in

				the configurable user interface.
RADTIB 8.2	position	int		This is the sequence of presentation of the computation rule parameter in the configurable user interface
RADTIB 8.3	parametertype	Integer float string integerdictionary floatdictionary stringdictionary		The parameter can be a scalar or a dictionary. The allowable values of scalars are integer, float and string. The parameter value can also be a dictionary. Dictionaries are not of predetermined length, and allow the user to add an arbitrary number of rows. This means that the user interfaces have to be sufficiently flexible to allow for entry of new dictionary rows which consist of an index value and a parameter value. The user interface should validate the type on data entry.
RADTIB 8.4	indextype	Integer, float, string		If the parameter is a dictionary, this indicates the index type. The user interface should validate the type on data entry.
RADTIB 8.5	identifier	UUID		A UUID used to match the item selection algorithm parameter in Test Authoring with the corresponding parameter in Test Delivery
RADTIB 9.0	The Test Authoring application will use the Smarter Balanced SSO component for user authentication and authorization.			This requirement documents the use of Shared Services component by Test Authoring.
RADTIB 10	The Test Authoring application will use the Smarter Balanced Permissions component to obtain a list of roles that have permissions on Test Authoring and the mapping between these roles and Test	Architecture		This requirement documents the use of Shared Services component by Test Authoring.

Level II Requirements: Test Authoring

	Authoring permissions.			
RADTIB 11	The Test Authoring application will use the Smarter Balanced Program Management component as a source of configuration management data to locate shared services and other components it depends on.	Architecture		This requirement documents the use of Shared Services component by Test Authoring.
RADTIB 11.1	The Test Authoring application will use the Smarter Balanced Program Management component to provide configuration management data that other components will use to interface with Test Authoring.	Architecture		This requirement documents the use of Shared Services component by Test Authoring.
RADTIB 12	The Test Authoring application will use the Smarter Balanced Monitoring and Alerting component to issue alerts and notifications.	Architecture		This requirement documents the use of Shared Services component by Test Authoring.
RADTIB 13	The Test Authoring application will use the Smarter Balanced Core Standards application as a source of standards information.	Architecture		This requirement documents the use of Shared Services component by Test Authoring.
RADTIB 14	The Test Authoring application will use the Smarter Balanced Test Item Bank component as a source of item information.	Architecture		This requirement documents the use of Shared Services component by Test Authoring.
RADTIB 15	The Test Authoring application will use the Smarter Balanced Test Spec Bank component to store and retrieve Test Specification information.	Architecture		This requirement documents the use of Shared Services component by Test Authoring.
RADTIB 16	The Test Authoring application shall be multi-tenant.	User Roles and Tenancy		Tenants are configured in Program Management, and particular Test Authoring users are associated with a

Level II Requirements: Test Authoring

				given tenant.
RADTIB 16.1	Test data and dependent elements shall be maintained separately for each tenant.	User Roles and Tenancy		Users will only be able to see data that is relevant to their specific tenancy and will not be able to see other tenant's test and dependent elements.
RADTIB 16.2	Tests and test elements created and managed by a given user are done so on behalf of the tenant identified by the entity associated by the user's Test Authoring role.	User Roles and Tenancy		All system roles are associated to an entity. Users logging on to Test Authoring must have a valid Test Authoring role that is associated to an entity that is a valid tenant.
RADTIB 16.3	Users that do not have a valid Test Authoring role will not be given access to the Test Authoring application.	User Roles and Tenancy		The Permissions component maps user roles to permissions that belong to a component. Only users with roles that are mapped to at least one Test Authoring role will be given access.
RADTIB 16.4	Users with a valid Test Authoring role that are not associated to a valid tenant will not be given access to the Test Authoring application.	User Roles and Tenancy		A user may have a valid Test Authoring role, but if their valid Test Authoring role is not associated with a valid tenant, then the user will not be given access.
RADTIB 16.5	Users with more than one valid Test Authoring role associated with more than one valid tenant will be given access to one and only one tenant at a time.	User Roles and Tenancy		If a user has valid test authoring roles for more than one tenant, Test Authoring must resolve the discrepancy. If it is unable to do so, the user must be presented with a choice and must select to represent one and only one tenant at a time.
RADTIB 16.6	The Test Authoring application will maintain a Setup user interface for creation and management of subject metadata and management of publication standards associations that will not be multi-tenant.	User Roles and Tenancy		Creation of subjects and subsequent association to various tenants requires that the Setup UI see across tenants.
RADTIB 17	The Authoring application shall create permissions that shall be	Roles and		A permission is a unit of access that is granted to a user, and is the most granular

	used for fine-grained authorization.	Permissions		level of access provided to a user. The granularity of permissions must be determined by the system designer.
RADTIB 17.1	Users shall be granted access to Test Authoring features and based on the permissions the user's role is has authorized to access to.	Roles and Permissions		All user roles are mapped to component permissions by the Permissions component, and this information is available to components through the Permissions API. Test Authoring must determine the specific user roles that are mapped to Test Authoring permissions, and when a user is authenticated, Test Authoring provides access to only those features based on the permissions that the user's role is allowed to access.
RADTIB 18	Elements originating internally to Test Authoring such as tests, segments or forms will use a unique identifier assigned by the Test Authoring application.	Identification		All elements must be uniquely identified. If the element is originated and managed by Test Authoring, it will also manage identification.
RADTIB 18.1	External elements such as content standards, items and passages that already provide unique identifiers will use the existing external unique identifier to identify the element.	Identifier		Elements that already have unique identifiers should be referred to by those identifiers.
RADTIB 19	Test specification elements at all levels will be versioned using a major.minor version scheme.	Versioning		What constitutes major and minor version changes will depend on the specific entity. Additional requirements will be added in the detailed requirements process.
RADTIB 20	The Test Authoring application will maintain Subject metadata for each tenant via a user interface that is not multi-tenant.	Subject Metadata		The list of subjects to choose from to create tests will be tenant-specific
RADTIB 20.1	Subject metadata shall include a full subject name as well as a three-character short code.	Subject Metadata		The short code is used for various purposes such as concatenation for a test name and for data exchange with other applications such as Test Delivery.

Level II Requirements: Test Authoring

RADTIB 20.2	Both the full subject name and three-character subject code shall be emitted in the test package XML.	Subject Metadata		The short code is needed by downstream systems.
RADTIB 21	Specific standards publications from Core Standards will be associated with a tenants and a subject specific to that tenant.	Publication Standards		When selecting a standards publication for a test, a user will only be able to select standards publications that have been previously associated with the test subject for their specific tenancy. This makes it possible to choose different standards publications for the same subject for different tenants. For example, California users wanting to create a Biology test will want to use California science standards, but North Carolina users will want to choose North Carolina Biology standards.
RADTIB 22.1	Item selection algorithms shall be configurable in Test Authoring so that new item selection algorithms and their parameters can be added at any time.	Item Selection Algorithms		Test Authoring should accommodate new Item selection algorithms by means of configuration.
RADTIB 22.2	Configured versions of item selection algorithms shall have a major.minor version number.	Item Selection Algorithms		This allows us to keep old versions of item selection algorithms for compatibility while making changes in new versions
RADTIB 22.2	The Test Segment user interface shall present configured item selection algorithms for selection by the Test Author, and shall provide a dynamic user interface for entering configuration parameters specific to the selected algorithm.	Item Selection Algorithms		The user interface will respond to new item selection algorithms by presenting them as a choice and providing facilities to enter their customized parameters.
RADTIB 22.3	Test Authoring shall provide a setup function that analyzes the external inputs for configuration of item selection algorithms and provide users with appropriate error messages if something is wrong	Item Selection Algorithms		If the configuration for Item Selection algorithms is provided via an external XML file, the file will be checked for correctness and conformance to a DTD or XSD. Any error messages based on this analysis is presented to the user so they

	with the external configuration.			can make appropriate modifications.
RADTIB 22.4	The system will support a minimum of three item selection algorithms: fixedform, field test and adaptive	Item Selection Algorithms		Test Authoring will support these three item selection algorithms by default. Other item selection algorithms can be added after Test Authoring is completed by means of the extensibility mechanism described in the above requirements.
RADTIB 23	Scoring functions shall be configurable in Test Authoring so that new scoring functions and their parameters can be added at any time.	Scoring Functions		Test Authoring should accommodate new scoring functions by means of configuration.
RADTIB 23.1	Configured versions of scoring functions shall have a major.minor version number.	Scoring Functions		This allows us to keep old versions of scoring functions for compatibility while making changes in new versions
RADTIB 23.2	The Scoring Rules user interface shall present configured scoring functions for selection by the Test Author, and shall provide a dynamic user interface for entering configuration parameters specific to the selected function.	Scoring Functions		The user interface will respond to new scoring functions algorithms by presenting them as a choice and providing facilities to enter their customized parameters.
RADTIB 23.3	Test Authoring shall provide a setup function that analyzes the external inputs for configuration of scoring functions and provide users with appropriate error messages if something is wrong with the external configuration.	Scoring Functions		If the configuration for scoring functions is provided via an external XML file, the file will be checked for correctness and conformance to a DTD or XSD. Any error messages based on this analysis is presented to the user so they can make appropriate modifications.
RADTIB 24	The Test Authoring application permits authorized users to create and manage Test entities.	Tests		The Test entity is the highest level domain entity. Deleting a Test will also delete all lower level entities, so access to test elements should be carefully controlled.

Level II Requirements: Test Authoring

RADTIB 24.1	The Test Authoring application permits authorized users to create and manage Test Segment entities.	Test Segment		This requirement documents the need for the creation and management of test segments.
RADTIB 24.2	The Test Authoring application permits authorized users to create and manage Test Blueprint entities.	Test Blueprint		This requirement documents the need for the creation and management of
RADTIB 24.3	If a non-leaf node standard is made inactive and its children are not made inactive, a message shall be displayed on each active child node: "Parent Inactive"	Test Blueprint Validation		If a standard is inactive, all of its child standards must also be inactive. However, an active standard can have inactive children.
RADTIB 25	The Test Authoring application permits authorized users to create and manage Item Pool entities.	Item Pool		This requirement documents the need for the creation and management of Item Pools.
RADTIB 26	The Test Authoring application permits authorized users to create and manage Test Form entities.	Test Form		This requirement documents the need for the creation and management of Test Forms.
RADTIB 27	The Test Authoring application permits authorized users to create and manage Performance Levels entities.	Performance Levels		This requirement documents the need for the creation and management of Performance Levels.
RADTIB 27.1	As a user I would like to record performance level information so I can include it in a test package	Performance Levels		TESTAUTHOR-424
RADTIB 28	The Test Authoring application permits authorized users to create and manage Scoring Rules entities.	Scoring Rules		This requirement documents the need for the creation and management of Scoring Rules.
RADTIB 29	The Test Authoring application permits authorized users to create and manage Reporting Measures entities.	Reporting Measures		This requirement documents the need for the creation and management of Reporting Measures.

Level II Requirements: Test Authoring

RADTIB 30	The Test Authoring application permits authorized users to create and manage Segment Blueprint entities.	Segment Blueprint		This requirement documents the need for the creation and management of Segment Blueprints.
RADTIB 31	The Test Authoring application permits authorized users to create and manage Segment Pool entities.	Segment Pool		This requirement documents the need for the creation and management of Segment Pools.
RADTIB 32	The Test Authoring application permits authorized users to create and manage Segment Form entities.	Segment Form		This requirement documents the need for the creation and management of Segment Forms.
RADTIB 33	The Test Authoring application permits authorized users to create and manage Form Partition entities.	Form Partition		This requirement documents the need for the creation and management of Form Partitions.
RADTIB 34	The Test Authoring application permits authorized users to create and manage Item Group entities.	Item Group		This requirement documents the need for the creation and management of Item Groups.
RADTIB 34.1	items to be associated with item groups			<p>Implicit associations</p> <p>i. Reference to same passage? As far as we can tell, the only thing that causes items to be grouped into an item group are they mutual association to the same passage. This means that Test Authoring should group items together into an item group if they refer to the same</p>

				passage. The passage reference should be part of the item metadata. Items that have no passage references or unique passage references are automatically grouped by Test Authoring into an itemgroup.
RADTIB 35	The Test Authoring application permits authorized users to create and manage Conversion Table entities.	Conversion Table		This requirement documents the need for the creation and management of Conversion Tables.
RADTIB 30	The Test Authoring application will maintain the workflow state (Not Submitted, Validated, Submitted, Approved and Published) for the following: Registration, Administration, Scoring, and Reporting.	Workflow		Each of these levels requires independent submission and approval. The levels are telection o be considered independent and not sequential.
RADTIB 30.1	The Test Authoring application will provide the means to perform the necessary validations upon user demand that will check whether a test is ready for promotion to the Registration, Administration, Scoring and Reporting states, and provides appropriate error messages and detailed information so the user can perform necessary corrections.	Workflow		This validation feature checks whether a test is ready for promotion from Not Submitted to Submitted.
RADTIB 30.2	The Test Authoring application will not permit a test to be submitted for approval that has not been validated and deemed ready for submission.	Workflow		The user must pass all validations for submitting a test for approval for one of the test purposes managed by Test authoring. A test may be deemed ready for submission for approval for Administration and Reporting and not for Administration and Scoring. This depends

Level II Requirements: Test Authoring

				on the completeness and consistency of the set of required elements for promotion to Approved for that purpose.
RADTIB 30.3	The user that submits a test for approval for a given purpose cannot be the same user that approves the test for promotion from the Submitted to Approved state.	Workflow		Regardless of the role of the user, the submitter and approver must be two different users.
RADTIB 31	Deleted domain entities will be copied to a separate space for deleted entities for possible recovery	Data Retention		This is especially critical if a high-level entity is deleted from the system. This will cause cascading deletion of all dependent entities.
RADTIB 31.1	Modified domain entity data shall be copied to history tables or collections to ensure an audit history of changes are maintained and shall include the user that made the change and a time/date stamp.	Data Retention		This is necessary to track who changed test data over time.
RADTIB 32	TBD	Test Packaging		Test Packaging requirements will be added in a subsequent revision of this document.
RADTIB 33	Test Authoring requires a considerable number of consistency checks and validations to ensure that all test specification elements are complete within themselves and consistent across each other. This section captures these validations in the form of additional requirements	Test Segments		
RADTIB 34	For every test in the system, a minimum of one test segment is required	Blueprint		Tests with "no" test segments are actually a single-segment test. This maintains consistency in the software and does not require any special cases. More than one

				test segment is certainly allowed, and the system will not impose a limit to the number of test segments because practical limits to this will typically prevail
RADTIB 34.1	The Test Blueprint screens shall implement adequate validation to ensure that the indicated mins and maxes are consistent throughout the standards hierarchy.	Blueprint		The values entered for blueprint screen mins and maxes must follow certain rules related
RADTIB 34.2	The operational item min value for every content standard must be less than or equal to the operational item max value for each standard Display message: "Min must be ≤ Max"	Blueprint		If this constraint is violated, display a message.
RADTIB 34.3	The sum of the operational item min values of a higher-level content standard's active children must be less than or equal to its operational item max value. Display message: "The max value of the content standard (N) is less than the sum if its children's min value (M)"	Blueprint		If a parent standard has a max of N items, its children cannot have a total min value M greater than N. Substitute actual values for N and M in the message.
RADTIB 34.4	The sum of the operational item max values of a higher-level content standard's active children must be greater than or equal to its operational item min value. Display message: "The max value of the content standard (N) is smaller than the sum if its children's min value (M)"	Blueprint		If a parent standard has a min of N items, its children cannot have a total max value M less than N. Substitute actual values for M and N in the message.
RADTIB	The field test item min value for every content standard must be less	Blueprint		If this constraint is violated, display a

Level II Requirements: Test Authoring

34.5	than or equal to the field test item max value for each standard Display message: "Min must be ≤ Max"			message.
RADTIB 34.6	The sum of the field test item min values of a higher-level content standard's active children must be less than or equal to its field test item max value. Display message: "The max value of the content standard (N) is less than the sum if its children's min value (M)"	Blueprint		If a parent standard has a max of N items, its children cannot have a total min value M greater than N. Substitute actual values for N and M in the message.
RADTIB 34.7	The sum of the field test item max values of a higher-level content standard's active children must be greater than or equal to its field test item min value. Display message: "The max value of the content standard (N) is smaller than the sum if its children's min value (M)"	Blueprint		If a parent standard has a min of N items, its children cannot have a total max value M less than N. Substitute actual values for M and N in the message.
RADTIB 35	If a standard is made inactive, a message must be displayed that indicates that the standard is inactive.	Test Forms		A visual cue is required so a user is clear that a standard has been made inactive.
RADTIB 36	A test form is required if any test segment has an item selection algorithm choice of fixedform	Form Partitions		Test forms are specifically for fixed form tests. If any test segment is marked as fixedform, at least one test form with at least one form partition is required
RADTIB 36.1	A form partition is required for every test segment with an item selection algorithm choice of fixedform			If there are N test segments marked with an item selection choice of with an item selection algorithm choice of fixedform, then N form partitions are required

RADTIB 36.2	This screen allows the creation of form partitions that belong to a form. Filters are provided to narrow the number of forms under which to create a partition.	Form Partitions		<ul style="list-style-type: none"> A form requires as many form partitions as there are fixed form segments
RADTIB 36.3	Edit Form Partitions allows the user to use the search screen to search the existing item pool for items to place on the form. Items can be sequenced and deleted as well.	Edit Form Partitions		<ul style="list-style-type: none"> Item searches are conducted against the item pool, not against the test item bank.
RADTIB 36.4	This screen allows the assignment of test form partitions to test segments.	Assign Form Partitions to Test Segments		<ul style="list-style-type: none"> Only fixed form test segments participate in form partition assignment.
RADTIB 36.5	This screen allows the user to build scoring rules by selecting computation rules and entering the parameters of the computation rule. The computation rules are configured into the Test Authoring system and the user interface presents the configured parameters dynamically.	Scoring Rules		<ul style="list-style-type: none"> The user is offered a preconfigured set of computation rules to choose from. Once created, computation rules can be moved up and down as the sequence of rules is critical. The user interface for entering the computation rule parameters is dynamic based on parameter configuration. Existing blueprint elements are referred to.
RADTIB 37	Roles and permissions should be managed in the following way by the Smarter Balanced Shared Services			Permissions are defined by component functions, and are an atomic unit of access control. Users that have access to a certain component permission are authorized by the component to access the feature or function. Although the Test Authoring permissions are defined by the Test

Level II Requirements: Test Authoring

				Authoring system and code, the list of permissions for Test Authoring is captured by the Permissions component.
RADTIB 37.1	Roles are not defined by components and are instead managed by the Permissions component			
RADTIB 37.2	Test Authoring is responsible for determining from the Permissions component the roles that matter to Test Authoring, and how each of those roles map to established permissions in Test Authoring			
RADTIB 37.3	When a user is authenticated, Test Authoring determines which of the user's roles are applicable, and from the role, the features that this individual user is authorized to access.			
RADTIB 38	The Test Authoring permissions will grant access to screens that correspond to the test specification entities separately for CRUD operations. This means that user roles with:			Delete is combined with Create because they are considered to be equal but higher level activities than update, which is in turn higher than read. Read implies read only. The User Interface Summary matrix below will include a column for permissions

				for each screen.
RADTIB 37.1	Create/Delete permissions will be able to create new tests,			
RADTIB 37.2	Read permissions is able to view test data but not create, modify or delete, and			
RADTIB 37.3	Update is able to modify existing tests , but not create or delete them.			
RADTIB 38	The following table summarizes the user interface screens in the Test Authoring application.			
RADTIB 38.1	This screen provides home screen navigation access to the various Test Authoring functions	Home		Individual users will be able to access only the screens that correspond to the permissions they are authorized to access. Only the features that the user is permitted to access are available and all others are disabled.
RADTIB 38.2	This screen provides navigation access to the Setup functions	Setup		Setup is an administrative function that few users should be able to access.
RADTIB	Screens to CRUD subject metadata are required. The list of publishers	Subjects		<ul style="list-style-type: none"> A subject cannot be deleted once a test using this publisher has been

38.3	offered come from here.			<p>created</p> <ul style="list-style-type: none"> A subject cannot be modified once a test using this publisher has been published
RADTIB 38.4	<p>Standards publications from Core Standards are browsed and associated with a publisher and a subject.</p> <ul style="list-style-type: none"> Show a grid of publications with the publication name, description and subject Provide a dropdown with every publisher that has a publication in the repository. Allow the user to select a publisher to filter the publications in the grid by publisher. Default is "Show all publishers" Show a dropdown of subjects from the subject metadata to select from Show a dropdown of publishers from the publisher metadata to select from Confirm association of a standards publication to a publisher and a subject. Allow disassociation of a publication from a subject/publisher 	Standards Publications		<ul style="list-style-type: none"> Once a test has been created with a given subject, publisher and standards publication, the publication/publisher/subject association cannot be disassociated Cardinality isn't a requirement, this is an item issue Required levels are primarily for items The required levels are needed only on the BP screen itself
RADTIB 38.5	<p>An XML configuration file for item selection algorithms is loaded on demand</p> <ul style="list-style-type: none"> The configuration file is analyzed and issues problems are reported to the user If the configuration file has no 	Item Selection Algorithms		<ul style="list-style-type: none"> Changes cannot be accepted unless the configuration file integrity check reveals no errors Changes to existing item selection configuration that changes an item selection algorithm already selected into a test are not allowed

	errors, the user can accept the changes into the Test Authoring configuration			
RADTIB 38.6	An XML configuration file for scoring functions is loaded on demand <ul style="list-style-type: none"> • The configuration file is analyzed and issues problems are reported to the user • If the configuration file has no errors, the user can accept the changes into the Test Authoring configuration 	Scoring Functions		<ul style="list-style-type: none"> • Changes cannot be accepted unless the configuration file integrity check reveals no errors • Changes to existing scoring function configuration that changes a scoring function already selected into a test are not allowed
RADTIB 38.7	<ul style="list-style-type: none"> • Show all tests in a grid • Provide dropdowns to filter by publisher, subject and grade • Provide a link for users to select a test to edit • Provide a means for users to delete a test • Provide the means for users to create a new test 	Tests		<ul style="list-style-type: none"> • A test cannot be deleted once it has been published • Deleting a test destroys all child elements
RADTIB 38.8	<ul style="list-style-type: none"> • Show all test segments in a grid • Provide dropdowns to filter test segments by publisher, subject, grade and test • The test dropdown is itself filtered by publisher, subject and grade • Provide a link for users to select a test segment to edit • Provide a means for users to delete segment a test • Provide the means for users to create a new test 	Test Segments		<ul style="list-style-type: none"> • A test segment cannot be deleted once it has been published • A test segment may be added, but it changes the versioning

RADTIB 38.9	<ul style="list-style-type: none"> • Show all test blueprints in a grid • Provide dropdowns to filter test segments by publisher, subject, grade, test and test segment • The test and test segment dropdowns are themselves filtered by publisher, subject and grade • Provide a link for users to select a test segment to edit • Provide a means for users to delete segment a test • Provide the means for users to create a new test 	Blueprints		<p>Detailed requirements in section 9 <u>Additional Functional Requirements</u> and in section 10 <u>Consistency Checks and Validations</u>. See Test Blueprint Validations for detailed business rules for this screen.</p>
RADTIB 38.10	<p>This screen is organized by the same standards that are selected for the blueprint.</p> <ul style="list-style-type: none"> • Blueprint information is shown in a read-only manner • Total operational and field test items in the pool • Switch between Standards and SOCKs view • View total number of operational and field test items • Switch between a master test item pool or a segment item pool • Summary of item searches are shown as item operational and field test item counts (counts roll up hierarchically) • Item summary counts for operational and field test and for test and segment (when viewing segment pool) • Edit links take you to an Edit Item Pools screen to the means to view items individually (all items, or items for a particular content 	Item Pools		<ul style="list-style-type: none"> • The Test Item Bank is searched for items that correspond to the content standards selected for the blueprint • New searches may repopulate standards with items previously pruned from the item pool • Searches can be conducted for any all content standards or for any individual content standard at any level in the hierarchy • Searching for items at a content standard level higher in the hierarchy than the lowest level will include all child standards as well

	<p>standard and their children if any)</p> <ul style="list-style-type: none"> ● Search links allow the user to search (or search again) for items for all standards or for a particular content standard and their children if any) ● Searches are conducted for both operational and field test items 			
RADTIB 38.11	<p>This screen shows individual items (not just item summaries) organized by the same standards selected for the blueprint</p> <ul style="list-style-type: none"> ● Toggle between viewing items and passages ● Filter items by an individual content standard (and its children) or see items for all standards ● Display of total number of items in pool, filter and segment (if editing a segment pool) ● Switch between master test pool and segment pools ● Delete individual items from the master pool ● See more details about an item 	Edit Item Pools		<ul style="list-style-type: none"> ● Cannot change OP/FT, Required or Active status of an item from the segment pool view ● Cannot delete an item from the segment pool view, only deselect it from the pool ● Cannot change passage status or delete a passage: passages are pulled in by item references
RADTIB 38.12	This screen allows the creation of test forms. Forms are simple container elements that contain form partitions.	Forms		<ul style="list-style-type: none"> ● A test with at least one fixed form segment requires at least one form
RADTIB 38.13	This is pending as more information is required.	Conversion Tables		

RADTIB 38.14	Performance levels are a sequential set of level cutters. They refer to a blueprint element and permit the user to enter a low and high value.	Performance Levels		Existing blueprint elements are referred to.
RADTIB 38.15	This screen allows the user to create reporting measures, which are a collection of references to existing computation rules from Scoring Rules. The user adds computation rules to a reporting measure until they are satisfied.	Reporting Measures		<ul style="list-style-type: none"> • Existing blueprint elements are referred to • Existing computation rules are referred to
RADTIB 38.16	The Workflow screens are left to DRC engineers to design	Workflow		
RADTIB 39	The selection of active standards on a test blueprint is a prerequisite for selection of an item pool because the item pool screen searches items according to their alignment with active standards.			The Item Pool screen has a selector to view the master test item pool, or to choose the test's child segments to view and edit the segment pool. This feature is similar to the blueprint screens. At the segment level, no items can be added to the master pool, items deleted from the segment pool are not deleted from the master item pool. However, items removed from the master item pool are also deleted from the segment item pool. Invalidation of items follows a similar pattern: it can only be done in the master item pool and not in the segment pools, and the item becomes invalid in the segment pools as well.

Level II Requirements: Test Authoring

RADTIB 39.1	User should have the option to do the following:-			
RADTIB 39.1	Repeat more limited searches on individual standards in the pool			
RADTIB 39.2	Prune the item pool by removing items from the pool or making them inactive in the pool			
RADTIB 40	Create Item Pools (Segment Pool)			Switching from Standards to Socks in the button bar allows you to see the item pool summary of items allocated to SOCKs for the master test or to child test segments.
RADTIB 40.1	No item searches are allowed in the SOCKs view to expand the item pool. Rather, the item pool SOCK assignment can be pruned by selecting the Edit feature for a SOCK.			Screen 10-3: Item Pools (SOCKs)
RADTIB 41	The Test Authoring application will provide a user interface so that the psychometric evaluation result records can be created and stored with the test at any time during the test workflow, including after approval or publishing.	Psychometric Information	Psychometric evaluation records do not alter any of the test's test elements.	

RADTIB 41.1	The Test Authoring application will provide a user interface that allows the attachment of arbitrary files such as text, spreadsheet or pdf files to particular psychometric evaluation result records at any time during the test workflow, including after approval or publishing.	Psychometric Information	Same as above: psychometric Evaluation records do not alter any of the test's test elements, so they can be updated at any time without changing the test.	This is straightforward and initiated in the TestAuth UI. User adds a psychometric record and provides the minimal input shown in the mockups. The upload feature permits the user to upload, attach and subsequently download any file attachment: pdf, spreadsheet, etc. This can happen even to a test that has been published and isn't locked down. - Per David Tuesday, February 25, 2014 1:58 PM
RADTIB 42.1	Test Authoring Component will meet requirements detailed in the SBAC IT Systems Architecture (Appendix C)		Outlined in the RFP, this is the header of the main requirement for Test Packaging.	
RADTIB 42.2	Create XML file containing data required for the DTD. testpackage.dtd	Test Package	<p>A formal specification of the XML elements that represent a test. XML files that conform to this DTD are suitable the following purposes:</p> <ol style="list-style-type: none"> 1. Administration: an XML representation of a test that contains all elements necessary to administer the test to a student. 2. An Administration XML packaged with item metadata is suitable for adaptive test simulation 3. An Administration XML packaged with full item content, metadata and item assets can be used to configure a Test Delivery system for administration of a test to students 4. Scoring: an XML representation of a test that contains sufficient elements for configuration of a test scoring system with custom rules for scale scoring this test 5. Reporting: an XML representation of a test that contains sufficient elements for configuration of a test reporting system for reporting of the scale scores for a test 6. Complete: A superset of all the test elements separately required in all of the test representations above 	<p>For each test specification domain entity, information about how interface elements map into XML fields emitted by the test packager.</p> <p>I discussed this with Jon Cohen. He told me that this mapping was a detailed design activity that DRC should do. In one of the tech calls, I discussed with Mike and the attendees. I indicated on that occasion, and subsequent occasions, that when designing and implementing any feature, the designer should:</p> <ol style="list-style-type: none"> 1. Pay attention to the DTD and ensure that the UI provided all of the elements required. 2. Consult the examples and determine if there were any usage patterns of XML elements that were required but the UI didn't provide. 3. Determine what elements from the UI mapped into XML elements in the test package. <p>In many of the tech calls, whenever a question was asked, I retraced these</p>

				steps during the call with the intention of ingraining this analysis process into the development team. In many email communications, I also retraced these steps analyzing the DTD, the example XMLs and the UI mockups and indicating how they are related. Ping Mike if you have any questions about this.
RADTIB 42.3	Create XML file containing data required for the DTD. DE_ALG_registration.xml	Test Package	An abridged test specification that is used to configure the Test Registration component. No item data is packaged in this format. DE_ALG_registration.xml	
RADTIB 42.4	Create XML file containing data required for the DTD. DE_ALG_registration.xml	Test Package	An abridged test specification that is used to configure the Test Registration component. No item data is packaged in this format.	
RADTIB 42.5	Create XML file containing data required for the DTD. DE_PT_ALG_admin.xml	Test Package	A test specification that is used to perform adaptive simulations or to configure a test delivery system. A simulation XML must be packaged with the item metadata, and an XML intended for a test delivery component requires full item metadata, content and all item assets such as graphics and multimedia files.	
RADTIB 42.6	Create XML file containing data required for the DTD. DE_read4_scoring.xml	Test Package	A test specification that is used to configure a test scoring system. Only item references and scoring information is packaged in this format.	
RADTIB 42.7	Create XML file containing data required for the DTD. DE_read4_reporting.xml	Test Package	A test specification that is used to configure a reporting system. No item data is packaged in this format.	
RADTIB 42.8	Create XML file containing data required for the DTD. DE_ALG_complete.xml	Test Package	A test specification that is a superset of all the elements described above. The four test specification types above contain some common elements and some unique elements. The Complete package contains a superset of all elements.	
RADTIB	A Test Packaging function must be included with the Test Authoring	Test Package		

42.9	application to create Test Packages for various purposes. In order to create a test package explicitly using the Test Packaging user interface, the test and its sub-elements must have been fully approved and be in the Approved or Published workflow level described in section 7.2 Test Approval Workflow			
RADTIB 43	Create CAT Simulator that uses REST interface to create a simulation record and store simulation information. The user can subsequently go into this screen and add attachments to the simulation information record.	<p>Tests must be in the Not Submitted or Validated states to be eligible for simulation, so simulation records and attachments can only be done when the test is in these states.</p> <ul style="list-style-type: none"> · Tests must be in the Not Submitted or Validated state for simulation results to be written back to the test repository 	<p>Simulation results look identical, but the simulation record isn't added by the user as with psychometric records. There are interfaces on TestAuth that the simulator needs. The sequence is as follows:</p> <ol style="list-style-type: none"> 1. Simulator browses TestAuth for in-process tests ready for simulation. 2. Simulator is able to extract in-process tests for simulation. 3. Simulator performs simulations. 4. Simulator writes modified results back to TestAuth and updates in-process elements: blueprints, forms, item pools, etc. When it does so, it creates a simulation record. 5. When the tests go through the normal approval workflow, they become ineligible to be modified by the simulator. 6. Simulation records permit the upload of arbitrary attachments (PDFs, documents, spreadsheets, etc.) at any time, even after tests are published. 	

User Interface Summary

The following table summarizes the user interface screens in the Test Authoring application. The screens correspond to the specific domain elements managed by Test Authoring, but there are additional screens provided for management of independent metadata, for workflow management and for test packaging.

Function	Screen	Mockup Identifier	Description	Business Rules	Permissions
Home	Home	01	This screen provides home screen navigation access to the various Test Authoring functions	Individual users will be able to access only the screens that correspond to the permissions they are authorized to access. Only the features that the user is permitted to access are available and all others are disabled.	All users can access this screen
Setup	Setup	02	This screen provides navigation access to the Setup functions	Setup is an administrative function that few users should be able to access.	Setup Create/Read/Update/Modify
Setup	Subjects	03	Screens to CRUD subject metadata are required. The list of publishers offered come from here.	<ul style="list-style-type: none"> A subject cannot be deleted once a test using this publisher has been created A subject cannot be modified once a test using this publisher has been published 	Setup Create/Read/Update/Modify
Setup	Standards Publications	04a1, 04a2	<p>Standards publications from Core Standards are browsed and associated with a publisher and a subject.</p> <ul style="list-style-type: none"> Show a grid of publications with the publication name, description and subject Provide a dropdown with every publisher that has a publication in the repository. Allow the user to select a publisher to filter the publications in the grid by publisher. Default is "Show all publishers" Show a dropdown of subjects from the subject metadata to select from Show a dropdown of publishers from the publisher metadata to select from Confirm association of a standards publication to a publisher and a subject. Allow disassociation of a publication from a subject/publisher 	<ul style="list-style-type: none"> Once a test has been created with a given subject, publisher and standards publication, the publication/publisher/subject association cannot be disassociated Cardinality isn't a requirement, this is an item issue Required levels are primarily for items The required levels are needed only on the BP screen itself 	Setup Create/Read/Update/Modify
Setup	Item Selection Algorithms	05	An XML configuration file for item selection algorithms is loaded on demand <ul style="list-style-type: none"> The configuration file is analyzed and issues problems are reported to the user If the configuration file has no errors, the user can accept the changes into the Test Authoring 	<ul style="list-style-type: none"> Changes cannot be accepted unless the configuration file integrity check reveals no errors Changes to existing item selection configuration that changes an item selection algorithm already selected into a test are not allowed 	Setup Create/Read/Update/Modify

Level II Requirements: Test Authoring

			configuration		
Setup	Scoring Functions	06	An XML configuration file for scoring functions is loaded on demand <ul style="list-style-type: none"> • The configuration file is analyzed and issues problems are reported to the user • If the configuration file has no errors, the user can accept the changes into the Test Authoring configuration 	<ul style="list-style-type: none"> • Changes cannot be accepted unless the configuration file integrity check reveals no errors • Changes to existing scoring function configuration that changes a scoring function already selected into a test are not allowed 	Setup Create/Read/Update/Modify
Test	Tests	07, 1-2	<ul style="list-style-type: none"> • Show all tests in a grid • Provide dropdowns to filter by publisher, subject and grade • Provide a link for users to select a test to edit • Provide a means for users to delete a test • Provide the means for users to create a new test 	<ul style="list-style-type: none"> • A test cannot be deleted once it has been published • Deleting a test destroys all child elements 	Test Create/Delete Test Read Test Modify
Test Segment	Test Segments	08, 1-2	<ul style="list-style-type: none"> • Show all test segments in a grid • Provide dropdowns to filter test segments by publisher, subject, grade and test • The test dropdown is itself filtered by publisher, subject and grade • Provide a link for users to select a test segment to edit • Provide a means for users to delete segment a test • Provide the means for users to create a new test 	<ul style="list-style-type: none"> • A test segment cannot be deleted once it has been published • A test segment may be added, but it changes the versioning 	Test Segment Create/Delete Test Segment Read Test Segment Modify
Test Blueprint	Blueprints	09, 1-4	<ul style="list-style-type: none"> • Show all test blueprints in a grid • Provide dropdowns to filter test segments by publisher, subject, grade, test and test segment • The test and test segment dropdowns are themselves filtered by publisher, subject and grade • Provide a link for users to select a test segment to edit • Provide a means for users to delete segment a test • Provide the means for users to create a new test 	Detailed requirements in section 9 <u>Additional Functional Requirements</u> and in section 10 <u>Consistency Checks and Validations</u> . See Test Blueprint Validations for detailed business rules for this screen.	Blueprint Create/Delete Blueprint Read Blueprint Modify
Item Pool	Item Pools	10, 1-5	This screen is organized by the same standards that are selected for the blueprint. <ul style="list-style-type: none"> • Blueprint information is shown in a read-only manner • Total operational and field test items in the pool 	<ul style="list-style-type: none"> • The Test Item Bank is searched for items that correspond to the content standards selected for the blueprint • New searches may repopulate standards with items previously pruned from the item pool 	Item Pool Create/Delete Item Pool Read Item Pool Modify

			<ul style="list-style-type: none"> ● Switch between Standards and SOCKs view ● View total number of operational and field test items ● Switch between a master test item pool or a segment item pool ● Summary of item searches are shown as item operational and field test item counts (counts roll up hierarchically) ● Item summary counts for operational and field test and for test and segment (when viewing segment pool) ● Edit links take you to an Edit Item Pools screen to the means to view items individually (all items, or items for a particular content standard and their children if any) ● Search links allow the user to search (or search again) for items for all standards or for a particular content standard and their children if any) ● Searches are conducted for both operational and field test items 	<ul style="list-style-type: none"> ● Searches can be conducted for any all content standards or for any individual content standard at any level in the hierarchy ● Searching for items at a content standard level higher in the hierarchy than the lowest level will include all child standards as well 	
Item Pool	Edit Item Pools	10, 6-9	<p>This screen shows individual items (not just item summaries) organized by the same standards selected for the blueprint</p> <ul style="list-style-type: none"> ● Toggle between viewing items and passages ● Filter items by an individual content standard (and its children) or see items for all standards ● Display of total number of items in pool, filter and segment (if editing a segment pool) ● Switch between master test pool and segment pools ● Delete individual items from the master pool ● See more details about an item 	<ul style="list-style-type: none"> ● Cannot change OP/FT, Required or Active status of an item from the segment pool view ● Cannot delete an item from the segment pool view, only deselect it from the pool ● Cannot change passage status or delete a passage: passages are pulled in by item references 	Item Pool Read Item Pool Modify
Test Forms	Forms	11	This screen allows the creation of test forms. Forms are simple container elements that contain form partitions.	<ul style="list-style-type: none"> ● A test with at least one fixed form segment requires at least one form 	Test Form Create/Delete Test Form Read Test Form Modify
Form Partitions	Form Partitions	12-1	This screen allows the creation of form partitions that belong to a form. Filters are provided to narrow the number of forms under which to create a partition.	<ul style="list-style-type: none"> ● A form requires as many form partitions as there are fixed form segments 	Test Form Create/Delete Test Form Read Test Form Modify
Form Partitions	Edit Form Partitions	12-2	Edit Form Partitions allows the user to use the search screen to search the existing item pool for items to place on the form. Items can be sequenced and deleted as well.	<ul style="list-style-type: none"> ● Item searches are conducted against the item pool, not against the test item bank. 	Test Form Create/Delete Test Form Read Test Form Modify

Level II Requirements: Test Authoring

Test Segments and Form Partitions	Assign Form Partitions to Test Segments	13	This screen allows the assignment of test form partitions to test segments.	<ul style="list-style-type: none"> Only fixed form test segments participate in form partition assignment. 	Test Segment Create/Delete Test Segment Read Test Segment Modify
Scoring Rules	Scoring Rules	14	This screen allows the user to build scoring rules by selecting computation rules and entering the parameters of the computation rule. The computation rules are configured into the Test Authoring system and the user interface presents the configured parameters dynamically.	<ul style="list-style-type: none"> The user is offered a preconfigured set of computation rules to choose from. Once created, computation rules can be moved up and down as the sequence of rules is critical. The user interface for entering the computation rule parameters is dynamic based on parameter configuration. Existing blueprint elements are referred to. 	Computation Rule Create/Delete Computation Rule Read Computation Rule Modify
Conversion Tables	Conversion Tables	15	This is pending as more information is required.		
Performance Levels	Performance Levels	16	Performance levels are a sequential set of level cutters. They refer to a blueprint element and permit the user to enter a low and high value.	Existing blueprint elements are referred to.	Performance Level Create/Delete Performance Level Read Performance Level Modify
Reporting Measures	Reporting Measures	17	This screen allows the user to create reporting measures, which are a collection of references to existing computation rules from Scoring Rules. The user adds computation rules to a reporting measure until they are satisfied.	<ul style="list-style-type: none"> Existing blueprint elements are referred to Existing computation rules are referred to 	Reporting Measure Create/Delete Reporting Measure Read Reporting Measure Modify
Workflow	Workflow	18	The Workflow screens are left to DRC engineers to design		Test Package Create Test Package Read
Test Packaging	Test Packaging	19	This document will be extended with packaging requirements when the Test Packager requirements are due.		
Simulation	Simulation	19	CAT Simulator uses REST interface to create a simulation record and store simulation information. The user can subsequently go into this screen and add attachments to the simulation information record.	<ul style="list-style-type: none"> Tests must be in the Not Submitted or Validated states to be eligible for simulation, so simulation records and attachments can only be done when the test is in these states. Tests must be in the Not Submitted or Validated state for simulation results to be written back to the test repository 	Simulation Information Create Simulation Information Read
Psychometric Information	Psychometric Information	20	Authorized users can create a Psychometric information record and add attachments to the record.	<ul style="list-style-type: none"> Tests can be in any state when psychometric information records are added because the addition of this information does not change the state of the test. All workflow rules governing what can be modified and when based on workflow state apply. 	Psychometric Information Create Psychometric Information Read

Table 9. User Interface Summary

User Interface Screen Mockups

This section presents mockups of the screens described in the summary above. They are presented as only as input to the design process and are not intended to be definitive or restrictive. Designers of Test Authoring are free to innovate and improve these layouts if they determine that an alternate layout enhances usability, user understanding or efficiency.

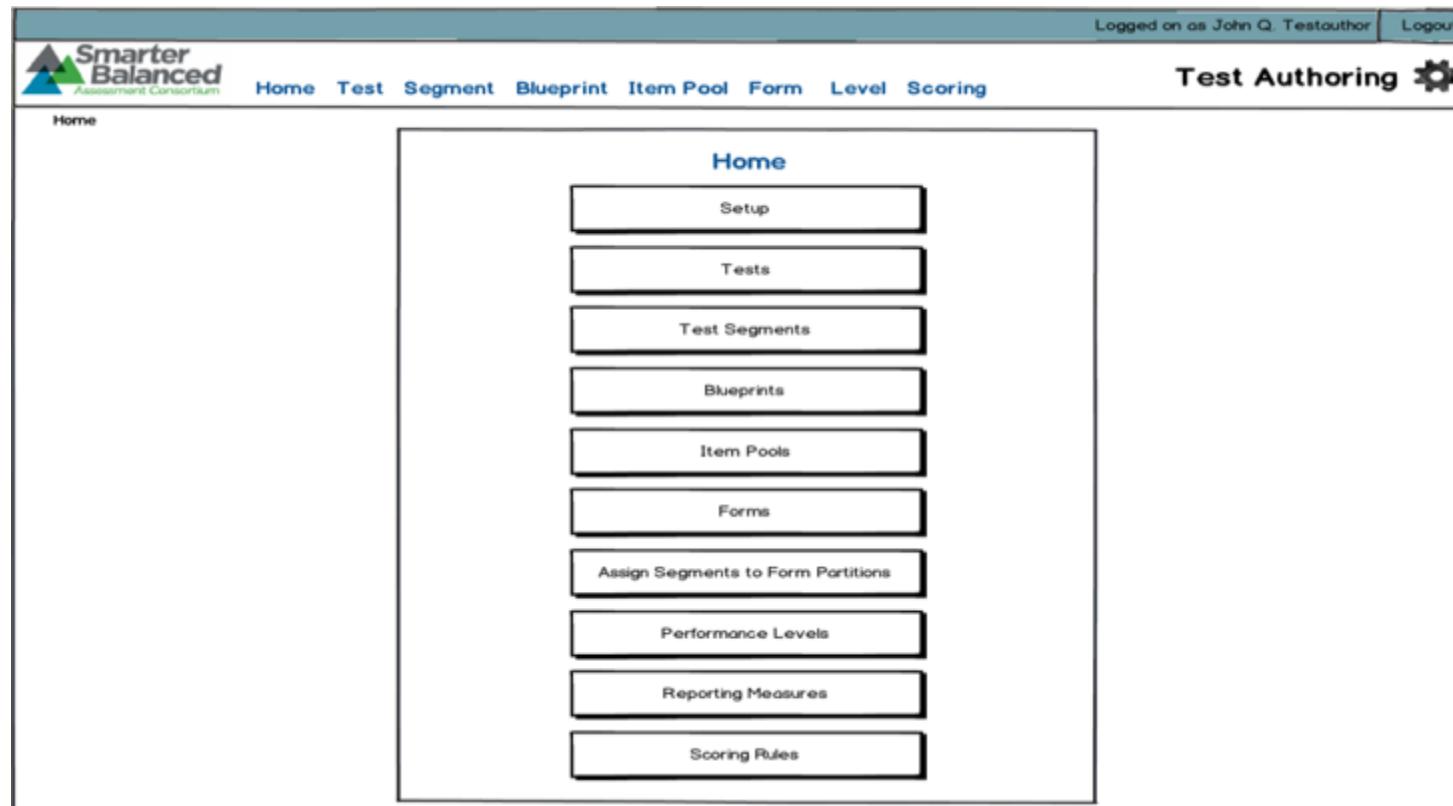


Figure 5. Screen 01: Home

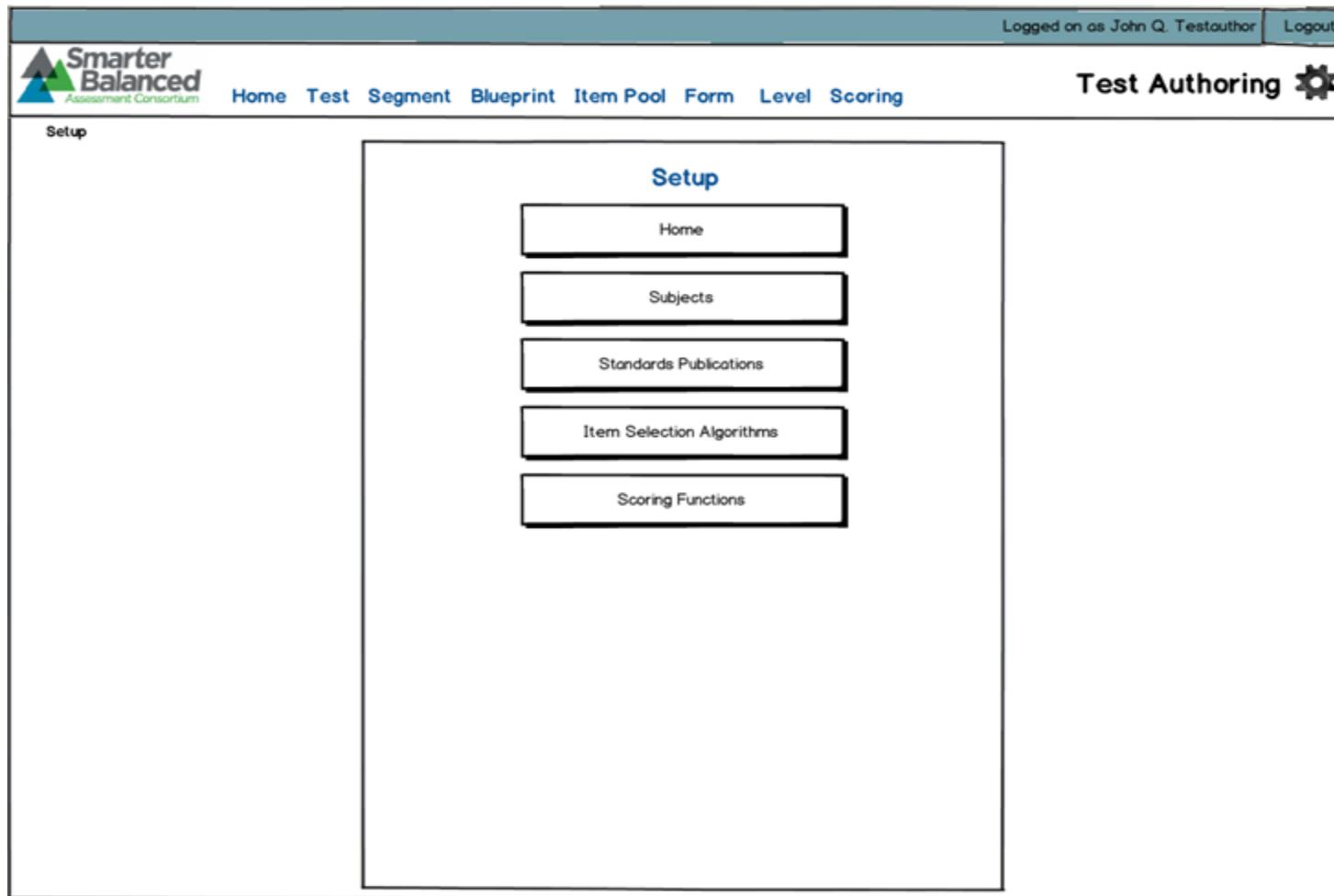


Figure 6. Screen 02: Setup

The screenshot shows the Smarter Balanced Test Authoring interface for managing subjects. The top navigation bar includes links for Home, Test, Segment, Blueprint, Item Pool, Form, Level, and Scoring. A yellow callout box points to the 'Add New Subject' button, which is highlighted in red. Another yellow callout box on the left side of the page states: "The list of Publishers are the list of tenants subscribed to Test Authoring and come from Program Management".

Add New Subject Dialog:

- Subject Name: Chemistry
- Abbreviation: CHE
- Publisher: Select Publisher (dropdown menu showing California, North Carolina, Smarter Balanced Assessment)
- Description: Chemistry is available as an elective for grades 10, 11 and 12.
- Buttons: Save and Add Another, Save and Exit, Cancel.

Edit Subject Dialog (for Biology):

- Subject Name: Biology
- Abbreviation: BIO
- Publisher: Select Publisher (dropdown menu showing California, North Carolina, Smarter Balanced Assessment)
- Description: Biology is available as an elective for grades 10, 11 and 12.
- Buttons: Save, Cancel.

List of Existing Subjects:

Subject	Abbreviation	Publisher
English Language Arts	ELA	Smarter Balanced Assessment Consortium
Mathematics	MA	Smarter Balanced Assessment Consortium
Biology	BIO	California

Annotations:

- A yellow callout box above the list of subjects says: "Click this button to add a new Subject".
- A yellow callout box on the left side of the list says: "These subjects have created tests against them, and so can only have limited edits and no deletes".
- A yellow callout box on the right side of the list says: "No biology tests have been created, so this subject metadata may be deleted and edited".
- A yellow callout box on the 'Edit Subject' dialog says: "Editing a subject brings up the same dialog with this heading".

Figure 7. Screen 03: Subjects

Figure 8. Screen 04-1: Publications

The screenshot shows the Smarter Balanced Test Authoring interface. At the top, there is a navigation bar with links: Home, Test, Segment, Blueprint, Item Pool, Form, Level, Scoring, Logout, and a gear icon for settings. Below the navigation bar, the page title is "Test Authoring". On the left, there is a sidebar with a "Setup > Publications" link. The main content area has two sections: "Add Publication" and "Select Standards Publication". The "Select Standards Publication" section contains dropdown menus for "Publication Publisher" (California, North Carolina, Smarter Balanced Assessment Consortium), "Publication Subject" (Biology, Chemistry, English Language Arts, Mathematics), and "Publication" (CA-BIO-V1, NC-BIO-V1, SBAC-ELA-V1, SBAC-MA-V1). The "Add Publication" section contains dropdown menus for "Publisher" (Select Test Authoring Publisher, Smarter Balanced Assessment Consortium) and "Test Authoring Subject" (English Language Arts, Mathematics). At the bottom of the "Add Publication" section are three buttons: "Save and Add Another", "Save and Exit", and "Cancel". To the right of these sections is a table titled "Add Publication" with columns for "Subject", "Action", and "Delete". The table lists two rows: "English Language Arts" and "Mathematics". A yellow callout box points to the "Delete" link for the English Language Arts row, containing the text: "The user selects a publication and a Test Authoring subject against which the publication will be associated. Clicking the Save and Add Another or Save and Exit buttons creates an association between this publication and the Test Authoring Publisher/Subject combination." Another yellow callout box points to the "Publisher" dropdown in the "Add Publication" section, containing the text: "The list of Publishers is the list of tenants subscribed to Test Authoring and come from Program Management".

Subject	Action
English Language Arts	Delete
Mathematics	Delete

Figure 9. Screen 04-2: Publications (Add Publication)

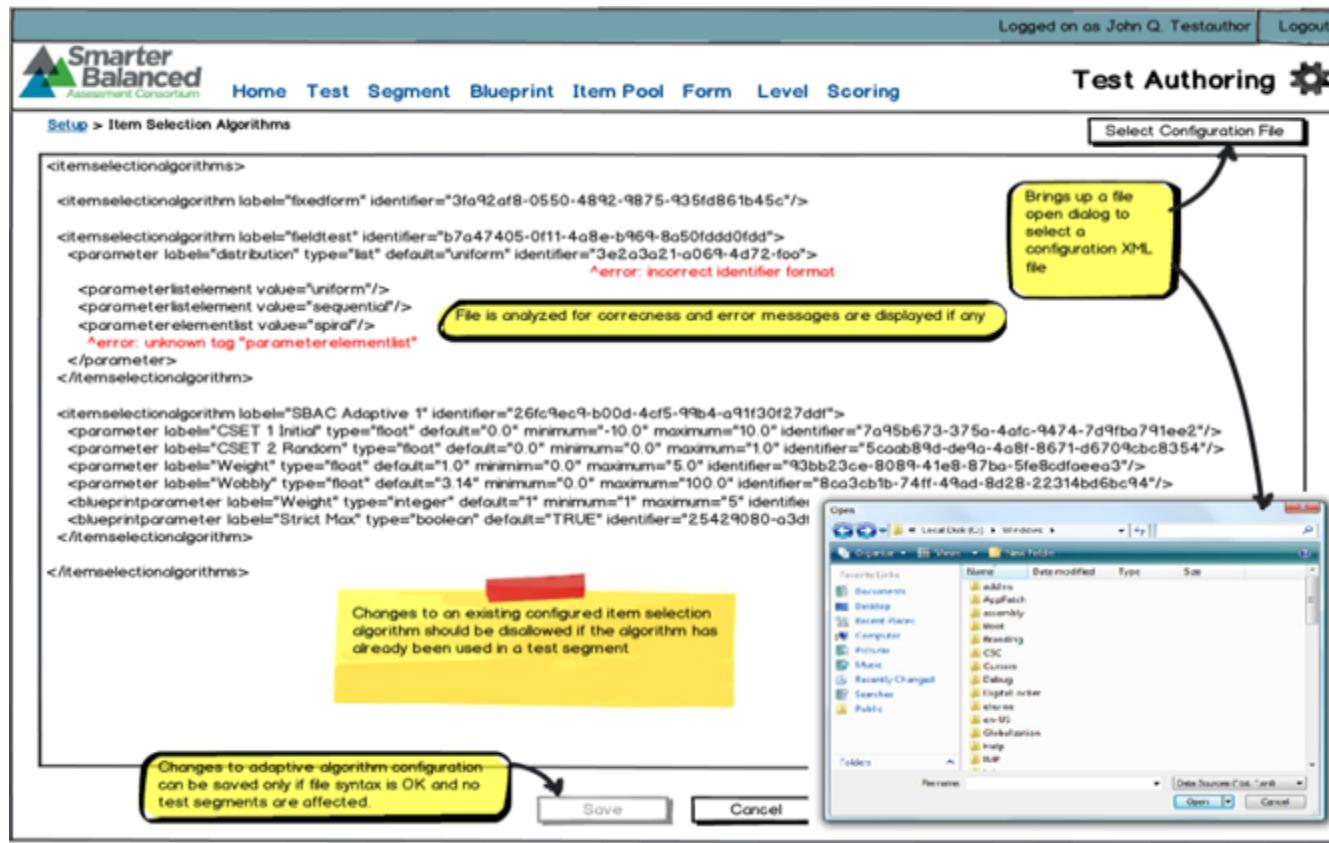


Figure 10. Screen 05: Item Selection Algorithm Configuration

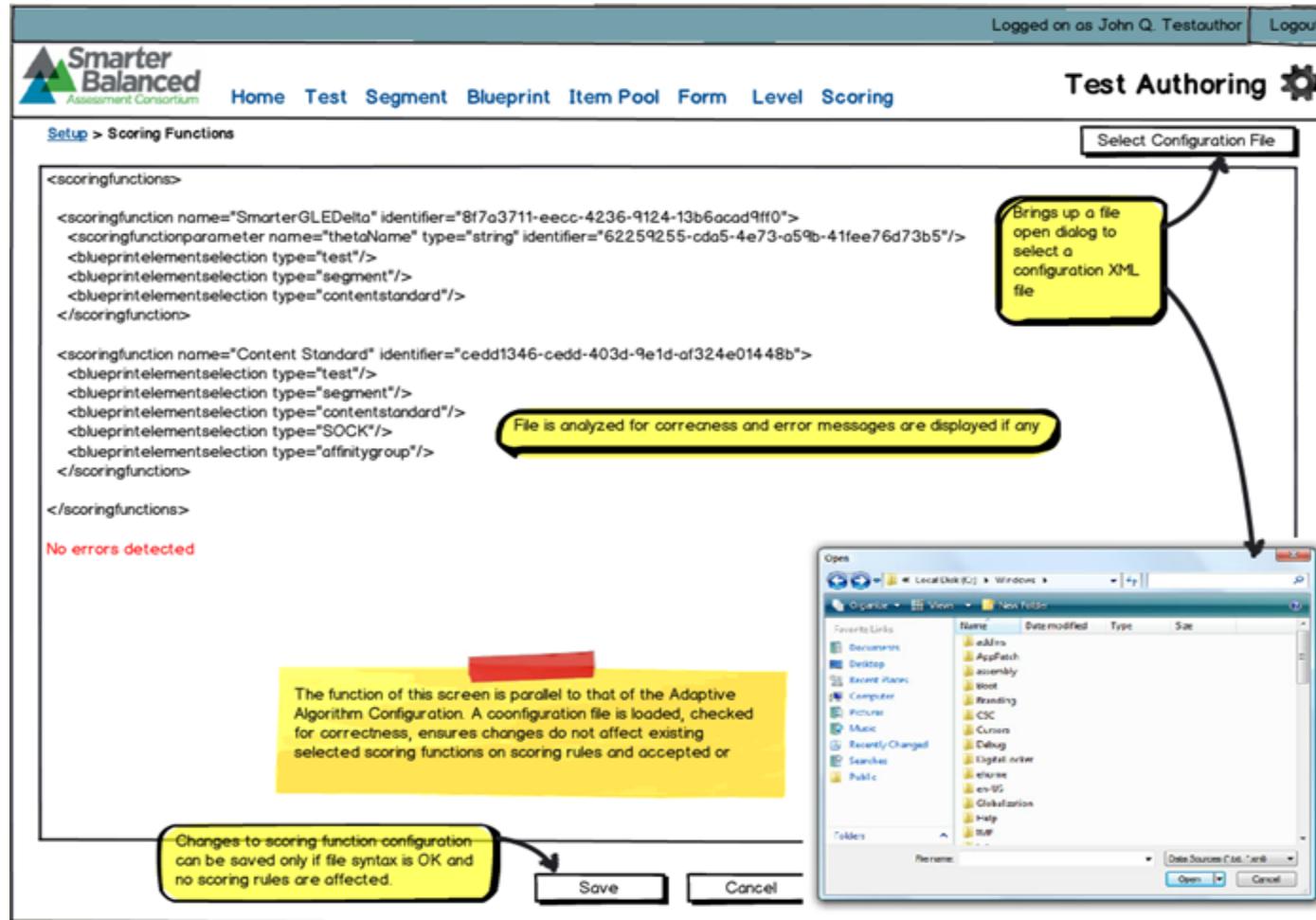


Figure 11. Screen 06: Scoring Function Configuration

The screenshot shows a table of tests with the following columns:

- Test Name**: SBAC-ELA-4-2014-2015-S, SBAC-ELA-5-2014-2015-S, SBAC-ELA-6-2014-2015-S, SBAC-MA-4-2014-2015-S, SBAC-MA-5-2014-2015-S, SBAC-MA-6-2014-2015-S.
- Label**: ELA Grade 4 2014-2014, ELA Grade 4 2014-2014.
- Subject**: English Language Arts, English Language Arts, English Language Arts, Mathematics, Mathematics, Mathematics.
- Grade**: 4, 5, 6, 4, 5, 6.
- Administration**: Fall 2014-2015, Fall 2014-2015, Fall 2014-2015, Fall 2014-2015, Fall 2014-2015, Fall 2014-2015.
- Type**: Summative, Summative, Summative, Summative, Summative, Summative.
- Category**: (empty), (empty), (empty), (empty), (empty), (empty).
- Publication**: SBAC-ELA-V1, SBAC-ELA-V1, SBAC-ELA-V1, SBAC-ELA-V1, SBAC-ELA-V1, SBAC-ELA-V1.
- Last Published**: 11/4/14 3:15 PM, not published, 11/11/14 2:42 PM, 11/20/14 4:23 PM, 10/11/14 9:00 AM, not published.
- Action**: Edit Delete Blueprint, Edit Delete Blueprint.

Callout boxes provide additional details:

- Test Name**: Test names are managed by Test Authoring, and are a concatenation of Publisher, Subject, Grade, Administration, and test type (I, F, S). The test name must be unique, and if these components aren't sufficient to disambiguate tests, the naming scheme will have to change to ensure unambiguous naming.
- Label**: The test label is entered by the user in a free text field and is not required to be unique.
- Subject**: The Subject is selected from the list of subject metadata when the user creates the test.
- Grade**: Grade is selected by the user when the test is created. There may be more than one grade, and an appropriate delimiter must be selected so the Name concatenation doesn't contain any "problem" characters.
- Administration**: Administration is entered by the user and is a free text field.
- Type**: Tests are either Interim, Formative or Summative. I, F or S is appended to the test name.
- Category**: The Category allows Test Administrators to filter the list of assessments to administer. It is a free text.
- Publication**: The Publication is a Core Standards publication selected by the user when the test is created.
- Last Published**: The concept of Test "Publication" needs to be fleshed out. A test is "published" when it is loaded into the Test Delivery system and administered to at least one student. Unfortunately, there is no feedback from Test Delivery to confirm this, so discussion is needed.
- Action**: Action links allow the user to edit this test or its blueprint. Deletion of a test will delete all child elements, so will require a confirmation dialog. Deletion is not allowed once a test has been published.

Figure 12. Screen 07-1: Tests

This view shows the Add New Test and Edit Test dialogs.

The screenshot shows the Smarter Balanced Test Authoring interface. At the top, there's a navigation bar with links for Home, Test, Segment, Blueprint, Item Pool, Form, Level, Scoring, and a user status message "Logged on as John Q. Testauthor" with a Logout link.

The main area has tabs for "Home > Tests", "Create Tests", "Select Grade", "Select Subject", "Select Publisher", and "Select Category". A "Test Authoring" button with a gear icon is also present.

Add New Test Dialog: This dialog is shown on the left. It contains fields for Label (empty), Subject (dropdown menu showing "Select Subject", "English Language Arts", and "Mathematics"), Grade (checkboxes for 3, 6, 9, 12, 4, 7, 10, 5, 8, 11, where 4 is checked), Administration (dropdown menu showing "Fall 2014-2015" and "Spring 2015"), Type (dropdown menu showing "Summative", "Formative", "Interim", "Summative"), Category (empty), Publication (dropdown menu showing "Select Standards Publication", "SBAC-ELA-V1", "SBAC-MA-V1"), Comment (empty), and Description (empty). Buttons at the bottom include "Save and Add Another", "Save and Exit", and "Cancel". A yellow callout points to the "Add New Test" button in the top right corner of the dialog, stating: "The Add New Test button allows brings up this dialog".

Edit Test Dialog: This dialog is shown on the right. It contains fields for Label ("ELA Grade 4 2014-2014"), Subject (dropdown menu showing "English Language Arts" and "Mathematics"), Grade (checkboxes for 3, 6, 9, 12, 4, 7, 10, 5, 8, 11, where 4 is checked), Administration ("Fall 2014-2015"), Type (dropdown menu showing "Summative", "Formative", "Interim", "Summative"), Category (empty), Publication (dropdown menu showing "SBAC-ELA-V1", "SBAC-ELA-V1", "SBAC-MA-V1"), Comment ("This is a 4th Grade ELA test"), and Description ("The most common description of a 4th grade ELA assessment is that it is valuable for the teacher and arduous for the 4th grader"). Buttons at the bottom include "Save" and "Cancel". A yellow callout points to the "Edit" link in the "Action" column of a table on the right, stating: "Clicking on a test's Edit link brings up this dialog". Another yellow callout points to the "Blueprint" link in the same table, stating: "Click on the Blueprint link to edit a test's blueprint".

Action Table: A table on the right lists actions for tests. It includes columns for "Time", "Action", and "Edit Delete Blueprint". The table rows show: "1 PM Edit Delete Blueprint", "2 PM Edit Delete Blueprint", "3 PM Edit Delete Blueprint", and "4 AM Edit Delete Blueprint".

Figure 13. Screen 07-2: Test (Add/Edit Test)

The screenshot shows the Smarter Balanced Test Authoring interface. The top navigation bar includes links for Home, Test, Segment, Blueprint, Item Pool, Form, Level, and Scoring. The user is logged in as John Q. Testauthor.

The main content area displays a table of existing test segments and provides options to add new ones. The table columns include Parent Test, Segment Position, Grade, Subject, Min Items, Max Items, Item Selection, and Action.

Annotations highlight specific features:

- A callout points to the "Add New Test Segment" button in the header, stating: "New test segments are always created in the last position, so there is no 'Position' entry in Add New Test Segment".
- An annotation on the "Min Items" column header says: "Filters allow a user to narrow down the list of test segments".
- An annotation on the "Max Items" column header says: "Click Blueprint to go directly to this segment's blueprint".
- An annotation on the "Item Selection" column header says: "Click Edit to opt to change a segment's adaptive".
- An annotation on the "Action" column header says: "Edit Delete Blueprint".
- An annotation on the "Add Test Segment" dialog box states: "Click Add to create a new test segment".
- An annotation on the "Edit Test Segment" dialog box states: "Edit Item Selection Parameters".

The "Add Test Segment" dialog box contains fields for Parent Test (dropdown menu showing SBAC-ELA-4-2014-2015-S, SBAC-ELA-5-2014-2015-S, SBAC-ELA-5-2014-2015-S), Min Items (text input), Max Items (text input), Item Selection (dropdown menu showing fixed form, field test, adoptive), Version (dropdown menu showing 1.0, 1.1), and Save/Cancel buttons.

The "Edit Test Segment" dialog box shows the current settings for SBAC-MA-4-2014-2015-S: Position 2, Min Items 22, Max Items 22, Item Selection (dropdown menu showing SBAC-Fixed, SBAC-Adaptive, SBAC-Fixed, SBAC-Adaptive), Version (dropdown menu showing 1.0, 1.1), and Save/Cancel buttons.

Figure 14: Screen 08-1: Test Segments

Figure 15. Screen 08-2: Test Segments (Edit Item Selection Parameters)

The screenshot shows the Test Authoring interface for the SBAC-ELA-V1 publication at Grade 5. The main content area displays a table for standard keys, levels, and grades. A yellow box highlights the 'SOCKS' tab in the top navigation bar. Another yellow box highlights the 'Change' button in the top right. A large yellow box covers the entire right side of the screen, containing several annotations:

- A callout points to the 'Grade' column header: "Brings up the grade filter dialog".
- A callout points to the 'OP Min' column: "Clear buttons set all values in the column to Zero after presenting a".
- A callout points to the 'Active' column: "Master checkbox checks or unchecks all checkboxes in this column".
- A callout points to the 'OP Sum of Min' column: "Clear buttons set all values in the column to Zero after presenting a".
- A callout points to the 'OP Max' column: "Clear buttons set all values in the column to Zero after presenting a".
- A callout points to the 'OP Sum of Max' column: "Clear buttons set all values in the column to Zero after presenting a".
- A callout points to the 'FT Min' column: "Clear buttons set all values in the column to Zero after presenting a".
- A callout points to the 'FT Sum of Min' column: "Clear buttons set all values in the column to Zero after presenting a".
- A callout points to the 'FT Max' column: "Clear buttons set all values in the column to Zero after presenting a".
- A callout points to the 'FT Sum of Max' column: "Clear buttons set all values in the column to Zero after presenting a".

Annotations on the left side of the table:

- A callout points to the 'Standard Text' column: "Standards come from the publication selected for the test".
- A callout points to the 'Level' column: "Standards are filtered according to Grade. The default grade is the grade of the test, but can be overridden by a".
- A callout points to the 'Grade' column: "Standards are indented according to level".
- A callout points to the 'OP Min' column: "Determine or clarify the meaning of unknown and multiple-meaning word and phrases based on grade 3 reading and content, choosing flexibly from a range of strategies. (C) Use a known root word as a clue to the meaning of an unknown word with the same root".

Annotations at the bottom of the table:

- A callout points to the 'OP Min' column: "The Blueprint UI allows editing of both test and segment blueprint. The segment blueprints are always constrained to be subsets of the test blueprint which is considered the master blueprint".

Figure 16. Screen 09-1: Blueprints

Segment blueprints may require adaptive algorithm configuration that is unique to a selected algorithm on a per-standard basis. Tags in the item configuration XML indicate the type and other information for each parameter to be added, which translates to new columns in the blueprint screen.

This is an example of how the adaptive algorithm configuration is capable of modifying the blueprint screen. Adaptive algorithms require global configuration parameters that are set in the test segment screens, but they may also require configuration parameters per standard. These are added in the Item Selection Algorithm Configuration screen by means of a <blueprintparameter> tag in the item selection configuration XML.

Please note that item selection is configured per segment, so the blueprint is customized only when editing a segment blueprint. Different segments may have different item selection configuration, so different segments may have different additional columns as a result.

The following two tags resulted in the addition of the Weight and Strict Max columns to the segment blueprint:

```
<blueprintparameter label="Weight" type="integer" default="1" minimum="1" maximum="5" identifier="ec31eeaf-256c-45e8-ae3c-4efc1a10642d"/>
<blueprintparameter label="Strict Max" type="boolean" default="TRUE" identifier="25429080-a3df-431f-97d7-4ea66bbaf09d"/>
```

Standard Key	Level	Grade	Standard Text	OP Min	OP Sum of Min	OP Max	OP Sum of Max	FT Min	FT Sum of Min	FT Max	Active	Weight	Strict Max
1-IT	1	5	(i) Read Analytically: Informational Text - Students c	Clear 4	4	7	8	Clear 0	0	0	All <input checked="" type="checkbox"/>	1	<input checked="" type="checkbox"/>
1-ITI10-1	2	5	(i) WORD MEANINGS: Determine intended meanings of word	1	0	2	0	0	0	0	All <input checked="" type="checkbox"/>	1	<input checked="" type="checkbox"/>
1-ITI10-2	2	5	(i) WORD MEANINGS: Determine intended meanings of word	1	0	2	0	0	0	0	All <input checked="" type="checkbox"/>	1	<input checked="" type="checkbox"/>
1-ITI10-3	2	5	(i) WORD MEANINGS: Determine intended meanings of word	1	1	2	7	0	0	0	All <input checked="" type="checkbox"/>	1	<input checked="" type="checkbox"/>
1-ITI10-3i3.L.4a	3	5	(i) Determine or clarify the meaning of unknown and mu	0	1	0	0	0	0	0	All <input checked="" type="checkbox"/>	1	<input checked="" type="checkbox"/>
1-ITI10-3i2.L.4b	2	5	(i) Determine or clarify the meaning of unknown and mu	0	1	0	0	0	0	0	All <input checked="" type="checkbox"/>	1	<input checked="" type="checkbox"/>
1-ITI10-4	1-ITI10-4	5	(i) Acquire and use accurately grade-appropriate gener	0	1	0	0	0	0	0	All <input checked="" type="checkbox"/>	1	<input checked="" type="checkbox"/>
1-ITI10-4i4.RI.4	3	5	(i) Determine the meaning of general academic and doma	0	1	0	0	0	0	0	All <input checked="" type="checkbox"/>	1	<input checked="" type="checkbox"/>
1-ITI10-5	2	5	(i) WORD MEANINGS: Determine intended or precise meani	1	0	2	1	0	0	0	All <input checked="" type="checkbox"/>	1	<input checked="" type="checkbox"/>
1-ITI10-5i5.L.4a	3	5	(i) Determine or clarify the meaning of unknown and mu	0	1	0	0	0	0	0	All <input checked="" type="checkbox"/>	1	<input checked="" type="checkbox"/>

Figure 17. Screen 09-2: Blueprints (Adaptive Algorithm Configuration)

This view of the Blueprint screen shows the Grade Filter dialog and the selection of the master test blueprint or segment blueprints.

The screenshot shows the Smarter Balanced Test Authoring interface. At the top, there's a navigation bar with links for Home, Test, Segment, Blueprint, Item Pool, Form, Level, and Scoring. The user is logged in as John Q. Testauthor.

The main area is titled "Test Authoring" and shows a "Blueprints" section. A dropdown menu for "Test/Segment" is open, listing options like SBAC-ELA-5-2014-2015-S, SBAC-ELA-5-2014-2015-S-S1, and SBAC-ELA-5-2014-2015-S-S2.

A "Standard Key" table on the left lists various standard codes such as 1-ITI10-5, 1-ITI10-3I3.L.4d, etc. A dropdown menu here is highlighted with a yellow box, stating: "This dropdown collapses standards down to the level selected or shows all levels".

The central part of the screen shows a grid of standards. One standard is highlighted with a yellow box: "1-ITI10-4I4.RI.4". A tooltip for this standard says: "This button bar toggles between standards and SOCKs. Both of these are part of a blueprint but they do not fit on one screen." Another tooltip for the same standard says: "Validation errors cause edit fields to be rendered with a red background, and cause numerical values to appear in bold red".

A "Change Grade Filter" dialog box is overlaid on the grid. It contains a checkbox group with options: 3, 6, 9, 12, 4, 7, 10, 5, 8, 11. The checkboxes for 6, 9, 12, 4, 7, 10, and 5 are checked. Below the dialog are "Save" and "Cancel" buttons.

To the right of the grid, there are several configuration tables with columns for OP Min, OP Sum of Min, OP Max, OP Sum of Max, FT Min, and FT Max. Some cells in these tables contain validation errors, indicated by red backgrounds and bold red numbers.

A large yellow callout box on the right side of the screen provides instructions: "The Test/Segment dropdown contains the selected test and all of its segments. Standards selected as active in the master blueprint will be active for segments and those made inactive in the master blueprint will be inactive in the segment blueprints." Another part of this callout says: "The workflow requires the test author to create a master test blueprint, then select segments and change the values so they are a subset of the master blueprint."

Figure 18. Screen 09-3: Blueprints (Segments, Grade Filter, Level)

This screen shows the selection of SOCKs rather than standards. The only option is to make them active or not.

The screenshot shows the Smarter Balanced Test Authoring interface. At the top, there's a navigation bar with links for Home, Test, Segment, Blueprint, Item Pool, Form, Level, and Scoring. The user is logged in as John Q. Testauthor. The main content area is titled "Test Authoring" with a gear icon. Below this, the title "Blueprints" is followed by tabs for "Standards" and "SOCKS". The "SOCKS" tab is selected. The publication information is listed as SBAC-ELA-V1 Grade: 5 with a "Change" button, and the test/segment is set to SBAC-ELA-5-2014-2015-S. A table lists "Knowledge Category" (Depth of Knowledge 1, 2, 3) under "Description", with a note that Depth of Knowledge level 1 corresponds to items of least difficulty on a scale from 1 to 3, level 2 to medium difficulty, and level 3 to most difficulty. To the right of the descriptions is a column labeled "Active" with checkboxes. Three checkboxes are checked for Depth of Knowledge 1, 2, and 3. Three yellow callout boxes provide additional context: one points to the "Standards" tab, another points to the "SOCKS" tab, and a third points to the "Active" column, stating "As with standards, SOCKs are either selected or unselected for the test or the test segment".

Figure 19. Screen 09-4: Blueprints (SOCKs)

The screenshot shows the Smarter Balanced Test Authoring interface. The top navigation bar includes links for Home, Test, Segment, Blueprint, Item Pool, Form, Level, Scoring, and Logout. The current page is the Item Pool tab, indicated by the highlighted 'Item Search' button. A yellow sticky note in the top right corner states: "This is a master item pool for a test because the master test and not a test segment is selected". The main content area displays a table of items from the SBAC-ELA-5-2014-2015-S pool. The table columns are: Test OP Items, BP FT Min, BP FT Max, Test FT Items, and Action. Annotations include:

- A callout box points to the 'Item Search' button with the text: "Click Item Search to add analytically: Information about WORD MEANINGS: Determine intended meanings of word elements for the entire item pool".
- A callout box points to the 'Total item count is presented for the selected segment or test' text with the text: "The same standards selected for the blueprint are available for allocation of items in an item pool".
- A callout box points to the 'Edit' link in the 'Action' column with the text: "Click Edit to delete individual items from the pool for a specific content level".
- A callout box points to the 'Search' link in the 'Action' column with the text: "Click Search to search for items for a specific content level".
- A callout box points to the '142' value in the 'Test OP Items' column with the text: "The number of items roll up hierarchically from the leaf nodes of the content standards".

Pool Items: 312 OP 42 FT				
Standards Key	Level	Grade	Standards Text	
Show All Levels				
1-IT				
1-ITI10-1	3	5	WORD MEANINGS: Determine intended meanings of word elements for the entire item pool	yes 4 7 312
1-ITI10-2	3	5	WORD MEANINGS: Determine intended meanings of word elements for the entire item pool	yes 1 2 19
The same standards selected for the blueprint are available for allocation of items in an item pool	3	5	WORD MEANINGS: Determine intended meanings of word elements for the entire item pool	yes 1 2 30
1-ITI10-3 3.L.4	3	5	Determining or clarifying the meaning of words in context	yes 1 2 106
1-ITI10-3 3.RI.4	3	5	Creating test and segment blueprints are a predecessor to creation of a test or segment item pool. The standards selected for a test or test segment and their level, grade, test, min, max and active status are presented to provide information to the user when editing the item pool.	1 25
1-ITI10-4	3	5	The Item Search feature searches the Test Item Bank for items that are aligned to each standard, as a starting point. Search allows additional filters on item metadata, and the items selected for each standard can be pared down with an individual standard edit feature that allows picking and choosing of individual items. Search is also permitted at individual content levels. This is done so items can be added to the pool at the individual content level and not disturb other content levels as a master search would do.	1 18
1-ITI10-4 4.L.4a	2	5	Initial item search and selection should be performed at the master test level. When editing a test segment item pool, items can only be selected from the master test pool. Items can only be added or deleted from the pool at the master test level.	1 20
1-ITI10-4 4.L.4b	3	5	If an content standard is made inactive after the master test pool is created, the number of test items for that content standards will be set to zero and the any affected rollups are recalculated. However, the item selection is retained, and if the content level is made active again, items previously selected are restored to the pool.	0 0 4
1-ITI10-4 4.L.4c	3	5		0 0 3
1-ITI10-4 4.L.5c	3	5		0 0 3
1-ITI10-4 4.L.6	3	5		0 0 3
1-ITI10-4 4.RI.4	3	5		0 0 4
1-ITI10-5	2	5		0 0 2
1-ITI10-5 5.L.4a	3	5		0 0 2

Figure 20. Screen 10-1: Item Pools

This is an item pool for a test segment because a child test segment and not a master test is selected

Segment: SBAC-ELA-5-2014-2015-S-S1

Standards Key	Level	Grade	Standards Text	BP Active	BP OP Min	BP OP Max	Test OP Items	Seg OP Items	BP FT Max	Test FT Items	Seg FT Items	Action
1-IT	1		Item Search is only available for the master test item pool. It is not available when editing the segment pool.	yes	4	7	312	198	0	42	23	Edit Search
1-ITI10-1	2		WORD MEANINGS: Determine intended meanings of word	yes	1	2	19	9	0	1	1	Edit Search
1-ITI10-2	2		WORD MEANINGS: Determine intended meanings of word	yes	70	70	70	0	0	4	2	Edit Search
1-ITI10-3	2		WORD MEANINGS: Determine intended meanings of word	yes	108	108	35	0	0	13	9	Edit Search
1-ITI10-3I3.L.4a	3	5	(i) Determine or clarify the meaning of unknown and mu	yes	22	5	0	1	1	1	1	Edit Search
1-ITI10-3I3.L.4b	3	5	(i) Determine or clarify the meaning of unknown and mu	yes	25	10	0	1	1	0	0	Edit Search
1-ITI10-3I3.L.4c	3	5	(i) Determine or clarify the meaning of unknown and mu	yes	18	6	0	4	4	2	2	Edit Search
1-ITI10-3I3.L.4d	3	5	(i) Determine or clarify [redacted] of unknown and mu	yes	20	6	0	4	4	4	4	Edit Search
1-ITI10-3I3.RI.4	3	5	(i) De	yes	21	8	0	0	0	0	0	Edit Search
1-ITI10-4	2	5	(i) W	yes	1	2	142	142	0	0	0	Edit Search
1-ITI10-4I4.L.4a	3	5	(i) De	yes	30	13	0	0	0	0	0	Edit Search
1-ITI10-4I4.L.4b	3	5	(i) De	yes	28	10	0	4	4	4	4	Edit Search
1-ITI10-4I4.L.4c	3	5	(i) De	yes	17	7	0	4	0	0	0	Edit Search
1-ITI10-4I4.L.5c	3	5	(i) De	yes	24	12	0	3	0	0	0	Edit Search
1-ITI10-4I4.L.6	3	5	(i) Acquire and use accurately grade-appropriate general	yes	22	5	0	3	0	0	0	Edit Search
1-ITI10-4I4.RI.4	3	5	(i) Determine the meaning of general academic and doma	yes	21	6	0	4	4	4	4	Edit Search
1-ITI10-5	2	5	(i) WORD MEANINGS: Determine intended or precise meani	yes	15	3	0	2	0	0	0	Edit Search
1-ITI10-5I5.L.4a	3	5	(i) Determine or clarify the meaning of unknown and mu	yes	15	3	0	2	0	0	0	Edit Search

Figure 21. Screen 10-2: Item Pools (Segment Pool)

Switching from Standards to Socks in the button bar allows you to see the item pool summary of items allocated to SOCKs for the master test or to child test segments. This screen shows the master item pool SOCK view and not the segment SOCK view. No item searches are allowed in the SOCKs view to expand the item pool. Rather, the item pool SOCK assignment can be pruned by selecting the Edit feature for a SOCK.

This screen allocates the test item pool to SOCKS at the test level because the master test and not a test segment is selected.

No Search features are available for SOCKS. Items can only be searched and added to the master test item pool for content standards.

Switching the Item Pool view from Standards to Socks allows the user to adjust the distribution of items that belong to SOCK categories. SOCKs, like test segments, cannot add any additional items that aren't already selected by the master test item pool.

The allocation of items to SOCKs for tests and test segments works similarly to the allocation of segment items to content standards.

1. The item pool is determined by item searches on the content standards
2. Items from the master item pool are available for allocation to SOCKs
3. Allocation of items to SOCKs for test segments are a subset of items allocated to SOCKs for the master test

Knowledge Category	Description	Active	Test OP Items	Test FT Items	Action
Depth of Knowledge 1	Depth of Knowledge level 1 corresponds to items of least difficulty on a scale from 1 to 3	yes	80	80	Edit
Depth of Knowledge 2	Depth of Knowledge level 2 corresponds to items of medium difficulty on a scale from 1 to 3	yes	127	127	Edit
Depth of Knowledge 3	Depth of Knowledge level 3 corresponds to items of most difficulty on a scale from 1 to 3	yes	105	105	Edit

Figure 22. Screen 10-3: Item Pools (SOCKs)

This screen is similar to the previous screen, but a segment is selected rather than the master test. The view changes to add a column to see the item allocation to SOCKs for the master test and the selected test segment.

The screenshot shows the Smarter Balanced Test Authoring interface. At the top, there's a navigation bar with links for Home, Test, Segment, Blueprint, Item Pool, Form, Level, and Scoring. The user is logged in as John Q. Testauthor. The main title is "Test Authoring". Below the navigation, there's a breadcrumb trail: Home > Item Pools > Standards > SOCKS. A search bar shows "Item Search" and "Pool/Segment Items: 312/198 OP 42/23 FT Segment: SBAC-ELA-5-2014-2015-S-S1".

Knowledge Category	Description	Active	Test OP Items	Segment OP Items	Test FT Items	Segment FT Items	Action
Depth of Knowledge	Depth of Knowledge level 1 corresponds to items of least difficulty	yes	80	65	13	8	Edit
Depth of Knowledge	Depth of Knowledge level 2 corresponds to items of medium difficulty	yes	127	61	15	6	Edit
Depth of Knowledge	Depth of Knowledge level 3 corresponds to items of most difficulty	yes	105	72	14	9	Edit

Annotations on the screen:

- A yellow callout box points to the "Edit" link in the first row: "This screen allocates the segment item pool to SOCKs at the test and segment level because a test segment is selected".
- A yellow callout box points to the "Edit" link in the third row: "Edit is available for selection of items from the master item pool to SOCKs".
- A yellow callout box points to the "Edit" link in the fourth row: "Switching to Segment view allows the user to see the allocation of items to SOCKs for the master test and segment, and change it for this segment".
- A large yellow callout box covers the bottom section of the table: "Switching the Item Pool view from Standards to Socks allows the user to adjust the distribution of items that belong to SOCK categories. SOCKs, like test segments, cannot add any additional items that aren't already selected by the master test item pool. The allocation of items to SOCKs for tests and test segments works similarly to the allocation of segment items to content standards.
1. The item pool is determined by item searches on the content standards
2. Items from the master item pool are available for allocation to SOCKs
3. Allocation of items to SOCKs for test segments are a subset of items allocated to SOCKs for the master test".

Figure 23. Screen 10-4: Item Pools (SOCKs, Test Segment)

This view shows a search for items for the Item Pool in progress. The search dialog shown here allows the user to search on additional item metadata than the base search for content standards. A dialog is shown warning the user that searching at a higher level can invalidate prior item pool pruning at lower levels.

Smarter Balanced Assessment Consortium

Home Test Segment Blueprint Item Pool Form Level

Logged on as John Q. Testauthor | Logout

Test Authoring

Item Search

Standard: 1-ITI10-414.L.5c | All Standards

Attribute 1: between 10 and 20 AND

Attribute 2: equal to foo AND

Attribute 3: less than 4.2 AND

select AND

select AND

OR

Select item property: Attribute 1, Attribute 2, ... Attribute n

Select operation: between, not between, equal to, not equal to, greater than, less than, greater than or equal to, less than or equal to

Search for Items

All searches are stored so that the next time that the user searches all standards or a single standard, the search dialog pre-populates with the parameters of the last search.

Item searches for all standards or for a given standard work as follows:
Standard AND (five expressions) OR (five expressions)

Test SBAC-ELA-5-2014-2015-S

FT	BP FT Max	Test FT Items	Action
4	7	312	42
1	2	19	1
1	2	30	4
1	2	30	13
1	2	30	1
1	2	30	1
1	2	30	4
1	2	30	4
1	2	30	3
1	2	30	22
1	2	30	4
1	2	30	4
1	2	30	3
1	2	30	4
1	2	30	3
1	2	30	4
1	2	30	2
1	2	30	2

! You are searching for items for a higher-level standard. This may change the item pool for standards at a lower level. Please be sure that you understand the risks before searching and that you intend to change the item pool for lower-level standards as well.

Don't show me this warning again

Go Ahead and Search Cancel

The screenshot shows the Smarter Balanced Test Authoring interface. At the top, there's a navigation bar with links for Home, Test, Segment, Blueprint, Item Pool, Form, Level, and Help. The user is logged on as John Q. Testauthor. The main area is titled "Test Authoring" with a gear icon. A yellow callout box contains instructions about item searches for all standards or a given standard work as follows: Standard AND (five expressions) OR (five expressions). Below this, another callout box states that all searches are stored so that the next time the user searches all standards or a single standard, the search dialog pre-populates with the parameters of the last search.

The central part of the screen is the "Item Search" dialog. It displays a search query: "Standard: 1-ITI10-414.L.5c I All Standards". The search interface includes fields for Attribute 1 through Attribute 3, each with dropdown menus for operators like "between", "equal to", "less than", and "select". There are also sections for "OR" and "Select item property". A "Search for Items" button is at the bottom of the dialog.

To the right of the search dialog is a table listing items from a pool. The columns include: Item ID, Standard, BP FT Min, BP FT Max, Test FT Items, and Action. A warning message is overlaid on the table, stating: "You are searching for items for a higher-level standard. This may change the item pool for standards at a lower level. Please be sure that you understand the risks before searching and that you intend to change the item pool for lower-level standards as well." It includes a checkbox for "Don't show me this warning again" and two buttons: "Go Ahead and Search" and "Cancel".

Figure 24. Screen 10-5: Item Pools (Search Dialog)

This screen shows the Edit Item Pool view. Individual items are shown, and their ownership by blueprint content standards is indicated in the first column. The user is able to delete an item, change various attributes or see more detail about an item. The user can choose to view items or passages.

Smarter Balanced Assessment Consortium

Home Test Segment Blueprint Item Pool Form Level Scoring

Test Authoring

Logged on as John Q. Testauthor Logout

Home > Edit Item Pools

Standards Key	Item ID	Passage ID	Type	OP/FT	Required	Active	Action
1-IT10-1	367268	807120	MC	OP	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Delete More
1-IT10-1	249122	807120	MC	OP	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Delete More
1-IT10-1	482524	807120	MC	OP	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Delete More
1-IT10-1	369007	807120	MC	OP	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Delete More
1-IT10-1	2775			OP	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Delete More
1-IT10-1	1258			OP	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Delete More
1-IT10-1	2042			OP	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Delete More
1-IT10-1	2072			OP	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Delete More
1-IT10-1	2108			OP	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Delete More
1-IT10-1	3679			OP	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Delete More
1-IT10-1	3636			OP	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Delete More
1-IT10-1	490195	663805	MC	Exercising regularly			
1-IT10-1	336183	585367	GI	Are you eating right?			
1-IT10-1	362746	585367	MC	Whistle while you stroll			
1-IT10-1	344739	585367	MC	A beautiful fall day...			
1-IT10-1	221046	585367	MC	Click into the OP/FT field to change an item's status. This will change the item's status in the master pool and all child segment pools.			
1-IT10-1	277977	585367	MC	In the Court of the Crimson King			
1-IT10-1	352606	585367	MC				

Changes made to an item pool can be saved or cancelled

Save Cancel

Click Delete to remove an item from the item pool. This will delete the item from the master test item pool as well as any child segment pools.

Click More to see more details about an item.

An item can be inactivated without deleting it from the item pool. This will deactivate the item in the master test pool and all child segment pools.

An item can be made required or not.

Figure 25. Screen 10-6: Edit Item Pools

This view shows the previous screen, but a segment is selected instead of a master test. The item OP/FT, Required and Active status for a segment is not available for segments. Items cannot be deleted from a segment view, only deselected from the segment.

The screenshot shows the Smarter Balanced Test Authoring interface. The top navigation bar includes links for Home, Test, Segment, Blueprint, Item Pool, Form, Level, and Scoring. The current page is 'Edit Item Pools' under 'Segment'. A search/filter section allows filtering by Standards Key, Item ID, Passage ID, Type, and a dropdown for 'Show All Standards'. The main content area displays a table of items, each with a 'More' link. Three yellow callout boxes highlight specific restrictions:

- Item pool status cannot be changed at the segment level, only at the master pool level. It is read-only at the segment level.**
- Items cannot be deleted from the pool at the segment level, only selected or deselected from the segment pool.**
- No Delete links are available at the segment level.**

Standards Key	Item ID	Passage ID	Type	Description	OP/FT	Required	Active	In Segment Pool	Action
1-IT10-1	367268	807120	MC	Love to read	OP	Yes	Yes	<input type="checkbox"/>	More
1-IT10-1	249122	807120	MC	Hiding in plain sight	OP	Yes	Yes	<input type="checkbox"/>	More
1-IT10-1	482524	807120	MC		OP	Yes	Yes	<input checked="" type="checkbox"/>	More
1-IT10-1	369007	807120	MC		OP	Yes	Yes	<input checked="" type="checkbox"/>	More
1-IT10-1	27756			Switching to a test segment from a master test shows you the segment pool rather than the master item pool. Item status such as OP/FT, Required and Active cannot be changed at the segment pool level. Items can be deleted from a segment pool, but this does not delete the whole	OP	Yes	Yes	<input checked="" type="checkbox"/>	More
1-IT10-1	12582				OP	Yes	Yes	<input type="checkbox"/>	More
1-IT10-1	20422				OP	Yes	Yes	<input type="checkbox"/>	More
1-IT10-1	20726				OP	Yes	Yes	<input type="checkbox"/>	More
1-IT10-1	21080				OP	Yes	Yes	<input type="checkbox"/>	More
1-IT10-1	36747				OP	Yes	Yes	<input checked="" type="checkbox"/>	More
1-IT10-1	35368				OP	Yes	Yes	<input type="checkbox"/>	More
1-IT10-1	490195	663805	MC	Exercising regularly	FT	Yes	Yes	<input checked="" type="checkbox"/>	More
1-IT10-1	336183	585367	GI	Are you eating right?	FT	Yes	Yes	<input type="checkbox"/>	More
1-IT10-1	362746	585367	MC	Whistle while you stroll	FT	Yes	Yes	<input checked="" type="checkbox"/>	More
1-IT10-1	344739	585367	MC	A beautiful fall day	FT	Yes	Yes	<input type="checkbox"/>	More
1-IT10-1	221046	585367	MC	Summer flowers	FT	Yes	Yes	<input type="checkbox"/>	More
1-IT10-1	277977	585367	MC	A day at the aquarium	FT	Yes	Yes	<input type="checkbox"/>	More

[Save](#) [Cancel](#)

Figure 26. Screen 10-7: Edit Item Pools (Segment)

This view shows a user switching to SOCKs view and selecting to see more detail about an individual item.

The screenshot shows the Smarter Balanced Test Authoring interface. At the top, there's a navigation bar with links for Home, Test, Segment, Blueprint, Item Pool, Form, Level, and Scoring. The user is logged in as John Q. Testauthor. The main area is titled "Test Authoring" and shows a table of items in a "SOCK" pool. The table has columns for Item ID, Passage ID, Filter, OP/FT, Required, Active, and Action. A yellow callout box points to the "More" link in the Action column for item ID 367268, with the text "Clicking More puts up a dialog with item details". Below the table, a modal window titled "Item Details" displays the following information for item ID 367268:

Item ID	367268
Description	Jane Finds a Bug
Subject	English Language Arts
Type	Multiple Choice
Version	22
Grade	7
Minimum Grade	6
Maximum Grade	8
Standard	1-IT10-1
Answer Key	B
Braille	BRF
Human Scored	Yes
Difficulty	High
Performance Task	Classroom-based

At the bottom of the modal, there are "Save" and "Cancel" buttons.

Figure 27. Screen 10-8: Edit Item Pools (SOCKs, More Item Details)

This view shows the user switching the view from Items to Passages. Please note than unlike items, passages cannot be manually deleted from the item pool. They are present because an item in the item pool references the passage. This screen is for informational use only.

The screenshot shows the Smarter Balanced Test Authoring interface. At the top, there is a navigation bar with links for Home, Test, Segment, Blueprint, Item Pool, Form, Level, and Scoring. On the right side of the header, it says "Logged on as John Q. Testauthor" and "Logout". The main title "Test Authoring" is displayed with a gear icon. Below the header, there is a breadcrumb trail "Home > Edit Item Pools" followed by tabs for "Item" (selected), "Passage" (highlighted in blue), and "Filter". A dropdown menu "Show All Standards" is open, displaying "1-IT" and three sub-options: "1-IT10-1", "1-IT-1012", and "1-IT-1013". To the right of this is a search bar "Pool/Filter: 312/115" and a "Test" dropdown set to "SBAC-ELA-5-2014-2015-S". The main content area contains a table with columns "Standards Key", "Passage ID", and "Action". The table has four rows:

Standards Key	Passage ID	Action
1-IT	No items	
1-IT10-1	807120	More
1-IT10-1	663805	More
1-IT10-1	585367	More

A yellow callout box is overlaid on the table, containing the following text:

Switching to Passage allows the user to see passages associated with items. Unlike items, passages cannot be independently deleted, made inactive or made required. Passages are pulled into an assessment whenever a corresponding item is selected for presentation. Therefore, this screen only shows you a list of passages and allows you to see details about each one.

At the bottom of the screen are two buttons: "Save" and "Cancel".

Figure 28. Screen 10-9: Item Pools (Passages)

The screenshot shows the Smarter Balanced Test Authoring interface. At the top, there is a navigation bar with links for Home, Test, Segment, Blueprint, Item Pool, Form, Level, and Scoring. On the right side of the header, it says "Logged on as John Q. Testauthor" and "Logout". Below the header is a search bar with dropdowns for "Select Grade", "Select Subject", "Select Test", and a button for "Add Test Form".

The main area displays a table titled "Test Form" with columns: Test Form, Parent Test, Number of Partitions, Grade, Subject, and Action (Edit Delete). The table lists various test forms, such as F1-ENU through F6-ENU and F1-ESN, each associated with a specific parent test like SBAC-MA-3-2014-2015-S.

A yellow callout box with a red arrow points from the "Edit Test Form" button in the table to the "Edit Test Form" dialog box. The dialog box has fields for "Parent Test" (set to SBAC-ELA-3-2014-2015-S) and "Form Language" (with options English, Spanish, Hawaiian, Braille-English, and Braille-Spanish). It also contains "Save" and "Cancel" buttons.

Two arrows point from the "Add Test Form" button in the table to the "Add Test Form" dialog box. This dialog box has fields for "Parent Test" (dropdown menu showing "Select Parent Test" and options SBAC-MA-4-2014-2015-S and SBAC-ELA-4-2014-2015-S) and "Form Language" (dropdown menu showing "Select Form Language" and options English, Spanish, Hawaiian, Braille-English, and Braille-Spanish). It also contains "Save and Add Another", "Save and Exit", and "Cancel" buttons.

Test Form	Parent Test	Number of Partitions	Grade	Subject	Action
F1-ENU	SBAC-MA-3-2014-2015-S	2	3	Mathematics	Edit Delete
F2-ENU	SBAC-MA-3-2014-2015-S	2	3	Mathematics	Edit Delete
F3-ENU	SBAC-MA-3-2014-2015-S	2	3	Mathematics	Edit Delete
F4-ENU	SBAC-MA-3-2014-2015-S	2	3	Mathematics	Edit Delete
F5-ENU	SBAC-MA-3-2014-2015-S	2	3	Mathematics	Edit Delete
F6-ENU	SBAC-MA-3-2014-2015-S	1	3	Mathematics	Edit Delete
F1-ESN	SBAC-MA-3-2014-2015-S	1	3	Mathematics	Edit Delete
F2-ESN					
F1-ESN					
F3-ESN					
F1-BR-E					
F2-BR-E					
F1-ENU					
F2-ENU					
F3-ENU					
F4-ENU					
F5-ENU					
F6-ENU					
F1-ESN	SBAC-MA-4-2014-2015-S	1	3	Mathematics	Edit Delete

Figure 29. Screen 11: Test Forms

The screenshot shows the Smarter Balanced Test Authoring interface. At the top, there are navigation links: Home, Test, Segment, Blueprint, Item Pool, Form, Level, Scoring. Below this is a breadcrumb trail: Home > Form Partitions. There are four dropdown menus: Select Grade, Select Subject, Select Test, and Select Form. An 'Add Form Partition' button is located to the right of these dropdowns. The main area contains a table of 'Test Form' and 'Partition' pairs. A modal window titled 'Add Form Partition' is overlaid on the table. This modal has four dropdowns: 'Grade' (with options 3 and 4), 'Subject' (with options 'Select Parent Test', 'English Language Arts', and 'Mathematics'), 'Test' (with options 'Select Parent Test', 'SBAC-MA-3-2014-2015-S', and 'SBAC-ELA-3-2014-2015-S'), and 'Form' (with options 'Select Form', 'F1-ENU', and 'F2-ENU'). At the bottom of the modal are three buttons: 'Save and Add Another', 'Save and Exit', and 'Cancel'. To the right of the table, there is a column of 'Action' buttons labeled 'Up Down Edit Del' repeated multiple times. Two yellow callout boxes with black borders provide instructions: one pointing to the 'Edit' button in the 'Action' column, stating 'Filters allow the user to reduce the number of forms in the dropdown, but are not case sensitive'; and another pointing to the 'Edit' button in the 'Action' column of the modal, stating 'Clicking Edit takes you to the Edit Form Partitions screen'.

Figure 30. Screen 12-1: Form Partitions

Once a form partition is created, Edit Form Partitions allow the user to select items on a form, sequence them, change their form attributes and see related passages.

Smarter
Balanced
Assessment Consortium

Home Test Segment Blueprint Item Pool Form Level Scoring

Test Authoring

Logged on as John Q. Testauthor Logout

Home > Edit Form Partitions Items Passages Search for Items Form SBAC-ELA-3-2014-2015-S-F-ENU-12 Partition 2

Item ID	Sequence	Admin Required	Response Required	Field Test	Active on Form	Block ID	Action
150-15869	1	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	A	Up Down Delete
150-12310	2	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	A	Up Down Delete
150-12225	3	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	A	Up Down Delete
150-11927	4	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	A	Up Down Delete
150-13490	5	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	A	Up Down Delete
150-12044	6	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	A	Up Down Delete
150-12197	7	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	A	Up Down Delete
150-12349	8	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	A	Up Down Delete
150-12304	9	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	A	Up Down Delete
150-12033	10	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	A	Up Down Delete
150-8982	11	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	A	Up Down Delete
150-12355	12	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	A	Up Down Delete
150-11466	13	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	A	Up Down Delete
150-12571	14	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	A	Up Down Delete
150-12252	15	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	A	Up Down Delete
150-12296	16	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	A	Up Down Delete
150-12292	17	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	A	Up Down Delete

Save Cancel

Figure 31. Screen 12-2: Edit Form Partitions

Smarter
Balanced
Assessment Consortium

Test Authoring

Home Test Segment Blueprint Filters allow selection of a test - segments and forms follow

Home > Edit Form Partitions

Grade: 4 Subject: English Language Arts Test: SBAC-MA-3-2014-2015-S

Form	SBAC-MA-3-2014-2015-S-S1	SBAC-MA-3-2014-2015-S-S2	SBAC-MA-3-2014-2015-S-S3
F1-ENU	1	2	3
F2-ENU	2	3	1
F3-ENU	2	3	1
F4-ENU	2	3	1
F5-ENU	2	3	1
F6-ENU	2	3	1
F1-ESN	3	1	2
F2-ESN	3	1	2
F1-ESN	2	3	1
F3-ESN	2	3	1
F1-BR-ENU	2	3	1
F2-BR-ENU	1	2	1
F1-ENU	3	1	2
F2-ENU	3	1	2
F3-ENU	3	1	2
F4-ENU	3	1	2
F5-ENU	2	3	1
F6-ENU	2	3	1

Save Cancel

Figure 32. Screen 13: Assign Form Partitions to Test Segments

The screenshot shows the Smarter Balanced Test Authoring interface. On the left, there is a table titled "Scoring Rules" with columns: Order, Computation Rule, Version, Label, BP Ref Type, BP Reference, and Action. The table contains 19 rows of data. On the right, there is a modal dialog titled "Add Computation Rule" with fields for Label, Computation Rule, Version, BP Reference Type, BP Reference, cutscores, scaleScoreName, and buttons for Save and Add Another, Save and Exit, and Cancel.

Order	Computation Rule	Version	Label	BP Ref Type	BP Reference	Action
1	ThetaScore	1.0	ThetaScore	Standard	ELA-2	Up Down Edit Del
2	ItemCount	1.0	ItemCount	Standard	ELA-2	Up Down Edit Del
3	RawScore	1.0	RawScore	Standard	ELA-2	Up Down Edit Del
4	ItemCountScored	1.0	ItemCountScored	Standard	ELA-2	Up Down Edit Del
5	OnGradeRawScore	1.0	OnGradeRawScore	Standard	ELA-2	Up Down Edit Del
6	ScaleScore	2.0	ScaleScore	Standard	ELA-2	Up Down Edit Del
7	OnGradeItemCountScored	1.1	OnGradeItemCountScored	Standard	ELA-2	Up Down Edit Del
8	GLEDelta	1.3	BenchmarkScore	Standard	ELA-2I2.1	Up Down Edit Del
9	GLEDelta	1.3	BenchmarkScore	Standard	ELA-2I2.2a	Up Down Edit Del
10	GLEDelta	1.3	BenchmarkScore	Standard	ELA-2I2.3a	Up Down Edit Del
11	GLEDelta	1.3	BenchmarkScore	Standard	ELA-2I2.3c	Up Down Edit Del
12	GLEDelta	1.3	BenchmarkScore	Standard	ELA-2I2.4a	Up Down Edit Del
13	GLEDelta	1.3	BenchmarkScore	Standard	ELA-2I2.4b/T	Up Down Edit Del
14	GLEDelta	1.3	BenchmarkScore	Standard	ELA-2I2.4bL	Up Down Edit Del
15	GLEDelta	1.3	BenchmarkScore	Standard	ELA-2I2.4c	Up Down Edit Del
16	GLEDelta	1.3	BenchmarkScore	Standard	ELA-2I2.4d	Up Down Edit Del
17	GLEDelta	1.3	BenchmarkScore	Standard	ELA-2I2.4e	Up Down Edit Del
18	GLEDelta	1.3	BenchmarkScore	Standard	ELA-2I2.4f	Up Down Edit Del
19	GLEDelta	1.3	BenchmarkScore	Standard	ELA-2I2.4g	Up Down Edit Del

Add Computation Rule

Parent Test: SBAC-MA-4-2014-2015-S

Label: DSTPProficiency

Computation Rule: DelawarePerformanceLevels

Select Computation Rule Type: ThetaScore, ItemCount, RowScore, ItemCountScored

Version: 1.0

BP Reference Type: Test

Select BP Reference Type: Segment, Content Standard, SOCK

BP Reference: (Delaware)OCAS-Reading-4-Fall-2012-2013

Select BP Reference

cutscores:

Index	Value
1	0
2	606
3	653
4	769
5	815

scaleScoreName: AccountabilityScore

Buttons: Save and Add Another, Save and Exit, Cancel

Figure 33. Screen 14: Scoring Rules

The screenshot shows the Smarter Balanced Test Authoring interface. At the top, there is a navigation bar with links for Home, Test Segment, Blueprint, Item Pool, Form, Level, and Scoring. The user is logged in as John Q. Testauthor. On the right, there is a "Test Authoring" button with a gear icon.

The main content area displays a table titled "Performance Levels". The table has columns for "Level", "BP Reference Type", "BP Reference", "Scaled Lo", "Scaled Hi", and "Action". There are four rows in the table:

Level	BP Reference Type	BP Reference	Scaled Lo	Scaled Hi	Action
1	Test	SBAC-ELA-5-2014-2015-S	0.00	700.00	Up Down Edit Del
2	Test	SBAC-ELA-5-2014-2015-S	700.00	1,400.00	Up Down Edit Del
3	Test	SBAC-ELA-5-2014-2015-S	1,400.00	2,100.00	Up Down Edit Del
4	Test	SBAC-ELA-5-2014-2015-S	2,100.00	2,800.00	Up Down Edit Del

Two modal windows are overlaid on the page:

- Add Performance Level**: A modal window for adding a new performance level. It includes fields for "Parent Test" (set to SBAC-MA-4-2014-2015-S), "BP Reference Type" (dropdown menu showing "Test", "Segment", "Content Standard", "SOCK"), "BP Reference" (dropdown menu showing "SBAC-ELA-5-2014-2015-S"), "Scaled Lo" (text input field), "Scaled Hi" (text input field), and three buttons: "Save and Add Another", "Save and Exit", and "Cancel".
- Edit Performance Level**: A modal window for editing an existing performance level. It includes fields for "Parent Test" (set to SBAC-MA-4-2014-2015-S), "BP Reference Type" (dropdown menu showing "Test", "Segment", "Content Standard", "SOCK"), "BP Reference" (dropdown menu showing "SBAC-ELA-5-2014-2015-S"), "Scaled Lo" (text input field set to 700.00), "Scaled Hi" (text input field set to 1,400.00), and two buttons: "Save" and "Cancel".

Arrows from the "Add Performance Level" and "Edit Performance Level" buttons in the main table point to their respective modal windows.

Figure 34. Screen 16: Performance Levels

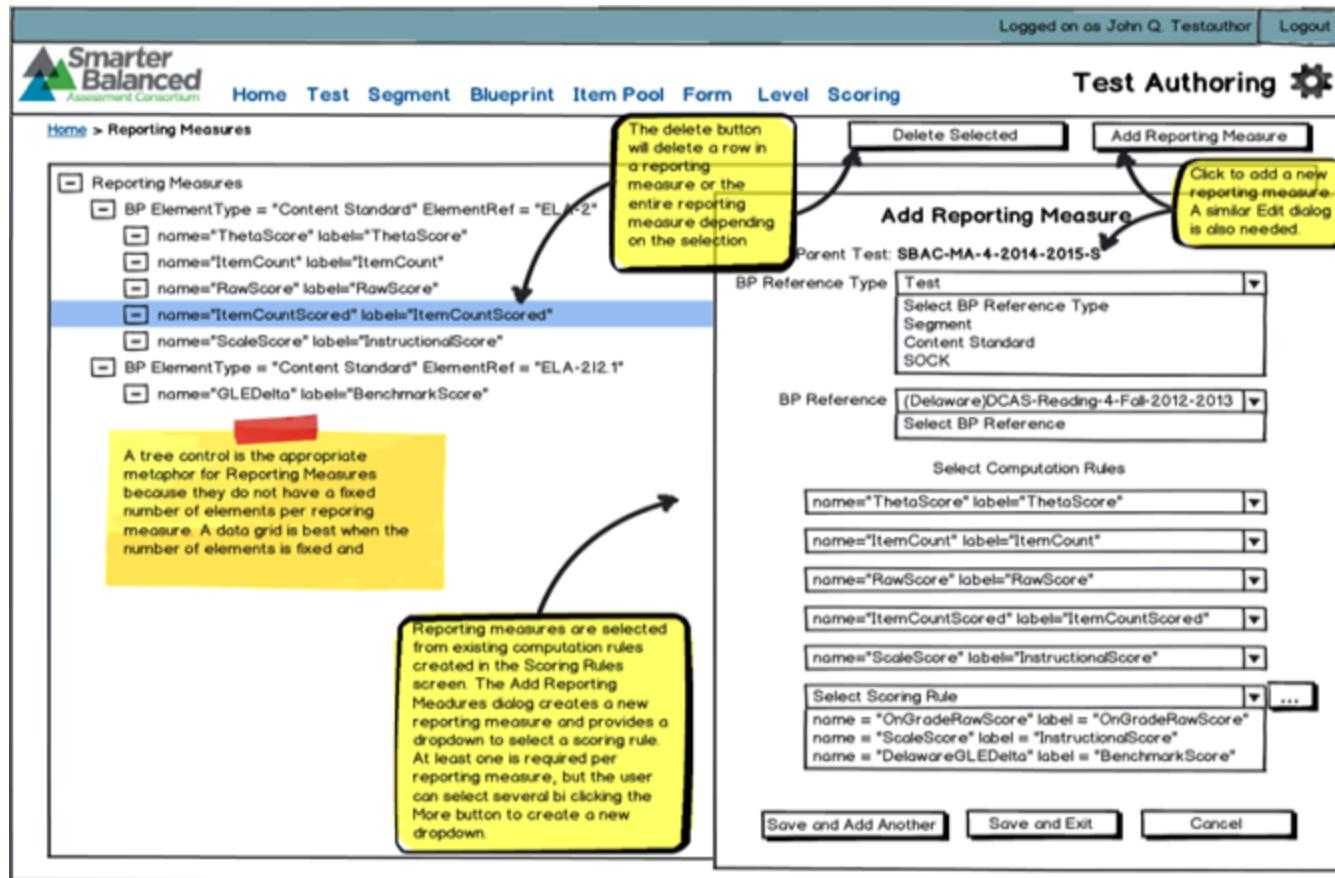


Figure 35. Screen 17: Reporting Measures

The following are areas that require follow-up and more investigation

Item	Concept
Affinity	Currently organized under SOCKs, there are no explicit affinity groups. There may be a need for explicit affinity groups to control certain aspects of the adaptive

Level II Requirements: Test Authoring

Groups	algorithm, but there is no DTD element to do this. We will keep things this way for now but follow-up is needed.
Enemy list	Enemy lists are organized at the item pool level, but no interface is defined. More clarity is needed about item groups before enemy lists can be defined.
Item groups	It isn't clear whether itemgroups are explicit entities managed by users at the user interface or hidden elements organized by the application according to implicit groupings such as item passage references. More clarity is needed.
Conversion tables	Additional information is needed about conversion table references before a user interface can be defined
Test Packaging	This document will be expanded with the mappings between UI elements and XML fields in the test specifications