Structuring Knowledge of a Non-linear Interface for Training Effectiveness

Martin Bink
U.S. Army Research Institute
Fort Benning, Georgia
martin.bink@us.army.mil

ABSTRACT

The new generation of Army command-post digital systems, e.g., Command Post of the Future (CPOF), increasingly utilizes non-linear interfaces. Non-linear interfaces use a customizable workspace that is based on the user's needs rather than a static data format, and interaction with the interface is not based on prescribed or hierarchical sequences of steps. Successful application of non-linear interfaces requires the user to decide which functions will best address a problem or need. One major difficulty in learning non-linear interfaces such as CPOF is that the interfaces do not support the organization of knowledge necessary for the individual to successfully interact with the system. This paper describes a two-part research effort to define the structured knowledge of CPOF that can serve as a basis for CPOF training. In the first part, a cognitive task analysis based on functional use of CPOF was conducted to produce a framework of CPOF knowledge. In the second part, data were collected on the retention of CPOF skills. Thirty-six participants from CPOF training at two battle command training centers completed a skills test immediately following training and again five weeks after training. The retention data were then integrated with the CPOF knowledge framework by analyzing the patterns of skill retention within the major categories of the framework. For example, the number of skills retained differed between the "Construct" category of the framework and the "Collaborate" category. The pattern of retention was then used to identify the specific skills and the progression of skills that are critical in developing CPOF expertise. The results can be used to indicate that some training techniques are more appropriate to CPOF skills than other techniques. For example, training techniques that leverage the execution of sub-goals and that illustrate overlapping CPOF procedures should most efficiently train CPOF skills.

ABOUT THE AUTHOR

Dr. Martin Bink is a Senior Research Psychologist at the U.S. Army Research Institute for the Social and Behavioral Sciences – Fort Benning (ARI). Dr. Bink holds a Ph.D. in Cognitive Psychology from The University of Georgia. His research expertise is in human learning, memory, and cognition especially as applied to education and training, and he has published many professional articles in those areas. For the past three years, Dr. Bink's research has focused on digital systems training, marksmanship training, and aviation collective training for ARI.

Structuring Knowledge of a Non-linear Interface for Training Effectiveness

Martin Bink
U.S. Army Research Institute
Fort Benning, Georgia
martin.bink@us.army.mil

INTRODUCTION

Many of the Army's digital system training programs depend more on learning a skill set as opposed to learning task execution. Learning a skill set is particularly the case with the Command Post of the Future (CPOF). CPOF is touted as a dynamic visualization tool that supports collaborative decision-making in tactical units. Staff personnel can use CPOF to assist in planning tactical operations, to monitor battlefield operations, and to provide update briefings to leaders. To accomplish these functions, the CPOF interface uses a customizable workspace that is based on the user's needs rather than a static data format. Successful application of CPOF requires the user to decide which functions will best address a problem or need.

Because the CPOF interface is mostly non-linear (i.e., interaction with the system is not based on prescribed sequences of steps and data), there is less internal cuing in the interface. Proficiency with non-linear interfaces requires a higher-level of understanding of task goals and interface capabilities (Farrell & Moore, 2000). Traditional instructional approaches (e.g., lecture and practical exercises) might not help a learner to develop such higher-level understanding as efficiently as other learning approaches. Thus, the challenge for CPOF training is to develop training approaches that are based on the underlying knowledge structure for the system.

Training Implications of Knowledge Structures for Digital-System

Knowledge structures can be defined as the cognitive organization of concepts that reflects the functional relations among conceptual features as a result of experience (Johnson-Laird, 1983; Collins & Gentner, 1987). In other words, knowledge structures are higher-level knowledge that guides inference and action (Collins & Gentner). In one view, the goal of training is to transform the knowledge structures of novices to an expert structure (e.g., Dorsey, Campbell, Foster, & Miles, 1999; Cooke, Durso, & Schvaneveldt, 1986). As a result, it is important to assess knowledge structures to determine training effectiveness (Kraiger, Ford, & Salas, 1993; Day, Arthur, & Gettman, 2001). However, the purpose of the work presented here is to suggest a way to organize training based on expert knowledge structures (e.g., Ericsson, 2009; Kraiger, Salas, Connon-Bowers, 1995) rather than to assess knowledge structures per se.

While there is little empirical research on the training requirements of non-linear interfaces, the research on learning from hypertext is an analogy for training non-linear digital-system interfaces. Two particular problems noted in research on learning from hypertext are (a) that learners (i.e., users) can become disoriented as they click link after link into the text and get further away from the starting point (e.g., Chen, 2002; Ellis & Kurniawan, 2000) and (b) that learning involves an independent and active learning process (Chen, 2002; Ford & Ford, 1993). Both of these problems seem to be related to the learner's level of conceptual knowledge of the task. That is, given that there is flexibility in the manner in which tasks are completed, people who have hierarchical-task knowledge are better able to monitor task progress and to select alternatives that will lead to efficient task completion (Chen, 2002; Dunlosky & Lipko, 2007).

Thus, the difficulty in learning non-linear digital interfaces such as CPOF appears to be based on the fact that novices do not have, and the interfaces do not support, the organization of knowledge necessary to successfully interact with the system. One possible solution to the potential problems of learning non-linear digital system interfaces is to base training on the development of hierarchical conceptual knowledge instead of on the memorization of steps (cf. Newell & Simon, 1972). The difficulty with this approach is that the development and use of hierarchical knowledge is associated with expert performance (Larkin, McDermott, Simon, & Simon, 1980) rather than as a method of training novices. However, some studies have demonstrated that novices benefit from learning hierarchical knowledge as compared to learning step-by-step procedures (e.g., Catrambone, 1998; Dufresne, Gerace, Hardiman, & Mestre, 1992).

Hierarchical knowledge of digital systems (i.e., the knowledge structure) should include the steps necessary to support the execution of a given procedure, the structure of the procedure, the underlying purpose of the procedure in executing a task, and some understanding of how the system functions as a whole. In general, developing hierarchical knowledge involves learning meaningful components of the overall concepts and then structuring those components based on the requirements of task goals (cf. Catrambone, 1998). In the case of learning non-linear digital systems such as CPOF, it is assumed that developing hierarchical knowledge is based upon learning the skills that are most critical to the intended functions of the system, applying those critical skills in task execution, and structuring the critical skills based on the application of those skills across tasks. In this case, "skills" refer to manipulations of the system interface that support completion of multiple tasks. It is this type of skill that is the focus of the present research.

The knowledge structure for digital systems identifies "what" needs to be instructed (and, presumably, assessed) but does not dictate "how" to do the instruction. The organization of the knowledge structure does provide a guide, though, to the order of instruction and the relations that should be emphasized in training. In addition, the hierarchy of concepts in the knowledge structure as well as procedural overlap among skills could be leveraged to organize training.

The purpose of this paper is to document one attempt at identifying knowledge structures for the application of CPOF that can be used to organize training and help develop training approaches. To do so, the knowledge structure for CPOF was identified by an analysis of CPOF critical skills. Skill-retention data from CPOF training was then analyzed to determine the training priority of individual skills.

A CRITICAL-SKILL HIERARCHY FOR CPOF

Three types of analyses were used to define CPOF critical skills and to synthesize those skills into a knowledge structure for CPOF. The preliminary analysis involved reviewing the CPOF Unser's Guide and other technical manuals for the CPOF system. This review helped to develop an understanding of the technical specifications of the systems and of the intended functionality of the system. The technical review was followed by a review of unit CPOF standard operating procedures (SOP). The SOP's provide an understanding of how CPOF is applied and of the functional requirements of the system. The final analysis was a cognitive-task analysis of CPOF in which the hierarchy of CPOF critical skill was identified. Together, the technical review and the SOP review provided the general structure for types of critical CPOF skills, for the functionality of CPOF, and for the general framework of the cognitive task analysis. The critical-skills cognitive task analysis provided data with which it was possible to detail the knowledge structure among CPOF skills based on expert knowledge and application. The cognitive task analysis was based on a specific knowledge extraction method (Catrambone, 1998). The details of the method and the results of the task analysis are described next.

Knowledge Extraction Method

Two separate knowledge extraction sessions were conducted. In each case the knowledge extraction expert (KEE) worked with a domain expert (DE) to uncover the knowledge needed to use CPOF. The DEs were CPOF trainers from the digital training facilities at two different installations. Both DEs had operational experience with CPOF. The DE performed a series of tasks based on the practical exercises developed for training Soldiers and that reflect how CPOF is operationally employed. The KEE took detailed notes and continually required the DE to explain why he was doing each step. The KEE tested the accuracy of the notes (called a Critical Skills Document) by doing tasks provided by the DE. The Critical Skills Document was iteratively updated and reorganized by the KEE and other members of the research team in order to identify the major components of the system and the procedures for accomplishing various goals. Commonality among sub-goals and functional dependencies across skills were aligned and uses to structure the Critical Skills Document.

The resulting Critical Skills Document covered the major capabilities of CPOF and organizes them in a way that makes the relations, including hierarchical ones, among those capabilities clear. It identified the various procedures and sub-procedures needed to use CPOF and represented them in a way to show their generality and applicability. The Critical Skills Document was then used to develop a skill hierarchy that was focused on the critical skills required for users to operate the CPOF system in an operational environment. The intent was to formalize the skill hierarchy into a representation of the knowledge structure for CPOF.

The knowledge structure was intended to accomplish two purposes. First, the knowledge structure needed to reflect an organization that was functionally sensible.

Second, the knowledge structure would need to be organized so it could guide the development of training materials that would maximize learning efficiency, retention, and problem solving flexibility. Multiple hierarchies that divided the CPOF capabilities and skills to emphasize different functional aspects of the system were considered. For instance, an organization based on military goals or an organization that focused on software capabilities regardless of the domain of application could be used. However, the most practical organization was one based on the construction, display, and sharing of tactical "products" within CPOF. The logic applied was that the user would first need to create or construct some tactical product (e.g., a map overlay). The completed product could then be visualized on the CPOF screen in relation to its operational context. Once completed and visualized, the user could share the product in collaboration with other CPOF users. As a result, the knowledge structure consisted of the functional groups of Construct, Visualize, and Collaborate. Within each of these functional groups are the knowledge about the tools, the processes, and the products related to the functions. This organization emphasized the main functional components of CPOF with its military application implicit within those components.

Knowledge Extraction Results: A Knowledge Structure

The results of the knowledge extraction analysis were used to create a visual representation of the functional components of the identified knowledge structure. Figure 1 illustrates the schematic architecture of the knowledge structure for CPOF skills produced by the analysis. Accordingly, the CPOF system is characterized by four main functional groupings: Construct, Visualize, and Collaborate, and System Basics. Each of these major categories is further divided, to one degree or another, into the elements of Processes, Products, and Tools. "Processes" are the CPOF procedures for executing tasks (e.g., the steps for drawing a Main Supply Route). "Products" are the results of the procedures (e.g., the resulting Main Supply Route). "Tools" are the CPOF software features used in the procedures (e.g., the Graphics Palette).

The most basic CPOF function represented in the knowledge structure is System Basics. This function refers to the tools and processes needed for system operation and the interface. For example, one needs to know how to properly operate the mouse (e.g., left-clicking versus right-clicking) in order to properly operate the system. The Construct function refers both to the generic capability of CPOF to construct virtual products and to the construction of specific hierarchical products for Visualization and Collaboration. Thus, the Construct function can be viewed as subordinate to Visualize and Collaborate. Visualize and Collaborate are the primary CPOF functions in the schematic (i.e., Figure 1). Each of these primary functions contains products that allows for the execution of the specific function. For example, a Mapboard (a product under Visualize) allows the CPOF user to view the terrain, battle graphics, Events, etc. that provide a picture of the current situation. The Visualize function and the Collaborate function are interdependent. That is, the CPOF products that are typically shared during collaboration are Visualization products. Likewise, the main purpose for Collaboration is to produce Visualization products.

Training Implications for the Knowledge Structure

While the represented knowledge structure does not specifically indicate how the CPOF skills are best trained, the structure and content of the knowledge structure hint at some general training approaches. For instance, the knowledge structure indicates that Soldiers need to learn aspects of system basics (e.g., the Frame Dispenser) in order to effectively learn other tasks that depend on those basics (e.g., drawing graphical objects). Likewise, Soldiers need to construct a product before it can be visualized or used for collaboration. Therefore, Soldiers must learn how to construct at least some of the basic products (i.e., Construct Products) before training skills in the Visualize Products and Collaborate Products levels. It is not the case, however, that all of the processes and products from these basic functional groups (i.e., System Basics and Construct) need to be trained before the processes and products in other functional groups.

The knowledge structure also generally implies that training techniques that leverage the execution of sub-goals and that illustrate overlapping CPOF procedures should most efficiently train CPOF skills. One such training technique is problem-based training in which learning occurs as the result of facilitated problem solving (Hmelo-Silver, 2004). Of course, problem-based techniques might be only one tool for training CPOF skills based on the knowledge structure. Likewise, a mix of training techniques may be appropriate. For example, providing direct instruction on the System Basics should precede problem-based exercises in order to provide requisite system knowledge. From that point, a series of problems that

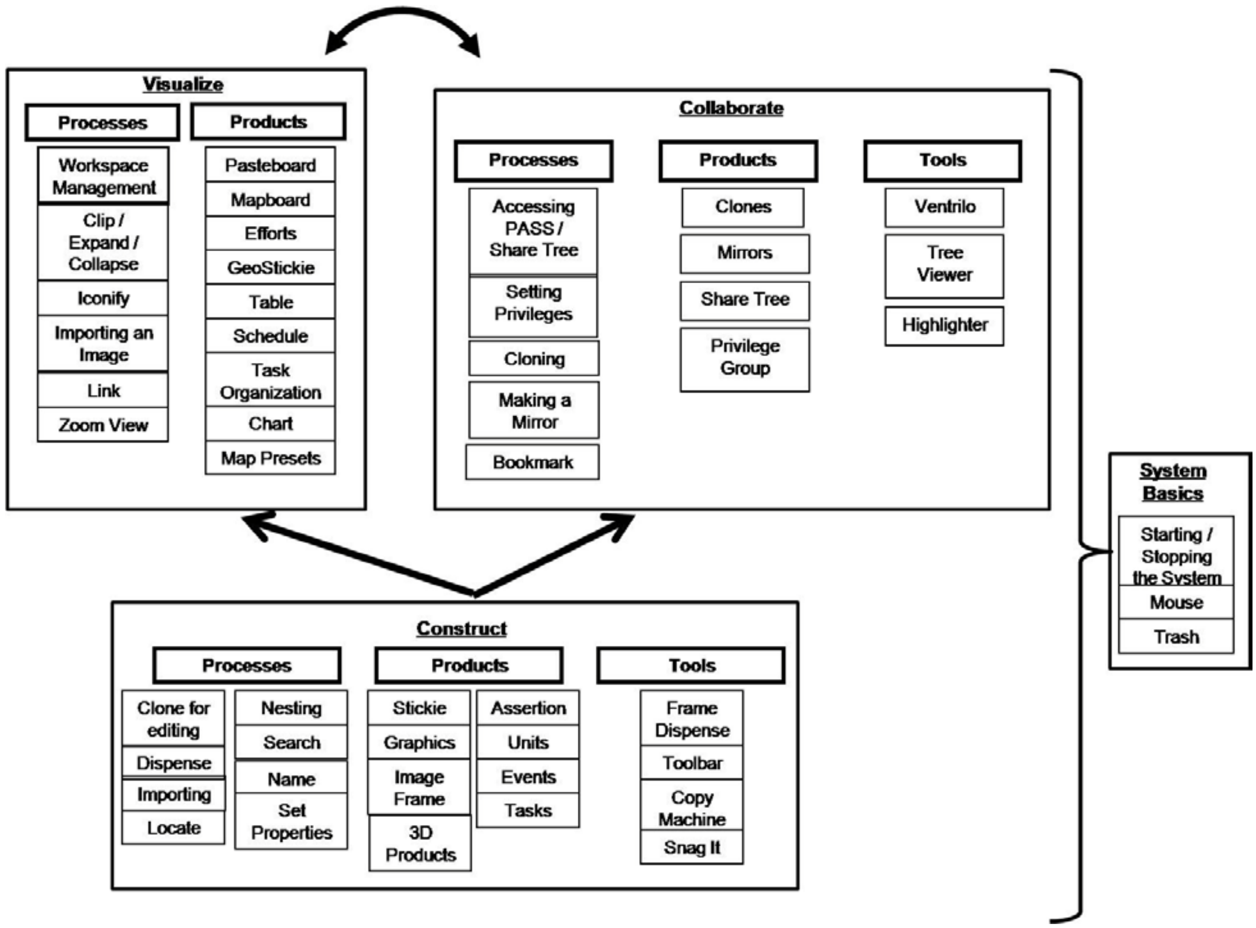

Figure 1. Schematic representation of the knowledge structure for CPOF skills derived from knowledge extraction analysis.

focus on simple sub-goals and procedural overlaps should be executed. Finally, more complex problems that require the construction and use of higher-level products can be addressed.

RETENTION OF CPOF SKILLS

Even though the knowledge structure for CPOF skills represented in Figure 1 carries implications for training, it was not intended to prescribe *what* CPOF skills need training. In order to understand what CPOF skills require more (or less) training and to further structure the progression of skills to be trained, a more empirical approach was needed. Skill retention data was collected in order to understand what people know as a result of CPOF training. The analysis of high- and low-retention skills was used to further refine the knowledge structure for CPOF and to further structure possible CPOF training.

In general, digital skills are quite perishable, and as a consequence, digital-skills training methods should be sensitive to patterns of skill retention (Goodwin, 2006; Goodwin, Leibrecht, Wampler, Livingston, & Dyer, 2007). As already stated, the complex and non-linear format of the CPOF interface makes the training and retention of CPOF skills particularly challenging. In particular, novices do not have the appropriate knowledge structures to support the integration of new information, and the interface does not necessarily cue procedural steps to complete a given task. What is more, there are few opportunities for individuals to practice CPOF skills because systems are generally only available in theater or in staff exercises. Thus, understanding the pattern of skill retention should impact *how* CPOF is trained.

Method for Collecting Skill-retention Data

Participants

Thirty-six participants from CPOF training at two Battle Command Training Centers (BCTC) completed a skills test immediately following training and again five weeks after training. The participants ranged in rank from Private First-Class to Command Sergeant Major and Captain and ranged in time-in-service from 15 months to 240 months. In general, the participants were Specialists or junior non-commissioned officers with less than 60 months time-in-service. Most participants had some tactical operations center experience. Six participants were unable to successfully complete the retention test because of duty requirements. As a consequence, the analyses reported are based on a total sample of 30 Soldiers.

Materials and Procedure

All participants completed 24-hour CPOF training at a BCTC. As part of the training course, participants completed an end of class practical exercise. The practical exercise required each participant to apply skills learned in training by preparing CPOF overlays and products that might be used for a battle-update brief. The practical exercise took one to two hours for participants to complete.

At the completion of the practical exercise, the course instructor (or assistant instructor) reviewed each participant's practical exercise and noted on a checksheet whether each task was successfully completed. The checksheet listed each specific CPOF task component required to complete the practical-exercise tasks, and the participant was given a "GO" for successful completion of the task component or a "NoGo" if the task component was not successfully completed. It is important to note that the specific task components were associated with specific CPOF skills identified in the Critical Skills Document developed from the task analysis. Thus, it was possible to quantify each participant's skill proficiency on 18 specific CPOF skills. It is also important to note that there were multiple instances of a given task component (e.g., "Create a Unit") on the practical exam.

Participants also completed a brief demographic questionnaire at the conclusion of the classroom practical exercise. Participants then returned to the BCTC four to six weeks later depending on duty schedule. Upon return to the BCTC, each participant completed a second practical exercise to assess the retention of CPOF skills. This retention practical exam was a similar format and content as the initial classroom practical exercise. Participants were given two hours to complete the exercise and were debriefed at the conclusion of the exercise. Again, a CPOF course instructor determined if each practical-exercise task was successfully completed and marked the results of each exercise on a checksheet.

Results and Discussion of Skill-retention Data

Throughout this paper, statistical significance was based on the five-percent level of alpha error. The means were analyzed with one-tail comparisons because only decreases in performance values were of interest. That is, the main purpose for the analyses was to identify CPOF skills that were *not* retained (i.e., statistically significant lower performance on the retention exercise than the initial exercise). If a skill was retained, it was of no of consequence to these analyses if the skill increased or stayed the same across the retention interval. Post-hoc differences in means were determined by pair-wise comparisons of 95% confidence intervals. Where appropriate, group means and standard errors of the means are given in the text.

Individual CPOF Skills

Individual items from each practical exercise were aggregated according to the individual CPOF skills from the Critical Skills Document in order to allow comparisons across exercises. The proportion correct of each item within each type of skill was calculated for both the classroom practical exercise and the retention practical exercise. Comparisons (i.e., paired *t*-tests) were performed on the proportions from each exercise for each skill. Performance on each skill was also aggregated across skill categories from the knowledge structure (i.e., Construct, Visualize, and Collaborate) for a comparison across CPOF skill types.

In general, retention of CPOF skills was fairly good over the four- to six- week retention interval. Overall, participants showed statistically significant forgetting of CPOF skills from the initial exercise to the retention exercise.[1] However, the proportion of correctly executed CPOF skills only decreased by about eleven percent from the initial exercise (m = .94, SEm = .02) to the retention exercise (m = .83, SEm = .04), and the proportions were fairly high even after a five-week retention period.

Each of the 18 CPOF skills assessed in the practical exercises was individually analyzed for forgetting across the retention interval. Nine of the CPOF skills showed no statistically significant forgetting (i.e., the skills were "retained"). Likewise, nine of the CPOF

[1] $t(29) = 2.81$, $MSe = .037$

skills showed a statistically significant decrease in proportion correct across the initial exercise and the retention exercise (i.e., the skills were "not retained"). Table 1 presents the specific skills that were retained and that were not retained. The largest difference between proportion correct across initial and retention[2] exercises was for "PASS: import + display product," which showed about a 21-percent decrease in performance (initial mean = .94, *SEm* = .03; retention mean = .74, *SEm* = .07).

Table 1. CPOF Skills Retained and Not Retained as a Function of the Knowledge-Structure Categories.

	Retained	Not Retained
Construct	Automatic Layout	Locate Product
	Create Graphic	Populate Effort (Clone product)
	Name/Label Graphic	Create Unit
	Set Event Table Properties	Create Event
		Event Properties
		Create Stickie
Visualize	Create and Name Pasteboard	Create + Name Map Preset
	Master Schedule	Set Preset View
	Create, Name, + Nest Map	
	Create, Name, + Place Effort	
Collaborate	Set Privileges	PASS: import + display product

Table 1 displays the skills according to their knowledge-structure categories. It is important to note that skills from each knowledge-structure category were both retained and not retained, but Visualize skills had the greater proportion of retained skills. It is also important to note that the retention of individual CPOF skills varied more as a function of the type of product (e.g., Units, Graphics, and Maps) than as the types of processes (e.g., Create and Name). That is, "creating" a map was a skill that was retained, but creating a unit was not retained. Thus, there appeared to be a disconnect between the sub-goals of the skills (e.g., "create") and the execution of whole skills (e.g., Create a Unit) with regard to how the individuals were learning and retaining CPOF skills.

Skill Categories

In order to specifically understand how CPOF-skill retention varied as a function of the knowledge-structure categories, proportions of correct responses for the practical exercises were compared across knowledge-structure categories. The resulting analysis yielded a statistically significant interaction between skill retention and knowledge-structure category.[3] Figure 2 displays the nature of the interaction. Accordingly, there was statistically significant lower[4] proportion correct for the retention exercise (*m* = .83, *SEm* = .04) than for the initial exercise (*m* = .93, *SEm* = .04) regardless of knowledge-structure categories. However, there was a larger difference between proportions correct for Collaborate than either Construct or Visualize. Thus, there was *less* retention of Collaboration skills than the other types of skills.

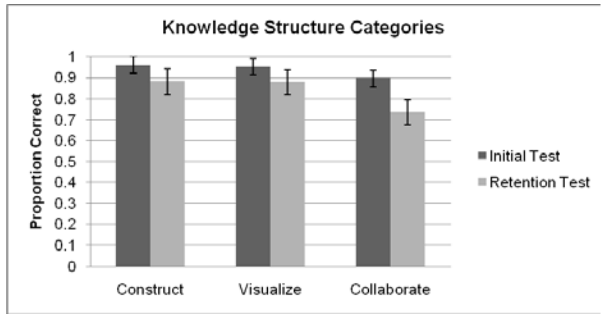

Figure 2. Skill Retention as a Function of Knowledge Structure Categories. Error Bars Represent 95% Confidence Intervals.

In addition to the patterns of retention, the correlations of retention performance among the knowledge-structure categories indicated that Collaborate skills did not share the retention properties of the other skills. While Construct and Visualize were highly correlated (*r* = .84), Collaborate was only moderately correlated with both Construct (*r* = .55) and Visualize (*r* = .56). Thus, it appeared that Visualize and Construct skills shared characteristics that likely contributed to the retention of the skills. Those shared characteristics are most likely the sub-goals of the skills, but Visualize and Construct skills may also have shared training overlap. That is, the training program of instruction for the CPOF courses observed focused heavily on Visualize skills, and as a consequence, the Construct skills were introduced in support for the developing Visualize products.

Summary of Results

In summary, even though overall retention of CPOF skills was high, there were differences in retention across skills. In particular, Collaborate skills had the largest decline in retention, but skills in each knowledge-structure category demonstrated forgetting. These differences reflect some combination of the inherent differences in the difficulty of the skills and the differential emphasis on the skills in training. More importantly, the differences in patterns of skill retention

[2] $t(29) = 3.03$, $MSe = .068$

[3] $F(2, 58) = 3.22$, $MSe = .019$

[4] $F(1, 29) = 7.11$, $MSe = .068$

provided a means to (a) identify which skills were retained and which skills were not retained, (b) discover possible disconnects among skill sub-goals, and (c) identify the relations among skills. These results should help determine how to prioritize the training of CPOF skills within the context of the knowledge structure.

STRUCTURING TRAINING

Again, the purpose of this paper was to utilize hierarchical knowledge structures of CPOF skills to suggest new ways to train the digital system. The advantages of aligning CPOF training with underlying knowledge structures are (a) to reduce the cognitive load as material is learned, (b) to provide a referential context for learning, and (c) to facilitate the ability to make inferences and apply learning (cf. Kieras & Bovair, 1984). Because CPOF has a non-linear interface, one key to developing new training techniques based on knowledge structures is to provide instruction in a way that minimizes cognitive load while maximizing efficiency (Byrne, Catrambone, & Stasko, 1999; Gerjets, Scheiter, & Catrambone, 2004; Mayer, 2005; Sweller, 2005). Likewise, using the knowledge structure to define the types of skills addressed in training will link the learning to the application of the skills (i.e., transfer) (Glaser & Bassok, 1989). The following discussion describes one way in which the preceding research results can be formalized into a training approach.

Based on the results of the knowledge-structure development and of the skill-retention analysis, a training approach was developed that has three primary characteristics. First, the training approach specifies that the knowledge structure is used as an explicit training context. Second, the training methodology specifically matches the characteristics of the knowledge structure. Finally, the training approach specifies a progression of skills training to continually reinforce learning.

Training Context.

The knowledge structure can be used as a general context for CPOF training. A representation of the knowledge structure such as the schematic presented in Figure 1 can be given to students as an advanced organizer. This schematic information should be explained in order to allow students to develop a general understanding of the relations among skills. Such an approach has been shown to increase retention and facilitate inference (Kieras and Bovair, 1984). The training of specific skills should be accomplished with reference to the knowledge structure when possible. That is, CPOF instructors should sequence the training of skills within a given knowledge-structure category and should reference the knowledge-structure schematic when transitioning from the training of one skill to the next.

Training Method.

The knowledge structure for CPOF can also be used to define the method of training. The structure and content of the CPOF skills hint that some training techniques better lend themselves to CPOF skills than other techniques. More specifically, training techniques that leverage the execution of sub-goals and that illustrate overlapping CPOF procedures should most efficiently train CPOF skills. One such training technique is problem-based training in which learning occurs as the result of facilitated problem solving (Hmelo-Silver, 2004).

In problem-based training, the trainer provides learners with structured problems designed to develop a specific skill. The trainer does not necessarily provide information, but rather serves to facilitate the problem-solving process. The learners typically work in groups to solve the problem. The solution process requires the learners to develop learning strategies and to explore content knowledge. The important aspect of problem-based training is that each problem is structured around the use of a specific skill requirement (Hmelo-Silver, 2004). In the case of CPOF skills, problems should be based on operationally-relevant tasks and should require learners to discover the overlap among tools and procedures across the knowledge-structure categories. Moreover, problems should focus on the common sub-goals across skills with different retention properties. For example, a problem that requires learners to construct and locate a Graphic, and a Unit on a map would reinforce an important CPOF skill (i.e., "Locate Product") that was not well retained and would allow the learners to discover the similarities and distinctions among these CPOF products.

Progression of Skills.

A facilitated problem-solving training approach suggested by the knowledge structure highlights the importance of training CPOF skills in specific sequence. Identifying the optimal sequence comes from information in the knowledge structure. The knowledge structure indicates the hierarchical sequence of CPOF skills. For example, System Basic and Construct skills should be trained first followed by

Visualize and Collaborate. The knowledge structure can also help identify common sub-goals that can be leveraged in training. In addition, the retention properties of the CPOF skills should be used to determine the sequence of training. That is, training skills that were better retained should precede the training of skills that were not retained. What is more, training the better-retained skills should be reinforced as more focus is given to training the skills that were not retained.

In general, sequencing training for complex skills should begin with skills that provide effective strategies (Clawson, Healy, Ericsson, & Bourne, 2001) or with skills that leverage common sub-goals (Catrambone, 1998). In the case of CPOF skills, applying these two general guidelines was a matter of determining which skill sub-goals were common across both retained skills and skills not retained. The set of common sub-goals are listed in Table 2. These skill sub-goals represent the general procedural steps for producing many of the CPOF products and can be generalized as new skills are introduced in training. Training these specific sub-goals up front and continually reinforcing them as new skills are trained is a way to provide an effective learning strategy.

Table 2. Common Sub-Goals for CPOF Skills.*

CPOF Skills Sub-Goals
Retrieve item from Frame Dispenser
Input product information and Name
Drag product to desktop
Use drop down boxes to select features
Drag to desired location on Pasteboard
Click "Nesting Icon" while available

*Listed in order of execution

Training the common sub-goals is just the first step in the defining the sequence of CPOF-skills training. Based on the knowledge structure, the retention properties of CPOF skills, and the general guidance on sequencing the progression of skills, a sequence of skills training can be defined. The knowledge structure indicated that the general system knowledge supports the execution of specific CPOF skills. As a result, System Basics skills should be introduced first. These skills address the basic functions of the system and the layout of the interface. After System Basics, a set of basic skills should be introduced. These basic skills represent the general functionality of CPOF and are common across the knowledge-structure categories. The three most basic CPOF skills are using the Frame Dispenser (i.e., the source for creating most other products), creating a Pasteboard (i.e., the highest level product and most operationally relevant), and creating a Map (i.e., the most basic visualization tool).

After these basic skills are introduced, a set of skills that are better retained and that have the sub-goals listed in Table 2 should be trained. Doing so will reinforce the basic sub-goal procedures and will introduce relatively easy skills. Likewise, these initial skills should be Construct skills because these skills support the higher-level skills. The next steps in the progression is to introduce more difficult (i.e., skills not retained) Construct skills, introduce Visualize skills that were retained, and continue to train retained and not-retained skills as higher-level skills are introduced. Table 3 displays a typical progression of skills training and provides examples of specific CPOF skills that apply to each level of the progression.

Table 3. Progression of CPOF Skills Training.

Progression of Skills	Example Skills to be Trained
Train some basic system skills.	Starting and Stopping the System
Introduce basic CPOF products.	Create Pasteboard; Create Map
Introduce Retained Construct skills, and have students apply that knowledge in an operationally-relevant problem-solving task.	Create Graphic
Introduce Not Retained Construct skills and Retained Visualize skills that incorporate common sub-goals, and have students apply that knowledge.	Create Unit; Create Effort
Introduce Not Retained Visualize skills and Retained Collaborate skills that incorporate common sub-goals, and have students apply that knowledge.	Set Preset view; Set Privileges
Continue to introduce and incorporate more complex skills while reinforcing the training with application.	PASS: Import + Share product

By following this progression, increasingly complex skills or skills that are more susceptible to forgetting can be trained in the context "easier" skills. The specific skills introduced at each step of the sequence can be determined not only by the procedural commonality with already learned skills but also by the operational relevance of the skill. With these two factors in mind (i.e., the progression of skills and operational relevance), specific training problems can be constructed to form the basis of the training approach. The problems should be developed across the different CPOF purposes and should require decision making and collaboration at every level.

DISCUSSION

This paper offered a method whereby the knowledge structure for and learning characteristics of an Army digital system could be used to define the content and structure of system training. That is, a training approach was developed specifically to address the results of two lines of research on the human performance properties of CPOF. First, knowledge extraction was used to develop a representation of expert knowledge that provided a general training framework. Second, skill-retention data provided specific skill-performance information that was used to prioritize training. The resulting training approach utilized the developed knowledge structure as a training context, prescribed a facilitated problem-solving method for training, and identified a method to sequence the training of CPOF skills.

While the proposed training approach is a logical extension of the available data, several issues remain unresolved. These issues may impact the actual efficiency of the proposed approach and the way in which the proposed approach can be implemented. First, not all important CPOF skills were tested for retention (e.g., Workspace Management). Including these skills would be necessary for any new CPOF-training approach, but not enough information is available at this time to determine where these important skills would be introduced in the training sequence. Second, the retention properties of CPOF skills would likely change as a result of the type of training used (Arthur, Bennett, Stanush, & McNelly, 1998). Thus, in order to properly sequence skills training, the retention of CPOF skills would need to be assessed after the proposed training approach is implemented, and the sequence of training should be modified according to those results.

Finally, no direct empirical evidence yet exists to support the assertion that problem-based training will be effective for CPOF skills. However, research on training other digital systems suggests that problem-based training was effective for complex systems like CPOF. For example, in the comparison of constructivist training techniques (e.g., problem-based training) to lecture-based training for the All Source Analysis System and the Advanced Field Artillery Tactical Data System, results showed that problem-based training produced higher scores than lecture-based training on the performance-based practical exercises at the end of digital-skills courses. (Childs, Blankenbeckler, & Dudley, 2001; Childs, Schaab, & Blankenbeckler, 2002). In addition, Childs et al. (2002) reported that problem-based training allowed for more material to be trained in less time without the perception of additional workload.

Of course, problem-based techniques might be only one tool for training CPOF skills based on the results presented in this paper. The structure of the proposed training approach suggested that a mix of training techniques may be appropriate. For example, providing direct instruction on the System Basics should precede problem-based exercises in order to provide requisite system knowledge. From that point, a series of problems that focus on simple sub-goals and a progression of skills should be executed. It may a case that the progression of skills training could be accomplished with an effective technique such as deliberate practice (e.g,. Ericsson, Krampe, & Tesch-Romer, 1993). However, the main advantage of a problem-solving approach is that the relations among skills can be implicitly trained without additional explicit training on the structure among skills.

Given the increasing complexity of Army digital-systems interfaces, traditional digital-system training approaches (i.e., memorization of key strokes and menus) will limit the degree of training efficiency. In order to avail oneself of the capabilities of complex non-linear systems, the user must know the total system capability and understand how the system can be applied to meet operational needs. That is, the user must not only know how to do things, but also know when to do them (i.e., decision rules). The approach to training development offered in this paper specifically allows hierarchical knowledge of complex non-linear digital systems to be trained.

ACKNOWLEDGEMENTS

Richard Wampler, Chris Strauss, and Richard Catrambone are recognized for their contributions to the data collection and to knowledge-structure analysis.

REFERENCES

Arthur, W., Bennett, W., Stanush, P. L., & McNelly, T. L. (1998). Factors that influence skill decay and retention: A quantitative review and analysis. *Human Performance, 11,* 57 – 101.

Byrne, M.D., Catrambone, R., & Stasko, J.T. (1999). Evaluating animations as student aids in learning computer algorithms. *Computers & Education, 33,* 253-278.

Catrambone, R. (1998). The subgoal learning model: Creating better examples so that students can solve novel problems. *Journal of Experimental Psychology: General. 127*, 355-376.

Chen, S. (2002). A cognitive model for non-linear learning in hypermedia programmes. *British Journal of Educational Technology, 33*, 449-460.

Childs, J. M., Blankenbeckler, P. N., & Dudley, M. G. (2001). *Digital skills training research* (ARI Contractor Report No. 2001-04). Alexandria, VA: U.S. Army Research Institute for the Behavioral and Social Sciences. (DTIC No. AD A391 790).

Childs, J.M., Schaab, B.B., Blankenbeckler, P.N., (2002). *Digital Skill Learning Using Constructivist Training Methods*. (Interservice/ Industry Training Simulation & Education Conference Proceedings). Orlando, FL: National Training Systems Association.

Clawson, D. M., Healy, A. F, Ericsson, K. A., Bourne, L. E. (2001). Retention and transfer of morse code reception skill by novices: Part-whole training. *Journal of Experimental Psychology: Applied, 7,* 129-142.

Collins, A. M., & Gentner, D. (1987). How people construct mental models. In D. Holland & N. Quinn (Eds.), *Cultural models in thought and language* (pp. 243-265). Cambridge, England: Cambridge University Press.

Cooke, N. M., Durso, F. T., & Schvaneveldt, R. W. (1986). Recall and measures of memory organization. J*ournal of Experimental Psychology: Learning, Memory, and Cognition, 12*, 538-549.

Day, E. A., Arthur, W., & Gettman, D. (2001). Knowledge Structures and the Acquisition of a Complex Skill. J*ournal of Applied Psychology, 86*, 1022-1033.

Dorsey, D. W., Campbell, G. E., Foster, L. L., & Miles, D. E. (1999). Assessing knowledge structures: Relations with experience and posttraining performance. *Human Performance, 12*, 31-57.

Dufresne, R. J., Gerace, W. J., Hardiman P. T., & Mestre, J. P. (1992). Constraining novices to perform expertlike problem analyses: Effects on a schema acquisition. *The Journal of Learning Sciences, 2,* 307-331.

Dunlosky, J., & Lipko, A. R. (2007). Metacomprehension: A brief history and how to improve its accuracy. *Current Directions in Psychological Science, 16*, 228-232.

Ellis, R. D., & Kurniawan, S. H. (2000). Increasing the usability of online information for older users: A case study in participatory design. *International Journal of Human-Computer Interaction, 12*, 263-276.

Ericsson, K. A. (2009). Development of professional expertise: Toward measurement of expert performance and design of optimal learning environments. New York: Cambridge University Press.

Ericsson, K. A., Krampe, R. T., & Tesch-Romer, C. (1993). The role of deliberate practice in the acquisition of expert performance. *Psychological Review*, *100*, 363 – 406.

Farrell, I. H., & Moore, D. M. (2000). The effects of navigation tools on learners' achievement and attitude in a hypermedia environment. Journal of Educational Technology Systems, 29, 169-181.

Ford, N., & Ford, R. (1993). Towards a cognitive theory of information accessing: An empirical study. *Information Processing and Management, 29*, 569-585.

Gerjets, P., Scheiter, K., & Catrambone, R. (2004). Designing instructional examples to reduce intrinsic cognitive load: Molar versus modular presentation of solution procedures. *Instructional Science, 32*, 33-58.

Goodwin, G. A. (2006). The training, retention, and assessment of digital skills: A review and integration of the literature (ARI Research report 1864). Arlington, VA: U.S. Army Research Institute for the Behavioral and Social Sciences. (DTIC No. ADA470707)

Goodwin, G. A., Leibrecht, B. C., Wampler, R. L., Livingston, S. L. & Dyer, J.L. (2007). *Retention of selected FBCB2 operating skills among Infantry Captains Career Course (ICCC) students.* (ARI Research Report 1872). Arlington, VA: U.S. Army Research Institute for the Behavioral and Social Sciences. (DTIC No. AD A470 741).

Glaser, R., & Bassok, M. (1989). Learning theory and the study of instruction. In M. R. Rosenzweig & L. W. Porter (Eds.), *Annual review of psychology, 40*, 631-666. Palo Alto, CA: Annual Reviews.

Hmelo-Silver, C. E. (2004). Problem-Based Learning: What and How Do Students Learn? *Educational Psychology Review, 16*, 235 – 266.

Johnson-Laird, P. N. (1983). *Mental models: Towards a cognitive science of language, inference, and consciousness.* Cambridge, MA: Harvard University Press.

Kieras, D. E., & Bovair, S. (1986). The acquisition of procedures from text: A production-system analysis of transfer of training. *Journal of Memory and Language*, *25*, 507-524.

Kraiger, K., Ford, J. K., & Salas, E. (1993). Application of cognitive, skill-based, and affective theories of learning outcomes to new methods of training evaluation. *Journal of Applied Psychology, 78*, 311-328.

Kraiger, K., Salas, E., & Cannon-Bowers, J. A. (1995). Measuring knowledge organization as a method for assessing learning during training. *Human Factors, 37*, 804-816.

Larkin, J., McDermott, J., Simon, D. P., & Simon, H. A. (1980). Expert and novice performance in solving physics problems. *Science*, *208*, 1335-1342.

Mayer, R. E. (2005). Cognitive Theory of Multimedia Learning. In R. E. Mayer (Ed.), *The Cambridge Handbook of Multimedia Learning* (pp. 30-48). New York: Cambridge University Press.

Sweller, J. (2005). Implications of Cognitive Load Theory for Multimedia Learning. In R. E. Mayer (Ed.), *The Cambridge Handbook of Multimedia Learning* (pp. 19-30). New York: Cambridge University Press.