# Final Cut Pro X XML Format

Developer

# Contents

# Tables and Listings

# About Final Cut Pro X XML Version 1.2

This document describes the elements of the Final Cut Pro X XML (FCPXML) file format, version 1.2. This file format lets you import and export XML files between Final Cut Pro X and third-party applications. The FCPXML v1.2 file format requires Final Cut Pro X v10.0.6 or later.

> **Important:** FCPXML v1.2 describes certain project data that is useful for exchanging with other applications. It does not describe all possible data, and therefore is not a substitute for the native project file when archiving a project or moving a complete project from one system to another.

## At a Glance

You use the Final Cut Pro X application to create, edit, and produce high-quality video. The FCPXML file format allows you to transfer the details of your Events and projects between Final Cut Pro X and third-party applications, devices, and media asset management tools that do not natively recognize Final Cut Pro X Events or projects.

Final Cut Pro X allows you to export and import FCPXML files to accomplish the following tasks:

- Exchange Final Cut Pro X Event and project data with other applications.
- Create new Final Cut Pro X Events and projects.
- Import new media into existing Events.

FCPXML v1.2 adds support for the following Final Cut Pro X features:

- Audio Component Editing—allows you to define how audio components are used (for example, channel mappings, enabled, role, effects, and mute ranges).

  **Relevant Section:** "Audio Components" (page 15)

- Metadata Export/Import—allows you to import and export metadata associated with media assets or clips in a project or Event.

  **Relevant Chapter:** "Metadata" (page 29)

FCPXML v1.1 adds support for the following Final Cut Pro X features:

- Multicam editing—allows you to manage media from multiple cameras or other synchronized footage.

  **Relevant Chapter:** "Resources" (page 17)

- Adjustments
  - Video and image transformation—allows you to crop, trim, pan and zoom, distort, conform, move, resize, rotate, and blend videos and images.
  - Audio adjustment—allows you to adjust audio volume and panning.

  **Relevant Chapter:** "Adjustments" (page 24)

- Retrieve ASC CDL color information—allows you to export the first primary color correction definition in ASC CDL (American Society of Cinematographers Color Decision List) format.

  **Relevant Chapter:** "Exports with Color Corrections" (page 33)

## How to Use This Document

For definitions and discussions of Final Cut Pro X concepts and terminology, see "Events and Projects" (page 8), "Story Elements" (page 12), and "Resources" (page 17). Discussions are supported by examples.

To understand how elements are used, and for reference lists of available elements and the time values used with them, see "Story Elements" (page 12).

To enhance your project with effects, "Adjustments" (page 24) describes audio and video adjustment elements that allow you to crop, zoom, distort, blend, rotate, and more.

To understand how to import and export asset and clip metadata in your Events and projects, including how to incorporate custom keys, see "Metadata" (page 29).

If your project includes color corrections, "Exports with Color Corrections" (page 33) describes how Final Cut Pro X uses the `<info-asc-cdl>` element.

To use the Final Cut Pro X Document Type Definition (DTD) to validate FCPXML documents in your own applications prior to import, see "Appendix D: FCPXML Version 1.2 DTD" (page 65). Earlier DTD versions are also available in "Appendix B: FCPXML Version 1.0 DTD" (page 42) and "Appendix C: FCPXML Version 1.1 DTD" (page 51).

## Prerequisites

This document assumes you have an understanding of XML and have used Final Cut Pro X.

## See Also

The following resources may be helpful as you work with the FCPXML format:

- The Final Cut Pro X User Manual.
- The Extensible Markup Language (XML) 1.0 specification.

# Events and Projects

## Overview

In Final Cut Pro X, you use FCPXML (`.fcpxml`) files to export and import Event and project documents. Both document types manage collections of clips.

- A **clip** is a reference to media, such as a video, audio, or still image file, that allows you to edit and annotate the media without directly modifying it. A clip controls which portions of the media you would like to use, and it allows you to organize the media based on keywords you have applied. Clips can also contain other clips to represent composite media.
- Use a Final Cut Pro X **Event** to store and organize media. You can import media files into a new or existing Event. You can choose to copy these files into an Event's own media folder, or reference them in their original locations. Final Cut Pro X tracks each imported file as an asset and ensures your Event contains at least one clip per asset.
- Use a Final Cut Pro X **project** to build a finished movie. A project has a primary container called a sequence that defines your movie's final appearance. You build a sequence by bringing clips into it from one or more Events, or by creating new clips within the sequence. You adjust and arrange the clips, along with other story elements in the sequence, to produce your movie. Every clip in a project is unique to that project (not shared), but referenced media always resides in an Event and may be shared across more than one project.

## About Events and Projects

An FCPXML file contains a root `<fcpxml>` element with a `<project>` element that describes a Final Cut Pro X Event or project.

A `<project>` element contains either:

- A `<sequence>` element (for a Final Cut Pro X project)
- A set of `<clip>`, `<audition>`, `<mc-clip>`, and `<ref-clip>` elements (for a Final Cut Pro X Event)

See "Story Elements" (page 12) for further details on these elements.

A <project> element also includes a <resources> element that lists additional data the project depends upon, including external assets, video formats, and effects. See “Resources” (page 17) for details on the supported resource types.

Listing 1-1 provides an example of an FCPXML project:

**Listing 1-1** A project referencing audio and video in an external Event

```
<fcpxml version="1.0">
    <project name="MyProject">
        <!-- Project Resources -->
        <resources>
            <projectRef id="r1" name="MyEvent"/>
           <asset id="r2" projectRef="r1" src="file:/Volumes/Media/MyVideo.mov"/>
            <format id="r3" name="FFVideoFormat1080p30"/>
        </resources>
        <!-- Project Story Elements -->
        <sequence format="r3">
            <spine>
                <video ref="r2" duration="5s">
                    <audio lane="-1" ref="r2" duration="5s"/>
                </video>
            </spine>
        </sequence>
    </project>
</fcpxml>
```

In Listing 1-1, the MyProject project contains three resources:

- A reference to an external Event called MyEvent.
- A reference to an external asset located at /Volumes/Media/MyVideo.mov, managed by MyEvent.
- A built-in video format, FFVideoFormat1080p30.

When Final Cut Pro X imports this file, it:

- creates a new Event called MyEvent in the default location.
- imports MyVideo.mov into MyEvent using the default preference for copying versus linking media.

- creates a new project called `MyProject` with a sequence using a 1080p, 30fps video format specified by `FFVideoFormat1080p30`.
- inserts the first 5 seconds of video from `MyVideo.mov` into the project.
- anchors the first 5 seconds of audio from `MyVideo.mov` one lane below the video. See "Story Elements" (page 12) for details on the `lane` attribute.

Listing 1-2 provides an example of an FCPXML Event:

**Listing 1-2** An Event with multiple clips

```
<fcpxml version="1.0">
    <project name="MyEvent">
        <!-- Project Resources -->
        <resources>
            <format id="r1" name="FFVideoFormat1080p30"/>
            <asset id="r2" src="file:/Volumes/Media/MyVideo1.mov"/>
            <asset id="r3" src="file:/Volumes/Media/MyVideo2.mov"/>
            <asset id="r4" src="file:/Volumes/Media/MyVideo3.mov"/>
        </resources>
        <!-- Event Clip Elements -->
        <clip name="MyVideo1" duration="45s" format="r1">
            <video ref="r2" duration="45s">
                <audio lane="-1" ref="r2" duration="45s"/>
            </video>
        </clip>
        <clip name="MyVideo2" duration="15s" format="r1">
            <video ref="r3" duration="15s">
                <audio lane="-1" ref="r3" duration="15s"/>
            </video>
        </clip>
        <clip name="MyVideo3" duration="30s" format="r1">
            <video ref="r4" duration="30s">
                <audio lane="-1" ref="r4" duration="30s"/>
            </video>
        </clip>
    </project>
```

```
</fcpxml>
```

In Listing 1-2, the Event `MyEvent` contains four resources:

- Three external assets named `MyVideo1.mov`, `MyVideo2.mov`, and `MyVideo3.mov` located at `/Volumes/Media/`.
- A built-in video format, `FFVideoFormat1080p30`.

When Final Cut Pro X imports this file, it:

- creates a new Event called `MyEvent` in the default location.
- imports `MyVideo1.mov`, `MyVideo2.mov`, and `MyVideo3.mov` into `MyEvent` using the default preference for copying versus linking.
- creates three clips (`MyVideo1`, `MyVideo2`, and `MyVideo3`) containing audio and video data from the imported media and using a 1080p, 30fps video format specified by `FFVideoFormat1080p30`.

# Story Elements

Story elements are the building blocks of your Final Cut Pro X Events and projects.

## Structure

An Event or project may include the story elements listed in Table 2-1.

**Table 2-1** Story Elements

| Element | Description |
|---|---|
| `<clip>` | A container for editing/compositing media and other story elements. |
| `<audio>` | A reference to audio data from an `<asset>` or `<effect>`. |
| `<video>` | A reference to video data from an `<asset>` or `<effect>`. |
| `<mc-clip>` | A reference to audio/video data from a `<multicam>` media source. |
| `<ref-clip>` | A reference to audio/video data from a `<sequence>` media source. |
| `<gap>` | A placeholder element with no intrinsic audio or video data. |
| `<transition>` | An effect that combines zero, one, or two neighboring elements. |
| `<title>` | An effect with custom text elements. |
| `<spine>` | A container that schedules elements sequentially in time. |
| `<audition>` | A container of alternative elements, exactly one of which is currently active. |
| `<sequence>` | A top-level element for a Final Cut Pro X project. |

Each story element allows one or more other elements to be *anchored* to it. An anchored item has a positive or negative `lane` index which positions the item either above or below its base element in the timeline.

For video elements, lane order also implies compositing order—items with higher lane indexes composite over elements with lower lane indexes. For audio elements, lane order does not affect compositing. Items that reside inside (rather than above or below) a container are called *contained items* and have an implied lane index of zero.

Events and projects may contain the keywords, markers, and notes listed in Table 2-2 that annotate certain elements.

**Table 2-2** Annotation Elements

| Element | Description |
|---|---|
| `<keyword>` | An annotation that applies to a range of time. |
| `<marker>` | An annotation (with optional *to-do* flag) that applies to a range of time. |
| `<chapter-marker>` | An annotation that applies to a range of time and indicates a chapter. |
| `<note>` | An annotation that applies to an entire element. |
| `<rating>` | A `favorite` or `reject` annotation that applies to a range of time. |

Events and projects may also contain filters (Table 2-3) that apply effects to certain elements.

**Table 2-3** Filter Element

| Element | Description |
|---|---|
| `<filter>` | A reference to an effect that applies to the parent element.<br>Refer to the DTD for specific information on the types of filters (audio, video, both) each parent element supports. |

## Time Values

Each story element has up to three time values, listed in Table 2-4, that schedule it (and its contained or anchored items) in a timeline.

**Table 2-4** Time Value Attributes

| Attribute | Description |
|---|---|
| `offset` | An element's location in parent time (or base element’s time for an anchor). |

| Attribute | Description |
|---|---|
| `duration` | An element's extent in parent time. |
| `start` | The start of an element's local timeline (to schedule its contained and anchored items). |

Time values are expressed as a rational number of seconds with a 64-bit numerator and a 32-bit denominator. Frame rates for NTSC-compatible media, for example, use a frame duration of "1001/30000s" (29.97 fps) or "1001/60000s" (59.94 fps). If a time value is equal to a whole number of seconds, the fraction may be reduced into whole seconds (for example, “5s”).

A time map allows you to adjust, or remap, the playback speed of an element. A time map is defined with the `<timeMap>` element which contains one or more `<timept>` elements that specify the new timing. The parent element’s playback speed is adjusted over a range of time using points on a curve to interpolate values. For example, if the local time range for a clip is originally 0-5s, then the time map for playing this clip at 50 percent speed, followed by playback at -50 percent speed, might look as follows:

```
<timeMap>
    <timept time="0s" value="0s" interp="linear"/>
    <timept time="10s" value="5s" interp="linear"/>    <!-- In 0-10s, play the
original 0-5s -->
    <timept time="20s" value="0s" interp="linear"/>    <!-- In 10-20s, play the
original 5-0s -->
</timeMap>
```

A time map affects only an element’s own duration and the offsets of the element’s anchored items; it does not modify the offsets of the element’s contained items (such as, items with `lane=0`).

Final Cut Pro X automatically applies rate conforming when the timeline frame rate and the media frame rate are certain combinations (see Table 2-5) by converting the media frame rate to match the timeline frame rate. When this occurs, the Rate Conform section appears in the application’s Video Inspector, and is also indicated in the exported XML (the media’s `<clip>` element contains a `<timeMap>` element with a `rateConform` attribute value of 1); the duration is also adjusted.

Table 2-5 illustrates how Final Cut Pro X applies rate conforming.

**Table 2-5** Rate Conforming

| Media Frame Rate | Timeline Frame Rate | | | | |
|---|---|---|---|---|---|
| | 23.98p | 24p | 25p<br>25i<br>50p | 29.97p<br>59.94p<br>29.97i | 30p<br>60p |
| **23.98p** | - | 24p | 25p | - | - |
| **24p** | 23.98p | - | 25p | - | - |
| **25p** | 23.98p | 24p | - | - | - |
| **29.97p** | - | - | - | - | 30p |
| **30p** | - | - | - | 29.97p | - |
| **50p** | 47.96p | 48p | - | - | - |
| **59.94p** | - | - | - | - | 60p |
| **60p** | - | - | - | 59.94p | - |
| **25i** | 47.96p | 48p | - | - | - |
| **29.97i** | - | - | - | - | 60p |

## Audio Components

For clips with audio, you can specify the configuration, such as channel mapping, effects, roles, mute ranges, and whether the configuration is enabled, at the component level using the audio component elements listed in Table 2-6.

**Table 2-6** Audio Component Elements

| Element | Description |
|---|---|
| `<audio-source>` | Defines the configuration for a single audio component from the contained items. |
| `<audio-aux-source>` | Defines the configuration for a single audio component from the anchored items. |
| `<mute>` | Suppresses audio output for the audio component over a range of time. |

The <audio-source> and <audio-aux-source> elements are very similar, the only difference being that the <audio-source> element specifies the configuration for the primary timeline, while the <audio-aux-source> element applies to the anchored timeline.

You can use <mute> or <filter> elements as children to specify mute ranges or audio effects.

In the following example, the audio from the primary timeline is reconfigured as dual mono while the audio from the anchored timeline is mapped to reverse stereo. Additionally, the last 20 seconds of the anchored audio is muted.

```
<clip duration="120s">
    <video ref="r2" duration="120s">
        <audio lane="-1" ref="r2" duration="120s" outCh="L, R"/>
    </video>
    <clip lane="-1" duration="120s">
        <audio ref="r3" duration="120s" outCh="L, R"/>
    </clip>
    <audio-source name="mono 1" srcCh="1" outCh="L"/>
    <audio-source name="mono 2" srcCh="2" outCh="R"/>
    <audio-aux-source name="reverse stereo" srcCh="2, 1" outCh="R, L"/>
        <mute start="100s" duration="20s"/>
    </audio-aux-source>
</clip>
```

# Resources

## Structure

An FCPXML file contains multiple references to shared elements, such as media files, video formats, and external Events. Each `<project>` element (for an Event or project) contains a `<resources>` section to track these shared references.

The `<resources>` section may include the elements listed in Table 3-1.

**Table 3-1** Resource Elements

| Element | Description |
|---|---|
| `<asset>` | A reference to a media file managed by an Event. |
| `<effect>` | A reference to an effect plug-in (for example, FxPlug, Motion document, or Audio Unit). |
| `<format>` | A reference to a Final Cut Pro X video format definition. |
| `<media>` | A reference to a new or existing media definition in an Event. The optional child element specifies the new media object. |
| `<projectRef>` | A reference to a new or existing Event. |

Each resource has an `id` attribute that specifies a *local identifier* for use within the FCPXML file. For example, an asset resource with a local identifier of “r1” might be declared as follows:

```
<asset id="r1" name="MyMovieFile" src="/path/to/MyMovieFile.mov"/>
```

Later in the same file, a video or audio element that uses the asset can refer to it by its local identifier as follows:

```
<video ref="r1" ... >
```

**Note:** Third-party tools may use any string for a local identifier, as long as the string begins with a letter and no two resources in the same FCPXML file share the same local identifier.

For most resources, Final Cut Pro X also assigns a lengthier *unique identifier* that it writes when exporting the FCPXML file. This unique identifier is stored as a `uid` attribute in the resource.

**Note:** Unique identifiers are private to Final Cut Pro X and should not be generated by third-party tools.

The reason for having both a local identifier and a unique identifier is to allow Final Cut Pro X to distinguish between resources that already exist and resources that are imported for the first time. Exported FCPXML files that refer to existing resources have unique identifiers assigned by Final Cut Pro X, whereas imported FCPXML files containing new resources have local identifiers only. A local identifier provides an identity for a new resource in an FCPXML file before Final Cut Pro X assigns it a permanent unique identifier.

## Bookmark Data

A security-scoped bookmark is required for sandboxed applications that need to access a media asset (file-system resource) outside the sandboxed environment. Refer to the *Security-Scoped Bookmarks and Persistent Resource Access* section in the App Sandbox Design Guide for more information.

Because of this requirement, Final Cut Pro X creates a document-scoped bookmark (see Table 3-2) for each media asset the user grants access to and includes this bookmark data in the asset element when exporting an FCPXML project or Event.

**Table 3-2** Bookmark Element

| Element | Description |
|---|---|
| `<bookmark>` | A Base64-encoded string that represents security-scoped bookmark data. It is used by a sandboxed application to reference an asset outside the sandboxed environment.<br>A bookmark is associated with an `<asset>` element. |

Follow these steps to use a security-scoped bookmark in a sandboxed application:

1. Decode the bookmark data—The bookmark data in the bookmark element is Base64-encoded so you must first decode the bookmark before it can be used. You can use the *Security Transforms API* to encode or decode Base64 data—refer to the Security Transforms Programming Guide for more details.

2. Obtain a security-scoped URL from the decoded bookmark data—Resolve the decoded bookmark data into a security-scoped URL that points to the media asset.
3. Indicate that you want to use the security-scoped URL to access the file-system resource—Before you can access the media asset using the security-scoped URL you must explicitly indicate that you want to use the security-scoped URL to gain access to the file-system resource.
4. Access the file-system resource—Use the security-scoped URL in your application to access the file-system resource.
5. Release the file-system resource—Once you are finished accessing the media asset you must explicitly release the file-system resource.

These steps are outlined in the following code snippet (refer to the NSURL Class Reference for details on the NSURL methods used).

```
// Decode the Base64 data using the Security Transforms API.
NSData* decodedBookmark = yourBase64DataDecodeRoutine(bookmark);

// Resolve the decoded bookmark data into a security-scoped URL.
NSError* err = nil;
NSURL* url = [NSURL URLByResolvingBookmarkData:decodedBookmark
                                 options:NSURLBookmarkResolutionWithSecurityScope
                                 relativeToURL:sourceURL
                           bookmarkDataIsStale:nil
                                         error:&err];

if (url) {
    // Indicate that you want to access the file-system resource.
    [url startAccessingSecurityScopedResource];

    // Use the resolved security scoped URL.
    ...

    // Release the file-system resource when you are done.
    [url stopAccessingSecurityScopedResource];
}
```

**Note:** To use security-scoped bookmarks, your application must have the `com.apple.security.files.bookmarks.document-scope` entitlement set to `true`. For more details see *Enabling Security-Scoped Bookmark and URL Access* in the Entitlement Key Reference.

## Multicam Media

A `<multicam>` media element allows you to assemble footage from multiple cameras, or angles, that are synchronized in time. A `<multicam>` element contains one or more `<mc-angle>` elements that each manage a series of other story elements.

These multicam media elements are listed in Table 3-3.

**Table 3-3** Multicam Media Elements

| Element | Description |
|---|---|
| `<multicam>` | A root container for *angles* of related media, organized as `<mc-angle>` elements. |
| `<mc-angle>` | A container of story elements organized sequentially in time. |

You define a `<multicam>` element inside a `<media>` resource for an Event.

A multicam definition with two angle definitions “a1” and “a2” might look similar to the following:

**Listing 3-1** Multicam definition with two angles

```
<resources>
    <asset id="r1" name="angle1" src="angle1.mov"/>
    <asset id="r2" name="angle2" src="angle2.mov"/>
    <format id="r3" name="FFVideoFormat720p2997"/>
    <media id="r4" name="MC Clip" uid="67890">
        <multicam format="r3">
            <mc-angle name="Angle1" angleID="a1">
                <clip duration="10s">
                    <video ref="r1" duration="10s">
                        <audio lane="-1" ref="r1" duration="10s"/>
                    </video>
                </clip>
            </mc-angle>
            <mc-angle name="Angle2" angleID="a2">
                <clip duration="10s">
                    <video ref="r2" duration="10s">
                        <audio lane="-1" ref="r2" duration="10s"/>
                    </video>
                </clip>
            </mc-angle>
        </multicam>
    </media>
</resources>
```

## Using Multicam Media

To use the multicam media in an Event or project, add an `<mc-clip>` element which references the media.

Each `<mc-clip>` element may use audio and video from the same angle, or mix the audio and video from separate angles from the same `<multicam>` container. You use `<mc-source>` elements to specify which angle the audio, video, or both come from.

**Table 3-4** Multicam Source Element

| Element | Description |
|---|---|
| `<mc-source>` | Specifies which angle a particular part of the clip comes from. This element has the following attributes:<br>• `angleID`—specifies the angle<br>• `srcEnable`—indicates which source to use, if any, from the angle (one of `audio`, `video`, `all`, or `none`). |

The following example references the multicam media defined in Listing 3-1:

```
<mc-clip ref="r4" name="MC Clip" duration="10s">
    <mc-source angleID="a1" srcEnable="all"/>
</mc-clip>
```

**Note:** In FCPXML v1.1, the angle was specified using the `videoAngleID` and `audioAngleID` attributes of the `<mc-clip>` element. In FCPXML v1.2, you use the `<mc-source>` element's `angleID` attribute.

## Audio Components and Multicam

When you use multicam clips in your project, you can also specify the configuration of audio components in each clip, such as channel mapping, effects, roles, mute ranges, and whether the configuration is enabled. To do so, you use the `<audio-source>` element just as you do with regular clip elements.

Listing 3-2 is an example project with two multicam clips whose audio components are reconfigured. The first clip has its audio reconfigured to dual mono, and the second clip has its audio reconfigured to reverse stereo with the last two seconds muted.

**Listing 3-2** Multicam project with two multicam clips and reconfigured audio

```
<fcpxml version="1.2">
    <project name="Multicam Project">
        <resources>
            <!-- reference to external Event #12345 -->
            <projectRef id="r1" uid="12345"/>
            <!-- reference to external media #67890 -->
            <media id="r2" uid="67890" projectRef="r1"/>
            <format id="r3" name="FFVideoFormat720p2997"/>
        </resources>
        <sequence format="r3">
            <spine>
                <mc-clip ref="r2" offset="0s" duration="5s" start="0s"
                    <mc-source angleID="a1" srcEnable="all">
                        <audio-source name="mono1" srcCh="1" outCh="L"/>
                        <audio-source name="momo2" srcCh="2" outCh="R"/>
                    </mc-source>
                </mc-clip>
                <mc-clip ref="r2" offset="5s" duration="5s" start="5s"
                    <mc-source angleID="a1" srcEnable="audio" />
                        <audio-source name="reverse stereo" srcCh="2, 1"
                                outCh="R, L"/>
```

```
                            <mute start="3s" duration="2s"/>
                        </audio-soruce>
                    </mc-source>
                    <mc-source angleID="a2" srcEnable="video"/>
                </mc-clip>
            </spine>
        </sequence>
    </project>
</fcpxml>
```

**Note:** As of FCPXML v1.2, `<mc-source>` elements no longer support audio filters. Instead, you apply audio filters to an `<mc-clip>` element, which applies the effect to all active audio source angles, or to an `<audio-source>` element, which only applies the effect to the individual audio source, if active. Existing FCPXML v1.1 projects with audio angles on an `<mc-clip>` element are converted to v1.2 format on import.

# Adjustments

## Overview

FCPXML supports adjustment elements to alter the audio and video output of story elements.

You can apply adjustment elements to control:

- Audio volume and panning
- Video frame size, position, scale, rotation, crop, pan and zoom, and blend mode

Each adjustment element has attribute values that affect the kind and amount of adjustment you wish to make. You specify the values you wish to override and all other values will remain at their default settings. You can also create animated adjustments by varying attribute values over time.

Some adjustment attributes expect a multi-dimensional (or compound) value, such as an (*x*,*y*) point. These attributes are specified with a space separating each value. For example, the two-dimensional values (0,0) or (10,-25) would be specified as `"0 0"` or `"10 -25"`.

The `<clip>`, `<mc-clip>`, `<audio>`, `<video>`, and `<title>` story elements may contain the adjustment elements listed in Table 4-1.

**Table 4-1** Adjustment Elements

| Element | Description |
|---|---|
| `<adjust-crop>` | Modifies the visible image width and height by cropping, trimming, or panning. This element has the following attributes:<br>• `enabled`—enables (1, the default) or disables (0) this crop adjustment.<br>• `mode`—specifies the active crop mode (one of `trim`, `crop`, or `pan`). Only one crop mode is active at any given time. Crop modes are defined via the following child elements, whose attribute values (`left`, `top`, `right`, and `bottom`) are expressed as a percentage of the original frame height:<br>  • `<crop-rect>`—defines the crop values.<br>  • `<trim-rect>`—defines the trim values.<br>  • `<pan-rect>`—defines the pan and zoom animation ("Ken Burns" effect) values. |
| `<adjust-corners>` | Modifies the corners of the visible image, adding a distort (or skew) effect using the `botLeft`, `topLeft`, `topRight`, and `botRight` attributes. These attributes are two-dimensional (*x y*) values expressed as a percentage of frame height. |
| `<adjust-conform>` | Modifies the initial image size of a clip to fill, fit, or remain unchanged within its container's frame size. The `type` attribute specifies the conform method to apply, which defaults to `"fit"`. |

| Element | Description |
|---|---|
| `<adjust-transform>` | Modifies the visible image by resizing, moving, or rotating using the following attributes:<br>• `enabled`—enables (1, the default) or disables (0) this transform adjustment.<br>• `position`—a two-dimensional (*x y*) value that specifies the amount to move the frame in each dimension from its original location. These values are percentages based on the original frame height. For example, if the original frame height is 1,080 pixels, specifying a `position="10 10"` moves the frame 108 pixels to the right and 108 pixels up.<br>• `scale`—a two-dimensional (*x y*) value that specifies the percentage amount to resize the frame in each dimension from its original frame size (for example, `"25 25"`).<br>• `rotation`—the amount, in degrees, to rotate the frame. A positive value rotates counter clock-wise and a negative value rotates clock-wise (for example, `"-180"`).<br>• `anchor`—a two-dimensional (*x y*) value that offsets the origin of the video image and its center of rotation. This value is expressed as a percentage of the scaled image height and defaults to `"0 0"`. |
| `<adjust-blend>` | Modifies the compositing blend mode and opacity percentage [0.0..1.0] of the visible image using the following attributes:<br>• `amount`—a percentage value that specifies the opacity amount (for example, `"0.75"`).<br>• `mode`—an integer value corresponding to a built-in FCP X blend mode. See "Blend Modes" (page 38) for a list of supported values. |
| `<adjust-volume>` | Modifies the volume of a clip in dB (for example, `"-6dB"`). |

| Element | Description |
|---|---|
| `<adjust-panner>` | Modifies the audio panning levels using the following attributes:<br>• `mode`—the pan mode. See "Pan Modes" (page 39) for a list of supported values.<br>• `amount`—the pan amount.<br>• `original_decoded_mix`—the balance between the original and decoded signals.<br>• `ambient_direct_mix`—the balance between the decoded surround and center signals.<br>• `surround_width`—the separation, in dB, between the decoded surround signals.<br>• `left_right_mix`—the balance between the left and right speakers.<br>• `front_back_mix`—the balance between the front and back speakers.<br>• `LFE_balance`—the low-frequency effects signal.<br>• `rotation`—the rotation amount for the surround signals.<br>• `stereo_spread`—the stereo effect amount from the left and right channels to the center and surround speakers<br>• `attenuate_collapse_mix`—the panning effect amount.<br>• `center_balance`—the balance between the center and surround signals. |

## Animation

You can create effects and animations by changing certain adjustment attributes over time. You do this by adding a parameter (`<param>`) element for each adjustment attribute you would like to override.

The `<param>` element attributes are defined in Table 4-2.

**Table 4-2** Parameter Element Attributes

| Attribute | Description |
|---|---|
| `name` | A case-sensitive string that identifies the parent attribute to override. |
| `value` | A new initial value to apply to the parent attribute (optional). This value overrides the initial value specified by the parent attribute. |

| Attribute | Description |
|---|---|
| `keyValues` | A list of keyframe values, delimited by a semicolon (;) (for example, `"5 10;10 20"`). |
| `keyTimes` | A list of keyframe times, delimited by a semicolon (;) (for example, `"0s;10s"`). |

If specified, adjustment values are set using the following priority (with 1 being the highest priority):

1. The `keyValues` and `keyTimes` attributes.
2. The parameter's `value` attribute.
3. The adjustment's attribute.

For example, the following XML adjusts the audio volume of a clip by -3dB:

```
<audio>
    <adjust-volume>
        <param name="amount" value= "-3dB"/>
    </adjust-volume>
</audio>
```

To produce a keyframe animation, you specify the values and time points for each keyframe using the `keyValues` and `keyTimes` attributes respectively; the FCP application will interpolate between them. The following example performs a position and scale/size transform adjustment animation using keyframes:

```
<adjust-transform position="-10 10">
    <param name="position" keyValues="-5 10;10 20" keyTimes="0s;10s"/>
    <param name="scale" keyValues="1 1;0.75 0.75" keyTimes="0s;10s"/>
</adjust-transform>
```

Initially, at time=0s, the frame is moved left 5 percent and up 10 percent (`"-5 10"`) from the original location, with no change in size (`"1 1"`). At time=10s, the frame has moved 10 percent right and 20 percent up (`"10 20"`) from the original location, with a 25 percent (`"0.75 0.75"`) reduction in size from the original (`"1 1"`).

In the previous example, it does not matter that an initial position of `"-10 10"` is specified because the keyframe defined at time=0s takes priority and overrides it to `"-5 10"`.

# Metadata

## Overview

Final Cut Pro X maintains various metadata item values (for example, Camera Name, Reel, Copyright). Metadata typically comes from the media and may be of interest to other applications, but is not critical for Final Cut Pro X to perform its tasks.

Some metadata items are editable through Final Cut Pro X; see Table 5-1 for a complete list.

On export, Final Cut Pro X includes only the metadata items in the selected metadata view. On import, Final Cut Pro X uses the values specified in the FCPXML for editable metadata items; the values for non-editable metadata items are ignored during import, and are instead retrieved from the media asset.

Table 5-1 lists the FCPXML metadata elements.

**Table 5-1** Metadata Elements

| Element | Description |
|---|---|
| `<metadata>` | A container for a collection of `<md>` elements. |
| `<md>` | Specifies an individual metadata item. This element has the following attributes:<br>• `key`—a structured Final Cut Pro X key string that identifies the metadata item (for example, `com.apple.proapps.mio.cameraName`).<br>• `value`—the value of the metadata item. When the value is an array, an `<array>` element is used instead.<br>• `editable`—indicates whether the metadata item is editable (`1`=Yes, `0`=No). The default is 0. Most metadata items are not editable, and this attribute may be omitted. If editable, Final Cut Pro X applies the new value for the metadata item on import.<br>• `type`—defines the metadata item value type.<br>• `displayName`—the metadata item name that appears in Final Cut Pro X.<br>• `description`—the metadata item description that appears in Final Cut Pro X.<br>• `source`—the metadata item origin. |

| Element | Description |
|---|---|
| `<array>` | A container for a collection of `<string>` elements that define the value(s) for an array of strings. |
| `<string>` | Defines one metadata value in an array of strings. |

The following is an example of metadata with primitive data types:

```
<metadata>
    <md key="com.apple.proapps.mio.cameraName" value="The Big Camera"/>
    <md key="com.apple.proapps.spotlight.kMDItemCopyright"
            value="© 2009 All Rights Reserved"/>
    <md key="com.apple.proapps.studio.reel" value="R945"/>
    <md key="com.apple.proapps.studio.scene" value="Deep Water"/>
</metadata>
```

The following is an example of metadata with complex data types:

```
<metadata>
    <md key="com.apple.proapps.spotlight.kMDItemCodecs" displayName="Codecs">
        <array>
            <string>AAC</string>
            <string>H.264</string>
        </array>
    </md>
</metadata>
```

## Keys and Sources

Metadata items Final Cut Pro X maintains are grouped based on where they come from (their source). The source is indicated by the `<md>` element's `source` attribute (for example, `studio`) or the prefix of the `key` attribute (for example, `com.apple.proapps.studio`).

The sources and key prefixes currently used are:

**Camera**

Metadata from a camera or capturing device that Final Cut Pro X or the device driver/plug-in picks up. These are not editable, except for `com.apple.proapps.mio.cameraName`.

*Key Prefixes:*

- `com.apple.proapps.mio`
- `org.smpte.mxf`

*Example:*

- `com.apple.proapps.mio.cameraName`

**Exif**

Metadata from files with Exif information (for example, a JPEG file).

*Key Prefix:*

- `com.apple.proapps.exif.{Exif}`

*Example:*

- `com.apple.proapps.exif.{Exif}.FocalLength`

**Image**

Metadata from an image file (for example, TIFF or PNG), or camera specific metadata saved with the image.

*Key Prefixes:*

- `com.apple.proapps.image.{TIFF}`
- `com.apple.proapps.image.{CIFF}`
- `com.apple.proapps.image.{GIF}`
- `com.apple.proapps.image.{PNG}`
- `com.apple.proapps.image.{DNG}`
- `com.apple.proapps.image.{GPS}`
- A prefix that identifies the camera manufacturer.

*Examples:*

- `com.apple.proapps.image.{TIFF}.WhitePoint`
- `com.apple.proapps.image.{CIFF}.FocusMode`
- `com.apple.proapps.image.{PNG}.Gamma`
- `com.apple.proapps.image.{GPS}.MapDatum`

**IPTC**

International Press Telecommunication Council defined photo metadata.

*Key Prefix:*

- `com.apple.proapps.iptc.{IPTC}`

*Example:*

- `com.apple.proapps.iptc.{IPTC}.Headline`

**Share** Metadata added to a share output.

*Key Prefix:*

- `com.apple.proapps.share`

*Example:*

- `com.apple.proapps.share.FFShareCopyrightKey`

**Spotlight** Spotlight metadata.

*Key Prefix:*

- `com.apple.proapps.spotlight`

*Example:*

- `com.apple.proapps.spotlight.kMDItemCopyright`

**Studio** Final Cut Pro X-defined metadata. See Table 5-1 (page 29) for a list of editable Studio keys. Additional *Studio* keys exist that are not editable.

*Key Prefix:*

- `com.apple.proapps.studio`

*Examples:*

- `com.apple.proapps.studio.reel`
- `com.apple.proapps.studio.scene`

**Custom** Metadata the user added in Final Cut Pro X (has prefix: `com.apple.proapps.custom`), or metadata that comes from a third-party application or camera (prefix is a Reverse DNS string that identifies a third-party application or a camera manufacturer).

*Key Prefixes:*

- `com.apple.proapps.custom`
- A prefix that identifies a third-party application or a camera manufacturer.

*Examples:*

- `com.apple.proapps.custom.myCustomMetadata`
- `com.yourCompany.yourApp.yourCustomMetadata`

# Exports with Color Corrections

Final Cut Pro X supports the American Society of Cinematographers Color Decision List (ASC CDL) format for the exchange of basic primary color grading information between equipment and software from different manufacturers.

Final Cut Pro X exports the `<info-asc-cdl>` element if a color correction is applied to a clip in your project. This element describes the first primary color correction in ASC CDL format using three attributes: `slope`, `offset`, and `power`. Each attribute is represented as a vector of red, green, and blue adjustment values (for example, "`1.0 1.0 1.0`").

> **Note:** On import, all `<info-acs-cdl>` elements are ignored and have no effect.

The following is an example export of ASC CDL information:

```
<clip>
    <info-asc-cdl slope="1.05 1.05 1.05" offset="0.0275 0.0275 0.0275"
                  power="1.25 1.2 1"/>
</clip>
```

# Appendix A: FCPXML Supported Identifiers

This appendix lists the supported identifiers you can use to specify rendering and video formats, and blend and pan modes in your FCPXML files.

## Rending Formats

```
FFRenderFormatProRes4444
FFRenderFormatProRes422HQ
FFRenderFormatProRes422
FFRenderFormatUncompressed10bit422
```

## Video Formats

```
FFVideoFormatRateUndefined
FFVideoFormat1080i5994
FFVideoFormat1080p2398
FFVideoFormat1080p24
FFVideoFormat1080p25
FFVideoFormat1080p2997
FFVideoFormat1080p30
FFVideoFormat1080p50
FFVideoFormat1080p5994
FFVideoFormat1080p60
FFVideoFormat1280x1080i5994
FFVideoFormat1280x1080p2398
FFVideoFormat1280x1080p24
FFVideoFormat1280x1080p2997
FFVideoFormat1280x1080p30
FFVideoFormat1280x1080p5994
```

```
FFVideoFormat1280x1080p60
FFVideoFormat1440x1080i50
FFVideoFormat1440x1080i5994
FFVideoFormat1440x1080p2398
FFVideoFormat1440x1080p24
FFVideoFormat1440x1080p25
FFVideoFormat1440x1080p2997
FFVideoFormat1440x1080p30
FFVideoFormat1440x1080p50
FFVideoFormat1440x1080p5994
FFVideoFormat1440x1080p60
FFVideoFormat2048x1024p2398
FFVideoFormat2048x1024p24
FFVideoFormat2048x1024p25
FFVideoFormat2048x1024p2997
FFVideoFormat2048x1024p30
FFVideoFormat2048x1024p50
FFVideoFormat2048x1024p5994
FFVideoFormat2048x1024p60
FFVideoFormat2048x1080p2398
FFVideoFormat2048x1080p24
FFVideoFormat2048x1080p25
FFVideoFormat2048x1080p2997
FFVideoFormat2048x1080p30
FFVideoFormat2048x1080p50
FFVideoFormat2048x1080p5994
FFVideoFormat2048x1080p60
FFVideoFormat2048x1152p2398
FFVideoFormat2048x1152p24
FFVideoFormat2048x1152p25
FFVideoFormat2048x1152p2997
FFVideoFormat2048x1152p30
FFVideoFormat2048x1152p50
FFVideoFormat2048x1152p5994
FFVideoFormat2048x1152p60
```

```
FFVideoFormat2048x1556p2398
FFVideoFormat2048x1556p24
FFVideoFormat2048x1556p25
FFVideoFormat2048x1556p2997
FFVideoFormat2048x1556p30
FFVideoFormat2048x1556p50
FFVideoFormat2048x1556p5994
FFVideoFormat2048x1556p60
FFVideoFormat4096x2048p2398
FFVideoFormat4096x2048p24
FFVideoFormat4096x2048p25
FFVideoFormat4096x2048p2997
FFVideoFormat4096x2048p30
FFVideoFormat4096x2048p50
FFVideoFormat4096x2048p5994
FFVideoFormat4096x2048p60
FFVideoFormat4096x2160p2398
FFVideoFormat4096x2160p24
FFVideoFormat4096x2160p25
FFVideoFormat4096x2160p2997
FFVideoFormat4096x2160p30
FFVideoFormat4096x2160p50
FFVideoFormat4096x2160p5994
FFVideoFormat4096x2160p60
FFVideoFormat4096x2304p2398
FFVideoFormat4096x2304p24
FFVideoFormat4096x2304p25
FFVideoFormat4096x2304p2997
FFVideoFormat4096x2304p30
FFVideoFormat4096x2304p50
FFVideoFormat4096x2304p5994
FFVideoFormat4096x2304p60
FFVideoFormat4096x3112p2398
FFVideoFormat4096x3112p24
```

```
FFVideoFormat4096x3112p25
FFVideoFormat4096x3112p2997
FFVideoFormat4096x3112p30
FFVideoFormat4096x3112p50
FFVideoFormat4096x3112p5994
FFVideoFormat4096x3112p60
FFVideoFormat640x480p2398
FFVideoFormat640x480p24
FFVideoFormat640x480p25
FFVideoFormat640x480p2997
FFVideoFormat640x480p30
FFVideoFormat720p2398
FFVideoFormat720p24
FFVideoFormat720p25
FFVideoFormat720p2997
FFVideoFormat720p30
FFVideoFormat720p50
FFVideoFormat720p5994
FFVideoFormat720p60
FFVideoFormat720x486i5994
FFVideoFormat720x486i5994_16x9
FFVideoFormat720x486p2398
FFVideoFormat720x486p2398_16x9
FFVideoFormat720x486p2997
FFVideoFormat720x486p2997_16x9
FFVideoFormat720x576i50
FFVideoFormat720x576i50_16x9
FFVideoFormat720x576p25
FFVideoFormat720x576p25_16x9
FFVideoFormat960x540p2398
FFVideoFormat960x540p24
FFVideoFormat960x540p25
FFVideoFormat960x540p2997
FFVideoFormat960x540p30
FFVideoFormat960x720p2398
```

```
FFVideoFormat960x720p24
FFVideoFormat960x720p25
FFVideoFormat960x720p2997
FFVideoFormat960x720p30
FFVideoFormat960x720p50
FFVideoFormat960x720p5994
FFVideoFormat960x720p60
FFVideoFormatDV720x480i5994
FFVideoFormatDV720x480i5994_16x9
FFVideoFormatDV720x480p2398
FFVideoFormatDV720x480p2398_16x9
FFVideoFormatDV720x480p2997
FFVideoFormatDV720x480p2997_16x9
FFVideoFormatDV720x576i50
FFVideoFormatDV720x576i50_16x9
```

## Blend Modes

FCPXML v1.1 and later support the following blend modes.

> **Note:** XML generators can omit the value in parentheses, and parsers should scan for the numeric value only.

```
0 (Normal)
2 (Subtract)
3 (Darken)
4 (Multiply)
5 (Color Burn)
6 (Linear Burn)
8 (Add)
9 (Lighten)
10 (Screen)
11 (Color Dodge)
```

```
12 (Linear Dodge)
14 (Overlay)
15 (Soft Light)
16 (Hard Light)
17 (Vivid Light)
18 (Linear Light)
19 (Pin Light)
20 (Hard Mix)
22 (Difference)
23 (Exclusion)
25 (Stencil Alpha)
26 (Stencil Luma)
27 (Silhouette Alpha)
28 (Silhouette Luma)
29 (Behind)
31 (Alpha Add)
32 (Premultiplied Mix)
```

## Pan Modes

FCPXML v1.1 and later support the following pan modes.

**Note:** XML generators can omit the value in parentheses, and parsers should scan for the numeric value only.

```
0 (Default)
1 (Stereo Left/Right)
2 (Create Space)
3 (Dialogue)
4 (Music)
5 (Ambience)
6 (Circle)
7 (Rotate)
8 (Back to Front)
```

```
9 (Left Surround to Right Front)
10 (Right Surround to Left Front)
```

## Metadata Keys

Final Cut Pro X imports/exports the following metadata keys.

| Key | Display Name | Type | Source |
|---|---|---|---|
| com.apple.proapps.mio.cameraName | Camera Name | String | Camera |
| com.apple.proapps.studio.alphaHandling | Alpha Handling | Integer | Studio |
| com.apple.proapps.studio.angle | Camera Angle | String | Studio |
| com.apple.proapps.studio.colorSpaceOverride | ColorSpace Override | Integer | Studio |
| com.apple.proapps.studio.metadataAnamorphicType | Anamorphic Override | Integer | Studio |
| com.apple.proapps.studio.metadataDeinterlaceType | Deinterlace | Boolean | Studio |
| com.apple.proapps.studio.metadataFieldDominanceOverride | Field Dominance Override | Integer | Studio |
| com.apple.proapps.studio.metadataLocation | Location | String | Studio |
| com.apple.proapps.studio.reel | Reel | String | Studio |
| com.apple.proapps.studio.scene | Scene | String | Studio |
| com.apple.proapps.studio.shot | Take | String | Studio |

The supported values for *Integer* types are provided in the following table.

| Integer Type | Value | Description |
|---|---|---|
| Alpha Handling | `0` | Premultiply |

| Integer Type | Value | Description |
|---|---|---|
| | 1 | Straight |
| | 2 | None/Ignore Alpha |
| ColorSpace Override | 0 | None Set |
| | 1 | 601 170M (NTSC) |
| | 2 | 601 EBU (PAL) |
| | 3 | 709 (HD) |
| | 4 | Adobe RGB (1998) |
| | 5 | sRGB (IEC 61966-2-1) |
| Anamorphic Override | 0 | None Set |
| | 1 | Standard |
| | 2 | Widescreen |
| Field Dominance Override | 0 | None Set |
| | 1 | Progressive |
| | 2 | Upper First |
| | 3 | Lower First |

# Appendix B: FCPXML Version 1.0 DTD

The following is the DTD for the Final Cut Pro X XML Interchange Format v1.0.

> **Note:** Even though an imported FCPXML document matches this DTD, import errors may still occur due to invalid data. If the document does not match the DTD, Final Cut Pro X rejects the import operation completely.

```
<!-- FCP X Interchange Format, Version 1.0 -->
<!-- Copyright (c) 2011 Apple Inc. All rights reserved. -->


<!-- FCPXML -->
<!ELEMENT fcpxml (project)>
<!ATTLIST fcpxml version CDATA #FIXED "1.0">


<!-- COMMON ENTITIES -->


<!-- 'time' attributes are expressed as a rational number of seconds (e.g.,
"1001/30000s") with a 64-bit numerator and 32-bit denominator. -->
<!-- Integer 'time' values, such as 5 seconds, may be expressed as whole numbers
(e.g., "5s"). -->
<!ENTITY % time "CDATA">


<!-- An exported Event contains zero or more 'clip' or 'audition' elements. -->
<!-- An exported project contains exactly one 'sequence' element. -->
<!ELEMENT project (resources, (sequence | (clip | audition)*))>
<!ATTLIST project name CDATA #IMPLIED>
<!ATTLIST project uid CDATA #IMPLIED>
<!ATTLIST project eventID CDATA #IMPLIED>       <!-- 'uid' of default Event -->
<!ATTLIST project location CDATA #IMPLIED>     <!-- URL of project storage location
 -->
```

```
<!-- RESOURCE ELEMENTS -->


<!-- A 'resource' is an asset, effect, output format, or external project shared
throughout a project. -->

<!ELEMENT resources (asset | effect | format | projectRef)*>


<!-- A 'projectRef' defines a reference to an external project via an FCP-assigned
 unique identifier. -->

<!-- 'uid' is an FCP-assigned unique ID; if missing, FCP will create a new Event.
 -->

<!ELEMENT projectRef EMPTY>

<!ATTLIST projectRef id ID #REQUIRED>

<!ATTLIST projectRef name CDATA #IMPLIED>

<!ATTLIST projectRef uid CDATA #IMPLIED>


<!-- A 'format' defines video output properties. -->

<!ELEMENT format EMPTY>

<!ATTLIST format id ID #REQUIRED>

<!ATTLIST format name CDATA #IMPLIED>

<!ATTLIST format frameDuration %time; #IMPLIED>

<!ATTLIST format fieldOrder CDATA #IMPLIED>       <!-- (progressive | upper first
 | lower first) -->

<!ATTLIST format width CDATA #IMPLIED>

<!ATTLIST format height CDATA #IMPLIED>

<!ATTLIST format paspH CDATA #IMPLIED>

<!ATTLIST format paspV CDATA #IMPLIED>

<!ATTLIST format colorOverride CDATA #IMPLIED>


<!-- An 'asset' defines a reference to the original source media (i.e., a local
file). -->

<!-- 'uid' is an FCP-assigned unique ID; if missing, FCP will create a new asset
-->

<!ELEMENT asset EMPTY>

<!ATTLIST asset id ID #REQUIRED>

<!ATTLIST asset name CDATA #IMPLIED>

<!ATTLIST asset uid CDATA #IMPLIED>

<!ATTLIST asset projectRef IDREF #IMPLIED>
```

```
<!ATTLIST asset src CDATA #REQUIRED>              <!-- file: URL -->
<!ATTLIST asset start %time; #IMPLIED>
<!ATTLIST asset duration %time; #IMPLIED>
<!ATTLIST asset hasVideo CDATA #IMPLIED>
<!ATTLIST asset hasAudio CDATA #IMPLIED>
<!ATTLIST asset audioSources CDATA #IMPLIED>
<!ATTLIST asset audioChannels CDATA #IMPLIED>
<!ATTLIST asset audioRate CDATA #IMPLIED>

<!-- An 'effect' defines a reference to a built-in or user-defined Motion effect,
 FxPlug plug-in, audio bundle, or audio unit. -->
<!ELEMENT effect EMPTY>
<!ATTLIST effect id ID #REQUIRED>
<!ATTLIST effect name CDATA #IMPLIED>
<!ATTLIST effect uid CDATA #REQUIRED>

<!-- STORY ELEMENTS -->

<!-- The 'ao_attrs' entity declares the attributes common to 'anchorable' objects.
 -->
<!-- The 'lane' attribute specifies where the object is contained/anchored relative
 to its parent: -->
<!--      0 = contained inside its parent (default) -->
<!--      >0 = anchored above its parent -->
<!--      <0 = anchored below its parent -->
<!-- The 'offset' attribute defines the location of the object in the parent
timeline (default is '0s'). -->
<!ENTITY % ao_attrs "
    lane CDATA #IMPLIED
    offset %time; #IMPLIED
">

<!-- The 'clip_attrs' entity declares the attributes common to all story elements.
 -->
<!--The 'start' attribute defines a local timeline to schedule contained and
anchored items. -->
```

```
<!ENTITY % clip_attrs "
    %ao_attrs;
    name CDATA #IMPLIED
    start %time; #IMPLIED
    duration %time; #REQUIRED
">

<!ENTITY % audioHz "(32k | 44.1k | 48k | 88.2k | 96k | 176.4k | 192k)">

<!ENTITY % outputChannel "(L | R | C | LFE | Ls | Rs | X)">

<!ENTITY % fadeType "(linear | easeIn | easeOut | easeInOut)">

<!-- A 'fadeIn' animates a parameter from its min value to its implied value over
 a specified duration. -->
<!ELEMENT fadeIn EMPTY>
<!ATTLIST fadeIn type %fadeType; #IMPLIED>        <!-- default is 'easeIn' -->
<!ATTLIST fadeIn duration %time; #REQUIRED>

<!-- A 'fadeOut' animates a parameter from its implied value to its min value over
 a specified duration. -->
<!ELEMENT fadeOut EMPTY>
<!ATTLIST fadeOut type %fadeType; #IMPLIED>       <!-- default is 'easeOut' -->
<!ATTLIST fadeOut duration %time; #REQUIRED>

<!-- A 'param' specifies the range for a parameter over time. -->
<!-- The 'name' and 'value' attributes support only constant values. -->
<!-- Fade-in and fade-out are optional. -->
<!ELEMENT param (fadeIn?, fadeOut?)>
<!ATTLIST param name CDATA #REQUIRED>
<!ATTLIST param value CDATA #IMPLIED>

<!-- The 'anchor_item' entity declares the valid anchorable story elements. -->
<!-- When present, anchored items must have a non-zero 'lane' value. -->
<!ENTITY % anchor_item "audio | video | clip | title | audition | spine">
```

```
<!-- The 'clip_item' entity declares the primary story elements that may appear
inside a clip. -->

<!ENTITY % clip_item "audio | video | clip | title | audition | gap">


<!-- A 'spine' (or storyline) contains elements ordered serially in time. -->

<!-- Only one story element is active at a given time, except when a transition
is present. -->

<!ELEMENT spine (audio | video | clip | title | audition | gap | transition)*>

<!ATTLIST spine %ao_attrs;>

<!ATTLIST spine name CDATA #IMPLIED>

<!ATTLIST spine format IDREF #IMPLIED>        <!-- default is same as parent -->


<!-- An 'audition' is a container with one active story element followed by
alternative story elements. -->

<!ELEMENT audition (audio | video | title | clip)+ >

<!ATTLIST audition %ao_attrs;>


<!-- A 'sequence' is the root/parent container for a 'spine' of story elements.
-->

<!-- Sequences have a local timeline which is used to schedule the story elements
 in its spine. -->

<!-- For example, in a sequence with a 'tcStart' of '3600s', a child element that
 starts 30 seconds from the start has an 'offset' of '3630s' -->

<!ELEMENT sequence (note?, spine)>

<!ATTLIST sequence format IDREF #REQUIRED>           <!-- output 'format' resource
 ID -->

<!ATTLIST sequence duration %time; #IMPLIED>

<!ATTLIST sequence tcStart %time; #IMPLIED>            <!-- sequence timecode
origin -->

<!ATTLIST sequence tcFormat (DF | NDF) #IMPLIED>      <!-- timecode display format
 (DF=drop frame, NDF=non-drop frame) -->

<!ATTLIST sequence audioLayout (stereo | surround) #IMPLIED>

<!ATTLIST sequence audioRate %audioHz; #IMPLIED>

<!ATTLIST sequence renderFormat CDATA #IMPLIED>


<!-- A 'clip' is a container for story elements. -->

<!-- Clips have only one primary item, and zero or more anchored items. -->
```

```
<!-- Use 'audioStart' and 'audioDuration' to define J/L cuts (i.e., split edits)
on composite A/V clips. -->

<!ELEMENT clip (note?, timeMap?, param*, (spine | (%clip_item;))*, (marker | rating
 | keyword)*, filter*)>

<!ATTLIST clip %clip_attrs;>

<!ATTLIST clip format IDREF #IMPLIED>            <!-- default is same as parent
-->

<!ATTLIST clip audioStart %time; #IMPLIED>

<!ATTLIST clip audioDuration %time; #IMPLIED>

<!ATTLIST clip tcStart %time; #IMPLIED>          <!-- clip timecode origin -->

<!ATTLIST clip tcFormat (DF | NDF) #IMPLIED>     <!-- timecode display format
(DF=drop frame, NDF=non-drop frame) -->

<!ATTLIST clip audioLayout (stereo | surround) #IMPLIED>

<!ATTLIST clip audioRate %audioHz; #IMPLIED>

<!ATTLIST clip enabled CDATA #IMPLIED>           <!-- default is '1' (0=disabled,
 1=enabled) -->


<!-- An 'audio' element defines a range of audio data in a source asset. -->

<!ELEMENT audio (note?, timeMap?, param*, (%anchor_item;)*, marker*, filter*)>

<!ATTLIST audio ref IDREF #REQUIRED>             <!-- 'asset' or 'effect' ID -->

<!ATTLIST audio %clip_attrs;>

<!ATTLIST audio srcID CDATA #IMPLIED>            <!-- source/track identifier in
'asset' (if not '1') -->

<!ATTLIST audio enabled CDATA #IMPLIED>          <!-- default is '1' (0=disabled,
 1=enabled) -->

<!ATTLIST audio role CDATA #IMPLIED>

<!ATTLIST audio srcCh CDATA #IMPLIED>            <!-- source audio channels in
'asset' (comma-separated, 1-based index) -->

<!ATTLIST audio outCh CDATA #IMPLIED>            <!-- output audio channels
(comma-separated, from: L, R, C, LFE, Ls, Rs, X) %outputChannel -->


<!-- A 'video' element defines a range of video data in a source asset. -->

<!ELEMENT video (note?, timeMap?, param*, (%anchor_item;)*, marker*, filter*)>

<!ATTLIST video ref IDREF #REQUIRED>             <!-- 'asset' or 'effect' ID -->

<!ATTLIST video %clip_attrs;>

<!ATTLIST video srcID CDATA #IMPLIED>            <!-- source/track identifier in
'asset' (if not '1') -->

<!ATTLIST video enabled CDATA #IMPLIED>          <!-- default is '1' (0=disabled,
 1=enabled) -->
```

```
<!ATTLIST video role CDATA #IMPLIED>             <!-- default is 'video' -->


<!-- A 'gap' element defines a placeholder with no associated media. -->

<!-- Gaps cannot be anchored to other objects. -->

<!ELEMENT gap (note?, (%anchor_item;)*, marker*)>

<!ATTLIST gap name CDATA #IMPLIED>

<!ATTLIST gap offset %time; #IMPLIED>

<!ATTLIST gap start %time; #IMPLIED>

<!ATTLIST gap duration %time; #REQUIRED>


<!-- A 'title' element contains one or more 'text' elements that customize a
referenced effect.  -->

<!ELEMENT title (note?, timeMap?, (%anchor_item;)*, marker*, filter*, text*)>

<!ATTLIST title ref IDREF #REQUIRED>              <!-- 'effect' ID for a Motion
template -->

<!ATTLIST title %clip_attrs;>

<!ATTLIST title enabled CDATA #IMPLIED>          <!-- default is '1' (0=disabled,
 1=enabled) -->

<!ATTLIST title role CDATA #IMPLIED>


<!-- A 'text' element defines an unformatted text string for a 'title' element.
-->

<!ELEMENT text (#PCDATA)>


<!-- A 'transition' element defines an effect that overlaps two adjacent story
elements. -->

<!-- For example,

             <video ref="1" duration="5s" />

             <transition ref="2" duration="2s" />

             <video ref="3" duration="5s" />

Here, the transition element overlaps the last 2 seconds of the previous video
(ref="1") and the first 2 seconds of the next video (ref="3"). -->

<!ELEMENT transition EMPTY>

<!ATTLIST transition ref IDREF #REQUIRED>         <!-- 'effect' ID -->

<!ATTLIST transition name CDATA #IMPLIED>

<!ATTLIST transition offset %time; #IMPLIED>
```

```
<!ATTLIST transition duration %time; #REQUIRED>


<!-- A 'filter' defines effect adjustments to apply to its parent element. -->

<!-- Filters are concatenated in the order in which they appear. -->

<!ELEMENT filter (param*)>

<!ATTLIST filter ref IDREF #REQUIRED>             <!-- 'effect' ID -->

<!ATTLIST filter name CDATA #IMPLIED>

<!ATTLIST filter enabled CDATA #IMPLIED>         <!-- default is '1' (0=disabled,
 1=enabled) -->


<!-- A 'timeMap' is a container for 'timept' elements that change the output speed
 of the clip's local timeline. -->

<!-- When present, a 'timeMap' defines a new adjusted time range for the clip using
 the first and last 'timept' elements. -->

<!-- All other time values are interpolated from the specified 'timept' elements.
 -->

<!ELEMENT timeMap (timept)+>


<!-- A 'timept' defines the re-mapped time values for a 'timeMap'. -->

<!-- For example,

             <timeMap>

                <timept time="0s" value="0s" interp="linear" />

                <timept time="10s" value="5s" interp="linear" />

                <timept time="20s" value="0s" interp="linear" />

             </timeMap>

Here, when applied to a clip whose original timeline was 0-5s, the 'timeMap' will
 adjust the clip's timeline to 0-20s and play the original content at 50% speed,
followed by -50% speed. -->

<!ELEMENT timept EMPTY>

<!ATTLIST timept time %time; #REQUIRED>               <!-- new adjusted clip time
 -->

<!ATTLIST timept value CDATA #REQUIRED>               <!-- original clip time -->

<!ATTLIST timept interp (smooth | linear) #REQUIRED>  <!-- interpolation type for
 next segment -->


<!-- KEYWORDS, MARKERS, NOTES -->


<!-- If 'completed' is specified, this marker becomes a to-do item. -->
```

```
<!ELEMENT marker EMPTY>
<!ATTLIST marker start %time; #REQUIRED>
<!ATTLIST marker duration %time; #IMPLIED>
<!ATTLIST marker value CDATA #REQUIRED>
<!ATTLIST marker completed CDATA #IMPLIED>             <!-- (0=not completed |
1=completed) -->

<!ELEMENT rating EMPTY>
<!ATTLIST rating start %time; #IMPLIED>
<!ATTLIST rating duration %time; #IMPLIED>
<!ATTLIST rating value (favorite | reject) #REQUIRED>

<!ELEMENT keyword EMPTY>
<!ATTLIST keyword start %time; #IMPLIED>
<!ATTLIST keyword duration %time; #IMPLIED>
<!ATTLIST keyword value CDATA #REQUIRED>               <!-- comma-separated list
of keywords -->

<!ELEMENT note (#PCDATA)>
```

# Appendix C: FCPXML Version 1.1 DTD

The following is the DTD for the Final Cut Pro X XML Interchange Format v1.1.

> **Note:** Even though an imported FCPXML document matches this DTD, import errors may still occur due to invalid data. If the document does not match the DTD, Final Cut Pro X rejects the import operation completely.

```
<!-- FCP X Interchange Format, Version 1.1 -->
<!-- Copyright (c) 2011 Apple Inc. All rights reserved. -->

<!-- FCPXML -->
<!ELEMENT fcpxml (project)>
<!ATTLIST fcpxml version CDATA #FIXED "1.1">


<!-- COMMON ENTITIES -->

<!-- 'time' attributes are expressed as a rational number of seconds (e.g.,
"1001/30000s") -->
<!-- with a 64-bit numerator and 32-bit denominator. -->
<!-- Integer 'time' values, such as 5 seconds, may be expressed as whole numbers
(e.g., '5s'). -->
<!ENTITY % time "CDATA">

<!-- A 'timelist' is a semicolon-separated list of time values (e.g.,
"0s;5s;10s;20s") -->
<!ENTITY % timelist "CDATA">


<!-- PROJECT ELEMENTS -->
```

```
<!-- An exported Event contains zero or more 'clip', 'audition', or 'multicam'
elements. -->

<!-- An exported project contains exactly one 'sequence' element. -->

<!ELEMENT project (resources, (sequence | (clip | audition | mc-clip)*))>

<!ATTLIST project name CDATA #IMPLIED>

<!ATTLIST project uid CDATA #IMPLIED>

<!ATTLIST project eventID CDATA #IMPLIED>        <!-- 'uid' of default Event -->

<!ATTLIST project location CDATA #IMPLIED>       <!-- URL of project storage
location -->




<!-- RESOURCE ELEMENTS -->


<!-- A 'resource' is a project element potentially referenced by other project
elements. -->

<!-- To support such references, all resource instances require a local ID attribute.
 -->

<!ELEMENT resources (asset | effect | format | media | projectRef)*>


<!-- A 'projectRef' defines a reference to an external project via an FCP-assigned
 unique identifier ('uid'). -->

<!-- If 'uid' is not specified, FCP creates a new project instance for this
reference. -->

<!ELEMENT projectRef EMPTY>

<!ATTLIST projectRef id ID #REQUIRED>

<!ATTLIST projectRef name CDATA #IMPLIED>

<!ATTLIST projectRef uid CDATA #IMPLIED>


<!-- A 'media' defines a reference to new or existing media via an FCP-assigned
unique identifier ('uid'). -->

<!-- If 'uid' is not specified, FCP creates a new media object as specified by the
 optional child element. -->

<!-- If 'projectRef' is not specified, FCP uses the default instance. -->

<!ELEMENT media (multicam)?>

<!ATTLIST media id ID #REQUIRED>

<!ATTLIST media name CDATA #IMPLIED>

<!ATTLIST media uid CDATA #IMPLIED>
```

```
<!ATTLIST media projectRef IDREF #IMPLIED>

<!-- A 'format' describes video properties. -->
<!ELEMENT format EMPTY>
<!ATTLIST format id ID #REQUIRED>
<!ATTLIST format name CDATA #IMPLIED>
<!ATTLIST format frameDuration %time; #IMPLIED>
<!ATTLIST format fieldOrder CDATA #IMPLIED>       <!-- (progressive | upper first
 | lower first ) -->
<!ATTLIST format width CDATA #IMPLIED>
<!ATTLIST format height CDATA #IMPLIED>
<!ATTLIST format paspH CDATA #IMPLIED>
<!ATTLIST format paspV CDATA #IMPLIED>

<!-- An 'asset' defines a reference to external source media (i.e., a local file).
 -->
<!-- 'uid' is an FCP-assigned unique ID; if not specified, FCP creates a new default
 clip for the asset. -->
<!ELEMENT asset EMPTY>
<!ATTLIST asset id ID #REQUIRED>
<!ATTLIST asset name CDATA #IMPLIED>
<!ATTLIST asset uid CDATA #IMPLIED>
<!ATTLIST asset projectRef IDREF #IMPLIED>
<!ATTLIST asset src CDATA #REQUIRED>        <!-- file: URL -->
<!ATTLIST asset start %time; #IMPLIED>
<!ATTLIST asset duration %time; #IMPLIED>
<!ATTLIST asset hasVideo CDATA #IMPLIED>
<!ATTLIST asset hasAudio CDATA #IMPLIED>
<!ATTLIST asset audioSources CDATA #IMPLIED>
<!ATTLIST asset audioChannels CDATA #IMPLIED>
<!ATTLIST asset audioRate CDATA #IMPLIED>
<!ATTLIST asset colorOverride CDATA #IMPLIED>

<!-- An 'effect' defines a reference to a built-in or user-defined Motion effect,
 FxPlug plug-in, audio bundle, or audio unit. -->
<!ELEMENT effect EMPTY>
```

```
<!ATTLIST effect id ID #REQUIRED>
<!ATTLIST effect name CDATA #IMPLIED>
<!ATTLIST effect uid CDATA #REQUIRED>



<!-- STORY ELEMENTS -->


<!-- The 'ao_attrs' entity declares the attributes common to 'anchorable' objects.
 -->
<!-- The 'lane' attribute specifies where the object is contained/anchored relative
 to its parent: -->
<!--    0 = contained inside its parent (default) -->
<!--    >0 = anchored above its parent -->
<!--    <0 = anchored below its parent -->
<!-- The 'offset' attribute defines the location of the object in the parent's
timeline (default is '0s'). -->
<!ENTITY % ao_attrs "
    lane CDATA #IMPLIED
    offset %time; #IMPLIED
">


<!-- The 'clip_attrs' entity declares the attributes common to all story elements.
 -->
<!-- The 'start' attribute defines a local timeline to schedule contained and
anchored items. -->
<!-- The default start value is '0s'. -->
<!ENTITY % clip_attrs "
    %ao_attrs;
    name CDATA #IMPLIED
    start %time; #IMPLIED
    duration %time; #REQUIRED
    enabled (1 | 0) '1'
">


<!ENTITY % audioHz "( 32k | 44.1k | 48k | 88.2k | 96k | 176.4k | 192k )">
```

```
<!-- The 'media_attrs' entity declares the attributes common to media instances.
-->

<!-- 'format' specifies a <format> resource ID. -->

<!-- 'tcStart' specifies the timecode origin of the media. -->

<!-- 'tcFormat' specifies the timecode display format (DF=drop frame; NDF=non-drop
 frame). -->

<!ENTITY % media_attrs "

    format IDREF #REQUIRED

    duration %time; #IMPLIED

    tcStart %time; #IMPLIED

    tcFormat (DF | NDF) #IMPLIED

    audioLayout (stereo | surround) #IMPLIED

    audioRate %audioHz; #IMPLIED

">


<!ENTITY % outputChannel "( L | R | C | LFE | Ls | Rs | X )">


<!ENTITY % fadeType "(linear | easeIn | easeOut | easeInOut)">


<!-- A 'fadeIn' element animates a parameter from its min value to its implied
value over a specified duration. -->

<!ELEMENT fadeIn EMPTY>

<!ATTLIST fadeIn type %fadeType; #IMPLIED>        <!-- default is 'easeIn' -->

<!ATTLIST fadeIn duration %time; #REQUIRED>


<!-- A 'fadeOut' element animates a parameter from its implied value to its min
value over a specified duration. -->

<!ELEMENT fadeOut EMPTY>

<!ATTLIST fadeOut type %fadeType; #IMPLIED>       <!-- default is 'easeOut' -->

<!ATTLIST fadeOut duration %time; #REQUIRED>


<!-- A 'param' specifies the range for a parameter over time, optionally including
 key-framed values. -->

<!-- 'fade-in’ and ‘fade-out’ are optional. -->

<!ELEMENT param (fadeIn?, fadeOut?, param*)>

<!ATTLIST param name CDATA #REQUIRED>

<!ATTLIST param value CDATA #IMPLIED>            <!-- initial value -->
```

```
<!ATTLIST param keyValues CDATA #IMPLIED>        <!-- list of keyframed values
(delimited by ;) -->

<!ATTLIST param keyTimes %timelist; #IMPLIED>    <!-- list of time offsets for
keyframed values (delimited by ;) -->


<!-- A 'crop-rect' specifies crop values as a percentage of original frame height.
 -->

<!ELEMENT crop-rect (param*)>

<!ATTLIST crop-rect left CDATA "0">

<!ATTLIST crop-rect top CDATA "0">

<!ATTLIST crop-rect right CDATA "0">

<!ATTLIST crop-rect bottom CDATA "0">


<!-- A 'trim-rect' specifies trim values as a percentage of original frame height.
 -->

<!ELEMENT trim-rect (param*)>

<!ATTLIST trim-rect left CDATA "0">

<!ATTLIST trim-rect top CDATA "0">

<!ATTLIST trim-rect right CDATA "0">

<!ATTLIST trim-rect bottom CDATA "0">


<!-- A 'pan-rect' specifies the initial or final crop values for a "Ken Burns"
animation. -->

<!-- The attributes of a pan-rect cannot be keyframed. -->

<!ELEMENT pan-rect EMPTY>

<!ATTLIST pan-rect left CDATA "0">

<!ATTLIST pan-rect top CDATA "0">

<!ATTLIST pan-rect right CDATA "0">

<!ATTLIST pan-rect bottom CDATA "0">


<!-- An 'info-asc-cdl' describes a primary color correction in the form of an ASC
 CDL (American Society of Cinematographers Color Decision List). -->

<!-- Each attribute is a vector of "red green blue" adjustments. -->

<!-- This element is not processed during import. -->

<!ELEMENT info-asc-cdl EMPTY>

<!ATTLIST info-asc-cdl slope CDATA "1.0 1.0 1.0">
```

```
<!ATTLIST info-asc-cdl offset CDATA "0.0 0.0 0.0">

<!ATTLIST info-asc-cdl power CDATA "1.0 1.0 1.0">


<!-- The 'adjust-crop' element modifies the visible image width and height. -->

<!-- This element contains an optional adjustment for each crop mode, although
only one mode is active. -->

<!ELEMENT adjust-crop (crop-rect?, trim-rect?, (pan-rect, pan-rect)?)>

<!ATTLIST adjust-crop mode (trim | crop | pan) #REQUIRED>

<!ATTLIST adjust-crop enabled (0 | 1) "1">


<!-- The 'adjust-corners' element modifies the corner positions of the visible
image adding a distort (or skew) effect. -->

<!ELEMENT adjust-corners (param*)>

<!ATTLIST adjust-corners enabled (0 | 1) "1">

<!ATTLIST adjust-corners botLeft CDATA "0 0">

<!ATTLIST adjust-corners topLeft CDATA "0 0">

<!ATTLIST adjust-corners topRight CDATA "0 0">

<!ATTLIST adjust-corners botRight CDATA "0 0">


<!-- The 'adjust-conform' element modifies the frame size of a clip to match the
project’s frame size. -->

<!-- 'fit' resizes the clip to fit entirely in the project’s frame, adding black
bars as needed. -->

<!-- 'fill' resizes the clip to fill the entire project’s frame, cropping the clip
 as needed. -->

<!-- 'none' does not resize the clip, black bars may be added (for smaller clips)
 or cropping may occur (for larger clips) as needed. -->

<!ELEMENT adjust-conform EMPTY>

<!ATTLIST adjust-conform type (fit | fill | none) "fit">


<!-- The 'adjust-transform' element modifies the visible image by resizing, moving,
 or rotating it. -->

<!ELEMENT adjust-transform (param*)>

<!ATTLIST adjust-transform enabled (0 | 1) "1">

<!ATTLIST adjust-transform position CDATA "0 0">    <!-- new 'x y' position as a
percentage of frame height -->

<!ATTLIST adjust-transform scale CDATA "1 1">       <!-- new 'x y' size as a
percentage of frame height -->
```

```
<!ATTLIST adjust-transform rotation CDATA "0">      <!-- amount to rotate in
degrees; + rotates CCW, - rotates CW -->
<!ATTLIST adjust-transform anchor CDATA "0 0">      <!-- the 'x y' point at which
 the image rotates around -->


<!-- The 'adjust-blend' element modifies the opacity of the visible image. -->
<!ELEMENT adjust-blend (param*)>
<!ATTLIST adjust-blend amount CDATA "1.0">     <!-- a percentage in the range 0.0
 - 1.0 -->
<!ATTLIST adjust-blend mode CDATA #IMPLIED>    <!-- the built-in FCPX blend mode
to apply -->


<!-- The 'adjust-volume' element modifies the audio volume of the clip. -->
<!ELEMENT adjust-volume (param*)>
<!ATTLIST adjust-volume amount CDATA "0dB">


<!-- The 'adjust-panner' element modifies the audio pan mode of the clip. -->
<!ELEMENT adjust-panner (param*)>
<!ATTLIST adjust-panner mode CDATA #IMPLIED>
<!ATTLIST adjust-panner amount CDATA "0">
<!ATTLIST adjust-panner original_decoded_mix CDATA #IMPLIED>
<!ATTLIST adjust-panner ambient_direct_mix CDATA #IMPLIED>
<!ATTLIST adjust-panner surround_width CDATA #IMPLIED>
<!ATTLIST adjust-panner left_right_mix CDATA #IMPLIED>
<!ATTLIST adjust-panner front_back_mix CDATA #IMPLIED>
<!ATTLIST adjust-panner LFE_balance CDATA #IMPLIED>
<!ATTLIST adjust-panner rotation CDATA #IMPLIED>
<!ATTLIST adjust-panner stereo_spread CDATA #IMPLIED>
<!ATTLIST adjust-panner attenuate_collapse_mix CDATA #IMPLIED>
<!ATTLIST adjust-panner center_balance CDATA #IMPLIED>


<!-- The 'intrinsic-params' entities declare intrinsic video and audio adjustments.
 -->
<!ENTITY % intrinsic-params-video "(info-asc-cdl?, adjust-crop?, adjust-corners?,
 adjust-conform?, adjust-transform?, adjust-blend?)">
<!ENTITY % intrinsic-params-audio "(adjust-volume?, adjust-panner?)">
```

```
<!ENTITY % intrinsic-params "(%intrinsic-params-video;, %intrinsic-params-audio;)">


<!-- The 'anchor_item' entity declares the valid anchorable story elements. -->

<!-- When present, anchored items must have a non-zero 'lane' value. -->

<!ENTITY % anchor_item "audio | video | clip | title | mc-clip | audition | spine">


<!-- The 'clip_item' entity declares the primary story elements that may appear
inside a clip. -->

<!ENTITY % clip_item "audio | video | clip | title | mc-clip | audition | gap">


<!-- An 'audition' is a container with one active story element followed by
alternative story elements. -->

<!ELEMENT audition (audio | video | title | mc-clip | clip)+ >

<!ATTLIST audition %ao_attrs;>


<!-- A 'spine' is a container for elements ordered serially in time. -->

<!-- Only one story element is active at a given time, except when a transition
is present. -->

<!ELEMENT spine (%clip_item; | transition)* >

<!ATTLIST spine %ao_attrs;>

<!ATTLIST spine name CDATA #IMPLIED>

<!ATTLIST spine format IDREF #IMPLIED>    <!-- default is same as parent -->


<!-- A 'sequence' is a container for a spine of story elements in a sequence
project. -->

<!ELEMENT sequence (note?, spine)>

<!ATTLIST sequence %media_attrs;>

<!ATTLIST sequence renderFormat CDATA #IMPLIED>


<!-- A 'multicam' is a container for multiple "angles" of related content. -->

<!ELEMENT multicam (mc-angle*)>

<!ATTLIST multicam %media_attrs;>

<!ATTLIST multicam renderFormat CDATA #IMPLIED>


<!-- An 'mc-angle' is a container for elements ordered serially in time for one
angle of a multicam clip. -->
```

```
<!-- Only one story element is active at a given time, except when a transition
is present. -->

<!ELEMENT mc-angle ((%clip_item; | transition)*) >

<!ATTLIST mc-angle name CDATA #IMPLIED>

<!ATTLIST mc-angle angleID CDATA #REQUIRED>


<!-- An 'mc-clip' element defines an edited range of a/v data from a source
'multicam' media. -->

<!ELEMENT mc-clip (note?, timeMap?, %intrinsic-params;, mc-source*, (%anchor_item;)*,
 (marker | rating | keyword)*)>

<!ATTLIST mc-clip ref IDREF #REQUIRED>            <!-- 'media' ID -->

<!ATTLIST mc-clip %clip_attrs;>

<!ATTLIST mc-clip audioAngleID CDATA #IMPLIED>    <!-- angleID of the active audio
 angle -->

<!ATTLIST mc-clip videoAngleID CDATA #IMPLIED>    <!-- angleID of the active video
 angle -->

<!ATTLIST mc-clip audioStart %time; #IMPLIED>

<!ATTLIST mc-clip audioDuration %time; #IMPLIED>


<!-- An 'mc-source' element defines custom filters to apply to an angle of a
multicam clip or edit. -->

<!ELEMENT mc-source (filter*)>

<!ATTLIST mc-source angleID CDATA #REQUIRED>


<!-- A 'clip' is a container for other story elements. -->

<!-- Clips have only one primary item, and zero or more anchored items. -->

<!-- Use 'audioStart' and 'audioDuration' to define J/L cuts (i.e., split edits)
on composite A/V clips. -->

<!ELEMENT clip (note?, timeMap?, %intrinsic-params;, (spine | (%clip_item;))*,
(marker | rating | keyword)*, filter*)>

<!ATTLIST clip %clip_attrs;>

<!ATTLIST clip format IDREF #IMPLIED>            <!-- default is same as parent
-->

<!ATTLIST clip audioStart %time; #IMPLIED>

<!ATTLIST clip audioDuration %time; #IMPLIED>

<!ATTLIST clip tcStart %time; #IMPLIED>          <!-- clip timecode origin -->

<!ATTLIST clip tcFormat (DF | NDF) #IMPLIED>     <!-- timecode display format
(DF=drop frame; NDF=non-drop frame) -->
```

```
<!ATTLIST clip audioLayout (stereo | surround) #IMPLIED>
<!ATTLIST clip audioRate %audioHz; #IMPLIED>


<!-- An 'audio' element defines a range of audio data in a source asset. -->
<!ELEMENT audio (note?, timeMap?, %intrinsic-params-audio;, (%anchor_item;)*,
marker*, filter*)>
<!ATTLIST audio ref IDREF #REQUIRED>     <!-- 'asset' or 'effect' ID -->
<!ATTLIST audio %clip_attrs;>
<!ATTLIST audio srcID CDATA #IMPLIED>    <!-- source/track identifier in asset (if
 not '1') -->
<!ATTLIST audio role CDATA #IMPLIED>
<!ATTLIST audio srcCh CDATA #IMPLIED>    <!-- source audio channels in asset (comma
 separated, 1-based index) -->
<!ATTLIST audio outCh CDATA #IMPLIED>    <!-- output audio channels (comma separated,
 from: L,R,C,LFE,Ls,Rs,X) %outputChannel -->


<!-- A 'video' element defines a range of video data in a source asset. -->
<!ELEMENT video (note?, timeMap?, %intrinsic-params-video;, (%anchor_item;)*,
marker*, filter*)>
<!ATTLIST video ref IDREF #REQUIRED>     <!-- 'asset' or 'effect' ID -->
<!ATTLIST video %clip_attrs;>
<!ATTLIST video srcID CDATA #IMPLIED>    <!-- source/track identifier in asset (if
 not '1') -->
<!ATTLIST video role CDATA #IMPLIED>     <!-- default is 'video' -->


<!-- A 'gap' element defines a placeholder with no associated media. -->
<!-- Gaps cannot be anchored to other objects. -->
<!ELEMENT gap (note?, (%anchor_item;)*, marker*)>
<!ATTLIST gap name CDATA #IMPLIED>
<!ATTLIST gap offset %time; #IMPLIED>
<!ATTLIST gap start %time; #IMPLIED>
<!ATTLIST gap duration %time; #REQUIRED>
<!ATTLIST gap enabled (0 | 1) "1">


<!-- A 'title' element contains one or more 'text' elements that customize a
referenced effect. -->
<!ELEMENT title (note?, timeMap?, %intrinsic-params-video;, (%anchor_item;)*,
marker*, filter*, text*)>
```

```
<!ATTLIST title ref IDREF #REQUIRED>     <!-- 'effect' ID for a Motion template
-->

<!ATTLIST title %clip_attrs;>

<!ATTLIST title role CDATA #IMPLIED>


<!-- A 'text' element defines an unformatted text string for a 'title' element.
-->

<!ELEMENT text (#PCDATA)>


<!-- A 'transition' element defines an effect that overlaps two adjacent story
elements. -->

<!-- For example,

    <video ref="1" duration="5s"/>

    <transition ref="2" duration="2s"/>

    <video ref="3" duration="5s"/>

Here, the transition element overlaps the last 2 seconds of the previous video
(ref="1") and the first 2 seconds of the next video (ref="3"). -->

<!ELEMENT transition (marker*)>

<!ATTLIST transition ref IDREF #REQUIRED>        <!-- 'effect' ID -->

<!ATTLIST transition name CDATA #IMPLIED>

<!ATTLIST transition offset %time; #IMPLIED>

<!ATTLIST transition duration %time; #REQUIRED>


<!-- A 'filter' defines an effect that's applied to its parent element. -->

<!-- Filters are concatenated in the order in which they appear. -->

<!ELEMENT filter EMPTY>

<!ATTLIST filter ref IDREF #REQUIRED>     <!-- 'effect' ID -->

<!ATTLIST filter name CDATA #IMPLIED>

<!ATTLIST filter enabled (0 | 1) "1">


<!-- A 'timeMap' is a container for 'timept' elements that change the output speed
 of the clip's local timeline. -->

<!-- When present, a 'timeMap' defines a new adjusted time range for the clip using
 the first and last 'timept' elements. -->

<!-- All other time values are interpolated from the specified 'timept' elements.
 -->

<!ELEMENT timeMap (timept)+>
```

```
<!-- A 'timept' defines the re-mapped time values for a 'timeMap'. -->
<!-- For example,
    <timeMap>
        <timept time="0s" value="0s" interp="linear"/>
        <timept time="10s" value="5s" interp="linear"/>
        <timept time="20s" value="0s" interp="linear"/>
    </timeMap>
Here, when applied to a clip whose original timeline was 0-5s, the 'timeMap' will
 adjust the clip's timeline to 0-20s and play the original content at 50% speed,
followed by -50% speed. -->
<!ELEMENT timept EMPTY>
<!ATTLIST timept time %time; #REQUIRED>        <!-- new adjusted clip time -->
<!ATTLIST timept value CDATA #REQUIRED>        <!-- original clip time -->
<!ATTLIST timept interp (smooth | linear) #REQUIRED>     <!-- interpolation type
for next segment -->


<!-- KEYWORDS, MARKERS, NOTES -->

<!-- If 'completed' is specified, this marker becomes a to-do item. -->
<!ELEMENT marker EMPTY>
<!ATTLIST marker start %time; #REQUIRED>
<!ATTLIST marker duration %time; #IMPLIED>
<!ATTLIST marker value CDATA #REQUIRED>
<!ATTLIST marker completed CDATA #IMPLIED>     <!-- (0=not completed, 1=completed)
 -->
<!ATTLIST marker note CDATA #IMPLIED>

<!ELEMENT rating EMPTY>
<!ATTLIST rating start %time; #IMPLIED>
<!ATTLIST rating duration %time; #IMPLIED>
<!ATTLIST rating value (favorite | reject) #REQUIRED>
<!ATTLIST rating note CDATA #IMPLIED>

<!ELEMENT keyword EMPTY>
```

```
<!ATTLIST keyword start %time; #IMPLIED>

<!ATTLIST keyword duration %time; #IMPLIED>

<!ATTLIST keyword value CDATA #REQUIRED>      <!-- comma-separated list of keywords
 -->

<!ATTLIST keyword note CDATA #IMPLIED>


<!ELEMENT note (#PCDATA)>
```

# Appendix D: FCPXML Version 1.2 DTD

The following is the DTD for the Final Cut Pro X XML Interchange Format v1.2.

> **Note:** Even though an imported FCPXML document matches this DTD, import errors may still occur due to invalid data. If the document does not match the DTD, Final Cut Pro X rejects the import operation completely.

```
<!-- FCP X Interchange Format, Version 1.2 -->
<!-- Copyright (c) 2011-2012 Apple Inc. All rights reserved. -->


<!-- FCPXML -->
<!ELEMENT fcpxml (project)>
<!ATTLIST fcpxml version CDATA #FIXED "1.2">



<!-- COMMON ENTITIES -->


<!-- 'time' attributes are expressed as a rational number of seconds (e.g.,
"1001/30000s") -->
<!-- with a 64-bit numerator and 32-bit denominator. -->
<!-- Integer 'time' values, such as 5 seconds, may be expressed as whole numbers
(e.g., '5s'). -->
<!ENTITY % time "CDATA">


<!-- A 'timelist' is a semicolon-separated list of time values -->
<!ENTITY % timelist "CDATA">



<!-- PROJECT ELEMENTS -->
```

```
<!-- An exported Event contains zero or more 'clip', 'audition', or 'multicam'
elements. -->

<!-- An exported project contains exactly one 'sequence' element. -->

<!ELEMENT project (resources, (sequence | (clip | audition | mc-clip | ref-clip
)*))>

<!ATTLIST project name CDATA #IMPLIED>

<!ATTLIST project uid CDATA #IMPLIED>

<!ATTLIST project eventID CDATA #IMPLIED>            <!-- 'uid' of default Event
-->

<!ATTLIST project location CDATA #IMPLIED>           <!-- URL of project storage
location -->



<!-- RESOURCE ELEMENTS -->


<!-- A 'resource' is a project element potentially referenced by other project
elements. -->

<!-- To support such references, all resource instances require a local ID attribute.
 -->

<!ELEMENT resources (asset | effect | format | media | projectRef)*>


<!-- A 'projectRef' defines a reference to an external project via an FCP-assigned
 unique identifier ('uid'). -->

<!-- If 'uid' is not specified, FCP creates a new project instance for this
reference. -->

<!ELEMENT projectRef EMPTY>

<!ATTLIST projectRef id ID #REQUIRED>

<!ATTLIST projectRef name CDATA #IMPLIED>

<!ATTLIST projectRef uid CDATA #IMPLIED>


<!-- A 'media' defines a reference to new or existing media via an FCP-assigned
unique identifier ('uid'). -->

<!-- If 'uid' is not specified, FCP creates a new media object as specified by the
 optional child element. -->

<!-- If 'projectRef' is not specified, FCP uses the default instance. -->

<!ELEMENT media (multicam | sequence)?>

<!ATTLIST media id ID #REQUIRED>

<!ATTLIST media name CDATA #IMPLIED>
```

```
<!ATTLIST media uid CDATA #IMPLIED>
<!ATTLIST media projectRef IDREF #IMPLIED>

<!-- A 'format' describes video properties. -->
<!ELEMENT format EMPTY>
<!ATTLIST format id ID #REQUIRED>
<!ATTLIST format name CDATA #IMPLIED>
<!ATTLIST format frameDuration %time; #IMPLIED>
<!ATTLIST format fieldOrder CDATA #IMPLIED>            <!-- (progressive | upper
first | lower first ) -->
<!ATTLIST format width CDATA #IMPLIED>
<!ATTLIST format height CDATA #IMPLIED>
<!ATTLIST format paspH CDATA #IMPLIED>
<!ATTLIST format paspV CDATA #IMPLIED>

<!-- An 'asset' defines a reference to external source media (i.e., a local file).
 -->
<!-- 'uid' is an FCP-assigned unique ID; if not specified, FCP creates a new default
 clip for the asset. -->
<!ELEMENT asset (bookmark?,metadata?)>
<!ATTLIST asset id ID #REQUIRED>
<!ATTLIST asset name CDATA #IMPLIED>
<!ATTLIST asset uid CDATA #IMPLIED>
<!ATTLIST asset projectRef IDREF #IMPLIED>
<!ATTLIST asset src CDATA #REQUIRED>                <!-- file: URL -->
<!ATTLIST asset start %time; #IMPLIED>
<!ATTLIST asset duration %time; #IMPLIED>
<!ATTLIST asset hasVideo CDATA #IMPLIED>
<!ATTLIST asset hasAudio CDATA #IMPLIED>
<!ATTLIST asset audioSources CDATA #IMPLIED>
<!ATTLIST asset audioChannels CDATA #IMPLIED>
<!ATTLIST asset audioRate CDATA #IMPLIED>
<!ATTLIST asset colorOverride CDATA #IMPLIED>

<!-- The 'md-type' entity declares the metadata data type. -->
<!ENTITY % md-type "( string | boolean | integer | float | date | timecode )">
```

```
<!-- The 'metadata' element is a container for a collection of 'md' metadata
definition elements. -->

<!ELEMENT metadata (md*)>


<!-- The 'md' element defines an individual metadata item. -->

<!-- The 'key' attribute is a structured reverse DNS string that identifies the
metadata item (e.g., com.apple.proapps.studio.cameraName). -->

<!-- Metadata values are specified using either the 'value' attribute or the 'array'
 child element:

       - The 'value' attribute is used to specify the metadata value for primitive
 data types (e.g., boolean, integer, string).

      - The 'array' child element is a container for 'string' elements that specify
 the metadata value(s) for complex data types. -->

<!ELEMENT md (array?)>

<!ATTLIST md key CDATA #REQUIRED>

<!ATTLIST md value CDATA #IMPLIED>

<!ATTLIST md editable (0 | 1) "0">

<!ATTLIST md type %md-type; #IMPLIED>

<!ATTLIST md displayName CDATA #IMPLIED>

<!ATTLIST md description CDATA #IMPLIED>

<!ATTLIST md source CDATA #IMPLIED>


<!-- An 'effect' defines a reference to a built-in or user-defined Motion effect,
 FxPlug plug-in, audio bundle, or audio unit. -->

<!ELEMENT effect EMPTY>

<!ATTLIST effect id ID #REQUIRED>

<!ATTLIST effect name CDATA #IMPLIED>

<!ATTLIST effect uid CDATA #REQUIRED>




<!-- STORY ELEMENTS -->


<!-- The 'ao_attrs' entity declares the attributes common to 'anchorable' objects.
 -->

<!-- The 'lane' attribute sepcifies where the object is contained/anchored relative
 to its parent: -->
```

```
<!--    0 = contained inside its parent (default) -->
<!--    >0 = anchored above its parent -->
<!--    <0 = anchored below its parent -->
<!-- The 'offset' attribute defines the location of the object in the parent
timeline (default is '0s'). -->
<!ENTITY % ao_attrs "
    lane CDATA #IMPLIED
    offset %time; #IMPLIED
">

<!-- The 'clip_attrs' entity declares the attributes common to all story elements.
 -->
<!-- The 'start' attribute defines a local timeline to schedule contained and
anchored items. -->
<!-- The default start value is '0s'. -->
<!ENTITY % clip_attrs "
    %ao_attrs;
    name CDATA #IMPLIED
    start %time; #IMPLIED
    duration %time; #REQUIRED
    enabled (0 | 1) '1'
">

<!ENTITY % audioHz "( 32k | 44.1k | 48k | 88.2k | 96k | 176.4k | 192k )">

<!-- The 'media_attrs' entity declares the attributes common to media instances.
-->
<!-- 'format' specifies a <format> resource ID. -->
<!-- 'tcStart' specifies the timecode origin of the media. -->
<!-- 'tcFormat' specifies the timecode display format (DF=drop frame; NDF=non-drop
 frame). -->
<!ENTITY % media_attrs "
    format IDREF #REQUIRED
    duration %time; #IMPLIED
    tcStart %time; #IMPLIED
    tcFormat (DF | NDF) #IMPLIED
    audioLayout (stereo | surround) #IMPLIED
    audioRate %audioHz; #IMPLIED
">

<!ENTITY % outputChannel "( L | R | C | LFE | Ls | Rs | X )">

<!ENTITY % fadeType "(linear | easeIn | easeOut | easeInOut)">

<!-- A 'fadeIn' element animates a parameter from its min value to its implied
value over a specified duration. -->
<!ELEMENT fadeIn EMPTY>
<!ATTLIST fadeIn type %fadeType; #IMPLIED>        <!-- default is 'easeIn' -->
<!ATTLIST fadeIn duration %time; #REQUIRED>

<!-- A 'fadeOut' element animates a parameter from its implied value to its min
value over a specified duration. -->
<!ELEMENT fadeOut EMPTY>
<!ATTLIST fadeOut type %fadeType; #IMPLIED>       <!-- default is 'easeOut' -->
<!ATTLIST fadeOut duration %time; #REQUIRED>

<!-- A 'mute' element suppresses audio output for a range of source media time -->
<!ELEMENT mute (fadeIn?, fadeOut?)>
<!ATTLIST mute start %time; #IMPLIED>
<!ATTLIST mute duration %time; #IMPLIED>

<!-- A 'param' specifies the range for a parameter over time, optionally including
 key-framed values. -->
<!-- Fade-in and fade-out are optional. -->
<!ELEMENT param (fadeIn?, fadeOut?, param*)>
<!ATTLIST param name CDATA #REQUIRED>
<!ATTLIST param value CDATA #IMPLIED>                <!-- initial value -->
<!ATTLIST param keyValues CDATA #IMPLIED>           <!-- list of keyframed values
 (delimited by ;) -->
<!ATTLIST param keyTimes %timelist; #IMPLIED>       <!-- list of time offsets for
 keyframed values (delimited by ;) -->
```

```
<!-- A 'crop-rect' specifies crop values as a percentage of original frame height.
 -->
<!ELEMENT crop-rect (param*)>
<!ATTLIST crop-rect left CDATA "0">
<!ATTLIST crop-rect top CDATA "0">
<!ATTLIST crop-rect right CDATA "0">
<!ATTLIST crop-rect bottom CDATA "0">

<!-- A 'trim-rect' specifies trim values as a percentage of original frame height.
 -->
<!ELEMENT trim-rect (param*)>
<!ATTLIST trim-rect left CDATA "0">
<!ATTLIST trim-rect top CDATA "0">
<!ATTLIST trim-rect right CDATA "0">
<!ATTLIST trim-rect bottom CDATA "0">

<!-- A 'pan-rect' specifies the initial or final crop values for a "Ken Burns"
animation. -->
<!-- The attributes of a pan-rect cannot be keyframed. -->
<!ELEMENT pan-rect EMPTY>
<!ATTLIST pan-rect left CDATA "0">
<!ATTLIST pan-rect top CDATA "0">
<!ATTLIST pan-rect right CDATA "0">
<!ATTLIST pan-rect bottom CDATA "0">

<!-- An 'info-asc-cdl' describes a primary color correction in the form of an ASC
 CDL (American Society of Cinematographers Color Decision List). -->
<!-- Each attribute is a vector of "red green blue" adjustments. -->
<!-- This element is not processed during import. -->
<!ELEMENT info-asc-cdl EMPTY>
<!ATTLIST info-asc-cdl slope CDATA "1.0 1.0 1.0">
<!ATTLIST info-asc-cdl offset CDATA "0.0 0.0 0.0">
<!ATTLIST info-asc-cdl power CDATA "1.0 1.0 1.0">

<!-- The 'adjust-crop' element modifies the visible image width and height. -->
<!-- This element contains an optional adjustment for each crop mode, although
only one mode is active. -->
```

```
<!ELEMENT adjust-crop (crop-rect?, trim-rect?, (pan-rect, pan-rect)?)>
<!ATTLIST adjust-crop mode (trim | crop | pan) #REQUIRED>
<!ATTLIST adjust-crop enabled (0 | 1) "1">


<!-- The 'adjust-corners' element modifies the corner positions of the visible
image adding a distort (or skew) effect. -->
<!ELEMENT adjust-corners (param*)>
<!ATTLIST adjust-corners enabled (0 | 1) "1">
<!ATTLIST adjust-corners botLeft CDATA "0 0">
<!ATTLIST adjust-corners topLeft CDATA "0 0">
<!ATTLIST adjust-corners topRight CDATA "0 0">
<!ATTLIST adjust-corners botRight CDATA "0 0">


<!-- The 'adjust-conform' element modifies the frame size of a clip to match the
project’s frame size. -->
<!-- 'fit' resizes the clip to fit entirely in the project’s frame, adding black
bars as needed. -->
<!-- 'fill' resizes the clip to fill the entire project’s frame, cropping the clip
 as needed. -->
<!-- 'none' does not resize the clip, black bars may be added (for smaller clips)
 or cropping may occur (for larger clips) as needed. -->
<!ELEMENT adjust-conform EMPTY>
<!ATTLIST adjust-conform type (fit | fill | none) "fit">


<!-- The 'adjust-transform' element modifies the visible image by resizing, moving,
 or rotating it. -->
<!ELEMENT adjust-transform (param*)>
<!ATTLIST adjust-transform enabled (0 | 1) "1">
<!ATTLIST adjust-transform position CDATA "0 0">
<!ATTLIST adjust-transform scale CDATA "1 1">
<!ATTLIST adjust-transform rotation CDATA "0">
<!ATTLIST adjust-transform anchor CDATA "0 0">


<!-- The 'adjust-blend' element modifies the opacity of the visible image. -->
<!ELEMENT adjust-blend (param*)>
<!ATTLIST adjust-blend amount CDATA "1.0">
```

```
<!ATTLIST adjust-blend mode CDATA #IMPLIED>

<!-- The 'adjust-volume' element modifies the audio volume of the clip. -->
<!ELEMENT adjust-volume (param*)>
<!ATTLIST adjust-volume amount CDATA "0dB">

<!-- The 'adjust-panner' element modifies the audio pan mode of the clip. -->
<!ELEMENT adjust-panner (param*)>
<!ATTLIST adjust-panner mode CDATA #IMPLIED>
<!ATTLIST adjust-panner amount CDATA "0">
<!ATTLIST adjust-panner original_decoded_mix CDATA #IMPLIED>
<!ATTLIST adjust-panner ambient_direct_mix CDATA #IMPLIED>
<!ATTLIST adjust-panner surround_width CDATA #IMPLIED>
<!ATTLIST adjust-panner left_right_mix CDATA #IMPLIED>
<!ATTLIST adjust-panner front_back_mix CDATA #IMPLIED>
<!ATTLIST adjust-panner LFE_balance CDATA #IMPLIED>
<!ATTLIST adjust-panner rotation CDATA #IMPLIED>
<!ATTLIST adjust-panner stereo_spread CDATA #IMPLIED>
<!ATTLIST adjust-panner attenuate_collapse_mix CDATA #IMPLIED>
<!ATTLIST adjust-panner center_balance CDATA #IMPLIED>

<!-- The 'intrinsic-params' entities declare instrinsic video and audio adjustments.
 -->
<!ENTITY % intrinsic-params-video "(info-asc-cdl?, adjust-crop?, adjust-corners?,
 adjust-conform?, adjust-transform?, adjust-blend?)">
<!ENTITY % intrinsic-params-audio "(adjust-volume?, adjust-panner?)">
<!ENTITY % intrinsic-params "(%intrinsic-params-video;, %intrinsic-params-audio;)">

<!-- The 'audio-filter' and 'video-filter' entities help to document which contexts
 expect one type of filter versus the other. -->
<!ENTITY % audio-filter "filter">
<!ENTITY % video-filter "filter">

<!-- The 'anchor_item' entity declares the valid anchorable story elements. -->
<!-- When present, anchored items must have a non-zero 'lane' value. -->
```

```
<!ENTITY % anchor_item "audio | video | clip | title | mc-clip | ref-clip | audition
 | spine">


<!-- The 'clip_item' entity declares the primary story elements that may appear
inside a clip. -->

<!ENTITY % clip_item "audio | video | clip | title | mc-clip | ref-clip | audition
 | gap">


<!ENTITY % marker_item "(marker | chapter-marker)">


<!-- The 'audio_comp_items' entity declares the component-level audio adjustments
 for a clip. -->

<!ENTITY % audio_comp_items "(audio-source*, audio-aux-source*)">


<!-- An 'audio-source' element adjusts playback settings for a single audio component
 in a clip's primary audio layout -->

<!-- The primary audio layout is comprised of all audio from elements in the primary
 (lane 0) storyline. -->

<!ELEMENT audio-source (%intrinsic-params-audio;, (%audio-filter;)*, mute*)>

<!ATTLIST audio-source name CDATA #IMPLIED>

<!ATTLIST audio-source srcCh CDATA #REQUIRED>        <!-- source audio channels
(comma separated, 1-based index) -->

<!ATTLIST audio-source outCh CDATA #IMPLIED>        <!-- output audio channels
(comma separated, from: L,R,C,LFE,Ls,Rs,X) %outputChannel -->

<!ATTLIST audio-source role CDATA #IMPLIED>           <!-- output role assignment
 -->

<!ATTLIST audio-source enabled (0 | 1) '1'>


<!-- An 'audio-aux-source' element adjusts playback settings for a single audio
component in a clip's auxiliary audio layout. -->

<!-- The auxiliary audio layout is comprised of all audio from elements outside
(i.e. anchored to) the primary storyline. -->

<!ELEMENT audio-aux-source (%intrinsic-params-audio;, (%audio-filter;)*, mute*)>

<!ATTLIST audio-aux-source name CDATA #IMPLIED>

<!ATTLIST audio-aux-source srcCh CDATA #REQUIRED>       <!-- source audio channels
 (comma separated, 1-based index) -->

<!ATTLIST audio-aux-source outCh CDATA #IMPLIED>       <!-- output audio channels
 (comma separated, from: L,R,C,LFE,Ls,Rs,X) %outputChannel -->
```

```
<!ATTLIST audio-aux-source role CDATA #IMPLIED>            <!-- output role
assignment -->
<!ATTLIST audio-aux-source enabled (0 | 1) '1'>

<!-- An 'audition' is a container with one active story element followed by
alternative story elements. -->
<!ELEMENT audition (audio | video | title | mc-clip | ref-clip | clip)+ >
<!ATTLIST audition %ao_attrs;>

<!-- A 'spine' is a container for elements ordered serially in time. -->
<!-- Only one story element is active at a given time, except when a transition
is present. -->
<!ELEMENT spine (%clip_item; | transition)* >
<!ATTLIST spine    %ao_attrs;>
<!ATTLIST spine name CDATA #IMPLIED>
<!ATTLIST spine format IDREF #IMPLIED>                <!-- default is same as
parent -->

<!-- A 'sequence' is a container for a spine of story elements in a sequence
project. -->
<!ELEMENT sequence (note?, spine, %audio_comp_items;, metadata?)>
<!ATTLIST sequence %media_attrs;>
<!ATTLIST sequence renderFormat CDATA #IMPLIED>

<!-- A 'multicam' is a container for multiple "angles" of related content. -->
<!ELEMENT multicam (mc-angle*, metadata?)>
<!ATTLIST multicam %media_attrs;>
<!ATTLIST multicam renderFormat CDATA #IMPLIED>

<!-- An 'mc-angle' is a container for elements ordered serially in time for one
angle of a multicam clip. -->
<!-- Only one story element is active at a given time, except when a transition
is present. -->
<!ELEMENT mc-angle ((%clip_item; | transition)*, %audio_comp_items;) >
<!ATTLIST mc-angle name CDATA #IMPLIED>
<!ATTLIST mc-angle angleID CDATA #REQUIRED>
```

```
<!-- An 'mc-clip' element defines an edited range of a/v data from a source
'multicam' media. -->

<!ELEMENT mc-clip (note?, timeMap?, %intrinsic-params;, (%audio-filter;)*,
mc-source*, (%anchor_item;)*, ((%marker_item;) | rating | keyword)*, metadata?)>

<!ATTLIST mc-clip ref IDREF #REQUIRED>                <!-- 'media' ID -->

<!ATTLIST mc-clip %clip_attrs;>

<!ATTLIST mc-clip audioStart %time; #IMPLIED>

<!ATTLIST mc-clip audioDuration %time; #IMPLIED>


<!-- An 'mc-source' element defines custom settings and filters to apply to an
angle of a multicam clip or edit. -->

<!ELEMENT mc-source (%audio_comp_items;, (%video-filter;)*)>

<!ATTLIST mc-source angleID CDATA #REQUIRED>

<!ATTLIST mc-source srcEnable (all | audio | video | none) "all">


<!-- A 'clip' is a container for other story elements. -->

<!-- Clips have only one primary item, and zero or more anchored items. -->

<!-- Use 'audioStart' and 'audioDuration' to define J/L cuts (i.e., split edits)
on composite A/V clips. -->

<!ELEMENT clip (note?, timeMap?, %intrinsic-params;, (spine | (%clip_item;))*,
((%marker_item;) | rating | keyword)*, %audio_comp_items;, filter*, metadata?)>

<!ATTLIST clip %clip_attrs;>

<!ATTLIST clip format IDREF #IMPLIED>            <!-- default is same as parent
-->

<!ATTLIST clip audioStart %time; #IMPLIED>

<!ATTLIST clip audioDuration %time; #IMPLIED>

<!ATTLIST clip tcStart %time; #IMPLIED>          <!-- clip timecode origin -->

<!ATTLIST clip tcFormat (DF | NDF) #IMPLIED>     <!-- timecode display format
(DF=drop frame; NDF=non-drop frame) -->

<!ATTLIST clip audioLayout (stereo | surround) #IMPLIED>

<!ATTLIST clip audioRate %audioHz; #IMPLIED>


<!-- A 'ref-clip' is a clip that references (rather than contains) other story
elements. -->

<!-- Clips have a media reference and zero or more anchored items. -->

<!-- Use 'audioStart' and 'audioDuration' to define J/L cuts (i.e., split edits)
on composite A/V clips. -->
```

```
<!ELEMENT ref-clip (note?, timeMap?, %intrinsic-params;, (%anchor_item;)*,
((%marker_item;) | rating | keyword)*, %audio_comp_items;, filter*, metadata?)>

<!ATTLIST ref-clip ref IDREF #REQUIRED>                <!-- 'media' ID -->

<!ATTLIST ref-clip %clip_attrs;>

<!ATTLIST ref-clip srcEnable (all | audio | video) "all">

<!ATTLIST ref-clip audioStart %time; #IMPLIED>

<!ATTLIST ref-clip audioDuration %time; #IMPLIED>


<!-- An 'audio' element defines a range of audio data in a source asset. -->

<!ELEMENT audio (note?, timeMap?, %intrinsic-params-audio;, (%anchor_item;)*,
(%marker_item;)*, (%audio-filter;)*)>

<!ATTLIST audio ref IDREF #REQUIRED>                <!-- 'asset' or 'effect' ID
-->

<!ATTLIST audio %clip_attrs;>

<!ATTLIST audio srcID CDATA #IMPLIED>                <!-- source/track identifier
 in asset (if not '1') -->

<!ATTLIST audio role CDATA #IMPLIED>

<!ATTLIST audio srcCh CDATA #IMPLIED>                <!-- source audio channels
in asset (comma separated, 1-based index) -->

<!ATTLIST audio outCh CDATA #IMPLIED>                <!-- output audio channels
(comma separated, from: L,R,C,LFE,Ls,Rs,X) %outputChannel -->


<!-- A 'video' element defines a range of video data in a source asset. -->

<!ELEMENT video (note?, timeMap?, %intrinsic-params-video;, (%anchor_item;)*,
(%marker_item;)*, (%video-filter;)*)>

<!ATTLIST video ref IDREF #REQUIRED>                <!-- 'asset' or 'effect' ID
-->

<!ATTLIST video %clip_attrs;>

<!ATTLIST video srcID CDATA #IMPLIED>                <!-- source/track identifier
 in asset (if not '1') -->

<!ATTLIST video role CDATA #IMPLIED>                <!-- default is 'video' -->


<!-- A 'gap' element defines a placeholder with no associated media. -->

<!-- Gaps cannot be anchored to other objects. -->

<!ELEMENT gap (note?, (%anchor_item;)*, (%marker_item;)*, metadata?)>

<!ATTLIST gap name CDATA #IMPLIED>

<!ATTLIST gap offset %time; #IMPLIED>

<!ATTLIST gap start %time; #IMPLIED>
```

```
<!ATTLIST gap duration %time; #REQUIRED>
<!ATTLIST gap enabled (0 | 1) "1">


<!-- A 'title' element contains one or more 'text' elements that customize a
referenced effect. -->
<!ELEMENT title (note?, timeMap?, %intrinsic-params-video;, (%anchor_item;)*,
(%marker_item;)*, (%video-filter;)*, metadata?, text*)>
<!ATTLIST title ref IDREF #REQUIRED>               <!-- 'effect' ID for a Motion
 template -->
<!ATTLIST title %clip_attrs;>
<!ATTLIST title role CDATA #IMPLIED>


<!-- A 'text' element defines an unformatted text string for a 'title' element.
-->
<!ELEMENT text (#PCDATA)>


<!-- A 'transition' element defines an effect that overlaps two adjacent story
elements. -->
<!-- For example,
    <video ref="1" duration="5s"/>
    <transition ref="2" duration="2s"/>
    <video ref="3" duration="5s"/>
Here, the transition element overlaps the last 2 seconds of the previous video
(ref="1") and the first 2 seconds of the next video (ref="3"). -->
<!ELEMENT transition ((%marker_item;)*, metadata?)>
<!ATTLIST transition ref IDREF #REQUIRED>            <!-- 'effect' ID -->
<!ATTLIST transition name CDATA #IMPLIED>
<!ATTLIST transition offset %time; #IMPLIED>
<!ATTLIST transition duration %time; #REQUIRED>


<!-- A 'filter' defines an effect that's applied to its parent element. -->
<!-- Filters are concatenated in the order in which they appear. -->
<!ELEMENT filter EMPTY>
<!ATTLIST filter ref IDREF #REQUIRED>                <!-- 'effect' ID -->
<!ATTLIST filter name CDATA #IMPLIED>
<!ATTLIST filter enabled (0 | 1) "1">
```

```
<!-- A 'timeMap' is a container for 'timept' elements that change the output speed
 of the clip's local timeline. -->

<!-- When present, a 'timeMap' defines a new adjusted time range for the clip using
 the first and last 'timept' elements. -->

<!-- All other time values are interpolated from the specified 'timept' elements.
 -->

<!ELEMENT timeMap (timept)*>

<!ATTLIST timeMap rateConform (0 | 1) #IMPLIED>


<!-- A 'timept' defines the re-mapped time values for a 'timeMap'. -->

<!-- For example,

    <timeMap>

        <timept time="0s" value="0s" interp="linear"/>

        <timept time="10s" value="5s" interp="linear"/>

        <timept time="20s" value="0s" interp="linear"/>

    </timeMap>

Here, when applied to a clip whose original timeline was 0-5s, the 'timeMap' will
 adjust the clip's timeline to 0-20s and play the original content at 50% speed,
followed by -50% speed. -->

<!ELEMENT timept EMPTY>

<!ATTLIST timept time %time; #REQUIRED>               <!-- new adjusted clip time
 -->

<!ATTLIST timept value CDATA #REQUIRED>               <!-- original clip time -->

<!ATTLIST timept interp (smooth | linear) #REQUIRED> <!-- interpolation type for
next segment -->




<!-- KEYWORDS, MARKERS, NOTES -->


<!-- If 'completed' is specified, this marker becomes a to-do item. -->

<!ELEMENT marker EMPTY>

<!ATTLIST marker start %time; #REQUIRED>

<!ATTLIST marker duration %time; #IMPLIED>

<!ATTLIST marker value CDATA #REQUIRED>

<!ATTLIST marker completed CDATA #IMPLIED>            <!-- (0=not completed,
1=completed) -->

<!ATTLIST marker note CDATA #IMPLIED>
```

```
<!ELEMENT rating EMPTY>
<!ATTLIST rating start %time; #IMPLIED>
<!ATTLIST rating duration %time; #IMPLIED>
<!ATTLIST rating value (favorite | reject) #REQUIRED>
<!ATTLIST rating note CDATA #IMPLIED>

<!ELEMENT keyword EMPTY>
<!ATTLIST keyword start %time; #IMPLIED>
<!ATTLIST keyword duration %time; #IMPLIED>
<!ATTLIST keyword value CDATA #REQUIRED>            <!-- comma-separated list of
keywords -->
<!ATTLIST keyword note CDATA #IMPLIED>

<!ELEMENT chapter-marker EMPTY>
<!ATTLIST chapter-marker start %time; #REQUIRED>
<!ATTLIST chapter-marker duration %time; #IMPLIED>
<!ATTLIST chapter-marker value CDATA #REQUIRED>
<!ATTLIST chapter-marker note CDATA #IMPLIED>
<!ATTLIST chapter-marker posterOffset %time; #IMPLIED>

<!ELEMENT note (#PCDATA)>

<!ELEMENT bookmark (#PCDATA)>

<!ELEMENT array (string*)>
<!ELEMENT string (#PCDATA)>
```

# Document Revision History

This table describes the changes to *Final Cut Pro X XML Format* .

| Date | Notes |
|---|---|
| 2012-10-23 | Updated for Final Cut Pro X XML v1.2. |
| 2012-01-30 | Updated for Final Cut Pro X XML v1.1. |
| 2011-09-19 | New document that describes the XML format used by Final Cut Pro X. |

Apple Inc.
1 Infinite Loop
Cupertino, CA 95014
408-996-1010

**Even though Apple has reviewed this document, APPLE MAKES NO WARRANTY OR REPRESENTATION, EITHER EXPRESS OR IMPLIED, WITH RESPECT TO THIS DOCUMENT, ITS QUALITY, ACCURACY, MERCHANTABILITY, OR FITNESS FOR A PARTICULAR PURPOSE. AS A RESULT, THIS DOCUMENT IS PROVIDED "AS IS," AND YOU, THE READER, ARE ASSUMING THE ENTIRE RISK AS TO ITS QUALITY AND ACCURACY.**

**IN NO EVENT WILL APPLE BE LIABLE FOR DIRECT, INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES RESULTING FROM ANY DEFECT OR INACCURACY IN THIS DOCUMENT, even if advised of the possibility of such damages.**

**THE WARRANTY AND REMEDIES SET FORTH ABOVE ARE EXCLUSIVE AND IN LIEU OF ALL OTHERS, ORAL OR WRITTEN, EXPRESS OR IMPLIED. No Apple dealer, agent, or employee is authorized to make any modification, extension, or addition to this warranty.**

**Some states do not allow the exclusion or limitation of implied warranties or liability for incidental or consequential damages, so the above limitation or exclusion may not apply to you. This warranty gives you specific legal rights, and you may also have other rights which vary from state to state.**