AL/HR-TR-1995-0159

STEP WRITER FINAL REPORT: DEVELOPMENT AND DELIVERY TOOLS FOR PROCEDURAL TRAINING

Jeffrey E. Norton
Daniel F. Thompson

Galaxy Scientific Corporation, Information Division
2310 Parklake Drive, Suite 325
Atlanta, GA 30345

HUMAN RESOURCES DIRECTORATE
TECHNICAL TRAINING RESEARCH DIVISION
7909 Lindbergh Drive
Brooks AFB, Texas 78235-5352

November 1995

Final Technical Report for Period July 1991-October 1994

ARMSTRONG LABORATORY

AIR FORCE MATERIEL COMMAND
BROOKS AIR FORCE BASE, TEXAS

NOTICES

Publication of this report does not constitute approval or disapproval of the ideas or findings. It is published in the interest of STINFO exchange.

When Government drawings, specifications, or other data are used for any purpose other than in connection with a definitely Government-related procurement, the United States Government incurs no responsibility or any obligation whatsoever. The fact that the Government may have formulated or in any way supplied the said drawings, specifications, or other data, is not to be regarded by implication, or otherwise in any manner construed, as licensing the holder, or any other person or corporation; or as conveying any rights or permission to manufacture, use, or sell any patented invention that may in any way be related thereto.

The Office of Public Affairs has reviewed this report, and it is releasable to the National Technical Information Service, where it will be available to the general public, including foreign nationals.

This report has been reviewed and is approved for publication.

JIMMY L. FLEMING
Contract Monitor
Intelligent Training Branch

R. BRUCE GOULD
Technical Director
Technical Training Research Division

JAMES B. BUSHMAN, Lt Col, USAF
Chief, Technical Training Research Division
Human Resources Directorate

Please notify this office, AL/HRPP, 7909 Lindbergh Drive, Brooks AFB TX 78235-5352, if your address changes, or if you no longer want to receive our techinical reports. You may write or call the STINFO office at DSN 240-3853 or commercial (210) 536-3853.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY (Leave blank)	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	November 1995	Final Report - July 1991- October 1994

4. TITLE AND SUBTITLE

Step Writer Final Report: Development and Delivery Tools for Procedural Training

5. FUNDING NUMBERS

C - F41622-92-D-0006
PE - 63227F
PR - 2949
TA - 01
WU - 05

6. AUTHOR(S)

Jeffrey E. Norton
Daniel F. Thompson

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Galaxy Scientific Corporation, Information Division
2310 Parklake Drive, Suite 325
Atlanta, GA 30345

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

Armstrong Labortory
Human Resources Directorate
Technical Training Research Division
7909 Lindbergh Drive
Brooks Air Force Base TX 78235-5352

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

AL/HR-TR-1995-0159

11. SUPPLEMENTARY NOTES

Armstrong Laboratory Technical Monitor: Jimmy L. Fleming, (210) 536-2034

12a. DISTRIBUTION/AVAILABILITY STATEMENT

Approved for public release; distribution is unlimited

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*

For several years artificial intelligence techniques have been applied to computer-based training systems. The products of this merger are referred to as intelligent tutoring systems (ITSs). Typically, ITSs are instructionally effective but costly to develop and maintain. Reducing the resources necessary to build and maintain ITSs is the objective of the Intelligent Computer-Assisted Training Testbed (ICATT) project. The approach taken in ICATT to achieve this is to build authoring shells which instructional designers and subject matter experts can use to produce ITSs. This significantly reduces or eliminates the need for computer programmers and knowledge engineers in the process. This paper describes one such shell called Step Writer and its application to the procedural tasks of configuring the ARC-164 Radio for testing and assembling/disassembling the M-16 Rifle.

14. SUBJECT TERMS

Artificial Intelligence
Authoring Shells
Intelligent Tutoring Systems
Technical Training

15. NUMBER OF PAGES
26

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	UL

CONTENTS

Figures

ACKNOWLEDGMENTS

Step Writer was developed for the U. S. Air Force's Armstrong Laboratory (AL/HRTI). Jimmy L. Fleming served as contract monitor for the project. Dr. J. Wesley Regian, Maj H. A. Goodman, Chuck Swanberg, Capt Leonard Mackie, and Capt Roger Riendeau supported the development of Step Writer.

MSgt Gordon L. Morrison and TSgt Mark R. Collins of the 332 Technical Training Squadron at Keesler AFB, MS provided technical assistance for the ARC-164 Radio. MSgt Edward Scott of 343 TRS/TCAT and his associates provided technical assistance for the M-16 Rifle.

I. INTRODUCTION

The Air Force has been investigating the use of computer-based training (CBT) and intelligent tutoring systems (ITSs) to help meet its training needs (from fundamental skills training to troubleshooting training). The authors developed Microcomputer Intelligence for Technical Training (MITT) and MITT Writer in the late 1980's to meet the Air Force's troubleshooting needs. MITT and MITT Writer allow non-programmers to develop and deliver low-cost ITSs for troubleshooting (Norton, et al., 1991; Wiederholt, et al., 1992). However, troubleshooting is but one part of the Air Force's training needs.

An equally great need exists for effective training on non-troubleshooting procedures. Therefore, the Air Force pursued the creation of instructional development and delivery tools to create, present, and maintain tutors for procedural tasks. This paper describes the end product - Step Writer. Step Writer is an integrated authoring and delivery system to help instructors and other non-programmers develop computer-based training systems for procedural tasks.

II. WHAT IS A PROCEDURAL TUTOR?

A Procedural Tutor is a tutor which teaches background and support information and permits adaptive practice of procedures (to include monitoring and feedback). A Procedural Tutor should also be able to support a wide variety of procedures: from assembly/disassembly procedures to checklists and control panel operations.

The authors started with this primitive definition and began to refine it by studying different Air Force domains. Armstrong Laboratory identified several candidate domains for investigation. As each domain was studied, different properties emerged for each domain. The remainder of this section describes these domains and the Step Writer design that emerged.

Domains Investigated

Armstrong Laboratory identified several candidate domains for investigation. These systems were the testing configuration for the ARC-164 Radio and assembly/disassembly of the M-16 Rifle at Lackland AFB, Texas. Each of these domains presented their own properties to be considered in support of procedural tasks.

ARC-164 Radio

The ARC-164 Radio was a good candidate domain for study for several reasons. First, the ARC-164 contained an assembly procedure - the student had to know how to configure the system for initial operation from a schematic diagram (Schematic Assembly), similar to **Figure 1**. The ability to represent assembly / disassembly procedures has wide applicability across the Air Force. In fact, many Technical Orders specifically instruct the student to assemble the equipment according to an attached schematic.

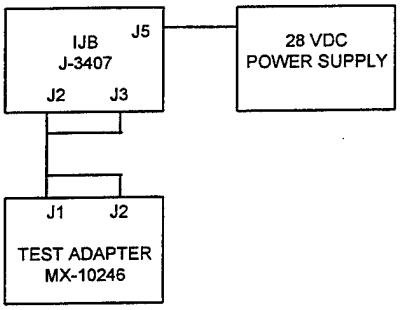


Figure 1. *Schematic Assembly*

The ARC-164 Radio was also a good candidate for Step Writer because it had well-established procedures for the operation of the radio. This allowed the authors to investigate another type of procedure: Control Panel Operations. Step Writer focused on one particular procedure of the Radio, the AJ Mode Test. The AJ Mode Test is a very time-consuming, tedious procedure with very few opportunities within it for feedback that could verify correct execution of the procedure. However, a computer-based, procedural tutor could intervene to tell the students when they made a mistake.

The ARC-164 Radio Tutor was a prototype that helped to define the core components required to define procedures. While this prototype did not allow for any authoring of the procedure, it helped to define the general properties that would later be needed in the authoring system. Some of these general properties are discussed in the Step Writer design in **Sec. II Resulting Step Writer Design**.

M-16 Rifle

The M-16 Rifle was also an ideal candidate domain for Step Writer because it represented a piece of equipment that large numbers of incoming Air Force recruits learned to assemble and disassemble. Even though the M-16 also involves assembly/disassembly, it represents a different class of assembly procedure - Composition Assembly. Composition Assembly is defined in further detail in **Sec. II** below.

In addition, the M-16 Rifle domain provided a testbed for the authoring system. As new features were developed for the authoring system and the student run-time system, the authors used portions of the M-16 domain to test Step Writer.

Resulting Step Writer Design

The ARC-164 and the M-16 Rifle domains demonstrated many of the attributes that a procedural system must have. This section describes the design that resulted from investigation of these domains in three separate areas:

- Defining Procedures
- Defining Background and Support Information
- Defining the Pedagogy

Defining a Procedure

Step Writer supports two different types of procedures: Schematic Assembly and Composition Assembly. "Schematic Assembly" involves a representation of the system that is similar to a schematic diagram, as shown earlier in **Figure 1**. The student assembles a system by selecting components and telling the Step Writer to connect them (by performing the appropriate action). When two components are "connected", a line is drawn between the two components to indicate

the connection. As the assembly continues the screen should look more like the completed schematic diagram.

"Composition Assembly" involves a representation of the system that shows pictures of the equipment in various states of assembly. The student selects Components and then chooses Actions to perform on the selected component. After the Action has been completed, the system shows the components in a more complete state of assembly or disassembly. **Figure 2** shows the M-16 after pressing the magazine release button to release the magazine from the lower receiver group.

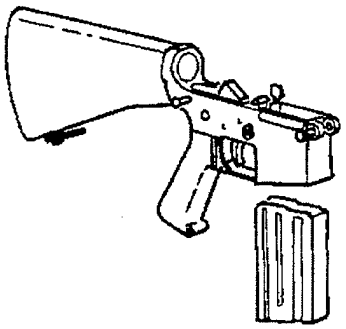


Figure 2. *Composition Assembly*

Whether using a Schematic Assembly procedure or a Composition Assembly procedure, Step Writer defines a procedure as an ordered collection of steps. A step in Step Writer is defined as an action performed on component(s) with certain preconditions (where preconditions are other steps which need to be performed). For example, one step might be to "Open the Door". Within Step Writer, "Open" would be the action and "Door" would be the component. Once a step is successfully performed, Step Writer shows the results of correct performance of the step.

Actions

An action is a word or phrase that the author deems appropriate. For example, an action can be as general as "Tighten" or as specific as "Tighten to 20 ft. lbs." The author decides how general or specific the action needs to be.

Components/Views

A component is principally defined as a collection of pictures. Each picture represents a different view or perspective of the component. A view may contain hotspots, where hotspots define areas on a component of particular importance. Each component can have up to seven different views.

Preconditions

Each step may have preconditions. A precondition is another step or steps that must be performed before the current step can be performed. For example, a precondition to "Open the Door" may be a step to "Unlock the Door".

Results of Correct Performance

When students correctly perform a step they receive feedback. What happens depends on the type of procedure being performed. In Schematic Assembly, the components in the schematic are connected by a line to indicate success. In Composition Assembly, the feedback is more natural - the currently selected components disappear and are replaced with new components to reflect a change in state. Sometimes the component change is subtle, while other component changes are quite dramatic. Using the "Open the Door" example, the component picture of a closed door would be replaced by a picture of an open door.

Defining Background and Support Information

Previous research identified essential categories of information for assembly tasks (Bieger & Glock, 1984). Step Writer provides different kinds of background and support information for the procedures using the following modules: Component Identification, Component Function, and Procedure Information. These modules support each of the four information categories shown in **Table 1**.

Category	Description
Inventory	names and pictures of objects
Operational	action to be performed on objects
Spatial	location or position of object orientation of object composition of object
Contextual	general outcome or goal, such as picture or description of the final product

Table 1. *Categories of Information to Support Assembly Tasks*

The Component Identification and Component Function modules allow the student to learn about the components involved in the procedure and how they function. As implemented in Step Writer, these modules support the Inventory and Spatial information categories in **Table 1**. The idea is that the students must be able to correctly identify a component and know how it functions before they are able to effectively perform the procedure.

Step Writer's Procedure Information module (as well as the Procedures themselves) accommodate the Operational, Spatial, and Contextual categories of information. The Procedure Information module allows the student to optionally obtain detail about the procedure beyond that provided by Component Identification and Component Function.

All three background and support information modules provide the ability to display text, graphics, and multimedia. When Step Writer authors provide multimedia, they must also provide orienting information for the student. This will let the novice know what to attend to, allowing the student to interpret the information in the multimedia correctly (Rieber, 1990).

Defining the Pedagogy

The previous sections briefly defined the different components that make up a Step Writer tutor. This section describes the pedagogy that allows this procedural information to be presented to the student in a meaningful way. These pedagogical aspects include the following: remediation, timing of feedback, chunking, mental models, repetition, and mnemonics.

Remediation and Timing of Feedback

Effective types of feedback vary from situation to situation (Kulik & Kulik, 1988). In applied studies, immediate feedback was better than delayed feedback. However, in experimental studies, results were mixed. Overall, Kulik & Kulik felt that immediate feedback was better than delayed feedback for testing situations. However, McKendree, (1990) indicates that feedback should take place immediately and be placed in the proper context based on Anderson's ACT* theory.

Cohen found that immediate feedback is best for learners with a low level of mastery (Cohen, 1985). The use of immediate feedback facilitates short-term memory and initial acquisition for the novice. End-of-session feedback seems to be more appropriate for learners with a high level of mastery.

Based on these results, Step Writer uses an adaptive feedback (immediate vs. delayed) strategy. When performing a procedure, Step Writer provides immediate feedback at the beginning levels of the procedure, but then transitions to end-of-session feedback at the highest level. **Sec. III Student-Run Module**, provides specific examples of how this is implemented for the Procedures in Step Writer.

Chunking and Mental Models

Step Writer employs chunking to help the student manage all of the information required to perform a procedure. Miller's research argues that the capacity of short-term memory is from five to nine items, with seven items being the average (Miller, 1956). Therefore, Step enforces chunking of component and procedure information to no more than nine elements per chunk.

Step Writer also attempts to help the student derive a mental model of the procedure being learned. Kieras states that students who have an adequate mental model retained procedures much better than students who do not have adequate mental models (Kieras, 198x).

Step Writer uses a hierarchical decomposition of the components and procedures to provide its own type of mental model, as shown in **Figure 3**. This diagram helps to reinforce the chunking and force the student to think about the components and the procedures in logical increments.

The dotted lines in **Figure 3** outline the different chunks of the procedures. A different diagram is generated for components.

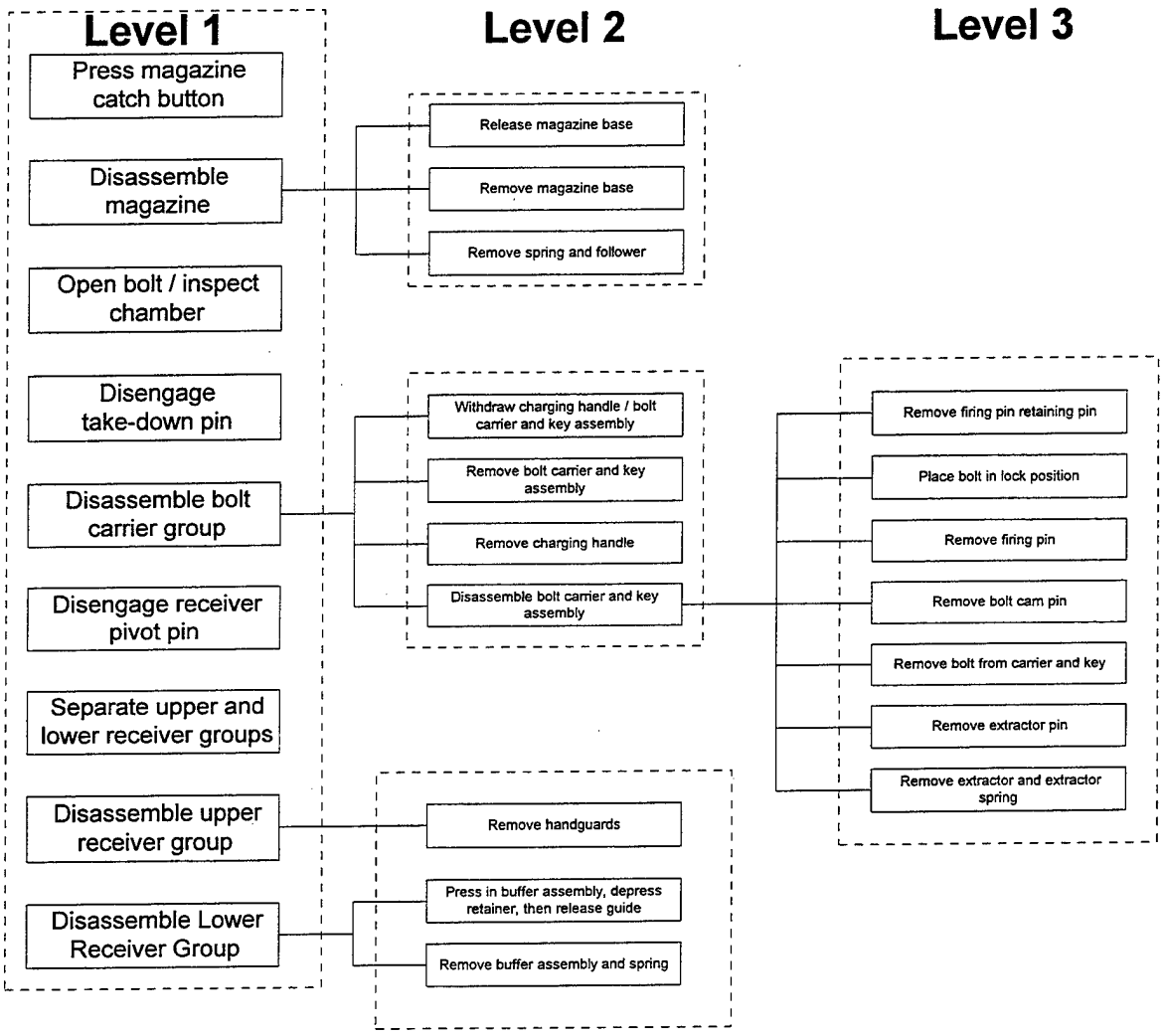


Figure 3. *Hierarchical Mental Model*

Repetition

Step Writer provides repetition to help reinforce the material that students learn. It does this several ways. First, the student is shown the information in a Guided Tour. The Guided Tour tells the student what to do and how to do it. After navigating the Guided Tour, the student graduates to the Demonstrate Knowledge level. This level forces the students to demonstrate

their mastery of the material just learned. If they do not exceed an author-defined performance threshold for Demonstrate Knowledge, they must repeat the Guided Tour. the exact implementation of the Guided Tour/Demonstrate Knowledge cycle varies in Component ID/Function and the Procedures. **See Sec. III Student Run-Time Module**, for more details.

After graduating from Demonstrate Knowledge for Procedures, Step Writer presents the Final Exercise. The Final Exercise is a chance for the student to demonstrate mastery of the entire procedure, not just chunks. The Final Exercise forces the students to review material if they do not exceed author-defined performance thresholds.

Mnemonics

Step Writer also supports mnemonics for procedures. The author provides text to present and explain a mnemonic that he feels is appropriate for the current procedure.

Order of Presentation

Step Writer presents the information shown in **Figure 3** in a breadth-first manner. It presents all information within a chunk (shown by dotted lines) before proceeding to the next chunk. For example (in Level 2 of **Figure 3**), Step Writer presents the steps which comprise the Disassemble Magazine procedure (Release Magazine Base, Remove Magazine Base, Remove Spring and Follower) before proceeding to the steps of the Disassemble Bolt Carrier Group procedure. Step Writer employs the same approach for the Components.

III. THE SOFTWARE

In an effort to support the capabilities described above, Step Writer includes a Student Run-Time Module and the Authoring Module. The Student Run-Time Module is what the student sees when running a Step tutor, while the Authoring Module is where the authors (instructors, subject matter experts, etc.) enter the domain-specific information for a Step tutor. This section describes each of these modules in further detail.

Student Run-Time Module

The Student Run-Time Module consists of four general sections, presented in the following order: Component Identification, Component Function, the Procedures, and the Final Exercise, as shown in **Figure 4**. The idea is that students must learn how to identify a component and know how it functions before they will be able to effectively interact with that component in a procedure. After the student has progressed through the different stages of the procedure, a final exercise will test that the student can perform the entire procedure. The Final Exercise is a chance for the student to demonstrate mastery of the entire procedure, not just mastery of the individual chunks.

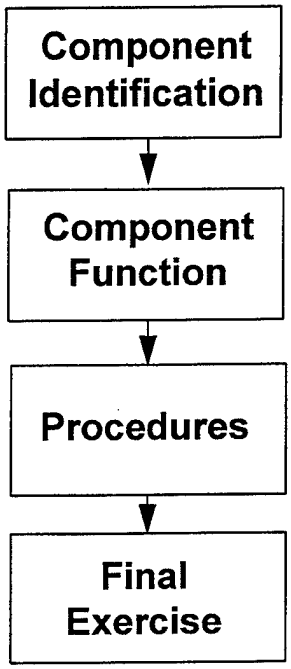


Figure 4. *Student Run-Time Modules*

Component ID / Function

Figure 5 shows an example of the Component Identification section for the M-16 Rifle. The student first sees a Guided Tour of all items at Level 1 (Magazine Group, Bolt Carrier Group, Upper Receiver Group, and Lower Receiver Group). At the beginning of the Guided Tour, the system presents a diagram (similar to **Figure 5**) to serve as a hierarchical representation of the components in the procedure.

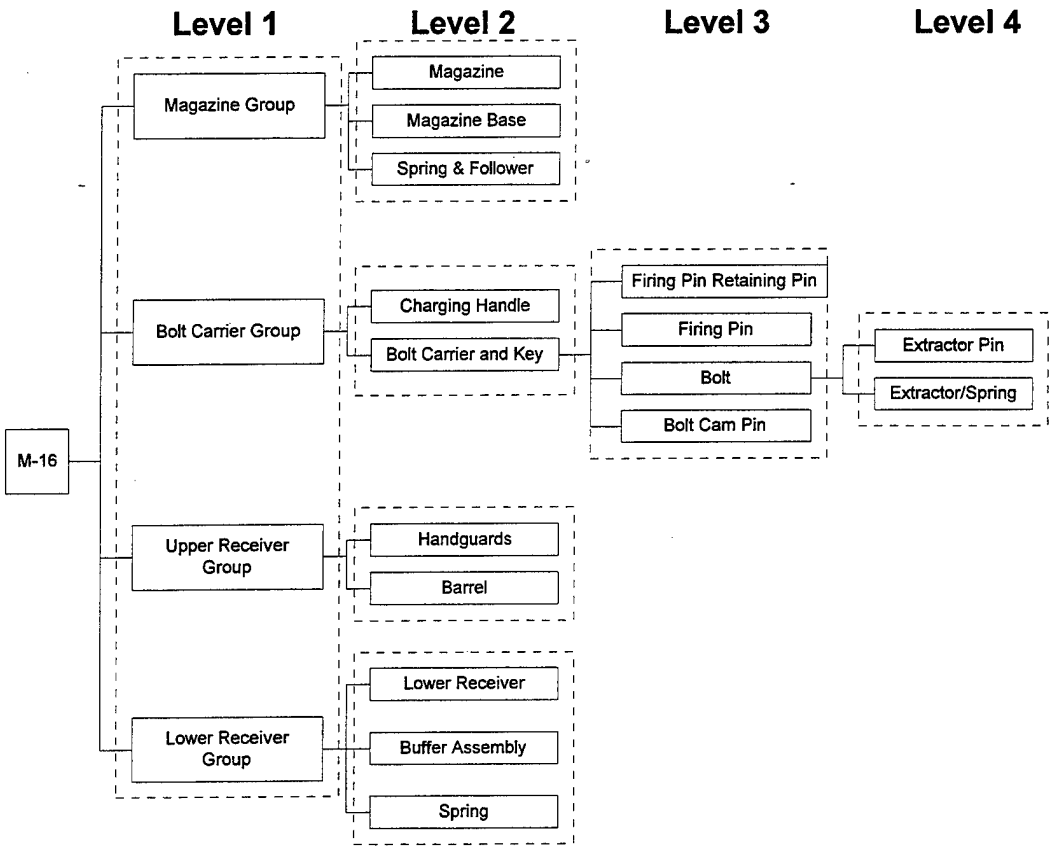


Figure 5. *Component ID/Function*

After the Guided Tour, the student then Demonstrates Knowledge for each item at Level 1. After successfully completing Level 1, the student moves on to the first chunk (shown by the dotted lines) of Level 2. The student receives a Guided Tour of the first chunk at Level 2 (Magazine, Magazine Base, Spring and Follower), then the student Demonstrates Knowledge for each of the items in the chunk. After successfully demonstrating mastery of the first chunk at Level 2, the student then moves to each of the remaining chunks at Level 2 (Guided Tour followed by Demonstrate Knowledge). The student repeats the process for all of the remaining chunks at the different levels.

The Guided Tour shows each component, identifies each component with a label and asks the student to acknowledge each component. The Demonstrate Knowledge portion shows each component, and prompts the student to identify the appropriate component by selecting it with the mouse.

The Demonstrate Knowledge portion of Component Identification and Component Function have a level of proficiency required to graduate to the next chunk. If the defined level of proficiency is not met, then the system will drop back and present the Guided Tour for that chunk again. Reverting to the Guided Tour will allow the student to review the component or procedure information. After reviewing the Guided Tour, the system again presents the Demonstrate Knowledge portion.

The Component Function module uses the same pictures as Component Identification, but substitutes the function of each component for the component names. The Component Identification module is mandatory, while the author determines whether the Component Function module is presented.

Procedures

When performing a procedure, Step Writer provides three different levels to the student: Guided Tour, Demonstrate Knowledge (Level 1), and Demonstrate Knowledge (Level 2), as shown in **Figure 6**. The Guided Tour shows each step within the current chunk of the procedure and specifically tells the student how to perform the step. For the Guided Tour, the only components and actions that are available for selection are the components and actions that are appropriate for the step. At the Guided Tour level, Step Writer provides explicit instructions for use.

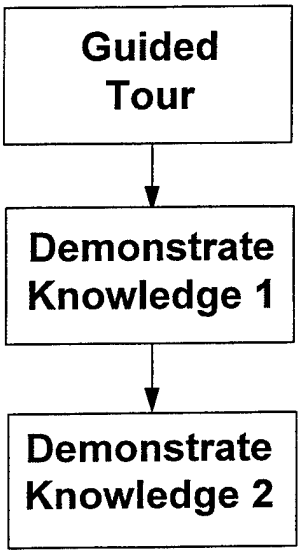


Figure 6. *Levels for Procedures*

Since only the correct action and components are available, the student cannot make a mistake. The Demonstrate Knowledge (Level 1) portion expands on the Guided Tour to include all available actions and components. Therefore, for each step, the students have many actions and components from which to choose. Also, during The Demonstrate Knowledge (Level 1) portion, if the students make a mistake, they will receive immediate feedback about that mistake. The Demonstrate Knowledge (Level 2) is similar to the Demonstrate Knowledge (Level 1) portion, but it does not provide immediate feedback for mistakes that the students make until the end of that chunk.

The Demonstrate Knowledge portions of the Procedures have a level of proficiency required to graduate to the next chunk. If the defined level of proficiency is not met, then the system will revert to the previous level for that "chunk - Demonstrate" Knowledge (Level 1) down to Guided Tour or Demonstrate Knowledge (Level 2) down to Demonstrate Knowledge (Level 1). The students must demonstrate their proficiency at the current level to be promoted to the next level.

Final Exercise

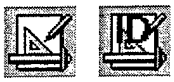
After the students have successfully completed the first three sections (Component ID, Component Function, and Procedures), the system presents the Final Exercise. The Final Exercise is a chance for the students to demonstrate mastery of the entire procedure, not just mastery of the individual chunks. Therefore, the Final Exercise is <u>not</u> limited to 9 steps.

The Final Exercise is a type of procedure. Therefore, it behaves very similarly to the Procedures. However, the Final Exercise has only one level - Demonstrate Knowledge Level 2. If the student exceeds the author-defined performance threshold, then the student graduates. Otherwise, Step Writer forces the students to review the section of the procedure where they had trouble. After successful review of the section students return to the Final Exercise.

Authoring Module

Step Writer's Authoring Module allows authors to construct Tutors using core building blocks for assembly and disassembly tasks (described earlier in **Sec. II Resulting Step Writer Design**). The building blocks include the following: Components, Actions, Step, Procedures, and Screens.

Step Writer uses separate editors for each of these building blocks. This section describes the different editors and support tools that Step Writer employs to build a Tutor.

Components 

Step Writer has two editors which support components: The Component Editor and the Component Identification/Function Editor. The Component Editor allows the author to define the components used in the procedures. A component can have up to seven different views (where each view consists of a bitmap). Within each view, the author may define hotspots which indicate areas of specific interest.

The Component Identification/Function Editor lets the author hierarchically define information about components. The editor guides authors to describe the components in a hierarchical fashion, proceeding from major components to small subcomponents.

Actions 

Since steps are defined as actions on components, Step Writer provides an Action Editor with which the author defines actions for the procedures. Each action is defined with a text string. Therefore, the authors may be as general or as specific as they wish when defining actions.

Steps 

The Step Definition Editor allows the author to define specific steps that make up a procedure. The author provides an Action/Component group which specifies which action must be performed on which component(s) to properly perform the step. The author may specify preconditions, if necessary. The author also specifies what happens when the step is successfully completed. For Composition Assembly, the author specifies which components become available. For Schematic Assembly, the author specifies the line color to be used to connect components.

Procedures 

Two different editors let the author define procedures: the Procedure Editor and the Final Exercise Editor. Both editors let the author define procedures by specifying the steps to include in the procedure. The Procedure Editor lets the author associate a procedure with each step. This lets the author establish a hierarchy of procedures. The Procedure Editor limits the author to 7 +/- 2 steps per procedure.

The Final Exercise Editor lets the author define the final test for the entire procedure. Because the Final Exercise is meant to test a student's knowledge of the entire procedure (not just small chunks), this editor does not limit the number of steps.

Screens 

Step Writer supports three different types of screens: Introduction Screen, Main Menu, and Procedure Screens. The Introduction Screen Editor lets the author define text and graphics as the first screen the student sees. The Main Menu Editor lets the author define a screen with links to the different modules for Step Writer student run-time. The Procedure Screen Definition Editor lets the author define which components appear and where these components appear for the procedure.

Pedagogy/Info 

The Introductory Text Editor and Instructor Editor provide ways to enter miscellaneous information about pedagogy and other supporting information. Each of these editors provide default values so the authors don't have to fill in values unless they want to customize the text messages or feedback messages. The Procedure Information Editor lets author enter text, graphics, and multimedia to support procedures which are not already supported by the Component Identification/Function Editor.

Consistency Check and On-Line Help

Step Writer provides a Consistency Check tool that validates the current Step database for completeness and correctness. The Consistency Check tool will notify the author of inconsistencies of error in the database. The author must correct any inconsistencies before attempting to run the Student Run-Time Module.

Also, Step Writer provides on-line help for the authoring system. This on-line help provides additional information for each of the editors in the authoring system.

VI. Future Directions

As stated earlier, Step Writer currently supports assembly and disassembly tasks. However, many of the Air Force's tasks go beyond assembly and disassembly tasks to include checklists, operation of equipment, etc. Even though the ARC-164 Radio Tutor prototype contained Control Panel Operations, this capability is not available in the current version of the Authoring Module or Student Run-Time Module. In order for Control Panel Operation to be fully supported, Step Writer needs to incorporate simulation capabilities to represent a wide variety of Control Panel domains. Otherwise, the current pedagogy and architecture of Step Writer will easily support the inclusion of additional types of procedures.

Step Writer currently supports a breadth-first order of instruction, as described in **Sec. II Defining the Pedagogy**. Step Writer could be modified to support both breadth-first and depth-first order of instruction to experimentally determine which is more effective.

Also, a large body of procedural data currently exists within the Air Force. While a great majority of it is in paper form, some of it exists in electronic form. Step Writer can be enhanced to take advantage of these existing data sources. One possible extension to Step Writer is to integrate it with the troubleshooting capabilities of MITT and MITT Writer. Another possibility is to import maintenance data from other Air Force systems such as the Integrated Maintenance Information System (IMIS) (Link, et al., 1987).

These are but a few of the existing opportunities within Air Force. As more people use Step Writer, additional needs will be identified. Based on the author's experience with MITT Writer, one area of enhancement will probably center around additional tools and capabilities to support the author. These additional tools could take the form of support for external data (both import and export of data), refined reporting capabilities, enhanced editing capabilities, more detailed definition of actions (to include additional modifiers), etc.

Finally, Step Writer workshops and classes can be held to help increase exposure within the Air Force. These workshops will not only educate the users, but will also help to establish a user community that will drive future enhancements to the system.

V. REFERENCES

Bieger, G. R. & Glock, M. D. (Winter 1984-85). The information content of picture-text instructions. The Journal of Experimental Education, 53, 68-76.

Cohen, V. B. (1985). A reexamination of feedback in computer-based instruction: implications for instructional design. Educational Technology, 25, 33-7.

Kieras, D. E. (1988). What mental model should be taught: Choosing instructional content for complex engineered systems. In J. Psotka, L.D. Massey and S.A. Mutter (Eds.), Intelligent Tutoring Systems: Lessons Learned (pp. 85-111). Hillsdale, NJ: Lawrence Erlbaum Associates, Inc.

Kulik, J. A., & Kulik, C. C. (1988). Timing of feedback and verbal learning. Review of Educational Research, 58, 79-97.

Link, W. R., VonHolle, J. C., & Mason, D. (1987). Integrated Maintenance Information System (IMIS): A maintenance information delivery concept. (AFHRL-TP-87-27). Wright-Patterson AFB, OH: Armstrong Laboratory.

McKendree, J. (1990). Effective feedback content for tutoring complex skills. Human-Computer Interaction, 5, 381-413.

Miller, G. A. (1956). The magical number seven, plus or minus two: Some limits on our capacity for processing information. Psychological Review, 63, 81-97.

Norton, J. E., Wiederholt, B. J. & Johnson, W. B. (1991). Microcomputer intelligence for technical training (MITT): The evolution of an intelligent tutoring system. Proceedings of the 1991 NASA-Air Force Conference on Intelligent Computer-Aided Training, Houston, TX: L.B. Johnson Space Center.

Rieber, L. P. (1990). Animation in computer-based instruction. Educational Technology Research and Design, 38(1), 77-86.

Wiederholt, B. J., Norton, J. E., Johnson, W. B., Browning, E. J. (1992). MITT Writer and MITT Writer Advanced Development: Developing Authoring and Training Systems For Complex Technical Domains. (AL-TR-1991-0122). Brooks AFB, Texas: Air Force Systems Command.