Content Management Middleware for the Support of Distributed Teaching

Hariklia Tsalapatas[1], John B Stav[2] and Christos Kalantzis[1]
[1]University of Thessaly, Volos, Greece
[2]Sor Trondelag University College, Trondheim, Norway
htsalapa@uth.gr
john.b.stav@iaf.hist.no
ckalant@uth.gr

Abstract: eCMS is a web-based federated content management system for the support of distributed teaching based on an open, distributed middleware architecture for the publication, discovery, retrieval, and integration of educational material. The infrastructure supports the management of both standalone material and structured courses, as well as the integration of existing organized external repositories. The infrastructure is complemented with services targeting the specific needs of user groups involved in the e-Learning process, namely learners, content providers, course managers, and repository administrators.

Keywords: e-Learning, distributed teaching, content management, middleware, integration, content reuse, metadata, information discovery, federated, autonomy, services, scalability, 3-tier architecture, structured courses, examination development, statistics analysis

1. Introduction

Asynchronous e-Learning overcomes geographical and temporal constraints transforming learning into a process that can occur at the independently determined convenience of instructor and learner (Harris D., DiPaolo A., Goodman J. 1994). Demand for asynchronous e-Learning has been developing driven both by corporations and professionals striving to remain competitive in a constantly evolving job market and by individuals seeking degrees, typically at the higher education and post graduate level, but facing time and location constraints. Furthermore, content management infrastructures are recently being used as a complementary educational tool in traditional classroom education (e-Learning 2001) for the publication and dissemination of course related material.

The success of asynchronous e-Learning relies on the effective management of the wealth of available educational content. Recognizing the importance of educational content management in distributed teaching, the eCMS project focused on a web-based federated content management system for the support of the distance education process. The system is implemented on state-of-the-art digital library technology for the management of heterogeneous, possibly distributed content located locally or on remote servers and managed by independent content masters. The core of the system is open, distributed middleware for the publication, discovery, retrieval, and integration of educational material. The infrastructure is complemented with services targeting the specific needs of user groups involved in the e-Learning process, i.e. learners, content providers, course managers, and repository administrators. eCMS supports standalone as well as distributed configurations, allowing installations to evolve into educational content portals and ensuring the autonomy of institutions participating in an extended eCMS educational network.

eCMS services for learners include information discovery through metadata keyword searches, navigation through organized content, participation in self-assessment examinations, educational module reviews, and on-line collaboration tools. Services for content providers include optional content hosting, metadata authoring wizards, and support for the on-line development and management of structured courses as well as multiple-choice examinations. The system supports the reuse of content, possibly developed by peers, facilitating the creation of value-adding educational entities under the control of the owning organizations and with respect to the intellectual property rights of content developers. Services for repository administrators include publisher and user management, easy to use educational content metadata indexing for information discovery purposes, and transparent gathering and analysis of system use statistics aiming at the constant improvement of services and published material.

eCMS recognizes that course structuring often reflects the pedagogical practices and internal organization of academic institutions and thus may differ significantly from one organization to another. In addition, educational content contributors are typically professors who have already well developed pedagogical practices, existing material, and limited time to restructure content or teaching methods. Course managers wish to convert material used in traditional classroom settings to on-line content with as few modifications as possible. For the above reasons, success of an on-line educational content publication service depends on its flexibility on course structuring. Rather than enforcing a predefined structure, eCMS allows providers to organize course content in a manner that best fits their academic practices through the support of flexible tree-like course hierarchies with no additional limitations. Course content is developed in private workspaces controlled by the publishers. Thus publishers have the discretion to make content available to the general public when they feel that it is mature. Content can be published either as complete entities, e.g. an entire course including several lectures, handouts, exercises, etc, or in parts. This flexibility is necessary for the support of current teaching practices, which often involve the periodic publication of material. For example, instructors often publish reading lists before a lecture so that students can prepare for class discussion. Instructors publish class presentations after the lecture as reference material. This process ensures that students reach their conclusions through class participation instead of being fed information by the instructor.

On the other hand, while organizations recognize the benefits of on-line content dissemination they are reluctant to publish material through a system that they do not fully control. The federated nature of the eCMS middleware allows institutions to install and locally manage an eCMS node that may optionally be connected to a larger eCMS network. Thus, educational institutions enjoy maximum autonomy through the distribution of metadata and content. eCMS nodes can be installed at the institution or the department level. That is, the separation of content into logical repositories is flexible and can be decided by the educational institutions.

Finally, eCMS supports the integration of existing organized repositories through metadata importing and mapping agents, a feature not currently supported by available e-Learning management systems (Blackboard 2003), (WebCT 2003). eCMS can be configured to periodically import external repository metadata in a manner that is transparent to each repository's normal operations. This feature stresses the federated nature of eCMS networks as opposed to standalone content management configurations.

To facilitate the on-line learning process eCMS offers a basic collaboration environment that enables learner-learner and learner-instructor communication. Given the wealth of relevant software and the fact that eCMS focuses on federated content management infrastructures the collaboration environment is implemented by taking advantage of available off-the-shelf tools.

External learner groups consisting of secondary school teachers and higher education learners in the north and south of Europe have tested the system with positive results. However, the proposed infrastructure is independent of the targeted learner group and may be used in a wide range of e-Learning initiatives, such as higher education, post graduate education, life long learning of professionals, training of trainers, etc.

2. eCMS Educational Content Management Middleware

Figure 1 displays the eCMS educational content management node. Information is logically organized into separate data stores for metadata, content, and system statistics. The business logic is separated from the data and is implemented in a separate service layer to ensure flexibility in educational offerings, software updates, and interface improvements.

The *Metadata Store* holds a description record for each published educational module. The system supports optional hosting of material into a designated *Content Hosting Area*. Material is uploaded into the system through the *Hosting* Service. It is expected that this mechanism is typically used for the publication of standalone educational units originating from institutions that do not currently offer organized distance education programs or the infrastructure and know-how to manage content independently but still wish to publish educational content through external services. Finally the *Statistics Store* holds transparently gathered statistics on system use. Analyzed statistics are available for review by both system administrators and course managers, who may use the information to evaluate their practices and identify points of improvement.

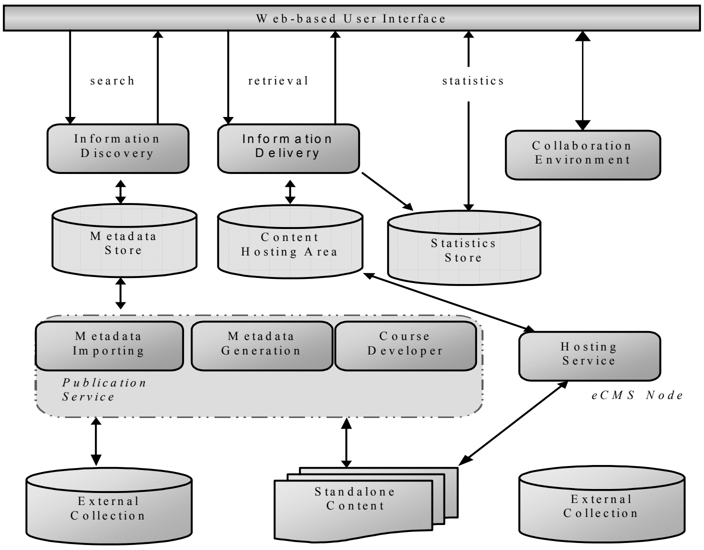

Figure 1: eCMS system architecture

Users interact with the eCMS Content Management Node through a web-based interface providing customized library views targeting the needs of learners, content providers, course managers, and repository administrators.

Content providers can publish educational material in one of the following ways: (i) through the *eCMS Metadata Submission and Editing Wizard*, a facility that guides publishers through the step-by-step generation of metadata descriptions for educational entities. (ii) Through the *eCMS Course Developer*, which supports the publication of structured courses through the development of metadata descriptions for educational modules as well as the definition of parent-child relationships creating tree-like hierarchies of content. (iii) Through the *Metadata Importing Wizard for the* transparent integration of external repository metadata, thus enabling the publication through the system of entire collections of information in a manner that does not interfere with the external repository normal operations.

Learners have access to published content through search and navigation interfaces that support text-based queries against the educational metadata. The system is extensible to support text-based queries against content, for example the content of a text file or a file stored in a proprietary format such as PDF. Content may be retrieved through the *Information Delivery Service* and presented to the learner in an appropriate format depending on the storage method and the intended use of the module (image, text, video, etc).

The *Statistics Gathering and Analysis* service automatically collects statistics on the use of offered services during the regular eCMS operation. Statistics include information on keywords used in queries, metadata records matched through queries, reviewed metadata records, and user information. The goal of this operation is the constant improvement of the system functionality and content, as described in more detail in section *“System statistics”.*

Finally, a collaboration environment developed through off-the-shelf tools facilitates communication between learners (e.g. group work) as well as learner-instructor interaction (e.g. office hours) thus aiding the learning process. The tools consist of chat-rooms, forums, and application sharing modules.

Additional Content Management Nodes are supported to ensure scalability of the system as the amount of managed information increases. Each eCMS node manages a distinct metadata collection and corresponding content. As described in more detail in section *“Support for metadata distribution and distributed queries”*, distributed information is periodically and transparently synchronized to ensure that users have access to the entirety of the distributed content through an interface that hides the underlying system complexity.

The system is designed as a 3-tier web-based application to ensure modularity of offered services and minimum requirements on the user side. The back end (server side) of the architecture serves as metadata, content, and statistics storage. Metadata and content is published on the web through a commercial Application Server. The middle tier implements the business logic as servlets for specific services. The front end (client side) is developed as JSP applications. The 3-tier structure disengages the logic from the interface, which ensures easy adaptation of services and user screens for addressing the needs of additional target groups.

Figure 2 displays the 3-tier implementation. As shown in the figure, independent services and interfaces are developed for each of the identified target user groups that participate in the asynchronous e-Learning process: *Public Services* aimed at learners, *Provider Services* aimed at content providers and course managers, and *Administrator Services* aimed at repository administrators, who manage the eCMS nodes.

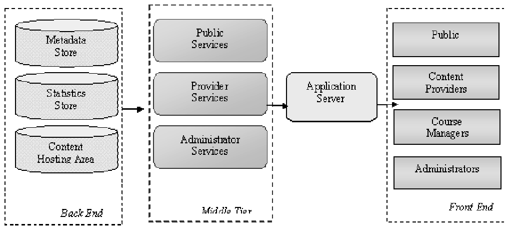

Figure 2: The 3-Tier implementation

3. Course structure

One of the key difficulties and hindrances for publishing content, particularly already existing modules, into e-Learning libraries is the fact that many systems support a very specific course-structuring model. At first glance this may seem as a feature that simplifies the publication process and guides publishers through content development. However, this choice may prove to be a significant disadvantage when academic institutions wish to create distance education programs based on existing content and educational practices. These programs are typically developed by professors who on the one hand have very limited time and on the other have already well developed teaching practices that they wish to transfer to e-Learning courses. Furthermore, in a lot of situations the structuring of courses may represent a larger view on the educational process adopted at an institutional level.

It becomes apparent that for the success of any open federated library for educational content flexibility in the structuring of courses is a vital feature that encourages content publication. For this reason, the eCMS system allows maximum flexibility to content providers on the structuring of material. eCMS achieves this flexibility by imposing the least possible structuring constraints. A hierarchical approach has been adopted. The providers may decide the types of educational modules they will use for course structuring purposes. Examples of currently supported module types include course, module, section, and subsection. However, the supported educational module types are dynamically managed by eCMS administrators, as opposed to being hard-coded into the system, and may be easily extended through straightforward on-line services. Thus, an institution that wishes a different structure, for example lecture driven as opposed to the current book-content index table inspired one, can easily implement it.

Figure 3 displays the representation of structured courses in the eCMS system. In the presented example a number of independent modules are displayed, interconnected with *“contains”* parent-child relationships. Thus, modules published through the system may be reused, provided that the owning organization permits. This example displays two courses: “Computing” which includes “Computer Architecture” and “Desktop Tools” (e.g. a lab), which in turn includes “Imaging”, “Doc Editing”, and “Spread Sheets”. “European History” includes a module for a historic period and is reusing parts of “Computing” to provide learners with basic desktop skills. Reuse of educational modules is a value-adding function that takes advantage of the federated distributed library nature of the eCMS system, which is not available in standalone systems. Information reuse can result in the dramatic increase of content published through the system through combinations targting the needs of different learner groups.

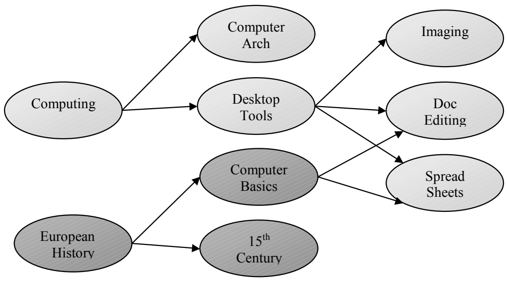

Figure 3: eCMS course structure supporting flexibility and module reuse

4. Educational content metadata

Figure 4 displays a high level overview of the eCMS Educational Module metadata schema. The metadata schema has been developed taking into account earlier work in this area (The Dublin Core), (ETB), (Learning Technology Standards Committee 2000), (Nikolau C., Georgakopoulos G., Tsalapatas H. 2000). Past work on metadata definitions has been extended to support detailed descriptions of structured courses and examinations, as well as course hierarchies.

The information is organized through relational database tables. Each published educational entity corresponds to a record in the Metadata table and is uniquely identified. Information is also maintained on Publishers (e.g. the Department of Computer Engineering) as well as the Institutions the Publishers belong to (e.g. the University of Thessaly). More than one Publisher may be associated with each Institution, thus creating a hierarchy of Publishers in the organization of published educational modules. This hierarchy allows additional granularity in the organization of content in local collections within an academic institution.

Information is organized in Thematic Areas for navigation and information discovery purposes. The supported Thematic Areas are kept in a corresponding table, which is easily updateable through available on-line services for repository administrators. Managing dynamically information on thematic areas allows the creation of user-friendly information discovery interfaces. The value of such flexibility becomes clearer at the face of growing educational repository content leading to the need of effective interfaces for information discovery.

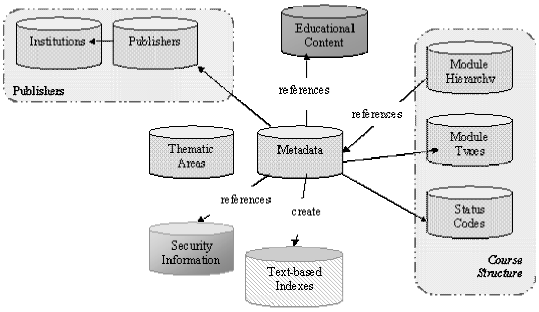

Figure 4: Overview of eCMS educational content metadata schema

The Module Type table holds information on the types of educational modules supported by the eCMS system (e.g. courses, chapters, sections, subsections, etc). The table enables flexible course structures as the update of supported module types and the addition of new ones is dynamic and involves simple database operations through on-line functions for repository administrators. Course structure is maintained through the Module Hierarchy table, which holds parent-child relationships forming tree-like hierarchies as described above.

The Module Status Codes table enables the effective management of educational modules by providing a means of identifying entities that are under construction, completed, public, or available for reuse by other content providers. This information may be used to customize the presentation of information to users. For example, learners may only access public modules whereas publishers have access to their own records but may only reuse modules developed by others if these modules are marked as “available for reuse” by the owning organization. The status code granularity is at the educational module level, as opposed to the course level, implying that parts of a course may be marked as “completed” while other parts may be marked as “under construction” and others as “public” to ensure maximum flexibility in course presentation. Status codes, similarly to educational module types, are dynamically managed through on-line functions, ensuring the straightforward modification of existing and the addition of new ones as needed.

Security information is held in a separate space in the schema and includes user accounts and groups that control access to content and metadata in the system repositories.

In order to support distribution of metadata, a feature that supports the autonomy of participating organizations as well as efficient queries against the entire content space, the eCMS system maintains a directory of available repositories connected into a wider eCMS network. The directory is used for metadata synchronization and caching as described in more detail in section *“Support for metadata distribution and distributed queries”.*

Finally, text-based indexes are built on the Module Metadata to allow free-text search. Indexes are built on a set of metadata fields defined as searchable, while regular SQL queries are supported against the entire metadata record description. Examples of searchable metadata fields include a course’s title, author, summary, and keywords. Non-searchable metadata fields are the ones that describe details about an educational module, such as course prerequisites, class-meeting schedules, recommended bandwidth for accessing the material, required course examinations, etc.

5. Summary of eCMS services for the support of distributed teaching

The following sections provide a high-level overview of the eCMS services implemented on the educational content management infrastructure described in earlier sections. Services are customized for each targeted user group, namely learners, content providers, course managers, and repository administrators.

5.1 Services for learners:

- Querying support:
  - Free-text search, which is the most common and easiest to use information discovery method.
  - Search with options, used for increased search result accuracy by matching user-entered keywords against specific metadata fields as opposed to the entire metadata record description; for example matching particular keywords against an educational module's title.
  - Search by eCMS record ID, useful for the quick access of frequently used records.
- Navigation:
  - By publisher, for example by academic institution or by department.
  - By thematic area.
  - By author or course instructor.
- Collaboration environment:
  - Chat-rooms for real-time quick communications.
  - Forums for the support of discussions.
  - Application sharing, for the collaboration on common projects. Examples include collaborative work (modification and review) on shared files by users in different locations.
- Participation in self-assessment multiple-choice examinations.
- Educational module reviews, which can be useful to other students wishing, for example, to participate in a course:
  - Submission of module ratings on a predefined scale of 1 to 5.
  - Submission of text reviews with the student's opinion on the quality and applicability of a module's content.
- Management of personal record in eCMS:
  - Learner registration to the eCMS system.
  - Personal data and personal eCMS account editing.
- Feedback forms on the eCMS services:
  - On-line eCMS service questionnaire.
  - Email-based feedback.
- Services manual.

5.2 Additional services for educational content providers:

In addition to the above services for learners, the following are available for content providers:

- Private workspaces for content development.
- On-line step-by-step metadata generation and publication.
- Management of educational content metadata (editing, deleting).
- Optional content hosting.
- Access to learner educational content reviews and ratings aimed as an aid to the instructor for the constant improvement of the published material.

5.3 Additional services for course managers:

- On-line structured course developer and editor.
- On-line multiple-choice examination developer and editor.

5.4 Additional services for repository administrators:

- User account and group management.
- Supported educational module type management.
- Institution and publisher management.
- On-line text-based metadata indexing.
- Importing of organized external repositories and mapping of external to eCMS metadata.
- Support for the efficient distribution of metadata and content through remote metadata caching.

The following sections provide more details on key eCMS services and features. Additional information can be found in the eCMS Design and Implementation Reports (Tsalapata H., Brna P., Stav J. 2002, Tsalapata H., Kalantzis C., Brna P., Stav J. 2003).

6. Course developer and editor

The eCMS Course Developer and Editor utilities are part of the eCMS content and metadata publication services. Since they constitute an important aspect of the eCMS system, they are discussed as separate entities.

6.1 Course developer

The eCMS Course Developer facilitates the on-line construction of structured courses. This value-adding functionality enables providers to make the most of published content by supporting the creation of new structured entities either using new modules or based on already published content. The service complements the eCMS Metadata Submission and Editing Wizard for the publication of standalone educational modules to create a complete on-line content authoring environment.

To ensure maximum flexibility on course structuring the eCMS Course Developer adopts a recursive top-down construction approach for the creation of tree-like hierarchies. In addition, the tool supports the possibility of information reuse. The approach is displayed in Figure 5. To gain access to the service, providers must log into the eCMS system with their user account *(1)*. The first step of course creation is the submission of a general metadata description for the course that documents the course title, the thematic area, the covered topics, the targeted audience, the required background on the learner's behalf, the required effort for completing the course work, examinations, hand-ins, instructor availability, and other information. Subsequently, providers are requested to add educational modules to the course.

Course sub-modules may be:

- New entities that the provider creates recursively at the time of course creation, or
- Existing entities already published through eCMS, which the provider reuses.

In the case of a new entity, the provider creates a metadata description for the module *(3)* and either associates the module with a remote content address or optionally uploads the corresponding content into the eCMS repositories.

The provider may reuse an already published entity as a sub-module of the course under development by selecting the entity's title from a drop-down list *(3)*. Reusable entities may be:

- Entities that are owned by the provider, in which case the provider has by default full rights, or
- Entities that are owned by other providers and have been marked by their owners as "available for reuse".

Once the sub-module has been created a parent-child relationship between the existing node and the newly created sub-module is automatically inserted into the eCMS Course Structure area, extending the course hierarchy.

The above process is repeated recursively in a depth-first manner until the course is completed. Upon completion the provider may review the structure and use the Course Editor to make any necessary changes/improvements.

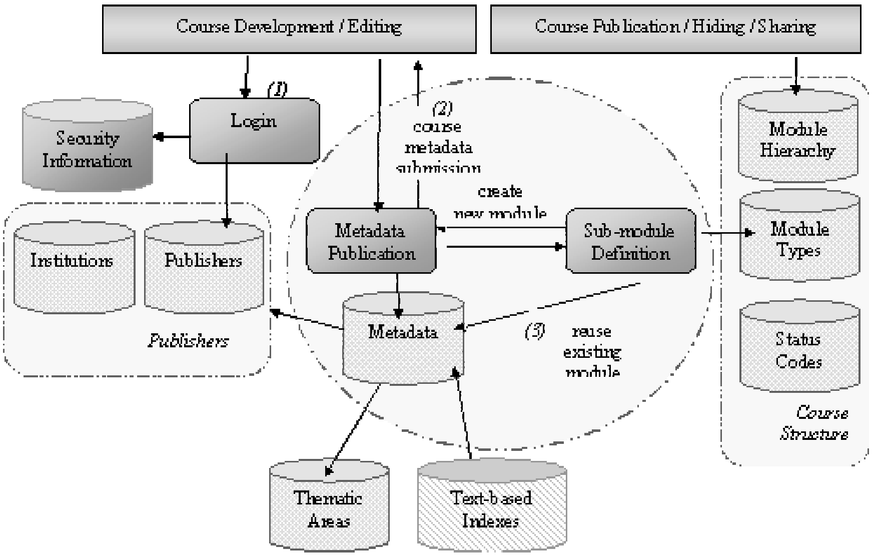

Figure 5: Course developer and editor

6.2 Course editor

The eCMS Course Editor complements the Course Developer to form a complete structured-course authoring tool. Through the Course Editor providers may perform the following operations:

- *Edit the metadata description of a node in a course hierarchy*: Both the top level "course" description and the inner "course module" descriptions can be edited.
- *Upload new content and associate it with a node in a course hierarchy*: This function allows the update of the content itself, and complements metadata updates.
- *Publish a node*: This function moves a node and all its children to level "public". Upon completion the sub-tree rooted at the node is visible by the general public.
- *Hide a node from the public*: This function moves a node and all its children to level "complete". Upon completion the sub-tree rooted at the node is visible only to the owner / publisher.
- *Allow reuse of a node*: This function moves a node and all its children to level "reuse allowed". Upon completion other publishers may reuse the sub-tree rooted at the node for the development of value-adding educational entities.
- *Delete a node*: This function deletes a node and all of its children. Also, it deletes all parent-child relationships between the node and its children and between the node and its parents and repeats this operation recursively for each child. Upon completion the sub-tree rooted at the node is deleted.
- *Detach a node*: This function detaches a node, i.e. removes the corresponding parent-child relationship. The difference with the *delete* operation is that no nodes or other relationships are deleted. Figure 6 displays the result of detaching Node B from Node A.

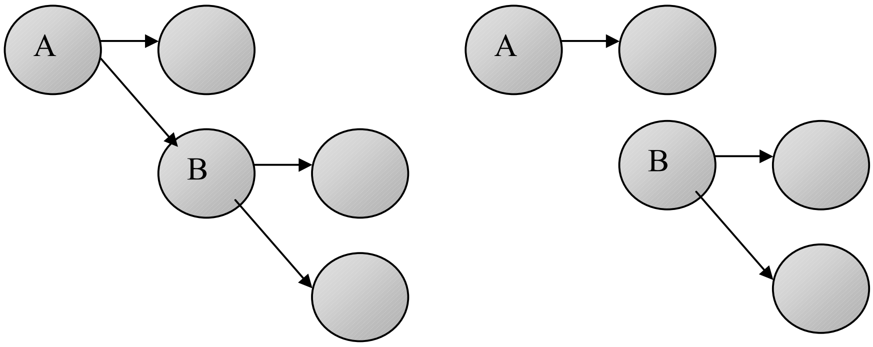

Figure 6: Node Detachment

- Add a node: This function inserts a new node in the course hierarchy at a specific location. The new node can be added before or after a specific node in the existing course hierarchy. Figure 7 displays the result of adding Node C before Node B.

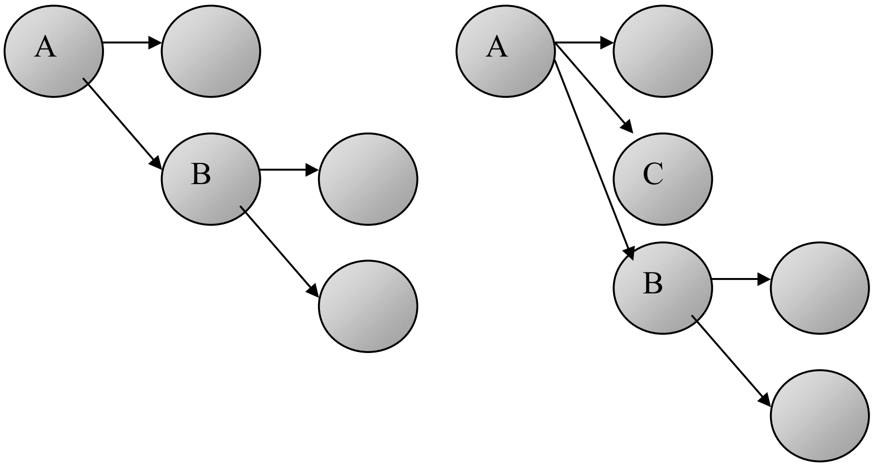

Figure 7: Node Insertion

All actions described above are allowed on nodes that are owned by the publisher. During the recursive execution of the operations on a sub-tree only nodes that are owned by the publisher are affected. Nodes that are part of the sub-tree and are not owned by the publisher, i.e. they are reused, are not altered in any way.

7. Examination developer and editor

The eCMS *Examination Developer* and *Editor* is an on-line authoring tool that supports course managers in the development of multiple-choice self-assessment examinations for learners. The examinations may or may not be associated with courses or other content published through eCMS and may be incorporated into a course hierarchy.

In a manner similar to the one used for course development, examination development is achieved through a top-down approach. A general description of the examination is created first, in which the developer may document the scope of the examination, the knowledge being assessed, the educational material it is associated with, and other information. An eCMS metadata record is automatically created for the examination and may be used as a handle for optionally incorporating the examination into a course hierarchy, as mentioned above.

Once the general examination description is complete, the eCMS Examination Developer guides the user through the addition of multiple-choice questions to the examination through the *Examination Question Wizard*. For each question, the wizard prompts for the following information:

- The question text, to be presented to the learner taking the examination.
- Up to 5 possible answer texts.
- The correct answer.
- An optional hint for the correct answer, to be presented to the learner.
- An optional eCMS address (URL) of the educational module where the correct answer may be located, to be presented to the learner.

This process is repeated for each question to be added to the examination.

The Examination Editor supports the editing of examination metadata description fields, the deletion of questions, and the insertion of additional questions through the Examination Question Wizard.

8. Support for metadata distribution and distributed queries

In order to ensure autonomy of participating organizations, eCMS supports the installation of more than one independently managed eCMS nodes optionally interconnected into a federated digital library of educational content. Thus, institutions may install an eCMS node in their premises and manage it through the provided Repository Administrator services. Remote eCMS nodes may be easily connected into a wider eCMS network through the support for *Metadata Distribution* and *Distributed Queries*. This feature ensures that publishers may get the best of both worlds: autonomy in the management of the content, scalability with respect to metadata volume, and participation in educational networks.

eCMS hides the complexity of the federated content management infrastructure from users, who may submit a query through an eCMS entry point and receive merged results of query hits from all eCMS nodes. Users may review query results from a single list without being required to know details about the nodes that manage individual records.

Two approaches were considered for the implementation of metadata distribution and distributed query support:

- *Distributed queries*: This approach involves the propagation of a user query, which has been entered through an eCMS entry point, to all eCMS nodes, the execution of the query locally at each eCMS node, the collection of query results from all nodes, the merging of results, and the presentation of a single, merged result list to the user.
- *Metadata caching*: This approach involves "synchronization" of eCMS metadata repositories. Specifically, it involves caching of remote eCMS metadata, which results in equivalent eCMS nodes, i.e. into eCMS nodes that hold the same metadata information, some of which is local and some replicated from remote servers.

The second approach, i.e. metadata caching, has been implemented in the eCMS system to alleviate the need for remote queries, thus resulting in faster query responses.

The *Repository Synchronization* service is available on-line to repository administrators, who may invoke it through a simple click of a button. It may also be programmed by administrators to be executed periodically, e.g. nightly. Repository synchronization is achieved through the following steps:

1 Identification of the addresses of all remote repositories connected into an eCMS network.

2 For each remote repository:

   a Deletion of the locally cached information that originates from the specific repository.

   b Importing of the remote repository's metadata records.

   c Importing of the remote repository's course hierarchy structures.

To facilitate the above actions eCMS stores, for each metadata record, information on the eCMS repository of origin, i.e. the repository where the record is stored and managed. Other repositories simply cache copies of the original records, thus ensuring the integrity of metadata information.

9. External repository integration

External organized educational content repositories, which have their own educational metadata definition sets, may be integrated into eCMS through the Metadata Importing Wizard*.* The wizard operates transparently to the external repository's normal operation as a background demon that may be configured to poll the external repository periodically, e.g. nightly or weekly. The wizard imports external metadata records, maps the metadata to the eCMS metadata definition set through a mapping scheme that is specific to the external repository, and stores the mapped metadata into the Metadata Store as eCMS records that are subsequently indexed through the eCMS metadata free-text indexer.

This functionality requires cooperation between the eCMS and external repository administrators, who must provide the mapping of the external repository metadata fields onto the eCMS metadata definition set.

This functionality allows eCMS installations to evolve into e-Learning portals for the access of information originating from independent sources.

10. System statistics

Gathering and analysis of statistics can lead to valuable feedback on the system infrastructure, the organization of content, and the costs and benefits of the provided services to the users. For this reason, in addition to gathering user input through questionnaires and email, statistics gathering and analysis mechanisms have been developed that operate transparently and gather information on system use.

Gathered statistics currently include the following:

- Query keywords entered by users, for both basic search and search with options
- eCMS identifiers of records matched by queries
- Reviewed metadata records of educational modules
- User information
- Date of statistics record entry

The above information is gathered by automatically inserting records into a designated Raw Statistics area of the Statistics Store each time a user enters a query or reviews an educational module. Once analysed, the statistics information is moved to a corresponding Analysed Statistics area of the Statistics Store. The analysed statistics provide views and summaries of the gathered raw information in a format that can be easily interpreted by repository administrators and course providers and can provide valuable feedback leading to system and service improvements, such as more accurate metadata information and interfaces that are easier to use.

Table 1 displays examples of analysed statistics and possible improvements they can lead to.

Table 1: Analysed statistics and resulting possible system improvements

Analysed Statistics	**Possible System and Service Improvements**
Occurrences of a particular keyword in basic free-text queries	Adjustment of the navigation interface according to frequently used keywords.
Occurrences of a particular keyword in particular fields for queries with options	
eCMS identifiers of matched records	Rarely matched records may imply unintuitive metadata descriptions. The metadata descriptions of such records may have to be modified to ensure easier discovery.

Analysed Statistics	Possible System and Service Improvements
Reviewed metadata records	Rarely reviewed metadata records may imply reduced interest of users for the corresponding content. Such content may have to be adjusted to better meet user needs.

11. Conclusions and future work

This paper presented the eCMS educational content management system for the support of asynchronous e-Learning. The purpose of the system is to provide an open scalable platform for the publication, management, and dissemination of possibly distributed, heterogeneous educational material developed by independent educational content providers while maintaining the autonomy of participating organizations. In addition to providing a platform for the publication and management of educational content, the system provides services for all user groups participating in the asynchronous e-Learning process, namely learners, content providers, course managers, and repository administrators.

The presented architecture has been implemented in the context of MINERVA-SOCRATES action project *“Content Management Middleware for the Support of e-Learning across Europe”* (eCMS 2001).

Current work focuses on services for the support of distributed teaching in engineering and natural sciences, which addresses the additional challenges for the development and dynamic representation of media rich educational content including mathematical and other formulas, graphs, sketches, etc through the latest standards including MathML XML, and CML. This work is currently being implemented in the context of the MINERVA-SOCRATES action project *“Content Management and Collaboration Environment for Natural Sciences”* (Stav J.B. and Tsalapatas H. 2003, NS-eCMS 2003).

References

Blackboard (2003) [on-line] http://www.blackboard.com

eCMS (2001): Content Management Middleware for the support of e-Learning across Europe, MINERVA-SOCRATES funded project 90613-CP-1-2001-1-GR-MINERVA-M, [on-line], http://e-Learning.noc.uth.gr

e-Learning (2001) [on-line] ftp://ftp.cordis.lu/pub/documents_r5/natdir0000001/s_1530009_20010125_143306_GPRC001349en.pdf

ETB: European Treasury Browser, [on-line] http://www.eun.eun.org

Harris D., DiPaolo A., Goodman J. (1994), ADEPT proposal, [on line], http://stanford-online.stanford.edu/history/fulldesc.html

Learning Technology Standards Committee (2000). Approved Working Draft of the IEEE Learning Technology Standards Committee (LTSC) Learning Object Metadata Working Group. IEEE P1484.12 http://ltsc.ieee.org/doc/wg1/ltsa/ltsa_05.pdf

Nikolau C., Georgakopoulos G., Tsalapatas H. 2000, A Federated Digital Library for Universities' Grey-Literature: Practical Issues and a "Transient" Analysis. University of Crete Technical Report

NS-eCMS (2003): Content Management and Collaboration Environment for Natural Sciences, MINERVA-SOCRATES funded project 110159-CP-1-2003-1-NO-MINERVA-M, [on-line], http://ns-ecms.noc.uth.gr

Stav J. B. and Tsalapatas H. (2003), Open, Dynamic Content and e-Learning Management Infrastructure for Engineering and Natural Sciences, to be published at the Electronic Journal of e-Learning (EJEL), *http://www.ejel.org*, June 2004.

The Dublin Core, [on-line] http://purl.org/metadata/dublin-core

Tsalapata H., Brna P., Stav J. (2002) “eCMS System Design Report”, [online], University of Thessaly Technical Report, http://e-Learning.noc.uth.gr/docs/DRV01.doc

Tsalapata H., Kalantzis C., Brna P., Stav J. (2003) “eCMS System Implementation Report”, [online], University of Thessaly Technical Report, http://e-Learning.noc.uth.gr/docs/IRV01.doc

WebCT (2003) http://www.webct.com