Smarter Balanced Adaptive Algorithm – Summary and Explanation Version 5.2.1

This paper describes the adaptive algorithm design for the Smarter Balanced Test Delivery System. Whether states and their service providers use the open source software, or have a different engine certified to deliver Smarter Balanced assessments, they must adopt an algorithm that delivers the published blueprint. Three potential scenarios through which this could be accomplished are listed below:

- States may deliver Smarter Balanced assessments using the open source software for both the test delivery system and adaptive algorithm.
- States may use the open source software for one component and a service provider solution for the other (e.g., open source test delivery system, and a vendor's algorithm that can be appropriately configured).
- States may use service provider solutions for both components, provided that in concert, they can deliver the blueprint as expected.

This document describes the method used in the Smarter Balanced system to satisfy the blueprint and provide optimal precision.. The implementation described here is released under the Creative Commons Attribution Only, No Derivatives license. This document is a summary with supplemental explanations and examples of explicit functionality found in the separate, *Smarter Balanced Adaptive Item Selection Algorithm Design Report (version 3 5/9/2014)* by Jon Cohen and Larry Albright. Interested readers can refer to the more detailed document for more technical information and specific formulas the algorithm employs. In this document, many features are described as "configurable". This means that the software can deliver a variety of test types that may be used in the future or may be used by member states for their local tests. Configuration of features for Smarter summative and interim tests will be set by the consortium. Detailed information about configuration values can be found in the Cohen and Albright paper.

In general, an adaptive algorithm is the method used to carry out a blueprint design by acting on an item pool. The algorithm finds the items expected to comprise the best test for each student, selecting items from the pool that match blueprint demands while using information from student responses to find the most accurate score. The blueprint describes in detail the content and other attributes for each student's test.. In Smarter tests, data describing content is based on the hierarchical structure of the test (e.g., the blueprint) and on substantive attributes of test questions (e.g., Depth of Knowledge). A table showing the general, Smarter Balanced test structure is presented below, to orient readers of this paper on the types of considerations both the algorithm and items in the pool must support in order to deliver an accurate, efficient test.

General Test Structure:

Structure Level	Examples	
Subject	ELA	Math
Grade	6	4
Claim/Reporting Category	Reading	Concepts & procedures
Sub-category	Informational text	Priority cluster
Target	Reasoning and evaluation	Extend understanding of fractional equivalence and ordering

Other item attributes specified in blueprints and needed to run the algorithm include depth of knowledge, response type, scoring type, common stimulus membership and mathematical domain. All items in the bank must have complete information about these elements available to the algorithm software, in order to be considered eligible for test administration. The minimum and maximum number of items in each element is specified in the adaptive software, serving as a constraint to balance aspects such as blueprint coverage with test length. Each element can be given weights used in the selection process that will affect test delivery. By allowing users to specify weights, the general algorithm can be customized for varying conditions of population and pool distribution. This function can help assure that a test best matches the purpose for which it is designed. For example, weights can be shifted to emphasize measurement precision or content coverage, depending on policy priorities. Final weights are usually established during the last stages of test design when all item parameters are known and simulations can be run. In the case of Smarter Balanced, this will occur as part of the achievement level setting process, using data from the spring 2014 field test.

Item measurement data: In addition to the blueprint attributes listed above, each item has a set of parameters that provide measurement information. The purpose of the algorithm is to satisfy the content blueprint while providing the most accurate student score, in the most efficient manner. In measurement terms, the most information is obtained when the difficulty of the item is close to the functional level of the student. At the beginning of the test, item difficulty and discriminating power are known, and student ability is unknown. The job of the algorithm is to find out the student's ability in the content area being assessed..

Test operation walkthrough

Preparation: The system must have in place a sufficient item pool with the full set of parameters and metadata. Smarter pools contain all items for the intended grade level and items from adjacent grades that address on-grade content. Items from upper grades address content the student has had an opportunity to learn. Items from lower grades are screened for age-appropriateness. Initially, the pool is filtered to contain only items written for the intended grade. Under certain circumstances (described below) the filter is dropped and the adjacent grade items are added. The adaptive engine needs to be

populated with all hierarchical and content elements, including the minimum and maximum number of items allowed for each facet of the blueprint.

Initialization. Adaptive tests require methods for avoiding overuse of items. If the same initial item is used for all or most students, security issues arise as students share their experiences. Consequently operational and summative tests have been configured to choose each test's initial item from a set of items with medium difficulty relative to the population of the grade. The initial claim can be chosen at random as long as passages and hand-scored items are not presented first. The algorithm then cycles through the claims.

Item selection. The initialization and selection processes control underuse and overuse of items, also known as exposure control. Exposure control enhances item security, discouraging copying and cheating by presenting a variety of items. It also leads to more efficient pool use, assuring that all items developed to cover the content are used. Rather than choosing the single best item for initialization and selection, which would cause some items to be used repeatedly and others rarely or never, the algorithm selects randomly from targeted sets of items. To prevent overuse of highly discriminating items, the discrimination (a) parameter is not taken into account in selection ranking. The optimal size of the first content-based set and the subsequent subset, which takes information into account, will be determined through simulation with actual pool parameters.

Once the initial item response has been given, the selection process is launched and will be repeated for every subsequent response. The software uses the set of weights described earlier to determine a group of items with the best match to the blueprint, excluding items from categories that have reached maximum n and items previously seen by the examinee. When this mini pool has been chosen, information value is calculated for each item using the current student ability estimate and known item parameters. Overall item value is calculated using both information and content data. The item set is then sorted according to overall value and a set of the most preferred items are identified. The item to be administered is chosen randomly from within this set. After each response, the student ability estimate is updated and the selection procedure is repeated until the blueprint has been satisfied.

The algorithm proceeds in this manner a percentage of the test has been administered, sampling items from all claim areas. (Coverage in mathematics, 61%; ELA, 62%.) At this point the distance of the estimated score from the college content readiness cut score is evaluated. This is Level 3 as defined in the Achievement Level Descriptor document (link).If there is a determination that the student is in either Level 1 or Level 4 as defined by the Achievement Level Setting Report, the item pool is expanded to include items from no more than two adjacent grades in either direction. In grade 3, the expansion includes items from adjacent upper grades only; in grade 11 only adjacent lower grades are included. Items from adjacent grades have been screened for appropriateness by content experts to assure that they are instructionally and developmentally appropriate for the target grade. For the remainder of the test, both on-grade and off-grade items can be administered. The item with the best content and measurement characteristics is chosen from the pool. When a determination of being in Level 1 or level

4 cannot be made, the test continues with on-grade items. The algorithm delivers the remainder of the blueprint until termination.

Termination: The engine allows users to choose termination conditions for different purposes. Current test designs call for ending the test when the blueprint has been met. It is also possible to have the test end when scores are sufficiently precise, a condition often used in adaptive mastery or graduation tests. In either case, the item pool must be sufficiently robust to support the desired termination criteria. Therefore, any decisions about altering termination rules to support different purposes must begin with careful examination of both the blueprint and corresponding item pool.

Scoring-Results for the overall test are determined after all human-scored items and PT responses are collected and merged together. Student responses and item parameters provide data for maximum likelihood estimation of overall and claim scores. During the adaptive portion of the test, a running estimate of the student score is kept using a Bayesian technique that makes use of prior information to predict the most informational items. Students can go back and change their answers within a test segment. When this occurs, the ongoing student score estimate is updated with the new response.