DLNET: A Digital Library Architecture for Lifelong Learning

Vara Prashanth Pushpagiri
Alexandria Research Institute
Virginia Tech, USA
E-mail: vpushpag@vt.edu

S. Rahman
Professor, Alexandria Research Institute
Virginia Tech, USA
E-mail: srahman@vt.edu

Abstract

The Digital Library Network for Engineering and Technology (DLNET) is a part of the National Science Foundation's digital library initiative (NSDL) and is envisaged as a repository of learning resources that will facilitate the "lifelong learning" of the engineering community. It will provide technologies to build and maintain a quality collection of engineering resources thus bridging the gap between "the knowledge developer" and "the learner". This paper presents a Learning Object Model for representing engineering resources and a prototype system implementation of a component-based architecture for a modern-profile digital library conformant to the Open Archives Initiative.

1. Introduction

A number of initiatives have been proposed to explore learning technologies for online education. Many of these initiatives, planned towards continuing education have gained popularity of late. One such effort to provide quality-assured learning material in engineering and technology related areas is currently underway at Virginia Tech. The project entitled Digital Library Network for Engineering and Technology (DLNET) is an endeavor to provide life-long learning for students, faculty, practicing engineers and technical professionals of the engineering community. DLNET is an initiative of the National Science Foundation's distributed digital library initiative (National Science, Technology, Engineering and Mathematics Education Digital Library, NSDL)[5,10]. It is a collaborative effort of four institutions, namely, the American Society of Engineering Education (ASEE), the Institute of Electrical and Electronics Engineers (IEEE), Iowa State University and Virginia Polytechnic Institute and State University (Virginia Tech). DLNET provides technologies for developing, discovering and delivering learning programs. The design and implementation have been carefully tailored to ensure portability of the library in all aspects with much emphasis on using the latest technology to portray itself as a modern profile digital library, being more a metadata-centric service provider (and hence a more learner-centric service provider) than being a teacher-centric service provider.

This paper provides an overview of DLNET's approach to the learner-centric model. Section 2 delves into our learning object model and its inception and representation and characteristics. Section 3 provides an overview of the DLNET system architecture where various components of the architecture are described and the advantages of having a component-based architecture are stressed followed by summary and conclusions.

2. Learning Object Model

Digital libraries are all about hosting and serving digitized content. In a library such as DLNET, knowledge providing content comes in a variety of sizes and formats. Such content can vary from being as small as an image to being as large as an application, from being as simple as a sequence of hypertext markup language pages to being as complex as a course module with audio-video components. When such materials are made available over the Internet, a suitable technology must be provided which would enable users to perceive the learning value of the content, i.e. help them gain knowledge about the learning resources upfront without actually playing the resource. This aids users in filtering irrelevant learning resources without difficulty. Such information referred to as metadata [7] provides additional information about a learning resource and can figuratively be considered to be the face of the content. The objective of metadata of providing information about the content would be fulfilled only when metadata is transported along with the content to every platform the content resides on. It becomes logical now to provide some packaging scheme [3,4,6] that would transport metadata along with data. Such a packaging scheme also makes possible a methodology wherein learning resources can be re-used during the creation of other learning resources. We at DLNET have proposed one such representation of learning objects. Our learning object definition follows:

A learning object (LO) is defined as a structured electronic resource that encapsulates high quality *information in a manner that facilitates learning and pedagogy. It has a stated objective and a designated audience. It has ownership and associated intellectual property rights.* [1]

DLNET learning objects [1] have semantic and syntactic properties; have a pre-defined objective, targeted audience and provide the learner with a quality learning experience. These learning objects have ownership and quality as requisite attributes. DLNET learning objects are bounded by copyright issues, as is any other publication material. Any learning resource generated by reusing a learning resource must be referenced and credited appropriately. Learning resources are subject to quality considerations as well. Quality relates to the following facets:

– Subject matter accuracy and authenticity,
– Pedagogical effectiveness or the educational value,
– Relevance of the information in a resource relative to its objective,
– LO features that represent post-publication usefulness such as easy discovery and use/re-use by learners and teachers,
– Technical "soundness" of LOs.

Learning Objects are digital constructs comprising of information and behavior and are analogous to object-oriented modeling elements. Such an analogous representation eases the understanding of how LOs will be packaged, processed and archived in a digital library. DLNET provides tools and interfaces to assist in this process. A tailor-made learning object development tool can be used by *knowledge developers* to package content.

The tool provides user-friendly interfaces and assists in gathering metadata and organize resources. The LO development tool provides a level of abstraction which waives the need for an author to have knowledge about metadata and XML. A sequence of steps provided in the contributors guide helps in metadata harvesting and resource aggregation. The tool then validates the resource by checking and collecting dependent resources required for smooth operation of the resource. This resource validation has been implemented for hypertext resources only. A ready to go *package interchange format* file is created that symbolizes completion of the development phase of the learning resource. This completes the first of six possible lifecycle stages we envision for a learning object. The second stage involves cataloguing, annotating and ranking the learning resource as part of the review process. The ranked content is then indexed and stored (Archive process) making it accessible via the search process. Such resources can then be used as a part of the learning/teaching process or can be reused as a part of a new learning resource. A schematic overview of the lifecycle of a learning object is shown in figure 1.

Before continuing with the learning object model, let us look at the metadata requirements for the learning object. Metadata as we all know is defined as data about data. With the introduction of XML as a standard for representing and packaging structured content, [7,8,9] we have seen a number of recommendations emerge that propose novel ways of representing metadata. One such organized representation is the IMS metadata and content packaging specification [4]. We have chosen to adopt this specification with some tailor-made modifications to suit DLNET requirements [2].

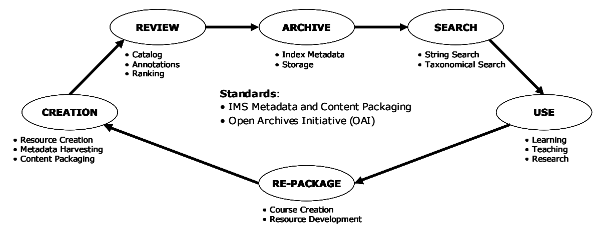

Figure 1. Learning Object lifecycle.

A wide-range of metadata can be collected to assist a variety of users. Metadata often acts as a catalogue–card for the content. Hence, any and all metadata that can compliment the discovery of a learning resource is very helpful. Table 1 shows some of the metadata elements we use in the cataloguing process.

Table 1. Metadata Elements of Interest.

Category	Metadata Element
Information about the Contributor	• Author Name • Email Address • Department • Organization
Basic Metadata	• Title • Language • Description • Keywords • Date of Creation
Educational/Pedagogic Information	• Interactivity Type • Learning Resource Type • Interactivity Level • Intended User • Context of Use • Learning Duration
Other Additional Information	• Format • Rights • Classification

2.1. Learning Object Representation

A visual representation of a DLNET learning object is shown in Figure 2. As seen in the figure, a learning object is represented in a frameset called the preamble. The preamble serves as the face of the learning object by displaying metadata information to the learner. All learning objects are initialized using this file. The top frame is used for playing the learning object while the bottom frame is reserved for navigational purposes.

Learning Object representation in a file system is "tree-

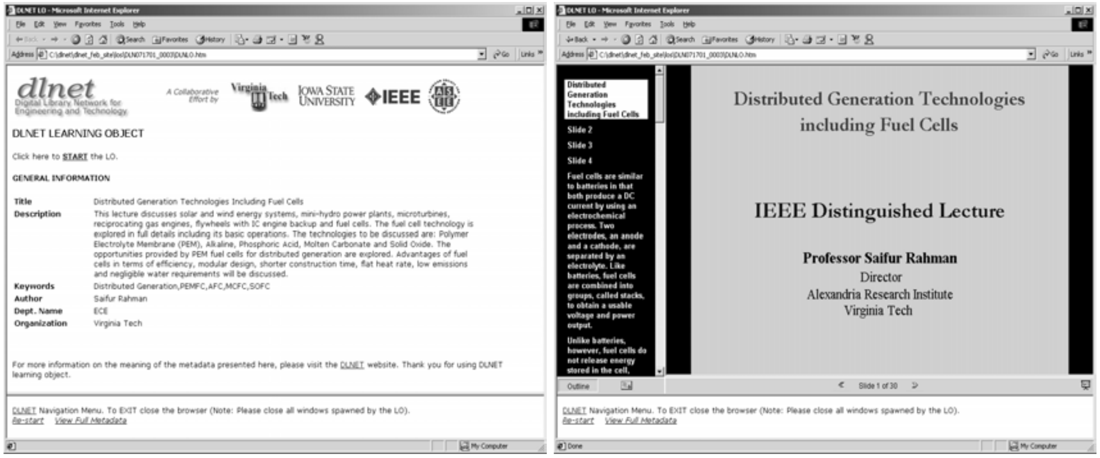

Figure 2. Visualization of a DLNET Learning Object.

based" to facilitate scalability and organization. The learning object can be viewed as comprising of four clusters, a *manifest* file, a *preamble* file, a *support files* folder and a *resources* folder. The *manifest* file (appears as imsmanifest.xml in the top LO folder) contains one or more manifests pertaining to one or more learning resources, metadata comprising of at least the basic set illustrated above, organizational details of resources present in the learning object, etc. We have adopted an extended IMS Content Packaging Specification (v.1.1.2) [4] for representing learning object metadata. The *preamble* file, as mentioned before, is an html frameset that initializes the learning object. It displays the metadata from the manifest file, thus providing the learner with pedagogical information about the resource. The bottom frame controls the learning object presentation and is used for navigational purposes. The *support files* directory provides additional files or methods that may be required for proper functioning of the LO. The *resources* folder is a package containing resources that provide the learning experience to a learner. The entities in this directory can be items or collections and may provide their own metadata. Individual collections can be treated as embedded learning objects if they provide their own manifests. A sample learning object file structure is shown in figure 3.

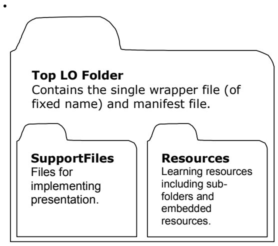

Figure 3. Learning Object file structure.

3. DLNET System Architecture

The DLNET architecture can best be described as one that bridges the gap between the "learner" and a "knowledge developer". Acting as the intermediary between them, DLNET provides learning technologies towards continuing education. It acts as a repository that stores and distributes quality educational material in engineering. The system architecture can be organized to serve four categories of users, *Knowledge Developers*, *Reviewers*, *Learners* and the *NSDL community*. DLNET provides interfaces for each of the four categories of users to perform supported functions at the portal. Figure 4 provides a more elaborate representation of the architecture. Functionalities provided for each group are enumerated below:

Knowledge Developers

The knowledge developer is one of key groups necessary for providing continuing education. This group of users comprises of specialized members who share research and teaching material at DLNET. A knowledge developer can make use of the tool to package content for submission to DLNET. Since a degree of abstraction has been imparted into the tool, developers need not be aware of the rather complex XML specifications and syntaxes. Once a package has been created, content can be uploaded at the DLNET website. To contribute however, the developer will have to be registered as a contributor at DLNET. A simple background check follows the signup process that ratifies the validity of the contributor following which the contributor's content would be processed. We are currently in the process of collecting URLs of engineering resources residing elsewhere. Although these resources are not archived at DLNET (except for the metadata), the method opens a window of opportunity for us to collect resources hosted elsewhere.

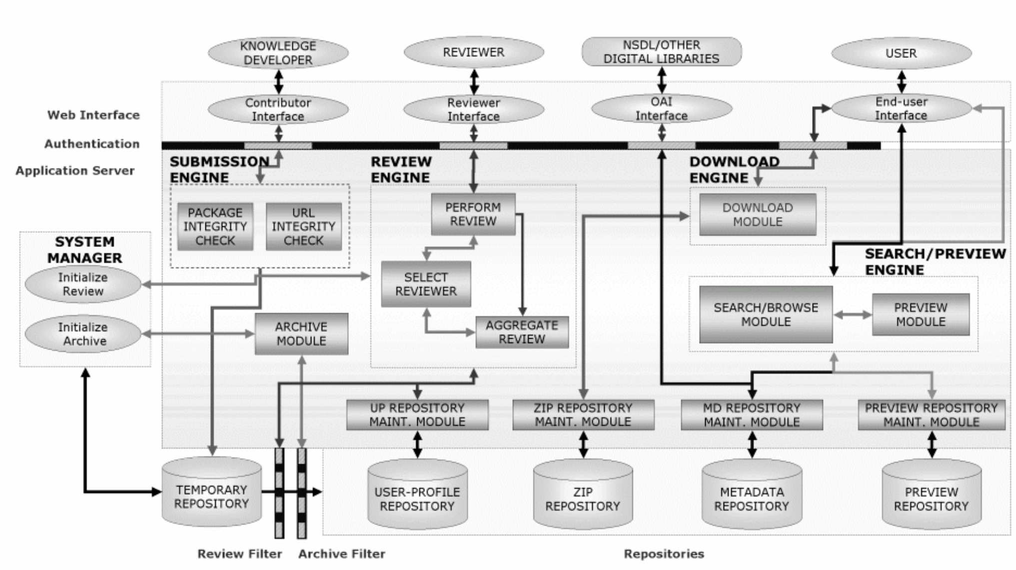

Figure 4. DLNET System Architecture.

In this case, the author of the material must provide metadata for the resource. DLNET will process such resources in a manner similar to learning objects and provide metadata-based search functionality. The submission engine handles submissions from knowledge developers. It verifies the integrity of a learning object or URL before storing it in a temporary repository.

Reviewers

Reviewers are a category of users that have volunteered for peer-review of content submitted to DLNET. Like knowledge developers, reviewers perform specialized tasks. They evaluate the learning objects for pedagogical effectiveness, quality, ease of usage, suitability and conformity to the area of submission. Reviewers are selected by an apt algorithm, which matches the content classification (specified during learning object creation/URL submission) with the area of specialization of a reviewer (specified during registration). Three such reviewers are selected and contacted about an opportunity to review. The review engine interacts with reviewers in collecting reviews. The system manager that polls the temporary database for new/under-review content is responsible for choosing reviewers and re-allocating reviewers if necessary. The algorithm avoids overloading reviewers over periods of time. A simple user-friendly form collects review and comments for a learning resource. The feedback from the reviewers is aggregated and analyzed by the review engine and a rank assigned to the learning object. This rank weighs the pedagogical effectiveness of the resource. The developer of the content is then notified of review completion and asked for permission to archive the resource. The content is then archived and catalogued by the system manager making it publicly available. The metadata, compressed content and preview content are stored in designated repositories that can be accessed via the appropriate maintenance modules.

NSDL Community

The NSDL community comprises of federation of digital libraries including the NSDL portal. DLNET plans to share its collection-level and item-level metadata resources using the Open Archives Initiative (OAI) Protocol. If need be, a screening mechanism will be developed to grant special access to these federations. Partner digital libraries from NSDL domain will have access to DLNET's metadata repository thus harvesting DLNET's metadata.

End-Users

The end-users in a digital library comprise of all users seeking a learning experience from resources. We classify them as "learners". Learners are the primary end-users of the digital library who seek knowledge via the learning resources. DLNET provides easy-to-use interfaces for users to search, preview and download learning objects. Each of these functionalities is explained further below:

Search Functionality. DLNET provides its users with three types of search functionalities, a basic keyword search, an advanced search and a browse search. A keyword search is executed by collecting the search query and parsing it for keywords. Stop words are dropped and stemming is done wherever possible. Once this is complete a sequence of queries is executed (either in a sequential manner or simultaneously) using sets of stored procedures and results are obtained from the database containing indexed metadata. The results of the search are sorted in the order of degree of match and precedence, and presented to the user. The user then has the option of viewing the metadata, previewing or even downloading the learning object. A similar functionality is provided for advanced search, but with better precision. Currently, we support advanced search based on author, title, description, keyword and format. We plan to expand this functionality to enable search by interactivity type, intended user, context, semantic density, difficulty and other metadata fields.

Browse Functionality. DLNET provides its users the opportunity to browse through its collection using a pre-defined taxonomy. A three-level taxonomy has been designed to classify engineering fields into two subgroups. This provides a user with the unique opportunity of searching for learning resources based on taxonomical classification. Future plans include making available a combination of advanced/keyword search along with browse functionality.

Preview Functionality. DLNET provides the opportunity to preview a learning object before deciding to download the entire learning object. For this purpose, a preview repository has been provided with direct access even to unregistered members to preview learning objects. The extent of preview supported is more useful in a business model of the digital library. Although DLNET has no intentions of providing a business model for its architecture, care will be taken to leave plug-in modules to explore such a possibility.

Download Functionality. The download functionality is provided in DLNET so as to enable users personalize their learning experience and even to perhaps re-use parts of the learning module in creating new improved learning objects. The requirement however for downloading a learning object is that the downloader must be a registered DLNET user. DLNET stores a compressed copy of the learning object in the zip repository for this purpose.

DLNET's platform independent modular architecture has been made possible by using java components. DLNET's web service is provided by Apache's Tomcat Service running on Microsoft's IIS web server. The front end and user interfaces are provided by a combination of cutting-edge technological elements. Java Beans™, JSP and Java Servlets are the primary components that provide front and back-end support. In addition a powerful database service (SQL Server 2000) aids in information retrieval.

4. Summary and Conclusions

Digital libraries are on the forefront of providing education with a difference. Whether it be towards technological development, lifelong learning or for institutional learning, these libraries hold the key to the future of electronic learning. We envision DLNET to be a major contributor of learning resources to the lifelong learning of the engineering community. To realize this dream requires a sizeable collection of high-quality learning resources hosted on a platform that provides friendly interfaces for content development, information retrieval and an amicable learning environment. The learning object model we have proposed provides a way of packaging content that can be easily read and indexed thus assisting in information retrieval. This model fits perfectly into the component-based prototype architecture we have proposed. The architecture provides friendly interfaces to assist knowledge developers, reviewers and end-users alike. To add to this, the platform independent nature of the implementation allows the system to be ported to a variety of domains. An alpha-version of the system has been tested and further improvements are currently in progress. These characteristics make DLNET a prospective library for lifelong learning programs and extended electronic learning environments.

5. Acknowledgement

The work reported in this paper is supported in part by a grant from the US National Science Foundation, grant DUE - #0085849 under the NSDL initiative.

6. References

[1] DLNET Team, "Learning Object Definition", *White Paper*, http://www.dlnet.vt.edu/, Oct 2001.

[2] DLNET Team, "DLNET Schemas for Metadata and Content Packaging", *White Paper*, http://www.dlnet.vt.edu/ , Nov 2001.

[3] IMS Content Packaging Specification, http://www.imsproje ct.org/content/packaging/index.html.

[4] Kahn, R. and Wilensky, R., "A framework for distributed digital object services", cnri.dlib/tn95-01, http://www.cnri.reston .va.us/home/cstr/arch/k-w.html, 1995.

[5] "Pathways to Progress: Vision and Plans for Developing the NSDL", http://www.smete.org/nsdl/meetings/grantees0901/whitep

aper.pdf, March 20, 2001.

[6] "Sharable Content Object Reference Model (SCORM)", http://www.adlnet.org/Scorm/scorm.cfm

[7] Thomas D. Wason, "Dr. Tom's metadata Guide", http:// www.imsproject.org/drtommeta.html

[8] Thomas D. Wason, "Dr. Tom's XML Guide", http://www. imsproject.org/drtomxml.html

[9] Thomas D. Wason, "Dr. Tom's Guide to XML Extensions and Incorporations", http://www.imsproject.org/drtomextensions .html

[10] William Y. Arms, Diane Hillmann, Carl Lagoze, Dean Krafft, Richard Marisa, John Saylor, Carol Terrizzi, Cornell University. A Spectrum of Interoperability: The Site for Science Prototype for NSDL. D-Lib magazine, http://www.dlib.org/dlib/january02/arms/01arms.html, January 2002.