USING COGNITIVE SIMULATIONS IN MULTIMEDIA FOR MAINTENANCE TROUBLESHOOTING TRAINING: PRACTICAL, COST-EFFECTIVE SIMULATIONS

T. Kent Thomas
Allen Communication, Inc.
5225 Wiley Post Way
Salt Lake City, Utah 84116

ABSTRACT

Experts appear to master the art of critical thinking in troubleshooting. It's as if they have a mental model of the system etched on the inside of their forehead. How can this mental model be transferred to the novice? Through carefully crafted multimedia courseware and free-play simulations, novices can match wits with the expert in a delivery environment that doesn't require either expensive expert systems software development nor complex hardware simulators. Preliminary training results from a 200-plus hour program suggest that interactive multimedia courseware may produce results approaching both of those methods, with substantially lower development and delivery costs. Small-group tryout results from 21 courses developed by Allen Communication for Air Force maintenance technicians show a 25% aggregate increase in knowledge, and a striking 79% aggregate leap in the ability to successfully apply expert troubleshooting strategies to simulated problems.

The mental models of experts, the sequence of troubleshooting actions they perform, and their reasoning have been captured using cognitive task analysis methods and used as the basis of courseware design. Experts' mental models form the foundation of the tutorials that comprise approximately 70% of the courseware. Their performance on complex troubleshooting problems is the basis of the simulated troubleshooting scenarios. Combining this detailed cognitive task analysis with high-impact motivational video, focused in-depth tutorials that directly depict the mental models of experts, and extensive free-play simulations, this F-15/F-16 Maintenance Continuation Training Program won the 1993 *Nebraska Interactive Media Award* for the most significant achievement in the Government/Military category and an Intermedia *Invision* Bronze Medal.

The author will present an overview of the methods used to design and develop these simulation-focused multimedia courses, including: knowledge engineering, design, programming, and evaluation. Courseware samples will be demonstrated and preliminary results reported.

ABOUT THE AUTHOR

Mr. Thomas is a former Air Force education/training officer and computer systems requirements analyst. He has a B.A. from Carson-Newman College and is completing a Masters in Instructional Technology from Utah State University. While previously on active duty, he planned and managed a variety of new applications of training technology, including the design and implementation of Air Education and Training Command's largest operational computer-managed instruction system and Air Combat Command's maintenance continuation training program using interactive videodisc. He has managed the design and development of over 70 multimedia courses, including four winners of the *Nebraska Interactive Media Award* (three consecutive winners in the government/military category) and three winners of the Intermedia *Invision Award*. A frequent presenter at TITE/IITSEC, Mr. Thomas is Vice President, Research and Development, at Allen Communication and a Major in the Air Force Reserve.

USING COGNITIVE SIMULATIONS IN MULTIMEDIA FOR MAINTENANCE TROUBLESHOOTING TRAINING: PRACTICAL, COST-EFFECTIVE SIMULATIONS

T. Kent Thomas
Allen Communication, Inc.
5225 Wiley Post Way
Salt Lake City, Utah 84116

INTRODUCTION

Air Combat Command (ACC) has been using interactive videodisc courseware for several years to provide maintenance continuation training (Thomas, 1987). Although quite effective, this courseware has primarily focused on systems knowledge and procedural training to support weapons systems conversions such as to the F-16 Block 40 and F-15E. Consequently, the maintenance problems related to troubleshooting these complex, highly integrated systems have persisted. Further, this courseware for conversion training has been based *primarily on the design principles of behavioral* psychology (Hannafin and Rieber, 1990), with either weak or nonexistent simulation capabilities. Meanwhile, ACC has been a strong supporter of research programs that evaluate technologies (such as "cognitive task analysis" and "simulation-based intelligent tutors") that address these persistent problems -- especially the ongoing Basic Job Skills (BJS) project conducted by Armstrong Laboratories. This paper documents the early results of transitioning some of this research to full scale development in the F-15/F-16 Maintenance Continuation Training Program (MCTP). It describes both lessons learned from cognitive-based courseware design and indicates areas with high potential for further research.

According to the Productivity Investment Funding (PIF) package that supported the MCTP, approximately one quarter of the components removed from the F-15 and F-16 during flight line maintenance are actually serviceable (F-15, 25%; F-16, 27%) when bench-checked. While there are many factors other than training (or the lack of it) that directly contribute to these statistics (including management pressures for "quick fixes"), poor *troubleshooting skills are obviously a* part of the problem. These challenges are further compounded by initiatives such as Rivet Workforce (which tripled the technicians' responsibilities in some cases) and two-level maintenance (which provides no local "back-shop" support to test and verify the condition of the components removed during maintenance). These troubleshooting weaknesses are expensive in terms of time, logistics funds, and their potential impact on readiness. The potential paybacks identified on this \$4.2M courseware investment are \$11M in the first year and over \$100M in savings across the anticipated life-cycle of these weapons systems.

PROGRAM DEVELOPMENT GOALS AND REQUIREMENTS

The F-15/F-16 MCTP contract required the development of 21 courses (10 for the F-15 and 11 for the F-16), each approximately 10 hours of average student contact time, which addressed ten separate maintenance specialties. It represented a ground-breaking sythesis of several isolated Air Force research and development projects. These innovations, when compared to other interactive courseware development efforts, lie in four primary areas: Knowledge Assessment Tools (KATs), Cognitive Task Analysis, Simulation-Based Instruction, and Automated Analysis Package.

Knowledge Assessment Tools (KATs)

Drawing upon previous Air Force research in assessing the *knowledge and skills* of technicians on the A-10 Stability Augmentation System and the F-15 APG-63 Radar System, the MCTP contract required the development and validation of tests prior to designing the associated course. This approach formalizes and implements the concept of "test before you train." If a technician does not need the training on a given topic or system, the test results allow the student to be "routed around" that particular portion of the courseware. This *was done to significantly improve the training efficiency* and the students' motivation since the students were to be trained only where needed.

Validating the KATs early in the development process (in the analysis phase, prior to design) allows the pretests to be much smaller and more efficient. Rather than a comprehensive test of all the content included in a given course or module, only key questions or problems that have been statistically validated to discriminate between novices and experts are posed to the student. This further reduces the amount of time the students are tested and trained. Validating the tests and analyzing the data prior to design eliminated many potential topics from the training -- topics whose data did not support their inclusion since the content was commonly known -- while other topics were either included or their coverage expanded as a result of this analysis. This process ensured that the training focused on only the most needed areas. Finally, much invaluable data about common misconceptions and misunderstandings was gathered for inclusion in the later design of the associated courses.

Cognitive Task Analysis

Cognitive task analysis is a set of procedures to define the differences between how novices and experts think about a given system or problem. As previously mentioned, ACC staff personnel have been strong supporters of the Basic Job Skills (BJS) program which researches the use of a cognitive task analysis methodology to design "intelligent tutors" for teaching complex troubleshooting performance. By focusing on the experts' thought processes and presenting them in the context of robust training simulations, preliminary indicators from the initial test of the BJS F-15 Avionics Manual Test Station Tutor revealed a significant potential to condense the on-the-job troubleshooting experience into a short training course (Gott, 1989).

Though the MCTP contract did not require a specific cognitive task analysis methodology, it did require that it be performed in the analysis phase, prior to design, and that the results be used as the basis of designing the courseware and, specifically, the troubleshooting simulations. It clearly stated that the goal was to teach novices to model the experts' performance on troubleshooting problems -- i.e. to teach them to think like experts.

Simulation-Based Instruction

Acknowledging the importance of simulations to teach complex tasks related to complex systems, the MCTP contract required approximately 30% of the courseware to be simulation-based. Simulations are generally regarded as necessary in order to provide relevance to "real-world" tasks, yielding more valid assessment of knowledge and skills while enhancing the transfer of learning to the actual job environment. Further, the MCTP required that these simulations be based on the results of the cognitive task analysis as previously discussed.

While this emphasis on simulations incorporates some of the research findings from the BJS project, note that this approach differs significantly from that used in the actual design of the BJS tutors. The BJS tutors use only simulations with "coaching" as an instructional strategy -- all learning takes place in the context of a simulation, with "hints" or "help" available upon the student's request. The MCTP contract allowed approximately 70% of each course to be provided by other instructional strategies, primarily the more common (and less expensive) tutorials.

Automated Analysis Package

Intended for use by senior managers and staff, the MCTP contract required the development of automated programs so that planners and managers would have the necessary data to show both the training-effectiveness and cost-effectiveness of the program. Specifically, it required automated programs that correlated pretest to post-test scores by course and module to show training effectiveness of each course. It also required pretest versus post-test correlation by core maintenance tasks (from the Specialty Training Standard or Job Qualification Standard) and by the work unit codes reported in the Core Automated Maintenance System (CAMS) to show cost-effectiveness. In a nutshell, it required automated programs to point toward maintenance trends on those troublesome systems that would "prove" the program met its goals. Much of the data reported here came from those programs.

METHODS USED

The original timelines specified in the MCTP contract reflected an intense development cycle, given all the data requirements, analysis, and reviews. The first four courses were to be delivered within 10 months of contract award and the remaining 17 on a 12-month development cycle. All were to be delivered within 32 months. As the first four courses were under development, it became evident that these time frames were too stringent. Changes to the requirements were *negotiated. The development cycle was extended to 14* months for each course (at no additional cost to the government), though overall program duration remained fixed. The KAT development and cognitive task analysis schedules were merged to allow them to occur concurrently rather than sequentially, as specified, yet still prior to courseware design.

Further, this effort was ongoing during the *entire* duration of Desert Shield and Desert Storm. Due to high reliance of this project on Air Force personnel and support, the impact of this large-scale military operation was all-pervasive. Schedule changes and personnel changes were frequent, as were limitations on reviewers, sample students, and access to equipment or systems. The design and development methods used were under constant refinement, change, and full of "work-arounds." The following discussion of the methods used is provided as a *general*, somewhat "ideal" process, that likely was *never fully incorporated on any one course.*

KATs and Cognitive Task Analysis

As discussed earlier, the goal of the KAT was to develop a diagnostic device that was "better, faster, and *cheaper" than a traditional pretest. It must sample* learner knowledge and skill in critical areas, assess current level of performance, and direct the learner to the appropriate instruction. The following development process was used.

1. Screen and identify project subject matter experts (i.e. SMEs). A critical part of any cognitive task analysis or "knowledge engineering" effort is identifying *the right SMEs*. Since their understanding of the subject system is to be captured as the basis of the software system to be developed, they must truly be experts. A variety of methods was used to identify SMEs, including: 1) surveys for screening subject-matter experts and review their experience, 2) interviews with SMEs, managers, and supervisors, and 3) perhaps most valuable, soliciting recommendations from their peers. Though the peer recommendations were almost invariably more accurate and were relied upon more, they were not used without corroboration by other means.

Questions were posed to these members of the target audience such as "If you were faced with a very difficult problem on system XXX that you have been unable to solve, who would you choose to help you solve it (from any base in the Air Force) and why?" Several interesting *points came to light in this process, including: 1) the* peer groups understand and acknowledge expertise in troubleshooting, 2) individual expertise (even for those acknowledged as "experts") varies widely from one system to another system or to another part of the same system, and 3) there are very few, if any, experts on entire systems -- their expertise is *very domain specific*. Frequently the experts on one subsystem were no more than average performers on a related subsystem.

2. Develop knowledge surveys using initial SMEs. Usually two SMEs identified by the above process were employed to develop initial knowledge surveys used to further screen their peers. Note that this survey was not intended to truly determine expertise, only to place the individuals in an initial category for further analysis. Note also that the questions included in the survey were frequently word problems that described a troubleshooting scenario or very technical questions about the specific functioning of a system or subsystem. At the same time, these SMEs were asked to define representative troubleshooting problems (including problem set-ups and step-by-step solutions) for possible inclusion in the course. A preliminary cognitive *task analysis was performed for these problems using* the PARI method (described below).

3. Perform cognitive task analysis of both novices and experts. Developed as part of the BJS program, the PARI methodology is a set of data collection procedures, including structured interviews, designed to capture and document an expert's actions and reasoning processes simultaneously (Hall et al, 1990). Each step in the expert's performance is analyzed to determine the Precursor (i.e. pre-existing condition), their specific Action and the reasons for it, the associated Results of that action, and their Interpretation of those results, hence the acronym PARI. A somewhat streamlined version of the PARI method was used to interview approximately four experts and four novices from at least two different Air Force bases. The changes to the published PARI methodology were minor, intended to support "mass production" and (in the author's opinion) do not substantively change the results.

Note three minor, yet perhaps valuable, changes to the PARI technique. First, the term "problem" was used in lieu of "precursor" since it communicates better to laymen and since the troubleshooting strategy to be *taught was space-splitting (also known as split-half or* half-split). This change in terminology kept both the interviewers and interviewees focused on the "problem space" and how each step did or did not continue to reduce it. Note that each of the courses presents the PARI technique directly to the students, in the context of a "generic" Troubleshooting Techniques lesson, and again in the context of a guided, system-specific simulation, using this terminology of Problem-Action-Result-Interpretation.

PARI interviews are very dependent on visual representations of a mental model of the system. The second change was to provide the "experts" with drawings of the mental models developed by the initial SMEs, rather than having them draw their own for illustrating their solutions. They could critique or modify them as necessary. The goal was to develop and document a common, *fundamental mental model* (Lesgold et al, 1988) that could be incorporated directly in the courseware to illustrate the troubleshooting *process to the learners, and to simultaneously document* and depict how it is expanded, modified, or otherwise manipulated as the troubleshooter delves deeper into the problem. While this technique generated considerable discussion between the SMEs, consensus could almost always be gained. This resulted in a common, yet robust, mental model of the system (though perhaps not an optimum nor universal one) that can be validated for functionality and illustrated via computer graphics and animation in the courseware. [See Wilson and Rutherford (1989) for a further discussion of mental models.]

Finally, the interviewing of novices was treated almost as critically as the interviewing of experts. The mistakes and misunderstandings of novices were documented as clearly as possible since common mistakes and misconceptions were to be directly addressed in the courseware. The combination of actions taken at each point by both novices and experts would define the limits of free-play to be designed into the simulations.

4. Develop alpha version of KAT and review it with SMEs. Based on the cognitive task analysis performed above, the knowledge survey was expanded to include more "key tests or checks" and the interpretation of results, in addition to the word problems and *technical questions previously developed. This is the* foundation of the KAT. The KAT items were then reviewed by the initial SMEs for accuracy prior to field testing.

5. Develop beta version of KAT and validate with field SMEs and novices. After any necessary revisions, the beta versions were administered to representative members of the target audience at two or more Air Force bases. The sample size was ideally 20 members, with a fully representative range of experience on this specific weapons system. More subjective data about the subjects was gathered simultaneously, including evaluations of their performance from their supervisor(s) and a summary of their background and experience.

6. Analyze results and select final items. Responses to KAT items were analyzed, item by item, in three different ways using a point-biserial method to determine item validity. In fundamental terms, this validation identified KAT questions that discriminate between novices and experts. Subjects were first categorized for analysis by a subjective evaluation (based on experience, and supervisor and peer input) as *experts, average performers, and novices. The central* category was added to accommodate those who are truly not experts nor novices. Then, the subjects were categorized by the skill level from their Air Force Specialty Code (AFSC), i.e. 3-, 5-, and 7-level. Each item was evaluated for significant differences between the results for the subjective grouping and the subject's AFSC skill level. Finally, each student's overall KAT score was used to assign him/her to the subjective grouping and the items analyzed to provide an internal referent. Minor or occasional anomalies were expected and accepted. By definition, any test item with a positive point biserial value will discriminate. However, only the items with the highest positive values were included in the courseware, since the higher the value the better the item discriminates and the more confidence that can be placed in the results. Though there were significant variations across all 21 KATs, the vast majority of item validity scores were above 0.33. The KR20 reliability index was calculated for each test to evaluate its reliability. By its simplest definition, reliability is the ability of the test to provide consistent results, all external factors being equal. A composite threshold value of 0.80 was established for the reliability of each test, considered as a whole. However, given the sample sizes for KAT validation efforts, this should still be *considered a rough estimate of reliability.*

Given the administration of the KAT to two pools of students at two bases and the fairly high validity and reliability criteria used, there was confidence that the KAT would make consistent, accurate diagnoses of knowledge. The final step was to refine the objectives for each course module to be developed, correlate the KAT items to the objectives, and ensure adequate numbers of valid and reliable items were available for content coverage. In those isolated cases where adequate, validated items were not available, additional items were generated and validated in conjunction with small group tryouts.

Simulation-Based Instruction

While the simulations are the "heart" of the instruction, they are only a part of the overall design. The KAT and cognitive task analysis results are incorporated throughout each course, as is careful attention to the target audience.

General Structure and Design. Since the target audience for this training includes both apprentice and journeyman maintenance technicians, and the training is performed on a time-available basis, modularity is critical. Each course is designed so that students may enter quickly, access their previous position in the course, receive intensive instruction, and then exit easily. Overviews, introductions, and navigational lessons are mandatory for first-time students and then available as options. The courseware allows the student access to various options such as Help, Take a Break, Course Status, Expert Tips, and flight line Logbooks. Advance organizers and summaries orient students to the overall content of the modules. Students can exit the course quickly by selecting a Logoff option, which places a "bookmark" allowing them to reenter the course at the point they left it. Bookmarking also allows the students to browse through and repeat instruction they have already seen. Each course uses the same structure and conventions, since one student may take several courses in the MCTP series.

Motivational and affective learning strategies were integrated to make the instruction appealing to the students and encourage knowledge transfer. For example, a theme song was commissioned to a professional song-writer and mixed with high-impact video (including considerable Desert Storm footage). It builds and reinforces the intrinsic motivational theme of *"When The World Has Its Eyes On Me"* and is incorporated throughout each course. On-camera role models introduce, guide the student, and conclude each module. Learning games similar to *Concentration*(tm) or roulette wheels (where students can wager points against their ability to answer technical questions) are used for review, practice, and reinforcement. "Commercial breaks" are interjected throughout to surprise and entertain the student, helping to maintain motivation.

The instruction in each course consists of four modules covering: Subject System, Related Systems (and their interfaces with the subject system), Diagnostic Tools and Procedures, and Troubleshooting. The troubleshooting module is simulation-based, as discussed earlier, while the other modules are largely tutorials, with linear introductions and gaming strategies for reviews and practice. Test items from the KAT development process are included as separate pretests and progress checks for each of the first three modules. Students enter the troubleshooting instruction and free-play simulations only after passing KATs or completing the previous modules. If the KAT determines that the students need instruction, they are routed to the appropriate module (which contains two or more lessons). Each lesson instructs the student about a system's main components, functions, working relationships, and common troubleshooting problems. These "chunks" of information, over 30 per course, are called clusters.

Each cluster further evaluates the student's knowledge and ability via a conversation-like Socratic dialogue technique, and then, depending on the results, will route the student to in-depth tutorials. This dialogue provides a more discrete "diagnosis" once the KAT has indicated that a weakness exists. This design allows the more knowledgeable students to progress very quickly through the course by correctly answering questions. Weaker students receive training that is individualized to address their specific weaknesses. This approach has allowed some students to complete a course in four hours or less, while others spent the two days allocated for tryouts and still were unable to complete the course.

In these tutorials, students receive extensive training about the system which fills in their specific "knowledge gaps." The instruction in each lesson is accomplished by using voice-over narration or text, supported by full-color graphics and video of proper maintenance procedures. The mental model for the system is introduced in the Subject System module and is explained in the context of describing the system's functionality. It is reinforced in the Related Systems module, where it depicts the interface(s) to the related systems on the aircraft. In the Diagnostics module, it is *used to illustrate where certain tests are performed*, test equipment is used, etc. It is also incorporated in the guided simulation of the Troubleshooting Module as feedback to depict and reinforce effective space-splitting.

Troubleshooting Simulations. Each course contains six to nine specific troubleshooting simulations, used throughout the final module. Simulations are used as the pretest for the module, the specific lessons within it (a generic Troubleshooting Techniques simulation as discussed earlier, plus a course-specific guided simulation), to provide practice in application, and they serve as the progress check. These are fairly robust and powerful free-play simulations. The student can perform between 60 and about 150 actions (depending on the course) on the system in question, in any sequence at any time. These actions are as varied as talking to the pilot, researching the maintenance history, performing tests with diagnostic equipment, making visual inspections, replacing parts, and performing operational checks. Each simulation contains well over 1,000,000 possible paths, and some significantly more.

All the while, performance indicators track and display a running evaluation of the student's performance. Each step is evaluated to meet one of four criteria: 1) critical to successfully solving the problem, 2) a reasonable step, though usually neutral in value toward solving the problem, 3) an unreasonable step, or costly in terms of time or parts, or 4) a safety violation, which results in immediate termination of the simulation. The total number of steps taken is displayed, as is the number of critical steps and unreasonable steps, the time expended (in terms of the actual maintenance time it would take to perform the action), and the cost of parts used. Students are allowed to take three unreasonable steps before the simulation is terminated to provide feedback and a summary evaluation. The simulation is also terminated if one or more of the efficient time, total steps, or cost limits (140% of optimum) is exceeded. In the summary, the student's performance is directly compared and contrasted with the expert's performance in terms of all these factors. A step-by-step comparison of each action taken and the corresponding result is displayed. Students must successfully troubleshoot at least two randomly generated aircraft system faults before passing the course.

In addition to being instructionally powerful to depict cause and effect relationships in a complex system, *these simulations were consciously designed and used* to provide the most value for the least cost to the government. While definitely effective, simulation-based training is expensive -- interactive courseware simulations can often cost at least three times as much to develop as simple interactive tutorials. Yet, the relative training effectiveness of simulations versus other instructional strategies is not clear (Fletcher, 1990). The MCTP attempts to use a more optimum mix of simulations with other strategies to increase *cost-effectiveness*. Instead of all the content being presented in the context of simulations (as do most "intelligent tutors" for maintenance skills), simulations are used here almost exclusively for illustration, practice application, and assessment. Tutorials that are focused on the carefully-analyzed knowledge and cognitive abilities that support troubleshooting are used as the primary presentation strategy.

Further, the courseware uses direct instruction methods to impart the information in the most time-efficient manner. Rather than leaving the students to formulate their own mental model of the system, a common one is *presented directly*. *Rather than inferring the expert's* approach, their specific troubleshooting sequence is presented directly for step-by-step comparison with the student. Rather than implying the cost and/or value associated with each step or action, they are displayed constantly as running totals in the performance indicators, and again in a detailed step-by-step summary after each simulation.

What is simulated also differs from most instructional simulations (Alessi and Trollip, 1991). Rather than building an extensive, realistic simulation of the system under study, likely human actions are modeled. The actions that novices and experts are likely to take are incorporated as limits to the free-play choices. The realism (or fidelity) was also focused on the areas perceived to have the most instructional value. While the results of a student's actions are clearly depicted with pictures (and sometimes sound or explanatory text) to support the student's interpretation, quite often the action choices are merely "menu selection items." Similarly, these simulation scenarios are used fully and thoroughly. A typical student will experience almost every simulation scenario in each course, even though they are presented randomly (with extinction, then "reshuffled") to prevent compromise. In comparison, a typical computer-driven flat-panel avionics trainer may provide the capability to simulate over 100 problems, yet the student will only experience about ten of these in a *typical course, due to the limitations on training time* and course length.

Also impacting cost-effectiveness over the life cycle of training programs is the way the simulations (and all the courseware) were developed with a commercial off-the-shelf (COTS) authoring system. They were developed entirely in the *Quest* (tm) Multimedia Authoring System, as required by the MCTP contract. Consequently, these simulations can be updated and maintained by experienced Air Force subject-matter personnel, while neither intelligent tutors nor flat-panel trainers can be. Updates can also be affordably done via competitive contract if necessary. The simulations are "driven" by a small, yet very powerful "simulation engine" written in the underlying *Quest* (tm) Authoring Language that is compiled for speed and used for all the courses. The screens used to present specific actions and results are in lesson files, with no inherent branching or paths, while all logic and scoring criteria are contained in a delimited text file (i.e. table) that is created and updated using a spreadsheet. The "simulation engine" reads the action taken, determines the appropriate result based upon the rules table, and displays the correct result screen. This modular approach, separating logic from content, speeds up initial development and makes the simulations much easier to update and maintain, reducing life cycle costs even more.

Automated Analysis Package

All these innovations may be moot without the final one -- an automated analysis package that links learning gains back to the maintenance environment. The previous innovations were transitioned from research and development (R&D) to large-scale implementation. This last one is still an R&D effort -- to the author's knowledge, nothing like it has been attempted in either the military or industry.

Pretest and progress check scores for each question or simulation scenario are captured for each individual in each course. This data is then loaded to a *dBase III* (tm) compatible database format for consolidation and further analysis by specialized programs. These programs provide correlation of these scores by module and course to show training effectiveness data, a common practice. More importantly, they provide a similar correlation of test questions and scores to core tasks and to work unit codes, as discussed earlier. Summaries are provided by student, an aircraft maintenance unit, a fighter wing, an Air Force Specialty (AFS) or an aggregate of all students who have taken a given course (regardless of AFSC or unit of assignment). This same test item data can also be loaded to an Air Force-developed test-item analysis package for evaluating individual test items and tests in terms of difficulty, reliability, validity, etc.

In addition, critique data from a standardized, Air Force-provided 17-question automated survey is gathered from each student for analysis. There is also a vehicle for unsolicited feedback in a "free-form, anonymous mode." The results described below were gathered using these standardized packages.

RESULTS

The F-15/F-16 MCTP appears to have successfully transferred research and development tools and techniques to a "production line." All 21 courses have been delivered and accepted by the government. The data available at this point is from small group tryouts of each course, since some of the courses have just been delivered and several have yet to be fielded. More meaningful data should become available when each course receives a scheduled follow-up data extraction and analysis. By the time this paper is published, that data should start becoming available.

Limitations of Available Data

Given the source, the sample size per course is small. Conclusions would be premature for any specific course. Note also that the skill distribution of students at small group tryouts was not truly representative of the target audience. An equal representation of novices, average performers, and experts was sought for small group tryouts. However, since experts could logically be expected to score highest on pretests, yet make up the smallest percentage of the target population, the data from a numerically representative sample should show even higher gains in knowledge or skills. The data has not been separately analyzed by skill level.

Another limitation is the varying numbers of students who completed a given module, especially the final one, Troubleshooting. Much of this is due to the inability of students to complete the course in two days (the maximum time available). There were also a few cases of power interruptions and hardware or software failure during tryouts which impacted limited portions of student data. These inconsistencies have been accommodated and compensated for in the analysis of results where possible. For example, only those completing a progress check are calculated in the pretest data. Though the available data is *preliminary* by definition, coming from small group tryouts, each course successfully met the tryout criteria specified by the Air Force. When considered in a holistic fashion, some indications should be valid related to the effectiveness of the basic approach and the courseware design, due to the commonality across all courses and the aggregate sample size of all tryouts.

Table 1. MCTP Knowledge Gains -- Pretest vs. Progress Check Increases (Data From Small Group Tryouts)

	Subject Systems Module			Related Systems Module				Diagnostics Module		Total
Course	Pretest	Prog Ck	Difference	Pretest	Prog Ck	Difference	Pretest	Prog Ck	Difference	Difference
F-15/8	50	89	39%	NA	NA	NA	51	94	43%	41%
F-16/3	53	88	35%	54	96	43%	58	90	32%	37%
F-15/2	42	84	42%	59	98	39%	69	98	28%	36%
F-16/9	48	86	38%	55	92	37%	56	84	28%	34%
F-16/1	56	83	26%	58	89	31%	46	88	43%	33%
F-16/10	63	92	28%	59	92	33%	60	83	23%	28%
F-16/11	53	85	31%	63	86	23%	61	90	29%	28%
F-16/7	65	91	25%	58	89	31%	69	93	25%	27%
F-15/5	72	82	10%	68	100	32%	70	100	30%	24%
F-15/10	68	91	23%	69	88	19%	62	91	29%	24%
F-16/4	64	86	22%	61	81	20%	57	86	29%	24%
F-16/6	72	88	21%	56	88	31%	65	85	18%	23%
F-15/1	59	87	29%	62	96	34%	83	88	5%	23%
F-16/8	72	88	16%	56	88	32%	65	85	20%	23%
F-15/9	63	91	28%	NA	NA	NA	74	91	17%	23%
F-15/7	61	92	31%	70	89	19%	76	93	17%	22%
F-15/4	74	86	12%	64	83	20%	56	90	34%	22%
F-16/2	62	85	23%	75	93	18%	68	86	18%	20%
F-16/5	63	88	25%	72	85	13%	68	88	20%	19%
F-15/6	70	85	15%	73	82	9%	70	89	19%	14%
F-15/3	74	89	16%	78	91	13%	93	90	-3%	9%
Mean	62	87	25%	64	90	26%	66	90	24%	25%
Median	63	88	25%	62	93	31%	65	90	25%	24%

Training Effectiveness

Aggregate data regarding knowledge gains from a sample of 183 students at small group tryouts shows overall knowledge gains (from the first three modules) average 25% (see Table 1). The data from these three modules reflect testing of the background knowledge required for troubleshooting. This amount of increase is not unexpected and is of no particular significance, based on the author's experience, other than showing that considerable learning took place.

Note that the pretest scores by module average 62%, 64%, and 66% respectively, else the increase in knowledge scores may have been higher. This can be attributed primarily to two factors. First, the students had a fairly high entry knowledge, since they have already graduated from formal technical training and had flightline experience. Secondly, all tests are "open-book" with no time limits, so students had the opportunity to research their answers. This ready access to technical data is *intentional* since the students are *required* to use the technical data in actual flightline performance of all job tasks.

The increases in performance as measured by the simulations used for pretests and progress checks in the Troubleshooting modules appear to be quite significant. Only 14% of the students could solve the problems in the pretest while 93% solved them in the progress checks, an increase of 79% (see Table 2). The low pretest scores appear especially significant given that the students are supplied and encouraged to use technical data (whicH often includes very specific fault-isolation guides) during the *entire* course.

A general finding is that while students may know "facts and figures" about a system, they still do not truly understand how it functions -- especially when a component of the system has failed. While they may

Table 2. MCTP Performance Gains on Troubleshooting Simulations (Data From Small Group Tryouts)

Course	Pretest Attempts	Pretest Passed	Pretest %	Progress Check Attempts	Progress Check Passed	Progress Check %	Difference
F-16/1	7	1	14%	7	6	86%	71%
F-15/1	9	1	11%	8	8	100%	89%
F-16/2	8	1	13%	7	6	86%	73%
F-15/2	7	0	0%	7	6	86%	86%
F-16/3	NA	NA	NA	NA	NA	NA	NA
F-15/3	8	1	13%	7	5	71%	59%
F-16/4	9	0	0%	9	9	100%	100%
F-15/4	8	2	25%	6	6	100%	75%
F-16/5	10	2	20%	9	9	100%	80%
F-15/5	11	4	36%	9	7	78%	41%
F-16/6	12	1	8%	11	11	100%	92%
F-15/6	11	0	0%	11	11	100%	100%
F-16/7	12	8	67%	7	6	86%	19%
F-15/7	7	0	0%	7	6	86%	86%
F-16/8	12	1	8%	12	11	92%	83%
F-15/8	9	1	11%	9	9	100%	89%
F-16/9	8	2	25%	7	6	86%	61%
F-15/9	8	1	13%	7	7	100%	100%
F-16/10	8	0	0%	8	8	100%	100%
F-15/10	10	0	0%	10	10	100%	100%
F-16/11	9	0	0%	9	9	100%	100%
Totals	183	26	14%	167	156	93%	79%

Table 3. F-16 Pretest to Progress Check Score Increases Correlated by Work Unit Code -- Partial List

Course	WUC	Descriptor	Pretest (Average)	Progress Check Avg.	Difference
16/3	14BCO	Integrated Servoactuator, Flaperon	29%	100%	71%
16/2	13AAD	Handle Assembly, Landing Gear, Pilot's	38%	100%	62%
16/2	14AQO	Panel Assembly, Digital Flight Control	43%	100%	57%
16/4	74AWO	Cable, Transmit Drive	44%	100%	56%
16/4	74CEO	General Avionics Computer	28%	83%	55%
16/4	74KAO	Multifunction Display	47%	100%	53%
16/7	27GMO	Hydraulic System	42%	94%	52%
16/8	75BOO	External Stores	32%	83%	51%
16/2	14DAO	Leading Edge Flaps	50%	100%	50%
16/3	14BAO	Integrated Servoactuator, Rudder	50%	100%	50%
16/3	14BBO	Integrated Servoactuator, Horizontal Tail	50%	100%	50%
16/3	14AQO	Panel Assembly, Digital Flight Controls	42%	92%	50%

Table 4. Pretest to Progress Check Score Increases Correlated by Task (From STS or JQS) -- Partial List

Course	Core Task	Pretest (Average)	Progress Check Avg.	Difference
16/2	Troubleshoot Aircraft Wiring	39%	100%	61%
16/2	Use Data Transfer Terminal Switch	39%	100%	61%
16/5	Rig Engine Power Control System	40%	100%	60%
16/1	Use Jet Fuel Starter (JFS) Analyzer	34%	90%	56%
15/2	Perform Operational Checkout on Manual Flight Controls	45%	100%	55%
16/3	Use Portable Hydraulic Test Stand	42%	95%	53%
16/4	Troubleshoot RF Wiring	48%	95%	47%
16/3	Trace Signal/Data Flow in DFLCS	53%	100%	47%
15/2	Trace Wiring/System/Interface on Manual Flight Controls	47%	94%	47%
15/5	Troubleshoot External Fuel Tanks System	53%	100%	47%
16/4	Isolate Faulty Antennas	41%	87%	46%
16/3	Use Hydraulic Test Stand - Support Equipment	50%	95%	45%

know how to perform a given troubleshooting procedure, they do not understand when and why to perform it in the context of actual problems -- they cannot *efficiently* select which procedure to use in specific circumstances. These findings are substantiated by the high pretest scores on the previous modules (especially the Diagnostics module, which covers diagnostic tests and test equipment), yet poor performance on the Troubleshooting pretests. Note that the modules (including pretests and progress checks) are presented in sequence, so the students have a firm knowledge foundation prior to taking the Troubleshooting pretest simulations. Yet, only 14% of the students could pass these problem-solving pretests.

Cost Effectiveness

As discussed earlier, the small sample size from small group tryouts does not support conclusive analysis of data on a specific course. However, this preliminary data is quite encouraging. When analyzed by Work Unit Code (i.e. a maintenance action related to a specific component, assembly, or subsystem), twelve F-16 components show a troubleshooting knowledge increase of 50% or more (see Table 3). Dozens show more than a 30% increase. Similarly, nine F-16 tasks show a 45% or more increase (see Table 4). Should these preliminary indications bear out, savings related to just one F-16 maintenance task (e.g. Isolate Faulty Antenna for the APG-68 Radar, with a 46% increase as in Table 4) or reductions in supplies of just one F-16 component (e.g. General Avionics Computer, with a 55% increase as in Table 3) could feasibly pay for this entire courseware development effort.

Subjective Feedback from Critiques

Analysis of the aggregate data from the standardized, automated, Air Force-provided critiques is quite positive overall (see Table 5). It averages 3.97 on a 5-point Likert scale (1 = low, 3 = average, 5 = high). There was a wide range in critique input when analyzed by question and by course, as shown by the High and Low scores. When all questions were summarized by course, the overall evaluation of each course ranged from a low of 3.36 for course F-15/8 to a high of 4.43 for course F-16/7, with a mean of 3.97 and a median of 3.95. The lowest evaluation on 10 of 16 questions and the lowest overall evaluation came from course F-15/8 -- which had the highest knowledge gains recorded.

Note the overall scores of 3.76 on Question 1 and 3.94 on Question 7, yet overall scores of 4.11 on Question 15 and 4.19 on Question 19. Though the students appeared to think that the courses did not prepare them for actual job performance as well as they could have (though 3.76 on a 5-point scale is still quite positive), they felt that they were relatively better prepared for the progress checks (3.94). Further, the courses were rated both very beneficial personally (4.11) and very valuable for further review (4.19). Perhaps this disparity between perceived value and the lower score on preparation for job performance can be attributed to a desire for more practice on more problems.

Table 5. Critique Results From Small-Group Tryouts (Air Force-provided questions, 5-point Likert scale)

Question:	High	Low	Avg.
1. How well did the course prepare you for job tasks taught in the course?	4.54	3.08•	3.76
2. How well did the course hold your attention?	4.56	3.33•	4.01
3. How well is the course divided into "bite-sized" segments of instruction?	4.80	3.17•	3.99
4. How easy is it to exit and return to the course?	4.77	3.63	4.13
5. How clear were the directions for using the course materials?	4.67	3.42•	4.13
6. How well did the pretest results allow you to bypass parts of the course you already knew and did not need?	4.33	3.00•	3.79
7. How well did the course prepare you for the progress check?	4.62	3.25•	3.94
8. How well did the progress check test what you learned in the course?	4.62	3.25•	4.00
9. How clear were the test questions?	4.15	3.25•	3.70
10. How adequate (free from noise and interruptions) is the interactive courseware workstation?	4.33	3.09	3.93
11. How well did the course encourage use of appropriate T.O.s or other technical data?	4.54	2.71	3.86
12. How easy was it to get a copy of the technical data needed to take the course?	4.62	3.25•	4.16
13. How available was the course for use during your duty hours?	4.60	3.13	3.84
14. How well did the course hardware and software operate?	4.70	3.25•	4.12
15. *How beneficial was the course to you personally?*	4.85	3.40	4.11
16. How helpful do you think it might be to have this course available for your review at some future date?	4.92	3.53	4.19
		TOT	3.97

• From course 15/8, whose knowledge increase was the highest recorded (41%).

CONCLUSIONS AND RECOMMENDATIONS

Methodology Issues. The MCTP effort validates the widespread utility of the PARI process, especially for less experienced interviewers. The author can recommend the PARI process without reservation for those instructional simulations where the learners' actions are the key input to the outcome -- as opposed to a process simulation, for example. [See Alessi and Trollip (1991) for an excellent description of the types and uses of instructional simulations.]

The validation of tests and the analysis of results prior to designing and developing the instruction helped *significantly in focusing the training on the most needed* areas. Further, it also provided considerable data that was incorporated directly in the course design. While most instructional design models call for developing the tests prior to designing the instruction, none (to the author's knowledge) describe the benefits of actually administering it and analyzing the results. This practice should be considered for more widespread use, since it does not take that much additional effort yet it yields significantly valuable data.

With performance improvements more than three times the size of the knowledge gains, the data would suggest that the courses are quite effective in teaching students *how to apply* their knowledge to complex troubleshooting problems. The data would also suggest that while essential for measuring troubleshooting skills, simulations are not the *only means* of teaching troubleshooting. When combined with carefully designed tutorials, a more cost-effective mix of instructional strategies may be possible. Finally, the data would support the proposition that mental models *can effectively be taught directly* rather than through inference or self-discovery. More research and evaluation of this issue is definitely *warranted, given the development cost of simulations as* compared to tutorials. A later enhancement of this simulation development methodology used for a commercial client (troubleshooting electrical problems on diesel/electric locomotives) was to add a "Rationale" *section to the summaries presented after each* simulation. This section grouped the actual steps performed in a troubleshooting sequence into a higher level of "key tests or checks," explaining the experts' rationale for performing them. (Often, it requires several distinct steps to perform one complete test in order to split the problem-space.) This enhancement appears to reduce the amount of practice needed for the novice to understand and start modeling the experts' performance, yet this was not a controlled study so any conclusions would be premature.

The data would also suggest that psychological (i.e. cognitive) fidelity is the key component of simulation fidelity for complex maintenance tasks rather than physical or functional device fidelity. Note that these courses were developed on an 80286 microcomputer, 640 kB RAM, *dual floppy drives (no hard disk)*, CGA graphics, and an interactive videodisc player. Physical realism was limited to visuals of actual equipment, and these were used primarily to depict the results (and not the possible actions) as mentioned previously. There *was no true physical fidelity. Functional device fidelity* was also quite limited. Care was taken to depict accurate results for any of the action choices provided. However, these action choices were limited to the actions performed by either novices or experts during the cognitive task analysis -- likely actions, not all possible actions, were simulated. This resulted in a *much simpler simulation model* than exists in either computer-driven part-task trainers or intelligent tutors that are used to present the same (or similar) content. This impacts the design, development, delivery, and maintenance costs of the simulations. While further research is definitely needed, this data would suggest that significant increases in the cost-effectiveness throughout the life cycle of maintenance troubleshooting simulations are possible by using cognitive task analysis techniques to build high psychological fidelity.

Research Issues. The author would echo the opinions of Lane and Alluisi that discussions of fidelity are confounded by the terminology: "...unless we add a great many additional modifiers, the term fidelity is so general as to be almost meaningless in discussing simulations" (1992, p5), and further that paying the high *price of high fidelity does not ensure training* effectiveness: "...all the fidelity you can afford may be *too much* (their emphasis) for optimum training (1992, p10)." The author would also contend that the issue of fidelity is further compounded by the difference in maintenance versus operator tasks. *Most published research seems to apply to operator training, yet many seem to apply it directly to maintenance.* Most maintenance tasks do not seem to require the precise psychomotor reactions, the constant monitoring of a myriad of visual, auditory, or tactile cues, the real-time event simulation, nor the potentially unlimited emergency conditions that are common in operator tasks.

The author would encourage more specific research on instructional simulations to address the following questions:

1. "How much simulation is enough" to reach the *most cost-effective mix* of instructional strategies? Under what conditions and for what audience? While simulation appears essential for some things, it is unquestionably the most expensive instructional strategy being widely used in interactive courseware. When must it be used and what alternative strategies are both efficient and cost-effective? How much practice is optimum for cost-effectiveness, both in terms of actual learning and to provide sufficient student confidence to facilitate transfer to the job environment?

2. How much fidelity (and of what type) is needed for the *most cost-effective development of simulations?* For what subject matter and audience? Just as in "hardware simulators," the fidelity of simulations in interactive courseware is the primary *determinant of cost. Again, how much is enough?*

3. What are the quantifiable differences (if any) between the requirements for *operations versus maintenance simulations?* How do they impact fidelity requirements? As discussed above, maintenance and operator tasks seem to differ substantially in some of the key areas related to the fidelity of simulations. For example, maintenance troubleshooting simulations seem to have more in common with medical diagnostic simulations than with flight simulations.

4. What are the quantifiable differences (if any) in fidelity requirements between maintenance *initial skills training and advanced troubleshooting training*: At least in the key areas of physical, psychological, and device/functional fidelity? It appears that high physical fidelity and low functional fidelity is needed for initial skills training, while high psychological fidelity is necessary for advanced training. Building either unneeded functional fidelity into initial skills training systems or physical fidelity into advanced training systems significantly increases both complexity and cost. How much is truly needed?

Management Issues. In conclusion, the author *would strongly recommend that the data from* implementation of this courseware be promptly gathered, analyzed, and compared with CAMS maintenance data, then widely disseminated. In addition to more insight *into the instructional design and training effectiveness* issues above, the cost-effectiveness data could be of significant benefit for other potential courseware programs, as follows:

1. Reductions in Maintenance Manhours or Spare Parts. These two areas were originally targeted as the potential payback for this training investment in the PIF package (as discussed earlier). Most other investments in interactive courseware use reductions in training time, instructor salaries, training facilities, or travel costs as the potential offsets. Instead, the MCTP attempts to tie the return on training investment back to operations and maintenance costs, a much larger potential return on investment (potentially more than a 100:1 payback ratio) and an area directly related to mission capability. If successful, this attempt could provide both a significant, large-scale precedent and a proven methodology in how to do so -- allowing others to use similar training investment strategies.

2. *Reductions in Depot-Level Repairs. Aircraft* wing commanders must now pay for depot-level repair of ALL components processed by a maintenance depot, whether they are defective or not. This was not the case when the MCTP was originally proposed and funded. Should the data confirm that this troubleshooting training decreases those depot repair costs (as it should, since it should reduce the number of components removed erroneously), the local wing commander can have more budget flexibility and will likely be a stronger proponent of training. "Two-level maintenance" (as described earlier) will likely increase the cost of depot-level repairs (and the accompanying transportation costs) otherwise, since there will be no local "check" to prevent components being sent to the depots unnecessarily.

One of the persistent challenges of training efforts has been to show a direct relationship of training results to either mission capability or local management initiatives. The results of training and the accompanying impact on performance have often been either intangible or difficult to measure. The MCTP's automated analysis programs and the data that they generate could demonstrate a methodology for providing the local commander (and higher levels of management) quantifiable results (in both time and money savings) in direct relationship to *their emphasis on and support of training.*

Based on the preliminary results of this MCTP project and the author's experience, the answers to some of these issues may not be as obvious as they seem. Meanwhile, those of us frequently tasked to "transition research into reality" sure need to know...and to have the data to support and justify our program management actions or instructional design decisions.

REFERENCES

Alessi S.M. and Trollip, S.R. (1991). Computer-based instruction: Methods and development. Englewood Cliffs NJ: Prentice-Hall.

Fletcher, J.D. (1990). Effectiveness and cost of interactive videodisc instruction in defense training and education. IDA paper P-2372, prepared for Department of Defense. Alexandria VA: Institute for Defense Analyses.

Hall, E.M., Gott, S.P. and Pokorny, R.A.. (1990). A procedural guide to cognitive task analysis: The PARI methodology. Draft AFHRL Technical Paper 90-XX, Brooks AFB TX: Air Force Human Resources Laboratory, Manpower and Personnel Division.

Gott, S.P. (1989). Apprenticeship instruction for real-world tasks: The coordination of procedures, mental models, and strategies. In E.Z. Rothkoph (Ed), Review of Research in Education, Vol 15, 97-169. Washington DC: American Educational Research Association.

Hannafin, M.J. and Rieber, L.P. (1990). Psychological foundations of instructional design for emerging computer-based instructional technologies: Part I. Educational Technology Research & Development, 37(2), 91-114.

Lane N.E. and Alluisi, E.A. (1992). Fidelity and validity in distributed interactive simulation: Questions and answers. IDA Document D-1066, prepared for Department of Defense. Alexandria VA: Institute for Defense Analyses.

Lesgold, A.M., Lajoie, S.P., Bunzo, M., & Eggan, G.M. (1988). Sherlock: A coached environment for an electronics troubleshooting job. Paper presented at the J.S. Macdonald Foundation Conference on CAI and ITSs, Carnegie-Mellon University, Pittsburgh, PA.

Thomas, T.K. (1987). The use of interactive videodisc training systems for tactical aircraft maintenance training. Proceedings of the Fifth Annual Technology and Innovations in Training and Education (TITE) Conference. Biloxi MS: U.S. Air Force.

Wilson, J.R. and Rutherford, A. (1989). Mental models: Theory and application in human factors. Human Factors, 31(6), 617-634.