

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/243764432>

# Apprenticeship Training in the Workplace: Computer-Coached Practice Environment as a New Form of Apprenticeship

Chapter · January 1989

CITATIONS

133

READS

321

2 authors:



[Susanne P. Lajoie](#)

McGill University

154 PUBLICATIONS 2,212 CITATIONS

SEE PROFILE



[Alan Lesgold](#)

University of Pittsburgh

40 PUBLICATIONS 937 CITATIONS

SEE PROFILE

## Chapter 1

# Apprenticeship Training in the Workplace: Computer-Coached Practice Environment as a New Form of Apprenticeship

SUSANNE P. LAJOIE  
ALAN LESGOLD

Learning Research and Development Center  
University of Pittsburgh  
Pittsburgh, Pennsylvania 15260

**Abstract** *Air Force technicians who practice with SHERLOCK, a computer-based coached practice environment, show marked improvement in difficult troubleshooting skills. SHERLOCK's strategy is to provide holistic practice in a realistic context, supported by tailored coaching on request. This article compares the SHERLOCK approach to other recent cognitive apprenticeship proposals.*

It is widely recognized that many people in our society fail to acquire adequate skills from their schooling. A less-recognized problem is the problem of "inert knowledge."<sup>1</sup> Time after time it has been demonstrated that students who are successful in school—who have demonstrated on tests that they know schooled facts and concepts—nevertheless fail to use these facts and concepts on appropriate occasions [1, 2].

Technical training, even though it is intended to enable skillful practice in a particular job, is not exempt from this problem. We developed SHERLOCK, a computer-based coached practice environment, as a new approach to training specific cognitive skills. SHERLOCK is an intelligent coached practice environment. It has much in common with intelligent tutoring systems, but is not as much driven by a dynamic student model. Also, its intelligence is used primarily to respond to student requests rather than to intervene actively. Extensive field testing shows that SHERLOCK works. Air Force trainees who used SHERLOCK learned the difficult skills of troubleshooting the electronic equipment they use to make diagnoses in faulty devices—to diagnose the diagnoser [3, 4]. In fact, after less experienced technicians practiced on SHERLOCK for about twenty to twenty-five hours, they compared in their ability to troubleshoot test station failures with colleagues who have four more years of on-the-job experience [4]. It seems worthwhile to reflect on SHERLOCK as a "success model" within which we can identify effective strategies for tackling difficult instructional problems. For example:

- How can we motivate the learning typically conducted in schools?
- How can we ensure that schooled knowledge will be usable rather than "inert?"
- How can we enable skillful performance in complex technical jobs?

Present address for Susanne P. Lajoie is Applied Cognitive Science Laboratory, McGill University, 3700 McTravish Street, Montreal, Quebec H3A 1Y2, Canada.

In this article, we consider what SHERLOCK can tell us about one particular proposal for improving the instruction of cognitive skills, "cognitive apprenticeship" [5, 6, 7]. Collins and Brown and their colleagues have proposed a "new apprenticeship" as a model for the instruction of cognitive skills. In their view, such a model enables students—the apprentices—to use their school knowledge in the world because it teaches knowledge *in the context of its use*. They identify *modeling*, *coaching*, and *fading* as the essential activities of masters in relation to apprentices. In apprenticeship, the learners perform with support (coaching) that is gradually withdrawn (fading). Cognitive processes can be modeled for students just as physical skills like tailoring and weaving are modeled for apprentices. One complexity of cognitive apprenticeship, however, is that the components of skill are often exercised only mentally and thus are less observable than physical skills. In a traditional apprenticeship setting, many of the skills required for performing a task are physical, such as sewing a hem. In writing, reading, or diagnosing faulty equipment, the skills and procedures may take place "in the head;" they are not as easy to observe [7, 8]. Ways of externalizing cognitive processes, therefore, are essential. Finally, apprenticeship learning is "situated;" it is embedded in a context of activity rather than being taught as abstract knowledge to be applied later.

Collins et al. identified three "success models" of teaching cognitive processes: one for teaching the processes of comprehending texts, one for writing, and one for solving mathematics problems [6, 9, 10, 11]. These success models emphasize the role of the teacher as an expert practitioner who can model skilled performance for the students. Brown et al. developed the theme of cognitive apprenticeship with a somewhat different emphasis [5]. They regard knowledge as fundamentally situated in the context of a specific activity within a specific culture. They suggest a theory of "situated learning" in which knowledge is socially constructed.

If "cognitive apprenticeship" is to be a useful model, we must identify which of its features are essential to successful learning. For example, among the features of cognitive apprenticeship as it is described and exemplified are the following:

- Learning is situated in a social context similar to those in which the skills will be used;
- Both novice and master are active participants in the learning environment;
- Cognitive processes are externalized and displayed for inspection.

Further, the success models have an element of role reversal, in which novices both evaluate others' performances and are themselves evaluated by others. This role reversal might promote self-esteem and the ability to tolerate feedback and to learn from it. Palincsar and Brown used this role reversal concept in their producer/critic paradigm, called reciprocal teaching, in which students take turns either producing an answer or critiquing a peer's solution [9]. Scardamalia and Bereiter fostered collaborative efforts in which students critiqued both the teacher's and other student's work [10]. Schoenfeld had students propose math problems for the teacher to solve impromptu [11]. The teacher's sometimes floundering efforts modeled realistic problem solving, with its false paths and revisions. When students take on the role of the teacher, evaluating other students, they can come to understand the levels of competence that can, with practice, be reached. SHERLOCK has modest capability to support comparisons of performances, and more such capability is being added in the new version now being prepared.

The presence of students at various stages of performance can support the sense of learning as an incremental process and the externalization of expert and novice thought processes. In the "success models," the environment of the classroom supported these

processes  
for produ  
other wh  
will be u  
be preser  
of more  
others at  
an incre  
This situ  
the need

Gran  
ments of  
before th  
SHERLOC  
skills. A  
will have  
computer  
every stu  
demand

Tradi  
learning,  
compass  
vary in t  
what is e  
student d  
ery appr

But l  
others. I  
could be  
do not a  
reasons,  
that are  
and com  
tively, c  
are at v  
needs as  
This puts

Altho  
fore they  
his trade  
[15]. M  
goals of  
provide  
not to c  
goals.

The  
compon  
assistanc  
the conte

processes. An ideal apprenticeship learning situation might have only limited pressure for product output, allowing trainees to work at their own pace, to provide help to each other when necessary, and to learn skills in the social and practical context in which they will be used. Ideally, a group of apprentices, each at different stages of expertise, would be present. Such a group enables the apprentice to observe the problem-solving process of more capable peers and to compare his or her own strategy and tactics with those of others at different stages of competence. The apprentice is encouraged to see learning as an incrementally staged process and has benchmarks for his or her own progress [6]. This situation is generally not fully practical; real-world apprenticeship is constrained by the need for efficient production.

Granting, even, that cognitive apprenticeship can be successful, there are still elements of the approach that need to be tested more completely and stated more explicitly before the approach will generalize across subject matters, teachers, and students. SHERLOCK can be viewed as a formalization of apprenticeship training for complex skills. As a computer tool, it forced us to specify the principles it embodies: those that will have to be taught to teachers if they are to become cognitive "masters." And such computer tools as SHERLOCK may supplement a classroom teacher, who cannot coach every student at once. In its current form the cognitive apprenticeship approach may demand too much of the teacher's monitoring skills and intuitions.

Traditional apprenticeship training was relatively unconcerned with accelerating learning, since the apprenticeship performed useful work. Apprenticeship training encompasses years. Learning in such situations is often slow and unfocused. Masters may vary in their experience as teachers, and novices may vary in their ability to learn in what is essentially a discovery learning environment. Apprenticeship learning is largely student driven. This is good when it suits the student's learning style and when a discovery approach to knowledge acquisition is feasible in the work environment.

But learning is facilitated by a discovery approach in some situations and not in others. In a complex domain, "learning by discovery" may take years, even though skill could be fostered earlier by focused instruction [12, 13, 14]. It is likely, too, that masters do not always recognize the current performance level of their apprentices. For these reasons, and for economic reasons, apprentices may be performing fragmentary tasks that are below their capability even after they are capable of performing more complex and complete tasks, alone or with assistance. Novices must observe the master attentively, comparing their skills to the master or other apprentices in the setting who are at varying levels of competence. The master must recognize when the novice needs assistance (coaching) and decide when assistance is no longer needed (fading). This puts a heavy burden on both learner and teacher—more than some can handle well.

Although novices are afforded the opportunity to observe skilled performances before they are expected to enact them, the traditional master's main agenda is to practice his trade or profession—to produce a product rather than to teach a novice new skills [15]. Much of the instruction then is modulated by the economic or product-oriented goals of the master's shop rather than by any search for the work situations that could provide the most instructional value, in terms of guidelines on when to coach and when not to coach, and undercontrolled, with educational goals subordinate to production goals.

The cognitive apprenticeship approach to instruction shares several instructional components with traditional apprenticeship approaches: observation, coaching, fading of assistance, shared problem solving between teacher and student, and situated learning in the context of subsequent knowledge use. Where cognitive apprenticeship approaches go

beyond the traditional approach is in developing ways to facilitate the externalization of expert problem solving activity and the student's skills of learning. But does cognitive apprenticeship depend on the special qualities of exceptional master teachers?

Like expert cognition, expert teaching is often inaccessible to study. SHERLOCK can be viewed as an externalization of expertise: it is specified by rules and is replayable. It can be studied. There can then be debate about whether and to what extent it embodies the approaches labeled cognitive apprenticeship. Furthermore, to the extent that it is effective, SHERLOCK can provide information about the efficacy of those approaches. Specifically, we believe that SHERLOCK provides an opportunity to answer questions such as the following:

*When should coaching be provided and when should it be faded?* The expert teacher probably depends on intuition. SHERLOCK uses explicit models of the student's competence to drive the coaching and fading of feedback.

*How can coaching be individualized for a student?* The expert teacher may deal with many students at once, or at best rely on a general sense of a student's ability. SHERLOCK gives different levels of help according to the student's current state of achievement.

Experience with SHERLOCK alone cannot directly address questions about the social environment of learning. For example, what principles should guide students' evaluation of their peers? How should a cooperative environment be maintained (if it should)? In field testing, however, SHERLOCK was embedded in a social situation as well as a context of application. In discussing possible future developments of SHERLOCK, we suggest means by which SHERLOCK can be used to foster such aspects of socially situated learning as peer tutoring and role reversal.

### Computer-Coached Practice Environments

SHERLOCK is a specific example of a form of job-situated training that we call a *computer-coached practice environment*. It was developed for the Air Force as a research vehicle, to allow us to address the issues raised above. In the remainder of this article, we discuss why such an approach was needed, how the approach was developed, and how well it worked.

#### Background

SHERLOCK was developed for a particular job in the Air Force, that of F-15 manual avionics technician. People in this job repair electronic navigation equipment from F-15 aircraft, using a *test station* to isolate the sources of failures. With this test station, the diagnosis task is relatively easy, and trainees have no difficulty learning the basic job from extant "schoolhouse" instruction. However, the test station itself can fail. Indeed, it fails relatively often. It is composed of massive amounts (40 to 100 ft.<sup>3</sup>) of discrete electronic components soldered to printed circuit cards. Furthermore, the special tools for diagnosing its failures are minimal (there are confidence tests that can be run on the test station, but they tend not to isolate faults very completely or efficiently). So while the everyday job of these technicians is to use good tools to make simple diagnoses, their occasional job is to use minimal tools to make complex diagnoses. It was this second, more difficult, chore that we set out to teach with SHERLOCK. SHERLOCK is de-

signed for  
tice in the  
shooting t  
isolation p

The tes  
of the train  
test station  
There is li  
least) is a  
troublesho  
the electro  
cal approa  
than thirty  
technical p  
greater wa  
tional conc  
job learnin  
experience

The el  
does not s  
technician  
the test sta  
that novice  
navigation  
shop could  
peers solv  
process an  
sion. Regi  
mons help

Beclus  
work. In p  
it. Often,  
time, so th  
practice th  
cannot be  
relatively  
tionally op  
(4) inexpe  
(5) the go  
Even if th  
would stil  
invisible b  
cognitive  
learned an

What is a  
opportunit  
supplied s  
computer

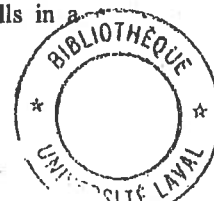
signed for airmen who have just completed technical school and have started to apprentice in their assigned F-15 manual avionics shops. It teaches trainees skill in troubleshooting the manual avionics test station by coaching them through realistic fault isolation problems.

The technical training for the manual avionics job is modeled after schooling. Much of the training is didactic, and the rest involves following specific directions in using the test station, much as a student in a high school science lab might follow directions. There is little direct instruction in problem solving, even though fixing the test station (at least) is a problem solving task. Basically, technical schools like those for this job teach troubleshooting by focusing on formal principles such as the electrical laws that underlie the electronics instrumentation domain. There is evidence, however, that such a theoretical approach fails to produce good troubleshooters. Morris and Rouse reviewed more than thirty training studies and concluded that the closer the training was to actual technical performance in terms of task-related specificity, completeness, and realism, the greater was the training effectiveness [16]. As Gott and Davis suggest, teaching functional concepts of electronics systems, grounded in experience, might facilitate on-the-job learning in avionics better than teaching theoretical principles that are not situated in experience [17].

The electronics troubleshooting training that technicians receive in the Air Force does not stop at technical school. The purpose of preliminary schooling is to make the technician able to do useful work on the job. The harder parts of the job, such as fixing the test station, are meant to be acquired through on-the-job experience. It is on the job that novices have some opportunity to observe experts diagnose and repair faults in the navigation equipment or the test stations. Thus, in a sense, the F-15 manual avionics shop could be a natural apprenticeship setting, where novices observe as more capable peers solve troubleshooting problems, developing mental models of the troubleshooting process and later exercising those models by attempting harder diagnoses under supervision. Regrettably, though, the organizational rules make this a solitary job. One summons help when necessary, but one usually works alone.

Because there is much work to be done, there is little time for watching experts work. In particular, when the test station breaks, an expert is usually assigned to repair it. Often, the novice and journeyman technicians have work at other test stations at this time, so they cannot observe the expert performance. There are also few opportunities to practice the complex skills of test station diagnosis. Troubleshooting the test station cannot be practiced efficiently in the shop, because (1) failures of the test station are relatively rare; (2) failures of the test station are diverse and do not occur in an instructionally optimal sequence; (3) inexpert troubleshooting of the test station is often unsafe; (4) inexpert troubleshooting of the test station is time consuming; and, more generally, (5) the goals of training conflict with the goal of repairing test station faults quickly. Even if the conditions were otherwise ideal for apprenticeship, the on-the-job learner would still face the difficulty that many of the skills and procedures of the jobs are invisible because they take place "in the head." SHERLOCK was developed to make the cognitive processes of test station troubleshooting more observable so they could be learned and practiced.

*What is a Computer-Coached Practice Environment?* A practice environment provides opportunities for practicing cognitive activity in a realistic setting, with computer-supplied support when needed. The practice environment in SHERLOCK is a realistic computer simulation of a job environment; trainees acquire and practice skills in a



context similar to those in which they will be used. So, for example, SHERLOCK's simulated work environment matches that of the real job. Each control and display panel of the test station, with knobs and dials, is accurately simulated. The schematic diagrams, instructions for testing units from aircraft, and other documentation, while slightly simplified, retain many aspects of those used in the real job: the symbols in diagrams are the same, the layout of testing algorithms is the same, the indexing schemes are the same, etc. We do not believe that it is critical to have the simulation entirely based in computer graphics; in fact, the next version of SHERLOCK will instead use interactive video for part of the simulated work environment. What is critical, we believe, is the combination of intelligent coaching with realistic simulation of the actual work environment.

*How is SHERLOCK Different from Other Cognitive Apprenticeship Settings?* SHERLOCK's fundamental purpose is to provide an efficient and realistic, but artificial, means of practicing, with support and feedback, a skill that cannot be efficiently practiced in its real application context. There are several factors that make practice in a computer-coached environment more effective than either ordinary on-the-job experience or the classroom approach to cognitive apprenticeship. A human coach could also provide efficient practice, but adequate coaches and work environments are often not available.

What differentiates a computer-coached practice environment from shop experience is efficiency and focus. In the shop, a technician sees only the equipment failures that happen to occur while he is there, and often he cannot participate in the most useful troubleshooting cases. A computer-coached practice environment can quickly and safely give trainees experience with sequences of fault diagnosis tasks selected for instructional purposes. In the shop, trainees receive advice of varying quality when they need help solving a problem. A computer-coached practice environment can provide instruction when the trainee needs it and can tailor the help provided so that outside help is not an unnecessary crutch. In the shop, complex troubleshooting problems may take as long as two shifts (sixteen hours); airmen on the second shift may have to begin troubleshooting where the previous shift stopped. Problems that take twelve hours on the job can be completed in less than a hour on SHERLOCK. In the shop, time is spent waiting for parts, dismantling components, and carrying out long sequences of performances that are already well learned. Using SHERLOCK, all the activity is focused on the problem solving that is to be learned. SHERLOCK can ensure "coverage" of the set of instructive experiences as well as provide appropriate levels of support for those trainees who need assistance. The practice environment preserves the full context of what technicians experience in the shop, presenting a problem-oriented interface along with support when needed.

SHERLOCK shares with the cognitive apprenticeship models that we reviewed the principles of observation, practice, coaching, fading, and externalization of problem-solving processes. Similarly, SHERLOCK shares with other approaches a basis in extensive knowledge of the subject matter and of common individual differences in subject-matter knowledge. One advantage, however, of a computer-coached practice environment over other forms of cognitive apprenticeship is that a computer coach provides support or coaching based on a detailed picture of the trainee's performance, which many human teachers lack. SHERLOCK models performance and competence dynamically and uses the models to determine the level of assistance that the trainee needs. SHERLOCK also features a strong link to empirical research, extensive involve-

ment of  
instructio

### Empirica

SHERLO  
proficien  
15 manu  
identified  
shooters  
proprie  
differ fro  
comprehe  
problem.

### A Close

### What Do

The real  
drawers,  
called the  
is itself a  
front pan  
of the SH  
drawers;  
package  
be expan  
show its  
circuit di  
and indic  
ment ins  
using SH  
SHERLC  
since fin  
have.

SHEF  
the prop  
test static  
compon  
stations a  
generic,  
tronic sy  
station co  
accuratel  
lems that



ment of domain experts in developing instruction, a pedagogical approach of holistic instruction of the problem-solving activity, and the use of the computer as coach.

### ***Empirical Research Behind SHERLOCK***

SHERLOCK's instructional goals reflect the cognitive components of troubleshooting proficiency identified in empirical research. Extensive cognitive task analyses of the F-15 manual avionics test station job compared more skilled with less skilled airmen and identified several skill components that differentiated them [3, 18]. Proficient troubleshooters were skilled at generating and testing fault isolation plans, and at using appropriate methods and strategies to confirm their hypotheses. They did not, however, differ from less skilled troubleshooters on isolated skill components such as logic gate comprehension. The skill differences emerged only in the context of a troubleshooting problem.

### **A Closer Look at SHERLOCK**

#### ***What Does SHERLOCK Look Like?***

The real work environment consists of (1) the test station, which is divided into twelve drawers, (2) whichever unit from the airplane is being tested, and (c) a cabling package called the *test package* which connects the unit to the station. Each drawer of the station is itself a large entity, containing a number of printed circuit cards and having a complex front panel of knobs, switches, and indicator displays. Figure 1 shows a wide-angle view of the SHERLOCK work environment. It depicts the test station, including its twelve drawers; the specific navigation unit from the plane that is being tested; and the test package which interconnects the station with that unit. Each of the drawer displays can be expanded (by pointing to the drawer with a mouse and pressing the mouse button) to show its front display and control panel.<sup>2</sup> Tests can be performed by retrieving schematic circuit diagrams from a data base (by pointing to entries in realistic tables of contents and indices) and then indicating on the diagrams where the test probes of measurement instruments should be placed. The trainee's basic job, in real life and while using SHERLOCK, is to isolate where the faults are in the airplane units. However, SHERLOCK is designed to simulate a high rate of test station and test package failures, since finding faults in these devices is the specific problem airmen in the job seem to have.

SHERLOCK provides a simulation of a real-world apprenticeship setting, where all the proper tools and equipment are available for efficient troubleshooting. Although the test station as simulated by SHERLOCK is reduced in complexity, all of the critical components contained in the four different types of Air Force manual avionics test stations are represented. To some extent, the test station that SHERLOCK simulates is generic, an abstraction of those used by the Air Force for maintenance of various electronic systems for several different types of military aircraft. Since the critical test station components, technical manuals, schematics, and checkout technical orders are accurately portrayed by SHERLOCK, it has the ability to replicate fault isolation problems that actually occur with real test stations.





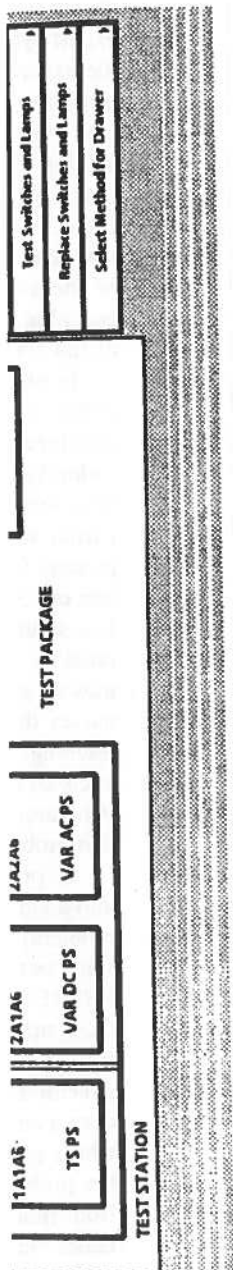
**SHERLOCK'S Domain Expertise**

One way to develop computer-based instruction is to embody domain expertise in an expert system and to correct or instruct the student whenever his activity departs from the expert ideal. Both empirical research and expert advice suggested that such an approach would both misrepresent troubleshooting expertise and overly constrain the trainees' problem-solving activity. Often, there are multiple "best" paths or sequences for solving a troubleshooting problem. SHERLOCK's basic mode of interaction with trainees is to pose realistic troubleshooting problems and to provide expert advice on demand. Accordingly, the knowledge in SHERLOCK is organized around problems, problem solution methods, and the constraints imposed on problem solution by the structure of the work environment. Domain expertise in SHERLOCK is reflected both in its coaching and in its work environment, so skills are learned in an accurate representation of the context in which they will be used.

In order to discuss SHERLOCK's domain knowledge more completely, we need to define some terms.<sup>3</sup> Much of SHERLOCK's knowledge is represented in a separate structure specialized to each problem that it presents. Problem-based knowledge organization is efficient because it is based on what we call the *effective problem space*. The term *problem space* is often used to refer to a graph structure in which nodes represent partial solution states for problem and links represent actions that move the solution process from one partial solution state to another [19]. In the course of solving a problem one "traverses" the problem space via a path that represents the sequence of problem solution steps taken. For a test station diagnosis problem, there are thousands of possible moves. We did not represent all these moves. Rather, we considered only moves, good or poor, that are likely to be made by experts or trainees, plus additional moves that are infrequent but consistent with the troubleshooting approaches we are teaching. Rare and bizarre moves were omitted, since we probably would not know how to coach them anyway, except through generic techniques of reviewing the problem goal structure. It is the remaining critical subset of the problem space that we call the *effective problem space*.

To provide further structure, we grouped the various pieces of problem space paths into a hierarchy of larger units which we called subgoals (the overall goal is to solve the problem). Each subgoal is a piece of problem-solving procedure. Some are strategic, some tactical. Some involve heuristic plans; others involve fixed procedures. The actual SHERLOCK knowledge structure for a problem is a network in which the nodes are these subgoals rather than the more microscopic actions which they summarize. That is, most subgoals subsume a lot of mental activity; they are *abstracted* problem subgoal structures. For some problems, a number of different sequences of subgoals will produce a solution. For such problems, we can picture the subgoals as forming a lattice in which each specific solution approach is represented by a subset tree. The top node of the problem subgoal lattice represents the complete navigation of the problem space from first attack on the problem to final solution. Branches underneath any node in the lattice refer to the subgoals that must be achieved before that higher-level node can be achieved. Higher-level nodes generally reflect problem solution strategies, while lower-level nodes represent tactics and specific actions. SHERLOCK's representation of a problem, then, consists of an abstracted subgoal lattice for the problem's effective problem space (EPS), which we abbreviate as an *EPS subgoal structure*. These problem representations have fifty to sixty nodes, rather than the thousands of nodes that a "complete" problem subgoal tree for test station diagnosis would have.

Figure 1 The SHERLOCK work environment.



The EPS subgoal structures are represented as *objects* in an object-oriented programming language. That is, for each node of the EPS subgoal structure, there is a specialist miniprogram, or object. In our *object-oriented programming* approach, a small number of generic template objects are first defined, and then each node's "personal" object is specified as an instance of the generic template. Most of the unique information in such an instance object consists of data, not program, enabling substantial use of nonprogrammers for developing such a system. In the case of SHERLOCK, the data specify hinting information, test point values, etc. Even though there are many different test measurements that a trainee might make on a component while diagnosing a test station failure, only the abstracted notion of testing the component is represented in an EPS subgoal node, along with a procedure to determine whether the appropriate testing has been done. The ability to provide test values in response to a measurement action is part of the device simulation, not the problem solution process, so it is represented separately in the model of the device.

### *SHERLOCK's Pedagogical Framework*

We concluded from the empirical research on troubleshooting that it would be best to train troubleshooting competence holistically. With SHERLOCK, the trainee learns to do complex procedures by doing them. When he cannot handle the whole performance, he is given reminders and hints to support him through the parts he cannot handle, but at the same time he is still enacting the whole target cognitive performance. Instead of teaching separate didactic lessons, we situated our coaching of specific cognitive components in the contexts of specific problem situations in which they should be used. This facilitates mapping new knowledge to appropriate situations. Rather than teach any one curriculum goal in isolation, each goal is addressed whenever it is relevant to overall problem-solving performance. More specifically, SHERLOCK talks only when the trainee is likely to listen—when he needs to know the information SHERLOCK is presenting. Our pedagogy parallels the cognitive apprenticeship approach in providing holistic instruction. However, instead of decomposing the troubleshooting activity into processes each of which needs to be externalized for trainees, SHERLOCK provides coaching on specific skills when trainees ask for help and demonstrate through their performance that assistance is needed for strengthening those skills.

### *Coaching*

Our principal expert, Gary Eggan, had worked in the job as an enlisted airman and later become a civilian technical advisor to airmen doing the job. Thus, he was expert both in doing the job and in helping others learn to do it. As an articulate expert, he provided us with a rationale for problem selection and sequencing, with the coaching content, and with clear representations of the effective problem spaces for each problem. SHERLOCK permits a large number of different paths toward isolating the fault in each problem. The coaching emphasizes relevant work environment information first, following up next with broad strategy (the application of relevant "rules of thumb"), and providing specific tactics only if the trainee is still unable to reason them out on his or her own. Help is provided only on request or when the trainee's performance necessitates SHERLOCK's attention (for example, when an unsafe action is attempted).

Each of SHERLOCK's problems has an EPS subgoal structure built by our domain

expert. The  
Each EPS s  
lar hypothe  
relevant to  
test outcom  
on an up-t  
planned ah  
well as for

Expert  
"rules of th  
for help wi  
that are mo  
tutored by  
tion genera  
component  
volved in  
functions fr  
derived fro  
component  
tions. For i  
prior and s  
then it prol  
two steps).  
suspect.

SHERI  
solving pr  
asking que  
minimal hi  
problem-se  
first hint h  
can reflect  
trace of th  
problem se  
to the curr  
ing structu  
so that the  
node can g  
example,

Thes  
set.  
do  
thun  
Pins  
mine

The assign  
with suppl

expert. The trainee's performance is shadowed with reference to this subgoal structure. Each EPS subgoal node has one or more expert ways to confirm or disconfirm a particular hypothesis. When the trainee asks for help, SHERLOCK knows which EPS node is relevant to his needs. The help the trainee receives depends on the appropriate actions, test outcomes, conclusions, and best next moves for the EPS node where he is, and also on an up-to-date assessment of his skills. By having both expert and novice paths planned ahead of time, it is possible to specify coaching for different levels of skill as well as for the specific nodes in the problem space.

Expert mental models of the test station, in addition to informing the coaching of "rules of thumb," are also used more broadly in the hinting process. A trainee who asks for help will often get hints that direct attention to the functional areas of the test station that are most likely to be the source of the failure. For example, one of the heuristics tutored by SHERLOCK is the history-taking procedure. History taking uses the information generated by the routine actions of the test station as it systematically exercises its components. In history taking, an expert examines the devices that were actively involved in the function which is manifesting the system failure. He also considers the functions that the system has just handled correctly. Troubleshooting hypotheses are then derived from a comparison of the successful and the failure situations to see which components are needed for the faulty function but not needed in prior successful functions. For instance, if a device active in the current testing function was also active in the prior and successful testing step, and if both functions used the device in the same way, then it probably is not faulty (of course, there is a small chance that it failed between the two steps). Otherwise, as an active component in the function that failed, it is a prime suspect.

SHERLOCK is designed to offer the least hint that can enable further problem-solving progress. Much of its coaching is designed to stimulate a trainee's thinking by asking questions rather than generating answers for what the trainee should do next. A minimal hinting approach was selected in order to encourage active construction of the problem-solving knowledge by the trainee. Thus, each time a trainee asks for help, the first hint he receives is a structured rehearsal of what he has done on that problem, so he can reflect on his plans and actions. Further, when he requests additional help he gets a trace of the hints he had already received. More elaborate hints support the trainee's problem solving much as a shop supervisor might. The content of these hints is specific to the current work environment and to the state of the trainee's knowledge. Our coaching structure was developed to provide understanding of skills in many problem contexts so that the trainee can see the conditions under which a new procedure applies. Each node can give hints about the conditions under which certain things would be true. For example,

These data signals should cause relays K3 and K2 in the A1A1A13 card to set. One way to determine if the A1A1A13 card is operating properly is to do an ohms check on the relays being set by the stimuli code select thumbwheel switches. You could ohm out the lower contacts of K3 (between Pins 1 & 3) and the lower contacts of K2 (between Pins 55 & 57) to determine if the relays are being set.<sup>4</sup>

The assigned tasks, though at the limits of the trainee's capabilities, are all completed, with support from the environment available on demand.

### *Does SHERLOCK Have the Right Knowledge?*

We began the development of SHERLOCK with the belief that it was impractical to provide knowledge of troubleshooting procedures and conceptual understanding of the entire test station in complete form, although it would be desirable to do that in an intelligent instruction system. Accordingly, we developed the approach just described. More recently, we have come to realize that what SHERLOCK really needs to know about is not the entire test station but only the circuit configurations whereby the test stations accomplish various bits of work. With a sufficient collection of circuit path models for the test station tasks, it is possible for an intelligent coach to have deep knowledge of both concepts and procedures relevant to troubleshooting and to derive specific problem-solving plans from more general knowledge on demand. Thus, more explanations can be provided, and, by introducing faults into the circuit paths, problems can be generated to meet individual training needs. The plans for SHERLOCK II are not the topic of this paper, but we want to clarify that other, and perhaps more productive, designs can be developed for intelligent coached practice environments.

### *Student Modeling and Adaptive Instruction*

SHERLOCK provides all trainees with the same type of problems in the same sequence of graded difficulty, gradually building up the trainee's conceptual model of troubleshooting. Nonetheless, SHERLOCK is adaptive to individual differences, diagnosing the trainee's learning performance relative to the part of the problem he or she is currently attacking. Feedback is both expert driven, based on SHERLOCK's calculation of the student's performance model or learned state, and student driven, in that if a trainee asks for help, it is provided, with additional elaboration on demand. Because it is given sparingly, based on an assessment of the trainee, hinting also fades automatically as the trainee's skill develops.

SHERLOCK predicts and models a trainee's performance on each specific problem and also maintains a more enduring model of the trainee's emerging competence. Assessing the trainee's competence enables SHERLOCK to determine how much assistance is required when an impasse is reached. The level of hint provided to the trainee depends on SHERLOCK's assessment of how well the training goals that are immediately relevant have been attained. Pedagogically, SHERLOCK approximates apprenticeship—a novice trainee can independently practice goal-oriented whole performance with SHERLOCK providing individualized assistance when an impasse is reached.

The classroom cognitive apprenticeship approaches do not have SHERLOCK's approach to assessment and pedagogy. Although they evaluate a student's progress, they have no concrete rules for when to provide hints and when to fade assistance. This is pedagogically important, because SHERLOCK's hint structure allows us to examine emerging competence that might not be tapped if no assistance were given. Novices have "islands of knowledge" that can be explored only by providing the necessary bridges between these islands. For instance, a trainee may have a well-developed measurement-taking skill. But if he or she has not mastered the early steps of planning a strategy for finding a fault, the point at which the mastered skill can be demonstrated or practiced may never be reached. By supplying bridging knowledge, the hint structure enables us to assess information about the subject's knowledge that would be inaccessible in a testing environment that terminated a performance whenever it reached an impasse.

This approach taps what Vygotsky called a "zone of proximal development," emerg-

ing compet  
gotsky (p.  
children as

the di  
dent  
throu  
ble p

SHERLOC  
would be ju  
skills that i

### *Externaliz*

An interest  
ronments a  
practice, as  
the trainee  
based on r  
than human  
of coach ca  
can intern  
internalize  
as advice f  
tage of suc

Further  
these temp  
with the co  
actions. Tl  
moves from  
select a lo  
already sel  
tional unit  
each of the  
the unit. T  
tive tool fo  
from which

Others  
activity. R  
novices by  
a hierarchi  
physics ca  
ment that  
lems. The  
selection.  
solving the  
HAT prov  
questions



ing competence that can be assessed only by supporting it with coaching [20,21]. Vygotsky (p. 86) defined the zone of proximal development in developmentally disabled children as

the distance between the actual development level as determined by independent problem solving and the level of potential development as determined through problem solving under adult guidance or collaboration of more capable peers.

SHERLOCK's hinting structure challenges trainees with learning opportunities that would be just beyond their reach without coaching. This permits the assessment of their skills that in turn enables appropriate coaching.

#### *Externalization of the Trainee's Troubleshooting Process*

An interesting comparison can be made between computer-based coached practice environments and apprenticeship models that have human coaches.<sup>5</sup> In computer-coached practice, as in traditional apprenticeship settings, a "supervisor" is available to answer the trainee's questions. The advantage of the computer is that it can provide feedback based on more detailed and accurate individual performance and competence models than humans can readily determine or remember [24]. Salomon suggests that both kinds of coach can foster the internalization of expert skills and reflective processes. A student can internalize the cognitive skills displayed by a computer coach just as one could internalize processes modeled by a human coach. Computer tools can be internalized just as advice from "more capable peers" can be. SHERLOCK was designed to take advantage of such internalization. It provides hints that externalize troubleshooting strategies.

Furthermore, SHERLOCK provides templates for cognitive skills. One source of these templates is the interface the computer presents to the trainee. The trainee interacts with the computer largely by selecting options from a menu of troubleshooting goals and actions. The menus are hierarchical, with an underlying goal structure. The trainee moves from a menu of higher-level goals to submenus of subgoals. Thus, in order to select a local goal, like extending a particular card for testing, the trainee must have already selected a higher-level goal. Higher-level subgoals represent plans to test functional units: *Test the test station*, *Test the test package*, or *Test the airplane unit*. Within each of these major subgoals are sub-subgoals, such as *Test the circuit cards located in the unit*. The hierarchical planning structure modeled in these menus serves as a cognitive tool for structuring the problem-solving process. Figure 1 shows a menu of actions from which a subgoal has been selected and the submenu that appears in response.

Others have attempted to use menu selection to teach structured problem-solving activity. Research has been conducted in physics to promote expert-like behavior among novices by constraining them to follow an expert-like approach. Mestre et al. developed a hierarchical computer-based problem analysis environment for mechanics problems in physics called Hierarchical Analysis Tool (HAT) [25]. HAT is a menu-driven environment that constrains users to perform a top-down analysis of classical mechanics problems. The computer poses well-defined questions, and students answer them via menu selection. The first level menu consists of general principles that could be applied to solving the problems, followed by secondary menus that consist of ancillary concepts. HAT provides no feedback or tutoring, but merely constrains the type and order of questions that need to be considered when analyzing a problem. The HAT treatment

group performed better than a control group on a task that required deep structure analysis but were no more likely to apply a concept-based approach to a problem-solving task. Mestre et al. concluded that a feedback approach using HAT would be more powerful for improving problem solving. SHERLOCK incorporates hierarchical structuring and feedback and was successful. However, it is difficult to attribute SHERLOCK's success with certainty to a single instructional strategy such as hierarchical structuring, because SHERLOCK incorporates multiple instructional strategies [26, 27].

The effective problem space (EPS) subgoal structure supports another pedagogical tool. One of the purposes of the EPS, from a programming perspective, was to serve as a record-keeping device of the trainee's movements through the problem space. By monitoring the solution trace, we could summarize and display the trainee's problem-solving steps every time help was requested. From a cognitive perspective, this form of recapitulation serves as a reminder of what hypotheses the trainee generated. This reminder externalizes the trainee's thought processes and is often enough to make the trainee reflect on what parts of the problem space have been examined. Replaying the trainee's problem-solving trace is a way to encourage self-regulation of planning behavior. Repeated experience with such a tool can lead to a trainee's anticipation of the trace and eventually perhaps to self-generation of a trace.<sup>6</sup>

The same record-keeping capabilities are also used to generate hints for a special-purpose option we called *the panic button*. The panic button can be selected at any time during the course of problem solution. Trainees are told that the panic button should be selected when they are totally lost and do not know what step to take next. If a trainee reaches an impasse and selects the panic button, then SHERLOCK can replay the plan subgoals that the trainee has (directly or indirectly) completed. This externalizes the trainee's plans up to the point of impasse. In this manner, the panic button constrains the problem space for the trainee by explicitly telling the trainee what he has ruled out in the problem space and implicitly telling him what is left in the problem space. It helps constrain the problem space and forces concentration on plan components that have not yet been completed or proposed. This computer tool is something like a more capable peer telling the technician that he should perhaps try to "check out the unit" instead of "the test package" because his past actions have already eliminated the test package as a possible fault area. The panic button is more than a summary; it uses the trainee's prior actions to generate a statement that constrains the problem space.

Recapitulation and the panic button are both designed to foster hypothesis generation, self-regulation, and reflection. This instructional concept is similar to Schoenfeld's use of post mortems in mathematics [11]. Schoenfeld presents postmortems at the completion of a problem to draw attention to expert/novice differences at key decision points. The panic button is an interactive rather than retrospective tool in that it plays back the trainee's thought process at the point where an impasse is reached, rather than waiting until the student is done with the problem.<sup>7</sup> It highlights for the trainee which hypotheses were ruled out by previous actions. In addition, trainees can call up an expert's performance by asking for help and thus compare their solution to an expert model.

### Evidence that a Computer-Coached Practice Environment Works

The Air Force evaluation of SHERLOCK found that subjects who spent twenty to twenty-five hours working with SHERLOCK were as proficient in troubleshooting the

test station :  
our own evi  
groups of 2  
average of 1  
sessions we  
working da  
avionics sh  
shooting to  
failures enc  
technical or

The pre  
areas as the  
ages, test s  
in our tutor  
experience  
experiment  
the posttest  
significantl

Qualita  
group was  
emerging  
Eggen, our  
shooting st  
step reflect  
currently b  
one very i  
informatio  
determine  
tion that h  
shooting st  
resistance  
active whe  
formance  
trolling fo  
like probl  
(mean =  
moves (me

A qual  
encing a te  
middle, an  
change in  
according  
U.S. Arm  
expert dev  
tion traces  
computed  
solution s  
peated me  
time spent



test station as technicians who had been on the job four years longer [4]. We performed our own evaluation of SHERLOCK's effectiveness on matched experimental and control groups of airmen at two Air Force F-15 bases [3]. The experimental group spent an average of twenty hours working through SHERLOCK's thirty-four problems. Tutoring sessions were conducted in two- to three-hour blocks that spanned an average of twelve working days. The control group subjects went about their daily activities in the manual avionics shop. Structured interviews were administered as pre- and posttests of troubleshooting to thirty-two trainees. These pre- and posttests posed problems based on actual failures encountered in the shop, rather than on SHERLOCK problems, and Air Force technical orders were used rather than SHERLOCK's modified technical orders.

The pre- and posttests of troubleshooting proficiency targeted the same functional areas as the tutor problems, but there were some surface differences. Units, test packages, test stations, and the schematics for test station drawers were different from those in our tutor. Thus, pre- and posttests measured *near transfer* from the simulated tutoring experience to on-the-job performance. There were no significant differences between the experimental and control groups at pretesting ( $\chi^2(1, N = 63) = 0.00, p = 1.00$ ), but on the posttest performance ( $\chi^2(1, N = 62) = 10.29, p < .001$ ) the tutored group solved significantly more problems (mean = 30) than the control group (mean = 21).

Qualitative analyses of the pre- and posttest data revealed that the experimental group was significantly different from the control group on several factors that reflected emerging competence. Pre- and posttest scoring templates were developed by Gary Eggan, our domain expert. They addressed levels of competence by classifying troubleshooting steps as *expert*, *good*, *redundant*, or *inappropriate*. An expert troubleshooting step reflected a step that an expert would make at the particular point in the problem currently being addressed, such as eliminating a device as the fault source by making one very informative measurement. A good step was one that collected meaningful information but was not necessarily expert-like, such as taking three measurements to determine what an expert might establish with one. A redundant step collected information that had been determined by a previous move. An inappropriate or bad troubleshooting step was one that was totally meaningless or irrelevant, such as measuring the resistance across a power source or measuring the voltage drop in a path that was not active when the system failed. Analyses of covariance were performed examining performance differences between the two groups on posttest criterion variables while controlling for the qualitative pretest variables. The tutored group displayed more expert-like problem solving steps (mean = 19.33) in the posttest than the control group (mean = 9.06,  $F(1,27) = 28.85, p < .001$ ) and made fewer inappropriate or bad moves (mean = 1.89,  $F(1,27) = 7.54, p < .01$ ) than the control group (mean = 4.19).

A qualitative analysis was performed to see how effective SHERLOCK was in influencing a technician's overall problem-solving solutions. Three problems—one early, one middle, and one late in the tutor sequence—were scored to see whether there was a change in expertise over the course of the tutored practice. Trainees were also split according to aptitude, as measured by the standard selection and placement test of the U.S. Armed Services, the Armed Services Vocational Aptitude Battery (ASVAB). An expert developed an ideal solution path for each problem and scored the trainees' solution traces using the ideal path as a template. A deviation score from the expert path was computed for each subject by using the following formula: the subject's number of solution steps minus the number of steps that overlapped with the expert path. A repeated measures analysis of variance examined the effects of aptitude (high, low) and time spent in the tutoring environment (early, middle, and late tutor problem) on type of

ired deep structure  
o a problem-solving  
AT would be more  
hierarchical struc-  
to attribute SHER-  
uch as hierarchical  
strategies [26, 27].  
another pedagogical  
ive, was to serve as  
problem space. By  
trainee's problem-  
ective, this form of  
generated. This re-  
nough to make the  
ined. Replaying the  
of planning behav-  
icipation of the trace

hints for a special-  
selected at any time  
ic button should be  
ke next. If a trainee  
can replay the plan  
his externalizes the  
button constrains the  
has ruled out in the  
lem space. It helps  
onents that have not  
like a more capable  
the unit" instead of  
the test package as a  
as the trainee's prior

r hypothesis genera-  
nilar to Schoenfeld's  
mortems at the com-  
ces at key decision  
tool in that it plays  
reached, rather than  
or the trainee which  
nees can call up an  
solution to an expert

## It Works

who spent twenty to  
1 troubleshooting the

moves (deviation scores and total steps to solution). There was a significant main effect due to time spent in the tutoring environment ( $F(2,42) = 4.19, p < .02$ ) on the trainee's deviation scores and steps to solution. The more problems the trainee had solved using SHERLOCK, the closer the trainee's troubleshooting solution processes were to expert troubleshooting solution processes, and the fewer steps were taken to reach a solution. With practice on SHERLOCK, trainees became more expert-like and efficient. Both the high- and low-ability trainees responded in the same manner, with the low-ability individuals showing slightly more improvement than the high-ability group.

Additional data confirm that the more specific goals of SHERLOCK were achieved. The proportion of actions that involved making measurements, as opposed to swapping components, increased over tutoring sessions ( $F(2,45) = 3.30, p < .05$ ), for example. Furthermore, the effects of improved efficiency and decreases in inappropriate steps were found whether the problems involved conceptually simple or more complex failures. Overall, our data suggest that trainees acquired expertise in all phases of troubleshooting activity, including the most difficult aspects of measurement taking, as they used SHERLOCK. Our cognitive task analysis findings have been validated by a successful tutoring environment. Tutoring trainees to test their hypotheses, to reflect on the results of their measurements, to understand the procedures and conditions under which such procedures work, resulted in positive performance gains. SHERLOCK improved troubleshooting proficiency, and it is effective for trainees of various aptitudes. In addition, SHERLOCK's student modeling indicates that improvements occurred in every major category of knowledge that was targeted for training, which is confirmed by the analyses of posttest performance.

### *To What Do We Attribute SHERLOCK's Success?*

Since SHERLOCK has multiple instructional strategies, it is hard to attribute its success to one strategy or element. As educators we are pleased with the many indicators of its success, but as scientists we want to know why it worked and which principles it embodies are most important. Sometimes these two goals are in conflict, since ensuring positive learning outcomes may destroy the controlled experimental manipulation of instructional variables. Over the years, reviews of research on aptitude learning and instructional interactions have concurred that no one form of instruction is ideal for all learners [28, 29]. Consequently, it may be inevitable that successful training requires confounding of approaches. However, we believe that with detailed analyses of skill changes over the course of learning, and with some variation in approach from one tutor to the next, we can begin to untangle and validate the multiple instructional strategies that SHERLOCK embodies.

One way to test what is effective in SHERLOCK would be to test its underlying mechanisms in a new domain. We might term this the *cloning approach*, replicating both the instructional mechanisms and the assessment tools. An alternative way to test SHERLOCK's effectiveness would be a kind of "swapping." Having developed a successful coached practice environment, we could take out different components and see whether it still produced the same success. Whichever empirical method is chosen, we believe that the following instructional strategies are candidate contributors to successful training:

1. *Teach specific domain knowledge along with metacognitive skills and heuristics that leverage the domain knowledge.*

2. En
- wh
- ate
- on-
3. Pr
- loc
- shc
- fee
- pro
- ma
4. En
- pro
- exj
- ref
- sup
- pej
- shc
- pej
5. Tr
- coi
- gei
- gro
- ate
6. Co
- suu
- cej
- the
7. Bu
- pej
- rea
- we
8. En
- vic
9. Ad
- coi
- boj
- ual

### **Future I**

### *Reflective*

SHERLO  
ever, it d  
postsoluti  
that of ar  
enhancerr  
egies sugj

2. *Embed learning within a problem-solving environment that reflects the uses to which knowledge will be put.* SHERLOCK embodies activity-based learning situated in the context of the shop environment, where skills can be generalized to on-the-job learning.
3. *Provide opportunities for learning through a combination of guided practice and locally situated opportunities for observation* (i.e., tell the trainee what to do, or show him, only when he is stuck and needs a specific bit of assistance). Situate feedback as guided, on-request support. In this situated way, articulate expert problem-solving methods and foster the comparison of expert and trainee performance details.
4. *Emphasize holistic procedural training.* People learn to do complex cognitive procedures by doing them. Core mental models should be revealed and should be explicit in hints, but practice should focus on doing the thinking, not on rote repetition or playing abstractly with the background of conceptual knowledge that supports successful problem solving. When a trainee cannot handle the whole performance, he should be supported through the parts he cannot handle but should still, more or less, be enacting, with help, the whole target cognitive performance.
5. *Train the specific competence required rather than training abstractions.* Transfer comes from training concepts and procedures that are common to or can be generalized from experience in multiple situations. Abstractions must be grounded in prior concrete experience. This is the fundamental lesson about situated learning!
6. *Cognitive task analyses of the hardest thinking tasks in a job are critical to success in designing effective thinking.* These analyses identify the critical concepts and procedures situated within the job environment. Use the vocabulary of the workplace to further situate explanations and guidance.
7. *Build from the highest levels of domain expertise.* For a system to be able to shape performance toward expertise, it must reflect, in its coaching, the expertise of real workers who have learned to do the job, not just the designer's model of the work environment or the job as represented in doctrine.
8. *Enable the development of complex subgoal structures.* In SHERLOCK, we provided a tool that encouraged thinking about higher-order goals before subgoals.
9. *Adapt instruction.* Accommodate individual differences in learning by giving coaching only when it is needed. In SHERLOCK, detailed dynamic assessment, both of global competence and of local performance details, supports this individualization.

### Future Enhancements

#### *Reflective Follow-Up to Problem Solving*

SHERLOCK externalizes both the trainee's thought processes and the expert's. However, it does this in a local and fragmented way. It might be helpful to also provide a postsolution opportunity for the trainee to analyze his performance in comparison with that of an expert. Our colleagues Marilyn Bunzo and Garry Egan have designed an enhancement to SHERLOCK that combines the abstracted replay and post mortem strategies suggested by Collins et al. and Schoenfeld, respectively [7, 11]. At the end of each

ificant main effect (2) on the trainee's e had solved using ses were to expert o reach a solution. efficient. Both the ie low-ability indi-up. CK were achieved. posed to swapping .05), for example. nappropriate steps ore complex fail-phases of trouble-nt taking, as they alidated by a suc-s, to reflect on the itions under which RLOCK improved aptitudes. In addi-occurred in every ; confirmed by the

attribute its success ay indicators of its h principles it em-ict, since ensuring d manipulation of itude learning and ion is ideal for all l training requires l analyses of skill ach from one tutor uctional strategies

test its underlying h, replicating both way to test SHER-oped a successful ts and see whether osen, we believe o successful train-

ills and heuristics

problem a trainee will be provided with a visual representation of an expert trace of solving the problem. This representation will illustrate the plans and actions undertaken to reach a solution. Modest animation highlights, in turn, each step that was taken. Subsequently, the student's own problem-solving trace (see Figure 2) is provided below the expert's graph. Next, a third graph appears in which the student and expert traces are overlaid so that the trainee can see where and how much his solution overlapped an expert solution.

In addition, a trainee can explore the expert problem-solving trace by pointing to any node in the graph. When pointed to, each expert node will reveal an expert's reasoning for selecting this node in the solution, as well as hints as to what the appropriate actions, outcomes, conclusions, and next steps would be for that particular node (see Figure 2). Articulating the problem solution structure makes the role of SHERLOCK as a more capable peer more explicit and allows the user to evaluate his or her own responses in comparison to an expert's.

At a finer grain size, we have a second approach for comparative analysis of the testing done on a particular circuit component. Trainees would be able to see the active circuit paths marked on a schematic diagram of a printed circuit card and would receive explanations of appropriate testing strategy that referred to the illustrated paths (see Figure 3). In this way mental models of testing functional areas can become more salient to the trainee, and testing strategies may become less mysterious.

### Social Facilitation of Learning

The success models of cognitive apprenticeship in the classroom have one more element that could be incorporated into SHERLOCK's experiments on learning: the role reversal or procedure/critic element in which the trainee learns to generate and critique his own and others' work. Each of the cognitive apprenticeship approaches demonstrated to the students that learning was transitional and that perfect performance all the time was not

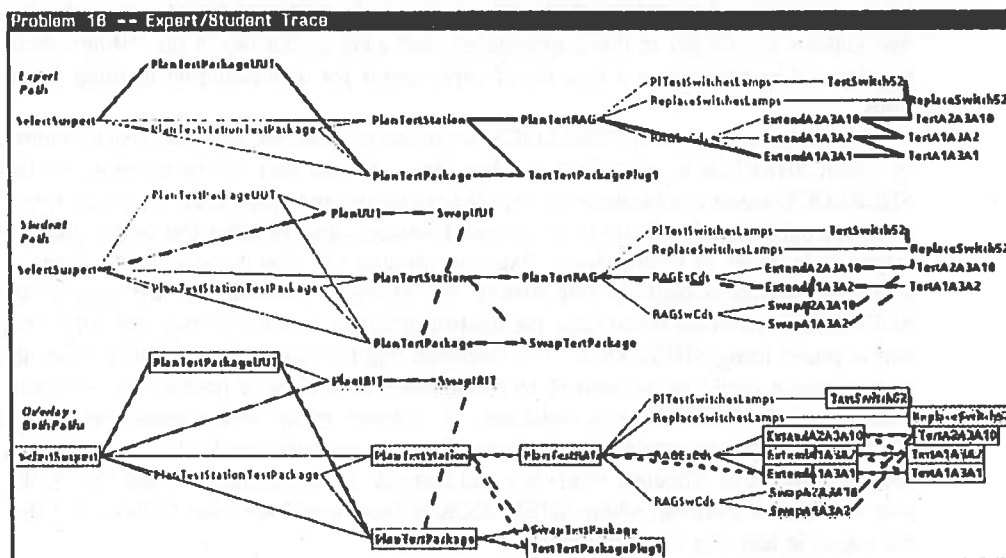


Figure 2 Comparison of expert and trainee problem solutions.

Relay Assembly	
by	
PLAN7 DECODED OVER ZUNE A05 (TEST POINT TERS SELECTION)	
SELECTION ZUMA FOR TEST POINT SELECTION RELAY'S	
SELECTION AND FOR GROUP TEST POINT RELAY: 00-04	
PLAN10 TEST POINT 5 TURNED AND IN ZUNE A21	
UNIT RELAY SELECTION (GND)	4 3 2 1 0

the "norm reading pa peers both to be transi Math prob problems 1 solution ju interaction mechanism ture" coul tions.

A poss in which SHERLOC tions in tro stages in 1 structured SHERLOC had acqui control gr where ins group. Th causing tr play trans ent stages

At the off compl

expert trace of  
ions undertaken  
that was taken.  
provided below  
expert traces are  
1 overlapped an

/ pointing to any  
pert's reasoning  
ropriate actions,  
: (see Figure 2).  
OCK as a more  
wn responses in

: analysis of the  
to see the active  
d would receive  
rated paths (see  
me more salient

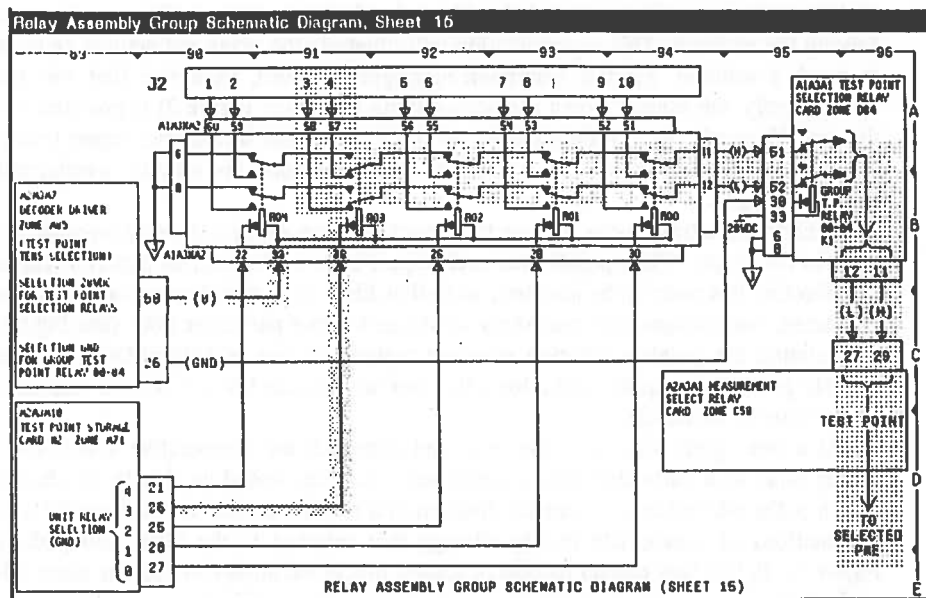
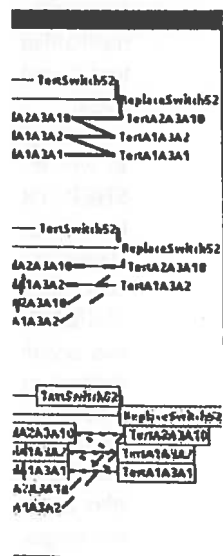


Figure 3 Display of relevant path in a printed circuit card.

the "norm." Reading comprehension was demonstrated to be transitional by discussing reading passages in a group of readers at different stages of development and having peers both produce work and critique each other's work [9]. Writing was demonstrated to be transitional in that revision was a significant component in the writing process [10]. Math problem solving was shown to be transitional by having teachers solve math problems that students generated, demonstrating that teachers can flounder toward a solution just as students can [11]. Each of these approaches has an aspect of social interaction not used in SHERLOCK. However, many of SHERLOCK's externalization mechanisms are similar to these approaches, and a controlled use of the "shop subculture" could be fostered in a new set of experiments for understanding learning transitions.

A possible extension of SHERLOCK would be to make the informal social context, in which SHERLOCK is used at the job site, a formal part of the training, so that SHERLOCK-based troubleshooting experiences are shared with peers. Learning transitions in troubleshooting might be enhanced if trainees observe other trainees at different stages in learning to troubleshoot. Experiments that use peer tutoring in the form of structured feedbacks could be informative. For example, trainees who had received the SHERLOCK treatment could tutor the control group on troubleshooting skills that they had acquired using SHERLOCK. The feedback that the tutor group would provide the control group could be structured by using SHERLOCK as a practice environment, where instead of SHERLOCK coaching, the tutored group would coach the control group. This approach would provide further learning opportunities for the tutor group by causing trainees to articulate what they had learned. Furthermore, it would clearly display transitional learning, where SHERLOCK-trainees could see other trainees at different stages in learning to troubleshoot.

At the completion of training, SHERLOCK's coaching mechanisms could be turned off completely and the trainees could be asked to coach their peers through a problem.



These teachbacks could assist trainees in developing the evaluative mechanisms necessary for successful problem solving. Trainees could take on the roles of both producer and critic, by generating efficient problem-solving traces and critiquing various solution paths, and having discussion groups that focus on elaborating why one solution path is more effective than another. This would be an interesting application of the producer/critic model described by Palincsar and Brown, in which the students are responsible for both generating their own responses and being their own critics [9]. It would also provide an additional external measure of their learning. We could observe how much of SHERLOCK's instruction was internalized and how well trainees could critique their peers in the troubleshooting process.

### *Tailoring to Local Needs*

We previously described how SHERLOCK was designed with the assistance of avionics experts who contributed to the knowledge engineering as well as to the teaching expertise reflected in SHERLOCK. We could extend the teaching expertise aspect of SHERLOCK by having more involvement of masters at different avionics shops, so that we could have different models of expertise in SHERLOCK. Using different experts from different shops may tailor instruction more to the specific shop demands.

### *Technological Enhancements*

Increasing the number of avionics units, test packages, and active test station drawers in SHERLOCK becomes time consuming to the problem developer. More technical documentation must be appropriated, more schematics must be developed and input into the computer, and a great deal of cross-checking of the test station setup with the TO test procedures is necessary. Creating schematics on the computer is a time-consuming activity and led us to restrict much of our instruction to a small part of the test station. With advances in interactive videodisc technology it will take less time to incorporate elaborate graphic displays. This will allow rapid development of tutors with a broader scope of devices and technical materials available in the work environment simulations, increasing the scope and realism of computer-based cognitive apprenticeship.

### *Acknowledgments*

The work was done as part of subcontracts with HumRRO and Universal Energy Systems, who were prime contractors with the Air Force Human Resources Laboratory. Many of the tools and much of the infrastructure that permitted us to conduct this project efficiently were put in place by earlier efforts and the foresight of the Office of Naval Research, where Henry Halff, Susan Chipman, Marshall Farr, and Michael Shafto patiently invested in a future that took a number of years to unfold. SHERLOCK was a massive undertaking. It took a lot of work from a lot of people. As a real-world applied activity in a setting with real work that could not be halted, it was a test not only of ingenuity and hard work but also of courage, patience, tolerance, perseverance, and the love and respect of many players for each other. Much of the design of SHERLOCK came from joint work with Marilyn Bunzo and Gary Eggan. Joining us in the effort at one point or another were Jaya Bajpayee, Jeffrey Bonar, Lloyd Bond, James Collins, Keith Curran, Richard Eastman, Drew Gitomer, Robert Glaser, Bruce Glymour, Linda

Greenberg, lin, Cassand  
tant were S  
Human Resc  
who helped  
33rd Tactica  
the Compon  
its many ret  
script, restr

### *Notes*

1. The te
2. The m
3. We thi
- SHERLOCK.
4. Note t
- chance that m
5. There
- reviewed this
- simulation for
6. A secc
- load in early
- simply will n
7. The m

### *References*

1. A. Carar  
Misconce
2. L. C. M  
July 1984
3. A. M. La  
D. Logar  
hance tec  
the Learn
4. P. Nichol  
troublesh  
sources I
5. J. S. Bro  
Education
6. A. Collin  
reading,  
tion: Ess
7. A. Collir  
H. Mand  
York: Spi
8. L. B. Re
9. A. S. Pa  
comprehe
10. M. Scard



mechanisms necessary of both producer and various solution one solution path is on of the producer/ are responsible for [9]. It would also serve how much of could critique their

sistance of avionics the teaching expertise aspect of SHERLOCK shops, so that we ferent experts from and.

station drawers in ore technical documented and input into the up with the TO test re-consuming active test station. With incorporate elaborate with a broader scope ent simulations, iniceship.

iversal Energy Systems Laboratory. conduct this project the Office of Naval Michael Shafto past SHERLOCK was a a real-world applied s a test not only of perseverance, and the ign of SHERLOCK g us in the effort at and, James Collins, ice Glymour, Linda

Greenberg, Denise Lensky, Debra Logan, Maria Magone, Tom McGinnis, Valerie Shalin, Cassandra Stanley, Arlene Weiner, Richard Wolf, and Laurie Yengo. Equally important were Sherrie Gott and the many people that worked with her at the Air Force Human Resources Laboratory, Dennis Collins and other Air Force subject matter experts who helped us when we needed it most, and the officers and personnel of the 1st and 33rd Tactical Fighter Wings, at Langley and Eglin Air Force Bases, especially those in the Component Repair Squadrons who made room in their worlds for SHERLOCK and its many retainers. Arlene Weiner put considerable effort and expertise into this manuscript, restructuring much of it and making it much more coherent.

### Notes

1. The term *inert knowledge* is due to Alfred North Whitehead.
2. The mouse is a common input device for graphical interfaces.
3. We thank Jill Larkin for helping us find understandable ways to discuss these aspects of SHERLOCK.
4. Note that SHERLOCK uses the jargon that trainees are known to use whenever there is a chance that more formal terms may not be as well understood.
5. There are other computer-based approaches to apprenticeship. Gott has comprehensively reviewed this literature [22]. Frederiksen, White, and Cooper have developed a computer-based simulation for teaching mental models in a domain closely related to SHERLOCK's [23].
6. A second purpose of the recapitulation hint is to compensate for processing capacity overload in early stages of training. We assume that the trainee will sometimes know what to do but simply will not be able to keep track of his progress, due to processing overload.
7. The next version of SHERLOCK will also have postproblem reflective opportunities.

### References

1. A. Caramazza, M. McCloskey, and B. Green, "Naive beliefs in 'sophisticated' subjects: Misconceptions about trajectories of objects," *Cognition*, 9, 117-23, 1981.
2. L. C. McDermott, "Research on conceptual understanding in mechanics," *Physics Today*, July 1984.
3. A. M. Lesgold, S. P. Lajoie, R. Eastman, G. Eggan, D. Gitomer, R. Glaser, L. Greenberg, D. Logan, M. Magone, A. Weiner, R. Wolf, and L. Yengo, *Cognitive task analysis to enhance technical skills training and assessment*, Technical Report, University of Pittsburgh, the Learning Research and Development Center, April 1986.
4. P. Nichols, R. Pokorny, G. Jones, S. P. Gott, and W. E. Alley, *Evaluation of an avionics troubleshooting tutoring system*, Special Report, Brooks AFB, TX: Air Force Human Resources Laboratory, in press.
5. J. S. Brown, A. Collins, and P. Duguid, "Situated cognition and the culture of learning," *Educational Researcher*, 18, 32-41, 1989.
6. A. Collins, J. S. Brown, and S. E. Newman, "Cognitive apprenticeship: Teaching the craft of reading, writing, and mathematics," in L. B. Resnick (ed.), *Knowing, learning, and instruction: Essays in honor of Robert Glaser*, Hillsdale, NJ: Lawrence Erlbaum Associates, 1989.
7. A. Collins and J. S. Brown, "The computer as a tool for learning through reflection," in H. Mandl and A. M. Lesgold (eds.), *Learning issues for intelligent tutoring systems*, New York: Springer, Verlag, 1988.
8. L. B. Resnick, "Learning in school and out," *Educational Researcher*, 16, 13-20, 1973.
9. A. S. Palincsar and A. L. Brown, "Reciprocal teaching of comprehension-fostering and comprehension monitoring activities," *Cognition and Instruction*, 1, 117-75, 1984.
10. M. Scardamalia and C. Bereiter, "Fostering the development of self-regulation in children's



- knowledge processing." in S. F. Chipman, J. W. Segal, and R. Glaser (eds.), *Thinking and learning skills: Research and open questions*, Hillsdale, NJ: Lawrence Erlbaum Associates, 1985.
11. A. H. Schoenfeld, *Problem solving in the mathematics curriculum: A report, recommendations and an annotated bibliography*, The Mathematical Association of America, MAA Notes, No. 1, 1983.
  12. L. J. Cronbach, "The logic of experiments on discovery," in L. S. Shulman and E. R. Keislar (eds.), *Learning by discovery: A critical appraisal*, Chicago: Rand McNally, 1966.
  13. J. Greeno, "A study of problem solving," in R. Glaser (ed.), *Advances in instructional psychology*, vol. 1, Hillsdale, NJ: Lawrence Erlbaum Associates, 1978.
  14. S. P. Lajoie and V. Shute, "Two intelligent tutors: Principles of instructional design," presented at the 93d annual convention of the American Psychological Association, Los Angeles, CA, 1985.
  15. A. L. Brown and R. A. Reeve, "Bandwidths of competence: The role of supportive contexts in learning and development," in L. S. Liben and D. H. Feldman (eds.), *Development and learning: Conflict or congruence?* Hillsdale, NJ: Lawrence Erlbaum Associates.
  16. N. M. Morris and W. B. Rouse, "Review and evaluation of empirical research on troubleshooting," *Human Factors*, 27, 503-30, 1985.
  17. S. P. Gott and T. S. Davis, "Introducing specific knowledge domains into basic skills instruction: From generalized power to specified knowledge," paper presented at the National Adult Education Conference, Philadelphia, PA, 1983.
  18. D. H. Gitomer, *A cognitive analysis of a complex troubleshooting task*, Doctoral diss., University of Pittsburgh, 1984.
  19. G. W. Ernst and A. Newell, *GPS: A case study in generality and problem solving*, New York: Academic Press, 1969.
  20. L. S. Vygotsky, *Mind in society: The development of higher psychological processes*, Cambridge, MA: Harvard University Press, 1978.
  21. S. P. Gott, "Apprenticeship instruction for real world cognitive tasks," in E. Z. Rothkopf (ed.), *Review of research in education*, vol. XV, Washington, DC: AERA, in press.
  22. A. M. Lesgold, "The integration of instruction and assessment in technical jobs," in *Proceedings of 1987 ETS invitational conference*, 81-88, Princeton, NJ: Educational Testing Service, 1988.
  23. J. R. Frederiksen, B. Y. White, and E. Cooper, "Restoring lost apprenticeship: An approach based upon intelligent tutoring," paper presented at the annual meeting of the American Education Research Association, New Orleans, LA, 1988.
  24. G. Salomon, "AI in reverse: Computer tools that become cognitive," invited address at the American Educational Research Association, New Orleans, April 1988.
  25. J. Mestre, R. Defresne, W. Gerace, and P. T. Hardiman, "Hierarchical problem solving as a means of promoting expertise," paper presented at the annual Cognitive Science Conference, Montreal, Canada, August 1988.
  26. S. P. Lajoie, G. M. Eggan, and A. M. Lesgold, "Instructional strategies for a coached practice environment," paper presented at the annual Cognitive Science Conference, Montreal, Canada, August 1988.
  27. A. M. Lesgold, S. P. Lajoie, M. Bunzo, and G. Eggan, "Sherlock: A coached practice environment for an electronics troubleshooting job," in J. Larkin, R. Chabay, and C. Scheftic (eds.), *Computer assisted instruction and intelligent tutoring systems: Establishing communication and collaboration*, Hillsdale, NJ: Lawrence Erlbaum Associates, in press.
  28. L. J. Cronbach and R. E. Snow, *Aptitudes and instructional methods: A handbook for research on interactions*, New York: Irvington, 1977.
  29. R. E. Snow, "Research on aptitudes: A progress report," in L. S. Shulman (ed.), *Review of research in education*, Vol. IV, Itasca, IL: Peacock, 1977.

Autom

UR  
KA  
GA  
Col  
Nev

Introducti

Interactive c  
range of use  
be as simple  
bases, desig  
dural in that  
construct pi  
through whi  
expertise. St  
their immed  
solution thr

Ursula Wol  
51438, a gra  
Kathleen R  
K-0256, by a  
ogy.  
Gail E. Ka  
AT&T, DEC,  
Center for Te

