Present and Future of the Virtual Campus

The Future of University Teaching?
Multimedia, Web and New Technologies
JTAP Workshop
Friday 23rd April 1999,

The University of Edinburgh

Daniel Schneider

TECFA
Faculté de Psychologie et des Sciences de l'Education
Université de Genève

PDF: http://tecfa.unige.ch/tecfa/talks/schneide/epcc-99/epcc-talk.pdf (including links)
HTML: http://tecfa.unige.ch/tecfa/talks/schneide/epcc-99/epcc-talk.html (very ugly!)
PS: http://tecfa.unige.ch/tecfa/talks/schneide/epcc-99/epcc-talk.ps.gz (2 slides / page)

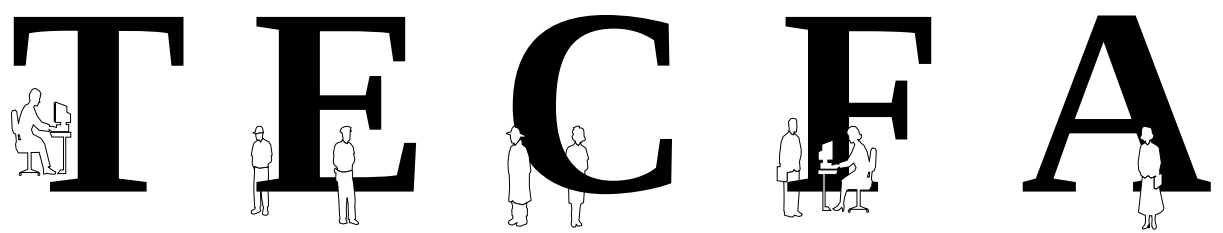
1. Introduction

1.1 Outline

Focus on the overall environment !

1.2 About TECFA

- A teaching and research unit in the School of Psychology and Education
- TECFA = Technologies de formation et apprentissage (educational technology)



Research		Education
☐	Computer Aided Instruction	☒
☒	Artificial Intelligence & Education	☐
☐	Multi-media & software ergonomics	☒
☒	Distance teaching	☒
☒	Communication & collaboration	☒

- About 15 people (4 permanent)

1.3 The case study: Our master in Educational Technology (STAF)

structure: **6 periods: [each 1 week at TECFA + 4 weeks at home]**

audience: **15-20 adults [most work part time]**

curriculum: **1st year: 8 courses / year**
2nd year: 3 optional courses + master thesis + internship

staff: **4 teachers + teaching assistants**

timing: **4 courses extend over the whole year**
4 are project-based and last 1 period
(1/2 week + part of 4 weeks at home)

benefits of the formula:

Presence time:
Students can work half-time in other locations
During presence time we can “catch up”
Some teaching works better in the classroom
Student projects can be launched and debriefed during presence
Some technical problems are better discussed face-to-face

Distance time:
flexibility

STAF **first year courses (rough translation from french)**

STAF-11: Learning theories and educational applications

STAF-12: Man-machine interaction and interactive programming

STAF-13: Image, sound and communication in multimedia software

STAF-14: Technical basis of information and communication systems

STAF-15: Hyperdocuments and educational help systems

STAF-16: Educational Software development

STAF-17: Distance Education

STAF-18: Teaching and learning in a virtual environment

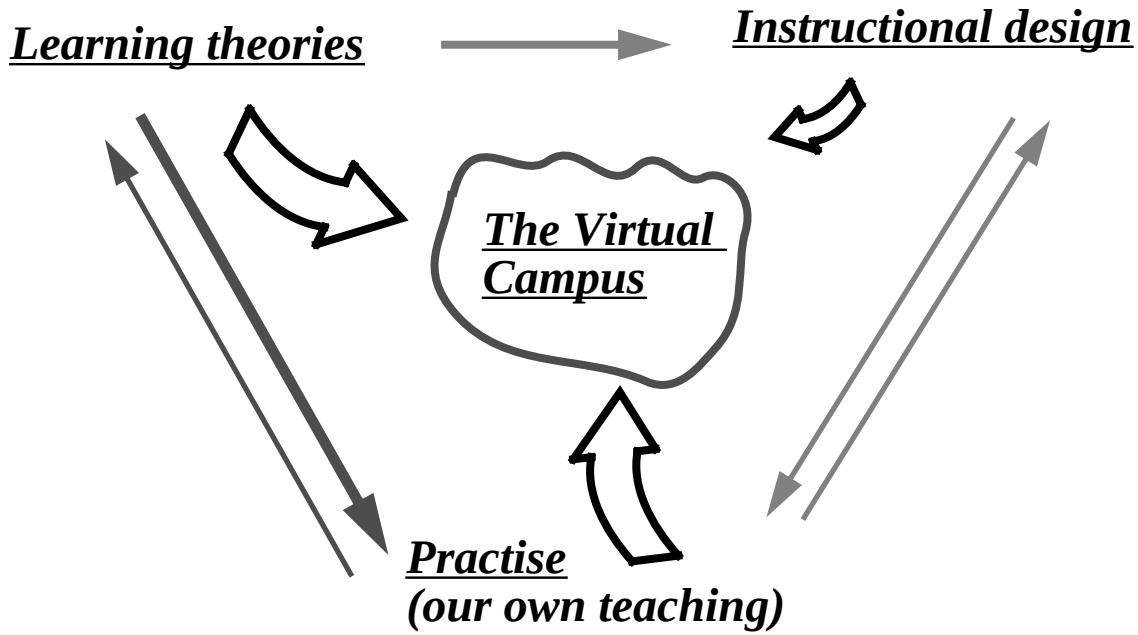
Wide range: psychology, education, multimedia, ergonomics, semiotics, ...

My talk is slightly incestuous

- We teach Internet and use Internet to teach
- However, most ideas apply to wide range of teaching subjects and situations (you can steal ideas from us !)
- Some of our ideas are starting to spread (e.g. in medical school)

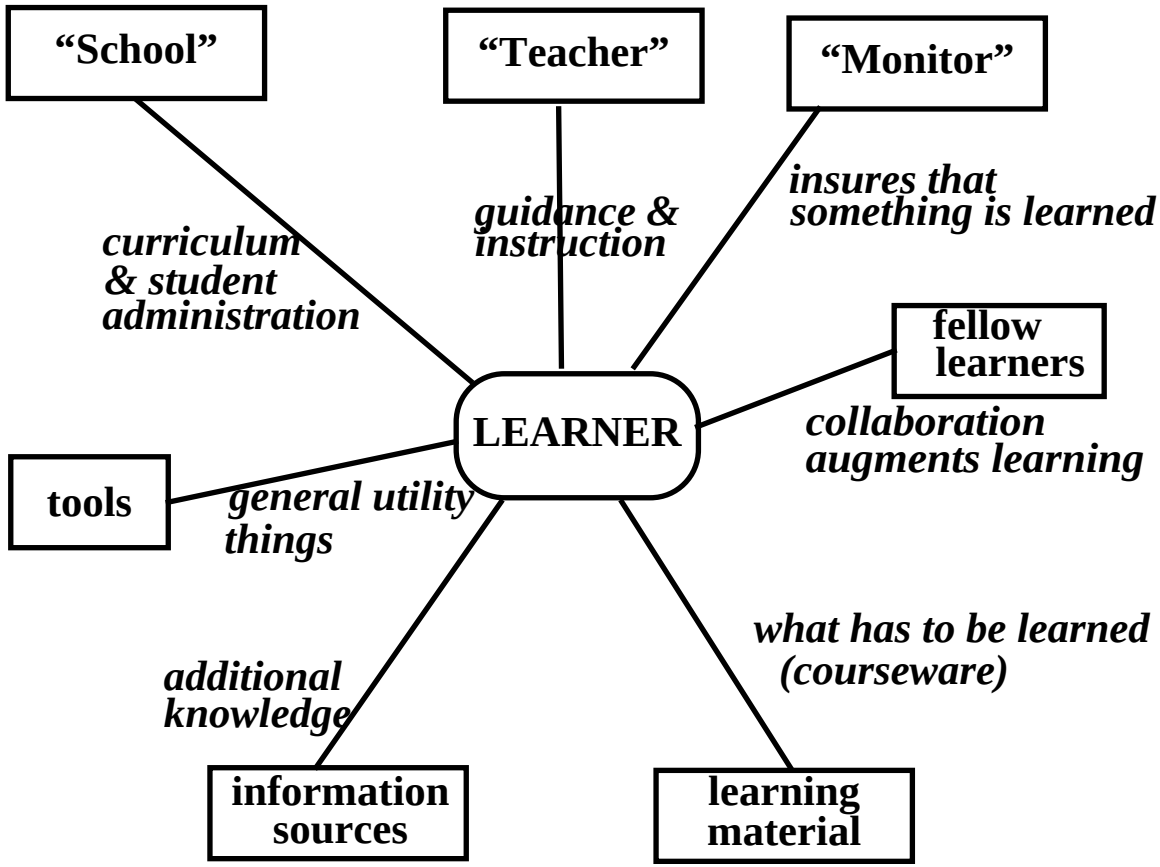
2. The Tecfa Approach to teaching and learning

2.1 Main focus on Learning



- Using Internet for changing education
- Learner & activity centred
- based on our own experience

2.2 The Learning Environment



These are functions, not necessarily people, organisations, things, ...

- It's a whole with several components,
- all of them ought to be supported !
- Specially important in full distance teaching !

2.3 Acknowledgment of variety

"Learning" is not a general thing:

- Learning Types: Attitudes, Factual Information, Concepts, Reasoning, Procedure Learning, Problem solving, Learning Strategies,...
  See: Theory into Practice Database
- Degrees: beginners vs. advanced, easy vs. difficult, personal preferences
- All major learning paradigms have a point
  (Constructivism, Instructionalism, Socio-culturalism etc.)

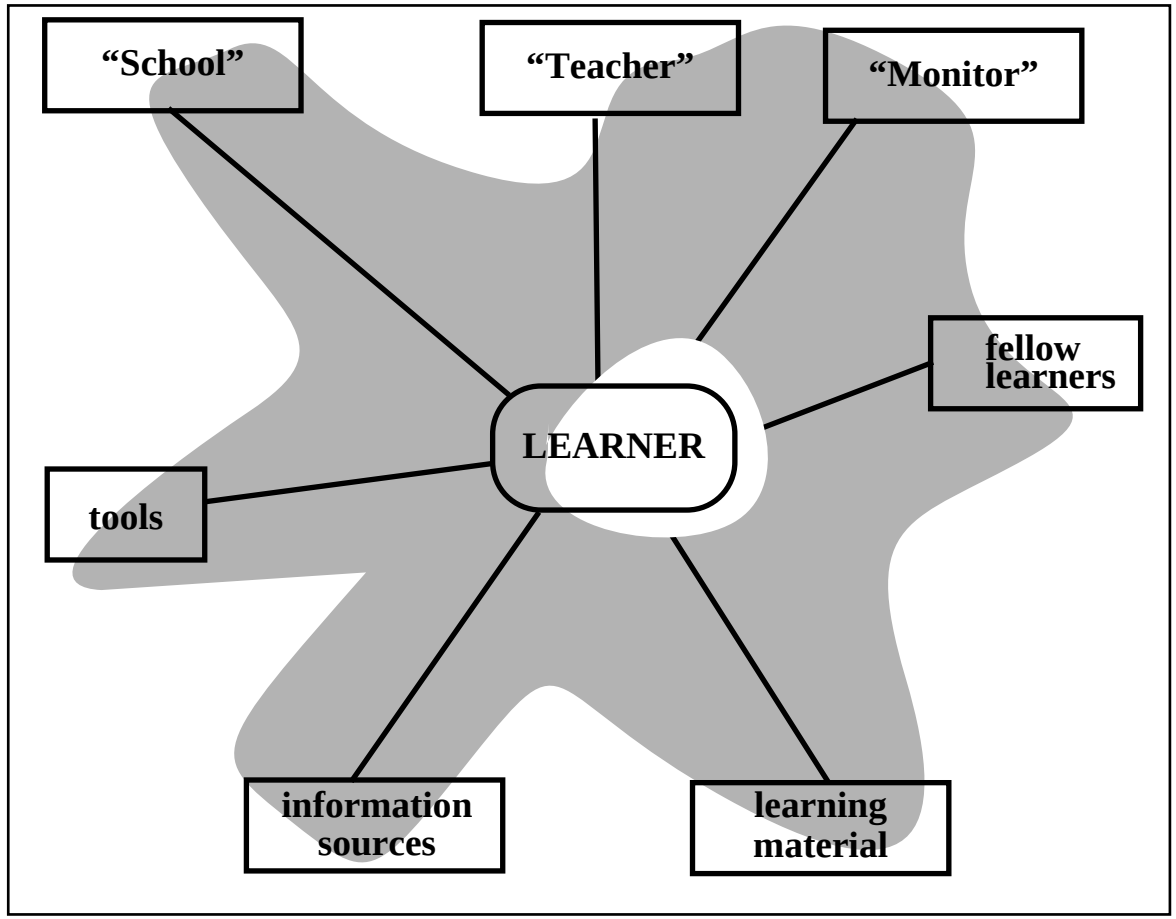
But there are common features:

- Reading is NOT learning
  => On must "do" to learn
- ... and Internet is NOT interactive (per se)
  => What counts is activity (triggered by the system and/or by the task)
- Most people need additional guidance to achieve instructional goals
  => External conditioning (teaching, monitoring)
- Learning and teaching paradigms are best combined

☞ The Virtual Campus must be flexible to accommodate these requirements

3. Virtual Campus software: a short overview

- We only get “some of it”
- There isn’t any globally satisfactory virtual campus software



3.1 Mainstream virtual campus software

- See: http://www.ctt.bc.ca/landonline/ [Landon's comparative Analysis]

Some market leaders:

- WebCT: http://homebrew1.cs.ubc.ca/webct/
  At Tecfa we ***teach*** WebCT (but don't use it for teaching)
- Topclass: http://www.wbtsystems.com/
- Lotus Learning Space: http://www.lotus.com/home.nsf/tabs/learnspace

Main features:

- Asynchronous Communication: email, forums
- Synchronous Communication: chat, whiteboard, teleconferencing, group browsing, application sharing,...
- Student tools: home page, self tests, bookmarks, progress tracking, ...
- Student Management Tools: progress tracking, on-line grading, ...
- Lessons tools: authoring (structured HTML text), testing (e.g. JavaScript generators)

Main differences:

- Authoring (ability to import, hand coding, HTML forms-based editing tools)
- Student Management (at course- and general level)
- Application sharing and co-editing
- TeleConferencing
- Flexibility and possibility to adapt the software

Missing features:

- Virtual Space
- Interactive applications (besides HTML forms)
- Interactive shared applications
- “openness” (most are quite locked)

2 main teaching approaches supported:

☞ Traditionalist teaching

- Mostly reading and answering questions

☞ Resource and project-based learning

- This saves Internet-based teaching !

3.2 Other interesting things on the Internet

- See: http://tecfa.unige.ch/guides/cspace-pointers.html (unsorted list)
- Thee is a lot of creative software, but it's not integrated in standard campus tools

Various Groupware

- featuring: whiteboards, shared applications, conferencing, etc.

Virtual Environments

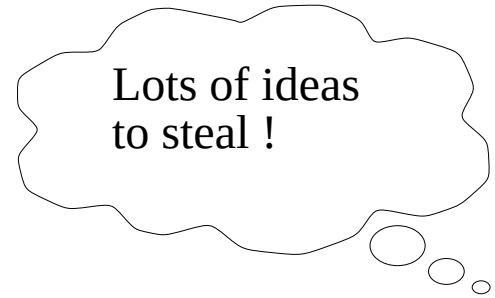
- featuring: spacial organization, avatars (identities), building spaces, forums, sophisticated chat and more
- there are several varieties, e.g. MUDs, VRML-based, custom 2D

More recent combinations of the two

- There is a promising future for integrated "Cyberspace desktops"

Training applications, e.g. simulations

- topic not addressed in this talk
- increasing amount of good and interesting things (specially in science)



4. Ordinary Internet supported teaching at TECFA

4.1 The four basic tools

1. WWW (hypertext), e.g. for:

- planning, curricula, agendas, assignments
- texts, manuals
- resources and pointers
- assignments (student productions)
- collaboration within group projects

2. Email, e.g. for:

- agenda planning (teacher)
- search for information (student)
- information about updates (student, teacher)
- short comments (teacher)

3. Discussion Forums, e.g. for:

- debates (about articles or themes)
- technical Q/A
- student-student help (!)

4.The MOO, e.g. for:

Short definition (see http://tecfa.unige.ch/moo/tecfamoo.html **for more)**

- MOOs are text-based Virtual Environments
- Metaphor: A living evolving book with persons
- sophisticated chat, mail, forums, rooms, objects
- user programmability, integrated www servers

Our usage:

- urgent things
- co-presence (common virtual space, radio channels)
- virtual meetings
- non-intrusive (almost) real time questions/answers
- supports 1-to-1 dialogue with multiple students
- Special Java/HTML interfaces for some purposes

Not used:

- Video-conferencing (except for demo purposes)
  - (we meet our students every 5-6 weeks, so need for that is low)
- Groupware (like whiteboards, calendar tools)

4.2 Problems with the "basic four"

- "*Please send Email*" does not work very well
  - Typically, students hesitate to ask questions (or too late)
  - Teachers are overloaded: overlook details, do not prompt
- "*Use the forums*"
  - Need constant coaching/animation else they die out
  - Students think their problem is private
  - Slowness of WWW-based tools,
    features missing within standard News
- "*Be on the MOO*"
  - Connection costs
  - difficulty to find a date for large group meetings
  - Not sexy enough
  - No teacher animation (lack of time)
- Web pages
  - updating
  - structuring and navigation, meta-information, etc.

Summary:

- Students need a lot of attention
- They have a tendency to get lost (they rather read than produce)
- We don’t always cope with student management

 Build more structured activity into the virtual campus

Note:

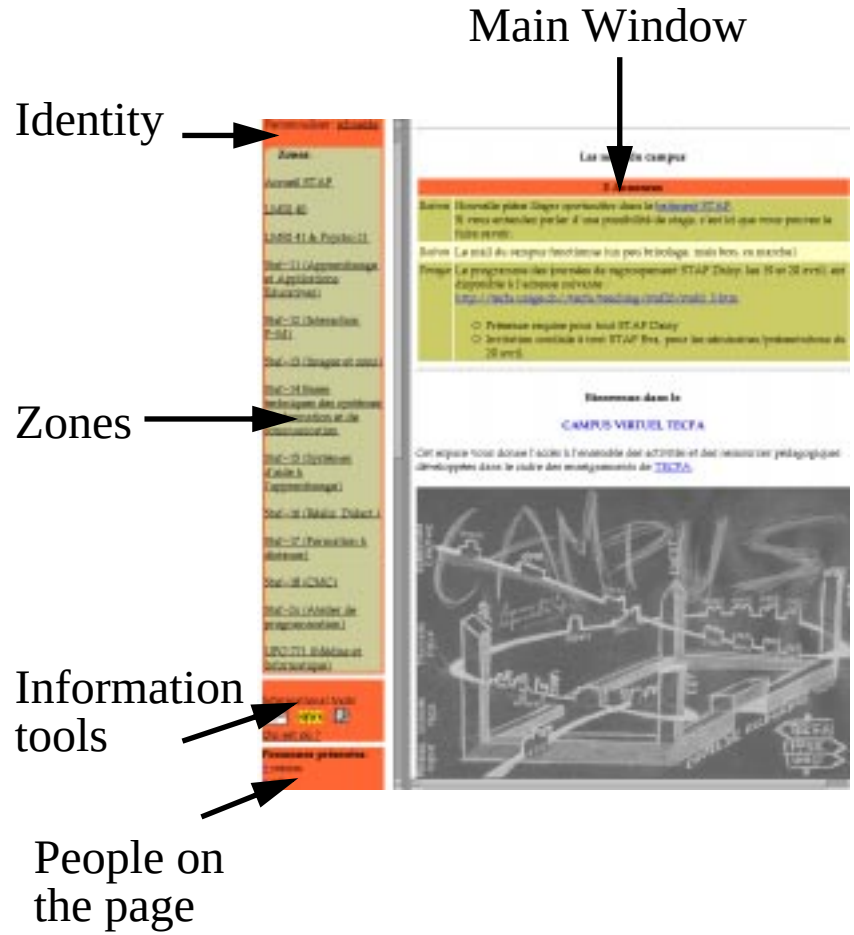
♡ Our experience hasn’t been really bad, but we want to do better

- E.g. my technical Internet course (Staf-14) works all right with a "standard" approach:
  - 1 exercise per period,
  - grading for each, marks count for the final evaluation
  - good resources on the Web (including examples)
  - students have to meet requirements, but can freely choose the subject
  - Auto-motivation exists for easier technical subjects (Html, PHP, JavaScript, VRML, etc.)
- On the other hand, my Java course is rather a disaster
  - e.g. I failed to check on student’s work every week or so.

5. TECFA's new Campus project

Overall Design

- Spacial metaphor
  - Zones = courses
  - buildings = main activities
  - rooms = sub activities
- HTML interface
- Other interfaces under way:
  3-D abstract VRML
  3-D VRML island
  2-D (MOO linked ?)



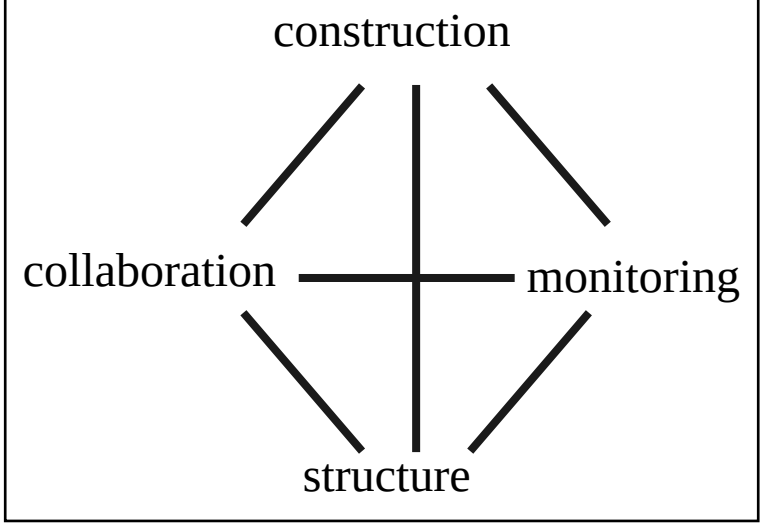
- Featuring a growing amount of structured learning/teaching activities
- A very recent project (its architecture will be revised this summer)

5.1 Our growing activity collection

Features (each has some)

- Sharing
- Project monitoring
- Construction
- Discussion & argument
- Information classification
- Navigation tool construction

The principle:



Current growth rate of implementations

- about 2/month

Planned for later

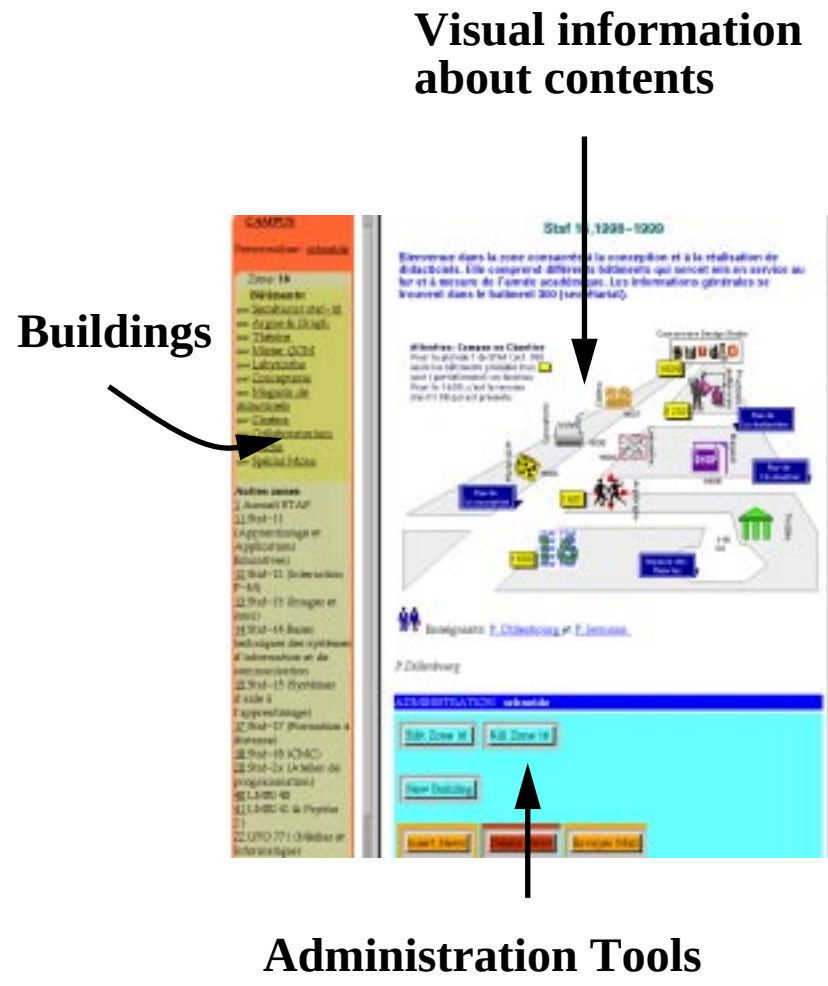
- 3D multi-user environment navigation and chat interface
- Central authentication
- Common tools (each activity basically is programmed from scratch)
- More information tools
- Better integration of synchronous communication

5.2 Zone Example: The Staf 16 - Educational software course

Main buildings (activities):

- Argue & Graph
- Mister QCM
- Design Studio
- Concept Factory

- A building fore each activity (teaching/learning activity class)
- Each building:
  - sets up various tasks with special tools
  - contains the necessary information
  - features collaboration



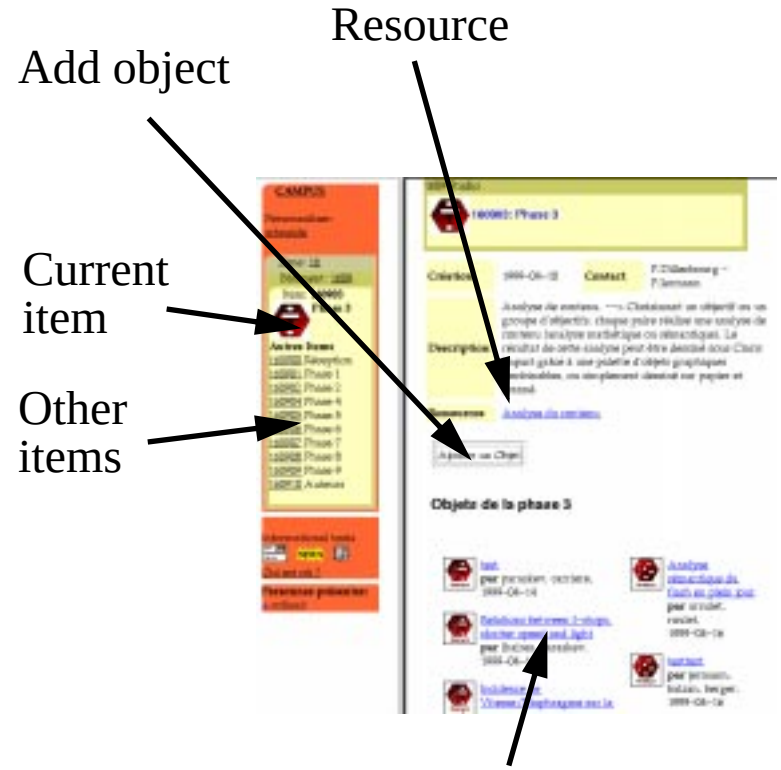
5.3 Activity examples

A. The Courseware Design Studio

- Part of STAF-16
- Goal: Learn to design educational software

Phases (learner activities):

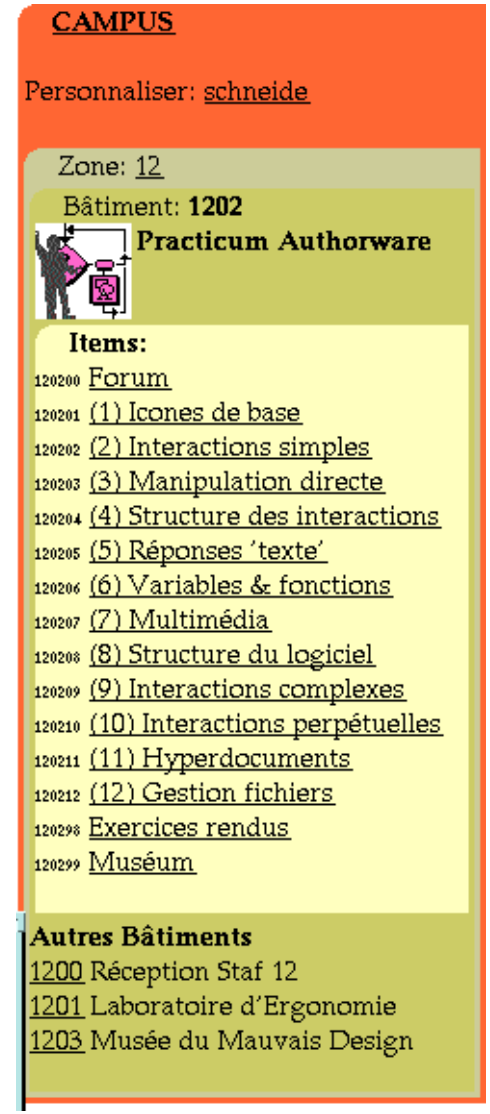
1. **Topic specification**
2. **Objectives specification**
3. **Content analysis (analyse the objectives)**
4. **Operational pedagogical objectives spec (of other pairs)**
5. **Architecture and scenario spec**
6. **Learning activities spec (of 3 learning activities)**
7. **Preparation of Materials**
8. **Development of a module (with Authorware)**
9. **Assembly of all the modules**



Objects of phase 3 produced by students

B. Practicum Authorware

- Staf 12: Man-machine Interaction (Ergonomics)
- Goal: Learn Authorware
- Method: Instructional Design
- Support:
  - Authorware manual (on-line & paper)
  - News Forum
  - Examples
- Task:
  - Each student has to solve series of grouped exercises
  - Some exercises must be deposited in the campus
- Motivation:
  - Exam
  - Prerequisite for STAF-16

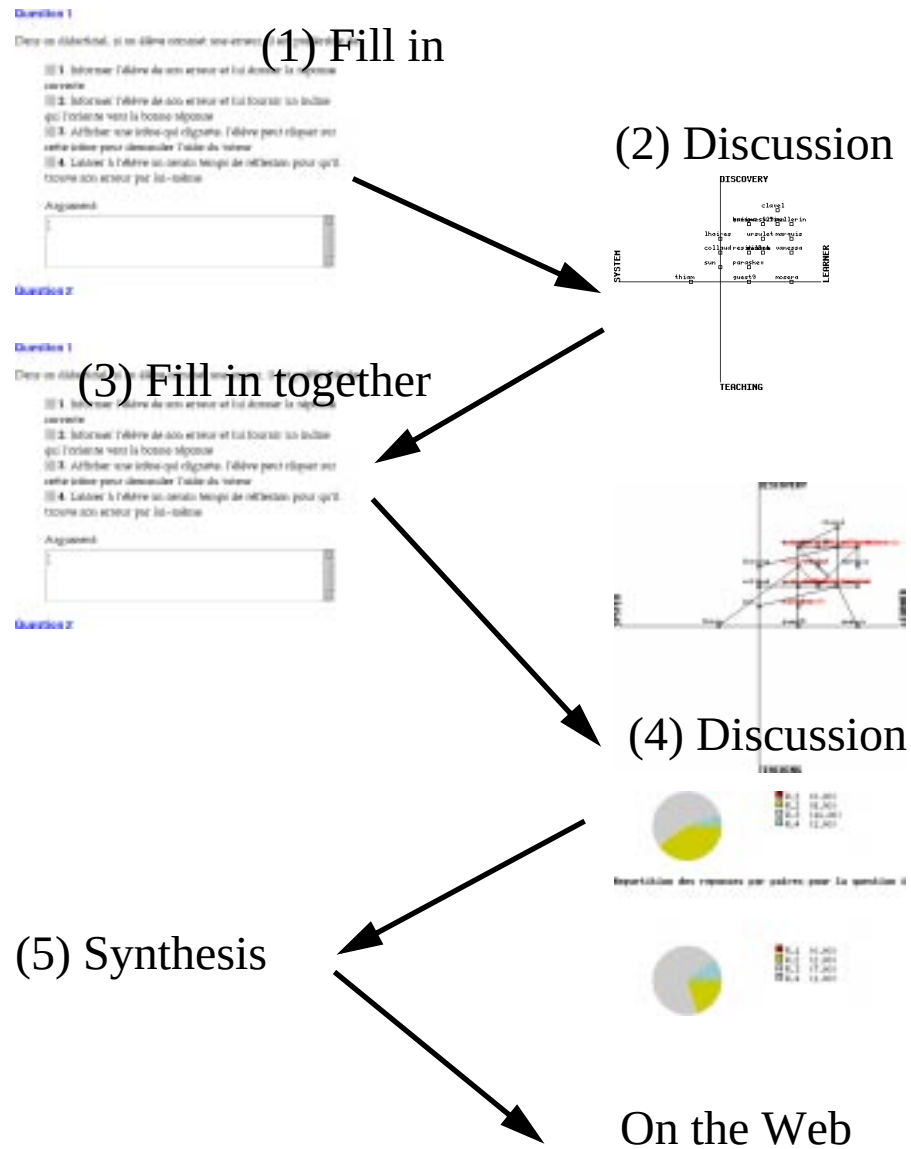


C. Argue & Graph

- part of STAF-16
- Goal: Support classroom discussion about pedagogical style

Scenario:

1. **Students fill in questionnaire about design choices**
   - Campus produces summary information (incl. graphs) (system vs. learner control discovery vs. teaching)
2. **Classroom discussion**
3. **Collaborative fill in (again)**
   - Teacher selects opposite pairs
   - Pairs fill in questionnaire
   - Campus produces summary + details
4. **Classroom discussion**
5. **Synthesis (HomeWork)**
   - Each student writes a text



D. The Iconometer

- STAF-13 course
- Goals:
  - Test icons used in web pages
  - Learn about "monosémie"

Scenario

1.Look at an icon

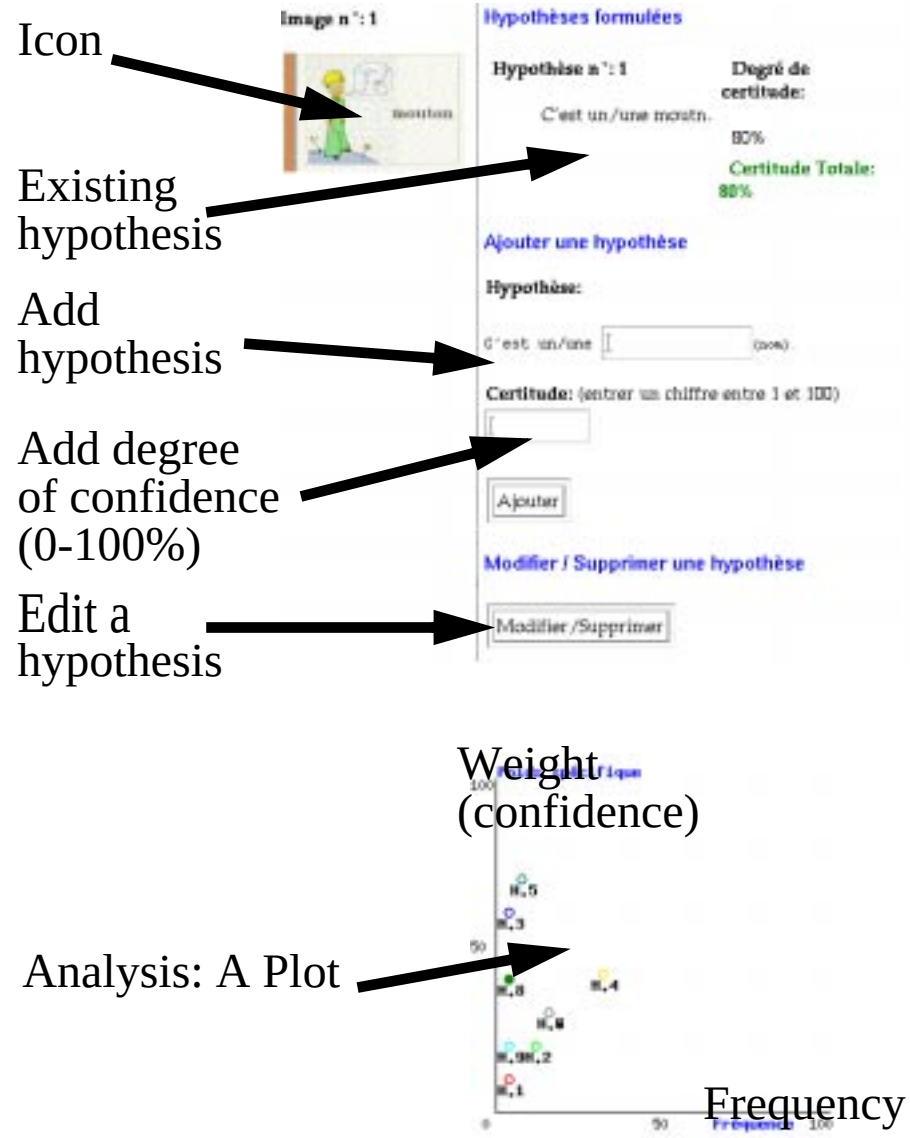
2.Formulate hypothesis

- one or several
- each with a confidence factor
- total must not exceed 100%

3.Look at results

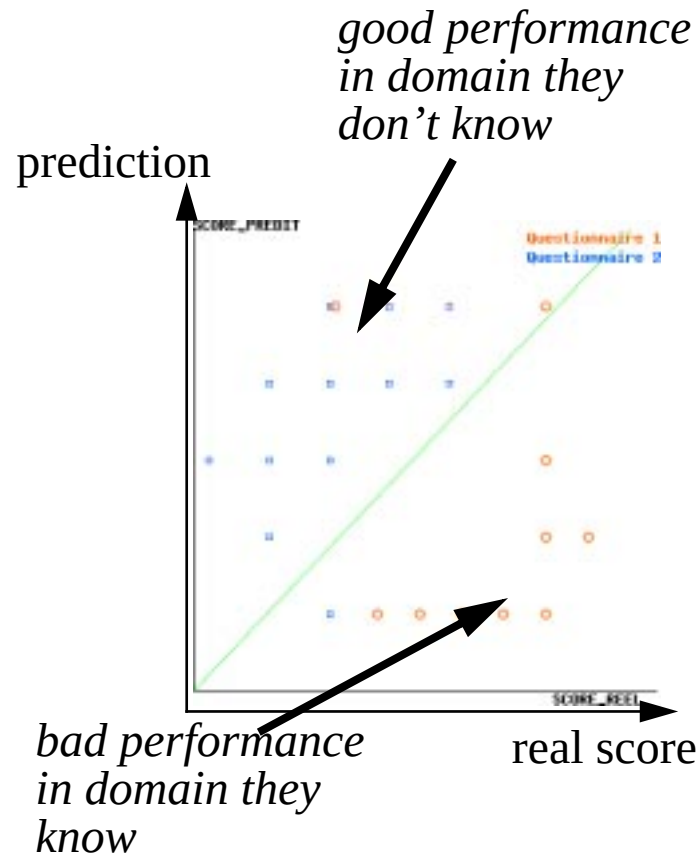
- Look at each hypothesis
- Look at summary information

4.Discuss ....



E. Mister QCM

- STAF-16 course (ergonomics)
- Goal: Students have to experience QCM design effects
- Scenario:
  - Students fill in 2 badly designed QCMs about Belgian Athletes and European capitals. They also indicate their level of competence.
  - Students look at results (question by question and overall plot)
  - They are very angry at the results (bad performance on european capitals and good one on Belgian athletes)
  - Now they are motivated to understand QCM design and participate in classroom discussion
- The tool has design notes on each question (including "don't"s)



F. Analysis of Information Units

- Goal: Learn to classify information units in texts

G. Simple project Supervisor

- Goal: Simple project management

H. Student project toolset (XML)

- Goal: Project management (software development and research papers)

I. Ergonomics Lab

- Goal: Play with 6 versions of the same application and determine features

J. Scenario Construction Workshop (Staf-15)

- Goal: Students design scenarios for help systems and implement parts of it

..... there is more .....

5.4 Other tools

A. MapLinks

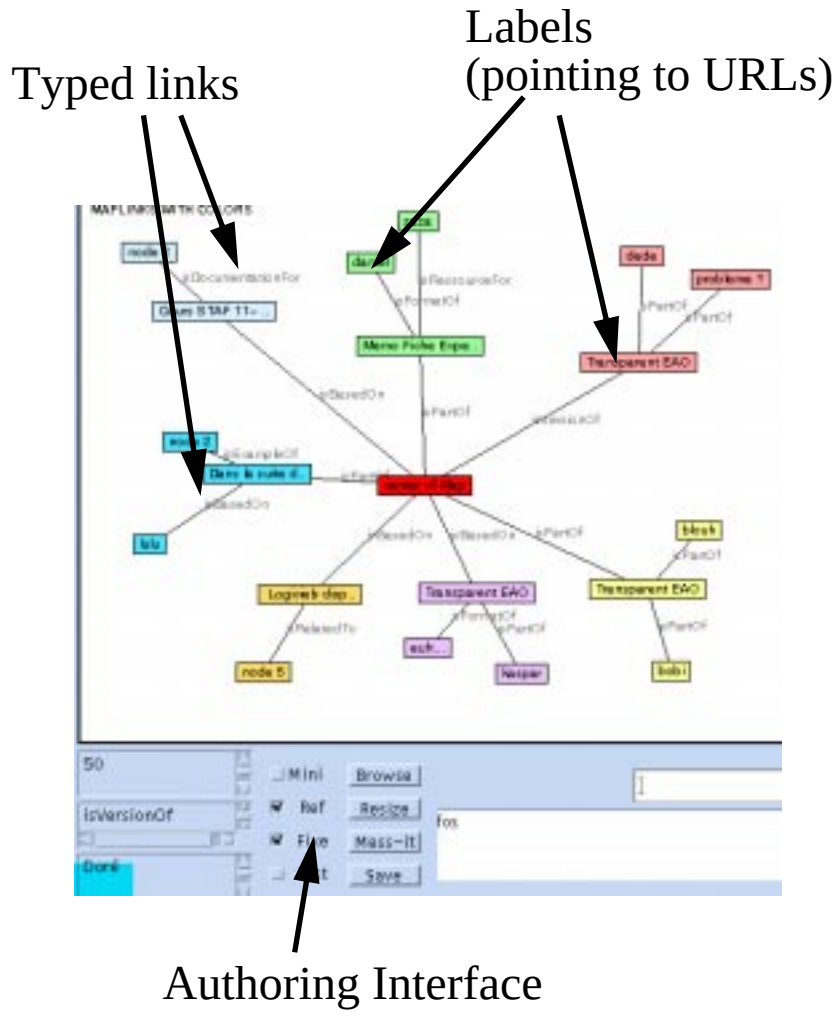
- Goal: maps of information space
  - to ease navigation
  - to organize knowledge
  - to clarify links between pages
- Related activities:
  - under construction
    (this tool is very new)

B. Other info tools

- E.g. RDF style classification of files

Motivation:

- Better support for less structured scenarios ...



5.5 Immediate Plans

1. **More activities**
2. **Work on information space**
3. **Integration of a multi-user environment**

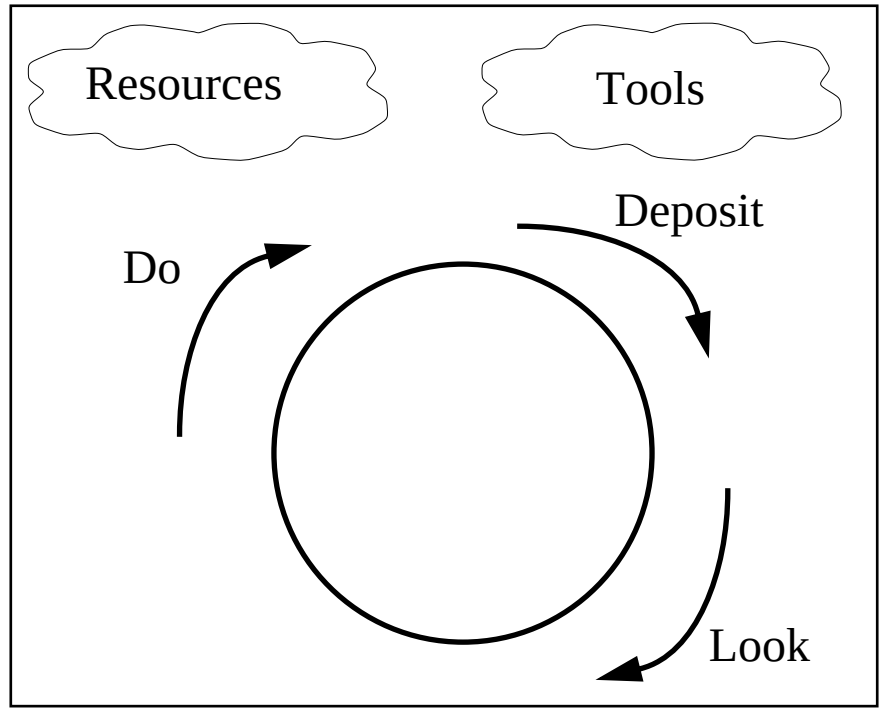
Why multi-user environments ?

- People collaborate better when they have a feeling of co-presence
  (same space = same context)
- People working in pairs can infer what the other is doing from his location
- 3D space metaphors might make hyperspace navigation easier
- Community building

5.6 Summary

A lot of structured activities, mostly:

1. **Students do something**
   - using tools and resources
2. **Results are collected**
3. **Debriefing, synthesis, feedback**



- tasks are often pseudo tasks
- these phases are sometimes repetitive
- most are collaborative
- collaboration is structured
- teacher can monitor, add, help, edit ...

Student acceptance

- Overally good

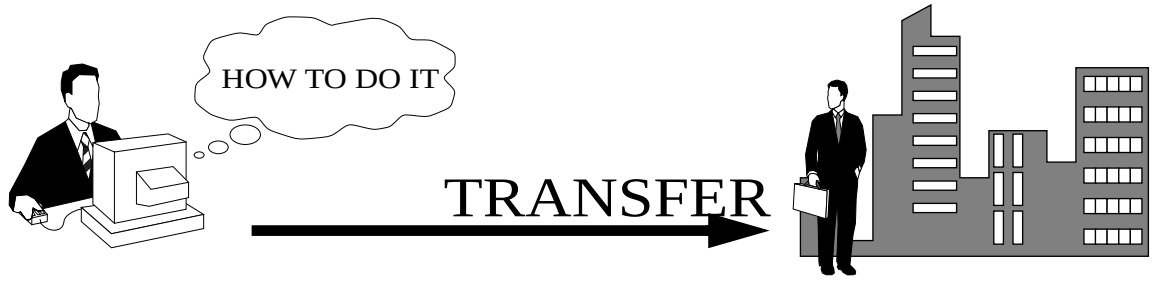
Assessment

- Unknown yet (well, our students like it)

A word of caution

- Some student projects/exercises are "open" and will remain so
  - We have to avoid the "video game effect"
  - We must plan the mastering of the content, not of the tool
- We are looking further into project management and information tools

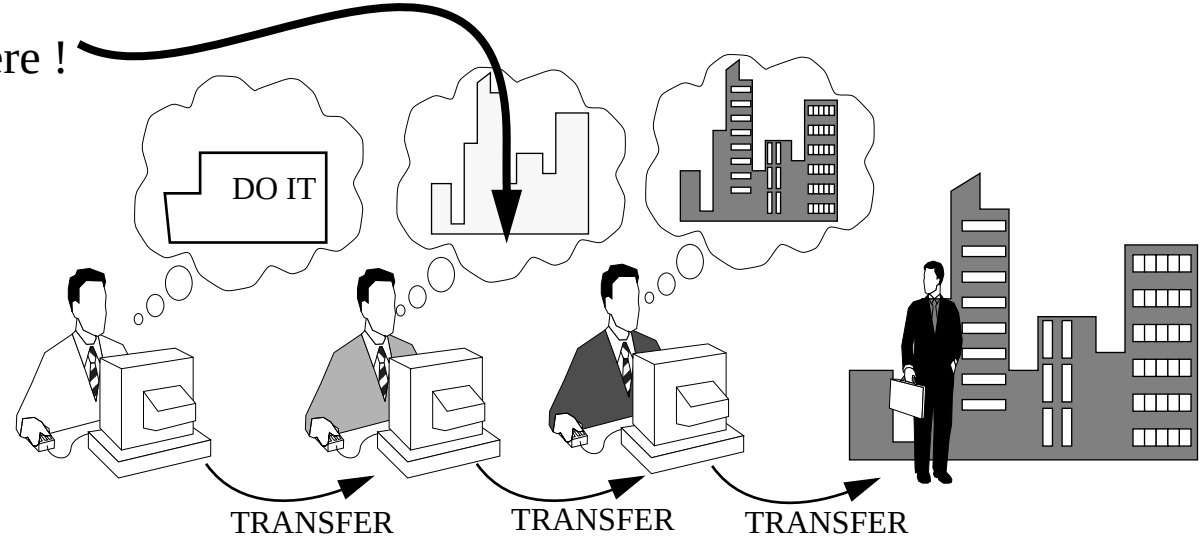
Traditional project based teaching



Don't stop here !

With:

scaffolding
guidance
learning tools

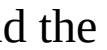


Problems

- "Mushroom style development" will make maintenance difficult
- Often decentralized authentication
  .... in addition to all the other tools...
- Tedious navigation:
  .... several clicks to get "there"
- Klunky HTML-based interfaces
- Yet more work for us...
  - cost/benefit yet undetermined
- Students don't have much of a collaboration culture
- Our information web is badly managed so far
  - resource collections
  - on-line manuals, etc.
- Nobody wants do the boring stuff:
  - navigation aids, updating, etc.
  - resource indexes, information sheets, etc.

We have ideas about our information web

- we just got started (XML, RDF, visualisation tools)



Mostly "behind the scene":

- bad software engineering,
- rapid prototype view of things
- learning while we are doing, ...
- lack of time

6. Practical Questions

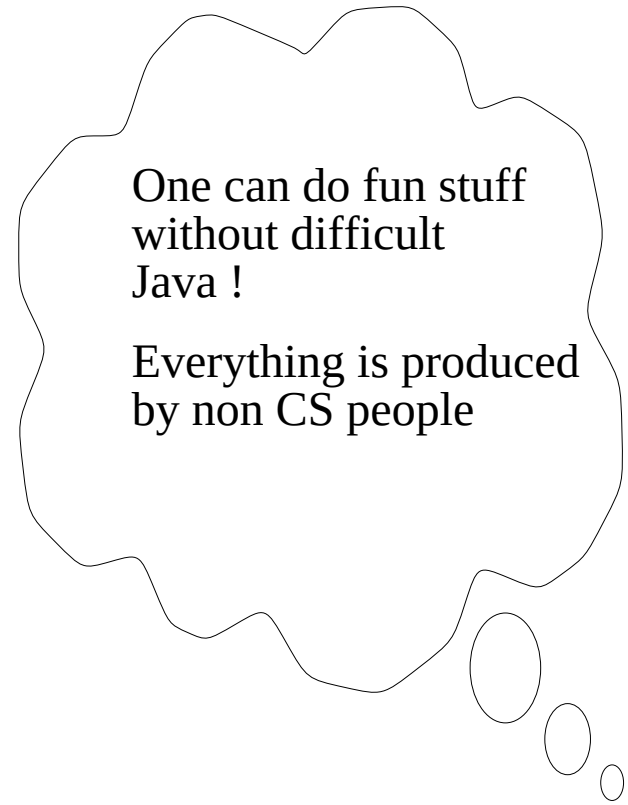
6.1 Technology used at Tecfa

Main:

- Simple HTML, soon: XML
- MySQL (a free relational SQL database)
- PHP
  - server-side html-embedded scripting language
  - supports SQL, XML, graphics, LDAP, etc.
- Apache Web Server

A bit:

- Javascript, Java (applets)
- Authorware and plug-in
- Python, Perl
- VRML
- Java Web Server (servlets)
- MOO



6.2 Closed vs. open environments and standard vs. custom tools

- See also: 7. “Outlook”

Our evaluation of WebCT and TopClass

- lack of flexibility (closedness)
- Incorporated tools don’t add much to the combination of separate tools (standard or custom)
- Student management is not that hot (no real project management tools)
- Authoring is not easy to learn (in particular finding work arounds)
- However we teach WebCT
  - STAF-17 (distance teaching)
  - a fine tool in non-innovative contexts

Course content Bulletin board Mail FAQ

Calendar Quiz Tools e.g. search Chat

A WebCT Menu

Lotus (not really tested at Tecfa)

- might be the good answer if you start from scratch
- but it’s a different world and requires manpower.

6.3 People

- It took us a quite a long time, mostly because we lacked ideas & resources

  1993: Our first WWW Server

  1994: Internet used in teaching:
  Web pages for courses and students, teaching materials, indexes,
  Email, forums, MOO

  1997: Experiences with commercial educational Web Servers
  First custom tools for activities

  1998: Start of our virtual campus project

- We can “do it” now:
  - because we take the best of our students as assistants
    (we teach the technology we use)
  - we now have examples from which each can copy
  - server-side html-embedded scripting languages are easy to learn
  - we can talk to each other
- I am the only full-time technical person
  - ..... but spend most time on exploring technologies, teaching and (some) system admin
  - All you need is one support person + graduate students who like to program

6.4 About ....

- http://tecfa.unige.ch/campus/infospace/index.php

Availability

- We will not make a product out of it
  - ... but any medium-rated programmer can copy ideas
- Guest accounts available upon request
  - need to understand french
  - most areas are locked (because activities need to be tied to a user)
  - we did not figure out how to quickly implement “just browsing”
- ... documentation for visitors is under way
  e.g. http://tecfa.unige.ch/campus/tecfarama/campustour.php

Pointers to technology used at Tecfa

- http://tecfa.unige.ch/guides/toolbox.html

7. Outlook

A. Closed environments

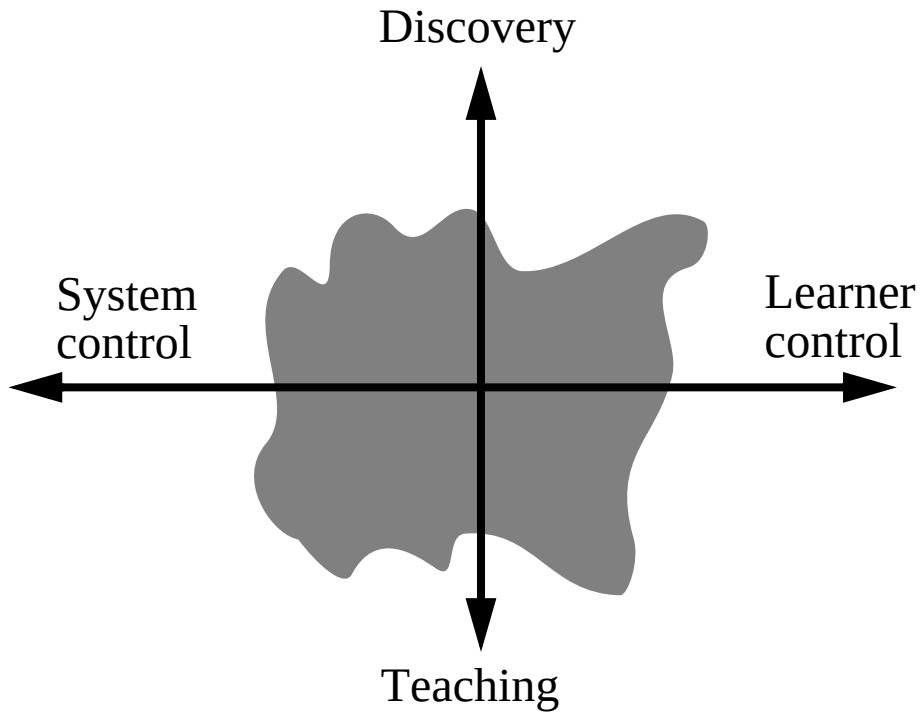
- Current integrated campus software is ok for traditional or project-based teaching
- It puts severe limits on “on-line” creativity
- It will become better, but will probably not do a lot in the next few years (unless it’s totally open source and easy to understand)

B. The open environment

- Standard communication tools (possibly better integrated)
- SQL server and maybe an object store
- Some Java applets and servlets
- Custom plugins (e.g. Authorware)
- LDAP (for central authentication and person look up)
- XML and less HTML (at Tecfa: Next Fall)
- Server-side html-embedded scripting languages (PHP, ASP, ColdFusion, JHTML)
- Advanced Groupware (like CVW, Workplace)
- Multi-user environment servers (DeepMatrix, Blaxxun, etc.)
- .....

C. Conclusion

- Student activities need some degree of monitoring and scaffolding
- Main stream virtual campus software is not flexible enough
- Internet supported teaching is expensive: *we* need more resources
  - teachers (and support) need to have some training or experience
  - but you get better results
  - ... and you do not necessarily need a “hot shot” development team



There is a wide range of possibilities

What counts is:
- to select appropriate strategies for a given goal
- support the learner in some way

A lot of things go
.... if they are well done !