The World-Wide Web in Education

Daniel K. Schneider [*] Kayla Block [†]

June 12, 1995

published in ANDREA,
Volume 2 No 5, June 12, 1995,
A Network for Distance Education
Reporting from European Activities

Abstract

This article gives an overview on what can be done with the World-Wide Web in education. The discussion is based on a short technical description of the World-Wide Web and some theoretical considerations on teaching and learning.

Contents

1 Technical Aspects of the WWW, the Integrator of "Cyberspace"

In order to discuss "when the Web" works in education, let's recall shortly "how the web" works. The World-Wide Web was conceived by Tim Berners-Lee at CERN in 1989 as an information integrator within which all available information on the Internet could be accessed in a simple and consistent way on every kind of machine architecture. A standard WWW browser (i.e. the client program for the WWW) can access at least the following communication protocols: HTTP (WWW's

* TECFA, School of Psychology and Educational Sciences University of Geneva
† Computer Consultant, Los Angeles

Hypertext Transfer Protocol), FTP, NNTP, WAIS and Gopher. Central for information retrieval is the Universal Resource Locator (URL). A URL (e.g. "http://tecfa.unige.ch/welcome.html") is composed of a protocol indicator (e.g. "http"), an Internet machine name (e.g. "tecfa.unige.ch") and a file name (e.g. "welcome.html"). Usually the file name stands for a document to be retrieved. Sometimes, a program will be launched, e.g. "http://www.ucc.ie/htbin/acronym" will launch a program for looking up Internet Acronyms). Note that in principle any networked software can be turned into an HTTP server and this is important for future educational WWW-based systems.

The standard WWW information vehicle is a hypertext document, a text file encoded with the so-called "Hypertext Mark-Up Language" (HTML[1]). An HTML file will display on your screen according to special mark-up commands and according to the settings and capabilities of your client-program. Special highlighted buttons in a displayed "page" allow retrieval of other pages from anywhere in the world (including any kind of file formats that your setup can handle).

The server tells the client what file type is accessed. When a file format is met that the browser does not understand (e.g. a Postscript, a VRML or a Toolbook file), a "helper" application is called, if it is available on your machine. You can configure your client to use any kind of program to display any multimedia image, sound, video and data format. Note also, that most WWW clients can display in-line images. HTML is an evolving language: The latest version (HTML 3) allows for quite sophisticated layout (e.g. tables, text floating around images).

Forms are an important feature of HTML. They allow a server to query the user with a few standard graphic user interface (GUI) widgets like push-buttons, radio-buttons, text editing window and scrolling lists. This information can then be processed by the server. It makes up a powerful query-interface to various kinds of data bases, but it can be used as interface to any program running on a server. Because HTML pages can be generated dynamically by a server, such pages can be tailored according to the needs of the user, which is an interesting feature for educational systems. Another feature is interactive maps which report to the server the position of mouse clicks. Such maps can be used to build user navigation aids.

Given this technical description, the WWW can be characterized with several functionalities:

1. A Knowledge Integrator
2. A distributed Hypertext
3. An Interface to any kind of remote program
4. An Interface to certain local programs

For producing HTML, there are 4 basic strategies:

1. Using a simple text editor (inserting html tags manually),
2. Using a "smart" text editor or an HTML tool that includes special editing extensions for automatically inserting HTML tags (like HTML assistant for Windows; or an "HTML tool" like HotMetal)
3. Using "filters" (translation programs that translate from text or other formats into HTML)
4. Using programs that create HTML "on the fly" (a feature frequently used to display output resulting from a data base request, on-line questionnaires and assessment programs).

Option 1 is not very productive, Option 2 is best for producing short Hypertext material, but using those tools requires some training. Option 3 is a good solution for producing larger document structures that also can be printed. However, currently these filters are not powerful enough (especially in the world of micro-computers).

We can distinguish several levels of WWW Use in Education. In order of difficulty they are:

1. The Web as information tool: Curricula and courses information, etc.)

[1] http://www.w3.org/hypertext/WWW/MarkUp/MarkUp.html

2. Distribution of learning material: E.g. books (in various formats like postscript, word-processor, etc.), programs, applications needing a specific interpreter/language for the client

3. Collaboration tools

4. Interactive educational applications:

The realization of level 3 and level 4 WWW-interfaced applications are hindered both by the design of the WWW and the fact that advanced programming knowledge is required.

Recently a few interesting new developments can be observed in the WWW world, like: HotJava (a WWW client that can execute programs), WWW-MOO interfaces, VRML (virtual reality mark-up language) interfaces, Netscape's "server-pushes", Educational packages for CGI-scripts. These extensions and additions will add additional power to the WWW and increase its popularity.

2 A Note about Teaching and Learning

What teaching and learning functions can the WWW provide for what educational setups? Before we answer that question let's have a closer look at "learning" and "teaching":

What is learning? In short:

1. One learns by doing something (psychology)

2. One learns by pursuing an instructional goal (education)

For the Psychologist, learning is a complex phenomenon AND there are several types of learning. Learning needs external "conditioning" (behaviorism); it is related to active problem solving; it involves integration, construction and compilation of new content (cognitivism) is constrained by human cognitive capacities (e.g. how much hypertext can we handle?). There are several kinds of learning: e.g. Factual information, Concepts, Reasoning, Procedures, Problem Solving,...) One does not learn by browsing, we need a variety of learning tasks, not just exploration (see [Kearsley et al., 1993]).

Teaching can be characterized by 2 aspects:

(1) Teaching "setups": e.g. Distance Teaching, Open Learning, Semi-Distance Teaching, Traditional Class room teaching, etc. Each of those setups needs different instructional strategies and tactics.

(2) Instructional processes that can be looked at from many points of view. Let's have a short look at normative instructional design theory: What is the optimal sequencing of course-ware and how is it related to various types of learning? Gagné suggests nine universal steps of instruction (cf. [Gagné, 1985] or [Aronson et Briggs, 1983]) which should be found in any instructional context:

1. Gain attention: e.g. present a good problem, a new situation, use a multimedia advertisement.

2. Describe the goal: e.g. describe the goal of a lesson (task,...), state what students will be able to accomplish and how they will be able to use the knowledge, give a demonstration if appropriate.

3. Stimulate recall of prior knowledge: e.g. remind the student of prior knowledge relevant to the current lesson (facts, rules, procedures or skills). Show how knowledge is connected, provide the student with a framework that helps learning and remembering. Tests can be included.

4. Present the material to be learned: e.g. text, graphics, simulations, figures, pictures, sound, etc. e.g. follow a consistent presentation style, chunking of information (avoid memory overload, recall information)

5. Provide guidance for learning: e.g. presentation of content is different from instructions on how to learn. Should be simpler and easier than content. Use of different channel.

6. Elicit performance: "practice", let the learner do something with the newly acquired behavior, practice skills or apply knowledge

7. Provide informative feedback: show correctness of the trainee's response, analyze learner's behavior (or let him do it), maybe present a good (step-by-step) solution of the problem

8. Assess performance: test, if the lesson has been learned. Also give information on general progress

9. Enhance retention and transfer: inform the learner about similar problem situations, provide additional practice. Put the learner in a transfer situation. Maybe let the learner review the lesson.

Instructional Design Theory provides a detailed prescription on how to organize teaching and learning at the global (curricula), lesson and task level. Most work is also grounded in some learning theory. The practical use of those approaches (despite or maybe because their level of detail) is often debated. Some argument against reading much instructional design theory is that a good teacher with good practice intuitively knows and uses things like Gagné's steps. Most people agree that instruction needs principles, however some researchers feel that instructional theory should not just be grounded in learning theory but BE applied learning theory and implement optimal learning conditions according to what we know about learning. This is the way most research in Advanced Learning Systems operates.

3 The Learning and Teaching Environment

Teaching and Learning "meet" in the teaching and learning environment. Building a good learning environment means taking into account both the psychologist's and the instructionalist's perspectives.

Advanced Research on Learning Environments (mostly in the field of artificial intelligence and education, see [Wenger, 1987]) can give us some insights on how to build a good learning and teaching environment. An Advanced Learning Environment refers to a category of educational software in which the learner is 'put' into a problem solving situation. A learning environment is quite different from traditional course-ware based on a sequence of questions, answers and feedback. The best known example of a learning environment is a flight simulator: the learner does not answer questions about how to pilot an aircraft, he learns how to behave like a "real" pilot in a rich flying context. Experience with learning environments (like LOGO) showed that those systems gain efficiency if the learner is not left on his own but receives some assistance. Learning is also increased when two or more learners work together on a problem. This assistance or co-learner function may be provided by humans or by some system components. In our flight simulator example, the future pilot would gain from discussing his actions with an experienced pilot. In advanced experimental learning environments, the implementation of these agents is based on artificial intelligence techniques. In summary, we use the word 'intelligent learning environment' for learning environments which include (1) a problem solving situation, (2) a reflective architecture that shows the students what he is doing or even helps him to actively structure his cognitive activities, (3) one or more agents that assist the learner in his task and monitor his learning and maybe (4) a co-learner.

Advanced Learning environments are difficult to implement and currently they are not cost effective. However, ideas from this type of research have to be taken seriously and a very important lesson is that new media should be used in a creative way. There is not much point in investing time and money into computer technology that replicates what can be done as well with traditional means. Now lets look at a few postulates that can be found in such research:

- The Learner must be active (again and always: people don't learn by browsing hypertext and by answering questions!)

- A learning environment should be designed to be as powerful as dedicated working environments. It must be rich and complex reflecting the essential properties of what has to be learned.

- The environment must be structured. If the richness of a learning environment is a quality, its complexity may reduce learning. It must provide optimal learning conditions as a function of the learner's stage of knowledge.

- Learning environments should be designed as hierarchical knowledge base generators

- Learning environments should present knowledge as a communication system. A learner must interact with agents, tutors.

Of course, such an environment is difficult to implement on the computer and even more over the network. But educational setups can be constructed with those principles in mind! Now let's have a more "distant" look at the concept of "learning environment".

In discussing the role of technological support in education, [Sandberg, 1994, 225] identifies the components of a (technologically rich) learning environment. These components must all be there in order to optimize learning. However, they can be "implemented" in many different ways. Each component has functionalities that we should insure:

1. "Teacher" component: Its role is to provide something between loose guidance and direct instruction. It can be a human agent (present or distant), an intelligent agent, instructions like some text books provide, etc. This component provides information from the syllabus to the task level.

2. "Monitor component: Ensures that something is learned. A role taken by either the human teacher, the learner (self-control) or by some program.

3. "Fellow learners" component: Improves the learning process (some research tries to implement artificial ones).

4. "Learning material": Contains what has to be learned in a very broad sense (knowing what, knowing how). It can be computational in various ways (exploratory hypertext, lesson and task oriented hypertext, simulation software, task solving environments, etc.).

5. "External information sources": All kinds of information which is not directly stored in the learning material (e.g. additional material, handbooks, manuals, etc.).

6. "Tools": Everything which may help the learning process other then the learning material (e.g. calculators, communication software, etc.)

7. "School" [a category we added]: Something that provides a curriculum.

4 What is Course-ware?

For many educational technologists "course-ware" is the production of computerized learning materials that would fall into one of the following categories:

1. Programmed Instruction (transfer of content proceeds step-by-step)

2. Computer Assisted Instruction (Drills and Tutorials)

3. Intelligent Computer Assisted Instruction (ITS Tutorials)

4. Computer Based Learning (Simulations, Hypertext and Micro-worlds)

5. Intelligent Learning Environments (Micro-worlds + tutors, helpers, experts)
6. Cognitive Learning Support Environments (some hypertexts)
7. Knowledge Construction Environments

Currently, The Web and other on-line tools fit in rather poorly (except as far as the Webs distribution quality is concerned). The potential of the Web should be addressed in a more general "course-ware perspective", understood as:

1. optimizing access to educational "information" via an appropriate interface and structuring of the material
2. implementation of instructional strategies, i.e. sequencing of teaching materials
3. implementation of instructional tactics, e.g.: giving examples, multiple choice questions, asking the student to perform a task, telling what learning strategy to adopt with some material, etc.

In this perspective, the learning material contains what has to be learned in a very broad sense (e.g. knowing what, knowing how). It can be computational in various ways (exploratory hypertext, lesson and task oriented hypertext, simulation software, task solving environments, etc.)

The Web has potential for new pedagogies: For teachers, the focus shifts from "information transfer" to "organization of information access" + "organization of collaboration".

5 Who needs the Web and what are the Difficulties?

Who needs the Web (and the Net in general)? Since the WWW is becoming a de-facto standard as information tool on the Internet, frequently cited candidates are Distance Teaching Institutions. Such institutions do in principle have the financial resources for producing WWW course-ware. However student's access to the Web has to be insured! Arguments against the WWW are the quite staggering re-organization costs and the Web's technical gaps. "Just in Time Open Learning" Providers face the same issues.

Semi-Distance curricula that combine intensive course modules in some location with distance learning (e.g. TECFA's postgraduate diploma) are more flexible in that respect. Such diplomas are a rather new phenomenon and can be constructed with the possibilities that the Internet offers in mind. Instructors can produce up-to-date learning materials. However, this means additional work for them (at least in the short term) and coping with the lack of experience with Distance Teaching. Similar candidates are single Distance Teaching Courses offered by regular institutions. Such courses will become increasingly popular due to increasing specialization and shrinking of funding. However similar problems have to be faced, i.e. lack of time and training.

"Classical" Institutions can use the WWW to enhance Class Room teaching and try out new pedagogies, e.g. facilitate information instead of giving ex-cathedra lectures, have students collaborate in exploratory research, etc.

One of the most popular uses of the WWW is Departmental Research and Training Information. Such servers are quite easy to implement, but getting all department members involved and having them UPDATE their material can be a real problem.

All Institutions can of course use the WWW to build a Campus-Wide Information System. The only need here is "organizational planning" and at least one full-time information system manager!

Very generally speaking, five main aspects need to be considered before planning involvement with WWW technology:

1. Technological aspects (e.g. handling filters for generating HTML, Server-side programs, Data-Bases, etc.)
2. Organizational aspects: Who is going to do the extra-work? How can we train all involved persons?

3. Pedagogical/psychological aspects: How can we produce good course-ware on the Net? How can we build full teaching and learning environments?
4. Media aspects: How do we "mediatize" existing educational content? (This is an important issue for Distance Teaching institutions)
5. Cost: Internet access is expensive (rural areas!), producing WWW material can be quite time-consuming (including instructor's training), the students themselves do need some basic "Internet training" too.

6 Education on the Web

(1) Because of its hyper media and distance delivery capacities, the Web has certainly shown its potential for building "EDUCATIONAL INFORMATION SYSTEMS" in a larger sense. We can distinguish among:

1. Campus-Wide Information Systems. Some universities already do pretty well, the ones that don't lack probably one (or more) full-time professionals
2. "Just in Time Open Learning" Information
3. Departmental Education and Research. This work is often coordinated by few persons and it needs involvement of all members
4. Courses syllabus (program, grading, exercises, etc.): Needs training of all members
5. Distribution of learning material

There are some difficulties (e.g. all necessary tools need to be found and installed, people need to be trained, updating needs to be insured) but they are mostly "social" and "organizational" ones.

(2) As easy on the technical level, EDUCATIONAL (NON-INTERACTIVE) HYPER MEDIA on the Web is the other popular use of the WWW in education. HTML can be used to produce on-line text-books, "museums and exhibitions", manuals etc. Such material can be produced for a single class only or it can be shared over the network. The main problem with this use is that hypertext is not always the best vehicle to produce study material that is mainly textually based. Information on the screen is very hard to read (more so on the kind of 15" screens most students have). Furthermore, orientation is easier in a book. Such material is best used for exploration or for accessory information (manuals, "further readings", etc.)

(3) INTERACTIVE FORMS pages start enjoying popularity for tests and quizzes used for both assessment and self-assessment. This technology (questions and answers) also can be used to produce a certain type of learning material, but note that both assessment by questionnaires and Q&A based learning software do not provide central functionalities for most types of learning and instruction. However, combined with good hypertext "books", some results can be achieved.

More generally, it is not really productive today to produce interactive software on the Web. For people with little or very specialized training in computer science it is not easy to write scripts that handle forms and interact with other software on the server (see [Ibrahim et Franklin, 1995]). The statelessness of the http protocol (meaning that the connection to a server is closed after a requested document is delivered) adds additional difficulties. Note that a certain amount of code for writing interactive scripts is available, e.g. "Zot-Dispatch"[2]. It is more efficient to use a "classical" educational software generator such as Toolbook and distribute Toolbook files over the network, that can then be interpreted locally. The technical requirements for such a setup are not overly difficult (see for instance: Multimedia World Wide Web PC: How to distribute interactive applications on the Internet[3]).

[2]http://www.oac.uci.edu/X/W6/forms/zot-dispatch/ZD-Doc.html
[3]http://www.univ.trieste.it/ nirital/lughi/strumlav/mmwwwpc/mmwwwpc.html

(4) The WWW can also be used as interface to scientific data, papers, on-line journals and so forth. Here lies a great potential for higher education that is not yet fully exploited. Those scientific resources are not necessarily structured in a way that students can use them. In most cases, teachers have to have prepare structured text to facilitate access. At the same time, kiosk modes available in some browsers allow for the suppression of the menu bar, which disallows random access to the Web as a whole, helping to maintain a tighter focus of learning. In some cases, it can be useful to ask students to produce WWW "index" pages. Such an exercise will help them to learn navigation and "mining" of interesting things.

(5) RECENT TECHNOLOGICAL DEVELOPMENTS will increase the INTERACTIVE POTENTIAL of the Web:

1. Newly proposed Client Communication Interfaces (CCIs)[4] allow more sophisticated interaction between a local WWW browser and a client program than just launching an external program for a given file type. The drawback of such technology is that external clients need to be installed first (like in the Toolbook example mentioned above).
2. HotJava[5] from Sun Micro Systems (and recently licensed to Netscape) takes a different approach. The client itself (currently only running on Unix workstations) will be able to interpret programs sent over the network written in the "Java" language.
3. VRML[6] and other 3-D Interfaces: e.g. the W3 Kit[7].
4. Educational "cgi-bin" add-ons will facilitate production of interactive forms WWW pages. An example is WEST[8]
5. Interfaces to text-based virtual realities (see below) or other specialized HTTP servers (e.g. see [Mallery, 1994])

The general issue with WWW-based educational material is as always, "productivity versus power". Currently, powerful interactive WWW applications are hard to produce.

7 Collaboration on the Net

Two basic claims can be made about the collaborative character of the Network:

1. With networked computers, teachers become rather: "information organizers", "tutors", "facilitators" instead of "information transmitters".
2. The Network favors collaborative work, which in turn favors learning (see [Dillenburg et Schneider, 1995])

As it stands now, the WWW is not very well suited for interactive collaborative work. However, remember that the WWW has been designed initially as a collaboration tool between researchers and it is easy to use it in this spirit:

- Organizing better information exchange among students (who does what, students have to turn in html pages)
- Distribution of work to be shared Students have to coordinate over the network
- Co-writing of text Students have to co-produce over the network
- Discussion and Conferencing Students have to comment text and argue about ideas

[4] http://yahoo.ncsa.uiuc.edu/mosaic/cci.spec.html
[5] http://java.sun.com/
[6] http://www.sdsc.edu/vrml
[7] http://www.geom.umn.edu/docs/W3Kit/W3Kit.html
[8] http://west.ucd.ie/

A certain amount of WWW based conferencing and annotation software currently exists (e.g. see the "WWW Collaboration Projects"[9] page). Asynchronous collaboration software can easily be built upon the WWW and its cgi-bin extensions. However, currently most of that software is still a bit experimental and it needs to installed by a knowledgeable person. But educators can use the WWW in conjunction with mail, mailing-lists, Usenet News and so forth. Prototypes for synchronous communication such as Web-Chat exist, but besides being experimental they are rather slow.

One of the most exciting recent developments for education are WWW interfaces to MOOs, a variant of text-based virtual worlds.

> "A MOO is a network-accessible, multi-user, programmable, interactive system well-suited to the construction of text-based adventure games, conferencing systems, and other collaborative software. Its most common use, however, is as a multi-participant, low-bandwidth virtual reality" Participants (usually referred to as "players") connect to LambdaMOO using Telnet or some other, more specialized, "client" program. Having connected to a character, players then give one-line commands that are parsed and interpreted by LambdaMOO as appropriate. Such commands may cause changes in the virtual reality, such as the location of a character, or may simply report on the current state of that reality, such as the appearance of some object. [....] The job of interpreting those commands is shared between the two major components in the LambdaMOO system: the "server" and the "database". The server is a program, written in a standard programming language, that manages the network connections, maintains queues of commands and other tasks to be executed, controls all access to the database, and executes other programs written in the MOO programming language. The database contains representations of all of the objects in the virtual reality, including the MOO programs that the server executes to give those objects their specific behaviors." [Curtis et Nichols, 1993].

MOOs allow for individual users to extend the environment both programmatically, and by "building" or creating new objects. In an educational context, this can allow the student to become an active participant in the learning experience. In addition, it is well documented in the literature that MOOs provide a strong sense of "place", bringing back some of the advantages of "campus" life that is lost in distance education.

A MOO server can also be configured to act as an HTTP server. This means that a WWW browser can be used to look at locations, rooms, people, artifacts, etc. in the MOO. Those objects can have hypertext (URL's) attached and therefore be used to structure information on the Web. Those interested in more details should look at: http://tecfa.unige.ch/tecfamoo.html (information about the author's MOO project) and the WWW-VL library on educational technology cited below for more general information on educational MOOs. See also [Butts et al., 1994] and [Speh, 1994] for a good example on the educational use of a MUD.

8 So what can we do with the Web?

The table in figure 1 attempts a summary according to Sandberg's functions. Each column is covered! - We "can do it". But a LOT of development work is still needed. The role of server-side scripts (e.g. see [Kay et Kummerfeld, 1994] or [Brusilovsky, 1995] for recent work) or client-side computing (à la "HotJava") is still not clear though. "Smart server-side computing" (e.g. replacing a standard WWW server) by a specialized educational server is another road that has not be full explored. There is always a trade-off between using the Web as a closed system, using it to distribute software, using it in combination with other software, etc. Only time will tell us what solution is best for what case.

[9] http://union.ncsa.uiuc.edu:80/HyperNews/get/www/collaboration.html

Function:	school	teacher	monitor	fellow learners	learning material	Ext.info sources	tools
Simple WWW	***	*	-	*	**	***	*
WWW with server-side scripts	**	**	*	**	**	***	**
WWW & local clients	*	*	-	**	**		
WWW with "smart" server-side computing		**	**	**	***	**	**
WWW interfaces to Internet tools		**	**	**		**	
WWW multi-user VR interfaces	**	***	***	***	**	**	**

Figure 1: Components of the Learning Environment

9 Further Resources on the Net

Interested readers can start browsing from the authors "Virtual WWW Library on Educational Technology"[10]

A few recommended books on the Internet: [December et Randall, 1995], [Gaffin et Heitkoetter, 1994], [Wiggins, 1994]

Persons interested in Educational Technology and/or "Virtual Environments for Education and Research" may join us at the TecfaMOO[11]!

10 About the Authors

Daniel K. Schneider is a research associate at the Educational Technology Unit (TECFA), Faculty of Psychology and Education, University of Geneva, Switzerland. His current research interest include computer mediated communication and information tools on the Internet

Address: 9 route de Drize, CH-1227 Carouge, Switzerland
Phone: +41 22 70 9694
MOO: daniel at tecfamoo.unige.ch 7777
WWW: http://tecfa.unige.ch/tecfa-people/schneider.html
Email: schneide@divsun.unige.ch

Kayla Block has a masters degree in Clinical Psychology and is currently working as an Independant Computer Consultant. Her current research interests include computer user interfaces.

Email: luxuria@netcom.com

[10] http://tecfa.unige.ch/info-edu-comp.html
[11] http://tecfa.unige.ch/tecfamoo.html

References

[Aronson et Briggs, 1983] Aronson, D. T. et Briggs, L. J. (1983). Contributions of Gagné and Briggs to a Prescriptive Model of Instruction. In Reigeluth, C., (ed.), *Instructional Design Theories and Models: An Overview of their Current Status.* Prentice-Hall, Hillsdale, NJ.

[Brusilovsky, 1995] Brusilovsky, P. (1995). Intelligent Tutoring Systems for World-Wide Web. In *WWW'95 Posters and Demonstrations*, Darmstadt. URL: http://www.igd.fhg.de/www/www95/posters/48/index.html.

[Butts et al., 1994] Butts, C., Reilly, C., Speh, M., et Wang, J. (1994). WWW and the Globewide Network Academy. In *Proceedings of the First International WWW Conference.* URL: http://uu-gna.mit.edu:8001/uu-gna/admin/papers/www/index.html.

[Curtis et Nichols, 1993] Curtis, P. et Nichols, D. (1993). MUDs Grow UP: Social Virtual Reality in the Real World. In *Third International Conference on Cyberspace*, Austin, Texas. URL: http://lucien.berkeley.edu/MOO/MUDsGrowUp.ps, also: http://www.oise.on.ca/%7Ejnolan/muds/about_muds/mudsgrowup.

[December et Randall, 1995] December, J. et Randall, N. (1995). *The World Wide Web Unleashed.* Macmillan.

[Dillenburg et Schneider, 1995] Dillenburg, P. et Schneider, D. (1995). Collaborative learning and the Internet. In *Paper presented to ICCAI 95.* URL: http://tecfa.unige.ch/tecfa/tecfa-research/CMC/colla/iccai95_1.html.

[Gaffin et Heitkoetter, 1994] Gaffin, A. et Heitkoetter, J. (1994). *Big Dummy's Guide to the Internet.* The Electronic Frontier Foundation, (available on-line). URL: http://www.germany.eu.net/books/eegtti/eegtti.html.

[Gagné, 1985] Gagné, R. (1985). *The Conditions of Learning.* Holt Rinehart, and Winston, New York, 4th edition.

[Ibrahim et Franklin, 1995] Ibrahim, B. et Franklin, Stephen, D. (1995). Advanced Educational Uses of the World-Wide Web. In *Proceedings of Third World-Wide Web Conference - WWW'95, Darmstadt, Germany, April 10-14, 1995*, volume 27, pages 871–877. special issue of Computer Networks and ISDN Systems. URL: http://cuiwww.unige.ch/eao/www/WWW95/paper.html.

[Kay et Kummerfeld, 1994] Kay, J. et Kummerfeld, B. (1994). An Individualised Course for the C Programming Language. In *Proceedings of the Second International WWW Conference '94 Mosaic and the Web.* URL: http://www.cs.su.oz.au/%7Ebob/kay-kummerfeld.html, also at: http://www.ncsa.uiuc.edu/SDG/IT94/Proceedings/Educ/kummerfeld/kummerfeld.html.

[Kearsley et al., 1993] Kearsley, G., Seidel, R., et Park, D. (1993). Theory Into Practice. A hypertext Database for Learning and Instruction. US Army Research Institute. (Note: The Theory Into Practice program (Mac, DOS and Windows versions) is being published by Wadsworth as a book/disk.

[Mallery, 1994] Mallery, J. (1994). A Common LISP Hypermedia Server,. In *Proceedings of The First International Conference on The World-Wide Web.* http://www.ai.mit.edu/projects/iiip/doc/cl-http/server.html.

[Sandberg, 1994] Sandberg, J. A. (1994). Educational paradigms: issues and trends. In Lewis, R. Mendelsohn, P., (ed.), *Lessons from Learning, (IFIP TC3/WG3.3 Working Conference 1993)*, pages 13–22, Amsterdam. North-Holland.

[Speh, 1994] Speh, M. (1994). A Virtual Programming Course using the MOO. *The Diversity University Journal of Virtual Reality Education*, 1(1):21–27. URL: http://info.desy.de/www/gna/DUVEJ.html, ASCII: http://tecfa.unige.ch/edu-comp/DUJVRE/vol1/DUJVRE.1.1.text, email: (Marcus) marcus@andersen.co.uk.

[Wenger, 1987] Wenger, E. (1987). *Artificial Intelligence and Tutoring Systems.* Morgan Kaufmann, Los Altos, CA 94022.

[Wiggins, 1994] Wiggins, R. W. (1994). *The Internet for Everyone: A Guide for Users and Providers.* McGraw-Hill, New York.