

Final Cut Pro 7 XML Interchange Format



Developer

Contents

About This Document 7

Organization 7

XML Resources 8

Document Export and Import 9

Exporting Documents 9

Importing Documents 10

Errors During Import 12

Basics of Encoding 13

Special Conventions 14

Inheritance 14

The id Attribute 14

Encoding a Clip 15

Associating Media With a Clip 18

Encoding Audio Information 21

Exporting a Clip from the Browser 23

Encoding a Subclip 24

Encoding a Multiclip 26

Encoding a Sequence 29

Format and Outputs 29

Media in a Sequence 32

Linking 35

Transitions 38

PSD Files as Sequences 40

Encoding a Project 41

Encoding Effects 43

Keyframes 45

XMEML Topics 48

Notes on the XML Interchange Format 48

Timing Values 49

Speed and Special Keyframes 51

Special and Virtual Keyframes 51

The anchoroffset Element	51
Keyframe Adjustments	52
Importing Speed Keyframes	52
Exporting Speed Keyframes	52
Identifying Media	52
Managing Project Components	53
The uuid and updatebehavior Elements	53
The itemhistory Element	55
Managing QuickTime Metadata	56
Applications of the Interchange Format	59
Enhancing a Batch List	59
Simplifying Subtitling	64
Choosing Effect Parameters	70
Other Possibilities	77
Elements Catalog	79
Major Elements	79
Clips	82
Video and Audio	90
Common Elements	94
Rate and Timecode	96
Effects	98
The Time Remap Effect	103
Application Specific Data	104
Film Data	107
QuickTime Metadata	109
Import Options	110
DTD for the Interchange Format	113
Using the DTDs	113
DTD for Version 1 of the Interchange Format	114
DTD for Version 2 of the Interchange Format	122
DTD for Version 3 of the Interchange Format	131
DTD for Version 4 of the Interchange Format	140
DTD for Version 5 of the Interchange Format	150
Frame Rates	160
Keyframe Interpolation	161

About Final Cut Pro Interpolation 161

Apple Events and Final Cut Pro 166

Final Cut Pro Apple Events 168

Versions of XMEML and Final Cut 173

Versions of XMEML 173

Version 1 173

Version 2 173

Version 3 174

Version 4 174

Version 5 175

Versions of Final Cut Pro 175

Changes in Final Cut Pro 6.0.2 175

Changes in Final Cut Pro 7.0 175

Document Revision History 177

Index 179

Figures, Tables, and Listings

Document Export and Import 9

- Figure 2-1 Export XML dialog 10
- Figure 2-2 Final Cut Pro XML file icon 10
- Figure 2-3 Import XML dialog 11

Basics of Encoding 13

- Listing 3-1 A minimum clip 15
- Listing 3-2 A clip with additional information 16
- Listing 3-3 A clip with associated media 18
- Listing 3-4 A clip using inheritance 19
- Listing 3-5 A clip with timecode and media 19
- Listing 3-6 Encoding audio information 21
- Listing 3-7 An exported Browser clip 23
- Listing 3-8 A subclip 24
- Listing 3-9 A video multiclip 27
- Listing 3-10 An empty sequence 29
- Listing 3-11 Format information for a video track 30
- Listing 3-12 Format information for an audio track 31
- Listing 3-13 A sequence with a video track 32
- Listing 3-14 An audio element with two tracks 33
- Listing 3-15 The link element 35
- Listing 3-16 The linkclipref element 37
- Listing 3-17 A transition between video clips 38
- Listing 3-18 A project 41
- Listing 3-19 A gaussian blur effect 43
- Listing 3-20 Radius parameter with keyframes and Bezier handles 45

XMEML Topics 48

- Figure 4-1 Media Timing Values 49
- Figure 4-2 Sequence Timing Values 49
- Figure 4-3 Multiclip Timing Values 50
- Listing 4-1 Replacing a Clip in the Browser 53
- Listing 4-2 Replacing a Clip in a Bin 54
- Listing 4-3 Sample Metadata 56

- Listing 4-4 Adding Metadata 57
- Listing 4-5 Removing Metadata 57

Applications of the Interchange Format 59

- Figure 5-1 Multiple parameter variations in the Canvas 70
- Listing 5-1 An enhanced batch list 59
- Listing 5-2 Subtitling 65
- Listing 5-3 Effect parameter settings 71

DTD for the Interchange Format 113

- Listing A-1 Version 1 DTD 114
- Listing A-2 Version 2 DTD 122
- Listing A-3 Version 3 DTD 131
- Listing A-4 Version 4 DTD 140
- Listing A-5 Version 5 DTD 150

Frame Rates 160

- Table B-1 Frames per second 160

Keyframe Interpolation 161

- Listing C-1 Acceleration coefficients 161
- Listing C-2 Control points 162
- Listing C-3 Locations 163
- Listing C-4 Linear segments 163
- Listing C-5 Bezier curve 164

About This Document

The Final Cut Pro XML Interchange Format provides extensive access to the contents of Final Cut Pro projects, including edits and transitions, effects, layer-compositing information, and organizational structures. Using the interchange format, you can process project content in ways that supplement the capabilities of the Final Cut Pro application itself. You can also share Final Cut Pro information with other applications or systems that support XML—including nonlinear editors, asset management systems, database systems, and broadcast servers.

Important: Final Cut Studio includes the Final Cut Pro application and companion applications such as Cinema Tools, Compressor, Color, DVD Studio Pro, Motion, and Soundtrack. Color, Cinema Tools, and Final Cut Pro itself make use of the Final Cut Pro XML Interchange Format.

The main body of this document describes version 5 of the Final Cut Pro XML Interchange Format. The appendix [Versions of XMEML and Final Cut](#) (page 173) compares versions 1, 2, 3, 4, and 5 of the interchange format.

This document assumes that you are familiar with XML conventions and with the Final Cut Pro 7 application.

Organization

This document contains the following chapters and appendixes:

- [Document Export and Import](#) (page 9) discusses exporting, importing, and validating interchange format documents.
- [Basics of Encoding](#) (page 13) reviews the key elements of the interchange format.
- [XMEML Topics](#) (page 48) provides information about selected Final Cut Pro XML topics.
- [Applications of the Interchange Format](#) (page 59) illustrates some applications of the interchange format to various tasks.
- [Elements Catalog](#) (page 79) provides reference information about the elements of the interchange format.
- [DTD for the Interchange Format](#) (page 113) lists the DTDs for versions 1, 2, 3, 4, and 5 of the interchange format.
- [Frame Rates](#) (page 160) indicates the values required to specify various types of video and associated frame rates.

- [Keyframe Interpolation](#) (page 161) explains the interpolation method Final Cut Pro uses to construct Bezier curves in keyframes.
- [Apple Events and Final Cut Pro](#) (page 166) discusses using Apple Events to export or import interchange format documents.
- [Versions of XMEML and Final Cut](#) (page 173) provides information about various versions of XMEML and Final Cut Pro.
- [Document Revision History](#) (page 177) provides a history of changes to this document.

This document also has an index.

XML Resources

Here are some recommended XML resources:

- For a general introduction XML and the Final Cut Pro XML Interchange Format, see the section “Using Final Cut Pro XML and QuickTime Metadata” in the *Final Cut Pro 7 User Manual*.
- For a comprehensive reference guide to XML-related topics, see *XML In a Nutshell, Third Edition*, published by O'Reilly. ISBN 0-596-00764-7.
- For a useful resource of XML-related information, see *XML From the Inside Out* (<http://xml.com>).
- On the Apple developer website, the document [Core Foundation XML](#) describes the Core Foundation objects you can use to parse XML. (Cocoa developers please visit this [link](#).)

Document Export and Import

Documents in the Final Cut Pro XML Interchange Format are stored on disk as plain text documents that you can view, parse, and edit on any platform.

You can validate an interchange format document using an interchange format DTD. (See [DTD for the Interchange Format](#) (page 113))

You can export an interchange format document from Final Cut Pro, or import an interchange format document into Final Cut Pro, using either the Final Cut Pro user interface or, alternatively, Apple Events from an independent application. This chapter describes importing and exporting with the user interface. For information about using Apple Events see [Apple Events and Final Cut Pro](#) (page 166)

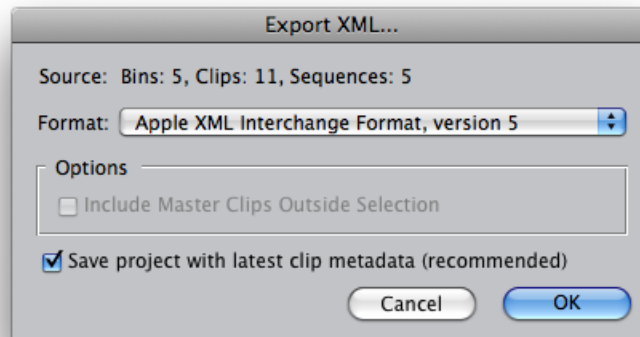
Exporting Documents

To export an interchange format document from Final Cut Pro:

1. Select an item in the Browser that you want to describe in an interchange format document. (If nothing is selected, a description of the entire contents of the Browser is exported.)
2. Choose File > Export > XML.
3. Check the Export XML dialog to make sure that the source for the interchange format document is correct. In Figure 2-1 the source is a single sequence.
4. Choose the version of the XML Interchange Format for the document.
5. If clips in your source have master clips, you can select the option to Include Master Clips Outside Selection.
6. Select or deselect the option “Save project with latest clip metadata.” Since XML Export may make changes that affect project metadata (such as `itemhistory`), saving the project at the time you export ensures that this metadata is persistent. On the other hand, not saving the project can ensure the user’s ability to undo changes.
7. Click OK.

8. Enter a title for the interchange format document and choose a location. (You may want to use `.xml` as a suffix.)

Figure 2-1 Export XML dialog



Interchange format documents you export from Final Cut Pro have the Final Cut Pro XML file icon.

Figure 2-2 Final Cut Pro XML file icon



Importing Documents

There are several ways you can import an interchange format document into Final Cut Pro.

You can use the Import menu to import any interchange format document:

1. Choose File > Import > XML.
2. Navigate to the document, select it, and then click Choose.

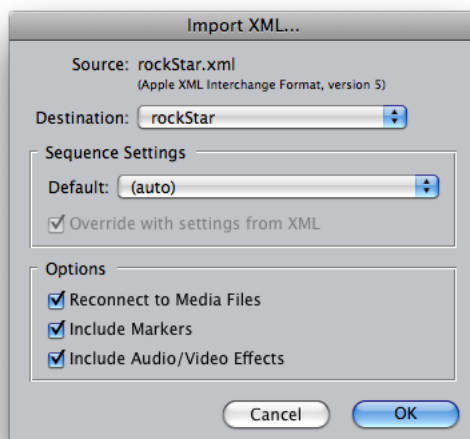
If the document file has a Final Cut Pro XML file icon, you can do any of the following:

- Double-click the document.
- Drag the document to the Final Cut Pro icon on the desktop or in the Dock.

- Use the Import menu.

When you import an interchange format document, the Import XML dialog appears.

Figure 2-3 Import XML dialog



The settings in this dialog are:

- *Destination*: Select an existing project as the destination of the import, or create a new project for the import.
- *Sequence Settings*: Select the sequence settings for the import. The default setting is (auto). This means that Final Cut Pro conforms the sequence settings of the imported XML to the settings specified in the XML. (If the XML doesn't specify sequence settings, Final Cut uses the settings of the first edited clipitem in each sequence. If no settings are available, Final Cut uses the default Sequence Preset.)
- *Reconnect to Media Files*: Select this option to tell Final Cut Pro to reconnect browser and sequence clips encoded in the file with associated media when possible. If you do not select this option, all imported clips are offline.
- *Include Markers*: Select this option to import any clip or sequence marker information encoded in the file.
- *Include Audio/Video Effects*: Select this option to import any encoded audio or video filters for Browser or sequence clips. This setting does not affect encoded audio or video transitions.

Note: You can bypass the Import XML dialog and import a document directly into Final Cut Pro by using the element `importoptions` and its subelements. For details, see [Import Options](#) (page 110)

Errors During Import

When you import an interchange format document, Final Cut Pro first checks to make sure the document is well-formed and that all required elements are present. It then translates the interchange format document into Final Cut Pro components.

If Final Cut Pro finds critical errors during import, the interchange format document is not imported. If noncritical errors occur, the document is imported, though the components constructed from the document may differ from your expectations.

Information about noncritical errors appears in an Final Cut Pro XML Error Log Window. (Note that you can suppress the reporting of noncritical errors by setting `displaynoncriticalerrors` to `FALSE`. See [Import Options](#) (page 110)

In general, you should study error information about a document you are attempting to import, modify the document, and then reimport it, continuing this process until you can import the document without errors.

Basics of Encoding

This chapter provides an introduction to the basic features of the Final Cut Pro XML Interchange Format. In particular, it explains how the interchange format can encode a clip, a sequence, a project, and an effect.

For examples of applying the interchange format to specific tasks, see [Applications of the Interchange Format](#) (page 59)

For reference documentation about the elements of the interchange format, see [Elements Catalog](#) (page 79)

For information about the various versions of the interchange format, see [Versions of XMEML and Final Cut](#) (page 173)

This chapter contains the following sections:

- [Special Conventions](#) (page 14) introduces the inheritance and `id` conventions of the interchange format.
- [Encoding a Clip](#) (page 15) shows how to encode a clip.
- [Encoding a Sequence](#) (page 29) details encoding a sequence.
- [Encoding a Project](#) (page 41) explains the requirements for encoding a project.
- [Encoding Effects](#) (page 43) looks at encoding effects.

Special Conventions

The Final Cut Pro XML Interchange Format uses a set of special conventions beyond the base conventions of XML that affect how a document is parsed and translated when you import it into Final Cut Pro. These conventions—inheritance and the `id` attribute— make it possible to efficiently construct more concise interchange format documents.

Note: Interchange format documents you export from Final Cut Pro only use these conventions in certain circumstances.

Inheritance

Information for various components of a project is frequently redundant. For example, a sequence and its contents usually have the same frame rate; most clips share the same source media file across multiple tracks; multiple clips in a batch list are likely share the same Reel/Tape identifier, and so on.

You can explicitly encode all the information for each component in a project. Alternatively, you can use the inheritance convention of the Final Cut Pro XML Interchange Format with certain elements to encode shared information just once. For example, you can encode a sequence and specify a frame rate of 29.97 fps for the sequence. Clips that are subelements of this sequence inherit this frame rate unless you specifically encode a different rate for a particular clip.

An example of the inheritance convention appears in [Listing 3-4](#) (page 19)

The `id` Attribute

The `id` attribute lets you share information between certain "peer" elements—elements that are not in an element/subelement relationship with each other. If an element provides an `id` attribute, Final Cut Pro registers this element in a reference table during import and translation. Then, during subsequent translation, other elements can reference this registered information.

For example, the clip information for a video track in a sequence may be identical to the clip information in an audio track in the same sequence. Rather than repeating this information for both tracks, you can encode it once and then reference the information using the `id` attribute.

Note: Not all elements support the use of the `id` attribute. See [Elements Catalog](#) (page 79) for information about the `id` attribute for specific elements.

An example of the `id` convention appears in [Listing 3-14](#) (page 33)

Encoding a Clip

A clip represents a virtual piece of time-based media. A `clip` element has three required subelements: `name`, `duration`, and `rate`. (When a clip appears in a sequence, you also need to specify `start` and `end` subelements.)

Listing 3-1 shows an example of a minimum clip:

Listing 3-1 A minimum `clip`

```
<?xml version="1.0" encoding="UTF-8"?> // 1
<!DOCTYPE xmeml> // 2
<xmeml version="5"> // 3
  <clip>
    <name>Jeremy Solo</name>
    <duration>188</duration> // 4
    <rate> // 5
      <ntsc>TRUE</ntsc>
      <timebase>30</timebase>
    </rate>
  </clip>
</xmeml>
```

Note: The Final Cut Pro application uses the internal name `xmeml` for the Final Cut Pro XML Interchange Format.

Lines numbered 1 and 2 are the prologue section. Line numbered 1 declares the version of XML and the text encoding; line numbered 2 the document type. Line numbered 3 is the start tag for the root element `xmeml` that contains all the other elements in an interchange format document. The version attribute in this start tag specifies the version of the Final Cut Pro XML Interchange Format.

At line numbered 4, the `duration` element encodes the total number of frames in the clip. This value does not change even if you set In and Out points for the clip using the `in` and `out` elements. (It is the equivalent of the Length timing property, rather than the Duration property.)

At line numbered 5, the `rate` element defines the frame rate for the clip—in this case, an NTSC clip with a frame rate of 29.97 fps. Note that this frame rate is specified using the `ntsc` subelement, which takes a Boolean value, and the `timebase` subelement, which takes an integer value (30). The NTSC frame rate variant is defined as a 0.01 percent reduction in frame rate. The actual mathematical value of this variant is 29.970029970029...

(repeating). The decimal 29.97 is not an accurate representation of this number. Using an interger and an NTSC indicator flag allows any application to process the rate information in its own manner. See [Table B-1](#) (page 160) for a complete listing of the possible settings for `timebase` and `ntsc`.

Listing 3-2 shows a clip with more information, including In and Out points, master clip, logging, label, and comment information.

Listing 3-2 A clip with additional information

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE xmeml>
<xmeml version="5">
  <clip id="Jeremy Solo">
    <name>Jeremy Solo</name>
    <duration>188</duration>
    <rate>
      <ntsc>TRUE</ntsc>
      <timebase>30</timebase>
    </rate>
    <in>14</in>
    <out>149</out>
    <masterclipid>Jeremy Solo</masterclipid> // 1
    <ismasterclip>TRUE</ismasterclip> // 2
    <logginginfo>
      <description>solo male</description>
      <scene>ballroom</scene>
      <shottake>Shot1/Take1</shottake>
      <lognote>LogNote1</lognote>
      <good>TRUE</good>
    </logginginfo>
    <filmdata> . . . </filmdata>
    <labels>
      <label>Good Take</label>
      <label2>Label 2 (1)</label2>
    </labels>
    <comments>
```



```

    <mastercomment1>MC1</mastercomment1>
    <mastercomment2>MC2</mastercomment2>
    <mastercomment3>MC3</mastercomment3>
    <mastercomment4>MC4</mastercomment4>
    <clipcommenta>CommentA</clipcommenta>
    <clipcommentb>CommentB</clipcommentb>
  </comments>
</clip>
</xmml>
```

At line numbered 1, the `masterclipid` element provides a reference to the master clip's `clip_id` attribute. (As in this example, a master clip lists its own `clip_id` attribute.)

At line numbered 2, the `ismasterclip` element indicates whether or not a clip is a master clip. (In this case, it is.)

About master clips: For information about master clips, see the chapter “Working with Master and Affiliate Clips” in Volume 4 of the *Final Cut Pro User Manual*.

For details about the other elements in this listing, see the reference information in [Elements Catalog](#) (page 79). Check the index to locate a particular element.

Associating Media With a Clip

To associate media with a clip, you use the `file` element. Listing 3-3 shows a clip associated with a media file.

Listing 3-3 A clip with associated media

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE xmeml>
<xmeml version="5">
  <clip id="Jeremy Solo">
    <name>Jeremy Solo</name>
    <duration>188</duration>
    <rate>
      <ntsc>TRUE</ntsc>
      <timebase>30</timebase>
    </rate>
    <file id="Jeremy Solo1">
      <name>Jeremy Solo1</name>
      <pathurl>file://localhost/Jeremy%20Solo</pathurl>           // 1
      <rate>                                                       // 2
        <timebase>30</timebase>
        <ntsc>TRUE</ntsc>
      </rate>
      <duration>188</duration>                                     // 3
    </file>
  </clip>
</xmeml>
```

At line numbered 1, the `pathurl` element specifies the location of the media. Note that the name of the file includes a space, which is encoded by the escape character (%) and the hexadecimal value for the space character (20). See [pathurl](#) (page 83) for more details.

At lines numbered 2 and 3, the `rate` and `duration` subelements are required for the `file` element. However, because they are specified earlier in the document by the parent element, the `file` element can inherit this information. Listing 3-4 shows how this works.

Listing 3-4 A clip using inheritance

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE xmeml>
<xmeml version="5">
  <clip id="Jeremy Solo">
    <name>Jeremy Solo</name>
    <duration>188</duration>
    <rate>
      <ntsc>TRUE</ntsc>
      <timebase>30</timebase>
    </rate>
    <file id="Jeremy Solo1">
      <name>Jeremy Solo1</name>
      <pathurl>file://localhost/Jeremy%20Solo</pathurl>
    </file>
  </clip>
</xmeml>
```

Note: Some subelements may be inherited, others may not. [Elements Catalog](#) (page 79) indicates if a subelement can be inherited or not.

The file element can also have optional subelements: `timecode` and `media`. Listing 3-5 is an example.

Listing 3-5 A clip with timecode and media

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE xmeml>
<xmeml version="5">
  <clip id="Jeremy Solo">
    <name>Jeremy Solo</name>
    <duration>188</duration>
    <rate>
      <ntsc>TRUE</ntsc>
      <timebase>30</timebase>
    </rate>
```

```
<file id="Jeremy Solo1">
  <name>Jeremy Solo1</name>
  <pathurl>file://localhost/Jeremy%20Solo</pathurl>
  <timecode>                                     // 1
    <string>02:42:35;14</string>
    <frame>292372</frame>
    <displayformat>DF</displayformat>
    <source>source</source>
    <reel>
      <name>002</name>
    </reel>
  </timecode>
  <media>                                         // 2
    <video>
      <duration>188</duration>
      <samplecharacteristics>
        <width>720</width>
        <height>480</height>
      </samplecharacteristics>
    </video>
  </media>
</file>
</clip>
</xmml>
```

At line numbered 1, the `timecode` element encodes the media starting point. It includes the subelements `string`, which specifies the starting timecode in SMPTE format; `frame`, which specifies the starting frame count; and `displayformat`, which specifies drop frame or non-drop frame format. Note that either the element `string`, or the elements `frame` and `format` are required, but not all three. Other subelements include `source`, which categorizes the timecode as a source timecode from the tape, and `reel`.

At line numbered 2, the `media` element encodes information about the media, including the `height` and `width` in pixels.

Encoding Audio Information

Media can also include audio tracks. You use the `audio` element to specify information about these tracks, as seen in Listing 3-6.

Listing 3-6 Encoding audio information

```
<file>
  <name>myFile.mov</name>
    <pathurl> file://localhost/myFile.mov</pathurl>
    <rate>
      <timebase>24</timebase>
      <ntsc>TRUE</ntsc>
    </rate>
    <duration>3671</duration>
    <media>
      <video>...</video>
      <audio>                                     // 1
        <samplecharacteristics>
          <samplerate>48000</samplerate>
          <depth>16</depth>
        </samplecharacteristics>
        <layout>stereo</layout>
        <channelcount>2</channelcount>
        <audiochannel>
          <channellabel>right</channellabel>
        </audiochannel>
        <audiochannel>
          <channellabel>left</channellabel>
        </audiochannel>
      </audio>
      <audio>                                     // 2
        <samplecharacteristics>
          <samplerate>48000</samplerate>
          <depth>16</depth>
        </samplecharacteristics>
        <channelcount>2</channelcount>
```

```
        </audio>  
    </media>  
</file>
```

At line numbered 1, the first stereo audio track is encoded. It has two channels. The first channel is specified as the right channel, the second as the left channel. (This is the reverse of the default settings.)

At line numbered 2, the second audio track in the media is encoded. When layout is not specified, stereo is assumed

See [Elements Catalog](#) (page 79) for more information about the subelements used to encode audio information.

Exporting a Clip from the Browser

When you export a single clip from the Browser, the resulting listing includes the `media` element with track information. Listing 3-7 shows excerpts from such an export.

Listing 3-7 An exported Browser clip

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE xmeml>
<xmeml version="5">
  <clip id="Jeremy Solo">
    <uuid>970C270C-8692-4E6A-A58C-F378CC3EB776</uuid> // 1
    <updatebehavior>add</updatebehavior>
    <name>Jeremy Solo</name>
    <duration>188</duration>
    <rate>
      <ntsc>TRUE</ntsc>
      <timebase>30</timebase>
    </rate>
    <in>14</in> // 2
    <out>149</out>
    <masterclipid>Jeremy Solo</masterclipid>
    <logginginfo>
      <good>FALSE</good>
    </logginginfo>
    <media>
      <video> // 3
        <track> // 4
          <clipitem> // 5
            <name>Jeremy Solo</name>
            <duration>188</duration>
            <rate>
              <ntsc>TRUE</ntsc>
              <timebase>30</timebase>
            </rate>
            <enabled>TRUE</enabled>
            <anamorphic>FALSE</anamorphic>
```

```
        <alphatype>none</alphatype>
        <file id="Jeremy Solo1">
            . . .
        </file>
        <sourcetrack>
            <mediatype>video</mediatype>
        </sourcetrack>
    </clipitem>
    <enabled>TRUE</enabled>
    <locked>FALSE</locked>
</track>
</video>
</media>
</clip>
</xmml>
```

At line numbered 1, the `uuid` element and the `updatebehavior` element in the next line let you manage project components. See [Managing Project Components](#) (page 53) You can also use `updatebehavior` to manage QuickTime metadata. See [About master clips](#) (page 17)

At line numbered 3, the `video` element identifies the type of media.

At line numbered 4, the `track` element starts the encoding for the first (and only) track in the media.

At line numbered 5, the `clipitem` element encodes information that is specific to the associated media. See [Media in a Sequence](#) (page 32).

Encoding a Subclip

You could use the clip encoded in Listing 3-7 to generate a subclip. Listing 3-8 shows the results (with some details omitted).

Listing 3-8 A subclip

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE xmml>
<xmml version="5">
    <clip id="Jeremy Solo Subclip">
```



```
<name>Jeremy Solo Subclip</name>
<duration>135</duration>                                     // 1
<rate>
    <ntsc>TRUE</ntsc>
    <timebase>30</timebase>
</rate>
<in>-1</in>                                                  // 2
<out>-1</out>
<masterclipid>Jeremy Solo Subclip</masterclipid>
<subclipmasterid>Jeremy Solo</subclipmasterid>              // 3
. . .
<media>
    <video>
        <track>
            <clipitem>
                <name>Jeremy Solo Subclip</name>
                <duration>135</duration>
                . . .
                <in>0</in>
                <out>135</out>
                <start>0</start>
                <end>135</end>
                <subclipinfo>                                  // 4
                    <startoffset>14</startoffset>
                    <endoffset>39</endoffset>
                </subclipinfo>
                . . .
                <file id="Jeremy Solo">
                    . . .
                    <media>
                        <video>
                            <duration>188</duration>          // 5
                            <samplecharacteristics>
                                <width>720</width>
                                <height>480</height>
```

```
                </samplecharacteristics>
            </video>
        </media>
    </file>
    <sourcetrack>
        <mediatype>video</mediatype>
    </sourcetrack>
</clipitem>
<enabled>TRUE</enabled>
<locked>FALSE</locked>
</track>
</video>
</media>
</clip>
</xmml>
```

At line numbered 1, the duration of the subclip is 135.

At line numbered 2, the values for `in` and `out` are `-1`, indicating that no In and Out points are set.

New Browser-level subclips automatically become their own master clips. At line numbered 3, the `subclipmasterid` element specifies the id of the subclip's parent clip.

At line numbered 4, the `subclipinfo` element indicates the starting offset and ending offset for the subclip. These values are offsets in frames from the media start and media end of the associated media file, calculated using the `in` and `out` points for the clip from which the subclip was generated. (See Listing 3-7 line numbered 1.)

At line numbered 5, the value for the `duration` element for the media associated with the subclip remains at 188, compared to the value of 135 at line numbered 2.

Encoding a Multiclip

Multiclips let you group multiple camera angle clips together and switch or cut between angles in real time. You can also make a multiclip with multiple audio tracks. A video multiclip is represented by Listing 3-9

Listing 3-9 A video multclip

```
<clip id="Billiards1">
  <name>Billiards1</name>
  <duration>2700</duration>
  <rate>
    <ntsc>TRUE</ntsc>
    <timebase>30</timebase>
  </rate>
  <in>-1</in>
  <out>-1</out>
  <ismasterclip>FALSE</ismasterclip>
  <media>
    <video>
      <track>
        <clipitem>
          <name>Billiards1</name>
          <duration>2700</duration>
          . . .
          <multiclip id="Multiclip 1 ">                                // 1
            <name>Multiclip 1</name>
            <collapsed>FALSE</collapsed>
            <synctype>1</synctype>
            <angle>                                                    // 2
              <activevideoangle>TRUE</activevideoangle>
              <activeaudioangle>TRUE</activeaudioangle>
              <clip id="Billiards1"/>
            </angle>
            <angle>
              <clip id="Billiards2">
                <name>Billiards2</name>
                <duration>2700</duration>
                . . .
                <defaultangle>2</defaultangle>                        // 3
              </clip>
            </angle>
```

```
        <angle>
            <clip id="Billiards3">
                <name>Billiards3</name>
                <duration>2700</duration>
                . . .
                <defaultangle>3</defaultangle>
            </clip>
        </angle>
    </multiclip>
    <mediadelay>900</mediadelay> // 4
</clipitem>
<enabled>TRUE</enabled>
<locked>FALSE</locked>
</track>
</video>
</media>
</clip>
```

At line numbered 1, the `multiclip` element is a subelement of `clipitem`. It has two required subelements: `name` and `angle`. Other subelements include `collapsed`, which indicates if the multiclip is collapsed or not, and `synctype`, which specifies how the angles in the multiclip should be synchronized.

At line numbered 2, the `angle` element includes the optional subelements `activevideoangle` and `activeaudioangle`. These take Boolean values (TRUE or FALSE) and indicate whether an angle is active or not. (Only one angle in a multiclip can be active at a time.) If these elements are not specified in an angle, the default settings are false. If no angles in a multiclip are encoded with these subelements, the first angle is assumed to be the active angle.

At line numbered 3, the element `defaultangle` is used for clips in the Browser. It encodes identifying information (a number or a letter) about the camera angle.

At line numbered 4, the element `mediadelay` is a subelement of `clipitem`. It shifts the multiclip sync point from the start of the clip item to the specified delay. (See [Figure 4-3](#) (page 50)).

Encoding a Sequence

To encode a sequence in the Final Cut Pro XML Interchange Format, you use the `sequence` element. You specify subelements for the name of the sequence, the duration, the frame rate, and the starting timecode. You also use the `media` element to encode subelements for the video and audio tracks.

Listing 3-10 shows an empty sequence with one video track and two audio tracks.

Listing 3-10 An empty sequence

```
<?xml version="1.0" encoding="UTF-8"?>
<xmeml version="5">
  <sequence>
    <name>Sequence 1</name>
    <duration></duration>
    <rate>. . .</rate>
    <timecode>. . .</timecode>
    <media>
      <video>
        <format></format> // 1
        <track></track>
      </video>
      <audio>
        <format></format> // 2
        <outputs></outputs> // 3
        <track></track>
        <track></track>
      </audio>
    </media>
  </sequence>
</xmeml>
```

Format and Outputs

At lines numbered 1 and 2, the `format` element in Listing 3-10 encodes information about the video and audio media, while, at line numbered 3, the `outputs` element encodes information about the audio media. These elements and their subelements are equivalent to the sequence settings in Final Cut Pro.

Listing 3-11 shows sample format information for a video track. See [Elements Catalog](#) (page 79) for details about the elements displayed here.

Listing 3-11 Format information for a video track

```
<video>
  <format>
    <samplecharacteristics>
      <width>720</width>
      <height>480</height>
      <anamorphic>FALSE</anamorphic>
      <pixelaspectratio>NTSC-CCIR-601</pixelaspectratio>
      <fielddominance>lower</fielddominance>
      <rate>
        <ntsc>TRUE</ntsc>
        <timebase>30</timebase>
      </rate>
      <colordepth>24</colordepth>
      <codec>
        <name>Apple DV - NTSC</name>
        <appspecificdata>
          <appname>Final Cut Pro</appname>
          <appmanufacturer>Apple Computer, Inc.</appmanufacturer>
          <appversion>4.0</appversion>
          <data>
            <qtcodec>                                     // 1
              . . .
            </qtcodec>
          </data>
        </appspecificdata>
      </codec>
    </samplecharacteristics>
    <appspecificdata>
      <appname>Final Cut Pro</appname>
      <appmanufacturer>Apple Computer, Inc.</appmanufacturer>
      <appversion>4.0</appversion>
```

```
        <data>
            <fcpimageprocessing>                                // 2
                <useyuv>TRUE</useyuv>
                <usesuperwhite>FALSE</usesuperwhite>
                <rendermode>YUV8BPP</rendermode>
            </fcpimageprocessing>
        </data>
    </appspecificdata>
</format>
. . .
</video>
```

At line numbered 1, the `qtcodec` element provides information (omitted) about a QuickTime codec.

At line numbered 2, the `fcpimageprocessing` element encodes details about image processing settings.

Listing 3-12 shows sample `format` and `outputs` information for an audio track.

Listing 3-12 Format information for an audio track

```
<audio>
    <format>
        <samplecharacteristics>                                // 1
            <depth>16</depth>
            <samplerate>48000</samplerate>
        </samplecharacteristics>
    </format>
    <outputs>                                                  // 2
        <group>
            <index>1</index>
            <numchannels>2</numchannels>
            <downmix>4</downmix>
            <channel>
                <index>1</index>
            </channel>
            <channel>
                <index>2</index>
```

```
        </channel>
    </group>
</outputs>
```

At line numbered 1, the `samplecharacteristics` element encodes information about the audio depth and sample rate.

At line numbered 2, the `outputs` element specifies information about the audio output settings.

Media in a Sequence

To add media to a sequence, you use a `clipitem` subelement in the `track` element.

Listing 3-13 A sequence with a video track

```
<sequence id = "Sequence 1">
. . .
    <media>
        <video>
            <format>
                . . .
            </format>
            <track>
                <clipitem id = "Seq1Clip1">
                    <name>Rob Dialog</name>
                    <duration>751</duration>
                    <in>0</in> // 1
                    <out>751</out>
                    <start>0</start> // 2
                    <end>751</end>
                    . . .
                <file id = "Rob Dialog">
                    <name>Rob Dialog</name>
                    <pathurl>. . .</pathurl>
                    <duration>751</duration>
                    . . .
```



```
        </file>
        <sourcetrack>                                     // 3
            <mediatype>video</mediatype>
            <trackindex>1</trackindex>
        </sourcetrack>
    </clipitem>
</track>
</video>
</media>
. . .
</sequence>
```

A `clipitem` encoded in a track is similar to a `clip` in the Browser. It has a name, duration, timecode, associated media file, and so on. A `clipitem` in a sequence track is similar to a `clipitem` in a track of a `clip`. The only difference is that you can have more than one `clipitem` per track in a sequence.

At line numbered 1, it has `in` and `out` subelements that indicate the portion of the source media file to reference. At line numbered 2, it has `start` and `end` subelements that specify the relative position of the clip in the parent sequence. (See [Timing Values](#) (page 49).)

At line numbered 3, the `sourcetrack` element indicates the media track to use in the source media file; in this case, video track 1.

To encode the stereo audio that goes with this video clip, you use an `audio` element and the first two `track` subelements within it.

Listing 3-14 An audio element with two tracks

```
<sequence id = "Sequence 1">
. . .
    <media>
        <video>
            <format>
                . . .
            </format>
            <track>
                <clipitem id = "Seq1Clip1">               // 1
                    <name>Rob Dialog</name>
```

```
        <duration>751</duration>
        <in>0</in>
        <out>751</out>
        <start>0</start>
        <end>751</end>
        <file id = "Rob Dialog">
        . . .
        </file>
        . . .
    </clipitem>
</track>
</video>
<audio> // 2
    <format>
    . . .
    </format>
    <outputs>
    . . .
    </outputs>
    <track>
        <clipitem id = "Seq1Clip1"> // 3
            <file id = "Rob Dialog"/> // 4
            <sourcetrack> // 5
                <mediatype>audio</mediatype>
                <trackindex>1</trackindex>
            </sourcetrack>
        </clipitem>
    </track>
    <track> // 6
        <clipitem id = "Seq1Clip1">
            <file id = "Rob Dialog"/>
            <sourcetrack>
                <mediatype>audio</mediatype>
                <trackindex>2</trackindex>
```

```
        </sourcetrack>
      </clipitem>
    </track>
  </audio>
</media>
. . .
</sequence>
```

At line numbered 2, you have the start tag of the `audio` element.

At line numbered 3, you can reference the `id` attribute specified earlier at line numbered 1. See [The id Attribute](#) (page 14) You don't have to repeat all the information for duration, the in and out points, and so on.

At line numbered 4, you can reference the `id` attribute for the media file and then override the source track information at line numbered 5. You are using the same source media that you used for the video track, but are now referencing audio track 1. Starting at line numbered 6, you can construct the information for the second audio track in the same manner.

Linking

With the video track and the two audio tracks encoded, you now need to link the tracks together so that they are managed as a single component within the sequence. You do this using the `link` element.

Listing 3-15 The `link` element

```
<media>
  <video>
    <format> ... </format>
    <track>
      <clipitem id = "Seq1Clip1">
        <name>Rob Dialog</name>
        <duration>751</duration>
        <in>0</in>
        <out>751</out>
        <start>0</start>
        <end>751</end>
      . . .
```

```
<link>                                                                    // 1
    <mediatype>video</mediatype>
    <trackindex>1</trackindex>
    <clipindex>1</clipindex>
</link>
<link>
    <mediatype>audio</mediatype>
    <trackindex>1</trackindex>
    <clipindex>1</clipindex>
    <groupindex>1</groupindex>
</link>
<link>
    <mediatype>audio</mediatype>
    <trackindex>2</trackindex>
    <clipindex>1</clipindex>
    <groupindex>1</groupindex>
</link>
<file id = "Rob Dialog">
    . . .
</file>
</clipitem>
</track>
</video>
<audio>
    <format> . . . </format>
    <outputs> . . . </outputs>
    <track>
        <clipitem id = "Seq1Clip1">
            <file id = "Rob Dialog">
                . . .
            </file>
        </clipitem>
    </track>
</track>
```

```
<clipitem id = "Seq1Clip1">
  <file id = "Rob Dialog">
    . . .
  </file>
</clipitem>
</track>
</audio>
</media>
```

The link information always appears in the first track in the group of tracks that you are linking together. Here, this is the video track at line numbered 1, which is linked with the two audio tracks. Since the two audio clips use the `id` convention to reference the contents of the video clip, all three clips use the same link information.

Alternatively, you can link tracks together using the `id` attribute and the `linkclipref` element.

Listing 3-16 The `linkclipref` element

```
<media>
  <video>
    <format> . . .</format>
    <track>
      <clipitem id = "Seq1Clip1v">
        <name>Clip 001</name>
        <duration>751</duration>
        <in>0</in>
        <out>751</out>
        <start>0</start>
        <end>751</end>
        <link>
          <linkclipref>Seq1Clip1v</linkclipref> // 1
        </link>
        <link>
          <linkclipref>Seq1Clip1a1</linkclipref> // 2
          <groupindex>1</groupindex>
        </link>
        <link>
```

```
        <linkclipref>Seq1Clip1a2</linkclipref>                                // 3
        <groupindex>1</groupindex>
    </link>
    <file id = "Rob Dialog">
        . . .
    </file>
</clipitem>
</track>
</video>
<audio>
    <format> . . . </format>
    <outputs> . . . </outputs>
    <track>
        <clipitem id = "Seq1Clip1a1">
            <file id="Rob Dialog"/>
        </clipitem>
    </track>
    <track>
        <clipitem id = "Seq1Clip1a2">
            <file id="Rob Dialog"/>
        </clipitem>
    </track>
</audio>
</media>
```

Note that you cannot use `linkclipref` for multiple clips in a track with the same ID.

Transitions

A sequence usually includes transitions between video clips you can encode with a `transitionitem` element. Listing 3-17 shows an example of a sequence with a single video track with a transition between two clips.

Listing 3-17 A transition between video clips

```
<sequence>
    <name>Sequence 1</name>
```

```
<duration>243</duration>
. . .
<media>
  <video>
    <format> . . . </format>
    <track>
      <clipitem>
        <name>Bloomies</name>
        <duration>248</duration>
        <in>74</in>
        <out>227</out> // 1
        <start>0</start>
        <end>-1</end> // 2
        <file id = "Bloomies1">
          <duration>248</duration>
          . . .
        </file>
        . . .
      </clipitem>
      <transitionitem> // 3
        <rate>. . .</rate>
        <start>123</start>
        <end>153</end>
        <alignment>center</alignment>
        <effect> // 4
          <name>Cross Dissolve</name>
          <!-- encoding details for the effect -->
        </effect>
      </transitionitem>
      <clipitem>
        <name>Jeremy Solo</name>
        <duration>188</duration>
        <in>36</in> // 5
        <out>156</out>
        <start>-1</start> // 6
```

```
        <end>243</end>
        <file id = "Jeremy Solo">
            <duration>188</duration>
            . . .
        </file>
        . . .
    </clipitem>
</track>
</video>
</media>
</sequence>
```

At line numbered 3, the `transitionitem` element encodes the start and end of the transition.

At line numbered 4, the `effect` subelement determines the specific transition that is used. See [Encoding Effects](#) (page 43) for a discussion of effects.

In the Viewer, the Out point for the first clip in the sequence (Bloomies) was set at 212. Note that at line numbered 1, the Out point is extended to 227 to provide additional frames for input into the transition effect. In a similar fashion, the In point for the second clip (Jeremy Solo) at line numbered 5 is changed to 36 from its original setting of 51. Also, because the end time for the first clip now depends on the next item in the sequence, the end element at line numbered 2 is set to `-1`. This indicates that the value should be computed. Similarly, the `start` element for the second clip at line numbered 6 is set to `-1`.

PSD Files as Sequences

Final Cut encodes a Photoshop PSD file with multiple layers as a sequence (rather than a clip) when you import it into the Browser. The `stillframe` element is set to `TRUE` and the `layerindex` element is used to specify the various layers in the file.

Encoding a Project

Listing 3-18 shows skeleton XML for a Final Cut Pro project that has two bins and one sequence.

Listing 3-18 A project

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE xmeml>
<xmeml version="5">
  <project> // 1
    <name>Swing</name>
    <children> // 2
      <bin> // 3
        <name>Audio</name>
        <children> // 4
          <clip id="Jumptown.aif">
            <name>Jumptown.aif</name>
            . . .
          </clip>
        </children>
      </bin>
      <sequence id="Dance"> // 5
        <name>Dance</name>
        . . .
      </sequence>
      <bin> // 6
        <name>Video</name>
        <children>
          <clip id="Bloomies">
            <name>Bloomies</name>
            . . .
          </clip>
          <clip id="Jeremy Solo">
            <name>Jeremy Solo</name>
            . . .
          </clip>
          <clip id="Johnny n Cari">
```

```
        <name>Johnny n Cari</name>
        . . .
    </clip>
    <clip id="Overhead">
        <name>Overhead</name>
        . . .
    </clip>
    <clip id="Rob Dialog">
        <name>Rob Dialog</name>
        . . .
    </clip>
    <clip id="Round n Round">
        <name>Round n Round</name>
        . . .
    </clip>
</children>
</bin>
</children>
</project>
</xmml>
```

At line numbered 1, the `project` start-tag initiates the encoding for the project. The next line specifies the name of the project (Swing).

At lines numbered 2 and 4, the required `children` elements enclose elements in a project or bin.

At line numbered 3, the first bin (`Audio`) is encoded. It contains a single audio clip.

At line numbered 5, the sequence (`Dance`) is encoded. Note that a `children` subelement is not specified.

At line numbered 6, the second bin (`Video`) is encoded. It contains six clips.

Encoding Effects

Effects in Final Cut Pro are generally categorized as transition effects or filter effects. You encode an effect using either the `transitionitem` or `filter` element, and the `effect` subelement. Listing 3-17 at lines numbered 3 and 4 shows a skeleton example of a transition effect (a cross dissolve). Listing 3-19 shows the encoding for a filter effect (a gaussian blur).

Listing 3-19 A gaussian blur effect

```
<filter>
  <start>-1</start>
  <end>-1</end>
  <effect>
    <name>Gaussian Blur</name> // 1
    <effectid>Gaussian Blur</effectid> // 2
    <effectcategory>Blur</effectcategory>
    <effecttype>filter</effecttype> // 3
    <mediatype>video</mediatype> // 4
    <parameter>
      <parameterid>channel</parameterid> // 5
      <name>Channel</name>
      <valuemin>1</valuemin>
      <valuemax>7</valuemax>
      <valuelist> // 6
        <valueentry>
          <name>Alpha+RGB</name>
          <value>1</value>
        </valueentry>
        <valueentry>
          <name>Alpha</name>
          <value>2</value>
        </valueentry>
        <valueentry>
          <name>RGB</name>
          <value>3</value>
        </valueentry>
        <valueentry>
```

```
        <name>Red</name>
        <value>4</value>
    </valueentry>
    <valueentry>
        <name>Green</name>
        <value>5</value>
    </valueentry>
    <valueentry>
        <name>Blue</name>
        <value>6</value>
    </valueentry>
    <valueentry>
        <name>Luminance</name>
        <value>7</value>
    </valueentry>
</valuelist>
<value>1</value>                                // 7
</parameter>
<parameter>
    <parameterid>radius</parameterid>
    <name>Radius</name>
    <valuemin>0</valuemin>
    <valuemax>100</valuemax>
    <value>2</value>                                // 8
</parameter>
</effect>
</filter>
```

At lines numbered 1 and 2, either the element `name` or the element `effectid` is required to identify a particular effect. The element `effectid` is language independent. The element `name` may be localized, depending on the application generating the document. (When parsing installed effects, applications should look first at `effectid` and use `name` as a fallback if no matching effect is found.)

At lines numbered 3 and 4, both elements `effecttype` and `mediatype` are required in order to provide the application parsing the document with enough information to correctly identify the effect, or to map it to a reasonable substitute. (The `mediatype` element also allows inherited effects to be propagated to the correct type of track or clip.)

The gaussian blur effect has two parameters: `channel` and `radius`. Both parameters use `parameterid` and `name`. The element `parameterid` is language independent, while `name` can be localized.

At line numbered 5, the `channel` parameter is a pop-up list with seven choices. At line numbered 6, the `valuelist` element encodes the names and ordinal value of each of these choices. The encoding for the actual selection from the list occurs at line numbered 7, where `value` is set to 1. Alternatively, you can use the name of the pop-up list choice for the value element (`<value>Alpha+RGB</value>`).

Important: The `valuelist` element is not required for import. When you export an effect with a pop-up list parameter, `valuelist` provides a convenient summary of the pop-up options. In general, this document does not provide detailed information about effects and their parameters in the Final Cut Pro XML Interchange Format. To see the interchange format details for an effect, you can apply the effect to a clip and then export that clip in XML.

Keyframes

At line numbered 8 in Listing 3-19 the `radius` parameter is encoded with a static value of 2. Suppose, however, that you want to use keyframes to modify the effect over time. (A keyframe is a combination of a time value and optional curve or interpolation information.)

Listing 3-20 shows the `radius` parameter encoded with three `keyframe` elements, which replace the `value` element at line numbered 7 in Listing 3-19. To "smooth out" the behavior of the effect, you can add Bezier handles for the last two `keyframe` elements.

Listing 3-20 Radius parameter with keyframes and Bezier handles

```
<parameter>
  <parameterid>radius</parameterid>
  <name>Radius</name>
  <interpolation>                                     // 1
    <name>FCPCurve</name>
  </interpolation>
  <keyframe>                                           // 2
    <when>35</when>
    <value>5</value>
```

```
</keyframe>
<keyframe> // 3
  <when>50</when>
  <value>17</value>
  <inscale>0</inscale>
  <inbez>
    <horiz>9.5</horiz>
    <vert>3.14159</vert>
  </inbez>
  <outscale>0</outscale>
  <outbez>
    <horiz>9.5</horiz>
    <vert>0</vert>
  </outbez>
</keyframe>
<keyframe> // 4
  <when>88</when>
  <value>4</value>
  <inscale>0</inscale>
  <inbez>
    <horiz>9.5</horiz>
    <vert>-3.14159</vert>
  </inbez>
</keyframe>
</parameter>
```

At line numbered 1, the `interpolation` element identifies the interpolation method for the keyframes. Since this clip is for use in Final Cut Pro, the interpolation method is `FCPCurve`. See [Keyframe Interpolation](#) (page 161) for details of the Final Cut Pro interpolation method.

At line numbered 2, the encoding for the first keyframe sets `when` at 35 and `value` at 5.

At line numbered 3, the second keyframe includes information for a Bezier handle that is specific to the `FCPCurve` format. It consists of an offset (`inbez` and `outbez`) and a magnitude (`inscale` and `outscale`).

At line numbered 4, note that the last keyframe has a one-sided Bezier handle.

You can also set the range of an effect. By default, an effect uses the `in` and `out` values from its parent clip. The effect modifies the media within this range. You can alter this by encoding specific `start` and `end` points for `filter` or `transitionitem`.

XMEML Topics

This chapter details information about selected Final Cut Pro XML topics, including

- Notes on the XML Interchange Format
- Timing Values
- Managing Project Components
- Managing QuickTime Metadata
- Speed XML

Notes on the XML Interchange Format

Here are some additional details about the Interchange Format.

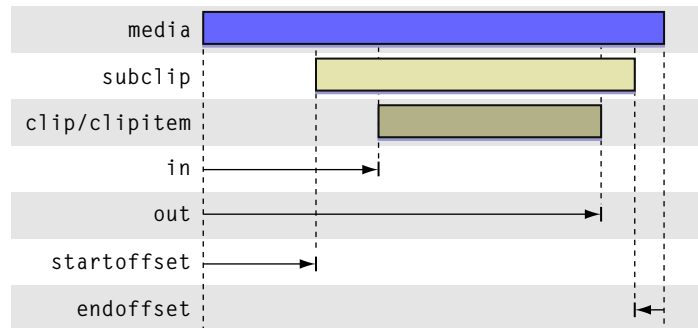
- You cannot encode user interface information such as the size and position of a window, the source and destination settings for a track, or the height of a sequence track in an XML interchange format document.
- Transitions or filters in the Browser are not included in an XML export.
- Custom names for columns in the Browser are not included in an XML export. Exporting a project with custom column names and then reimporting it causes loss of data.
- The drop/non-drop character of a timecode value is not included in an XML export.
- Third party effects and custom master templates are not exported with the XML. If you export a sequence using one and then import the sequence on a machine where the effect or template is not available, the system replaces it with a default effect or default master template.
- If you create a transition between two clips in a sequence, export the sequence, and then import it into the same project, the icon on the transition in the original sequence may lose its name.
- The volume level for the audio track of a clip is encoded by the `Audio Levels` effect. The parameter `Level` expresses linear gain rather than decibels. To convert gain to decibels, use the formula $\text{decibels} = 20 * \log_{10}(\text{Level})$. Conversely, to convert decibels to gain, use $\text{Level} = 10 ^ {(\text{decibels} / 20)}$.

For a summary description of changes to the Final Cut Pro XML Interchange Format and Final Cut, see [Versions of XMEML and Final Cut](#) (page 173).

Timing Values

Figure 4-1 illustrates the timing elements that encode the relation between clips, subclips and media.

Figure 4-1 Media Timing Values

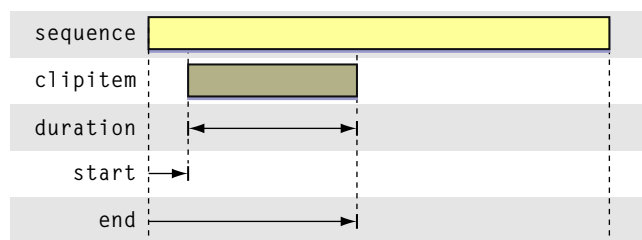


The elements `in` and `out` encode the initial and final points that specify the media content associated with a `clip` or `clipitem`. (In contrast, the elements `start` and `end` illustrated below specify the placement of a `clipitem` in a sequence.) The elements `startoffset` and `endoffset` encode the offset values specifying a `subclip`.

To convert `in` and `out` to timecode values, use the starting timecode in the media, if there is one. For media without timecodes, the default starting timecode is 00:00:00:00. For media with more than one timecode (such as media with `source`, `aux1`, `aux2`, `sound`), look for the `primarytimecode` subelement and use that value.

Figure 4-2 illustrates the elements that encode timing values for a `clipitem` in a sequence.

Figure 4-2 Sequence Timing Values



The elements `start` and `end` encode the start and end of a `clipitem` in an enclosing sequence. The element `duration` encodes the length of a `clipitem`. Note that the `duration` value for a `clipitem` may not match the `duration` value of the corresponding media, even when the `clipitem` and the media file are perfectly aligned. Applying speed to a `clipitem`, changing its framerate, or other edits can alter the `duration` of the `clipitem`.

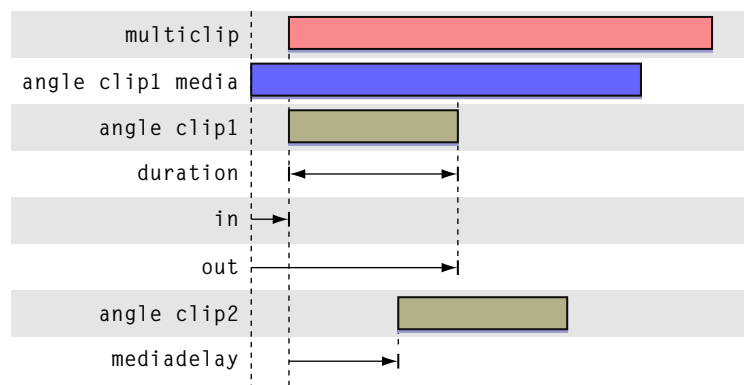
To convert `start` and `end` to timecode values, use the timecode in the containing sequence, or, for Browser clips, the timecode on the `file` of the containing `clip`.

Note: In Final Cut Pro, the timing values for the elements `out` and `end` are computed differently from the timing values for `start` and `in`.

To calculate or interpret the value for `start` and `in`, Final Cut numbers the first frame as frame 0. On the other hand, to calculate or interpret the value for `out` and `end`, Final Cut numbers the first frame as frame 1. In other words, Final Cut uses two distinct ordinal numbering schemes when calculating or interpreting the values for these elements

Figure 4-3 illustrates the relationships of timing elements for a multiclip and multiclip angles.

Figure 4-3 Multiclip Timing Values



The `angle` encoding for clip 1 shows that an angle clip can use `in` and `out` encoding to indicate which portion of the media should be used for the clip.

Syncing by `in` point (synctype 1) means that all the `in` points for each angle clip are lined up at the beginning of the multiclip. In that case no `mediadelay` is required.

The simplified `angle` encoding for clip 2 shows how you can line up angle clips in a multiclip. The second angle will be black from the beginning of the multiclip for the duration specified in `mediadelay`. Then the second angle clip is displayed. (Note that it doesn't make sense for the `mediadelay` to be negative since you can simply mark a later `in` on the clip to achieve the same result.)

A good way to understand these timing values is to create the desired multiclip alignment in Final Cut Pro, then modify something (like the `in` point for one clip) and make another multiclip, then compare the XML for the multiclips and see where they differ.

Speed and Special Keyframes

Final Cut encodes speed on a `clipitem` with an array of special keyframes.

The specification of a special keyframe includes the elements `when` and `value`, where `when` is the position of the keyframe in the `clipitem` and `value` is the corresponding position in the underlying media. The element `when` is always integral; `value` may be fractional, which indicates that it will be blended into the next frame.

The first keyframe always has a value of 0 for `when`, and the last keyframe always has the duration of the `clipitem` as the value for `when`.

Speed can be either constant or variable.

- If speed is constant, a graph of the `when`–`value` pairs is linear.
- If speed is variable, the values of the `when`–`value` pairs are arbitrary and the resulting graph is non-linear.

Special and Virtual Keyframes

Final Cut exports four special keyframes for a `clipitem` with speed. They are:

1. `speedkfstart`—the first keyframe. Not present if `speedkfend` is on the first frame.
2. `speedkfstart`—the keyframe at the in point.
3. `speedkfend`—the keyframe at the out point.
4. `speedkfend`—the last keyframe. Not present if `speedkfstart` is at the end.

At times, Final Cut writes out a speed keyframe in the XML that is not actually on the `clipitem`. These “virtual” keyframes are flagged with the Boolean element `speedvirtualkf` set to `TRUE`.

Final Cut ignores virtual keyframes on import unless they are on the first or last keyframe.

The `anchoroffset` Element

The `anchoroffset` element is new in version 5 of the Final Cut Pro XML Interchange Format (xmeme). It is an optional value that describes an offset from a keyframe for playback. Final Cut Pro uses it to preserve accuracy over speed changes.

For example, if you apply 200% speed to a `clipitem`, the frames would play back at: 0, 2, 4, ... But if `speedkfstart` is set at 3, playback would start from a frame that is no longer available. To adjust, Final Cut adds an `anchoroffset` of 1.0 so that the frames would play back at 1, 3, 5, 7, ... starting correctly from the frame specified by `speedkfstart`.

Keyframe Adjustments

For historical reasons, there are two modifications you should keep in mind when reading or writing special keyframes for speed.

- **Last Frame Adjustment**—Final Cut increments `when` and `value` in the last keyframe by 1.
- **Constant Reverse Adjustment**—For constant reverse keyframes, Final Cut increments all `value` settings by 1, except for `value` in the last keyframe if it has already been incremented by a Last Frame adjustment.

Importing Speed Keyframes

Final Cut interprets speed keyframes at import differently for variable and constant speed.

- For variable speed, Final Cut interprets speed keyframes as specified.
- For constant speed, Final Cut constructs its speed graph by:
 1. Reading the `value` specification of the first keyframe.
 2. Determining the number of frames used by reading the `value` specification of the last keyframe and subtracting `value` specification of the first keyframe.
 3. Setting the `duration` of the speed graph as the `when` specification of the last keyframe.
 4. Adjusting the speed graph as required to ensure that the graph's setting at the in point matches the `speedkfin` specification. Note that if a keyframe is flagged as virtual and is not the first or last keyframe, Final Cut ignores that keyframe.

Exporting Speed Keyframes

Final Cut writes keyframes for speed following these conventions:

- If a keyframe does not exist at `speedkfin` or `speedkfout`, Final Cut adds one and marks it as virtual.
- Final Cut applies the adjustments mentioned in the “Keyframe Adjustments” section above. Note these points:
 - The setting for `when` in the last keyframe must agree with the `duration` of the `clipitem`.
 - If speed is constant and if the keyframe settings do not produce a linear graph, Final Cut adjusts the settings to produce a linear graph, while maintaining the settings for `speedkfin`.

Identifying Media

One common reason for reading speed keyframes is to identify the media frames used at `speedkfin`, or at `speedkfout`, or between `speedkfin` and `speedkfout`.

- For the in point in the media, use the setting of `value` in the `speedkfin` keyframe.
- For the out point in the media, start with the setting of `value` in the `speedkfout` keyframe. Subtract from this the slope between the `speedkfout` keyframe and the preceding keyframe. This calculation is required because `speedkfout` uses 1-based ordering—see [Note](#) (page 50).
- For the extent of the media used when speed is constant, inspect the `value` settings at `speedkfin` and `speedkfout`; for variable speed, inspect the keyframes between `speedkfin` and `speedkfout`.

Managing Project Components

The Final Cut Pro XML Interchange Format lets you modify or update open projects using the elements `uuid` and `updatebehavior`. As well, you can use the `itemhistory` element to track the edit history of a `clipitem`.

The `uuid` and `updatebehavior` Elements

The `uuid` element uniquely and persistently identifies individual components (clips, bins, and sequences) in a project. Since it is saved with the project file, you can use it to track detailed changes during your workflow.

Important: A `uuid` entry is only guaranteed uniqueness on a per project basis. If you copy a clip from one project to another, the `uuid` entry for the clip remains the same. Likewise, if you export a clip and then import it into a different project, the `uuid` entry is unchanged. If you then modify the clip in one project, the clip in the other project has a matching UUID but is no longer identical. The UUID has ceased to be a unique identifier.

The `updatebehavior` element allows you to modify UUID-identified components in an open project. The target component in a project can be a clip, bin, or sequence. As well, you can use `updatebehavior` to modify metadata in a QuickTime file. (See [About master clips](#) (page 17))

A target component is identified by its `uuid` entry. For example, here is a XML snippet that replaces a clip in the Browser:

Listing 4-1 Replacing a Clip in the Browser

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE xmeml>
<xmeml version="5">
  <clip id="clip2 ">
    <uuid>7ACBD5FC-C166-40F9-AE36-55B39476D8E6</uuid>
    <updatebehavior>replaceiffound</updatebehavior>
```

```
<!--Information about the replacement clip -->
</clip>
```

When this snippet is imported, Final Cut Pro looks for a component in the project with a matching UUID. If it finds one, it replaces the component with the clip specified in the interchange format document. (In this case, the replacement clip appears at the topmost level of the Browser.) If it doesn't find the target clip, nothing happens.

To replace a clip in a bin, you have to specify the bin and all its contents. For example, Listing 4-2 replaces a clip in Bin 1.

Listing 4-2 Replacing a Clip in a Bin

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE xmeml>
<xmeml version="5">
  <bin>
    <uuid>4FDDECFA-D312-4272-8DE8-BE98E4BA5F92</uuid>
    <updatebehavior>replaceiffound</updatebehavior>
    <name>Bin 1</name>
    <children>
      <clip id="clip2 ">
        <uuid>7ACBD5FC-C166-40F9-AE36-55B39476D8E6</uuid>
        <updatebehavior>add</updatebehavior>
        <!-- Information about the replacement clip -->
      </clip>
    </children>
  </bin>
```

Similarly, to replace a particular `clipitem` in a sequence, you must replace the entire sequence.

The way you specify the location of the replacement component in the import document has precedence. For example, if the target clip is in Bin 1 and if you import a snippet such as Listing 4-1 Final Cut Pro removes the clip from Bin 1 and puts the replacement clip at the top level of the Browser.

The desired behavior at the time of import is specified by the entry for `updatebehavior`. The valid entries are:

replaceiffound Replaces the target component if found. If not found, does nothing.

replaceoradd	Replaces the target component if found. If not found, adds the imported component to the project.
addifnotfound	Adds the imported component to the project if the target component is not found. If the target component is found and if the target component is a bin, the attributes of the bin are ignored and the children of the bin are processed according to their <code>updatebehavior</code> settings.
add	Adds the imported component to the project. If a component with the same <code>uuid</code> entry is present, changes the <code>uuid</code> entry of the imported object.
remove	Removes the target object if found. If not found, does nothing.

Note: Only add is supported for PSD files or multiclips.

The `itemhistory` Element

The first time you export the XML for a `clipitem`, Final Cut creates a UUID for the item. If you modify the `clipitem` and export it again, Final Cut generates a new UUID for the item. The element `itemhistory` encodes a list of the UUIDs associated with a particular `clipitem`. The top UUID in the list is the current UUID for the item and is guaranteed to be unique. The rest of the list is a historical record of the UUIDs associated with the item. (The bottom UUID item in the list is the initial UUID generated for the `clipitem` or the UUID of the original item from which the current item was derived.)

For example, suppose you export a `clipitem` for the first time. It is assigned an initial UUID:

```
<itemhistory>
  <uuid>3510A3C8-5C79-4AA4-9A81-38B40397C0DC </uuid>
</itemhistory>
```

You then blade the `clipitem` and export both items. The XML for the first item is:

```
<itemhistory>
  <uuid>CC70F2BC-1777-4FF6-8846-58A8A3D2DA0E</uuid>
  <uuid>3510A3C8-5C79-4AA4-9A81-38B40397C0DC</uuid>
</itemhistory>
```

The XML for the second item is:

```
<itemhistory>
  <uuid>DD30375E-5CF3-4EB7-B209-36624768C4BB</uuid>
```

```
<uuid>3510A3C8-5C79-4AA4-9A81-38B40397C0DC</uuid>  
</itemhistory>
```

The common final UUID for the two items shows that they have a common origin.

Managing QuickTime Metadata

Final Cut Pro 5.1.2 supports metadata in QuickTime files. When Final Cut Pro opens or re-reads a QuickTime file with metadata, it caches a copy of the metadata in the project. (The file is the authoritative source of the metadata; if the file changes, the cached copy changes.)

To manage QuickTime metadata, version 3 of the Final Cut Pro XML Interchange Format provides the element `metadata` and its subelements. (See [QuickTime Metadata](#) (page 109)) With these new elements, you can display the metadata in a project by exporting an interchange format document. In addition, you can actually modify the metadata in a QuickTime file by importing an interchange format document with appropriately specified elements.

In Final Cut Pro, QuickTime metadata appears at the movie or track levels; that is, the parents of `metadata` can be `file`, `video`, or `audio`. Here is an XML fragment that displays some metadata at the movie level.

Listing 4-3 Sample Metadata

```
<file id="clouds">  
  <name>clouds</name>  
  <pathurl>file://localhost/Users/. . .</pathurl>  
  <rate>  
    <timebase>8</timebase>  
  </rate>  
  <duration>91</duration>  
  <metadata>  
    <storage>QuickTime</storage>  
    <key>com.mycompany.myapp.myid.mymd</key>  
    <size>11</size>  
    <type>UTF8</type>  
    <value>My metadata</value>  
  </metadata>
```



```
<media>  
.  
.  
.
```

When you import interchange format documents with metadata information, Final Cut Pro caches the imported metadata. However, when Final Cut Pro reconnects referenced QuickTime files to the project, the metadata in these files replaces the metadata you imported.

You can permanently alter the metadata in a QuickTime file by using the element `updatebehavior` as a subelement of `metadata`. In this context, `updatebehavior` has two valid entries: `add` or `remove`. Here is an XML fragment using `add`.

Listing 4-4 Adding Metadata

```
<metadata>  
  <updatebehavior>add</updatebehavior>  
  <storage>QuickTime</storage>  
  <key>com.mycompany.myapp.myid.mymd</key>  
  <size>22</size>  
  <type>UTF8</type>  
  <value>My additional metadata</value>  
</metadata>
```

Metadata in QuickTime files is stored as a key-value pair. In Listing 4-4 the value and the key are added to the metadata in the file. If the file already has an identical key, the imported value does not replace any existing values. Instead, the new value is stored alongside the existing values for that key. Note that a key is specified with a reverse-dns string to help ensure uniqueness.

Using `remove`, on the other hand, does affect existing metadata in a file. Here is an example:

Listing 4-5 Removing Metadata

```
<metadata>  
  <updatebehavior>remove</updatebehavior>  
  <storage>QuickTime</storage>  
  <key>com.mycompany.myapp.myid.mymd</key>  
</metadata>
```

This removes the specified key and all its associated values from the file. (Note that it is not necessary to specify the subelements `type`, `size`, or `value`.)

When you import an interchange format document with `metadata` specifications, Final Cut Pro first processes the specifications with `remove`, and then those with `add`. (This allows you to easily replace values for a particular key from a single import document.)

Applications of the Interchange Format

This chapter describes three tools based on the Final Cut Pro XML Interchange Format that developers might create to help streamline certain postproduction tasks. It includes these sections:

- [Enhancing a Batch List](#) (page 59) shows how a batch list can include extra features usually associated with a "paper edit."
- [Simplifying Subtitling](#) (page 64) demonstrates how the time-consuming subtitling process can be simplified and made significantly more efficient.
- [Choosing Effect Parameters](#) (page 70) illustrates how an editor can more easily choose appropriate parameters for an effect.
- [Other Possibilities](#) (page 77) lists some other ideas for applications of the interchange format.

Enhancing a Batch List

A filmmaker may generate tens or hundreds of hours of footage during production. Frequently, this footage is logged into a word processor, spreadsheet, or database program. The filmmaker may then use this information to create a batch list for import into Final Cut Pro.

Using the Final Cut Pro XML Interchange Format, a developer can create a simple tool that greatly enhances the capabilities of the traditional batch list, integrating into it many of the editing decisions usually expressed in a paper edit. For example, the tool can translate information from a spreadsheet file into an interchange format document that represents separated bins of clips, predefined sequences containing selected categories of clips, and even clips with effects already applied.

Listing 5-1 shows a brief example of an enhanced batch list in the interchange format. In this example, there is a single bin with five offline clips. The logging information entered for each clip in the spreadsheet has also allowed the tool (at the user's request) to set up a sequence of all the clips of a couple dancing.

Listing 5-1 An enhanced batch list

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE xmeml>
<xmeml version="5">
```

```
<project>
  <name>EnhancedBatchList</name>
  <children>
    <bin>
      <name>Dance shots</name>
      <children>
        <clip id="Bloomies">
          <name>Bloomies</name>
          <duration>248</duration>
          <rate>
            <ntsc>TRUE</ntsc>
            <timebase>30</timebase>
          </rate>
          <logginginfo>
            <lognote>couple</lognote>
          </logginginfo>
          <timecode>
            <string>02:12:02;17</string>
            <source>source</source>
            <reel>
              <name>002</name>
            </reel>
          </timecode>
        </clip>
        <clip id="Jeremy Solo">
          <name>Jeremy Solo</name>
          <duration>188</duration>
          <rate>
            <ntsc>TRUE</ntsc>
            <timebase>30</timebase>
          </rate>
          <logginginfo>
            <lognote>solo male</lognote>
            <good>FALSE</good>
```

```
</logginginfo>
<timecode>
  <string>02:42:35;14</string>
  <source>source</source>
  <reel>
    <name>002</name>
  </reel>
</timecode>
</clip>
<clip id="Johnny n Cari">
  <name>Johnny n Cari</name>
  <duration>148</duration>
  <rate>
    <ntsc>TRUE</ntsc>
    <timebase>30</timebase>
  </rate>
  <logginginfo>
    <lognote>couple</lognote>
    <good>FALSE</good>
  </logginginfo>
  <timecode>
    <string>02:41:42;10</string>
    <source>source</source>
    <reel>
      <name>002</name>
    </reel>
  </timecode>
</clip>
<clip id="Overhead">
  <name>Overhead</name>
  <duration>321</duration>
  <rate>
    <ntsc>TRUE</ntsc>
    <timebase>30</timebase>
  </rate>
```

```
<logginginfo>
  <lognote>group</lognote>
  <good>FALSE</good>
</logginginfo>
<timecode>
  <string>02:39:26;28</string>
  <source>source</source>
  <reel>
    <name>002</name>
  </reel>
</timecode>
</clip>
<clip id="Round n Round">
  <name>Round n Round</name>
  <duration>308</duration>
  <rate>
    <ntsc>TRUE</ntsc>
    <timebase>30</timebase>
  </rate>
  <logginginfo>
    <lognote>couple</lognote>
    <good>FALSE</good>
  </logginginfo>
  <timecode>
    <string>02:12:18;07</string>
    <source>source</source>
    <reel>
      <name>002</name>
    </reel>
  </timecode>
</clip>
</children>
</bin>
<sequence id="Couples">
```

```
<name>Couples</name>
<duration>704</duration>
<rate>
  <ntsc>TRUE</ntsc>
  <timebase>30</timebase>
</rate>
<timecode>
  <rate>
    <ntsc>TRUE</ntsc>
    <timebase>30</timebase>
  </rate>
  <string>01:00:00;00</string>
  <frame>107892</frame>
  <source>source</source>
  <displayformat>DF</displayformat>
</timecode>
<media>
  <video>
    <track>
      <clipitem>
        <name>Bloomies</name>
        <start>0</start>
        <end>248</end>
        <file id="Bloomies1"/>
      </clipitem>
      <clipitem>
        <name>Johnny n Cari</name>
        <start>248</start>
        <end>396</end>
        <file id="Johnny n Cari"/>
      </clipitem>
      <clipitem>
        <name>Round n Round</name>
        <start>396</start>
        <end>704</end>
```

```
        <file id="Round n Round"/>
    </clipitem>
</track>
</video>
</media>
</sequence>
</children>
</project>
</xmml>
```

Simplifying Subtitling

Using the capabilities of the Final Cut Pro XML Interchange Format and text generators, a developer can create a tool or suite of tools to help simplify and streamline the subtitling process.

First, an editor completes the offline edit without subtitles. Then the editor adds a new slug video track and creates through edits at the points where the subtitling should change. For those parts of the video where no subtitling should appear, the editor deletes the matching portion of the slug track.

The editor then exports the sequence in the interchange format. The slug track in the resulting document contains the start and end information required to correctly specify the start and end of a text generator for each subtitle.

In addition, the editor puts text for the subtitles in a database or spreadsheet.

The subtitling tool now reads the exported file for the start and end information, and the spreadsheet or database for the actual subtitle text. It then generates a new interchange format document that substitutes a text generator for each section of the slug track, providing the appropriate subtitle text for each generator.

The editor then imports this new interchange format file and make any necessary refinements or adjustments.

Listing 5-2 shows an abbreviated example of a project with two subtitles.

At line numbered 1, the first subtitle generator effect uses the `id` attribute. The first generator effect in the track can set all the parameters, such as `fontname`, `fontsize`, and `origin`, and so on.

At line numbered 2, the next generator effect in the track can reference this `id` and use the same settings, overriding only the `str` parameter to change the subtitle text. This means you can change parameters, such as the font or text style, in the first generator effect and have the changes apply to all the subtitles in the sequence. (At line numbered 3, the encoding details for the audio tracks are omitted.)

Listing 5-2 Subtitling

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE xmeml>
<xmeml version="5">
<sequence id="SampleSubtitles">
  <name>SampleSubtitles</name>
  <duration>219</duration>
  <rate>
    <ntsc>TRUE</ntsc>
    <timebase>30</timebase>
  </rate>
  <timecode>
    <rate>
      <ntsc>TRUE</ntsc>
      <timebase>30</timebase>
    </rate>
    <string>01:00:00;00</string>
    <frame>107892</frame>
    <source>source</source>
    <displayformat>DF</displayformat>
  </timecode>
  <media>
    <video>
      <track>
        <clipitem>
          <name>Rob Dialog</name>
          <duration>751</duration>
          <rate>
            <ntsc>TRUE</ntsc>
            <timebase>30</timebase>
          </rate>
          <in>0</in>
          <out>219</out>
          <start>0</start>
          <end>219</end>
```

```
<file id="Rob Dialog1">
  <name>Rob Dialog</name>

<pathurl>file://localhost/NTSC%20TutorialDance%20Shots/Rob%20Dialog</pathurl>
  <media>
    <video/>
  </media>
</file>
</clipitem>
</track>
<track>
  <generatoritem>
    <name>Text</name>
    <duration>3600</duration>
    <rate>
      <ntsc>TRUE</ntsc>
      <timebase>30</timebase>
    </rate>
    <in>1650</in>
    <out>1784</out>
    <start>0</start>
    <end>134</end>
    <enabled>TRUE</enabled>
    <anamorphic>FALSE</anamorphic>
    <alphatype>black</alphatype>
    <effect id = "subtitle">                                     // 1
      <name>Text</name>
      <effectid>Text</effectid>
      <effectcategory>Text</effectcategory>
      <effecttype>generator</effecttype>
      <mediatype>video</mediatype>
      <parameter>
        <parameterid>str</parameterid>
        <name>Text</name>
```

<value>If you look at the Lindy Hopyou'll see a
couple just moving</value>

</parameter>

<parameter>

<parameterid>fontname</parameterid>

<name>Font</name>

<value>Futura</value>

</parameter>

<parameter>

<parameterid>fontsize</parameterid>

<name>Size</name>

<valuemin>0</valuemin>

<valuemax>1000</valuemax>

<value>36</value></parameter>

<parameter>

<parameterid>fontstyle</parameterid>

<name>Style</name>

<valuemin>1</valuemin>

<valuemax>4</valuemax>

<valuelist>

<valueentry>

<name>Plain</name>

<value>1</value>

</valueentry>

<valueentry>

<name>Bold</name>

<value>2</value>

</valueentry>

<valueentry>

<name>Italic</name>

<value>3</value>

</valueentry>

<valueentry>

<name>Bold/Italic</name>

<value>4</value>

```
        </valueentry>
    </valuelist>
    <value>3</value>
</parameter>
<parameter>
<parameterid>fontalign</parameterid>
<name>Alignment</name>
<valuemin>1</valuemin>
<valuemax>3</valuemax>
<valuelist>
    <valueentry>
        <name>Left</name>
        <value>1</value>
    </valueentry>
    <valueentry>
        <name>Center</name>
        <value>2</value>
    </valueentry>
    <valueentry><name>Right</name>
        <value>3</value>
    </valueentry>
</valuelist>
<value>2</value>
</parameter>
<parameter>
    <parameterid>fontcolor</parameterid>
    <name>Font Color</name>
    <value>
        <alpha>255</alpha>
        <red>255</red>
        <green>255</green>
        <blue>255</blue>
    </value>
</parameter>
```

```
        <parameter>
        <parameterid>origin</parameterid>
        <name>Origin</name>
        <value>
        <horiz>0</horiz>
        <vert>0.34375</vert>
        </value>
        </parameter>
    </effect>
    <sourcetrack>
        <mediatype>video</mediatype>
    </sourcetrack>
</generatoritem>
<generatoritem>
    <start>135</start>
    <end>219</end>
    <effect id ="subtitle2"> // 2
        <name>Text</name>
        <effectid>Text</effectid>
        <effectcategory>Text</effectcategory>
        <effecttype>generator</effecttype>
        <mediatype>video</mediatype>
        <parameter>
            <parameterid>str</parameterid>
            <name>Text</name><value>They're constantly
moving&#13;they're always on the go</value>
        </parameter>
        </effect>
    </generatoritem>
    <enabled>TRUE</enabled>
    <locked>FALSE</locked>
</track>
</video>
<audio> // 3
    . . .
```

```
</audio>
</media>
</sequence>
</xmml>
```

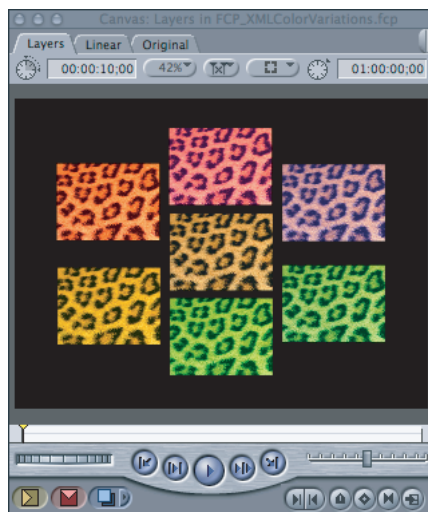
Choosing Effect Parameters

Many effects have multiple parameters that offer a wide range of choices. Finding the right combination of settings for a desired effect can be a time-consuming task. To help an editor more quickly identify appropriate parameter settings, a developer can create a "parameter exploration" tool that quickly displays a range of alternative treatments for a clip in the Canvas.

The tool asks an editor to choose a clip and an effect, and then to specify the parameters to be varied and the increments for the variations. The tool then generates an interchange format document for import into Final Cut Pro.

Figure 5-1 shows what the tool might produce. In this case, the clip is a still frame and the applied effect is Color Corrector 3-way. The original clip (in the center) and six variations are displayed in the Canvas.

Figure 5-1 Multiple parameter variations in the Canvas



Listing 5-3 displays part of the interchange format document created by the tool. It shows two of the seven tracks. In particular, note these points:

At line numbered 1, the `stillframe` element indicates that the clip is a still frame.

At line numbered 2, `duration` is set to 2. (Note that for still frames, the value for `duration` indicates time in minutes, rather than the number of frames.)

Lines numbered 3 through 8 are the parameters in the effect that vary from clip to clip. In the second track, they change at lines numbered 11 through 16.

At line numbered 9, the clip is scaled to fit with the other clips in the Canvas window.

At line numbered 10, the horizontal and vertical settings for the Center parameter control the position of the clip in the Canvas. These values are changed for the second clip at line numbered 17.

Listing 5-3 Effect parameter settings

```
<track> <!-- the first track -->
  <clipitem>
    <name>Faux Fur.jpg</name>
    <duration>3901</duration>
    <rate>
      <ntsc>TRUE</ntsc>
      <timebase>30</timebase>
    </rate>
    <in>1800</in>
    <out>2100</out>
    <start>0</start>
    <end>300</end>
    <stillframe>TRUE</stillframe>                                     // 1
    . . .
    <file id="Faux Fur.jpg1">
      <name>Faux Fur.jpg</name>
      <pathurl>file://localhost/Faux%20Fur.jpg</pathurl>
      <duration>2</duration>
      <media>
        <video>
          <duration>2</duration>                                       // 2
          <stillframe>TRUE</stillframe>
          <samplecharacteristics>
            <width>720</width>
            <height>480</height>
```

```
        </samplecharacteristics>
    </video>
</media>
</file>
<filter>
    <enabled>TRUE</enabled>
    <start>-1</start>
    <end>-1</end>
    <effect id = "colorcorrector3way">
        <name>Color Corrector 3-way</name>
        <effectid>Color Corrector 3-way</effectid>
        <effectcategory>Color Correction</effectcategory>
        <effecttype>filter</effecttype>
        <mediatype>video</mediatype>
        <parameter>
            <parameterid>dispmode</parameterid>
            <name>Display Mode</name>
            . . .
            <value>1</value>
        </parameter>
        <parameter>
            <parameterid>label1</parameterid>
            <name>Blacks Controls</name>
        </parameter>
        <parameter>
            <parameterid>blacklevel</parameterid>
            <name>Level</name>
            <valuemin>-196</valuemin>
            <valuemax>254</valuemax>
            <value>0</value>
        </parameter>
        <parameter>
            <parameterid>b_hue</parameterid>
            <name>Angle</name>
```



```
        <valuemin>-180</valuemin>
        <valuemax>180</valuemax>
        <value>-33</value>                                     // 3
    </parameter>
    <parameter>
        <parameterid>b_mag</parameterid>
        <name>Magnitude</name>
        <valuemin>0</valuemin>
        <valuemax>200</valuemax>
        <value>0</value>                                       // 4
    </parameter>
    <parameter>
        <parameterid>label2</parameterid>
        <name>Midrange Controls</name>
    </parameter>
    <parameter>
        <parameterid>mids</parameterid>
        <name>Level</name>
        <valuemin>0</valuemin>
        <valuemax>200</valuemax>
        <value>100</value>
    </parameter>
    <parameter>
        <parameterid>m_hue</parameterid>
        <name>Angle</name>
        <valuemin>-180</valuemin>
        <valuemax>180</valuemax>
        <value>-33</value>                                     // 5
    </parameter>
    <parameter>
        <parameterid>m_mag</parameterid>
        <name>Magnitude</name>
        <valuemin>0</valuemin>
        <valuemax>200</valuemax>
        <value>0</value>                                       // 6
```

```
</parameter>
<parameter>
  <parameterid>label3</parameterid>
  <name>Highlight Controls</name>
</parameter>
<parameter>
  <parameterid>highlights</parameterid>
  <name>Level</name>
  <valuemin>64</valuemin>
  <valuemax>509</valuemax>
  <value>255</value>
</parameter>
<parameter>
  <parameterid>h_hue</parameterid>
  <name>Angle</name>
  <valuemin>-180</valuemin>
  <valuemax>180</valuemax>
  <value>-33</value> // 7
</parameter>
<parameter>
  <parameterid>h_mag</parameterid>
  <name>Magnitude</name>
  <valuemin>0</valuemin>
  <valuemax>200</valuemax>
  <value>0</value> // 8
</parameter>
. . . <!-- other parameter settings are defaults -->
</effect>
</filter>
<filter>
  <effect id = "basicmotion">
    <name>Basic Motion</name>
    <effectid>basic</effectid>
    <effectcategory>motion</effectcategory>
```

```
<effecttype>motion</effecttype>
<mediatype>video</mediatype>
<parameter>
  <parameterid>scale</parameterid>
  <name>Scale</name>
  <valuemin>0</valuemin>
  <valuemax>1000</valuemax>
  <value>25</value> // 9
</parameter>
<parameter>
  <parameterid>center</parameterid>
  <name>Center</name>
  <value> // 10
    <horiz>-2.98023e-08</horiz>
    <vert>-0.00524935</vert>
  </value>
</parameter>
</effect>
</filter>
. . .
</clipitem>
<enabled>TRUE</enabled>
<locked>FALSE</locked>
</track>
<track> <!-- start of the second track -->
  <clipitem>
    . . .
    <filter>
      <enabled>TRUE</enabled>
      <start>-1</start>
      <end>-1</end>
      <effect id = "colorcorrector3way">
        . . .
        <parameter>
          <parameterid>b_hue</parameterid>
```

```
        <name>Angle</name>
        <valuemin>-180</valuemin>
        <valuemax>180</valuemax>
        <value>-12</value> // 11
    </parameter>
    <parameter>
        <parameterid>b_mag</parameterid>
        <name>Magnitude</name>
        <valuemin>0</valuemin>
        <valuemax>200</valuemax>
        <value>20</value> // 12
    </parameter>
    <parameter>
        <parameterid>m_hue</parameterid>
        <name>Angle</name>
        <valuemin>-180</valuemin>
        <valuemax>180</valuemax>
        <value>-12</value> // 13
    </parameter>
    <parameter>
        <parameterid>m_mag</parameterid>
        <name>Magnitude</name>
        <valuemin>0</valuemin>
        <valuemax>200</valuemax>
        <value>20</value> // 14
    </parameter>

    <parameter>
        <parameterid>h_hue</parameterid>
        <name>Angle</name>
        <valuemin>-180</valuemin>
        <valuemax>180</valuemax>
        <value>-12</value> // 15
    </parameter>
```

```
        <parameter>
            <parameterid>h_mag</parameterid>
            <name>Magnitude</name>
            <valuemin>0</valuemin>
            <valuemax>200</valuemax>
            <value>20</value>                                // 16
        </parameter>
    </effect>
</filter>
<filter>
    <effect id = "basicmotion">
        . . .
        <parameter>
            <parameterid>center</parameterid>
            <name>Center</name>
            <value>                                // 17
                <horiz>--0.275591</horiz>
                <vert>-0.167979</vert>
            </value>
        </parameter>
    </effect>
</filter>
    . . .
</clipitem>
<enabled>TRUE</enabled>
<locked>FALSE</locked>
</track>
```

Other Possibilities

Here are some other possible projects and tools based on the Final Cut Pro XML Interchange Format:

- Production companies often use a database program to track all film or video shots during production. Exporting this database information to interchange format documents could allow for the quick creation of video dailies from film, HD, or SD footage. The interchange format document could edit together footage into a sequence with text generator slates before each shot.

- A production house could archive all clips ever used in Final Cut Pro projects in one central database by exporting each project as an interchange format document and then translating the clip information in that document into a central database.
- A production house could maintain version-tracking of sequences. For example, a sequence could be exported and the interchange format document stored in a central repository. Later, if an editor wants to see a previous version of a sequence, she can retrieve the earlier version from the repository and import it into Final Cut Pro.
- An editor could easily find and replace clips with other clips. For example, an editor might want to change all medium shots of scene 10 to close ups.
- A tool could automatically generate Final Cut Pro sequences with foreground and background plates in place, and the appropriate chromakey effect already applied to the foreground elements. This could save an editor significant time.
- A tool could randomly sort the clips in a sequence, instead of the pseudo-random editing done by hand. For example, starting with 100 different clips, the tool could generate 100 sequences, each with the clips in a different random order.

Elements Catalog

This chapter describes all the elements of the Final Cut Pro XML Interchange Format.

Important: This chapter uses two descriptive conventions. The plus (+) sign precedes a subelement that is required. An asterisk (*) precedes a subelement that cannot be inherited and must be explicitly encoded.

Information about the elements of the Final Cut Pro Interchange Format is presented in the following sections:

- [Major Elements](#) (page 79)
- [Clips](#) (page 82)
- [Video and Audio](#) (page 90)
- [Common Elements](#) (page 94)
- [Rate and Timecode](#) (page 96)
- [Effects](#) (page 98)
- [The Time Remap Effect](#) (page 103)
- [Application Specific Data](#) (page 104)
- [Film Data](#) (page 107)
- [QuickTime Metadata](#) (page 109)
- [Import Options](#) (page 110)

Major Elements

xmeml

Description Root element for an XML Interchange Format document.

Attribute version (for example: `version = "5"`)

Subelements `importoptions`, `project`, `bin`, `clip`, `sequence`

Notes You can encode multiple `xmeml` elements in a single document, as long as there is an `xmeml` document type declaration at the top of the file.

project

Description Encodes a Final Cut Pro project.

Parent xmeml

Subelements +name, +children

bin

Description Encodes a bin, a named container within a project.

Parents xmeml, children

Subelements +*name, +*children, *labels, *comments, uuid, updatebehavior

children

Description Encloses a collection of elements within a bin or project.

Parents bin, project

Subelements clip, sequence, bin

sequence

Description Encodes a collection of clips and generators sequenced in relation to each other by time, layer, and position.

Parents xmeml, children, clipitem

Subelements +name, +duration, +rate, in, out, *timecode, media, *marker, sequence, *labels, *comment, *masterclipid, *ismasterclip, *descriptionlogginginfo, filmdata, file, pixelaspectratio, uuid, updatebehavior

Attribute id

media

Description Encodes specific media tracks for a clip or a sequence.

Parents clip, sequence, file

Subelements *video, *audio

track

Description Encodes data specific to one or more video or audio elements for a track.

Parents video, audio

Subelements *clipitem, *generatoritem, *transitionitem, enabled, locked, *outputchannelindex

locked

Description	A Boolean value specifying whether or not a track is locked.
Parent	track
Notes	If TRUE, the parent track is locked and cannot be modified in the Timeline. The default is FALSE.

outputchannelindex

Description	An integer specifying the output channel index for a track.
Parent	track

link

Description	Encodes a link between different clips in a sequence.
Parent	clipitem
Subelement	+linkclipref or (+mediatype, +trackindex, +clipindex, +groupindex)

linkclipref

Description	A string specifying a clip <code>id</code> attribute that encodes a link between sequence items sharing the same clip reference.
Parent	link
Notes	You cannot use <code>linkclipref</code> with multiple clips in a track with the same <code>id</code> .

clipindex

Description	An integer specifying the clip index.
Parent	link

groupindex

Description	An integer specifying the index of a group of audio outputs.
Parent	link
Notes	A <code>groupindex</code> element specifies a subset in an audio <code>link</code> array. (A video <code>link</code> should not have a <code>groupindex</code> subelement.) Items sharing a <code>groupindex</code> are part of a stereo pair. The <code>groupindex</code> should never exceed the number of audio items specified in the <code>link</code> array.

uuid

Description	A unique identifier.
Parents	clip, sequence, bin, itemhistory
Notes	Valid entries are in the standard UUID format. For the hex characters A through F, case is ignored. Note that UUID's for clips, sequences, and bins are guaranteed to be unique only within a single

project. If you copy or import a clip from one project to another, the `uuid` entry remains the same. Note also that the `UUID` element, which is a subelement of `data`, is distinct from `uuid`.

updatebehavior

Description	Allows an imported interchange format document to modify an open project or metadata in a QuickTime file.
Parents	<code>clip</code> , <code>sequence</code> , <code>bin</code> , <code>metadata</code>
Notes	Valid entries are <code>replaceiffound</code> , <code>replaceoradd</code> , <code>addifnotfound</code> , <code>add</code> , <code>remove</code> . (Only <code>add</code> is a valid entry for PSD files and multiclips.) See Managing Project Components (page 53) The default entry is <code>add</code> . If you use <code>updatebehavior</code> and <code>add</code> with a clip, and if there is clip with the same UUID already in the project, then Final Cut assigns a new UUID to the clip you're importing.

itemhistory

Description	A list of UUIDs associated with a <code>clipitem</code> .
Parent	<code>clipitem</code>
Subelements	<code>uuid</code>
Notes	Final Cut generates a UUID for a <code>clipitem</code> when the item is first exported, and subsequently whenever the item is modified and exported again. The top UUID in the list is the current UUID for the item and is guaranteed to be unique. The rest of the list is a historical record of the UUIDs associated with the item. (The bottom UUID item in the list is the initial UUID generated for the <code>clipitem</code> or the UUID of the original item from which the current item was derived.) See itemhistory (page 82)

Clips

clip

Description	Encodes a clip in the Browser.
Parents	<code>xmml</code> , <code>children</code>
Subelements	<code>+name</code> , <code>+duration</code> , <code>+rate</code> , <code>in</code> , <code>out</code> , <code>*masterclipid</code> , <code>*subclipmasterid</code> , <code>*ismasterclip</code> , <code>*logginginfo</code> , <code>*enabled</code> , <code>*file</code> , <code>*timecode</code> , <code>*media</code> , <code>marker</code> , <code>*anamorphic</code> , <code>*alphatype</code> , <code>*alphareverse</code> , <code>*labels</code> , <code>*comments</code> , <code>sourcetrack</code> , <code>*compositemode</code> , <code>subclipinfo</code> , <code>filter</code> , <code>stillframe</code> , <code>*stillframeoffset</code> , <code>*startoffset</code> , <code>*endoffset</code> , <code>defaultangle</code> , <code>subframeoffset</code> , <code>start</code> , <code>end</code> , <code>pixelaspectratio</code> , <code>filmdata</code> , <code>uuid</code> , <code>updatebehavior</code>
Attribute	<code>id</code>

clipitem

Description Encodes a clip in a track.

Parent track

Subelements `+*name, +duration, +rate, +*start, +*end, link, syncoffset, *enabled, *in, *out, *masterclipid, *subclipmasterid, ismasterclip, *logginginfo, file, *timecode, *marker, *anamorphic, *alphatype, *alphareverse, *labels, *comments, sourcetrack, *compositemode, subclipinfo, *filter, stillframe, *stillframeoffset, *sequence, multiclip, mediadelay, subframeoffset, *mixedratesoffset, filmdata, pixelaspectratio, fielddominance, gamma, primarytimecode*itemhistory`

Attribute id

Notes Note that start, end, link, syncoffset, and enabled are subelements of clipitem, but not of clip.

file

Description Encodes a media file reference for use within a clip.

Parents clip, clipitem

Subelements `+duration, +rate, +*name or +pathurl (or both), *timecode, *media, width, height, mediaSource`

Attribute id

pathurl

Description A UTF-8 string specifying a pathname for a file.

Parent file

Notes A file referenced by pathurl must be on a local volume. The URL must start with `file://localhost` or `file:///`.

A character that is not a legal URL character according to the RFC 2396 standard needs to be escaped by writing it in hexadecimal notation preceded by the % character. (To process URL strings programmatically, you can use functions such as `CFURLCreateStringByAddingPercentEscapes` in the Carbon library.)

mediaSource

Description A string describing a media source (such as a slug) built in to Final Cut Pro.

Parent file

anamorphic

Description A Boolean value characterizing a piece of video used in a track.

Parents clip, clipitem, generatoritem, samplecharacteristics

Notes If TRUE, video has a 16:9 aspect ratio. If FALSE, video has a 4:3 aspect ratio.

alphatype

Description A string specifying the alpha mode for a clip.

Parents clip, clipitem, generatoritem, media, video

Notes Valid entries are none, straight, black, or white. Note that these entries are always in English, even for versions of Final Cut Pro in other languages.

alphareverse

Description A Boolean value specifying whether or not Reverse Alpha is set.

Parents clip, clipitem

compositemode

Description A string specifying the composite mode for a clip.

Parents clip, clipitem

Notes Valid entries are normal, add, subtract, difference, multiply, screen, texturize, hardlight, softlight, darken, lighten, mask, or lumamask. Note that these entries are always in English, even for versions of Final Cut Pro in other languages.

masterclipid

Description A string specifying the master clip id.

Parents clip, clipitem, sequence

Notes You do not have to encode a masterclipid element before you encode any affiliate clips. (In a sequence, all clips are affiliate or independent clips. Independent clips have no masterclipid element.)

subclipmasterid

Description A string specifying the id of the parent clip for a subclip.

Parents clip, clipitem, generatoritem

ismasterclip

Description A Boolean value specifying whether or not a clip is a master clip.

Parents clip, clipitem, sequence

Notes If ismasterclip is not encoded, the clip is assumed to be an affiliate or independent clip.

logginginfo

Description Encodes logging information for a clip.

Parents clip, clipitem

Subelements *description, *scene, *shottake, *lognote, *good, scenenote, shotnote, takenote

description

Description A string describing a clip or sequence.

Parents logginginfo, sequence

scene

Description A string specifying a scene identifier or name for a clip.

Parent logginginfo

shottake

Description A string specifying a shot or take identifier.

Parent logginginfo

lognote

Description A string specifying the logging note for a clip.

Parent logginginfo

good

Description A Boolean value specifying the state for a clip.

Parent logginginfo

scenenote

Description A string with scene information from the Cinema Tools database.

Parent logginginfo

shotnote

Description A string with shot information from the Cinema Tools database.

Parent logginginfo

takenote

Description A string with take information from the Cinema Tools database.

Parent logginginfo

labels

Description Encodes Label and Label 2 information for a clip.

Parents clip, clipitem, bin, sequence

Subelements *label, *label2

label

Description Specifies Label information for a clip.

Parent labels

Notes Since the user can modify label names in User Preferences, the system matches imported information for this element against the current Label settings in Final Cut Pro.

label2

Description A string specifying Label 2 information.

Parent labels

comments

Description Encodes comment information for a clip.

Parents clip, clipitem, bin, sequence

Subelements *mastercomment[n], *clipcommenta, *clipcommentb

Notes Four masterComment subelements, and two unique comment subelements are available.

mastercomment[n]

Description A string specifying a master comment.

Parent comments

Notes Valid elements are mastercomment1 through mastercomment4. The mastercomment[n] element specifies text stored inside a clip definition. This information is shared between master and affiliate clips and is kept consistent if modified inside Final Cut Pro.

clipcommenta, clipcommentb

Description A string specifying a unique clip comment.

Parent comments

Notes The elements clipcommenta and clipcommentb contain comment text stored inside a clip definition. This information is unique to each clip instance.

sourcetrack

Description	Encodes details of the media connected with a clip.
Parents	<code>clip</code> , <code>clipitem</code> , <code>generatoritem</code>
Subelements	<code>mediatype</code> , <code>trackindex</code>
Notes	The designated media can be a specific piece of media or a nested sequence with multiple types of media.

start

Description	An integer specifying the relative starting point for a clip in a track. (See Timing Values (page 49).)
Parents	<code>clipitem</code> , <code>generatoritem</code>
Notes	A value of <code>-1</code> indicates that a <code>clipitem</code> in a sequence is preceeded by a transition. The actual value of <code>start</code> is computed from the beginning of the transition.

end

Description	An integer specifying the relative ending point for a clip in a track. (See Timing Values (page 49).)
Parents	<code>clipitem</code> , <code>generatoritem</code>
Notes	A value of <code>-1</code> indicates that a <code>clipitem</code> in a sequence is followed by a transition. The actual value of <code>end</code> is computed from the end of the transition.

subclipinfo

Description	Encodes offset information for a subclip.
Parents	<code>clip</code> , <code>clipitem</code>
Subelements	<code>*startoffset</code> , <code>*endoffset</code>

startoffset

Description	An integer specifying the offset in frames from the start media value, calculated using the In point of the clip from which the subclip was created.
Parents	<code>subclipinfo</code> , <code>clip</code> , <code>clipitem</code>

endoffset

Description	An integer specifying the offset in frames from the end media value, calculated using the Out point of the clip from which the subclip was created.
Parents	<code>subclipinfo</code> , <code>clip</code> , <code>clipitem</code>

stillframe

Description	A Boolean value specifying whether or not a clip is a still frame.
Parents	clip, clipitem
Notes	Photoshop PSD files, jpeg files, tiff files, or other still images imported into Final Cut have stillframe set to TRUE.

stillframeoffset

Description	An integer specifying the offset in frames from the start media value to the frame to use as a freeze frame.
Parents	clip, clipitem

syncoffset

Description	An integer specifying the offset in frames from the beginning of the first track in a group of linked tracks that have been marked in sync.
Parents	clip, clipitem

multiclip

Description	Encodes a multiclip.
Parents	clipitem
Subelements	+name, +angle, collapsed, synctype

angle

Description	Encodes an angle is a multiclip.
Parents	multiclip
Subelements	activevideoangle, activeaudioangle, +clip

activevideoangle

Description	A Boolean value specifying if an angle is the active angle in a multiclip.
Parents	angle
Notes	Only one angle in a multiclip can be active at a time. Default setting is FALSE. If no angles are specified as active, the first angle is assumed to be the active angle.

activeaudioangle

Description	A Boolean value specifying if an angle is the active angle in a multiclip.
Parents	angle

Notes Only one angle in a multiclip can be active at a time. Default setting is FALSE. If no angles are specified as active, the first angle is assumed to be the active angle.

collapsed

Description A Boolean value specifying if a multiclip is collapsed or not.

Parents `multiclip`

Notes Default is FALSE.

synctype

Description An integer specifying the synchronizatrion for the multiclip.

Parents `multiclip`

Notes Valid entries are 1 (In point), 2 (Out point), 3 (timecode), 4 (aux 1 timecode), and 5 (aux 2 timecode).

mediadelay

Description Integer specifying the starting offset (in frames) for a `clipitem` used in a `multiclip` or `subclip`. (See [Figure 4-3](#) (page 50)).

Parents `clipitem`

defaultangle

Description A string identifying the angle of a clip in the Browser.

Parents `clip`

subframeoffset

Description Specifies the offset when a track is slipped by less than a frame.

Parents `clip`, `clipitem`

mixedratesoffset

Description An integer specifying an offset for footage played at mixed rates

Parent `clipitem`

Notes In points are stored in the frame rate of a sequence. Before the introduction of `mixedratesoffset` in version 6 of Final Cut, this could cause problems. For example, if you edited a 60fps clip with an in point of 3 into a 30fps sequence, the in point became 1 and Final Cut played frame 2 instead of frame 3. Final Cut 6, however, uses `mixedratesoffset` to handle these mixed rates situations. In this example, the value of `mixedratesoffset` would be set to +1 and the sequence would play correctly from frame 3. Note that `mixedratesoffset` is only available in version 4 of `xmeml`.

gamma

Description Number specifying the gamma setting for a still.

Parents clipitem

Notes Specifies a still gamma setting when a gamma setting is supported by the codec. The entry is a decimal with three digits of precision. The thousandths place is rounded up. For example, an entry of 1.2345 becomes 1.235. If gamma is unspecified, Final Cut Pro uses the source gamma setting.

Video and Audio

video

Description Encodes data specific to video media.

Parent media

Subelements track,*duration,format,*samplecharacteristics,in,out,stillframe,*layerindex,*alphatype,trackcount (version 1 only)

Notes Video data that you can encode for specific video tracks includes In and Out points, video format, and rate information.

layerindex

Description An integer specifying a layer in source media such as a PhotoShop PSD file.

Parent video

Notes Valid entries are from 1 to the number of layers in the media.

audio

Description Encodes data specific to audio media.

Parent media

Subelements *in,*out,track,format,outputs,*channelcount,channeldescription,rate,*samplecharacteristics,*duration,layout,audiochannel,trackcount (version 1 only)

Notes Audio data that you can encode for specific audio tracks include In and Out points and audio format information.

channelcount

Description An integer specifying the number of audio channels.

Parent audio

trackcount

Description For version 1 of the interchange format only. An integer specifying the number of tracks defined for the parent media type.

Parents video, audio

Notes The `trackcount` element is required when explicitly encoding the subelements of a `file` element where there is more than one track of a particular media type. It is optional for clips or sequences, where `trackcount` is implied by the number of `track` elements encoded.

channeldescription

Description A string specifying the audio channels.

Parent audio

Notes Valid entries are `mono`, `stereo`, `A1`, `A2`, or `A1+A2`.

layout

Description A string specifying the layout of an the audio channels.

Parent audio

Notes Valid entries are `mono`, and `stereo`. Default is `stereo`.

audiochannel

Description Encodes information about an audio channel.

Parent audio

Subelements `channellabel`, `sourcechannel`

channellabel

Description A string labelling the channel

Parent `audiochannel`

Notes Valid entries are `left`, `right`, and `discrete`.

sourcechannel

Description An integer specifying the source channel.

Parent `audiochannel`

Notes Valid entries are 1 or greater.

outputs

Description Encodes information about audio outputs.

Parent audio

Subelement *group

Notes The information corresponds to the outputs tab of the sequence settings.

group

Description Encodes information about a group of audio output channels.

Parent outputs

Subelements index, numchannels, downmix, channel

index

Description An integer specifying the index of a group or channel.

Parents group, channel

numchannels

Description An integer specifying the number of channels in a group.

Parent group

downmix

Description An integer specifying the amount of volume adjustment, in decibels, to apply to each channel when performing a downmix for the specified group.

Parent group

Notes Valid entries are +10, +6, +4, +2, 0, -2, -4, -6, or -10.

channel

Description Encodes the output device index of a channel in a group.

Parent group

Subelement index

format

Description Encodes format information for video or audio media in a track.

Parents video, audio

Subelements samplecharacteristics, appspecificdata

samplecharacteristics

Description Encodes characteristics of video or audio media.

Parents video, audio, format

Subelements Video: *width, *height, anamorphic, pixelaspectratio, fielddominance, rate, colordepth, codec
Audio: depth, samplerate

width

Description An integer specifying the width, in pixels, of video media.

Parent samplecharacteristics, file

height

Description An integer specifying the height, in pixels, of video media.

Parent samplecharacteristics, file

pixelaspectratio

Description A string specifying the pixel aspect ratio.

Parent samplecharacteristics

Notes Valid entries are NTSC-601, PAL-601, square, DVCPROHD-720P or HD-(960x720), DVCPROHD-1080i60 or HD-(1280x1080), DVCPROHD-1080i50 or HD-(1440x1080).

fielddominance

Description A string specifying the field dominance.

Parent samplecharacteristics

Notes Valid entries are none, lower, upper, odd, or even.

colordepth

Description An integer specifying the color depth.

Parent samplecharacteristics

Notes Valid entries are 8, 6, or 24.

codec

Description Encodes details about a codec.

Parent samplecharacteristics

Subelements name, appspecificdata

depth

Description An integer specifying the bit depth of audio media.

Parent samplecharacteristics

Notes Valid entries are 8 or 16.

samplerate

Description An integer specifying the audio sample rate.

Parent samplecharacteristics

Notes Valid entries are 32000, 44100, or 48000.

Common Elements

Common elements are subelements of more than one parent element.

name

Description A string specifying a name.

Parents project, bin, sequence, clip, clipitem, multiclip, generatoritem, file, marker, effect, interpolation, source, transitionitem

duration

Description An integer specifying duration in number of frames. (See [Timing Values](#) (page 49).)

Parents sequence, clip, clipitem, generatoritem, file, video, audio, transitionitem

Notes The number is interpreted based on the frame rate in effect at the time. Note that in the case of still frames, the value for duration represents minutes rather than frames.

enabled

Description A Boolean value specifying whether or not the parent element is enabled.

Parents track, clipitem, clip, generatoritem, sequence, filter

Notes If you do not specify enabled, the default setting is TRUE.

marker

Description Encodes a named time or range of time in a clip or sequence.

Parents clip, clipitem, sequence

Subelements +*name, +in, +out, *marker, *comment, color

color

Description Encodes the color value of a marker.

Parent marker

Subelements red, green, blue, or alpha.

comment

Description A string describing a marker.

Parent marker

in

Description An integer specifying a starting time. (See [Timing Values](#) (page 49).)

Parents sequence, clip, clipitem, generatoritem, marker, effect

Notes The valid range is from 0 to the duration of the component. The value must be less than the value of out. A value of -1 indicates that no in point is set for a browser clip or a sequence. Also, you can use -1 to override an existing encoding with an undefined or default value. Like duration, the value of in is interpreted based on the frame rate in effect at the time.

out

Description An integer specifying an ending time. (See [Timing Values](#) (page 49).)

Parents sequence, clip, clipitem, generatoritem, marker, effect

Notes The valid range is from 0 to the duration of the component. The value must be greater than the value of in. A value of -1 indicates that no out point is set for a browser clip or a sequence. Also, you can use -1 to override an existing encoding with an undefined or default value. Like duration, the value of out is interpreted based on the frame rate in effect at the time.

mediatype

Description A string defining the media type.

Parents link, sourcetrack, effect

Notes Valid strings are video or audio.

trackindex

Description An integer specifying the track index in a link element or in a media source.

Parents link, sourcetrack

Notes If the media has a single video track and no audio tracks, specifying trackindex may cause an error.

Rate and Timecode

rate

Description	Encodes a time scale to interpret time values for a clip, sequence, timecode, or other component.
Parents	sequence, clip, clipitem, generatoritem, timecode, audio, effect
Subelements	+timebase (for video: ntsc)

timebase

Description	An integer specifying the timebase of the frame rate.
Parent	rate
Notes	You use <code>timebase</code> in conjunction with <code>ntsc</code> to define the time scale. See Table B-1 (page 160)

ntsc

Description	A Boolean value specifying the NTSC rate reduction.
Parent	rate
Notes	If you set <code>ntsc</code> to <code>TRUE</code> , the NTSC rate reduction (0.01%) is applied to <code>timebase</code> . See Table B-1 (page 160)

timecode

Description	Encodes a timecode value for a clip, sequence, or file.
Parents	sequence, clip, clipitem, file, value, generatoritem
Subelements	+*string or (+frame and +displayformat), +rate, field, reel, source, format
Attribute	id
Notes	A timecode represents the address of a particular video frame on a time-based media storage device, such as a videotape recorder. Note that <code>timecode</code> requires a rate definition (local or inherited), and either a <code>string</code> subelement, or <code>frame</code> and <code>displayformat</code> subelements. (You can also specify all three subelements.) For example, a <code>string</code> set to <code>01:00:2;05</code> with a rate of 29.97 fps translates to a frame count of 107657 formatted using drop frame notation.

string

Description	A string specifying a timecode.
Parent	timecode
Notes	The value is based on an SMPTE timecode string, with a semicolon indicating that the drop frame format is being used. The application uses the current frame rate to interpret this value. (If you specify <code>string</code> , you do not need to specify <code>frame</code> and <code>displayformat</code> .)

frame

Description	An integer specifying the frame count.
Parent	timecode
Notes	This element defines a frame count that Final Cut Pro interprets into a timecode using the parent rate and format definitions. Note that <code>frame</code> contains an actual frame count, not simply a time index. For a drop frame timecode, the skipped numbers are taken into account. The frame count starts at zero, and rolls over at 23 hours, 59 minutes, 59 seconds, and <i>n</i> frames (where <i>n</i> is the frame rate).

displayformat

Description	A string specifying the format modifier to use for the parent <code>timecode</code> element.
Parent	timecode
Notes	Valid values are DF (for drop frame) or NDF (for non-drop frame) Note that values other than NDF are valid only for a timecode with a rate of 30 frames per second. If not specified, non-drop frame timecode formatting is used.

field

Description	An integer specifying the field offset for a timecode.
Parent	timecode
Notes	Valid values are 0 or 1, with a default to 0 if <code>field</code> is not specified.

reel

Description	Encodes the source identifier for a timecode.
Parent	timecode
Subelement	+*name
Notes	In common usage, <code>reel</code> identifies the source archive device or media (videotape) from which the timecode was taken.

source

Description	A string specifying either the timecode source or the track identifier.
Parent	timecode
Notes	Valid entries are <code>source</code> , <code>aux1</code> , <code>aux2</code> , or <code>sound</code> . If not specified, <code>source</code> is assumed.

Effects

Note: To see the interchange format and parameter details for an effect, you can apply the effect to a clip and then export the clip as an interchange format document.

generatoritem

Description	Encodes information about a generator in a track.
Parent	track
Subelement	+name,+duration,+rate,in,out,start,end,enabled,anamorphic,alphatype,effect,sourcetrack,filter,timecode,primarytimecode,alphareverse,link,logginginfo,labels,comments,mediadelay,fielddominance,filmdata,subframeoffset,clipid,compositemode,syncoffset,marker,subclipmasterID

primarytimecode

Description	A string specifying the default timecode to display for this item.
Parents	generatoritem,clipitem

transitionitem

Description	Encodes a transition in a track.
Parent	track
Subelements	rate,*start,*end,*alignment,effect,*name

alignment

Description	A string specifying an alignment for a transition.
Parent	transitionitem
Notes	Valid entries are start, center, end, end-black, or start-black.

filter

Description	Encodes a filter effect.
Parents	clip,clipitem
Subelements	*enabled,*start,*end,effect

effect

Description	Encodes an effect or processing operation.
Parents	transitionitem, filter,generatoritem

Subelements `+*name, +*effectid, +*effecttype, +*mediatype, *effectcategory, parameter, keyframe, appspecificdata, wipecode, wipeaccuracy, rate, startratio, endratio, reverse, duration, privatestate, multiclip, effectclass`

Notes You should encode an effect so that the importing application can accurately find the matching effect. If the effect is not found, the application may use a generic proxy effect.

effectid

Description A string providing a language-independent constant identifier for an effect.

Parent effect

effectcategory

Description A string specifying the category of an effect.

Parent effect

effecttype

Description A string specifying the type of an effect.

Parent effect

Notes Valid entries are `filter`, `transition`, `motion`, or `generator`.

effectclass

Description A string specifying an FxPlug effect.

Parent effect

Notes The only valid entry is `FxPlug`. For any other effects, you should leave this element unspecified.

wipecode

Description An integer specifying the SMPTE wipe code.

Parent effect

wipeaccuracy

Description An integer specifying the SMPTE wipe accuracy.

Parent effect

startratio

Description For a transition effect, a number specifying what fraction of the effect has been executed at the start of the transition.

Parent effect

Notes Valid entries in the range `0.0` to `1.0`. The default is `0.0`.

endratio

Description	For a transition effect, a number specifying what fraction of the effect will have been executed at the end of the transition.
Parent	effect
Notes	Valid entries in the range 0.0 to 1.0. The default is 1.0.

reverse

Description	For a transition effect, a Boolean value indicating whether or not an effect should be reversed, with the end result applied to the start and vice versa.
Parent	effect
Notes	The default is FALSE.

privatestate

Description	Data for an After Effects plug-in effect.
Parent	effect

parameter

Description	Encodes a parameter for an effect.
Parent	effect
Subelements	+name or +parameterid (or both), +value or +keyframe (but not both), parameterspecifier, valuemin, valuemax, valuelist, interpolation, appspecificdata, privatestate, parameterspecifier

parameterid

Description	A string providing a constant identifier for an effect parameter.
Parent	parameter
Notes	For an AudioUnits effect, you should use both <code>parameterid</code> and <code>parameterspecifier</code> .

parameterspecifier

Description	A string providing a constant identifier for an AudioUnit effect parameter.
Parent	parameter
Notes	The <code>parameterspecifier</code> element is only supported for AudioUnit effects. When combined with the <code>effectid</code> element, it unambiguously identifies a particular parameter for an AudioUnit effect. The value of <code>parameterspecifier</code> is a combination of the values <code>AudioUnitParameterID</code> , <code>AudioUnitScope</code> , and <code>AudioUnitElement</code> . It is packaged as a string with colon-separated decimal numeric values. For example, if an AudioUnit parameter has an ID of 42, a scope of

kAudioUnitScope_Global, and an element of 1, the encoding for the parameterspecifier element would be the string 42:0:1.

valuemin

Description A number specifying the minimum value for a parameter.

Parent parameter

valuemax

Description A number specifying the maximum value for a parameter.

Parent parameter

valuelist

Description Encodes information about a pop-up list in a parameter.

Parent parameter

Subelements valueentry

valueentry

Description Encodes information about the choice in a pop-up list in a parameter.

Parent valuelist

Subelements name, value

Notes The entry for value indicates the ordinal position of the item in the pop-up list.

value

Description Encodes or specifies a fixed value for an effect parameter or a keyframe.

Parents parameter, keyframe, valueentry, timecode, (metadata)

Subelements red, green, blue, alpha, horiz, vert, timecode, clip, sequence

Notes For some parameters, the subelements red, green, and blue are required. For others, the horiz and vert subelements are required. In some cases, you can directly specify a numeric or Boolean value in value and not use any subelements.

keyframe

Description Encodes a keyframe for an effect.

Parent parameter

Subelements +*when, +*value, interpolation, *inscale, *outscale, *inbez, *outbez, *speedvirtualkf, *speedkfin, *speedkfout, *speedkfstart, *speedkfend, *origvalue, *hadbezierin, *hadbezierout

when

Description An integer specifying the time location for a keyframe.

Parent keyframe

inscale

Description A number specifying the incoming magnitude value for a keyframe.

Parent keyframe

Notes The value is specific for keyframes using the FCPCurve interpolation.

inbez

Description Encodes the incoming handle value for a keyframe.

Parent keyframe

Subelements horiz, vert

outscale

Description A number specifying the outgoing magnitude value for a keyframe.

Parent keyframe

Notes The value is specific for keyframes using the FCPCurve interpolation.

outbez

Description Encodes the outgoing handle value for a keyframe.

Parent keyframe

Subelements horiz, vert

red, green, blue

Description A number specifying color information for an effect parameter.

Parents value, color

alpha

Description An integer specifying the alpha value of a channel.

Parents value, color

Notes Valid entries are from 0 to 255. 0 is transparent; 255 is opaque.

horiz, vert

Description A number specifying the horizontal or vertical coordinate. When these values specify the Center parameters in the Motion settings, they are scale values ranging from -100 to 100. This allows the size of the image to change at import time while still maintaining the positioning.

Parents value, inbez, outbez

Notes The value is specific for keyframes using the FCPCurve interpolation. When used as subelements of inbez and outbez, these elements represent polar coordinates. The horiz element is the radius and the vert element is the angle in radians.

interpolation

Description Encodes the type of curve interpretation and data to use in the parent element

Parents keyframe, parameter

Subelement +*name

Notes The interpolation element only occurs in an effect parameter with keyframes. For Final Cut Pro, the interpolation method is FCPCurve. See [Keyframe Interpolation](#) (page 161) for more information about this method.

The Time Remap Effect

The Time Remap effect allows you to set the constant speed, variable speed, and playback direction for a clipitem. When you use this effect, Final Cut automatically creates two “virtual” keyframes in the graphdict parameter at the beginning and end of the item. For keyframes in this parameter, value encodes the frame in the media file to be remapped to a time encoded by when. Virtual keyframes in the graphdict parameter also use these additional elements:

speedvirtualkf

Description A Boolean value indicating whether a keyframe has been created by Final Cut (TRUE) or by the user (FALSE).

Parent keyframe

speedkfin

Definition A Boolean value indicating whether a virtual keyframe is at the setting for in.

Parent keyframe

speedkfout

Definition A Boolean value indicating whether a virtual keyframe is at the the setting for out.

Parent keyframe

speedkfstart

Definition A Boolean value indicating whether a virtual keyframe is at the setting for start.

Parent keyframe

speedk fend

Definition A Boolean value indicating whether a virtual keyframe is at the setting for end.

Parent keyframe

origvalue

Definition An integer indicating the original value of the `value` element.

Parent keyframe

anchoroffset

An optional double value that specifies an offset from a keyframe for playback. See [The anchoroffset Element](#) (page 51).

Parent keyframe

hadbezierin

A Boolean value indicating whether keyframe once had an in Bezier handle.

Parent keyframe

Notes

Not required, not inherited.

hadbezierout

Definition A Boolean value indicating whether a virtual keyframe once had an out Bezier handle.

Parent keyframe

Application Specific Data

appspecificdata

Description Encodes application-specific data.

Parents format, codec, parameter

Subelements appname, appmanufacturer, appversion, data

appname

Description A string specifying the application name.

Parent appspecificdata

appmanufacturer

Description A string specifying a manufacturer.

Parent appspecificdata

appversion

Description A string specifying a version number.

Parent appspecificdata

data

Description Encodes data about an application or codec.

Parent appspecificdata

Subelements fcpimageprocessing, qtcodec , qteffectid, UUID

fcpimageprocessing

Description Encodes information about image processing.

Parent data

Subelements useyuv, usesuperwhite, rendermode

useyuv

Description A Boolean value specifying the use of YUV color space.

Parent fcpimageprocessing

usesuperwhite

Description A Boolean value specifying the use of super-white.

Parent fcpimageprocessing

rendermode

Description A string specifying the color space to be used for rendering.

Parent fcpimageprocessing

Notes Valid entries YUV8BPP, RGB, Float10BPP, or Float

qtcodec

Description Encodes QuickTime-specific codec identification and configuration information.

Parent data

Subelements codecname, codectypename, codetypecode, codecvendorcode, spatialquality, temporalquality, keyframerate, datarate

codecname

Description A string specifying the registered name for the codec.

Parent qtcodec

codectypename

Description A string specifying the registered short name (type name) for the codec.

Parent qtcodec

codetypecode

Description A string specifying the registered fourCC (4-byte) type code for the codec.

Parent qtcodec

codecvendorcode

Description A string specifying the registered fourCC (4-byte) vendor or manufacturer code for the codec.

Parent qtcodec

spatialquality

Description An integer specifying the spatial compression quality.

Parent qtcodec

Notes Valid entries are in the range 0 through 1023. This property may not be supported by a particular codec.

temporalquality

Description An integer specifying the temporal compression quality.

Parent qtcodec

Notes Valid entries are in the range 0 through 1023. This property may not be supported by a particular codec.

keyframerate

Description An integer specifying the keyframe rate for the codec.

Parent	qtcodec
Notes	A value of 0 indicates that keyframes should not be used. This property may not be supported by a particular codec.

datarate

Description	An integer specifying the data rate limit for a codec and QuickTime.
Parent	qtcodec
Notes	A value of 0 indicates that no data rate limit should be used by QuickTime. This property may not be supported by a particular codec.

qteffectid

Description	An integer specifying a QuickTime effect id.
Parent	data

UUID

Description	A string specifying a unique identifier for an application or codec.
Notes	The UUID element is distinct from the <code>uuid</code> element.
Parent	data

Film Data

filmdata

Description	Encodes metadata imported from Cinema Tools.
Parent	clip, clipitem
Subelements	appspecificdata, filmslate, cameraroll, labroll, keycode, dailyroll, inknumber, filmstandard, telecinespeed
Notes	You cannot enter or edit this data from Final Cut Pro, but you can import it.

filmslate

Description	Encodes information about the scene and take.
Parent	filmdata
Subelements	scene, take, slate
Notes	The Cinema Tools database has separate text field for Scene and Take, which appear concatenated as Slate in the application.

scene

Description A string describing the scene.

Parent `filmslate`

take

Description A string describing the take.

Parent `filmslate`

slate

Description A string specifying the Cinema Tools Slate information.

Parent `filmslate`

cameraroll

Description A string describing the camera roll.

Parent `filmdata`

labroll

Description A string describing the lab roll.

Parent `filmdata`

dailyroll

Description A string describing the daily roll.

Parent `filmdata`

keycode

Description Encodes information about the keycode for a film.

Parent `filmdata`

Subelements `+prefix, +feetframes, perfoffset, reversed`

prefix

Description Prefix information for the keycode.

Parent `keycode, inknumber`

Notes Appers in Browser concatenated with `feetframes`.

feetframes

Description A string representing feet (four digits) and frames (two digits).

Parent keycode, inknumber

Notes Appers in Browser concatenated with prefix.

perfoffset

Description A string describing the 3-perf offset.

Parent keycode

reversed

Description A Boolean value set to TRUE if the keycode runs backwards.

Parent keycode

inknumber

Description Encodes information about the keycode for a film.

Parent filmdata

Subelements +prefix, +feetframes

filmstandard

Description A string specifying a supported format: 35mm4p, 35mm3p, or 16mm20.

Parent filmdata

telecinespeed

Description A string specifying the telecine speed: 24, 25, or 30.

Parent filmdata

QuickTime Metadata

metadata

Description Encapsulates metadata in a QuickTime file.

Parents file, video, audio

Subelements +storage, +key, updatebehavior, size, type, value

Notes A metadata specification, regardless of its purpose, requires storage and key subelements. To add or remove metadata from a QuickTime file, use the updatebehavior subelement with add or remove. See [About master clips](#) (page 17)

storage

Description A string defining the storage container for a specified metadata.

Parent metadata

Notes The only valid entry is QuickTime.

key

Description A reverse-dns string identifying the metadata.

Parent metadata

Notes Reverse-dns strings (such as `com.mycompany.myid.md1`) help ensure uniqueness.

size

Description A number defining the size in bytes of the metadata.

Parent metadata

Notes While not absolutely required for strings, this specification is important for numbers and binary data blocks.

type

Description A string defining the type of metadata.

Parent metadata

Notes Valid entries are `binary`, `UTF8`, `UTF16`, `MacRoman`, `signed`, `unsigned`, `float32`, and `float64`.

value

Description The data stored as metadata in the QuickTime file.

Parents metadata, (parameter, keyframe, valueentry, timecode)

Notes All data values are stored as plain text, with the exception of binary data elements. These are uuencoded and wrapped in a CDATA scope.

Import Options

importoptions

Description Encodes import options specific to Final Cut Pro.

Parent xmeml

Subelements `*createnewproject`, `*targetprojectname`, `*defsequencepresetname`,
`*filterreconnectmediafiles`, `*filterincludemarkers`, `*filterincludeeffects`,
`*filterincludesequencesettings`,

```
*displaynonfatalerrors,*createfcpprojectatxmlfilepath,  
*deletethisxmlfileatimport
```

Notes Using `importoptions` allows you to bypass the Import XML dialog shown in [Figure 2-3](#) (page 11) and import an interchange format document directly into Final Cut Pro. (You must specify the subelements `createnewproject` and `defsequencepresetname` to do this.) The subelements specify the same choices as those in the dialog.

createnewproject

Description A Boolean value specifying whether or not to create a new project.

Parent `importoptions`

Notes If `TRUE`, the system creates a new project with the name "Untitled Project [n]."

targetprojectname

Description A string specifying the target project.

Parent `importoptions`

Notes The components specified in the interchange format document are imported into the target project.

defsequencepresetname

Description A string specifying the name of a sequence preset.

Parent `importoptions`

Notes For a list of valid entries, see the pop-up list on the XML Import dialog. Using the entry `useFirstClipSettings` or any invalid entry is the same as using the Auto setting in the Import XML dialog. (See [Importing Documents](#) (page 10).)

filterreconnectmediafiles

Description A Boolean value specifying whether or not to reconnect clips to media files on import.

Parent `importoptions`

Notes The default setting is `TRUE`.

filterincludemarkers

Description A Boolean value specifying whether or not to import encoded markers.

Parent `importoptions`

Notes The default setting is `TRUE`.

filterincludeeffects

Description A Boolean value specifying whether or not to import encoded effects.

Parent `importoptions`

Notes The default setting is TRUE.

filterincludefiltersequencesettings

Description A Boolean value specifying whether or not to imported encoded sequence settings.

Parent `importoptions`

Notes The default setting is TRUE.

displaynonfatalerrors

Description A Boolean value specifying whether or not to display nonfatal errors.

Parent `importoptions`

Notes The default setting is TRUE.

createfcpprojectatxmlfilepath

Description A Boolean value indicating whether a project file should be created at the same location as the imported xml file.

Parent `importoptions`

Notes If this element is set to TRUE, importing the xml file `/foo/bar/clouds.xml` would create the file `/foo/bar/clouds.fcp`.

deletethisxmlfileatimport

Description A Boolean value indicating whether an xml file should be deleted at the end of import.

Parent `importoptions`

Notes By default, the value for this element is FALSE. If critical errors are encountered during import, the file is not deleted.

DTD for the Interchange Format

The Document Type Definition (DTD) for the Final Cut Pro XML Interchange Format lets you independently validate an interchange format document before you import it into Final Cut Pro.

Note: The Final Cut Pro XML DTD is not integrated into the Final Cut Pro import process. That is, Final Cut Pro does not validate a file against the DTD during import.

There are five separate DTDs for versions 1, 2, 3, 4, and 5 of the interchange format.

Using the DTDs

The DTDs for the Final Cut Pro XML Interchange Format are available in text file format. You can find these files on the Apple Developer web site at the following URL: <http://developer.apple.com/appleapplications/download>.

You can use these DTDs to validate an interchange format document prior to importing the document into Final Cut Pro.

Note: Successful validation does not necessarily guarantee successful import. That is, it is possible for a document to pass validation with the DTD and still not import into Final Cut Pro without issues.

You can use the `xmllint` program available in OS X version 10.3 to validate an interchange format document. Follow these steps:

1. Place a copy of the DTD file in the same folder with the interchange format document you want to validate.
2. Open the Terminal application and navigate to the directory with the DTD file.
3. Execute the following command line: `%xmllint --noout --dtdvalid fcpxmlv1.dtd myxmlfile.file` where `myxmlfile.xml` is the name of the file you want to validate

DTD for Version 1 of the Interchange Format

Listing A-1 Version 1 DTD

```
<!-- ===== -->
<!-- Copyright 2005 Apple Computer, Inc. -->
<!-- Final Cut Pro XML DTD v2.0 for Final Cut Pro XML Interchange Format v1.0-->
<!-- May 10, 2005 -->
<!-- ===== -->

<!ELEMENT xmeml (project | sequence | bin | clip | importoptions)*>
<!ATTLIST xmeml version NMTOKENS #REQUIRED>

<!-- ===== -->
<!-- MAJOR ELEMENTS -->
<!-- Top-level containers specifiable under xmeml root. -->
<!-- ===== -->

<!ELEMENT importoptions (createnewproject | targetprojectname |
defsequencepresetname | displaynonfatalerrors | filterreconnectmediafiles |
filterincludemarkers | filterincludeeffects | filterincludesettings)*>

<!ELEMENT project (name | children)*>

<!ELEMENT children (bin | clip | sequence)*>

<!ELEMENT bin (name | children | labels | comments)*>

<!ELEMENT clip (name | duration | rate | enabled | in | out | anamorphic |
alphatype | alphareverse | compositemode | masterclipid | ismasterclip | labels
| comments | stillframeoffset | subclipinfo | logginginfo | stillframe | file |
marker | filter | sourcetrack | subframeoffset | timecode | media | start | end
| startoffset | endoffset | pixelaspectratio)*>
<!ATTLIST clip id CDATA #IMPLIED>

<!ELEMENT sequence (name | duration | rate | timecode | in | out | marker | media
| sequence | ismasterclip | labels | comments | logginginfo | description |
masterclipid | file | pixelaspectratio | fielddominance | filter)*>
```

```
<!ATTLIST sequence id CDATA #IMPLIED>
```

```
<!-- ===== -->
```

```
<!-- COMMON ELEMENTS -->
```

```
<!-- Use as necessary to complete top-level containers. -->
```

```
<!-- ===== -->
```

```
<!ELEMENT clipitem (name | duration | rate | enabled | in | out | start | end |
  anamorphic | alphasize | alphareverse | compositemode | masterclipid |
  ismasterclip | labels | comments | stillframeoffset | sequence | subclipinfo |
  logginginfo | stillframe | timecode | syncoffset | file | primarytimecode | marker
  | filter | sourcetrack | link | subframeoffset | pixelaspectratio |
  fielddominance)*>
```

```
<!ATTLIST clipitem id CDATA #IMPLIED>
```

```
<!ELEMENT transitionitem (name | rate | start | end | alignment | effect)*>
```

```
<!ELEMENT generatoritem (name | duration | rate | enabled | in | out | start |
  end | anamorphic | alphasize | effect | sourcetrack | filter | timecode |
  primarytimecode | alphareverse | link | logginginfo | labels | comments |
  mediadelay | fielddominance | subframeoffset | masterclipid | compositemode |
  syncoffset | marker )*>
```

```
<!ATTLIST generatoritem id CDATA #IMPLIED>
```

```
<!ELEMENT filter (start | end | effect | enabled)*>
```

```
<!ELEMENT sourcetrack (mediatype | trackindex)*>
```

```
<!ELEMENT file (name | rate | duration | media | timecode | pathurl | width |
  height | mediaSource)*>
```

```
<!ATTLIST file id CDATA #IMPLIED>
```

```
<!ELEMENT media (video | audio)*>
```

```
<!ELEMENT video (in | out | track | format | samplecharacteristics | duration |
  trackcount | stillframe | alphasize | layerindex)*>
```

```
<!ELEMENT audio (in | out | track | format | outputs | trackcount |
  channeldescription | rate | samplecharacteristics | filter | duration)*>
```

```
<!ELEMENT track (clipitem | transitionitem | generatoritem | enabled | locked |
outputchannelindex)*>

<!ELEMENT format (#PCDATA | samplecharacteristics | appspecificdata)*>

<!ELEMENT samplecharacteristics (width | height | anamorphic | pixelaspectratio |
fielddominance | colordepth | codec | depth | samplerate | rate)*>

<!ELEMENT codec (name | appspecificdata)*>

<!ELEMENT appspecificdata (appname | appmanufacturer | appversion | data)*>

<!ELEMENT data (fcpimageprocessing | qtcodec | qteffectid)*>

<!ELEMENT fcpimageprocessing (useyuv | usesuperwhite | rendermode)*>

<!ELEMENT qtcodec (codecname | codectypename | codectypecode | codecvendorcode |
spatialquality | temporalquality | keyframerate | datarate)*>

<!ELEMENT logginginfo (description | good | scene | shottake | lognote)*>

<!ELEMENT outputs (group)*>

<!ELEMENT group (index | numchannels | downmix | channel)*>

<!ELEMENT channel (index)>

<!ELEMENT timecode (rate | string | frame | source | displayformat | format | reel
| field)*>

<!ELEMENT rate (ntsc | timebase)*>

<!ELEMENT reel (name)>

<!ELEMENT link (mediatype | trackindex | clipindex | groupindex | linkclipref)*>
```

```
<!ELEMENT valuelist (valueentry)*>

<!ELEMENT valueentry (name | value)*>

<!ELEMENT value (#PCDATA | horiz | vert | alpha | red | green | blue | timecode |
clip | sequence)*>

<!ELEMENT marker (name | in | out | comment)*>

<!ELEMENT effect (name | effectid | effectcategory | effecttype | mediatype |
parameter | wipecode | wipeaccuracy | startratio | endratio | reverse | rate |
duration | keyframe | appspecificdata | privatestate )*>

<!ELEMENT parameter (name | parameterid | interpolation | valuemin | valuemax |
keyframe | valuelist | value | appspecificdata | privatestate)*>

<!ELEMENT keyframe (when | value | inscale | outscale | inbez | outbez |
interpolation | hadbezierin | hadbezierout | speedkfstart | speedkfstend | speedkfin
| speedkfend | speedvirtualkf | origvalue)*>

<!ELEMENT inbez (horiz | vert)*>

<!ELEMENT outbez (horiz | vert)*>

<!ELEMENT interpolation (name)*>

<!ELEMENT labels (label | label2)*>

<!ELEMENT comments (mastercomment1 | mastercomment2 | mastercomment3 | mastercomment4
| clipcommenta | clipcommentb)*>

<!ELEMENT subclipinfo (startoffset | endoffset)*>

<!-- ===== -->
<!-- TERMINALS -->
<!-- ===== -->
```

```
<!ELEMENT in (#PCDATA)*>
<!ELEMENT out (#PCDATA)*>
<!ELEMENT index (#PCDATA)*>
<!ELEMENT numchannels (#PCDATA)*>
<!ELEMENT downmix (#PCDATA)*>
<!ELEMENT width (#PCDATA)*>
<!ELEMENT height (#PCDATA)*>
<!ELEMENT anamorphic (#PCDATA)*>
<!ELEMENT pixelaspectratio (#PCDATA)*>
<!ELEMENT fielddominance (#PCDATA)*>
<!ELEMENT name (#PCDATA)*>
<!ELEMENT duration (#PCDATA)*>
<!ELEMENT masterclipid (#PCDATA)*>
<!ELEMENT ismasterclip (#PCDATA)*>
<!ELEMENT good (#PCDATA)*>
<!ELEMENT lognote (#PCDATA)*>
<!ELEMENT scene (#PCDATA)*>
<!ELEMENT ntsc (#PCDATA)*>
<!ELEMENT timebase (#PCDATA)*>
<!ELEMENT codecname (#PCDATA)*>
<!ELEMENT codectypename (#PCDATA)*>
<!ELEMENT codectypecode (#PCDATA)*>
<!ELEMENT codecvendorcode (#PCDATA)*>
<!ELEMENT spatialquality (#PCDATA)*>
<!ELEMENT temporalquality (#PCDATA)*>
<!ELEMENT keyframerate (#PCDATA)*>
<!ELEMENT datarate (#PCDATA)*>
<!ELEMENT string (#PCDATA)*>
<!ELEMENT frame (#PCDATA)*>
<!ELEMENT source (#PCDATA)*>
<!ELEMENT displayformat (#PCDATA)*>
<!ELEMENT mediatype (#PCDATA)*>
<!ELEMENT trackindex (#PCDATA)*>
```

```
<!ELEMENT clipindex (#PCDATA)*>
<!ELEMENT groupindex (#PCDATA)*>
<!ELEMENT when (#PCDATA)*>
<!ELEMENT effectid (#PCDATA)*>
<!ELEMENT qteffectid (#PCDATA)*>
<!ELEMENT effectcategory (#PCDATA)*>
<!ELEMENT effecttype (#PCDATA)*>
<!ELEMENT privatestate (#PCDATA)*>
<!ELEMENT parameterid (#PCDATA)*>
<!ELEMENT valuemin (#PCDATA)*>
<!ELEMENT valuemax (#PCDATA)*>
<!ELEMENT inscale (#PCDATA)*>
<!ELEMENT outscale (#PCDATA)*>
<!ELEMENT alpha (#PCDATA)*>
<!ELEMENT red (#PCDATA)*>
<!ELEMENT green (#PCDATA)*>
<!ELEMENT blue (#PCDATA)*>
<!ELEMENT start (#PCDATA)*>
<!ELEMENT end (#PCDATA)*>
<!ELEMENT wipecode (#PCDATA)*>
<!ELEMENT wipeaccuracy (#PCDATA)*>
<!ELEMENT startratio (#PCDATA)*>
<!ELEMENT endratio (#PCDATA)*>
<!ELEMENT reverse (#PCDATA)*>
<!ELEMENT alphatype (#PCDATA)*>
<!ELEMENT alphareverse (#PCDATA)*>
<!ELEMENT compositemode (#PCDATA)*>
<!ELEMENT label (#PCDATA)*>
<!ELEMENT label2 (#PCDATA)*>
<!ELEMENT clipcommenta (#PCDATA)*>
<!ELEMENT clipcommentb (#PCDATA)*>
<!ELEMENT mastercomment1 (#PCDATA)*>
<!ELEMENT mastercomment2 (#PCDATA)*>
<!ELEMENT mastercomment3 (#PCDATA)*>
<!ELEMENT mastercomment4 (#PCDATA)*>
```

```
<!ELEMENT comment (#PCDATA)*>
<!ELEMENT stillframeoffset (#PCDATA)*>
<!ELEMENT stillframe (#PCDATA)*>
<!ELEMENT syncoffset (#PCDATA)*>
<!ELEMENT colordepth (#PCDATA)*>
<!ELEMENT appname (#PCDATA)*>
<!ELEMENT appmanufacturer (#PCDATA)*>
<!ELEMENT appversion (#PCDATA)*>
<!ELEMENT enabled (#PCDATA)*>
<!ELEMENT locked (#PCDATA)*>
<!ELEMENT outputchannelindex (#PCDATA)*>
<!ELEMENT depth (#PCDATA)*>
<!ELEMENT useyuv (#PCDATA)*>
<!ELEMENT usesuperwhite (#PCDATA)*>
<!ELEMENT rendermode (#PCDATA)*>
<!ELEMENT samplerate (#PCDATA)*>
<!ELEMENT createnewproject (#PCDATA)*>
<!ELEMENT targetprojectname (#PCDATA)*>
<!ELEMENT defsequencepresetname (#PCDATA)*>
<!ELEMENT displaynonfatalerrors (#PCDATA)*>
<!ELEMENT filterreconnectmediafiles (#PCDATA)*>
<!ELEMENT filterincludemarkers (#PCDATA)*>
<!ELEMENT filterincludeeffects (#PCDATA)*>
<!ELEMENT filterincludessequencesettings (#PCDATA)*>
<!ELEMENT trackcount (#PCDATA)*>
<!ELEMENT channeldescription (#PCDATA)*>
<!ELEMENT subframeoffset (#PCDATA)*>
<!ELEMENT startoffset (#PCDATA)*>
<!ELEMENT endoffset (#PCDATA)*>
<!ELEMENT linkclipref (#PCDATA)*>
<!ELEMENT description (#PCDATA)*>
<!ELEMENT shottake (#PCDATA)*>
<!ELEMENT alignment (#PCDATA)*>
<!ELEMENT horiz (#PCDATA)*>
```



```
<!ELEMENT vert (#PCDATA)*>
<!ELEMENT pathurl (#PCDATA)*>
<!ELEMENT layerindex (#PCDATA)*>
<!ELEMENT field (#PCDATA)*>
<!ELEMENT primarytimecode (#PCDATA)*>
<!ELEMENT hadbezierin (#PCDATA)*>
<!ELEMENT hadbezierout (#PCDATA)*>
<!ELEMENT speedkfstart (#PCDATA)*>
<!ELEMENT speedkfbend (#PCDATA)*>
<!ELEMENT speedkfend (#PCDATA)*>
<!ELEMENT speedkfstart (#PCDATA)*>
<!ELEMENT speedkfend (#PCDATA)*>
<!ELEMENT speedvirtualkf (#PCDATA)*>
<!ELEMENT origvalue (#PCDATA)*>
<!ELEMENT mediaSource (#PCDATA)*>
```

DTD for Version 2 of the Interchange Format

Listing A-2 Version 2 DTD

```
<!-- ===== -->
<!-- Copyright 2005 Apple Computer, Inc. -->
<!-- Final Cut Pro XML DTD v1.0 for Final Cut Pro XML Interchange Format v2.0-->
<!-- May 10, 2005 -->
<!-- ===== -->

<!ELEMENT xmeml (project | sequence | bin | clip | importoptions)*>
<!ATTLIST xmeml version NMTOKENS #REQUIRED>

<!-- ===== -->
<!-- MAJOR ELEMENTS -->
<!-- Top-level containers specifiable under xmeml root. -->
<!-- ===== -->

<!ELEMENT importoptions (createnewproject | targetprojectname |
defsequencepresetname | displaynonfatalerrors | filterreconnectmediafiles |
filterincludemarkers | filterincludeeffects | filterincludessequencesettings)*>

<!ELEMENT project (name | children)*>

<!ELEMENT children (bin | clip | sequence)*>

<!ELEMENT bin (name | children | labels | comments)*>

<!ELEMENT clip (name | duration | rate | enabled | in | out | anamorphic |
alphatype | alphareverse | compositemode | masterclipid | ismasterclip | labels
| comments | stillframeoffset | subclipinfo | logginginfo | stillframe | file |
marker | filter | sourcetrack | subframeoffset | timecode | media | start | end
| startoffset | endoffset | pixelaspectratio | defaultangle | filmdata)*>
<!ATTLIST clip id CDATA #IMPLIED>

<!ELEMENT sequence (name | duration | rate | timecode | in | out | marker | media
| sequence | ismasterclip | labels | comments | logginginfo | description |
masterclipid | filmdata | file | pixelaspectratio | fielddominance | filter)*>
```

```
<!ATTLIST sequence id CDATA #IMPLIED>
```

```
<!-- ===== -->
```

```
<!--          COMMON ELEMENTS          -->
```

```
<!--          Use as necessary to complete top-level containers.          -->
```

```
<!-- ===== -->
```

```
<!ELEMENT clipitem (name | duration | rate | enabled | in | out | start | end |
  anamorphic | alphatype | alphareverse | compositemode | masterclipid |
  ismasterclip | labels | comments | stillframeoffset | sequence | subclipinfo |
  logginginfo | stillframe | timecode | syncoffset | file | marker | filter |
  sourcetrack | link | subframeoffset | multiclip | primarytimecode | mediadelay |
  filmdata | pixelaspectratio | fielddominance)*>
```

```
<!ATTLIST clipitem id CDATA #IMPLIED>
```

```
<!ELEMENT filmdata (appspecificdata | filmslate | cameraroll | labroll | keycode
  | dailyroll | inknumber | filmstandard | telecinespeed)*>
```

```
<!ELEMENT filmslate (scene | take | slate )*>
```

```
<!ELEMENT keycode (prefix | feetframes | perfoffset | reversed )*>
```

```
<!ELEMENT inknumber (prefix | feetframes)*>
```

```
<!ELEMENT multiclip (name | angle | collapsed | synctype)*>
```

```
<!ATTLIST multiclip id CDATA #IMPLIED>
```

```
<!ELEMENT angle (clip | activevideoangle | activeaudioangle )*>
```

```
<!ELEMENT transitionitem (name | rate | start | end | alignment | effect)*>
```

```
<!ELEMENT generatoritem (name | duration | rate | enabled | in | out | start |
  end | anamorphic | alphatype | effect | sourcetrack | filter | timecode |
  primarytimecode | alphareverse | link | logginginfo | labels | comments |
  mediadelay | fielddominance | filmdata | subframeoffset | masterclipid |
  compositemode | syncoffset | marker )*>
```

```
<!ATTLIST generatoritem id CDATA #IMPLIED>
```

```
<!ELEMENT filter (start | end | effect | enabled)*>
```

```
<!ELEMENT sourcetrack (mediatype | trackindex)*>
```

```
<!ELEMENT file (name | rate | duration | media | timecode | pathurl | width |  
height | mediaSource)*>
```

```
<!ATTLIST file id CDATA #IMPLIED>
```

```
<!ELEMENT media (video | audio)*>
```

```
<!ELEMENT video (in | out | track | format | samplecharacteristics | duration |  
trackcount | stillframe | alphasize | layerindex)*>
```

```
<!ELEMENT audio (in | out | track | format | outputs | channelcount |  
channeldescription | rate | samplecharacteristics | layout | audiochannel |  
trackcount | filter | duration )*>
```

```
<!ELEMENT track (clipitem | transitionitem | generatoritem | enabled | locked |  
outputchannelindex)*>
```

```
<!ELEMENT format (#PCDATA | samplecharacteristics | appspecificdata)*>
```

```
<!ELEMENT samplecharacteristics (width | height | anamorphic | pixelaspectratio |  
fielddominance | colordepth | codec | depth | samplerate | rate)*>
```

```
<!ELEMENT codec (name | appspecificdata)*>
```

```
<!ELEMENT appspecificdata (appname | appmanufacturer | appversion | data)*>
```

```
<!ELEMENT data (fcpimageprocessing | qtcodec | qteffectid | UUID )*>
```

```
<!ELEMENT fcpimageprocessing (useyuv | usesuperwhite | rendermode)*>
```

```
<!ELEMENT qtcodec (codecname | codectypename | codetypecode | codecvendorcode |  
spatialquality | temporalquality | keyframerate | datarate)*>
```

```
<!ELEMENT logginginfo (description | good | scene | shottake | lognote | takenote  
| shotnote | scenenote)*>
```

```
<!ELEMENT outputs (group)*>
```

```
<!ELEMENT group (index | numchannels | downmix | channel)*>
```

```
<!ELEMENT channel (index)>
```

```
<!ELEMENT audiochannel ( channellabel | sourcechannel)*>
```

```
<!ELEMENT timecode (rate | string | frame | source | displayformat | format | reel  
| field)*>
```

```
<!ELEMENT rate (ntsc | timebase)*>
```

```
<!ELEMENT reel (name)>
```

```
<!ELEMENT link (mediatype | trackindex | clipindex | groupindex | linkclipref)*>
```

```
<!ELEMENT valuelist (valueentry)*>
```

```
<!ELEMENT valueentry (name | value)*>
```

```
<!ELEMENT value (#PCDATA | horiz | vert | alpha | red | green | blue | timecode |  
clip | sequence)*>
```

```
<!ELEMENT marker (name | in | out | comment)*>
```

```
<!ELEMENT effect (name | effectid | effectcategory | effecttype | mediatype |  
parameter | wipecode | wipeaccuracy | startratio | endratio | reverse | rate |  
duration | keyframe | appspecificdata | privatestate | multiclip)*>
```

```
<!ELEMENT parameter (name | parameterid | interpolation | valuemin | valuemax |  
keyframe | valuelist | value | appspecificdata | privatestate)*>
```

```
<!ELEMENT keyframe (when | value | inscale | outscale | inbez | outbez |
interpolation | hadbezierin | hadbezierout | speedkfstart | speedkfend | speedkfend
| speedkfend | speedvirtualkf | origvalue)*>
```

```
<!ELEMENT inbez (horiz | vert)*>
```

```
<!ELEMENT outbez (horiz | vert)*>
```

```
<!ELEMENT interpolation (name)*>
```

```
<!ELEMENT labels (label | label2)*>
```

```
<!ELEMENT comments (mastercomment1 | mastercomment2 | mastercomment3 | mastercomment4
| clipcommenta | clipcommentb)*>
```

```
<!ELEMENT subclipinfo (startoffset | endoffset)*>
```

```
<!-- ===== -->
```

```
<!--                                TERMINALS                                -->
```

```
<!-- ===== -->
```

```
<!ELEMENT in (#PCDATA)*>
```

```
<!ELEMENT out (#PCDATA)*>
```

```
<!ELEMENT index (#PCDATA)*>
```

```
<!ELEMENT numchannels (#PCDATA)*>
```

```
<!ELEMENT downmix (#PCDATA)*>
```

```
<!ELEMENT width (#PCDATA)*>
```

```
<!ELEMENT height (#PCDATA)*>
```

```
<!ELEMENT anamorphic (#PCDATA)*>
```

```
<!ELEMENT pixelaspectratio (#PCDATA)*>
```

```
<!ELEMENT fielddominance (#PCDATA)*>
```

```
<!ELEMENT name (#PCDATA)*>
```

```
<!ELEMENT duration (#PCDATA)*>
```

```
<!ELEMENT masterclipid (#PCDATA)*>
```

```
<!ELEMENT ismasterclip (#PCDATA)*>
```

```
<!ELEMENT good (#PCDATA)*>
<!ELEMENT lognote (#PCDATA)*>
<!ELEMENT scene (#PCDATA)*>
<!ELEMENT ntsc (#PCDATA)*>
<!ELEMENT timebase (#PCDATA)*>
<!ELEMENT codecname (#PCDATA)*>
<!ELEMENT codectypename (#PCDATA)*>
<!ELEMENT codectypecode (#PCDATA)*>
<!ELEMENT codecvendorcode (#PCDATA)*>
<!ELEMENT spatialquality (#PCDATA)*>
<!ELEMENT temporalquality (#PCDATA)*>
<!ELEMENT keyframerate (#PCDATA)*>
<!ELEMENT datarate (#PCDATA)*>
<!ELEMENT string (#PCDATA)*>
<!ELEMENT frame (#PCDATA)*>
<!ELEMENT source (#PCDATA)*>
<!ELEMENT displayformat (#PCDATA)*>
<!ELEMENT mediatype (#PCDATA)*>
<!ELEMENT trackindex (#PCDATA)*>
<!ELEMENT clipindex (#PCDATA)*>
<!ELEMENT groupindex (#PCDATA)*>
<!ELEMENT when (#PCDATA)*>
<!ELEMENT effectid (#PCDATA)*>
<!ELEMENT qteffectid (#PCDATA)*>
<!ELEMENT effectcategory (#PCDATA)*>
<!ELEMENT effecttype (#PCDATA)*>
<!ELEMENT privatestate (#PCDATA)*>
<!ELEMENT parameterid (#PCDATA)*>
<!ELEMENT valuemin (#PCDATA)*>
<!ELEMENT valuemax (#PCDATA)*>
<!ELEMENT inscale (#PCDATA)*>
<!ELEMENT outscale (#PCDATA)*>
<!ELEMENT alpha (#PCDATA)*>
<!ELEMENT red (#PCDATA)*>
<!ELEMENT green (#PCDATA)*>
```

```
<!ELEMENT blue (#PCDATA)*>
<!ELEMENT start (#PCDATA)*>
<!ELEMENT end (#PCDATA)*>
<!ELEMENT wipecode (#PCDATA)*>
<!ELEMENT wipeaccuracy (#PCDATA)*>
<!ELEMENT startratio (#PCDATA)*>
<!ELEMENT endratio (#PCDATA)*>
<!ELEMENT reverse (#PCDATA)*>
<!ELEMENT alphatype (#PCDATA)*>
<!ELEMENT alphareverse (#PCDATA)*>
<!ELEMENT compositemode (#PCDATA)*>
<!ELEMENT label (#PCDATA)*>
<!ELEMENT label2 (#PCDATA)*>
<!ELEMENT clipcommenta (#PCDATA)*>
<!ELEMENT clipcommentb (#PCDATA)*>
<!ELEMENT mastercomment1 (#PCDATA)*>
<!ELEMENT mastercomment2 (#PCDATA)*>
<!ELEMENT mastercomment3 (#PCDATA)*>
<!ELEMENT mastercomment4 (#PCDATA)*>
<!ELEMENT comment (#PCDATA)*>
<!ELEMENT stillframeoffset (#PCDATA)*>
<!ELEMENT stillframe (#PCDATA)*>
<!ELEMENT syncoffset (#PCDATA)*>
<!ELEMENT colordepth (#PCDATA)*>
<!ELEMENT appname (#PCDATA)*>
<!ELEMENT appmanufacturer (#PCDATA)*>
<!ELEMENT appversion (#PCDATA)*>
<!ELEMENT enabled (#PCDATA)*>
<!ELEMENT locked (#PCDATA)*>
<!ELEMENT outputchannelindex (#PCDATA)*>
<!ELEMENT depth (#PCDATA)*>
<!ELEMENT useyuv (#PCDATA)*>
<!ELEMENT usesuperwhite (#PCDATA)*>
<!ELEMENT rendermode (#PCDATA)*>
```



```
<!ELEMENT samplerate (#PCDATA)*>
<!ELEMENT createnewproject (#PCDATA)*>
<!ELEMENT targetprojectname (#PCDATA)*>
<!ELEMENT defsequencepresetname (#PCDATA)*>
<!ELEMENT displaynonfatalerrors (#PCDATA)*>
<!ELEMENT filterreconnectmediafiles (#PCDATA)*>
<!ELEMENT filterincludemarkers (#PCDATA)*>
<!ELEMENT filterincludeeffects (#PCDATA)*>
<!ELEMENT filterincludesequencesettings (#PCDATA)*>
<!ELEMENT trackcount (#PCDATA)*>
<!ELEMENT channeldescription (#PCDATA)*>
<!ELEMENT subframeoffset (#PCDATA)*>
<!ELEMENT startoffset (#PCDATA)*>
<!ELEMENT endoffset (#PCDATA)*>
<!ELEMENT linkclipref (#PCDATA)*>
<!ELEMENT description (#PCDATA)*>
<!ELEMENT shottake (#PCDATA)*>
<!ELEMENT alignment (#PCDATA)*>
<!ELEMENT horiz (#PCDATA)*>
<!ELEMENT vert (#PCDATA)*>
<!ELEMENT pathurl (#PCDATA)*>
<!ELEMENT layerindex (#PCDATA)*>
<!ELEMENT field (#PCDATA)*>
<!ELEMENT primarytimecode (#PCDATA)*>
<!ELEMENT hadbezierin (#PCDATA)*>
<!ELEMENT hadbezierout (#PCDATA)*>
<!ELEMENT speedkfstart (#PCDATA)*>
<!ELEMENT speedkfstend (#PCDATA)*>
<!ELEMENT speedkfmin (#PCDATA)*>
<!ELEMENT speedkfmax (#PCDATA)*>
<!ELEMENT speedvirtualkf (#PCDATA)*>
<!ELEMENT origvalue (#PCDATA)*>
<!ELEMENT mediaSource (#PCDATA)*>
<!ELEMENT collapsed (#PCDATA)*>
<!ELEMENT synctype (#PCDATA)*>
```

```
<!ELEMENT activevideoangle (#PCDATA)*>
<!ELEMENT activeaudioangle (#PCDATA)*>
<!ELEMENT mediadelay (#PCDATA)*>
<!ELEMENT defaultangle (#PCDATA)*>
<!ELEMENT channelcount (#PCDATA)*>
<!ELEMENT layout (#PCDATA)*>
<!ELEMENT takenote (#PCDATA)*>
<!ELEMENT shotnote (#PCDATA)*>
<!ELEMENT scenenote (#PCDATA)*>
<!ELEMENT take (#PCDATA)*>
<!ELEMENT slate (#PCDATA)*>
<!ELEMENT cameraroll (#PCDATA)*>
<!ELEMENT labroll (#PCDATA)*>
<!ELEMENT prefix (#PCDATA)*>
<!ELEMENT feetframes (#PCDATA)*>
<!ELEMENT perfoffset (#PCDATA)*>
<!ELEMENT reversed (#PCDATA)*>
<!ELEMENT dailyroll (#PCDATA)*>
<!ELEMENT filmstandard (#PCDATA)*>
<!ELEMENT telecinespeed (#PCDATA)*>
<!ELEMENT channellabel (#PCDATA)*>
<!ELEMENT sourcechannel (#PCDATA)*>
<!ELEMENT UUID (#PCDATA)*>
```

DTD for Version 3 of the Interchange Format

Listing A-3 Version 3 DTD

```
<!--===== -->
<!-- Copyright 2006 Apple Computer, Inc. -->
<!-- Final Cut Pro XML DTD v1.0 for Final Cut Pro XML Interchange Format v3.0-->
<!-- September 14, 2006 -->
<!-- ===== -->

<!ELEMENT xmeml (project | sequence | bin | clip | importoptions)*>

<!ATTLIST xmeml version NMTOKENS #REQUIRED>

<!-- ===== -->
<!-- MAJOR ELEMENTS -->
<!-- Top-level containers specifiable under xmeml root. -->
<!-- ===== -->

<!ELEMENT importoptions (createnewproject | targetprojectname | defsequencepresetname
| displaynonfatalerrors | filterreconnectmediafiles | filterincludemarkers |
filterincludeeffects | filterincludedesquenceesettings)*>

<!ELEMENT project (name | children)*>

<!ELEMENT children (bin | clip | sequence)*>

<!ELEMENT bin (name | children | labels | comments | uuid | updatebehavior)*>

<!ELEMENT clip (name | duration | rate | enabled | in | out | anamorphic | alphatype
| alphareverse | compositemode | masterclipid | ismasterclip | labels | comments
| stillframeoffset | subclipinfo | logginginfo | stillframe | file | marker |
filter | sourcetrack | subframeoffset | timecode | media | start | end | startoffset
| endoffset | pixelaspectratio | defaultangle | filmdata | uuid | updatebehavior)*>
<!ATTLIST clip id CDATA #IMPLIED>

<!ELEMENT sequence (name | duration | rate | timecode | in | out | marker | media
| sequence | ismasterclip | labels | comments | logginginfo | description |
```

```
masterclipid | filmdata | file | pixelaspectratio | fielddominance | filter | uuid
| updatebehavior)*>
<!-- ATTLIST sequence id CDATA #IMPLIED>

<!-- ===== -->
<!--          COMMON ELEMENTS          -->
<!--          Use as necessary to complete top-level containers.          -->
<!-- ===== -->

<!-- ELEMENT clipitem (name | duration | rate | enabled | in | out | start | end |
anamorphic | alphatype | alphareverse | compositemode | masterclipid | ismasterclip
| labels | comments | stillframeoffset | sequence | subclipinfo | logginginfo |
stillframe | timecode | syncoffset | file | marker | filter | sourcetrack | link
| subframeoffset | multiclip | mediadelay | filmdata | pixelaspectratio |
primarytimecode | fielddominance | gamma)*>
<!-- ATTLIST clipitem id CDATA #IMPLIED>

<!-- ELEMENT filmdata (appspecificdata | filmslate | cameraroll | labroll | keycode
| dailyroll | inknumber | filmstandard | telecinespeed)*>

<!-- ELEMENT filmslate (scene | take | slate )*>

<!-- ELEMENT keycode (prefix | feetframes | perfoffset | reversed )*>

<!-- ELEMENT inknumber (prefix | feetframes)*>

<!-- ELEMENT multiclip (name | angle | collapsed | synctype)*>
<!-- ATTLIST multiclip id CDATA #IMPLIED>

<!-- ELEMENT angle (clip | activevideoangle | activeaudioangle )*>

<!-- ELEMENT transitionitem (name | rate | start | end | alignment | effect)*>

<!-- ELEMENT generatoritem (name | duration | rate | enabled | in | out | start | end
| anamorphic | alphatype | effect | sourcetrack | filter | timecode |
primarytimecode | alphareverse | link | logginginfo | labels | comments | mediadelay
| fielddominance | filmdata | subframeoffset | masterclipid | compositemode |
syncoffset | marker )*>
```

```
<!ATTLIST generatoritem id CDATA #IMPLIED>

<!ELEMENT filter (start | end | effect | enabled)*>

<!ELEMENT sourcetrack (mediatype | trackindex)*>

<!ELEMENT file (name | rate | duration | media | timecode | pathurl | width |
height | mediaSource | metadata )*>
<!ATTLIST file id CDATA #IMPLIED>

<!ELEMENT media (video | audio)*>

<!ELEMENT video (in | out | track | format | samplecharacteristics | duration |
trackcount | stillframe | alphasize | layerindex | metadata)*>

<!ELEMENT audio (in | out | track | format | outputs | channelcount |
channeldescription | rate | samplecharacteristics | layout | audiochannel |
trackcount | filter | duration | metadata)*>

<!ELEMENT track (clipitem | transitionitem | generatoritem | enabled | locked |
outputchannelindex)*>

<!ELEMENT format (#PCDATA | samplecharacteristics | appspecificdata)*>

<!ELEMENT samplecharacteristics (width | height | anamorphic | pixelaspectratio |
fielddominance | colordepth | codec | depth | samplerate | rate)*>

<!ELEMENT codec (name | appspecificdata)*>

<!ELEMENT appspecificdata (appname | appmanufacturer | appversion | data)*>

<!ELEMENT data (fcpimageprocessing | qtcodec | qteffectid | UUID )*>

<!ELEMENT fcpimageprocessing (useyuv | usesuperwhite | rendermode)*>

<!ELEMENT qtcodec (codecname | codectypename | codectypecode | codecvendorcode |
spatialquality | temporalquality | keyframerate | datarate)*>
```

```
<!ELEMENT logginginfo (description | good | scene | shottake | lognote | takenote  
| shotnote | scenenote)*>
```

```
<!ELEMENT outputs (group)*>
```

```
<!ELEMENT group (index | numchannels | downmix | channel)*>
```

```
<!ELEMENT channel (index)>
```

```
<!ELEMENT audiochannel ( channellabel | sourcechannel)*>
```

```
<!ELEMENT timecode (rate | string | frame | source | displayformat | format | reel  
| field)*>
```

```
<!ELEMENT rate (ntsc | timebase)*>
```

```
<!ELEMENT reel (name)>
```

```
<!ELEMENT link (mediatype | trackindex | clipindex | groupindex | linkclipref)*>
```

```
<!ELEMENT valuelist (valueentry)*>
```

```
<!ELEMENT valueentry (name | value)*>
```

```
<!ELEMENT value (#PCDATA | horiz | vert | alpha | red | green | blue | timecode |  
clip | sequence)*>
```

```
<!ELEMENT marker (name | in | out | comment)*>
```

```
<!ELEMENT effect (name | effectid | effectcategory | effecttype | mediatype |  
parameter | wipecode | wipeaccuracy | startratio | endratio | reverse | rate |  
duration | keyframe | appspecificdata | privatestate | multiclip | effectclass)*>
```

```
<!ELEMENT parameter (name | parameterid | interpolation | valuemin | valuemax |  
keyframe | valuelist | value | appspecificdata | privatestate)*>
```

```
<!ELEMENT keyframe (when | value | inscale | outscale | inbez | outbez |  
interpolation | hadbezierin | hadbezierout | speedkfstart | speedkfstend | speedkfend | speedkfstart  
| speedkfend | speedkfstart | speedkfend | origvalue)*>
```

```
<!ELEMENT inbez (horiz | vert)*>
```

```
<!ELEMENT outbez (horiz | vert)*>
```

```
<!ELEMENT interpolation (name)*>
```

```
<!ELEMENT labels (label | label2)*>
```

```
<!ELEMENT comments (mastercomment1 | mastercomment2 | mastercomment3 | mastercomment4  
| clipcommenta | clipcommentb)*>
```

```
<!ELEMENT subclipinfo (startoffset | endoffset)*>
```

```
<!ELEMENT metadata (storage | key | updatebehavior | size | type | value)*>
```

```
<!-- ===== -->
```

```
<!--                                TERMINALS                                -->
```

```
<!-- ===== -->
```

```
<!ELEMENT in (#PCDATA)*>
```

```
<!ELEMENT out (#PCDATA)*>
```

```
<!ELEMENT index (#PCDATA)*>
```

```
<!ELEMENT numchannels (#PCDATA)*>
```

```
<!ELEMENT downmix (#PCDATA)*>
```

```
<!ELEMENT width (#PCDATA)*>
```

```
<!ELEMENT height (#PCDATA)*>
```

```
<!ELEMENT anamorphic (#PCDATA)*>
```

```
<!ELEMENT pixelaspectratio (#PCDATA)*>
```

```
<!ELEMENT fielddominance (#PCDATA)*>
```

```
<!ELEMENT name (#PCDATA)*>
```

```
<!ELEMENT duration (#PCDATA)*>
```

```
<!ELEMENT masterclipid (#PCDATA)*>
<!ELEMENT ismasterclip (#PCDATA)*>
<!ELEMENT good (#PCDATA)*>
<!ELEMENT lognote (#PCDATA)*>
<!ELEMENT scene (#PCDATA)*>
<!ELEMENT ntsc (#PCDATA)*>
<!ELEMENT timebase (#PCDATA)*>
<!ELEMENT codecname (#PCDATA)*>
<!ELEMENT codectypename (#PCDATA)*>
<!ELEMENT codectypecode (#PCDATA)*>
<!ELEMENT codecvendorcode (#PCDATA)*>
<!ELEMENT spatialquality (#PCDATA)*>
<!ELEMENT temporalquality (#PCDATA)*>
<!ELEMENT keyframerate (#PCDATA)*>
<!ELEMENT datarate (#PCDATA)*>
<!ELEMENT string (#PCDATA)*>
<!ELEMENT frame (#PCDATA)*>
<!ELEMENT source (#PCDATA)*>
<!ELEMENT displayformat (#PCDATA)*>
<!ELEMENT mediatype (#PCDATA)*>
<!ELEMENT trackindex (#PCDATA)*>
<!ELEMENT clipindex (#PCDATA)*>
<!ELEMENT groupindex (#PCDATA)*>
<!ELEMENT when (#PCDATA)*>
<!ELEMENT effectid (#PCDATA)*>
<!ELEMENT qteffectid (#PCDATA)*>
<!ELEMENT effectcategory (#PCDATA)*>
<!ELEMENT effecttype (#PCDATA)*>
<!ELEMENT effectclass (#PCDATA)*>
<!ELEMENT privatestate (#PCDATA)*>
<!ELEMENT parameterid (#PCDATA)*>
<!ELEMENT valuemin (#PCDATA)*>
<!ELEMENT valuemax (#PCDATA)*>
<!ELEMENT inscale (#PCDATA)*>
```



```
<!ELEMENT outscale (#PCDATA)*>
<!ELEMENT alpha (#PCDATA)*>
<!ELEMENT red (#PCDATA)*>
<!ELEMENT green (#PCDATA)*>
<!ELEMENT blue (#PCDATA)*>
<!ELEMENT start (#PCDATA)*>
<!ELEMENT end (#PCDATA)*>
<!ELEMENT wipecode (#PCDATA)*>
<!ELEMENT wipeaccuracy (#PCDATA)*>
<!ELEMENT startratio (#PCDATA)*>
<!ELEMENT endratio (#PCDATA)*>
<!ELEMENT reverse (#PCDATA)*>
<!ELEMENT alphasize (#PCDATA)*>
<!ELEMENT alphareverse (#PCDATA)*>
<!ELEMENT compositemode (#PCDATA)*>
<!ELEMENT label1 (#PCDATA)*>
<!ELEMENT label2 (#PCDATA)*>
<!ELEMENT clipcommenta (#PCDATA)*>
<!ELEMENT clipcommentb (#PCDATA)*>
<!ELEMENT mastercomment1 (#PCDATA)*>
<!ELEMENT mastercomment2 (#PCDATA)*>
<!ELEMENT mastercomment3 (#PCDATA)*>
<!ELEMENT mastercomment4 (#PCDATA)*>
<!ELEMENT comment (#PCDATA)*>
<!ELEMENT stillframeoffset (#PCDATA)*>
<!ELEMENT stillframe (#PCDATA)*>
<!ELEMENT syncoffset (#PCDATA)*>
<!ELEMENT colordepth (#PCDATA)*>
<!ELEMENT appname (#PCDATA)*>
<!ELEMENT appmanufacturer (#PCDATA)*>
<!ELEMENT appversion (#PCDATA)*>
<!ELEMENT enabled (#PCDATA)*>
<!ELEMENT locked (#PCDATA)*>
<!ELEMENT outputchannelindex (#PCDATA)*>
<!ELEMENT depth (#PCDATA)*>
```

```
<!ELEMENT useyuv (#PCDATA)*>
<!ELEMENT usesuperwhite (#PCDATA)*>
<!ELEMENT rendermode (#PCDATA)*>
<!ELEMENT samplerate (#PCDATA)*>
<!ELEMENT createnewproject (#PCDATA)*>
<!ELEMENT targetprojectname (#PCDATA)*>
<!ELEMENT defsequencepresetname (#PCDATA)*>
<!ELEMENT displaynonfatalerrors (#PCDATA)*>
<!ELEMENT filterreconnectmediafiles (#PCDATA)*>
<!ELEMENT filterincludemarkers (#PCDATA)*>
<!ELEMENT filterincludeeffects (#PCDATA)*>
<!ELEMENT filterincludesequencesettings (#PCDATA)*>
<!ELEMENT trackcount (#PCDATA)*>
<!ELEMENT channeldescription (#PCDATA)*>
<!ELEMENT subframeoffset (#PCDATA)*>
<!ELEMENT startoffset (#PCDATA)*>
<!ELEMENT endoffset (#PCDATA)*>
<!ELEMENT linkclipref (#PCDATA)*>
<!ELEMENT description (#PCDATA)*>
<!ELEMENT shottake (#PCDATA)*>
<!ELEMENT alignment (#PCDATA)*>
<!ELEMENT horiz (#PCDATA)*>
<!ELEMENT vert (#PCDATA)*>
<!ELEMENT pathurl (#PCDATA)*>
<!ELEMENT layerindex (#PCDATA)*>
<!ELEMENT field (#PCDATA)*>
<!ELEMENT primarytimecode (#PCDATA)*>
<!ELEMENT hadbezierin (#PCDATA)*>
<!ELEMENT hadbezierout (#PCDATA)*>
<!ELEMENT speedkfstart (#PCDATA)*>
<!ELEMENT speedkfstend (#PCDATA)*>
<!ELEMENT speedkfstart (#PCDATA)*>
<!ELEMENT speedkfend (#PCDATA)*>
<!ELEMENT speedkfstart (#PCDATA)*>
<!ELEMENT speedkfend (#PCDATA)*>
```

```
<!ELEMENT origvalue (#PCDATA)*>
<!ELEMENT mediaSource (#PCDATA)*>
<!ELEMENT collapsed (#PCDATA)*>
<!ELEMENT synctype (#PCDATA)*>
<!ELEMENT activevideoangle (#PCDATA)*>
<!ELEMENT activeaudioangle (#PCDATA)*>
<!ELEMENT mediadelay (#PCDATA)*>
<!ELEMENT defaultangle (#PCDATA)*>
<!ELEMENT channelcount (#PCDATA)*>
<!ELEMENT layout (#PCDATA)*>
<!ELEMENT takenote (#PCDATA)*>
<!ELEMENT shotnote (#PCDATA)*>
<!ELEMENT scenenote (#PCDATA)*>
<!ELEMENT take (#PCDATA)*>
<!ELEMENT slate (#PCDATA)*>
<!ELEMENT cameraroll (#PCDATA)*>
<!ELEMENT labroll (#PCDATA)*>
<!ELEMENT prefix (#PCDATA)*>
<!ELEMENT feetframes (#PCDATA)*>
<!ELEMENT perfoffset (#PCDATA)*>
<!ELEMENT reversed (#PCDATA)*>
<!ELEMENT dailyroll (#PCDATA)*>
<!ELEMENT filmstandard (#PCDATA)*>
<!ELEMENT telecinespeed (#PCDATA)*>
<!ELEMENT channellabel (#PCDATA)*>
<!ELEMENT sourcechannel (#PCDATA)*>
<!ELEMENT UUID (#PCDATA)*>
<!ELEMENT uuid (#PCDATA)*>
<!ELEMENT updatebehavior (#PCDATA)*>
<!ELEMENT storage (#PCDATA)*>
<!ELEMENT key (#PCDATA)*>
<!ELEMENT size (#PCDATA)*>
<!ELEMENT type (#PCDATA)*>
<!ELEMENT gamma (#PCDATA)*>
```

DTD for Version 4 of the Interchange Format

Listing A-4 Version 4 DTD

```
<!--===== -->
<!-- Copyright 2007 Apple Inc. -->
<!-- Final Cut Pro XML DTD v1.0 for Final Cut Pro XML Interchange Format v4.0-->
<!-- March 26, 2007 -->
<!-- ===== -->

<!ELEMENT xmeml (project | sequence | bin | clip | importoptions)*>

<!ATTLIST xmeml version NMTOKENS #REQUIRED>

<!-- ===== -->
<!-- MAJOR ELEMENTS -->
<!-- Top-level containers specifiable under xmeml root. -->
<!-- ===== -->

<!ELEMENT importoptions (createnewproject | targetprojectname | defsequencepresetname
| displaynonfatalerrors | filterreconnectmediafiles | filterincludemarkers |
filterincludeeffects | filterincludesequencesettings | deletethisxmlfileatimport
| createfcprojectatxmlfilepath)*>

<!ELEMENT project (name | children)*>

<!ELEMENT children (bin | clip | sequence)*>

<!ELEMENT bin (name | children | labels | comments | uuid | updatebehavior)*>

<!ELEMENT clip (name | duration | rate | enabled | in | out | anamorphic | alphasize
| alphasize | compositemode | masterclipid | ismasterclip | labels | comments
| stillframeoffset | subclipinfo | logginginfo | stillframe | file | marker |
filter | sourcetrack | subframeoffset | timecode | media | start | end | startoffset
| endoffset | pixelaspectratio | defaulttangle | filmdata | uuid | updatebehavior)*>
<!ATTLIST clip id CDATA #IMPLIED>
```

```
<!ELEMENT sequence (name | duration | rate | timecode | in | out | marker | media
| sequence | ismasterclip | labels | comments | logginginfo | description |
masterclipid | filmdata | file | pixelaspectratio | fielddominance | filter | uuid
| updatebehavior)*>
```

```
<!ATTLIST sequence id CDATA #IMPLIED>
```

```
<!-- ===== -->
```

```
<!--                COMMON ELEMENTS                -->
```

```
<!--                Use as necessary to complete top-level containers.                -->
```

```
<!-- ===== -->
```

```
<!ELEMENT clipitem (name | duration | rate | enabled | in | out | start | end |
anamorphic | alphasize | alphareverse | compositemode | masterclipid | ismasterclip
| labels | comments | stillframeoffset | sequence | subclipinfo | logginginfo |
stillframe | timecode | syncoffset | file | marker | filter | sourcetrack | link
| subframeoffset | multiclip | mediadelay | filmdata | pixelaspectratio |
fielddominance | gamma | itemhistory | primarytimecode | mixedratesoffset)*>
```

```
<!ATTLIST clipitem id CDATA #IMPLIED>
```

```
<!ELEMENT filmdata (appspecificdata | filmslate | cameraroll | labroll | keycode
| dailyroll | inknumber | filmstandard | telecinespeed)*>
```

```
<!ELEMENT filmslate (scene | take | slate )*>
```

```
<!ELEMENT keycode (prefix | feetframes | perfoffset | reversed )*>
```

```
<!ELEMENT inknumber (prefix | feetframes)*>
```

```
<!ELEMENT multiclip (name | angle | collapsed | synctype)*>
```

```
<!ATTLIST multiclip id CDATA #IMPLIED>
```

```
<!ELEMENT angle (clip | activevideoangle | activeaudioangle )*>
```

```
<!ELEMENT transitionitem (name | rate | start | end | alignment | effect)*>
```

```
<!ELEMENT generatoritem (name | duration | rate | enabled | in | out | start | end
| anamorphic | alphasize | effect | sourcetrack | filter | timecode |
primarytimecode | alphareverse | link | logginginfo | labels | comments | mediadelay
```

```
| fielddominance | filmdata | subframeoffset | masterclipid | compositemode |  
syncoffset | marker )*>  
<!ATTLIST generatoritem id CDATA #IMPLIED>  
  
<!ELEMENT filter (start | end | effect | enabled)*>  
  
<!ELEMENT sourcetrack (mediatype | trackindex)*>  
  
<!ELEMENT file (name | rate | duration | media | timecode | pathurl | width |  
height | mediaSource | metadata )*>  
<!ATTLIST file id CDATA #IMPLIED>  
  
<!ELEMENT media (video | audio)*>  
  
<!ELEMENT video (in | out | track | format | samplecharacteristics | duration |  
trackcount | stillframe | alphasize | layerindex | metadata)*>  
  
<!ELEMENT audio (in | out | track | format | outputs | channelcount |  
channeldescription | rate | samplecharacteristics | layout | audiochannel |  
trackcount | filter | duration | metadata)*>  
  
<!ELEMENT track (clipitem | transitionitem | generatoritem | enabled | locked |  
outputchannelindex)*>  
  
<!ELEMENT format (#PCDATA | samplecharacteristics | appspecificdata)*>  
  
<!ELEMENT samplecharacteristics (width | height | anamorphic | pixelaspectratio |  
fielddominance | colordepth | codec | depth | samplerate | rate)*>  
  
<!ELEMENT codec (name | appspecificdata)*>  
  
<!ELEMENT appspecificdata (appname | appmanufacturer | appversion | data)*>  
  
<!ELEMENT data (fcpimageprocessing | qtcodec | qteffectid | UUID )*>  
  
<!ELEMENT fcpimageprocessing (useyuv | usesuperwhite | rendermode)*>
```

```
<!ELEMENT qtcodec (codecname | codectypename | codectypecode | codecvendorcode |
spatialquality | temporalquality | keyframerate | datarate)*>

<!ELEMENT logginginfo (description | good | scene | shottake | lognote | takenote
| shotnote | scenenote)*>

<!ELEMENT outputs (group)*>

<!ELEMENT group (index | numchannels | downmix | channel)*>

<!ELEMENT channel (index)>

<!ELEMENT audiochannel ( channellabel | sourcechannel)*>

<!ELEMENT timecode (rate | string | frame | source | displayformat | format | reel
| field)*>

<!ELEMENT rate (ntsc | timebase)*>

<!ELEMENT reel (name)>

<!ELEMENT link (mediatype | trackindex | clipindex | groupindex | linkclipref)*>

<!ELEMENT valuelist (valueentry)*>

<!ELEMENT valueentry (name | value)*>

<!ELEMENT value (#PCDATA | horiz | vert | alpha | red | green | blue | timecode |
clip | sequence)*>

<!ELEMENT marker (name | in | out | comment)*>

<!ELEMENT effect (name | effectid | effectcategory | effecttype | mediatype |
parameter | wipecode | wipeaccuracy | startratio | endratio | reverse | rate |
duration | keyframe | appspecificdata | privatestate | multiclip | effectclass)*>
```

```
<!ELEMENT parameter (name | parameterid | interpolation | valuemin | valuemax |
keyframe | valuelist | value | appspecificdata | privatestate)*>

<!ELEMENT keyframe (when | value | inscale | outscale | inbez | outbez |
interpolation | hadbezierin | hadbezierout | speedkfstart | speedkfend | speedkfmin
| speedkfout | speedvirtualkf | origvalue)*>

<!ELEMENT inbez (horiz | vert)*>

<!ELEMENT outbez (horiz | vert)*>

<!ELEMENT interpolation (name)*>

<!ELEMENT labels (label | label2)*>

<!ELEMENT comments (mastercomment1 | mastercomment2 | mastercomment3 | mastercomment4
| clipcommenta | clipcommentb)*>

<!ELEMENT subclipinfo (startoffset | endoffset)*>

<!ELEMENT metadata (storage | key | updatebehavior | size | type | value)*>
<!ELEMENT itemhistory (uuid)*>

<!-- ===== -->
<!--                                TERMINALS                                -->
<!-- ===== -->

<!ELEMENT in (#PCDATA)*>
<!ELEMENT out (#PCDATA)*>
<!ELEMENT index (#PCDATA)*>
<!ELEMENT numchannels (#PCDATA)*>
<!ELEMENT downmix (#PCDATA)*>
<!ELEMENT width (#PCDATA)*>
<!ELEMENT height (#PCDATA)*>
<!ELEMENT anamorphic (#PCDATA)*>
<!ELEMENT pixelaspectratio (#PCDATA)*>
```



```
<!ELEMENT fielddominance (#PCDATA)*>
<!ELEMENT name (#PCDATA)*>
<!ELEMENT duration (#PCDATA)*>
<!ELEMENT masterclipid (#PCDATA)*>
<!ELEMENT ismasterclip (#PCDATA)*>
<!ELEMENT good (#PCDATA)*>
<!ELEMENT lognote (#PCDATA)*>
<!ELEMENT scene (#PCDATA)*>
<!ELEMENT ntsc (#PCDATA)*>
<!ELEMENT timebase (#PCDATA)*>
<!ELEMENT codecname (#PCDATA)*>
<!ELEMENT codectypename (#PCDATA)*>
<!ELEMENT codectypecode (#PCDATA)*>
<!ELEMENT codecvendorcode (#PCDATA)*>
<!ELEMENT spatialquality (#PCDATA)*>
<!ELEMENT temporalquality (#PCDATA)*>
<!ELEMENT keyframerate (#PCDATA)*>
<!ELEMENT datarate (#PCDATA)*>
<!ELEMENT string (#PCDATA)*>
<!ELEMENT frame (#PCDATA)*>
<!ELEMENT source (#PCDATA)*>
<!ELEMENT displayformat (#PCDATA)*>
<!ELEMENT mediatype (#PCDATA)*>
<!ELEMENT trackindex (#PCDATA)*>
<!ELEMENT clipindex (#PCDATA)*>
<!ELEMENT groupindex (#PCDATA)*>
<!ELEMENT when (#PCDATA)*>
<!ELEMENT effectid (#PCDATA)*>
<!ELEMENT qteffectid (#PCDATA)*>
<!ELEMENT effectcategory (#PCDATA)*>
<!ELEMENT effecttype (#PCDATA)*>
<!ELEMENT effectclass (#PCDATA)*>
<!ELEMENT privatestate (#PCDATA)*>
<!ELEMENT parameterid (#PCDATA)*>
<!ELEMENT valuemin (#PCDATA)*>
```

```
<!ELEMENT valuemax (#PCDATA)*>
<!ELEMENT inscale (#PCDATA)*>
<!ELEMENT outscale (#PCDATA)*>
<!ELEMENT alpha (#PCDATA)*>
<!ELEMENT red (#PCDATA)*>
<!ELEMENT green (#PCDATA)*>
<!ELEMENT blue (#PCDATA)*>
<!ELEMENT start (#PCDATA)*>
<!ELEMENT end (#PCDATA)*>
<!ELEMENT wipecode (#PCDATA)*>
<!ELEMENT wipeaccuracy (#PCDATA)*>
<!ELEMENT startratio (#PCDATA)*>
<!ELEMENT endratio (#PCDATA)*>
<!ELEMENT reverse (#PCDATA)*>
<!ELEMENT alphatype (#PCDATA)*>
<!ELEMENT alphareverse (#PCDATA)*>
<!ELEMENT compositemode (#PCDATA)*>
<!ELEMENT label (#PCDATA)*>
<!ELEMENT label2 (#PCDATA)*>
<!ELEMENT clipcommenta (#PCDATA)*>
<!ELEMENT clipcommentb (#PCDATA)*>
<!ELEMENT mastercomment1 (#PCDATA)*>
<!ELEMENT mastercomment2 (#PCDATA)*>
<!ELEMENT mastercomment3 (#PCDATA)*>
<!ELEMENT mastercomment4 (#PCDATA)*>
<!ELEMENT comment (#PCDATA)*>
<!ELEMENT stillframeoffset (#PCDATA)*>
<!ELEMENT stillframe (#PCDATA)*>
<!ELEMENT syncoffset (#PCDATA)*>
<!ELEMENT colordepth (#PCDATA)*>
<!ELEMENT appname (#PCDATA)*>
<!ELEMENT appmanufacturer (#PCDATA)*>
<!ELEMENT appversion (#PCDATA)*>
<!ELEMENT enabled (#PCDATA)*>
```

```
<!ELEMENT locked (#PCDATA)*>
<!ELEMENT outputchannelindex (#PCDATA)*>
<!ELEMENT depth (#PCDATA)*>
<!ELEMENT useyuv (#PCDATA)*>
<!ELEMENT usesuperwhite (#PCDATA)*>
<!ELEMENT rendermode (#PCDATA)*>
<!ELEMENT samplerate (#PCDATA)*>
<!ELEMENT createnewproject (#PCDATA)*>
<!ELEMENT targetprojectname (#PCDATA)*>
<!ELEMENT defsequencepresetname (#PCDATA)*>
<!ELEMENT displaynonfatalerrors (#PCDATA)*>
<!ELEMENT filterreconnectmediafiles (#PCDATA)*>
<!ELEMENT filterincludemarkers (#PCDATA)*>
<!ELEMENT filterincludeeffects (#PCDATA)*>
<!ELEMENT filterincludesequencesettings (#PCDATA)*>
<!ELEMENT createfcpprojectatxmlfilepath (#PCDATA)*>
<!ELEMENT deletethisxmlfileatimport (#PCDATA)*>
<!ELEMENT trackcount (#PCDATA)*>
<!ELEMENT channeldescription (#PCDATA)*>
<!ELEMENT subframeoffset (#PCDATA)*>
<!ELEMENT startoffset (#PCDATA)*>
<!ELEMENT endoffset (#PCDATA)*>
<!ELEMENT linkclipref (#PCDATA)*>
<!ELEMENT description (#PCDATA)*>
<!ELEMENT shottake (#PCDATA)*>
<!ELEMENT alignment (#PCDATA)*>
<!ELEMENT horiz (#PCDATA)*>
<!ELEMENT vert (#PCDATA)*>
<!ELEMENT pathurl (#PCDATA)*>
<!ELEMENT layerindex (#PCDATA)*>
<!ELEMENT field (#PCDATA)*>
<!ELEMENT primarytimecode (#PCDATA)*>
<!ELEMENT hadbezierin (#PCDATA)*>
<!ELEMENT hadbezierout (#PCDATA)*>
<!ELEMENT speedkfstart (#PCDATA)*>
```

```
<!ELEMENT speedkfend (#PCDATA)*>
<!ELEMENT speedkfin (#PCDATA)*>
<!ELEMENT speedkfout (#PCDATA)*>
<!ELEMENT speedvirtualkf (#PCDATA)*>
<!ELEMENT origvalue (#PCDATA)*>
<!ELEMENT mediaSource (#PCDATA)*>
<!ELEMENT collapsed (#PCDATA)*>
<!ELEMENT synctype (#PCDATA)*>
<!ELEMENT activevideoangle (#PCDATA)*>
<!ELEMENT activeaudioangle (#PCDATA)*>
<!ELEMENT mediadelay (#PCDATA)*>
<!ELEMENT defaultangle (#PCDATA)*>
<!ELEMENT channelcount (#PCDATA)*>
<!ELEMENT layout (#PCDATA)*>
<!ELEMENT takenote (#PCDATA)*>
<!ELEMENT shotnote (#PCDATA)*>
<!ELEMENT scenenote (#PCDATA)*>
<!ELEMENT take (#PCDATA)*>
<!ELEMENT slate (#PCDATA)*>
<!ELEMENT cameraroll (#PCDATA)*>
<!ELEMENT labroll (#PCDATA)*>
<!ELEMENT prefix (#PCDATA)*>
<!ELEMENT feetframes (#PCDATA)*>
<!ELEMENT perfoffset (#PCDATA)*>
<!ELEMENT reversed (#PCDATA)*>
<!ELEMENT dailyroll (#PCDATA)*>
<!ELEMENT filmstandard (#PCDATA)*>
<!ELEMENT telecinespeed (#PCDATA)*>
<!ELEMENT channellabel (#PCDATA)*>
<!ELEMENT sourcechannel (#PCDATA)*>
<!ELEMENT UUID (#PCDATA)*>
<!ELEMENT uuid (#PCDATA)*>
<!ELEMENT updatebehavior (#PCDATA)*>
<!ELEMENT storage (#PCDATA)*>
```

```
<!ELEMENT key (#PCDATA)*>  
<!ELEMENT size (#PCDATA)*>  
<!ELEMENT type (#PCDATA)*>  
<!ELEMENT gamma (#PCDATA)*>  
<!ELEMENT mixedratesoffset (#PCDATA)*>
```

DTD for Version 5 of the Interchange Format

Listing A-5 Version 5 DTD

```
<!--===== -->
<!-- Copyright 2009 Apple Inc. -->
<!-- Final Cut Pro XML DTD v1.0 for Final Cut Pro XML Interchange Format v5.0-->
<!-- July 9, 2009 -->
<!-- ===== -->

<!ELEMENT xmeml (project | sequence | bin | clip | importoptions)*>

<!ATTLIST xmeml version NMTOKENS #REQUIRED>

<!-- ===== -->
<!-- MAJOR ELEMENTS -->
<!-- Top-level containers specifiable under xmeml root. -->
<!-- ===== -->

<!ELEMENT importoptions (createnewproject | targetprojectname | defsequencepresetname
| displaynonfatalerrors | filterreconnectmediafiles | filterincludemarkers |
filterincludeeffects | filterincludesequencesettings | deletethisxmlfileatimport
| createfcprojectatxmlfilepath)*>

<!ELEMENT project (name | children)*>

<!ELEMENT children (bin | clip | sequence)*>

<!ELEMENT bin (name | children | labels | comments | uuid | updatebehavior)*>

<!ELEMENT clip (name | duration | rate | enabled | in | out | anamorphic | alphasize
| alphareverse | compositemode | masterclipid | ismasterclip | labels | comments
| stillframeoffset | subclipinfo | logginginfo | stillframe | file | marker |
filter | sourcetrack | subframeoffset | timecode | media | start | end | startoffset
| endoffset | pixelaspectratio | defaultangle | filmdata | uuid | updatebehavior
| subclipmasterid)*>

<!ATTLIST clip id CDATA #IMPLIED>
```

```
<!ELEMENT sequence (name | duration | rate | timecode | in | out | marker | media
| sequence | ismasterclip | labels | comments | logginginfo | description |
masterclipid | filmdata | file | pixelaspectratio | fielddominance | filter | uuid
| updatebehavior)*>
```

```
<!ATTLIST sequence id CDATA #IMPLIED>
```

```
<!-- ===== -->
```

```
<!--                COMMON ELEMENTS                -->
```

```
<!--                Use as necessary to complete top-level containers.                -->
```

```
<!-- ===== -->
```

```
<!ELEMENT clipitem (name | duration | rate | enabled | in | out | start | end |
anamorphic | alphasize | alphareverse | compositemode | masterclipid | ismasterclip
| labels | comments | stillframeoffset | sequence | subclipinfo | logginginfo |
stillframe | timecode | syncoffset | file | marker | filter | sourcetrack | link
| subframeoffset | multiclip | mediadelay | filmdata | pixelaspectratio |
fielddominance | gamma | itemhistory | primarytimecode | mixedratesoffset |
subclipmasterid)*>
```

```
<!ATTLIST clipitem id CDATA #IMPLIED>
```

```
<!ELEMENT filmdata (appspecificdata | filmslate | cameraroll | labroll | keycode
| dailyroll | inknumber | filmstandard | telecinespeed)*>
```

```
<!ELEMENT filmslate (scene | take | slate )*>
```

```
<!ELEMENT keycode (prefix | feetframes | perfoffset | reversed )*>
```

```
<!ELEMENT inknumber (prefix | feetframes)*>
```

```
<!ELEMENT multiclip (name | angle | collapsed | synctype)*>
```

```
<!ATTLIST multiclip id CDATA #IMPLIED>
```

```
<!ELEMENT angle (clip | activevideoangle | activeaudioangle )*>
```

```
<!ELEMENT transitionitem (name | rate | start | end | alignment | effect)*>
```

```
<!ELEMENT generatoritem (name | duration | rate | enabled | in | out | start | end
| anamorphic | alphasize | effect | sourcetrack | filter | timecode |
primarytimecode | alphareverse | link | logginginfo | labels | comments | mediadelay
```

```
| fielddominance | filmdata | subframeoffset | masterclipid | compositemode |  
syncoffset | marker | subclipmasterid)*>  
<!ATTLIST generatoritem id CDATA #IMPLIED>  
  
<!ELEMENT filter (start | end | effect | enabled)*>  
  
<!ELEMENT sourcetrack (mediatype | trackindex)*>  
  
<!ELEMENT file (name | rate | duration | media | timecode | pathurl | width |  
height | mediaSource | metadata )*>  
<!ATTLIST file id CDATA #IMPLIED>  
  
<!ELEMENT media (video | audio)*>  
  
<!ELEMENT video (in | out | track | format | samplecharacteristics | duration |  
trackcount | stillframe | alphasize | layerindex | metadata)*>  
  
<!ELEMENT audio (in | out | track | format | outputs | channelcount |  
channeldescription | rate | samplecharacteristics | layout | audiochannel |  
trackcount | filter | duration | metadata)*>  
  
<!ELEMENT track (clipitem | transitionitem | generatoritem | enabled | locked |  
outputchannelindex)*>  
  
<!ELEMENT format (#PCDATA | samplecharacteristics | appspecificdata)*>  
  
<!ELEMENT samplecharacteristics (width | height | anamorphic | pixelaspectratio |  
fielddominance | colordepth | codec | depth | samplerate | rate)*>  
  
<!ELEMENT codec (name | appspecificdata)*>  
  
<!ELEMENT appspecificdata (appname | appmanufacturer | appversion | data)*>  
  
<!ELEMENT data (fcpimageprocessing | qtcodec | qteffectid | UUID )*>  
  
<!ELEMENT fcpimageprocessing (useyuv | usesuperwhite | rendermode)*>
```



```
<!ELEMENT qtcodec (codecname | codectypename | codectypecode | codecvendorcode |
spatialquality | temporalquality | keyframerate | datarate)*>

<!ELEMENT logginginfo (description | good | scene | shottake | lognote | takenote
| shotnote | scenenote)*>

<!ELEMENT outputs (group)*>

<!ELEMENT group (index | numchannels | downmix | channel)*>

<!ELEMENT channel (index)>

<!ELEMENT audiochannel ( channellabel | sourcechannel)*>

<!ELEMENT timecode (rate | string | frame | source | displayformat | format | reel
| field)*>

<!ELEMENT rate (ntsc | timebase)*>

<!ELEMENT reel (name)>

<!ELEMENT link (mediatype | trackindex | clipindex | groupindex | linkclipref)*>

<!ELEMENT valuelist (valueentry)*>

<!ELEMENT valueentry (name | value)*>

<!ELEMENT value (#PCDATA | horiz | vert | alpha | red | green | blue | timecode |
clip | sequence)*>

<!ELEMENT marker (name | in | out | comment)*>

<!ELEMENT effect (name | effectid | effectcategory | effecttype | mediatype |
parameter | wipecode | wipeaccuracy | startratio | endratio | reverse | rate |
duration | keyframe | appspecificdata | privatestate | multiclip | effectclass)*>
```

```
<!ELEMENT parameter (name | parameterid | parameterspecifier | interpolation |
valuemin | valuemax | keyframe | valuelist | value | appspecificdata |
privatestate)*>

<!ELEMENT keyframe (when | value | inscale | outscale | inbez | outbez |
interpolation | hadbezierin | hadbezierout | speedkfstart | speedkfstend | speedkfend | speedkfin
| speedkfout | speedvirtualkf | origvalue | anchoroffset)*>

<!ELEMENT inbez (horiz | vert)*>

<!ELEMENT outbez (horiz | vert)*>

<!ELEMENT interpolation (name)*>

<!ELEMENT labels (label | label2)*>

<!ELEMENT comments (mastercomment1 | mastercomment2 | mastercomment3 | mastercomment4
| clipcommenta | clipcommentb)*>

<!ELEMENT subclipinfo (startoffset | endoffset)*>

<!ELEMENT metadata (storage | key | updatebehavior | size | type | value)*>
<!ELEMENT itemhistory (uuid)*>

<!-- ===== -->
<!--                                TERMINALS                                -->
<!-- ===== -->

<!ELEMENT in (#PCDATA)*>
<!ELEMENT out (#PCDATA)*>
<!ELEMENT index (#PCDATA)*>
<!ELEMENT numchannels (#PCDATA)*>
<!ELEMENT downmix (#PCDATA)*>
<!ELEMENT width (#PCDATA)*>
<!ELEMENT height (#PCDATA)*>
<!ELEMENT anamorphic (#PCDATA)*>
```

```
<!ELEMENT pixelaspectratio (#PCDATA)*>
<!ELEMENT fielddominance (#PCDATA)*>
<!ELEMENT name (#PCDATA)*>
<!ELEMENT duration (#PCDATA)*>
<!ELEMENT masterclipid (#PCDATA)*>
<!ELEMENT ismasterclip (#PCDATA)*>
<!ELEMENT subclipmasterid (#PCDATA)*>
<!ELEMENT good (#PCDATA)*>
<!ELEMENT lognote (#PCDATA)*>
<!ELEMENT scene (#PCDATA)*>
<!ELEMENT ntsc (#PCDATA)*>
<!ELEMENT timebase (#PCDATA)*>
<!ELEMENT codecname (#PCDATA)*>
<!ELEMENT codectypename (#PCDATA)*>
<!ELEMENT codectypecode (#PCDATA)*>
<!ELEMENT codecvendorcode (#PCDATA)*>
<!ELEMENT spatialquality (#PCDATA)*>
<!ELEMENT temporalquality (#PCDATA)*>
<!ELEMENT keyframerate (#PCDATA)*>
<!ELEMENT datarate (#PCDATA)*>
<!ELEMENT string (#PCDATA)*>
<!ELEMENT frame (#PCDATA)*>
<!ELEMENT source (#PCDATA)*>
<!ELEMENT displayformat (#PCDATA)*>
<!ELEMENT mediatype (#PCDATA)*>
<!ELEMENT trackindex (#PCDATA)*>
<!ELEMENT clipindex (#PCDATA)*>
<!ELEMENT groupindex (#PCDATA)*>
<!ELEMENT when (#PCDATA)*>
<!ELEMENT effectid (#PCDATA)*>
<!ELEMENT qteffectid (#PCDATA)*>
<!ELEMENT effectcategory (#PCDATA)*>
<!ELEMENT effecttype (#PCDATA)*>
<!ELEMENT effectclass (#PCDATA)*>
<!ELEMENT privatestate (#PCDATA)*>
```

```
<!ELEMENT parameterid (#PCDATA)*>
<!ELEMENT parameterspecifier (#PCDATA)*>
<!ELEMENT valuemin (#PCDATA)*>
<!ELEMENT valuemax (#PCDATA)*>
<!ELEMENT inscale (#PCDATA)*>
<!ELEMENT outscale (#PCDATA)*>
<!ELEMENT alpha (#PCDATA)*>
<!ELEMENT red (#PCDATA)*>
<!ELEMENT green (#PCDATA)*>
<!ELEMENT blue (#PCDATA)*>
<!ELEMENT start (#PCDATA)*>
<!ELEMENT end (#PCDATA)*>
<!ELEMENT wipecode (#PCDATA)*>
<!ELEMENT wipeaccuracy (#PCDATA)*>
<!ELEMENT startratio (#PCDATA)*>
<!ELEMENT endratio (#PCDATA)*>
<!ELEMENT reverse (#PCDATA)*>
<!ELEMENT alphatype (#PCDATA)*>
<!ELEMENT alphareverse (#PCDATA)*>
<!ELEMENT compositemode (#PCDATA)*>
<!ELEMENT label (#PCDATA)*>
<!ELEMENT label2 (#PCDATA)*>
<!ELEMENT clipcommenta (#PCDATA)*>
<!ELEMENT clipcommentb (#PCDATA)*>
<!ELEMENT mastercomment1 (#PCDATA)*>
<!ELEMENT mastercomment2 (#PCDATA)*>
<!ELEMENT mastercomment3 (#PCDATA)*>
<!ELEMENT mastercomment4 (#PCDATA)*>
<!ELEMENT comment (#PCDATA)*>
<!ELEMENT stillframeoffset (#PCDATA)*>
<!ELEMENT stillframe (#PCDATA)*>
<!ELEMENT syncoffset (#PCDATA)*>
<!ELEMENT colordepth (#PCDATA)*>
<!ELEMENT appname (#PCDATA)*>
```

```
<!ELEMENT appmanufacturer (#PCDATA)*>
<!ELEMENT appversion (#PCDATA)*>
<!ELEMENT enabled (#PCDATA)*>
<!ELEMENT locked (#PCDATA)*>
<!ELEMENT outputchannelindex (#PCDATA)*>
<!ELEMENT depth (#PCDATA)*>
<!ELEMENT useyuv (#PCDATA)*>
<!ELEMENT usesuperwhite (#PCDATA)*>
<!ELEMENT rendermode (#PCDATA)*>
<!ELEMENT samplerate (#PCDATA)*>
<!ELEMENT createnewproject (#PCDATA)*>
<!ELEMENT targetprojectname (#PCDATA)*>
<!ELEMENT defsequencepresetname (#PCDATA)*>
<!ELEMENT displaynonfatalerrors (#PCDATA)*>
<!ELEMENT filterreconnectmediafiles (#PCDATA)*>
<!ELEMENT filterincludemarkers (#PCDATA)*>
<!ELEMENT filterincludeeffects (#PCDATA)*>
<!ELEMENT filterincludesequencesettings (#PCDATA)*>
<!ELEMENT createfcpprojectatxmlfilepath (#PCDATA)*>
<!ELEMENT deletethisxmlfileatimport (#PCDATA)*>
<!ELEMENT trackcount (#PCDATA)*>
<!ELEMENT channeldescription (#PCDATA)*>
<!ELEMENT subframeoffset (#PCDATA)*>
<!ELEMENT startoffset (#PCDATA)*>
<!ELEMENT endoffset (#PCDATA)*>
<!ELEMENT linkclipref (#PCDATA)*>
<!ELEMENT description (#PCDATA)*>
<!ELEMENT shottake (#PCDATA)*>
<!ELEMENT alignment (#PCDATA)*>
<!ELEMENT horiz (#PCDATA)*>
<!ELEMENT vert (#PCDATA)*>
<!ELEMENT pathurl (#PCDATA)*>
<!ELEMENT layerindex (#PCDATA)*>
<!ELEMENT field (#PCDATA)*>
<!ELEMENT primarytimecode (#PCDATA)*>
```

```
<!ELEMENT hadbezierin (#PCDATA)*>
<!ELEMENT hadbezierout (#PCDATA)*>
<!ELEMENT speedkfstart (#PCDATA)*>
<!ELEMENT speedkffend (#PCDATA)*>
<!ELEMENT speedkfmin (#PCDATA)*>
<!ELEMENT speedkffout (#PCDATA)*>
<!ELEMENT speedvirtualkf (#PCDATA)*>
<!ELEMENT origvalue (#PCDATA)*>
<!ELEMENT mediaSource (#PCDATA)*>
<!ELEMENT collapsed (#PCDATA)*>
<!ELEMENT synctype (#PCDATA)*>
<!ELEMENT activevideoangle (#PCDATA)*>
<!ELEMENT activeaudioangle (#PCDATA)*>
<!ELEMENT mediadelay (#PCDATA)*>
<!ELEMENT defaultangle (#PCDATA)*>
<!ELEMENT channelcount (#PCDATA)*>
<!ELEMENT layout (#PCDATA)*>
<!ELEMENT takenote (#PCDATA)*>
<!ELEMENT shotnote (#PCDATA)*>
<!ELEMENT scenenote (#PCDATA)*>
<!ELEMENT take (#PCDATA)*>
<!ELEMENT slate (#PCDATA)*>
<!ELEMENT cameraroll (#PCDATA)*>
<!ELEMENT labroll (#PCDATA)*>
<!ELEMENT prefix (#PCDATA)*>
<!ELEMENT feetframes (#PCDATA)*>
<!ELEMENT perfoffset (#PCDATA)*>
<!ELEMENT reversed (#PCDATA)*>
<!ELEMENT dailyroll (#PCDATA)*>
<!ELEMENT filmstandard (#PCDATA)*>
<!ELEMENT telecinespeed (#PCDATA)*>
<!ELEMENT channellabel (#PCDATA)*>
<!ELEMENT sourcechannel (#PCDATA)*>
<!ELEMENT UUID (#PCDATA)*>
```

```
<!ELEMENT uuid (#PCDATA)*>  
<!ELEMENT updatebehavior (#PCDATA)*>  
<!ELEMENT storage (#PCDATA)*>  
<!ELEMENT key (#PCDATA)*>  
<!ELEMENT size (#PCDATA)*>  
<!ELEMENT type (#PCDATA)*>  
<!ELEMENT gamma (#PCDATA)*>  
<!ELEMENT mixedratesoffset (#PCDATA)*>  
<!ELEMENT anchoroffset (#PCDATA)*>
```

Frame Rates

Table B-1 displays the settings for the elements `timebase` and `ntsc` that are required to specify various video types and frame rates.

Table B-1 Frames per second

Type	Canonical rate	Actual rate	FCP rate	<code>timebase</code>	<code>ntsc</code>
24P	23.98/23.976	23.97602397...	23.976	24	true
Film	24	24.00	24	24	false
PAL	25	25.00	25	25	false
NTSC/HD	29.97	29.97002997...	29.97	30	true
Video/HD	30	30.00	30	30	false
HD	50	50.00	50	50	false
HD	59.94	59.94005994...	59.94	60	true
HD	60	60.00	60	60	false

Keyframe Interpolation

This appendix describes the algorithms that Final Cut Pro uses to interpolate keyframes with Bezier curves (other than speed).

About Final Cut Pro Interpolation

Final Cut Pro uses both 1D and 2D Bezier curves to interpolate keyframes. The 2D Bezier algorithm is a simple extension of the 1D algorithm.

The heart of the math Final Cut Pro uses to find a given point on a Bezier curve is described in the article "Quick and Simple Bezier Curve Drawing" by Robert Miller, which appears in *Graphic Gems* Volume 5, page 206. Final Cut Pro modifies this algorithm to account for acceleration in a curve. As a result, Final Cut Pro users can change the velocity of motion into and out of a keyframe.

In Final Cut Pro, Bezier curves are represented by an approximation of a curve made up of 40 linear segments. The Bezier code itself is divided into two parts. The first part constructs a Bezier structure and calculates the location of all 40 constituent segments of the curve. The second part of the code finds the location of a given time on the curve.

To create the structure that represents a Bezier curve, Final Cut Pro first initializes a number of variables. This first set of variables—the acceleration coefficients—enables the user to ease the effect into the endpoints of the curve. This is done by biasing them either into or out of the endpoints using an exponential scaling factor to create a power curve.

Listing C-1 Acceleration coefficients

```
leftScale = curve[i].scale[1];  
leftVal = pow(10, -leftScale)- 1.0;  
rightScale = curve[i+1].scale[0];  
rightVal = pow(10, -rightScale) - 1.0;
```

The scale values `leftScale` and `rightScale` are obtained from a UI widget. The values of this float scale factor range from 1 (biased away from the endpoint) to -1 (biased toward the endpoint).

Next, Final Cut Pro initializes the Cartesian location of the control points of the Bezier curve segment by looking at the values passed in as an array of Bezier points, `FloatPt *seg`. These values represent the location of the control points of the Bezier curve.

Listing C-2 Control points

```
MakeBezSegment(Bezier curve, int32 index, FloatPt *seg) {

    FloatPt temp;

    seg[0] = curve[index].location;
    seg[3] = curve[index+1].location;

    if (curve[index].vector[1].length == 0) {
        seg[1].h = seg[0].h + (seg[3].h - seg[0].h) / 2.0 / SCALE_MULT_CONST;
        seg[1].v = seg[0].v + (seg[3].v - seg[0].v) / 2.0 / SCALE_MULT_CONST;
    }
    else {
        PolarToCart(&curve[index].vector[1], &temp);
        seg[1].h = seg[0].h + temp.h * SCALE_MULT_CONST;
        seg[1].v = seg[0].v + temp.v * SCALE_MULT_CONST;
    }

    if (curve[index+1].vector[0].length == 0) {
        seg[2].h = seg[3].h + (seg[0].h - seg[3].h) / 2.0 / SCALE_MULT_CONST;
        seg[2].v = seg[3].v + (seg[0].v - seg[3].v) / 2.0 / SCALE_MULT_CONST;
    }
    else {
        PolarToCart(&curve[index+1].vector[0], &temp);
        seg[2].h = seg[3].h + temp.h * SCALE_MULT_CONST;
        seg[2].v = seg[3].v + temp.v * SCALE_MULT_CONST;
    }
}
```

With this initialization work complete, Final Cut Pro now employs the algorithm found in the Robert Miller article to quickly and iteratively compute the location of points on the curve. First, Final Cut Pro initializes the values of the vertices of the control polygon of the Bezier curve:

Listing C-3 Locations

```
BezierForm(FloatPt *p, FloatPt *c, int numCtlPoints) {  
  
    Int32 i, multiplier;  
  
    for (i = 0; i <= numCtlPoints - 1; i++) {  
        if (i == 0) multiplier = 1;  
        else if (i == 1) multiplier = numCtlPoints - 1;  
        else multiplier = multiplier * (numCtlPoints - i) / i;  
  
        c[k].h = p[k].h * multiplier;  
        c[k].v = p[k].v * multiplier;  
    }  
}
```

Now Final Cut Pro is ready to calculate the final position of each linear segment that represents the Bezier curve. It makes this calculation using the final control points of the curve and the acceleration coefficients calculated earlier.

Final Cut Pro loops over the 40 segments that make up the curve. For each point, it calculates the distance along the curve as a fraction, for example 0, 1/40th, 2/40th, 3/40th, ... 39/40th, 40/40th. Then it determines whether the point is on the left or right half of the curve and applies the appropriate acceleration value. It multiplies the acceleration constant calculated at the outset by a contribution factor that yields a desired acceleration factor.

Listing C-4 Linear segments

```
percent = (float)i / 40;  
if (percent <= 0.5) {  
    if (leftScale != 0.0) {  
        contribution = (1.0 - (percent * 2));  
        contribution *= contribution;           // ^2  
        contribution *= contribution;         // ^4  
    }  
}
```

```
        contribution *= contribution;           // ^8
        scale = 1.0 + leftVal * contribution;
    }
    else scale = 1.0;
}
else {
    if (rightScale != 0.0) {
        contribution = ((percent * 2) - 1.0);
        contribution *= contribution;           // ^2
        contribution *= contribution;           // ^4
        contribution *= contribution;           // ^8
        scale = 1.0 + rightVal * contribution;
    }
    else scale = 1.0;
}
```

In effect, these calculations push the segments of the curve in one direction or the other, either "squishing" the curve segments toward an endpoint, or "pulling" them away.

Now Final Cut Pro pulls these values together. First, it fixes the position of each control point which was calculated using Miller's algorithm `BezierForm`. See [Listing C-3](#) (page 163) Using the final scaling factor calculated in [Listing C-4](#) (page 163) the program can stretch or shrink the magnitude of this linear segment of the curve. It can calculate the value of the curve at any point by interpolating between these control points.

Listing C-5 Bezier curve

```
BezierCurve(FloatPt *c, FloatPt *pt, float t) {

    int32          i, n;
    float          t1, tt, u;
    FloatPt        b[NUM_CONTL_POINTS];

    n = NUM_CONTL_POINTS - 1;
    u = t;

    b[0].h = c[0].h;
```

```
b[0].v = c[0].v;
for (i =1; i <=n; i++) {
    b[i].h = c[i].h * u;
    b[i].v = c[i].v * u;
    u = u * t;
}

(*pt).h = b[n].h;  (*pt).v = b[n].v;

t1 = 1 - t;
tt = t1;
for (k = n - 1; i >= 0; i--) {
    (*pt).h += b[i].h * tt;
    (*pt).v += b[i].v * tt;
    tt = tt * t1;
}
}
```

Now that Final Cut Pro has all the information needed about the Cartesian position of the current line segment along the actual curve, it can update the Bezier structure representing the curve with this data. This process involves recording the start and end position of the segment and calculating the line connecting these points, while scaling the line segment by the calculated acceleration factor. Looping around from point to point, the application calculates the position of all 40 points of the curve.

Lastly, before finishing with this segment of the curve, Final Cut Pro calculates the time value for each segment point. Later, it uses these time values to look up and interpolate values between the segment points. This process is a simple matter of calculating the current position on the curve and then using this value to compute the total time along the curve this position represents.

Apple Events and Final Cut Pro

Final Cut Pro version 5.1.2 and later offers support for Apple events that external applications can use to directly manage the import and export of XML in a Final Cut project.

These custom Apple events provide the following functions:

- Open a project file.
- Close a project file, either saving or discarding any changes.
- Retrieve the XML representation of a project file.
- Send XML data to Final Cut Pro directly for import.
- Update the media file.

To alert the user to changes in a project that result from these import or export operations, there are also several Apple events that control the display of project information, including:

- Select a single item in the Browser and highlight it for the user.
- Open individual items from the Browser into the Viewer window.
- Use Final Cut's built-in Find mechanism to locate specific elements of a project.

Final Cut Pro 6.0.2 provided Apple Events in support of the following functions:

- Get a list of all open project files.
- Update information about a media file.
- Get an XML representation of all available effects.
- Retrieve the XML representation of a single item.

New in Final Cut Pro 7.0

Final Cut Pro 7.0 provides new Apple Events that support the following functions:

- Save a specified project file.
- Get a table of contents (bins, clips, sequences) for a specified project.

Note: The Apple Developer website provides [sample code](#) that demonstrates the use of Apple events in Final Cut Pro. The sample code includes a header file you can use in constructing your own application.

Final Cut Pro Apple Events

Open Project (`kFCPOpenProjectFile 'ofcP'`)

Description Opens a specified project file. If the project file is already open, the event brings it to the foreground inside Final Cut Pro.

Parameter `kFCPPProjectFileKey 'fcpP'` typeFSRef/typeFileURL. The project file to open.

Close Project (`kFCPSaveAndCloseProjectFile 'cfcP'`)

Description Closes the specified project file, either saving any changes or discarding them.

Parameters `kFCPPProjectFileKey 'fcpP'` typeFSRef/typeFileURL. The project to close. If the project is not open, nothing happens. If no project is specified, the most recently opened file is closed.

`kFCPPProjectFileCloseFlagsKey 'fcCF'` typeSInt32. A flag controlling the behavior of the close event. The possible values are:

`kFCPSaveAndCloseProject` - saves any changes and then closes the file.

`kFCPDiscardAndCloseProject` - discards any changes and closes the file.

If no flag is specified, the default behavior is save and close.

Save Project As (`kFCPSaveProjectFileAs 'sfcP'`)

Description Saves the specified project file, possibly to a new name and location. If the specified project is not open, nothing happens. If no project file is specified, the top-most open project is saved.

Parameters `kFCPPProjectFileKey 'fcpP'`. The project file to save (FSRef).

`kFCPPProjectFileURL 'fcpU'`. The name and location for the saved project file (file url).

`kFCPPProjectFileOverwriteFlag 'fsOF'`. An optional Boolean flag controlling overwriting. TRUE means Final Cut Pro overwrites a project file with the same name and location; FALSE means Final Cut Pro cancels the save operation if a project exists with the same name and location. The default setting is FALSE.

Get Table of Contents (`kFCPGetProjectTOC 'eTCX'`)

Description Returns a Table of Contents list of bins, clips, and sequences in the given project. If no project file parameter is sent, the top-most project is used.

Parameter `kFCPPProjectFileKey 'fcpP'`. The project file to list.

Results `kFCPXMLDataKey 'xmLD'`. An XML stream listing out a 'table of contents' for the project. Each listing includes the name, uuid, duration, and rate information for the specified item.

Get Open Projects List (kFCPGetAllOpenProjects 'fcLP')

Description Returns a list of open project files as FSRefs. Only projects with an associated file are returned.

Results kFCP0penProjectList 'fcOP'. An AEList of AERecords, each of which has a kFCP0neOpenProjectFile parameter.
kFCP0neOpenProjectFile 'fcPf'. A single path to a project file in the list of open projects. This parameter only occurs within the AEList returned by kFCP0penProjectList. Its value is a file FSRef.

Get Project XML (kFCPGetDocumentXML 'eXML')

Description Requests the XML representation of a Final Cut project. If the project file is closed, it is first opened. The project is always brought to the foreground in Final Cut. If no project file is specified, Final Cut Pro returns XML for the front-most project.

Parameters kFCPProjectFileKey 'fcpP' typeFSRef/typeFileURL. The project to operate on.
kFCPXMLDataVersion 'xmlV' typeFloat. The version of the Final Cut Pro XML Interchange Format to use. If unspecified, then the current version is used.

Results kFCPXMLDataKey 'xmlD' typeUTF8Text. The XML data for the project represented as a UTF-8 string.

Get Item XML (kFCPGetItemXML 'gXML')

Description Requests the XML representation of a single item in a Final Cut Pro project by UUID.

Parameters kFCPProjectFileKey 'fcpP' typeFSRef/typeFileURL. The project to operate on. If you omit this parameter and specify a UUID parameter only, Final Cut looks for the item in the front-most project.
kFCPIItemUUID 'fcIU' typeUTF8Text. The UUID for the item. If you omit this parameter and specify a project that is open, Final Cut returns the XML for the front-most sequence, whether this sequence is in the specified project or not. If you omit this parameter and specify a project that is closed, Final Cut opens the project, makes any open sequence in that project front-most, and returns the XML for that sequence.
kFCPXMLDataVersion 'xmlV' typeFloat. The version of the Final Cut Pro XML Interchange Format to use. If unspecified, then the current version is used.

Results kFCPXMLDataKey 'xmlD' typeUTF8Text. The XML data for the item represented as a UTF-8 string.

Import XML (kFCPImportXMLToDocument 'iXML')

Description Directly imports XML into Final Cut Pro without referencing XML saved on disk. You can specify a specific project file to import the XML into. If you don't specify a file, the XML is imported

into a new, untitled project. Using `updatebehavior`, you can specify whether the imported items replace items in the project or are simply appended to the project.

Parameter `kFCPPProjectFileKey 'fcpP' typeFSRef/typeFileURL`. The project to import into. If unspecified, a new untitled project is created.

`kFCPXMLDataKey 'xmlD' typeUTF8Text`. The XML data to import represented as a UTF-8 string. Note this is a buffer of XML data, not a URL. See the [sample code](#) for Apple Events in Final Cut Pro.

Get All Effects (`kFCPGetAllEffectsXML 'eFXX'`)

Description Allows an application to request the XML representation of all installed effects.

Parameters `kFCPXMLDataVersion 'xmlV' typeFloat`. An optional value controlling the version of the XML Interchange Format to use. The default is the current version.

Results `kFCPXMLDataKey 'xmlD' typeUTF8Text`. The XML data represented as a UTF-8 string.
NOTE: The XML for effects cannot be imported back into Final Cut Pro.

Update Media File (`kFCPUpdateMediaFile 'udfF'`)

Description Requests Final Cut Pro to examine a media file modification time and perform a silent reconnect or update if necessary.

Parameters `kFCPMediaFileKey 'fcpM' typeFSRef/typeFileURL`. The media file to examine and update.

`kFCPUpdateFileIgnoreModDate 'fIMD'`. An optional Boolean flag controlling updating. TRUE means Final Cut Pro updates the media file regardless of its modification time; FALSE means Final Cut Pro updates the media file only if it has changed since the last time Final Cut examined the file. The default setting is FALSE.

Select Item (`kFCPSelectItemInBrowser 'fcSI'`)

Description Highlights a specified item or items in the Browser window. Use this event to bring one or more items to the attention of the user.

Parameters `kFCPPProjectFileKey 'fcpP' typeFSRef/typeFileURL`. The project file to operate on.

`kFCPItemUUID 'fcIU' typeUTF8Text`. The UUID of a single item to select in the Browser.

`kFCPItemsToSelectList 'fcSL' AEDescList`. A list specifying multiple items to select at the same time. The parameter is an AEList of AERecords each of which has a single `kFCPItemUUID` parameter.

NOTE: You can use either a `kFCPItemUUID` parameter or a `kFCPItemsToSelectList` parameter, but not both.

Open Item (kFCPOpenItemInProject 'fcOI')

Description Opens a specified item. Clips open in the Viewer; sequences in the Timeline; bins in a separate window.

Parameters kFCPPProjectFileKey 'fcpP' typeFSRef/typeFileURL. The project file to operate on.
kFCPIItemUUID 'fcIU' typeUTF8Text. The UUID of the item to open.

Find Items (kFCPFindItemsInProject 'ffcP')

Description Searches for items. If the Find Results window is active, the search takes place among elements in this window. Otherwise, the search takes place in the Browser. The search results are displayed in a new Find Results window. This lets you bring multiple items to the attention of the user at one time.

The kFCPFindParameters list contains one or more entries that compose the search criteria. You can use multiple entries to precisely control the results obtained. For example, one entry can specify that the Name column of the items must start with a string 'Camera' and the next entry specifies that the Notes column must not contain the string 'bad take'. Each criteria entry is made up of four elements: the string to search for, the way to match the string, the column to search, and whether to use matching or non-matching items. Note that each criteria entry is an AERecord, so the list consists of records, each of which contains several parameters.

Parameters kFCPPProjectFileKey 'fcpP' typeFSRef/typeFileURL. The project file to operate on.
kFCPFindLogicMode 'fndL' typeSInt32. The logic mode for the search. You can combine search criteria using AND (all criteria apply) or OR (only one criteria need apply). Possible values are:

kFCPFindLogicAnd = 0

kFCPFindLogicOr = 1

kFCPFindParameters 'fndP' typeAEList typeAERecord. An AEDescList list of criteria AERecords. The list is made up of one or more "criteria" records that define the string to search for, the string matching logic to use, whether to find matching items or items that don't match (i.e. 'skip'), and the name of a Browser column to search in. There are four criteria you can specify: kFCPFindSearchString, kFCPFindOmitCriteria, kFCPFindMatchMode and kFCPFindColumnName .

Search criteria kFCPFindSearchString 'fndS' typeUTF8Text. The string to search for.

kFCPFindOmitCriteria 'fndO' typeSInt32. The way the search criteria should be used. A zero value means find items that match the search string. A non-zero value means find items that don't match the search string. (That is, skip any item matching the search string.)

kFCPFindMatchMode 'fndM' typeSInt32. The way a string should be matched. Valid entries are:

```
kFCPFindMatchStartsWith = (1 << 0)
kFCPFindMatchContains = (1 << 1)
kFCPFindMatchEquals = (1 << 2 )
kFCPFindMatchEndsWith = (1 << 3)
kFCPFindMatchLessThan = (1 << 4)
kFCPFindMatchGreaterThan = (1 << 5)
```

kFCPFindColumnName 'fndC' typeUTF8Text. Restricts the search to a specific Browser column. If not specified or if the passed string is not a valid entry, then all columns are searched. Valid column names are: name, duration, in, out, start, end, scene, shottake, lognote, good, label, label2, mastercomment1, mastercomment2, mastercomment3, mastercomment4, clipcommenta, clipcommentb.

Versions of XMEML and Final Cut

This appendix provides information about versions of the Final Cut Pro XML Interchange Format, as well as information about changes in versions of the Final Cut Pro application itself that are relevant to XMEML.

Versions of XMEML

Version 1

Version 1 of the interchange format was released with Final Cut Pro 4.0.

The elements of version 1 of the interchange format are specified in [DTD for Version 1 of the Interchange Format](#) (page 114) All of the elements of version 1 are valid in subsequent versions except for `trackcount`, which is deprecated in version 2 and later.

Version 2

Version 2 of the interchange format was released with Final Cut Pro 5.0

Version 2 of the interchange format introduced these new elements:

- `filmdata` and its subelements `appspecificdata`, `filmslate`, `cameraroll`, `labroll`, `keycode`, `dailyroll`, `inknumber`, `filmstandard`, `telecinespeed`.
- `filmslate` and its subelements `scene`, `take`, `slate`.
- `multiclip` and its subelements `name`, `angle`, `collapsed`, `synctype`.
- `angle` and its subelements `clip`, `activevideoangle`, `activeaudioangle`.
- `audiochannel` and its subelements `channellabel`, `sourcechannel`.
- `mediadelay`
- `channelcount`
- `layout`
- `takenote`
- `shotnote`

- scenenote
- feetframes
- perfoffset
- reversed
- channellabel
- UUID

Version 3

Version 3 of the interchange format was released with Final Cut Pro 5.1.2.

Version 3 introduces support for these new capabilities:

- Management of metadata in QuickTime files. See [About master clips](#) (page 17)
- Management of project components. See [Managing Project Components](#) (page 53)
- Support for FxPlug plug-in effects. See [effectclass](#) (page 99)
- Support for specifying a gamma setting for a still. See [gamma](#) (page 90)

The new elements in version 3 of the interchange format are:

- uuid
- updatebehavior
- metadata and its subelements storage, key, updatebehavior, size, type, value.
- effectclass
- gamma

Version 4

The new elements in version 4 of the interchange format (released with Final Cut Pro version 6.0) are:

- itemhistory - see [itemhistory](#) (page 82)
- mixedratesoffset - see [mixedratesoffset](#) (page 89)
- createfcpprojectatxmlfilepath - see [createfcpprojectatxmlfilepath](#) (page 112)
- deletethisxmlfileatimport - see [deletethisxmlfileatimport](#) (page 112)

Version 5

Version 5 of the interchange format was released with Final Cut Pro 7.0.

Version 5 of the interchange format includes these new elements:

- `subclipmasterid` - see [subclipmasterid](#) (page 84)
- `parameterspecifier` - see [parameterspecifier](#) (page 100)
- `color` - see [color](#) (page 94)
- `anchoroffset`—see [The anchoroffset Element](#) (page 51).

Versions of Final Cut Pro

Changes in Final Cut Pro 6.0.2

- The end-of-line delimiter in an exported XML file is now the LF (line feed) character, rather than the CR (carriage return) character. This facilitates editing with standard text editors.
- An empty element instance (such as `<value></value>`, or `<value/>`) is now treated as an empty string rather than 0. But an instance that references another instance with an id (for example, `<file id = "FooBar"/>`), is processed as if it were a duplicate of referenced instance.
- Only the filters specified are added when an XML document is imported into Final Cut Pro. For example, no Shift Field filter is added automatically when an imported clip is brought into a sequence with a different field dominance setting.
- The element `importoptions` lets you control options for importing an XML document into Final Cut Pro. Specifying the subelements `createnewproject` and `defsequencepresetname` suppresses the Import XML dialog. You no longer need to specify the subelement `targetprojectname` to do this. (This means you can import the XML into the default project and still suppress the dialog.)
- The `anamorphic` setting in the imported XML document is now respected even if a media file has a different property. If necessary, the user must change the `anamorphic` setting in the sequence by hand to conform to the actual property of the media.

Changes in Final Cut Pro 7.0

- Importing XML with timecode information retains the timecode information, even if the clips created are reconnected to media later, as long as the specified timecode data in the XML does not conflict with the specified tracks in the media file. (If the media file specifies a source timecode, XML should specify an Aux1 to avoid conflict). Any specified sound timecode is now exported.

- The behavior of `addifnotfound` has changed. Specifically, you can now update the contents of a bin without having to specify every object in the bin—see [updatebehavior](#) (page 82).
- Speed handling is improved to be more accurate—see [The anchoroffset Element](#) (page 51). FCP now pins bezier handles so that they cannot cause random, unexpected spikes in the wrong direction.
- A `clipitem` with a speed effect and an overlapping transition now imports properly.
- Using an audio unit effect across localized systems no longer loses parameter settings, provided the new element `parameterspecifier` is used.
- Importing a text string that starts with "true" or "false" now properly imports the string as text, while retaining TRUE and FALSE as Boolean values in other places.
- Audio only angles of multiclips now round-trip (export and import) correctly.
- The multiclip round-trip process is improved in terms of accuracy of edits as well as effects, transition, and marker handling.
- Mishandling of `link - itemindex` now logs an error.
- Improved memory management and other fixes increase the stability of XML export and import.
- FCP import is more strict about the UUID element. The element must have the proper format or it is rejected and an error logged.
- FCP escapes most illegal characters on XML export. Since the FCP Export engine is now completely revised, FCP may escape some of these characters differently.

Document Revision History

This table describes the changes to *Final Cut Pro 7 XML Interchange Format*.

Date	Notes
2009-07-23	Updated for version 5 of the interchange format released with Final Cut Pro 7.0.
2009-02-04	Minor changes to documentation.
2007-11-14	Minor changes for Final Cut Pro 6.0.2
2007-10-10	Minor corrections.
2007-05-21	Updated for version 4 of the interchange format released with Final Cut Pro 6.0.
2006-09-26	Updated for version 3 of the interchange format released with Final Cut Pro 5.1.2.
2004-03-16	Information about inheritance added to Elements Catalog (page 79)
2003-11-14	Updated for release of Final Cut Pro 4.1
2003-06-23	First version for beta 1 release of XML feature



Apple Inc.
Copyright © 2003, 2009 Apple Inc.
All rights reserved.

No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, mechanical, electronic, photocopying, recording, or otherwise, without prior written permission of Apple Inc., with the following exceptions: Any person is hereby authorized to store documentation on a single computer or device for personal use only and to print copies of documentation for personal use provided that the documentation contains Apple's copyright notice.

No licenses, express or implied, are granted with respect to any of the technology described in this document. Apple retains all intellectual property rights associated with the technology described in this document. This document is intended to assist application developers to develop applications only for Apple-branded products.

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

Apple, the Apple logo, Carbon, Cocoa, DVD Studio Pro, Final Cut, Final Cut Pro, Final Cut Studio, Logic, Mac, OS X, and QuickTime are trademarks of Apple Inc., registered in the U.S. and other countries.

DEC is a trademark of Digital Equipment Corporation.

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, ERROR OR INACCURACY IN THIS DOCUMENT, even if advised of the possibility of such damages.

Some jurisdictions do not allow the exclusion of implied warranties or liability, so the above exclusion may not apply to you.

Index

A

activeaudioangle element 88
activevideoangle element 88
alignment element 98
alpha element 102
alphareverse element 84
alphatype element 84
anamorphic element 83
anchoroffset element 104
angle element 88
appmanufacturer element 105
appname element 105
appspecificdata element 104
appversion element 105
audio element 33, 90
Audio Levels 48
audiochannel element 91

B

batch list encoding 59
Bezier curves 161
Bezier handles 45, 46
bin element 80
blue element 102

C

cameraroll element 108
channel element 92
channelcount element 90
channeldescription element 91
channellabel element 91
children element 42, 80

clip element 15, 82
clip encoding 15
clipcommenta element 86
clipcommentb element 86
clipindex element 81
clipitem element 24, 32, 83
codec element 93
codecname element 106
codectypecode element 106
codectypename element 106
codecvendorcode element 106
collapsed element 89
color element 94
colordepth element 93
comment element 95
comments element 86
compositemode element 84
createfcpprojectatxmlfilepath element 112
createnewproject element 111

D

dailyroll element 108
data element 105
datarate element 107
defaultangle element 89
defsequencepresetname element 111
deletethisxmlfileatimport element 112
depth element 93
description element 85
displayformat element 20, 97
displaynonfatalerrors element 112
documents
 errors during import 12

- exporting 9
- importing 10
- validating 113

downmix element 92

DTDs 113

duration element 15, 94

E

effect element 43, 98

effectcategory element 99

effectclass element 99

effectid element 44, 99

effecttype element 45, 99

enabled element 94

encoding

- effects 43

- multiclips 26

- projects 41

end element 15, 33, 87

endoffset element 87

endratio element 100

errors importing documents 12

exporting documents 9, 23

F

FCPCurve interpolation 46

fcpimageprocessing element 31, 105

feetframes element 108

field element 97

fielddominance element 93

file element 18, 19, 83

filmdata element 107

filmslate element 107

filmstandard element 109

filter element 43, 98

filterincludeeffects element 111

filterincludemarkers element 111

filterincludesequencesettings element 112

filterreconnectmediafiles element 111

format element 29, 92

frame element 20, 97

G

gamma element 90

generatoritem element 64, 98

good element 85

green element 102

group element 92

groupindex element 81

H

hadbezierin element 104

hadbezierout element 104

height element 93

horiz element 103

I

id attribute 14, 35, 37, 64

importing documents 10

importoptions element 12, 110

in element 95

inbez element 46, 102

index element 92

inheritance convention 14, 18

inknumber element 109

inscale element 46, 102

interpolation element 46, 103

interpolation method 46, 161

ismasterclip element 17, 84

itemhistory element 55, 82

K

key element 110

keycode element 108

keyframe element 45, 101

keyframe encoding [45](#)
keyframerate element [106](#)

L

label element [86](#)
label2 element [86](#)
labels element [86](#)
labroll element [108](#)
layerindex element [90](#)
layout element [91](#)
link element [35, 81](#)
linkclipref element [37, 81](#)
linking [35](#)
locked element [81](#)
logginginfo element [85](#)
lognote element [85](#)

M

marker element [94](#)
masterclipid element [17, 84](#)
mastercomment element [86](#)
media element [19, 20, 80](#)
mediadelay element [89](#)
mediaSource element [83](#)
mediatype element [45, 95](#)
metadata element [56, 109](#)
mixedratesoffset element [89](#)
multiclip element [88](#)
multiclip encoding [26](#)

N

name element [15, 44, 94](#)
ntsc element [96](#)
numchannels element [92](#)

O

origvalue element [104](#)

out element [95](#)
outbez element [46, 102](#)
outputchannelindex element [81](#)
outputs element [29, 32, 91](#)
outscale element [46, 102](#)

P

parameter element [100](#)
parameter settings [70](#)
parameterid element [45, 100](#)
parameterspecifier element [100](#)
pathurl element [18, 83](#)
perfoffset element [109](#)
pixelaspectratio element [93](#)
prefix element [108](#)
primarytimecode element [98](#)
privatestate element [100](#)
project element [42, 80](#)
project encoding [41](#)
PSD files [40, 88, 90](#)

Q

qtcodec element [31, 106](#)
qteffectid element [107](#)

R

rate element [15, 96](#)
red element [102](#)
reel element [20, 97](#)
rendermode element [105](#)
reverse element [100](#)
reversed element [109](#)

S

samplecharacteristics element [32, 92](#)
samplerate element [94](#)
scene element [85, 108](#)

scenernote element 85
sequence element 29, 80
sequence encoding 29
shotnote element 85
shottake element 85
size element 110
slate element 108
source element 20, 97
sourcechannel element 91
sourcetrack element 87
spatialquality element 106
speedkfstend element 104
speedkfstin element 103
speedkfstout element 103
speedkfststart element 104
speedvirtualkf element 103
start element 15, 33, 87
startoffset element 87
startratio element 99
stillframe element 70, 88
stillframeoffset element 88
storage element 109
string element 20, 96
subclip encoding 24
subclipinfo element 26, 87
subclipmasterid element 84
subframeoffset element 89
subtitle encoding 64
syncoffset element 88
synctype element 89

T

take element 108
takenote element 85
targetprojectname element 111
telecinespeed element 109
temporalquality element 106
timebase element 96
timecode element 19, 20, 96

track element 24, 32, 80
trackcount element 91
trackindex element 95
transitionitem element 38, 43, 98
type element 110

U

updatebehavior element 24, 53, 57, 82
usesuperwhite element 105
useyuv element 105
UUID element 107
uuid element 24, 53, 81

V

validating documents 113
value element 45, 101, 110
valueentry element 101
valuelist element 45, 101
valuemax element 101
valuemin element 101
vert element 103
video element 24, 90

W

when element 102
width element 93
wipeaccuracy element 99
wipecode element 99

X

xmml element 15, 79