Personal University: Building an Academic Environment

Giovanni Degli Antoni

Dipartimento di Scienze dell'Informazione
Università degli Studi di Milano
Italia

Riccardo Ferrari

Facultad de Ciencias y Tecnología
Universidad Católica "Nuestra Señora de la Asunción"
Paraguay

The introduction of ever more powerful Information Technologies (IT) will make possible radical changes in the higher instruction field. Portable and low cost workstations can support all academic activities, substituing in a single unit most of the traditional communication technologies (black-board, papers, pencils, books...). The translation of a complete academic environment into a workstation is well indicated by the term: Personal University. Personal Universities have to integrate all the applications and documents of an entire university curriculum in a single workstation to be used by teachers and students. To make this possible sound architectural criteria have to be devised. This work reports reviews and suggestions taken from the literature as well from a Personal University1 project still under development at Milan University jointly with Asunción Catholic University.

Reasons for personal university workstation

Weissman has already pointed out in 1988 [21] the significant changes that we could expect from an intensive and careful application of technology to

1 PU is in the frame of an accademic cooperation agreement between Milan University (Italy) and Catholic University of Asunción (Paraguay). It is sponsored by the Minister of Foreign Affairs jointly with the Minister of University and Scientific Research of Italian Government.

higher education. Notebook size computers, large storage devices like CD-ROM [8,12,15], communication facilities, will provide the base for the design of new instructional systems. Nevertheless, to become largely diffused, these systems have to cope with technological and commercial constraints that prevent their advance.

The existence of different and incompatible types of components is the primary source of technological problems. Actually building an instructional system implies the selection of several elements like:

- Computer platform
- Operating system
- Graphical user interface
- Author system
- Data support and format
- Communication devices.

The result of all the possible combinations is an atomized market, influenced also by commercial problems as:

- High cost of instructional software production
- Cost of organizational efforts
- Lack of established circuits for instructional s/w distribution.

The outlined situation make almost impossible the settlement of standards (h/w, s/w, courseware) that would accelerate the growing of the sector. So, even if no discipline has been untouched by academic computing tools and technology [7,20], today tools remain far from a full integration. This becomes more evident considering a lecture, the typical interface between faculty and students. Computer support for lecture performance is still limited. Preparation of lesson is time consuming, as most of the informations needed reside on paper and have to be entered to the machine. Once the lecture is prepared its presentation to the classroom represent a technical challenge. Tuning all system components is time consuming and troublesome, also even a single h/w and s/w platform is in use. Hardware configuration, operating system version, software installation often become problems as each professor uses his own set of tools. Using a single portable workstation is a simple alternative. Provided a video signal standard, professor has just to plug in and play his own workstation in the classroom. As standards for multimedia data interchange will be defined [2, 13, 17] communication among this kind of

workstation will increase, even if they were developed using competitive technologies.

These considerations let us understand the value of making available all documents and applications for a complete university department on a single integrated environment. We call this academic integrated environment: Personal University (PU).

In a Personal University framework a professor could work out the following activities:

1. Personal time management
2. Scientific writing
3. Scientific computation
4. Personal literature
5. Access to data and literature banks
6. Cooperative work with colleagues
7. Lecture preparation
8. Drill, practice and exercise preparation
9. Student's evaluation
10. Course management
11. Communication with students
12. Lecture performance using a lecture room projector

Being higher education an intensive form of training, we assume that teachers and students have to use almost the same tool. As today they can have access to the same library, tomorrow they will have available the same information sources. A personal time management system will allow students to take care of the different subject matters that they have to master. Through PU workstation student will be able to study, he will perform evaluation of his discipline mastering. PU will be also a tool for professor during examinations. As already outlined, an integrated environment including facilities for lectures and examinations will reduce the gap between research and study. Since teachers and students use the same environment to study, the training aspect of university life will become more evident. Teachers and students will become knowledge explorers working together. The use of PU could become an obvious activity as the use of paper.

Personal University Requirements

It is rather evident that the development from scratch of the software needed for a PU is beyond actual financial ranges of the instructional field. Indeed the budget for didactic purposes is generally limited, even if a certain number of big projects is underway[1,3]. The integration of existing application permits us to withstand economical constraints. This implies that the first mandatory requirement for PU is the *availability of operating systems that allow communication among applications*, possibly existing ones. Given the state of the art in this field, it seems correct to project the solution of a working Personal University within the next two years.

The second requirement is about *software selection*. The availability of future versions supporting interapplication communication is the primary criterion. This requirement is not stringent for developing prototypes. It will not be difficult or costly to substitute some components.

A design methodology arises naturally as third requirement: *realistic prototyping*. This methodology suggests the implementation of prototypes with the best available components (h/w or s/w), and with a minimum software development effort. The availability of communication among applications will give a large premium to this attitude when such an integration facility will be fully available. The fourth requirement (that sprang from PU experimentation) is that the *project goal has to be quite advanced.* In this way during the life time of the development cycle there will be little chance for the competition to prevail.

The fifth requirement for the development of a workstation for higher education is inherent to *publishing problems*. As we have previously observed, software distribution channels are not still adequate to distribute software instructional material. There are many reasons for this situation and it is outside our purpose to analyze them in this note.

We just want to affirm that a *good integration with paper publishers* would make things better. Juxtaposing "electronic" and hard-copy books is a suitable frame for research[22], but not for production[16,18]. Traditional editors have well established experience from various point of view regarding promotion, advertising, incentives for authors, quality control, distribution....

There are many technical routes to reach such a good integration. Here we suggest one, that could be fully sustained by paper publishers in the case that electronic media will emerge.

1 - Electronic version has to reproduce the appearance of its paper correspondent.
2 - All the snapshots of electronic version must be printable.
3 - Electronic version must be treated as an *added value* of the paper version. All the activities that can be performed by hand, with pencil and paper only, will be automated in the electronic version.

Personal University Architecture

Starting from previous requirements, only few possible choices remain. Integration with traditional publishing strongly suggests centering of Personal University on a library of books in electronic format. We call pu-book the electronic book of PU. Pu-book must satisfy the criteria pointed out in fifth requirement, and has to be well integrated with all the rest of PU to comply with all other requirements.

It easily follows that the rest of the PU has to communicate quite well with the pu-book. This will be the basis for the activities that we mentioned above both for the professors or for the students. The library, the data bank, the presentations, the communication channels, the personal documents file and all the other information sources or destinations will be integrated. Since some of these sources or destinations, elements of selected commercial applications, it follows that integration will rise from communication among applications; that is from higher order software not necessarily provided by commercial application developers. We suggest a name for this higher order software (software linking commercial applications): agents.

In our hypothesis, agents have to deal with:

- Filling up empty pu-books, that is developing or modifying a pu-book.
- Adding personal notes to the pu-book.
- Navigating the PU-library.
- Producing and performing lectures with suitable presentation systems.
- Doing computations (mathematics, graphics...) on the entities of the pu-book.

- Developing evaluation (examination) dialogues, for example multiple choice.
- Producing printed material.

...

We will not give the full list as this can grow and it is related to many factors. Obviously, standard packages suitable to make possible all the activities suggested in the first chapters have to be integrated.

The human interface: the hyper dimension

The human interface represents an important element of every computer system. There are many aspects that have to be taken into account for the choices to be done. Hypermedia is certainly a good choice for a scientific workstation [5,6,9]. Facilities to make buttons trigger other communicating applications have to be provided. For example mathematical formulas of pu-book have to be computed directly upon request to a mathematical agent. The result will be placed where it has been established during the book design, using a simple script language based on the notion of context. It is necessary to take care of similar considerations also for function graphs or other computational entities.

All reference entities are good candidates for buttons/links that can connect them in a bidirectional symmetrical way. Indexes are a collection of links to corresponding parts and all parts can be used to recover the corresponding index. The same can be carry out for figures identifiers, references, notes, formal content. Definitions will be easily referenced by a suitable agent.

To conserve the paper book structure, a default navigation path could be selected, based on the page structure. However, the reader will have other possibilities of navigation following associative links. To avoid cognitive difficulties, the path followed by the user is certainly the best entity to be used as an Arianna's wire. We call the user path *hyper dimension* or *hyper*.

Hyper is analogous to Bush concept of trail in Memex [4] and, as trail, it is essentially a medium for communication. All the information collected and the activities performed during navigation have to be stacked to hypers. The hyper dimension can obviously be stored and it is a way to build up a lecture. A lecture preparation is just a navigation in the PU-library. A lecture performance is just a reply, driven by the teacher, or by a clock, or by

students' questions, of previously prepared hyper. The information elements associated to hyper are just information taken from books, results of computations, computation requests, graphs and picture (animated or not), segment of vocal comments.

So the human interface suggests itself a style for building up lectures, which is not biased by pedagogical suggestions. Transferring traditional material into electronic form we allow an easier way of lecture construction and presentation, minimizing interferences with the teachers' educational habits. Integration between hyper dimension (an agent) and a presentation system is a natural result of this consideration, as suggested by efforts in developing the PU project. Last but not least, the hyper dimension is a good candidate for transporting work from teacher to student, from student to teacher or to other students or for cooperative learning. Hypers, just as files, can be opened, displayed, filled, closed, stored, mailed, copied, edited, combined, browsed, protected, shared....

Intelligent Tutoring in a Personal Universtiy

Navigation paths could be used by intelligent agents as a way for learner modeling [19]. Such an intelligent tutoring system will advice the user about learning strategies, retrieval of useful information, performance of practices and exercises.

Hypers are a good base for the development of a user observer as they collect all the user interactions with the system. Anyhow, to allow the use of hypers as knowledge base about learner habits, their content has to acquire a semantic significance suitable for machine interpretation.

A simple way to start intelligent tutoring is the observation of reading. A recording of the time spent on any page can provide a useful parameter to start a comparison among different learners. The sequence of browsed pages is another useful information. When a formal representation of the subject is available, a check about page sequence consistency can be performed. When some inconsistency is detected, the system could start a dialogue with the user discovering new association possibilities or advising a better sequence.

Integration with evaluation tool will offer another optimal opportunity to make the tutoring system aware of learner characteristics. Response to multiple-choice answer tests is the starting point of similar tool, but the

integration with intelligent applications will lead to more sophisticated approach. For example: intelligent tools for grammatical verification are already available. Integrating them with a pu-book would make it possible to compare detected mistakes and hyper contents, and discover where the user has to be addressed for improvements.

Formal disciplines and the pu-book

The development of pu-book has been a good chance to understand a possible use of the formal content of disciplines [11]. Indeed, given a network representing the formal description of the book subject, it becomes possible to browse through the book by a network navigation. Such a network could be based on the notion of conceptual dependency: a concept depends on others trough connections elements (nexuses). Taking concepts and nexuses together we produce a network.

Browsing in the network means stepping among concepts. For this reason we call the kind of browser we developed for the personal university, conceptual browser. Other entities can be associated to concepts and nexuses: examples, references, explanations, notes, comments, user notes, parts of text... these entities become detachable components of a formal structure of the text. In the pu-book developed by the authors Petri Nets have been used [10,14]. However any other net or graph can be used, as, for example, semantic nets.

Many suggestions emerge from the conceptual browser development. The usefulness of such a navigation for all readers is not always obvious: but if a user wants to enter deeper into the subject matter, it is a strong advantage to have the possibility of understanding its formal structure.

From the design point of view, the link between formal and informal content can be related to textual chunks producer and to maintenance facilities. Thes notions seem related to an extension of the mark-up techniques to the case where mark-up have a formally treatabl meaning.

It is probably a good line of experimentation to develop technologies that can finally show one real use of formal contents of the disciplines.

At that point theorem provers could be used for formal deductions. An intelligent dynamical book could result.

Conclusion

This paper is a mix of results and suggestions emerging from our experimental activity. The suggestions have all been tested with some prototyping activitiy. Sometimes this prototyping has been carried out in different machines, without any intention of being ready for an engineering effort, and in any case lacking a good operating system suitable for supporting communication among applications.
Results are mainly a set of simple architectural suggestions that the reader can easily find in our notes. With them we hope to contribute to minimizing mistakes in future developments.

Acknowledgements

We acknowledge HUG (Hypertext User Group) of Computer Science Department at the Universityof Milano and the Digital Lab at the University of Asunción (LED). They given a strong support to the activities. Then we give thanks to the many Italian and Paraguayan students who worked out all the prototyping. Their work will appear elsewhere and is available on request. Italian Minister of Education and the Minister for the Foreign Affairs have given financial and normative support: we deeply acknowledge them.

References

1. Balkovic E., S. Lerman., R. P. Parmelee, "Computing in Higher Education: The Athena Experience." *Computer* 18(11):112-125; November 1985.

2. Bono, P. R. "A Survey of Graphic Standards and Their Role in Information Interchange." *Computer* 18(10):15-30; October 1985.

3. Brand, S. "The Media Lab: Inventing the Future at MIT". New York: Viking Penguin Inc., 1987.

4. Bush, V. "As We May Think." *Atlantic Monthly*: 176, pp. 10-108; July 1945. Reprinted in *CD ROM: The New Papyrus*; Microsoft, 1986.

5. Conklin, J. "Hypertext: An Introduction and Survey." *Computer* 20(9):17-41; September 1987.

6. Cook, P. "Multimedia Technology." In *Interactive Multimedia.* Edited by S. Ambron and K. Hooper. Redmond, WA: Microsoft Press, 1988.

7. Degli Antoni, G. "Technology and Scientific Thinking." Atti del Congresso Internazionale SILFS. Viareggio; January 1990.

8. Elshami, A. "CD-ROM: An Annotated Bibliography." Libraries Unlimited, Inc., 1989.

9. Fraase, M. "Macintosh Hypermedia." Scott, Foresman and Co. Glenview, 1990.

10. Genrich, H. J. "The Petri Net Representation of Mathematical Knowledge." GMD Bonn, Interner Bericht, ISF-76-05, 1976.

11. Greselin, F., R. Ferrari, G. Degli Antoni, D. Viganó, "Petri Nets as 'Informal Knowledge Engineering' Solution." Proceedings XVI Conferencia Latinoamericana de Informática: 459-477; Asunción, September 1990.

12. Holtz, F., "CD-ROMs: Breackthrough in Information Storage." Tab Books, 1988.

13. Horak, W. "Office Document Architecture and Office Document Interchange Formats: Current Status of International Standardization." *Computer* 18(10):50-60; October 1985.

14. Jantzen, M. "Structured Representation of Knowledge by Petri Nets as an Aid for Teaching and Research."

15. Lyon, E. "Spoilt for Choice? Optical Discs and Online Databases in the Next Decade." Program 25(1):37-50;January 1991.

16. Myers, P. "Publishing with CD-ROM." Meckler Corporation, 1987.

17. Postel, J. B., G. G. Finn, A. R. Katz, J. K. Reynolds, "An Experimental Multimedia Mail System." ACM Transaction on Office Information Systems 6(1):63-81; January 1988.

18. Ropiequet, S., J. Einberger, B. Zoellick, "CD-ROM and Optical Pubblishing." Microsoft Press, 1987.

19. Weyer, S. A. "As We May Learn." In *Interactive Multimedia.* Edited by S. Ambron and K. Hooper. Redmond, WA: Microsoft Press, 1988.

20. Weissman, R. F. E. "Academic Computing: the Next Wave", *Brown Online* 2: 3-6; May 1989.

21. Weissman, R. F. E. "From the Personal Computer to the Scholar's Workstation", *Academic Computing* (10): 10-41; October 1988.

22. Yankelovich, N., N. Meyrowitz, A. van Dam, "Reading and Writing the Electronic Book." *Computer* 18(10):15-30; October 1985.