

INTERNATIONAL ASSESSMENT OF FUNCTIONAL COMPUTER ABILITIES

Ronald E. Anderson* and Betty Collist†

**University of Minnesota, U.S.A.*

†University of Twente, The Netherlands

Introduction

In 1987 the International Association for the Evaluation of Educational Achievement (IEA) began the International Computers in Education Study involving about 20 countries or educational systems. During the first stage each country conducted surveys of principals, computer coordinators, and teachers in elementary and secondary schools. Whereas this phase of the study was grounded in a conceptual framework of decision-making and educational change, the measurement of student achievement was planned for the second stage of the study. To accomplish this goal it was necessary to articulate what was intended as measurable outcomes of student exposure to computer use in education. In addition, these desired outcomes had to have associated test items in order to produce an internationally valid assessment.

In this paper we first identify various rationales as alternative conceptual frameworks for the domain of student assessment in the second stage of the IEA Computers in Education Study. Data from Stage 1 are presented to suggest some possible differences across systems in educational priorities. Then we elaborate the chosen conceptual framework into measurable objectives for student assessment, and finally describe the development and implications of an instrument to measure student achievement relative to these objectives.

Terms of Reference for the Domain of Assessment

"Computers in education" can encompass every purpose and procedure relevant to computers and learning, but the focus of the second stage of the IEA Computers in Education Study was narrowed to that of "computer education." The idiom "computer education" typically refers to learning or teaching about computers, their use, and products (including impacts). From the standpoint of student assessment, computer education is not what the teacher does or the student experiences, but what the student can demonstrate in terms of measurable conceptual and skill development relative to computers, their uses, and products. We use the term "computers" in its broadest sense to encompass any system with a CPU and all forms of computing, that is, all categories of procedures for controlling

this technology. Computing educators often distinguish between teaching about computers versus teaching with or through computers. In this discussion we presume that the domain of student assessment refers to learning about computers, how to use them, and their social implications, rather than what the student might learn about the content of traditional curriculum areas when computers are used as instructional agents.

Our domain is further defined by a focus upon "general computer education" in an attempt to narrow the problem space to education that is universal or nonexclusive and nonspecialist. Our interest is in computer education that is intended for all students in a system or a specific educational institution. For instance, an introductory college computing course intended for any student in the college would be considered "general computer education" while a first course in computing for majors in computer science is not.

"Computer Literacy" as a Basis for Student Assessment

General computer education has been discussed most often under the rubric of "computer literacy." Regrettably the phrase "computer literacy" developed a bad reputation after its popularity in 1980. (Its semantic history is like the term "socialist" in North America, where everyone enjoys its benefits but almost no one admits to supporting it.) When the 1985 National Assessment for Educational Progress (NAEP) conducted a nationwide testing of computing skills among US students, they called the domain of assessment "computer competence" to avoid the negative connotations of "computer literacy" at that time (Martinez & Mead, 1988). Nonetheless, "computer literacy" has become more semantically resilient and a recent search of the last 10 years of the educational literature using ERIC yielded over 1,000 items on "computer literacy" but only 10 on "computer competency". The concatenation of computer and literacy evokes expectations of communication skills that everyone should have in order to function in society. So, from the beginning of the debates on computer literacy, there has been the hidden premise that the requirements were universal and that it would become part of general education within schools. Indeed today many schools and businesses require basic computer skills, although more often it may be an implicit rather than an explicit requirement.

Rationales Underlying General Computer Education

A critical analysis of the domains of computer literacy and computer competence reveals different rationales implicitly or explicitly underlying the assessment domain of "general computer education" as delimited in the previous section. We have grouped general rationales into seven value-oriented positions expressed by protagonists for general computer education. These rationales are identified as: computer appreciation, programming, human capital, educational reform, constructivism, functionality, and empowerment. In contrast, antagonists toward general computer education tend to philosophically focus upon perspectives we will call: humanism, economic, and curricular integration. It should be noted that our conceptualization is similar to that of Hawkrige (1991) and associates (Hawkrige, Jaworski, & McMahon, 1991). However, our approach is different in that we have focused upon general computer education, not any type of educational computing. Furthermore, in delineating categories of rationales we

have identified interrelated sets of arguments, not just a single motive or a reason for something. Next we will review each of the protagonist rationales we have differentiated.

Computer Appreciation. This rationale is based mainly on the presumed need for a cultural perspective on the role of the computer. The most "intellectually elite" proponents of this approach are computer scientists who seek to educate the public in the value of computer technology. A recent television series, "The Machine that Changed the World," illustrates the prevalence of this perspective (WGBH-TV, 1992). Produced by the British Broadcasting Corporation, WGBH in Boston, and NDR/Germany, with funding from the Association for Computing Machinery, the National Science Foundation, and Unisys Corp., this six-hour series purports to educate the general public about what it needs to know about computers. Yet most of the content is either historical detail or futuristic projects in computing. Reflecting this rationale for computer education, almost every general computer text is filled with historical facts and tantalizing forecasts designed to get the student to "appreciate" computers, much like music appreciation classes. Many textbooks and some teachers seem to have an unspoken behavioral objective of getting the student to be impressed by the power of computers as many times as possible. This perspective is often packaged and promoted at the elementary/secondary school levels as "computer awareness." Whatever it is called, it is frequently conceptualized as an introduction to information technology vocabulary, elementary computer-related concepts, and a cursory review of computer applications. In the early years of the computer literacy curricular movement, many computer educators claimed that "computer awareness" was unimportant and too superficial. In contrast, in many countries, what is most often understood as "computer literacy" in the early 1990s is hands-on, computer- interaction skills such as keyboarding mixed with different simplified elements of "computer awareness." Regrettably, however, the consequences of the computer awareness tradition sometimes yield very superficial, misinformed classroom instruction (Rosenberg, 1991).

Programming. In the early years of educational computing, Luehrmann (1981) and many others believed that every student should be taught elementary programming. Their most common rationale for teaching programming was the logical discipline that came from problem analysis. Before the widespread availability of application software such as word processors, database packages, and spreadsheets, one often had to write a program in BASIC, Fortran, or Pascal to accomplish a computer-related task. Therefore, in the early 1980s the argument to teach programming was compelling, but by the late 1980s when application software became widely available, this approach was called into question. The current trend in software is toward macros and scripts, which are programming languages embedded within software packages. Ironically, many advanced software applications now require systems analysis and programming, even though a higher level language like BASIC or Pascal is no longer necessary. Proponents of the programming rationale often claim that a systems point of view and its exercise is useful generally for problem solving.

Human Capital. In contrast to the "computers for their own sake" perspective, many arguments for general computer education use terms such as labor force productivity, global competitiveness, career advancement, and training for the workplace as their conceptual framework. However, the concept of "computer literacy" from a human capital position, has been criticized; for example, Robins and Webster (1989) suggest that all

these concerns reduce to a view of computer education as little more than vocational skill training. While there are many good reasons for job training relative to computing skills, many educational systems do not accept job training as an appropriate dominant goal for general education. But as Attewell (1989) notes, there are learning areas such as keyboarding, where the training needs of the industrialists intersect with the educational needs of students, if not citizens in general.

Educational Reform. A very different perspective for general computer education relates to its relationship with educational reform. From this position, computers are seen as having the potential to become the single most potent agent of change in education during the twentieth century. Many educators have seized upon this as an opportunity for bringing all kinds of reforms into education including individualization, discovery, cooperative learning, higher-order thinking, performance testing, and so on (cf. Collins, 1991; Pogrow, 1990; Sheingold, 1991). In order for the computer to play a major role in broad reform, all students will need to know enough about and have enough skill to minimally use computer systems. Consequently, general computer education is a prerequisite of reform movements, at least in the short run. Some futurists claim that in the 21st century computers will be so usable that no significant learning time relative to their use will be required. A more realistic view is that future information technology will make many computer skills obsolete but will replace old learning requirements with new ones.

Constructivism. For the past two decades there have been pioneers who have been vigorous advocates for various types of student activities reflecting a constructivism rationale, such as LOGO programming (Papert, 1980; Taylor, 1980) and reasoning exercises (Pogrow, 1990). Some of the constructivist advocates appear to have little interest in general computer education. However, even while these constructivists concentrate upon specific, creative problem-solving activities through the mediation of certain types of computer use, widespread infusion of these constructivist approaches would require more and more attention to "entry level" general computer skills such as keyboarding and more advanced skills associated with higher-order informatics.

Functionality. The essence of this rationale is the ability to control one's resources to get things done, that is, to function effectively with one's information-related tasks. This is a very practical concern with optimal use of information technology. Within this framework, functioning with information technology requires certain knowledge, skills, and attitudes, the specifics of which depend upon one's environment and goals. While the "functionality" label has not been explicitly applied to computer education, the general rationale was articulated even in the early days of educational computing under different terminology, such as that of a "comprehensive view" (Anderson, Klassen & Johnson 1981), who proposed that each student should gain an "understanding of computers that enables one to evaluate computer applications as well as to do things with them," and that schools should "provide students with constructive computer experiences."

Current trends to teach students to use computers for writing, thinking, and other information-handling work, are certainly consistent with this functionality rationale. Teaching students to evaluate and assess the outcomes of computer applications also is consistent with this approach. The literature on general computer education is becoming more and more reflective of functionality as its basic rationale; however human capital and

constructivist arguments are also still strong (Collis & Oliveira, 1990; Hawkridge, 1991; Shaw, 1991). Optimal functionality lies in between minimal computer literacy and advanced computer expertise, hence we might call it "computer fitness."

Empowerment. "Empowerment" could be seen as the social dimension of functionality. The term empowerment has roots in social reform and is generally associated with attempts to equalize the relative power of different social groups. For instance, training disadvantaged groups in functional skills for solving their problems is considered to be empowerment education. Likewise the current concern with inequity in reference to computer access and skills is seen as an empowerment challenge. This may lead to the conclusion that general computer education is necessary to insure that women, minorities, the disabled, and others are not excluded from the advantages of using computing and other information technology. Since much computer learning occurs in the home and is therefore income- dependent, it is argued that educational systems have an unique obligation to provide for general computer education for all of their students (Becker, 1987; Collis, 1985).

In contrast to the above seven arguments underlying general computer education, what might be called "general computer education antagonists" tend to be based upon one of three types of arguments: humanism, economics, and curricular integration.

Humanism. Pessimism about general computer education is often founded upon humanistic arguments (Robins & Webster, 1989; Sloan, 1984). This reasoning generally contrasts humane and social values with technical, rational, and economic ones, claiming that computerization tends to detract from that which is most truly human. The more sophisticated reasoning against general computer education from a humanist perspective concentrates on educational needs and priorities and often justifies its scepticism about general computer education by arguing that the value of computer use in education, relative to "real" educational needs, has not been adequately substantiated by educational research (Clark, 1985; Robins & Webster, 1989). It is often noted that more research is needed on what the specific outcomes of different types of general computer education actually might be.

Economics. A conceptual framework emphasizing economics is not so much a rationale as a pragmatic issue; however, the importance of economically measured costs and benefits often does take on an ideological character. Critics of general computer education frequently point to the vast cost of equipping schools with computing facilities for all students. In addition, the cost of replacing obsolete hardware and software, to say nothing of teacher education, is far out of proportion relative to reduction of educational needs. In the absence of data and with the inability to predict the future, educational policy analysts tend to base their cost-related positions on their traditional or technological orientations, rather than on objective strategies to operationalize a cost-effectiveness approach in the context of general computer education (Moonen, 1990).

Curricular Integration. Long before computers arrived in the schools, there were curricular debates over the integration of technologies within curriculum areas as instructional resources. These debates often were fairly narrow, for instance, whether or not science experiments should be taught separately from science theory courses or whether applied

statistics should be taught in statistical theory. The now ubiquitous computer has made the issue of curriculum integration much broader, because now, arguably, every subject can benefit from some aspect of information technology. There are two types of integration that are very easily confused. One, which we can call infusion, is the transfer of computing technology into the classroom to support the instruction of existing subjects such as mathematics and language. (A related controversy is whether or not to group computers in separate computer laboratories or distribute them among traditional classrooms.)

The second dimension with respect to integration relates to the spreading of responsibility for student acquisition of computer-education competencies across various existing subject areas rather than establishing a new curriculum area called "general computer education" (Collis, 1988). In this discussion we use the term integration to refer to this latter type of development, even though much of the literature uses the term integration to refer to what we call infusion (see for example, van den Akker, Keursten, and Plomp, 1992; Pelgrum and Schipper, 1991).

In many countries, including the United States, a trend toward computer integration can be observed. For instance, the Minnesota State Department of Education (MDE) in 1990 adopted a policy of "integrated information technology." This policy "mandates" instruction in information technology, but provides no means of assessment and discourages separate courses in computing (Minnesota Department of Education, 1991). On the surface it would seem that the integration rationale would promote greater general computer education, but in practice it may not, due to inadequate planning and commitment of resources, especially teacher training (Maddux, 1991). The integration rationale is inadequate to the extent that it is a "sink or swim" model in which students and teachers are given computing resources to use without a system for assessing computer functionality with systematic instruction whenever such skills are lacking. Such systems have not been formalized and all too often the students, and their teachers, may be metaphorically pictured as sinking into some sort of "electronic cyberspace", never to return.

Evidence from Stage 1 of Different Rationales for Computer Education

The foregoing rationales were not explicitly articulated prior to the planning of Stage 1 of the IEA Computers in Education Study, however some of the findings relate to these approaches and will be discussed below.

The international survey data plan of Stage 1 called for randomly sampling at least 200 schools in at least one of the three grade levels: elementary schools, lower secondary, and upper secondary (Pelgrum & Plomp, 1991). In each school three separate surveys were administered: one for general information from the principal, another for information from the computer coordinator or another technical person, and one for selected teachers. Screening questions appeared in the principals questionnaire to determine whether or not each school currently used computers for instruction.

Table 1 shows for each country the percent of lower and upper secondary schools that used computers for instruction. In addition, it gives the percents by country of lower and upper secondary schools that offer general computer education, and within that, offer such material in a "separate course." The question asked of the computer coordinator was: "In which of the following contexts or subjects do students get a substantial amount of instruction about computers (computer education, computer literacy, programming, etc. --

not just how to run software)?" Respondents were instructed to check as many of the following that apply: "separate course," mathematics, science, native language, foreign language, creative arts, social studies, commercial studies, technology, specific technical courses, home economics, "informal", or other.

Table 1: IEA Comparisons of Secondary Schools: Percentage of Schools Using Computers for Instruction and Percentage of Computer-Using Schools Providing General Computer Education in a Separate Course

Country	LOWER SECONDARY		UPPER SECONDARY	
	% Schools Using Computers	Separate Comp. Course	% Schools Using Computers	Separate Comp. Course
Belgium (Flem)	78	87	98	83
Belgium (Fren)	93	90	93	84
Canada (BC)	99	79	99	79
China	61	96	-	-
France	99	10	99	25
Germany (Fed)	94	81	99	79
Greece	5	99	4	96
Hungary	-	-	99	55
India	-	-	7	81
Israel	-	-	82	94
Japan	36	24	94	30
Luxembourg	99	93	-	-
Netherlands	87	91	69	53
New Zealand	99	67	99	94
Poland	-	-	72	89
Portugal	53	35	72	68
Slovenia	-	-	94	95
Switzerland	74	90	98	84
United States	99	51	99	59

Note. - data not collected.

Source: Pelgrum & Plomp, 1991. p. 18 and 40.

Only the percentages of "separate courses" are included in Table 1, and these are shown as evidence of the worldwide trend toward establishing courses just to teach general computer education. Countering this trend are several countries with an apparent commitment to curricular integration. At the lower secondary level these countries include Portugal, Japan, France, and the USA. While Portugal reported their students get computer education from "informal" sources, France and Japan indicated that students receive it in general technology courses or mathematics courses, and in the United States it appears to have been dispersed across many different subjects.

A similar pattern holds for the upper secondary level where separate computer courses predominate except for France, where most students reportedly receive their computer education in "commercial studies," and in Japan where most students get it in

"commercial studies" or "general technology." In the Netherlands, Hungary, and the United States about half of the schools reported that their students got their computer education in other than separate courses.

We do not know exactly what computer coordinators viewed as a "separate course" when they responded to this question. They could have been describing an instructional unit of only a few days. Nonetheless, these results do supply evidence that in 1989 there was relatively little support for the curricular integration rationale as we defined it above.

It is extremely difficult to find good measures of rationales for computer education to which individuals, much less educational systems, subscribe. In further analysis from Stage 1 as well as the student data from Stage 2 it will be possible to examine the content of the general education courses. However, for the moment we will examine the content of the training computer education teachers reported they received. Obviously, there are many diverse reasons for the content of teacher training programs, but the overall patterns may give us some hints about the predominate underlying goals of computer education.

Table 2: IEA Comparisons of Secondary Schools: Percentage of Secondary Computer Education Teachers who Received Training in Three Topics: History, Editing or Word Processing, and Program Structure

Country	Lower Secondary			Upper Secondary		
	History	Editing/ Word Proc.	Program Structure	History	Editing/ Word Proc.	Program Structure
Belgium (Flem)	44	70	68	47	65	80
Belgium (Fren)	44	81	75	60	72	89
Canada (BC)	--	--	--	31	53	42
China	83	34	84	--	--	--
France	--	--	--	38	74	57
Germany (Fed)	28	46	50	44	43	76
Greece	96	91	84	69	49	81
Hungary	--	--	--	86	55	91
India	--	--	--	51	68	59
Israel	--	--	--	44	32	87
Japan	35	44	51	55	53	76
Luxembourg	52	82	71	--	--	--
Netherlands	56	80	70	--	--	--
New Zealand	--	--	--	--	--	--
Poland	--	--	--	68	57	82
Portugal	31	61	36	74	72	65
Slovenia	--	--	--	--	--	--
Switzerland	63	79	55	58	71	63
United States	53	79	59	45	72	56

Note. -- indicates either nonparticipation of country in survey or fewer than 20 cases in the percentage base.

Source: Pelgrum & Plomp, 1991, p. 126, 129.

Each teacher was asked to check off those computer education topics on which they had received either general training or in-service training. While there were about 25 topics

in the list, only 3 topics were selected for presentation in Table 2 as evidence of the substantive emphasis in their training. Table 2 gives the percent of computer education teachers that received training in each of three topics: "History/evolution," "Editing/Word Processing," and "Program Structure." Extensive training in the history of computers would indicate a "computer appreciation" orientation to computer education, training in word processing would suggest an emphasis in "functionality," and training in program structures implies a commitment to the programming rationale. The table reveals that almost none of the countries, at either secondary level, provided training in history more often than they did word processing or programming. Generally the history of computing would appear to be relatively unimportant as content for general computer education.

Secondly, the data reveal that teachers in a majority of countries are more likely to receive training in programming than in word processing. In contrast to this trend, computer education teachers were more likely to receive word processing than programming instruction at the lower secondary level in five countries: Luxembourg, The Netherlands, Portugal, Switzerland, and the United States; at the upper secondary level the countries were Canada, France, India, Switzerland, and the United States.

It may well be that in some systems word processing is a subject that teachers are expected to learn on their own or through informal channels. However, the greater likelihood of teacher training in programming within some countries undoubtedly reflects a greater commitment to the programming rationale for computer education. During the past few years in some countries, including the United States, there has been a noticeable movement away from the programming approach. It will be interesting to see if the Stage 2 data confirms such as trend.

Student Assessment in the IEA Computers in Education Study

Regardless of which considerations motivate general computer education, there is a growing recognition of the value of assessing student outcomes in computer-related activities in school. In particular, empirical data is needed to assess student growth and achievement and to monitor the payoff of future computer-related programs in schools. The extent to which perspectives are being translated into student competencies was not directly addressed in Stage 1 of the IEA Computers in Education Study.

The IEA General Assembly in 1984 authorized the development of a proposal for the study of Computers in Education, and in 1986 approved the plan which included direct student assessment (Wolf, Plomp & Pelgrum, 1986). As the Stage 1 data collection progressed in 1989, the International Coordinating Center (ICC) drafted several conceptual framework grids of computer-related content parallel to the grids of prior IEA studies such as the Second International Mathematics Study (Travers & Westbury, 1989), and the NAEP Computer Competence study (Martinez & Mead, 1988; National Assessment for Educational Progress, 1985). These grids included both a content grid (topic by process matrix) and curriculum grids (topic by country/system matrices). During the drafting of these grids, a number of specialists in computer education and representatives from participating countries reviewed them and suggested refinements.

The content grid was a matrix with the columns representing process categories and the rows giving topics and subtopics. These topics included information technology in society; computers: hardware and system software; applications of information technology; problem analysis, and programming. Many subtopics within these four topics appeared as

the rows of the preliminary grid. These topics were not expressed as student outcomes but only as topics or subtopics, e.g., "problem solving techniques", "software engineering", "Fortran", "mouse", "local area networks", "history of IT and computers", etc. Many of the suggested topics were very broad, such as "Impact of IT on society, education, economy, culture, job qualifications, types of jobs, quality of life" while other rows were narrow in focus, such as "printer". These and all the remaining topics and subtopics were derived from an international search of computer education literature, including curriculum materials, textbooks, and syllabuses.

The grid columns denoted the following cognitive process categories: "Knowledge", "Comprehension", "Operation", and "Design," which coincided with the NAEP categories except for the addition of the "Comprehension" category. Consultants and national research coordinators were asked to comment on the completeness and clarity of the test grid process categories as well as the substantive topics and subtopics. While discomfort was expressed over the clarity of the categories, the grid remained essentially the same for about a year.

During the same period, the IEA Computers in Education international center staff collected over 1,000 test items which were mostly multiple choice items, from sources around the world and entered these into a database. Various rounds of consultation occurred during this preliminary period, including preliminary ratings and classifications of the test items in the item bank.

By September, 1990, current versions of the content grid and the best or prototypical test items were mailed to the national research coordinator in each participating country. These coordinators along with the staff and consultants of the ICC assembled in Frascati, Italy in September, 1990. The primary task of this meeting was to develop a first draft of the student assessment instrument for Stage 2 of the IEA Study.

Establishing a Theoretical Framework for the Student Assessment

The specification of the domain for testing began in 1989, three years before the target testing date of the Spring of 1992. This posed a great challenge because not only were there numerous differences in curricula within and across countries, but with a rapidly evolving subject matter, the content could be obsolete by the date of testing.

One consequence of the method used to assemble the draft test grid was that, in general, the material was already a number of years old due to the inevitable time lapse due to development, publishing, and distribution. While a few years is of little consequence with regard to some established curriculum areas, with computer-related curriculum it is of critical consequence because terminology as well as procedures and concepts change substantially from year to year. A major dilemma for the meeting at Frascati was the need to produce a plan for content and measurement that would be piloted one year later, used for an international assessment two years later, analyzed three years later, and communicated to researchers and policy decision-makers four years later.

As was noted earlier, the collection of topics for the content grid had been done in a highly inductive manner. Such an approach will implicitly reflect the many conflicting assumptions that underlie the topics acquired from various countries. An alternative approach would have been to assemble computer education representatives from each country to vote on a position or positions to guide the specification of the IEA Computers in Education test domain.

A new direction emerged from an extended discussion with different rationales and combinations of rationales underpinning the perceived relevance of "general computer education." For example, an appropriate domain for an assessment based on a "constructivist" perspective will clearly be very different from that based on a "human capital" rationale. All of the orientations discussed in the first part of this paper were implicitly represented in the content grid, although several were much more pronounced. The most dominant orientations reflected in the underlying grid were the "programming" and the "functionality" perspectives, although the "computer appreciation," "human capital," and constructivist orientations were well represented too.

After prolonged discussion and critique of the content grid, the participants agreed that another approach for the domain specification was needed. An alternative was needed to minimize the obsolescence problem and to emphasize practical, functional students needs. We could have selected the items that most reflected the "functionality" perspective, however, most of the items of this type related to applications and application software and did not address the functional needs related to general user interaction, especially with equipment, operating systems, and system malfunctions.

After much discussion, the participants settled on an assessment compromise: the core or "mainstream" assessment would reflect the "functionality" perspective, but an optional "Programming Test" would be developed for those countries oriented toward programming. The new core test would include many of the functionality-oriented items from the pool, but would require some new ones dealing with general user actions. From this discussion emerged a name for the proposed core test: "Functional Information Technology Test," which of course had a fitting acronym, FITT.

The authors of this paper took the lead in delineating a test domain and drafting or revising individual test items based on this new "functionality" direction. Avoiding obsolescence and to measure practical requirements of user interaction were the two most significant goals of this effort.

The next step of the IEA Computers in Education student assessment test development did not follow traditional procedures. We had to work quickly in order to take advantage of the international gathering in Frascati. Therefore we used our own experience to predict measurable outcomes for FITT, asking ourselves what a person who could function well in terms of common uses of computers would know and be able to do in about five years ahead. Also, we did not limit ourselves to what might be taught formally in school settings, but considered what a well-informed consumer might learn from home or work.

Some criterion scenarios emerged as "rules of thumb" to guide us in developing both content objectives and test items. Here are some sample scenarios: What would a young adult need to know in order to have a reasonable understanding of what was being referred to in a newspaper advertisement for a local computer store? What sort of "trouble shooting" insights and strategies might a person have to know and then use with information technology for various commonplace applications such as word processing? What are typical types of computer applications likely to be found in the school, office, home, and workplace in 1995? What are different strategies for trying to get started with a new software package or for deciding what to do when something you are trying to do with a computer is not working as expected?

Within the conception of functionality that emerged we identified four themes, and defined a part of the content for each theme. The complete list of content objectives is

given in Appendix A, but the four themes, which also are the section headings of the content objectives, are:

- A. The Computer as Part of Information Technology: What are Computers and How do They Operate?
- B. Using Computers: What are Your Computer-Handling Skills?
- C. Applications: What Can You Do With Information Technology?
- D. Attitudes, Opinions, Social Context and Ethics.

Each section will be described briefly.

A. The Computer. This section consists largely of vocabulary and knowledge about basic aspects of information technology. The associated test items would be best described as "knowledge" items, although some might require comprehension of fundamental concepts.

B. Using Computers. This is the most innovative and most important part of the content objectives and the emergent test. In addition to elementary operations of user interaction such as cursor control, this section includes skills with handling disks, backing up data, and dealing with common malfunctions in the course of computer and software operation.

C. Applications. This section combines knowledge about common application software capabilities with operational skills in using the software. In this regard this section of the content grid resembles the applications section of the NAEP computer competence grid.

D. Attitudes. The last section on attitudes was not used as a guide for items in the cognitive test, but like many other large-scale assessments was used for defining attitude and opinion questions in another part of the survey. This will be discussed in a separate report.

From Content Objectives to Assessment Instruments

Next we turned our attention to the 1,000 item data base of test items and made a professional judgment as to the fit of each item to any of the behavioral objectives on the list. We literally regrouped the accumulated items, each printed on separate small pieces of paper until we identified 83 that fit reasonably well to the specified objectives. Ten of the stated objectives had no representation and 11 of the objectives had only one. We drafted new test items for the under-represented objectives and revised many of the selected items. One of the main considerations was the anticipated cultural difficulties in the translation of items to other languages. The meetings of the National Project Coordinators (NPCs) at Frascati and Kriopigi, Greece, one year later, provided an ideal context for immediate feedback regarding cultural differences and translation difficulties. In Frascati, each country's Coordinator commented on every draft item with respect to its suitability for students in their own country at different age levels.

After the meeting in Frascati, the objectives and test items continued to be reviewed and refined by other teams of national experts as well as by those involved in the development. In the Fall of 1990, the ICC distributed pilot test instrument drafts to all countries for translation followed by back translation. During the Spring and Summer of 1991, the FITT instrument as well as other survey instruments were pilot tested in eight countries. These results were taken to the annual meeting of the National Project

Coordinators, which in 1991 took place in Kriopigi, Greece. On the basis of various substantive as well as technical consideration, a final assessment instrument was specified for each of the three grade levels to be tested.

In this paper we purposely focused on only one aspect of student assessment for IEA Computers in Education. Other reports will detail other aspects of the test development which are of scientific and practical interest. For example, we have not yet described the parallel process involved in the construction of attitude items including ethics. Likewise this report has not described the extensive process involved in the construction of instruments for performance assessment. In addition, the parallel development of a "programming" test is noteworthy as well.

Discussion

We have sought to draw attention to the importance of articulating an educational rationale prior to domain and test development. Not all subject matter assessments will require such attention. But when the curricular content as well as the tools for delivering the instruction rapidly evolve, a strategic curricular analysis is needed. When the curriculum also is fragmented or relatively uncrystallized, a traditional curricular analysis may not be satisfactory without review, delineation and specification of the underlying orientations.

With schools throughout the world investing heavily in computer resources and in the support of student activities with information technology, the development of valid and reliable tests to assess student outcomes will become critical, particularly for administrators who must weigh many competing demands for limited funds and who need cost-effectiveness measures in order to justify particular computer costs in schools (Moonen, 1990).

The approach indicated in this paper, beginning with an articulation of the underlying rationale relating to computer use in education, and then translating that position into domain objectives which apply forward rather than backwards in time, can provide an effective strategy for ongoing impact assessment and planning. The choice of a rationale expression must be a reflection of the complex and unique mix of norms and context in a given situation. Nonetheless, we believe that the "functionality" perspective will prove to be the most useful for policy and planning in the future. We base this conclusion, in part, on observations and analyses of the implications of other rationales in general computer education (Collis & Oliveira, 1990; Hawkrige, 1991); however, the most compelling foundation for the functionality perspective is its emphasis upon practical adaptation to the changing technology and the evolving needs of students to cope with it.

Summary

After delineating the major rationale for computer education, data are presented from Stage 1 of the IEA Computers in Education Study showing international comparisons that may reflect differential priorities. Rapid technological change and the lack of consensus on goals of computer education impedes the establishment of stable curricula for "general computer education" or computer literacy. In this context the construction of instruments for student assessment remains a challenge. Seeking to anticipate and measure what educators will view as the essential computer-related abilities for students in the mid-

1990s, the second stage of the IEA Computers in Education Study developed a student assessment instrument grounded in the perspective of "functionality," student prerequisites to functioning effectively with practical information-related tasks. The threat of test obsolescence as well as philosophical differences among the experts in their goals for general computer education challenged traditional test construction procedures. The resulting content objectives and test procedures can serve as guideposts for research and planning in computer education.

Note

1. The authors wish to acknowledge the very significant contributions to this work by numerous persons in the IEA Computers in Education Study, especially Tjeerd Plomp, International Project Director, and Willem J. Pelgrum, International Coordinator.

References

- Akker, J.J.H., Keursten, P., Plomp, Tj. (1992). The integration of computer use in education. *International Journal for Educational Research*, 17 (1), 65-75.
- Anderson, R.E., Klassen, D.L., & Johnson, D. C. (1981). In defense of a comprehensive view of computer literacy. *The Mathematics Teacher*, 74 (9), (Dec), 687-690.
- Attewell, P. (1989). The attack on computer literacy. *Social Science Computer Review*, 7 (1), (Spring), 7-26.
- Becker, H. J., & Sterling, C. W. (1987). Equity in school computer use: National data and neglected considerations. *Journal of Educational Computing Research*, 3 (3), 289-311.
- Clark, R.E. (1985). Confounding in educational computing research. *Journal of Educational Computing Research*, 1 (2), 137-149.
- Collins, A. (1991). The role of computer technology in restructuring schools. *Phi Delta Kappan*, 73 (1), 28-36.
- Collis, B. A. (1985). Reflections on inequities in computer education: Do the rich get richer? *Education & Computing*, 1 (3), 179-186.
- Collis, B. (1988). *Computers, curriculum and whole-class instruction*. Belmont, CA, USA: Wadsworth.
- Collis, B. A., & Oliveira, J. O. (1990). Categorizing national computerrelated educational policy: A model and its applications. *Information Technology for Development*, 5 (1), 45-68.
- Hawkrige, D., Jaworski, J., & McMahon, H. (1991). *Computers in third-world schools: Examples, experience and issues*. New York: St. Martin's Press.
- Hawkrige, D. (1991). Machine-mediated learning in third-world schools? *Machine-Mediated Learning*, 3, 319-328.
- Luehrmann, A. (1981). Computer literacy - What should it be? *The Mathematics Teacher*, 74 (9), 682-86.

- Maddux, C. D. (1991). Integration versus computer labs: An either/or proposition? *Educational Technology* (October), 36-40.
- Martinez, M. E., & Mead, N. A. (1988). *Computer competence - The first national assessment*. Princeton, NJ: Educational Testing Service.
- Minnesota Department of Education (1991). *Model learner outcomes for information technology Education - Curriculum Integration Guide*. St. Paul, MN, USA: Minnesota Department of Education.
- Moonen, J.(1990). *Costs, effectiveness, and expectations: Concepts and issues with respect to educational technology*. Sidney, British Columbia: Educational Technology Centre.
- National Assessment for Educational Progress (1985). *A framework for assessing computer competence: Defining objectives*. Princeton, NJ, USA: Educational Testing Service.
- Papert, S. (1980). *Mindstorms - Children, computers, and powerful ideas*. New York, NY, USA: Basic Books.
- Pelgrum, W. J., & Plomp, T. (1991). *The use of computers in education worldwide*. Oxford, England: Pergamon.
- Pelgrum, W. J., & Schipper, A. T. (1991). Indicators of technology integration in education. Unpublished paper, Department of Education, University of Twente, The Netherlands.
- Pogrow, S. (1990). *HOTS - Using computers to develop thinking skills in students at risk*. New York: NY Scholastic.
- Robins, K., & Webster, F. (1989). Computer literacy: The employment myths. *Social Science Computer Review*, 7 (1), (Spring), 7-26.
- Rosenberg, R. (1991). Debunking computer literacy. *Technology Review*, 94 (1), 58-65.
- Shaw, M. (1991). Informatics for a new century: Computing education for the 1990s and beyond. *Education and Computing*, 7 , 9-17.
- Sheingold, K. (1991). Restructuring for learning with technology: The potential for synergy. *Phi Delta Kappan*, (September), 17-27.
- Sloan, D. (Ed.). (1984). *The computer in education. A critical perspective*. New York: Columbia University, Teachers College Press.
- Taylor, R. P. (1980). *The computer in the school: Tutor, tool, tutee*. New York: Teachers College Press.
- Travers, K.J., & Westbury, I. (Eds) (1989). *The IEA Study of Mathematics II: Analysis of Mathematics Curricula*. Oxford (UK): Pergamon.
- Van Den Akker, J., Keursten, P., & Plomp, T. (1992). The integration of computer use in education. *International Journal of Educational Research*, 17 (1), 65-76.
- WGBH-TV (1992). *A teacher's guide for The Machine that Changed the World*. Boston, MA, USA: WGBH-TV.

Wolf, R., Plomp, Tj., & Pelgrum, W. J. (1986). *Computers in education -Design and planning*. Enschede, Netherlands: Department of Education, University of Twente.

The Authors

RONALD E. ANDERSON obtained the M.A. and Ph.D. degrees at Stanford University, California, USA. He is Professor at the University of Minnesota, Minnesota, USA. He serves as the United States National Project Director for the IEA Computers in Education study.

BETTY COLLIS is on the faculty in the Department of Education at the University of Twente, Enschede, The Netherlands. Previously she served on the faculty of the Department of Psychological Foundations, University of Victoria, British Columbia, Canada.

Appendix A

CONTENT DOMAIN OBJECTIVES

Part A. The Computer as Part of Information Technology: What are computers and how do they operate?**A1. General Concepts:***The Input-Processing-Output Model*

- A1a. Show comprehension of the essential functions, relative to input, processing, and output, of information processing systems.
- A1b. Evaluate at the conceptual level the most likely sources of unsuccessful program operation relative to the "input-processing-output" model of computer operation.

Program-related concepts and vocabulary

- A1c. Distinguish between hardware and software.
- A1d. Be aware that a program directs a computer to carry out certain functions related to input, processing, or output in logically related steps.
- A1e. Be aware that programs are written according to the syntax of programming languages, such as, for example, BASIC, Logo, Pascal, and C.

Concepts related to processing

- A1f. Know that processing occurs in a special unit (the CPU) that can be located in a user's own terminal or in a "remote" computer system.
- A1g. Distinguish between information being processed in the active (working) memory of the computer and information saved on storage media.

A2. Characteristics of Components of Computer Systems

- A2a. Distinguish between a computer (processor or microprocessor) and peripherals.

Input devices

- A2b. Identify common input devices for computers such as: keyboard, mouse, optical reader, and sensor.

Output devices

- A2c. Be aware of different categories of the most commonly used output devices, namely printers and monitors.

Storage devices and media

- A2d. Identify some different forms of the most currently popular storage media for microcomputers: diskettes and hard disks.
- A2e. Identify different devices used to read information from commonly available microcomputer storage media: external drives, fixed or hard drives.

A 3. System Software (Operating Systems)

- A3a. Identify major functions of operating systems (system software) with respect to program operation and file management.

A 4. Trends with Respect to Technical Developments

- A4a. Be aware of current trends in the technical development of interactive video and other computer-related multimedia.

- A4b. Be aware of current information technology trends such as reductions in size combined with increases in speed, power, and storage capacity.

Part B. Using Computers: What Are Your Computer-Handling Skills?**B 1. Interacting with a Computer System**

- B1a. Indicate awareness of general strategies for starting up and exiting from a computer system.

- B1b. Indicate awareness of dealing with common access procedures, such as those involving passwords or user identification codes.

- B1c. Indicate awareness of the general functions of the most common special-purpose keys on the computer keyboard: cursor-movement keys, backspace key, shift keys, function keys, control keys, enter(return) and escape keys.

- B1d. Indicate awareness of how to "find one's way" in a program through interacting with menus.

- B1e. Indicate strategies for handling common peripherals such as printers, modems, or a mouse.

B 2. Disk Handling and Backing Up Data and Software

- B2a. Indicate awareness of the importance of backing up data and of making other sorts of back-up copies of software.

- B2b. Indicate awareness of procedures for backing up data and software.

- B2c. Indicate awareness of strategies for locating files on disks and for doing common file-handling operations relating to copying, deleting, and renaming files.

B 3. Dealing with Common Problems

- B3a. Identify common problems that the computer user typically faces such as:
- problems relative to file incompatibility;
 - operating system incompatibility;
 - problems relating to program operation;
 - problems relating to interfacing with peripherals;
 - problems relating to disk and hardware maintenance.

- B3b. Identify some strategies for dealing with common problems encountered by computer users.

Part C. Applications: What Can You Do With Information Technology?**C1. Common Applications of Information Technology***General categories and examples*

- C1a. Identify some categories of activities for which information technology, and computers in general, are often used and some categories for which they, as yet, have little application.
- C1b. Identify some of the applications of microtechnology in the individual's everyday life.

Categories of commonly used software applications

- C1c. Associate selected information processing tasks with the most appropriate category of commonly used software including
- word processing;
 - desktop publishing
 - data base management;
 - spreadsheet;
 - telecommunications;
 - graphical image generation and presentation;
 - music and other sound generation;
 - data capture and processing in scientific experiments;
 - process control and robotics;
 - mathematical and numerical analysis.
- C1d. Organize data for entry into categories of commonly used software.
- C1e. Interpret the output from various categories of commonly used software.

Features and Functions of Common Applications Software**C2. Word processing**

- C2a-f. Identify and perform the functions of some of the basic features of word processing:
- C2a. creating a file
 - C2b. retrieving a file
 - C2c. entering text
 - C2c. editing text
 - C2d. saving a file
 - C2e. formatting text
 - C2f. printing text or a file

C3. Spreadsheets

- C3a. Identify the meaning of some basic vocabulary relative to spreadsheets: rows and columns, cells, calculation formulas.
- C3b-d. Make basic decisions relative to the use of spreadsheets and perform the operations in each of the following areas:
- C3b. entering and organizing data
 - C3c. determining and entering calculation models for data
 - C3d. displaying and interpreting the results of calculations

C4. Data bases

- C4a. Know and apply concepts of data bases including file, record, field, search, sort, print.

C5. Telecommunications as a Computer Application

- C5a. Be aware of the components necessary for telecommunications applications on a computer: modem, telecommunications software, appropriate connections to other networks or computers; and be aware of and use some of the purposes for which telecommunications is commonly applied:
- electronic mail,
 - accessing of bulletin boards,
 - accessing of on-line data bases and other on-line resources,
 - electronic file transfer.

Part D. Attitudes, Opinions, Social Context, and Ethics

D1. Attitudes

- D1a. Be aware of positive and negative feelings (e.g., enjoyment, fear) about computers and other information technology.
- D1b. Feel informed confident about one's own abilities and inabilities related to computing. Beliefs of self-efficacy are generally warranted in the absence of reliable information to the contrary.

D2. Social Contexts

- D2a. Recognize and believe in the relevance of computers for education, for employment, and for the society in general.
- D2b. Recognize and report parental and other family encouragement toward participation in computing.
- D2c. Recognize gender equality in abilities to participate in computing.

D3. Ethics

- D3a. Recognize ways that computerization contributes to (and detracts from) improving the human condition, that is, the quality of life.
- D3b. Be aware of (and apply) principles of integrity (e.g., professional responsibility) to computer work.
- D3c. Identify different forms of intellectual property pertaining to computers and other information technology. Identify appropriate actions to protect intellectual property, e.g., ownership due to creation of designs and programs, and be aware of penalties for violations of associated rights.
- D3d. Understand that computer crimes are equally serious to parallel noncomputer crimes. Identify appropriate penalties.

D4. Risks and Liabilities

- D4a. Recognize the kinds of risks that can accompany computing applications, e.g., software mistakes, security risks, user mis-applications, and associate actions a system designer can take to reduce these risks.
- D4b. Recognize potential privacy violations in computer contexts.