QUIS - Quality, Interoperability and Standards in e-learning

2004-3538/001-001 ELE - ELEB14

QUIS analysis of commercial and experimental e-learning systems

Authored by the QUIS team.
Contact authors:
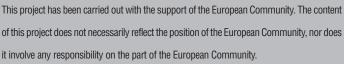F. Di Domenico, E. Panizzi, A. Sterbini, M. Temperini

Università La Sapienza
Roma, Italia
(marte@dis.uniroma1.it, panizzi@di.uniroma1.it, sterbini@di.uniroma1.it)

e Learning
Designing Tomorrow's Education

www.tisip.no/QUIS/

QUIS

Quality, Interoperability and Standars in e-learning

QUIS analysis of commercial and experimental e-learning systems

QUIS - Quality, Interoperability and Standards in e-learning

2004-3538/001-001 ELE - ELEB14

QUIS analysis of commercial and experimental e-learning systems

Authored by the QUIS team.
Contact authors:

F. Di Domenico, E. Panizzi, A. Sterbini, M. Temperini

Università La Sapienza
Roma, Italia
(marte@dis.uniroma1.it,
panizzi@di.uniroma1.it, sterbini@di.uniroma1.it)

QUIS

Quality, Interoperability and Standars in e-learning

Introduction

Work Package 4 describes and analyzes the main features of several Learning Management Systems with the aim to collect and summarize the state-of-the art of the products available both as commercial packages, open-source packages and research projects. This is done with the final goal of collecting a set of guidelines and suggestions from which the Work Package 6 of the QUIS project will profit, in order to describe and define requirements for the implementation of a "Next Generation LMS".
For each one of the LMSs considered, we have examined a wide set of features, trying to analyze an LMS both as a software system, by describing the main features and their possible impact on the different actors using the system, and as a learning tool, by analyzing the support given to pedagogical issues in each LMS.

The examined characteristics can be collected under the following set of main categories:

1. *Features available*: the features available to the teacher
2. *Users roles*: the support provided for different types of users: students, teachers, tutors, administration …
3. *Relationships with LMS standards*: standard used by the LMS for importing/exporting learning materials
4. *Usability*: information about usability studies on the LMS
5. *Pedagogical methodologies*: specific support for different pedagogical theories
6. *Installation*: details about the installation
7. *Maintainability*: how the software is updated
8. *Required hardware & software*
9. *Licensing model:* (open source/proprietary)
10. *Total cost of ownership*
11. *Usage:* (how many institutions etc)
12. *Support quality*: information about the available support at system level

The analysis of each LMS has been done here mainly by working on the documentation and the information collected about each LMS. In the following, a chapter of this report is dedicated to each LMS.
The examined Learning Management Systems are:

1. A-Tutor 1.5
2. Blackboard Academic Suite 6.2
3. Claroline 1.6.1
4. Microsoft Class Server 4.0
5. Desire2Learn 7.3
6. DoceboLMS 2.0.4
7. it's:learning 3.0
8. LAMS 1.1
9. LON-CAPA 1.3
10. Moodle 1.5
11. PaKMaS
12. Reload 2.1.1
13. Sakai project 2.0.0
14. WebCT Campus Edition 4.1
15. WebTeach/TWiki
16. WeBWorK 2.1

EP^2

To widen the analysis, we have developed a web-based system (called **EP^2**) to collect judgements about Learning Management Systems from LMS users. The system allows the user to express his/her judgment about an LMS as seen from one of 4 different roles: **student, teacher, LMS administrator, System Administrator**. For each role we have devised a distinct set of features, in order to allow to express evaluations on the role-related activities. In order to define such sets of features we had to do some choices.

Our final selection should not be considered the only one possible and acceptable, yet we think that these sets of features allows to build an effective comparison among the LMSs.

A user can express a judgement by selecting a role, then marking the perceived quality of the various features (possibly not all of them) in the system.

Users can also browse the expressed judgments and state how much they agree with each judgment examined: a judgment gains/loses importance depending on the number of agreements/disagreements collected.

The judgements can also be used to allows for the search of the "best LMS system" for a user, i.e. the LMS that has a better score in the set of features selected and weighted by the querying user.

To this aim, the user searching for "his best LMS" selects the set of features associated to a given role and assigns weights to each feature s/he perceives as more important. The EP^2 system computes the best LMS by considering both the weights given by the querying user, the marks provided through other user's judgements and the agreement/disagreement votes expressed about the judgements.

The EP^2 system is built as a web-based XML/PHP application.

Document organization

This document describes each Learning Management System in one separate chapter, following the scheme given below:

1) Features
    a. Contents
    b. Course's organization
    c. Learning assessment (quiz)
    d. Use assessment (survey)
    e. Learning assessment: assignment and journal
    f. Automated correction
    g. Managing learners' groups
    h. Access rights (to course and contents)
    i. Authoring tool   (tba see also: section 2a,2b,4c)
    j. Learning paths
    k. Calendar
    l. Hidden-in preparation material
    m. Synchronous communication: chat, videoconferencing etc.
    n. Asynchronous: Forums/topic forums/newsgroups
    o. Other
2) Users roles (details on use and functionalities)
    a. Student
    b. Teacher
    c. Tutor
    d. Administrator
    e. System manager
    f. Others
3) Relationships with standard
4) Usability
    a. Available usability evaluations
    b. Apprehension evaluations
    c. Accessibility
    d. Multi language support
5) Pedagogical methodologies
6) Installation
7) Maintainability
8) Required hw & sw
9) Licensing (open source/proprietary)
10) Total cost of ownership
11) Usage (how many institutions etc)
12) Support quality

1. LMS: A-Tutor 1.5

A-Tutor is an Open Source Web-based **Learning Content Management System (LCMS)** designed with accessibility and adaptability in mind. Administrators can install or update A-Tutor in minutes, and develop custom templates to give A-Tutor a new look. Educators can quickly assemble, package, and redistribute Web-based instructional content, easily retrieve and import pre-packaged content, and conduct their courses online. Students learn in an adaptive learning environment.

I. TEACHING FEATURES

a) Contents

As said above, content in A-Tutor can be entered manually, or created from HTML files and standards compliant content packages, or from a learning objects content repository. Content already existing in A-Tutor can also be exported into any of the above formats. Only instructors and students who are given content privileges can manage course content. By using only one tool, content can be created, its location, keywords, and date properties set, and glossary terms defined, it can also be previewed and have its accessibility checked (see also sections 1.k and 1.c).

Content can be created in either 'plain text' or 'HTML' mode. Plain text mode comes in handy for writing up text content quickly. HTML mode, despite being less straightforward, allows for extra features like text formatting and layout. In either plain text or HTML formatting mode, you can insert *code* which is useful for differentiating blocks of text (like math equations, program code, or quotations) from the rest of the text content.

A-Tutor allows importing and exporting of course content using IMS 1.1.3 or SCORM 1.2 content package specifications (see also section 3). Exported content can be viewed offline, and transferred to other systems. If enabled, students also can export content for offline viewing. An entire course, a chapter, or a single page of content can be exported as an IMS 1.1.3 or SCORM 1.2 content package. Exported packages are archived into a single file using ZIP compression. All content can be exported including the terms and glossary, colours, and code.

A-Tutor has a file system used for storing course resource files, and it is managed through the File Manager. The File Manager allows instructors to include files associated with course content into content pages. The File Manager also allows you to create, edit (in text or HTML mode), move, and delete files.

Links to external websites can be added to the student Links area, allowing course members to visit course related information elsewhere on the Web. Both students and instructors can add links, but student-submitted links must be approved by the instructor.

This system allows to publish announcement, too; they can be added through the manage section and will appear on the course Home page, sorted by date in descending order.

b) Course Organization

A-Tutor allows to group related courses in categories. This is helpful when viewing courses in the course browser. Courses can be associated with categories when created, or afterwards, by editing their properties.

Course properties include: description, export content (to toggle the availability of a course content for export), access (to restrict access right to specific categories, i.e. any user, only logged in user, or logged in and enrolled users) (see also section 1.g), initial content (to initialise the course content to be either empty, basic place-holder content, or a restored backup from other courses you teach).

c) Verifying the learning

This system allows students to keep an online journal, private or shared notes, associate notes with private or shared files, and print out compiled notes from within their personal work area.

Instructors can create self-assessment tests, that the students are allowed to take as many times as they wish. Automatically scored multiple choice and true/false, as well as random question tests are available. Instructor marked fill-in-the-blank questions can be added to self-tests, and custom feedback can be provided.

Instructors, and course members with test privileges, can create tests with multiple choice, true/false, Likert scaling, and a number of open ended question types. Multiple choice and true/false questions are marked automatically. A test release window can be created to make a test available only for a certain period, feedback can be customized, and test results can be archived. Self-marking tests can be created to provide students with instant feedback. Selecting from a pool of questions to generate random question quizzes and assigning tests to groups of students is also possible. Once a test has been created, teachers can add questions to the Question Database, and then add these questions to the new test.

The Question Database is where course test and survey questions are stored. Questions are created separately so that they may be reused in different tests and surveys.

Beside each question is a box in which a weight or mark for that question can be entered.

Using a button in each test page, the list of student submissions will be listed, and it can be filtered to show either all, marked or unmarked tests. Moreover, it is possible to view a test's statistics. There are two sets of statistics available for tests: the first is Question Statistics, showing beside each question the number/percentage of students who chose it, the second is Submission Statistics, showing each submission's overall mark and the marks given for each test question. The overall test average is also calculated.

d)   Verifying the use: survey

A-Tutor provides two different tools to look into final user satisfaction, Polls and Surveys.

A poll is a question with up to seven choices for answers. It's useful for quickly gathering course member opinions. Instructors and students with poll privileges can post it. Polls are not graded., and because Polls is a student tool, it can be enabled, disabled, and positioned according to the Student Tools preferences.

Surveys are the same as regular tests, with the exception that no marks will be assigned to questions (and no results be released), and in some cases it might be preferable to treat submissions as *a*nonymous. This can be done by choosing Yes from the anonymous property setting.

e)   How much knowledge: automatic correction

Instructors can create automatically scored true/false and multiple choice questions, and randomize test questions. Instructors can set dates and times when students can access a test. Instructors can provide individual feedback, override automated scoring, and create unit specific or course level tests. In fact in the Question Database, where all the question to be used in tests or surveys are stored, each question comes with a box to set a weight or a mark for the question; questions for survey (which are not evaluated) have an empty mark box (see also section 1.c).

f)   Organize a class: managing learners group

In A-Tutor, an instructor can create a group and add course members to it. This allows the instructor to administer different tests to different groups. When creating a test, a specific property allows to assign the test to existing groups. Moreover, instructors can appoint group leaders to create and manage groups. Each group comes with its own file exchange area, collaborative document authoring/versioning tools, discussion forum, chat room, group email list, and shared calendar.

Students can create study groups within their own forums and chat rooms. Students can send messages privately, or to a group. A shared notice board allows students to post news items or announcements.

g)   Who sees what: access rights to course and contents

A-Tutor allows to assign different access rights to different users, according to users categories. Students, instructors and administrators have different rights, but also among the students, for example, there could be different access rights. Through the Enrolment Manager, instructors can grant some students some instructor privileges, creating teaching assistants or co-instructors (see also section 2).

h)   Course creation: authoring tools

The software provides support for template-based content creation. An administrator or an instructor can create a course using the tools provided: moreover, they can share content through a central learning objects repository. Instructors can choose from the available course tools and menu modules, thus selecting the material to be used in a particular course. They can also optionally display tools in the main navigation bar, or link them into the course home page for quick access.

Instructor can create content in HTML or plain text (see also section 1.c). This content can either be imported from a local editor, or edited online. The system allows release dates to be set to control when content is viewable to learners. Content pages can be moved to different locations within a course. Related pages can be linked to content as references or relevant information.

Finally, instructors have access to all the learning tools available for learners. Help boxes are present throughout a newly created A-Tutor course to assist novice instructors with content development and course configuration. Full instructor documentation is also available in the A-Tutor HowTo Course (see also section 12).

Moreover, there are several ways to navigate an A-Tutor course. Some of these are Breadcrumbs, Global Links, Main Navigation, Sub-Navigation, and Content Navigation. There are also elements within content pages such as Previous and Next buttons, and Table of Contents. The Site-Map is available from either the course Home page or the Main Navigation page, depending on the instructor's settings.

i) Different users, different learning paths

Instructors can create both linear and nonlinear learning sequences. Moreover, by using groups, it is possible to determine learning processes, for example by grouping users by pre-existent knowledge or learning approach.

j) Time for everything: calendar

The system allows to use private, group, and course calendars to keep track of assignments, deadlines, due dates, etc.. But only instructors, group leaders, or group members can post to the calendar.

k) Hidden/in preparation material

In A-Tutor is always possible to see a preview of what one is creating: the system displays the content page as it will look with the formatting given. This is also how the content page will appear to a user, with custom HTML, Glossary Terms, Code, and colours.

l) Synchronous communication: chat, video conferencing, etc

This system provides a PHP-based chat tool for course or group level messaging. Students can see who else is online within their courses. The chat section is also used for managing chat transcripts. An active transcript will record all the chat messages as they are posted. There can be only one active transcript at a time and a unique name must be given to a new transcript when it is started.

A-Tutor doesn't provide tools to manage video services or shared whiteboards.

m) Asynchronous communication: forum, topic forum, newsgroup

The system allows administrators to create regular course-specific forums as well as shared cross-course forums.

A forum is an area within which course members can communicate in a structured manner through messages. The forums can either be enabled or disabled and can be linked from the main navigation, or the course home page, or displayed as a menu module. Course instructors and students with forum privileges can manage and moderate the forums by deleting, locking, and sticking threads and messages.

Posts can include URLs, and can be either plain text or formatted text. Discussion threads are expandable and collapsible: it is possible to view a list of topics or an entire conversation on one screen. Threads can be sorted by author, topic, post date, and activity level. Students can enable or disable notification of new posts sent to their email. Active threads move towards the top of the thread list.

For each thread in a forum, the following actions are available: stick thread (by using the exclamation point icon next to a thread: this keeps the specified thread at the very top of the forum's thread list and is useful for keeping important information visible to forum users), lock thread (by using the Lock icon, with two options - lock posting and reading, and lock posting only. Lock posting and reading closes the thread so that no one can read the contents or post replies, but lets the title of the thread listed in the forum. Lock posting only allows users to read the entire thread but not to post any replies) and delete thread (this will delete all messages within the thread and they cannot be undeleted).

Shared forums are special forums where members of different courses can engage in discussions with one another. The instructors, and students with forum privileges, of each participating course can manage and moderate the shared forum. Shared forums can only be created by an A-Tutor administrator.

Moreover students can use the internal email feature or instant messaging tool to send email or to communicate with other enrolled students.

n)   Other

The system allows to create a course backup, including all available course material within an archive in a format specific to A-Tutor. Backups are forwards compatible with future versions of A-Tutor but may not be backwards compatible with previous versions. Once a backup is created, it can be downloaded for safe-keeping, imported into another A-Tutor installation, used as a basis for a newly created course, and made available in the originating course's backup manager. Instructors can also create their own course backups from within a course.

II. ROLES

There are four types of user accounts in an A-Tutor installation, as defined by their Status: *disabled* (only administrators may disable an account and disabled accounts cannot login to the A-Tutor installation), *unconfirmed* (when the email confirmation required upon registration has not been given yet), *student* (regular account which can enrol in, but not create courses) and *instructor* (regular account which can enrol in as well as create courses). In order for a user to login to the A-Tutor system, a unique system account needs to be created; and if you have forgotten your password, just enter your email address into the password reminder screen, and the login and password information for that address will be mailed immediately.

a)   Learners

Students can configure the layout, tools displayed, presentation of text and graphics, etc. and save those setting for future visits. They can upload files in most document formats to a shared course library, or to a shared group library, and can share content from their personal folder with other students, and with an instructor or teaching assistants. Students' personal and public folders can be shared with other students, with group members, with a course instructor, or with teaching assistants.

Students can use keywords to simultaneously search all of their courses or all available courses.

b)   Authors

Instructors privileges include the management of the whole course. Also, they can allow to other users limited access to administrative and instructional tools based on pre-defined roles or permissions, and can create teaching assistants or additional instructors, each with their own custom privileges. Instructors can batch add students to a course using a delimited text file.

Concerning students evaluation, instructors can assign partial credit for certain answers, and can view grades in the gradebook by assignment, by student, and for all students on all assignments, they can also delegate the responsibility for grading assignments. Obviously, instructors can manually edit all grades.

Finally, in order to monitor the attendance at the course, instructors can get reports showing the number of times, the time and date on which, and the frequency with which each student accessed course content, discussion forums, and assignments, and can review the navigation record of each student.

c)   Tutors

In A-Tutors users who are only enabled to teach (and not to author a course) have no specific name. This role can be obtained giving different access right to instructors.

d)   Administrators

An A-Tutor installation can be maintained by multiple administrators, each with their own privilege access level. The three kinds of administrator accounts are: Super Administrator (has no restrictions and has access to all administrator options. This is the only administrator type that can create and delete other administrator accounts. There must always be at least one Super Administrator account), Active Administrator (with

limited access. This means that s/he only has privileged access to some sections, assigned when their account was created by Super Administrator) and Inactive Administrator (has not been assigned any access privileges. As a result, this administrator cannot login).

An administrator can create and manage courses, shared forums, and course categories without having to login as a course instructor. Administrators can set courses to be publicly accessible, or can protect access to individual courses with a username and password. System has a password reminder option. Administrator can manually add users to the system. Manually created accounts are automatically confirmed and the account status is set to Student, Instructor or Disabled as chosen in the Account Status field of the user account creation form. User accounts can also be created by individuals using the registration form available through the public pages of A-Tutor.

e) System managers

This kind of user, in terms of A-Tutor configuration, can be represented as the Super Administrator (see section 2.d).

f) Other

III. RELATIONSHIP WITH STANDARD

The software supports the IMS Content Packaging Specification 1.1.3 and has self-tested compliance with SCORM Content Packaging, allowing content developers to create reusable content that can be swapped between different e-learning systems. Content created in other IMS or SCORM conformant systems can be imported into A-Tutor, and vice versa. A-Tutor also includes a SCORM 1.2 Runtime Environment (LMS RTE3) for playing and managing SCORM based Sharable Content Objects (SCOs).

Moreover, A-Tutor is renowned for its compliance with the accessibility specifications, i.e. users with disabilities are able to fully access the system (see also section 4.c).

IV. USABILITY AND LEARNABILITY

a) Usability test: existence and results

No documentation has been found about A-Tutor usability tests.

b) Learnability test

No documentation has been found about A-Tutor learnability test.

c) Accessibility

A-Tutor is the first fully inclusive LCMS, complying with the W3C WCAG 1.0 accessibility specifications at the AA+ level, allowing access to all potential learners, instructors, and administrators, including those with disabilities, who may access the system using assistive technologies. Conformance with W3C XHTML 1.0 specifications ensures that A-Tutor is presented consistently in any standards compliant technology.

Many aspects of the system are obviously designed while keeping accessibility in mind. For example, there's a tool to perform analysis of the content for accessibility problems. It gives recommendations and you are given the option to implement or reverse corrections. When all known problems are solved, and decisions have been made on all potential problems, a "full pass" will be displayed, after which you can be sure the content will be accessible to all your students.

Another example of accessibility features is the online course included to help students, and instructors, to learn how to use the system. Students are provided with context sensitive help for any tool, or for fields within tools, and a functional feedback after each action indicates successes, warnings, or errors.

d) Multi-language support

A-Tutor can be displayed in many different languages. Systems are made available as importable package from the atutor.ca website only it they are completely translated. Moreover, the main A-Tutor website

provides a whole forum to discuss the translation of the system in different languages, and the most posted threads are about translation in Chinese, Russian, Vietnamese, and other rarer languages.

V. PEDAGOGICAL METHODS

As for other LMS systems, A-Tutor components (wiki, forum, quiz etc…) allow the free (hand-made) construction of courses following different pedagogical approaches depending on the Teacher preferences. A-Tutor has no specific automatic support for pedagogical methodologies, e.g. to course personalization (even if the Teacher could hand-craft several different versions of his Course for different groups of students). The tools provided (forums) allows the Teacher to follow Socio-constructivist approaches.

VI. INSTALLATION

The installation procedure is rather simple for a normal system-administrator.
After the installation, institutions can create their own look and feel templates, can apply their own institutional images, headers and footers. Instructors can alter the appearance of their courses, change the navigation icons and colour schemes, and change the location of the navigation menus for a course. Custom tools can be created and quickly added and removed from a course.

VII. MAINTAINABILITY

A-Tutor is based on the work of a dynamic development community, recognized feature of successful open source projects. The system was originally released in December 2002: known installations list only reports 18 servers, the present installation list is likely to be much longer, at least tenfold. Since then A-Tutors has grown steadily, with a succession of new features producing a new version every three months, the latest version, 1.5, has been released in June 2005.

VIII. REQUIRED HW or SW

The software supports any browser although Internet Explorer 4+, Netscape 6+, and Opera 5+ are recommended, requires a MySQL database and requires PHP, MySQL and web server software such as Apache or Microsoft IIS.

IX. LICENSING: OPEN SOURCE or PROPRIETARY

A-Tutor is Open Source software, distributed under the terms of he GNU General Public License.

X. TOTAL COST OF OWNERSHIP

In terms of hardware and licensing cost, A-Tutor is very cheap, because all licenses required are F/OSS (Free/Open Source Software) and hardware requirements are reasonable. Development and support, though, are not easy, because even if the code is straightforward, it lacks structure and it is unclear where new functionalities are to be added. About maintenance, the cost of ownership is low because configuration files are easily managed, there is more than one logging available through Apache and PHP, and logs contain warning during normal operation.

XI. USAGE

Since A-Tutor has been first been released in December 2002, it is difficult to track all users of the system, since they are not obliged to register when they download or install A-Tutor. In the A-Tutor community there are approximately 4000 registered members, though the total number of users is likely to be much larger than that, as this number only represents registered administrators: instructors and students users are not included. Since collection of installation statistics, which is also optional, has been included in A-Tutor 1.4.3, there have been about 3500 installations; the actual number is probably somewhat larger than that.

XII. SUPPORT QUALITY

A-Tutor's base in Open Source technology makes it a cost effective tool for both small and large organizations presenting their instructional materials on the Web, or delivering fully independent online courses. Comprehensive help is available through the A-Tutor HowTo documentation, through a number of support services, or through the public forums, and full language support is available through the A-Tutor Translation Site: instructors can access an online instructor manual, context sensitive help, and an instructor support forum hosted on the product provider's site.

2. LMS: Blackboard Academic Suite 6.2

Blackboard Inc licenses two main products.

The Blackboard Academic Suite includes: Blackboard Learning System (online courses and virtual learning environments), Blackboard Content System (enables users to collect, share, reuse and manage content) and Blackboard Community System (provides both an academic portal and eCommerce capabilities).

The Blackboard Commerce Suite includes: Blackboard Transaction System (administers online, real-time eCommerce, meal plans, vending and event access through a campus one-card), Bb One (enables the acceptance of the university ID card as a form of payment off-campus by connecting local merchants to student accounts) and Blackboard Community System (provides both an academic portal and eCommerce capabilities).

This report evaluates Blackboard Academic Suite because it's more suitable for our purposes.

Blackboard Inc. has recently acquired the WebCT company, thus we expect that the two systems will merge.

I. TEACHING FEATURES

a) Contents

Instructors can specify start and stop dates for the entire course.

Within each course, the control panel allows the management of course content areas, tools users and assessments.

Many types of learning objects can be included in a course: learning unit, survey (see also section 1.d), assignment (see also section 1.c), discussion board (see also section 1.m), chat (see also section 1.l), virtual classroom (see also section 1.l), group (see also section 1.f), syllabus, offline content, SCORM content (see also section 3), glossary (a list of terms that are commonly used within the course. Instructors can either create a glossary from scratch or upload an existing one from any common delimitated file) and more.

The system supports content sharing across course and institution boundaries. The system provides a central content repository where course content files can be stored and accessed by other instructors. In fact, instructors can designate their files to be private or publicly accessible. Content files can also be shared with specific groups or users as well as with individuals outside of the system. Instructors can create links to content files in the central content repository so that changes made to the linked content are automatically displayed in their courses.

b) Course Organization

In Blackboard an accurate course organization in categories is not available. Courses are simply presented as a list in the user's home page.

c) Verifying the learning

In Blackboard the three most important means to check learners level of knowledge are test, assignment and self-assessment.

A test needs only the specification of some basic properties: name, description and instruction. Many additional choices are available if desired: i.e. feedback for individual answers, addition of images, files (also audio, video and flash) and external links to questions and/or answers, questions classification by categories and keywords, setting of a default point value for each question or of particular credit options, randomization of the answers, different layouts of answers presentation, and, last, several numbering options for answers.

There are several question types, allowing instructors to create many different kinds of text: calculated formula (allows questions to contain formulas whose values change with each student attempt), essay (provides a free form text area in which students can enter their answer), short answer (similar to essay, but allows the instructor to limit the amount of text entered), file response (requires students to upload a file as their response), fill in multiple blanks (enable instructors to prompt students answer to complete multiple word missing within a sentence or paragraph), hot spot (enables an instructor to enter an image and designate the coordinate in which the answer resides; a student would have to click on the area that correctly answers

to the question), jumbled sentence (presents a drop down of answers to fill in multiple blanks within a sentence), and other types.

An assignment can be added to any content area and its set up is similarly straightforward, as only a few variables need to be set up: name, number of points (this value will appear in the course gradebook and can be used as part of calculating a student's overall grade), instructions, any attached files being part of the assignment, assignment availability, tracking for the number of views (if applicable), and any date restrictions concerning the assignment.

Finally, instructors can create anonymous timed or un-timed self-assessments that students can take more than once. Moreover, instructors can create a database of questions that the system will randomize to create a unique self-assessment for each student. Assessment questions can be given all at once or one at a time, can be timed or un-timed, and assessments can be taken multiple times or only once.

d) Verifying the use: survey

In Blackboard a survey can be created just like a test: there are the same basic properties to define, and the only difference is that for this test there's not evaluation. In particular the option scale/Likert scale question type is ideal for surveys.

e) How much knowledge: automatic correction

Instructors can deliver online, automatically-scored assessments and surveys. They can create these assessments from scratch or draw from personal, institutional, or available "test banks" of questions. The system automatically scores multiple choice, multiple answer, ordering, matching, fill-in-the-blank, and true-false questions, and can display instructor-created feedback and links to relevant course material. Also, instructors can create different levels of feedback messages, override the automated scoring and determine how to communicate test results to students.

Instructors can differentially weight tests and create grading rules. The system provides analysis data for surveys and test item results can be exported for analysis.

Instructors can publish student submissions as examples for other students to see.

When an instructor adds an assessment, or assignment, to the course, the software automatically adds it to the gradebook. Within the grade book, instructors can create a course grading scale that can employ raw scores, percentages, letter grades or pass/fail metrics, and then they can sort and view grades in the gradebook by assignment, by student, by category and for all students on all assignments. In addiction, they can search the gradebook to find all students who meet a specific performance criteria, mark, or status such as exam completion. Instructors can also add the grades for offline assignments to the online gradebook, or manually edit all grades.

In every moment the gradebook show also an automatically calculated average grade on each assignment, to compare each student result with the others.

Finally, instructors can download the results of a test across sections of a course and across semesters, and they can import and export a comma-delimited version of the gradebook from/to an external spreadsheet program.

f) Organize a class: managing learners group

To support peer collaboration, instructors can use the groups tool to form different groups of students. While creating a group there are some properties to set up: name, description and some communication tools availability. In fact each group can be given its own file exchange area, discussion board, virtual classroom and a group email tool to send messages to all group members. Then, an instructor might assign different groups for different assignments or projects.

To manage a group project, a discussion board can be set up for the group members to use for debates and conversations on their assigned topics. The group members also have access to collaboration tools including a virtual classroom and a chat environment. An easy-to-use email facility is also available. Also, instructors can set up shared file space for each group.

Students can belong to multiple groups simultaneously: to add a new user to a group, instructors can use the search tool (by name, by identifier or by email address) to find within their course those they wish to add. Finally, instructors can also list and remove groups members.

Students too can create online clubs, and also interest and study groups at the system level.

g) Who sees what: access rights to course and contents

As for course availability, instructors can choose between three options: available (or not) to all enrolled users, to a guest user and to an observer user (s/he can view only the first page of the course). Each of these options is independent from each other, so, for example, a course can be available to observer and enrolled user, but not to guest. In addiction, it's possible to set up an enrolment key to access course for the first time. Within a course, it's possible to define access right for learning objects, for example, restricting access to a group of users.

h) Course creation: authoring tools

Authors can quickly switch between the student view and the edit view clicking on a link in their contents pages.

Instructors can create a course from scratch, or can easily create it using the course creation wizard: it builds the structure of a course through a series of steps presented in an easy to use interface. Instructors can create a new course, or can create a course using an existing course template provided by their system administrator. Course materials can also be created by copying an existing course. Last but not least, the wizard also allows a course to be created using a course cartridge, which provides a complete set of content and tools and is supplied by an academic publisher.

During the course creation, wizard makes it easy for instructors to enter the necessary settings, such as the name and description of the course, the availability for guests or observers, the duration, the enrolment options, the categorization of the course in the course Catalogue, the entry point for the course (the area viewed by users at their access in the course) and the localization.

Within the wizard the text editor provides a rich text editing interface, including WYSIWYG and spell-check, to create effective learning content. Moreover, instructors can import e-learning content created in external authoring tools such as Macromedia Dreamweaver, Microsoft Frontpage, or any SCORM-compliant authoring tool.

i) Different users, different learning path

The system, by means of the adaptive release functionality, allows each instructor to create custom learning paths through course content and activities. Content items, discussions, assessments, assignments, or other activities can be released to students customizing a set of criteria, such as: date/time, username, group membership, institution role, grade on a particular test or assignment, or precedence (i.e. a certain content can be accessed only if the user has previously reviewed another piece of content).

The first option is to control the content availability by date. Another option is to select specific users to release content to. Instructors can either enter the user's system id (or a list separated with comma), if they know it, or browse to find it. An instructor can also release content to entire groups. Content release can also be linked directly to a gradebook item and specific criteria around it. Once an item is selected, an instructor can set content to be released only when a student has attempted a certain graded item, or if the student scored greater or less than a specific value, or when a student scores between two set values.

Instructors can add as many rules as they wish. By having multiple release rules, an instructor can indeed design various path within their course, based on student activities.

j) Time for everything: calendar

In Blackboard a calendar tool allows a user to manage and view course-specific events and announcements, populated by a course's instructor, as well as personal and institutional events. Instructors can assign tasks by using the calendar, assigning a priority and a due dates (to students, individuals or groups) and then, they can enable users to track their progresses by using specific options.

k) Hidden/in preparation material

While creating whatever Blackboard content, there is always the option to make the object unavailable; so if an instructor is preparing a document, a test or anything else, it's possible to prevent other users to see it.

l) Synchronous communication: chat, video conferencing, etc

The collaboration tool within Blackboard, designed for live, synchronous interaction, supports a text-based chat environment, as well as a full virtual classroom.

The chat environment supports unlimited simultaneous group discussions and private messages. Instructors may moderate chats and suspend students from the chat rooms. The system creates archive logs for all chat rooms. Instructors can view chat logs and share them with students.

In addition to text-based chat, the virtual classroom provides a collaborative whiteboard, group web browsing (web touring), private question-and-answer, and breakout room capability. It can be run in lecture mode or an open participation mode. Users can "raise their hand" to be called on or given full participation control.

While leading a virtual classroom instructors can access a variety of tools: they can navigate through the course (to find easily a piece of material to share with the session's participants), control a common whiteboard, set up a shared group browser in which they can navigate and share the content of any website with their own class, pose a question to their own class or click on their own inbox to review and respond to any questions that have been sent them.

Blackboard doesn't provide any video services.

m) Asynchronous communication: forum, topic forum, newsgroup

The discussion board tool within Blackboard enables threaded, asynchronous discussions. Instructors can set up multiple forums around different topics and embed those forums in appropriate content areas or lessons: instructors can associate a discussion with any course content. Instructors can determine whether students can modify, delete, post anonymously, include attachments, and other options. The threaded discussion software includes a formatting text editor which can create mathematic equations. Forums can be sorted/viewed by thread, author, date, or subject and are completely searchable: a user can set various search options to quickly find the posting of interest choosing to search in the subject, in subject and body or in all the post, and then to make a exploration by author (two options are "all enrolled in forum" or selecting only one user in the list of enrolled) or by keywords (two options are "any of inserted" or "all inserted" words).

Each forum has a title, description and count of how many messages are in the thread. There is also a note on how many messages have not been read by the user accessing the forum. Discussion threads are expandable and collapsible to view an entire conversation on one screen.

About file exchange, Blackboard provides students with a private folder into which they can upload and download files. Students can upload files to a shared group folder and instructors can upload files to the personal folder of a student. Students can also exchange content outside of course boundaries.

Finally students can use the Blackboard messages to have an email-like communication within a course without relying on external email systems or addresses.

n) Other

Instructors can publish course content on a CD-ROM that can be linked to dynamically from within the online course or viewed offline. Students can download course content into a format that can be accessed on a mobile device and synchronize calendar events with a PDA.

II. ROLES

Within Blackboard, a user can only be assigned one role. There are five user roles to choose from the system: instructor, teaching assistant, grader, course builder and student.

a) Learners

Students can access both communication and general course tools clicking on a link in the course home page. Communication tools are: announcements (displays all the announcements for the current week), collaboration session (can either be set up as a chat or a virtual classroom, see also section 1.l), discussion board (specific topic open for discussion where students can add to and review the threads, see also section 1.m), group pages (quick access to information pertaining to whichever group students have been assigned to, see also section 1.f), messages (allows the members of a course to exchange messages with one another, see also section 1.m), roster (allows a quick search of the students that are enrolled within the course) and send email (allows a student to send email messages).

Course tools are: calendar (allows students to see scheduled events for their class, see also section 1.j), course portfolios (presents related content in an organized and interactive manner - it may be private, shared

with other courses, other users or publicly available), dictionary and thesaurus, glossary (provides a list of terms commonly used within the course), my grades (allows a student to quickly access a report of their grades in the course), tasks (list of tasks required for the course) and user manual (online user manual).

Students can create a personal home page: it can include a photo, personal information, and links to important websites. This page contains also a list of all courses in which the student is enrolled, all their grades and all course and system-wide events and tasks.

Finally, students can make private notes about their course, and can create bookmarks in a private folder or share them in their portfolio.

It's important to note that if administrator changes the role of a student enrolled in a Blackboard course to teacher assistant or grader (see below), he will be removed from the gradebook and will not be able to submit work in Blackboard.

b) Authors

The instructor has master access over the Blackboard course functions.

Instructors can get reports showing the number of times and date on which each student accessed course content, discussion forums and assignments. They can also flag individual course components to track the frequency with which students are accessing those components.

Instructors can store student performance results in the course Gradebook. Scores from assessments delivered through Blackboard are automatically recorded in the gradebook. The gradebook supports custom grading scales, grade weighting, item analysis and multiple gradebook views. With the instructor's permission, students can view their own grades (but no one else's) in the course gradebook.

c) Tutors

This role can be described as similar to course builders. They have access to most content functions, but they cannot set assessments or polls, cannot access the gradebook and digital dropbox nor course statistics, and cannot create entries and course tasks.

d) Administrators

Administrators can batch add students to a course using a delimited text file (or students can self-register). They can also batch create courses, users, and enrolments in the system. The software supports integration with student information systems through an event-driven API or through a special tool based on scheduled system extracts. Administrators can distribute the roles across multiple institutions to facilitate school board-wide or consortia-style approaches.

Moreover, administrators can set courses to be publicly accessible or protect access to individual courses with a username and password.

Administrators can create an unlimited number of custom roles with specific access privileges to course content and tools. Instructors or students may be assigned different roles in different courses. Administrators and instructors can assign different levels of access to the system or to specific courses based on pre-defined roles. Administrators can delegate administration of the system to different organizational entities within an institution or consortia.

e) System managers

In Blackboard a specific role for this type of user doesn't exist.

f) Other

In Blackboard a teaching assistant have almost the same access level of an instructor, with the only exception that s/he cannot add instructors and teaching assistants to a Blackboard course. The role of teaching assistant is typically assigned to faculty assistants, teaching assistants or administrative assistants who help to develop and maintain course materials, communications, assessments and grade.

On the other hand, a grader is able to access all areas under assessment, so s/he can manage an existing assessment, for instance evaluating or deleting students attempts.

III. RELATIONSHIP WITH STANDARD

The software has been issued a certificate of compliance with SCORM 1.2 level LMS-RTE3. The system supports the following standards: IMS Metadata vocabulary, IMS Content Packaging 1.1.2, IMS Question & Test Interoperability Specifications 1.2, IMS Enterprise Specification 1.01. The system includes tools to facilitate the migration of course content between different versions of the software.

IV. USABILITY AND LEARNABILITY

a) Usability test: existence and results

It wasn't easy to find documentation about usability test for Blackboard. The Blackboard brochure contained only these few sentences: "Ease of use was a major reason that Red Rocks Community College switched to Blackboard after using another e-Learning platform for 3 years. Red Rocks also needed a scalable and reliable course management system that could support their whole institution. Rebecca Woulfe, Director of e-Learning, noted that faculty training is much easier using Blackboard. In addition, "student emails to the help desk at the beginning of the semester declined from 80 -100 emails per day with WebCT to 10 - 20 emails per day using Blackboard", according to Woulfe.".

b) Learnability test

In the Blackboard brochure, about learnability it is claimed: "The unique power of the Blackboard Academic Suite is that as a unified solution it enhances and maximizes the independent strengths of three best-of-breed applications: the Blackboard Learning System, Blackboard Community System and Blackboard Content System. Rather than learning to use multiple applications, once students and instructors become familiar with one Blackboard application they are familiar with all Blackboard applications.".

c) Accessibility

To comply with Section 508 of the US Rehabilitation Act, the software implements the following features: a tool for instructors to add alt tags to uploaded images, alt tags on all system images, data tables that are optimized for use with screen readers, documentation to assist students and instructors in the use of the assistive technologies the software supports, appropriately titled framesets describing the functionality of the frames layout, and support for most screen reader technology including JAWS.
Developers and designers on the staff of the product provider are trained in the use of Section 508 standards and development techniques.

d) Multi-language support

Blackboard enables institutions to run multiple languages on the same system. Instructors can even set the language of the course independently from the language setting of the overall system.
The system is currently available in English, Spanish, Italian and French versions. In addition to supporting most European languages, Blackboard supports multi-byte character sets such as Japanese and Chinese.

V. PEDAGOGICAL METHODS

About pedagogical approaches Blackboard claims: "Perhaps the most exciting promise of the NLE (Networked Learning Environment) is its ability to put the individual student at the centre of a more expansive learning process. Pedagogically, that means techniques like support for alternate learning styles, individualized programs and adaptive release of content and activities. It also means better integration with all aspects of student life. Not only is learning improved, so is recruitment, retention and alumni communication.".

VI. INSTALLATION

The product provider offers a **hosted system** that includes 99.75% Service Level Agreements with guaranteed system availability and performance on a network of high-performance, fault-tolerant servers with fail-over capability including load balancing and clustering, managed software installation, redundant Internet connections, redundant and conditioned power, redundant hosting platforms, 24x7x365 monitoring, 10GB – 20GB storage space to start, managed bandwidth usage, redundant T3 connections, daily data backups and weekly tape backups, and a secure facility with environmental control and a modern alarm/security system.

VII. MAINTAINABILITY

Blackboard is a popular software widely used around the world and partially owned by Microsoft and Sun. Blackboard has created a community of developers: more than 200 independent developers at educational institutions and software companies take part in the Blackboard Developer NetworkSM, plus countless others who develop software solutions with the free Blackboard Building Blocks SDK. There are more than 115 community-developed applications available in the Blackboard Building Blocks catalogue.

VIII. REQUIRED HW or SW

As a client browser, Blackboard supports Internet Explorer 6.0, Netscape 7.1+, and Firefox 1.0 on Windows PCs. The software supports Internet Explorer 5.2, Netscape 7.1, Firefox 1.0, and Safari 1.1 and 1.2 for Mac OS.
Regarding databases, the system requires either SQL Server 2000 on Windows 2000 Server or Windows 2003 Server, or Oracle 8 or 9 on Sun Solaris 8 or 9 or RedHat Linux.
Finally, regarding server software, Blackboard requires the Apache web server software or Microsoft Internet Information Server (IIS) software, and the Java 2 SDK.

IX. LICENSING: OPEN SOURCE or PROPRIETARY

Blackboard Academic Suite is a software released from Blackboard Inc.
The company offers a range of professional service offerings including consulting (strategic planning, implementation, and customization services), training, ASP services, and 24 x 7 support (Enterprise version).

X. TOTAL COST OF OWNERSHIP

The Blackboard Academic Suite is not itself versioned but is released within the Learning System and Community System Release 6.2 and Content System Release 2.0.
The annual license fee is based on number of FTE students in an institution (or school within an institution) or consortium. In some markets, the annual license fee is determined on a per-user basis.

XI. USAGE

The system can support multiple institutions, departments, schools or other organizational units on a single installation. More than 12 million people at over 2,000 institutions in 35 countries use Blackboard solutions. Blackboard is used by over 45% of U.S. postsecondary institutions that use course management systems.
For example, according to Dr. James Frazee, Associate Director, Instructional Technology Services at San Diego State University (SDSU), "Scalability and integration with our student information system were key to our success with Blackboard. Our adoption rates have been outstanding. In 1999 we had only six instructors using Blackboard. Now Blackboard is used by more than 700 faculty in over 1,200 courses. We could only

support that kind of growth with a scalable system that interacts with other mission-critical systems on campus. Blackboard continues to be our choice for e-learning at SDSU."

XII. SUPPORT QUALITY

Having a wide client base, Blackboard provides various types of support: a dedicated technical support manager (working with users on the phone or via email), self-help materials (created by a special documentation team), helpful tip sheets (for faculty and students) and account managers (within Blackboard site).

3.   LMS: Claroline 1.6.1

Claroline is an Open Source software based on PHP/MySQL. It's a collaborative learning environment allowing teachers or education institutions to create and administer courses through the web. The system provides group management, forums, document repositories, calendar, chat, assignment areas, links, user profile administration on a single and highly integrated package.

I.   TEACHING FEATURES

a)   Contents

Claroline courses can contain different content types: *agenda* (to display the course appointments ordered by date), *documents* (the area where instructors upload files, with an optional description to help students to know the file content without opening it), *announcements* (to display any information instructors wish to communicate. Instructors can choose to send the announcements both to this area and directly to students private e-mail), *forums* (see also section 1.m), *groups* (see also section 1.f), *links* (a list of links to useful web pages, as chosen by instructor), *student papers* (the area where students can upload them own files, to share material or to attempt assignment requiring file delivery), *users* (an area where is possible to see a list of all course members, instructor and students, and is possible to contact each of them using his private email addresses) and *content documents* (created within the system or imported, can be in several format such as pdf, doc, txt, html, organized in directories and sub-directories).
The course main page contains all the features visible to the students; however, additional options and sections are available for teachers on this page.

b)   Course Organization

Using Claroline it is possible to organize courses in categories, indicating name, number and type of courses contained and a short description of the features shared among them. It's always possible to modify their order, and even to set them in a level organization.

c)   Verifying the learning

Instructors can create self-assessments, including multiple choice, true/false, fill-in-the-blank, matching and multiple answer type questions. They can also contain images, and can display feedback either instructor-created, or produced by automatic correction (see also section 1.e). Instructors can create personal test banks, and the system can randomize the questions in a test.
Assignment tool is a publication area for students. The course manager can create one or several assignment (different zones of publication) where the students will be allowed to publish their work. Each assignment can have different objectives or requirements: publication may be a file, or a text or a text and a file, publications can be authored by groups or by single users... Also, students can be authorised to modify their work after submission and the course manager can give them feedback.

d)   Verifying the use: survey

Claroline doesn't provide any tool to insert surveys, or anonymous tests, to evaluate courses or instructors.

e)   How much knowledge: automatic correction

Instructors can create automatically scored multiple choice, multiple answer, fill-in-the-blank, true/false and matching questions, collected in tests or self-assessments, to give students feedback or to grade them after each assignment attempt.

f)   Organize a class: managing learners group

Instructors can create groups and assign students to them or allow self-registration.

Groups can have documents area for files exchange between group members: it is possible to upload, rename, delete and move files, create directories and subdirectories. Every file format is admitted, but not .php files. Each group can have its own internal discussion forums, too.
Students themselves are not allowed to build communities.

g) Who sees what: access rights to course and contents

Instructors can set courses to be publicly accessible or can protect access to individual courses with a username and password. The system has a password reminder option.

h) Course creation: authoring tools

All course contents creation is template based, so instructors can use different templates to create agendas, announcements, course content, discussion forums, links, syllabus and course descriptions.

i) Different users, different learning path

The learning path tool of Claroline allows to create a complete sequence of learning steps or activities for learners to follow. The main idea behind this tool is that you can suggest (or force) your students to read documents, pass exercises or follow any learning activity that you created yourself in a specific sequence, namely a learning path. This path is composed by modules (at least one). Modules are single learning objects, and they can be of three types: an *exercise* (made with the existing exercise tool of Claroline), a *document* (in this case, the document refers to a document or a link present in the document tool of Claroline. This can be anything that you made yourself and that is browseable and uploadable as files with document tool.) or a SCORM content imported module (those modules are the result of importing SCORM conformant content packages, their contents depend on what the package provider has put in it. Exercises, documents, whatever that is browseable and importable with the SCORM standard so that Claroline can use it properly.) The learning path tool is available in the list of available tools of a course in the home page.

j) Time for everything: calendar

Instructors can post events and announcements in the online course calendar.

k) Hidden/in preparation material

(to be completed)

l) Synchronous communication: chat, video conferencing, etc

Claroline provides a basic chat tool: the system can archive logs for all chat rooms.
There are not video services or a shared whiteboard.

m) Asynchronous communication: forum, topic forum, newsgroup

Claroline provides a discussion forum tool: for each forum is possible to see the last post date, the number of discussions and the number of post. It's possible to quote a post to answer to it or create a simple answer.
Claroline provides a file exchange tool, too: students can upload files to a shared course folder, and download files of other students of the same course.
Claroline doesn't allow to maintain online journals or notes.

n) Other

II. ROLES

In Claroline access rights are managed according to users roles. There are four predefined roles: administrator, instructor, user and guest. It's possible to give additional rights to specific users/students.

a) Learners

In Claroline students have a public folder for displaying their work in every course in which they are enrolled, and have the possibility to visit and download files from others students folder.

b) Authors

In Claroline, only course managers are allowed to edit, disable or delete features on the course pages. The creator of the course page can give course manager privileges to any user who has enrolled in the course page: for example to additional lecturers, tutors or students with administrative responsibilities.

c) Tutors

With Claroline instructors can add students to a course or can batch add students to a course using a delimited text file, and students can self-register, too.
Instructors can get reports showing the number of students accesses to a course, within the aggregated group accessed course content.

d) Administrators

(to be completed)

e) System managers

Claroline doesn't define specifically the system manager role, but his/her duties can be carried out by an administrator user.

f) Other

In Claroline it is possible to define a guest user who can enter the course, but doesn't have any other access right.

III. RELATIONSHIP WITH STANDARD

The system is SCORM 1.2 compliant, but only for packaging.

IV. USABILITY AND LEARNABILITY

a) Usability test: existence and results

In October 2004, the OSSITE (Open-Source Software for Education in Europe) evaluated Claroline and, about usability, proclaimed that, for content production, the system is really very easy to use; moreover the system is easy to navigate, but is customizable only at platform level, and the access rights module is not very flexible, but is easy to understand and can cover almost all cases.

b) Learnability test

No documentation has been found about Claroline learnability tests.

c) Accessibility

Instructors can access context sensitive help and several instructor support forums through the development community website.

d) Multi-language support

The Claroline current version is available in 20 languages, from all five continents: Arab, Catalan, Croatian, Chinese, Dutch, English, Finnish, French, German, Galician, Greek, Hungarian, Italian, Japanese, Polish, Portuguese, Spanish, Swedish, Thaï and Turkish.

V. PEDAGOGICAL METHODS

Claroline software philosophy is explained in the Catholic University of Louvain web site (www.icampus.ucl.be/CLARO01/document/philosophy/claroline_philosophie.html). About its pedagogical approach, Thomas De Praetere, the creator of Claroline, says: "(...) The stake (note: to create a new solution to supplant other existing LMS) is at the same time technological, economical, political and pedagogical. (...) Pedagogical because any e-learning platform convey, explicitly or implicitly, a certain model of training. The oriented contents platforms suggest a training centered on the knowledge acquirement, those which privilege to use interaction tolls support the development of more collaborative courses scenarios, those which offer many authoring tools to students themselves facilitate the implementation of training by the project it selves. Claroline is at a crossroad and does not avoid suggesting a method of training. Nevertheless, its starting philosophy is minimalist: the software must envisage empty boxes and make it possible to structure exchanges and contents in multiple ways. For example, instead of guiding the teacher in the classification of his contents, a tool simply stores it and awaits for a teacher's proposal about organization. That can lead to a certain absence of homogeneity from one course to another, but, on the other hand, facilitate the use of tool by the teacher, which constitutes one of the principal difficulties of the e-learning. To reconcile teacher's freedom with the requirements of a formation of quality, Claroline returns to the interaction of teachers between them, or with education counsellors, rather than to consider naively that the tool could itself be used as a teaching guide. Axiomatically and pedagogically as neutral as possible, Claroline allures much teachers and should make it possible for organization to guide those towards a better using technologies teaching. Teachers have indeed many reasons to refuse to use the NTIC, and organizations do not have an other strategic choice to only come to a compromise with this parameter. (...)".
As for other LMS systems, Claroline components allow the free (hand-made) construction of courses following different pedagogical approaches depending on the Teacher preferences. The tools provided (forums, assessment) allows the Teacher to follow Socio-constructivist approaches. Claroline has no specific automatic support for pedagogical methodologies, e.g. to course personalization.

VI. INSTALLATION

Claroline provides a good and detailed documentation, to be downloaded from the web site, where users can find all the system and MySql requirements, specifically about file system right, and a step by step description of installation procedure.

VII. MAINTAINABILITY

The software was initially started by the Catholic University of Louvain (Belgium) and released under Open Source license (GPL). A broad community of developers and users around the world have since contributed to its development. Claroline has been developed by an international network of teachers and developers scattered around the world. It recycles entire programs or pieces of code found in the vast library of the GPL Open Source internet mediated community. Since the beginning of 2004, the Cerdecam (research center of ECAM, Engineering higher school in Belgium) is collaborating on the development of Claroline.
The development pathway indicates an increasing commitment to open standards.

VIII. REQUIRED HW or SW

There aren't very strict requirements about hardware. Regarding software, these products are required: Apache, MySql (version 2.23 or more), PHP (version 4 or more) and some PHP module (mysql, zlib, preg). The software supports any browser type or version.

IX. LICENSING: OPEN SOURCE or PROPRIETARY

The software is free and distributed under the terms of the GNU General Public License.

X. TOTAL COST OF OWNERSHIP

Downloading and using Claroline is completely free of charge, and a complete system can be installed in less than a day.

From the instructor point of view, it is enough to be familiar with your own preferred browser: you will be asked to simply interact with different forms (text, buttons, upload objects). Many instructors become familiar with Claroline in two or three hours without any dedicated technical training, so they can concentrate on content and good educational scenario.

XI. USAGE

The Claroline project was initiated in 2000 at the Catholic University of Louvain (Belgium) by Thomas De Praetere and was financially supported by the Louvain Foundation.

Catholic University of Louvain encouraged the Institut de Pédagogie universitaire et des Multimédias (Institute for University Education and Multimedia) to develop and distribute this software. It became available as open source in January 2002. The development site for the project is at http://www.icampus.ucl.ac.be/ CLARO01/

Some 6.000 people downloaded the last Claroline version since its release, in April 2005. Total download count of all the different Claroline versions reaches 78.000. In the worldwide page of the Claroline web site there's a detailed list that collects all the 424 organisations from 61 countries using or testing Claroline (or one of its clones) that Claroline knows about.

Claroline has been designed for a 20.000 students university. Today, more than 400 organisations use this tool in 30 languages, 60 countries and help to improve it day after day.

XII. SUPPORT QUALITY

Claroline code is written in quite clearly and rules are, almost always, respected, but there are not API and is not implemented any plug-in to features extension. Moreover, at this moment, is not available a development documentation.

Users requests for new features are managed through a forum, and is possible to enter to repository (CVS) to see the version in progress at that moment.

However, the system has a very clear student manual, in English, and the translation community is working to produce the documentation in other language (Dutch, French, German, Italian, Portuguese and Spanish).

4. LMS: Microsoft Learning Gateway / Microsoft Class Server

The Microsoft Learning Gateway is a technological framework that makes available the tools and systems needed for integrating and sharing several knowledge sources across a wide set of participants with different roles. The framework is based on the Microsoft Windows Server technology and on the Microsoft SharePoint technological solution, that allows for collaborative editing, information exchange and document workflow across organizations.

This technological framework can be used to build custom solutions that handle the roles and needs of a Learning Institution, through the help of the Microsoft Learning Partners, i.e. firms specialized in the integration, customization and construction of integrated systems within the framework for Learning Institutions.

The framework is very general, solving several system-related problems, like: data replication and integration, role-depending access to information sources, single sign-on to the system, integration of data from many sources in the same document/web page, construction of workflows, publication of documents and resources on the web, system-wide search, data reporting.

The Microsoft Learning Gateway system can be viewed more as a set of integrated technologies than as a Learning Management System, for this reason, while we are aware of its technological strength, we have decided not to describe its features here, but to examine the specific component used as a Learning Management System within the framework: the **Microsoft Class Server**.

LMS: Microsoft Class Server 4.0

Microsoft Class Server is an integrated Microsoft Learning Gateway feature. The Microsoft Learning Gateway brings together a range of Microsoft server products to deliver a Web-based portal solution that is highly scalable, while supporting also the deployment in smaller school scenarios. Microsoft Class Server is integrated within the portal to deliver an online learning management system (LMS) enabling teachers to create lesson plans, manage classes and perform curriculum assessment electronically online and offline. On the other hand, Microsoft Windows SharePoint Portal is fully integrated with Learning Gateway to enable students, parents and teachers to collaborate and share ideas. Users can see when their friends and colleagues are online and work together using the instant messaging service while accessing shared documents and workspaces.

I. TEACHING FEATURES

a) Contents

Class Server is a learning management platform; it delivers over the Web lessons and assessments complying with state or district curriculum standards. It can be used as the most important data repository for a whole district, as it can include student information systems, data warehouse systems, and content from multiple suppliers.

System contents may be purchased separately from the system, which predominantly delivers the content to school districts on CD-ROM (which get loaded onto the server where Class Server is located). There are more than 5,000 individual resources of content available for purchase and use within Class Server from internationally-respected publishers. Microsoft itself does not publish learning resources for Class Server. Although Class Server does not come with any content "included," users can purchase more than 50 CD-ROMs of content separately. The content that is available today covers all major subjects (math, reading, science, and social studies) and grade ranges (K-5, 6-8, and 9-12). These content packages can be easily purchased and loaded into the server to be shared and used by the entire faculty. Some content being offered is similar to traditional "educational software", and there is also a significant amount of content from XSIQ that follows the Basal Curriculum. In addition, one of the partners, Scantron, offers an assessment

bank with more than 13,000 valid assessment items so that users can conduct regular diagnostic assessments of students.

Besides, within Class Server, using Microsoft Office Outlook Web Access, users can see messages, calendar, contacts, tasks, and public folders directly from the Learning Gateway portal. The interface closely matches that of Microsoft Office Outlook, so no new learning is required and students and teachers can be productive immediately.

b) Course Organization

At present no information is available about this point.

c) Verifying the learning

Class Server may be used to load assessments and quizzes, for instance, from the Scantron bank, or from other repositories.

A new feature in Class Server 4.0 is the ability to create auto-graded tests delivered over the Web, but at present no information is available about this feature.

d) Verifying the use: survey

At present no information is available about this point.

e) How much knowledge: automatic correction

To the best of our knowledge, no tools are available yet in order to automatically evaluate students' work. However, Class Server comes with tools which allow to collect and export accurately and in real-time student performances to create reports for a wide variety of student demographic categories (ethnicity, gender, special needs, and more) as required by the No Child Left Behind Act (see also section 5). Moreover, performance data can be exported out of the system into other data systems using a Software Development Kit that allows the solutions partners and IT-savvy school districts to access and collect student achievement data from Class Server.

A new feature in Class Server 4.0 is the ability to grade student work against local curriculum standards.

f) Organize a class: managing learners group

At present no information is available about this point.

g) Who see what: access rights to course and contents

At present no information is available about this point.

h) Course creation: authoring tools

At present no information is available about this point.

i) Different users, different learning path

At present no information is available about this point.

j) Time for everything: calendar

Microsoft Class Server doesn't (itself) offer calendar tools because this feature is provided by the Office Outlook Web Access, within the Learning Gateway portal.

k) Hidden/in preparation material

At present no information is available about this point.

l)   Synchronous communication: chat, video conferencing, etc

At present no information is available about this point.

m)   Asynchronous communication: forum, topic forum, newsgroup

Class Server doesn't (itself) offer specific tools to manage messages, contacts, public folders because this feature is provided by the Microsoft Office Outlook Web Access, within the Learning Gateway portal.

n)   Other

I. ROLES

Access to Microsoft Learning Gateway is role-based and controlled by a central administration according to the user's status. Users only see the information that is directly relevant to them, ensuring that the portal delivers the right information to the right person at the right time.
The system provide single sign in, so users have to sign in only once per session.
Class Server doesn't facilitate the transportation of individual student records to another server, because the classes, teachers, and assignments will all be different in the new school.
At present no information is available about users roles within the platform, so all the sub-section of this paragraph are empty.

a)   Learners

b)   Authors

c)   Tutors

d)   Administrators

e)   System managers

f)   Other

II. RELATIONSHIP WITH STANDARD

By embracing open-standards for content (IMS and SCORM) in Class Server, Microsoft delivers a less costly, more sustainable deployment and integration, as well as allowing the use of content from multiple providers. Class Server conforms to IMS content packaging and meta-data standards, as well as SCORM, including SCORM run-time support. Class Server also supports the specific application profile developed for the Curriculum Online project.
As a matter of fact, Class Server 4.0 comes with a new feature: the compliance with IMS and SCORM open standards for content, and with SIF open standards for Enterprise Application Integration.
Moreover, in Class Server, school district IT administrators (or the solution providers hired by them) can load an XML file of their own state standards directly into Class Server. A tool for curriculum directors is also available to edit these standards to be "local-district specific". Each learning resource and assignment can then be tagged with the applicable curriculum standard(s) using a series of drop-down menus.

III. USABILITY AND LEARNABILITY

a) Usability test: existence and results

No documentation is available about usability tests.

b) Learnability test

No documentation is available about learnability tests.

c) Accessibility

At present no information is available about this point.

d) Multi-language support

At present no information is available about this point.

IV. PEDAGOGICAL METHODS

Designed specifically to meet the needs of K-12 institutions, Microsoft Class Server is a learning management platform that enables school districts and teachers to track, analyze, and improve student achievement in accordance with the requirements of No Child Left Behind (NCLB). The focus of NCLB is "to ensure that all children have a fair, equitable, and significant opportunity to obtain a high-quality education, and reach, at the minimum, proficiency on challenging state academic standards and state academic assessments." In practice, this system doesn't force any kind of pedagogical approach, but simply sustains any process to deliver an improved learning method.

As for other LMS systems, Class Server components allow the free (hand-made) construction of courses following different pedagogical approaches depending on the Teacher preferences. The tools provided (forums, assessment) allows the Teacher to follow Socio-constructivist approaches. Class Server has no specific automatic support for pedagogical methodologies, e.g. to course personalization.

V. INSTALLATION

The easiest way to get Microsoft Class Server and begin to use it is to purchase it from one of the Microsoft-approved solution providers (see also section 10). However, users interested in creating their own Class Server solution can also contact an authorized local reseller (in the USA).

Most Class Server solution providers offer data integration or data import services to ensure that all school districts' data is properly loaded into Class Server at the beginning of each new semester (and also to ensure that grades are integrated or exported out of the platform). In technical terms, Class Server has bulk import/export capabilities, a Schools Interoperability Framework (SIF) agent (for interoperability with other systems that also employ SIF), and it also allows solution partners to populate or collect data in real-time by directly accessing the Application Programming Interfaces (APIs).

Moreover, Class Server can host multiple schools on the same server. The single copy of the application can handle multiple schools from the same server. Each school must have a unique database name, a unique primary virtual directory, and a unique file location. However, the SQL Server or MSDE sign-in name and password must be identical for each school.

VI. MAINTAINABILITY

Microsoft Class Server (and other Learning Gateway related products) is supported by Microsoft.

VII. REQUIRED HW or SW

Class Server has been designed to work for a wide range of sizes of school districts: whether there is one computer per classroom, extensive computer labs, mobile computer carts (where the "lab comes to the classroom"), or there is an environment where every teacher and every student has a computer. The system can handle situations where all students and all teachers "share computers," and the system can also be configured for 1:1 implementations.

In technical terms, system requirements for Class Server are: a PC with a 1.8 GHz or higher processor (Intel Xeon-compatible processor recommended), a Microsoft operating system such as Microsoft Windows 2000 Server or Advanced Server or Microsoft Windows Server 2003 family (or later), 256 MB of RAM, 100 MB of free hard disk space, a CD-ROM drive, and an Internet access. In addition, although Class Server comes bundled with a free copy of SQL Server 2000 Desktop Engine (MSDE), a sample database that handles up to 1,000 students, SQL Server is required for large deployments, because for implementations larger than pilot size, Class Server's data is stored in an associated Microsoft SQL Server database. Both Class Server and SQL Server can either be installed locally or hosted by an application service provider. Deployments larger than 2,000 students require a multi-computer solution. SQL Server needs to be purchased under "per processor licensing" for the institution to avoid having to buy SQL Server User Client Access License (CAL), which allows students to access SQL Server anywhere.

Then, system requirements for teachers are: 600 MHz or higher processor (Pentium III-compatible processor recommended), an operating system such as Microsoft Windows 98, Millennium Edition, 2000, XP (or later), 128 MB of RAM, a Super Vega (800x 600) monitor, 50 MB of free hard disk space, Microsoft Internet Explorer 6.0 or later, and an Internet or intranet access to Class Server. Class Server teachers need to do their work on Windows-based personal computers, because Microsoft grading and learning resource creation applications only work on a personal computer. There is a benefit that comes along with this requirement; if teachers wish to work from home, a special "offline capability" is provided into Class Server so that teachers do not need to tie up their phone line when using Class Server to grade papers or create lessons at home.

System requirements for students are: a Windows-compatible or Macintosh computer, Internet Explorer 5.0 (or later) or Netscape Navigator 7.1 (or later), an Internet or intranet access to Class Server.

VIII. LICENSING: OPEN SOURCE or PROPRIETARY

Microsoft Learning Gateway are proprietary programs of Microsoft.

IX. TOTAL COST OF OWNERSHIP

Microsoft Class Server is a proprietary software, so to purchase a Microsoft Class Server solution, users need to contact one of Microsoft-approved solution providers. On the Web no information is available about the cost of these services, perspective clients are invited to ask providers for an estimate.

About the cost to learn to use the system, in Class Server 4.0, was included the SharePoint interface, which is intuitive and easy to learn, however users have seen their best results when they have had some 'Best Practices' training with Class Server. Therefore, according to the system official web site, it is highly recommend a minimum of one full day of professional development training for teachers and administrators who plan to use Class Server.

X. USAGE

Microsoft Class Server has been designed specifically to address K-12 school districts and teachers. So, at date, it is helping many school districts to address the requirements of No Child Left Behind (see also section 5) and build connected learning communities.

XI. SUPPORT QUALITY

Microsoft offers special support for schools using Class Server: there is a complete Troubleshooting Guide and a newsgroup, where users can get help and share "best practices". Moreover, Microsoft has cultivated a network of Class Server Professional Development providers nationwide, each of whom provides one-day workshops to help new users. Finally, Microsoft has a Professional Development kit available that customers and evaluators can download for free.

As an additional technical support, Microsoft provides the Class Server Administration Guide, a detailed document explaining how users can configure and deploy the system, available on the Class Server CD-ROM or on the Web. Besides, Class Server 4.0 comes with a new teacher's guide, tailored specifically for teachers' needs.

5.   LMS: Desire2Learn 7.3

Desire2Learn is a Learning Platform rich in tools to create and manage course on the Web. Desire2Learn is a proprietary system and a demo version is not available, so it was not possible to verify all the information found about its features. The information in this document was obtained matching different sources.

I.   TEACHING FEATURES

a)   Contents

In each Desire2Learn course instructors can create *announcements* (to be published on the main page of the course to inform students about upcoming events, due dates, or corrections), *calendar* entries (see also section 1.j), discussions *forums* (see also section 1.m), *links* (indicating external web pages relevant to the course), *syllabus* (presenting subjects, exams, teacher's contacts and bibliography of the course), course *description*, (presenting a summary of the course contents), *FAQ* (providing a list of frequently asked questions related to the course content or policies) and *glossaries* (to be used to make key terms related to the course available to students).

b)   Course Organization

At the moment, there is no available information about this feature.

c)   Verifying the learning

The system allows students to make keep track of ideas and information they might want to use in class and share with their instructor in a personal journal: it is accessible almost anywhere and easily exportable.

Moreover, the system enables instructors to create easily, within the course content, interactive self-assessments that can be taken more than once, to provide instant feedback on question responses.

In addiction, Desire2Learn comes with a testing tool enabling instructors to create, edit and grade assessments (see also section 1.e) without any HTML knowledge or programming experience. Desire2Learn supports the following types of question: Multiple-choice, True/False, Short/Long Answer, Fill-in-the-blank, Matching, Ordering, Multi-select, Arithmetic, Significant Figures. Custom questions types can also be defined. Test questions can incorporate images, sound, video and other media types. Questions can either be built with the tool provided, or instructors can import and export questions from external test banks in the IMS QTI specification format. Instructors can create a database of questions: questions of various formats and difficulty levels can be stored in question libraries, thus permitting instructors to share questions across quizzes, to randomize the questions used, and to randomize the response options within a particular question. Regular expressions can be used to create unique variables and answers for every question. Instructors can also restrict submissions to the assignments folder by date, so that late or early submissions will not be allowed. Assessments can also be restricted by IP address. Assessments can be shared across several organizational units, and data can be aggregated during report generation. The testing tool can also provide the instructor with a gradebook of exam results and statistical analysis of the tests and the individual test items. Instructors can customize reports on each user and question data that can be exported.

d) Verifying the use: survey

Desire2Learn comes with a survey tool, thus allowing the instructors to collect instant feedback from students (even anonymously) and to monitor trends and opinions of the users. This tool provides easy distribution and results tabulation: options about restricted survey access and results release are also available.

e) How much knowledge: automatic correction

The system automatically and instantly grades quizzes and can display instructor-created feedback, explanations, hints and links to relevant course material, allowing students to receive feedback more quickly and efficiently.

f) Organize a class: managing learners group

Either instructors can assign students to groups or the system can create groups of a certain size or a set number of groups. Each group can have its own discussion forum, chat room, and group email list. Desire2Learn allows roles creation for user within a group: for example there may be a leader (who can start a discussion or respond to a question posted by the instructor, usually is one user only), facilitators (who makes sure the team's work is done in time by helping to keep the team on task), commoners (common member of the group) and summarizers (who, after each team member has had a chance to participate in the discussion, will read through the responses and post a summary for the entire class). Moreover the system allows community building by students themselves.

g) Who sees what: access rights to course and contents

Administrators can create an unlimited number of custom organizational units and roles with specific access privileges to course content and tools. Administrators can distribute the roles across multiple institutions to facilitate school board-wide or consortia-style approaches. Instructors or students may be assigned different roles in different courses and group contexts. Instructors can customize specific access permissions for each student.

h) Course creation: authoring tools

Course templates containing either layout or content can be created at any level above the specific section level. Templates include a WYSIWYG content editor. Instructors can also upload and choose content from the system-wide content library.

i) Different users, different learning path

Instructors can specify multiple paths through a course for different skill levels or job functions.

j) Time for everything: calendar

Instructors can post events and announcements in the online course calendar (or to the course homepage), or can link items on a learning objective checklist within the calendar. Students too can use a course calendar: to store private events, to view their grades on completed assignments and any other instructor feedback, and to compare their grades against the overall class performance. Students can also view their completed and pending course readings and activities.
Calendar doesn't provide an automatic event reminder.

k) Hidden/in preparation material

At the moment, there is no available information about this feature.

l) Synchronous communication: chat, video conferencing, etc

The chat tool supports private rooms and private messages, allowing users to converse with more than one other user at a time. The system creates archive logs for all chat rooms. There is a built-in instant messaging tool.
The whiteboard supports image uploading and annotation. There's no video services.

Desire2Learn provides another synchronous communication tool, called pager, that allows users to communicate one on one in real time, similar to instant messaging.

m) Asynchronous communication: forum, topic forum, newsgroup

The system comes with a tool to view, add, modify and remove discussion topics and forums. Within a specific course offering, administrators can edit or delete threads, and can control the access privileges of users, including the right to edit and/ or delete one's own messages. Instructors can enable or disable anonymous postings, and an entire discussion can be saved or printed for off-line reading. Discussions can be viewed by date, by thread, by title or by author, and users can access keyword searches and sort messages by date, topic, and sender. Discussions can be shared across courses, departments, or any institutional unit. Instructors may create separate discussion environments for small groups and can associate a discussion with any course content. Posts can include attachments or URL and can be either plain text or html.

The system provides a file exchange tool: students have a private folder into which they can upload and download files. If a course is divided into groups, students can upload files to a shared group folder and administrators can set folder permissions and quota tracking on their folders. Finally, students can email individual students, instructors or groups by using a searchable address book. This internal email system supports folders, searching and attachments; besides, it allows students either to forward their mail to an external address, or to read mail in any IMAP mailbox.

Moreover, a particular feature allows Instructors to publish course content on a CD-ROM that can be linked to dynamically from within the online course or viewed offline. Students can compile and download the content for an entire course into a format that can be printed or stored locally.

n) Other

The system can support multiple institutions, departments, schools or other organizational divisions on an individual server setup. Each unit can be separately branded.

II. ROLES

In Desire2Learn administrators can create an unlimited number of customized user roles with specific access rights or privileges both for instructors and students, so is possible to have a large gamma range of users type. Here only major roles are analysed.

a) Learners

Students can self-register to courses. They can search all course content, discussion threads, calendar postings in their course or organization, also by using filters, and can bookmark any content material in a course.

Students can create a personal home page, where is possible to insert artwork or pictures.

The system allows students to create their own portfolios; they can make notes in a personal work area in preformatted text or HTML-style with tags and can perform spell checking.

Students from different courses can interact in system-wide chat rooms or discussion forums. Students can create online clubs, interest, and study groups at the system level. Students can be assigned access privileges for system-wide text-chat rooms.

b) Authors

Instructors can manage courses: they can selectively release course materials and tools, based on previous course activity or specific start and end dates. The system can synchronize course dates defined by the institutional calendar. The system provides a central content repository where course content files can be stored and accessed by other instructors. The repository can be system-wide or for individual organizational units. Instructors can designate their files to be private or publicly accessible, and can create links to content files in the central content repository so that changes made to the linked content are automatically displayed in their courses. Instructors can describe course content using metadata.

Instructors can mark assignments online and provide feedback on all assignments, both editing grades manually or automatically. When an instructor adds an assignment to the course, the software automatically

adds it to the gradebook. Instructors can use the gradebook for basic statistical analysis and final grade calculation. Instructors may export the gradebook in PDF format. Instructors can view reports of student session tracking, recording both login/logout dates and times and discussion participation They can also view both individual and aggregate student performance data about assessments, viz. number of attempts and time per attempt. Instructors can share this tracking information with students.

c)   Tutors

In Desire2Learn this simply an instructor with only teaching rights, and his/her range of work is chosen by the administrator.

d)   Administrators

Administrators can protect access to individual courses with a username and password. The system has a password reminder option and users can maintain their own passwords. Administrators can set password length restrictions and require password changes after the initial logon and after a set period of time.
Administrators can batch add students to a course using a delimited text file.
The customizable features of Desire2Learn provide administrators with straightforward means of creating conferences for the members of a class or its sub-groups.

e)   System managers

Because the product provider installs and offers a hosted system for the system (see also section 6), this role is not relevant, and can be considered equal to another administrator.

f)   Other

III. RELATIONSHIP WITH STANDARD

Desire2Learn's LMS was the first academically focused LMS to be certified for SCORM 1.2 RTE3 and SCORM 2004 by the ADL co-Lab, and supports IMS Enterprise 1.1, IMS QTI 1.2, and IMS Content Packaging Specification.
Desire2Learn is XML standard-based, and uses a learning-object repository approach that allows instructors to store content (i.e., pages, graphics and test-item banks, etc.). These objects can be shared across any number of courses or programs.

IV. USABILITY AND LEARNABILITY

a)   Usability test: existence and results

In 2004, the University of Iowa made a research to adopt a single, centrally-supported course management system with full enterprise integration. Following a thorough evaluation process with broad campus participation, the University of Iowa has selected Desire2Learn as the University's enterprise learning management system. During this research usability was analysed, too, and results are available in the "E-Learning   Project   at   The   University   of   Iowa"   web   page (http://www.uiowa.edu/~provost/elearning/cms_selection/index.shtml).
Also the IRRODL (International Review of Research in Open and Distance Learning), in April 2004, evaluated Desire2Learn and this is its opinion about usability: "The product has a high degree of usability. Its course structure and appearance are easy to manipulate, and administrative control over the usage process is broad and easily specified. Owing to its high degree of flexibility and customizability, the product can take many forms" (http://www.irrodl.org/content/v5.1/technote_xxx.html).

b)   Learnability test

No documentation has been found about Desire2Learn learnability test.

c) Accessibility

To comply with Section 508 of the US Rehabilitation Act, the software implements the following features: alt tags on all system images, appropriately titled framesets that describe the functionality of the frames layout, data tables optimized for use with screen readers, content available without colour, and content readable without style sheets.

The system includes an online student guide, and students can access context sensitive help for each of the major tools, where help items are further customizable by the institution.

Instructors can access an online instructor manual, context sensitive help, take a free online course, and subscribe to an instructor mailing list. The product provider can setup an instructor group so instructors in the organization can access faculty resources and discussions.

d) Multi-language support

Desire2Learn has a growing list of users community, distributed in Europe, South America and North America, and its main language is English, but it also produces courses for French speaking universities, e.g. in Canada.

V. PEDAGOGICAL METHODS

In the Desire2Learn web site the producers claim: "Unlike many of our competitors, we believe an eLearning platform should not dictate your approach to teaching and learning. It should be tailored to your needs, desired look and feel, philosophy, vision, brand, and pedagogical approach." So it's not possible to define for it a specific pedagogical approach.

As for other LMS systems, Desire2Learn components allow the free (hand-made) construction of courses following different pedagogical approaches depending on the Teacher preferences. The tools provided (forums etc.) allows the Teacher to follow Socio-constructivist approaches. Desire2Learn has no specific automatic support for pedagogical methodologies, e.g. to course personalization (although the teacher can define several alternative paths through the course material).

VI. INSTALLATION

The product provider installs the system and provides new releases and patches over the network. Moreover the product provider offers a hosted system that includes 24x7x365 monitoring, redundant hosting platforms, intrusion detection, nightly backups, options for geographical disaster recovery, and service level agreements on a network of high-performance, fault-tolerant servers with fail-over capability with redundant Tier 1 network connections.

VII. MAINTAINABILITY

According to what it claims in its web site, Desire2Learn has "a very strong research and development team, and it is developing several next-generation technologies to give clients the ability to communicate more effectively. These include wireless, hand-held devices, and others. These other components, which may become integrated into the learning process, only serve to expand and enhance its current product offerings.

Working closely with its clients and R&D partners, it also has built several other key components for system's infrastructure: Registration System, Certificate Generation and Management System, Basic Administrative and Student Information Systems (typically used as a component add-on to existing solutions), Performance Management and Analysis, Wireless technology, and much more".

VIII. REQUIRED HW or SW

The software is available for Windows 2000 Server, Advanced Server and Datacenter Server, and Windows 2003 Standard, Enterprise and Datacenter Editions; it supports Internet Explorer 4+, Netscape 4+, Safari 1+ and Mozilla 1+. (see also section 10)

IX. LICENSING: OPEN SOURCE or PROPRIETARY

Desire2Learn Learning Platform is a proprietary software developed by the company founded in 1999, that employees more than 60 people and is privately held.

X. TOTAL COST OF OWNERSHIP

The Desire2Learn software has a start-up cost based on the number of features, the level of integration with existing systems, and the support needs of the institution. The product has the capability to be used as a course-by-course delivery tool, or scaled up to a larger entity such as a province-wide license. Annual licensing costs are based on a flat fee or on a per-student basis. Users are not required to download any software. The company's servers contain all the software to support their LMS platform with the exception of Java, which is used under the license. This minimal use of third-party software enables the company to give flexible responses to clients' requirements. The vendor provides access to a support person per 5000 students covered by the license.

XI. USAGE

A lot of public and private companies and universities chose this product to offer to their users a good e-learning environment. For example, in academic world, University of Oklahoma, University of Iowa and Ohio State University selected Desire2Learn products for their users. In particular at the end of 2004, the Ohio State University chos to replace the existing WebCT Course Management System with a new implementation of Desire2Learn (after a comparison with Blackboard and CyberLearning Lab's Angel): this implementation replaced all current instances of WebCT, created by the university itself!

XII. SUPPORT QUALITY

The company offers consulting services for implementation planning and advanced technical services including an initial technical assessment and evaluation of issues such as authentication, load balancing, and migration and upgrade planning. The product provider also offers full services for instructional design and custom courseware creation.

6. LMS: DoceboLMS 2.0.4

(add description)

I. TEACHING FEATURES

a) Contents

Learning object can be defined as they are all the object that the students can use and the DoceboLMS platform supports. These include: *FAQ* (a list of questions that users can navigate by question or by keywords), *glossary* (a list of words that can be selected by letters or trough an alphabetic list), *html page* (can be also created online, with formatted text, links and images), *uploaded file* (documents cannot be created within the platform, but is possible to import many types of files, for example images, PowerPoint presentations, movies, sounds), *link*, *poll* (see section 1.c), SCORM both for packaging or for runtime (a group of some learning objects, with an internal organization and sequence, based on international SCORM standard) and *test* (see section 1.c).

b) Course Organization

DoceboLMS allows to create courses categories for a better courses cataloguing, the use of filters for a quicker search and a clearer visualization of the LMS. In practice, while creating each course it is possible to set several features (name, status, difficulty, language, estimated finish time, course end option, students not allowed to access the course, etc), and its category is one of these.

c) Verifying the learning

DoceboLMS provides two tools to verify learners knowledge: polls and tests.

A poll can show opinions held by students about a certain topic, and can generate a report. While creating a poll, there are five fields to be filled in: question text, category (to be chosen among commons categories defined by administrator), kind of question (single answer, multiple answer, open answer), other options (whether question is mandatory, or others).

Poll and test interfaces are very similar, but there are some differences: tests have a different policy, so, rules must be assigned in the moment of the test creation. Also the generated reports are different: for example, it is possible to see the test results of one student only. Within a test the possible variables are: *kind of question* (single or multiple), *kind of question marks* (point or percentage), *total result after the test ordered by points, total result after the test ordered by right answers*. While creating a test there are six fields to be filled in: *title, description, question type* (multiple answer using checkbox or single answer using radio button), *evaluation type* (evaluate in points or evaluate in percentage), *toggle test solution* (if on, shows test solution after it has been finished) and *toggle test results* (if on, shows results after it has been finished).

After test creation, it's time to insert questions. If the questions will be "points questions" it is necessary to specify if and how many points will be added to the right/fail answers. It's possible to assign an unlimited number of answers, and also to comment (feedback) for the right/fail result.

d) Verifying the use: survey

Using DoceboLMS, instructors can create a survey to view reporting pages that show tests results and show how many students have already attempted it. Survey management is quite similar to test management, and has the same features.

e) How much knowledge: automatic correction

While creating questions for a test, teachers can insert a comment and a score for each possible answer. The comment is a text feedback. Answers scores are given by points, which add up to the test score if the answer is right or are subtracted if it's wrong. At the end of an attempt to the test, the total score is shown as attempt's mark.

DoceboLMS provides a tool to report users activities in the platform. The layout of the statistics is the same of other reports, with the additional option of a two layers analysis, on lessons statistics and on users statistics. Lessons statistics can be organized in three ways: general statistics, viewed pages, and mostly used functions. Polls results also come with this layout (for teachers). Users statistics can be viewed for every single user and organized for every single log to the platform, and/or for every activity taking place after log in. For example, teachers can view, for every single user, how many learning objects are being attempted and how many haven't been completed yet, and also tests results and detailed answers. Moreover, they can give their own votes, modify and remove votes already given and make votes visible/invisible to students.

f)   Organizing a class: managing learners group

Users can be organized in groups. Using a simple icon interface is possible to add and remove users in groups, create/delete groups and modify groups features (for example name and description).
Teachers can aggregate users in several groups (and users can be in more than one group). This allows teachers to use more efficiently the advanced functions of the learning manager: in forums (allowing different privileges on forum discussions to different groups), in messages (allowing group management of receivers), in advice (allowing to show advices collectively to group members), in learning objects (allowing to set some learning objects to be shown only to some groups), in statistics (allowing to create groups filters for statistics data) and in projects.
In fact, DoceboLMS provides a specific tool to involve groups. The project management system allows groups to work collectively using a common interface; by monitoring activity progress, publishing news, "to do" lists and new files to work on, communicating using the internal messages system, and so on. In order to create a project it is mandatory to create at least one group of users.

g)   Who sees what: access rights to course and contents

While creating a course, one is able also to set its status, by which access rights to the course itself can be limited: students can access only active courses, while teachers can access courses even if they are deactivated or in activation. Besides, it is also possible to define a policy about student access denials. For example, it is possible to forbid users having a particular status to enter (i.e., a teacher can decide if an user can enter the course after having finished it, or not). Furthermore, DoceboLMS allows a fine tuning of access rights: it is possible to assign to each student a level, which will determine the user's rights in that course.

h)   Course creation: authoring tools

Every object can be created using a page form containing all the options to be filled in. Once learning objects are created, it is also possible to open them sequentially by setting up "execution rules". For example, if a learning object is set as a prerequisite, the object must be seen before accessing any other object. It is also possible to choose how many times the object can be viewed or executed, which groups or users can access it, to show or hide it (see also section 1.k), and finally is possible to set it to be a "course end marker", so when a student will execute it, the course will be marked as "complete".

i)   Different users, different learning path

It's possible to (manually) design different learning paths by designing sets of learning components available to different groups of learners or (partially automatically) by using the execution rules assigned to learning objects.

j)   Time for everything: calendar

DoceboLMS comes with an agenda for planning activities.

k)   Hidden/in preparation material

While creating a course, the administrator can choose whether to show it or not in the generic list of its category.
In the content management area, teachers have a private area where they store his/her own objects, create new ones or import other teachers' objects. This is an extra-course area, so it will be viewed by the teacher independently from the course s/he's working on. Also, it is possible to organize the objects in folders/subfolders for a better usability.

Moreover, there is a public repository area, where teachers can share learning objects with other teachers and from which everyone can import learning objects to his private area or directly in the course (note that in DoceboLMS by importing an object one creates an independent copy of it, which will be a brand new object).

l) Synchronous communication: chat, video conferencing, etc

The chat system created for DoceboLMS is learning-based: beside the normal chat functions, it comes with specific functions in order to optimize the time spent in the synchronous learning activity. Chat functions are: create rooms, public and private (private can be on request, where users "knock" and the room admin decides what to do, or password protected, where users must use a password to access the room), use a "Can I talk?" system (users must have permission to speak in the chat), send files (is possible to share files into the chat), save chat (a static readable offline version) and create a personal chat between two users. When a user enters the chat, s/he can see a room list. Every course has its chat room. There can be also publics rooms or private rooms created by users. Policies for room creation, moderation issues, file upload, etc… are defined by the administrator.
Neither video services nor shared whiteboard are provided by DoceboLMS.

m) Asynchronous communication: forum, topic forum, newsgroup

For some topics, forums can be a very useful means of discussion. Forums are organized in categories (created by the teacher, who can also set access privileges for them, e.g. denying access to some groups if desired) and threads (discussions posted by the teacher, by students, etc). While using the forum it is possible to upload a file into a thread, to edit replies and also to mark every student reply and generate reports about them. Forum can be also moderated, in two ways: by blocking a thread (it is still possible to read the discussion but not to reply), or by closing a thread (the thread isn't viewable anymore).
The internal message system is used to share files and messages in every single course, so, every student will be able to post new messages for every course s/he joins. In the message area it is possible to send messages or attach files to a single user, a selection of users or groups of users.

n) Other

I. ROLES

In DoceboLMS are available four different pre-defined roles (admin, instructor, assistant, student) and it is not possible to define new roles.
Nevertheless, inside the learning manager there are different user's level and what every level can do can be decided at single course level, so we don't have a rigid model where a student can do only certain actions and not others. Available levels are: God Admin, Admin, Teacher, Tutor, Mentor, Student, Guest user and Ghost user. User levels are assigned in two different moments: the pre-defined level is assigned when the user is created (or edited) and then the level can change when administrators enrol the user in some course (imagine an admin that attempt a course as a student ...)

a) Learners

In DoceboLMS students can enrol themselves or administrators can create and enrol new students.

b) Authors

While using DoceboLMS, instructors can track students behaviour (about access time and test grading), administer course content and manage all the course features, according to their access rights set by the administrator.

c) Tutors

In DoceboLMS tutors don't exist as a specific role, but it is possible to assign pre-defined Mentor or Tutor level to an instructor or a learner, both during the process of course creation and while the same is in progress.

d)   Administrators

While using DoceboLMS, system administrators' task is to allow teachers, students and other intermediate levels to be free to concentrate on managing learning objects (for teachers) or on using them (for students). Main administrator activities are: user management (creation, editing, delete), course creation and activation, user course subscription, reporting activity. DoceboLMS comes with lots of function to make easy, powerful and flexible the administration activity. Two kinds of user can administer the LMS: the God Admin (they can be more than one, actually), who can do everything, and the Admin (also more than one), who can have their activities limited in functions or in user's groups management.

As we said, God Admin has full control of the administration area, without any limitation, either to functions or to users number/group. Admin, instead, can have some limitations: for example, Admin may see only some menu links, and not all of them (to every menu link a function is associated) or s/he may not perform all the activities in the same function. (For example: admin can see the list of users but s/he can't delete them). About users limitations, (see also section 1.f): an Admin will see and manage users in her/his own group, but not in the other groups (ex. Group A admin users list will be different by the users list that admin of group B will see).

e)   System managers

In DoceboLMS this role is not provided, and its tasks are carried out by administrators or instructors.

f)   Other

DoceboLMS allows the existence of a ghost user that can enter the course, but doesn't have any other access rights.

II. RELATIONSHIP WITH STANDARD

The system supports SCORM 1.2 standard (a very good guide is available to create SCORM packages (see also section 12).

III. USABILITY AND LEARNABILITY

a)   Usability test: existence and results

No documentation has been found about DoceboLMS usability tests.

b)   Learnability test

No documentation has been found about learnability tests done on DoceboLMS.

c)   Accessibility

No documentation has been found about accessibility tests done on DoceboLMS.

d)   Multi-language support

Several localizations are available: French, German, English, Italian, Russian, Spanish, Portuguese (from Brazil and from Portugal), Chinese (standard simplified) and Finnish.

IV. PEDAGOGICAL METHODS

The system can manage different learning models, from collaborative learning to self learning, and provides some functions particularly oriented to collaborative-learning.

As for other LMS systems, DoceboLMS components (chat, forum, quiz etc…) allow the free (hand-made) construction of courses following different pedagogical approaches depending on the Teacher preferences. DoceboLMS has no specific automatic support for pedagogical methodologies, e.g. to course personalization

(even if the Teacher could define several different learning paths in his Course or define different activation rules for different groups of students). The tools provided allows the Teacher to follow Socio-constructivist approaches.

V. INSTALLATION

In order to perform installation, it is possible to download an installation package (a zip file), which includes all the fixes from the previous versions, from the download section in the DoceboLMS web site. In the same page all the patches (zip files) to update DoceboLMS are also available, together with an installer manual in pdf format containing the suggested configuration for a satisfactory use of the system (about operating system, components version, server and file upload configuration). The manual explains also how to set up parameters in the configuration file to allow ftp operations. The manual presents a detailed case of installation on a Windows system, with lots of screenshots.

VI. MAINTAINABILITY

At the beginning, the system was not developed using an object oriented approach and plug-in concept wasn't introduced, nevertheless the system has a modular structure.
Recently, contact with system developers is available, either to suggest new features, or to obtain assistance, or to have product support through the forum in the web site. Also, hosting services are being offered and the team is fairly efficient: it is possible to have an answer in a short time.
The project is now developed by programmers located all over the world and the original team is coordinating more than 20 developers.

VII. REQUIRED HW or SW

There are no special requirements about hardware for a new installation and it is possible to have a usable system in a day. The software requirements include an environment with Apache, PHP (4.1.1 or successive versions) and MySql software.

VIII. LICENSING: OPEN SOURCE or PROPRIETARY

DoceboLMS is Open Source software, distributed under the terms of he GNU General Public License.

IX. TOTAL COST OF OWNERSHIP

The whole system is very easy to use for every role, the time to learn to use the platform being quite short. Sometimes, it could be worthwhile to spend some time upgrading the operative system or of one of the applications used, and, if necessary, to reboot periodically.

X. USAGE

At this moment installations for every kind of users have been deployed, both private and public, and it is possible to estimate that there are thousands of users in the whole world.
The platform is installed in various places, the most important being: 15.000 users in 10 Italian hospital network, 2.500 users in one of the most important Italian companies, 2.000 students in a Bergamo (Italy) High School, 500 Students in a "Science of education" Italian university, more than 120.000 teachers are using the DoceboLMS SCORM module in a big e-learning project in the world.
DoceboLMS doesn't manage only "University based teaching", lots of successful case histories are reported about, for example, installations in big companies that use the SCORM functions (especially the reporting

area, useful to be used as report for the top managers), or in governmental training centres, or in many Hospitals which are experimenting DoceboLMS. Also, in schools lot of teachers are now DoceboLMS addicted, in many Universities DoceboLMS obtained impressive results, in particular in exam simulation and preparation.

XI. SUPPORT QUALITY

On the DoceboLMS web site documentation is available to help teachers, administrators and installers. Beside a documentation is available to explain SCORM use and relevance in teaching and learning processes. All these are simple pdf files, and download is allowed for offline reading.

On the DoceboLMS web site free courses are available on DoceboLMS usage and installation. All these courses are in "self subscription mode" so users can simply auto-subscribe and attend whichever course is useful for them. Finally, for any user still having doubts about how to use the e-learning platform, an "How to start" movie is available.

In the web site different forums are available to help different kind of users: elementary schools users, middle and high schools users, university users, business professionals and health professionals. In the same page another forum is available, where DoceboLMS usage and obtained results are discussed.

After 3 years of growing business, Docebo S.R.L. is now a Company selling consulting services related to e-learning and content management to companies, government and universities.

Every consulting service is made by consultants all over the world. The Company is young but has many customers as big as major private owned and governmental companies.

Docebo is an international network and can support companies located worldwide: from Portugal to Hong Kong, from Germany to Brazil.

Finally, the system received some awards: for example, the UNESCO included the DoceboLMS Project in its "Trusted" project list, and the most important Italian economy Newspaper "Il Sole 24 Ore" included it in its list of the 10 most innovative open source projects in Italy.

7. LMS: it's:learning 3.0

It's:learning is produced by it:solutions, one of the leading e-learning companies in Scandinavia. It's:learning is a tool for the production, organisation, maintenance and reuse of learning resources. Users can make use of internal or external tools for content production. It's:learning handles several types of digital content, both self produced as well as those imported from other sources.

I. TEACHING FEATURES

II. Contents

By making use of the "Add" feature in It's:learning, users are provided access to tools/modules enabling them to create their own learning objects and then organise them within a subject. It's:learning provides users a built-in editor for creation of web pages/learning material, including a formula editor. Course content can be organised in different ways: by topic, chronologically or by function.

In It's:learning administrators can restrict the types of elements that can be created in a folder. For example, a grammar school may decide that pupils' folders are to contain only files and pictures and not conferences and discussions.

When instructors establish a new folder, they see a field, description, where they can enter a text that will be shown each time the folder is opened. If they intend to give others permission to add elements to the folder, they can decide which elements to be allowed among: discussion, process-oriented document, picture, folder, conference, note, link, file, survey, test, assignment, explaining sequence and lesson. When a user wants to add a new element to the folder, his options are limited to the types of elements administrator has allowed. If users can or not use HTML depends on whether the administrator allows them to. If users are allowed to use it, they will find that only certain tags work. This is a safety feature to prevent people from abusing the board by using tags which may destroy the layout or cause other problems. If HTML is enabled users can disable it on a per post basis from the posting form.

It's:learning has some form of built-in communication tools. It's:learning can be easily set up so that all or some of these tools are made available to users of the system. The system provides an arena where members work together, both synchronously and asynchronously. Some of the available tools for communication (see also section 1.l and 1.m) and cooperation are: internal messaging system, e-mail, chat, SMS notifications, discussion forums, process oriented writing tools, bulletin boards/news service on group level, and calendar/individual agenda.

a) Course Organization

At the moment, there is no available information about this feature.

b) Verifying the learning

Instructors have available different tools to verify student's level of knowledge, such as test and assignment. The test generating module allows instructors to create many types of tests for automatic correction, such as yes/no answers, multiple choice, and find pairs questions tests. Moreover, this tool allows to insert in tests images, formatted text, links to external resources, etc.

The function DirectEdit makes it possible for the teacher to write comments directly in the submitted assignment files. This function can be compared to the "red pencil" used when the teacher corrects assignments written on paper. By using a text editor, however, it is possible to enter longer remarks than those allowed by the limited space available in traditional methods.

c) Verifying the use: survey

It's:learning provide a module for production of surveys that can be inserted within a course to ask users their opinions about course material, teacher's approach or other arguments connected with the course.

Surveys production and evaluation can be anonymous or not anonymous, to be distributed internally in It's:learning, sent via e-mail or surveys can be presented on an external home page.

d)   How much knowledge: automatic correction

In It's:learning, the test module allows instructors to set up parameters to enable the automatic correction for some question types: yes/no answers, multiple choice, and find pairs.

e)   Organize a class: managing learners group

Using It's:learning in topics and projects it is possible to establish and manage groups directly within courses and projects. One person may participate in several groups, and the groups may consist of individual persons or sub-groups. When this facility is combined with access control for groups and individuals, it provides new possibilities for flexible teaching programs. In fact, when the groups have been created, instructor can decide what each group will have access to. For instance, this feature allows to: offer students different levels of teaching within the same topic, teaching adjusted to each student (see also section 1.i), give assignments or tasks to one group of students at a time, in order that the correction job is spaced over time.

To create a new group instructors have to enter the group's title and the persons who belong to the group. These persons must already have started the topic or be participants to the project. Not all groups are open access, some are closed and some may even have hidden memberships. If there is an open group, users that want to become group members can request to join it, and then the group moderator will need to approve the request.

Finally, creating a user group it's possible to define users role and the respective functions they have access to in the topics. It's:learning is delivered with the following standard profiles: administrator, teacher, student and guest. An administrator has full rights on all group features and all default permissions in the root folder. A teacher has full rights on group features, but he cannot edit synchronized courses features. This is the default profile chosen for persons creating a course, and a teacher, as the administrator, has all default permissions in the root folder. A student has no rights on group features, but he can see report on group members. This is the default profile chosen for membership, and a student has only participate and read permissions in the root folder. A guest has no rights on group features and has only read permission in the root folder.

f)   Who sees what: access rights to course and contents

It's possible to control who has access to each of the elements, the entire course or to structures. These rights can be given to individuals or to groups. The system provides differentiated keys for various types of access to groups and individuals: full control, edit, write, evaluate, participate or read. A full control permission allows to edit the folder permissions themselves; an edit permission allows to edit the folder, as well as permission to change the element sequence; a write permission allows to add elements to the folder and to copy and move elements to and from the folder; a read permission allows to see the folder contents. Then an evaluate or participate permission is not used by the folder directly: elements added to this folder will inherit the permission.

For elements to which a user himself has full access rights, he can change the access control. Elements added to a folder will use the same permissions rules, but adapted to the case. For example, about a discussion, full control permission allows to edit discussion permissions, edit permission allows to edit the discussion properties as well as to delete posts and threads, evaluate permission allows to view reports, participate permission allows to write posts and start new threads, and read permission allows to read posts.

When instructors have created groups (see also section 1.f), they can give them different rights of access to elements in the learning platform. In a hierarchy of elements, elements on lower levels will inherit the settings from the hierarchy above. For example: if instructor has defined that students are allowed to add content in a topic, the students will have the same right in a topic sub-folder, unless instructor specifies other rights for the subfolder.

When instructor has created a new element, for example a folder, s/he will be presented with some options. If s/he wishes to allocate other access rights than what is defined for the folder in the hierarchy above, can choose to set permissions in that folder. The screen will show two tables: the table to the left shows persons or groups of persons entered in the topic instructor can choose from. In this table instructor ticks off the persons or groups s/he want to have the right to view and participate in the folder. Then s/he moves over to the table to the right by pressing the add button. This operation can be repeated several times until the table

to the right contains the persons and/or the groups instructor wants to have access to the element. S/he can change his or her mind and remove persons or groups from the table to the right by pressing the remove button.

After that, instructor can also select who will have permission to add new elements to the folder in a similar way.

g)   Course creation: authoring tools

The production tools consist of two types of templates: graphic templates (many standard templates, users can create their own templates for their own organization/department/course, or personal templates) and structure templates.

h)   Different users, different learning path

Using groups (see also section 1.f), instructors can create different learning path for users with different capabilities. Since the same user can belong to different groups, instructors have the opportunity to offer to their student as many teaching environment as they need. For example, the same topic can be explained to different groups using different contents: images and tables for users having a photographic memory, or documents rich in details and examples for users who work preferably using an analytic approach.

i)   Time for everything: calendar

The system provides users with a tool to plan course activities. A calendar view can be obtained by every user who has the right access permission, by default it can be viewed both by students and teachers but can be edited only by teachers, unless they change permissions for some others users. Users can also see a personal calendar view restricted to their own group or personal activities. The timetable is often used as a communication tool for course leaders. Information issued in a course appears in the course delegates personal calendars. Among other things, the calendar is linked to the assignment module and the survey and evaluation tool and thus gives the possibility to set deadlines with pertinent notice, follow-up, etc. The calendar may be linked to other calendar tools according to a further defined integration project.

j)   Hidden-in preparation material

Each element, for example a test, can be in an active or inactive status. Instructor will activate the test when s/he wants the students to take it. Moreover it is also possible to decide that an element is to be active from a certain date and hour, to a certain date and hour, or during a defined time interval. This gives teachers the possibility to create teaching arrangements where lessons, tests and assignments are posted to students in accordance with a predetermined plan.

k)   Synchronous communication: chat, video conferencing, etc

It's:learning contains a synchronous Chat tool. This tool allows to create open and closed rooms, password-protected rooms, and allows the leader to throw out participants, etc. (Chat requires that port 8002 or 6667 is open in the firewall).

l)   Asynchronous communication: forum, topic forum, newsgroup

Some forums may be limited to certain users or groups. To view, read, post, etc. users may need special authorization which only the forum moderator and course administrator can grant. Images can also be shown in users' posts. However, there is no facility at present for uploading images directly to this board, so to display the image user must use the appropriate HTML tag (if allowed). Therefore users must link to an image stored on a publicly accessible web server and they cannot link to pictures stored on their own PC (unless it is a publicly accessible server) nor to images stored behind authentication mechanisms, password-protected sites, etc.

There may be two images below a username when viewing posts. The first is an image associated with user's rank; generally these take the form of stars or blocks indicating how many posts he has made or his status on the forums (moderator, administrator, etc). In general users cannot directly change the wording of any rank. Below this may be a larger image known as an avatar; this is generally unique or personal to each user. It is up to the board administrator to enable avatars and they have a choice over the way in which avatars can be made available.

Unless user is the board administrator or forum moderator he can only edit or delete his own posts. Users can modify a post, but sometimes for only a limited time after it was made. If someone has already replied to the post, the user will find a small piece of text output below the post when he return to the topic that lists the number of times he edited it. This will only appear if no one has replied; it also will not appear if moderators or administrators edit the post (they should leave a message saying what they altered and why). Please note that normal users cannot delete a post once someone has replied.

Announcements appear at the top of every page in the forum to which they are posted. Announcements often contain important information and users should read them as soon as possible. Whether or not users can post an announcement depends on the permissions required, which are set by the administrator. Sticky topics appear below any announcements and only on the first page. They are often quite important so users should read them where possible. As with announcements administrator determines what permissions are required to post sticky topics in each forum.

Locked topics are set this way by either the forum moderator or administrator. Users cannot reply to locked topics and any poll contained inside is automatically ended.

When an user posts a new topic (or edit the first post of a topic, if he has permission) he should see a form below the main posting box to add a poll. He should enter a title for the poll and then at least two options. He can also set a time limit for the poll, 0 being an infinite amount. There will be a limit to the number of options you can list, which is set by the board administrator. As with posts, polls can only be edited by the original poster, a moderator, or board administrator. If no one has cast a vote, then users can delete the poll or edit any poll option. However, if people have already placed votes only moderators or administrators can edit or delete it; this is to prevent people rigging polls by changing options mid-way through a poll.

Finally it's:learning provides a private messages service that can be disabled by the administrator for the entire board, or can be forbidden to an individual user.

m) Other

III. ROLES

User's rights to use the system depend on whether s/he is a student, a teacher or an administrator in the system. it's:learning is delivered with the following standard profiles: system administrator, administrator, teacher, student and guest.

a) Learners

Sometimes, even if users have registered by themselves, it can happen that their account needs activation. Some boards will require all new registrations to be activated, either by users themselves or by the administrator, before an user can log on. If this is the case, when a new user registers, the board would have told them whether activation was required by means of an email with instructions. One reason to use activation is to reduce the possibility of rogue users abusing the board anonymously.

Then, if an user forgets his password, while his password cannot be retrieved, it can be reset. To do this he can go to the login page and click "I've forgotten my password". By following the instructions, he should be back online in no time.

Users can change their settings, stored in the database, using the profile tool. They should change their profile setting, for example, for the time-zone to match their particular area, e.g. London, Paris, New York, Sydney, etc. However the board is not designed to handle the changeovers between standard and daylight time so during summer months the time may be an hour late/early from the real local time. In the profile is also possible to set an option to hide user's own status and prevent it from appearing in the online user listing. The user will be counted as a hidden user, and he'll only appear to administrators or to himself.

By default a student can access almost all the functionalities in the platform (calendar, messages, email, search content or persons, private files, web files, personal settings, subscriptions, contacts, access to course, create and access to projects) except administration, of course. Moreover, by default, only users in his courses and projects get access to his personal information.

b)   Authors

It's:learning is an administrative tool for course administrators and course leaders as well as system administrators. The core idea behind the platform is that teachers themselves can decide to make use of their own methods as part of their course content. Therefore teachers are not dependant on administrators to set up and administrate their students and/or course content. In addiction, the system provides a range of automatically generated reports that quickly and efficiently provide a comprehensive overview of a group or and individuals progress within the learning cycle.

However, teachers will not have access, for example, to other users' profiles to change them.

By default, a teacher can access all the functionalities in the platform (calendar, messages, email, search content or persons, detailed information about users, private files, web files, personal settings, subscriptions, contacts, create and access to course, create and access to projects) except administration. Moreover, by default, everybody gets access to his personal information.

c)   Tutors

In it's:learning doesn't exist, by default, a role for a user who helps instructor in his tasks. However, instructors can assign some additional permission to current users. For example, moderators are individuals (or groups of individuals) whose job it is to look after the running of the forums on a daily basis. They have the power to edit or delete posts and lock, unlock, move, delete and split topics in the forum they moderate. Generally moderators are there to prevent people going *off-topic* or posting abusive or offensive material.

d)   Administrators

Administrators are people assigned the highest level of control over the entire board. These people can control all facets of board operation which include setting permissions, banning users, creating groups or moderators, etc. They also have full moderator capabilities in all the forums. However, it's recommended that the system administrator's user account only is used to change settings and to take out information that applies to the entire site. If the system administrator also works as a teacher, all work in connection with the topics and teaching arrangements should be performed under a different user name and password. To keep these roles apart, logging on as a system administrator in it's:learning gives access to a separate user interface where it is impossible to work with topics.

By default an administrator can access all the functionalities in the platform (calendar, messages, email, search content or persons, detailed information about users, private files, web files, personal settings, subscriptions, contacts, create and access to course, create and access to projects). About administration, he has almost the same rights of a system manager (see section below), but he cannot modifies global settings, cannot view logs or license statistics, and cannot administrate search engines. Moreover, by default, nobody gets access to his personal information.

e)   System managers

The system administrator has the possibility to edit three sets of profiles: for the whole site, for course, and for project. The profiles will apply to the entire site. On the other hand, it is possible to create an unlimited number of profiles in each set.

For example: by creating new profiles it is possible to specify that a group of users shall have limited access to functions in the learning platform; students at a lower level of knowledge can have a different profile than more prepared students (in this manner, students can be given access to functions according to their level of knowledge); a college may want to specify that some students shall have the possibility to correct assignments, and thus create a special profile for teaching assistant or student assistant.

By default a system administrator can, of course, access the main page, as all the other user roles, but cannot create, manage or delete any teaching page in the platform. He works on the site from the administration area. He have full rights on global settings, administration of courses, projects, persons and hierarchy, can view either logs and license statistics, and can administrate search engines. Moreover, by default, nobody gets access to system administrator's personal information. Finally, the system administrator uses a simple editor (so ActiveX and Applets aren't permitted), while all the other users use a Java applet editor (excepting the guest user, see section below).

f) Other

By default, a guest user can access the main page of the system, can search content and persons, can access to courses and to projects. He is an anonymous user so nobody has access to his personal information.

IV. RELATIONSHIP WITH STANDARD

It's:learning is a standardised learning platform that supports existing as well as emerging e-learning standards.
It's:learning supports international e-learning standards, such as AICC tracking of standardised course from external suppliers (HACP), SCORM API tracking (SCORM 1.2), and SCORM C&P: however users don't have to know anything about SCORM/AICC or other technical aspects to use them.
In addiction, it's:learning supports accessibility standards, such as WAI-A, WAI-AA and Section 508.

V. USABILITY AND LEARNABILITY

a) Usability test: existence and results

On the system web site, system creators claimed about the importance of usability within their project: "User-friendliness is the critical factor when implementing an e-learning platform, irrespective of whether it is being used by the private or public sectors, or by an educational institution. Despite the fact that different e-learning systems appear to contain the same functionality, it is often the intuitiveness of a systems user-interface that decides whether or not it is accepted and taken into use by the end-user. Most users of the system when asked say that it's:learning's user-friendliness was the primary criteria for their selecting the system."
However no proof of usability test been performed on it's:learning has been found.

b) Learnability test

No documentation has been found about this topic.

c) Accessibility

It's:learning is the first VLE in Europe that supports global accessibility standards. It's:learning supports WAI-A, WAI-AA and Section 508. In addition to this it supports use of Braille readers, screen reader w/speech synthesizer (ex. JAWS) and screen enlarger software like Magic.

d) Multi-language support

At this moment, it's:learning is installed only in few European country: Norway, Sweden and United Kingdom. So its multi-language support is limited only to English, Swedish and Norwegian.

VI. PEDAGOGICAL METHODS

It's:learning allows different pedagogical approaches, because according to its pedagogical philosophy a system like it's:learning is an instrument to improve effectiveness and efficiency in teaching and learning process, and so it has to give instructors as much freedom as possible, even while guiding them.
As for other LMS systems, It's:learning components (chat, forum, quiz etc…) allow the free (hand-made) construction of courses following different pedagogical approaches depending on the Teacher preferences. It's:learning has no specific automatic support for pedagogical methodologies, e.g. to course personalization (even if the Teacher could define several different learning paths in his Course for different groups of students). The tools provided allows the Teacher to follow Socio-constructivist approaches.

VII. INSTALLATION

It:solutions supplies as a basis a pure ASP solution, which does not require any local installations, either on the server or client side. This has many advantages, but primarily users can take part in the continuous development of it's:learning. Users who choose local installation of it's:learning are responsible for running the solution and will only receive updates of it's:learning every six months (2 versions per year).

VIII. MAINTAINABILITY

It's:learning is a result of a thesis done at the College of Bergen, in Norway, in 1998. It:solutions in its first three years of development has gotten the position as market leader in Norway with more than 55% market share in the secondary high schools.

It:solutions is dedicated to the continuous development and distribution of it's:learning.

At first users received new versions of it's:learning perhaps several times each month; all customer specific adjustments and new functions were distributed to all users as they were developed. This development model was obsolete and system providers now introduce new versions twice a year in connection with the start of spring and autumn terms.

Moreover, it:solutions was recently rated by Deloitte & Touché as one of Europe's fastest growing technology companies (Fast 500).

IX. REQUIRED HW or SW

The database must run on a separate server, and users can scale up on the hardware side (Windows 2000 supports up to 32 processors, depending on which server version is used). Users can also set up a redundant solution, with one or more extra database servers.

There are no requirements regarding the client operating system, but it is recommended that the latest service release/package of this system has been installed.

In addition to basic installation of the operating system and IIS, some other components must also be installed by customers that want to run it's:learning on own computer, such as myODBC, Java2 Runtime Environment, Windows Media Server, and mySQL (win32). All service packages, fixer and security patches must be installed as soon as these are available from the suppliers.

About client requisites, there aren't great requests: a web browser, such as Microsoft Explorer 5 (or higher), Netscape 4.7 or 6.2, or Opera 6.0, and a minimum screen resolution of 600X800 (1024X768 recommended).

For example a standard rack machines with Pentium III processors, with 500 simultaneous users, 3 machines (2 web, 1 db) with double CPU 1,4MHz, 2GB RAM, supports up to 900 simultaneous users (approximately). Machines with better processors (faster speed, or e.g., Xeon and the like) will allow users to make their own configuration, with lower requirements regarding the number of processors and machines.

X. LICENSING: OPEN SOURCE or PROPRIETARY

It's:learning is wholly owned, developed and provided by it:solutions as an ASP solution.

XI. TOTAL COST OF OWNERSHIP

It's:learning is a product developed by it:solutions and its licensing costs are available only on request. Some companies provides It's:learning as distributors, and supports its usage. For instance, WorldOnWeb offers It's:learning both on large and small volumes: their price model is based on the fact that customers only pay for actual users, not potential users. That means that users don't need to know exactly how many course participants will use the system, because they are only registered as users the day they actively log into the system. This can give big advantages over a large-scale rollout of the system, since it is very reasonable to start a pilot project. At the same time WorldOnWeb have a price matrix which give user discount when the number of users rise.

However its costs could be high (but there aren't sure information), It's:learning has a low user-threshold with only a minimum requirement for PC and Internet competencies, so it doesn't need cost to training users or to update hardware supports.

XII. USAGE

It's:learning is based on Microsoft technology, and has almost unlimited scaling possibilities.
It:solutions, the it's:learning company provider, is one of the leading e-learning companies in Scandinavia with more than 350.000 users.
Moreover, the Norway's largest university (NTNU), after an year and half of testing with four different LMS, chose it's:learning as its Learning Platform.

XIII. SUPPORT QUALITY

It:solutions consultancy services provide the customer with the opportunity to increase the value of their it's:learning solution in a variety of different areas. The consultancy department focuses on three primary areas: the system integration (it:solutions has extensive experience in integrating student administration systems, user administration systems and financial and personnel systems), application enhancement (it:solutions can tailor existing it's:learning tools, or create new ones, to fit the customers specific needs), and project management (it:solutions has wide experience in implementing e-learning solutions in a variety of different types of organisation, and can offer assistance, advice and administration through all phases of the project).

it:solutions offers a range of differing training courses for users of it's:learning. The standard pack includes the Beginners course, the Super User course and the Administrator course. In addition it's also available the Advanced Users course.

The Beginners course concentrates on making novices accustomed to the user interface and the platform, e.g. giving an overview of the possibilities provided by the system, telling them how to get started, how to use the online help function and setting practical and theoretical assignments. The course is designed to focus on the various modules in the system and to provide an overview of what options and functionality the platform provides.

The Super User course is function oriented, as this is exactly what a super-user requires. A super-user is a resource person that has comprehensive knowledge of it's:learning as an LMS. This course is of course tailored to the needs of the individual customer.

The Administrator course is designed to provide the necessary knowledge for an administrator to be able to the platform for their organisation. In the event that the super-user cannot solve a specific problem, then the administrators have additional access rights enabling them to tackle a wider range of problems, otherwise they can contact it:solutions for help. This course is of course tailored to the needs of the individual customer, too.

The Advancer Users course is for teachers who wish to use the system as a tool for project oriented work flows and portfolio assessment. This course covers a period of one day and is structured so that participants can work with their own subjects and ideas. A template for problem based learning will also be reviewed and made available to participants after the completion of the course.

8. LMS: LAMS 1.1

LAMS (Learning Activity Management System) is a tool for designing, managing and delivering online collaborative learning activities. It provides teachers with a visual authoring environment for creating sequences of learning activities. In addition LAMS relieves users from administration duties.

LAMS research and development is managed by the non-profit LAMS Foundation, with implementation services provided by LAMS International Pty Ltd. Both organisations are supported by Macquarie University in Sydney Australia as part of the Macquarie E-learning Centre of Excellence (MELCOE).

The LAMS system is in the process of being integrated in the Moodle environment.

I. TEACHING FEATURES

a) Contents

LAMS is focussed on a very specific aspect of e-learning: it provides functionality for designing a "flow" of learning activities, especially those driven by collaboration between students. LAMS is a comprehensive system for the design and implementation of learning activity sequences.

However, LAMS does not perform the functions of a typical LMS, and consequently does neither feature any course administration nor news, email, and calendar functions. It does not provide either any content authoring tools for building web pages, etc. While it is easy to put an URL link to LAMS in a LMS, at present it is not possible to author a sequence in LAMS and run it on a LMS.

Here are all activities allowed in a LAMS sequence: chat (see section 1.l), forum (see section 1.m), grouping (allows to modify the behaviour of other tools, see also section 1.f), multiple choice (see section 1.c), journal (provides a task for learners to record their thoughts, viewable by the teacher, on aspects of a sequence), notice-board (see section 1.m), question&answer (see section 1.c), share resources (allows teachers to point to any e-learning content both online and offline, such as web pages, resources on a CD-ROM, Learning Objects in online repositories, zipped file containing web pages and single PDFs, PowerPoint slides, word processed documents, or photo files). The tool can also be set to allow teachers and/or learners to add URLs or individual files in "run-time", i.e., during the activity within a sequence, rather than during authoring), submit files (allows learners to submit a file for assessment by the teacher, such as an essay, report, etc.), voting (see section 1.c), survey (see section 1.d), and some combinations of these activities, such as Chat&Scribe, Resources&Forum, Chat&Scribe&Journal, Question&Answer&Journal, and Voting&Journal.

b) Course Organization

In LAMS two "systems" are interlaced together.

All users must belong to at least one organisation, called home organisation. The details about the user (username, first name, last name, address, etc) are accessed via their home organisation screen. Most users belong to one organisation only. These organisations are also used for the management of learning designs. In the authoring tool, staff members should see their home organisation in the drop down menu in the open/save dialogue box. This is used to share the learning design with other staff members in the same organisation.

A session class is a list of learners who may join a learning design session. When a learning design session is created, a session organisation is assigned to the session. Learners in the session organisation do not necessarily have to join the session. Therefore the list of learners who take part in a particular session is a subset of the learners in the session organisation. When a user brings up the learner interface, only the sessions that appear are the sessions that are attached to a session class to which the learner belongs.

c)   Verifying the learning

LAMS includes three different tools to verify the level of knowledge of learners which are to be included in sequences.

The **Question&Answer** tool allows teachers to pose a question to learners individually, and after they have entered their response, to see the responses of all their peers presented on a single answer screen. Teachers can set some properties: name, description, title (the web page title), question (the question to be considered, and the instructions to answer it, if any), "Show student's name with answer" option (if this is toggled off, the answers will be presented anonymously), "Define Question in Monitor" option (allows the teacher to type in the question to be answered while the sequence is running), grouping (allows to toggle from whole class mode to small group mode). This activity presents learners with two screens, first the question, then the collated answers: the content of this second screen is not editable by the author.

The **Voting** tool allows teachers to provide learners with a list of options to "vote" on. It is similar to the Question&Answer tool in that Voting is presented over more than one screen, first a list of voting options, followed by a screen showing the individual learner's selection, then a screen showing "progressive" voting results (that is, a screen where learners can see the results coming in from other learners over time), and a final summary screen of group results. Teachers can set some properties: name, description, title (the web page title), instructions (about the purpose of the vote and how to vote), nominations (the categories to be voted on), "Allow Text Entry" option (to allows learners to enter their own nomination as a vote, rather than just selecting from the teacher list), "Define Nominations in Monitor" option (allows the teacher to define the Nominations while the sequence is running), max votes (the maximum number of votes a learner can give), grouping (to toggle from whole class mode to small group mode). Like the answer screen in Question&Answer, the results screen is generated by LAMS, so it is not editable by authors.

The **Multiple Choice** allows teachers to create simple automated assessment questions, including multiple choice and true/false questions. The monitoring area includes additional tools for managing learners marks arising from the multiple choice tool (i.e., download marks in text format). Teachers can set some properties: name, description, title (the web pages title), instructions (e.g. the purpose of the activity), question text (shows the current questions, that can be edited by clicking the edit question button thus opening the appropriate sub-screen), "Allow re-tries" option (to choose whether learners are allowed one or more than one attempts, re-tries can also be linked to a pass mark), "Show top users" option (if this is toggled on, the system shows the name of the "top scoring" users), "Show feedback" option (if this is toggled on, the test includes feedback text on completion), and grouping (to toggle from whole class mode to small group mode).

d)   Verifying the use: survey

The **Survey** tool is similar to Multiple Choice tool, in that it presents learners with a number of questions for responses, but unlike Multiple Choice, there are no right or wrong answers (the system just collates all answers for the teacher to analyse in the monitoring area). Question types include single response, multiple response and open text entry, and questions can be mandatory or optional. Teachers can set some properties: name, description, title (the web page title), instructions (such as its purpose, the use of the data, etc...), questions (a list of questions in the survey that can be modified by selecting any of them and clicking the edit, or delete, or add button, thus opening a sub-screen), and completion message (some text to be showed after the survey is completed). In a survey each question can be either a single answer, multiple answer, or text entry question.

e)   How much knowledge: automatic correction

LAMS allows teachers to monitor the students performances on the fly. The systems allows teachers to help groups and individuals from the monitoring console thus allowing a greater efficiency in the guiding and mentoring activities. All the checking and evaluating tools are available on the monitor screen (see section 2.c).

f)   Organize a class: managing learners group

Teachers can manage and coordinate large groups of students simultaneously and engage as active participants in real time while the system runs the activities. It's possible to have more than one grouping per sequence, and to use different activity tools on different groupings. Currently the Grouping tool only provides random grouping options, but two additional options are planned for the future: teacher-allocated grouping and learner-selected grouping.

g)   Who see what: access rights to course and contents

Teachers can choose to share their own sequences with other users in two ways. First, when a teacher saves a sequence in LAMS, there are three possible areas to save it in: his own "Private" area (which is only viewable by her or himself); a "Group" area (which could be the Department/Faculty/etc, this is set up by the system administrator), or the "Public" area (which is viewable by all users on the LAMS server). If a teacher places the sequence in the Group or Public areas, this will be viewable by those who can access to this area of the server: they can view it, use it, adapt it, but are not allowed to edit the original file (only the original author can).

Teachers can share sequences also via the authoring Import and Export feature. They can export a sequence as a file (typically it is quite small – around 20-50K, if no other resource objects are embedded in it), which they can then store on a disk or on a network drive. They can even attach a sequence to an email to be sent to colleagues, who can import it back into their LAMS server, and use it. This process allows users to share sequences with others who are not on the same LAMS server.

h)   Course creation: authoring tools

LAMS allows user to adapt and share collaborative sequences, building libraries of best practice objects to be used in many subject areas, and adapted through the visual drag and drop authoring environment. This allows users to save time, to leverage best practice and to adapt it for their own particular purposes.

i)   Different users, different learning path

LAMS copes with "non-linear" sequence structures using the "optional box" feature in the menu. This box collects all sequence management tools (see also section 2.b), which allows flexibility of pathways through different sets of material according to student preferences, etc. This approach can be used to allow students to follow different activities or content paths according to their interests, or to mix individual and group tasks according to their learning style.

LAMS plans to implement branching or conditional sequencing in future versions. In the interim, the combination of boxes of optional activities plus instruction text from teachers is able to fulfil simple "non-linear" requirements.

j)   Time for everything: calendar

LAMS does not have a calendar, or other similar features.

k)   Hidden/in preparation material

In LAMS teachers can hide an activity simply creating a sequence without any transitions coming into it: so learners will not be able to reach it, nor to see it in any way.

Moreover, teachers can have a preview of their work. Within the "authoring" environment teachers can preview LAMS sequences by saving their current work and clicking the preview button: this will immediately show them a learner view of their sequences. Then, they can return to authoring, edit and save changes, and preview the new version immediately. This approach avoids problems associated with "locking" of sequences when they are run in "monitor" or "learner" environments. However, once a sequence has been "attempted", it is fixed forever. If teacher decides he wants to change the sequence mid-way through, he could go back to authoring, open the original sequence, save a copy and then delete the early (completed) activities, change the structure of the remaining activities, and then start it as a new sequence for students (thus geting them to leave the earlier sequence half-way through, and begin the new sequence).

l)   Synchronous communication: chat, video conferencing, etc

The Chat tool runs a live chat session for learners. Teachers can set some properties: name, description, title (the web page title), instructions (to tell learners what they can discuss within the chat session, or any other information about this tool), grouping (to toggle from whole class mode to in small group mode), and "lock when finished" option (to control whether learners can continue to use the chat tool after they finished the sequence).

LAMS don't offer other synchronous communication tools like videoconferencing or a shared whiteboard.

m)   Asynchronous communication: forum, topic forum, newsgroup

LAMS provides two different activities tools which can be included in sequences to help users communication.

The Forum tool provides an asynchronous discussion environment for learners, with discussion threads created by the teacher. Teachers can set some properties: name, description, title (the web page title), instructions (about the purpose of the forum, etc...), topics (shows the titles of the current topics, creation privileges (note that those who are able to create topics will be able also to delete them, using the "Edit Topic" button to open the Edit Topic sub-screen), grouping (to toggle from whole class mode to in small group mode), and "Lock when finished" option (to stop learners from contributing to the forum after they finished the activity).

The Notice-board tool allows a teacher to present text to learners, for example to communicate exams date or news about the course. Teachers can set some properties: name, description, title (the web page title), content (they can enter, using a Rich Text Editor, the text content, such as instructions, educational content, etc...), "Define content in Monitor" option (to change the text of the page while a sequence is running), and grouping (to toggle from whole class mode to in small group mode).

n)   Other

II. ROLES

LAMS access rights are based on four roles, each with its own privileges: administrator, author, staff and learner. These roles are connected with the four main areas in LAMS, viz. author, monitor, learner and administrator. Each area has the appropriate tools and help facilities. Even if a user can have multiple system-wide roles, each user needs only one username (with password) to access the platform, independently from his or her roles. Moreover each area is accessible using a simple button in the welcome page, after the login, so users can see the accessible areas when they enter the platform.

a)   Learners

The learner area is where a group of learners (viz. users with learner role) participate in a sequence that has been activated for them by a teacher.

The main learner page shows the list of LAMS sequences each user can access, a progress bar, a notebook and an area where the activity tools are displayed. To start a sequence is as simple as clicking it in the list, and instructions and activities will be displayed in the activity tool area. Sequences will be listed according to their status: available, new, started or finished. In fact, the progress bar shows users where he is up to in his activity sequence and it's colour coded: pink indicates the current activity, blue is for finished tasks, and green indicates tasks not attempted, that cannot be accessed until previous activity is completed. Every activity tool has a Finished button: when an user has completed his current activity, he will click this button to move on to the next activity. Finally the notebook area is a space where users can record personal notes during a sequence: it is always available in a frame on the left.

b)   Authors

The author area allows teachers (or users with author role) to create and modify sequences of learning activities, and store these in the sequence repository. The authoring area is made of three main parts: the main panel where sequences are created or edited, the menu with tools for sequences management and the panel with activity tools.

In the main central area icons represent activities in the sequence and arrows (that connect an icon to another) represents transitions from an activity to another. To edit, move and delete activities or transitions properties is easy: to create them, double click them and edit the pop up panel; to move activities click, hold and drag the icon to a different location; finally, to delete activities or transitions click, hold and drag the object to the trash icon on the area (deleting an activity will delete any related transitions).

The menu collecting all sequence management tools contains features applying to a whole sequence (open, save). An operation can be performed either using the drop down menus (Files, Tools and Help), or using the "frequent task" buttons. The menus allow, for example, to create a new sequence, to revert to the last saved version of a sequence, to import (or export) a LAMS sequence file from (or to) the user desktop, to draw simple transition lines between two activities (a teacher can choose to close the access to the next activity until he releases it, this setting can also be changed later), or to draw optional boxes around two or more activities (see also section 1.i). Teachers can add to the optional box a title and some instructions, and then set the minimum number of activities to be completed, so learners must complete at least this number of tasks before being allowed to move on to the next activity. Optional boxes cannot contain other optional boxes. The eight frequent task buttons provide easy access to common tasks, i.e. New, Copy, Paste, etc. The Copy and Paste buttons allows to duplicate the original activity or optional box (an exception to this rule holds for Share Resources: any files incorporated into this activity will not be copied when the activity is pasted).

The activity tools panel shows all the main components of a sequence: chat, forum, grouping, etc. Clicking on one of these activities, teachers can insert in their panel, and then set the relevant parameters.

c)   Tutors

The monitor area allows teachers (or users with staff role) to select a sequence from the sequences repository, assign to it a group of learners, activate it for learners, and then monitor their progresses. The monitor screen has two main parts: the session management area and the individual sequence area.

The session management area allows users to manage all the active sequences with their learner groups (including an area for archiving old sequences). Here teachers can: create new sessions based on their sequences, archive sequences moving them into the archive, open or close the active sessions folder or the archive to see their content.

The individual sequence session area is for managing a single session: by clicking on a sequence in the session management area on the left, the details for running the session are shown on the right. The three tabs at the top right of the window provide different information to monitor a running sequence. The Session Info tab provides basic information on the sequence session besides allowing to perform some actions, such as: status (the current status of the sequence, i.e. new, started, etc...), learners (the number of learners from the selected class who have joined this sequence so far, i.e. 2 of 9), move to archive (moves permanently a session to the Archive area, making the session unavailable to learners), and start/disable/enable (starting a session makes it available to learners, disabling a session deletes it from the available sequences area and enabling a session makes a disabled sequence available again). The Sequence tab provides a "sequence-based" view of a learner progression through a live session; teachers can choose to see a green circle for each learner in the session at their current location within the set of activities (in real time). Finally, the Learners tab provides a "learner-based" view of learner progression through a live sequence session: learners are presented in a layered horizontal view of their progresses, with a blue square indicated a completed activity, a red square indicating the current location of the learner, and a green square indicating activities yet to be performed.

d) Administrators

The administration area is for system administrators (or users with administrator role) to set up LAMS users (including teachers and learners), manage user groups and perform other system administration functions. It's divided in two areas: Maintain Organisation and Maintain Session Classes.

The Maintain Users function in the Organisation screen allows administrators to create new learners and to update the personal details of learners and staff. This screen lists all the users belonging to the same home organisation (see also section 1.b). To add a new staff member, the system shows a new screen: the top part of the screen lists all the organisations and session classes which this user will be enabled to access, then some personal data (name and e-mail), a form to insert the password (or to change it), and other details (job title, department, ...). Finally the screen presents four checkboxes (learner, staff, author, administrator) to choose users' roles: staff will usually have all of them ticked. To add a new learner, the system shows a new screen that will list user membership data (username, first and last name, and e-mail), a form to insert the password (or to change it), and other details. Finally the screen shows the four user's roles checkboxes, but in this case author and administrator roles are disabled.

The Maintain Session Classes screen allows administrator to add both staff and learner users to a session class. To do it, just select an organisation from the drop down list to see the list, and then tick the checkbox next to the people who you wish to add to this session class. Administrator can also add users to a Session Class by uploading an Excel file containing the users to be added.

e) System managers

In LAMS there isn't a specific role for system managers, but tools are available to perform advanced operations in the administration area, that give to administrator privileges usually associated with this role. The System Administration screens are the "power user" version of the User Management screens, available for administrator. Using these screens a user can, for example, add a new top level organisation, create new organisations, add or remove an existing user to an organisation, change a user's personal details, etc...

f) Other

III. RELATIONSHIP WITH STANDARD

LAMS is based on the concepts of the IMS Learning Design standard, whose diffusion is increasing. So, maybe, when more systems will implement this approach, it will be possible to author sequences in LAMS and run them on another platform that also implements IMS Learning Design. LAMS provides upload and delivery features for IMS Content Packages as far as they conform to the "index.htm" requirement (if the content has a central webpage named index.html in the root folder, it can be uploaded into LAMS). In future versions LAMS will support SCORM objects, too.

IV. USABILITY AND LEARNABILITY

a) Usability test: existence and results

No documentation is available about LAMS usability tests.

b) Learnability test

No documentation is available about learnability test for LAMS.

c) Accessibility

LAMS has implemented W3C Level A accessibility requirements for the learner interface (within the constraints of the tools used, such as Flash).

d) **Multi-language support**

No documentation is available about this.

V. *PEDAGOGICAL METHODS*

According to LAMS creators, "the lack of a mature approach to sequencing of multi-learner activities is a significant blind spot in e-learning today. This is surprising given that "lesson planning" (the process of determining the sequence of activities to be followed by a teacher and students when studying a topic) is well understood in education, but is mainly absent from e-learning. However, there is a growing body of work addressing this blind spot, based on the concept of "Learning Design". Learning Design provides the first practical ways of describing multi-learner activity sequences and the tools required to support these. Learning Design has emerged as one of the most significant recent developments in e-learning (...). Hence, the successful implementation of the IMS Learning Design specification has the potential to transform e-learning, and expertise in implementing IMS Learning Design will prove to be a very valuable skill. While definitions of Learning Design vary, the main elements tend to include greater focus on "context" dimensions of e-learning (rather than simply "content"), a more "activity" based view of e-learning (rather than "absorption"), and greater recognition of the role of "multi-learner" (rather than just single learner) environments. While Learning Design does not exclude single learner, self-paced modes of e-learning, it draws attention to a wider range of collaborative e-learning approaches in addition to single learner approaches. Much of the focus on Learning Design arises from a desire for re-use and adaptation at a level above simply re-using and adapting content objects.".

But, in their opinion, the existing approaches to Learning Design have several major drawbacks: "First, they focus too narrowly on one particular discipline area or style of Learning Design (...). Second, the attempts at a more generally applicable approach to Learning Design have been conceptually flawed, particularly as they relate to describing multi-learner activities, hence limiting their ability to generalise to all educational settings. Third, most Learning Design projects have not separated their conceptual model from the use of technology, and hence the outcomes only relate to e-learning, not any form of learning. The challenge for Learning Design is to describe a foundational conceptual framework, applicable to all educational settings, which is independent of the mode of delivery (face to face, technology, etc).".

Finally, LAMS developers stress that "LAMS certainly doesn't ensure good pedagogy: it is possible to build lousy pedagogy in LAMS just as much as it is possible to build good pedagogy. It is hoped that LAMS encourages good pedagogy as a result of its activity-centric view of the educational process, coupled with its visual authoring environment that helps to make explicit to teachers the nature of their educational processes.".

VI. *INSTALLATION*

The LAMS official website allows to download LAMS installers and packages. The installer includes all software packages necessary for an offline install.

Moreover, LAMS provides two detailed guides for Windows installer (basic and advanced installation, in two different pdf files, one for a quick installation with default settings, the other for a customised installation, allowing changing of ports, locations and storage settings), with step by step instructions and screenshots on how to fill out and successfully complete a LAMS installation on Windows, and a manual installation guide that details how LAMS can be set up from scratch, for Windows, Red Hat or Solaris.

VII. *MAINTAINABILITY*

LAMS is built using standard components and highly scalable back end systems. Users can download LAMS source code from the website and integrate it into the client's existing Learning Management System or set it as a stand alone system. Since LAMS has been implemented to allow the application to have presentation, logic and workflow cleanly separated, thus allowing a simplified development and

modifications to look and feel, it's quite simple to make changes. LAMS users can even suggest new features to be added to the system in the LAMS Community website, that keeps a public list of requested features and plans to implement them.

Finally, the long term sustainability of LAMS is based on two strategies. First, the LAMS Foundation plans to get funding for LAMS research and development from governments and charities to continue the development of LAMS until 2007 (after which, it is anticipated that the developer community will have grown to the point to make it self-sustaining). In parallel, there is a company, LAMS International Pty Ltd, working together with the LAMS Foundation to develop a sustainable revenue stream that can assist in funding future development of LAMS.

VIII. REQUIRED HW or SW

LAMS is built to run on all the most popular server platforms, since it's based on the Java programming language. Talking about server minimum requisites, the recommended operating systems are Windows 2000 Server or later, RedHat Enterprise Linux 3 AS or later and Solaris 9 or later, the hardware specification depends on the number of classes and user; for instance, for more than ten classes with hundreds of users LAMS requires an Intel P4 Dual 3GHz processor or equivalent, with more than 2GB of RAM, 1GB of HDD space, plus backup space. Finally, talking about the network requirements, LAMS needs one port for general access (configurable, port 8080 by default), and another for chat functionality (configurable, port 9800 by default).

There is no desktop software for users to install to use LAMS. LAMS runs through users internet browser and connects them automatically to their class with real-time communication. The Flash User Interface runs inside the browser and no extra client software is required. About client minimum requisites, hardware required is a Pentium III 600MHz processor or higher, with 128MB or more RAM. Then the recommended operating systems are Windows 98 or higher, Mac OS X 10.1 or higher, Red Hat Linux 7.3 or higher, while the recommended browsers are Internet Explorer 5.5 or higher, Firefox 1.0 or higher, Safari 1.2 or higher. Browsers should have Flash 7.0 or higher browser plug-in installed, pop-up blocking (if any) disabled, and Cookies enabled. Finally, about network connectivity, LAMS needs at least a 14.4kbps connection to the LAMS Server (broadband recommended), and the outgoing connections should be allowed to ports 80, 8080, 9800 on the LAMS Server.

IX. LICENSING: OPEN SOURCE or PROPRIETARY

LAMS is freely available as open source software under the GNU General Public License (GPL); this means that users will be able to look at, use and modify it freely. Open source GPL software is also characterised by the fact that any user who modifies it and then redistributes it, must make the modified version also 'open source' (GPL).

X. TOTAL COST OF OWNERSHIP

Anyone who wishes to take the publicly available software source code, and then compile it, install it, configure it and maintain it can use LAMS without incurring any fees. Alternatively, for many organisations, it may be more cost effective to have LAMS International (the LAMS services and support company) provide support services for a modest fee, but this decision is up to each potential user/organisation.

XI. USAGE

LAMS has impressed many senior educators who have followed its progress in a series of trials around the world prior to its official launch, with the software already being used at several Australian universities. In fact, during its testing phase in 2004 and in early 2005, some partners and organisation contributed supporting and testing sites: Oxford University, University of Cambridge, Australian

National University, University of Canberra, University of Huddersfield, University of Sussex, Nanyang Technological University, National Library of New Zeeland, Christchurch City Council Libraries, and Department of Education in Tasmania. At now, LAMS can be used in any education and training setting: it is equally applicable in primary/elementary schools, vocational and corporate training, and in adult/community learning.

XII. SUPPORT QUALITY

LAMS allow several types of support for its users.

LAMS International and its associated LAMS Foundation are available to provide consulting services and expertise in the set up and implementation of LAMS in various situations. LAMS International offers a variety of ways to contact it: through e-mail (for general or technical enquires and information), through phone (both for LAMS Foundation and LAMS International), and through a form in the official website to submit general enquiries, the last option allowing users to receive answers directly via e-mail. Developers have a broad experience both in the technical and pedagogical elements of a LAMS implementation. They can help users to plan and implement an educational strategy for their institution or business using the powerful functionality delivered within LAMS. Moreover, LAMS developers provides teacher training and technical support using e-mail to all their customers. On the LAMS official website there is a page indexing the installation guides/instructions and the manuals for LAMS: users can find step by step installation instructions, advanced options to further customise their installation, and also manuals for using LAMS.

The LAMS Community website is the online community for all teachers and administrators using LAMS. This website provides teachers with interactive tools that can be used to share their views on latest developments of LAMS and new learning designs as well as to share their sequences with others on a global scale.

Finally, the LAMS Launch CD is a compilation of documents, videos, case studies, news, general resources and the LAMS Installers for most platforms. Most resources are part of the official website, but some third party resources are provided as links to the relevant internet site.

9. LMS: LON-CAPA 1.3

(add description)

I. TEACHING FEATURES

a) Contents

LON-CAPA provides several types of contents: *quizzes, exams and tests* (see also section 1.c), *calendar* (see also section 1.j), *chat* (see also section 1.l), *forum* (see also section 1.m), *syllabus* (to present subjects, exams, teacher's contacts and bibliography of the course) and *announcements* (can be inserted everywhere within a course, can be easily posted, managed and archived, and there is an option which allows to email class as well as to post).

LON-CAPA allows a complete content sharing and reusability: a network of shared resources from tens of participating institutions and a shared repository of more than 50000 resources, including assessment questions and multimedia content, is available. Contents creators can use a large set of resources in physics/astronomy, chemistry, biology, mathematics/statistics, and geology, or can add new resources to the existing content pool. Course contents can be organized by week or by module, and is possible to insert time-released contents.

The system is compatible with all the most diffused media software: PowerPoint (with audio, too), Real Video, QuickTime, Windows media player and Flash.

The rich text editor allows the creation of mathematical equations in LaTEX format.

b)   Course Organization

In LON-CAPA courses can be organized in semesters: a course can be offered to several separate sections and instructors can view previous and current semester, by section. Instructors can copy contents or structures to new semester, too.

c)   Verifying the learning

LON-CAPA allows creation and grading (see also section 1.e) of randomized homework, quizzes or exams, including the ability to reduce blind copying of answers by giving a different version of the problems to each student, and to print randomized bubble-sheet exams (answered by marking answer paper forms to be optical scanned) and quizzes,

Instructors can create "sophisticated" question types: true/false, multiple choice, multiple answer, ordering, fill-in-the-blank, image map (click on the correct part of the image), matching, short answer/essay and calculated answer. Questions can contain images, video or other media files. If needed, instructors can import questions from existing test banks. Instructors can create personal, course specific, and system-wide test banks and use these test banks to create tests for students. The system can randomize the questions in a test, the alternatives for multiple choice questions as well as the numbers in numerical or symbolic questions. Instructors can set times for when students can access tests and a time limit on a test. Instructors can permit multiple attempts, and toggle the display of correct results. Finally instructors can use IP addresses to restrict access to tests. The system provides test analysis data for individual test items.

d)   Verifying the use: survey

LON-CAPA doesn't provide a specific survey tool, but instructors can create survey questions by using the quiz question tool, blocking answer evaluation and allowing anonymous attempt.

e)   How much knowledge: automatic correction

The system automatically scores multiple choice, true/false, and multiple answer type questions, and can display instructor-created feedback (answer-specific, too), hints, explanations, and links to relevant course material. The reporting of grades and feedbacks can be configured with a wide variety of options, weighting tests in different ways and creating grading rules.

Finally instructors can override the automated scoring.

f)   Organize a class: managing learners group

Instructors cannot create groups, so the system doesn't allow groupwork. Students cannot create student communities, either.

g)   Who sees what: access rights to course and contents

Instructors can customize specific access permissions for each student, banning them from accessing some course sections, or contents. Teachers can also allow them to manage course components, differently barred.

h)   Course creation: authoring tools

Instructors can use templates to create course content, course units, syllabus and course descriptions. Course content may be uploaded (using a form) or chosen from a system-wide content library.

i)   Different users, different learning path

Instructors can specify multiple paths through courses for different skill levels or job functions. Instructors can create both linear and nonlinear learning sequences using a content library. Learning sequences are reusable and become part of the content library. For example, instructors can set up specific course content that must be completed before students are allowed to continue. This system provides a sophisticated assignment engine that can create unique homework assignments and examinations for each student in a class. Its formative and summative assessment tools grade a broad variety of objective problems and assist essay evaluation.

j)   Time for everything: calendar

Students can view all their assignments, deadlines, and due dates in an online course calendar. Instructors can post events and announcements in the online course calendar or course homepage. The calendar tool provides also a calendar view which combines all course-specific calendars. Students cannot add personal items to calendar.

k)   Hidden/in preparation material

The system allows instructors to hide material in preparation so that students do not have access while they are building contents. For example, while creating an exam, is possible to hide the folder where chosen questions are listed.
In general, if desired, it is always possible to hide a folder in the documents section of a course.

l)   Synchronous communication: chat, video conferencing, etc

There is a basic chat tool which supports anonymous posting, and allows to archive conversations. There's no video services or shared whiteboard.

m)   Asynchronous communication: forum, topic forum, newsgroup

LON-CAPA provides a discussion forums tool, where discussions can be viewed by thread and then can be saved or printed for offline reading. Students can post anonymously and can associate a discussion with any course content. Only instructors may delete posts. Posts can be either formatted text, or html, and can include attachments, images or URLs.
Students can use the internal email feature to email individuals or can forward message notifications to an external email account. While using LON-CAPA file sharing is not available.

n)   Other

II. ROLES

In LON-CAPA system there are a lot of different user roles (staff/admin, instructor, student, guest, public) and individual privileges can be modified, including creating custom roles.

a)   Learners

All students have a personal home page that lists new email, all courses in which the student is enrolled and all course and system-wide events. Personal home pages may include a photo, personal information, and links, for example, to a private folder where student's work is uploaded. Students can view their completed and pending course activities, and can view their grades on completed assignments or attempts.
Students can compile course content into a format that can be printed and browsed offline, and can download the content for an entire course to a version usable on a PDA.
Students can attach personal notes and bookmark to any content material in a course, and categorize their bookmarks in a private folder. Moreover they can make a search among all documents and threaded discussions.

b)   Authors

The LON-CAPA software provides instructors with a common, scalable platform to assist them in all aspects of teaching a course, from lecture preparation to administration of homework assignments and exams. It allows instructors to create educational materials and to share these learning resources with colleagues across institutions in a simple and efficient manner.
Instructors can assign different levels of access to their course, by giving different privileges to the following pre-defined roles: instructors, students, teaching assistants, guests and staff, but they can customize specific access permissions for each student, too.

Instructors can view grades in the gradebook by assignment, by student, and for all students on all assignments. Moreover, they can add the grades given to offline assignments to the online gradebook and can create a comma-delimited version of the gradebook for export to an external spreadsheet program.

Instructors can get reports either showing the access number for all students in a course (as an aggregated group), or showing attempts number and time per attempt on each assessment for individual students, or summarising individual student performance on all assignments. Instructors can maintain private notes about each student in a secure area.

Finally, instructors can control start and end date of a course, allowing a class to start before the beginning of the semester, or stay open after the end of the semester; they can selectively release materials based on specific start and end dates; or they can personalize access to specific course materials based on specific start and end dates.

c) Tutors

In LON-CAPA these rules can be easily changed by an author with higher privileges, modifying access rights: for example, deleting his/her ability of modifying course contents.

d) Administrators

Administrators can protect access to individual courses with a username and password, and can batch add students to a course using a delimited text file. The software supports integration with student information systems via a tool based on scheduled system exports. Then Administrators can delegate their power to an instructor, so this figure is not mandatory to manage the system.

e) System managers

In LON-CAPA this role is not provided, and its tasks are carried out by administrators or instructors.

f) Other

LON-CAPA allows the existence of a guest user that can enter the course, but doesn't have any other access right.

III. RELATIONSHIP WITH STANDARD

The software can import course content that is IMS compliant. The product provider is committed to give assistance to any institution that choose LON-CAPA, leaving another system, to ensure a complete migration of existing courses into the system.

IV. USABILITY AND LEARNABILITY

a) Usability test: existence and results

In Fall 2003, the LON-CAPA group conducted a survey for a pilot study regarding course management systems. Approximately 300 surveys were sent to faculty members and administrators randomly gathered from the web pages of institutions of higher education in the United States and Canada. So far, 37 surveys were returned. The preliminary results from this pilot study will be used in the development of a full-scale study, and are presented in the site (http://www.lon-capa.org/pilotsurvey/index.html), but no one of the question presented is about usability.

b) Learnability test

No documentation has been found about LON-CAPA learnability test.

c) Accessibility

To enable accessibility, the software implements many features: contents are available without colour or in a text-only versions, there's the ability to suppress Java applets and to increase font size.

The system includes an online user manual, a FAQ page and context sensitive help.
Finally, in the system web site there is a detailed documentation, that includes a FAQ section, two short guide, to build a course and an exam, a course coordinator manual, an author manual and a training manual.

d) Multi-language support

The system provides support for multiple languages within one document. The interface is internationalized and provides partial localization for German, Japanese, Portuguese, Russian, Arabic, Farsi, and Turkish.

V. PEDAGOGICAL METHODS

LON-CAPA is a teaching tool, not a curriculum, and as such does not dictates course design, content or goals. Instead, it enables faculty to augment their courses with individualized relevant exercises.
On the web site there is an interesting document (rich in bibliography) written by Gerd Kortemeyer in November 2003 about research themes and methodology (http://www.lon-capa.org/researchthemes.html) that presents several theories about interactive learning environments and other themes connected with the pedagogical approach in an e-learning environment, like LON-CAPA.
Here there are some extracts from this document about "The impact of LON-CAPA feedback and help on learners with different epistemological beliefs": "(...) Previous research indicates that in general learners do not necessarily like LON-CAPA. (...) In the case of CAPA, Pascarella [note: author of "CAPA in a large university setting"] found evidence of different learning impact on students with different epistemologies (students' views about the nature of knowledge and also their views about the nature of learning) and problem-solving strategies: students with expert-like views of learning judged traditional homework to be the better learning tool. (...) Pascarella attributed the undesirable impact of online homework on metacognition to the immediate "correct/incorrect" outcome feedback (...) Yet, immediate feedback is widely recognized as an important benefit of online exercises (...) Beyond simple "yes/no" outcome feedback, already available in CAPA, feedback can consist of hints, which in their simplest form are supposed to help the students find the correct solution. Pasceralla found that students in general did not find these hints helpful (...) Since in CAPA and LON-CAPA material is provided by a large number of authors, hints vary strongly in their approach to being "helpful" (...) At the time Pascarella conducted her study, this system only had a very limited feedback and hint mechanism. Today, LON-CAPA allows for hints to be adapted to pre-anticipated wrong solutions, hints and feedback also on correct solutions, follow-up questions, the disabling of any correct/incorrect-feedback, as well as far more flexible control over the number and reward-structure of multiple tries. (...)".
Here there are some extracts from this document about "The impact of LON-CAPA personalization on student collaboration and cheating": " (...) it is frequently claimed that personalization prevents students from cheating themselves, and that it enables students to solve problems in groups without the temptation of simply exchanging solutions, and instead to discuss the problem on a more "conceptual" level (...) In most group work scenarios toward solving a problem, participants quickly assume or are assigned different roles, both active and passive. It can be hypothesized that personalized assignments will reduce this tendency, especially that of assuming a more passive role, since it is not enough for the group to solve one problem – every participant in the end has to solve their own. (...)".
As for other LMS systems, LON-CAPA components (chat, forum, quiz etc…) allow the free (hand-made) construction of courses following different pedagogical approaches depending on the Teacher preferences. LON-CAPA has no specific automatic support for pedagogical methodologies, e.g. to course personalization (even if the Teacher could define several different learning paths in the course). The tools provided allows the Teacher to follow Socio-constructivist approaches.

VI. INSTALLATION

LON-CAPA is based upon many open source modules, so it's important to have the right environment on the computer where it will be installed. It needs a dedicated machine. It does not get along well at all with other applications, especially other web applications.
If it's necessary to make extensive use of LON-CAPA on a departmental or campus-wide level, it is generally a good idea to locally install the system.

The system can be downloaded and installed for free. It could be useful to contact LON-CAPA to have it linked to the existing cluster, but it's possible also to install a standalone machine, or to start a new cluster. LON-CAPA system would like users to succeed in installing the system and linking it to the existing cluster, and would like to offer remote installation assistance within its capacity. Also, it offers a system administrator mailing list and an on-site installation support and technical assistance, as well as remote maintenance options.

VII. MAINTAINABILITY

LON-CAPA is based on a development group that sustains a twice-yearly release cycle to insure rapid incorporation of instructor requested enhancements.

VIII. REQUIRED HW or SW

The software works with most versions of Netscape, Internet Explorer, Safari, Firebird, Camino, Galeon, and Mozilla. Javascript and cookies must be enabled. The software is not compatible with Opera, Konqueror, and Mosaic, except in a limited text-based interface mode. The same limited text-based interface is available on some PDAs, namely Opera on Sharp Zaurus and Thunderhawk on Windows CE.
The separate database software required for the system is MySQL.
The software requires that ports 5663, 8080 and 80 be available. The system can load-balance within server clusters.
About hardware suggested, minimum recommendations are: a multiprocessor machine, 2Ghz CPU, 4 GB RAM and at least 60 to 120 GB disk space.

IX. LICENSING: OPEN SOURCE or PROPRIETARY

The software is distributed under the terms of the GNU General Public License.

X. TOTAL COST OF OWNERSHIP

LON-CAPA is an open-source freeware system. Neither the software nor the cluster membership is on payment. But running LON-CAPA is not for free - it will need a server (which is the cheapest of its needs), and above all it will need time, energy, and human resources for the maintenance of the system and the support of faculty and students. In terms of server maintenance, LON-CAPA invites its cluster members to keep the operating system updated and secure, and install new versions of LON-CAPA as they become available. It do offer on-site installation and remote maintenance support for a fee, as well as hosting services.

XI. USAGE

Current members of the LON-CAPA project include faculty from universities, colleges, and K-12 schools in the US, Canada, Asia, Africa, and Europe. The core development group is located at Michigan State University.
A lot of institutions have servers in the LON-CAPA production cluster: middle and high schools in USA, supported and hosted by Florida State University K-12 Consortium, Ohio University, community colleges in USA and universities (at department level), from the Michigan State University, to University of Illinois at Urbana-Champaign, from Florida State University to Universidad de Sao Paulo, from The Hebrew University of Jerusalem to Kanawaza University.
Then, all these schools have hundreds of students: for example the Florida State University has an average of students course enrolments per term of 1300 units, while the Michigan State University has 16600! In particular, in the LON-CAPA web site (http://www.lon-capa.org/usage.html) there is a report about LON-CAPA usage history at MSU that shows that from spring 2001 to spring 2005 the number of using LON-

CAPA courses in the respective systems grew to include almost all courses. In spring 2004, LON-CAPA was serving over 13,000 course enrolments per semester at MSU alone, and well over 23,000 course enrolments system-wide, ranging from middle school to graduate level courses. Disciplines include astronomy, biology, business, chemistry, civil engineering, computer science, family and child ecology, geology, human food and nutrition, human medicine, mathematics, medical technology, physics, and psychology.

XII. SUPPORT QUALITY

The software was developed by the Laboratory for Instructional Technology in Education at Michigan State University. Funding is provided by Michigan State University, the National Science Foundation, as well as the Alfred P. Sloan Foundation, and the Andrew. W. Mellon Foundation. The software has origins in two earlier projects, CAPA (a Computer-Assisted Personalized Approach), that provided students with personalized problem sets, quizzes, and exams, and LectureOnline, a project to serve physics course material over the web. The project provides gateways to NSF's National Science Digital Library.

In the system web page there is a section where are listed all the publications connected to the LON-CAPA project since 1993, which confirms the relevance of this system in the LMS development environment, and a section where are listed all the awards and recognitions obtained by LON-CAPA project. For example, in 2003, the project won the ComputerWorld Honors 21st Century Achievement Award and, in 2004, it won the Sloan-C Award for Most Outstanding Effective Practice.

10. LMS: Moodle 1.5

(add description)

I. TEACHING FEATURES

a) Contents

Two different types of content can be inserted in a course: resources or activities.
A *resource* can be:
- a text page (meaning a text file, written in plain text),
- an HTML page (meaning a single web page developed within Moodle),
- a directory available from the course-files area,
- a *label* (simple text and images that are actually embedded directly among the other activity links in the course page),
- a link to any other resource available on the web (in alternative to such linking, if the resource is directly available on the author's pc, it can also be directly uploaded in the course-files area).

An *activity* can be:
- an assignment (see also section 1.e),
- a chat (see also section 1.m),
- a choice (see also section 1.c),
- a forum (see also section 1.n),
- a glossary (allows participants to create and maintain a list of definitions, like a dictionary),
- a journal (see also section 1.e),
- a label (a "dummy" activity that allows to insert text and graphics among the other activities on the course page),
- a lesson (which is basically a set of pages, through which the learner can navigate, also depending on answers to test questions (?)),
- a quiz (see also section 1.c),
- a resource (it may be a file uploaded to the course server, or a page edited directly in Moodle or an external web page linked in such a way to appear part of this course),

- a SCORM package (a bundle of web content packaged in a way that follows the SCORM standard for learning objects),
- a survey (see also section 1.d),
- a wiki (allows to author documents collectively in a simple markup language, by using a web browser);
- a workshop (allows participants to assess each other's project, as well as exemplar projects).

All these activities can be generated using authoring functionalities, and it is always possible to modify the created objects.

Moodle supports a lot of different formats for applications (word, pdf, excel, powerpoint, rtf, flash,...), for images (bmp, gif, png, tif,...) and for texts (html, richtext,...). So the resources can be prepared using a large variety of applications.

b)   Course Organization

The administrator may set up several *course categories* (for example, "Science", "Humanities", "Public Health") and insert in each category several courses. This makes it easier for the learners to find their course of interest.

Each course can be organised in three different formats: weekly, topics and social format.

Under the *weekly* format, the course is organised week by week and start/finish dates are clearly defined. Each week consists of several activities. Some activities, like journals, may be organised to be usable only during a defined period (they are labelled as "open windows" activities). After the expiration date they aren't available any more.

In *topics* format, the course is very similar to a weekly course, except that each "week" is called a topic, and is not restricted by a time limit.

In *social* format, the course is oriented around one main forum, the *Social forum*, which appears listed on the main page. This format fits well in informal situations such as a departmental notice board.

Thanks to its modularity, Moodle allows to export/import whole courses through different instances of the system. This allows for a preferential exchange of course material among sites built with Moodle.

c)   Verifying the learning: quiz, choice, assignment and journal

Moodle features different means to evaluate learners' capability and level of learning: for instance there are *quiz, choice, assignment* and *journal*.

A *quiz* allows the teacher to design and set quiz tests, consisting of multiple choice, true-false, and short answer questions. These questions are kept in a categorised database, and can be re-used within the same course and even in different courses. Quizzes can be set to allow the student multiple attempts. Each attempt is automatically marked, and when an answer is given the teacher can choose whether the system is to give a feedback or to show the correct answer. This module includes grading facilities. The teacher can specify a time range for the quiz to be accessible: before the opening time and after the closing time the quiz is unavailable. By default, quizzes do not have a time limit, which allows students as much time as they need to complete the quiz; but it is also possible to set a time limit: in this case a floating time window will be shown with a countdown during the attempt and, when the timer runs out, the quiz will be submitted automatically with whatever answer have been filled in so far. The teacher can also choose if quiz will have shuffle questions or shuffle answer, and how many attempts each student is allowed for the same quiz. If the system is set for multiple attempts, a further option allows the teacher to decide whether each new attempt must start from scratch, viz. if the answers of the previous are to be shown or not. Moreover, when multiple attempts are allowed, there are different ways the teacher can modulate grades to calculate the student's final grade for the quiz: highest grade in any attempt, the average grade of all attempts, the grade gained on the first attempt or the grade gained on the most recent attempt. If no choice is made, the quiz module performs an automatic evaluation of each attempt, giving a grade to each question composing the quiz. The teacher can also choose not to consider that particular quiz, so for it no grade will be calculated. Finally the teacher can specify a password to access the quiz, or can restrict access to particular subnets on the LAN or Internet by specifying a comma-separated list of partial or full IP address numbers (this can be used for instance in order to allow only people in a certain room to access the quiz for an exam).

A choice is a very simple activity: the teacher asks a question and specifies a choice of multiple responses. It can be used in many ways: as a quick poll to stimulate thinking about a topic, as a voting system for the class to decide future developments of the course. It can also be used to gather research consent. For each choice it's possible to define a time window within which participants are allowed to make their choice, it is also possible for the teacher to modulate how the final results are to be available to students (always, only after the student's answer, only after the choice is closed or never); moreover it is possible to set the system to publish anonymous results (without students' name) or full results (showing names and their choice).

An *assignment* allows the teacher to specify a task for students: they are requested to prepare digital content (any format) and submit it by uploading it to the server. Typical assignments are essays, projects, reports and so on. This module includes grading facilities. There are two different types of assignments. An *offline activity* is defined as an assignment that is to be performed outside of Moodle: students can see a description of the assignment, but are not requested to upload files, teachers can grade all students though, and students will see notifications of their grades. An *online activity* allows all students to upload a single file: this may be a Word processor document, or an image, a zipped web site, or anything asked to submit, teachers can grade submitted assignments online. By default, students cannot resubmit assignments once the teacher has graded them, but while creating the assignment it is possible to set the system to allow students to resubmit assignments also after their grading. This may be useful if the teacher wants to encourage students to work better using an iterative process. Obviously, this option is not relevant for offline assignments. Finally, the teacher can choose to grade assignments to evaluate his/her students' attempts, or can choose not to evaluate them.

A *journal* is a very important reflective activity. The teacher asks the student to think about a particular topic, and the students can edit and refine their answer in time. This answer is private and can only be seen by the teacher, who can offer feedback and a grade on each journal entry. It's usually a good idea to have about one Journal activity per week. This activity uses the same evaluation system of an assignment, and the only other choice available for the teacher is to set the availability for the students.

d)   Verifying the use: survey

The survey module provides a number of survey means that have been found to be useful in assessing and stimulating learning in online environments. Teachers can use them to gather data from their students. This is supposedly helpful for teachers, who can thus learn about their own class and think about how to improve their own teaching.

Currently, Moodle only offers specific types of surveys (it is claimed that future versions will enable users to create their own). The available surveys, which have been chosen as being particularly useful for evaluating online learning environments, use a constructivist pedagogy (see also section 5). They are useful to recognise certain trends among participants. At present there are two survey families available:

- COLLES (Constructivist On-Line Learning Environment Survey), comprising an economical 24 statements grouped into six scales, each of which helps addressing key questions about the quality of the on-line learning environment;
- ATTLS (Attitudes Towards Thinking and Learning Survey), to examine the quality of discourse within a collaborative environment, to measure the extent to which a person is a "connected knower" or a "separate knower".

e)   How much knowledge: automatic correction

In Moodle some of the activities allow grades to be set. By default, the results of all grades within the course can be seen in the *Grades* page, accessible from the main course page.

In case the teacher decides not to show grades to students, grades display can be disabled in the Course Settings area. This does not implies that grades cannot be used in individual activities; it just disables the results being displayed to students.

For each valuable activity inserted in a course, Moodle allows to present the learner with a feedback. It is advisable to add proper feedback to each answer in a multiple choice question, or in a quiz. If the system is configured to show feedback, it will be shown to the respondent next to each given answer

For some types of activities, Moodle generates an automatic numerical evaluation of learners' attempts, based on the evaluation assigned to each question composing the test.

For example, by creating a quiz, which is a set of different questions, the teacher can give each question a mark and then, for each question, assign a weight to each possible answer: this weight is expressed by a percentage and can be positive or negative. The sum of the positive weight for all the possible answers to a question must be 100%, so if a student give the correct answer s/he will receive full mark for that question.

f) Managing learners' groups

Moodle allows to create groups of users. It's possible to create many groups (a group can have no members); and each user can be member of only one group.

For each group it's possible to edit a *profile* (name, description, and a picture), add a new member from the list of users enrolled to the course that are not already members of another group, or remove a member (who thus go back to the list of enrolled users that aren't in a group.

g) Who sees what: access rights to course and contents

Moodle has features for basic authentication and a system to manage privileges and rights, dividing users in different categories (see also section 2).

Moreover it allows to assign a password to enter a course, so only users knowing the password can visualize the course. The password can be given via e-mail, publishing it on the site, in the feedback of a test or, if the course is also face-to-face, in the classroom.

h) For course creators: authoring tools

To create a course Moodle offers a template, a form, quite rich in details. In this form, for example, the author can insert the name and a description of the course, and set the course options (see also section 1.b). This form is pre-filled so, when the author simply submits the pre-set data, Moodle automatically creates a course with default options (only the name of the course is necessary). Nevertheless the system does not allow authors to define their own course template, so for some special needs the available options can be restrictive. Moodle allows also to assign a password to a course (see section 1.h).

After the course has been created, it can be edited it to insert contents. To perform this operation, the author must turn editing mode on; this is can be annoying (see also section 4) because every time the author wants to modify an object in one of his/her courses s/he has to to turn editing mode on to be enabled to work.

A similar template is presented to the author of a course to create all the activities or the resources included in it. As said in previous sections (see 1.c and 1.e) each activity is created filling in several fields, thus setting all the activity features.

Finally, once created, all this activities are presented to the author together with a simple sequence of clickable icons showing all the operations that can be performed on that object: they can be moved within the course to change the order of creation of the objects, can be updated to modify the statements decided at the beginning, can be deleted or can simply become invisible to course's users or, with the last icon, can be assigned to a particular group of users.

i) Different users, different learning paths

Using Moodle, is quite difficult to define alternative learning paths. There are some instruments that are included in Moodle that allow to manage accesses to a resource or to a course, but there's not a simple way to create distinct learning paths, for the learner to choose or for the teacher to set, according to students' previous knowledge.

j) Time for everything: calendar

Moodle supports learning plans using a calendar. This tool allows teachers to enter several types of events: global events, group events, course events or user events. The author can insert a new event in the calendar through the same template, specifying event type, name and description, date (with time), duration (in days, minutes or nothing if the event doesn't have a duration) and repeats (specifying, if there are weekly repeats, and how many they are).

Events in the calendar are shown as links within their day cell, and may have different colours to show what type they are.

k)   Hidden-in preparation material

As said in previous sections, Moodle allows to create content within the system as well as to import them on the platform. In both cases it is possible to make contents invisible until they are ready: the process is very simple. Beside each resource there is a little icon (see also section 1.i), of an eye; that can be set as open thus making the corresponding resource visible and accessible to all the users, otherwise (if it is closed) the resource is not visible: the author of the course sees the corresponding resource link in grey and all the other users don't see the link at all. The eye's state can be changed simply clicking on the icon.

The eye is available whenever it can be useful to hide an object within the site; so, it's possible to hide a simple resource within a course, or a whole course, and also a whole category.

l)   Synchronous communication: chat, videoconferencing, etc

In Moodle synchronous communication is supported by a chat system: this activity can be inserted in each course or in the main area of the system.

The chat module allows participants to have a real-time synchronous discussion via the web. This is a useful way to get a better understanding of each other and to learn about different points of view about the topic being discussed. The mode of using a chat room is quite different from the asynchronous forums. The chat module contains a number of features for managing and reviewing chat discussions. While creating a chat, the template allows to choose name and description, to display next chat time if desired, to choose for how long past sessions are to be saved and, finally, if all the users can view past sessions. Moreover, there are some features that make chatting a little nicer: any smiley faces (emoticons) that you can type elsewhere in Moodle can also be typed in here and they will be displayed correctly, Internet addresses will be turned into links automatically, users can start a line with "/me" or ":" to emote (for example, if user's name is Kim and he types ":laughs!" or "/me laughs!" then everyone will see "Kim laughs!"), users can send a sound to others by hitting the "beep" link next to their name (a useful shortcut to beep all the people in the chat at once is to type "beep all") and then if an user knows some HTML code, s/he can use it in his text to do things like insert images, play sounds or create different coloured and sized text.

Moodle doesn't offer other types of synchronous communication, for example it is not possible to set-up videoconferences.

m)   Asynchronous communication: forum, topic forum, newsgroups

In Moodle asynchronous communication is supported by forums and e-mail.

A forum can be inserted in each course or in the main course of the system: it can be structured in different ways ("single simple discussion", "standard forum for general use" or "each person posts one discussion") and can include peer rating of each posting. While creating a forum the author can choose if students are to be free to start new threads and to post replies within those threads, or to restrict this ability (allowing reply but not start new discussions) or to disable it at all . By subscribing to a forum, participants will receive copies of each new posting in their email. People can usually choose whether or not to subscribe to each forum. However, if a teacher forces subscription on a particular forum then this choice is not available and everyone in the class will get email copies. This is especially useful in the News forum and at the beginning of a course. The postings can be viewed in a variety of formats, and can include attachments.

In Moodle each user has his/her profile where s/he can insert the e-mail address: profile options allow user to choose whether his/her address will be visible and to which kind of users. The profile is linked wherever an user has left a track of his passage, so it's quite simple for users to communicate each other using their e-mail.

n)   Other

II. ROLES

This system can be set to allow different types of users (administrator, creator of course, teacher of course, learner), and each person requires only one account for the whole server (each account can have different access roles).

a)   learners

A simple user account is only allowed to login courses (possibly not all), and to play an active role in the learning process of the whole class (see also section 5).

In Moodle students are encouraged to build an online profile including photos, description. and email addresses; this can be protected from display if required; every time a learner enrols in a new course, this course is added to his/her course list, visible by the teacher, with other actions of the learner (see also section 2.b). Nevertheless Moodle doesn't support learners' portfolios, viz. does not allow to keep a record of all successful learning activities for each learner.

b)   authors

A course creator account is allowed to create courses and teach in them. Moodle is customizable and flexible: authors can change the platform theme, its time zone, and also the structure of the page (adding or removing its constructing blocks), moreover they can define their own ranking for grading forums, assignments and journals; an author has full control over all the course settings, this includes access restrictions for other teachers. Authors can enrol and un-enrol students manually if desired (otherwise they are automatically un-enrolled after a certain period of inactivity, set by the admin). For each student, activity reports with graphs and details about each module (last access, number of read) are also available, as well as a detailed "history" of each students involvement, including postings, journal entries etc on one separate page.

c)   tutors

Courses authors may have editing privileges removed so that they can't modify the course. This role is only allowed to teach his/her course: s/he can see all that can see an author, can add new resources, but can't modify the course settings.

d)   Administrators

In Moodle the whole site is managed by an admin user, defined during setup: this account controls the creation of courses and creates authors and tutors.

One of the first goals of Moodle's creators was to reduce admin involvement to a minimum, while retaining high security; so the administrator's work ends when s/he sets all the configuration variables of the site (language, maximum size for uploaded files in the whole site, time zone, headings, theme) and defines the identities of authors and teachers within the system.

e)   system managers

In Moodle there's not a system manager because the whole site can be administrated by the admin user or by the authors.

f)   other

Moodle can allow unregistered users to login, using a guest account. This account will have strict limitations and no privileges, of course, but it can be useful to enable people to browse the system without enrolling.

III. RELATIONSHIPS WITH STANDARDS

(To be filled)

IV. USABILITY AND LEARNABILITY

a)   Usability test: existence and results

We are currently looking for documentation about Moodle usability tests. So far we have found documentation about how important this topic is considered by the Moodlers' community.

b) Learnability test

As for now, we did not find any available documentation about learnability tests conducted on Moodle.

c) Accessibility

Moodle contains a lot of help files, linked in every page where help can be useful, mostly in pages where authors create new objects (for example, in the form for course creation, next to every field there is a little icon linking to a help file), and at the end of each help page there's a link to an index of all the help files.

This system is easy to use both for authors, tutors and other users: little expertise is required and all its structure is intuitive, because within it each operation is performed always in the same way. For example, where is possible, to make an object (an activity, a whole course or a whole category) not visible to simple users, Moodle uses a little icon, viz. an eye, which is open when the object is visible and close when it is not: this condition can be modified clicking on the icon, so when the author sees an eye close to a link to a resource s/he knows that can change its condition of visibility. Moreover, Moodle is very easy to navigate because in every page there are both an *header* - containing the whole navigation path to the previous page or simply to a site map of the part the user is navigating in - , and a *footer* containing a link to the site (or to the course) main page. In the footer there's always also a link to the profile of the current user , and a link to logout anytime.

Finally, among the headline features for the next release of Moodle there are accessibility improvements,, aiming for compliance with WAI (W3C), SENDA (UK) and Section 508 (US) criteria.

d) Multi-language support

Moodle is a multi-language system. Presently it supports 60 different languages. Plug-in language packs allow full localisation to supported languages. These packs can be edited using a built-in web-based editor.

V. PEDAGOGICAL METHODS

Moodle supports different models of learning, not only distance learning scenarios but also scenarios in which distance learning and classroom learning are blended. In both cases the core theory behind Moodle is the same: social constructionism, so Moodle can be defined as a learning-centered system, whereas others are more tool-centered. Social constructionism is based on the idea that people learn best when they are engaged in a social process (a group's work) aimed to the construction of knowledge. This is achieved by constructing an artefact for others. From this point of view, learning is a process of negotiating meaningful information in a culture of shared artefacts and symbols. We need to test new learnings against our old beliefs and incorporate it into our existing knowledge structures. Moodle makes use of this theory, for example, in its interface. While tool-centric LMS systems give a list of tools as an interface, Moodle builds the tools into an interface that makes the learning task central. Additionally, while other LMS support a content model that encourages authors to upload lots of static content, Moodle focuses on tools for discussion and means of sharing artefacts. So the focus is not on delivering information, it's rather on sharing ideas and engaging in the construction of knowledge.

VI. INSTALLATION

Moodle offers a guide that explains how to install Moodle for the first time, and, in case of trouble, it's possible to seek help from a rich list of FAQ. The requirements for Moodle are a web server software (Apache or any other web server that support PHP, such as IIS on Windows platforms), PHP scripting language (version 4.1.0 or later) and a working database server (MySQL or PostgreSQL are completely supported and recommended). Although Moodle is a growing system with many features, its installation is very simple. To run the installer script, you have only to access Moodle main URL using a web browser (or access http://sitename/install.php): Moodle will detect that a configuration is necessary and will lead the user-installer through some screens to help him to create a new configuration file called config.php. At the end of this process, Moodle will try and write the file into the right location (if desired the user can press a button to download it from installer and then upload config.php into the main Moodle directory on the

server). Along the way, the installer will test server environment and give suggestions about how to fix any problems.

VII. MAINTAINABILITY

Moodle gathers a very large and active community both of users and of developers. This community has produced, in collaboration with the Moodle creator, Martin Dougiamas, a multi-language and quite detailed documentation for developers, where mechanisms for the extension of features and some basic rules to preserve a good quality of the source code of the project are documented. Furthermore Moodle community offers a very good assistance through the official site of the platform (www.moodle.org) where many forums about different topics are available (development, bugs, translating) in almost all the languages of Moodle. Moreover Moodle community has been critical for the success of the system. With so many global users, there is always someone who can answer a question or give some advice. At the same time, the Moodle developers and users work together to ensure quality, add new modules and features, and suggest new ideas for development. The development of Moodle is made easy thanks to its high modularity, so that a developer can construct a new functionality for the system without having to trouble about the whole structure, being thus enabled to keep/he focused on his/her work. Dougiamas and his core team are responsible for deciding what features are mature enough for official releases and for elaborating new strategies. Because users are free to experiment, many people use and test new features, acting as a large quality control department.

VIII. REQUIRED HW or SW

Moodle doesn't need any additional products for the authoring options, nor for the learning management functions.

IX. LICENSING: OPEN SOURCE OR PROPRIETARY

Moodle is Open Source software, and can be redistributed and/or modified under the terms of the GNU General Public License.

X. TOTAL COST OF OWNERSHIP

Being open-source, Moodle costs is zero. Speaking about the time needed to make this LMS ready to use, it's important to say that the procedure of installation is very simple (see also section 6) and it's possible to change most of the configuration variables also after the installation. Making a rough estimate of the training time on the Learning Management System for different type of users, we think an administrator needs about 30 hours to have a good knowledge of the possibility of his system, and so does an author. For tutors and users all the training needed is simply to use the system, because they all they need is to navigate: what they can do while surfing within the platform is shown by links. Finally, to modify Moodle at the program level a good knowledge of PHP, HTML and MySQL (or PostgreSQL) is needed, for example to modify layout or functionality of the site, but this knowledge can be obtained using Moodle itself, too.

XI. USAGE

Moodle is an open-source learning management system used by universities, community colleges, K-12 schools, businesses, and even individual instructors to add Web technology to their courses. Moodle has a large and diverse user community with over 50,000 users registered on the official website (www.moodle.org), speaking 60 languages in 115 countries; currently it is used by more than 1800 educational organizations around the world to deliver online courses and to supplement traditional face-to-face courses.

At this moment there are 3332 sites from 114 countries who have registered on the official website, and all these sites are linked at the website www.moodle.org/sites/.

XII. SUPPORT QUALITY

Moodle is an open source LMS created by Martin Dougiamas, a computer scientist and educator, who realized that a system built always keeping in mind the educational process, rather than the engineering process, would be infinitely better than what he used to see in his everyday work. He set his graduate degrees in education and computer science at work and started developing Moodle as an alternative. He now works on Moodle full time. Currently working with him, a community of dedicated open source developers from around the world is committed to a collaborative effort to make Moodle the best LMS available. Moodle community is continuously adding new features and modules according to the requirements of other "moodlers" (see also section 7). Then Dougiamas and his core team release a new version of Moodle when developers have created enough innovations to justify a new step.

For any problem is possible to contact, via forum, developers, installer, administrator or simply users of Moodle, and while looking for help is often possible to exchange useful information about other Moodle aspects.

11. LMS: PaKMaS

PaKMaS stands for Passauer Knowledge Management System.

PaKMaS's purpose is to produce and use multimedia knowledge modules, with a flexible re-use of knowledge modules of arbitrary dimensions. Its flexibility allows to operate on knowledge modules of any dimension, as well as on heterogeneous information sources and on different kinds of topics for the exchange of knowledge.

However, none of the available information are more recent than 2003, so we think the project has not been maintained by the developer team.

I. TEACHING FEATURES

a) Contents

Within PaKMaS, teaching material can be presented hierarchically using the *book metaphor* where chapters and sections act as "structure modules". It can also be presented sequentially starting from a first "basic module" using a *guided tour metaphor*. Both of these metaphors operate according to behaviouristic strategies (see also section 5) viz. the route through the given learning material is explicitly plotted by the author. If an author, for example, wants to create a section of a book-like course or a guided tour, he can build on new "basic modules" using "course units" which can be extended e.g. by question elements. He also can query the systems repository for suitable basic modules, i.e. for modules with strategy attribute set to "beh" (behaviouristic). The system can also suggest a linear ordering or check the consistency of a sequence using the modules "prerequisite" and "objectives" attributes. As in guided tours, sections or chapters act as "structure modules" which force a linear ordering to the children modules, and a teachware management system can offer navigation facilities with which the learner can sequentially walk step-by-step through the given "basic modules".

PaKMaS is flexible enough to support constructivist strategies too (see also section 5), as it comes with "structure modules" like indexes, glossaries, time lines and unordered collections of learning material. They all allow direct, non-hierarchical access. Special "basic modules" representing "coaching units" give some hints to the learner about the appropriate usage of the available material. PaKMaS supports cognitive apprenticeship both by providing these modules and also offering easy-to-use communication facilities with human coaches (see also section 1.l and 1.m). In addition to a simple tool allowing learners to add individual notes to learning material, PaKMaS provides cognitive tools which are of paramount importance for the application of constructivist strategies. Using these tools, learners can structure and conform their learning environment conforming to this meta-model. For example, they can use existing "structure modules" to form

their own collections and to create their own "guided tours". To build their own learning material, learners can use existing content objects like definitions or examples. Moreover, learners can create new types of structure modules like "slide show" or new types of conceptual units like slides. Thereby learners can develop their learning environment, according to the meta-model.

In practice, in PaKMaS a module can be a basic structure or can be a GuidedTour, a Glossary, a Collection or an Index. Each module comes with one or more resources, viz. ConceptualUnit or ContentObject. A ConceptualUnit can be a CourseUnit, an EvaluationUnit, a TrainingUnit, an AnnotationUnit, or an ExampleUnit. A ContentObject can be a Paragraph, a Table, a List, a Multimedia, a Motivation, a Definition, a Remark, an Example, an Exercise, or a Question.

b) Course Organization

At this moment no information is available about this point.

c) Verifying the learning

At this moment no information is available about this point.

d) Verifying the use: survey

At this moment no information is available about this point.

e) How much knowledge: automatic correction

At this moment no information is available about this point.

f) Organize a class: managing learners group

At this moment no information is available about this point.

g) Who sees what: access rights to course and contents

At this moment no information is available about this point.

h) Course creation: authoring tools

At this moment no information is available about this point.

i) Different users, different learning path

The system allows a full adaptation and individualization of content, structure and navigation.

Within the framework of current teaching and learning theories, authors create learning material as a net of knowledge modules. Learners are thus offered modules which can be adapted to suit the individual needs of each student. The modules in PaKMaS can be seen as sites on a map of a "knowledge world", connected through different roads. The author can offer several different paths (so-called "guided tours") through this knowledge world, while the students can modify and extend these paths or even create paths of their own. For a first reading of a topic or its fast revision for an exam, one will rather use the "highway", whereas courses going into details may require travelling on narrower, slower streets to visit as many places as possible.

A knowledge pool managed by PaKMaS can easily be managed and updated by authors (professors, students, staff) using repository technologies; as all knowledge modules and their connections can be found in one place and are no longer spread over many different scripts, documents, folders or file systems. Also, users can handle material more flexibly than in usual documents, teaching systems or CBT systems and they are invited to individual tours through the topic.

j) Time for everything: calendar

At this moment no information is available about this point.

k) Hidden/in preparation material

At this moment no information is available about this point.

l) Synchronous communication: chat, video conferencing, etc

PaKMaS supports several communication tools. Communication via chat with other users of the system, with coaches or with other learners in this domain of application allows the creation of knowledge communities. Furthermore, the system allows the wrapping of simulations or real life systems into basic modules.

m) Asynchronous communication: forum, topic forum, newsgroup

PaKMaS can foster the creation of knowledge communities by enabling email communication among users as well as posting to appropriate newsgroups managed by the same.

n) Other

II. ROLES

In all the material available on the web, there are no specific information about this aspect of the system. The following information were deduced from others topics.

a) Learners

All PaKMaS users are supported by adaptable presentation of knowledge modules on various levels of granularity; adaptable navigation using conceptual relationships; sophisticated query facilities, both for particular teaching modules (and their interdependencies) and fully or partially specified courses; topic and type oriented filtering of knowledge modules and their contents; and electronic communication with authors and other users.

b) Authors

Authors can use different tools in the XML-based "Learning Material Markup Language" (LMML) which enable them to: create new material, find knowledge modules with similar metadata, integrate existing material like LaTeX or MS-Word documents (either by converting to LMML or by wrapping), annotate knowledge modules with metadata, and to reuse and replace knowledge modules.

c) Tutors

No documentation has been found about this role.

d) Administrators

No documentation has been found about this role.

e) System managers

No documentation has been found about this role.

f) Other

III. RELATIONSHIP WITH STANDARD

At this moment no information is available about this point.

IV. USABILITY AND LEARNABILITY

a) Usability test: existence and results

No documentation has been found about usability test for PaKMaS.

b) Learnability test

No documentation has been found about learnability test for PaKMaS.

c) Accessibility

No documentation has been found about accessibility for PaKMaS.

d) Multi-language support

At this moment no information is available about this point.

V. PEDAGOGICAL METHODS

A full description of the system pedagogical approach is discussed in a publication titled "A Teachware Management Framework for Multiple Teaching Strategies", written by C.Süß, R.Kammerl, and B.Freitag, and presented at Proceedings of ED-MEDIA 2000, World Conference on Educational Multimedia, Hypermedia & Telecommunications, in Montreal (Quebec) (http://www.im.uni-passau.de/publikationen/SKF00/paper.pdf). Here are some interesting extracts.

"(...) Learning processes depend on the individual properties of learners like age, level of experience, knowledge and interest which usually span a wide range. Thus, there is no best teaching strategy that applies to every learning situation. In fact, even for a single learner the best strategy may change over time. (...) Apparently, teachware management systems or computer managed instruction (CMI) systems should support multiple teaching strategies. Moreover, they should provide an appropriate data model, suitable abstraction mechanisms, and tools not only to authors but also to learners. It is obvious, that a flexible multi-strategy support and powerful application interface both require system properties that are beyond most traditional hypermedia systems and common techniques and tools for computer based teaching. (...) Due to the heterogeneity of the goals aimed at, practical arrangements for teaching and learning are influenced by various didactical and learn-theoretical considerations. Consequently, teachware appreciating this variety of approaches and thus satisfying the requirements of practical teaching has to account for and give support to different learn-theoretical foundations. The latter can be classified into behaviouristic, cognitivist and constructivist theories. (...) The behaviourist approach is based on the assumption that behaviours can be steered by external stimuli. (...) Teaching goals are to be achieved by presenting selected information and questions or exercises as stimuli, that are expected to cause a certain behaviour, to the learner. (...) The behaviouristic learning theory is chiefly materialized as a learning strategy by guided tours or linear presentations using multiple choice questionnaires or drill & practice modules. The basic principle of these approaches is the presentation of information followed by a collection of questions and according feed-back to the learner. (...)

Cognitivism which relates to the theories of Piaget and Bruner considers the learner as an individual that processes external stimuli actively and independently and that can therefore not be steered by external impulses. (...) It is assumed that the learner perceives, interprets and processes impressions in a selective way according to his or her level of experience and personal evolution. (...) The cognitivistic learning theory together with behaviouristic ideas has influenced the Instructional Design. However, essential aspects received attention only in connection with constructivist approaches. (...)

From the perspective of constructivism the potential of teaching media to steer learning processes is considered even lower than from the cognitivistic point of view. The active processing of external impulses is rated even higher. The learner is assumed to construct reality individually based on his or her subjective experience structures. (...) Constructivism considers knowledge not as a reflection of an external reality but rather as a function of perceptual processes. From this point of view learning material is to consist just of collections of information units and tools that form the input of the learning processes built by the individual. (...)

We are convinced that all three approaches sketched above should be taken into consideration. However, which learning theory is best suited to support learning processes depends on the teaching goals and the individual properties of the learner, i.e. level of experience and education, age, learning habits. (...)Consequently, a teachware management system should have the capability to support different teaching strategies. These can be materialized in the form of behaviouristic components like guided tours with multiple-choice questions. Likewise, a constructivist strategy could be provided by a collection of

informative components and tools that allow the learner to construct or change the learning environment, in a certain way being an author him- or herself. (...)".

VI. INSTALLATION

We didn't find the download site for the PaKMaS software or any documentation the installation procedure for PaKMaS.

VII. MAINTAINABILITY

PaKMaS was developed by the Passau University in Bayern, at the Management Information Department. None of the available information are more recent than 2003, so we think the project has not been maintained by the developer team.

VIII. REQUIRED HW or SW

PaKMaS is platform independent, with intranet/internet-based author and user interface.
To use PaKMaS there are some minimum hardware and software requests: a Pentium III 600MHz (800 MHz recommended), RAM of 128 MB (256 MB recommended), 20 MB of free Hard Disk, operative systems Win98 or Win2000 (also Windows ME, but there are some bugs), and Internet Explorer 5.5.
The PaKMaS distribution encloses among other things software from the Apache XML FOP project and the Apache Jakarta project, as well as the Sun Java 2 to run time Environment Plugin.

IX. LICENSING: OPEN SOURCE or PROPRIETARY

At this moment no information is available about this point.

X. TOTAL COST OF OWNERSHIP

At this moment no information is available about this point.

XI. USAGE

At this moment no information is available about this point.

XII. SUPPORT QUALITY

At   this   moment   no   information   is   available   about   this   point.

12. LMS: Reload 2.1.1

Reload stands for Reusable Learning Object Authoring and Delivery.
Reload Metadata and Content Package Editor is not a LMS, so the structure of its description in this document was modified, but the main points have been maintained.
The key aim of the Reload project is the implementation of a reference-standard Content Package and Metadata Editor. Reload focuses on developing a set of open source tools for learning objects packaging and

delivery The Reload Editor enables users to organise, aggregate and package learning objects in standard IMS and SCORM content packages tagged with Metadata (in various subsets) and vocabularies.

I.   TEACHING FEATURES

Reload Editor is a content package and metadata editor, that allows both for the creation and the modification of educational material, and for their sharing and adaptation. With Reload, users can handle their own electronic content (web pages, images, flash animations, Java applets etc.) and make it ready for storage in content repositories by adding packaging and descriptions. Creating a content package is a good way to share educational materials, as all the files needed are conveniently stored within a single zip archive and the standard structure created can be understood both by repository software and by Managed or Virtual Learning Environments (MLE or VLE). Moreover, a package usually comes with metadata describing its contents, and also providing information about technical requirements, usage restrictions and third party annotations. Adding metadata to describe a resource allows it to be sorted, indexed and retrieved from repositories such as those established by JISC (Joint Information Services Committee) (JORUM project) and other bodies. As said, content packages are used to store educational content, but of course educational content can mean anything – from a single image which may be used for illustration to a full web-based course with video and sound clips, Acrobat files, Java simulations, Flash animations, Word documents and assessment tests. Metadata added to the content package allows other users to assess its suitability for their needs.

Content packaging is easy. Firstly, all the content needed to make up a piece of educational material are collected together. Then, a special file that collects information about the resource, the *manifest* (which must be called imsmanifest.xml) is created. The manifest contains the following information: metadata (stored for the resource as a whole, or for individual files), resources (what files are needed and where they reside in the content package - this section also records dependencies, for instance, it may specify for a certain web page its requirements of style sheet and images in order to display properly), organizations (describing how the individual resources inter-relate). For convenience, content packages are usually stored as zip archives. In this case, the important imsmanifest.xml file must be placed in the base directory of the zip file so that it can be easily found, should any software need it. Creating the manifest file involves creating a file in the XML format. Although similar to HTML, XML is much more complex and therefore less easy to work with. By providing a visual user interface for the manifest creation task, Reload Editor provides a very simple way of creating content packages and metadata just by dragging files around a file browser and typing information into fields in forms. Reload Editor also allows users to easily preview their content packages to check that they behave as expected. Reload Editor comes with the following functions: packaging content created by other tools, repurposing existing content through reassortment and reorganization, preparing content for storage in repositories (such as JORUM), and delivery of content to end users using the Save Content Package Preview facility.

Users working with Content Packages can fully appreciate the facilities offered by Reload Editor. A Content Package is a means of transferring a set of resources from one location to another whilst retaining their structure and inter-relationship. When creating the Content Package users create a space in which all local files are kept, a base folder. As users build the Content Package, all the files they use will be stored under this working folder, their original location being anywhere on the local machine (external resources can be referenced by hyperlink).

In order to show the Reload capabilities, we will describe how a Content Package can be created. Before beginning the work with Reload Editor, a user will create a folder on his/her own computer, where all the files needed for the Content Package (documents, images, slides, web pages,...) will be placed, and another folder at the same level to be used as a working folder for the Content Package itself. When an user launches Reload Editor, by default, it opens up with no file open. Then, to start the actual work, a user needs to set up a Workspace, which is done by creating a new IMS Content Package and then selecting those previously created as working folder. The Reload Workspace basically consist of three panes: a resource pane, a manifest pane, and an attribute pane. The resource pane shows a tree structure of files and folders which are to be inserted in the package. The manifest pane is the key area: this space represents the structure of the Content package (with its Manifest containing Metadata, Organizations and Resources). The attribute pane includes a section with context sensitive information on the currently selected element as well as a table of values and boxes for editing attributes. Now the user can also change the default IMS Metadata and Content

Package schemas. When an user saves his/her choices a new window appears, showing the Workspace. Each window represents a Content package and multiple windows may be open at one time. If users wish to view the Metadata associated with a particular Content Package, then a new metadata window is launched in addition to the Content Package window. The Reload Editor comes with two toolbars: the Main Toolbar, with general options, and the Resource Pool Toolbar giving access to actions specific to the Resource Pool. The Main Toolbar allows to: create a new Content Package (SCORM or IMS) or Metadata Record, open an existing Content Package or Metadata Record, save the current Content Package or Metadata Record, undo/redo the last action, cut, copy and paste the current selection, move up and down the selected file within the Content Package, edit Metadata for the selected manifest, edit SCORM properties for a given item, create a .zip package, view an individual file (via browser), and preview the Content Package. The Resource Pool Toolbar allows to: import to Resource Pool, create new folder, delete from Resource Pool, and refresh the file list for the Resource Pool.

While creating the Content Package, the Reload Editor automatically creates the main data file, imsmanifest.xml, i.e. the manifest for the current package, where all the information will be stored (the name is mandatory, and this file must appear in the root of any valid Content Package). At the moment the Content Package has no content at all; but before the user adds any content, he should work with Metadata, which can also be described as placeholders, and with Organizations. First of all, in the metadata section of the manifest, user has to specify a schema , such as IMS Content, and a version, such as v.1.2.2. In this way, the format of any metadata s/he will add is defined as conforming to the IMS Metadata specification v.1.2.2. Besides, the real core of a Content Package is not its content, but one or more Organizations which allow these contents to work properly. By default the package created has no organizations, so one has to add one in the Organization section, choosing a title for it. Then, the user can import some content. As all files needed for the Content Package are in one folder (created by the user before the launch of Reload Editor) s/he has just to select all files and folder in this directory and import them. The files thus imported now appear in the tree view of the Reload Workspace. Basically, he creates the structure for the Content Package by adding items to an organization. Adding items to the organization can be carried out by using menus, but it is far easier to just "drag and drop" content from the tree pane to the manifest pane. Firstly, user selects the resource s/he wants to add from the resource list in the tree pane, then drags the file to the existing organization in the manifest view: as user adds content to the organization, the file (and its dependents) are added to the resource section in the manifest. User doesn't need to worry about adding the files in order, because if he wants to rearrange them, he can always use the move up/down commands to reorganize individual items. If user's content is kept within a single folder, then s/he can drag the whole folder across to the organizations section from the tree pane and the whole process of adding organizations and itemizing is carried out automatically (a little tidying up may be needed).

Once all contents has been added to the main organization, the Content Package is ready for preview in any web browser. In the browser, a three-framed page will load: the banner frame provides some navigation tools, the left frame will mirror the previously defined structure of the Content Package, and the right pane will usually contain the first item of the default organization. In the banner frame, the name of the currently loaded item is displayed. To the right hand side of this banner, three icons allows the user to show or hide the package structure and to navigate through the content sequentially. In the left pane, the active organization is shown, with its title in black and the content in blue: clicking on any of the blue links will load the corresponding content into the main frame. An extra drop down menu will appear if there are multiple organization elements. To tidy up the package before final export, an user can, for instance, rename the organization with a name more pertinent with its content, since it will appear in the browser as the title of the content group, or rearrange the content to follow a different structure, or rename the items so that their names are more meaningful than a pure numeration. After changes have been made, to have a new preview the user has to close and re-launch the browser, to flush the old preview files from its cache.

If a more complicated organization of content is needed, the system allows the use of multiple organizations: these can specify alternative (parallel) structures for the same material, or can provide alternative content for the same learning outcome, to support different learning paths. Adding new organizations is simple, just like building the first, and also populating them follows the same process. When there are multiple organization elements, a default should be specified: if no default is specified then the first organization in the manifest is assumed to be the default. As said before, when there are multiple organization in a content package, the content preview will display a drop-down menu in the left navigation frame.

Reload Editor allows user to save not just the Content Package, but a copy of the resource he has created for preview, ready for presentation in a Web Browser. Usually, Content Packages are imported into software

(LMS or repositories) which interpret information held within the manifest and create a navigational structure for the content. Sometimes, user may wish to utilize the content s/he has organized without an LMS or repository, perhaps as a simple web site. For this purpose, it is useful to have the navigational structure generated by Reload for Web preview included. It is important to stress that the package preview is not in itself a Content Package, as it doesn't include the necessary imsmanifest.xml. In fact it is no more Content Package than any collection of web pages. A preview of the contents is very useful however and it is possible to store the preview within the package itself. By including extra preview files, the Content Package can be quickly previewed by a third party who does not have access to a package viewer.

The eventual destination of a Content Package will probably be within a LMS but it is also likely that it will be placed in a repository to facilitate re-discovery and re-use. Adding metadata to the Content Package will create information that can be searched by users of the database. When user initially set up the Content Package, he decided the standard schema to conform to, but didn't actually add any metadata. It's possible to do this by clicking on the Metadata section in the Manifest view. A new window appears, showing various metadata fields: there are two views, form view and tree view. Form view lets user adds metadata into specific fields, while tree view allows to see the entire metadata record and to create more sophisticated metadata. The form view is really just a simpler representation of the full metadata record, so it is possible to add information in the tree view which is not shown in the form view. For easiness, one can simply go through the form adding valid (i.e. compliant with the chosen standard) metadata to it. But if user wishes to add more then one entry for a given element, he needs to use the tree view, rather than the form view.

Although primarily a Content Package editor, the Reload Editor incorporates a fully fledged metadata editor and can be used to create and view metadata files, which are independent from content. To create a new metadata file, the user clicks the corresponding menu item, chooses the schema version to be used and the desired application profile. S/he now can add metadata to the new blank file as before, either using the form view or tree view (or a combination) until all necessary entries have been completed, and then save.

II. ROLES

Since Reload Editor isn't a LMS but a Metadata and Content Packaging editor, the user role concept doesn't exist within it (i.e. the only role available is the course writer/teacher one).

III. RELATIONSHIP WITH STANDARD

By default, Reload supports the IMS Metadata v1.2.2 and Content Packaging v1.1.3 specifications. However, in practice, individual organizations will finely adjust these specifications to meet their own requirements (either translating them into another language, or by customizing the way they represent the specifications).

The Reload Editor supports the editing of SCORM 1.2 packages, too. SCORM builds heavily on the IMS Content Package specification, and so they are very similar, but there are a few important differences. First of all, SCORM packages can be only either resource packages or content aggregation packages. Secondly, metadata can be kept in an external file and referenced, rather than being kept within the Content Package as a part of the main manifest. Finally SCORM packages come with five extra elements which provide information on how the package behaves: prerequisites, max time allowed, time limit action, data from LMS, and mastery score. *Prerequisites* define any other item which the learner must have completed satisfactorily before they are allowed to access the current item, and can be used to define multiple routes through learning materials. *Max time allowed* expresses in seconds how much time LMS allows an item to be accessed for: for instance user may wish to restrict access to a simple assessment, forcing all learners to complete it within 15 minutes. *Time limit action* is the action specified if the "max time allowed" is exceeded: the user is provided with a choice of four built-in actions (exit without message, exit with message, continue without message and continue with message). The *Data from LMS* allows initialisation data to be sent from the LMS to the item on launch. *Mastery score* is a normalised score (between 0 e 100) which has to be achieved to indicate satisfactory completion of an item.

IV. USABILITY AND LEARNABILITY

a) Usability test: existence and results

No documentation has been found about usability test for Reload Metadata and Content Packaging Editor.

b) Learnability test

No documentation has been found about learnability test for Reload Metadata and Content Packaging Editor.

c) Accessibility

No documentation has been found about accessibility in Reload Metadata and Content Packaging Editor.

d) Multi-language support

Reload provides Brazilian Portuguese, Bulgarian, Italian, German, French, Spanish, Catalan and Dutch language pack in addition to the English default version.

In addiction, Reload Editor introduces the option of providing different language versions of Reload for different locales. A translator can store translations of the Reload Program messages, tool-tips and menu items in text helper files stored in the user home directory. System providers will try and provide additional translation helper files for Reload through their web site.

V. PEDAGOGICAL METHODS

The Reload web site has a whole section about its pedagogic context. The text describes how to improve teaching and learning processes using or improving learning objects and lesson plan. This section is available from the "pedagogic context" link in the main menu of the Reload web site.

Here are some extracts from Reload Philosophy section, in the web site: "The degree to which educational content can be reused and re-purposed depends on how it is stored, made available, and delivered to the academic user community. (...) Emerging learning technology interoperability specifications (...) provide pedagogical frameworks which enable significantly more effective use of learning resources. These specifications are therefore relevant to the basis of the X4L Programme, which focuses on pedagogical outcomes. This project focuses on the development of tools that are based on interoperability specifications (...). The primary aims of this project are to facilitate the creation, sharing and reuse of learning objects and services and enhance the range of pedagogical approaches realisable through the use of lesson plans. These aims will be achieved through the production of a suite of software tools for authoring and delivery standard-compliant learning objects incorporating comprehensive user guides and exemplar resources."

VI. INSTALLATION

From the Reload web site, installation packages are available for the Microsoft Windows and Macintosh (OSX only) platforms, with a binary package (.bin) available for Linux. The install file for Windows includes a copy of the Java Runtime Environment (JRE) which is installed under the Reload Editor installation directory. This JRE (version 1.4.2.04) should not interfere with other Java Virtual Machines installed elsewhere on user's system. The Mac installer does not include a JVM, but utilises the system JVM. No JVM is included with the Linux package either. Instructions on Reload web site explain how to obtain and install a JVM for Linux.

To install the Reload software, download the most recent install file from the section Download in the web site. The download file is approximately 20Mb. Once download is complete, simply double click on the Installer file and follow the on-screen instructions. The installation allows user to choose where to install the software: Reload Editor is installed to "C:\Program Files\ReloadTools\Editor\" by default, although user can of course change the location. Once installation is finished user can launch the Reload Editor by clicking on the Start Menu and finding its Program Group.

VII. MAINTAINABILITY

Reload is funded by JISC under X4L Programme (see also section 5) and has been developed at the Bolton Institute and the University of Strathclyde by Phillip Beauvoir and Paul Sharples. It's under active development, but since version 1.1 (11 August 2003), it has been stable and fully functional. Subsequent versions have sought to provide improvements in usability and customisability.

Reload is rapidly becoming the *de facto* suite of tools for viewing, editing and creating IMS and SCORM Content Packages. Its creators are constantly being informed of, and collaborating with, international developments from such institutions as IMS, ADL, Collegis, Penn State University, Utah State University, WebCT, Calgary University, Heriot-Watt University, and other commercial companies. This is in part due to the Open Source nature of the software.

VIII. REQUIRED HW or SW

The Reload Editor is a Java application, and should run on any platform capable of running Java applications: it works in Windows, Mac OSX and Linux.

The system provider allows system requirements pertinent to the Microsoft Windows system, but for Linux and Mac (OSX) an equivalent specification would be required. For the Windows versions, users will need a fairly good PC with a good chunk of RAM. Say, at least an Intel Pentium 3 (or equivalent) processor (700Mhz), with 256Mb of RAM. The PC versions ships with their own local Java Virtual Machine. The preferred Operating System is Windows XP or Windows 2000, although it will run on Windows 98. It will not run on Windows 95 or earlier. Finally users will need a web browser for viewing Content Packages.

For a Macintosh, OS X and Java 1.4.1 are required (Java 1.4.1 is included with the latest version of OS X.)

The Linux versions of the software require that users install at least Java version 1.4.1 prior to installing the software.

Users do not need to install any other software to work with Reload products.

IX. LICENSING: OPEN SOURCE or PROPRIETARY

Reload Metadata and Content Packaging Editor is a free software released under the MIT Open Source License and is available for download from the project web site.

X. TOTAL COST OF OWNERSHIP

Since Reload Editor is a free software, no payment is needed use this product, and because of its simplicity it doesn't require a particular effort to learn it. Besides, the system doesn't need any other software to work.

XI. USAGE

The Reload Metadata and Content Packaging Editor is of significant value to the UK Higher Education community and beyond, since it provides the crucial "missing link" which allows users to author and transfer learning objects, in specification compliant format, between authoring and design tools, local and distributed digital repositories, and Virtual Learning Environments. At present Reload providers have information about practical interoperability of their products, so of this Editor too, with different LMS or LCMS, such as Tecknical Virtual Campus, .LRN, ATutor, Dokeos, Claroline, WebCT, Learnwise, and Moodle, but no other information are available.

XII. SUPPORT QUALITY

The Reload Editor provides a detailed step by step guide, as an Acrobat file, to a first Content Package creation. This guide is made up of ten steps only, each of them coming with images showing the system interface for clarity.

Moreover, the Reload Editor provides a Manual, which is an Acrobat file, with all useful information both for a novice and an expert user, ranging from the installation process, to the whole process of creating a simple Content Package, and from information on the Metadata features, to customization of Reload.

Finally, the Reload Editor provide a quick start guide, again as an Acrobat file, that gives in two pages some essential information about the system, its application and its features.

Reload is a distributed project, with staff in Bolton, Glasgow and North Wales. Because the core team is so distributed, and not all full time on Reload, email is usually the best way to contact it.

Moreover, the project has produced user guides and exemplars for disseminating and integrating the project tools into the JISC community. Also available, user guide and help files, a FAQ section, information for developer, some publicity material, and links to external sources of information.

13. LMS: Sakai project 2.0.0

(add description)

I. TEACHING FEATURES

a)   Contents

Sakai offers a variety of tools for use in course and project sites, including: *announcements* (to inform site participants of current items of interest, changes in the schedule or policies of the class, as well as to answer questions or make general comments, see also below), *assignments* (see section 1.c), *chat rooms* (see section 1.l), *discussions* (see section 1.m), *drop boxes* (see section 1.m), *email archives* (see section 1.m), *gradebooks* (see section 1.e), *workspaces* (private areas for each users, see below), *news* (topical news from online news sources, such as RSS, see below), *presentations* (sequences of slides, see below), *profile* (see section 2), *resources* (areas where publishing any kind of learning material, see below), *schedules* (see section 1.j), *syllabus* (official outline of a course, see below), *tests/quizzes* (see section 1.c) and *web content* (area for other website publication in frame, see below).

Announcements are a useful way to post a notice about an important change in deadlines, meeting times, or meeting locations. Announcements can have multiple attachments, such as local files, resources or URLs. Instructors can choose to have an announcement automatically emailed to all of the site participants by using the "required" or "high" notification setting; this means all site participants will receive the announcement in email as well as seeing it on the site. Announcements list can be arranged by title (alphabetically), by authors last name or chronologically. Instructors can draft and save an announcement before they send it out to site participants; drafts will be shown with a red "Draft" marker in their own list of announcements.

In Sakai each user has an individual worksite, called MyWorkspace: when users log in, they will automatically open this area. In MyWorkspace, users can perform various tasks, including: posting files in their own private resource area, viewing an integrated schedule for all sites in which they participate, viewing announcements from all their sites, creating their own private schedule items, viewing a list of all sites they belong to, revising worksites they own by adding users or changing tools, choosing how they would like to be notified of new or changed items on sites they belong to, adding themselves to publicly joinable sites. Moreover, in MyWorkspace, users can print a PDF file of all their scheduled events for the day, week, or month, and they can also save drafts or backup copies of work in progress.

Within MyWorkspace, Sakai provides the Membership tool where you can join and leave available worksites. On the Membership tool page, users will see a list of all the worksites to which they belong: this includes sites they have joined, and sites that they have been added to as participants by the creator of the site. They'll also see a list of all the sites that they are able to join, but haven't yet. By default, users will receive email notification about announcements, email, and resources that are posted to worksites to which they belong,

and they can change their preferences so that they receive this notification as a daily digest, or they can block these messages entirely.

An RSS feed is a specially formatted list of news articles: if a user has the necessary permissions, he can set up a worksite to display news. User must enter a URL to get an RSS news feed. Note that when users set up the site, or also later, they can add more than one News tool.

The Sakai Presentation tool allows one or more users to present a set of slides to many viewers. The presenter can move forward and backwards through the slides and the viewers follow each move within a few seconds. The viewers can watch the presentation in join mode or in view mode and navigate through the slides independently from the presenters (viewers can re-join the show at any time). While watching a presentation, it may take up to 10 seconds to see the new slide after the presenter gives the command. The viewer can refresh manually to see the latest slide if the automatic forward seems to be taking too long. To create a slide presentation, users must upload a series of files into the folder Presentations: within the Presentations folder, each sub-folder will contain a separate presentation. Slide order is determined by a sort of the file names or file titles (if provided). This tool can handle any web displayable image file including GIF, PNG, and JPG.

Resources is the most widely used feature in classes and collaborations. In Resources, users can make many kinds of material available online. There are three types of resources: documents (e.g., word processing documents, spreadsheets, slide presentations, plain text), links to other websites, and simple text documents that display right on the page.

Users can post up to 10 resources at a time, post items out of folders, create folders inside other folders, and create HTML documents. If they get the relevant permissions, users can control whether other users can post, delete, and read documents in specific folders or not. It so also possible to make a resource available to anyone by toggling on the "public view" setting. When users enter a resource area, they see a list of the resources and/or folders containing resources. These resources can be sorted by title, size, creator or last modification, in ascending or descending order.

The syllabus is the course official outline. Instructors can direct the syllabus tool to link to an already prepared online syllabus, if it is present. Otherwise, it's possible to post material to build it. While creating a syllabus, users can designate it to be viewable by the general public or just to members of the course. There are multiple options for making Web resources available: to include HTML tags, users can choose to format their syllabus content in HTML instead of plain text; to insert an image, they must provide the image's URL, because users can't upload or attach an image file to their syllabus.

The Sakai Web Content tool allows site owners to choose a website to display within their worksite's home frame. Users may display information in the Web Content window in different formats, including a text description, a web page, or an HTML file from their resources. Instructors who want to provide links to websites for their students may choose to list all of them in the resources section, and maybe to show only the most important ones with the Web Content tool. Departments, collaborations, or other groups may use the Web Content tool to provide a prominent link to their public worksite.

Sakai system supports several kinds of files: .gif, .doc, .html, .jpg, ...

b)   Course Organization

At this moment, no documentation has been found about this subject.

c)   Verifying the learning

In Sakai courses instructors have two different tools to check students level of learning: test and assignment.

The tests&quizzes tool enables instructors to create, administer and correct tests and quizzes online, and automatically feed the results to the gradebook (see also section 1.e). Instructors can create a new test from scratch, by using a template, or by using an import feature. To create a new assessment they have to follow four steps. First, they name the new assessment and choose its title. Then, there are several parameters to be set, such as the delivery dates, the time limit, the permission for multiple attempts, whether correct results will be shown, whether test questions will be randomized or not, and security settings. After that, instructors will create assessment parts. Every assessment must contain at least one part: this is the first one and is called Default. Parts can be added, modified, their order can be modified, and they can be removed, too. As a final step, questions must be inserted in the assessment. The system will show a question editing screen, with different kind of options depending on the type of question. The types available are: multiple choice, survey, short answer/essay, fill-in-the-blank, matching, true/false, and file upload. There are some options common

to all these types: answer point value (the score for each response in a question, see also section 1.e, it can be zero for a survey or ungraded question, see also section 1.d), question text (the actual question to be asked), answer (the correct answer or answers, each of them may include feedback), assign to part (if the test parts has been defined by the user, s/he may assign a question to any of those parts, otherwise, it will be assigned to the default part), and feedback (for correct or incorrect response to the question, this is optional). When instructors have inserted all their questions, they can preview the assessment 'as it is', and then insert it into their site.

Within courses, the assignments tool allows instructors to create, modify, grade, review, duplicate and delete online assignments. Assignments are private, so student submissions are not visible to other users of the site. Instructors can download all submissions to an assignment to their computer at once. Moreover, they can return assignments and allow for re-submission. This feature can be used to evaluate drafts of final projects or papers, or to allow students to correct and re-submit an assignment.

While creating an assignment several options are available: title, open date (when assignment will be available for work to begin), due date, "accept submissions until" date (no assignments can be submitted after this date, either before or after the due date), type of student submission (assignment can be typed into a text box, submitted as an attachment, or both), grade scale (ungraded, letter grade, points going from 1 to 200, pass/fail or checkmark), additional resource (such as attachments), and a description of the assignment with instructions for students. At the end of assignment creation, instructors can have students' grades released to the students either on command or immediately, and they can decide whether the assignment due date is to be added to the site's schedule. When the assignment is posted, the open date can be announced automatically by an announcement on their site's homepage. Instructors can also have the honour pledge displayed when students are submitting an assignment (the text of the Honours Pledge is as follows: "I have neither given nor received aid on this assignment.". Students must respond yes to submit their assignment). Finally, instructors can see an assignment preview, to see how it will look to students, and then save a draft copy of the assignment (only instructors can view draft) and post it.

Instructors can revise a posted assignment either before or after its open date. When students view the revised assignment, they will be notified that changes have been made.

Instructors can grade and review assignments. To give grades, instructors view a student's submission at a time and score it according to the grade scale chosen; they can also add comments, which will show in red. At the end, instructors will release grades and each students will be able to access his/her own. Instructors can review student status for all assignments: they can view a list of students and for each of them see their status for each of the assignments in the course. Instructors can see whether the student has submitted the assignment, the assignment status, and the grade if it has been given.

An instructor can also duplicate or delete assignments: the duplicate assignment (which will appear with the word "Copy" added to the title in the assignment list) is a draft and it will have to be explicitly posted. When instructors delete an assignment, they will see a confirmation message prompting them either to confirm or cancel the deletion.

d)   Verifying the use: survey

In Sakai instructor can create a survey using the same process as he would to create an assessment, since the survey question type allows to choose from a predefined set of scales commonly used for survey reporting. Those scales are: Yes/No, Disagree/Agree, Disagree/Undecided/Agree, Below Average/Above Average, Strongly Disagree/Strongly Agree, Unacceptable/Excellent, 1-5 and 1-10. Hence, to create a survey, simply create an assessment consisting of survey questions.

e)   How much knowledge: automatic correction

The system can automatically score true/false, multiple-choice, multiple-answer, matching, fill-in the blank, and short answer/essay questions. All results of the grading activities are collected in one of the course's most important objects: the gradebook.

The gradebook is a tool for instructors to calculate, store, and distribute grade information to students online. By using the gradebook, instructors can: automatically calculate course grades, add, view, edit, and release scored assignments and/or assessments to students, and besides, they can manually enter, view, edit, and release to students their scores and grades, and export them to Microsoft Excel (in .csv format). There are different ways to view information in the gradebook for instructors: general overview of each course,

assignment detail, course grade detail, student roster, [adding or editing assignments], gradebook options, and student view.

The gradebook will allow students to view their cumulative scores for all assignments and the corresponding percentage, and also their course grades (once an instructor releases them) and sort their assignments by title, due date, grade, and total possible points per assignment.

By default, a course grade is automatically calculated by the gradebook. This grade is simply calculated as a percentage of current points scored out of the total points which can be obtained if all assignments were entered in the gradebook. Therefore, if there are ungraded assignments, the auto-calculated course grade may not be accurate. Instructors can choose to manually override the course grade or use the auto-calculated grade. When manually entering a course grade, instructors must use the appropriate grade type for the gradebook they are working in. For example, if the course is graded on a simple letter grade scale, instructors will only be able to enter the letters A, B, C, D, or F. If the course is graded on a standard letter grade scale, they will also be able to add + or - to the letter grade. Instructors can overwrite all enrolled students grades at a time. However, if instructors have overridden a course grade, and they want to return to the auto-calculated course grade instead, they have manually delete their grade for each student, so that the automatic procedure can take place.

Instructors can also set student feedback options in the gradebook. By default, the system displays assignment grade, but it's possible to choose to display course grade, so that instructors may release their grades at the end of the grading period. It's possible to change grade type, viz. to make a conversion. It's possible to show the minimum percentage for each grade (manually entered course grades are not affected by the percentages in the grade/percentage table).

f)   Organizing a class: managing learners group

The software allows instructors or students to create groups by using distinct project sites, separated from the main course site. Each project site can have its own shared file exchange, discussion tool, calendar, announcements, chat, and group email list.

g)   Who sees what: access rights to course and contents

In the Sakai project, when creating a course or a project site, instructors can choose to edit access right in two different ways: if they wish to give access permissions to the entire community, they must select the appropriate option so that the site can be joined by anyone authorized to login, and then choose a role for all the users who may join the site; similarly, if instructors want their site open only to users they personally add, they must select the appropriate option.

Also, instructors can give access rights while adding and removing participants in a course or a project site. While adding participants, they will choose either to give all their newly added participants the same role or different roles (see also section 2). In fact, the Sakai system implements the concept of *permission*, which helps instructors to define which users can access certain features of a course or project website, depending on their roles. While creating a course or project website, instructors choose which tools or functions (e.g., discussion, schedule, resource, etc.) they want the site to have. For each of these functions, they can set permissions that allow or prevent users from seeing or performing certain tasks depending on their role. Furthermore, using permissions, it's possible to define access rights to specific contents: for example, a large collaboration site can use a folder in the resources area as a space to archive important documents, and allow only certain site participants to modify those documents.

h)   Course creation: authoring tools

In the Sakai system all authoring actions start from the Menu bar, an object available in most of the system's pages, which displays one button for each of the functions the user is permitted to use. By clicking on one of the buttons, the system opens the connected area and the user can access the allowed operation. Using the Menu bar, instructors can modify their course according to their tastes and students can, for example, edit their personal area within the course.

In some cases, on clicking on the Menu bar button to create a new content within the course, the system provides a Wizard to give help. It guides the user to complete a complex task by dividing it in of three or more steps (view). Each view of the Wizard contains a logical grouping of input objects or an item summary. Wizards are used especially when the number of inputs to be given for a particular item is large, so that

putting them all on one view would be lengthy or confusing, such as a schedule, or for infrequent tasks, to help users to follow the correct sequence.

Frequent tasks, such as adding a document or posting to a discussion board should use a one-view form.

i)   Different users, different learning path

Instructors can create linear learning sequences organized hierarchically by course, lesson, and topic.

j)   Time for everything: calendar

In Sakai project, the schedule keeps track of important dates and deadlines on the worksite calendar: this object has day, week, month, year, and a flat list views. Moreover, the schedule allows instructors or site organizers to post items in a calendar format. Any schedule item can have multiple attachments.

All schedules users have access to are merged in the MyWorkspace schedule of each user, and they can also selectively merge schedules from sites they have access to. Finally, users can print an Adobe PDF file of any view of a schedule.

k)   Hidden/in preparation material

Each user, independently from his/her role, can save in his own MyWorkspace resource area draft of his/her own work in progress, so none will access these documents, until they will be published.

l)   Synchronous communication: chat, video conferencing, etc

Chat is a tool for real-time, unstructured conversations among those users who are logged in to the site at the same time. For example, instructors can easily create an "Online Office Hours" chat room for student questions and answers, or dispersed collaboration groups can use the room as a space to have conversations from remote or catch up with conversations they may have missed.

The chat tool allows instructors or site owners to create more than one room, thus permitting to differentiate chats. The chat room tool alerts users on the chat room page when other participants are visiting that page. This way, users know who is available to talk. Instructors are able to change the chat room display; in particular, they can hide the date and/or the time next to each respondent's name, or set how many messages will be displayed.

By default, chat messages are saved and visible to all users so that all site participants can benefit from clarifying conversations and from asking questions and getting answers. Only an instructor or a participant who has a special permission can delete chat messages. By default, all chat messages are shown for at least three days.

The system doesn't provide video services or a shared whiteboard.

m)   Asynchronous communication: forum, topic forum, newsgroup

The Sakai discussion feature allows structured conversations, not real-time, organized into categories. Opening the discussions section, users will see a list of categories and, under each category, any topics and responses that have been added. It's possible to view only the categories by collapsing the list. Any post content can be seen by clicking it. The discussion tool displays discussion topics and replies within two frames. The layout can be changed to vertical or horizontal (this is a personal preference affecting only the personal view): one frame shows the hierarchy of categories, topics, and replies - by clicking a topic or a reply the system shows the content of the message in the other. The content window is the right frame in the two column layout (vertical), and the bottom frame in the two rows layout (horizontal). Only instructors and users with special permissions can delete discussion items. Instructors can choose whether site participants are allowed to post their own topics or not, teachers can also lock topics. Specifically, when posting a topic in the discussion area, instructors can specify whether they wish to allow students to respond only to that topic or to any topic. If they allow postings only to some topics, then they create a "flat" discussion. If, on the other hand, they allow replies to any posting, then they create a "threaded" discussion. Threaded discussions are naturally more interactive because users are able to reply to replies. Flat discussions, in contrast, tend to be more structured and for this reason are more easily controlled.

First of all, in order to create a discussion topic users will choose a category or create a new one. Then, they can choose replies options (flat or threaded discussion), type text in the message box, and, if desired, add attachments (local files from the user's computer, URLs or files from the website resources section). At the

end, users can either save a draft of their entry or post the topic to the discussion. If they save it as a draft, it will show up in the list with other discussion items, but will be marked as a draft and will be visible only to the author.

The Sakai email archive tool maintains an archive of all email sent to the worksites email list. Each site has an automatically generated site email address (site owners can create an easy-to-remember alias for it), and emails sent to this address are copied to all site participants and owners and stored in the email archive. The email sent to site participants will show if an attachment is present, and will provide a link to get the attachment from the worksite itself. The messages can be sorted by subject (alphabetically), by sender, or by date (in ascending or descending order). As students drop or add to the course, they will be dropped or added from the email group accordingly. Users who participate in more than one site can choose to receive email either "as it is", or in digest mode, with all posts for the day sent in a single email message.

The Sakai drop box feature allows private file-sharing between instructors and students. It works like the resources tool to allow users to upload many types of files and many files at a time, and allows nested folders, too.

n) Other

For any content posted to a course or in the personal areas, a copyright statement can be added.

II. ROLES

The Sakai project provides several kinds of users: affiliate, assistant, candidate, institutor, member, observer, student. All these roles have different permissions that have their extremes in the two system predefined roles: access, similar to the student role (can read content and add content to a site where appropriate), and maintain, that gives users full permissions throughout the site (can read, revise, delete, and add both content and participants to a site), and is automatically given to the site creator. Each of this roles has its default permissions; however instructors can add additional permissions, but they cannot remove any default permissions.

So in a Sakai system user permissions are almost completely customizable. Here we will specify only the features of the most important user types.

a) Learners

These users can surf within the course in the open area, read contents and, when appropriate, edit course material. For example, the simplest action that they are allowed to perform is to interact with their profile. The Sakai profile tool allows users to add and publish personal information, including pictures, resources or links. Moreover, students can self-register.

b) Authors

These members can specify the site parameters by editing course options. The Sakai Site Info tool allows users with appropriate permissions, such as instructors, to view worksite profile and participants list about the site they are navigating. If an user's role on that site is "maintain", he can use this tool to make changes to the site's information, tools, and access.

c) Tutors

This traditional role can be performed by users with affiliate, assistant, or member permissions.

d) Administrators

This is the role that allows both to access author-like to sites and to use administrator tools, too. Administrators can batch add students to the system and courses using providers or scripts. Administrators can create an unlimited number of custom organizational units and roles with specific access privileges to course content and tools. Instructors or students may be assigned different roles in different courses and group contexts. Administrators can distribute permissions and roles across multiple institutions or departments hosted in the server environment.

e) System managers

This role can be performed by users with institutor permissions, or by other users with a maintain role.

f) Other

III. RELATIONSHIP WITH STANDARD

The standards that form the foundation of the Sakai Tool and Service Portability Profile include: JavaServer Faces and OKI OSIDs. In addiction, the Sakai Project will define its own internal standards that are part of the Portability Profile. These standards will build upon and add detail to the OKI and JSF Standards to define their use within the Sakai Framework: Sakai GUI Elements, Sakai/OKI Façade Services, Sakai Application Services and Sakai Foundational Services.

Finally, using the Melete open source lesson building tool, the system can export course content using the IMS Content Packaging standard. The system can import assessment content in the IMS QTI 1.2 format.

IV. USABILITY AND LEARNABILITY

a) Usability test: existence and results

SAM is one of the reusable software modules created by the Sakai team. Indiana University will be the first school to utilize it. Indiana's User Experience Group lab was launched in 1998. The lab manages as many as 50 projects annually, primarily supporting enterprise and department-wide application development initiatives. To replace and upgrade the lab's eight-year-old hardware with next-generation technology would have costed over $20,000. But before purchasing a hardware-based solution, the school investigated more affordable and flexible alternatives.

The school in the end decided to abandon its clunky hardware in favour of Morae, a software-based solution from TechSmith Corporation. A brief documentation about Morae evaluation of the Sakai Assessment Manager is at the website address http://www.techsmith.com/casestudies/morae/indianauniversity.asp .

b) Learnability test

On the whole, the system is quite simple to learn and to use, thanks to the particular attention developers took in following some fundamental rules to design the user interfaces of a Sakai application. All these rules are summarized in the Sakai Style Guide, with a particular emphasis on principles of accessible and usable interface design.

c) Accessibility

The Sakai project website contains HTML renderings of the wireframes with integrated accessibility notes. The notes contain considerations for those using adaptive technology, which includes the following: screen readers, text magnifiers, voice activated controls, specialized keyboards, and Braille displays.

d) Multi-language support

The existing Sakai tools have been internationalized, allowing translation of Sakai 2.0 into multiple languages.

V. PEDAGOGICAL METHODS

In the Sakai project web site there is a whole section about the relationship between Sakai and pedagogy, and the SEPP (the Sakai Educational Partners Program) has a pedagogy discussion group to focus on the issue of the practice of teaching and learning and their relationship to the Sakai project. This is the web address to the section of site http://bugs.sakaiproject.org/confluence/display/ENC/Sakai+and+Pedagogy .

As for other LMS systems, Sakai components (chat, forum, quiz etc…) allow the free (hand-made) construction of courses following different pedagogical approaches depending on the Teacher preferences. Sakai has no specific automatic support for pedagogical methodologies, e.g. to course personalization. The communication tools provided allows the Teacher to follow Socio-constructivist approaches.

VI. INSTALLATION

The packaged distributions of Sakai include: Sakai demo, Sakai source and Sakai binary. The packaged archive distributions of Sakai come in two formats: a .zip file for use on Microsoft Windows operating systems, and a .tar.gz file for use on unix, linux, other *nix, and Macintosh OSX operating systems. The file permissions and the text file line ends are tuned for these different systems in the different distributions.

To fully appraise system's capabilities, user needs to create his own custom Sakai configuration. It allows to customize used database, the server id for each of the application servers, the URLs of the Sakai service, the skin and appearance setup, the control of optional Sakai features, the email configuration and the Sakai service identity information. Any configuration of any Sakai component can be overridden by this custom configuration. User may also want to configure Tomcat and the Java virtual machine for optimal Sakai operation and performance.

As a further help, the system provider makes available an Installation FAQ section in the web site. This section is sorted by release and reports information on how to upgrade, install, configure, modify and, when necessary, troubleshoot a Sakai installation.

VII. MAINTAINABILITY

The Sakai Project follows what is called the community source model, which is an extension to the already successful, economically feasible, open source movement forged by projects such as Apache, Linux, and Mozilla. The Sakai Project is an actively evolving ecosystem that currently consists of a community of core schools, which are developing the framework and basic set of collaboration and learning tools, the Sakai Educational Partners Program (SEPP) with a membership of over 80 actively engaged institutions, the Sakai Commercial Affiliates vendors who offer support and hosting for the Sakai CLE, and the public contributors to the open source project. All of these participants contribute to the project in a variety of ways and, ultimately, to the development and support of the open source collaboration and learning environment and its tools.

On the web, there are a lot of interesting articles about the Sakai System. For example, "Sakai and the 4 Cs of Open Source Campus Technologies" by K.C.Green.

VIII. REQUIRED HW or SW

Sakai requires Tomcat 5.5.9 or later, and the Tomcat compatibility package for use with Java 1.4.
Sakai also requires Java 1.4.2 or later (Note: Java 1.5 has not been tested with this release).
Moreover, Building Sakai requires the Maven project management tool, and system providers recommends the Eclipse Java IDE for developing Sakai code.

IX. LICENSING: OPEN SOURCE or PROPRIETARY

The Sakai Project Software is available for free download and use without fee under the Educational Community License Version 1.0.

X. TOTAL COST OF OWNERSHIP

The Sakai project product is a free software, so it can be freely downloaded, installed, managed and developed.

On the other hand, the Sakai Commercial Affiliates are commercial firms that offer for-fee support and expertise for the Sakai Project's community source software. Affiliates offer their experience with open source software to provide hosting, consulting, installation, integration, and support services. This group of companies includes: Embanet, HarvestRoad, IBM, justStudents, Ostrakon, the rSmart Group, Sun Microsystems, SunGard SCT, Unicon, and Unisys.

XI. USAGE

There are a lot of institutions that are in various stages of Sakai exploration and implementation. Here are some examples.

The Massachusetts Institute of Technology will deploy Sakai as a production pilot in fall 2005, with a goal of expanding the pilot and incorporating additional Sakai tools throughout the 2005-2006 academic year. MIT plans a campus-wide deployment of Sakai to replace the current Stellar system by fall 2006. The MIT implementation will be guided by three key considerations, including conformance to Institute requirements for security, accessibility, and enterprise integration; usability performance that is commensurate with our support capacity; and functionality enhancements to the current Stellar suite.

At Indiana University, Sakai 2.0 is in full deployment for summer 2005; faculty, staff and students have been encouraged to become familiar with the system by creating worksites, and faculty members can choose to use the system for their courses. Statistics as of July 2005 claims that have been created 1146 sites all semester and have been logged in at least once 3668 unique users on a number of 5237 with access to one or more worksite.

The University of California in Berkley is currently working on its Sakai implementation design and on campus integration issues. Plans include releasing a preview to a controlled number of users (5,000-10,000) in fall 2005. A January 2006 implementation will be open to all interested users. Migration from current systems will occur in a rolling fashion as needed.

In fall 2004, the University of Michigan began a migration of its course management software, CTools, to Sakai. For the fall 2004 term, approximately 1,591 class sites were set up, as well as 2,810 project sites. By mid-February 2005, CTools had more than 25,000 users. The number of daily users ranged from 5,000 (Fridays and Saturdays) to 15,000 (Mondays and Tuesdays). The average student logged in about twice a day (1.77 times) for an average of 43 minutes per session. Max concurrent users reached 700+ at peak usage times. On July 21, 2005, in preparation for the 2005 fall term, UM deployed a version of CTools based on Sakai 2.0. System providers expect 30,000+ users with concurrent usage above 1,000 at peak times. Load testing is in progress to help determine the number of application servers needed at start of term.

The Lubeck University of Applied Science (Germany) have began a pilot in spring 2005, with full deployment beginning 2006.

The University of Cape Town (South Africa) intends to deploy Sakai in February 2006.

The University Fernando Pessoa (Portugal), a private university in Porto, with 4500 students (undergraduate and graduate) is running Sakai 1.0 in a pilot with 450 users involving 60 courses. The university plans to fully adopt Sakai as the CLE platform.

The University of Lleida (Spain) fully adopted Sakai in 2005.

XII. SUPPORT QUALITY

The Sakai Project is a community source software development effort to design, build and deploy a new Collaboration and Learning Environment (CLE) for higher education. The Project began in January, 2004.

The Sakai Project had its origins at the University of Michigan and Indiana University, where both universities independently began open source efforts to replicate and enhance the functionality of their existing CMSs . Soon after, MIT and Stanford joined in and, along with the Open Knowledge Initiaitive (OKI) and the uPortal consortium, and a generous grant from the Mellon Foundation, they formed the Sakai Project.

The tools are being built by designers, software architects and developers at different institutions. To provide a support system for institutions that want to be involved in the Sakai Project, either by adopting Sakai tools or by developing tools for inter-institutional portability, the Sakai Project has also formed the SEPP and the Sakai Commercial Affiliates Program.

In particular, to support Sakai users, in the web site are available two different types of discussion groups: public groups (there are two of them), where the Sakai user community can exchange best practices, share ideas, pose and answer questions, and communicate about their experiences; and SEPP groups (fifteen different forums), created by SEPP members in response to a need to exchange ideas, explore issues, and generate discussions and activities surrounding key development or user issues.

14. LMS: WebCT Campus Edition 4.1

WebCT currently sells and supports two product lines: the WebCT Campus Edition (course management system, with a complete set of teaching and learning tools for course preparation, delivery, and management) and the WebCT Vista (academic enterprise system, specifically built to support enterprise-class e-learning deployments).
This report evaluates WebCT Campus Edition because it's more suitable for this study.
Blackboard Inc. has recently acquired the WebCT company, thus we expect that the two systems will merge.

I. TEACHING FEATURES

a) Contents
WebCT's digital content solutions make it possible to easily identify, use, and re-use online content.
The system provides many tools for content creation and delivery within individual courses. Within a WebCT course it's possible to have students' *annotations*, *glossaries* (where students can view and download a printable version of a list of commonly used terms, divided by initial or in a unique block), *assignments* (also called exams or quizzes, see also section 1.c), *self-assessments* (see also section 1.c), *gradebooks* (a gradebook backup can be downloaded, and opened offline, for example, as an Excel file, see also section 1.e), *discussions* (see also section 1.m), *chats* (see also section 1.l), *email service* (see also section 1.m), *syllabus* (see also section 1.h), *calendar* (see also section1.j) and *resources*.
Instructors can categorize course items such as calendar entries, course units, discussion forums, glossaries, syllabus, tips, and resources. Course content may be uploaded through a form, or may also be exported for later use.

b) Course Organization
Using WebCT the course categorization coincides nearly always with the semester division, i.e. it is a temporal categorization, more than content related.

c) Verifying the learning
In WebCT there are several tools to verify learners level of knowledge.
Instructors can create self-assessments, that will be automatically corrected by the system (see also section 1.e). They can insert quizzes into a course: it's simple. They need only to invent questions and then insert them into a question database. Then instructors can import in their course some questions chosen from these test banks and add them to the quiz itself. WebCT provides multiple choice, true/false, matching, calculated, short-answer, multiple answer, essay questions and more; they also can use the Mathematics Mark-up Language equation editor to enable students to enter and edit mathematical notations in their answer.
Finally instructors can set a time limit on a test, use IP address to restrict access to tests, (for example only to users attempting an exam from a certain laboratory room), and can create proctored exams.

d) Verifying the use: survey
Surveys are very similar to quizzes. The concepts of creating a questions database, creating a survey and then adding the questions to it are identical for both. To view the results, the process is also the same. The differences are that since surveys are not graded, the point value into the questions is not relevant, that surveys are anonymous, and that a survey can be taken only once without being reset.

e)   How much knowledge: automatic correction

The system automatically marks multiple choice, true/false, calculated, matching, short-answer and multiple answer type questions and can display instructor-created feedback.

When an instructor adds an assignment to the course, the software automatically adds it to the online gradebook. To make this tool more compliant with instructor demands, the gradebook supports the creation of custom columns which can contain either grade information or other instructor-determined details.

Instructors can mark all assessments which are not automatically scored online and can add the grades for offline assignments to the online gradebook. In general, they can manually edit all grades. Finally, instructors can assign partial credit for some answers.

Instructors can export a comma-delimited version of the gradebook to an external spreadsheet program.

f)   Organize a class: managing learners group

Either instructors can assign students to groups or the system can randomly create groups. Each group can have its own shared presentation folder and discussion forum.

Students cannot build communities by themselves.

g)   Who sees what: access rights to course and contents

Instructors can personalize access to specific course materials based on group membership and can set up specific course content to be released on a specific date.

h)   Course creation: authoring tools

The software provides support for template-based course creation. The templates include a WYSIWYG content editor. Instructors can use templates to create syllabus, course descriptions, course units, discussion forums, glossaries, calendar entries, chat, and resources. The system provides course design wizards i.e. step-by-step guides to help faculty and course designers to carry out common course tasks, such as setting up the course homepage, syllabus, organizer pages, content modules, discussions, mail, calendar and chat.

For example, to create syllabus, system provide a wizard that will stepwise prompt instructors to: add the title, determine syllabus location within the course, choose how to display it (title and/or icon). At this point the syllabus is only an empty shell, it doesn't contain any information. Now, to create or add information to the syllabus, the wizard will prompt the instructor to add as many different sections as s/he needs, selecting from a drop-down list: course information, instructor information, course goals, policies, textbooks, course requirements, lesson and custom. Finally the wizard builds a preview of the syllabus to allow instructors to see what they have created and, once they are satisfied, he can save the syllabus and exit the wizard.

The system provides default course look and feel templates, too. Institutions can create their own look and feel templates across the entire system, including their own institutional logos, headers, and footers. Instructors can alter the appearance of their course.

i)   Different users, different learning path

Instructors can specify multiple paths through a course for different skill levels or job functions. They can personalize access to specific course materials based on previous course activity or student performance, can set up specific course content that must be completed by students before they continue, or can design courses for facilitated learning or system managed for self-study.

Instructors can create both linear and nonlinear learning sequences. They can organize learning objects, course tools, and content into learning sequences that are reusable. Learning objects can be imported from external authoring tools and added to courses.

j)   Time for everything: calendar

Instructors and students can post events and announcements in the online course calendar. Students can keep track of all their assignments (they can also see their grades), deadlines, and due dates in an online course calendar.

k)   Hidden/in preparation material

In WebCT it's impossible to hide portions of a content module, but user instructors can hide whole content modules using the selective release feature. So, to hide something it is necessary to divide up the content and create several modules, giving each the proper release criteria.

l)   Synchronous communication: chat, video conferencing, etc

WebCT provides users a Java-based chat tool. The chat window contains three main areas: the "output interaction" box (displays all messages sent and received), the "users logged on" box (shows the users presently in the room) and the "enter your message below" text box (where users type their messages). Users may either send a message to everyone in the room or send a private message to someone in the room and activate the entry chime. The chat tool supports private rooms and private messages, too, up to four simultaneous group discussions. Finally, the system creates archive logs for all chat rooms.

The software provides an instructor-controlled shared whiteboard that supports image uploading and annotation; it can also archive snapshots of whiteboard sessions for future viewing.

WebCT doesn't provide video services.

m)   Asynchronous communication: forum, topic forum, newsgroup

WebCT provides users a discussion tool, that allows to view discussions by date, by thread, by subject, by author and by status (read or unread post). Messages can include attachments and URLs and a formatting text editor is available to write them. Discussion threads are expandable and collapsible - an entire conversation can be seen on one screen; moreover the discussion can be saved or printed for off-line reading. Finally, instructors can determine the level of involvement (read, write, or post anonymously) for students and they may create separate discussion environments for small groups.

The platform doesn't allow the file exchange between all system users, but students can submit assignments using drop boxes and upload files to a shared group folder.

WebCT has an internal email service. Students can attach files to their messages, spell-check outgoing messages, forward them to external email accounts, and also search in their email folders either by date, or object, or receiver, etc...).

n)   Other

Instructors can publish course content on a CD-ROM that can be linked to dynamically from within the online course or viewed offline.

II. ROLES

In WebCT there are only three different user types: designer, teaching assistant and student. These are fixed roles and there are no options to allow for additional custom roles. There are two additional roles at the administrative level: administrator and help desk.

a)   Learners

Students can create a personal home page in each course they are enrolled in. It lists for all courses in which the student is enrolled, the new emails and all course and system-wide events, from his/her personal calendar. Moreover, students can use their personal home page to display their work in the course, and may even export their homepage.

Students can attach notes to any page and combine their notes with the course content to create a printable (or locally stored) study guide. They can create bookmarks in a private folder, too.

Students can search all course content, discussion postings, and email messages within a course, also by using filters.

Upon re-entering a course, students have the option of resuming at the last page viewed.

Finally, students can access context sensitive help for any tool.

b) Authors

Instructors can add students to their courses manually or allow students to self-register.

Instructors can get reports about attendance: the number of times each student or all of them (or an aggregated group) accessed course content, or specific course units and discussion forums; but also date and frequency of these events. Instructors can also get a report showing the time each student spends on course content, specific course units and discussion forums. Moreover, instructors can to share this information with students.

c) Tutors

As said above, in WebCT users roles are very restricted, but instructors can delegate the responsibility of grading assignments, thus defining the teaching assistant role.

Moreover, the administrator can give a particular user designer privileges within a specific course, to help instructors to maintain course contents.

d) Administrators

Administrators can batch add students to the system using a delimited text file. They can also transfer student information bi-directionally between the system and an SIS.

Administrators can protect access to individual courses with a username and password (the system has a password reminder option). Access can also be restricted based on IP address.

Administrators can assign different levels of access to the system or courses based on the following pre-defined roles: instructors, students, designers, teaching assistants, and system administrators. Instructors or students may be assigned different roles in different courses.

e) System managers

As said above, WebCT administrators can assign the system administrator level to a user. In general this is not very common, because the system is able to self-administer and a user administrator is enough. However for big environments it can be useful to make less burdensome the administrator task.

f) Other

III. RELATIONSHIP WITH STANDARD

The product provider has self-tested the software and reports compliance with the following: IMS Content Packaging 1.1.2, IMS QTI 1.1, IMS Enterprise 1.1, and Microsoft LRN 2.0.

IV. USABILITY AND LEARNABILITY

a) Usability test: existence and results

A usability test on WebCT was made in 2001 by University of Patras (Greece) and published on "International Journal of Educational Telecommunications" with the title "Evaluation of Distance-learning Environments: Impact of Usability on Student Performance". It's available at the web address http://www.ee.upatras.gr/hci/papers/j30_Tselios_Avouris_Dimitracopoulou_Daskalaki_ 2001.pdf , but it refers to an old version of the system, 2.1.

b) Learnability test

No documentation has been found about WebCT learnability test.

c) Accessibility

To comply with Section 508 of the US Rehabilitation Act, the software implements the following features: alternative text for non-text elements, content available in a high contrast colour scheme, collapsible menus, content is only presented in text and graphics, content readable without style sheets, keyboard access to chat

tool, appropriately titled framesets that describe the functionality of the frames layout, and allowing invisible navigation links to be used by screen readers.

d) Multi-language support

WebCT is available in 14 different languages, by plug-ins. A WebCT language plug-in is a set of library files that allow WebCT to be displayed in a different language. Institutions can install multiple language plug-ins on a single system. For WebCT Campus Edition 4.1 these plug-in are available: Arabic, Brazilian Portuguese, Croatian, French, German, Spanish, Swedish and UK English; for the previous version (4.0) were available: Arabic, Chinese simplified, Chinese Traditional, Dutch, Farsi, Finnish, French, Galician, German, Spanish and UK English.

It's important to note that the software supports right-to-left languages, such as Arabic and Hebrew, and for these languages, any left-to-right language words or phrases, which are embedded in the text, are display correctly (i.e. left-to-right).

V. PEDAGOGICAL METHODS

In the web site (www.webct.com) the system provider claims to have a policy of flexible solutions for their users environments. This flexibility determines also the pedagogical approach followed by the system. This is an extract from the text in the web site: "E-learning has become pervasive among educational institutions of all types, from community colleges, to K-12 school districts, to private universities, to educational systems with hundreds of thousands of students. The diversity of these institutions demands that e-learning technology provide the highest level of flexibility to meet their needs. WebCT calls this "learning without limits". "Learning without limits" means enabling every institution to achieve its unique e-learning objectives. WebCT's products and services are designed to support a broad range of: stages of implementation, institutional strategies, user experience levels and pedagogical and learning styles.".

As for other LMS systems, WebCT components allow the free (hand-made) construction of courses following different pedagogical approaches depending on the Teacher preferences. The tools provided (forums, assessment) allows the Teacher to follow Socio-constructivist approaches. WebCT has no specific automatic support for pedagogical methodologies, e.g. to course personalization (even if the teacher can design different learning paths for different groups of students or can use a rule system to release learning content).

VI. INSTALLATION

Typically, local administrators install the software. Usually only few hours are needed to complete the procedure, specially if there is data migrations from older versions, from a server system to another. However there is a quite detailed documentation and is strongly recommended to contact WebCT Support before starting.

VII. MAINTAINABILITY

WebCT, or World Wide Web Course Tools, began in 1995 as part of a research project led by the University of British Columbia professor Murray Goldberg to study the effects of online teaching on learning. Murray founded WebCT in 1997 at UBC, and delivered it as a commercial product at that time. In 1999 the company was acquired by Universal Learning Technology (ULT) and combined company was renamed WebCT, and headquarters moved to Lynnfield, Massachusetts. WebCT is a privately held company backed by a group of investors (which include CMGI@Ventures, JPMorgan Partners, SCT, and Thomson Corporation). Today WebCT Inc. is the world's leading provider of e-learning systems for higher education institutions.

VIII. REQUIRED HW or SW

As a client browser, the software requires Internet Explorer 5.1+, Netscape 6.2.1+, and AOL 7.0. Javascript must be enabled in all browsers. Internet Explorer 5.5 SP1 is not supported. Use of some features requires Java Virtual Machine (JVM).

The following tools for servers are available: resource monitoring, crash recovery, backup of a course to a desktop machine. The server administration tools are accessed over the Web. The software requires Perl 5.6.1 and Apache 2.0 both of which are bundled with the system.

IX. LICENSING: OPEN SOURCE or PROPRIETARY

WebCT Campus Edition is a proprietary software delivered by WebCT Inc.

WebCT Campus Edition is licensed in two variants: the Focus License allows a subset of the functionality included in the Institution License by restricting the number of student seats and eliminating access to APIs that allow integration with campus systems. The company offers consulting services for implementation planning and advanced technical services including an initial technical assessment and evaluation of issues such as authentication, load balancing, and migration and upgrade planning. Premium support services are available for quicker response time, direct access, and 24/7 support. The company also offers training.

The product provider offers a hosted system that includes service level agreements with guaranteed system availability, utilizes a network of high-performance, fault-tolerant servers, a secure facility with environmental control, redundant and conditioned power, a modern alarm/security system, 24x7x365 monitoring, a direct T3 connection with second redundant T3 connection, and daily offsite tape backups. Individuals can access course and system information from any computer that has a web browser and Internet connection. Hosting is also available from Embanet, which provides daily and offsite tape backups, system clustering, managed bandwidth usage, and multiple Internet service providers to provide redundant fail-over capabilities.

X. TOTAL COST OF OWNERSHIP

The Campus Edition Institution License is priced based on number of full-time equivalent (FTE) students in an institution. The Campus Edition Focus License is priced based on a limited number of student seats (normally 3000).

The software is licensed on an annual subscription basis. Technical support is provided by email, web form, or phone. The annual license fee includes support for two administrators per license.

XI. USAGE

From 1995, starting year in the University of British Columbia, to today WebCT has grown enormously. Today, it is used by 10 million students at over 2,500 universities and colleges in 80 countries, and is described as having "revolutionized the way people teach and learn" (UBC, 2004). In 1999, the WebCT interest merged with Universal Learning Technologies of Massachusetts, and has offices in Canada, the US, and Australia. In the process, the product's mission has expanded from serving purely educational purposes to providing "enterprise-wide" learning management solutions in all areas of e-learning and training.

The WebCT Campus Edition is designed to support single institution use with limited customization and branding capabilities. It is scalable and supports integration with campus portals based on JASIG's uPortal framework a variety of portals and other campus systems.

XII. SUPPORT QUALITY

WebCT offers a full range of services to support successful implementations of WebCT Campus Edition, including technical services, professional development, hosting, and expanded support options. The WebCT Professional Services Team has on-campus experience deploying e-learning solutions at institutions

worldwide. Rely on their expertise to save time, maximize resources, and leverage best practices when deploying users e-learning solution.

Moreover, at user level, instructors can access the online help manual, context sensitive help, and numerous instructor support communities to share information in a number of discipline-specific or general interest forums; they can subscribe to an instructor mailing list. Finally they can take online courses about instructional design strategies for online courses and how to use the product.

15. LMS: WebTeach/TWiki

WebTeach is a prototype LMS developed and used by the Engineering faculty of Florence University, Italy. It's a collection of tools built by extending a collaborative web-based CMS (Content Management System) following the Wiki idea (all pages are editable through the browser).

The wiki implementation used is TWiki (www.twiki.org), that "as-is" fulfils the content-management requirements of a LMS. A straightforward TWiki is in use at the Computer Science Department of Rome 1 University.

I. TEACHING FEATURES

a) Contents

The TWiki CMS allows the creation and editing of web pages, and also the "attaching" of any kind of file-based resource to a web page. In TWiki the following features are available:

- **Automatic linking**: Wikis automatically link names of other pages appearing inside a page text, thus simplifying the handling of links,
- **Full-text search**: pages can be searched either or by simple keywords or by full regular-expressions,
- **Attachments**: any kind of file (pictures, handouts, multimedia) can be "attached" to a page, thus simplifying the development of content-rich pages,
- **Versioned pages**: all pages are kept under the RCS versioning system, thus allowing the reversal of any edit error,
- **Automatic notification of changes**: Users can be automatically notified about changes in a Web if they like,
- **RSS syndication**: each Web publishes an RSS feed of its last changes, thus allowing both the construction of a web portal merging the RSS feeds of different webs, and the subscription of any interested user to the "last news" feed of their interest,
- **Simplified "wiki" format**: pages can be written without knowing HTML by using simplified formatting rules similar to text emails, including: headings, bullets, automatic URL linking, tables,
- **Page inclusion**: a page can be included in another page, thus allowing a good modularization of the site,
- **User and Group authentication**: the access rights of the user/groups can be set both at page, web and site level, both for the view and edit/rename/attach actions.

TWiki comes with many "server-plugins", which can extend its features, such as:

- **Calendar:** a calendar to collect and display events
- **LaTeX formulas:** LaTeX formulas written in the page are rendered and displayed
- **Threads:** a page can be easily transformed into a discussion thread
- **Graphs:** graphs can be embedded in a page, automatically rendered and displayed
- **Spreadsheets:** a page can embed tables containing formulas computed on-the flight,
- **Presentations:** a page can be transformed to a presentation, one slide for each section.
- **And more …**

WebTeach extends TWiki by defining:

- **Parametric multiple-choice quizzes**: for generation and handling of homeworks and exams. The parameterization allows the generation of many slightly different quizzes from a single parametric template. The generated quizzes can be randomized in many ways: either according to parameters, or according to choice and order of questions or according to choice and order of answers.

  Exams can be delivered via web or can be printed and delivered in-class; answers can be collected by scanning the answer-sheets filled by the students.

- **WebCheck**: a web-based database system to handle courses, events, class allocation and reservations.
- **2D and 3D plots:** a page can embed a 2D or 3D plot of a set of functions or of a collection of data.

b) Course Organization

TWiki allows the organization of pages in different groups (called Webs), each course is usually contained in a separate Web. Each course Web has its automatic notification page, where students can enrol to stay informed about course modifications. The flexibility and ease of use of the Wiki CMS allows teachers to create the organization that best suits them. Every page can embed several elements, thus the teacher can shape the course Web as s/he likes. Two styles of course have been observed:

- the teacher uses the pages just to give course information, and attaches all handouts as files to the course pages,
- the teacher makes full use of system to write rich interlinked course pages.

c) Verifying the learning: quiz and choice

WebTeach extends the simple TWiki CMS by allowing the construction of parametric quizzes. The quizzes are mainly multiple-choice with several levels of randomization: Quizzes are organized in groups (eg. "Differential Calculus") of questions, and can contain numeric parameters that will be chosen and computed at generation time. Exams can be specified by listing how many questions should be chosen from each group. The generation can introduce several levels of randomization:

- the questions chosen from the groups and their order
- the values of the parameters
- the answers chosen to be displayed and their order

Exams can be delivered both as web pages and printed, getting answers by optic scanning the answer sheets filled by the students.

d) Verifying the use: survey

There are server plugins to run polls, that can be used to verify the usage of the system. Else a comment page can be easily organized to collect comments or suggestions.

e) Verifying the learning: assignment and journal

Other cooperative tasks can be organized with the students, by shared editing of pages or by structuring a page as a forum. Thus a journal is very easy to implement.

f) How much knowledge: automatic correction

Quizzes can be automatically checked by WebTeach. A system for the automatic testing and grading of programming exercises is currently under development.

g) Managing learners' groups

TWiki allows the creation of groups of users. A group can contain any set of users and even include other groups. A user can participate to several groups at once. Each user has its personal page in the Users web, containing his/hers personal details and preferences. Users can attach files to their personal pages, that can be used as a personal repository in the system.

h) Who sees what: access rights to course and contents

Each user can register himself in the system, choosing their password and username. TWiki authenticates the user for each action in the system (view, edit, rename, attach, post a comment). The access rights can be specified: globally, at web level or at page level, thus the teacher can decide which pages can be seen/edited/commented and from whom.

i)   For course creators: authoring tools

WebTeach makes available for parametric quizzes a JavaScript browser-based editor. TWiki allows the definition of "template topics" that can be filled automatically by data inserted in a form, thus it's very easy to design standardized pages.

j)   Different users, different learning paths

An automatic learning paths configuration tool is under development at Rome 1 University to allow the construction of personalized courses spanning. The student answers to a set of quizzes to determine his/hers "initial knowledge", the teacher specifies the "target knowledge" of the course, the system then selects and orders the course topics needed to reach the "target knowledge" from the "initial knowledge". A set of personalized graphic and textual navigation elements are generated so that the student interface reflects his/hers personal view of the course.

k)   Time for everything: calendar

A calendar object can be embedded in any course page, thus several different calendars can be generated. A calendar displays events listed in the current page or in other pages. A simple form can be used to insert new events in the page/calendar. An event can happen once, or be repeated, or can last several days. Events are shown in the calendar as links to the page that contains their description.

l)   Hidden-in preparation material

As the access of any web/page can be restricted, it's easy to work on "in preparation" material by allowing its view to only the author/teacher. When the material is ready the author can lift the restrictions. This can be done both at web (course) level and at page level.

m)   Synchronous communication: chat, videoconferencing, etc

At this moment no chat, videoconferencing or shared whiteboard tools are integrated in TWiki/WebTeach.

n)   Asynchronous communication: forum, topic forum, newsgroups

Asynchronous communication is supported by TWiki both as shared editing of pages, by adding comments to a page (acting as a forum), or by inserting a "thread object" in a Wiki page to handle simple threads of discussion. All changes to the course web are automatically notified to the students that enrol to the WebNotify page of the course. A server-plugin is available to allow contribution to web pages through email.

o)   Other

II. ROLES

TWiki shows no real distinction among groups of users, except for the group of administrators, that has full power over the whole set of pages. Any kind of authorization structure can be implemented with access groups. A very common setting for teaching gives edit power at course level to the course teachers/tutors and leaves only view rights to the students. Students can be allowed the edit rights selectively to single pages.

a)   Learners

Any user has normally view access to all course webs, but access rights can be easily set-up to restrict the access only to the students enrolled in the course.

b)   authors

A group of teachers/authors can be easily defined so that they have full editing powers inside their course web.

c)   tutors

As for teachers, it's easy to define a group of tutors and restrict their rights to specific actions in the course web.

d)   Administrators

The whole TWiki site is managed by the users listed in the TWikiAdminGroup, thus the administration duties can be distributed over several managers. The administrators have full edit powers over the whole site, creates new course webs and sets-up all the site preferences.

e)   system managers

TWiki has no system manager because the whole site can be administrated by the admin user or by the authors.

f)   other

TWiki can allow unregistered users to login, using a guest account. This account normally has strict limitations and no privileges, but it can be useful to enable people to browse the system without enrolling.

III. RELATIONSHIPS WITH STANDARDS

The TWiki/WebTeach system does not use IMS or SCORM formats.

IV. USABILITY AND LEARNABILITY

a)   Usability test: existence and results

Some studies on TWiki usability have been done.
(We must link their results here).

b)   Learnability test

No documentation is available about learnability tests conducted on TWiki/WebTeach.

c)   Accessibility

TWiki is rather well documented, with documentation spanning both the features of the system and all the server-plugins installed. Useful documentation is normally linked in each web form where it could be helpful. Wysiwyg editors are available to edit all pages. WebTeach quizzes are edited through a JavaScript editor. All TWiki pages are enclosed in a common template containing navigation menus and tools, thus allowing both an easy and consistent usage of the site and consistent personalization of its appearance (with "skins").

d)   Multi-language support

TWiki is multi-language enabled and allows for the definition multiple translations of the same page. The system messages for each language can be easily edited by editing a page of the system.

V. PEDAGOGICAL METHODS

Wikis are usually used for collaborative learning, by enabling the students to shared editing of pages.
As for other LMS systems, TWiki/WebTeach components allow the free (hand-made) construction of courses following different pedagogical approaches depending on the Teacher preferences. The tools provided (forums, wiki etc.) allows the Teacher to follow Socio-constructivist approaches. TWiki/WebTeach has no specific automatic support for pedagogical methodologies, e.g. to course personalization (although a tool for automatic construction of multiple personalized learning paths is under construction).

VI. INSTALLATION

The TWiki installation is rather easy, but the installer needs some experience in setting-up the prerequisite Perl packages and the web server. TWiki can be installed both under the Unix and Windows OS.
WebTeach extensions are now being packaged as a set of server-plugins, thus making their installation easier. WebTeach uses the MySQL database server.

VII. MAINTAINABILITY

VIII. REQUIRED HW or SW

TWiki and WebTeach are written in Perl and needs some Perl packages easily installed from CPAN (Comprehensive Perl Active Network). Moreover, WebTeach uses the MySQL database. Other packages can be required depending on the server-plugins installed.

IX. LICENSING: OPEN SOURCE OR PROPRIETARY

TWiki and WebTeach are Open Source software, and can be redistributed and/or modified under the terms of the GNU General Public License.

X. TOTAL COST OF OWNERSHIP

Both the TWiki system and its WebTeach extension are free for usage by anyone (they are released under GPL). The WebTeach setup is not yet simple as other tools, thus requiring a system administrator with some good Unix experience.

XI. USAGE

WebTeach is used by the Engineering department of the Florence University. More than 5000 students are currently enrolled, with 80 courses and teachers. The tool is heavily used by the students, with an average number of connection of 500 a day. The system is in use from 1999. The quiz facility has been used also for the admission tests of the Florence University, with almost 1600 students attending the tests each year.
TWiki is used by the Computer Science department of the Rome 1 University, with more than 2500 registered users, containing more than 30 courses. The tool is used by the students. It has reached a maximum of 300000 views/month with an average number of connections of 6800 a day. The system is in use from 2001.
In general, the TWiki CMS is used by many enterprises and several big software projects as a knowledge tool, for shared editing, customer support and cooperative design of systems.

XII. SUPPORT QUALITY

TWiki, though is open-source, is well supported, with very strong commitment to enterprise-level needs. The TWiki community is very wide and creative, improving continuously both the core system and the server-plugins available. WebTeach, being an in-house extension, is less supported, even though the developers are very responsive to bug reports, comments and suggestions.

16. LMS: WeBWorK 2.1.

WeBWorK is an Internet-based system for generating and delivering homework problems to students. Its goal is to make homework more effective and efficient.

I. TEACHING FEATURES

a) Contents

WeBWorK is an Internet-based system for generating and delivering Mathematics and Physics homework problems to students. The purpose of this program is to make homework more effective and efficient, both by allowing teachers to author, display and print with typeset quality advanced mathematics problems, and also by recording and delivering students' answers. WeBWorK can handle most standard homework problems assigned in calculus, vector calculus, and differential equations courses. Professors can write their own problems or edit library problems. WeBWorK comes with a Rochester library of over 2500 problems covering pre-calculus, first year calculus, vector calculus, differential equations, probability, and statistics syllabuses. As a matter of fact, several colleges, universities, and high schools are distributing their own libraries of problems, which are listed in a section of the WeBWorK web site. Many famous institutions currently using WeBWorK, such as the Arizona State University, the University of Rochester, the Detroit County Day School and the University of Rochester, gave their contributions to this libraries collection.

Graphs of functions can be generated "on the fly" by a single statement thus enabling teachers to ask questions involving individualized graphs for each student. GIF, PNG, and EPS illustrations, animated GIF's, HTML hyperlinks, JavaScript code and Java applets can all be embedded in WeBWorK problems in order to enhance their educational effectiveness.

The pg language which has been especially developed for writing WeBWorK problems is built on the widely used scripting language Perl. Mathematical formulas can be written in LaTeX, the mathematical typesetting language, and as with TeX, ease-of-use has been added in the form of macro packages. Even complicated numerical subroutines can be included to help check the answers to problems (see also section 1.e). Novice problem writers will use these macro packages to write problems, while expert problem writers can extend the capabilities of the language by writing new macro packages.

Problems can have individualized solutions and/or hints (e.g., solutions use individualized constants for each student). After the due date, students can review the homework, including the answers expected by the instructor. Solutions to problems, if provided by the instructor, are also available after the due date. Students frequently work on old assignments to prepare for exams.

b) Course Organization

There are a few course specific variables to be set in a WeBWorK course by the administrator or by the course instructor, such as feedback addresses, statistical data configuration on the Professor Page, and the toggling of a "professor view" of the source file names of each problem. Also in the webworkCourse.ph file there are variables to be defined using course_webwork_setup.pl. If the default set-up is convenient for them, users may not need to edit these variables.

The first variable defined in webworkCourse.ph is the feedbackAddress, that should contain the addresses of the instructor (and also assistants, if any), so that feedback from students goes to the right person. Next, user shall enter a single valid email address for the smtpSender variable. Normally this is set to the address of the course administrator. Undeliverable email from the Send Mail page will be returned to the smtpSender address. Then, if an user is to write essay type problems or questionnaires where the results are to be emailed back, s/he will here enable and edit the email addresses to which the answers are to be sent. On the Professor page, the user can view statistical data on problem sets for the whole course by section or by recitation. User may want to exclude certain sections or recitations (e.g. those containing practice users, TA's, or professors) from the overall statistics. Statistics for excluded sections are reported separately. To do so, the user will edit the webworkCourse.ph file and list the names of all sections and recitations to be excluded in a comma separated list: if this list is empty, no sections will be excluded. Finally, the user will give her/ himself the "professor view", so that when s/he views a problem or downloads a set, s/he will be enabled to see a list of

the file names of the source files. To do this, the user will edit the file webworkCourse.ph, and list the appropriate login names.

c)   Verifying the learning

WeBWorK remembers each student's problems, so they can connect to WeBWorK, try to solve a problem, receive immediate feedback about the validity of their answers, try again or logout and give the problem more thought if necessary, and then reconnect to WeBWorK to make a fresh attempt to solve their own individualized problem. Students can attempt a problem as many times as they wish until the due date unless the instructor desires to place a limit on the number of allowed attempts. Each problem in a set can have a different limit on the number of allowed attempts. For example, instructors may wish to limit the number of attempts on T/F questions while allowing unlimited attempts on problems requiring numeric and symbolic answers.

Flexible mechanisms are available for handling numeric, symbolic, and string answers. WeBWorK will accept any correct answer, no matter if it's an elementary function (which the system will evaluate), or a simple number (such as 6, or 4+2, or 3*2) , or a symbolic answer (such as an anti-derivate of a given function), and will think of equivalent answers. For physics problems, WeBWorK can check units attached to numeric answers and make the proper conversions.

String answers allow for T/F, matching, multiple-choice, and short answer questions.

d)   Verifying the use: survey

Teachers can choose to set questionnaires allowing anonymous answers and only one attempt for each student, thus they can create a survey in the course, even if there is not an explicit tool for that.

e)   How much knowledge: automatic correction

WeBWorK increases the effectiveness of traditional homework as a learning tool by providing students with immediate feedback on the validity of their answers and giving them the opportunity to correct mistakes while they are still thinking about the problem.

It increases the efficiency of traditional homework by providing automatic grading of assignments and information both about the performance of individual students and the course (or section) as a whole.

Problem sets are graded automatically, and the resulting scores are easily exported to and imported from spreadsheet programs such as Excel. Moreover, detailed statistical information on the current progress of a class or an individual in completing any assignment is available.

Instructors can do scoring on the internet using the Scoring Tools page. Instructors can either score one set at a time, or get a total score for all the sets they want. The scoring procedure creates files (in .csv format) that include the status of the problems, the number of attempts, and also the final score of the problems. Instructors can download these different files onto their personal computer, too.

There is no fixed procedure of scoring. Instructors have a few options, for instance they can score each problem set right after the due date: this way, they can email their students their own homework grades throughout the semester. The trouble with this option is that, if the teachers score the set before the student has completed it, and then decide to give him/her an extension, any work the student does after the old due date will not be recorded. Once the student has completed the set, they would have to edit the scoring files by hand.

Instructors can also score all of the problem sets at the end of the semester. This way, no problems can possibly arise with extensions and no major editing needs to be done. The drawback is that they will not be able to send emails to their students while semester is in progress.

The final option is a combination of the two. Instructors can score each problem set right after the problem set is due, email their students if they want, and then score once more at the end of the semester. Scoring a set just appends the scores to the totals file so if you score a set twice, it will appear twice in the totals file. Of course, if they do this, they will want to keep it in mind while handling the totals file. It should be noted that once a scoring file is created, student names, sections, and recitations will be taken from the current class-list information. Whenever instructors score a set after the creation of the scoring file, the new information is appended to that file by matching student ID's. If instructors change a student's recitation (or name, etc) but keep their student ID, the new recitation will not be written in that scoring file. If students add or drop the course, this will be handled automatically.

WeBWorK immediately tells a students whether their answers are correct or not, but does not tell them the correct answers. Students are allowed to a new try, until they get the right answer.

f) Organize a class: managing learners group

WeBWorK produces similar but individualized problems for each student. This makes WeBWorK particularly effective in a group learning setting, since students can collaborate without copying. Each student receives a slightly different version of each WeBWorK problem so that they can collaborate in finding solutions, but they still have to find the answer to their own problem individually.

To adding students to a problem set, instructors can choose to assign the set to all users, or to assign it to some subset of users, checking the ones to be assigned, one at a time (instructors must be careful not to uncheck anybody who's already started since all his/her data will be lost).

g) Who sees what: access rights to course and contents

At this moment, there are not available information about this feature.

h) Course creation: authoring tools

Very few WeBWorK professors write their own problems. Most subject material in courses well suited to WeBWorK has already been covered by previous WeBWorK professors, and there is no need to reinvent the wheel.

To select problems, instructors have to go to the Library Browser page, pick one of the libraries and, then, one of the contained directories. After making their choices, they can either change the display mode or the number of problems per page (more problems per page makes it longer to load the page); then they can have a preview of the problems. Now instructors are enabled to go through the problems and mark the ones they want to add to their list: when they will update the page, all the problems marked will display according to their choices.

While editing an existing problem set, instructors can modify several features, i.e. dates, weights and number of attempts.

The 'open date' is when the problems will become available to those who are assigned them. Between the 'open date' and the 'due date' the problems answers will be recorded. After the due date no new answers will be recorded. The answers will become visible to students after the 'answer date'. These three dates must be sequential. After these modifications have been made, instructors can save their work. Once all the dates have been saved, an instructor can edit the list of problems in this set. The order of problems can be changed; any problem can be deleted, but if an instructor accidentally deletes one, s/he'll have to go back to the Library Browser page and find its original location to add it back again.

S/he can also change the problem's weight i.e. how much the problem counts when scored.

Usually students are allowed an unlimited amount of attempts at a problem (which is tagged as "unlim"), but sometimes, if the problems are multiple choice, an instructor wants to set a limit to the number of times a student can try. When finished, to save changes the instructor needs only to press a button.

In order to create a problem template file (i.e. a .pg file), usually instructors start with a .pg file containing a problem similar to the one they want and then modify it. This is easy, since close to almost every occurrence of a problem there is an "Edit it" link. Clicking on this link, they will be taken to an editor where they can edit this problem. This may not be an easy task, since they will have to edit computer code. However, there is a link on the editing page which can help them learn to do this. When writing their own problems, instructors will usually have a few errors here and there. It is usually a good idea to click on the refresh button often to make sure the problem is working.

Writing totally new problems may be difficult for a novice, but even if s/he just wants to change the wording or the variable parameters, it may be worthwhile to save the modified problem to a new directory, as often the best way to learn how problems are written is to make small changes to a bunch of them.

i) Different users, different learning path

WeBWorK increases the effectiveness of traditional homework as a learning tool by providing students with individualized versions of problems, which means that instructors can encourage students to work together, while still requiring that each student develop an answer to his or her own version of the problem.

WeBWorK allows great flexibility in administering individual homework. For example, there are occasions when instructors need to give a student an extension on a problem set, or change their score on a problem, or

change the variables of a problem for a particular student. They can do this by going to the Instructor Tools page and selecting the wished student and set, and then choosing the edit option. To give an extension, they have just to overwrite the due date and answer date. If instructors want to change the variables in a given problem, they have just to modify the "problem seed" for it, and this will give it different variables. The students score on a particular problem can be changed, too.

The flexibility of WeBWorK allows its use by instructors with very different teaching styles.

j)   Time for everything: calendar

WeBWorK allows creating a calendar on the home page of the course web site. This tool also allows users to go back to earlier home pages, in case they missed a news bulletin.

k)   Hidden/in preparation material

The default for new sets is "invisible", so that students can't see what instructors are working on. Once everything is ready for the students to see, instructors can make the sets "visible", just clicking on a checkbox in the Homework Sets Editor page.

l)   Synchronous communication: chat, video conferencing, etc

At this moment, there is not available information about this feature.

m)   Asynchronous communication: forum, topic forum, newsgroup

Sometimes, after instructors score problem sets, they may need to send an email to an entire class (or any subset of the class) reporting individual homework grades, exam grades etc., or to email students with general announcements about their course. This can be done on the Email page, which allows teachers to send personalized emails. The personalized variables are: first and last name, student id number, student's section, student's recitation, student's status (current, drop, withdrawal, audit, etc), student's email address, and student's login name. To save time, instructors may want to save one email to use as a template for future emails. They can save as many as they want, but the one they save as default is the one that appears in the email page when accessed, so it is best if it is the template they use most often.

Additionally, all pages have a feedback button to send an e-mail message directly to the instructor (or whomever the instructor designates). Students find it a convenient way to communicate with their instructor, especially when needing help on a particular point.

The system also provides a discussion group tool to create topic forum within a course, either for all users enrolled, or only for restricted groups.

n)   Other

II.   ROLES

In WeBWorK each user has a status determining his/her ability to navigate within the course: current users can navigate within the course, drop users cannot access the platform, because their username and password aren't recognized by the platform, withdrawal users can access the platform, but cannot navigate within the course, and audit users can navigate within the course, but their actions are inspected and any damage can be prevented
.

a)   Learners

Students have their own personal page, where an help button is provided to give specific instructions and hints.

b)   Authors

Instructors can add many students at once using a class list which is to fulfil the ClassList specification. After editing it (there are many different ways users can build and import a classlist; moreover, there are many

tools on the Internet that allow classlist editing), instructors shall upload the file and import it through the Class List Editor Page. The file imported must have a .lst extension, otherwise WeBWorK won't recognize it as a class list.

If there are only few students in a class, instructor can use the Add Students option and the linked Add Users page. They have just to fill in the information for each student: the only required fields are the student identifier and the login name, but in any case it is advisable to fill in the first and last name, and the no email address, because otherwise the student will not receive emails (However, the student is enabled to enter or modify these fields on her/his own).

The student's progress can be viewed by accessing the Student Progress page. Instructors can either view the statistics on one set for all users, or on all sets for one user. This is a good way to find out how an individual student is doing in the course, or to find out how the class as a whole is doing on a particular set. In the report page there is also a statistical success indicator of how well the students are grasping the concepts for the problem set.

When a student is having trouble with a particular problem it is often helpful to see the version of the problem given to that particular student, making it easy to answer specific questions via e-mail or in person. Instructors can do this by selecting the student and the set and clicking the correspondent button: instructors can enter answers into the student's problem, they will not be recorded. Once they end "acting" as that student, they have just to stop this option to go back to their usual view.

c) Tutors

This user role doesn't exist in WeBWorK.

d) Administrators

Administrators may change users status. When they want to prevent a student's access to a course, instead of deleting him/her they can 'drop' him/her, since deleting them destroys all their data which may be needed later. In order to 'drop' the student, instructors have to access the ClassList Editor page, select the user, and enter the word DROP or the letter D in the status box for the student classlist information. Administrator may want to enter a comment stating the drop date. They can later 'resurrect' the student by changing his or her status to 'current'. In that case, they may want to check that the student has been assigned the problem sets.

e) System managers

This user role doesn't exist in WeBWorK.

f) Other

III. RELATIONSHIP WITH STANDARD

At this moment, there is not any available information about this feature.

IV. USABILITY AND LEARNABILITY

a) Usability test: existence and results

No documentation has been found about this subject.

b) Learnability test

No documentation is available about WeBWorK learnability tests. In spite of this, in the system web site page producers claims:"...In designing the problem programming interface for the professor we seek to emulate some of the successful text processing languages such as TeX and Perl. First, the instructor should be able to use the system in simple ways without learning the entire system. If your problems are similar to existing problems, then you should be able to safely modify these template problems to achieve your goals. Along with this comes the principle of "minimum surprise", once the instructor is somewhat familiar with the system things that should work will work, or at the very least they will fail gracefully. At the other

extreme comes the principle of extensibility: if the instructor understands the system he or she should be able to extend its functionality without rewriting the entire system from scratch.".

c) Accessibility

No documentation has been found about this subject.

d) Multi-language support

WeBWorK makes it easy for user to provide multi-language support for GUI display by utilizing Resource Bundles, i.e. property files with special names.

V. PEDAGOGICAL METHODS

The WeBWorK pedagogical approach is very ductile and compatible with several teaching and learning methods. For example, within the PAREA (help program for teaching and learning research) program WeBWorK was used to study "Male self-efficacy and success in calculus through WeBWorK" (documentation at the internet web address http://doe.concordia.ca/cslp/Grants/RA-PAREA_MSSCW.php). This report tells: "...we will study the impact of three pedagogical approaches to homework assignments: pencil/paper assignments with correct answer feedback; assignments through an online assessment tool, WeBWorK, with correct answer feedback; and, assignments through an online assessment tool, WeBWorK, with elaborative answer feedback...".

On the other hand, on the "Journal of Computers in Mathematics and Science Teaching" appeared an article titled "Student perceptions of the web-based homework program WeBWorK in moderate enrolment college algebra classes" written by S.Hauk and A.Segalla (http://hopper.unco.edu/faculty/personal/hauk/segalla/WBHqualJCMSTpost.pdf). According to this article, "... the educational philosophy underpinning both the development of WeBWorK and this study is constructivist: understandings are conceptual structures built mentally by a learner. (...) WeBWorK may facilitate both factual and conceptual knowledge building for individual learners. (...) What WeBWorK does do is replace the unevenly implemented pedagogical interaction of homework grading with a uniform method of feedback.".

As for other LMS systems, WeBWorK components allow the free (hand-made) construction of courses following different pedagogical approaches depending on the Teacher preferences. The tools provided allows the Teacher to follow Socio-constructivist approaches. WeBWorK has no specific automatic support for pedagogical methodologies, e.g. to course personalization.

VI. INSTALLATION

WeBWorK is available for download on its developer's SourceForge project page.
Installation directly from CVS is also possible.
On the WeBWorK web site there is a installation dedicated section, with a full link list to all the existent documentation about this procedure.

VII. MAINTAINABILITY

In 1999 WeBWorK was awarded the International Conference on Technology in Collegiate Mathematics Award for Excellence and Innovation with the Use of Technology in Collegiate Mathematics. Since that date, WeBWorK was released in several different versions, each of them with improvements requested by users, and implemented by other users or members of system core-team. The WeBWorK team releases an average of three new versions per year, and the latest version was released on 10 June 2005.

VIII. REQUIRED HW or SW

WeBWorK requires a Unix server, and some Unix expertise to set it up. The students can access WeBWorK with any internet browser. Professors can administer much of the course using only an internet browser. A few operations require some knowledge of simple Unix commands.
WeBWorK 2.1 requires PGLanguageRelease2pt1. Both WeBWorK and PG are available on the SourceForge project page.

IX. LICENSING: OPEN SOURCE or PROPRIETARY

WeBWorK is copyrighted by the WeBWorK team and is freely available for non-commercial use by academic institutions. In the system web site there is a section with license explanations (permissions to use, copy, and modify the software and its documentation for educational, research and non-profit purposes, without fee).

X. TOTAL COST OF OWNERSHIP

As just said, WeBWorK is a freely available software, and also other required software are free. Its use doesn't require significant investments in hardware, and in order to use it, users don't need training, if they have a minimum experience with their browser and Internet. Instructors only need to know a few simple Unix commands.

XI. USAGE

WeBWorK is a web-based homework checker currently used to check calculus homework by the University of Rochester's math department and by math departments at other universities. WeBWorK is currently in use by approximately 50 colleges, universities, and high schools. Most of WeBWorK courses are mathematics courses, such as at Arizona University, California State University at Dominguez Hills, Fresno and Long Beach, Central Michigan University, Columbia University, Harvard University, Ohio State University, and, obviously, University of Rochester.
Outside US, WeBWorK is used also at McGill University (Canada), Ming Hsin University of Science and Technology (Taiwan), National Chiao Tung University (Taiwan), Seoul National University (Korea), and Vanier College (Canada). Moreover, more than seventy courses are hosted at the University of Rochester for almost thirty different institutions.
In the WeBWorK web site there is a section showing a simple system performances report, considering as parameters different numbers of users, courses size, operating system, processor, and system memory. For example, at the Indiana University, with 2500 users, largest course size equal to 290 users, a SunOS 5.7 system with a Quad Processor Sun Sparc Ultra-80 and 2 GB of memory, system performances are claimed excellent. On the other hand, at the University of Rochester, with 1100 users, largest course size equal to 300 users, a FreeBSD system with a Dell Dual Processor 500 MHz Intel Pentium III and 512 MB of memory, system performances are claimed good.

XII. SUPPORT QUALITY

WeBWorK entrusts its users to different kinds of support, such as a web community that regroups all users from all courses. Moreover, these users can exchange their ideas and their comments using the discussion groups available in the WeBWorK web page. The WeBWorK mailing list is open to developers and end users. The list is hosted in message-by-message and digest format. An archive of this list is also available from the same page.
Finally, the system provides some brief documents to help users in basic operation, from installation procedure to first steps in set up, from setting up a whole course to editing course environment; and detailed

reference documents for managing course, such as a manual to describe the structure of the headers used to define the format of the TeX output or a report about what the command line script do and how to use them.

Conclusions

We have examined several (16) Learning Management Systems, with the aim of describing their peculiarities and of suggesting ways to improve the current state-of the art. The conclusions stated below are meant as a set of guidelines and suggestions from which the Work Package 6 of the QUIS project will profit, in order to describe and define requirements for the implementation of a "Next Generation LMS".

The main part of the LMS examined choose to leave the teacher free of designing his courses by using his preferred pedagogical methods, without forcing him to follow a given pedagogy.

Thus they tend to propose a wide set of technological tools focused mainly on four areas:
- content delivery, reuse and management,
- class and student management,
- group communication and cooperation,
- self assessment – quizzes.

Such tools allow for the (hand-crafted) construction of courses that follow different pedagogical styles; there are no specific <u>automatic</u> tools available to help the teacher implementing more complex pedagogical settings (e.g. best practices).

It should be noted that the presence of group communication/cooperation tools allows for the application of the Socio-constructivist pedagogy theories.

Some effort towards the personalization of the learning experience is beginning to appear, either by defining different groups of users to which different learning paths/activities are presented, or by "releasing" learning components for student consumption only when a set of rules are satisfied.
PaKMaS is a notable exception, being able to automatically build learning paths in the material; moreover, in PaKMaS the student himself can annotate the learning material and construct his personal learning paths.

The LAMS and Reload systems start addressing the issues of multi-learner personalization by following the "Learning Design" methodology.

Yet, there are no LMS available that use a formal semantic-based approach to the construction, analysis and delivery of learning elements, neither are available model-based tools that adapt the system to a model of the student.

Author team of QUIS reports

Contributions to QUIS reports are produced by staff members at the partner institutions.
All of these persons have taken part in discussions and production leading to this and other reports.
Contact authors for this particular report are listed on the front page.

The activities in the QUIS project will be directed towards QUality in e-learning, Interoperability and reusability of e-learning material and development of Standards. The project will also look at cost beffectiveness in e-learning.

Quality in e-learning is important to be able to exchange both learning materials and learning practices across HEI's in Europe. To establish joint study programs it is essential that cooperating institutions accept each others Quality Assurance Systems (QAS).

Partner	Institution	Country	Staff members contributing
P1	TISIP Research Foundation	NO	Tor Atle Hjeltnes, Thorleif Hjeltnes, Geir Maribu, Arne B. Mikalsen
P2	Norwegian Technical & Natural science University, NTNU	NO	Line Kolås, Arvid Staupe
P3	Mid Sweden University, MIUN	SE	Bertil Andersson, Börje Hansson, Åke Malmberg
P4	Universita' La Sapienza, UoR	IT	Maria de Marsico, Andrea Sterbini, Marco Temperini, Emanuele Panizzi
P5	SZÁMALK Education and Information Technology Ltd., SEL	SEL	Lászlo Kómaroni

Other staff members may have been involved with activities related to the project, course development, dissemination, secretarial work etc, but are not directly involved with the content and authoring of these reports.

QUIS

Quality, Interoperability and Standars in e-learning

www.tisip.no/QUIS/

tisip
- it og forskning