Generation of Test Based on Test Ontology

Larisa Soldatova, Riichiro Mizoguchi
ISIR, Osaka University
8-1 Mihogaoka, Ibaraki, Osaka, 567-0047, Japan
{larisa, miz}@ei.sanken.osaka-u.ac.jp

Abstract: In this paper design of test generation systems (TGS) based on a test ontology is discussed. Top level concepts of the test ontology are described. Suggested test ontology permits to analyze process of test composition, identify test tasks, and standardize components of TGS. Main functions, regimes, and components of TGS are considered. Test generation is described for the example integration of functions. System explanations and examples of generated test items are given.

Introduction

Test construction is important as to traditional educational technology, as to technology based on using IES (Intelligent Education Systems). In order to correctly select an educational strategy it is necessary to have a regular feedback from a student to control the student's level of understanding. The most popular measurement tool of students' knowledge is a test. Preparation of good tests and in-depth analysis of the test results take much time. The process of professional test construction is complicated and consists of many steps (Crocker, L., Algina, J., 1986). There are some information systems (IS), which facilitate the test composition, but the majority of them concentrate mainly on editing of test items or selecting them from an item bank, analyzing of the test results, and administrating of the test (i.e., TestMaster, TestPro, Sigma Test Management System, and so on).

The purpose of our research is analysis of the test design process on the basis of ontology methodology, construction of the test ontology, and design of an IS for test composition, which will contain knowledge about the test construction process.

Test Ontology

In the knowledge base community, ontology is defined as "a system of primitive vocabulary/concepts used for building artificial systems" (Mizoguchi, R., Sinitsa, K., 1996).

At the top level of the test ontology we consider "world of tests" or sets of tests which consists of particular tests. On the other hand, test is a tool for measurement of student's knowledge. Any test has specification: purpose, type, and target group; structure as number of blocks of different types; test and test result characteristics (see Fig. 1 made in OE (Kozaki, K., 2000), where p/o is "part of" link with a slot, a/o is "attribute of" link, upper part of a slot corresponds to role of the concept, and right side of a slot represents a class).

The purpose of a test determines the type of the test: mastering, kind of exam (pre-exam, midterm exam, quizzes, final exam, and post-exam), placement, admission, or diagnostics. A test could be in the form of paper, a computer, listening, demonstration, or oral. A test consists of blocks, and each block has specific item form, scoring scheme, direction, examples, and number of items. Each test item has stem, response alternatives with a correct answer and destructors, and characterized by level of understanding. We classify test items as *open question:* fill-in the blank, giving a short answer, and writing an essay; *multiple choices* items divided according to the amount of the given answers (2, 3, etc), and logic of answer; and *matching* problems with different amount of
columns.

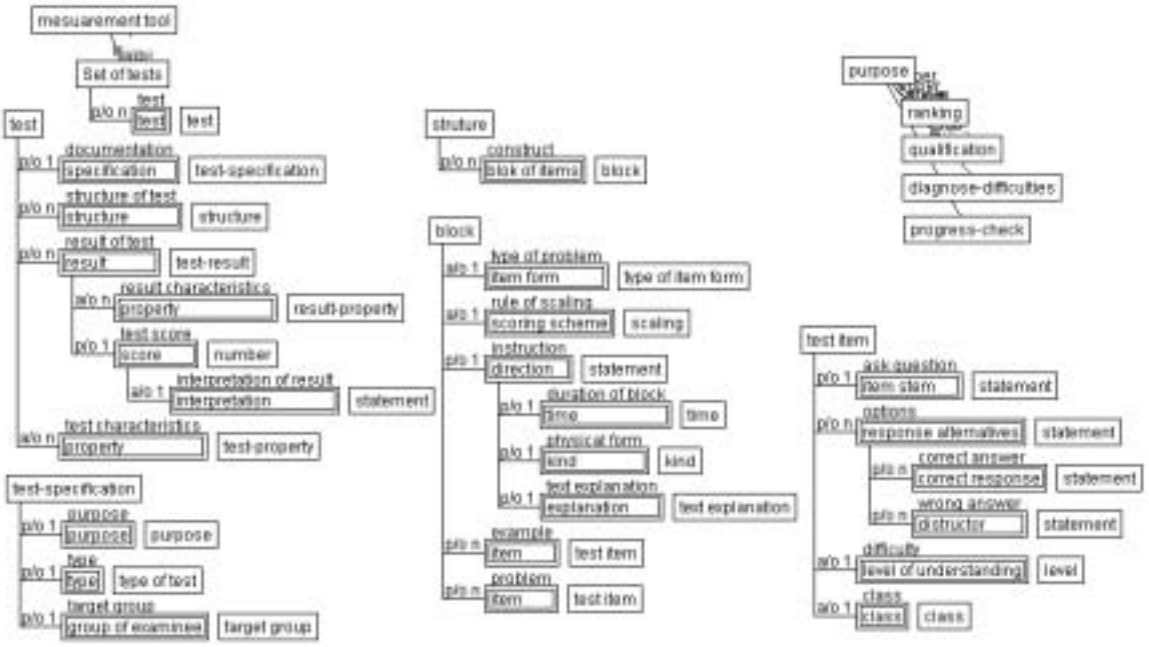


Fig. 1: Specification of main concepts of the test ontology (fragment).

The suggested test ontology can help to create a test, not to miss important components, to determine correctly a purpose and a way of testing, and to provide appropriate quality of a test. The ontology is based on classic test composition theory. It is possible to construct different ontologies of test or to change the suggested one. However, changes of for example test purposes might affect changes of test item forms, or omit the level of difficulty will give less information about a student stage. We suppose that our test ontology provides better quality of a test.

One of important test item characteristics is level of difficulty (see fig. 1). We would like to use popular in pedagogy Bloom's 6-layers model (Anderson, L.W., 2001):

1st level, ***knowledge*** in a narrow sense, is knowing facts and definitions of notions;

2nd level, ***comprehension***, is understanding of meaning of notions, objects, abstracts, and knowing simple rules;

3rd level, ***application***, corresponds to ability to apply known rules;

4th level, ***analysis***, checks understanding of relationships between elements, ability to select and compound different rules;

5th level, ***synthesis***, corresponds to ability to generalize knowledge;

and 6th level, ***evaluation***, uses meta knowledge.

The first three levels correspond to the domain knowledge, and the latter three levels reflect mostly the domain independent knowledge.

Test Generation Systems

Users of test generation system (TGS) are: a professor or an instructor who would like to design a test and/or analyze the test results, a learner or an examinee whose knowledge the instructor would like to check, and an administrator who manages the system. Also TGS might be useful for professional test developers. TGS can work in several regimes: ***test composition*** (either automatically or interactively with the instructor) and ***test***

application to check examinee's knowledge and to analyze the test result characteristics.

Functions of TGS:

Design: to determine the approach for the test composition, the test structure, and the method of scaling.
Input: a purpose of testing, a target group of examinees, requirements for the test characteristics. Output: a recommendable test structure and a scoring scheme.

Adapt: to fit the test structure (in an interactive mode).
Input: instructor's preferences about test characteristics. Output: a test structure and a scoring scheme.

Generate: to form test items.
Input: a type of item form, a level of difficulty. Output: test items.

Select: to extract test items from data base of ready test items.
Input: a type of item form, a level of difficulty. Output: test items.

Observe: to get information about the examinee's knowledge.
Input: test items. Output: answers on given questions.

Evaluate: to analyze the examinee's answers to obtain quantitative characteristics.
Input: answer on given questions; quantitative or qualitative evaluation made by the instructor. Output: quantitative and qualitative characteristics.

Improve: to enhance quality of the test on the base of analysis of test result characteristics.
Input: the test. Output: upgraded test.

Administrate: to provide security and access to the system recourses.
Input: commands. Output: system information.

At the present time TGS is at the stage of knowledge acquisition and representation, and KB of the system contains 30 rules.

Example: Integration of Functions

Let us consider TGS for a problem domain integration of functions (Soldatova, L., Mizoguchi, R., 2003). Suppose that an instructor gives to TGS the following input data: the purpose of test is ranking, the target group is master course student of "ISs in economy" specialty, and the amount of test items is 20. The system uses the follows KB rules:

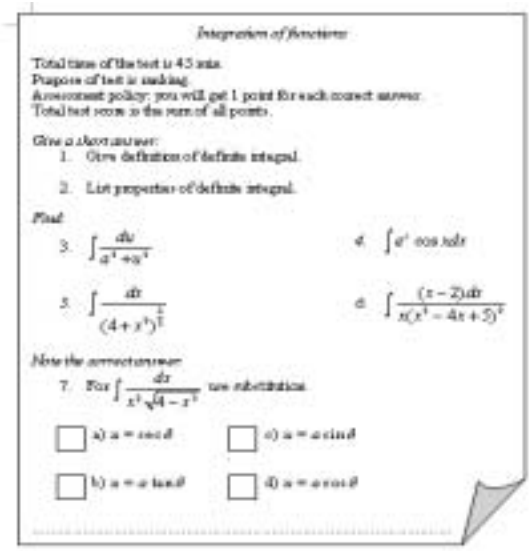
Rule 1: If a purpose is ranking, then the type of the test is placement.
Rule 2: If a type of the test is placement, then difficulty of items is middle.
Rule 3: If amount of test items is less then 100, then include more T/F items,
and so on.

Then TGS calculates the test structure and automatically generates test items of each form according to the level of difficulty or uses ready-made items from a test bank. The output is the test (see fig. 2); test documentation, and system explanation (see fig. 3).

Conclusions

A suggested approach for design of information systems for test construction, based on the test ontology, enables to systemize, analyze, and accumulate knowledge about test composition process. The test ontology contains description of structure of test, test and result properties, types of a test and items form, scoring schemas, and test composition approaches. For better diagnostics of student's knowledge Bloom's 6-layers model of understanding is used. Problem of test item generation is considered in the domain of integration of functions.

TGS can be used as a subsystem of an IES or as a separate IS for generation of test, sets of tests with identical level of difficulty, diagnostic of student's knowledge, and analysis of teaching materials, how fully all kinds of knowledge are presented. At the present time the process of test composition and knowledge assessment can be automated only partially. TGS can neither generate complicated test items nor check *open question* items.



Integration of functions

Total time of the test is 45 min.
Purpose of test is ranking.
Assessment policy: you will get 1 point for each correct answer.
Total test score is the sum of all points.

Give a short answer:

1. Give definition of definite integral.
2. List properties of definite integral.

Find

3. $\int \frac{du}{a^2+u^2}$
4. $\int e^x \cos kx\,dx$
5. $\int \frac{dx}{(4+x^2)^{\frac{1}{2}}}$
6. $\int \frac{(x-2)dx}{x(x^2-4x+5)^2}$

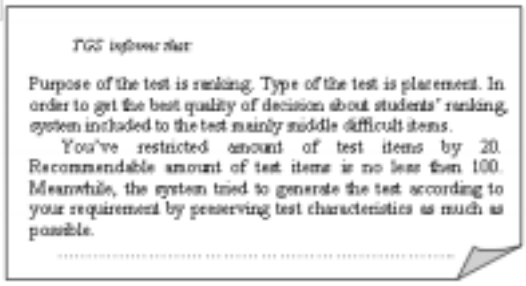
Note the correct answer

7. For $\int \frac{dx}{x^2\sqrt{4-x^2}}$ use substitution

☐ a) $x = \sec\theta$ ☐ c) $x = a\sin\theta$

☐ b) $x = a\tan\theta$ ☐ d) $x = a\cos\theta$

Fig. 2: An example of a test (fragment).



TGS informs that

Purpose of the test is ranking. Type of the test is placement. In order to get the best quality of decision about students' ranking, system included to the test mainly middle difficult items.

You've restricted amount of test items by 20. Recommendable amount of test items is no less then 100. Meanwhile, the system tried to generate the test according to your requirement by preserving test characteristics as much as possible.

Fig. 3: An example of system explanation (fragment).

References

Crocker, L., Algina, J. (1986). *Introduction to Classical and Modern Test Theory*. Wadsworth Group, Thomson Learning.

Mizoguchi, R., Sinitsa, K. (1996). Workshop on "Architectures and Methods for Designing Cost-Effective and Reusable ITSs." *Task Ontology Design for Intelligent Educational/Training Systems*. (pp. 1-21) in Proc. ITS'96, Montreal.

Anderson, L.W., Krathwohl, D.R. (2001). *A Taxonomy for Learning, Teaching, and Assessing*. Addison Wesley Longman.

Kozaki, K., Kitamura, Y., Ikeda, M., & Mizoguchi, R. (2000). Workshop on PKAW. *Development of an Environment for Building Ontologies which is based on a Fundamental Consideration of "Relationship" and "Role"*. (pp. 205-221) in Proc. PKAW, Australia.

Leithold, L. (1976).*The Calculus with analytic geometry*. Harper & Row, Publishers.

Soldatova, L., Mizoguchi, R. (2003). *Test Generation Systems*. (pp. 51-56) in Proc. SIG-IES-A203-09, Japan.

Soldatova, L. (1999). *Comparison of the Problems of Complicity*. (pp. 169-172, V. 2) in Proc. ICCE'99, IOS Press.