macromedia® white paper

Creating Learning Objects With Macromedia Flash MX

by Tanya Heins and Frances Himes

April 2002

The information contained in this document represents the current view of Macromedia on the issue discussed as of the date of publication. Because Macromedia must respond to changing market conditions, it should not be interpreted to be a commitment on the part of Macromedia, and Macromedia cannot guarantee the accuracy of any information presented after the date of publication.

This white paper is for information purposes only. MACROMEDIA MAKES NO WARRANTIES, EXPRESS OR IMPLIED, IN THIS DOCUMENT.

Macromedia may have patents, patent applications, trademark, copyright or other intellectual property rights covering the subject matter of this document. Except as expressly provided in any written license agreement from Macromedia, the furnishing of this document does not give you any license to these patents, trademarks, copyrights or other intellectual property.

Macromedia, Inc.
600 Townsend Street
San Francisco, CA 94103
415–252–2000

Contents

The acceptance of standards as a foundation for creating and reusing content offers today's instructional developers a challenge in selecting appropriate authoring tools. To date, it has not been easy to find the right tool – one that enables easy development of high quality instructional content for integration into various standards-based management systems, as well as enabling reuse in different contextual situations.

Macromedia is committed to facilitating support for authoring Learning Components, or Objects, across key products used for learning. In March 2002, Macromedia Flash MX, was launched. This new version of Flash offers customers the capability of authoring Learning Objects that support key standards. Now, eLearning content developers can use the fully integrated development environment of Macromedia Flash MX to combine a variety of media and external applications to build rich, immersive online learning experiences.

In this paper, the first of several that Macromedia will post on its Flash MX, the authors describe how instructional developers can use Macromedia Flash MX to design, develop, and share Learning Object files, Movie Clips, and other Components.

The Landscape of Online Learning Is Changing

Both instructional technology and information technology have evolved during their transition to the web. Recent advances in online learning, such as the influence of international standards and the advancement of learning management systems, are continually altering the online learning landscape. What once was a linear, static collection of text-driven content can be transformed into a well-organized, dynamic, and engaging learning experience.

As technology itself evolves, eLearning is reaching beyond the desktop computer to other web-aware devices like mobile phones, PDAs, webpads, set-top boxes, and game consoles. Today's developers are challenged by the need to build quality online content that can be delivered across a growing variety of devices and platforms.

The new vision dynamic of online learning can be accomplished only by changing the way we design, develop, and deploy content. The goal of anyplace, anytime, just-for-me learning *is* technically possible today; but, to achieve it, we must change our paradigm from the traditional linear "course" approach to a more granular, component approach.

Learning Standards Are Essential

Today's eLearning standards provide a strong new framework, creating a bridge from abstract components to the practical world of implementation. In a recent Macromedia survey of corporate and government eLearning developers, a convincing 93% said that standards were either "very important" or important. This emphasizes how critical it is to understand the value of standards, and the challenges faced by developers in changing their design and implementation models in support of standards.

According to an industry report by the Masie Center (2002), eLearning standards are of value in that they support investments in:

1. **Interoperability, defined as interchangeability.**
2. **Durability, defined as having lasting value.**
3. **Manageability, defined by enabling value to be assessed.**
4. **Re-usability, defined as usage in different contextual situations.**
5. **Accessibility, defined as access to content for individuals with disabilities.**

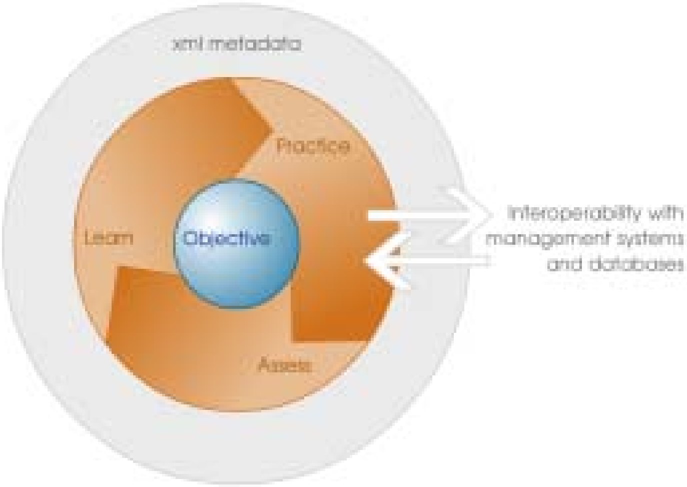
These standards focus on content metadata, content packaging, and run-time communication to support tracking of student activities. All of the standards are predicated upon the notion of a reusable Learning Object.

The Anatomy of Learning Object Design and Development

The concept behind reusable Learning Objects is that the Learning Object allows developers and users to leverage databases, Internet, and other digital technologies to prepare learning content as discrete small "chunks" or Learning Objects that can be used alone or dynamically assembled to provide "just enough" and "just-in-time" learning. [iii]

The use of Learning Objects contribute to the scalability and reuse of content and media assets within an organization. There are varying academic and instructional definitions of a Learning Object. Simply stated, a Learning Object is smaller than a course, unit, or section. It represents a small unit of instruction that teaches a focused concept. There is also heated debate about the granularity of a Learning Object, *aka* a shareable content object, knowledge object, reusable Learning Object, nugget, and by a number of other names. For the purposes of this paper, a Learning Object is defined as consisting of three fundamental elements:

Figure 1: *Anatomy of an eLearning Object*



- **Instructionally sound content can contain opportunity for practice, simulation, collaborative interaction, and assessment around a Learning Objective or outcome.**
- **Metadata or keywords that describe the attributes of the Learning Object. This allows the object to be searched, re-contextualized, and leveraged in other courses, devices, or just-in-time delivery of information.**
- **Interoperability that allows the Learning Object to communicate with management systems, databases, and web applications.**

This communication with a management system takes place through web transport protocols (XML data, HTTP, HTTPS, or URL encoded data). The purpose is to exchange or update metadata information about the use of the Learning Object. This may involve variables, such as learner scores, preferences, completion times, and progression, identifying which Learning Object to deliver next.

eLearning development teams need powerful and flexible content development tools to build Learning Objects. These tools should allow teams to draw upon shared templates and development methodologies, code resources, and media assets. This common sense approach not only saves time and money, but also contributes to a consistent process and end product. Content development tools also need to be flexible enough to adapt to the pedagogical structure and development standards of an organization. This is essential, because a developer may be creating a SCO (SCORM Learning Object[ii]) for one course and a RLO (CISCO Reusable Learning Object[iii]) for another context. Interfaces for Learning Objects must be consistent, rich, and engaging to hold the solitary online learner's attention. The final presentation of content should stand up to different platforms and devices and be able to communicate with external systems through standard web protocols.

Quality Content

While Macromedia Flash is already well known in advertising and animation circles, this new version of Macromedia Flash MX technology offers features that extend far beyond simple animation for the web. The importation of video, chat collaboration between users, multilingual language and Unicode support, and dynamic loading of sound and images are just a few of the new features in Macromedia Flash MX that improve a learner's experience and interaction with Learning Objects.

With Macromedia Flash MX, developers can quickly design rich, user-centered, collaborative environments specifically tailored to an organization's particular instructional design, metatagging standards, and content delivery needs. The added depth of the rich interface makes online learning a more vivid experience.

To help reach the widest range of learners possible, Macromedia Flash MX provides features to support compliance with the United States Section 508 Accessibility guidelines and the World Wide Web Consortium's (W3C's) Web Accessibility Initiative priority 1 guidelines through the new accessibility panel, auto-labeling of buttons, tab-order controls, and access to leading assistive technologies.

For the developer, the new component-based feature set allows sharing, standardization, and improved workflow within Learning Object development teams. To further boost their productivity and capabilities, Macromedia Flash MX supports key technology standards, protocols, and formats such as ECMAScript-262, HTML, MP3, Unicode, and XML.

Finally, Macromedia Flash players are readily available for a wide variety of leading platforms and devices. Because of this broad adoption, the development process is simplified, allowing eLearning developers to focus attention on the process of authoring content, rather than on compatibility-testing and other deployment issues.

Features for Building Learning Objects

Macromedia Flash MX and Flash Player

Macromedia Flash MX is a powerful development tool, with the ability to pull in and deliver a wide variety of web media and data sources. It has wide adoption, with Macromedia Flash Player loaded in 98% of all connected web browsers. The Macromedia Flash player is integrated into Microsoft WindowsXP. By using the flexibility of Macromedia Flash MX to design interactions, developers have greater control over the behavior of their eLearning content. Unlike a linear, static HTML page, a Macromedia Flash web application can respond on the fly and be programmed to stream in only relevant information, and only as needed. Multiple layers of content and interaction are available from a loaded Macromedia Flash file, as Macromedia Flash interfaces move seamlessly from one interaction to another, in contrast to the jarring change of a HTML page load or re-direction.

For example, a central Macromedia Flash file could contain instructions for navigation for the content in a Learning Object. As a navigation button for a particular area (a “practice” button, for example) is clicked, the appropriate Macromedia Flash files or external media assets can be seamlessly loaded into the interface.

Rapid Development Environment

The Macromedia Flash MX environment works with movie metaphors using a stage and timeline as primary structuring devices. The Stage is the screen area visible in a Learning Object. From the surrounding panels in the Macromedia Flash MX development environment, the developer can drag and drop items into the Learning Object screen to create interactions for the learner, by selecting an item on the stage and then applying the functions of the surrounding panels to that item. For example, to align three buttons, the developer selects the three buttons in the stage area and then selects the appropriate alignment function in the Align panel. To edit the actions of these buttons, the developer selects a button and then uses the Action panel to control what happens when the button is clicked.

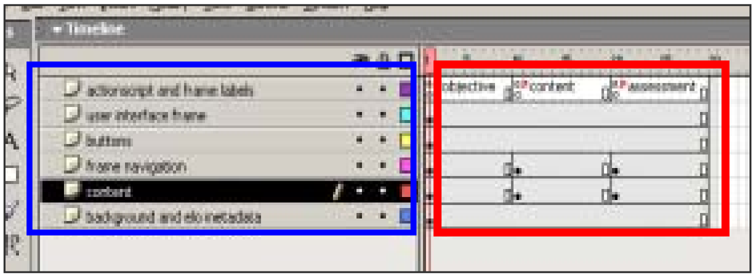
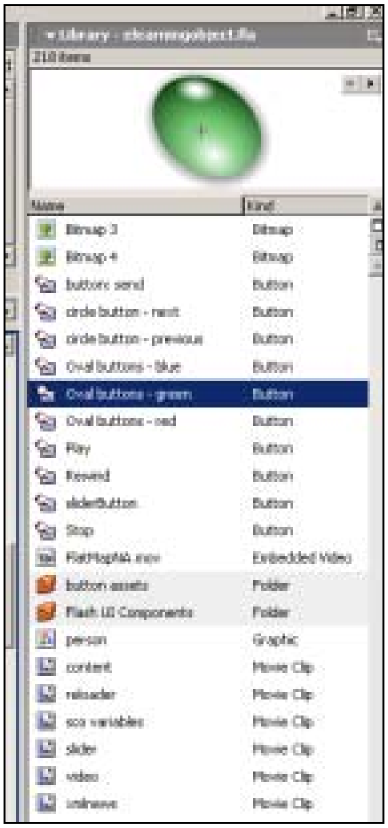


Figure 2: *Timeline containing the frames in red and layers in blue*

Frames within a timeline help structure the presentation of content items on the stage. The timeline contains layers. Layers are a way to order the appearance of content on a stage. If an item, a button for example, is on the uppermost layer, it will overlap and appear to be above items on a lower layer, such as a background tile. Notice, in the example above, the “button” layer is above the “background” layer.

Further, a timeline can be divided into framesets that define segments within that timeline. There are three main sets of frames in this Learning Object, for example. The first set of frames is used to display the objective or outcome and any appropriate instructions to the learner. The second set of frames shows where the related content interactions can be inserted. The third grouping of frames allows the developer to insert assessment interactions. A developer can rearrange frames in any order or structure appropriate to an organization’s instructional design needs.

Libraries



▾ Library - elearningobject.fla

210 items

Name	Kind
Bitmap 3	Bitmap
Bitmap 4	Bitmap
button: send	Button
circle button - next	Button
circle button - previous	Button
Oval buttons - blue	Button
Oval buttons - green	Button
Oval buttons - red	Button
Play	Button
Rewind	Button
sliderButton	Button
Stop	Button
FlatMapNA.mov	Embedded Video
button assets	Folder
Flash UI Components	Folder
person	Graphic
content	Movie Clip
reloader	Movie Clip
sco variables	Movie Clip
slider	Movie Clip
video	Movie Clip
sinewave	Movie Clip

Once the stage, frame, and layer structures for the Macromedia Flash MX file are set up, the appropriate media assets (the sound, video, images, buttons, animations, diagrams, etc.) need to be stored. In the Macromedia Flash development interface [at left], there is a library of media assets, referred to as Symbols. These Symbols may be internal to the Macromedia Flash MX file or pulled in at run-time from an external source. These libraries of information allow the developer to store a media asset, such as a sound or graphic, as a single Symbol and reuse it in many instances (much in the same way as the concept of Learning Objects). Further, these libraries can be shared between Macromedia Flash MX Learning Object developers to leverage development time and to help an organization maintain a consistent look and feel.

Figure 3: *Library panel in the Macromedia Flash MX development environment*

Movie Clips

One kind of asset that can be stored within the Library is the Movie Clip. A Movie Clip is like a mini-movie in Macromedia Flash MX, with its own independent timeline and properties. This allows for creation of frames and timelines within a Movie Clip that can be nested for use in the parent Macromedia Flash Movie. A movie clip symbol in the Library can be used multiple times in a Macromedia Flash document, with each use called an "instance." Movie Clips can be used to deliver information to the user in a particular presentation format. As shown below, a Movie Clip can be standardized to present different narrative files. It can be reused in any instance where narration is needed. Once created, a dynamic variable for its sound file and narration script may be passed to the Movie Clip at run-time. There is no need to worry about the type of plug-in on a learner's browser, as the sound file will play through Macromedia Flash Player in the web browser.

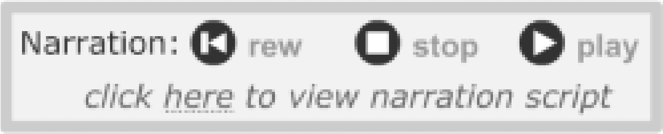


Figure 4: *Movie Clip to present narration*

The technique of designing presentation methods, such as Movie Clips, is a key to developing with Macromedia Flash MX. The developers and organizations that build eLearning applications benefit from this approach because standardization helps improve efficiency of production, reuse, and consistency in delivery.

Components

Components are complex Movie Clips that have defined parameters, which are set during document authoring or passed to the Macromedia Flash Learning Object file at run-time. These parameters can indicate which Macromedia Flash Library video to use and can provide details on how to communicate changes in metatag information to a management system. Components replace and extend Smart Clips, which were introduced in earlier versions of Macromedia Flash.

All Macromedia Flash user interface components are stored in a panel and displayed once the Flash program is installed and launched. In addition, the developer can display custom files imported or created in the Components panel by placing the Macromedia Flash MX file containing the component movie clips in the Flash MX/First Run/Components folder.

These Components might be placed, for example, in a Macromedia Flash MX file to display a quiz question at the end of a Learning Object. A PushButton at the end of a quiz, when clicked by the learner, could trigger communication with the management system. This communication could involve changing metatag information to indicate the learner has finished the Learning Object. The management system would then send the location of the next Learning Object for the learner to access. Other examples of UI Component use include: user feedback forms, screens that let learners to upload assignments, and post peer-reviews of other students' work.

The practice of using libraries and reusable components in the Macromedia Flash MX development architecture allows organizations to leverage their developers' efforts, maintain a consistent look and feel to their Learning Objects, and work together efficiently across levels as they share Macromedia Flash MX libraries and Components.

Entire classes of interactions can be built using Components as building blocks, as was done with the Macromedia Flash Learning Interactions that accompany Macromedia Flash MX. Because of the Component-based implementation, developers can easily customize the look and feel of these interactions, as well as their functionality. (See Figure 5.)

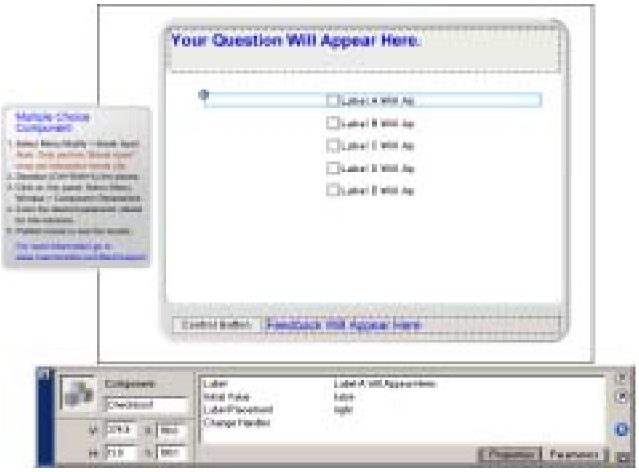


Figure 5: *Use of Checkbox Component within Macromedia Flash Learning Interactions*

Many developers have adopted a component-based authoring system with Flash. Northwest Airlines, for example, is able to leverage development time by reusing assets across different models of airplanes by creating editable components. By keeping text external to the component, Northwest is able to have subject matter experts create this in simple text documents, which can be held in a central repository for ease of updating. (An example of the Northwest Airlines model is shown below as Figure 6.)

An important issue to keep in mind while building content using components is that not all components are accessible for all users. Interactions involving drag and drop behaviors are difficult for users who are blind or have difficulty manipulating a mouse. Designers may choose to provide alternative interactions for users who are not able to complete these interactions.

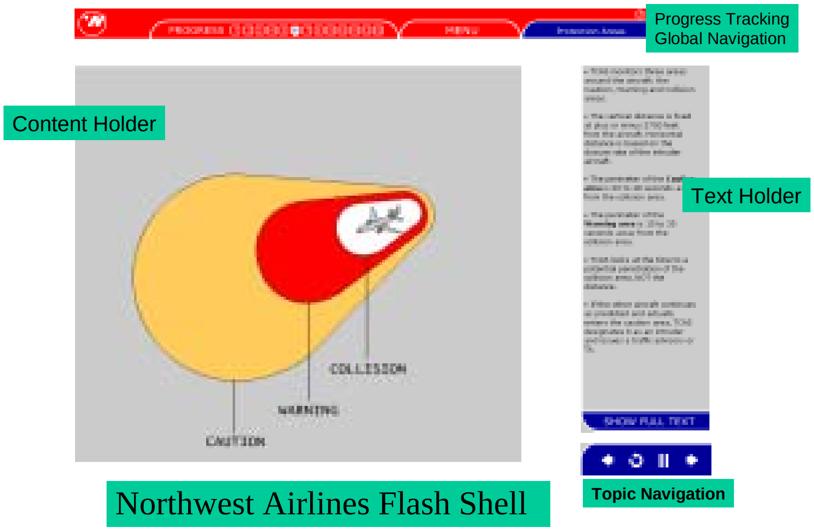
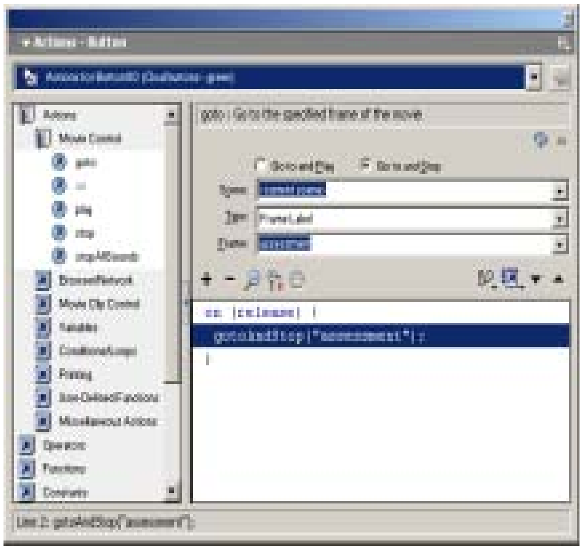


Figure 6: *Northwest Airline Macromedia Flash Shell*

ActionScript



goto : Go to the specified frame of the movie

○ Go to and Play ● Go to and Stop

Scene: [illegible]
Type: Frame Label
Frame: assessment

```
on (release) {
  gotoAndStop("assessment");
}
```

Line 2: gotoAndStop("assessment");

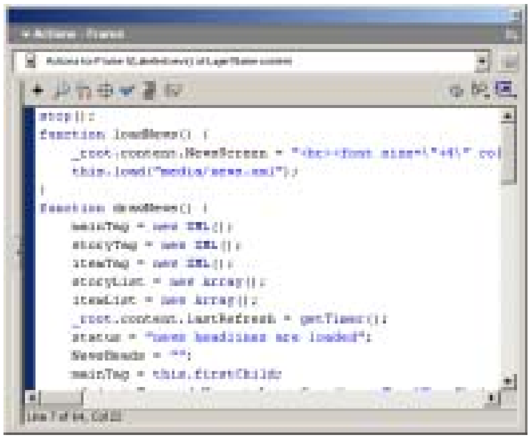
Macromedia Flash MX's ActionScript allows developers to easily create movies that adapt to learner activities without an intricate knowledge of code writing. Snippets of pre-written code (at far left in Figure 7) are available in Macromedia Flash MX to assist novice developers to begin scripting. Once a clear goal is established, it is easy to start building scripts with the actions provided.

Figure 7: *ActionScript panel in Normal mode with pre-scripted snippets on left*

Because ActionScript is based on ECMAScript-262, the internal standard for Javascript, web-savvy programmers will appreciate the power and familiarity of ActionScript's shared syntax and structure. ActionScript provides the flexibility to store and retrieve metatag variables, to control navigation, to communicate with external sources of information, and to dynamically adapt the display of the Learning Object interface to learner needs.

Using the ActionScript window in the Expert mode (in Figure 8, below) allows a developer to directly write and modify codes with a powerful context-sensitive editor. ActionScript code can even be stored externally as *.as* files that are dynamically read into the Macromedia Flash file during playback.

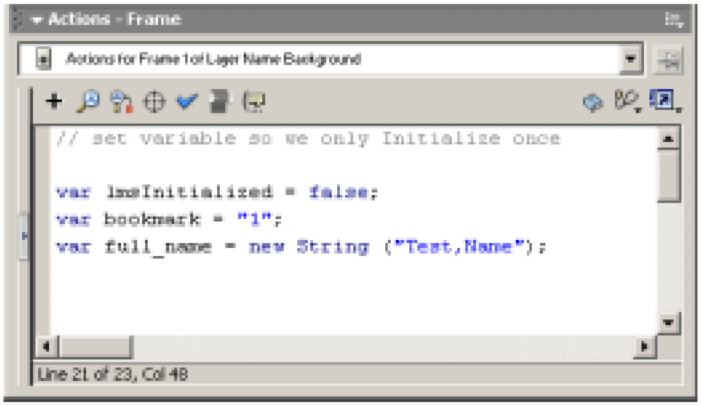
Developers familiar with other languages, such as Visual Basic, JavaScript, Perl, Java, and ColdFusion, will feel comfortable programming in the advanced mode with ActionScript.



```
stop();
function loadNews() {
    _root.content.NewsScreen = "<br><font size=\"+4\" col
    this.load("media/news.xml");
}
function drawNews() {
    mainTag = new XML();
    storyTag = new XML();
    itemTag = new XML();
    storyList = new Array();
    itemList = new Array();
    _root.content.LastRefresh = getTimer();
    Status = "news headlines are loaded";
    NewsHeads = "";
    mainTag = this.firstChild;
```

Line 7 of 64, Col 22

Figure 8: *ActionScript panel in Expert model*



```
// set variable so we only Initialize once

var lmsInitialized = false;
var bookmark = "1";
var full_name = new String ("Test,Name");
```

Line 21 of 23, Col 48

Figure 9: *Showing variables*

ActionScript objects and variables can hold virtually any data type and are useful as containers for metadata in Learning Objects. Stored data can easily be created or updated from anywhere in a Macromedia Flash movie. In the example (above, in Figure 9), the identifiers on the left side of the equal signs are variables or objects; the information to be stored is to the right of the equals signs.

Use of Bandwidth

Flash MX also offers features that will improve bandwidth efficiency. The faster page loading, cross-platform reliability, and enhanced streaming media technology will enrich the eLearning experience.

Flash MX includes a high-performance compression/decompression model. This allows Learning Object developers to compress their code, media, and data when published, for later decompression at run-time on the learner's computer.

Flash MX supports video. "Video can be easily imported from any standard video format, and played back at a variety of speeds and frame-rates. The video support in Macromedia Flash MX is optimized for low-bandwidth, high quality playback through the use of the Sorenson Spark Codec."[iv] Macromedia Flash Video Symbols can be manipulated, scaled, rotated, skewed, masked, and animated. They can also be made interactive using scripting. Video playback is supported in the browser through Macromedia Flash Player 6.

Flash files are also loaded using an efficient streaming model, in which the first few frames of content become immediately available once loaded, regardless of whether a file contains code, media, or data. This means that developers can very easily design their applications and content to quickly load specific portions of an application, while the remaining portions are loading in the background.

The Bandwidth Profiler in Macromedia Flash MX allows the developer to see how much data is being sent for each frame in the movie, according to a defined modem speed. In this way, the developer can customize a streaming Macromedia Flash file to load the most important functions and visual clues first, while the secondary components load more slowly.

Develop Once, Publish to Many

With Macromedia Flash MX, Flash source files can be exported to a wide variety of file formats. This capacity enables flexible delivery options, as well as opportunities to reuse content in many ways – for example, in HTML publishing templates that include Javascript code to communicate with either AICC or ADL SCORM compatible LMS products.

Similarly, publishing options allow developers to more easily re-purpose content. For instance, a Learning Object can be created containing interactive diagrams that may also be useful in a course seminar or program, even though the context of a seminar – a linear progression of slides supported by commentary from the presenter – may be unlike the interactive context of the Learning Object. By outputting a sequence of frames as images or animated movie file, an existing Learning Object can be leveraged into a format that suits the seminar's linear presentation and, thus, reduce the development time in preparing the seminar.

Files may also be repurposed for distribution as CD-ROM-based courses, stand-alone executables for both Mac and PC, files viewable in a browser (such as a QuickTime movie), or in image format, such as Gif and Jpg files. This degree of flexibility extends to the delivery of content to other platforms, like Portable Devices and tablet PCs, an example of which is shown below:

Figure 10: *A .net training application, created by Global Learning Systems*

Macromedia Flash MX's eLearning Framework

Components, Movie Clip, Symbols, Library, and ActionScript are a few of the features that make Macromedia Flash MX useful in Learning Object development. Because the organization of the hierarchy of information in a course may vary, Macromedia Flash MX can be used at a number of levels, as appropriate to instructional design and management system specifications. Macromedia Flash MX can be used to create a Learning Object file that contains framesets that structure the Objective, Learn, and Assessment screens.

Within those framesets, many learning interaction Movie Clips or Components may be used. Some examples of learning interactions that can be developed in Macromedia Flash MX include dynamic text, dynamic image, video seminars, simulation, mentored exercises, workshops, expert-led chats, case studies based on live data, role playing, learning styles presentation of content, intelligent pre-assessments (that direct learners to back into the content they still need to learn), peer-to-peer chat and data sharing, posted assignments with peer assessment, graphs and charts, screen narration, and collaborative exercises.

The Learning Object as Integrated Presentation Layer

Flash can be used to present the learner with different types of information from various sources in a seamless progression of interactions on the screen. Macromedia Flash MX can integrate content and interactivity using libraries of symbols, Movie Clips or Components, external management systems, web applications, and databases. Without ever being aware of it, the learner will be utilizing Macromedia Flash MX capabilities to access data and code from other web programming environments such as ColdFusion MX, .ASP, .JSP, .PHP, .NET, and Java Application Servers.

Rich possibilities for user interface development in Macromedia Flash allow for advanced visualizations of complex or abstract data relationships. Macromedia Flash interfaces can be developed in a manner that allows for dynamic input of data and consequent reuse of the Flash file within multiple contexts. Macromedia Flash is a "valuable tool to help users make sense of large amounts of data by presenting it in an easily accessible graphical representation."[v]

Depending upon how autonomous a developer wishes a Macromedia Flash file to remain, media assets can be left external to the Flash development environment or pulled into the Flash file Library. Keeping assets external allows easy updating of a file by replacing an external text, graphic, or sound file without having to republish the original Macromedia Flash file. External Macromedia Flash files containing media can also be loaded from a central Flash file, even as they are being viewed in the Flash Player.

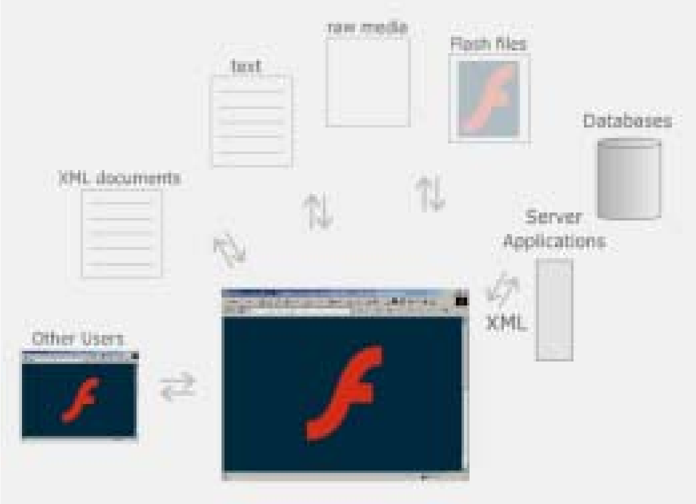


Figure 11: *Flash MX as a presentation layer using internal and external resources*

The Macromedia Flash MX Learning Object Development Process

Pre-Assess

A typical eLearning development team might consist of three to five specialists: a developer, the user interface designer, an instructional designer, an online instructor, and a subject matter expert. Sometimes, the developer and user interface designer are the same person, just as the online instructor may also be the subject matter expert. However, it is important to realize that there are five distinct skill-sets collaborating in the development process. The role of this team is for members to work together to create rich, interactive content that will integrate and communicate with management systems, server applications, and databases.

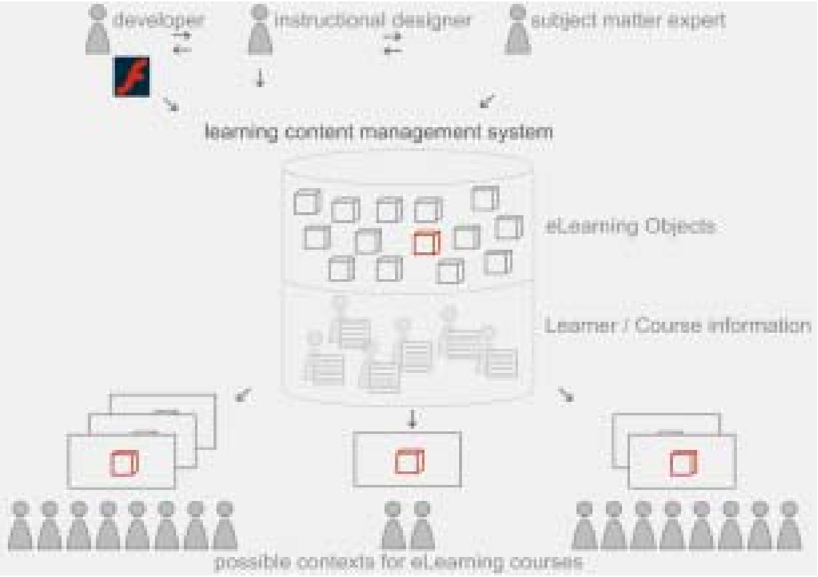
Before beginning the process of building Learning Objects, the development team should first consider the following key questions:

- What application(s) are being used to support content and learner management?
- What eLearning specifications or standards are being adopted to metatag and exchange information within the management system?
- Does the management system support these standards or specifications?
- How do these specifications and standards affect the instructional design of Learning Objects?
- What technical standards are being set to access courses through the management system?
- What are the accessibility standards at the institutional, state, and national levels?

With these questions answered, the Learning Object development team can begin to tailor an efficient system of sharable parts that will create consistent, scalable resources for the development of Learning Objects and then easily integrate them into a management system, server application, or database.

The following diagram illustrates the flow of content through the Learning Object development team to the management system. The management system then delivers, reuses, and tracks the use of the content to a variety of course delivery situations.

Figure 12: *Learning Object development team collaborates on the development, places the Learning Object in a management system which then delivers to a variety of eLearning situations.*


developer
instructional designer
subject matter expert
learning content management system
eLearning Objects
Learner / Course information
possible contexts for eLearning courses

Analyze and Design

The Learning Object development team can begin to collaborate on the transformation of content into engaging Learning Objects. The Macromedia Flash authoring environment allows Learning Object developers to create interactions around various technical specifications and instructional design models. Throughout the analysis and design process, data and decisions should be documented in a structured manner. This will be very valuable in setting the scope of a Learning Object, determining the structure of a course (content packaging and sequencing), and properly tagging high-value assets (media and Learning Objects) for potential re-use.

Hypothetical Example

> *The subject matter expert, a fantastic eCommerce classroom teacher and new to eLearning, faxes handwritten ideas for content to the developer and the instructional designer.*
>
> *The patient developer and the instructional designer use these faxed notes as a starting point in the development process. In collaboration, the team of three breaks the content down into manageable chunks and asks the subject matter expert many questions to focus the subject content. The content is structured around outcomes or objectives. The team considers assessment methods and brainstorms methods for meaningful learning interactions that capitalize on the subject matter expert's inspiring classroom presence. At the same time, the developer and the instructional designer are considering the potential reuse of the Components and Movie Clips for the development of other Learning Objects.*

> *The developer shows the team innovative uses of Macromedia Flash in examples of other Learning Objects. This helps the subject matter expert, who is new to the concept of Learning Objects, to understand the possibilities for the potential of rich interactive content.*
>
> *The original, faxed notes evolve into a series of interactive diagrams, drag and drop exercises, and an assignment with peer assessment to test a central learning objective. With this plan in mind, the developer is ready to begin work in the Macromedia Flash authoring environment.*

Develop

In this stage of the Learning Object development process, team members will collect the media assets and make decisions as to whether they will be stored externally or internally to the *Macromedia Flash* file. Appropriate Movie Clips and Components are created to present content on the screens within the Learning Object. These Movie Clips and Components will draw upon shared Libraries of symbols that ensure a consistent look and feel to an organization's Learning Objects. When creating new Movie Clips and Components for learning interaction, consideration should be given to their reusability, so that, once developed, they can be shared with other Learning Object development teams, either within or external to an organization.

Hypothetical Example (continued)

> *The specifications for the Learning Object have been analyzed and designed by the Learning Object development team. The developer now gathers all existing resources that will contribute to the development process. These may include existing media assets such as diagrams, videos, images, and sounds. Some new media assets may need to be created, such as sound files from narration scripts. (The developer is aware that some learners access the courses through slow modems; so, wherever a sound file is used, a textual narration script will also be made available for the slower connections.)*
>
> *The then sets up a Learning Object file to contain framesets for the Objective, Learn, and Assess sections of each Learning Object. Within each of these framesets, Library symbols, Movie Clips or Components are inserted to facilitate the delivery of the content in the Learning Object. If there are Components or Movie Clips that are new additions, the developer will design and make them available to other developers within the organization. To facilitate this sharing, it is a good idea for the developers to create and maintain a common style guide of considerations used in developing symbols for the Library, Movie Clips, shared ActionScript, and Components for Learning Objects. Such a guide help to ensure the consistent development of Learning Objects, as well as to provide a possible record of the organization's Macromedia Flash MX development files.*

It is even possible to document which raw media elements or components are used for a given Learning Object within the XML file for IMS Content metadata or ADL SCORM Content metadata. An XML editor should be used to populate an XML template from the standards bodies and to extend these templates as necessary for a project. The next section deals with examples of learning interactions (such as Movie Clips, Components, or Symbols in Macromedia Flash Libraries) that can be built to increase a developer's productivity.

Learning Interactions

Flash MX has the capability to allow for the creation of elements for re-use with a wide variety of eLearning content. These elements, constructed as Movie Clips, Components, or Libraries, are sharable from one developer to another.

In Macromedia Flash 5, there were learning interactions in the core product, with additional learning interactions were available through the Macromedia product exchange located at: http://www.macromedia.com/exchange. In Macromedia Flash MX, these "Smart Clips" have been renamed "Components," and the approach to authoring them has been slightly modified. To insert a Learning Interaction Component, it is only necessary to open the Window menu and select Learning Interactions under the Common Libraries tab.

Macromedia Flash MX also provides a Learning Interaction template that is pre-packaged to work with SCORM or AICC eLearning specifications. To access these templates, the developer chooses File > New from Template. In the New Document dialog box, he/she selects the Quiz option from the Category list, chooses a Quiz Style from the Category Items list, and then clicks OK.

To create Learning Interaction Quiz samples, the developer just needs to follow the onscreen instructions. They provide a walk-through to the launching and editing of information in the Component Parameters Panel. Further documentation on using Macromedia Flash Learning Interactions is available at: http://www.macromedia.com/learning

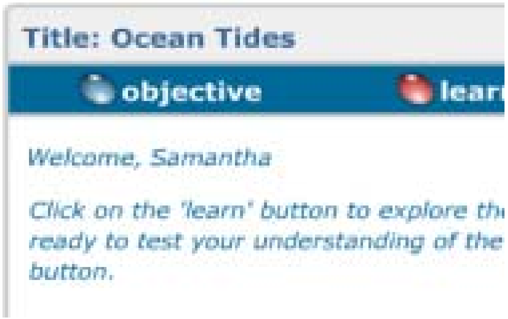
Developers can customize and create their own Learning Interaction Components by placing the Macromedia Flash MX file containing the Component or Movie Clips in the Flash MX/First Run/Components folder. Further information on Component development is available at: http://www.macromedia.com/support/flash/applications/creating_comps/

When using, creating, or editing these elements, it is important to first consider the learning environment in which they will be used. Developers should collaborate with the subject matter experts, the learners, and the instructional designers, as well as those responsible for their organization's technology infrastructure (whether it is a web server or a full-blown LMS). When designing Learning Objects, it is essential for a developer to maintain focus on the content and usability of the interface. Developers can leverage their work efforts by centralizing Macromedia Flash MX publishing templates, Libraries, and Components for reuse.

To stay updated on Macromedia Flash MX, development teams should regularly check the Macromedia exchange (http://www.macromedia.com/exchange) for any components that have been added since this document was published. To share some components with others, developers can post theirs to this same site.

The following scenarios illustrate but a few of the endless possibilities for Learning Object development:

Scenario – Objective Screen


Title: Ocean Tides

Welcome, Samantha

Click on the 'learn' button to explore th
ready to test your understanding of the
button.

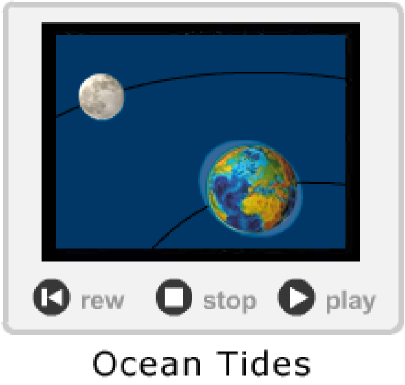
This Component greets the learner by name through the use of dynamic text. This is an indication of the level of communication with the management system. The developer may also design Learning Objects to store information on each learner's learning style or language preferences, and then dynamically update the interface to meet these needs.

Figure 13: *Component greeting the user by first name through use of dynamic text.*

Related Macromedia Flash MX Features:

- Dynamic text fields
- Storage of data through Shared Objects
- Setting and passing of variables
- Dynamic movie layout control through ActionScript
- Communication through transport protocols (XML data, HTTP, HTTPS, or URL encoded)

Scenario – Video Display



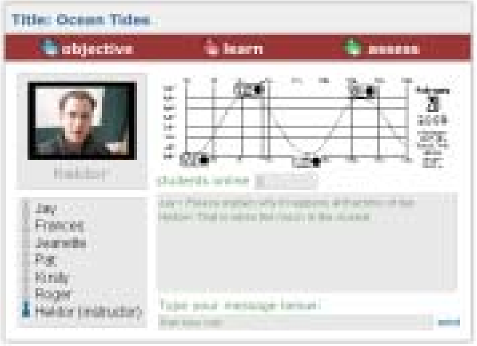
This Learning Object pulls Macromedia Flash Library video Symbols into a consistent interface for the presentation of video media. This interface ensures efficient playback on all devices and platforms and allows the Learning Object developer to reuse the presentation Component by simply replacing the Macromedia Flash video Symbol. This movie clip can be reused on any screen in a Flash file.

Figure 14: *Component for standardizing the display of video assets.*

Related Macromedia Flash MX Features

- Video Importation
- Sorenson Video Compression
- Buttons and Movie Clips
- Support for Captioning using MAGpie

Scenario – Expert-Led Chat



This example allows learners real-time access to expert-led chat sessions to discuss particular topics. In this example, learners can type in a message to fellow classmates and their instructor. They are discussing a dynamic image pulled in from an external source by the instructor. Discussions could be recorded in a database and played back later to learners unable to participate in the synchronous session.

Figure 15: *Expert-led chat showing participants on the left and chat on the right.*

Related Macromedia Flash MX Features

- Dynamic image
- ActionScript set variables
- ActionScript Local Connection
- Shared Object
- Connection through transport protocols (XML data or HTTP, HTTPS, URL encoded

Implement and Evaluate

A Learning Object should be implemented and evaluated in a test run before its release to learners. This involves creating a content package, such as an ADL SCORM Content Manifest. Macromedia Dreamweaver and the “Manifest Maker” extension (available at http://www.macromedia.com/exchange/dreamweaver) can be used to easily create a SCORM-compatible XML manifest that includes course level metadata, a table of contents, and a resource list identifying all support files *and* the supporting content metadata created during the development of the Learning Object. The content package generated by the Macromedia Manifest Maker can be used to load the entire course framework, or even a sub-group of Learning Objects, into the management system.

Both the individual Learning Object and the larger framework should be tested against the instructional design and technical criteria they were intended to meet. This requires a rigorous Quality Assurance (QA) period of testing and revision; then, Learning Objects should be revised and retested accordingly. It is essential to test content under a Learning Management System early in the process to isolate and resolve any communication issues. The Macromedia Flash Learning Interactions and Quiz templates should work well with any robust AICC- or ADL SCORM-compatible LMS product. However, there are subtleties to the standards that may need to be shaken out, especially if there will be an exchange of custom data extending beyond the typical elements. During the QA process, the modular design and effective re-use of assets will streamline debugging and revision, regardless of the type of issue —aesthetic, content, instructional or technical.

Hypothetical Example (continued)

Once the Learning Object development team has completed the initial development process, a series of Learning Objects are loaded into the management system. A Quality Assurance Tester, preferably someone with no previous involvement with the development process, will request the Learning Object from the management system and test the Learning Objects against a usability, instructional design, and technical specifications checklist. As the file loads, it creates a connection to collect appropriate metadata from the learning manager and then personalizes the content. As the “objective” screen loads (see below), the Learning Object welcomes the learner by name and records the learner’s path as he or she navigates the content. This path could be recorded as a series of variables that is then communicated back to the management system. It is particularly useful to record learner activity when analyzing the effectiveness of eLearning content.

Objective

The ”objective” screen:

- relays information about what is about to be studied,
- gives an indication of the time commitment needed to complete the Learning Object, and
- contextualizes the content, with instructions on how to proceed.

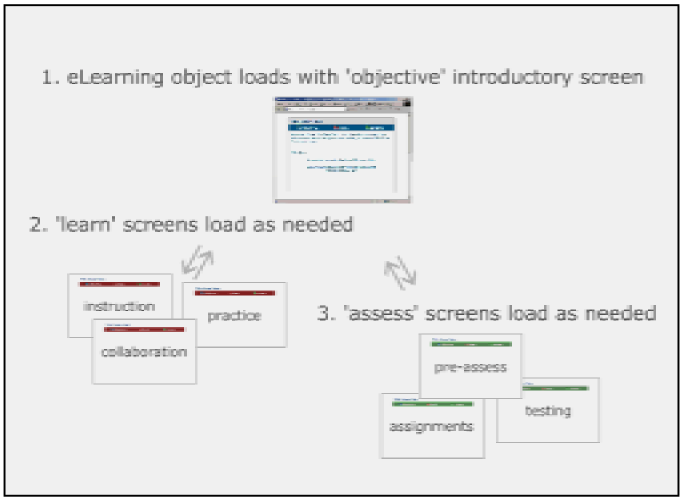

1. eLearning object loads with 'objective' introductory screen
2. 'learn' screens load as needed
instruction
practice
collaboration
3. 'assess' screens load as needed
pre-assess
testing
assignments

Figure 16: *Screens of learning interactions within a Learning Object*

Learning

The “learn” screens display the content necessary to teach to the learning objectives. These may take the form of instruction, collaboration, or practice. Effective learning interaction is achieved by thoughtfully combining media assets, such as sound files, graphics, charts, and graphs from real-time data, Macromedia Flash video Symbols, external resources, simulations, and live communication through chat or shared data. All of this rich, engaging user interaction is possible for the learner using the Macromedia Flash Player in a browser without any need for additional plug-ins or concerns about how the media might perform on a different platform or device.

Further, Macromedia Flash MX allows the originating Flash file to load other Flash "learn" files, as they are needed. This prevents the unnecessary use of bandwidth, when the learner may not wish to view particular screens. While designing a Macromedia Flash Learning Object, it is important to take the time to anticipate the needs of the Learner. For instance, while displaying the first learn screen, the developer may choose to stream in the loading of screens two and three.

Assessment

Once the learner is comfortable with the content, he or she may proceed to the "assess" screen. Some learners may even jump to the "assess" screen right at the beginning to specifically identify what they will need to know. While the learner is working through the content and assessment within a Learning Object, metatag values that have been tracked within the Macromedia MX file can be updated in the management system. After a Learning Object is successfully completed, the learner can then be directed to the next Learning Object.

Learning Objects should not be made available to learners within a course until after the Objects are tested, evaluated, and revised. It is good practice to provide opportunities for learner feedback within Learning Objects from both a technical and instructional design perspective.

Maintain

eLearning development is an evolutionary process. As the development team arrives at the end of one course delivery cycle and embarks on another, it will be faced with a variety of maintenance issues. The first such issue is the need to update content. Wherever possible, the developer should provide access for the subject matter expert to make these changes. This can be achieved by keeping content external to the presentation layer. The content could be kept in a database or text file and dynamically loaded into the Macromedia Flash MX Learning Object during course delivery. Text files are easily editable for a subject matter expert. Database content can be accessed through a web form interface that allows the subject matter expert control over the content.

Documentation is another maintenance issue. An up-to-date, well-maintained record of sharable learning interactions will create an awareness of what each Learning Object development team is working on, while, at the same time, providing a useful showcase to demonstrate the richness of Learning Objects for instructors new to eLearning. The meta- and content-level data and content package for the entire course allows the team to easily catalog, re-use, and re-structure content for a variety of users and situations. The metadata, which may once have seemed so tedious to create, now makes it easy to find the tiniest re-usable elements, track content changes through an audit trail, and create "cloned" courses with subtle differences, by creating a near-duplicate content manifest.

It is a good idea to hold regular meetings with developers, instructional designers, and subject matter experts to review and plan improvements in response to the course feedback and input received from learners and online instructors. Development teams that have used AICC or ADL SCORM content communication can even perform sophisticated statistical analysis on student usage and performance patterns, at either the interaction- or the lesson-level.

Conclusion

The scenarios (mentioned above) illustrate just a few of the possibilities available to developers of eLearning content using Macromedia Flash MX. This list can be expanded to an even wider variety of Learning Object interactions, such as:

- Case studies based on live data,
- Synchronous and asynchronous workshops (including new Macromedia Flash MX video capabilities),
- Mentoring environments,
- Role-playing exercises,
- Access and navigation of content-based on learning styles preferences,
- Intelligent pre-assessments that direct learners back in the content they still need to learn,
- Peer-to-peer chat and data sharing, and
- Interfaces to post and review class assignments.

Soon, we may see the development of online exchanges for Learning Object interactions, through which developers, instructional designers, and online instructors can share learning interactions and development ideas.

This is just the beginning of what will surely be a very creative time in the eLearning industry.

For additional information and white papers on eLearning, please check http://www.macromedia.com/learning frequently.

Tanya Heins is an eLearning consultant with a background in teaching, fine arts, and applied research in eLearning. She holds an M.F.A from the University of Victoria in British Columbia, and has had more than eight years of professional interactive design and development experience, ranging from designing CD-ROMs and database-driven websites to custom development of award winning learner and content management systems.

You can contact Tanya at tanya@numen.ca

Frances Himes is the Associate Vice President of Higher Education at Macromedia, in which position she facilitates the development and implementation of cutting-edge learning technologies that advance distributed learning efforts. During the past 17 years, her work has focused on adaptive learning architecture, dynamic eLearning environments, intercultural forms of learning, and teaching and learning with technology. She has served at the University of Arizona, as both the co-author and Project Director for the Virtual Adaptive Learning Architecture Project and as Director of Global Programs, at Western Michigan University, as Director of Research and Development for Information Technology, and at the University of Wisconsin-Eau Claire, as the Associate Director of International Education.

You can contact Frances at fhimes@macromedia.com

Resources

Allaire, Jeremy (2002). Macromedia Flash MX - A next-generation rich client. Macromedia. Available at: http://www.macromedia.com/software/flash/whitepapers/

Advanced Distributed Learning Initiative (2001). The SCORM Content Aggregation Model. Available at: http://www.adlnet.org/scorm/scorm_download_index.cfm

Bloom, B. S., Englehart, M. D., Furst, E. J., Hill, W. H., & Krathwell, D. R. (Eds.) (1956). A taxonomy of educational objectives: Handbook I. Cognitive domain. New York, NY: David McKay.

Brogan, Pat (1999). Using the Web for Interactive Teaching and Learning. Macromedia. Available at: http://www.macromedia.com/resources/elearning/whitepapers.html

Chambers, Mike (2002). Macromedia Flash MX Security. Macromedia. Available at: http://www.macromedia.com/software/flash/whitepapers/

Cisco (2001). E-Learning Glossary. Available at: http://www.cisco.com/warp/public/10/wwtraining/elearning/pdf/elearn_glossary.pdf

Cisco (2001). Reusable Learning Object Strategy, Version 4. Available at: http://www.cisco.com/warp/public/10/wwtraining/elearning/implement/rlo_strategy.pdf

Dolence, M. G., & Norris, D. M. (1995). Transforming higher education: A vision for learning in the 21st century. Ann Arbor, MI: Society for College and University Planning.

Downes, Stephen (2001) Learning Objects: Resources For Distance Education Worldwide. In International Review of Research in Open and Distance Learning (2001). Available at: http://www.irrodl.org/content/v2.1/downes.html

DLN Test Lab (2002), Canadian Department of National Defense, SCORM Dynamic Appearance Model, Version 1.0 Available at: http://www.online-learning.com/papers/SCORMModel.pdf

Eduworks Corporation (2001). Conformance, Compliance and *Caveat Emptor.* Available at: http://www.eduworks.com/web/Tutorials/MakingStandards/index.cfm

Eduworks Corporation (2001). Pros and Cons of Learning Objects. Available at: http://www.eduworks.com/LOTT/tutorial/prosandcons.html

Fisher, Sue (2001). POOL Indexing Guidelines, Version 1.0. Fredericton, NB: University of New Brunswick. Available at: http://www.cancore.ca/POOL_Indexing_Guidelines_1.0.doc

Guldman, Andrew (2001). Building Rich Internet Applications with Macromedia Flash and ColdFusion. Macromedia. Available at: http://www.macromedia.com/resources/business/solutions/user_centric/rich_internet_apps.pdf

Hall, Brandon (2001). LMS 2001: How to Choose the Right System for Your Organization. brandon-hall.com. Available at: http://www.brandon-hall.com/learmansys.html

Hartnett, John (2002). Where have all the Legos gone? Vol. 6, No. 2. Online Learning Magazine. Minneapolis MN. VNU Business Publications.

Jacobsen, Peder, (2001). Reusable Learning Objects. e-Learning Conference and Expo. Available at: http://www.elearningmag.com/elearning/article/articleDetail.jsp?id=5043

LearnSource (2001). E-Learning: Some of the Basics. Available at: http://www.learn-source.com/learn/ls.nsf/rlookup/gs01?open

Macromedia. Accessibility Starter Kit. Available at: http://www.macromedia.com/macromedia/accessibility/tools/accessibility_kit.html

McGreal, Rory & Roberts, Toni (2001). A Primer on Metadata for Learning Objects. e-Learning Conference and Expo. Available at: http://www.elearningmag.com/elearning/article/articleDetail.jsp?id=2031

The Masie Center; 2002. Making Sense of Learning Specifications and Standards; A Decision Maker's Guide to their Adoption. Available at: http://www.masie.com/standards/

Muzio, Jeanette, Heins, Tanya & Mundell, Roger (2001). Experiences with Reusable eLearning Objects: From Theory to Practice, Victoria, BC: Centre for Economic Development and Applied Research (CEDAR), Royal Roads University. Available at: http://www.cedarlearning.com/CL/elo/eLearningObjects_sml.pdf

Nichani, Maish (2001). LCMS = LMS + CMS [RLOs] – How does this affect the learner? The instructional designer? elearningpost. Available at: http://www.elearningpost.com/elthemes/lcms.asp

Perfetti, Christine & Spool, Jared (2002). Macromedia Flash: A New Hope for Web Applications. Bradford, MA. User Interface Engineering. Available at: http://www.macromedia.com/resources/business/solutions/user_centric/flash_web_apps.pdf

Wiley, David (1999). The Post-LEGO Learning Object. Available at: http://works.opencontent.org/docs/post-lego/

Citations

[i] Page 8, The Masie Center; 2002, Making Sense of Learning Specifications and Standards: A Decision Maker's Guide to their Adoption. Available at: http://www.masie.com/standards/

[ii]Advanced Distributed Learning Initiative (2001). The SCORM Content Aggregation Model.

[iii] Cisco (2001). Reusable Learning Object Strategy, Version 4.

[iv] Page 6, Allaire, Jeremy (2002). Macromedia Flash MX - A next-generation rich client. Macromedia.

[v] Page 12, Perfetti, Christine & Spool, Jared (2002). Macromedia Flash: A New Hope for Web Applications. Bradford, MA. User Interface Engineering. Available at: http://www.macromedia.com/resources/business/solutions/user_centric/flash_web_apps.pdf