Emerging Technology and Concepts for Computer-Based Training

Jeffrey M. Bradshaw, The Boeing Company
David Madigan, University of Washington
Tyde Richards, Apple Computer
Guy A. Boy, EURISCO
Peter Fairweather, Jostens
Chuck Buchanan, United Airlines
DRAFT: 26 April 1993

ABSTRACT

This paper describes the purpose and current activities of a new emerging technologies and concepts (*etc*.) subcommittee of the Aviation Industry Computer-Based Training Committee (AICC). The objective of the *etc*. subcommittee is to help computer-based training (CBT) software and hardware suppliers, developers, and users in aerospace to better respond to important industry problems, including the accelerating pace of technological change, reductions in CBT budgets and resources, and the need to increase leverage in computing industry standards efforts. The subcommittee sponsors cross-industry projects that develop and evaluate prototypes of new technology in order to understand changing requirements, and to address cost-of-technology and standards issues. Interoperability is discussed on three levels: cross-authoring-tool, cross-application, and intelligent interoperability.

1. Background: The Aviation Industry Computer-Based Training Committee

During the past decade, the aerospace industry has become a major proponent of Computer-Based Training (CBT).[1] A significant portion of the aerospace CBT in use is supplied to the airlines by the airframe manufacturers. Manufacturers such as Boeing, McDonnell-Douglas, and Airbus Industrie (Aéroformation) each delivers the courseware in its own proprietary manner, with widely varying interfaces, and conflicting hardware and software requirements. This would not be a problem, except that in today's world it is common for major airlines to have airplanes from several airframe manufacturers in their fleets, each type requiring the purchase, maintenance, and operation of a different dedicated instructional delivery system.

The Aviation Industry Computer-Based Training Committee (AICC) was formed in 1988 to promote cooperation among those involved in designing, producing, and using courseware. Members of the AICC include European and U.S. airframe manufacturers, CBT suppliers, airlines

[1] The terms *computer-based training* and *computer-aided instruction* (CAI) are often used interchangeably. We use CAI to signify the use of the computer to aid or support learning, and CBT for the use of the computer as the primary mode of instruction (AICC, 1992).

and third-party courseware developers, major computing hardware and software companies, and various airline and CBT industry associations.

Because of its diverse composition, the group is in a unique position to bring its resources to bear on *solving real-world problems* of compatibility, portability, interoperability, and technological change. While not itself a standards-setting body, the AICC works closely to coordinate its work with standards-settings bodies (e.g., ATA, AIA, ISO[2]), and produces guidelines that can be referenced in standards documents where appropriate. The guidelines are generated and refined by subcommittees on digital audio, platforms, computer-managed instruction, networking and operating systems, courseware technology, and communications. Compliance with guidelines is assessed by an independent test laboratory at the University of North Dakota.

2. Addressing Emerging Technology and Concepts for CBT

At the Spring 1993 meeting of the AICC, a new *emerging technology and concepts (etc.)* subcommittee was approved. Impetus to the effort was given by the recognition of three current challenges:

> *Accelerating pace of technological change*. Increasing competitiveness among technology providers is dramatically shortening the interval between initial conceptualization and delivery of new products (Davidow & Malone, 1992). Capabilities that were only dreamed of a very short time ago (e.g., handheld multimedia devices, wireless networking) will soon be widely available. Such emerging technologies portend revolutionary changes in the nature of CBT itself; without timely cooperation on new concept evaluation, the technology base of the CBT industry will become even more fragmented and incompatible than before (Masie, 1993).
>
> *Tighter budgets and fewer resources*. Rapid technological change compounds the problems of airlines and other CBT users who are strapped for cash and resources. CBT developers and users cannot afford to replace existing hardware and software wholesale, but must adopt new technology incrementally. Tools and strategies to reduce the cost and effort new technology adoption and for migration of “legacy courseware” must be actively investigated.
>
> *Insufficient leverage in computing industry standards efforts*. Until recently, there has been little direct participation in the AICC by major computing hardware and software suppliers and standards groups. As a result, input from the CBT community has come too late and in too fragmented a form to receive serious consideration before standards are adopted. A proactive stance involving early evaluation of new standards, before they become standards, is needed.

This paper surveys some of the significant emerging technologies and concepts relevant to the future of CBT in aerospace. The *etc.* subcommittee is addressing these problems through the sponsorship of projects designed to help AICC members better anticipate and more quickly respond

---

[2] Airline Transport Association, ???, International Standards Organization.

to new CBT technology and concepts. We have organized project objectives in terms of progressive levels of interoperability: from cross-authoring-tool interoperability to cross-application interoperability to intelligent interoperability.[3] We discuss these concepts in more detail below.

3. Defining Interoperability for CBT

Interoperability is very simply the ability to mix and match system components from different manufacturers, without worrying about incompatibilities between them. As Nance (1991) observes, one of the most interesting characteristics of interoperability is the fact that the word is no longer used in reference to things that actually *do* interoperate. For example, if you have ever connected a telephone to a wall socket using an RJ-11 jack or inserted a cassette tape into a tape player, you have experienced interoperability in its highest form. But you would never use the word to describe the experience. Within the telephone and consumer electronics industry, interoperability has become a thing *expected,* not a thing discussed.

On the other hand, the very popularity of the term among people who use information technology evidences their frustration at not being able to "plug and play" hardware and software components from different sources. Though progress is being made on many fronts, it is still far too slow to keep pace with technological change.

Interoperability is technically feasible. However, the efforts to achieve it require consensus on a wide range of standards. The first question to be resolved in the realm of CBT systems is what should be standardized. We can imagine standards based on four things: (1) *hardware*, (2) *operating systems*, (3) *authoring software*, or (4) the *interfaces* between these components. Fletcher (1992) compares these approaches and concludes that there are serious problems with the first three of them.

First, basing all interactive courseware on the same generic hardware platform, operating system, or authoring software requires a common elements specification. Standards based on common elements tend to migrate toward the least common denominator. Unique capabilities of a platform may languish for many years before they are included in the standards, while users increasingly seek ways of justifying exceptions to them. Second, providing for upward compatibility can become very difficult when standards require specific hardware, operating systems, or authoring software. Third, different users have different needs. Standards must accommodate situations where factors such as cost and performance have a legitimate impact on platform and software choices. Fourth, competition is best preserved when interface-based standards are adopted. Although standards based on hardware, operating systems, or authoring software can be expressed in generic terms, they still have the effect of reducing the number of competing players.

---

[3] Although the general charter of the *etc*. subcommittee was discussed favorably at a previous gathering of the AICC, the specifics of individual projects are not due to be presented until the March 1993 meeting. Thus, until that time, the perspectives and proposals outlined in this paper should be regarded as the views of the individual authors alone and not necessarily those of the AICC.

For these reasons, we advocate an approach to interoperability based on standardizing *interfaces* between components rather than standardizing on the components themselves. This allows for unique capabilities of hardware and software components, minimizes the changes required to accommodate technological evolution, permits users to select the most appropriate and cost-effective configuration for their situation, and encourages competition among CBT system suppliers.

Early CBT systems generally made strong assumptions about their operating environment. At first, dedicated, proprietary development and delivery systems that ran on specialized hardware were the rule. Over time, nearly all CBT development tool suppliers have produced versions of their software that run on variously configured DOS, Macintosh, or Unix platforms. Significant progress is being made as the AICC identifies hardware and operating system configurations that can play courseware from several sources. Other AICC efforts have supported the goal of interoperability by producing guidelines that allow the use of audio cards from different hardware suppliers, and that allow lessons produced using different development tools to be sequenced and managed programmatically.

Ultimately, we would like an approach that is operating-system- and hardware-independent. A few tools currently run under more than one operating system (Smith, 1993). Though this has made the CBT software more *portable,* it unfortunately has not made it more *interoperable*. Because software interfaces and data formats are usually proprietary, components from different suppliers cannot be easily be substituted or incorporated within a given CBT development or delivery system.

A CBT architecture for interoperability incorporates well-defined interfaces between system components. Each architectural layer addresses only the layers adjacent to it. Developers have complete freedom in designing approaches within their layer of concern so long as they adhere to standards at the interfaces. We believe that consensus on three kinds of interfaces is necessary for the achievement of CBT interoperability: multimedia database, virtual device, and interapplication communication. Additionally, next generation CBT will call for an understanding of interface requirements for *intelligent* interoperability.

4. Current Emerging Technology Projects

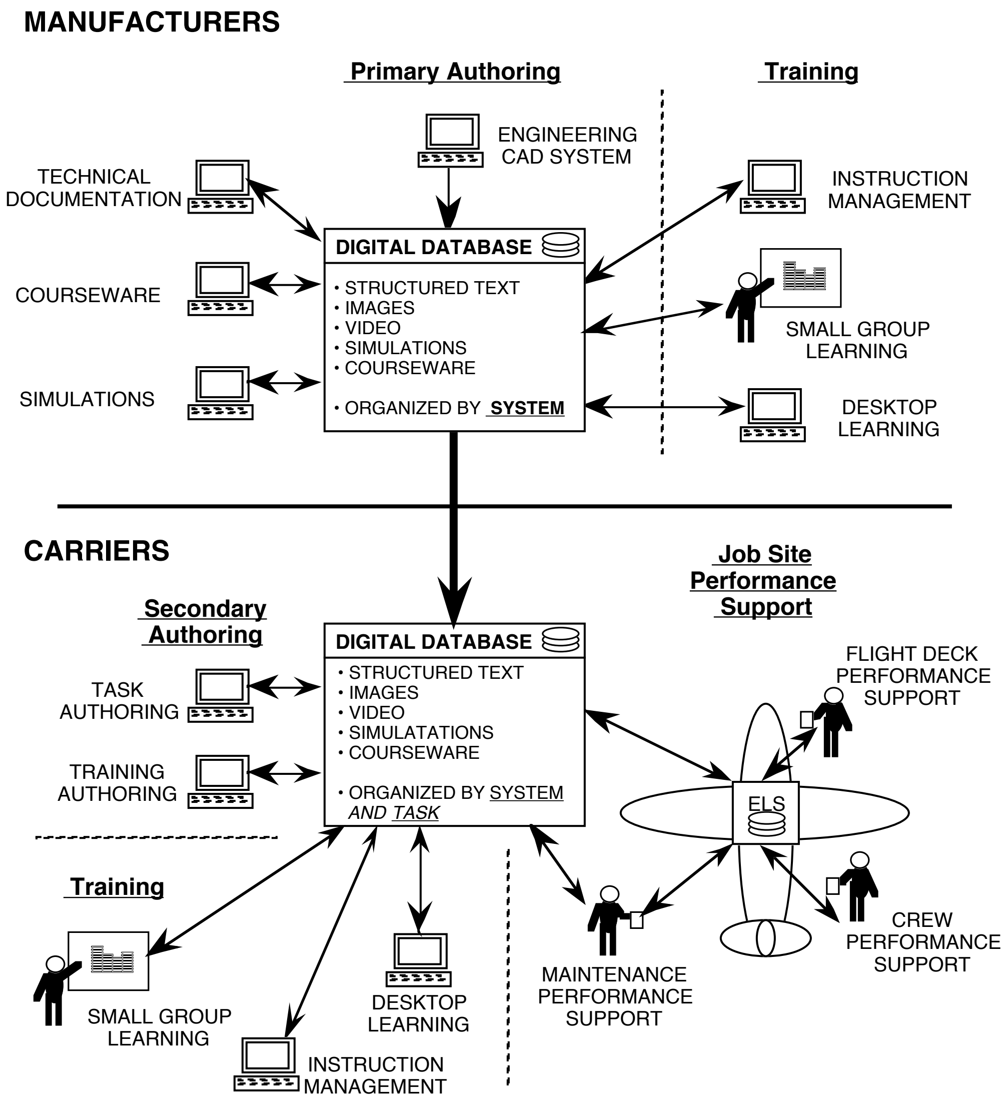


Figure **. Vision of future computer-based training in aerospace

Figure ** illustrates our vision of the future of computer-based training in aerospace:

1. Primary authoring. from engineering CAD, technical documentation, courseware, accounting, and simulation software.

2. Integration in a digital database. Airframe manufacturers are working toward the concept of digital repositories integrating data. These efforts are motivated by the desire to enhance configuration control, change management, and reuse of common data across multiple applications (e.g., use of CAD-generated data in courseware and simulation; use of courseware-generated data as part of technical documentation).

3. General system familiarization training. Reuse of data organized by system for general familiarization training in various settings. Shared digital data is generally organized by system in a manner that allows it to be straightforwardly used as part of general familiarization training courses.

Carrier's perspective:

4. Secondary authoring.

5. Carrier-specific and task-specific training.

6. Job-site performance support.

When AICC members identify a common interest in a strategic technology requirement, they may take steps to initiate what is called a *Sisyphus project* (see attached Sisyphus project criteria).[4] Technology requirements motivating the project are stated clearly, in a manner that enables technology providers to solve the same problem differently, or to each tackle different aspects of the same problem. Project team members agree to share results and work toward agreement on open standards. Projects report on their activities regularly at AICC quarterly meetings, organizing demonstrations of new technology and distributing technical reports. The output of a Sisyphus project typically becomes input for future AICC guideline creation activities.

Sisyphus projects address three issues:

Proof of Concept. Certain requirements can best be understood by developing and evaluating a proof-of-concept prototype. The goal of a such an effort is not to build a polished, operational system. Rather we want to demonstrate new potential solutions to identified problems, to discover the limitations of particular approaches, and to raise important *new* issues and questions. This creates a technology *pull* for CBT suppliers and technology providers based on authentic customer requirements.

Cost-of-Technology. Where appropriate, Sisyphus projects will address questions relating to the costs and benefits of new technology adoption, and will explore and demonstrate strategies and tools to minimize the effort and time involved in migrating existing courseware to take advantage of new capabilities.

Standards. Projects will emphasize early evaluation of proposed standards in real-world settings, to assure that inevitable problems of performance and completeness are made apparent before it is too late to do anything about them.

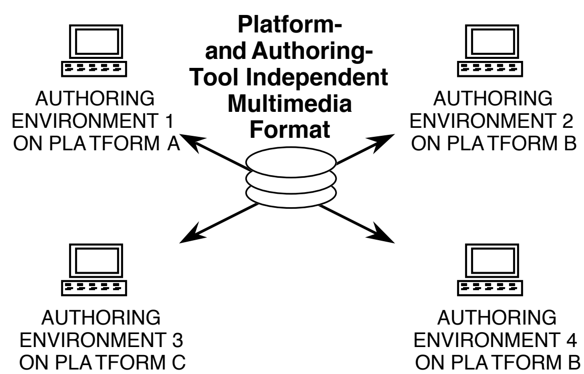
We will discuss three currently proposed Sisyphus projects: cross-authoring tool interoperability, cross-application interoperability, and intelligent interoperability.

4.1. Cross-Authoring-Tool Interoperability

[4] Similar Sisyphus projects have been organized for some years by the knowledge acquisition research community. Sisyphus, the mythical character who was condemned by the gods to continually roll a boulder from the bottom of a peak to the top only to watch it roll to the bottom again, seemed a fitting hero for the aspiring technological researcher.

As an initial step towards cross-authoring tool interoperability, several AICC members have expressed an interest in participating in a Sisyphus project structured around the topic of platform-independent and authoring-tool independent multimedia. We are actively soliciting the participation of technology providers who are willing to cooperate in developing a viable approach to this problem. We will work in cooperation with the Interactive Multimedia Association (IMA) and similar organizations toward the eventual goal of defining international consensus on relevant standards. These *de jure* standards will be likely be built on top of *de facto* industry standards such as Microsoft's AVI (audio-video interleaved) specification and Video for Windows tools and drivers, Apple's QuickTime for the Macintosh and Windows, IBM's Multimedia Presentation Manager/2, and multimedia extensions and utilities for X-Windows environments being developed by several groups including MIT, HP, DEC, and Gain Technology (Cole, 1993).

A group of AICC members have formed an initial project addressing some of these issues. Apple Computer and Kaleida (a joint venture of Apple Computer and IBM) have agreed to support an evaluation of ScriptX, an object-oriented extensible multimedia scripting format. ScriptX permits authoring and playback of multimedia on Macintosh, IBM-Compatible, and Unix platforms, as well as handheld and consumer electronic devices. It will provide the basis for a cross-platform standard that can be used by multiple developers of authoring tools. Cross-platform support is provided by defining a set of hardware- and software-independent integrated media data types and formats and a common way of describing the logical structure of the presentation. The object-oriented nature of the language allows differences between platforms to be abstracted, so that migration to a more capable platform becomes more a matter of detailing than reinvention.



Apple's Open Scripting Architecture allows ScriptX to be integrated with other forthcoming scripting languages such as AppleScript (a user-scripting system for application integration) and TeleScript (a communications scripting system jointly announced by General Magic, AT&T, Motorola, Sony, Philips, Matsushita, and Apple). Scripts are stored in a compiled, coded format, making it possible for them to be editable in foreign languages or in forms resembling other programming languages.

As part of the project, developers and users of several authoring tools across different platforms are being encouraged to evaluate the performance of ScriptX as a common multimedia format. One might imagine, for example, a demonstration of multimedia CBT courseware being authored in one tool on a given platform, played back using another tool on a second platform, and edited using a third tool on yet another platform.

One important component of the study will be the development of extensions to ScriptX to allow it to interpret graphics in formats that are being adopted as standards within the aerospace industry, such as ISO computer graphics metafiles (CGM) for vector drawings and CCITT tagged image file format (TIFF) bitmaps. The project will also explore ScriptX extensions to allow it to interpret documents prepared using the Hypermedia/Time-based Structuring Language (HyTime), an international standard with facilities for representing static and dynamic information that is processed and interchanged by hypertext and multimedia applications (ISO, 1992). HyTime is an application of the Standard Generalized Markup Language (SGML, ISO 8879), which is in wide use in the aerospace industry.

Another important component will be the participation of CBT suppliers, developers, and tool providers in assessing the cost and benefits of conversion of software and courseware to the new formats.

4.2. Cross-Application Interoperability

Over the next ten years, we expect the convergence of advanced documentation and knowledge representation tools. Current standards efforts in areas such as interactive electronic technical manuals (IETM) (Fuller & Rainey, 1992) will culminate in the development of a wide variety of *knowledge media,* computational environments in which explicitly represented knowledge serves as a means of communication among people and their programs (Glicksman, Weber, & Gruber, 1992; Gruber, Tenenbaum, & Weber, 1992). Knowledge media will integrate the functions of design documentation, operations and maintenance manuals, and computer-based training tools. They will transcend the artificial boundaries between documentation, modeling, database, and hypermedia applications, freely incorporating not only static drawings and graphical animations, but also simulations and full working models (Bradshaw & Boy, 1993).

With the increased integration of software will come advances in miniaturization allowing a single hardware device to combine entertainment, computing, and communications functions (Glaser, 1992). We expect computing to become "ubiquitous" as microprocessors become an integral part of office and home surroundings, and as inexpensive notebook and handheld units with wireless communication systems proliferate (Ryan, 1991; Tesler, 1991; Weiser, 1991).[5] Heads-up display systems will become routine in areas such as manufacturing, where they will allow workers to "see through" a blueprint that appears superimposed on the actual work surface. The source data will be accessible to a wide variety of computing devices in a multiplicity of formats. Docking bays will allow smaller devices to be easily connected to larger ones. New architectures to support the relay of data between mobile devices, desktop systems, and servers will also be developed. Examples of the precursors to these new architectures supporting roving users include Apple's *client-client-server* concept and Bellcore's *Touring Machine System Model* (Arango, Bahler, Bates, Cochinwala, Cohrs, Fish, Gopal, Griffeth, Herman, et al., 1993).

[5] Industry forecasters estimate an installed base of 100 million small computers and 1 billion intelligent digital devices by 1999.

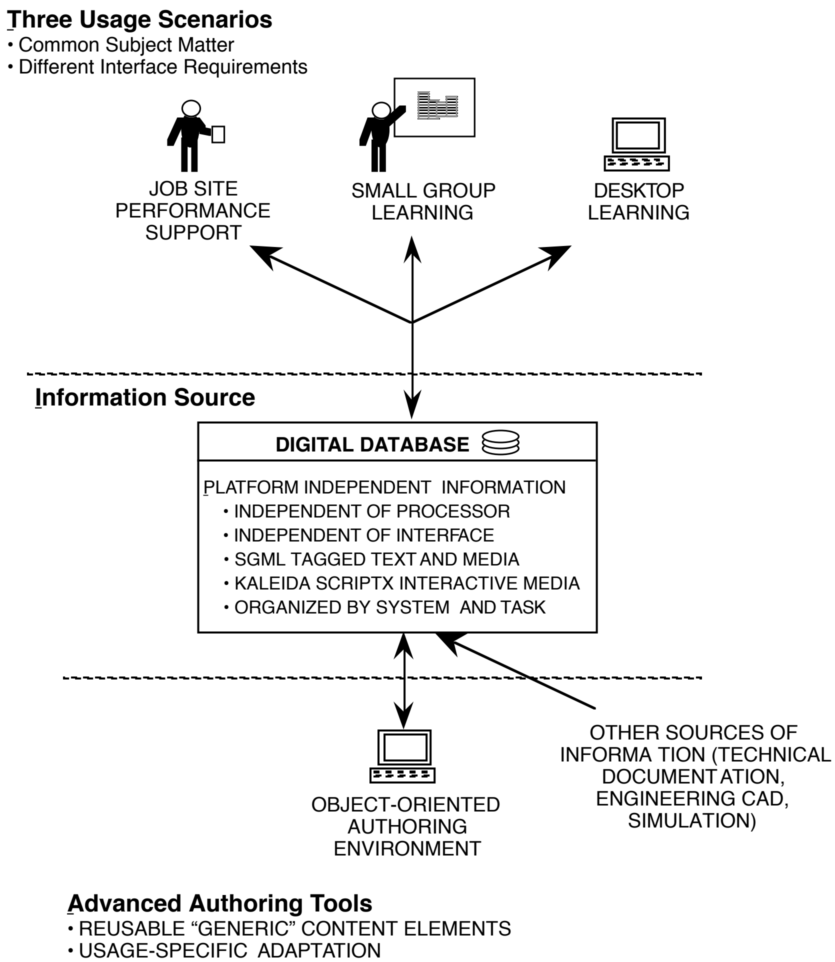
Researchers in CBT have begun exploiting advances in integration and miniaturization through the development of prototypical performance support systems (PSS). The purpose of a PSS is to enhance the performance of workers in the field by making training and crucial data instantly available to them, wherever they happen to be (Gery, 1991). One of the most active areas of research concerns how to better train and support those involved in essential aircraft maintenance activities, until recently a relatively neglected area (Marx, 1992). In the future, aircraft maintenance mechanics might carry portable devices with them to the hangar floor to support more reliable troubleshooting and quicker dispatching of the plane. Instead of volumes of fault isolation manuals,[6] the engineer would have access to required information about faults, parts-on-hand, and diagnostic tests through a wireless computer network.[7] Data about the status and history of the plane systems can be downloaded to the device from portable maintenance access terminals located on the plane itself. Integrated computer-based training tools would provide interactive demonstrations of required maintenance procedures as a form of “just-in-time” training. We believe that by the end of the decade CBT will be as likely to be delivered piecemeal on the job as part of recurrent training and performance assistance as on the classroom desktop during a dedicated training program.

There are a number of emerging standards for cross-application data exchange and object management that need to mature in order for PSS to reach their full potential. These include Apple’s Apple Events and Object Model, Microsoft’s Object Linking and Embedding (OLE), the Object Management Group’s Distributed Object Management Facility and Common Object Request Broker Architecture (DOMF, CORBA, supported by Sun and Hewlett-Packard among others), DEC’s Application Control Architecture (ACA), and Taligent’s object framework for system software and application components. For the purposessecondary to the more fundamental questions of how a PSS should be designed in order to be most useful in actual work settings.

For several years, The Boeing Company has been working in partnership with airlines and technology providers to understand and evaluate various aspects of the PSS concept. To pool resources and broaden the effort so that consensus on cross-application standards can be established, one of these projectthe *etc.* subcommittee is proposing a Sisyphus project to explore the implications of PSS technology for CBT suppliers, developers, users, and technology providers. The goal is to evaluate technical issues and cost for the entire PSS, from authoring and database tools to delivery vehicles to collaboration support and maintenance aids.

---

[6] A maintenance manual for a typical Boeing plane numbers close to 20,000 pages.
[7] One airline recently found that no-fault-found part removals in such situations cost them twelve million dollars over the course of a year.



4.3. Intelligent Interoperability

Now that some CBT systems are achieving sufficient interoperability to allow them to coexist comfortably on the same hardware and operating system configuration, users are beginning to demand the ability to mix and match lessons from different sources. The AICC activity to define computer-managed instruction (CMI) guidelines for interoperability has begun to address this requirement by producing an initial approach allowing sequences of lessons to be described and controlled programmatically.

But what about CBT developers and users who want to be able to describe and decompose a lesson into even smaller building blocks that can be reused as part of multiple lessons or even outside the CBT context? One airline customer of Boeing has lamented that things were much easier in the days when the airframe manufacturers delivered their courseware as boxes of overhead transparencies. At least then it was easy to select and reorder particular pieces of a lesson to adapt it to the airline's preferred method of instruction.

In the future, we expect the demand for adaptivity and intelligence to require course content and its description to become even more modular and finely-grained (Gruber, 1993; Rappaport, 1991). Eventually it will no longer be appropriate to think of "documents" and "lessons" as the right level of granularity for interoperability and adaptivity. Rather we will think in terms of a “knowledge soup,” in which small bits of information in a database can be assembled dynamically as appropriate to a given context (Neches, Fikes, Finin, Gruber, Patil, Senator, & Swartout, 1991; Sowa, 1990).

Such a capability will require even greater integration of CBT system components and the addition of increasingly sophisticated artificial intelligence capability. Second-generation Personal Digital Assistants (PDA) and Personal Communicators, with simple artificial intelligence enhancements and mothership-daughtership architectures, will give way to third generation Intelligent Personal Assistants (IPA), with sophisticated AI capabilities and fully-integrated wireless communication (Boy & Mathé, 1993). We expect CBT in such systems to become increasingly document-centric. Adaptive document systems will tailor the content and presentation of multimedia produced to the interests, abilities, and situation of the user (Bradshaw & Boy, 1993; Jansen & Bray, in preparation; Spohrer, Vronay, & Kleiman, 1991). Advanced human interfaces will exploit both built-in knowledge that shapes its “personality” for specific domains, as well as knowledge acquired through use in specific situations.

Brodie (1989) has discussed the need for *intelligent interoperability* in such future information systems. He defines the term to mean intelligent cooperation among systems to optimally achieve specified goals. A high level of interoperability requires knowledge of the capabilities of each system, so that task planning, resource allocation, execution, monitoring, and, possibly, intervention between the systems can take place. Ideally, a mechanism functioning as a global resource planner or intelligent agent would manage cooperation activities (Figure 2). In the CBT context, this can be envisioned simply as a more flexible and sophisticated CMI facility.

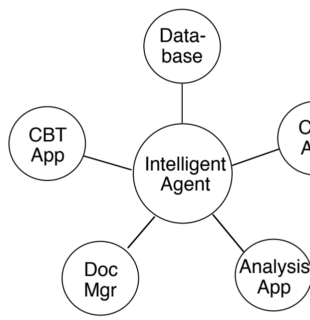


Figure 2. Cooperating systems with intelligent agent (adapted From Brodie, 1989).

While a single intelligent agent would be workable for small networks of systems, such a scheme would quickly become impractical as the number of cooperating systems grew. The activity of the global resource planner would become a bottleneck for the (otherwise distributed) system. A final step toward intelligent interoperability would be to embed an intelligent agent within each cooperating system (Figure 3). Applications would ask their intelligent agent for the needed resources, thus providing a level of encapsulation at the planning level analogous to the encapsulation provided in many existing applications at the level of data exchange protocols.

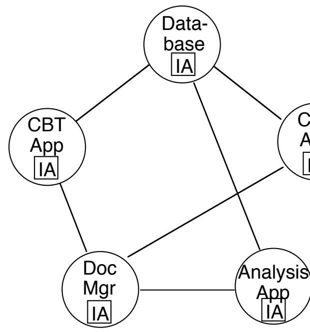
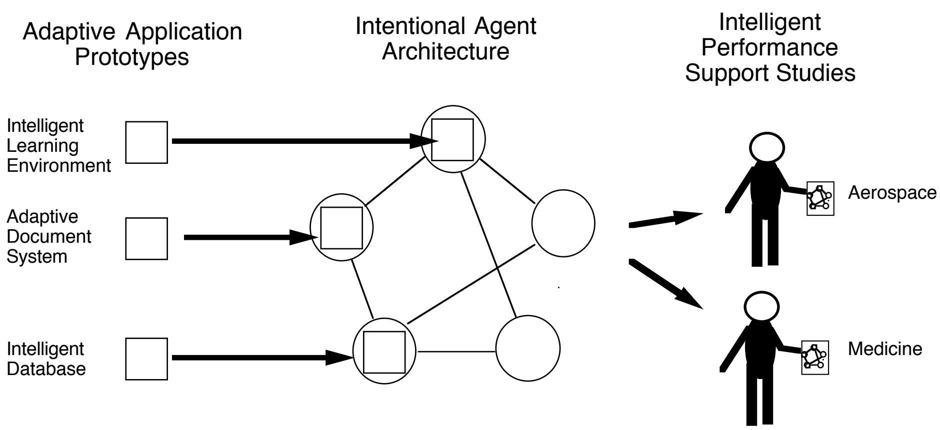


Figure 3. Cooperating systems with distributed intelligent agents (adapted From Brodie, 1989).



Intelligent performance support within an intentional agent architecture

Adaptivity and context sensitivity are key markers of intelligence
for future applications on mobile devices

- Mediating knowledge representations for constructive learning
- Explicit models of student, domain, instructional knowledge
- Uncertainty management

- Situation Recogntion and Analytical Reasoning Model
- Context-dependent background learning mechanisms
- Tags capture semantics, not just structure

- Object-oriented multimedia databases
- Intelligent information retrieval
- Sharing and reuse of ontologies

Convergence of advanced documentation, database, simulation, knowledge representation, and learning tools will
culminate in the development of a wide variety of knowledge media in support of specific user tasks

What needs to go into a CBT system in order to make it intelligent? What are the benefits and drawbacks to intelligent courseware in particular situations? Research in intelligent learning environments (ILE) addresses many of these questions and highlights differences between the traditional CBT approach and what will be required in future systems (Burns, Parlett, & Redfield, 1991; Clancey & Soloway, 1990; Larkin & Chabay, 1992; Polson, Richardson, & Soloway, 1988).

The distinction between traditional CBT and ILE mirrors that between classroom lecturing to large groups of students and one-on-one tutoring. In the former situation, the teacher attempts to meet the instructional needs of as many students as possible, making sweeping assumptions about each student's understanding of the subject matter, and presenting material in a rigid sequence. In the latter, the instructor focuses entirely on one student's needs, tailoring the material precisely to the student. At the University of Washington, Hunt and colleagues (Hunt & Minstrel, 1992; Levidow, Hunt, & McKee, 1991) have developed a set of instructional methods and interactive computer-based tools based on a theory of student knowledge organization. This approach has proven effective for an application in physics and is now being extended by Madigan and Hunt for use in a statistics domain (Madigan & Donnell, 1992).

O'Neill, Slawson, and Baker (1991) identify three important components of intelligent computer-supported training environments: (1) a *domain model,* which represents what the student has to learn; (2) a *student model,* which represents the current state of the student's subject area knowledge; and finally (3) an *instructional model,* which represents techniques for applying the right instructional tactics at the right times. Additional components might include (4) a *domain simulator* to make the learning environment similar to the real environment (Hollan, Hutchins, & Weitzman, 1984); (5) an *intelligent user interface,* based on principles from human factors and cognitive science research, to manage communication between the student and the ILE (Bradshaw & Boose, 1992; Fairweather, Gibbons, Rogers, Waki, & O'Neal, 1992; Norman, 1992) (6) a *context management system* to integrate and adapt information from the domain, student, and instructional models to a particular situation (Boy, 1991; Boy & Mathé, 1993); (7) an *uncertainty management system* to deal with the problem of predicting the outcome of teaching interventions and the state of the student's knowledge (Madigan & York, submitted for publication); and (8) a *concept management system* to formulate, communicate, and draw inferences about model semantics (Bradshaw, Holm, Boose, Skuce, & Lethbridge, 1992). A possible architecture for such a system is illustrated in Figure 4. As components of ILE architectures become better understood, the interfaces between them will themselves become subjects for standardization efforts.

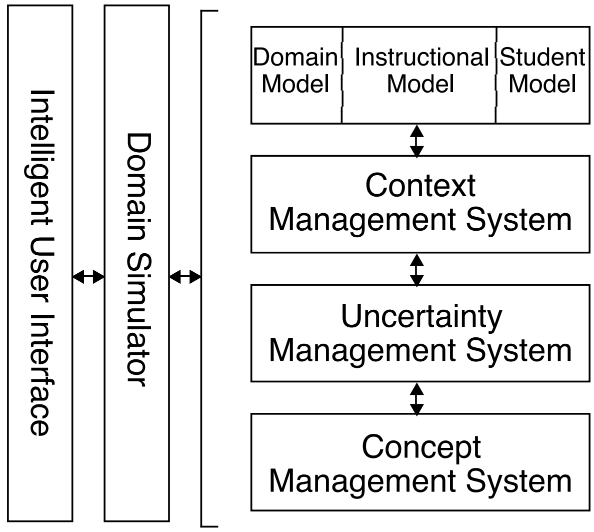


Figure 4. A proposed high-level architecture for an intelligent learning environment.

Under the auspices of the AICC, we are proposing a third Sisyphus project to build a prototypical intelligent tutoring system. Work on the project will be coordinated by the European Institute for Cognitive Sciences and Engineering (EURISCO). EURISCO was recently created by a European consortium of universities, research, aerospace, and industrial organizations to foster and develop focused research in cognitive sciences and engineering. Among its main research topics are CBT, cognitive aspects of human factors, knowledge acquisition, and multimedia. Several of the researchers mentioned in the previous two paragraphs have already agreed to participate (Boose, Boy, Bradshaw, Fairweather, Madigan). The goal of the project is to determine the appropriate building blocks for intelligent authoring tools and runtime environments.

5. Conclusion

The application of advanced technology to CBT provides a unique proving ground for a host of theoretical and practical issues. Concepts and approaches to achieving interoperability will succeed or fail as they meet the diverse and demanding requirements of the aviation industry. It is encouraging to witness the resolve shown by fierce competitors to increase levels of trust and eliminate artificial barriers to cooperation that frustrate and alienate customers (Davidow & Malone, 1992). With such a view of enlightened self-interest, everyone wins.

References

AICC (1992). *Glossary of Terms Related to Computer-Based Training*. Document No. CRS003. Aviation Industry Computer-Based Training Committee Courseware Technology Subcommittee, 18 November.

Arango, M., Bahler, L., Bates, P., Cochinwala, M., Cohrs, D., Fish, R., Gopal, G., Griffeth, N., Herman, G. E., Hickey, T., Lee, K. C., Leland, W. E., Lowery, C., Mak, V., Patterson, J., Ruston, L., Segal, M., Sekar, R. C., Vecchi, M. P., Weinrib, A., & Wuu, S.-Y. (1993). The Touring Machine System. *Communications of the ACM*, 36(1), 68-77.

Boy, G. A. (1991). Indexing hypertext documents in context. *Proceedings of the Third ACM Conference on Hypertext,* (pp. 1-11). San Antonio, Texas, , ,

Boy, G. A., & Mathé, N. (1993). Operator assistant systems: An experimental approach using a telerobotics application. In K. M. Ford & J. M. Bradshaw (Ed.), *Knowledge Acquisition as Modeling*. (pp. 271-286). New York: John Wiley.

Bradshaw, J. M., & Boose, J. H. (1992). *Mediating representations for knowledge acquisition.* Seattle Washington: Boeing Computer Services, March.

Bradshaw, J. M., & Boy, G. A. (1993). Adaptive documents. *Submitted to the IJCAI '93 Workshop on Using Knowledge in its Context,* . Chambery, France, , , August.

Bradshaw, J. M., Holm, P. D., Boose, J. H., Skuce, D., & Lethbridge, T. C. (1992). Sharable ontologies as a basis for communication and collaboration in conceptual modeling. *Proceedings of the Seventh Knowledge Acquisition for Knowledge-Based Systems Workshop,* (pp. 3.1-3.25). Banff, Alberta, Canada, , ,

Burns, H., Parlett, J., & Redfield, C. (Ed.). (1991). *Intelligent Tutoring Systems: Evolutions in Design*. Hillsdale, N.J.: Lawrence Erlbaum.

Clancey, W. J., & Soloway, E. (Ed.). (1990). *Artificial Intelligence and Learning Environments*. Amsterdam: Elsevier.

Cole, B. (1993). Special report on interactive multimedia: The technology framework. *IEEE Spectrum*, March, 32-39.

Davidow, W. H., & Malone, M. S. (1992). *The Virtual Corporation: Structuring and Revitalizing the Corporation for the 21st Century*., New York: Harper-Collins.

Fairweather, P., Gibbons, A. S., Rogers, D. H., Waki, R., & O'Neal, A. F. (1992). A MODEL for computed-based training. *AI Expert*, December, 30-35.

Fletcher, J. D. (1992). *Courseware Portability*. IDA Paper P-2648. Alexandria, VA: Institute for Defense Analyses, August.

Fuller, J., & Rainey, S. C. (1992). The interactive electronic technical manual. *CALS Journal*, 1(4), 63-69.

Gery, G. J. (1991). *Electronic Performance Support Systems*., Boston, MA: Weingarten Press.

Glaser, M. B. (1992). *Portable Intelligence*. Report No. 811. SRI International, Winter.

Glicksman, J., Weber, J. C., & Gruber, T. R. (1992). The NoteMail project for computer supported cooperative mechanical design. *Working Notes of the AAAI-92 Workshop on Design Rationale Capture and Use,* (pp. 105-110). San Jose, CA, , ,

Gruber, T. R. (1993). Model formulation as a problem solving task: Computer-assisted engineering modeling. In K. M. Ford & J. M. Bradshaw (Ed.), *Knowledge Acquisition as Modeling*. (pp. 105-128). New York: John Wiley.

Gruber, T. R., Tenenbaum, J. M., & Weber, J. C. (1992). Toward a knowledge medium for collaborative product development. J. S. Gero (Ed.), *Proceedings of the Second International Conference on Artificial Intelligence in Design,* . Pittsburgh, PA, , ,

Hollan, J. D., Hutchins, E. L., & Weitzman, L. (1984). STEAMER: An interactive inspectable simulation-based training system. *AI Magazine*, 5(2), 15-27.

Hunt, E., & Minstrel, J. (1992). A cognitive approach to the teaching of physics. In K. McGilly (Ed.), *Classroom Lessons: Integrating Cognitive Theory and Classroom Practice*. Cambridge, MA: MIT Press/Bradford Books.

ISO (1992). *Information Technology — Hypermedia/Time-based Structuring Language (HyTime)*. ISO/IEC International Standard 10744, First edition. International Organization for Standardization, November 11, 1992.

Jansen, B., & Bray, G. (in preparation). Context, knowledge types, and serendipity. *Submitted to Avignon '93,* . , , , ,

Larkin, J. H., & Chabay, R. W. (Ed.). (1992). *Computer Assisted Instruction and Intelligent Tutoring Systems: Shared Goals and Complementary Approaches*. Hillsdale, N.J.: L. Erlbaum Associates.

Levidow, B., Hunt, E., & McKee, C. (1991). The DIAGNOSER: A HyperCard tool for building theoretically-based tutorials. *Behavioral Research Methods, Instrumentation and Computers*, .

Madigan, D., & Donnell, D. (1992). *VITAL: An intelligent tutoring system for statistics*. Unpublished Technical Report. Seattle, WA: University of Washington,

Madigan, D., & York, J. (submitted for publication). Bayesian graphical models. , .

Marx, D. (1992). Looking toward 2000: The evolution of human factors in maintenance. *Proceedings of the FAAHIS Conference on Human Factors Issues in Aircraft Maintenance and Inspection,* . Alexandria, VA, , , January 22-23.

Masie, E. (1993). It's all changing…. *CBT Directions*, January/February, 16-20.

Nance, B. (1991). Interoperability today. *BYTE*, 16(12), 187-196.

Neches, R., Fikes, R., Finin, T., Gruber, T., Patil, R., Senator, T., & Swartout, W. R. (1991). Enabling technology for knowledge sharing. *AI Magazine*, 36-55.

Norman, D. A. (1992). Cognitive artifacts. In J. M. Carroll (Ed.), *Designing Interaction: Psychology at the Human-Computer Interface*. (pp. 17-38). Cambridge: Cambridge University Press.

O'Neill, H., Slawson, D., & Baker, E. (1991). Design of a domain independent problem solving instructional strategy for intelligent computer assisted instruction. In H. Burns, J. Parlett, & C. Redfield (Ed.), *Intelligent Tutoring Systems: Evolutions in Design*. Hillsdale, N.J.: Lawrence Erlbaum Associates.

Polson, M. C., Richardson, J. J., & Soloway, E. (Ed.). (1988). *Foundations of Intelligent Tutoring Systems*. Hillsdale, N.J.: L. Erlbaum Associates.

Rappaport, A. (1991). A theory of local and interstitial knowledge. *Proceedings of the Sixth Knowledge Acquisition for Knowledge-Based Systems Workshop,* (pp. 21.1-21.18). Banff, Alberta, Canada, , ,

Ryan, B. (1991). Dynabook revisited with Alan Kay. *BYTE*, February, 203-208.

Smith, S. (1993). Cross-platform CBT development. *CBT Directions*, January/February, 10-15.

Sowa, J. F. (1990). Crystallizing theories out of knowledge soup. In Z. W. Ras & M. Zemankova (Ed.), *Intelligent Systems: State of the Art and Future Systems*. London: Ellis Horwood Ltd.

Spohrer, J. C., Vronay, D., & Kleiman, R. (1991). Authoring intelligent multimedia applications: Finding familiar representations for expressing knowledge. *Proceedings of the 1991 IEEE International Conference on Systems, Man, and Cybernetics,* (pp. 1751-1756). Charlottesville, VA, , ,

Tesler, L. G. (1991). Networked computers in the 1990s. *Scientific American*, September, 86-93.

Weiser, M. (1991). The computer for the 21st century. *Scientific American*, September, 94-

104.

Sisyphus Project Criteria

1. Any proposed solution should actually solve the problem, i.e. not just be a problem analysis or technology analysis.

2. The solution should be complete and correct in terms of specified requirements, i.e. the problem itself should be sufficiently well-defined for these requirements to be applied.

3. The cost and performance of the solution should be evaluated.

4. The sensitivity of the solution to data or technology changes should be discussed.

5. Project team members agree to share results and work toward agreement on open standards.

6. All of the above criteria are recommendations and specific approaches may violate them, but the reasons for this should be stated clearly and justified.