THE le@rning FEDERATION

schools online curriculum content initiative

EDITORIAL PROCESS GUIDELINES

for Developers

VERSION: 1.4

DATE: 21 MAY 2007

Disclaimer

If the material inserted into this template may be relied upon by third parties and consequently may lead to loss or damage, it is important to include an appropriate disclaimer. Please seek advice from The Le@rning Federation's legal counsel in relation to the inclusion and drafting of a disclaimer.

If you are unable to seek advice for any reason, the general disclaimer below may serve as interim protection for the joint venture companies:

"The material contained in the Editorial Process Guidelines for Developers 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 these guidelines."

Table of contents

Introduction

This document outlines The Le@rning Federation's (TLF) editorial process. It provides an overview of the process for content developers commissioned by TLF. For style guidelines, content developers should refer to the separate document entitled *Editorial Guide for Learning Objects*.

Editorial reviews at TLF form a key part of the quality assurance (QA) process. The editorial process described within this document is the 'default' process. Any decision to deviate from the standard editorial process must be agreed jointly by the relevant project manager (PM) and the Coordinating Multimedia Editor (CME).

1 Collaborative approach

Members of TLF's editorial team perform QA tasks. They work collaboratively with TLF project managers, as well as other TLF QA staff. TLF editors may also liaise with commissioned developers, writers and subject matter experts.

2 Overall aim

The overall editorial aim is to increase a learning object's chances of success by ensuring the content is clear, consistent and achieves its intended learning outcomes. Integral to this aim is the production of a high-quality learning object. In order to achieve this goal, an editorial review process has been identified. This process incorporates a number of stages, or review types.

3 Review types

Each head of series (HOS) or first of series (FOS) learning object undergoes three key editorial reviews. These reviews take place at defined points during the development phase. The reviews are:

- structural edit
- multimedia copy-edit
- conformance review.

The following review is also usually required:

- editorial recheck.

The key aims of each review type are summarised in the following table. See Figure 1 at the end of this document for a workflow diagram.

Review name	Aims
Structural edit	To perform a structural edit of the content within the functional specification. To identify any potential issues with the on-screen realisation of the content. To identify and list any missing content. To ensure that issues arising from in-school evaluation (ISE) have been adequately addressed.
Multimedia copy-edit	To copy-edit the scripted content in the functional specification in conjunction with the Beta version of the learning object. To ensure that the content 'works' on screen. To review and finalise the audio elements of the script before the professional recording session. To check that any issues identified in the previous editorial review have been satisfactorily addressed.
Conformance review	To check that the fully functional learning object is editorially conformant. To check that all open issues from the previous editorial review have been fixed.
Editorial recheck	To check that all open issues from the previous editorial review have been fixed.

NB: Most HOS and FOS learning objects undergo ISE. An ISE editorial review takes place before a learning object is taken into a school. This review is performed by the relevant TLF project manager.

4 Process assumptions

The following table summarises key editorial process assumptions. Developers should adhere to these processes. Failure to do so can result in unnecessary work and expense for both parties.

	Structural edit	**Copy-edit**	**Conformance review**
Learning object (LO)	Prototype if relevant	Beta version of object (content complete minus final audio)	Fully functional object
Process assumptions	All on-screen text and audio text has been documented within a logically organised functional specification. Each element of text and each audio component have been assigned a unique ID within the functional specification.	The professional recording session should not take place until after this review. Placeholder audio files have been used in the Beta object. An up-to-date functional specification has been submitted to TLF.	Content is complete. Final audio files have been integrated into the object. An up-to-date functional specification has been submitted to TLF.

4.1 Documentation

Up-to-date documentation is a key tool in reducing QA risks. In order for editorial reviews to take place efficiently and effectively, content developers must supply TLF with complete and up-to-date documentation. Furthermore, up-to-date documentation is essential for future content updates or localisation of content.

The following table lists the documents that content developers must provide for each type of editorial review.

Structural edit	**Copy-edit**	**Conformance review**	**Editorial recheck**
Complete functional specification, which includes all proposed on-screen and audio text If applicable, photocopies of third-party materials such as extracts from published texts	Up-to-date functional specification If applicable, developer response to issues from previous editorial review	Up-to-date functional specification Developer response to issues from editorial previous review	Up-to-date functional specification Developer response to issues from editorial previous review

5 Types of series

The majority of TLF's learning objects are developed as part of a series. For development purposes, TLF has identified three main types of series:

1. Head of series (HOS) aggregate
2. Head of series (HOS) template
3. First of series (FOS)

Each series of objects consists of a HOS or FOS object, and at least one rest of series (ROS) object. The type of learning object, and its place within a series of learning objects, will determine the applicable QA stages. The TLF PM agrees the editorial development approach for each series of objects with the CME.

In some circumstances, it may be agreed to develop stand-alone objects. That is, objects that are not part of a series. Stand-alone objects undergo all the editorial reviews outlined in Section 3.

1 HOS aggregate

HOS LO: The learning object (LO) is built as an aggregate that will eventually be split into parts, or subobjects (known as rest of series, or ROS, subobjects).

Example: sc_003_csiro_800

ROS LOs: The ROS LOs share the same code base as the relevant parts of the aggregate.

Examples: sc_003_csiro_810, 820, 830, 840

Differences: Entry and exit screens of each subobject will be slightly different in content, though usually identical in functionality to the HOS.

Implication: There is one functional specification for the series of learning objects. The document should state how the aggregate will be split into parts. This information should be included in the version of the functional specification that is submitted for the HOS structural edit.

2 HOS template

HOS LO: The LO is built within a template that will eventually be used for other LOs in the series.

Example: lo_002_nl_111

ROS LOs: The ROS LOs share exactly the same code base as the HOS LO.

Example: lo_002_nl_214

Differences: The content differs in each LO within the series, but the structure and functionality is identical ('content' includes on-screen text, audio files and graphics). In some cases, only the on-screen text will change.

Implication: Each LO in the series will have its own functional specification. The HOS specification is created and reviewed before any writing commences on the ROS functional specifications. Each ROS document should be created by repopulating the content in a copy of the latest version of the HOS functional specification.

3 FOS

FOS LO: The LO is the first in a series of LOs that are linked thematically and share some functionality, but do not use exactly the same code base.

Example: ma_002_cadre_120

ROS LOs: The ROS LOs share the same theme and some, but not all, functionality with the FOS LO.

Examples: ma_002_cadre_100, 110

Differences: The content differs, either slightly or significantly, between each LO within the series, and so does the structure and functionality.

Implication: Each LO in the series will have its own functional specification. The HOS specification is created and reviewed before any writing commences on the ROS documents.

To ensure an efficient development process, it may be necessary to identify subseries of LOs.

5.1 Implications of series types

The table below shows which editorial reviews are required for each type of learning object.

	Structural	Copy-edit	Conformance
HOS aggregate	yes	yes	yes
ROS subobjects	no	no	yes
HOS template	yes	yes	yes
ROS objects	no	yes (paper-based, no Beta needed)	yes
FOS object	yes	yes	yes
ROS objects	negotiable: TLF PM & CME to discuss	yes (Beta needed)	yes
Stand-alone objects	yes	yes	yes

6 Recording sessions: editorial considerations

Before a recording session takes place, a TLF editor will check the proposed script in conjunction with an interactive version of the learning object. During this check, the editor may identify words that need special treatment during a recording session. For example, the editor may note the correct pronunciation of an unusual word or name. The editor may also include 'directions' to the voice artist. For example, the editor may note words that need to be stressed within a sentence. It is vital that this information is taken to the recording session.

In many cases, audio files have on-screen text equivalents. To reduce the likelihood of mismatches between audio and on-screen text, and to ensure all scripts are recorded, developers should:

- provide the recording studio with an up-to-date copy of the script
- ensure the session is appropriately directed.

At the recording session, someone needs to direct the voice artist. This person should not be the sound engineer. The person who directs the session should:

- check that the voice artist reads the script word-for-word
- ensure the voice artist uses appropriate stress, expression and characterisation.

See the table on the next page for a list of audio-related editorial issues, as well as recommended steps to avoid each problem.

Potential problem in learning object	Risk management
Missing audio files	Use an up-to-date script at the recording session. Use the script as a checklist. Ensure an appropriate person directs the recording session.
Mismatch of audio to on-screen text	Use an up-to-date script at the recording session. Use the script as a checklist. Record an alternative version if there is any uncertainty about the correct way to structure or pronounce an audio element. Ensure an appropriate person directs the recording session. Ensure the script within the functional specification is updated with any decisions made at the recording session.
Poor delivery of script	Ensure an appropriate person directs the recording session. Brief the voice artist appropriately. Use a professional voice artist.

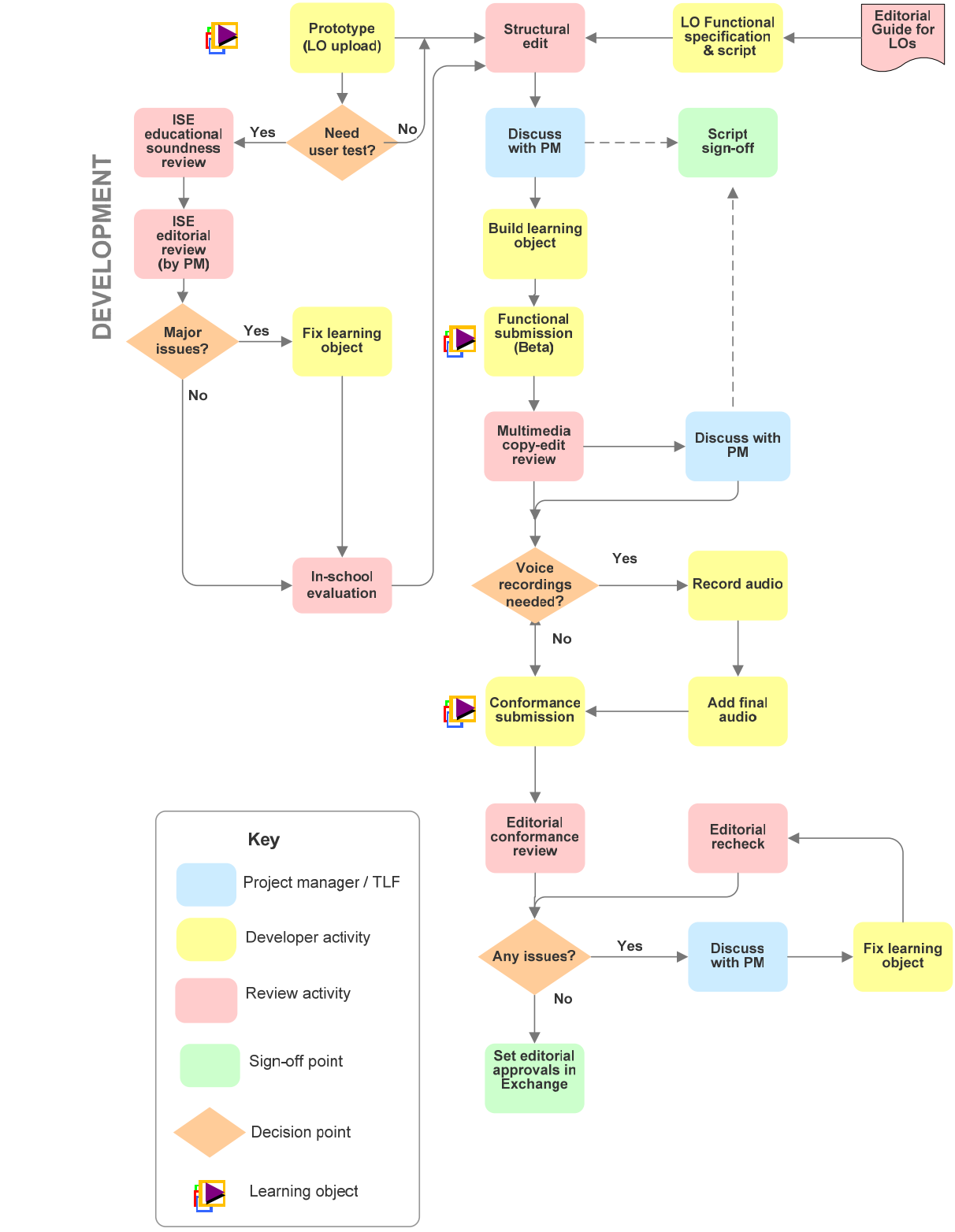

Figure 1: Editorial workflow diagram

