



THE
le@rning
FEDERATION

schools online curriculum content initiative

TECHNICAL SPECIFICATION FOR CONTENT DEVELOPMENT

VERSION: 5.2

DATE: MARCH, 2008

Table of Contents

1	INTRODUCTION	1
1.1	Purpose.....	2
1.2	Obligations of The Le@rning Federation	2
1.3	Monitoring and usage.....	2
1.4	Conformance.....	2
2	PRINCIPLES	3
2.1	Accessibility	3
2.2	Useability	3
2.3	Interoperability	3
2.4	Flexibility.....	3
2.5	Durability.....	3
2.6	Scalability	4
3	INFORMATION MODEL	4
3.1	Learning content model	4
3.2	Packaging learning objects	5
3.3	Separating structure, presentation and functionality	5
4	CONTENT MODEL REQUIREMENTS	6
4.1	Summary.....	6
4.2	XML.....	7
4.3	Style sheets	11
4.4	Images and graphics	11
4.5	Audio	12
4.6	Video	12
4.7	Colours.....	12
4.8	Device independence	12
4.9	Content delivery considerations	14
4.10	Document formats	15
4.11	Scripts.....	15
4.12	Web applications	15
4.13	Performance	16
4.14	Persistence.....	17
4.15	Screen layout.....	17
4.16	Directory structure.....	17
4.17	Rights information	18
4.18	Fonts	19
4.19	Client software requirements	19
4.20	Client hardware requirements	20

Disclaimer

"The material contained in the *Technical Specification for Content Development* is for general information purposes only. Any use of the material is at your own risk. To the extent permitted by law, Curriculum Corporation will not be liable for any loss or damage suffered as a result of any party relying upon this *Technical Specification*."

1 Introduction

The Le@rning Federation's *Technical Specification for Content Development* refers to the

standards and specifications that are to be adopted when creating content for The Le@rning Federation (TLF).

1.1 Purpose

The *Technical Specification* defines the use of technical standards and specifications for learning content developed by The Le@rning Federation initiative. The document defines the standard operating environment required for use of The Le@rning Federation content.

The *Technical Specification* aims to ensure that content development follows internationally accepted specifications. The guidelines outlined in this document will provide The Le@rning Federation with content of a consistent type and style that will maximise the viability, integrity and portability of the content over the life of the project and beyond.

1.2 Obligations of The Le@rning Federation

Curriculum Corporation will fulfil the following obligations in undertaking and delivering The Le@rning Federation outcomes:

- consult with relevant organisations, user groups and multimedia content developers in establishing, implementing and reviewing this specification
- proactively review and, as appropriate, evolve this specification to reflect knowledge and practice derived from development, deployment and use of content for the project
- contribute to national and international standards development in the area of multimedia content development through the evolving body of knowledge and practice generated under the auspices of the project.

1.3 Monitoring and usage

The *Technical Specification for Content Development* applies to all content commissioned as part of The Le@rning Federation.

The specification will also be applied to non-commissioned content considered for distribution by The Le@rning Federation. The Le@rning Federation realises, however, that not all of this specification may be applicable to non-commissioned content. The Le@rning Federation will *not* necessarily exclude non-commissioned content that deviates from this specification.

It is expected that technologies will evolve over the course of The Le@rning Federation Initiative. For that reason, TLF monitors technologies and standards likely to be deployed in classrooms over the course of The Le@rning Federation and beyond.

The *Technical Specification* will be updated and enhanced during development, deployment and use of content. Updated specifications and related guidelines will be documented and published on The Le@rning Federation website at <http://www.thelearningfederation.edu.au> and in The Le@rning Federation knowledge base at <http://jira.thelearningfederation.edu.au/confluence/x/IAE>

1.4 Conformance

Content conforming to this specification must adhere to the principles described in Section 2, the information model in Section 3 and the requirements defined in Section 4 of this document.

The key words "must", "must not", "required", "shall", "shall not", "should", "should not", "recommended", "may", and "optional" utilised in the requirements are to be interpreted as described in The Internet Engineering Task Force RFC 2119 <http://www.ietf.org/rfc/rfc2119.txt>

2 Principles

The *Technical Specification* is used to assess whether The Le@rning Federation's content conforms to the following six principles:

- accessibility
- useability
- interoperability
- flexibility
- durability
- scalability

2.1 Accessibility

The first principle aims to ensure that The Le@rning Federation's online content and services are inclusive of a range of teaching and learning capacities, contexts and environments. It aims to ensure content conforms to Commonwealth laws concerning accessibility, and State and Territory policy regarding inclusive education provision. This accessibility principle informs The Le@rning Federation specifications on technical content, accessibility and educational soundness.

2.2 Useability

The Le@rning Federation's content must be useable by a range of teachers and students. That is, the content's interface must be learnable, efficient, memorable, have a low error rate and be pleasant to use¹.

2.3 Interoperability

The aim of this principle is to ensure that The Le@rning Federation's content can be used on and shared between a wide range of software and hardware platforms. It assumes that the content will be used within operating environments that support internationally adopted standards and specifications.

2.4 Flexibility

This principle relates to mixing and reusing learning content from a range of sources into multiple applications and environments. It is realised by ensuring learning content and its constituent components is adequately described and packaged so that it can be found and re-contextualised.

2.5 Durability

This principle aims to ensure that The Le@rning Federation's online content can withstand technology changes without requiring redesign or recoding. The principle is often realised by encoding content in formats that can be easily interpreted and translated into future technologies. For example, images may need to be provided in both low- and high-resolution

¹ This definition of usability is derived from *Usability Engineering* by Jakob Nielsen, published by Academic Press, USA, 1993.

formats. The low-resolution format is for inclusion in the content, and the high-resolution format would be used for possible future translations.

2.6 Scalability

This principle relates to using technologies that allow The Le@rning Federation to benefit from both growth in demand for its services and an increase in new inputs, such as technologies that may enhance the re-use of the resources collected.

3 Information model

The Le@rning Federation is procuring online content in the form of learning objects and digital resources.

The Le@rning Federation uses the following definitions:

- A **learning object** is 'an interactive digital resource facilitating learning experiences related to a particular educational purpose'. Learning objects are designed in accordance with deliberate decisions about learning design and learning outcomes.
- A **digital resource learning asset** is 'a resource that contains an item, such as an image or moving image footage or an audio file, accompanied by information about the item and interpretation of its educational value'. Digital resource learning assets are built around items sourced from cultural institutions, scientific organisations and private individuals.

3.1 Learning content model

The information model for learning content (learning objects and learning assets) is illustrated in Figure 3-1: Learning Content Information Model. Learning objects contain resources, organisations and metadata.

- **Resources** are files and subordinate learning content that are used to facilitate the learning experiences. Resources may be any of the file types described in Section 4 'Content model requirements' such as XHTML files, CSS style sheets, Flash interactive files and MP3 audio files.
- An **organisation** specifies a navigation path through the learning content. A piece of learning content may have many organisations, and hence many possible navigation paths. For example, a learning object may have two educationally equivalent organisations: one optimised for visual learners and one optimised for use by visually impaired learners.
- Within learning content, **metadata** is structured information about the learning content and its resources. For example, metadata may include the learning content's title, description and educational purpose. Metadata is described in The Le@rning Federation's *Metadata Application Profile*. It supports the learning object and resource management, description of educational purpose, technical interoperability, digital rights management and accessibility.

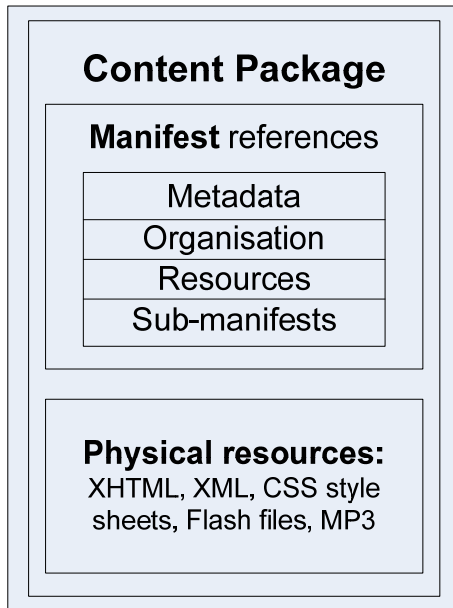


FIGURE 3-1: LEARNING CONTENT INFORMATION MODEL

3.2 Packaging learning objects

The Le@rning Federation delivers learning content to stakeholders as either IMS Content Packages - refer to 'Content Packaging Specification' on the IMS Global Learning Consortium, Inc. website at <http://www.imsproject.org/content/packaging/> or SCORM 2004 2nd edition version 1.3 Content Packages - refer to the Advanced Distributed Learning (ADL) website at <http://www.adlnet.gov/scorm/20043ED/Documentation.cfm>

Developers can upload resources to The Le@rning Federation's Exchange content management system and organise those resources into learning objects using the Exchange. These learning objects are then exported from the Exchange as either IMS or SCORM 1.3 Content Packages.

3.3 Separating structure, presentation and functionality

As much as possible, The Le@rning Federation's content must be constructed using technologies that separate the content's structure, presentation and functionality.

The structure of a resource refers to how it is organised. In print media information in a document may be organised by chapters, which includes an introduction and a table of contents. In web media information in a document may be organised within elements and nodes within an XML file.

The presentation of a resource refers to how the resource is rendered. For example, a document may be represented in print, as a web page or as an audio file such as synthesised speech. The presentation of a resource may also be effected by the application of another resource for example a CSS or XSLT may alter the presentation of an XML file.

The functionality of a resource refers to how the resource interacts with a user. For example, a web page may contain links that open other web pages in separate windows, and a Flash simulation of a calculator may perform calculations in reaction to mouse clicks.

Where possible, the resource structure, presentation and functionality should be separated.

Separating structure, presentation and functionality enhances the stated principles of accessibility, usability, flexibility and durability. It allows graceful transformation of content into different renderings for different devices, easy re-purposing of content for different users. Learning content structure, presentation and functionality should facilitate the development maintenance and repurposing of content.

The externalisation of data should be maximised as it decreases costs of production including development time and increases the capacity to re-use content.

4 Content model requirements

Learning objects procured by The Le@rning Federation must only contain resources of the types as listed in this section of the *Technical Specification*.

Other resource types may be considered, based on negotiation with The Le@rning Federation. Decisions about the use of other resource types will be influenced by consideration of the availability of appropriate means for users to utilise them.

4.1 Summary

Area	Specifications
XML markup	UTF-8 character set DOM Level 2 XHTML 1.1 SVG 1.0 SMIL 2.0 MathML 2.0 QTI 1.2
Style sheets	CSS 1, CSS 2.1
Images and graphics	PNG / MNG 1.0 JPEG SVG 1.0 (where appropriate)
Audio and video	MPEG-1 Audio Layer 3 (MP3) QuickTime video, playable in QuickTime Player v7.1.6 MPEG-4 Video
Colours	Do not rely on colour to convey information Provide sufficient contrast between foreground and background colours
Device independence	Support both keyboard and mouse access Keyboard navigation via TAB, ENTER and ARROW keys
Document formats	Text files for simple data (.txt) PDF Acceptable alternatives subject to approval: MS Word, MS Excel, MS PowerPoint

Client-side scripting	ECMAScript Revision 3 (JavaScript) ActionScript Lingo
Web applications	XHTML and SVG with ECMAScript. Flash, playable in Flash Player 9 Shockwave, playable in Shockwave Player v10 (full installer) Java applets (J2SE Runtime Environment (JRE) 5.0) QuickTime VR, playable in QuickTime Player v7.1.6
Bandwidth	Interaction time of less than 10 seconds over dedicated 256 kbit/s connection Maximum size guideline: 5 MB
Persistence	State must <i>not</i> be maintained across learning object or browser sessions
Screen layout	Scalable objects, optimised for screen resolution of 1024 x 768 pixels
Start file	index.html as starting point
File and directory names	RFC 2396 acceptable characters
Rights information	The copyright statement must be placed in the XHTML start file of each learning object and digital resource, digital resources may also incorporate rights information for partner institutions: Copyright TLF: © Curriculum Corporation, 2006, except where indicated under Acknowledgements The printable output footer must be placed on all printable outputs: Printable worksheet made available by The Learning Federation: © Curriculum, 2006, except where indicated under Acknowledgements Copyright TLF and partner institutions: © Curriculum Corporation and Partner institution, 2006, except where indicated under Acknowledgements
Fonts	Windows 2000 and XP: default installation fonts plus East Asian fonts OS X: default installation fonts
Client software requirements	Mozilla Firefox 2.0 and Microsoft Internet Explorer 6.0 on Microsoft Windows 2000 Mozilla Firefox 2.0 and Microsoft Internet Explorer 7.0 on Microsoft Windows XP Apple Safari 2.0 on Apple OS X

4.2 XML

XML is a key technology for separating structure and presentation. XML should be used to represent structured information. Technologies such as style sheets should be used to present that information.

XML that conforms to a published schema enhances the stated principle of interoperability by allowing automated validation of information structure.

XML enhances the stated principle of durability by conveying some of the information's purpose within the XML elements and schemas used to represent the structure.

4.2.1 Structured information as XML

Structured information should be represented using XML.

XML element and attribute names should represent the semantics of the data they describe.

4.2.2 UTF-8

XML documents, including XHTML documents, must be encoded using the UTF-8 character set.

4.2.3 XML DOM

The use of the XML Document Object Model Level 2 is permitted.

4.2.4 XHTML

Web page markup must conform to the XHTML 1.1 specification; refer to the W3C website at <http://www.w3.org/TR/xhtml11/>.

4.2.5 Objects embedded within XHTML

To conform to the XHTML specification, the `object` element must be used to embed external web applications such as Flash and Director objects within XHTML web pages. The `object` element must also be used to embed external audio and video in XHTML web pages.

Standalone web applications should scale and be centred (see Section 4.15 'Screen layout').

Menus must be disabled within Flash and Director players.

4.2.5.1 Flash embed syntax

The full syntax for an index.html file containing embedded Flash content with TLF branding is:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"
"http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
  <head>
    <title><!-- tlf:content-title --></title>
    <link rel="stylesheet" type="text/css" media="screen"
href="css/mystyle.css" id="default" title="default" />
  </head>
  <body>
    <div id="divcenter">
      <object type="application/x-shockwave-flash" data="loader.swf"
width="100%" height="100%" id="loader">
        <param name="movie" value="loader.swf" />
        <param name="quality" value="high" />
        <param name="scale" value="showall" />
        <param name="bgcolor" value="#ffffff" />
        <param name="menu" value="false" />
        <param name="seamlesstabbing" value="false" />
      </object>
      <br />
      <div class="logo">
        
    </div>
    <div class="partnerlogo">
        <!--  -
-->
    </div>
    <div class="copyright-conditions">
        <span id="copyright"><!-- tlf:content-rights --></span>
        <span id="conditions"><!-- tlf:conditions-of-use --></span>
    </div>
</div>
</body>
</html>
```

This should be styled with the following CSS²:

```
html, body
{ height: 100%; color: black; background: #ffffff; font-size: 1em; padding-bottom:
17px; margin-top: 1px;}
#divcenter
{height: 95%; text-align: center;}
#conditions
{font-family: Arial Unicode MS, Helvetica, Lucida Sans Unicode, sans-serif; font-size:
8pt; color: #000000; display: block;}
#copyright
{font-family: Arial Unicode MS, Helvetica, Lucida Sans Unicode, sans-serif; font-size:
8pt; color: #000000;}
#bold-italic {font-style: italic; font-weight: bold;}
.logo {width: 20%; float: left;}
#logo {width: 103px; height: 48px; float: left;}
.partnerlogo {width: 20%; float: right;}
.copyright-conditions {float: left; width: 59.5%;height: 48px; vertical-align: text-
bottom;}
#conditions a,#conditions a:link {color: #0000ff;}
#conditions a:visited {color:#990099;}
#conditions a:active {color:#aaffaa;}
#conditions a:hover, #conditions a:focus {color:#ff0000; outline:#000000 solid 1px;}
```

4.2.5.2 Director (Shockwave) embed syntax

The full syntax for an index.html file containing embedded Director content with TLF branding is:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"
"http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
  <head>
```

² This CSS facilitates the rendering of content using TLF's embed syntax in Mozilla Firefox, Mozilla and Netscape 7 For more information on embedding content in XHTML, see <http://jira.thelearningfederation.edu.au/confluence/x/0wM>

```

<title><!-- tlf:content-title --></title>
<link rel="stylesheet" type="text/css" media="screen"
href="css/mystyle.css" id="default" title="default" />
</head>
<body>
  <div id="divcenter">
    <object type="application/x-director" data="index.dcr" width="760"
height="570" id="object">
      <param name="src" value="index.dcr" />
      <param name="swStretchStyle" value="none" />
      <param name="swRemote"
value="swSaveEnabled='true'
swVolume='false' swRestart='false'
swPausePlay='false'
swFastForward='false'
swTitle='My object'
swContextMenu='false'" />
      <param name="bgColor" value="#FFFFFF" />
    </object>
    <br />
    <div class="logo">
      
    </div>
    <div class="partnerlogo">
      <!-- 
-->
    </div>
    <div class="copyright-conditions">
      <span id="copyright"><!-- tlf:content-rights --></span>
      <span id="conditions"><!-- tlf:conditions-of-use --></span>
    </div>
  </div>
</body>
</html>

```

4.2.6 SVG

Where appropriate, graphics should be represented in Scalable Vector Graphics (SVG) version 1.0 format. Refer to the W3C recommendation at <http://www.w3.org/TR/SVG/>. SVG is a language for describing two-dimensional graphics in XML.

4.2.7 SMIL

Where appropriate, the structure of simple interactive audiovisual presentations should be represented using Synchronised Multimedia Integration Language (SMIL) version 2.0. Refer to the W3C recommendation at <http://www.w3.org/TR/smil20/>. SMIL is used for synchronising simple multimedia presentations, which integrate streaming audio and video with images, text or any other media type.

4.2.8 MathML

Where appropriate, mathematics should be represented in Mathematical Markup Language (MathML) version 2.0. Refer to the W3C recommendation at <http://www.w3.org/TR/MathML2/>. It facilitates creation of mathematical expressions within web pages and the use of these expressions by other applications such as voice synthesis.

4.2.9 QTI

Activities for evaluating and supporting student understanding may be represented using the IMS Question & Test Interoperability (QTI) specification XML binding, version 1.2. Refer to the IMS Global Learning Consortium, Inc. website at <http://www.imsglobal.org/question/>. QTI describes an XML structure for the representation of basic question (item) and test (assessment) data and their corresponding results reports.

4.3 Style sheets

4.3.1 CSS

Document presentation must be defined using Cascading Style Sheets CSS1 and/or CSS2.1. Refer to 'Cascading Style Sheets', W3C website at <http://www.w3.org/Style/CSS/>.

4.3.2 Linking style sheets

Styles should be contained in an external linked .css file, using the <link> element.

Example:

```
<link rel="stylesheet" type="text/css" href="mystyle.css"/>
```

4.4 Images and graphics

To support the stated principle of durability, The Learning Federation may require supply of nominated images in high-resolution formats and supply of graphics in SVG format.

4.4.1 Images

Images must be specified in one of the following formats:

- PNG / MNG version 1.0 format – refer to 'PNG', W3C website at <http://www.w3.org/Graphics/PNG/>
- JPEG format – refer to the JPEG website at <http://www.jpeg.org/>.

Images must *not* be represented in GIF format.

4.4.2 Graphics

Graphics must be represented in one of the following formats:

- Scalable Vector Graphics (SVG) version 1.0 format – refer to 'Scalable Vector Graphics (SVG), XML graphics for the web', W3C website at <http://www.w3.org/Graphics/SVG/>
- PNG / MNG version 1.0 format – refer to 'PNG', W3C website at <http://www.w3.org/Graphics/PNG/>
- JPEG format – refer to the JPEG website at <http://www.jpeg.org/>

4.5 Audio

Audio must be supplied in MPEG Audio Layer 3 (MP3) format. Refer to 'Coding of moving pictures and audio', the International Organisation for Standardisation website at <http://www.chiariglione.org/mpeg/standards/mpeg-1/mpeg-1.htm>

To support the stated principle of durability, The Learning Federation may require supply of nominated audio in a high quality, uncompressed format.

4.6 Video

Video must be supplied in one of the following formats:

- QuickTime (v7.1.6) format – refer to <http://www.apple.com/quicktime/>
- MPEG-4 video format ('ISO/IEC 13818) – refer to 'MPEG standards', the International Organisation for Standardisation website at <http://www.chiariglione.org/mpeg/standards.htm>.

To support the stated principle of durability, The Learning Federation may require supply of nominated video in a high-resolution, uncompressed format.

4.7 Colours

4.7.1 Colour independence

Do not rely on colour alone to convey information.

If colour alone is used to convey information, people who cannot differentiate between certain colours and users with devices that have non-colour or non-visual displays will not receive the information.

Examples:

- An instruction on a screen that states: 'Select the blue button for yes' may not be perceived by a user without a colour display. In this case, the button should also contain the word 'yes' so that the instruction is clear and understandable.
- If a button changes from red to green when it is activated, a colour-blind user may not be able to perceive that it has been activated. In this case, activation should also be indicated using a button offset, resizing the button, or movement surrounding the button.
- A graph or chart that uses colour-alone to indicate the results of a test may mean that a user with colour discrimination issues may not be able to complete the activity or may misinterpret the data. In this case, texture, shading or labels on the data should also be included to ensure that the user can differentiate different data.

4.7.2 Contrast

Foreground and background colour combinations must provide sufficient contrast when viewed by someone having colour deficits or when viewed on a black-and-white screen or when printed.

4.8 Device independence

All content should be designed to be device independent. In other words, a user should be able to interact with the content with their preferred input (or output) device and switch between devices should they wish to do so.

Generally, interactive content that allows keyboard interaction is also accessible through speech input or a command line interface. Where learning objects include sophisticated

interactivity such as a timed game activity, keyboard access and tab order may not be appropriate.

This section uses the following definitions of user interface events³:

- Gain focus

An interactive element **gains focus** when a pointing device is moved onto the element or tabbing navigation moves onto the element.

- Lose focus

An interactive element **loses focus** when a pointing device is moved off the element or when tabbing navigation moves out/off of the element.

- Activated

An interactive element is **activated** when it has focus and a mouse is clicked or key is pressed.

4.8.1 Keyboard and pointing device access

Every interactive element, script and applet should be able to **gain focus, lose focus and be activated** with both a keyboard and a mouse.

Available interactive elements must show a clear change of state when they gain focus, lose focus and are activated. There must be a discernable difference between the unfocused, focused, and activated states.

Disabled or unavailable interactive elements must be clearly unavailable, for example masked but visible, and should not show any change of state.

Focus areas must be clearly defined and must not obscure other elements.

4.8.2 Default keyboard interaction

Keyboard navigation should consist of logical keyboard interaction with all interactive elements. The tab order should facilitate the user's navigation through the object in the order required to complete the learning task.

Please note: The SPACE BAR must not be used as the only mechanism to activate interactive elements. It may be used to provide an alternative to the ENTER key but the ENTER key must always operate.

The following keyboard interaction must be supported:

Key	Purpose
TAB	Move focus between interactive elements. Tab order should facilitate the user's navigation through the object in the order required to complete the learning task.
ENTER	Activate an interactive element that has focus.
ARROW keys	Move interactive components as part of drag and drop functionality.

³ User interface event definitions are based on the definitions in the W3C Document Object Model (DOM) Level 2 Events Specification at <http://www.w3.org/TR/DOM-Level-2-Events>

4.8.3 Grouping interactive elements

Where appropriate, related interactive elements should be grouped and identified. In these cases a way to bypass interaction with the group of elements must be provided. The keyboard interaction outlined for grouping should be applied to drop-down menus or pick lists.

When element grouping is supported, the following keyboard interaction must be supported in addition to the default keyboard interaction:

Key	Purpose
TAB	Move focus between groups of interactive elements, allow the group to be bypassed and allow the user to exit the group.
ENTER	Enter and activate a group of interactive elements.
Arrow keys	Move between/cycle through elements in a group of elements.

Example:

- a) Allow a user to TAB to a group, press ENTER to enter a group, use arrow keys to move around a group, use ENTER to select an element within a group, and TAB to exit a group.
- b) Allow a user to TAB to a drop down menu, press ENTER to enter the drop-down list, use arrow keys to move up and down the list, use ENTER to select an item within the list, and TAB to exit the list.

4.8.4 Pop-up windows

Users must be informed before a child browser window (HTML pop-up) is opened over the top of the parent window. This is not necessary for Flash/Shockwave pop-up windows.

HTML and Flash/Shockwave pop-up windows must include a mechanism (such as a button) to close the child window and return to the parent window. On launch of the pop-up focus must be given to the child window and no elements of the parent window should be accessible until such time as the child window is closed. Focus must return to the parent window after a child window has been closed.

4.8.5 Data input

Data input fields (such as text boxes) must be accessible with both a keyboard and a pointing device.

Content must indicate when invalid data has been entered into a data input field.

4.8.6 Data output

Where a print function is included in the learning object, the output must be printer independent and legible.

4.9 Content delivery considerations

All presentation formats must be able to be retrieved via a standard http connection using a standard HTTP/1.1 server.

Content must *not* rely on:

- a streaming server
- client or server-side caching.

4.10 Document formats

4.10.1 Text documents

Text files may be used to store simple data, such as the value of a configuration variable. When more than a few bytes of information are stored in a text file, or when the information contains detailed structure, consideration should be given to storing the information as XML.

4.10.2 PDF

Documents used in offline activities, such as activity sheets or templates, should be represented in PDF format.

4.10.3 Other document formats

The Le@rning Federation will consider the following alternatives to PDF. Approval of these formats will be considered on a case-by-case basis:

- Microsoft Word
- Microsoft Excel
- Microsoft PowerPoint.

To support the stated principle of durability, The Le@rning Federation may require supply of nominated documents in an accompanying RTF format version.

4.11 Scripts

4.11.1 ECMAScript (JavaScript)

ECMAScript is a scripting programming language, **JavaScript** is an extension of the ECMA-262 standard. It should be used to add functionality to web pages, SVG and SMIL resources.

Client-side ECMAScript must conform to ECMA Standard-262, ECMAScript Language Specification, Revision 3 – refer to Ecma International website at <http://www.ecma-international.org/publications/standards/Ecma-262.htm>.

Client-side ECMAScript must be contained in an external linked .js file, using the XHTML `<script>` element.

Example:

```
<script type="text/javascript" src="js/script.js"></script>
```

4.11.2 ActionScript

ActionScript should be used to add functionality to Flash objects.

4.11.3 Lingo

Lingo should be used to add functionality to Shockwave objects.

4.12 Web applications

4.12.1 Server independence

All web applications must operate as standalone objects that do not require interaction with a server.

4.12.2 Allowed application formats

Web applications must be supplied in one of the following formats:

- Scalable Vector Graphics (SVG) version 1.0 format, W3C website at <http://www.w3.org/Graphics/SVG/>, using ECMAScript to implement interactive functionality
- Flash, using ActionScript, playable in Flash Player 9
- Shockwave using Lingo, playable in Shockwave Player v10 (full installer)
- Java applets that run with version 1.4.2 (and higher) of the Java Virtual Machine
- QuickTime VR, playable in QuickTime Player v7.1.6

4.13 Performance

4.13.1 Interaction time

Learning content will be frequently used over limited bandwidth connections. For this reason, a restriction is placed on user interaction time with the learning content.

Interaction time is defined as the time between a user making a request to the learning content, such as a mouse click or keyboard press, and the user actively and continuously interacting with the requested component of the learning content. Performance is not evaluated solely on initial load time; the user must be able to continuously interact with the learning content at 10 second intervals.

As a baseline all Learning content produced for The Learning Federation must have an interaction time of less than 10 seconds over a dedicated 64 kbit/s connection⁴, however in circumstances where restricting the bandwidth to 64 kbit/s limits the educational integrity of the content and decreases the capacity of the content to meet educational soundness requirements The Learning Federation may agree to an exception to the requirements and test to an interaction time of less than 10 seconds over a dedicated 256 kbit/s connection. Where an exception is agreed the exception will be recorded in the 'Other platform requirements' metadata element in the Exchange and that information will be available in the content package.

4.13.2 Overall size

The learning content will be downloaded and replicated over limited bandwidth connections. For this reason, developers should endeavour to minimise the overall size of the learning content. Additionally, large learning content should be divided into small components in order to maximise load efficiency.

As a guide, the maximum size of TLF content should be 5 MB.

4.13.3 Ordering interactivity

Components of a learning task must load in the correct sequence for the user to accomplish that task.

Example:

- a) If an introduction to an activity includes text instructions, an image and an enter button, then the enter button should not be available before the text and image appear onscreen or there is a risk that the student may progress without sufficient information.

⁴ Where 'dedicated' means only one user is using the connection.

- b) If a task includes text instructions, a video and an interactive quiz dependent upon the user viewing the video, then the video should completely load before the user is required to complete the quiz.

Interactivity order must be maintained over varying bandwidth connections. That is, components of the task must load in the same order over both low bandwidth and high bandwidth connections.

4.14 Persistence

Learning objects may remember state accumulated during an interaction with the user. For example, a learning object may remember the user's name so the on-screen text includes the user's name.

For privacy reasons, state must expire when the user has completed the learning object or at the end of the browser session. State must *not* be maintained across a learning object or the browser sessions.

4.15 Screen layout

4.15.1 Dimensions

Learning content should be optimised to display on a screen resolution of 1024 x 768 pixels.

As many factors ultimately determine the final dimensions of the available area, the following should be used as a guide.

- Where a web page contains standalone embedded objects, each object should be completely visible and operate successfully when it is displayed in an area of 760 x 570 pixels. This represents a 4:3 aspect ratio on a monitor with a screen resolution of 1024 x 768 pixels (less the header and navigation areas).

4.15.2 Scaling

Resources should scale to fill available space gracefully when a browser is resized. Techniques for graceful scaling include using relative rather than absolute size definitions within XHTML and use of scalable vector graphics within Flash objects.

When a standalone object does not have the ability to scale, the recommended fixed size of the object is 760 x 570 pixels. The fixed size of the learning object may be smaller, but no larger than 760 x 570 pixels.

4.15.3 Centre standalone objects

Where a web page has a single standalone object embedded, such as SVG, Flash, Shockwave or a QuickTime object, it must be centred vertically and horizontally in the page.

4.16 Directory structure

Learning objects must be delivered to The Learning Federation as a directory structure containing resources.

4.16.1 XHTML start file

The top level of the directory structure must contain an XHTML file named `index.html` that represents a common starting point for interacting with the learning object.

4.16.2 Flash and Director files

Flash and Shockwave applications must have a Flash or Director file named `index.swf` or

`index.dcr` in the top level of the directory structure.

These files are crucial because Flash and Director applications cannot resolve relative paths on some platforms.

The top level of the directory structure for Flash and Director learning objects must also contain a `container/loader.swf` to combat a known interoperability issue related to streaming and the `<object>` element with certain browsers.⁵

Shockwave applications must have all `.dcr` files located within the top level of the directory structure of the learning object and external assets located within a sub-folder under a `dswmedia` directory. This is required for Shockwave to load external assets from local disks.

All objects will contain a `shared_assets` directory on publication; this directory will contain standard set of files; any logo files and applicable system images and icons.

If not affected by the requirements outlined above all other files must be stored in directories named by file extension or mime type.

4.16.3 File and directory names

File and directory names must use the following:

- lower case alphanumeric characters (a to z, 0 to 9)
- the characters - (dash) and _ (underscore)
- the slash character '/' to separate hierarchical parts of a file path.

File and directory names must *not* use the space character.

These requirements ensure that file and directory names can be easily referenced in URLs and are portable between operating systems.

4.17 Rights information

4.17.1 Copyright statement

The Le@rning Federation copyright statement must be placed in the XHTML start file of each learning object and digital resource, digital resources may also incorporate rights information for partner institutions:

The standard statement should read: © Curriculum Corporation, 2007, except where indicated under Acknowledgements

- The year date refers to the year in which the majority of the development occurred.
- The copyright statement must be coded in the XHTML start file and styled using CSS stylesheet as follows:
 1. Arial 8 point font.
 2. High contrast should be provided between foreground and background colours.
 3. The statement footer should be centred.

Note: The statement may be varied to include partner institutions:

© Curriculum Corporation and Partner institution, 2007, except where indicated under Acknowledgements

⁵ For more information, see 'Streaming and the object element' at <http://jira.thelearningfederation.edu.au/confluence/x/0wM>

4.17.2 Printable outputs

The printable output footer must be placed on all printable outputs.

The footer should read: Printable worksheet made available by The Le@rning Federation: © Curriculum Corporation, 2007, except where indicated under Acknowledgements

- The year date refers to the year in which the majority of the development occurred.
- The printable output footer should be styled as follows:
 1. Onscreen appearance of Arial 8 point font.
 2. High contrast should be provided between foreground and background colours.
 3. The statement footer should be left aligned.

4.18 Fonts

In order for a font to display correctly on a user's computer, the computer should have access to that font. The content gains access to fonts by the following:

- accessing a font outline embedded within the content
- referencing the font from within the content and assuming that the font has been installed on the user's computer.

For reasons related to guaranteed presentation and layout, developers should use embedded font outlines rather than using device fonts (referencing fonts).

If with agreement of TLF fonts are referenced, developers must assume that only the following fonts are available on the user's computer:

Operating system	Font
Windows 2000	Fonts available with the default installation plus fonts installed with the 'Japanese, Simple Chinese, Traditional Chinese font' installation option.
Windows XP	Fonts available with the default installation plus fonts installed with the 'Install East Asian characters' installation option.
OS X	Fonts available with the default installation. Asian language fonts are available by default

4.19 Client software requirements

The Le@rning Federation content is viewed within Learning Management Systems (LMS) and web browsers. At the time of publication, no LMS or web browser supports all the content model requirements described in this document.

4.19.1 Minimum browser and operating system requirements

The Le@rning Federation tests all content in the following environments:

- Mozilla Firefox 2.0 and Microsoft Internet Explorer 6.0 on Microsoft Windows 2000

- Mozilla Firefox 2.0 and Microsoft Internet Explorer 7.0 on Microsoft Windows XP
- Apple Safari 2.0 on Apple OS X.

The operating systems must also have the fonts described in Section 4.18 'Fonts' installed.

Until software supports the standards and specifications defined in this document, content metadata should include information about the environments in which the content has successfully been tested. The metadata must also include information about any extensions to these environments needed to operate the content.

4.19.2 Latest browsers

Content should also be tested and operate in the latest versions of Mozilla Firefox and Microsoft Internet Explorer on Microsoft Windows 2000 and Windows XP, and Safari on Apple OS X.

4.19.3 Browser plug-ins

The browser plug-ins listed in the table are commonly used to render The Learning Federation content. Content containing resources with the formats must be tested and operate with the plug-ins listed in the table.

Format	Plug-in
SVG	Adobe SVG Viewer Plug-in http://www.adobe.com/svg/
QuickTime	QuickTime 7.1.6 Player http://www.apple.com/quicktime/download
PDF	Adobe Acrobat Reader 5 http://www.adobe.com/products/acrobat/readstep.html
Flash MX	Adobe Flash Player 9 http://www.adobe.com/shockwave/download/download.cgi?P1_Prod_Version=ShockwaveFlash
Shockwave	Macromedia Shockwave Player 10 (full installer) http://www.macromedia.com/shockwave/download/alternates/
Java applets	J2SE Runtime Environment (JRE) 5.0 http://www.java.com/en/download/manual.jsp

4.20 Client hardware requirements

Content must at least operate on systems configured according to the minimum hardware requirements as specified by the nominated browser and plug-in manufacturers:

- Internet Explorer
<http://www.microsoft.com/windows/ie/evaluation/sysreqs/default.asp>
- Mozilla Firefox
<http://www.mozilla.org/products/firefox/>
- Flash
http://www.macromedia.com/support/flash/releasenotes/mx/rn_mx.html#systemrequirements
- Shockwave
<http://www.macromedia.com/software/shockwaveplayer/productinfo/systemreqs/>