Metrics for the Evaluation of Test-Delivery Systems

Salvatore Valenti
Università Politecnica delle Marche-Ancona, Italy

INTRODUCTION

Most solutions to the problem of delivering course content supporting both student learning and assessment nowadays imply the use of computers, thanks to the continuous advances of information technology. According to Bull (1999), using computers to perform assessment is more contentious than using them to deliver content and to support student learning. In many papers, the terms computer-assisted assessment (CAA) and computer-based assessment (CBA) are often used interchangeably and somewhat inconsistently. The former refers to the use of computers in assessment. The term encompasses the uses of computers to deliver, mark, and analyze assignments or examinations. It also includes the collation and analysis of data gathered from optical mark readers. The latter (that will be used in this paper) addresses the use of computers for the entire process, including assessment delivery and feedback provision (Charman & Elmes, 1998).

A typical CBA system is composed of the following.

- Test-Management System (TMS) - that is, a tool providing the instructor with an easy-to-use interface, the ability to create questions and to assemble them into tests, and the possibility of grading the tests and making some statistical evaluations of the results
- Test-Delivery System (TDS) - that is, a tool for the delivery of tests to the students. The tool may be used to deliver tests using paper and pencil, or a stand-alone computer on a LAN (local area network) or over the Web. The TDS may be augmented with a Web enabler used to deliver the tests over the Internet. In many cases, producers distribute two different versions of the same TDS: one to deliver tests either on single computers or on a LAN and the other to deliver tests over the WWW (World Wide Web). This is the policy adopted, for instance, by Cogent Computing Co. (2004) with CQuest LAN and CQuest Net.

The TMS and TDS modules may be integrated in a single application as, for instance, Perception developed by Question Mark Computing (2004), or may be delivered as separate applications as it occurs for MicroTest and MicroGrade developed by Chariot Software Group (2004).

BACKGROUND

The interest in developing CBA tools has increased in recent years thanks to the potential market of their application. Many commercial products, as well as freeware and shareware tools, are the result of studies and research in this field made by companies and public institutions.

Thus, for instance, 42 quiz software products are referenced by the Soft411 (2004) directory, 23 by the Educational Software (2004) directory, and 8 by Assessment System Co. (2004). Moreover, it must be noted that almost all course management systems (Edutools, 2004) provide facilities for CBA. This noteworthy growth in the market raises the problem of identifying a set of criteria that may be useful to an educational team wishing to select the most appropriate tool for their assessment needs. The literature on guidelines to support the selection of CBA systems seems to be very poor since no other up-to-date papers are available on the Internet apart from the works by the author and his colleagues (Valenti, Cucchiarelli, & Panti, 2002a, 2002b).

The purpose of this paper is to provide a framework for the evaluation of a test-delivery system.

METRICS FOR THE EVALUATION OF A TDS

Three main functional modules roughly compose a TDS: a student interface, a question-management unit, and a test-delivery unit. Therefore, our framework for the evaluation of a TDS is defined in terms of criteria that may support the evaluation of each functional module and other criteria for the evaluation of the whole system, as shown in Table 1.

The evaluation of the interface is a qualifying aspect for the evaluation of a CBA system and obviously for a TDS. This becomes dramatically true if we take into account the fact that neither the teacher nor the students involved in the use of a TDS necessarily have a degree in computer science, nor may be interested in acquiring skills in this field. According to Nielsen and Molich (1990), the interface must be easy to learn, efficient to use, easy to remember, error free, and subjectively pleasing. Some further criteria that may be adopted to evaluate the usability of the interface are summarized in the following list.

Table 1. Metrics for the evaluation of a TDS

Issue		Metrics
Component Level	Interface	Friendly GUI (graphical user interface)
	Question Management	Types of Questions
		Question Structure (retries, tutorial building)
	Test Management	Help and Hints
		Restricted Availability
		Grading
System Level		Security
		Survivability
		Communication

- speak the users' language (multilinguality and multiculturality)
- be accessible
- provide feedback
- provide clearly marked exit points

The question-management unit of a TDS can be evaluated with respect to its ability to provide

- multiple attempts at solving a question (retries),
- feedback and tutorials on the topic covered by the questions, and
- capabilities for the inclusion of multimedia in questions.

The ability of providing retries may be of great importance for self-assessment since it is useful to improve the knowledge of the student whilst reducing the need for providing feedback and/or tutoring. On the other hand, the impossibility to change the answer to a question during an examination is often perceived as unfair by the students (Valenti et al., 2002b). It is worth outlining that allowing multiple attempts at question answering may affect the use of adaptive systems whenever item presentation depends on previous responses.

The feedback may be provided after each question (this solution being preferable for self-assessment), after a set of questions covering a given topic, or at the end of the test, and can be based on the overall performance. Furthermore, the feedback may be used to indicate the correctness of the answer, to correct misconceptions, or to deliver additional material for deepening and/or broadening the coverage of the topic assessed by the question. Tutorials represent an extended approach to provide additional information to the students. The existence of some facility to ease inclusion of tutorials in the TDS represents an important feedback aid. As an example, Perception provides explanation-type questions that may be used for "information screens, title pages, or to display large bodies of text" (Question Mark Computing Ltd., 2004).

The use of questions incorporating multimedia, such as sound and video clips or images, may improve the level of knowledge evaluation. This aspect may be of great importance, for example, in language assessment, where the comprehension of a talk or a movie can be assessed by recurring to multimedia only. The use of multimedia can raise issues related to portability and interoperability since it may require special hardware and software, both for the server delivering the questions and for the client used by the students. Furthermore, it may raise the costs for the adopted solution. These issues may not represent a problem whenever a Web-enabled TDS is selected since the nature of the World Wide Web is inherently multimedial. In this case, the choice of standard plug-ins for the most common browsers may reduce risks of portability and of interoperability. Since most plug-ins used to grant access to multimedia sources are usually free of charge, their use may not interfere with cost problems.

Among the issues taken into account to evaluate the test-management unit of a TDS, we have identified the ability to

- provide help and hints,
- make tests available at a given time, and
- allow scoring procedures.

The capability of a TDS to provide directions about the completion of the test and hints that usually are related to the contents of the questions represents a further measure of the ease of use of the application from the student's point of view.

Tests can be made either available or unavailable at a specified date and time. This allows test designers to specify exactly when people can access a test. It should be possible to leave out either or both restrictions to provide maximum flexibility. This lends itself nicely to the computer-lab set-