

1 TITLE

2 Adaptive Engine Logic Used in Training Academic Proficiency

3

4 CLAIM OF PRIORITY/CROSS REFERENCE OF RELATED

5 APPLICATION(S)

6 This application claims the benefit of priority of United States Provisional
7 Application Number 60/459,773, filed April 2, 2003, entitled "Adaptive Engine
8 Logic Used in Training Academic Proficiency," hereby incorporated in its
9 entirety herein.

10

11 COPYRIGHT/TRADEMARK STATEMENT

12 A portion of the disclosure of this patent document may contain material which is
13 subject to copyright and/or trademark protection. The copyright/trademark owner
14 has no objection to the facsimile reproduction by anyone of the patent document
15 or the patent disclosure, as it appears in the Patent Office patent files or records,
16 but otherwise reserves all copyrights and trademarks.

17

18 STATEMENT REGARDING FEDERALLY SPONSORED RESEARCH OR
19 DEVELOPMENT

20 Not applicable.

21

1

2 BACKGROUND

3 1. Field of the Invention

4 The present invention relates generally to computerized learning and more
5 particularly to an adaptive learning system and method that utilizes a set of
6 heuristics to provide a learning environment unique to an individual.

7

8 2. Description of Related Art

9

10 The Problem

11 A child's learning pace varies from child to child. Schools often provide
12 education that is tailored to a general standard, to the "normal" child. Teachers
13 and facilitators often gear materials, e.g. static curriculum, and pedagogical
14 direction toward the majority of the classroom – the so-called normal child – and
15 therefore neglect children with different needs on either end of the spectrum.

16

17 Because the collection of concepts mastered by different students varies, without
18 a personalized curriculum tailored for the student, it is oftentimes difficult to help
19 different students with different abilities to develop a solid foundation in a
20 particular subject.

21

1 Prior Art Solutions to the Problem

2 There are a number of education-based, and more specifically math-based,
3 Internet web sites available today. Also, there are many offline products, such as
4 workbooks, CD-ROMs, and games that also address this issue. In addition there is
5 also traditional human help, such as a teacher and/or tutor.

6

7 Commercial examples in the math arena:

8

9 www.aleks.com - A fully automated online math tutor for K-12 and Higher
10 Education students. Below is an excerpt from their corporate website.

11

12 ALEKS is a revolutionary Internet technology, developed at the University of
13 California by a team of gifted software engineers and cognitive scientists, with the
14 support of a multi-million dollar grant from the National Science Foundation.
15 ALEKS is fundamentally different from previous educational software. At the
16 heart of ALEKS is an artificial intelligence engine -- an adaptive form of
17 computerized intelligence -- which contains a detailed structural model of the
18 multiplicity of the feasible knowledge states in a particular subject. Taking
19 advantage of state of the art software technology, ALEKS is capable of searching
20 an enormous knowledge structure efficiently, and ascertaining the exact
21 knowledge state of the individual student. Like "Deep Blue," the IBM computer

1 system that defeated international Chess Grand master Garry Kasparov, ALEKS
2 interacts with its environment and adapts its output to complex and changing
3 circumstances. ALEKS is based upon path breaking theoretical work in Cognitive
4 Psychology and Applied Mathematics in a field of study called "Knowledge
5 Space Theory." Work in Knowledge Space Theory was begun in the early 1980's
6 by an internationally renowned Professor of Cognitive Sciences who is the
7 Chairman and founder of ALEKS Corporation.

- 8 • Using state-of-the-art computerized intelligence and Web-based
9 programming, ALEKS interacts with each individual student, and functions as
10 an experienced one-on-one tutor.
- 11 • Continuously adapting to the student, ALEKS develops and maintains a
12 precise and comprehensive assessment of your knowledge state.
- 13 • ALEKS always teaches what the individual is most ready to learn.
- 14 • For a small fraction of the cost of a human tutor, ALEKS can be used at any
15 time: 24 hours per day; 7 days per week, for an unlimited number of hours.

16

17 Kumon Math Program- a linear and offline paper-based math program that helps
18 children develop mechanical math skills. 2.5 million students or more worldwide.

19

20 Math Blasters- A CD-ROM that provides some math training through fun games.

21

1 Ms. Lindquist: The Tutor – a web-based math tutor specialized in helping
2 children solving algebraic problems using a set of artificial intelligence
3 algorithms. It was developed by a researcher at Carnegie Mellon University
4
5 Cognitive Tutor – Developed by another researcher at Carnegie Mellon
6 University. It helps students solve various word-based algebraic and geometric
7 problems with real-time feedback as students perform their tasks. The software
8 predicts human behavior, makes recommendations, and tracks student-user
9 performance in real time. The software is sold by Carnegie Learning.

10

11 Limitations of the Prior Art

12 Many internet/web sites do not offer a truly personalized experience. In their
13 systems, each student-user answers the same 10 questions (for example),
14 regardless of whether they answer the first questions correctly or incorrectly.
15 These are examples of non-intelligence, or limited intelligence, backed by a
16 linear, not relational, curriculum.

17

18 Other offline products (like CD-ROMs) have the ability to provide a somewhat
19 personalized path, depending on questions answered correctly or incorrectly, but
20 their number of questions is limited to the storage capacity of the CD-ROM. CD-
21 ROMs and off-line products are also not flexible to real-time changes to content.

CD-ROMs also must be installed on a computer. Some may only work with certain computer types (e.g., Mac or PC), and if the computer breaks, one must reinstall it on another machine, and start all over with the product.

The Present Solution to the Problem

Embodiments of the present invention solve the aforementioned limitations of the prior art. Embodiments of the present invention fill in the gaps of what schools cannot provide— an individualized curriculum that is driven by the child's own learning pace and standards. Embodiments of the invention help each child build a solid foundation in the subject as early as possible, and then move on to more difficult material. Embodiments of the present invention provide an intelligent, adaptive system that takes in information and reacts to the specific information given to it, using a set of predefined heuristics. Therefore, each individual's information (which can and is unique) may feed the engine, and then provide a unique experience to that individual. One embodiment of the present invention discussed herein focuses on Mathematics however the invention is not limited thereby as the same logic can be applied to other academic subjects.

In a first aspect of the present invention, there is provided an adaptive learning system for presenting an appropriate topic and question to a user, said system comprising: a processor configured to: generate and store in a database a set of hierarchical topics having a plurality of questions associated with each one of said topics; each of said plurality of questions within a topic having an assigned

difficulty level value; determine an adjustable state level value for a user based on said user's topic performance consistency; said state level initialized to and having a range of predetermined value; determine an adjustable water level value for said user based on said user's proficiency in at least a subset of said hierarchical topics; said water level initialized to and having a range of predetermined value; determine a relevant topic for said user from said set of hierarchical topics by performing the following: cull said set of hierarchical topics to determine one or more eligible academic topics; and evaluate for relevance said one or more eligible academic topics using heuristic relevance ranking to determine said relevant academic topic; determine an appropriate question for said user from said plurality of relevant academic topic questions by performing the following: determine said user's water level, search said database for one or more questions within a threshold range from said user's water level, randomly select a relevant question from this one or more questions depending on the user's answer to said selected question, adjust said user's water level according to a predetermined adjustment table.

Disclosed herein are unlimited curriculum paths that respond to students' different learning patterns and pace based on a curriculum chart with correlation coefficients and prerequisite information.

Topics are connected with each other based on pre-requisite/post-requisite relationship thus creating a complex 3-D curriculum web. Each relationship is also quantified by a correlation coefficient. Each topic contains a carefully designed set of questions in increasing difficulty levels (e.g., 1-100). Thus, without acquiring a certain percentage of pre-requisites, a student-user may be deemed not ready to go into a specific topic.

The programming for the heuristics and the logic may be done in the Java programming language. In addition, an embodiment of the present invention has been adapted to accept information, via the Internet, using a browser as a client. Furthermore, information is stored in a database, to help optimize the processing of the information.

Certain features and advantages of embodiments of the present invention include: a high level of personalization, continuous programs accessible anytime and anywhere, real-time performance tracking systems that allow users, e.g., parents to track progress information online, a relational curriculum, enabling individualized paths from question to question and from topic to topic, worldwide comparison mechanisms that allow parents to compare child performance against peers in other locations.

1 The above aspects, features and advantages of the present invention will become
2 better understood with regard to the following description.

3

4

5 BRIEF DESCRIPTION OF THE DRAWING(S)

6 Referring briefly to the drawings, embodiments of the present invention will be
7 described with reference to the accompanying drawings in which:

8

9 Figures 1 – 15 depict various aspects and features of the present invention in
10 accordance with the teachings expressed herein.

11

12 DETAILED DESCRIPTION OF THE PRESENT INVENTION

13 Although what follows is a description of a preferred embodiment of the
14 invention, it should be apparent to those skilled in the art that the following is
15 illustrative only and not limiting, having been presented by way of example only.
16 All the features disclosed herein may be replaced by alternative features serving
17 the same purpose, and equivalents of similar purpose, unless expressly stated
18 otherwise. Therefore, numerous other embodiments of the modifications thereof
19 are contemplated as falling within the scope of the present invention. However,
20 all specific details may be replaced with generic ones. Furthermore, well-known

1 features have not been described in detail so as not to obfuscate the principles
2 expressed herein.

3

4 Moreover, the techniques may be implemented in hardware or software, or a
5 combination of the two. In one embodiment, the techniques are implemented in
6 computer programs executing on programmable computers that each include a
7 processor, a storage medium readable by the processor (including volatile and
8 non-volatile memory and/or storage elements), at least one input device and one
9 or more output devices. Program code is applied to data entered using the input
10 device to perform the functions described and to generate output information.
11 The output information is applied to one or more output devices.

12

13 Each program is preferably implemented in a high level procedural or object
14 oriented programming language to communicate with a computer system,
15 however, the programs can be implemented in assembly or machine language, if
16 desired. In any case, the language may be a compiled or interpreted language.

17

18 Each such computer program is preferably stored on a storage medium or device
19 (e.g., CD-ROM, NVRAM, ROM, hard disk, magnetic diskette or carrier wave)
20 that is readable by a general or special purpose programmable computer for
21 configuring and operating the computer when the storage medium or device is

1 read by the computer to perform the procedures described in this document. The
2 system may also be considered to be implemented as a computer-readable storage
3 medium, configured with a computer program, where the storage medium so
4 configured causes a computer to operate in a specific and predefined manner.

5

6 The engine and the algorithms and methodology that it was developed for, is
7 currently specific for Mathematics at this time. But, using the same structure, it
8 can be broadened and used in any numbers of scenarios. The function of the
9 engine is primarily to react on information, or data, given to it. Then, based on a
10 set of rules or governing heuristics, it will react to the data, and provide
11 meaningful output. This ideology can be used in a number of different
12 applications.

13

14 Figures 1 and 2 illustrate exemplary hardware configurations of a processor-
15 controlled system on which the present invention is implemented. One skilled in
16 the art will appreciate that the present invention is not limited by the depicted
17 configuration as the present invention may be implemented on any past, present
18 and future configuration, including for example,
19 workstation/desktop/laptop/handheld configurations, client-server configurations,
20 n-tier configurations, distributed configurations, networked configurations, etc.,
21 having the necessary components for carrying out the principles expressed herein.

1
2 In its most basic embodiment however, Figure 1 depicts a system 700 comprising,
3 but not limited to, a bus 705 that allows for communication among at least one
4 processor 710, at least one memory 715 and at least one storage device 720. The bus
5 705 is also coupled to receive inputs from at least one input device 725 and provide
6 outputs to at least one output device 730. The at least one processor 710 is
7 configured to perform the techniques provided herein, and more particularly, to
8 execute the following exemplary computer program product embodiment of the
9 present invention. Alternatively, the logical functions of the computer program
10 product embodiment may be distributed among processors connected through
11 networks or other communication means used to couple processors. The computer
12 program product also executes under various operating systems, such as versions of
13 Microsoft Windowsä, Apple Macintoshä, UNIX, etc. Additionally, in a preferred
14 embodiment, the present invention makes use of conventional database technology
15 740 such as that found in the commercial product SQL Server®; which is marketed
16 by Microsoft Corporation, to store, among other things, the body of questions.
17 Figures 3A-3C, 4A-4F, 5A-5F, 6A-6F, 7A-7F and 8A-8C illustrate one such order
18 data organization comprising Learning Dimensions, Proficiency Levels, Topics,
19 Questions, etc..
20

1 As shown in Figure 2, in another embodiment, the present invention is
2 implemented as a networked system having at least one client (e.g., desktop,
3 workstation, laptop, handheld, etc) in communication with at least one server
4 (e.g., application, web, and/or database servers, etc.,) via a network, such as the
5 Internet.

6

7 The present invention utilizes a comprehensive curriculum map that outlines
8 relational correlations between distinct base-level categories of mathematical
9 topics, concepts and skill sets.

10

11 The present invention generates an individually tailored curriculum for each user,
12 which is a result of the user's unique progression through the curriculum map, and
13 is dynamically determined in response to the user's ongoing performance and
14 proficiency measurements within each mathematical topic category. To illustrate
15 the mechanisms behind this process, attention must first be paid to the
16 mathematical topic category entity itself and its many features.

17

18 Each of the distinct mathematical topic category entities defined on the
19 curriculum map is represented technically as an object, with a vast member
20 collection of related exercise questions and solutions designed to develop skills
21 and proficiency in the particular topic represented. Each category object also

1 maintains a Student-user Proficiency Level measurement that continually
2 indicates each user's demonstrated performance level in that particular category.
3 In addition, each category object also maintains a Question Difficulty Level that
4 determines the difficulty of any questions that may be chosen from the object's
5 question collection and presented to the user. As expected, the movement of an
6 object's Question Difficulty Level is directly correlated to the movement of the
7 Student-user Proficiency Level. Referring to Figure 9, conceptually, each
8 category object may be depicted as a container, for example a water bucket. With
9 this analogy, the height of the water level within each bucket could then represent
10 the Student-user Proficiency Level, rising and falling accordingly. Directly
11 correlated to the water level, the Question Difficulty Level may then be
12 represented by graduated markings along the height of the bucket's inner wall,
13 ranging from low difficulty near the bottom to high difficulty near the top. The
14 rise and fall of the water level would therefore relate directly to the markings
15 along the bucket's wall.
16
17 As a student-user answers questions from a particular bucket, their Proficiency
18 Level in that topic area is gleaned from the accuracy of each answer, as well as
19 their overall performance history and consistency in the category. In general, a
20 correct answer will increase the user's proficiency measurement in that category,
21 while an incorrect answer will decrease it. A bucket's water level therefore

1 responds to each of the user's attempts to solve a question from that bucket's
2 collection. The issue left unresolved here is the incremental change in height
3 applied to the bucket's water level with each answered question.
4
5 On a per question basis, the magnitude of the incremental change in Proficiency
6 Level should vary, and will be determined by the user's recent performance
7 history in the category, specifically the consistency of their demonstrated
8 competence on previous questions from that bucket. Hence, a student-user who
9 has answered most questions in a category correctly will be posed with
10 progressively larger incremental increases in their Proficiency Level for an
11 additional correct answer, and progressively smaller incremental decreases for an
12 additional incorrect answer. The opposite conditions apply to a student-user that
13 has answered most questions in a category incorrectly. A student-user whose
14 performance history sits on the median will face an equally-sized increase or
15 decrease in Proficiency Level for their next answer.
16
17 The bucket property that will track and update a user's performance history is the
18 Student-user State rating. This rating identifies a user's recent performance
19 history in a particular bucket, ranging from unsatisfactory to excellent
20 competence. A student-user may qualify for only one State rating at a time. Each
21 State rating determines the magnitude of incremental change that will be applied

1 to a user's Proficiency Level in that bucket upon the next answered question, as
2 discussed in the previous paragraph. The user's performance on the next question
3 will then update the user's recent performance history, and adjust the user's State
4 accordingly before the next question is presented. In terms of the water bucket
5 analogy, a user's State may be illustrated as a range of cups, each of a different
6 size, which can add and remove varying amounts of water to and from the bucket.
7 Before answering each question from a bucket, a student-user is equipped with a
8 particular cup in one hand for adding water and a particular cup in the other hand
9 for removing water, depending on the user's State. The potential incremental
10 change in water level per question is therefore determined based on the user's
11 State. As the user's State rating changes, so do the cup sizes in the user's hands.
12
13 Revisiting the discussed functionality of the Proficiency Level in each bucket, it
14 becomes apparent that the full range of the Proficiency scale must be finite, and
15 therefore some other mechanisms must come into play once a user's Proficiency
16 Level in a bucket approaches the extreme boundaries of its defined range. It
17 would be nonsensical to continue adding water to a bucket that is filled to the
18 brim, or removing water from an empty bucket. Instead, approaching these
19 extreme scenarios should trigger a specialized mechanism to either promote or
20 demote the user's focus appropriately to another bucket. This is in fact the case,
21 and the new mechanisms that take over in these situations will lead the discussion

1 into inter-bucket relationships and traversing the curriculum map's links between
2 multiple buckets.
3

4 If a user's Proficiency Level in a particular bucket reaches a high enough level,
5 the student-user then qualifies to begin learning about content and attempting
6 questions from the "next" category bucket defined on the curriculum map.
7 Likewise, if a student-user demonstrates insufficient competence in a particular
8 bucket, their Proficiency Level in that bucket drops to a low enough level to begin
9 presenting the student-user with questions from the "previous" category bucket
10 defined on the curriculum map. These upper and lower Proficiency Threshold
11 Levels determine transitional events between buckets and facilitate the
12 development of a user's personalized progression rate and traversal paths through
13 the various conceptual categories on the curriculum map.
14

15 The direct relationships between category buckets on the curriculum map are
16 defined based on parallel groupings of similar level concept topics, and
17 prerequisite standards between immediately linked buckets of consecutive parallel
18 groups. These relationships help to determine the general progression paths that
19 may be taken from one bucket to the "next" or "previous" bucket in a curriculum.
20 Beyond the simple path connections, buckets that are immediately linked in the
21 curriculum map also carry a Correlation Index between them, which indicates

1 how directly the buckets are related, and how requisite the “previous” bucket’s
2 material is to learning the content of the “next” bucket. These metrics not only
3 determine the transition process between buckets, but also help to dynamically
4 determine the probability of selecting questions from two correlated buckets as a
5 student-user gradually traverses from one to the other (this selection functionality
6 will be addressed shortly under the Question Selection Algorithm section).

7

8 Briefly summarizing, there are several levels of mechanisms operating on the
9 curriculum map, both within each category bucket as well as between related
10 category buckets. Within each bucket, a user’s performance generates
11 Proficiency measurements, which set Difficulty Level ranges that ultimately
12 determine the difficulty levels of questions selected from that particular category.
13 Between related buckets, directly relevant topics are connected by links on the
14 curriculum map, and characterized by Correlation Indexes that reflect how
15 essential one topic is to learning another.

16

17

1 The present invention is a network (e.g., web-based) computer program product
2 application comprising one or more client and server application modules. The
3 client side application module communicates with the server side application
4 modules, based on student-user input/interaction.

5

6 In one exemplary embodiment of the present invention, the client tier comprises a
7 web browser application such as Internet Explorer™ by Microsoft™, and more
8 specifically, a client application based on Flash animated graphics technology and
9 format by Macromedia™.

10

11 In one exemplary embodiment of the present invention, the server tier comprises a
12 collection of server processes including a Knowledge Assessment Test module, a
13 Topic Selection module, and a Question Selection module. (collectively also
14 called "Engine"), discussed below.

15

16

1 KNOWLEDGE ASSESSMENT MODULE

2

3 The Knowledge Assessment component has the following objectives:

4 To efficiently identify for each student-user the most appropriate starting
5 topic from a plurality of topics.

6 To gauge student-user knowledge level across different learning
7 dimensions.

8

9 The Knowledge Assessment comprises 3 phases:

10 Phase 1 consists of several questions (e.g., 5-10) purely numerical
11 questions designed to assess the user's arithmetic foundations.

12 Phase 2 consists of a dynamic number (depending on user's success) of
13 word problem-oriented numerical questions designed to gauge the user's
14 knowledge of and readiness for the curriculum. The aim of Phase 2 is to quickly
15 and accurately find an appropriate starting topic for each user.

16 Phase 3 consists of several questions (e.g., 10-20) word problem-oriented
17 questions designed to test the user's ability in all other learning dimensions. If the
18 student-user exhibits particularly poor results in Phase 3, more questions may be
19 posed

20

21 **Initial Test Selection**

●

In one embodiment, to enhance the system's intelligence, the system prompts the student-user for date of birth and grade information. After entering the requested date of birth and grade information, the system prompts the student-user with one of several (e.g., six) Phase I Tests, based on the following calculation:

6

7 Date of Birth is used to compute Age according to the following formula:

8

9 SecondsAlive = Number of seconds since midnight on the user's Date of
10 Birth

```
11      Age = Floor( SecondsAlive ÷ 31556736 )
```

12

13 Grade is an integer between 1 and 12.

14

15 The system determines an appropriate Test Number as follows: note that where
16 grade and/or date of birth data is missing, the system uses predetermined logic.

17

18 If no data is known (Note: this case should not happen), then Test Number = 1

19

20 If only date of birth is known, then Test Number = $\max \{ 1, \min \{ \text{Age} - 5, 6 \} \}$

21

1 If only grade is known (Note: this case should not happen), then Test Number =
2 $\min\{\text{Grade}, 6\}$

3

4 If both date of birth and grade are known, then Test Number = $\min\{\text{Floor}[(2 \times$
5 $\text{Grade}) + (\text{Age} - 5)] \div 3, 6\}$

6

7 **Test Jumps**

8

9 Depending on the user's progress or level of proficiency, the student-user may
10 jump from one test to another.

11

12

13 *Test Jump Logic*

14

15 If the student-user answers a certain number of consecutive questions correctly
16 (incorrectly), the student-user will jump up (down) to the root node of the next
17 (previous) test. The requisite number depends on the particular test and is hard-
18 coded into each test. For example, a student-user starting in Test 1 must answer
19 the first four Phase 2 questions correctly in order to jump to Test 2.

20

21 *Test Jump Caps*

1
2 If the student-user jumps up (down) from one Test to another, in one embodiment, the system
3 will prevent the student-user from jumping back down (up) in the future to revisit a Test.

4
5 In another embodiment, the student-user may revisit a Test however, the user's starting topic
6 is set to the highest topic answered successfully in the lower level Test. For example,
7 referring to Figure 10, if the student-user jumps from Test 1 to Test 2, and then subsequently
8 falls back to Test 1, the starting topic is set at the 01N05 test, Phase 2 ends, and Phase 3 of the
9 01N05 test begins.

10 11 **Test Progression**

12
13 In one embodiment, a student-user proceeds through the Knowledge Assessment module
14 linearly, beginning with Phase 1 and ending with Phase 3. Phase 1 and Phase 2 are linked to
15 specific test levels. Phase 3 is linked to a specific Number topic, namely the Number topic
16 determined in Phase 2 to be the user's starting topic. Two users who start with the same
17 Phase 1 test will take at least part of the same Phase 2 test (though depending on their
18 individual success, one may surpass the other and see more questions), but may take very
19 different Phase 3 tests depending on their performance in Phase 2.

1

2 Knowledge Assessment Question Selection Approach

3

4 Each Knowledge Assessment question tests one or both of two skills: word
5 problem-solving skill, and skill in one of the five other learning dimensions. The
6 following variables are used for scoring purposes:

7

8 NScore – A running tally of the number of Number-related questions the student-
9 user has answered correctly.

10 NTotal – A running tally of the number of Number-related questions the student-
11 user has attempted.

12 PScore – A running tally of the number of Problem Solving-related questions the
13 student-user has answered correctly.

14 PTotal – A running tally of the number of Problem Solving-related questions the
15 student-user has attempted.

16 PSkill – Codes whether the question tests proficiency in Word Problems. In
17 general, will be set to 0 for Phase 1 questions, and to 1 for Phase 2 and Phase 3
18 questions

19

20 At the beginning of the Knowledge Assessment, all four of these variables are
21 initialized to zero.

1

2 **Assessments Test Phases**

3 The various assessments tests consists of three phases, namely Phase 1, Phase 2
4 and Phase 3.

5

6 **Phase 1**

7

8 *Overview*

9 Phase 1 is used to assess the user's foundation in numerical problems.

10 Phase 1 consists of a predetermined number (e.g., 5-10) of hard-coded questions.

11 The system presents the questions to the student-user in a linear fashion..

12

13 *Phase 1 Logic:*

14 1. If the student-user answers a question correctly:

15 a. NScore is increased by 1.

16 b. NTotal is increased by 1.

17 c. The student-user proceeds to the next question referenced in the
18 question's "Correct" field.

19

20 2. If the student-user answers a question incorrectly:

21 a. NScore is not affected.

- 1 b. NTotal is increased by 1.
- 2 c. The student-user proceeds to the next question referenced in the
- 3 question's "Incorrect" field.
- 4

5 **Phase 2**

6 *Overview*

7

8 Phase 2 establishes the user's starting topic. Phase 2 follows a binary tree

9 traversal algorithm. See Figure 10. Figure 10 depicts an exemplary binary tree

10 representing Phase 2 of an Assessment Test 1. The top level is the root node. The

11 bottom level is the placement level, where the user's starting topic is determined.

12 All levels in between are question levels. Nodes that contain pointers to other

13 Tests (indicated by a Test level and Phase number) (See Figure 10) are called jump

14 nodes. Each Test Level Phase 2 tree looks look similar to Figure 10 with varying

15 tree depths (levels).

16

17 An exemplary Phase 2 binary tree traversal algorithm is as follows:

18

19 Leftward movement corresponds to a correct answer. Rightward

20 movement corresponds to an incorrect answer.

21

1 The topmost topic is the root node. This is where the student-user starts
2 after finishing Phase 1. At the root node, the student-user is asked two questions
3 from the specified topic. This is the only node at which two questions are asked.
4 At all other nodes, only one question is asked.

5 At the root node, the student-user must answer both questions correctly to
6 register a correct answer for that node (and hence move leftward down the tree).
7 Otherwise, the student-user registers an incorrect answer and moves rightward
8 down the tree.

9 The student-user proceeds in this manner down through each question
10 level of the tree.

11 The student-user proceeds in this manner until he reaches the placement
12 level of the tree. At this point, he either jumps to Phase 1 of the specified test (if
13 he reaches a jump node) or the system registers a starting topic as indicated in the
14 node.

15

16 *Phase 2 Logic:*

- 17 1. If the student-user answers a question correctly:
- 18 a. NScore increases by 1.
- 19 b. NTotal increases by 1.
- 20 c. If the question's Pskill is set to 1, then
- 21 i. PScore increases by 1.

- 1 ii. PTotal increases by 1.
- 2 d. Else if the question's PSkill is set to 0, then
- 3 i. PScore is unaffected.
- 4 ii. PTotal is unaffected.
- 5 e. The student-user proceeds to the next question referenced in the
- 6 question's "Correct" field.
- 7 2. If the student-user answers a question incorrectly:
- 8 a. NScore is unaffected.
- 9 b. NTotal increases by 1.
- 10 c. If the question's PSkill is set to 1, then
- 11 i. PScore is unaffected.
- 12 ii. PTotal increases by 1.
- 13 d. Else if the question's PSkill is set to 0, then
- 14 i. PScore is unaffected.
- 15 ii. PTotal is unaffected.
- 16 e. The student-user proceeds to the next question referenced in the
- 17 question's "Incorrect" field.

1 **Phase 3**

2

3 Phase 3 is designed to assess the user's ability in several learning dimensions
4 (e.g., the Measure (M), Data Handling (D), Shapes and Space (S), and Algebra
5 (A) learning dimensions) at a level commensurate with the user's starting Number
6 topic determined in Phase 2. Phase 3 consists of a predetermined number of
7 questions (e.g., 9-27) hard-coded to each starting Number topic. For example, if
8 the user's starting Number topic is determined in Phase 2 to be 01N03, then the
9 student-user is presented with an corresponding 01N03 Phase 3 test.

10

11 The Knowledge Assessment lookup tables contain 3 questions from each M, D, S,
12 and A learning dimensions in the PLANETii curriculum.

13

14 Each Phase 3 test pulls questions from between 1 and 3 topics in each learning
15 dimension.

16

17 *Phase 3 Logic:*

18

- 19 1. If the student-user answers a question correctly:
- 20 a. If the question's PSkill is set to 1, then
- 21 i. PScore increases by 1.

- ii. PTotal increases by 1.
 - b. Else if the question's PSkill is set to 0, then
 - i. PScore is unaffected.
 - ii. PTotal is unaffected.
 - c. The student-user proceeds to the next question referenced in the question's "Correct" field.
 2. If the student-user answers a question incorrectly:
 - a. If the question's PSkill is set to 1, then
 - i. PScore is unaffected.
 - ii. The PTotal increases by 1.
 - b. Else if the question's PSkill is set to 0, then
 - i. PScore is unaffected.
 - ii. The PTotal is unaffected.
 - c. The student-user proceeds to the next question referenced in the question's "Incorrect" field.
 3. If the student-user answered all three questions in any topic incorrectly, the system provides a fallback topic at the end of Phase 3.

Each topic in the M, D, S, and A learning dimensions is coded with a fallback topic. If the student-user fails a topic, the student-user is given the

1 opportunity to attempt the fallback topic. For example, if a student-user answers
2 all three questions in 03M01 (Length and Distance IV) incorrectly, after the
3 student-user completes Phase 3, the system prompts the student-user with a
4 suggestion to try a fallback topic, e.g., 01M03 (Length and Distance II).

5

6 DATA STORAGE OF KNOWLEDGE ASSESSMENT INFORMATION – 7 DATABASE ORGANIZATION

8

9 The content/questions used during the Knowledge Assessment module are
10 stored in a main content-question database. One or more look up tables are
11 associated with the database for indexing and retrieving knowledge assessment
12 information. Exemplary knowledge assessment lookup tables comprise the
13 following fields A-W and optionally fields X-Y:

14

15 Field A: AQID

16 Field A contains the Knowledge Assessment Question ID code (AQID). This
17 should include the Test level (01-06, different for Phase 3), Phase number (P1-
18 P3), and unique Phase position (see below). Each of the three Phases has a
19 slightly different labeling scheme. For example: 01.P1.05 is the fifth question in
20 Phase 1 of the Level 1 Knowledge Assessment; 03.P2.I1C2 is the third question
21 that a student-user would see in Phase 2 of the Level 3 Knowledge Assessment

- 1 following an Incorrect and a Correct response, respectively; and 01N03.P3.02 is
2 the second question in the 01N03 Phase 3 Knowledge Assessment.
- 3 Field B: QID
4 Field C: Topic Code
5 Field D: Index
6 Field E: PSL
7 Field F: Question Text
- 8 -Fields B-F are pulled directly from the main content-question database and are
9 used for referencing questions.
- 10 Field G: Answer Choice A Text
11 Field H: Answer Choice B Text
12 Field I: Answer Choice C Text
13 Field J: Answer Choice D Text
14 Field K: Answer Choice E Text
- 15 -Fields G-K contain the five possible Answer Choices (a-e).
- 16 Field L : Correct Answer Text.
- 17 Fields M-Q contain Incorrect Answer Explanations corresponding to the Answer
18 Choices in fields G-K. The field corresponding to the correct answer is grayed-
19 out.
- 20 Field R: Visual Aid Description - The Visual Aid Description is used by
21 Content to create Incorrect Answer Explanations.

- 1 Field S: Correct – A pointer to the QID of the next question to ask if the
2 student-user answers the current question correctly.
- 3 Field T: Incorrect – A pointer to the QID of the next question to ask if the
4 student-user answers the current question incorrectly.
- 5 Field U: NSkill – 0 or 1. Codes whether the question involves Number skill.
6 Used for scoring purposes.
- 7 Field V: PSkill – 0 or 1. Codes whether the question involves Word
8 problem skill. In general, will be set to 0 for Phase 1 questions, and to 1 for Phase
9 2 and Phase 3 questions. Used for scoring purposes.
- 10 Field W: LDPoint – 1, 1.2, or 1.8 points for questions in Phase 3, blank for
11 questions in Phase 1 and Phase 2. Depends on PSL of question and is used for
12 evaluation purposes.
- 13 Field X: Concepts – Concepts related to the question material. May be used
14 for evaluation purposes in the future.
- 15 Field Y: Related Topics – Topics related to the question material. May be
16 used for evaluation purposes in the future.
- 17
- 18

1 FORMULAS FOR TEST SCORING

2

3 During the Knowledge Assessment Test module, the system calculates several
4 scores as follows:

5

6 The user's number score in the Numbers learning dimension is calculated via the
7 following formula:

8

9 $\text{Number Score} = \min[\text{Floor}\{[\text{NScore} / (\text{NTotal} - 1)] * 5\}, 5]$

10

11 The user's score in other learning dimensions (e.g., Measure, Data Handling,
12 Shapes and Space and Algebra) is calculated as follows:

13

14 First, a score is computed in each topic. In each Measure, Data Handling,
15 Shapes and Space and Algebra learning dimension, there are three questions, one
16 each with a LDPoint value of 1, 1.2, and 1.8. The user's topic score is calculated
17 via the following formula:

18

19 $\text{Topic Score} = \text{Round}\{\text{Sum of LDPoints of All 3 Questions} * (5/4)\}$

20

1 All Topic Scores in a given Learning Dimension are averaged (and floored) to
2 obtain the Learning Dimension Score.

3

4 Finally, the user's word problem score is calculated using the following formula:

5

6 Word Problem Score = min[Floor{ [PScore / (PTotal - 1)] * 5 }, 5]

7

8 EVALUATION OF KNOWLEDGE ASSESSMENT RESULTS

9

10 Overview

11 At the end of the Knowledge Assessment module, the system prompts the
12 student-user student-user to log out and the parent/instructor to log in to access
13 test results. The system then presents the parent/instructor with a screen relaying
14 the following evaluation information: 1) the name of each of the learning
15 dimensions (currently, five) in which the student-user student-user was tested is
16 listed, along with a 0-5 scale displaying the user's performance and 2) the user's
17 "Word Problem Skill" is assessed on a 0-5 scale.

18

19 The parent/instructor can then select a learning dimension or the "Word
20 Problem Skill" to see all relevant questions attempted by the student-user user,
21 along with incorrect answers and suggested explanations.

1

2 *Evaluation Standards*

3 Using an exemplary 0-5 scale, a 5 corresponds to full proficiency in a
4 topic. If a student-user scores a 5 in any learning dimension or in word problem
5 solving, the system displays the following message: “[Child Name] has
6 demonstrated full proficiency in [Topic Name].”

7

8 A 3-4 corresponds to some ability in that topic. If a student-user scores a
9 3-4 in any learning dimension or in word problem-solving, the system displays
10 the following message: “[Child Name] has demonstrated some ability in [Topic
11 Name]. PLANETii system will help him/her to achieve full proficiency.”

12

13 A 0-2 generally means that the student-user is unfamiliar with the topic
14 and needs to practice the material or master its prerequisites.

15

16 Full proficiency in a topic is defined as ability demonstrated repeatedly in
17 all questions in the topic. In the current implementation described herein, a
18 student-user has full proficiency only when he/she answers every question
19 correctly.

20

1 Some ability in a topic is defined as ability demonstrated repeatedly in a
2 majority of questions in the topic. In the current implementation, the student-user
3 must answer 2 of 3 questions in any topic correctly.

4

5

6 INITIALIZATION OF WATER LEVELS

7

8 After completion of the Knowledge Assessment Test module, the water
9 levels of the user's starting topic, any pre-requisites and related topics are
10 initialized (pre-assigned values) according to the following logic:

11

- 12 ▪ The water level in the user's starting topic is not initialized.
- 13 ▪ The water level in any Number topics that are pre-requisites (with a high
14 correlation coefficient (NEW) to the user's starting topic is initialized to
15 85.
- 16 ▪ For the other learning dimensions, topics are organized into subcategories.

17

18 Consider the following example where one family of topics organized into related
19 sub-topic categories include:

20 1. 01M01 Length and Distance I

21 2. 01M03 Length and Distance II

1 3. 02M01 Length and Distance III

2 4. 03M01 Length and Distance IV

3 Suppose a user, after completing the Knowledge Assessment Test module, is
4 tested in topic 03M01 Length and Distance IV: if his/her topic score in 03M01
5 Length and Distance IV is 5, then a). the water level in 03M01 Length and
6 Distance IV is set to 85 and b) the water level in related topics 01M01 Length and
7 Distance I, 01M03 Length and Distance II, 02M01 02M01 Length and Distance
8 III is set to 85.

9 If his/her topic score in 03M01 Length and Distance IV is 4, then a) the
10 water level in 03M01 Length and Distance IV is set to 50; and b) the water level
11 in related topics 01M01 Length and Distance I, 01M03 Length and Distance II,
12 02M01 Length and Distance III is set to 85.

13 If his/her topic score in 03M01 Length and Distance IV is 3 or below, then
14 a) the water level in 03M01 Length and Distance IV is not initialized; b) the water
15 level in related topic 02M01 Length and Distance III is not initialized; and c) the
16 water level in any related topic in the subcategory at least twice removed from
17 03M01 Length and Distance IV (in this case, 01M01 Length and Distance I and
18 01M03 Length and Distance II) is initialized to 85.

19 The water level for a given topic can be assigned during initialization or
20 after a student-user successfully completes a topic. Thus, a pre-assigned water
21 level of 85 during initialization is not the same as an earned water level of 85 by

1 the user. Therefore, a student-user can fall back into a topic with a pre-assigned
2 water level of 85 if need be.

3

4 TOPIC SELECTION ALGORITHM MODULE

5

6 The Topic Selection module is a three step multi-heuristic intelligence algorithm
7 which assesses the eligibility of topics and then ranks them based on their
8 relevance to a given student's past performance. During step one, the Topic
9 Selection module prunes (culls) the list of uncompleted topics to exclude those
10 topics which are not relevant to the student's path and progress. During step two,
11 the Topic Selection module evaluates each eligible topic for relevance using the
12 multi-heuristic ranking system. Each heuristic contributes to an overall ranking of
13 relevance for each eligible topic and then the topics are ordered according to this
14 relevance. During step three, the Topic Selection module assesses the list of
15 recommendations to determine whether to display the recommended most
16 relevant topics.

17

18 FIG. 11 depicts an exemplary process flow for the Topic Selection Algorithm
19 module.

20

21 *Step 1 - Culling eligible topics*

1
2 The Topic Selection module employs several culling mechanisms which
3 allow for the exclusion of topics based on the current state of a user's curriculum.
4 The topics that are considered eligible are placed in the list of eligible topics.
5 The first step includes all topics that have an eligibility factor greater than 0, a
6 water level less than 85 and no value from the placement test. This ensures that
7 the student-user will not enter into a topic that they are not ready for or one that
8 they have already completed or tested out of. The last topic a student-user
9 answered questions in is explicitly excluded from the list which prevents the
10 engine from recommending the same topic twice in a row particularly if the
11 student-user fails out of the topic.
12 After these initial eligibility assertions take place, some additional considerations
13 are made. If there are any topics that are current failed in the user's curriculum, all
14 of the uncompleted pre-requisites of these topics are added to the eligible list.
15 This includes topics that received values from the placement test.
16 Finally, if there are no failed topics in the student's curriculum and all the topics
17 in the recommendation list that are greater than 1 level away from the student's
18 average level, the list is cleared and no topics are included. This will indicate a
19 "Dead End" situation.

20

21 *Step 2 - Calculating Relevance*

1
2 After the list of eligible topics has been compiled, the Topic Selection module
3 calculates a relevance score for each topic. The relevance score is calculated
4 using several independent heuristic functions which evaluate various aspects of a
5 topic's relevance based upon the current state of the user's curriculum. Each
6 heuristic is weighted so that the known range of its values can be combined with
7 the other heuristics to provide an accurate relevance score. The weights are
8 designed specifically for each heuristic so that one particular relevance score can
9 cancel or compliment the values of other heuristics. The interaction between all
10 the heuristics creates a dynamic tension in the overall relevance score which
11 enables the recognition of the most relevant topic for the student-user based on
12 their previous performance.

13

14 *Relevance Heuristics Explained*

15

16 1) Average Level Relevance

17 Overview:

18 This heuristic determines a student's average overall level and then
19 rewards topics which are within a one-level window of the average while
20 punishing topics that are further away.

21

1 Formula:

2 For each level:

3

4 $\text{LevelAverage} = \text{sum}(\text{topicWaterLevel} * \text{topicLevel}) / \text{sum}(\text{topicLevel})$

5 $\text{Average Level} = \text{Sum}(\text{LevelAverage})$

6 Topic relevance: $(0.5 - \text{ABS}(\text{topicLevel} - \text{Average Level})) * 5$

7

8 Range of Possible Values:

9 (in current curriculum 1-4): 2.5 to -17.5

10

11 Weighted Range of Possible Values:

12 (in current curriculum 1-4): 7.5 to -52.5

13

14 2) Eligibility Relevance

15 Overview:

16 This heuristic assesses the student's readiness for the topic, found by determining
17 how much of each direct pre-requisite a student-user has completed.

18

19 Formula:

20

21 If $W(\text{PrqN}) \geq 85$, then set $W(\text{PrqN}) = 85$;

1

2

3

4 wherein: $E(X)$ be the Eligibility Index of Bucket X,
5 $W(\text{PrqN})$ be the Water Level of Pre-requisite N of Bucket X
6 $\text{Cor}(X, \text{PrqN})$ be the Correlation Index between Bucket X and its
7 Pre-requisite N, where N is the number of pre-requisite buckets
8 for X
9 t be the constant 100/85

10

11 Range of Possible Values:

12 (in current curriculum 1-4): 100 to 0

13

14 Weighted Range of Possible Values:

15 (in current curriculum 1-4): 20 to 0

16

17 3) Concept Importance (Static Multiplier) Relevance

18 Overview:

19 Concept importance is a predetermined measure of how important a topic is. For
20 example, a topic like "Basic Multiplication" is deemed more important than "The
21 Four Directions."

1

2 Formula:

3 $1 - (\text{Topic Multiplier})$

4

5 Range of Possible Values:

6 (in current curriculum 1-4): 1 to 0

7

8 Weighted Range of Possible Values:

9 (in current curriculum 1-4): 5 to 0

10

11 4) Contribution Relevance

12 Overview:

13 This heuristic measures the potential benefit completing this topic would provide,
14 by adding its post-requisites' correlations.

15

16 Formula:

17 $\text{SUM}(\text{post requisite correlation})$

18

19 Range of Possible Values:

20 (in current curriculum 1-4): ~6 to 0

21

1 Weighted Range of Possible Values:

2 (in current curriculum 1-4): ~3 to 0

3

4 5) Learning Dimension Repetition Relevance

5 Overview:

6 This heuristic is meant to ensure a degree of coherence to the student-user while
 7 developing a broad base in multiple learning dimensions. The heuristic favors 2
 8 consecutive topics in a particular learning dimension, and then gives precedence
 9 to any other learning dimension, so a student-user doesn't overextend his/her
 10 knowledge in any one learning dimension.

11

12 Formula:

13 This heuristic uses a lookup table (see below) of values based on the number of
 14 consecutive completed topics in a particular learning dimension.

15

16	1) Repetitions	2) 0	3) 1	4) 2	5) 3	6) 4	7) 5	8) 6	9) 7	10) 8
17	11) Value	12) 2	13) 7.5	14) -1	15) -5	16) -9	17) -12	18) -17	19) -22	20) -27
18										
19										

20

21

1 Range of Possible Values:

2 (in current curriculum 1-4): 7.5 to -27.5

3

4 Weighted Range of Possible Values:

5 (in current curriculum 1-4): 9.38 to -34.375

6

7 6) Failure Relevance

8 Overview:

9 This heuristic gives a bonus to topics that are important pre-requisites to
10 previously failed topics. For example, if a student-user fails 01M01 (Length and
11 Distance I), then the pre-requisites of 01M01 will receive a bonus based on their
12 correlation to 01M01. It treats assessment test topics differently than the normal
13 unattempted topics and weights the bonuses it gives to each according to the
14 balance of the correlation between these prerequisites. For example, an
15 assessment test topic's correlation to the failed topic must be higher than the sum
16 of the other unattempted topics or it receives no bonus. All unattempted topics
17 receive a bonus relative to their correlation to the failed topic.

18

19 Formula:

20 get the kid/bucket data

21 loop through the failed topics

1 get this failed topic ID
2 get the topic data for the failed topic ID
3 if we are a pre-req of the failed topic
4 sum the unattempted pre-req buckets' correlations
5 if the AT topic's correlation is higher than the sum of the unattempted pre-reqs
6 add $5 + (5 * \text{our correlation} - \text{the unattempted sum})$ to the bonus
7 otherwise return nothing
8 otherwise return $10 * \text{the pre-req's correlation}$
9 return the bonus
10
11 Range of Possible Values:
12 (in current curriculum 1-4): 10 to 0
13
14 Weighted Range of Possible Values:
15 (in current curriculum 1-4): 10 to 0
16
17 7) Additional Failure (Re-Recommend) Relevance
18 Overview:
19 This heuristic promotes failed topics if the student-user has completed most of the
20 pre-requisite knowledge, and demotes topics for which a high percentage of the

1 pre-requisite knowledge has not been satisfied. If the last topic completed was a
2 pre-requisite of this failed topic, this topic receives a flat bonus.
3
4 Formula:
5 $\text{score} += (80 - EI) / 10;$
6 $\text{if}(\text{preReq.equals}(\text{EngineUtilities.getLastBucket}(\text{userId}))) \{ \text{score} += 3; \}$
7
8 Range of Possible Values:
9 (in current curriculum 1-4): 11 to -2
10
11 Weighted Range of Possible Values:
12 (in current curriculum 1-4): 11 to -2
13
14 `public double calculateRelevance(String userId, String topicId) {`
15 `double score = 0;`
16 `// get the kid/bucket data`
17 `KidBucketWrapper kbw = new KidBucketWrapper(userId, topicId);`
18 `// loop through the failed topics`
19 `for(Iterator i = curriculum.getFailedTopics(userId).iterator(); i.hasNext();) {`
20 `// get this failed topic Id`
21 `String fTopicId = (String)i.next();`

```
1  // get the Topic data for the failed topic id
2  Topic fTopic = curriculum.getTopic(fTopicId);
3  // if we are a pre-req of the failed topic
4  if(fTopic.getPreRequisite(topicId) != null) {
5  // if we are an AT topic
6  if(kbw.getAssessmentLevel() > 0) {
7  double preSum = 0;
8  // sum the unattempted pre-req buckets' corellations
9  for(Iterator i2 = fTopic.getPreRequisites();i2.hasNext();) {
10 String pre = (String)i2.next();
11 Topic preTopic = curriculum.getTopic(pre);
12 KidBucketWrapper prebw = new KidBucketWrapper(userId, pre);
13 If(!pre.equals(topicId) && prebw.getAssessmentLevel() == 0 &&
14 prebw.getWaterLevel() == 0) {
15 preSum+=preTopic.getPostRequisite(fTopicId).getCorrelationCoefficient();
16 }
17 }
18
19 // if the AT topic's corellation is higher than the sum of the unattempted pre-reqs
20 if(fTopic.getPreRequisite(topicId).getCorrelationCoefficient() > preSum) {
21 // add 5 + (5 * our correlation - the unattempted sum) to the bonus
```

```
1   score += 5 + (5 * (fTopic.getPreRequisite(topicId).getCorrelationCoefficient() -  
2   preSum));  
3   }  
4   // otherwise return nothing  
5   else {  
6   return 0;  
7   }  
8   }  
9   // otherwise return 10 * the pre-req's correlation  
10  else {return 10 * fTopic.getPreRequisite(topicId).getCorrelationCoefficient();  
11  }  
12  }  
13  }  
14  // return the bonus  
15  return score;  
16  }  
17
```

18 *Step 3 – Assess Recommendations*

19 During the third and final step, the system assesses the list of
20 recommendations to determine whether to display the recommended most
21 relevant topics.

1 ELIGIBILITY INDEX

2

3 The Eligibility Index represents the level of readiness for the bucket to be chosen.

4 In other words, we ask the question "How ready is the student-user to enter into

5 this bucket?" Hence, the Eligibility Index of a bucket is a measure of the total

6 percentage of pre-requisites being completed by the user. The Eligibility Index is

7 calculated as follow:

8

9 Let $E(X)$ be the Eligibility Index of Bucket X,10 Let $W(\text{PrqN})$ be the Water Level of Pre-requisite N of Bucket X11 Let $\text{Cor}(X, \text{PrqN})$ be the Correlation Index between Bucket X and its Pre-requisite

12 N, where N is the number of pre-requisite buckets for X

13 Let t be the constant 100/85

14

15 If $W(\text{PrqN}) \geq 85$, then set $W(\text{PrqN}) = 85$;

16

$$17 \quad E(X) = \frac{\sum_{N=1}^N [t * W(\text{PrqN}) * \text{Cor}(X, \text{PrqN})]}{\sum_{N=1}^N \text{Cor}(X, \text{PrqN})}$$

18

19

20

1 To increase the effectiveness of choosing an appropriate bucket for the user, we
2 introduce a new criteria called Eligibility Index Threshold. If the eligibility index
3 does not reach the Eligibility Index Threshold, then the bucket is considered not
4 ready to be chosen.

5

6 Summary of Relevant Numbers for Implementation

- 7 1. Question selection starts at Water Level 25 for any new bucket
- 8 2. Proficiency Range (Water Level Range) is 0 to 100
- 9 3. Lower Threshold = 10
- 10 4. Upper Threshold = 85
- 11 5. Force Jump Backward at Water Level 0
- 12 6. Force Jump Forward at Water Level 100
- 13 7. Eligibility Index Threshold = 80

14

15 *Ranking and special case recognition*

16

17 Once the relevance has been calculated for each eligible topic, the Topic Selection
18 module recommends the two most relevant topics. If there are no topics to
19 recommend (i.e the Culling phase eliminated all possible recommendations), one
20 of two states is identified. The first state is called "Dead Beginning" and occurs
21 when a student-user fails the 01N01 "Numbers to 10" topic. In this case, the

1 student-user is not ready to begin using the Smart Practice training and a message
2 instructing them to contact their parent or supervisor is issued. The second state
3 is called "Dead End" and occurs when a student-user has reached the end of the
4 curriculum or the end of the available content. In this case, the student-user has
5 progressed as far as possible and an appropriate message is issued.

6 QUESTION SELECTION MODULE

7

8 *Overview*

9 Once a topic has been determined for the student-user, the Question Selection
10 Module delivers an appropriately challenging question to the student-user. In
11 doing so, the Question Selection Module constantly monitors the student-user's
12 current water level and locates the question(s) that most closely matches the
13 difficulty level the student-user is prepared to handle. Since water level and
14 difficulty level are virtually synonymous, this means that a student-user currently
15 at (for example) water level 56 should get a question at difficulty level 55 before
16 one at difficulty level 60. If the student-user answers the question correct, his/her
17 water level increases by an appropriate margin; if he/she answers incorrectly,
18 his/her water level will decrease.

19

20 Additionally, the Question Selection Module provides that all questions in a topic
21 should be exhausted before delivering a question the student-user has previously

1 answered. If all of the questions in a topic have been answered, the Question
2 Selection Module will search for and deliver any incorrectly answered questions
3 before delivering correctly answered questions. Alternatively and preferably, the
4 system will have an abundance of questions in each topic, therefore, it is not
5 anticipated that student-users will see a question more than once.

6 *Question Search Process*

7 All questions are each assigned a specific difficulty level from 1-100. Depending
8 on the capabilities of the system processor(s), the system may search all of the
9 questions for the one at the closest difficulty level to a student-user's current
10 water level. Alternatively, during the search process, the system searches within a
11 pre-set range around the student-user's water level. For example, if a student-
12 user's water level is 43, the system will search for all the questions within 5
13 difficulty levels (from 38 to 48) and will select one at random for the student.

14

15 The threshold for that range is a variable that can be set to any number. The
16 smaller the number, the tighter the selection set around the student's water level.
17 The tighter the range, the greater the likelihood of finding the most appropriate
18 question, but the greater the likelihood that the system will have to search
19 multiple times before finding any question.

20

21 General Flow

- 1 1. Get the student's current water level
- 2 2. Search the database for all questions within (+ or -) 5 difficulty levels of
- 3 the student's water level. (NOTE: This threshold + or - 5 can become tighter to
- 4 find more appropriate questions, but doing so will increase the demands on the
- 5 processor.)
- 6 3. Serve a question at random from this set.
- 7 4. Depending on the students answer, adjust his/her water level according to
- 8 the water level adjustment table.
- 9 5. Repeat the process.

10

11 Governing Guidelines

- 12 1. Questions should be chosen from difficulty levels closest the student's
- 13 current water level. If no questions are found within the stated threshold (in our
- 14 example, + or - 5 difficulty levels), the algorithm will continue to look further and
- 15 further out (+ or - 10, + or - 15, and so on).
- 16 2. A previously answered question should not be picked again for any
- 17 particular student-user unless all the possible questions in the topic have been
- 18 answered.
- 19 3. If all questions in a topic have been answered, search for the closest
- 20 incorrectly answered question.

1 4. If all questions have been answered correctly, refresh the topic and start
2 again.

3
4 Figure 15 depicts an exemplary process flow for picking a question from a
5 selected topic-bucket.

6 7 STATE LEVEL AND WATER LEVEL CALCULATIONS

8
9 A State Level indicates the student's consistency in performance for any bucket.
10 When a student-user answers a question correctly, the state level will increase by
11 1, and similarly, if a student-user answers incorrectly, the state level will decrease
12 by 1. Preferably, the state level has a range from 1 to 6 and is initialized at 3.

13
14 A Water Level represents a student's proficiency in a bucket. Preferably, the
15 water level has a range from 0 to 100 and is initialized at 25 when a student-user
16 enters a new bucket.

17
18 A Bucket Multiplier is pre-determined for each bucket depending on the
19 importance of the material to be covered in the bucket. The multiplier is applied
20 to the increments/decrements of the water level. If the bucket is a major topic, the
21 multiplier will prolong the time for the student-user to reach Upper Threshold. If

1 the bucket is a minor topic, the multiplier will allow the student-user to complete
2 the topic quicker.

3
4 To locate the corresponding water level from the user's current question to the
5 next question, the adjustment of the water level based on the current state of the
6 bucket is as follows:

7

State Level that the student-user is currently in:	Adjustment in water level when a question is answered correctly:	Adjustment of water level when a question is answered incorrectly:
1	+0m	-5m
2	+1m	-3m
3	+1m	-2m
4	+2m	-1m
5	+3m	-1m
6	+5m	-0m
<i>m</i> = Bucket Multiplier		

8

9

10 DATA TRANSFER

11

12 The communications are handled securely, using a 128-bit SSL Certificate signed
13 with a 1024-bit key. This is currently the highest level of security supported by
14 the most popular browsers in-use today.

15

1 The data that is exchanged between the client and server has 2 paths: 1) from the
2 server to the client, and 2) from the client to the server. The data sent from the
3 client to the server is sent as a POST method. There are two main ways to send
4 information from a browser to a web server, GET and POST. POST is the more
5 secure method. The data sent from the server to the client is sent via the
6 Extensible Markup Language (XML) format, which is widely accepted as the
7 standard for exchanging data. This format was chosen because of its flexibility,
8 and allows the system to re-use, change, or extend the data being used more
9 quickly and efficiently.

10

11 CONCLUSION

12 Having now described one or more exemplary embodiments of the invention, it
13 should be apparent to those skilled in the art that the foregoing is illustrative only
14 and not limiting, having been presented by way of example only. All the features
15 disclosed in this specification (including any accompanying claims, abstract, and
16 drawings) may be replaced by alternative features serving the same purpose, and
17 equivalents or similar purpose, unless expressly stated otherwise. Therefore,
18 numerous other embodiments of the modifications thereof are contemplated as
19 falling within the scope of the present invention as defined by the appended
20 claims and equivalents thereto.

21

1 Moreover, the techniques may be implemented in hardware or software, or a
2 combination of the two. In one embodiment, the techniques are implemented in
3 computer programs executing on programmable computers that each include a
4 processor, a storage medium readable by the processor (including volatile and
5 non-volatile memory and/or storage elements), at least one input device and one
6 or more output devices. Program code is applied to data entered using the input
7 device to perform the functions described and to generate output information.
8 The output information is applied to one or more output devices.
9
10 Each program is preferably implemented in a high level procedural or object
11 oriented programming language to communicate with a computer system,
12 however, the programs can be implemented in assembly or machine language, if
13 desired. In any case, the language may be a compiled or interpreted language.
14
15 Each such computer program is preferably stored on a storage medium or device
16 (e.g., CD-ROM, NVRAM, ROM, hard disk, magnetic diskette or carrier wave)
17 that is readable by a general or special purpose programmable computer for
18 configuring and operating the computer when the storage medium or device is
19 read by the computer to perform the procedures described in this document. The
20 system may also be considered to be implemented as a computer-readable storage

1 medium, configured with a computer program, where the storage medium so
2 configured causes a computer to operate in a specific and predefined manner.

3

4 Finally, an embodiment of the present invention having potential commercial
5 success is integrated in the Planetii™ Math System™, an online math education
6 software product, available at <<http://www.planetii.com/home/>>.

7

8 Figure 14 depicts an exemplary user interface depicting the various elements for
9 display. As shown, the question text data is presented as Display Area 2, the
10 potential answer choice(s) data is presented as Display Area 4, the correct answer
11 data is presented as Display Area 6, the Visual Aid data is presented as Display
12 Area 8 and the Descriptive Solution data is presented as Display Area 10.

13

1 CLAIMS

2

3 What is claimed is:

4

5 1. An adaptive learning system for presenting an appropriate topic and question
6 to a user, said system comprising:

7

8 a processor configured to:

9

10 generate and store in a database a set of hierarchical topics having a plurality of
11 questions associated with each one of said topics; each of said plurality of
12 questions within a topic having an assigned difficulty level value;

13

14 determine an adjustable state level value for a user based on said user's topic
15 performance consistency; said state level initialized to and having a range of
16 predetermined value;

17

18 determine an adjustable water level value for said user based on said user's
19 proficiency in at least a subset of said hierarchical topics; said water level
20 initialized to and having a range of predetermined value;

1 determine a relevant topic for said user from said set of hierarchical topics by
2 performing the following:
3
4 cull said set of hierarchical topics to determine one or more eligible
5 academic topics; and
6
7 evaluate for relevance said one or more eligible academic topics using
8 heuristic relevance ranking to determine said relevant academic topic;
9
10 determine an appropriate question for said user from said plurality of relevant
11 academic topic questions by performing the following:
12
13 determine said user's water level,
14
15 search said database for one or more questions within a threshold range
16 from said user's water level,
17
18 randomly select a relevant question from this one or more questions
19
20 depending on the user's answer to said selected question, adjust said user's
21 water level according to a predetermined adjustment table.

1

2

2. The system as in claim 1 wherein said processor is further configured to

3

evaluate for relevance said one or more eligible academic topics using at least

4

one of a Average Level Relavance heuristic, Eligibility Relevance heuristic,

5

Static Multiplier Relevance heuristic, Contribution Relevance heuristic,

6

Learning Dimension Repetition Relevance heuristic, Failure Relevance

7

heuristic and Re-recommend Failure Relevance heuristic.

8

9

3. The system as in claim 1 wherein said processor further defines a multiplier

10

value m , said state level value is initialized to 3 and ranging from 1 to 6, said

11

water level value is initialized to 25 and ranging from 0 to 100 and said

12

predetermined adjustment table comprises:

State Level that said user is currently in:	Adjustment in water level when a question is answered correctly:	Adjustment of water level when a question is answered incorrectly:
1	+0m	-5m
2	+1m	-3m
3	+1m	-2m
4	+2m	-1m
5	+3m	-1m
6	+5m	-0m
m = Multiplier		

13

14

1 4. The system as in claim 1 wherein said difficulty level value ranges from 1 to

2 100;

3

4 5. The system as in claim 1 wherein said threshold range is from ± 0 to ± 5 .

5

6 6. The system as in claim 1 wherein said threshold range is greater than ± 5 .

7

8