The process of writing test items is analyzed, and a blueprint is presented for an authoring system for test item writing to reduce invalidity and to structure the process of item writing. The developmental methodology is introduced, and the first steps in the process are reported. A historical review traces the advances made in the field and the formal systems developed for reviewing test items. A Computer Aided Item Construction Project has been initiated to integrate the results of test writing research into an Authoring System for Item Construction (ASIC) program. An existing systems development method was used to guide the process of developing the authoring system. This method, Information Systems Work and Analysis of Changes (ISAC), starts with analyzing the needs, ideas, and problems of those who handle information. Only this first stage, Change Analysis, is described in this report. Steps in change analysis are: (1) problem analysis; (2) description of the existing situation; (3) making a test plan; (4) choosing an item format; (5) creating the test item; (6) checking the items; (7) study of change alternatives; (8) developing a blueprint; (9) choice of the item format; (10) generating items; and (11) checking items. Problems of item writing are described and listed in two tables with reference to whether an authoring system would contribute to their solution and the validity of the test. In all, 4 tables and 12 figures illustrate aspects of item writing. (SLD)
Towards an Authoring System for Item Construction

Jos H.A.N. Rikers

Department of Educational Measurement and Data Analysis, University of Twente
Colofon:
Typing: Mevr. L.A.M. Padberg
Cover design: Audiovisuele Sectie TOLAB Toegepaste Onderwijskunde
Printed by: Centrale Reproductie-afdeling
Towards an Authoring System for Item Construction

J.H.A.N. Rikers
Towards an authoring system for item construction / J.H.A.N. Rikers – Enschede: University of Twente Department of Education. April, 1988. – 43 pages
Abstract

The process of writing test items has many sources of possible invalidity. In this report, the process of item writing as it is used today is analyzed, and some sources of invalidity are described. In order to reduce the invalidity and to structure the process of item writing an authoring system for test item construction is proposed.
Towards an Authoring System for
Test Item Construction

Improvement of the quality of education is a permanent topic. One way to improve quality of education is to improve testing. This report deals with the very first stage of testing, namely the construction of test items.

The first part of this report gives a historical review of new ideas about test item construction. The second part of this report presents a blueprint for an authoring system for item writing. A developmental methodology (ISAC) is introduced and first results are reported.

Historical Review

Improving the quality of test items has always been of great concern to test developers and researchers using tests to collect data. With the emergence of the criterion-referenced testing movement the concern about the quality of test items has become a focus of special interest. The paper of Glaser (1963) marks the beginning of the criterion-referenced testing movement. The interest in criterion-referenced testing coincides with changing demands upon testing. These changes were concerned with the search for assessment methods that could provide information for individual and programmatic decisions arising from specific competencies. Hambleton & Rogers (1986, p.208) put it like this: "Norm-referenced tests were judged to be inappropriate because they provided information that facilitated..."
comparisons among examinees on broad traits or constructs. Norm-referenced tests were not intended to measure specific competencies. And even if items in a norm-referenced test could be matched to competencies, typically there would be too few test items per competency to permit valid criterion-referenced test score interpretations."

The growing interest in criterion-referenced testing as a basis for student selection, policy making and research has changed the demands on requirements of the test item. In the criterion-referenced testing approach the test consists of a representative sample of test items from the universe of test items describing the domain of knowledge or skills to be tested. Therefore, there should exist a direct relationship between the item and the domain. Moreover, it should be clear what cognitive ability is measured by the items in the test. The demands made upon the test items by the criterion-referenced testing approach emphasizes shortcomings of the traditional item writing methods. In the latter a direct relationship between domain and item content is not always present, and it seems to be very difficult to classify items according to the cognitive ability they measure (Baker, 1974; Zwarts, 1982).

Outside the criterion-referenced testing movement other prominent researchers also expressed their concern about the quality of test items. Problems with respect to the quality of test items have been formulated by Bormuth (1970, p.6): "At the present time we seem to be in the position of having to accept the assertion that a test measures whatever the
Item Construction

5
test writers claim it measures without recause to definitive independent evidence." We find that the problem boils down to the fact that there is no existing methodology of test item writing. Writing test items still is considered a skill to be learned by experience. According to Roid, Haladyna, and Finn (1980) methods for writing test items, particularly for criterion-referenced testing, are needed that are (1) based on a logically defined relationship between the instructional materials and the test items and (2) capable of producing items that can easily be replicated by many test developers.

Guttman and Schlesinger (1967) proposed the mapping-sentence method for specifying an item domain. The mapping sentence method is based on facet design theory. A mapping sentence is produced by content analysis; it pays attention to the relationship between the required achievement and the cognitive abilities needed to perform adequately. An example of a mapping sentence is provided by Roid and Haladyna (1982, p. 139) and is shown in Table 1.

Insert Table 1 about here

This mapping sentence is derived from an instructional objective and can be used as a source for a large number of items. The objective is to compute and distinguish correlations between two variables (see Table 1). The mapping sentence consists of fixed and variable sentence parts. A
variable sentence part is called a facet. Items are constructed by filling in the facets provided by content analysis. This means that all items have the same syntactical structure. In this example a multiple choice item is produced, including the foils that are directly related to the results of the content analysis. This means that the test responses, including the foils can be used to measure achievement. Millman (1980) is convinced that the development of mapping sentences and the item generation from these sentences can be structured.

Since a domain is described by a set of mapping sentences and each mapping sentence consists of sets of facets, the description of a domain can easily result in large amounts of documentation, and the use of a computer to store the mapping sentences and the sets of facets would be inevitable. The generation of the items by computer is then a logical next step.

Hively, Maxwell, Rabehl, Sension, and Lundin (1973) used a so-called item-form technique to generate items directly related to objectives and content. Defining the domain with this technique has some resemblance with the mapping sentence method. Like a mapping sentence, an item form is a genuine operationalisation of an objective. Hively et al. (1973) developed a format to display item forms. An example of an item form is shown in Figure 1.
The item form consists of an item form shell (to be compared with fixed sentence parts of a mapping sentence). Figure 1 shows an item form for the concept of comparing cardinality of sets by one-to-one correspondence. The item form provides very detailed information about how the items must be constructed, but at the same time provides a source for numerous different items. Again it is possible to store item forms (written as a computer program) in a computer. (Roid and Haladyna, 1982) leaving the actual item generation to the machine.

The linguistic approach as introduced by Bormuth (1970) is another attempt to structure the item construction process. This technique constructs questions by selecting sentences from instructional text. The selected sentences are transformed into items. The selection of sentences is based on the presence of so-called high-information words. High-information words are words that are relatively rare in American English and have a low text frequency. When sentences are selected, their semantic structure is used to transform them into items. An example of this procedure is presented in Figure 2.
Roid et al (1980) developed an algorithm for generating test items by means of the linguistic technique.

Advances are made in preparing handbooks to aid novice (and experienced) item writers and reviewers, and formal systems have already been developed for the reviewing of test items (e.g. Herman & Winters, 1985). The use of these handbooks contributes to the clarification of the item writing process.

In reviewing the emerging technologies of item writing, Roid and Haladyna (1980) pointed out, that it would be best to consider a continuum of methods from (a) the informal, via (b) the objective-based to (c) the algorithmic method. Depending on the test requirements as well as time and resources available, the item writer should choose the most appropriate method.

But the application of all these different techniques requires a lot of organizational talent. It also takes an enormous effort to reach an adequate level of efficiency and skill in using these (new) methods. For these reasons, the author started a Computer Aided Item Construction (CAIC) project to integrate all these techniques and methods into an Authoring System for Item Construction (ASIC) program. Such an authoring system will contain procedures to provide all information needed to write items. This includes, e.g.,
information printed in handbooks as well as information on constructing item forms. The information will be stored in a database and can be consulted by an item writer in every processing phase. In addition, the system will provide decision rules for every step the item writer takes. In this way, the system assists in declaring mapping sentences, or in defining the replacement sets for an item form.

Two basic questions for the CAIC project are:

1. How to describe the process of item construction as detailed as possible.
2. What are the implications of automating every subprocess.

The purpose of this report is to answer the first question. If we consider item construction as an integrated process of information handling, we best use an information systems development methodology to describe this process. Using an information systems development methodology also enables us to answer the second question, because the methodology will provide detailed descriptions of every subprocess.

Item construction in an authoring systems environment will enable every item writer to use expert knowledge on item construction techniques. This means that an item writer can really use the whole continuum from informal to algorithmic methods.

Using an authoring system creates the possibility to link item construction and computer-aided testing. In the future, this may result in a revolutionary change in the
appearance of items. Traditional items can be replaced by computer simulations or even video games (Jones, 1984; van der Linden, 1985). This seems a promising future for ability testing, especially for testing higher level abilities.

Developing an Authoring System.

An existing systems development method will be used to guide this process. This method is known as the ISAC-method (Lundeberg, Goldkuhl & Nilsson, 1980). ISAC means Information Systems work and Analysis of Changes. The method starts with analyzing the needs, ideas and problems of those who handle information. The method ends with the specification of tasks that must be conducted by hand and computer programs. The method consists of a number of feasible and cohesive steps. Going through every step, a documentation file (describing the system as it is developed) is build. Finally, a fully described system can be developed, and every detail is documented. The reasons why this method is used are: (1) one does not have to be an experienced systems designer, (2) it permits adjustments of the procedures to the subject under study, (3) the method is orientated towards the activities that need support, and (4) the entire process of systems development is cut into manageable steps.

The method provides a step-wise development of an information system. During these steps, documentation is created that will function as the backbone of the system. The documentation consists of graphs or charts reflecting the information flows. The graphs indicate who is responsible for
the generation of information. In this way the system is described in every little detail. The graphs are accompanied by text pages that explain the short descriptions used in the graphs (see e.g. Figure 3). Outside the box on the left hand side in Figure 3 we can see the input at the top and the output at the bottom. The parallelograms contain a short description of the kind of information needed for an activity, or the product of an activity. The parallelograms with a double upper line contain information provided by e.g. the system database and materials supplied by the user. The parallelograms with a single upper line just contain information not provided by the user. The connections between the different activities are represented by lines. Thin lines indicate transport of non-user provided information; bold lines indicate a mixture of user generated and system generated information. Activities are represented by the small circles.

Insert Figure 3 about here

The first stage in the ISAC-method is change analysis. During this stage it is assessed whether the development of an information system is really the best solution to a particular problem. The following stages in the method are developmental stages (activity study; information-analysis; data system development; adjustment of means). This report is
only concerned with the change analysis. Later reports will cover the remaining stages of ISAC.

Change Analysis

Before an information system is developed a change analysis is performed. Change analysis attempts to analyze what kind of changes (in fact, improvements) of activities can be made in real-life situations. The reason for conducting a change analysis is that one is not only interested in tackling the symptoms which are caused by some problems, but also in finding and analyzing the problems themselves.

The results of change analysis can directly be used in the next steps of systems development, and, in fact, can be used as tools to speed up the development. If change analysis leads to the conclusion that the needs and problems of the users will not be solved by the development of an information system, then it would be better to change the strategy.

Change analysis consists of three steps. First, a list of problems is formulated. Next, the existing situation is described, and finally, a description of the resulting information system is given.

Problem Analysis

Starting a change analysis means starting a detailed description of the existing problems. These problems can be listed in a Problem list.
The problems listed in Table 2 can be grouped into four categories which are given in Table 3.
Making a Test Plan.

In today's item-writing practice, writing itself is considered to be part of the test development. This means that the need for a concrete test is the starting point. This differs from computerized testing where the generation of an itembank can be seen as a separate activity (van der Linden, 1986). Information on (1) the content area of the test, (2) the cognitive objectives, and (3) the test requirements are gathered in a test plan (see Figure 4).

The formation of a test plan is supported by guidelines. These guidelines lead to a two-way table where an outline of the domain is placed on one axis of the table and the cognitive objectives along the other (Mehrens & Lehmann, 1973). In this way it is possible to classify item types in the cells of the table simultaneously by domain and objective. The next thing to do is deciding on how many items have to be written for every item type in the table.

This procedure seems to be a valid one, since only the number of items and the classes have to be determined. Problems arise, however, when the outline of the content area and the detailed description of the objectives have to be obtained. It is well known that it is not easy to extract information from educational objectives, course descriptions.
information from educational objectives, course descriptions, id. No rules for this kind of activity are available. This means that every individual item writer has to rely on his/her own methods and experience. This, in turn, means that different item writers may have different results at the end of the procedure and thus come up with different plans for the same test. There are at least two possible kinds of test plans. One is a comprehensive review of the curriculum, which includes all the topics and behaviours connected to the course. The other is a selection of the materials based on the most important topics in terms of learning possibilities and acquisition of desirable behaviours. Since the latter kind of test plan may cause large discrepancies between item writers, we prefer the 'comprehensive' type of test plan, a choice also in line with the principles of criterion-referenced testing. The problems grouped in G2 and G3 (and partly in G1) arise (see Table 2 and 3) at this stage.

Choosing an Item Format.

The next decision to be made is the one on the format of the items in the test. This decision can be made after consulting the test plan and the information on different item formats, as is shown in Figure 5.

Insert Figure 5 about here
This choice is also a possible source of disagreement among item writers. See the problems mentioned in G2 (of Table 2 and 3).

Creating the Test Item.

Up till now the actual creation of the test item was seen as the exclusive domain of the test writer. The ideas for the item text had to come out of his/her creative mind, and there were no rules available to guide this process. The writing process (see Figure 6) can be based on (a) the information from the testplan, (b) the information from the chosen item format, (c) the information on how to write items (e.g., old items), and (d) the information from items created in another setting.

Insert Figure 6 about here

After the item text is created, graphics can be added if necessary. Before the item is filed, the lay-out must be checked and instructions can be added.

The freedom of the item writer in this phase can result in a serious bias comparable with the problems clustered in G2 and G4 (see Table 2 and 3). To improve the construct validity of test items, strict rules for this stage of the writing process are needed.
Checking the Items.

Before the items can be used in a test, a final check has to be made. Figure 7 shows what activities are needed for this check. First of all, the fulfilment of the requirements collected in the testplan has to be checked, and possible deviations from the testplan can be corrected. Next, the items have to be checked with the help of available checklists, and finally a check on spelling and grammatical errors is made. If the lay-out is considered to be correct, then the item can be stored. Any rejected item is placed in a pool and can be used again as input for a following writing stage.

Insert Figure 7 about here

In this stage a number of problems may be encountered. The proposed checking procedure is often not precise enough to detect every error in the items. In fact, all the problems mentioned in G2; G3 and G4 (see Table 2 and 3) should be taken care of in this stage, but usually this is not done.

Study of Change Alternatives

The last stage of change analysis consists of a description of the future system. The description enables us to compare the desired situation with the existing situation. The
description of the system starts with a listing of a number of goals of the system (see Table 4); Activity Graphs and Text Pages are used for this description.

A great advantage of an authoring system for item writing is that a structure can be imposed on the procedure. Every step (as shown in Figure 8) can be guided by rules, and some parts of the process can even be handled by applying heuristics and algorithms.

In principle the steps for developing test items do not differ much from the steps described earlier. Only the activities in every step are different.

**Developing a Blueprint.**

The development of a blueprint does not differ to much from the development of a test plan (compare Figure 4 with Figure 9). The differences are mainly a consequence of the fact that the items are no longer written for one specific test. This means that test requirements are no longer part of
the constraints for the blueprint. These requirements can be fulfilled during the testconstruction.

The development of the blueprint will lead to a two way table with content information as the row headings and cognitive objectives as column headings. The construction of this table is guided by rules (heuristics and/or algorithms), that must ensure the replicability of the table. This means that the freedom of the item writer must be limited in favor of the replicability of the results. An addition to the information in the table is the information on the difficulty level of the text material used in the course (also called readability level). The difficulty level is an index that can help the item writer to write items matching the index. Furthermore, the index is valuable when deciding what cognitive abilities should be tested. The cells of the table will be filled with information describing the items to construct as detailed as possible.

Choice of the Item Format.

In Figure 10 we see that for every cell of the table of specification an item format must be chosen. This choice must depend on the information in the cell. The test requirements for one specific test are no longer part of the decision.
Instead, a whole system of decision rules is added to aid the decision on the item format. The choice of the item format for a cell of the table of specification thus will be a compromise between the choice according to the item specifications in the cell, and the choice as a result of the use of decision rules. If the ultimate decision is left to the item writer, it is now clear on what grounds it was made.

---

**Insert Figure 10 about here**

---

**Generating Items.**

The table of specification and the item format connected with every cell in the table, are the input for the decision on what strategy should be applied to generate items (see Figure 11), and for each chosen strategy there will be a set of rules and additional information to structure the process.

---

**Insert Figure 11 about here**

---

The freedom of the item writer is restricted in favor of clarity and control of the process, and the severity of these restrictions will differ for each strategy. The
strategies themselves vary along the continuum proposed by Roid & Haladyna (1980). For every strategy, detailed rules should be worked out.

A very practical addition is the generation of graphics that could be integrated in the item writing process. This will save time, and results are immediately available. In the long run, graphics could become a very important part of the system. We predict a wide range of new possibilities in item generation processes when in computer aided testing graphics are combined with interactive video.

Checking Items.

The checking of items as shown in Figure 12 can be supported by such information as checklists; spell checkers; and, in the future, word processors using artificial intelligence to check on grammar: id.

Insert Figure 12 about here

Conclusions

To decide whether the problems described in Table 2 (and grouped in Table 3) can be solved best by developing an described authoring system, we must also consider alternative solutions.
G1 Item Construction is Time Consuming

Time can be reduced by integrating text and graphics generation (see Figure 11). When a word processor and other computer programs are used to check the items, the time needed to develop an item will be shortened. The time needed to search for information on various subjects (e.g. item formats or generation rules for multiple-choice items) will also be reduced by storing all information in a database.

It will cost time to prepare a test plan, as it will cost time to prepare item generation according to a chosen strategy (see Figure 11). But once time has been invested, it will be relatively easy to develop more items of the same quality. And once an item set has been developed, it can be used many times. The question whether this automatically implies that item generation should be automated cannot be answered yet. The suggested item writing techniques (Roid & Haladyna, 1980) can all be used without a computer. There is still work needed to show that the use of a computer is possible and will speed up the process.

The conclusion for problem group one therefore is that there can be a reduction in time needed to generate items.

G2 Itemwriter Bias Reduces the Validity of Items

This group of problems has been treated very thoroughly in the developed system. The process will be guided by rules to guard against item writer bias. The construction of the table of specification and the item generation itself are
guided by heuristics and algorithms to reduce differences among item writers, and differences in experience will be reduced by the information available during the whole process.

For some evidence that the item writing methods reduce item writer bias see Roid and Finn (1978; 1980).

G3 Content and Construct Validity Often Questionable

This problem is solved by guiding the building of the table of specification (see Figure 9). By making a comprehensive blueprint and checking the difficulty level of the instructional materials we are quite certain that the overall content validity of the items improves.

The construct validity of the items is a somewhat different matter. The relationship between students performances on an item and the ability the item is supposed to measure is not easy to establish. Up till now, the best help available is a set of taxonomies to classify items according to the ability they measure. The problem is that these taxonomies can only be used with existing items. This means that if we want to use them while generating items, we have to compromise. We could, for instance, use classified sample items to start from. Having generated some items, we could then check the ability they measure by classifying them again.
G4 Items Often Show Editorial Mistakes

Checking items using computer programs may reduce the number of mistakes and save time. With the use of artificial intelligence in combination with a word processor, it should be possible to eliminate most editorial mistakes.

G5 Item Writing as an Act of Creation

Subjecting the item writing process to rules, it would appear that there is no more room left for the creative item writer. This is not true. The rules allow control of the generation of items. But there still is a need for creativity when the item itself has to be made. Therefore, it would be better to conclude that there still is a lot of creativity involved in generating items, but that the generation process has a more scientific basis if the results can be replicated.
References


Table 1. Example of a mapping sentence developed from an instructional objective (Roid & Haladyna, 1982).

<table>
<thead>
<tr>
<th>Objectives</th>
<th>Mapping Sentence</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Given a set of ordered pairs of values on variable X and Y, correctly calculate r_{xy}.</td>
<td>A Presentation form</td>
</tr>
<tr>
<td>2. Given statements about the Pearson r, point biserial, phi, and rank order (correl.) correlation coefficients, the learner will identify those that accurately compare and contrast the various measures.</td>
<td>B Content form</td>
</tr>
</tbody>
</table>

A Presentation form

<table>
<thead>
<tr>
<th>Given 4</th>
<th>1 Table</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1 A set of ordered pairs of values on variables with 3 pairs</td>
<td></td>
</tr>
<tr>
<td>1.2</td>
<td>2 Pairs</td>
</tr>
<tr>
<td>1.3</td>
<td>5 Pairs</td>
</tr>
</tbody>
</table>

B Content form

<table>
<thead>
<tr>
<th>A Variable X</th>
<th>1.one digit</th>
</tr>
</thead>
<tbody>
<tr>
<td>2. two digits</td>
<td>1. one digit</td>
</tr>
<tr>
<td>3. three digits</td>
<td>2. two digits</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>E Variable Y</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. one digit</td>
</tr>
<tr>
<td>2. two digits</td>
</tr>
<tr>
<td>3. three digits</td>
</tr>
</tbody>
</table>

C A set of ordered pairs of values on variables with 3 pairs

D Given 3 |

E Variable Y

<table>
<thead>
<tr>
<th>Type of score</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. deviation</td>
</tr>
<tr>
<td>2. raw</td>
</tr>
</tbody>
</table>

F Type of score |

<table>
<thead>
<tr>
<th>Multiplication of signed numbers</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. correct</td>
</tr>
<tr>
<td>2. incorrect</td>
</tr>
</tbody>
</table>

G Division of SP (x,y) |

| 1. no |
| 2. yes that is SP (x,y) |

H Square root of SP (x,y) |

| 1. yes |
| 2. no |

I Type of unit |

| 1. no unit |
| 2. linear unit |
| 3. square unit |
| P1 | item construction is time consuming       |
| P2 | quality of items differs because of itemwriter bias |
| P3 | expertise and experience of itemwriters differs substantially |
| P4 | items are not always linked with curriculum. |
| P5 | items are not always linked with objectives |
| P6 | items don't frequently satisfy layout rules |
| P7 | items don't frequently satisfy construction rules |
| P8 | items often contain grammatical mistakes |
| P9 | items often contain spelling mistakes |
| P10 | items often contain deficiencies of style |
| P11 | items containing graphics cannot be constructed in one construction step |
| P12 | itemwriting is considered too much an act of creation and lacks a systematic approach |
Table 3. Problems grouped in clusters.

<table>
<thead>
<tr>
<th>G1</th>
<th>P1;P11</th>
</tr>
</thead>
<tbody>
<tr>
<td>G2</td>
<td>P2;P3</td>
</tr>
<tr>
<td>G3</td>
<td>P4;P5</td>
</tr>
<tr>
<td>G4</td>
<td>P6;P7;P8;P9;P10;</td>
</tr>
<tr>
<td>G5</td>
<td>P12</td>
</tr>
</tbody>
</table>
Table 4. Goals of the authoring system.

| G1 | the content of the items must have a direct and controlable relationship to the learning material |
| G2 | doubts about the cognitive ability an item measures should be minimized. |
| G3 | it should be possible that different item writers produce items not differing in quality |
| G4 | the item writer should have on-the-spot access to guidelines for item writing |
| G5 | the item writer should have on-the-spot access to means and tools to check the quality of items |
| G6 | integration of graphics and the item text should be made very easy. |
| G7 | the system should have an open structure so that new modules can be attached at any time |
Comparing numerosity of sets by one-to-one correspondence

The child is given either two or three sets of "counters," each having approximately 20 members (or less). The sets may have the same number of members or they may differ by one member. The child is asked to show whether or not the sets have the same number of members without counting.

STIMULUS AND RESPONSE CHARACTERISTICS

Constant for all Cells

Only standard "counters" (small colored disks) are used. Each set of counters is a different color (red, green, or yellow).

Number of Sets Compared: Two or three

Whether or not the sets have the same number of objects

Approximate number of objects in each set (about 5, about 13, or about 21)

Varying with Cells

No variation

<table>
<thead>
<tr>
<th>CELL MATRIX</th>
<th>NUMBER OF SETS IN SETS</th>
<th>NUMBER OF OBJECTS COMPARED</th>
<th>EQUALITY RELATIONS</th>
<th>EQUALITY RELATIONS</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>5</td>
<td>(1) (4)</td>
<td>(7) (10)</td>
<td>(13)</td>
</tr>
<tr>
<td></td>
<td>13</td>
<td>(2) (5)</td>
<td>(8) (11)</td>
<td>(14)</td>
</tr>
<tr>
<td></td>
<td>21</td>
<td>(3) (6)</td>
<td>(9) (12)</td>
<td>(15)</td>
</tr>
</tbody>
</table>

* Originally developed by Bruce Russell

Figure 1. Example of an item form (Hively et al., 1973).
EXAMPLE OF ITEMS PRODUCED FROM TEXT

1. **Keyword Noun — Metamorphosis.**
   a. **Text Sentence(s):** After hatching, all insects, except the most primitive, go through a series of steps in development. These steps are called metamorphosis.

   b. **Items (Stem and Foils) Produced by Item Writers:**

      (1) What are the series of steps in insect development called?
          (a) Maturation
          (b) Metamorphosis
          (c) Symbiosis
          (d) Meiosis

      (2) What are the steps insects go through in development called?
          (a) Metamorphosis
          (b) Arthropoda
          (c) Larva
          (d) Pupa

      (3) What are a series of steps in development called?
          (a) Reproduction
          (b) Larvac
          (c) Metamorphosis
          (d) Changes

      (4) What are the series of steps in insect development called?
          (a) Enyrtid
          (b) Instar
          (c) Arthorpoda
          (d) Metamorphosis

   c. **Foils Produced Algorithmically.**

      **Growths**
      Metamorphosis
      Types
      Activities

   **Figure 2.** Example of foils produced by the algorithm of Roid et al. (1980).
Figure 3. A GRAPH and TEXTPAGE showing an overview of the current state of the art of item writing.
Figure 4. A GRAPH and TEXTPAGE showing the construction of a blueprint.
Figure 5. A GRAPH and TEXTPAGE showing the choice of the item format
Figure 6. A GRAPH and TEXTPAGE showing the construction of items.
Figure 7. A GRAPH and TEXTPAGE showing the checking of the items.

<table>
<thead>
<tr>
<th>Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>4A4</td>
<td>information on checking the quality of items</td>
</tr>
<tr>
<td>7A</td>
<td>provisional items</td>
</tr>
<tr>
<td>5A3</td>
<td>matching the blueprint and the item set</td>
</tr>
<tr>
<td>C81</td>
<td>check if the item set matches the blueprint</td>
</tr>
<tr>
<td>C82</td>
<td>use checklist to ensure correctness of items with regard to chosen item format</td>
</tr>
<tr>
<td>C82A</td>
<td>item set and comments on construction errors</td>
</tr>
<tr>
<td>C83</td>
<td>check the items on spelling and grammatical errors</td>
</tr>
<tr>
<td>C83A</td>
<td>item set and comments from C82 and C83</td>
</tr>
<tr>
<td>C84</td>
<td>check the items on lay-out errors</td>
</tr>
<tr>
<td>8A</td>
<td>rejected/flaw items and comment</td>
</tr>
<tr>
<td>8B</td>
<td>items ready for use</td>
</tr>
</tbody>
</table>
Figure 8. A GRAPH and TEXTPAGE showing the wanted situation
Figure 9. A GRAPH and TEXTPAGE showing the wanted construction of the table of specification
Figure 10. A GRAPH and TEXTPAGE showing the procedure to chose the item format
Figure 11. A GRAPH and TEXTPAGE showing the construction of items in the new situation.
Figure 12. A GRAPH and TEXTPAGF showing the checking of items in the new situation.
Titles of recent Research Reports from the Division of
Educational Measurement and Data Analysis.
University of Twente, Enschede,
The Netherlands.

RR-87-1  R. Engelen, Semiparametric estimation in the Rasch model
RR-87-2  W.J. van der Linden (Ed.), IRT-based test construction
RR-87-3  R. Engelen, P. Thommassen, & W. Vervaat, Ignatov's theorem: A new and short proof
RR-87-4  E. van der Burg, & J. de Leeuw, Use of the multinomial jackknife and bootstrap in generalized nonlinear canonical correlation analysis
RR-87-5  H. Kelderman, Estimating a quasi-loglinear models for the Rasch table if the number of items is large
RR-87-6  R. Engelen, A review of different estimation procedures in the Rasch model
RR-87-7  D.L. Knol & J.M.F. ten Berge, Least-squares approximation of an improper by a proper correlation matrix using a semi-infinite convex program
RR-87-8  E. van der Burg & J. de Leeuw, Nonlinear canonical correlation analysis with k sets of variables
RR-87-9  W.J. van der Linden, Applications of decision theory to test-based decision making
RR-87-10 W.J. van der Linden & E. Boekkooi-Timminga, A maximin model for test design with practical constraints
RR-88-1  E. van der Burg & J. de Leeuw, Nonlinear redundancy analysis
RR-88-2  W.J. van der Linden & J.J. Adema, Algorithmic test design using classical item parameters
RR-88-3  E. Boekkooi-Timminga, A cluster-based method for test construction
RR-88-4  J.J. Adema, A note on solving large-scale zero-one programming problems
RR-88-5  W.J. van der Linden, Optimizing incomplete sample designs for item response model parameters
Rk-88-6  H.J. Vos, The use of decision theory in the Minnesota Adaptive Instructional System
RR-88-7  J.H.A.N. Rikers, Towards an authoring system for item construction

Research Reports can be obtained at cost from Bibliotheek, Department of Education, University of Twente, P.O. Box 217, 7500 AE Enschede, The Netherlands.