

# **DVDBuilder(tm) 2.1.2 for .NET**

## Table of Contents

<b>Release Notes</b>	<b>1</b>
<b>Error Handling</b>	<b>4</b>
<b>DVD Project File</b>	<b>7</b>
<dvd>	7
<videoManager>	8
<menus>	9
<menu>	9
<background>	13
<videoStream>	14
<audioStream>	14
<mask>	15
<display>	16
<selection>	17
<action>	17
<button>	18
<command>	20
<opcode>	22
<preCommands>	25
<command>	28
<opcode>	30
<postCommands>	33
<command>	35
<opcode>	38
<firstPlayCommands>	40
<command>	43
<opcode>	46
<titleSet>	48
<menus>	49
<menu>	49
<background>	53
<videoStream>	54
<audioStream>	55
<mask>	55
<display>	56
<selection>	57
<action>	58

<button>	59
<command>	60
<opcode>	63
<postCommands>	65
<command>	68
<opcode>	71
<preCommands>	73
<command>	76
<opcode>	78
<titles>	81
<title>	81
<videoObject>	84
<audioStream>	86
<videoStream>	87
<subpictureStream>	87
<postCommands>	88
<command>	91
<opcode>	93
<preCommands>	96
<command>	98
<opcode>	101
<subpicturePalette>	103
<color>	104
<audioStreams>	105
<stream>	105
<subpictureStreams>	106
<stream>	107

## **DVD Language Codes 109**

## **Symbol Reference 112**

<b>PrimoSoftware Namespace</b>	<b>112</b>
DVDBuilder Namespace	112
Interfaces	114
DataStream	114
DataStream.Close	115
DataStream.GetSize	115
DataStream.Open	116
DataStream.Read	116
DataStream.Seek	117
DataStream.Write	117

DVDBuilder	117
DVDBuilder.Error	119
DVDBuilder.ErrorFacility	120
DVDBuilder.ErrorHint	120
DVDBuilder.InputDataStreamFactory	120
DVDBuilder.OnContinue	121
DVDBuilder.OnProgress	121
DVDBuilder.OnStatus	121
DVDBuilder.OutputFolder	122
DVDBuilder.Project	122
DVDBuilder.ProjectFile	122
DVDBuilder.SystemErrorDomain	122
DVDBuilder.Build	123
InputDataStreamFactory	123
InputDataStreamFactory.Create	124
InputDataStreamFactory.Destroy	124
SubpictureEncoder	124
SubpictureEncoder.Error	125
SubpictureEncoder.ErrorFacility	126
SubpictureEncoder.ErrorHint	126
SubpictureEncoder.SystemErrorDomain	126
SubpictureEncoder.Encode	127
SubpictureEncoder.GetImageMaskColor	127
SubpictureEncoder.SetImageMaskColor	127
Classes	127
DVDBuilder_ProgressEventArgs	128
DVDBuilder_ProgressEventArgs.Percent	128
DVDBuilder_StatusEventArgs	129
DVDBuilder_StatusEventArgs.Status	129
Library	130
Library.CreateDVDBuilder	130
Library.CreateVideoRecorder	130
Library.CreateVideoRecorder	131
Library.Initialize	131
Library.SetLicenseInfo	131
Library.Terminate	132
VR Namespace	132
Interfaces	133
DeviceError	134
DeviceError.Error	135
DeviceError.ErrorFacility	135
DeviceError.SystemErrorDomain	136

FileSystemDeviceConfig	136
FileSystemDeviceConfig.Folder	136
OpticalDiscDeviceConfig	137
OpticalDiscDeviceConfig.DeviceHandle	137
OpticalDiscDeviceConfig.DriveLetter	138
OpticalDiscDeviceConfig.VolumeLabel	138
Title	138
Title.Duration	139
TitleEnumerator	139
TitleEnumerator.Count	140
TitleEnumerator.Item	140
VideoRecorder	140
VideoRecorder.AllowMuxBufferUnderflow	143
VideoRecorder.AverageBitrate	144
VideoRecorder.Devices	144
VideoRecorder.Error	144
VideoRecorder.ErrorFacility	145
VideoRecorder.IsFinalized	145
VideoRecorder.IsFinalizeSupported	145
VideoRecorder.MediaFreeSpace	146
VideoRecorder.SystemErrorDomain	146
VideoRecorder.FinalizeMedia	146
VideoRecorder.GetDeviceError	147
VideoRecorder.GetIsFinalized	147
VideoRecorder.GetMediaFreeSpace	148
VideoRecorder.GetTitles	149
VideoRecorder.LoadDevicePlugin	149
VideoRecorder.PrepareMedia	149
VideoRecorder.Start	150
VideoRecorder.StartAsync	150
VideoRecorder.Stop	151
VideoRecorder.Write	151
VRDevice	152
VRDevice.Config	154
VRDevice.Error	154
VRDevice.ErrorFacility	154
VRDevice.MediaFreeSpace	155
VRDevice.MedialsBlank	155
VRDevice.MedialsReWritable	155
VRDevice.SystemErrorDomain	156
VRDevice.Type	156
VRDevice.EraseMedia	156

VRDevice.Initialize	157
VRDevice.NotifyOSFileSystemChanged	157
VRDeviceList	157
VRDeviceList.Count	158
VRDeviceList.Item	158
VRDeviceList.Add	159
VRDeviceList.Clear	159
VRDeviceList.RemoveAt	159
VRDevicePlugin	160
VRDevicePlugin.Error	161
VRDevicePlugin.ErrorFacility	161
VRDevicePlugin.SystemErrorDomain	161
VRDevicePlugin.CreateFileSystemDevice	162
VRDevicePlugin.CreateOpticalDiscDevice	162
VRDevicePlugin.GetAPIRevision	163
Structs, Records, Enums	163
VideoRecorderError	163
VRDeviceError	165
VRDevicePluginError	166
VRDeviceType	166
Structs, Records, Enums	167
DVDBuilderError	167
DVDBuilderStatus	177
ErrorFacility	177
SubpictureEncoderError	178
SubpictureEncoderImageMaskColor	179
Types	179
DVDBuilder_ContinueEventHandler	180
DVDBuilder_ProgressEventHandler	181
DVDBuilder_StatusEventHandler	181

## Index

**a**

# DVDBuilder(tm) 2.1.2 for .NET

## 1 Release Notes

### Dependencies, Restrictions and Limitations

#### **C/C++ Runtime (CRT) Dependencies**

The DVDBuilder.NET assembly requires a particular version of the C/C++ runtime libraries. The following files: [DVDBuilder.NET.3.0.dll](#); [DVDBuilder.NET.3.0.x64.dll](#); [DVDBuilder.NET.3.5.dll](#); [DVDBuilder.NET.3.5.x64.dll](#) use Microsoft.VC90.CRT, version 9.0.30729.4148.

For more information and download links, please go to:

<http://www.microsoft.com/downloads/details.aspx?familyid=2051A0C1-C9B5-4B0A-A8F5-770A549FD78C&displaylang=en>

The following files: [DVDBuilder.NET.2.0.dll](#); [DVDBuilder.NET.2.0.x64.dll](#) use Microsoft.VC80.CRT, version 8.0.50727.4053.

For more information and download links, please go to:

<http://www.microsoft.com/downloads/details.aspx?familyid=766A6AF7-EC73-40FF-B072-9112BAB119C2&displaylang=en>

The DVDBuilder SDK setup package (MSI) already includes and installs the appropriate CRT. However when the DVDBuilder.NET assembly is deployed via different means (not through the DVDBuilder SDK setup), it has to be ensured that the correct version of the CRT is present on the target machine. One way to achieve this is to use the VC++ redistributives linked above ([vcredist\\_x86.exe](#) and [vcredist\\_x64.exe](#)). They contain the CRT version used by DVDBuilder.NET.

#### **Version 2.1.2**

##### New

- 1775: Added VM commands support.

#### **Version 2.1.1**

##### New

- 1761: SubpictureEncoder interface for producing DVD subtitle streams from STL subtitle format.
- 1762: Elementary audio and video streams, and subpicture streams can be used as input in DVDBuilder XML projects

**Version 2.0.8**New

- 1683: Added VideoDVDCreator sample

**Version 2.0.7**New

- 1615: Add DVDB version to IFO developer ID string
- 1610: Update DVDBuilder to use PrimoMpeg 1.0.6

Fixed

- 1612: VideoRecorder x64 sample does not start
- 1611: VideoRecorder.Write could fail with multiplexer error on a valid MPEG-2 stream.

**Version 2.0.6**Fixed

- 1545: Ambiguous system error returned by DVDBuilder
- 1534: Show system error message in DVDAuthorCmd.NET sample
- 1519: Handle bitmap color table for 8-bit bitmaps
- 1517: Support MPEG2 files without sequence end code
- 1509: Show the error hint in DVDAuthorCmd.NET

**Version 2.0.5**New

- 1496: DVDBuilder output successfully passes DVD-Video compatibility tests
- 1495: Added AllowMuxBufferUnderflow property to VideoRecorder
- 1461: VideoRecorder can now record DVD-video to Blu-ray Disc
- 1439: VideoRecorder optical plug-in now uses PrimoBurner 3.0.4

**Version 2.0.4**New

- Ability to record MPEG-2 program stream to a DVD disc (DVD-R, DVD+R, DVD-RW, DVD+RW, DVD-RAM) in real time.
- The DVDBuilder interface provides detailed error codes and hints when a DVD authoring project fails to build.

**Version 1.14.0**New

- <videoObject> XML Element: Added 'aspectRatio', 'resolution' and 'frameRate' attributes. This feature should be used with caution. It was added to allow input from encoders which generate wrong MPEG-2 headers.
- <title> XML Element: The 'postNavigate' attribute is now able to reference titles from different title sets. This

allows navigation across title sets.

- Ability to map MPEG audio and subtitle streams to DVD language codes.
- Ability to ready video object data from a stream provided by the application. To use this functionality your application must implement the `DataStream` and `InputDataStreamFactory` interfaces.

# 2 Error Handling

## Error State

Most operations in DVDBuilder produce a result that tells whether the operation has succeeded or not and also gives the reason for the failure. The operation result, also called the **error state** is accessible through the following 3 properties:

### Error, ErrorFacility and SystemErrorDomain.

A number of interfaces use these methods to expose the error state of an object.

In general the error state represents the result of the last operation performed by the object. The error state is updated after every operation.

### ErrorFacility Property

Gets the source of the last error. This value is indicative about the success of the last operation. If it is ErrorFacility.Success then the last operation has fully succeeded and the values returned by Error and SystemErrorDomain should be ignored. Any other value than ErrorFacility.Success means that the last operation has failed in some way.

If ErrorFacility is ErrorFacility.System then SystemErrorDomain may be necessary to interpret the error code returned by GetError.

**See** ErrorFacility (see page 177)

### Error Property

Returns the error code of the last operation. By itself the error code does not imply the success or the failure of the operation and should be ignored when ErrorFacility is ErrorFacility.Success.

The error code should be interpreted only when the failed component (error facility) is recognized. DVDBuilder provides the error codes for the built-in error facilities: ErrorFacility.VideoRecorder, ErrorFacility.VRDevice and ErrorFacility.VRDevicePlugin. However some errors may be generated by external components like the PrimoBurner Engine or by the operating system. In these cases additional information is required in order to interpret the error code.

Interpret the Error code as	When ErrorFacility is
VideoRecorderError (see page 163)	ErrorFacility.VideoRecorder
VRDeviceError (see page 165)	ErrorFacility.VRDevice
VRDevicePluginError (see page 166)	ErrorFacility.VRDevicePlugin
Operating System error. Refer to the OS documentation.	ErrorFacility.System
PrimoBurner engine error. Refer to PrimoBurner documentation.	ErrorFacility.PBEngine
External component error	A recognized external component with available documentation.

### SystemErrorDomain Property

Gets the error source of the last operation when there's a system error. This source may be necessary to interpret the value of Error when ErrorFacility is ErrorFacility.System, otherwise it should be ignored.

For Windows this value is always 0.

## Dealing with errors in parallel video recorder operations.

A number of video recorder operations are performed concurrently (in parallel) to all attached devices. These operations are:

VideoRecorder.PrepareMedia (see page 149), VideoRecorder.Start (see page 150), VideoRecorder.StartAsync (see page 150), VideoRecorder.Write (see page 151), VideoRecorder.Stop (see page 151) and VideoRecorder.FinalizeMedia (see page 146). All of these methods (with the exception of StartAsync) block while the operation is executed. They return *true* when the operation has succeeded for all attached devices and *false* when any of the attached devices has failed. When a parallel operation fails with a *DeviceError* (VideoRecorder.ErrorFacility (see page 145) == ErrorFacility.VideoRecorder and VideoRecorder.Error (see page 144) == VideoRecorderError.DeviceError) the error state of each device may be obtained by using VideoRecorder.GetDeviceError (see page 147). The device error state can be analyzed by using the properties Error, ErrorFacility and SystemErrorDomain as described already in this topic. When the video recorder writes to the attached devices (VideoRecorder.Write (see page 151)) it ignores the already failed devices, therefore their error state remains unchanged in the subsequent Write calls.

## A sample case of recording to several devices.

Consider that a video should be recorded concurrently to 7 devices, named *d0*, *d1*, ... and *d6* for the purpose of this sample. All possible steps involved in the recording will be discussed.

**1. VideoRecorder.PrepareMedia (see page 149)** - this step is optional but if used it can generate a *DeviceError* (described above).

Device *d0* fails during this operation and the other devices succeed.

PrepareMedia returns *false* and the *DeviceError* is set in the video recorder. The application enumerates the device error state and finds the failed device.

1.1. The application aborts the recording process and informs the user about the failed device. No calls to the DVDBuilder API are necessary in this case.

or

1.2. The application decides to start the recording process and only informs the user about the failed device. It is advisable to remove the failed device from the device list especially before calling the synchronous Start method. This guarantees that the video recording will start as soon as possible.

Either Start or StartAsync but the start step is mandatory.

### 2.1. VideoRecorder.Start (see page 150)

Device *d1* fails during this operation. So does device *d0* if it is still in the device list.

2.1.1. Upon seeing the failed devices the application stops the recording by calling the VideoRecorder.Stop (see page 151) method.

or

2.1.2. The application reports the failed devices and starts calling the VideoRecorder.Write (see page 151) method repeatedly.

### 2.2. VideoRecorder.StartAsync (see page 150)

Device *d1* fails during the asynchronous start but the application finds this only in one of the subsequent VideoRecorder.Write (see page 151) method calls. This is true for device *d0* if it is still in the device list.

2.2.1. `StartAsync` returns *false* and the application aborts the recording process because there's a logical and unrecoverable error. No further calls to the DVDBuilder API are necessary in this case.

or

2.2.2. `StartAsync` returns *true* and the application starts calling the `VideoRecorder.Write` (see page 151) method repeatedly.

**3. `VideoRecorder.Write` (see page 151)** - this step is mandatory and may be called many times to record the incoming video.

Initially the video recorder uses only devices from *d2* to *d6*.

(writing...)

Device *d2* fails at some point because the DVD disc is defective.

3.1. The application stops the recording process by calling the `VideoRecorder.Stop` (see page 151) method.

or

3.2. The application reports the failed devices so far (*d0*, *d1* and *d2*) and continues writing.

(writing...)

Device *d3* fails at some point because the DVD disc has run out of space.

The application either stops the recording or continues to record to devices *d4*, *d5* and *d6*.

(writing...)

Device *d4* fails at some point because the DVD disc has run out of space.

The application decides to stop the recording instead of trying to record only to *d5* and *d6*.

**4. `VideoRecorder.Stop` (see page 151)** - this step is mandatory whenever the `VideoRecorder.Start` (see page 150) method is called or the `VideoRecorder.StartAsync` (see page 150) method succeeds.

During this step the video recorder tries to stop devices *d3*, *d4*, *d5* and *d6*. The first three devices (*d0*, *d1* and *d2*) are considered unreliable because they have truly failed, not only run out of space as *d3* and *d4*, therefore their error state remains unchanged after this step.

Device *d4* fails during this step. `Stop` returns *false* and the `DeviceError` is set in the video recorder. The application enumerates the device error state and finds out that the only successfully recorded devices at this point are *d3*, *d5* and *d6*.

**5. `VideoRecorder.FinalizeMedia` (see page 146)** - this step should be called if no additional recording will be appended.

During this step device *d5* fails. `FinalizeMedia` returns *false* and the `DeviceError` is set in the video recorder.

The application enumerates the device error state and finds out that devices *d3* and *d6* have no errors and therefore are successfully finalized.

After the whole process is completed only the video content in devices *d3* and *d6* is guaranteed to be played back in a standalone DVD player. However the video in *d3* will be shorter as it has run out of space during the recording.

## 3 DVD Project File

DVD configuration file is a standard XML file. DVDBuilder defines a set of elements that implement DVD configuration settings. This section describes the schema for the DVD configuration file. You need to be familiar with XML to edit the configuration files. XML tags and attributes are case-sensitive.

### 3.1 <dvd> Element

#### Description

The root element in every configuration file used by DVDBuilder.

```
<dvd
  version="1.1.0.1"
  xmlns="http://www.primosoftware.com/dvdbuilder">
  <videoManager>
  <titleSet>
</dvd>
```

#### Required Attributes

Attribute	Description
version	Runtime version. Must be set to 1.1.0.1 or 2.1.1.1
xmlns	XML namespace. Must be set to "http://www.primosoftware.com/dvdbuilder"

#### Child Elements

Element	Description
<videoManager>	Configures the Video Manager (VMG).
<titleSet>	Configures a Video Title Set (VTS). There can be up to 99 VTS per DVD disc.

#### Remarks

A DVD compliant disc contains one Video Manager and multiple Video Title Sets.

#### Example

The following example shows a simple DVD configuration file.

```
<?xml version="1.0" encoding="utf-8"?>
<dvd version="1.1.0.1" xmlns="http://www.primosoftware.com/dvdbuilder">
  <videoManager firstPlayNavigate="Title=1"/>
  <titleSet>
    <titles>
```

```

<title id="1" chapters="00:00:00;">
  <videoObject file="1.mpg"/>
</title>
</titles>
</titleSet>
</dvd>

```

## 3.1.1 <videoManager> Element

Configures the DVD video manager.

```

<videoManager
  firstPlayNavigate="Menu=id;|Title=id;">
  <menus>
  <firstPlayCommands>
</videoManager/>

```

### Required Attributes

Attribute	Option	Description
firstPlayNavigate		This is the command that a DVD player executes when a disc is loaded into the DVD player.
	Menu=id;	Navigates to a video manager menu or a titleset root menu with ID equal to "id". The DVD player will automatically display the selected menu. Example: Menu=50001;
	Title=id;	Navigates to a Title with ID equal to "id". The DVD player will automatically play the first chapter of the selected title. Example: Title=1;

### Child Elements

Element	Description
<menus>	Configures the menus of the DVD video manager.
<firstPlayCommands>	Specifies the commands to be executed when the DVD player starts.

### Remarks

If <firstPlayCommands>child element is specified then the attribute "firstPlayNavigate" is not required. If both <firstPlayCommands>child element and attribute "firstPlayNavigate" are specified then the program will use only the <command>child element.

### Example

The following example shows <videoManager> configuration that will play the title with ID=1 when a disc is loaded into a DVD player.

```

<?xml version="1.0" encoding="utf-8"?>
<dvd version="1.1.0.1" xmlns="http://www.primosoftware.com/dvdbuilder">
  <videoManager firstPlayNavigate="Title=1"/>
  <titleSet>
  <titles>

```

```

    <title id="1" chapters="00:00:00;">
      <videoObject file="1.mpg"/>
    </title>
  </titles>
</titleSet>
</dvd>

```

### 3.1.1.1 <menus> Element

Defines Video Manager (VMG) or Video Title Set (VTS) menus.

```

<menus>
  <menu>
</menus>

```

#### Child Elements

Element	Description
<menu>	Adds a Video Manager (VMG) or Video Title Set (VTS) menu.

#### 3.1.1.1.1 <menu> Element

##### Description

Defines a Video Manager (VMG) or Video Title Set (VTS) menu.

```

<menu id="num" entry="title|root">
  <background>
  <button>
  <mask>
  <display>
  <selection>
  <action>
</menu>

```

##### Required Attributes

Attribute	Option	Description
id	1 to 99999	Menu unique identifier.

##### Optional Attributes

Attribute	Option	Description
entry		Specifies the menu type.

	title	<p>Indicates that the menu is the DVD Title menu.</p> <p>The title menu is used as a top-level menu through which the user can access root menus from other VTS on the disc, submenus in the VMG, and video titles available in video title sets.</p> <p>The title menu is displayed when the "Title" button on the remote control is pressed.</p> <p>In general, DVDs can have one title menu for each language unit defined in the VMG. DVDBuilder supports only the default "Unspecified" language unit.</p> <p>The "Unspecified" language unit is selected by the DVD player regardless of the DVD Player own language settings. The Title menu in this case can branch to a language selection menu and then to appropriate submenus of each supported language or to other VTS menus such as root, subpicture and angle menus.</p>
	root	<p>Indicates that the menu is a Video Title Set (VTS) root menu.</p> <p>The Root menu is used as a top-level entry point for other menus in the VTS.</p>

**Child Elements**

Element	Description
<background>	Specifies a video clip to be used for the menu background.
<mask>	Specifies a picture mask layer for the menu buttons.
<display>	Specifies the mask patterns colors that are used when the menu buttons are in unselected state.
<selection>	Specifies the mask patterns colors that are used when the menu buttons are in selected state.
<action>	Specifies the mask patterns colors that are used when the menu buttons are in depressed state.
<button>	Defines a menu button (highlight) position and navigation command. There can be up to 36 buttons per one menu.
<preCommands> (Version 2.1.2)	Specifies the VM commands to be executed before a menu is presented.
<postCommands> (Version 2.1.2)	Specifies the VM commands to be executed after a menu is presented.

**Example**

The following example shows a DVD with one VMG title menu.

```
<?xml version="1.0" encoding="utf-8"?>
<dvd version="1.1.0.1" xmlns="http://www.primosoftware.com/dvdbuilder">
  <videoManager firstPlayNavigate="Menu=50001;">
    <menus>
```



```

<menu id="50001" entry="title">
  <background file="menu_50001.mpg" />
  <mask
    file="menu_50001_mask.bmp"
    patternColor="#000000"
    backgroundColor="#FFFFFF"
    emphasisColor1="#0000FF"
    emphasisColor2="#00FF00" />

  <display
    patternColor    ="#FF0000" patternContrast    ="F"
    backgroundColor="#000000" backgroundContrast="0"
    emphasis1Color  ="#000000" emphasis1Contrast  ="0"
    emphasis2Color  ="#000000" emphasis2Contrast  ="0"/>

  <selection
    patternColor    ="#0000FF" patternContrast    ="F"
    backgroundColor="#000000" backgroundContrast="0"
    emphasis1Color  ="#000000" emphasis1Contrast  ="0"
    emphasis2Color  ="#000000" emphasis2Contrast  ="0"/>

  <action
    patternColor    ="#00FF00" patternContrast    ="F"
    backgroundColor="#000000" backgroundContrast="0"
    emphasis2Color  ="#000000" emphasis2Contrast  ="0"
    emphasis1Color  ="#000000" emphasis1Contrast  ="0"/>

  <button left="10" top="10" width="74" height="51" navigate="Title=1;" />
</menu>
</menus>
</videoManager>
<titleSet>
  <titles>
    <title id="1" chapters="00:00:00;00:01:00">
      <videoObject file="1.mpg"/>
    </title>
  </titles>
</titleSet>
</dvd>

```

The following example shows a DVD with one VMG menu (ID =1). The VMG menu has one button that navigates to a VTS root menu (ID=101). The VTS root menu has one button which navigates to another VTS menu (ID=10101) which has two buttons. Each button command specifies that a chapter from Title with ID =1 should be played. The title itself has a postNavigate command equal to "Menu=101;". That will instruct the DVD Player to display the VTS root menu (ID=101) when the title playback is completed.

```

<?xml version="1.0" encoding="utf-8"?>
<dvd version="1.1.0.1" xmlns="http://www.primosoftware.com/dvdbuilder">
  <videoManager firstPlayNavigate="Menu=1;">
    <menus>
      <menu id="1" entry="title">
        <background file="vmg_menu_background.mpg" />

        <mask file="vmg_menu_mask.bmp"
          patternColor    ="#000000"
          backgroundColor ="#FFFFFF"
          emphasisColor1  ="#00FF00"
          emphasisColor2  ="#0000FF" />

        <display
          patternColor    ="#000000" patternContrast    ="0"
          backgroundColor ="#000000" backgroundContrast ="0"
          emphasis1Color  ="#000000" emphasis1Contrast  ="0"
          emphasis2Color  ="#000000" emphasis2Contrast  ="0" />

        <selection
          patternColor    ="#0000FF" patternContrast    ="15"
          backgroundColor ="#000000" backgroundContrast ="0"
          emphasis1Color  ="#000000" emphasis1Contrast  ="0"
          emphasis2Color  ="#000000" emphasis2Contrast  ="0" />

```

```

    <action
      patternColor      = "#FF0000" patternContrast    = "15"
      backgroundColor   = "#000000" backgroundContrast = "0"
      emphasis1Color    = "#000000" emphasis1Contrast  = "0"
      emphasis2Color    = "#000000" emphasis2Contrast  = "0" />

    <button left="15" top="40" width="112" height="76" navigate="Menu=101;" />
  </menu>
</menus>
</videoManager>
<titleSet>
  <menus>
    <menu id="101" entry="root">
      <background file="vts_root_menu_background.mpg" />

      <mask file="vts_root_menu_mask.bmp"
        patternColor      = "#000000"
        backgroundColor   = "#FFFFFF"
        emphasisColor1    = "#00FF00"
        emphasisColor2    = "#0000FF" />

      <display
        patternColor      = "#000000" patternContrast    = "0"
        backgroundColor   = "#000000" backgroundContrast = "0"
        emphasis1Color    = "#000000" emphasis1Contrast  = "0"
        emphasis2Color    = "#000000" emphasis2Contrast  = "0" />

      <selection
        patternColor      = "#0000FF" patternContrast    = "15"
        backgroundColor   = "#000000" backgroundContrast = "0"
        emphasis1Color    = "#000000" emphasis1Contrast  = "0"
        emphasis2Color    = "#000000" emphasis2Contrast  = "0" />

      <action
        patternColor      = "#FF0000" patternContrast    = "15"
        backgroundColor   = "#000000" backgroundContrast = "0"
        emphasis1Color    = "#000000" emphasis1Contrast  = "0"
        emphasis2Color    = "#000000" emphasis2Contrast  = "0" />

      <button left="15" top="40" width="70" height="50" navigate="Menu=10101;" />
    </menu>
    <menu id="10101">
      <background file="title1_menu_background.mpg" />

      <mask file="title1_menu_mask.bmp"
        patternColor      = "#000000"
        backgroundColor   = "#FFFFFF"
        emphasisColor1    = "#00FF00"
        emphasisColor2    = "#0000FF" />

      <display
        patternColor      = "#000000" patternContrast    = "0"
        backgroundColor   = "#000000" backgroundContrast = "0"
        emphasis1Color    = "#000000" emphasis1Contrast  = "0"
        emphasis2Color    = "#000000" emphasis2Contrast  = "0" />

      <selection
        patternColor      = "#0000FF" patternContrast    = "15"
        backgroundColor   = "#000000" backgroundContrast = "0"
        emphasis1Color    = "#000000" emphasis1Contrast  = "0"
        emphasis2Color    = "#000000" emphasis2Contrast  = "0" />

      <action
        patternColor      = "#FF0000" patternContrast    = "15"
        backgroundColor   = "#000000" backgroundContrast = "0"
        emphasis1Color    = "#000000" emphasis1Contrast  = "0"
        emphasis2Color    = "#000000" emphasis2Contrast  = "0" />

      <button left="11" top="36" width="78" height="58" navigate="Title = 1; Chapter = 1"
    />
      <button left="96" top="36" width="78" height="58" navigate="Title = 1; Chapter = 2"
    />
  </menus>
</titleSet>

```

```

    </menu>
  </menus>
  <titles>
    <title id="1" postNavigate="Menu=101;" chapters="00:00:00;00:01:00;">
      <videoObject file="1.mpg" pause="0"/>
    </title>
  </titles>
</titleSet>
</dvd>

```

### 3.1.1.1.1.1 <background> Element

#### Description

Specifies a video clip to be used for the menu background.

```

<background
  file="file" />

```

#### Required Attributes

Attribute	Description
file	Specifies a file that contains an MPEG-2 video to be used as the background of a menu. It can be either a still or motion video.

#### Child Elements

Element	Description
<audioStream> (Version 2.1.1.1)	Specifies an audio elementary stream.
<videoAstream> (Version 2.1.1.1)	Specifies a video elementary stream.

#### Example

The following example shows a DVD with one VMG title menu.

```

<?xml version="1.0" encoding="utf-8"?>
<dvd version="1.1.0.1" xmlns="http://www.primosoftware.com/dvdbuilder">
  <videoManager firstPlayNavigate="Menu=50001;">
    <menus>
      <menu id="50001" entry="title">
        <background file="menu_50001.mpg" />
        <mask
          file="menu_50001_mask.bmp"
          patternColor="#000000"
          backgroundColor="#FFFFFF"
          emphasisColor1="#0000FF"
          emphasisColor2="#00FF00" />

        <display
          patternColor  ="#FF0000" patternContrast  ="F"
          backgroundColor="#000000" backgroundContrast="0"
          emphasis1Color ="#000000" emphasis1Contrast ="0"
          emphasis2Color ="#000000" emphasis2Contrast ="0"/>

        <selection
          patternColor  ="#0000FF" patternContrast  ="F"
          backgroundColor="#000000" backgroundContrast="0"
          emphasis1Color ="#000000" emphasis1Contrast ="0"
          emphasis2Color ="#000000" emphasis2Contrast ="0"/>
      </menu>
    </menus>
  </videoManager>
</dvd>

```

```

    <action
      patternColor    ="#00FF00" patternContrast    ="F"
      backgroundColor="#000000" backgroundContrast="0"
      emphasis2Color  ="#000000" emphasis2Contrast ="0"
      emphasis1Color  ="#000000" emphasis1Contrast ="0"/>

    <button left="10" top="10" width="74" height="51" navigate="Title=1;" />
  </menu>
</menus>
</videoManager>
<titleSet>
  <titles>
    <title id="1" chapters="00:00:00;00:01:00">
      <videoObject file="1.mpg"/>
    </title>
  </titles>
</titleSet>
</dvd>

```

#### 3.1.1.1.1.1 <videoStream> Element

Specifies a video stream.

```

<videoStream
  file="file" />

```

##### Required Attributes

Attribute	Description
file	Specifies a MPEG-2 video elementary stream file.

##### Example

The following example shows a DVD with one movie. Note that the version attribute in the <dvd> tag is set to 2.1.1.1, because the <audioStream> and <videoStream> tags require DVDBuilder 2.1.1.1 or later.

```

<?xml version="1.0" encoding="utf-8"?>
<dvd version="2.1.1.1" xmlns="http://www.primosoftware.com/dvdbuilder">
  <videoManager firstPlayNavigate="Title=1;" />
  <titleSet>

    <audioStreams>
      <stream languageCode="EN" mpegStreamID="0xC0" mpegSubstreamID="0x00" />
    </audioStreams>

    <titles>
      <title id="1" chapters="00:00:00">

        <videoObject>
          <audioStream file="movie1.mpa" format="MPA" />
          <videoStream file="movie1.mpv" />
        </videoObject>

      </title>
    </titles>
  </titleSet>
</dvd>

```

##### Summary

#### 3.1.1.1.1.2 <audioStream> Element

Specifies an audio stream.

```

<audioStream
  file="file" format="format" />

```

##### Required Attributes

Attribute	Description
file	Specifies an audio elementary stream file.
format	Specifies the audio elementary stream format. It must be AC3, MPA or DTS.

### Example

The following example shows a DVD with one movie. Note that the version attribute in the <dvd> tag is set to 2.1.1.1, because the <audioStream> and <videoStream> tags require DVDBuilder 2.1.1.1 or later.

```
<?xml version="1.0" encoding="utf-8"?>
<dvd version="2.1.1.1" xmlns="http://www.primosoftware.com/dvdbuilder">
  <videoManager firstPlayNavigate="Title=1;" />
  <titleSet>

    <audioStreams>
      <stream languageCode="EN" mpegStreamID="0xC0" mpegSubstreamID="0x00" />
    </audioStreams>

    <titles>
      <title id="1" chapters="00:00:00">

        <videoObject>
          <audioStream file="movie1.mpa" format="MPA" />
          <videoStream file="movie1.mpv" />
        </videoObject>

      </title>
    </titles>
  </titleSet>
</dvd>
```

### Summary

#### 3.1.1.1.1.2 <mask> Element

### Description

Specifies a picture mask layer for the menu buttons.

```
<mask
  file="file"
  patternColor = "#ffffff"
  backgroundColor = "#ffffff"
  emphasisColor1 = "#ffffff"
  emphasisColor2 = "#ffffff" />
```

### Required Attributes

Attribute	Description
file	Specifies a file that contains a 24-bit Color Windows Bitmap image. The bitmap image will be used as the mask layer.
patternColor	The color used for the mask primary pattern. This is an RGB color value expressed in hexadecimal format ("#ffffff").
backgroundColor	Mask background color. This is an RGB color value expressed in hexadecimal format ("#ffffff").
emphasisColor1	Mask first emphasis color. This is usually used to create an inner edge for the buttons frame. This is an RGB color value expressed in hexadecimal format ("#ffffff").

emphasisColor2	Mask second emphasis color. This is usually used to create an outer edge for the buttons frame. This is an RGB color value expressed in hexadecimal format ("#ffffff").
----------------	---

**Remarks**

Picture mask layer is usually a black-and-white or up to 4-color background image. In simple black-and-white masks, the background color is the white area of the image and the pattern color is the black patterns of the image. Additionally, in 4-color masks, there can be also the two emphasis colors.

It is important to know that the <mask> colors are used by the DVD player only to see the different patterns in the mask. The actual display color is determined by color and contrast settings in the <display>, <selection> and <action> elements.

**3.1.1.1.1.3 <display> Element****Description**

Specifies the mask patterns colors that are used when the menu buttons are in unselected state.

```
<display
  patternColor = "#ffffff"
  patternContrast = "num"
  backgroundColor = "#ffffff"
  backgroundContrast = "num"
  emphasis1Color = "#ffffff"
  emphasis1Contrast = "num"
  emphasis2Color = "#ffffff"
  emphasis2Contrast = "num" />
```

**Required Attributes**

Attribute	Option	Description
patternColor	This is an RGB color value expressed in hexadecimal format ("#ffffff").	Primary pattern color.
patternContrast	0 to 15	Contrast of the primary pattern. It is ranged from 0 to 15, 0 being totally transparent and 15 being solid.
backgroundColor	This is an RGB color value expressed in hexadecimal format ("#ffffff").	Background color.
backgroundContrast	0 to 15	Contrast of the background. It is ranged from 0 to 15, 0 being totally transparent and 15 being solid.
emphasisColor1	This is an RGB color value expressed in hexadecimal format ("#ffffff").	First emphasis pattern color.
emphasis1Contrast	0 to 15	Contrast of the first emphasis pattern. It is ranged from 0 to 15, 0 being totally transparent and 15 being solid.
emphasisColor2	This is an RGB color value expressed in hexadecimal format ("#ffffff").	Second emphasis pattern color.

emphasis2Contrast	0 to 15	Contrast of the second emphasis pattern. It is ranged from 0 to 15, 0 being totally transparent and 15 being solid.
-------------------	---------	---

### 3.1.1.1.1.4 <selection> Element

#### Description

Specifies the mask patterns colors that are used when the menu buttons are in selected state.

```
<selection
  patternColor = "#ffffff"
  patternContrast = "num"
  backgroundColor = "#ffffff"
  backgroundContrast = "num"
  emphasis1Color = "#ffffff"
  emphasis1Contrast = "num"
  emphasis2Color = "#ffffff"
  emphasis2Contrast = "num"/>
```

#### Required Attributes

Attribute	Option	Description
patternColor	This is an RGB color value expressed in hexadecimal format ("#ffffff").	Primary pattern color.
patternContrast	0 to 15	Contrast of the primary pattern. It is ranged from 0 to 15, 0 being totally transparent and 15 being solid.
backgroundColor	This is an RGB color value expressed in hexadecimal format ("#ffffff").	Background color.
backgroundContrast	0 to 15	Contrast of the background. It is ranged from 0 to 15, 0 being totally transparent and 15 being solid.
emphasisColor1	This is an RGB color value expressed in hexadecimal format ("#ffffff").	First emphasis pattern color.
emphasis1Contrast	0 to 15	Contrast of the first emphasis pattern. It is ranged from 0 to 15, 0 being totally transparent and 15 being solid.
emphasisColor2	This is an RGB color value expressed in hexadecimal format ("#ffffff").	Second emphasis pattern color.
emphasis2Contrast	0 to 15	Contrast of the second emphasis pattern. It is ranged from 0 to 15, 0 being totally transparent and 15 being solid.

### 3.1.1.1.1.5 <action> Element

#### Description

Specifies the mask patterns colors that are used when the menu buttons are in depressed state.

```
<action
  patternColor = "#ffffff"
```

```

patternContrast    = "num"
backgroundColor    = "#ffffff"
backgroundContrast = "num"
emphasis1Color    = "#ffffff"
emphasis1Contrast = "num"
emphasis2Color    = "#ffffff"
emphasis2Contrast = "num" />
    
```

**Required Attributes**

Attribute	Option	Description
patternColor	This is an RGB color value expressed in hexadecimal format ("#ffffff").	Primary pattern color.
patternContrast	0 to 15	Contrast of the primary pattern. It is ranged from 0 to 15, 0 being totally transparent and 15 being solid.
backgroundColor	This is an RGB color value expressed in hexadecimal format ("#ffffff").	Background color.
backgroundContrast	0 to 15	Contrast of the background. It is ranged from 0 to 15, 0 being totally transparent and 15 being solid.
emphasisColor1	This is an RGB color value expressed in hexadecimal format ("#ffffff").	First emphasis pattern color.
emphasis1Contrast	0 to 15	Contrast of the first emphasis pattern. It is ranged from 0 to 15, 0 being totally transparent and 15 being solid.
emphasisColor2	This is an RGB color value expressed in hexadecimal format ("#ffffff").	Second emphasis pattern color.
emphasis2Contrast	0 to 15	Contrast of the second emphasis pattern. It is ranged from 0 to 15, 0 being totally transparent and 15 being solid.



**3.1.1.1.1.6 <button> Element**

**Description**

Defines a highlight area for a menu button.

```

<button
  left="num"
  top="num"
  width="num"
  height="num"
  navigate="Menu=id;|Title=id;|Title=id; Chapter=num;" />
    
```

**Required Attributes**

Attribute	Option	Description
left	1 to 720 (NTSC) 1 to 720 (PAL)	Specifies the left position of the button rectangle.

top	1 to 480 (NTSC) 1 to 576 (PAL)	Specifies the top position of the button rectangle.
width	1 to 720 (NTSC) 1 to 720 (PAL)	Specifies the width of the button rectangle.
height	1 to 480 (NTSC) 1 to 576 (PAL)	Specifies the height of the button rectangle.
navigate	Menu=id;	Navigates to a menu with ID "id". Example: Menu=50001;
	Title=id;	Navigates to a title with ID "id". Example: Title=1;
	Title=id; Chapter=num;	Navigates to Chapter with number "num" from Title with ID "id". This navigation command can be used only in titleSet (VTS) menus. Example: Title=1;Chapter=1;

### Child Elements

Element	Description
<command>	Specifies the VM command to be executed when the button is activated.

### Remarks

When a button (or highlight area) is selected, the portion of the mask enclosed in button's rectangular area is used to decide what colors to display. DVD specification allows up to 36 buttons per display menu.

If <command>child element is specified then the attribute "navigate" is not required. If both <command>child element and attribute "navigate" are specified then the program will use only the <command>child element.

### Example

The following example shows a DVD that has one VMG menu with one button. When the button is selected the DVD player will play the contents of chapter 1 from title 1.

```
<?xml version="1.0" encoding="utf-8"?>
<dvd version="1.1.0.1" xmlns="http://www.primosoftware.com/dvdbuilder">
  <videoManager firstPlayNavigate="Menu=50001;">
    <menus>
      <menu id="50001" entry="title">
        <button left="10" top="10" width="74" height="51" navigate="Title=1;Chapter=1;" />
        <background file="menu_50001.mpg" />
        <mask
          file="menu_50001_mask.bmp"
          patternColor="#FFFFFF"
          backgroundColor="#000000"
          emphasisColor1="#000001"
          emphasisColor2="#000001" />
        <display
          patternColor="#000000" patternContrast="#000000"
          backgroundColor="#000000" backgroundContrast="#000000"
        />
      </menu>
    </menus>
  </videoManager>
</dvd>
```

```

    emphasis1Color="#000000" emphasis1Contrast="#000000"
    emphasis2Color="#000000" emphasis2Contrast="#000000"/>
<action
  patternColor="#0000FF" patternContrast="7"
  backgroundColor="#000000" backgroundContrast="#000000"
  emphasis2Color="#000000" emphasis2Contrast="#000000"
  emphasis1Color="#000000" emphasis1Contrast="#000000"/>
<selection
  patternColor="#FF0000" patternContrast="7"
  backgroundColor="#000000" backgroundContrast="#000000"
  emphasis1Color="#000000" emphasis1Contrast="#000000"
  emphasis2Color="#000000" emphasis2Contrast="#000000"/>
</menu>
</menus>
</videoManager>
<titleSet>
  <titles>
    <title id="1" chapters="00:00:00;">
      <videoObject file="1.mpg"/>
    </title>
  </titles>
</titleSet>
</dvd>

```

### 3.1.1.1.6.1 <command> Element

#### Description

Defines a Virtual Machine (VM) command.

```

<command>
<opcode></opcode>
</command>

```

#### Example

The following example shows a DVD that has title menu, root menu and three titles. The title menu is a dummy one. It simply redirects to the root menu. The root menu supports playing each clip individually and returns back or playing all clips in sequence.

```

<?xml version="1.0" encoding="utf-8"?>
<dvd xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" version="2.1.2.1"
  xmlns="http://www.primosoftware.com/dvdbuilder">
  <videoManager>
    <firstPlayCommands>
      <command>
        <!-- JumpSS VTSM (vts 1, tt 1, menu ROOT), 30 06 00 01 01 83 00 00 -->
        <opcode>[0x30][0x06][0x00][0x01][0x01][0x83][0x00][0x00]</opcode>
      </command>
    </firstPlayCommands>

    <menus>
      <menu id="1" entry="title">
        <preCommands>
          <command>
            <!-- JumpSS VTSM (vts 1, tt 1, menu ROOT), 30 06 00 01 01 83 00 00 -->
            <opcode>[0x30][0x06][0x00][0x01][0x01][0x83][0x00][0x00]</opcode>
          </command>
        </preCommands>

        <background file="menu\title_menu.mpg" />
      </menu>
    </menus>
  </videoManager>

```

```

<!-- GPRM11 -> Play All Mode (True=1, False=0) -->

<titleSet>
  <titles>
    <title id="100" chapters="00:00:00;">
      <videoObject file="../movies/3/columbiatristardvd_pal.mpg" />

      <postCommands>
        <command>
          <!-- if (GPRM11 == 1) Goto 3, 00 a1 00 0b 00 01 00 03 -->
          <opcode>[0x00][0xa1][0x00][0x0b][0x00][0x01][0x00][0x03]</opcode>
        </command>
        <command>
          <!-- CallSS VTSM (menu ROOT, rsm_cell 1), 30 08 00 00 01 83 00 00 -->
          <opcode>[0x30][0x08][0x00][0x00][0x01][0x83][0x00][0x00]</opcode>
        </command>

        <command>
          <!-- JumpVTS_PTT (tt 2, ptt 1), 30 05 00 01 00 02 00 00 -->
          <opcode>[0x30][0x05][0x00][0x01][0x00][0x02][0x00][0x00]</opcode>
        </command>
      </postCommands>

    </title>

    <title id="101" chapters="00:00:00;">
      <videoObject file="../movies/4/dreamworksanim_pal.mpg" />

      <postCommands>
        <command>
          <!-- if (GPRM11 == 1) Goto 3, 00 a1 00 0b 00 01 00 03 -->
          <opcode>[0x00][0xa1][0x00][0x0b][0x00][0x01][0x00][0x03]</opcode>
        </command>
        <command>
          <!-- CallSS VTSM (menu ROOT, rsm_cell 1), 30 08 00 00 01 83 00 00 -->
          <opcode>[0x30][0x08][0x00][0x00][0x01][0x83][0x00][0x00]</opcode>
        </command>

        <command>
          <!-- JumpVTS_PTT (tt 3, ptt 1), 30 05 00 01 00 03 00 00 -->
          <opcode>[0x30][0x05][0x00][0x01][0x00][0x03][0x00][0x00]</opcode>
        </command>
      </postCommands>
    </title>

    <title id="102" chapters="00:00:00;">
      <videoObject file="../movies/3/columbiatristardvd_pal.mpg" />

      <postCommands>
        <command>
          <!-- CallSS VTSM (menu ROOT, rsm_cell 1), 30 08 00 00 01 83 00 00 -->
          <opcode>[0x30][0x08][0x00][0x00][0x01][0x83][0x00][0x00]</opcode>
        </command>
      </postCommands>
    </title>
  </titles>
  <menus>
    <menu id="2" entry="root">
      <preCommands>
        <command>
          <!-- GPRM11 = 0, 71 00 00 0b 00 00 00 00 -->
          <opcode>[0x71][0x00][0x00][0x0b][0x00][0x00][0x00][0x00]</opcode>
        </command>
      </preCommands>

      <postCommands>
        <command>
          <opcode>[0x20][0x04][0x0][0x0][0x0][0x0][0x0][0x01]</opcode>
          <!-- LinkPGCN 1, 20 04 00 00 00 00 01 -->
        </command>
      </postCommands>
    </menu>
  </menus>

```

```

<!-- Play Title 1 button -->
<button left="107" top="96" width="166" height="35">
  <command>
    <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 01 00 00 -->
    <opcode>[0x30][0x05][0x00][0x01][0x00][0x01][0x00][0x00]</opcode>
  </command>
</button>

<!-- Play Title 2 button -->
<button left="107" top="155" width="170" height="37">
  <command>
    <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 02 00 00 -->
    <opcode>[0x30][0x05][0x00][0x01][0x00][0x02][0x00][0x00]</opcode>
  </command>
</button>

<!-- Play Title 3 button -->
<button left="107" top="216" width="171" height="36">
  <command>
    <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 03 00 00 -->
    <opcode>[0x30][0x05][0x00][0x01][0x00][0x03][0x00][0x00]</opcode>
  </command>
</button>

<!-- Play All button -->
<button left="107" top="401" width="112" height="36">
  <command>
    <!-- LinkPGCN 2, 20 04 00 00 00 00 02 -->
    <opcode>[0x20][0x04][0x00][0x00][0x00][0x00][0x00][0x02]</opcode>
  </command>
</button>

  <mask file="menu\root_menu_mask.bmp" patternColor="#000000"
backgroundColor="#FFFFFF" emphasisColor1="#000001" emphasisColor2="#000002" />
  <display emphasis2Contrast="0" emphasis1Contrast="0" backgroundContrast="0"
patternContrast="0" emphasis2Color="#000000" emphasis1Color="#000000"
patternColor="#000000" />
  <action emphasis2Contrast="0" emphasis1Contrast="0" backgroundContrast="0"
patternContrast="7" emphasis2Color="#000000" emphasis1Color="#000000"
backgroundColor="#000000" patternColor="#0000FF" />
  <selection emphasis2Contrast="0" emphasis1Contrast="0" backgroundContrast="0"
patternContrast="7" emphasis2Color="#000000" emphasis1Color="#000000"
backgroundColor="#000000" patternColor="#FF0000" />
  <background file="menu\root_menu.mpg" />
</menu>

<!-- Play All Dummy menu -->
<menu id="101">
  <preCommands>
    <command>
      <!-- GPRM11 = 1, 71 00 00 0b 00 01 00 00 -->
      <opcode>[0x71][0x00][0x00][0x0b][0x00][0x01][0x00][0x00]</opcode>
    </command>

    <command>
      <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 01 00 00 -->
      <opcode>[0x30][0x05][0x00][0x01][0x00][0x01][0x00][0x00]</opcode>
    </command>
  </preCommands>
</menu>
</menus>
</titleSet>
</dvd>

```

### 3.1.1.1.6.1.1 <opcode> Element

Specifies the Virtual Machine (VM) command opcode. The opcode is 8 bytes long. Each byte is specified in as hexadecimal value [0xXX].

```
<opcode> [0xXX] [0xXX] [0xXX] [0xXX] [0xXX] [0xXX] [0xXX] [0xXX]</opcode>
```

**Remarks****Example**

The following example shows a DVD that has title menu, root menu and three titles. The title menu is a dummy one. It simply redirects to the root menu. The root menu supports playing each clip individually and returns back or playing all clips in sequence.

```
<?xml version="1.0" encoding="utf-8"?>
<dvd xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" version="2.1.2.1"
xmlns="http://www.primosoftware.com/dvdbuilder">
  <videoManager>
    <firstPlayCommands>
      <command>
        <!-- JumpSS VTSM (vts 1, tt 1, menu ROOT), 30 06 00 01 01 83 00 00 -->
        <opcode>[0x30][0x06][0x00][0x01][0x01][0x83][0x00][0x00]</opcode>
      </command>
    </firstPlayCommands>

    <menus>
      <menu id="1" entry="title">
        <preCommands>
          <command>
            <!-- JumpSS VTSM (vts 1, tt 1, menu ROOT), 30 06 00 01 01 83 00 00 -->
            <opcode>[0x30][0x06][0x00][0x01][0x01][0x83][0x00][0x00]</opcode>
          </command>
        </preCommands>

        <background file="menu\title_menu.mpg" />
      </menu>
    </menus>

  </videoManager>

  <!-- GPRM11 -> Play All Mode (True=1, False=0) -->

  <titleSet>
    <titles>
      <title id="100" chapters="00:00:00;">
        <videoObject file="../movies/3/columbiatristardvd_pal.mpg" />

        <postCommands>
          <command>
            <!-- if (GPRM11 == 1) Goto 3, 00 a1 00 0b 00 01 00 03 -->
            <opcode>[0x00][0xa1][0x00][0x0b][0x00][0x01][0x00][0x03]</opcode>
          </command>
          <command>
            <!-- CallSS VTSM (menu ROOT, rsm_cell 1), 30 08 00 00 01 83 00 00 -->
            <opcode>[0x30][0x08][0x00][0x00][0x01][0x83][0x00][0x00]</opcode>
          </command>

          <command>
            <!-- JumpVTS_PTT (tt 2, ptt 1), 30 05 00 01 00 02 00 00 -->
            <opcode>[0x30][0x05][0x00][0x01][0x00][0x02][0x00][0x00]</opcode>
          </command>
        </postCommands>
      </title>

      <title id="101" chapters="00:00:00;">
        <videoObject file="../movies/4/dreamworksanim_pal.mpg" />

        <postCommands>
          <command>
            <!-- if (GPRM11 == 1) Goto 3, 00 a1 00 0b 00 01 00 03 -->
```

```

    <opcode>[0x00][0xa1][0x00][0x0b][0x00][0x01][0x00][0x03]</opcode>
  </command>
</command>
  <!-- CallSS VTSM (menu ROOT, rsm_cell 1), 30 08 00 00 01 83 00 00 -->
  <opcode>[0x30][0x08][0x00][0x00][0x01][0x83][0x00][0x00]</opcode>
</command>

  <command>
    <!-- JumpVTS_PTT (tt 3, ptt 1), 30 05 00 01 00 03 00 00 -->
    <opcode>[0x30][0x05][0x00][0x01][0x00][0x03][0x00][0x00]</opcode>
  </command>
</postCommands>
</title>

<title id="102" chapters="00:00:00;">
  <videoObject file="../movies/3/columbiatristardvd_pal.mpg" />

  <postCommands>
    <command>
      <!-- CallSS VTSM (menu ROOT, rsm_cell 1), 30 08 00 00 01 83 00 00 -->
      <opcode>[0x30][0x08][0x00][0x00][0x01][0x83][0x00][0x00]</opcode>
    </command>
  </postCommands>
</title>
</titles>
<menus>
  <menu id="2" entry="root">
    <preCommands>
      <command>
        <!-- GPRM11 = 0, 71 00 00 0b 00 00 00 00 -->
        <opcode>[0x71][0x00][0x00][0x0b][0x00][0x00][0x00][0x00]</opcode>
      </command>
    </preCommands>

    <postCommands>
      <command>
        <opcode>[0x20][0x04][0x0][0x0][0x0][0x0][0x0][0x01]</opcode>
        <!-- LinkPGCN 1, 20 04 00 00 00 00 01 -->
      </command>
    </postCommands>

    <!-- Play Title 1 button -->
    <button left="107" top="96" width="166" height="35">
      <command>
        <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 01 00 00 -->
        <opcode>[0x30][0x05][0x00][0x01][0x00][0x01][0x00][0x00]</opcode>
      </command>
    </button>

    <!-- Play Title 2 button -->
    <button left="107" top="155" width="170" height="37">
      <command>
        <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 02 00 00 -->
        <opcode>[0x30][0x05][0x00][0x01][0x00][0x02][0x00][0x00]</opcode>
      </command>
    </button>

    <!-- Play Title 3 button -->
    <button left="107" top="216" width="171" height="36">
      <command>
        <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 03 00 00 -->
        <opcode>[0x30][0x05][0x00][0x01][0x00][0x03][0x00][0x00]</opcode>
      </command>
    </button>

    <!-- Play All button -->
    <button left="107" top="401" width="112" height="36">
      <command>
        <!-- LinkPGCN 2, 20 04 00 00 00 00 02 -->
        <opcode>[0x20][0x04][0x0][0x0][0x0][0x0][0x0][0x02]</opcode>
      </command>

```

```

</button>

    <mask file="menu\root_menu_mask.bmp" patternColor="#000000"
backgroundColor="#FFFFFF" emphasisColor1="#000001" emphasisColor2="#000002" />
    <display emphasis2Contrast="0" emphasis1Contrast="0" backgroundContrast="0"
patternContrast="0" emphasis2Color="#000000" emphasis1Color="#000000"
patternColor="#000000" />
    <action emphasis2Contrast="0" emphasis1Contrast="0" backgroundContrast="0"
patternContrast="7" emphasis2Color="#000000" emphasis1Color="#000000"
backgroundColor="#000000" patternColor="#0000FF" />
    <selection emphasis2Contrast="0" emphasis1Contrast="0" backgroundContrast="0"
patternContrast="7" emphasis2Color="#000000" emphasis1Color="#000000"
backgroundColor="#000000" patternColor="#FF0000" />
    <background file="menu\root_menu.mpg" />
</menu>

<!-- Play All Dummy menu -->
<menu id="101">
  <preCommands>
    <command>
      <!-- GPRM11 = 1, 71 00 00 0b 00 01 00 00 -->
      <opcode>[0x71][0x00][0x00][0x0b][0x00][0x01][0x00][0x00]</opcode>
    </command>

    <command>
      <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 01 00 00 -->
      <opcode>[0x30][0x05][0x00][0x01][0x00][0x01][0x00][0x00]</opcode>
    </command>
  </preCommands>
</menu>
</menus>
</titleSet>
</dvd>

```

### 3.1.1.1.1.7 <preCommands> Element

Specifies the VM commands to be executed before a menu or a title is presented.

```

<preCommands>
  <command/>
</preCommands>

```

#### Child Elements

Element	Description
<command> (Version 2.1.2)	Specifies a VM command.

#### Example

The following example shows a DVD that has title menu, root menu and three titles. The title menu is a dummy one. It simply redirects to the root menu. The root menu supports playing each clip individually and returns back or playing all clips in sequence.

```

<?xml version="1.0" encoding="utf-8"?>
<dvd xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" version="2.1.2.1"
xmlns="http://www.primosoftware.com/dvdbuilder">
  <videoManager>
    <firstPlayCommands>
      <command>
        <!-- JumpSS VTSM (vts 1, tt 1, menu ROOT), 30 06 00 01 01 83 00 00 -->
        <opcode>[0x30][0x06][0x00][0x01][0x01][0x83][0x00][0x00]</opcode>
      </command>
    </firstPlayCommands>

    <menus>
      <menu id="1" entry="title">

```

```

    <preCommands>
      <command>
        <!-- JumpSS VTSM (vts 1, tt 1, menu ROOT), 30 06 00 01 01 83 00 00 -->
        <opcode>[0x30][0x06][0x0][0x01][0x01][0x83][0x0][0x0]</opcode>
      </command>
    </preCommands>

    <background file="menu\title_menu.mpg" />
  </menu>
</menus>

</videoManager>

<!-- GPRM11 -> Play All Mode (True=1, False=0) -->

<titleSet>
  <titles>
    <title id="100" chapters="00:00:00;">
      <videoObject file="../movies/3/columbiatristardvd_pal.mpg" />

      <postCommands>
        <command>
          <!-- if (GPRM11 == 1) Goto 3, 00 a1 00 0b 00 01 00 03 -->
          <opcode>[0x00][0xa1][0x00][0x0b][0x00][0x01][0x00][0x03]</opcode>
        </command>
        <command>
          <!-- CallSS VTSM (menu ROOT, rsm_cell 1), 30 08 00 00 01 83 00 00 -->
          <opcode>[0x30][0x08][0x00][0x00][0x01][0x83][0x00][0x00]</opcode>
        </command>

        <command>
          <!-- JumpVTS_PTT (tt 2, ptt 1), 30 05 00 01 00 02 00 00 -->
          <opcode>[0x30][0x05][0x00][0x01][0x00][0x02][0x00][0x00]</opcode>
        </command>
      </postCommands>
    </title>

    <title id="101" chapters="00:00:00;">
      <videoObject file="../movies/4/dreamworksanim_pal.mpg" />

      <postCommands>
        <command>
          <!-- if (GPRM11 == 1) Goto 3, 00 a1 00 0b 00 01 00 03 -->
          <opcode>[0x00][0xa1][0x00][0x0b][0x00][0x01][0x00][0x03]</opcode>
        </command>
        <command>
          <!-- CallSS VTSM (menu ROOT, rsm_cell 1), 30 08 00 00 01 83 00 00 -->
          <opcode>[0x30][0x08][0x00][0x00][0x01][0x83][0x00][0x00]</opcode>
        </command>

        <command>
          <!-- JumpVTS_PTT (tt 3, ptt 1), 30 05 00 01 00 03 00 00 -->
          <opcode>[0x30][0x05][0x00][0x01][0x00][0x03][0x00][0x00]</opcode>
        </command>
      </postCommands>
    </title>

    <title id="102" chapters="00:00:00;">
      <videoObject file="../movies/3/columbiatristardvd_pal.mpg" />

      <postCommands>
        <command>
          <!-- CallSS VTSM (menu ROOT, rsm_cell 1), 30 08 00 00 01 83 00 00 -->
          <opcode>[0x30][0x08][0x00][0x00][0x01][0x83][0x00][0x00]</opcode>
        </command>
      </postCommands>
    </title>
  </titles>
</menuSet>
  <menu id="2" entry="root">

```

```

<preCommands>
  <command>
    <!-- GPRM11 = 0, 71 00 00 0b 00 00 00 00 -->
    <opcode>[0x71][0x00][0x00][0x0b][0x00][0x00][0x00][0x00]</opcode>
  </command>
</preCommands>

<postCommands>
  <command>
    <opcode>[0x20][0x04][0x0][0x0][0x0][0x0][0x0][0x01]</opcode>
    <!-- LinkPGCN 1, 20 04 00 00 00 00 01 -->
  </command>
</postCommands>

<!-- Play Title 1 button -->
<button left="107" top="96" width="166" height="35">
  <command>
    <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 01 00 00 -->
    <opcode>[0x30][0x05][0x00][0x01][0x00][0x01][0x00][0x00]</opcode>
  </command>
</button>

<!-- Play Title 2 button -->
<button left="107" top="155" width="170" height="37">
  <command>
    <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 02 00 00 -->
    <opcode>[0x30][0x05][0x00][0x01][0x00][0x02][0x00][0x00]</opcode>
  </command>
</button>

<!-- Play Title 3 button -->
<button left="107" top="216" width="171" height="36">
  <command>
    <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 03 00 00 -->
    <opcode>[0x30][0x05][0x00][0x01][0x00][0x03][0x00][0x00]</opcode>
  </command>
</button>

<!-- Play All button -->
<button left="107" top="401" width="112" height="36">
  <command>
    <!-- LinkPGCN 2, 20 04 00 00 00 00 02 -->
    <opcode>[0x20][0x04][0x0][0x0][0x0][0x0][0x0][0x02]</opcode>
  </command>
</button>

  <mask file="menu\root_menu_mask.bmp" patternColor="#000000"
backgroundcolor="#FFFFFF" emphasisColor1="#000001" emphasisColor2="#000002" />
  <display emphasis2Contrast="0" emphasis1Contrast="0" backgroundContrast="0"
patternContrast="0" emphasis2Color="#000000" emphasis1Color="#000000"
patternColor="#000000" />
  <action emphasis2Contrast="0" emphasis1Contrast="0" backgroundContrast="0"
patternContrast="7" emphasis2Color="#000000" emphasis1Color="#000000"
backgroundcolor="#000000" patternColor="#0000FF" />
  <selection emphasis2Contrast="0" emphasis1Contrast="0" backgroundContrast="0"
patternContrast="7" emphasis2Color="#000000" emphasis1Color="#000000"
backgroundcolor="#000000" patternColor="#FF0000" />
  <background file="menu\root_menu.mpg" />
</menu>

<!-- Play All Dummy menu -->
<menu id="101">
  <preCommands>
    <command>
      <!-- GPRM11 = 1, 71 00 00 0b 00 01 00 00 -->
      <opcode>[0x71][0x00][0x00][0x0b][0x00][0x01][0x00][0x00]</opcode>
    </command>

    <command>
      <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 01 00 00 -->
      <opcode>[0x30][0x05][0x00][0x01][0x00][0x01][0x00][0x00]</opcode>
    </command>

```

```

    </preCommands>
  </menu>
</menus>
</titleSet>
</dvd>

```

### 3.1.1.1.1.7.1 <command> Element

#### Description

Defines a Virtual Machine (VM) command.

```

<command>
<opcode></opcode>
</command>

```

#### Example

The following example shows a DVD that has title menu, root menu and three titles. The title menu is a dummy one. It simply redirects to the root menu. The root menu supports playing each clip individually and returns back or playing all clips in sequence.

```

<?xml version="1.0" encoding="utf-8"?>
<dvd xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" version="2.1.2.1"
xmlns="http://www.primosoftware.com/dvdbuilder">
  <videoManager>
    <firstPlayCommands>
      <command>
        <!-- JumpSS VTSM (vts 1, tt 1, menu ROOT), 30 06 00 01 01 83 00 00 -->
        <opcode>[0x30][0x06][0x00][0x01][0x01][0x83][0x00][0x00]</opcode>
      </command>
    </firstPlayCommands>

    <menus>
      <menu id="1" entry="title">
        <preCommands>
          <command>
            <!-- JumpSS VTSM (vts 1, tt 1, menu ROOT), 30 06 00 01 01 83 00 00 -->
            <opcode>[0x30][0x06][0x00][0x01][0x01][0x83][0x00][0x00]</opcode>
          </command>
        </preCommands>

        <background file="menu\title_menu.mpg" />
      </menu>
    </menus>

  </videoManager>

  <!-- GPRM11 -> Play All Mode (True=1, False=0) -->

  <titleSet>
    <titles>
      <title id="100" chapters="00:00:00;">
        <videoObject file="../movies/3/columbiatristardvd_pal.mpg" />

        <postCommands>
          <command>
            <!-- if (GPRM11 == 1) Goto 3, 00 a1 00 0b 00 01 00 03 -->
            <opcode>[0x00][0xa1][0x00][0x0b][0x00][0x01][0x00][0x03]</opcode>
          </command>
          <command>
            <!-- CallSS VTSM (menu ROOT, rsm_cell 1), 30 08 00 00 01 83 00 00 -->
            <opcode>[0x30][0x08][0x00][0x00][0x01][0x83][0x00][0x00]</opcode>
          </command>

          <command>
            <!-- JumpVTS_PTT (tt 2, ptt 1), 30 05 00 01 00 02 00 00 -->
            <opcode>[0x30][0x05][0x00][0x01][0x00][0x02][0x00][0x00]</opcode>

```

```

    </command>
  </postCommands>

</title>

<title id="101" chapters="00:00:00;">
  <videoObject file="../movies/4/dreamworksanim_pal.mpg" />

  <postCommands>
    <command>
      <!-- if (GPRM11 == 1) Goto 3, 00 a1 00 0b 00 01 00 03 -->
      <opcode>[0x00][0xa1][0x00][0x0b][0x00][0x01][0x00][0x03]</opcode>
    </command>
    <command>
      <!-- CallSS VTSM (menu ROOT, rsm_cell 1), 30 08 00 00 01 83 00 00 -->
      <opcode>[0x30][0x08][0x00][0x00][0x01][0x83][0x00][0x00]</opcode>
    </command>

    <command>
      <!-- JumpVTS_PTT (tt 3, ptt 1), 30 05 00 01 00 03 00 00 -->
      <opcode>[0x30][0x05][0x00][0x01][0x00][0x03][0x00][0x00]</opcode>
    </command>
  </postCommands>
</title>

<title id="102" chapters="00:00:00;">
  <videoObject file="../movies/3/columbiatristardvd_pal.mpg" />

  <postCommands>
    <command>
      <!-- CallSS VTSM (menu ROOT, rsm_cell 1), 30 08 00 00 01 83 00 00 -->
      <opcode>[0x30][0x08][0x00][0x00][0x01][0x83][0x00][0x00]</opcode>
    </command>
  </postCommands>
</title>

</titles>
<menus>
  <menu id="2" entry="root">
    <preCommands>
      <command>
        <!-- GPRM11 = 0, 71 00 00 0b 00 00 00 00 -->
        <opcode>[0x71][0x00][0x00][0x0b][0x00][0x00][0x00][0x00]</opcode>
      </command>
    </preCommands>

    <postCommands>
      <command>
        <opcode>[0x20][0x04][0x0][0x0][0x0][0x0][0x0][0x01]</opcode>
        <!-- LinkPGCN 1, 20 04 00 00 00 00 01 -->
      </command>
    </postCommands>

    <!-- Play Title 1 button -->
    <button left="107" top="96" width="166" height="35">
      <command>
        <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 01 00 00 -->
        <opcode>[0x30][0x05][0x00][0x01][0x00][0x01][0x00][0x00]</opcode>
      </command>
    </button>

    <!-- Play Title 2 button -->
    <button left="107" top="155" width="170" height="37">
      <command>
        <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 02 00 00 -->
        <opcode>[0x30][0x05][0x00][0x01][0x00][0x02][0x00][0x00]</opcode>
      </command>
    </button>

    <!-- Play Title 3 button -->
    <button left="107" top="216" width="171" height="36">
      <command>

```

```

        <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 03 00 00 -->
        <opcode>[0x30][0x05][0x00][0x01][0x00][0x03][0x00][0x00]</opcode>
    </command>
</button>

<!-- Play All button -->
<button left="107" top="401" width="112" height="36">
    <command>
        <!-- LinkPGCN 2, 20 04 00 00 00 00 02 -->
        <opcode>[0x20][0x04][0x00][0x00][0x00][0x00][0x00][0x02]</opcode>
    </command>
</button>

    <mask file="menu\root_menu_mask.bmp" patternColor="#000000"
backgroundcolor="#FFFFFF" emphasisColor1="#000001" emphasisColor2="#000002" />
    <display emphasis2Contrast="0" emphasis1Contrast="0" backgroundContrast="0"
patternContrast="0" emphasis2Color="#000000" emphasis1Color="#000000"
patternColor="#000000" />
    <action emphasis2Contrast="0" emphasis1Contrast="0" backgroundContrast="0"
patternContrast="7" emphasis2Color="#000000" emphasis1Color="#000000"
backgroundcolor="#000000" patternColor="#0000FF" />
    <selection emphasis2Contrast="0" emphasis1Contrast="0" backgroundContrast="0"
patternContrast="7" emphasis2Color="#000000" emphasis1Color="#000000"
backgroundcolor="#000000" patternColor="#FF0000" />
    <background file="menu\root_menu.mpg" />
</menu>

<!-- Play All Dummy menu -->
<menu id="101">
    <preCommands>
        <command>
            <!-- GPRM11 = 1, 71 00 00 0b 00 01 00 00 -->
            <opcode>[0x71][0x00][0x00][0x0b][0x00][0x01][0x00][0x00]</opcode>
        </command>

        <command>
            <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 01 00 00 -->
            <opcode>[0x30][0x05][0x00][0x01][0x00][0x01][0x00][0x00]</opcode>
        </command>
    </preCommands>
</menu>
</menus>
</titleSet>
</dvd>

```

### 3.1.1.1.7.1.1 <opcode> Element

Specifies the Virtual Machine (VM) command opcode. The opcode is 8 bytes long. Each byte is specified in as hexadecimal value [0xXX].

```
<opcode> [0xXX][0xXX][0xXX][0xXX][0xXX][0xXX][0xXX][0xXX]</opcode>
```

#### Remarks

#### Example

The following example shows a DVD that has title menu, root menu and three titles. The title menu is a dummy one. It simply redirects to the root menu. The root menu supports playing each clip individually and returns back or playing all clips in sequence.

```

<?xml version="1.0" encoding="utf-8"?>
<dvd xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" version="2.1.2.1"
xmlns="http://www.primosoftware.com/dvdbuilder">
    <videoManager>
        <firstPlayCommands>
            <command>

```

```

    <!-- JumpSS VTSM (vts 1, tt 1, menu ROOT), 30 06 00 01 01 83 00 00 -->
    <opcode>[0x30][0x06][0x00][0x01][0x01][0x83][0x00][0x00]</opcode>
  </command>
</firstPlayCommands>

<menus>
  <menu id="1" entry="title">
    <preCommands>
      <command>
        <!-- JumpSS VTSM (vts 1, tt 1, menu ROOT), 30 06 00 01 01 83 00 00 -->
        <opcode>[0x30][0x06][0x00][0x01][0x01][0x83][0x00][0x00]</opcode>
      </command>
    </preCommands>

    <background file="menu\title_menu.mpg" />
  </menu>
</menus>

</videoManager>

<!-- GPRM11 -> Play All Mode (True=1, False=0) -->

<titleSet>
  <titles>
    <title id="100" chapters="00:00:00;">
      <videoObject file="../movies/3/columbiatristardvd_pal.mpg" />

      <postCommands>
        <command>
          <!-- if (GPRM11 == 1) Goto 3, 00 a1 00 0b 00 01 00 03 -->
          <opcode>[0x00][0xa1][0x00][0x0b][0x00][0x01][0x00][0x03]</opcode>
        </command>
        <command>
          <!-- CallSS VTSM (menu ROOT, rsm_cell 1), 30 08 00 00 01 83 00 00 -->
          <opcode>[0x30][0x08][0x00][0x00][0x01][0x83][0x00][0x00]</opcode>
        </command>

        <command>
          <!-- JumpVTS_PTT (tt 2, ptt 1), 30 05 00 01 00 02 00 00 -->
          <opcode>[0x30][0x05][0x00][0x01][0x00][0x02][0x00][0x00]</opcode>
        </command>
      </postCommands>
    </title>

    <title id="101" chapters="00:00:00;">
      <videoObject file="../movies/4/dreamworksanim_pal.mpg" />

      <postCommands>
        <command>
          <!-- if (GPRM11 == 1) Goto 3, 00 a1 00 0b 00 01 00 03 -->
          <opcode>[0x00][0xa1][0x00][0x0b][0x00][0x01][0x00][0x03]</opcode>
        </command>
        <command>
          <!-- CallSS VTSM (menu ROOT, rsm_cell 1), 30 08 00 00 01 83 00 00 -->
          <opcode>[0x30][0x08][0x00][0x00][0x01][0x83][0x00][0x00]</opcode>
        </command>

        <command>
          <!-- JumpVTS_PTT (tt 3, ptt 1), 30 05 00 01 00 03 00 00 -->
          <opcode>[0x30][0x05][0x00][0x01][0x00][0x03][0x00][0x00]</opcode>
        </command>
      </postCommands>
    </title>

    <title id="102" chapters="00:00:00;">
      <videoObject file="../movies/3/columbiatristardvd_pal.mpg" />

      <postCommands>
        <command>
          <!-- CallSS VTSM (menu ROOT, rsm_cell 1), 30 08 00 00 01 83 00 00 -->
          <opcode>[0x30][0x08][0x00][0x00][0x01][0x83][0x00][0x00]</opcode>

```

```

    </command>
  </postCommands>
</titles>
</titles>
<menu>
  <menu id="2" entry="root">
    <preCommands>
      <command>
        <!-- GPRM11 = 0, 71 00 00 0b 00 00 00 00 -->
        <opcode>[0x71][0x00][0x00][0x0b][0x00][0x00][0x00][0x00]</opcode>
      </command>
    </preCommands>

    <postCommands>
      <command>
        <opcode>[0x20][0x04][0x0][0x0][0x0][0x0][0x0][0x01]</opcode>
        <!-- LinkPGCN 1, 20 04 00 00 00 00 01 -->
      </command>
    </postCommands>

    <!-- Play Title 1 button -->
    <button left="107" top="96" width="166" height="35">
      <command>
        <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 01 00 00 -->
        <opcode>[0x30][0x05][0x00][0x01][0x00][0x01][0x00][0x00]</opcode>
      </command>
    </button>

    <!-- Play Title 2 button -->
    <button left="107" top="155" width="170" height="37">
      <command>
        <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 02 00 00 -->
        <opcode>[0x30][0x05][0x00][0x01][0x00][0x02][0x00][0x00]</opcode>
      </command>
    </button>

    <!-- Play Title 3 button -->
    <button left="107" top="216" width="171" height="36">
      <command>
        <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 03 00 00 -->
        <opcode>[0x30][0x05][0x00][0x01][0x00][0x03][0x00][0x00]</opcode>
      </command>
    </button>

    <!-- Play All button -->
    <button left="107" top="401" width="112" height="36">
      <command>
        <!-- LinkPGCN 2, 20 04 00 00 00 00 02 -->
        <opcode>[0x20][0x04][0x0][0x0][0x0][0x0][0x0][0x02]</opcode>
      </command>
    </button>

    <mask file="menu\root_menu_mask.bmp" patternColor="#000000"
backgroundColor="#FFFFFF" emphasisColor1="#000001" emphasisColor2="#000002" />
    <display emphasis2Contrast="0" emphasis1Contrast="0" backgroundContrast="0"
patternContrast="0" emphasis2Color="#000000" emphasis1Color="#000000"
patternColor="#000000" />
    <action emphasis2Contrast="0" emphasis1Contrast="0" backgroundContrast="0"
patternContrast="7" emphasis2Color="#000000" emphasis1Color="#000000"
backgroundColor="#000000" patternColor="#0000FF" />
    <selection emphasis2Contrast="0" emphasis1Contrast="0" backgroundContrast="0"
patternContrast="7" emphasis2Color="#000000" emphasis1Color="#000000"
backgroundColor="#000000" patternColor="#FF0000" />
    <background file="menu\root_menu.mpg" />
  </menu>

  <!-- Play All Dummy menu -->
  <menu id="101">
    <preCommands>
      <command>
        <!-- GPRM11 = 1, 71 00 00 0b 00 01 00 00 -->

```

```

    <opcode>[0x71][0x00][0x00][0x0b][0x00][0x01][0x00][0x00]</opcode>
  </command>

  <command>
    <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 01 00 00 -->
    <opcode>[0x30][0x05][0x00][0x01][0x00][0x01][0x00][0x00]</opcode>
  </command>
</preCommands>
</menu>
</menus>
</titleSet>
</dvd>

```

### 3.1.1.1.8 <postCommands> Element

Specifies the VM commands to be executed after a menu or a title is presented.

```

<preCommands>
  <command/>
</preCommands>

```

#### Child Elements

Element	Description
<command> (Version 2.1.2)	Specifies a VM command.

#### Example

The following example shows a DVD that has title menu, root menu and three titles. The title menu is a dummy one. It simply redirects to the root menu. The root menu supports playing each clip individually and returns back or playing all clips in sequence.

```

<?xml version="1.0" encoding="utf-8"?>
<dvd xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" version="2.1.2.1"
xmlns="http://www.primosoftware.com/dvdbuilder">
  <videoManager>
    <firstPlayCommands>
      <command>
        <!-- JumpSS VTSM (vts 1, tt 1, menu ROOT), 30 06 00 01 01 83 00 00 -->
        <opcode>[0x30][0x06][0x0][0x01][0x01][0x83][0x0][0x0]</opcode>
      </command>
    </firstPlayCommands>

    <menus>
      <menu id="1" entry="title">
        <preCommands>
          <command>
            <!-- JumpSS VTSM (vts 1, tt 1, menu ROOT), 30 06 00 01 01 83 00 00 -->
            <opcode>[0x30][0x06][0x0][0x01][0x01][0x83][0x0][0x0]</opcode>
          </command>
        </preCommands>

        <background file="menu\title_menu.mpg" />
      </menu>
    </menus>

  </videoManager>

  <!-- GPRM11 -> Play All Mode (True=1, False=0) -->

  <titleSet>
    <titles>
      <title id="100" chapters="00:00:00;">
        <videoObject file="../movies/3/columbiatristardvd_pal.mpg" />

        <postCommands>

```

```

    <command>
      <!-- if (GPRM11 == 1) Goto 3, 00 a1 00 0b 00 01 00 03 -->
      <opcode>[0x00][0xa1][0x00][0x0b][0x00][0x01][0x00][0x03]</opcode>
    </command>
    <command>
      <!-- CallSS VTSM (menu ROOT, rsm_cell 1), 30 08 00 00 01 83 00 00 -->
      <opcode>[0x30][0x08][0x00][0x00][0x01][0x83][0x00][0x00]</opcode>
    </command>

    <command>
      <!-- JumpVTS_PTT (tt 2, ptt 1), 30 05 00 01 00 02 00 00 -->
      <opcode>[0x30][0x05][0x00][0x01][0x00][0x02][0x00][0x00]</opcode>
    </command>
  </postCommands>

</title>

<title id="101" chapters="00:00:00;">
  <videoObject file="../movies/4/dreamworksanim_pal.mpg" />

  <postCommands>
    <command>
      <!-- if (GPRM11 == 1) Goto 3, 00 a1 00 0b 00 01 00 03 -->
      <opcode>[0x00][0xa1][0x00][0x0b][0x00][0x01][0x00][0x03]</opcode>
    </command>
    <command>
      <!-- CallSS VTSM (menu ROOT, rsm_cell 1), 30 08 00 00 01 83 00 00 -->
      <opcode>[0x30][0x08][0x00][0x00][0x01][0x83][0x00][0x00]</opcode>
    </command>

    <command>
      <!-- JumpVTS_PTT (tt 3, ptt 1), 30 05 00 01 00 03 00 00 -->
      <opcode>[0x30][0x05][0x00][0x01][0x00][0x03][0x00][0x00]</opcode>
    </command>
  </postCommands>
</title>

<title id="102" chapters="00:00:00;">
  <videoObject file="../movies/3/columbiatristardvd_pal.mpg" />

  <postCommands>
    <command>
      <!-- CallSS VTSM (menu ROOT, rsm_cell 1), 30 08 00 00 01 83 00 00 -->
      <opcode>[0x30][0x08][0x00][0x00][0x01][0x83][0x00][0x00]</opcode>
    </command>
  </postCommands>
</title>

</titles>
<menus>
  <menu id="2" entry="root">
    <preCommands>
      <command>
        <!-- GPRM11 = 0, 71 00 00 0b 00 00 00 00 -->
        <opcode>[0x71][0x00][0x00][0x0b][0x00][0x00][0x00][0x00]</opcode>
      </command>
    </preCommands>

    <postCommands>
      <command>
        <opcode>[0x20][0x04][0x0][0x0][0x0][0x0][0x0][0x01]</opcode>
        <!-- LinkPGCN 1, 20 04 00 00 00 00 01 -->
      </command>
    </postCommands>

    <!-- Play Title 1 button -->
    <button left="107" top="96" width="166" height="35">
      <command>
        <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 01 00 00 -->
        <opcode>[0x30][0x05][0x00][0x01][0x00][0x01][0x00][0x00]</opcode>
      </command>
    </button>

```

```

<!-- Play Title 2 button -->
<button left="107" top="155" width="170" height="37">
  <command>
    <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 02 00 00 -->
    <opcode>[0x30][0x05][0x00][0x01][0x00][0x02][0x00][0x00]</opcode>
  </command>
</button>

<!-- Play Title 3 button -->
<button left="107" top="216" width="171" height="36">
  <command>
    <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 03 00 00 -->
    <opcode>[0x30][0x05][0x00][0x01][0x00][0x03][0x00][0x00]</opcode>
  </command>
</button>

<!-- Play All button -->
<button left="107" top="401" width="112" height="36">
  <command>
    <!-- LinkPGCN 2, 20 04 00 00 00 00 02 -->
    <opcode>[0x20][0x04][0x0][0x0][0x0][0x0][0x0][0x02]</opcode>
  </command>
</button>

  <mask file="menu\root_menu_mask.bmp" patternColor="#000000"
backgroundColor="#FFFFFF" emphasisColor1="#000001" emphasisColor2="#000002" />
  <display emphasis2Contrast="0" emphasis1Contrast="0" backgroundContrast="0"
patternContrast="0" emphasis2Color="#000000" emphasis1Color="#000000"
patternColor="#000000" />
  <action emphasis2Contrast="0" emphasis1Contrast="0" backgroundContrast="0"
patternContrast="7" emphasis2Color="#000000" emphasis1Color="#000000"
backgroundColor="#000000" patternColor="#0000FF" />
  <selection emphasis2Contrast="0" emphasis1Contrast="0" backgroundContrast="0"
patternContrast="7" emphasis2Color="#000000" emphasis1Color="#000000"
backgroundColor="#000000" patternColor="#FF0000" />
  <background file="menu\root_menu.mpg" />
</menu>

<!-- Play All Dummy menu -->
<menu id="101">
  <preCommands>
    <command>
      <!-- GPRM11 = 1, 71 00 00 0b 00 01 00 00 -->
      <opcode>[0x71][0x00][0x00][0x0b][0x00][0x01][0x00][0x00]</opcode>
    </command>

    <command>
      <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 01 00 00 -->
      <opcode>[0x30][0x05][0x00][0x01][0x00][0x01][0x00][0x00]</opcode>
    </command>
  </preCommands>
</menu>
</menus>
</titleSet>
</dvd>

```

### 3.1.1.1.8.1 <command> Element

#### Description

Defines a Virtual Machine (VM) command.

```

<command>
<opcode></opcode>
</command>

```

#### Example

The following example shows a DVD that has title menu, root menu and three titles. The title menu is a dummy one. It

simply redirects to the root menu. The root menu supports playing each clip individually and returns back or playing all clips in sequence.

```
<?xml version="1.0" encoding="utf-8"?>
<dvd xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" version="2.1.2.1"
xmlns="http://www.primosoftware.com/dvdbuilder">
  <videoManager>
    <firstPlayCommands>
      <command>
        <!-- JumpSS VTSM (vts 1, tt 1, menu ROOT), 30 06 00 01 01 83 00 00 -->
        <opcode>[0x30][0x06][0x0][0x01][0x01][0x83][0x0][0x0]</opcode>
      </command>
    </firstPlayCommands>

    <menus>
      <menu id="1" entry="title">
        <preCommands>
          <command>
            <!-- JumpSS VTSM (vts 1, tt 1, menu ROOT), 30 06 00 01 01 83 00 00 -->
            <opcode>[0x30][0x06][0x0][0x01][0x01][0x83][0x0][0x0]</opcode>
          </command>
        </preCommands>

        <background file="menu\title_menu.mpg" />
      </menu>
    </menus>

  </videoManager>

  <!-- GPRM11 -> Play All Mode (True=1, False=0) -->

  <titleSet>
    <titles>
      <title id="100" chapters="00:00:00;">
        <videoObject file="../movies/3/columbiatristardvd_pal.mpg" />

        <postCommands>
          <command>
            <!-- if (GPRM11 == 1) Goto 3, 00 a1 00 0b 00 01 00 03 -->
            <opcode>[0x00][0xa1][0x00][0x0b][0x00][0x01][0x00][0x03]</opcode>
          </command>
          <command>
            <!-- CallSS VTSM (menu ROOT, rsm_cell 1), 30 08 00 00 01 83 00 00 -->
            <opcode>[0x30][0x08][0x00][0x00][0x01][0x83][0x00][0x00]</opcode>
          </command>

          <command>
            <!-- JumpVTS_PTT (tt 2, ptt 1), 30 05 00 01 00 02 00 00 -->
            <opcode>[0x30][0x05][0x00][0x01][0x00][0x02][0x00][0x00]</opcode>
          </command>
        </postCommands>

      </title>

      <title id="101" chapters="00:00:00;">
        <videoObject file="../movies/4/dreamworksanim_pal.mpg" />

        <postCommands>
          <command>
            <!-- if (GPRM11 == 1) Goto 3, 00 a1 00 0b 00 01 00 03 -->
            <opcode>[0x00][0xa1][0x00][0x0b][0x00][0x01][0x00][0x03]</opcode>
          </command>
          <command>
            <!-- CallSS VTSM (menu ROOT, rsm_cell 1), 30 08 00 00 01 83 00 00 -->
            <opcode>[0x30][0x08][0x00][0x00][0x01][0x83][0x00][0x00]</opcode>
          </command>

          <command>
            <!-- JumpVTS_PTT (tt 3, ptt 1), 30 05 00 01 00 03 00 00 -->
            <opcode>[0x30][0x05][0x00][0x01][0x00][0x03][0x00][0x00]</opcode>
          </command>
        </postCommands>
      </title>
    </titles>
  </titleSet>
</dvd>
```

```

        </command>
    </postCommands>
</title>

<title id="102" chapters="00:00:00;">
    <videoObject file="../movies/3/columbiatristardvd_pal.mpg" />

    <postCommands>
        <command>
            <!-- CallSS VTSM (menu ROOT, rsm_cell 1), 30 08 00 00 01 83 00 00 -->
            <opcode>[0x30][0x08][0x00][0x00][0x01][0x83][0x00][0x00]</opcode>
        </command>
    </postCommands>
</title>

</titles>
<menus>
    <menu id="2" entry="root">
        <preCommands>
            <command>
                <!-- GPRM11 = 0, 71 00 00 0b 00 00 00 00 -->
                <opcode>[0x71][0x00][0x00][0x0b][0x00][0x00][0x00][0x00]</opcode>
            </command>
        </preCommands>

        <postCommands>
            <command>
                <opcode>[0x20][0x04][0x0][0x0][0x0][0x0][0x0][0x01]</opcode>
                <!-- LinkPGCN 1, 20 04 00 00 00 00 01 -->
            </command>
        </postCommands>

        <!-- Play Title 1 button -->
        <button left="107" top="96" width="166" height="35">
            <command>
                <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 01 00 00 -->
                <opcode>[0x30][0x05][0x00][0x01][0x00][0x01][0x00][0x00]</opcode>
            </command>
        </button>

        <!-- Play Title 2 button -->
        <button left="107" top="155" width="170" height="37">
            <command>
                <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 02 00 00 -->
                <opcode>[0x30][0x05][0x00][0x01][0x00][0x02][0x00][0x00]</opcode>
            </command>
        </button>

        <!-- Play Title 3 button -->
        <button left="107" top="216" width="171" height="36">
            <command>
                <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 03 00 00 -->
                <opcode>[0x30][0x05][0x00][0x01][0x00][0x03][0x00][0x00]</opcode>
            </command>
        </button>

        <!-- Play All button -->
        <button left="107" top="401" width="112" height="36">
            <command>
                <!-- LinkPGCN 2, 20 04 00 00 00 00 02 -->
                <opcode>[0x20][0x04][0x0][0x0][0x0][0x0][0x0][0x02]</opcode>
            </command>
        </button>

        <mask file="menu\root_menu_mask.bmp" patternColor="#000000"
backgroundColor="#FFFFFF" emphasisColor1="#000001" emphasisColor2="#000002" />
        <display emphasis2Contrast="0" emphasis1Contrast="0" backgroundContrast="0"
patternContrast="0" emphasis2Color="#000000" emphasis1Color="#000000"
patternColor="#000000" />
        <action emphasis2Contrast="0" emphasis1Contrast="0" backgroundContrast="0"
patternContrast="7" emphasis2Color="#000000" emphasis1Color="#000000"
backgroundColor="#000000" patternColor="#0000FF" />

```

```

    <selection emphasis2Contrast="0" emphasis1Contrast="0" backgroundContrast="0"
patternContrast="7" emphasis2Color="#000000" emphasis1Color="#000000"
backgroundColor="#000000" patternColor="#FF0000" />
    <background file="menu\root_menu.mpg" />
  </menu>

  <!-- Play All Dummy menu -->
  <menu id="101">
    <preCommands>
      <command>
        <!-- GPRM11 = 1, 71 00 00 0b 00 01 00 00 -->
        <opcode>[0x71][0x00][0x00][0x0b][0x00][0x01][0x00][0x00]</opcode>
      </command>

      <command>
        <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 01 00 00 -->
        <opcode>[0x30][0x05][0x00][0x01][0x00][0x01][0x00][0x00]</opcode>
      </command>
    </preCommands>
  </menu>
</menus>
</titleSet>
</dvd>

```

### 3.1.1.1.8.1.1 <opcode> Element

Specifies the Virtual Machine (VM) command opcode. The opcode is 8 bytes long. Each byte is specified in as hexadecimal value [0xXX].

```
<opcode> [0xXX] [0xXX] [0xXX] [0xXX] [0xXX] [0xXX] [0xXX] [0xXX] </opcode>
```

#### Remarks

#### Example

The following example shows a DVD that has title menu, root menu and three titles. The title menu is a dummy one. It simply redirects to the root menu. The root menu supports playing each clip individually and returns back or playing all clips in sequence.

```

<?xml version="1.0" encoding="utf-8"?>
<dvd xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" version="2.1.2.1"
xmlns="http://www.primosoftware.com/dvdbuilder">
  <videoManager>
    <firstPlayCommands>
      <command>
        <!-- JumpSS VTSM (vts 1, tt 1, menu ROOT), 30 06 00 01 01 83 00 00 -->
        <opcode>[0x30][0x06][0x00][0x01][0x01][0x83][0x00][0x00]</opcode>
      </command>
    </firstPlayCommands>

    <menus>
      <menu id="1" entry="title">
        <preCommands>
          <command>
            <!-- JumpSS VTSM (vts 1, tt 1, menu ROOT), 30 06 00 01 01 83 00 00 -->
            <opcode>[0x30][0x06][0x00][0x01][0x01][0x83][0x00][0x00]</opcode>
          </command>
        </preCommands>

        <background file="menu\title_menu.mpg" />
      </menu>
    </menus>
  </videoManager>

  <!-- GPRM11 -> Play All Mode (True=1, False=0) -->

```

```

<titleSet>
  <titles>
    <title id="100" chapters="00:00:00;">
      <videoObject file="../movies/3/columbiatristardvd_pal.mpg" />

      <postCommands>
        <command>
          <!-- if (GPRM11 == 1) Goto 3, 00 a1 00 0b 00 01 00 03 -->
          <opcode>[0x00][0xa1][0x00][0x0b][0x00][0x01][0x00][0x03]</opcode>
        </command>
        <command>
          <!-- CallSS VTSM (menu ROOT, rsm_cell 1), 30 08 00 00 01 83 00 00 -->
          <opcode>[0x30][0x08][0x00][0x00][0x01][0x83][0x00][0x00]</opcode>
        </command>

        <command>
          <!-- JumpVTS_PTT (tt 2, ptt 1), 30 05 00 01 00 02 00 00 -->
          <opcode>[0x30][0x05][0x00][0x01][0x00][0x02][0x00][0x00]</opcode>
        </command>
      </postCommands>
    </title>

    <title id="101" chapters="00:00:00;">
      <videoObject file="../movies/4/dreamworksanim_pal.mpg" />

      <postCommands>
        <command>
          <!-- if (GPRM11 == 1) Goto 3, 00 a1 00 0b 00 01 00 03 -->
          <opcode>[0x00][0xa1][0x00][0x0b][0x00][0x01][0x00][0x03]</opcode>
        </command>
        <command>
          <!-- CallSS VTSM (menu ROOT, rsm_cell 1), 30 08 00 00 01 83 00 00 -->
          <opcode>[0x30][0x08][0x00][0x00][0x01][0x83][0x00][0x00]</opcode>
        </command>

        <command>
          <!-- JumpVTS_PTT (tt 3, ptt 1), 30 05 00 01 00 03 00 00 -->
          <opcode>[0x30][0x05][0x00][0x01][0x00][0x03][0x00][0x00]</opcode>
        </command>
      </postCommands>
    </title>

    <title id="102" chapters="00:00:00;">
      <videoObject file="../movies/3/columbiatristardvd_pal.mpg" />

      <postCommands>
        <command>
          <!-- CallSS VTSM (menu ROOT, rsm_cell 1), 30 08 00 00 01 83 00 00 -->
          <opcode>[0x30][0x08][0x00][0x00][0x01][0x83][0x00][0x00]</opcode>
        </command>
      </postCommands>
    </title>
  </titles>
  <menus>
    <menu id="2" entry="root">
      <preCommands>
        <command>
          <!-- GPRM11 = 0, 71 00 00 0b 00 00 00 00 -->
          <opcode>[0x71][0x00][0x00][0x0b][0x00][0x00][0x00][0x00]</opcode>
        </command>
      </preCommands>

      <postCommands>
        <command>
          <opcode>[0x20][0x04][0x0][0x0][0x0][0x0][0x0][0x01]</opcode>
          <!-- LinkPGCN 1, 20 04 00 00 00 00 00 01 -->
        </command>
      </postCommands>
    </menu>
  </menus>

```

```

<!-- Play Title 1 button -->
<button left="107" top="96" width="166" height="35">
  <command>
    <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 01 00 00 -->
    <opcode>[0x30][0x05][0x00][0x01][0x00][0x01][0x00][0x00]</opcode>
  </command>
</button>

<!-- Play Title 2 button -->
<button left="107" top="155" width="170" height="37">
  <command>
    <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 02 00 00 -->
    <opcode>[0x30][0x05][0x00][0x01][0x00][0x02][0x00][0x00]</opcode>
  </command>
</button>

<!-- Play Title 3 button -->
<button left="107" top="216" width="171" height="36">
  <command>
    <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 03 00 00 -->
    <opcode>[0x30][0x05][0x00][0x01][0x00][0x03][0x00][0x00]</opcode>
  </command>
</button>

<!-- Play All button -->
<button left="107" top="401" width="112" height="36">
  <command>
    <!-- LinkPGCN 2, 20 04 00 00 00 00 02 -->
    <opcode>[0x20][0x04][0x00][0x00][0x00][0x00][0x02]</opcode>
  </command>
</button>

  <mask file="menu\root_menu_mask.bmp" patternColor="#000000"
backgroundColor="#FFFFFF" emphasisColor1="#000001" emphasisColor2="#000002" />
  <display emphasis2Contrast="0" emphasis1Contrast="0" backgroundContrast="0"
patternContrast="0" emphasis2Color="#000000" emphasis1Color="#000000"
patternColor="#000000" />
  <action emphasis2Contrast="0" emphasis1Contrast="0" backgroundContrast="0"
patternContrast="7" emphasis2Color="#000000" emphasis1Color="#000000"
backgroundColor="#000000" patternColor="#0000FF" />
  <selection emphasis2Contrast="0" emphasis1Contrast="0" backgroundContrast="0"
patternContrast="7" emphasis2Color="#000000" emphasis1Color="#000000"
backgroundColor="#000000" patternColor="#FF0000" />
  <background file="menu\root_menu.mpg" />
</menu>

<!-- Play All Dummy menu -->
<menu id="101">
  <preCommands>
    <command>
      <!-- GPRM11 = 1, 71 00 00 0b 00 01 00 00 -->
      <opcode>[0x71][0x00][0x00][0x0b][0x00][0x01][0x00][0x00]</opcode>
    </command>

    <command>
      <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 01 00 00 -->
      <opcode>[0x30][0x05][0x00][0x01][0x00][0x01][0x00][0x00]</opcode>
    </command>
  </preCommands>
</menu>
</menus>
</titleSet>
</dvd>

```

### 3.1.1.2 <firstPlayCommands> Element

Specifies the commands to be executed when the DVD player starts. A DVD starts when a disc is inserted into a DVD player or when the Play button is pressed on the remote control.

```
<firstPlayCommands>
```

```
<command>
</firstPlayCommands>
```

### Child Elements

Element	Description
<command>	Adds a Virtual Machine (VM) command.

### Example

The following example shows a DVD that has title menu, root menu and three titles. The title menu is a dummy one. It simply redirects to the root menu. The root menu supports playing each clip individually and returns back or playing all clips in sequence.

```
<?xml version="1.0" encoding="utf-8"?>
<dvd xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" version="2.1.2.1"
xmlns="http://www.primosoftware.com/dvdbuilder">
  <videoManager>
    <firstPlayCommands>
      <command>
        <!-- JumpSS VTSM (vts 1, tt 1, menu ROOT), 30 06 00 01 01 83 00 00 -->
        <opcode>[0x30][0x06][0x0][0x01][0x01][0x83][0x0][0x0]</opcode>
      </command>
    </firstPlayCommands>

    <menus>
      <menu id="1" entry="title">
        <preCommands>
          <command>
            <!-- JumpSS VTSM (vts 1, tt 1, menu ROOT), 30 06 00 01 01 83 00 00 -->
            <opcode>[0x30][0x06][0x0][0x01][0x01][0x83][0x0][0x0]</opcode>
          </command>
        </preCommands>

        <background file="menu\title_menu.mpg" />
      </menu>
    </menus>

  </videoManager>

  <!-- GPRM11 -> Play All Mode (True=1, False=0) -->

  <titleSet>
    <titles>
      <title id="100" chapters="00:00:00;">
        <videoObject file="../movies/3/columbiatristardvd_pal.mpg" />

        <postCommands>
          <command>
            <!-- if (GPRM11 == 1) Goto 3, 00 a1 00 0b 00 01 00 03 -->
            <opcode>[0x00][0xa1][0x00][0x0b][0x00][0x01][0x00][0x03]</opcode>
          </command>
          <command>
            <!-- CallSS VTSM (menu ROOT, rsm_cell 1), 30 08 00 00 01 83 00 00 -->
            <opcode>[0x30][0x08][0x00][0x00][0x01][0x83][0x00][0x00]</opcode>
          </command>

          <command>
            <!-- JumpVTS_PTT (tt 2, ptt 1), 30 05 00 01 00 02 00 00 -->
            <opcode>[0x30][0x05][0x00][0x01][0x00][0x02][0x00][0x00]</opcode>
          </command>
        </postCommands>

      </title>
```

```

<title id="101" chapters="00:00:00;">
  <videoObject file="../movies/4/dreamworksanim_pal.mpg" />

  <postCommands>
    <command>
      <!-- if (GPRM11 == 1) Goto 3, 00 a1 00 0b 00 01 00 03 -->
      <opcode>[0x00][0xa1][0x00][0x0b][0x00][0x01][0x00][0x03]</opcode>
    </command>
    <command>
      <!-- CallSS VTSM (menu ROOT, rsm_cell 1), 30 08 00 00 01 83 00 00 -->
      <opcode>[0x30][0x08][0x00][0x00][0x01][0x83][0x00][0x00]</opcode>
    </command>

    <command>
      <!-- JumpVTS_PTT (tt 3, ptt 1), 30 05 00 01 00 03 00 00 -->
      <opcode>[0x30][0x05][0x00][0x01][0x00][0x03][0x00][0x00]</opcode>
    </command>
  </postCommands>
</title>

<title id="102" chapters="00:00:00;">
  <videoObject file="../movies/3/columbiatristardvd_pal.mpg" />

  <postCommands>
    <command>
      <!-- CallSS VTSM (menu ROOT, rsm_cell 1), 30 08 00 00 01 83 00 00 -->
      <opcode>[0x30][0x08][0x00][0x00][0x01][0x83][0x00][0x00]</opcode>
    </command>
  </postCommands>
</title>

</titles>
<menus>
  <menu id="2" entry="root">
    <preCommands>
      <command>
        <!-- GPRM11 = 0, 71 00 00 0b 00 00 00 00 -->
        <opcode>[0x71][0x00][0x00][0x0b][0x00][0x00][0x00][0x00]</opcode>
      </command>
    </preCommands>

    <postCommands>
      <command>
        <opcode>[0x20][0x04][0x0][0x0][0x0][0x0][0x0][0x01]</opcode>
        <!-- LinkPGCN 1, 20 04 00 00 00 00 01 -->
      </command>
    </postCommands>

    <!-- Play Title 1 button -->
    <button left="107" top="96" width="166" height="35">
      <command>
        <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 01 00 00 -->
        <opcode>[0x30][0x05][0x00][0x01][0x00][0x01][0x00][0x00]</opcode>
      </command>
    </button>

    <!-- Play Title 2 button -->
    <button left="107" top="155" width="170" height="37">
      <command>
        <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 02 00 00 -->
        <opcode>[0x30][0x05][0x00][0x01][0x00][0x02][0x00][0x00]</opcode>
      </command>
    </button>

    <!-- Play Title 3 button -->
    <button left="107" top="216" width="171" height="36">
      <command>
        <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 03 00 00 -->
        <opcode>[0x30][0x05][0x00][0x01][0x00][0x03][0x00][0x00]</opcode>
      </command>
    </button>
  </menu>

```

```

<!-- Play All button -->
<button left="107" top="401" width="112" height="36">
  <command>
    <!-- LinkPGCN 2, 20 04 00 00 00 00 02 -->
    <opcode>[0x20][0x04][0x0][0x0][0x0][0x0][0x0][0x02]</opcode>
  </command>
</button>

  <mask file="menu\root_menu_mask.bmp" patternColor="#000000"
backgroundColor="#FFFFFF" emphasisColor1="#000001" emphasisColor2="#000002" />
  <display emphasis2Contrast="0" emphasis1Contrast="0" backgroundContrast="0"
patternContrast="0" emphasis2Color="#000000" emphasis1Color="#000000"
patternColor="#000000" />
  <action emphasis2Contrast="0" emphasis1Contrast="0" backgroundContrast="0"
patternContrast="7" emphasis2Color="#000000" emphasis1Color="#000000"
backgroundColor="#000000" patternColor="#0000FF" />
  <selection emphasis2Contrast="0" emphasis1Contrast="0" backgroundContrast="0"
patternContrast="7" emphasis2Color="#000000" emphasis1Color="#000000"
backgroundColor="#000000" patternColor="#FF0000" />
  <background file="menu\root_menu.mpg" />
</menu>

<!-- Play All Dummy menu -->
<menu id="101">
  <preCommands>
    <command>
      <!-- GPRM11 = 1, 71 00 00 0b 00 01 00 00 -->
      <opcode>[0x71][0x00][0x00][0x0b][0x00][0x01][0x00][0x00]</opcode>
    </command>

    <command>
      <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 01 00 00 -->
      <opcode>[0x30][0x05][0x00][0x01][0x00][0x01][0x00][0x00]</opcode>
    </command>
  </preCommands>
</menu>
</menus>
</titleSet>
</dvd>

```

### 3.1.1.2.1 <command> Element

#### Description

Defines a Virtual Machine (VM) command.

```

<command>
<opcode></opcode>
</command>

```

#### Example

The following example shows a DVD that has title menu, root menu and three titles. The title menu is a dummy one. It simply redirects to the root menu. The root menu supports playing each clip individually and returns back or playing all clips in sequence.

```

<?xml version="1.0" encoding="utf-8"?>
<dvd xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" version="2.1.2.1"
xmlns="http://www.primosoftware.com/dvdbuilder">
  <videoManager>
    <firstPlayCommands>
      <command>
        <!-- JumpSS VTSM (vts 1, tt 1, menu ROOT), 30 06 00 01 01 83 00 00 -->
        <opcode>[0x30][0x06][0x0][0x01][0x01][0x83][0x0][0x0]</opcode>
      </command>
    </firstPlayCommands>

```

```

<menus>
  <menu id="1" entry="title">
    <preCommands>
      <command>
        <!-- JumpSS VTSM (vts 1, tt 1, menu ROOT), 30 06 00 01 01 83 00 00 -->
        <opcode>[0x30][0x06][0x00][0x01][0x01][0x83][0x00][0x00]</opcode>
      </command>
    </preCommands>

    <background file="menu\title_menu.mpg" />
  </menu>
</menus>

</videoManager>

<!-- GPRM11 -> Play All Mode (True=1, False=0) -->

<titleSet>
  <titles>
    <title id="100" chapters="00:00:00;">
      <videoObject file="../movies/3/columbiatristardvd_pal.mpg" />

      <postCommands>
        <command>
          <!-- if (GPRM11 == 1) Goto 3, 00 a1 00 0b 00 01 00 03 -->
          <opcode>[0x00][0xa1][0x00][0x0b][0x00][0x01][0x00][0x03]</opcode>
        </command>
        <command>
          <!-- CallSS VTSM (menu ROOT, rsm_cell 1), 30 08 00 00 01 83 00 00 -->
          <opcode>[0x30][0x08][0x00][0x00][0x01][0x83][0x00][0x00]</opcode>
        </command>

        <command>
          <!-- JumpVTS_PTT (tt 2, ptt 1), 30 05 00 01 00 02 00 00 -->
          <opcode>[0x30][0x05][0x00][0x01][0x00][0x02][0x00][0x00]</opcode>
        </command>
      </postCommands>

    </title>

    <title id="101" chapters="00:00:00;">
      <videoObject file="../movies/4/dreamworksanim_pal.mpg" />

      <postCommands>
        <command>
          <!-- if (GPRM11 == 1) Goto 3, 00 a1 00 0b 00 01 00 03 -->
          <opcode>[0x00][0xa1][0x00][0x0b][0x00][0x01][0x00][0x03]</opcode>
        </command>
        <command>
          <!-- CallSS VTSM (menu ROOT, rsm_cell 1), 30 08 00 00 01 83 00 00 -->
          <opcode>[0x30][0x08][0x00][0x00][0x01][0x83][0x00][0x00]</opcode>
        </command>

        <command>
          <!-- JumpVTS_PTT (tt 3, ptt 1), 30 05 00 01 00 03 00 00 -->
          <opcode>[0x30][0x05][0x00][0x01][0x00][0x03][0x00][0x00]</opcode>
        </command>
      </postCommands>
    </title>

    <title id="102" chapters="00:00:00;">
      <videoObject file="../movies/3/columbiatristardvd_pal.mpg" />

      <postCommands>
        <command>
          <!-- CallSS VTSM (menu ROOT, rsm_cell 1), 30 08 00 00 01 83 00 00 -->
          <opcode>[0x30][0x08][0x00][0x00][0x01][0x83][0x00][0x00]</opcode>
        </command>
      </postCommands>
    </title>

  </titles>

```

```

<menu>
  <menu id="2" entry="root">
    <preCommands>
      <command>
        <!-- GPRM11 = 0, 71 00 00 0b 00 00 00 00 -->
        <opcode>[0x71][0x00][0x00][0x0b][0x00][0x00][0x00][0x00]</opcode>
      </command>
    </preCommands>

    <postCommands>
      <command>
        <opcode>[0x20][0x04][0x0][0x0][0x0][0x0][0x0][0x01]</opcode>
        <!-- LinkPGCN 1, 20 04 00 00 00 00 01 -->
      </command>
    </postCommands>

    <!-- Play Title 1 button -->
    <button left="107" top="96" width="166" height="35">
      <command>
        <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 01 00 00 -->
        <opcode>[0x30][0x05][0x00][0x01][0x00][0x01][0x00][0x00]</opcode>
      </command>
    </button>

    <!-- Play Title 2 button -->
    <button left="107" top="155" width="170" height="37">
      <command>
        <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 02 00 00 -->
        <opcode>[0x30][0x05][0x00][0x01][0x00][0x02][0x00][0x00]</opcode>
      </command>
    </button>

    <!-- Play Title 3 button -->
    <button left="107" top="216" width="171" height="36">
      <command>
        <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 03 00 00 -->
        <opcode>[0x30][0x05][0x00][0x01][0x00][0x03][0x00][0x00]</opcode>
      </command>
    </button>

    <!-- Play All button -->
    <button left="107" top="401" width="112" height="36">
      <command>
        <!-- LinkPGCN 2, 20 04 00 00 00 00 02 -->
        <opcode>[0x20][0x04][0x0][0x0][0x0][0x0][0x0][0x02]</opcode>
      </command>
    </button>

    <mask file="menu\root_menu_mask.bmp" patternColor="#000000"
backgroundColor="#FFFFFF" emphasisColor1="#000001" emphasisColor2="#000002" />
    <display emphasis2Contrast="0" emphasis1Contrast="0" backgroundContrast="0"
patternContrast="0" emphasis2Color="#000000" emphasis1Color="#000000"
patternColor="#000000" />
    <action emphasis2Contrast="0" emphasis1Contrast="0" backgroundContrast="0"
patternContrast="7" emphasis2Color="#000000" emphasis1Color="#000000"
backgroundColor="#000000" patternColor="#0000FF" />
    <selection emphasis2Contrast="0" emphasis1Contrast="0" backgroundContrast="0"
patternContrast="7" emphasis2Color="#000000" emphasis1Color="#000000"
backgroundColor="#000000" patternColor="#FF0000" />
    <background file="menu\root_menu.mpg" />
  </menu>

  <!-- Play All Dummy menu -->
  <menu id="101">
    <preCommands>
      <command>
        <!-- GPRM11 = 1, 71 00 00 0b 00 01 00 00 -->
        <opcode>[0x71][0x00][0x00][0x0b][0x00][0x01][0x00][0x00]</opcode>
      </command>

      <command>
        <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 01 00 00 -->

```

```

        <opcode>[0x30][0x05][0x00][0x01][0x00][0x01][0x00][0x00]</opcode>
    </command>
</preCommands>
</menu>
</menus>
</titleSet>
</dvd>

```

### 3.1.1.2.1.1 <opcode> Element

Specifies the Virtual Machine (VM) command opcode. The opcode is 8 bytes long. Each byte is specified in as hexadecimal value [0xXX].

```
<opcode>[0xXX][0xXX][0xXX][0xXX][0xXX][0xXX][0xXX][0xXX]</opcode>
```

#### Remarks

#### Example

The following example shows a DVD that has title menu, root menu and three titles. The title menu is a dummy one. It simply redirects to the root menu. The root menu supports playing each clip individually and returns back or playing all clips in sequence.

```

<?xml version="1.0" encoding="utf-8"?>
<dvd xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" version="2.1.2.1"
xmlns="http://www.primosoftware.com/dvdbuilder">
  <videoManager>
    <firstPlayCommands>
      <command>
        <!-- JumpSS VTSM (vts 1, tt 1, menu ROOT), 30 06 00 01 01 83 00 00 -->
        <opcode>[0x30][0x06][0x00][0x01][0x01][0x83][0x00][0x00]</opcode>
      </command>
    </firstPlayCommands>

    <menus>
      <menu id="1" entry="title">
        <preCommands>
          <command>
            <!-- JumpSS VTSM (vts 1, tt 1, menu ROOT), 30 06 00 01 01 83 00 00 -->
            <opcode>[0x30][0x06][0x00][0x01][0x01][0x83][0x00][0x00]</opcode>
          </command>
        </preCommands>

        <background file="menu\title_menu.mpg" />
      </menu>
    </menus>

  </videoManager>

  <!-- GPRM11 -> Play All Mode (True=1, False=0) -->

  <titleSet>
    <titles>
      <title id="100" chapters="00:00:00;">
        <videoObject file="../movies/3/columbiatristardvd_pal.mpg" />

        <postCommands>
          <command>
            <!-- if (GPRM11 == 1) Goto 3, 00 a1 00 0b 00 01 00 03 -->
            <opcode>[0x00][0xa1][0x00][0x0b][0x00][0x01][0x00][0x03]</opcode>
          </command>
          <command>
            <!-- CallSS VTSM (menu ROOT, rsm_cell 1), 30 08 00 00 01 83 00 00 -->
            <opcode>[0x30][0x08][0x00][0x00][0x01][0x83][0x00][0x00]</opcode>
          </command>
        </postCommands>
      </title>
    </titles>
  </titleSet>

```

```

    <command>
      <!-- JumpVTS_PTT (tt 2, ptt 1), 30 05 00 01 00 02 00 00 -->
      <opcode>[0x30][0x05][0x00][0x01][0x00][0x02][0x00][0x00]</opcode>
    </command>
  </postCommands>
</title>

<title id="101" chapters="00:00:00;">
  <videoObject file="../movies/4/dreamworksanim_pal.mpg" />

  <postCommands>
    <command>
      <!-- if (GPRM11 == 1) Goto 3, 00 a1 00 0b 00 01 00 03 -->
      <opcode>[0x00][0xa1][0x00][0x0b][0x00][0x01][0x00][0x03]</opcode>
    </command>
    <command>
      <!-- CallSS VTSM (menu ROOT, rsm_cell 1), 30 08 00 00 01 83 00 00 -->
      <opcode>[0x30][0x08][0x00][0x00][0x01][0x83][0x00][0x00]</opcode>
    </command>

    <command>
      <!-- JumpVTS_PTT (tt 3, ptt 1), 30 05 00 01 00 03 00 00 -->
      <opcode>[0x30][0x05][0x00][0x01][0x00][0x03][0x00][0x00]</opcode>
    </command>
  </postCommands>
</title>

<title id="102" chapters="00:00:00;">
  <videoObject file="../movies/3/columbiatristardvd_pal.mpg" />

  <postCommands>
    <command>
      <!-- CallSS VTSM (menu ROOT, rsm_cell 1), 30 08 00 00 01 83 00 00 -->
      <opcode>[0x30][0x08][0x00][0x00][0x01][0x83][0x00][0x00]</opcode>
    </command>
  </postCommands>
</title>
</titles>
<menus>
  <menu id="2" entry="root">
    <preCommands>
      <command>
        <!-- GPRM11 = 0, 71 00 00 0b 00 00 00 00 -->
        <opcode>[0x71][0x00][0x00][0x0b][0x00][0x00][0x00][0x00]</opcode>
      </command>
    </preCommands>

    <postCommands>
      <command>
        <opcode>[0x20][0x04][0x0][0x0][0x0][0x0][0x0][0x01]</opcode>
        <!-- LinkPGCN 1, 20 04 00 00 00 00 01 -->
      </command>
    </postCommands>

    <!-- Play Title 1 button -->
    <button left="107" top="96" width="166" height="35">
      <command>
        <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 01 00 00 -->
        <opcode>[0x30][0x05][0x00][0x01][0x00][0x01][0x00][0x00]</opcode>
      </command>
    </button>

    <!-- Play Title 2 button -->
    <button left="107" top="155" width="170" height="37">
      <command>
        <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 02 00 00 -->
        <opcode>[0x30][0x05][0x00][0x01][0x00][0x02][0x00][0x00]</opcode>
      </command>
    </button>

```

```

<!-- Play Title 3 button -->
<button left="107" top="216" width="171" height="36">
  <command>
    <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 03 00 00 -->
    <opcode>[0x30][0x05][0x00][0x01][0x00][0x03][0x00][0x00]</opcode>
  </command>
</button>

<!-- Play All button -->
<button left="107" top="401" width="112" height="36">
  <command>
    <!-- LinkPGCN 2, 20 04 00 00 00 00 02 -->
    <opcode>[0x20][0x04][0x00][0x00][0x00][0x00][0x00][0x02]</opcode>
  </command>
</button>

  <mask file="menu\root_menu_mask.bmp" patternColor="#000000"
backgroundColor="#FFFFFF" emphasisColor1="#000001" emphasisColor2="#000002" />
  <display emphasis2Contrast="0" emphasis1Contrast="0" backgroundContrast="0"
patternContrast="0" emphasis2Color="#000000" emphasis1Color="#000000"
patternColor="#000000" />
  <action emphasis2Contrast="0" emphasis1Contrast="0" backgroundContrast="0"
patternContrast="7" emphasis2Color="#000000" emphasis1Color="#000000"
backgroundColor="#000000" patternColor="#0000FF" />
  <selection emphasis2Contrast="0" emphasis1Contrast="0" backgroundContrast="0"
patternContrast="7" emphasis2Color="#000000" emphasis1Color="#000000"
backgroundColor="#000000" patternColor="#FF0000" />
  <background file="menu\root_menu.mpg" />
</menu>

<!-- Play All Dummy menu -->
<menu id="101">
  <preCommands>
    <command>
      <!-- GPRM11 = 1, 71 00 00 0b 00 01 00 00 -->
      <opcode>[0x71][0x00][0x00][0x0b][0x00][0x01][0x00][0x00]</opcode>
    </command>

    <command>
      <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 01 00 00 -->
      <opcode>[0x30][0x05][0x00][0x01][0x00][0x01][0x00][0x00]</opcode>
    </command>
  </preCommands>
</menu>
</menus>
</titleSet>
</dvd>

```

## 3.1.2 <titleSet> Element

### Description

Configures a Video Title Set.

```

<titleSet>
  <menus>
  <titles>
  <audioStreams>
  <subpictureStreams>
</titleSet>

```

### Child Elements

Element	Description
<menus>	Configures the menus of a title set.
<titles>	Configures the titles of a title set.
<audioStreams> (Version 1.10)	Configures the audio streams of a title set.
<subpictureStreams> (Version 1.10)	Configures the sub-picture/subtitle streams of a title set.

### Remarks

There can be a maximum of 99 Video Title Sets (VTS) numbered from 01 to 99 on a DVD. Each title set can have up to 10 Video Objects (VOB) and menus. Each Video Title Set must have at least 1 Video Object. A Video Title Set is a container for a number of Titles. The maximum number of Titles on a DVD is also 99 - these can be all in one Video Title Set or spread across multiple title sets.

### Example

The following example shows the configuration file of a disc with two titles grouped in two title sets.

```
<dvd version="1.1.0.1" xmlns="http://www.primosoftware.com/dvdbuilder">
  <videoManager firstPlayNavigate="Title=1"/>
  <titleSet>
    <titles>
      <title id="1" chapters="00:00:00;">
        <videoObject file="1.mpg"/>
      </title>
    </titles>
  </titleSet>
  <titleSet>
    <titles>
      <title id="2" chapters="00:00:00;">
        <videoObject file="2.mpg"/>
      </title>
    </titles>
  </titleSet>
</dvd>
```

## 3.1.2.1 <menus> Element

Defines Video Manager (VMG) or Video Title Set (VTS) menus.

```
<menus>
  <menu>
</menus>
```

### Child Elements

Element	Description
<menu>	Adds a Video Manager (VMG) or Video Title Set (VTS) menu.

### 3.1.2.1.1 <menu> Element

#### Description

Defines a Video Manager (VMG) or Video Title Set (VTS) menu.

```

<menu id="num" entry="title|root">
  <background>
  <button>
  <mask>
  <display>
  <selection>
  <action>
</menu>

```

### Required Attributes

Attribute	Option	Description
id	1 to 99999	Menu unique identifier.

### Optional Attributes

Attribute	Option	Description
entry		Specifies the menu type.
	title	<p>Indicates that the menu is the DVD Title menu.</p> <p>The title menu is used as a top-level menu through which the user can access root menus from other VTS on the disc, submenus in the VMG, and video titles available in video title sets.</p> <p>The title menu is displayed when the "Title" button on the remote control is pressed.</p> <p>In general, DVDs can have one title menu for each language unit defined in the VMG. DVDBuilder supports only the default "Unspecified" language unit.</p> <p>The "Unspecified" language unit is selected by the DVD player regardless of the DVD Player own language settings. The Title menu in this case can branch to a language selection menu and then to appropriate submenus of each supported language or to other VTS menus such as root, subpicture and angle menus.</p>
	root	<p>Indicates that the menu is a Video Title Set (VTS) root menu.</p> <p>The Root menu is used as a top-level entry point for other menus in the VTS.</p>

### Child Elements

Element	Description
<background>	Specifies a video clip to be used for the menu background.
<mask>	Specifies a picture mask layer for the menu buttons.

<display>	Specifies the mask patterns colors that are used when the menu buttons are in unselected state.
<selection>	Specifies the mask patterns colors that are used when the menu buttons are in selected state.
<action>	Specifies the mask patterns colors that are used when the menu buttons are in depressed state.
<button>	Defines a menu button (highlight) position and navigation command. There can be up to 36 buttons per one menu.
<preCommands> (Version 2.1.2)	Specifies the VM commands to be executed before a menu is presented.
<postCommands> (Version 2.1.2)	Specifies the VM commands to be executed after a menu is presented.

### Example

The following example shows a DVD with one VMG title menu.

```
<?xml version="1.0" encoding="utf-8"?>
<dvd version="1.1.0.1" xmlns="http://www.primosoftware.com/dvdbuilder">
  <videoManager firstPlayNavigate="Menu=50001;">
    <menus>
      <menu id="50001" entry="title">
        <background file="menu_50001.mpg" />
        <mask
          file="menu_50001_mask.bmp"
          patternColor="#000000"
          backgroundColor="#FFFFFF"
          emphasisColor1="#0000FF"
          emphasisColor2="#00FF00" />

        <display
          patternColor="#FF0000" patternContrast="F"
          backgroundColor="#000000" backgroundContrast="0"
          emphasis1Color="#000000" emphasis1Contrast="0"
          emphasis2Color="#000000" emphasis2Contrast="0"/>

        <selection
          patternColor="#0000FF" patternContrast="F"
          backgroundColor="#000000" backgroundContrast="0"
          emphasis1Color="#000000" emphasis1Contrast="0"
          emphasis2Color="#000000" emphasis2Contrast="0"/>

        <action
          patternColor="#00FF00" patternContrast="F"
          backgroundColor="#000000" backgroundContrast="0"
          emphasis2Color="#000000" emphasis2Contrast="0"
          emphasis1Color="#000000" emphasis1Contrast="0"/>

        <button left="10" top="10" width="74" height="51" navigate="Title=1;" />
      </menu>
    </menus>
  </videoManager>
  <titleSet>
    <titles>
      <title id="1" chapters="00:00:00;00:01:00">
        <videoObject file="1.mpg"/>
      </title>
    </titles>
  </titleSet>
</dvd>
```

The following example shows a DVD with one VMG menu (ID =1). The VMG menu has one button that navigates to a VTS root menu (ID=101). The VTS root menu has one button which navigates to another VTS menu (ID=10101) which has two buttons. Each button command specifies that a chapter from Title with ID =1 should be played. The title itself has a postNavigate command equal to "Menu=101;". That will instruct the DVD Player to display the VTS root menu (ID=101)

when the title playback is completed.

```
<?xml version="1.0" encoding="utf-8"?>
<dvd version="1.1.0.1" xmlns="http://www.primosoftware.com/dvdbuilder">
  <videoManager firstPlayNavigate="Menu=1;">
    <menu>
      <menu id="1" entry="title">
        <background file="vmg_menu_background.mpg" />

        <mask file="vmg_menu_mask.bmp"
          patternColor      ="#000000"
          backgroundColor    ="#FFFFFF"
          emphasisColor1    ="#00FF00"
          emphasisColor2    ="#0000FF" />

        <display
          patternColor      ="#000000" patternContrast    ="0"
          backgroundColor    ="#000000" backgroundContrast ="0"
          emphasis1Color     ="#000000" emphasis1Contrast  ="0"
          emphasis2Color     ="#000000" emphasis2Contrast  ="0" />

        <selection
          patternColor      ="#0000FF" patternContrast    ="15"
          backgroundColor    ="#000000" backgroundContrast ="0"
          emphasis1Color     ="#000000" emphasis1Contrast  ="0"
          emphasis2Color     ="#000000" emphasis2Contrast  ="0" />

        <action
          patternColor      ="#FF0000" patternContrast    ="15"
          backgroundColor    ="#000000" backgroundContrast ="0"
          emphasis1Color     ="#000000" emphasis1Contrast  ="0"
          emphasis2Color     ="#000000" emphasis2Contrast  ="0" />

        <button left="15" top="40" width="112" height="76" navigate="Menu=101;" />
      </menu>
    </menu>
  </videoManager>
  <titleSet>
    <menu>
      <menu id="101" entry="root">
        <background file="vts_root_menu_background.mpg" />

        <mask file="vts_root_menu_mask.bmp"
          patternColor      ="#000000"
          backgroundColor    ="#FFFFFF"
          emphasisColor1    ="#00FF00"
          emphasisColor2    ="#0000FF" />

        <display
          patternColor      ="#000000" patternContrast    ="0"
          backgroundColor    ="#000000" backgroundContrast ="0"
          emphasis1Color     ="#000000" emphasis1Contrast  ="0"
          emphasis2Color     ="#000000" emphasis2Contrast  ="0" />

        <selection
          patternColor      ="#0000FF" patternContrast    ="15"
          backgroundColor    ="#000000" backgroundContrast ="0"
          emphasis1Color     ="#000000" emphasis1Contrast  ="0"
          emphasis2Color     ="#000000" emphasis2Contrast  ="0" />

        <action
          patternColor      ="#FF0000" patternContrast    ="15"
          backgroundColor    ="#000000" backgroundContrast ="0"
          emphasis1Color     ="#000000" emphasis1Contrast  ="0"
          emphasis2Color     ="#000000" emphasis2Contrast  ="0" />

        <button left="15" top="40" width="70" height="50" navigate="Menu=10101;" />
      </menu>
    <menu id="10101">
      <background file="title1_menu_background.mpg" />

      <mask file="title1_menu_mask.bmp" />
    </menu>
  </titleSet>
</dvd>
```

```

        patternColor      = "#000000"
        backgroundColor    = "#FFFFFF"
        emphasisColor1     = "#00FF00"
        emphasisColor2     = "#0000FF" />

<display
  patternColor      = "#000000" patternContrast = "0"
  backgroundColor    = "#000000" backgroundContrast = "0"
  emphasis1Color     = "#000000" emphasis1Contrast = "0"
  emphasis2Color     = "#000000" emphasis2Contrast = "0" />

<selection
  patternColor      = "#0000FF" patternContrast = "15"
  backgroundColor    = "#000000" backgroundContrast = "0"
  emphasis1Color     = "#000000" emphasis1Contrast = "0"
  emphasis2Color     = "#000000" emphasis2Contrast = "0" />

<action
  patternColor      = "#FF0000" patternContrast = "15"
  backgroundColor    = "#000000" backgroundContrast = "0"
  emphasis1Color     = "#000000" emphasis1Contrast = "0"
  emphasis2Color     = "#000000" emphasis2Contrast = "0" />

  <button left="11" top="36" width="78" height="58" navigate="Title = 1; Chapter = 1"
/>
  <button left="96" top="36" width="78" height="58" navigate="Title = 1; Chapter = 2"
/>
</menu>
</menus>
<titles>
  <title id="1" postNavigate="Menu=101;" chapters="00:00:00;00:01:00;">
    <videoObject file="1.mpg" pause="0"/>
  </title>
</titles>
</titleSet>
</dvd>

```

### 3.1.2.1.1.1 <background> Element

#### Description

Specifies a video clip to be used for the menu background.

```

<background
  file="file" />

```

#### Required Attributes

Attribute	Description
file	Specifies a file that contains an MPEG-2 video to be used as the background of a menu. It can be either a still or motion video.

#### Child Elements

Element	Description
<audioStream> (Version 2.1.1.1)	Specifies an audio elementary stream.
<videoAstream> (Version 2.1.1.1)	Specifies a video elementary stream.

**Example**

The following example shows a DVD with one VMG title menu.

```
<?xml version="1.0" encoding="utf-8"?>
<dvd version="1.1.0.1" xmlns="http://www.primosoftware.com/dvdbuilder">
  <videoManager firstPlayNavigate="Menu=50001;">
    <menus>
      <menu id="50001" entry="title">
        <background file="menu_50001.mpg" />
        <mask
          file="menu_50001_mask.bmp"
          patternColor="#000000"
          backgroundColor="#FFFFFF"
          emphasisColor1="#0000FF"
          emphasisColor2="#00FF00" />

        <display
          patternColor    ="#FF0000" patternContrast    ="F"
          backgroundColor="#000000" backgroundContrast="0"
          emphasis1Color  ="#000000" emphasis1Contrast  ="0"
          emphasis2Color  ="#000000" emphasis2Contrast  ="0"/>

        <selection
          patternColor    ="#0000FF" patternContrast    ="F"
          backgroundColor="#000000" backgroundContrast="0"
          emphasis1Color  ="#000000" emphasis1Contrast  ="0"
          emphasis2Color  ="#000000" emphasis2Contrast  ="0"/>

        <action
          patternColor    ="#00FF00" patternContrast    ="F"
          backgroundColor="#000000" backgroundContrast="0"
          emphasis2Color  ="#000000" emphasis2Contrast  ="0"
          emphasis1Color  ="#000000" emphasis1Contrast  ="0"/>

        <button left="10" top="10" width="74" height="51" navigate="Title=1;" />
      </menu>
    </menus>
  </videoManager>
  <titleSet>
    <titles>
      <title id="1" chapters="00:00:00;00:01:00">
        <videoObject file="1.mpg"/>
      </title>
    </titles>
  </titleSet>
</dvd>
```

**3.1.2.1.1.1.1 <videoStream> Element**

Specifies a video stream.

```
<videoStream
  file="file" />
```

**Required Attributes**

Attribute	Description
file	Specifies a MPEG-2 video elementary stream file.

**Example**

The following example shows a DVD with one movie. Note that the version attribute in the <dvd> tag is set to 2.1.1.1, because the <audioStream> and <videoStream> tags require DVDBuilder 2.1.1.1 or later.

```
<?xml version="1.0" encoding="utf-8"?>
<dvd version="2.1.1.1" xmlns="http://www.primosoftware.com/dvdbuilder">
  <videoManager firstPlayNavigate="Title=1;" />
```

```

<titleSet>
  <audioStreams>
    <stream languageCode="EN" mpegStreamID="0xC0" mpegSubstreamID="0x00" />
  </audioStreams>

  <titles>
    <title id="1" chapters="00:00:00">

      <videoObject>
        <audioStream file="movie1.mpa" format="MPA" />
        <videoStream file="movie1.mpv" />
      </videoObject>

    </title>
  </titles>
</titleSet>
</dvd>

```

### Summary

#### 3.1.2.1.1.2 <audioStream> Element

Specifies an audio stream.

```

<audioStream
  file="file" format="format" />

```

#### Required Attributes

Attribute	Description
file	Specifies an audio elementary stream file.
format	Specifies the audio elementary stream format. It must be AC3, MPA or DTS.

#### Example

The following example shows a DVD with one movie. Note that the version attribute in the <dvd> tag is set to 2.1.1.1, because the <audioStream> and <videoStream> tags require DVDBuilder 2.1.1.1 or later.

```

<?xml version="1.0" encoding="utf-8"?>
<dvd version="2.1.1.1" xmlns="http://www.primosoftware.com/dvdbuilder">
  <videoManager firstPlayNavigate="Title=1;" />
  <titleSet>

    <audioStreams>
      <stream languageCode="EN" mpegStreamID="0xC0" mpegSubstreamID="0x00" />
    </audioStreams>

    <titles>
      <title id="1" chapters="00:00:00">

        <videoObject>
          <audioStream file="movie1.mpa" format="MPA" />
          <videoStream file="movie1.mpv" />
        </videoObject>

      </title>
    </titles>
  </titleSet>
</dvd>

```

### Summary

#### 3.1.2.1.1.2 <mask> Element

**Description**

Specifies a picture mask layer for the menu buttons.

```
<mask
  file="file"
  patternColor = "#ffffff"
  backgroundColor = "#ffffff"
  emphasisColor1 = "#ffffff"
  emphasisColor2 = "#ffffff" />
```

**Required Attributes**

Attribute	Description
file	Specifies a file that contains a 24-bit Color Windows Bitmap image. The bitmap image will be used as the mask layer.
patternColor	The color used for the mask primary pattern. This is an RGB color value expressed in hexadecimal format ("#ffffff").
backgroundColor	Mask background color. This is an RGB color value expressed in hexadecimal format ("#ffffff").
emphasisColor1	Mask first emphasis color. This is usually used to create an inner edge for the buttons frame. This is an RGB color value expressed in hexadecimal format ("#ffffff").
emphasisColor2	Mask second emphasis color. This is usually used to create an outer edge for the buttons frame. This is an RGB color value expressed in hexadecimal format ("#ffffff").

**Remarks**

Picture mask layer is usually a black-and-white or up to 4-color background image. In simple black-and-white masks, the background color is the white area of the image and the pattern color is the black patterns of the image. Additionally, in 4-color masks, there can be also the two emphasis colors.

It is important to know that the <mask> colors are used by the DVD player only to see the different patterns in the mask. The actual display color is determined by color and contrast settings in the <display>, <selection> and <action> elements.

**3.1.2.1.1.3 < display > Element****Description**

Specifies the mask patterns colors that are used when the menu buttons are in unselected state.

```
<display
  patternColor = "#ffffff"
  patternContrast = "num"
  backgroundColor = "#ffffff"
  backgroundContrast = "num"
  emphasis1Color = "#ffffff"
  emphasis1Contrast = "num"
  emphasis2Color = "#ffffff"
  emphasis2Contrast = "num" />
```

**Required Attributes**

Attribute	Option	Description
patternColor	This is an RGB color value expressed in hexadecimal format ("#ffffff").	Primary pattern color.
patternContrast	0 to 15	Contrast of the primary pattern. It is ranged from 0 to 15, 0 being totally transparent and 15 being solid.
backgroundColor	This is an RGB color value expressed in hexadecimal format ("#ffffff").	Background color.
backgroundContrast	0 to 15	Contrast of the background. It is ranged from 0 to 15, 0 being totally transparent and 15 being solid.
emphasisColor1	This is an RGB color value expressed in hexadecimal format ("#ffffff").	First emphasis pattern color.
emphasis1Contrast	0 to 15	Contrast of the first emphasis pattern. It is ranged from 0 to 15, 0 being totally transparent and 15 being solid.
emphasisColor2	This is an RGB color value expressed in hexadecimal format ("#ffffff").	Second emphasis pattern color.
emphasis2Contrast	0 to 15	Contrast of the second emphasis pattern. It is ranged from 0 to 15, 0 being totally transparent and 15 being solid.

#### 3.1.2.1.1.4 <selection> Element

##### Description

Specifies the mask patterns colors that are used when the menu buttons are in selected state.

```
<selection
  patternColor = "#ffffff"
  patternContrast = "num"
  backgroundColor = "#ffffff"
  backgroundContrast = "num"
  emphasis1Color = "#ffffff"
  emphasis1Contrast = "num"
  emphasis2Color = "#ffffff"
  emphasis2Contrast = "num" />
```

##### Required Attributes

Attribute	Option	Description
patternColor	This is an RGB color value expressed in hexadecimal format ("#ffffff").	Primary pattern color.
patternContrast	0 to 15	Contrast of the primary pattern. It is ranged from 0 to 15, 0 being totally transparent and 15 being solid.
backgroundColor	This is an RGB color value expressed in hexadecimal format ("#ffffff").	Background color.

backgroundContrast	0 to 15	Contrast of the background. It is ranged from 0 to 15, 0 being totally transparent and 15 being solid.
emphasisColor1	This is an RGB color value expressed in hexadecimal format ("#ffffff").	First emphasis pattern color.
emphasis1Contrast	0 to 15	Contrast of the first emphasis pattern. It is ranged from 0 to 15, 0 being totally transparent and 15 being solid.
emphasisColor2	This is an RGB color value expressed in hexadecimal format ("#ffffff").	Second emphasis pattern color.
emphasis2Contrast	0 to 15	Contrast of the second emphasis pattern. It is ranged from 0 to 15, 0 being totally transparent and 15 being solid.

### 3.1.2.1.1.5 <action> Element

#### Description

Specifies the mask patterns colors that are used when the menu buttons are in depressed state.

```
<action
  patternColor = "#ffffff"
  patternContrast = "num"
  backgroundColor = "#ffffff"
  backgroundContrast = "num"
  emphasis1Color = "#ffffff"
  emphasis1Contrast = "num"
  emphasis2Color = "#ffffff"
  emphasis2Contrast = "num" />
```

#### Required Attributes

Attribute	Option	Description
patternColor	This is an RGB color value expressed in hexadecimal format ("#ffffff").	Primary pattern color.
patternContrast	0 to 15	Contrast of the primary pattern. It is ranged from 0 to 15, 0 being totally transparent and 15 being solid.
backgroundColor	This is an RGB color value expressed in hexadecimal format ("#ffffff").	Background color.
backgroundContrast	0 to 15	Contrast of the background. It is ranged from 0 to 15, 0 being totally transparent and 15 being solid.
emphasisColor1	This is an RGB color value expressed in hexadecimal format ("#ffffff").	First emphasis pattern color.
emphasis1Contrast	0 to 15	Contrast of the first emphasis pattern. It is ranged from 0 to 15, 0 being totally transparent and 15 being solid.

emphasisColor2	This is an RGB color value expressed in hexadecimal format ("#ffffff").	Second emphasis pattern color.
emphasis2Contrast	0 to 15	Contrast of the second emphasis pattern. It is ranged from 0 to 15, 0 being totally transparent and 15 being solid.

### 3.1.2.1.1.6 <button> Element

#### Description

Defines a highlight area for a menu button.

```
<button
  left="num"
  top="num"
  width="num"
  height="num"
  navigate="Menu=id;|Title=id;|Title=id; Chapter=num;" />
```

#### Required Attributes

Attribute	Option	Description
left	1 to 720 (NTSC) 1 to 720 (PAL)	Specifies the left position of the button rectangle.
top	1 to 480 (NTSC) 1 to 576 (PAL)	Specifies the top position of the button rectangle.
width	1 to 720 (NTSC) 1 to 720 (PAL)	Specifies the width of the button rectangle.
height	1 to 480 (NTSC) 1 to 576 (PAL)	Specifies the height of the button rectangle.
navigate	Menu=id;	Navigates to a menu with ID "id". Example: Menu=50001;
	Title=id;	Navigates to a title with ID "id". Example: Title=1;
	Title=id; Chapter=num;	Navigates to Chapter with number "num" from Title with ID "id". This navigation command can be used only in titleSet (VTS) menus. Example: Title=1;Chapter=1;

#### Child Elements

Element	Description
<command>	Specifies the VM command to be executed when the button is activated.

#### Remarks

When a button (or highlight area) is selected, the portion of the mask enclosed in button's rectangular area is used to decide what colors to display. DVD specification allows up to 36 buttons per display menu.

If <command>child element is specified then the attribute "navigate" is not required. If both <command>child element and attribute "navigate" are specified then the program will use only the <command>child element.

### Example

The following example shows a DVD that has one VMG menu with one button. When the button is selected the DVD player will play the contents of chapter 1 from title 1.

```
<?xml version="1.0" encoding="utf-8"?>
<dvd version="1.1.0.1" xmlns="http://www.primosoftware.com/dvdbuilder">
  <videoManager firstPlayNavigate="Menu=50001;">
    <menus>
      <menu id="50001" entry="title">

        <button left="10" top="10" width="74" height="51" navigate="Title=1;Chapter=1;" />

        <background file="menu_50001.mpg" />

        <mask
          file="menu_50001_mask.bmp"
          patternColor="#FFFFFF"
          backgroundColor="#000000"
          emphasisColor1="#000001"
          emphasisColor2="#000001" />

        <display
          patternColor="#000000" patternContrast="#000000"
          backgroundColor="#000000" backgroundContrast="#000000"
          emphasis1Color="#000000" emphasis1Contrast="#000000"
          emphasis2Color="#000000" emphasis2Contrast="#000000"/>

        <action
          patternColor="#0000FF" patternContrast="7"
          backgroundColor="#000000" backgroundContrast="#000000"
          emphasis2Color="#000000" emphasis2Contrast="#000000"
          emphasis1Color="#000000" emphasis1Contrast="#000000"/>

        <selection
          patternColor="#FF0000" patternContrast="7"
          backgroundColor="#000000" backgroundContrast="#000000"
          emphasis1Color="#000000" emphasis1Contrast="#000000"
          emphasis2Color="#000000" emphasis2Contrast="#000000"/>
      </menu>
    </menus>
  </videoManager>
  <titleSet>
    <titles>
      <title id="1" chapters="00:00:00;">
        <videoObject file="1.mpg"/>
      </title>
    </titles>
  </titleSet>
</dvd>
```

#### 3.1.2.1.1.6.1 <command> Element

##### Description

Defines a Virtual Machine (VM) command.

```
<command>
<opcode></opcode>
</command>
```

##### Example

The following example shows a DVD that has title menu, root

menu and three titles. The title menu is a dummy one. It simply redirects to the root menu. The root menu supports playing each clip individually and returns back or playing all clips in sequence.

```
<?xml version="1.0" encoding="utf-8"?>
<dvd xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" version="2.1.2.1"
xmlns="http://www.primosoftware.com/dvdbuilder">
  <videoManager>
    <firstPlayCommands>
      <command>
        <!-- JumpSS VTSM (vts 1, tt 1, menu ROOT), 30 06 00 01 01 83 00 00 -->
        <opcode>[0x30][0x06][0x0][0x01][0x01][0x83][0x0][0x0]</opcode>
      </command>
    </firstPlayCommands>

    <menus>
      <menu id="1" entry="title">
        <preCommands>
          <command>
            <!-- JumpSS VTSM (vts 1, tt 1, menu ROOT), 30 06 00 01 01 83 00 00 -->
            <opcode>[0x30][0x06][0x0][0x01][0x01][0x83][0x0][0x0]</opcode>
          </command>
        </preCommands>

        <background file="menu\title_menu.mpg" />
      </menu>
    </menus>

  </videoManager>

  <!-- GPRM11 -> Play All Mode (True=1, False=0) -->

  <titleSet>
    <titles>
      <title id="100" chapters="00:00:00;">
        <videoObject file="..\movies\3\columbiatristardvd_pal.mpg" />

        <postCommands>
          <command>
            <!-- if (GPRM11 == 1) Goto 3, 00 a1 00 0b 00 01 00 03 -->
            <opcode>[0x00][0xa1][0x00][0x0b][0x00][0x01][0x00][0x03]</opcode>
          </command>
          <command>
            <!-- CallSS VTSM (menu ROOT, rsm_cell 1), 30 08 00 00 01 83 00 00 -->
            <opcode>[0x30][0x08][0x00][0x00][0x01][0x83][0x00][0x00]</opcode>
          </command>

          <command>
            <!-- JumpVTS_PTT (tt 2, ptt 1), 30 05 00 01 00 02 00 00 -->
            <opcode>[0x30][0x05][0x00][0x01][0x00][0x02][0x00][0x00]</opcode>
          </command>
        </postCommands>
      </title>

      <title id="101" chapters="00:00:00;">
        <videoObject file="..\movies\4\dreamworksanim_pal.mpg" />

        <postCommands>
          <command>
            <!-- if (GPRM11 == 1) Goto 3, 00 a1 00 0b 00 01 00 03 -->
            <opcode>[0x00][0xa1][0x00][0x0b][0x00][0x01][0x00][0x03]</opcode>
          </command>
          <command>
            <!-- CallSS VTSM (menu ROOT, rsm_cell 1), 30 08 00 00 01 83 00 00 -->
            <opcode>[0x30][0x08][0x00][0x00][0x01][0x83][0x00][0x00]</opcode>
          </command>

          <command>
            <!-- JumpVTS_PTT (tt 3, ptt 1), 30 05 00 01 00 03 00 00 -->
```

```

        <opcode>[0x30][0x05][0x00][0x01][0x00][0x03][0x00][0x00]</opcode>
    </command>
</postCommands>
</title>

<title id="102" chapters="00:00:00;">
    <videoObject file="../movies/3/columbiatristardvd_pal.mpg" />

    <postCommands>
        <command>
            <!-- CallSS VTSM (menu ROOT, rsm_cell 1), 30 08 00 00 01 83 00 00 -->
            <opcode>[0x30][0x08][0x00][0x00][0x01][0x83][0x00][0x00]</opcode>
        </command>
    </postCommands>
</title>

</titles>
<menus>
    <menu id="2" entry="root">
        <preCommands>
            <command>
                <!-- GPRM11 = 0, 71 00 00 0b 00 00 00 00 -->
                <opcode>[0x71][0x00][0x00][0x0b][0x00][0x00][0x00][0x00]</opcode>
            </command>
        </preCommands>

        <postCommands>
            <command>
                <opcode>[0x20][0x04][0x0][0x0][0x0][0x0][0x0][0x01]</opcode>
                <!-- LinkPGCN 1, 20 04 00 00 00 00 01 -->
            </command>
        </postCommands>

        <!-- Play Title 1 button -->
        <button left="107" top="96" width="166" height="35">
            <command>
                <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 01 00 00 -->
                <opcode>[0x30][0x05][0x00][0x01][0x00][0x01][0x00][0x00]</opcode>
            </command>
        </button>

        <!-- Play Title 2 button -->
        <button left="107" top="155" width="170" height="37">
            <command>
                <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 02 00 00 -->
                <opcode>[0x30][0x05][0x00][0x01][0x00][0x02][0x00][0x00]</opcode>
            </command>
        </button>

        <!-- Play Title 3 button -->
        <button left="107" top="216" width="171" height="36">
            <command>
                <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 03 00 00 -->
                <opcode>[0x30][0x05][0x00][0x01][0x00][0x03][0x00][0x00]</opcode>
            </command>
        </button>

        <!-- Play All button -->
        <button left="107" top="401" width="112" height="36">
            <command>
                <!-- LinkPGCN 2, 20 04 00 00 00 00 02 -->
                <opcode>[0x20][0x04][0x0][0x0][0x0][0x0][0x0][0x02]</opcode>
            </command>
        </button>

        <mask file="menu\root_menu_mask.bmp" patternColor="#000000"
backgroundcolor="#FFFFFF" emphasisColor1="#000001" emphasisColor2="#000002" />
        <display emphasis2Contrast="0" emphasis1Contrast="0" backgroundContrast="0"
patternContrast="0" emphasis2Color="#000000" emphasis1Color="#000000"
patternColor="#000000" />
        <action emphasis2Contrast="0" emphasis1Contrast="0" backgroundContrast="0"
patternContrast="7" emphasis2Color="#000000" emphasis1Color="#000000"

```

```

backgroundColor="#000000" patternColor="#0000FF" />
  <selection emphasis2Contrast="0" emphasis1Contrast="0" backgroundContrast="0"
patternContrast="7" emphasis2Color="#000000" emphasis1Color="#000000"
backgroundColor="#000000" patternColor="#FF0000" />
  <background file="menu\root_menu.mpg" />
</menu>

<!-- Play All Dummy menu -->
<menu id="101">
  <preCommands>
    <command>
      <!-- GPRM11 = 1, 71 00 00 0b 00 01 00 00 -->
      <opcode>[0x71][0x00][0x00][0x0b][0x00][0x01][0x00][0x00]</opcode>
    </command>

    <command>
      <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 01 00 00 -->
      <opcode>[0x30][0x05][0x00][0x01][0x00][0x01][0x00][0x00]</opcode>
    </command>
  </preCommands>
</menu>
</menus>
</titleSet>
</dvd>

```

### 3.1.2.1.1.6.1.1 <opcode> Element

Specifies the Virtual Machine (VM) command opcode. The opcode is 8 bytes long. Each byte is specified in as hexadecimal value [0xXX].

```
<opcode>[0xXX][0xXX][0xXX][0xXX][0xXX][0xXX][0xXX][0xXX]</opcode>
```

#### Remarks

#### Example

The following example shows a DVD that has title menu, root menu and three titles. The title menu is a dummy one. It simply redirects to the root menu. The root menu supports playing each clip individually and returns back or playing all clips in sequence.

```

<?xml version="1.0" encoding="utf-8"?>
<dvd xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" version="2.1.2.1"
xmlns="http://www.primosoftware.com/dvdbuilder">
  <videoManager>
    <firstPlayCommands>
      <command>
        <!-- JumpSS VTSM (vts 1, tt 1, menu ROOT), 30 06 00 01 01 83 00 00 -->
        <opcode>[0x30][0x06][0x00][0x01][0x01][0x83][0x00][0x00]</opcode>
      </command>
    </firstPlayCommands>

    <menus>
      <menu id="1" entry="title">
        <preCommands>
          <command>
            <!-- JumpSS VTSM (vts 1, tt 1, menu ROOT), 30 06 00 01 01 83 00 00 -->
            <opcode>[0x30][0x06][0x00][0x01][0x01][0x83][0x00][0x00]</opcode>
          </command>
        </preCommands>

        <background file="menu\title_menu.mpg" />
      </menu>
    </menus>
  </videoManager>

```

```

<!-- GPRM11 -> Play All Mode (True=1, False=0) -->
<titleSet>
  <titles>
    <title id="100" chapters="00:00:00;">
      <videoObject file="../movies/3/columbiatristardvd_pal.mpg" />

      <postCommands>
        <command>
          <!-- if (GPRM11 == 1) Goto 3, 00 a1 00 0b 00 01 00 03 -->
          <opcode>[0x00][0xa1][0x00][0x0b][0x00][0x01][0x00][0x03]</opcode>
        </command>
        <command>
          <!-- CallSS VTSM (menu ROOT, rsm_cell 1), 30 08 00 00 01 83 00 00 -->
          <opcode>[0x30][0x08][0x00][0x00][0x01][0x83][0x00][0x00]</opcode>
        </command>

        <command>
          <!-- JumpVTS_PTT (tt 2, ptt 1), 30 05 00 01 00 02 00 00 -->
          <opcode>[0x30][0x05][0x00][0x01][0x00][0x02][0x00][0x00]</opcode>
        </command>
      </postCommands>

    </title>

    <title id="101" chapters="00:00:00;">
      <videoObject file="../movies/4/dreamworksanim_pal.mpg" />

      <postCommands>
        <command>
          <!-- if (GPRM11 == 1) Goto 3, 00 a1 00 0b 00 01 00 03 -->
          <opcode>[0x00][0xa1][0x00][0x0b][0x00][0x01][0x00][0x03]</opcode>
        </command>
        <command>
          <!-- CallSS VTSM (menu ROOT, rsm_cell 1), 30 08 00 00 01 83 00 00 -->
          <opcode>[0x30][0x08][0x00][0x00][0x01][0x83][0x00][0x00]</opcode>
        </command>

        <command>
          <!-- JumpVTS_PTT (tt 3, ptt 1), 30 05 00 01 00 03 00 00 -->
          <opcode>[0x30][0x05][0x00][0x01][0x00][0x03][0x00][0x00]</opcode>
        </command>
      </postCommands>
    </title>

    <title id="102" chapters="00:00:00;">
      <videoObject file="../movies/3/columbiatristardvd_pal.mpg" />

      <postCommands>
        <command>
          <!-- CallSS VTSM (menu ROOT, rsm_cell 1), 30 08 00 00 01 83 00 00 -->
          <opcode>[0x30][0x08][0x00][0x00][0x01][0x83][0x00][0x00]</opcode>
        </command>
      </postCommands>
    </title>

  </titles>
  <menus>
    <menu id="2" entry="root">
      <preCommands>
        <command>
          <!-- GPRM11 = 0, 71 00 00 0b 00 00 00 00 -->
          <opcode>[0x71][0x00][0x00][0x0b][0x00][0x00][0x00][0x00]</opcode>
        </command>
      </preCommands>

      <postCommands>
        <command>
          <opcode>[0x20][0x04][0x0][0x0][0x0][0x0][0x0][0x01]</opcode>
          <!-- LinkPGCN 1, 20 04 00 00 00 00 01 -->
        </command>
      </postCommands>
    </menu>
  </menus>

```

```

<!-- Play Title 1 button -->
<button left="107" top="96" width="166" height="35">
  <command>
    <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 01 00 00 -->
    <opcode>[0x30][0x05][0x00][0x01][0x00][0x01][0x00][0x00]</opcode>
  </command>
</button>

<!-- Play Title 2 button -->
<button left="107" top="155" width="170" height="37">
  <command>
    <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 02 00 00 -->
    <opcode>[0x30][0x05][0x00][0x01][0x00][0x02][0x00][0x00]</opcode>
  </command>
</button>

<!-- Play Title 3 button -->
<button left="107" top="216" width="171" height="36">
  <command>
    <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 03 00 00 -->
    <opcode>[0x30][0x05][0x00][0x01][0x00][0x03][0x00][0x00]</opcode>
  </command>
</button>

<!-- Play All button -->
<button left="107" top="401" width="112" height="36">
  <command>
    <!-- LinkPGCN 2, 20 04 00 00 00 00 02 -->
    <opcode>[0x20][0x04][0x00][0x00][0x00][0x00][0x00][0x02]</opcode>
  </command>
</button>

  <mask file="menu\root_menu_mask.bmp" patternColor="#000000"
backgroundColor="#FFFFFF" emphasisColor1="#000001" emphasisColor2="#000002" />
  <display emphasis2Contrast="0" emphasis1Contrast="0" backgroundContrast="0"
patternContrast="0" emphasis2Color="#000000" emphasis1Color="#000000"
patternColor="#000000" />
  <action emphasis2Contrast="0" emphasis1Contrast="0" backgroundContrast="0"
patternContrast="7" emphasis2Color="#000000" emphasis1Color="#000000"
backgroundColor="#000000" patternColor="#0000FF" />
  <selection emphasis2Contrast="0" emphasis1Contrast="0" backgroundContrast="0"
patternContrast="7" emphasis2Color="#000000" emphasis1Color="#000000"
backgroundColor="#000000" patternColor="#FF0000" />
  <background file="menu\root_menu.mpg" />
</menu>

<!-- Play All Dummy menu -->
<menu id="101">
  <preCommands>
    <command>
      <!-- GPRM11 = 1, 71 00 00 0b 00 01 00 00 -->
      <opcode>[0x71][0x00][0x00][0x0b][0x00][0x01][0x00][0x00]</opcode>
    </command>

    <command>
      <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 01 00 00 -->
      <opcode>[0x30][0x05][0x00][0x01][0x00][0x01][0x00][0x00]</opcode>
    </command>
  </preCommands>
</menu>
</menus>
</titleSet>
</dvd>

```

### 3.1.2.1.1.7 <postCommands> Element

Specifies the VM commands to be executed after a menu or a title is presented.

```

<preCommands>
  <command/>

```

```
</preCommands>
```

### Child Elements

Element	Description
<command> (Version 2.1.2)	Specifies a VM command.

### Example

The following example shows a DVD that has title menu, root menu and three titles. The title menu is a dummy one. It simply redirects to the root menu. The root menu supports playing each clip individually and returns back or playing all clips in sequence.

```
<?xml version="1.0" encoding="utf-8"?>
<dvd xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" version="2.1.2.1"
xmlns="http://www.primosoftware.com/dvdbuilder">
  <videoManager>
    <firstPlayCommands>
      <command>
        <!-- JumpSS VTSM (vts 1, tt 1, menu ROOT), 30 06 00 01 01 83 00 00 -->
        <opcode>[0x30][0x06][0x00][0x01][0x01][0x83][0x00][0x00]</opcode>
      </command>
    </firstPlayCommands>

    <menus>
      <menu id="1" entry="title">
        <preCommands>
          <command>
            <!-- JumpSS VTSM (vts 1, tt 1, menu ROOT), 30 06 00 01 01 83 00 00 -->
            <opcode>[0x30][0x06][0x00][0x01][0x01][0x83][0x00][0x00]</opcode>
          </command>
        </preCommands>

        <background file="menu\title_menu.mpg" />
      </menu>
    </menus>

  </videoManager>

  <!-- GPRM11 -> Play All Mode (True=1, False=0) -->

  <titleSet>
    <titles>
      <title id="100" chapters="00:00:00;">
        <videoObject file="../movies/3/columbiatristardvd_pal.mpg" />

        <postCommands>
          <command>
            <!-- if (GPRM11 == 1) Goto 3, 00 a1 00 0b 00 01 00 03 -->
            <opcode>[0x00][0xa1][0x00][0x0b][0x00][0x01][0x00][0x03]</opcode>
          </command>
          <command>
            <!-- CallSS VTSM (menu ROOT, rsm_cell 1), 30 08 00 00 01 83 00 00 -->
            <opcode>[0x30][0x08][0x00][0x00][0x01][0x83][0x00][0x00]</opcode>
          </command>

          <command>
            <!-- JumpVTS_PTT (tt 2, ptt 1), 30 05 00 01 00 02 00 00 -->
            <opcode>[0x30][0x05][0x00][0x01][0x00][0x02][0x00][0x00]</opcode>
          </command>
        </postCommands>

      </title>

      <title id="101" chapters="00:00:00;">
        <videoObject file="../movies/4/dreamworksanim_pal.mpg" />
      </title>
    </titles>
  </titleSet>
</dvd>
```

```

<postCommands>
  <command>
    <!-- if (GPRM11 == 1) Goto 3, 00 a1 00 0b 00 01 00 03 -->
    <opcode>[0x00][0xa1][0x00][0x0b][0x00][0x01][0x00][0x03]</opcode>
  </command>
  <command>
    <!-- CallSS VTSM (menu ROOT, rsm_cell 1), 30 08 00 00 01 83 00 00 -->
    <opcode>[0x30][0x08][0x00][0x00][0x01][0x83][0x00][0x00]</opcode>
  </command>

  <command>
    <!-- JumpVTS_PTT (tt 3, ptt 1), 30 05 00 01 00 03 00 00 -->
    <opcode>[0x30][0x05][0x00][0x01][0x00][0x03][0x00][0x00]</opcode>
  </command>
</postCommands>
</title>

<title id="102" chapters="00:00:00;">
  <videoObject file="../movies/3/columbiatristardvd_pal.mpg" />

  <postCommands>
    <command>
      <!-- CallSS VTSM (menu ROOT, rsm_cell 1), 30 08 00 00 01 83 00 00 -->
      <opcode>[0x30][0x08][0x00][0x00][0x01][0x83][0x00][0x00]</opcode>
    </command>
  </postCommands>
</title>

</titles>
<menus>
  <menu id="2" entry="root">
    <preCommands>
      <command>
        <!-- GPRM11 = 0, 71 00 00 0b 00 00 00 00 -->
        <opcode>[0x71][0x00][0x00][0x0b][0x00][0x00][0x00][0x00]</opcode>
      </command>
    </preCommands>

    <postCommands>
      <command>
        <opcode>[0x20][0x04][0x0][0x0][0x0][0x0][0x0][0x01]</opcode>
        <!-- LinkPGCN 1, 20 04 00 00 00 00 01 -->
      </command>
    </postCommands>

    <!-- Play Title 1 button -->
    <button left="107" top="96" width="166" height="35">
      <command>
        <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 01 00 00 -->
        <opcode>[0x30][0x05][0x00][0x01][0x00][0x01][0x00][0x00]</opcode>
      </command>
    </button>

    <!-- Play Title 2 button -->
    <button left="107" top="155" width="170" height="37">
      <command>
        <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 02 00 00 -->
        <opcode>[0x30][0x05][0x00][0x01][0x00][0x02][0x00][0x00]</opcode>
      </command>
    </button>

    <!-- Play Title 3 button -->
    <button left="107" top="216" width="171" height="36">
      <command>
        <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 03 00 00 -->
        <opcode>[0x30][0x05][0x00][0x01][0x00][0x03][0x00][0x00]</opcode>
      </command>
    </button>

    <!-- Play All button -->
    <button left="107" top="401" width="112" height="36">

```

```

    <command>
      <!-- LinkPGCN 2, 20 04 00 00 00 00 02 -->
      <opcode>[0x20][0x04][0x0][0x0][0x0][0x0][0x0][0x02]</opcode>
    </command>
  </button>

  <mask file="menu\root_menu_mask.bmp" patternColor="#000000"
backgroundColor="#FFFFFF" emphasisColor1="#000001" emphasisColor2="#000002" />
  <display emphasis2Contrast="0" emphasis1Contrast="0" backgroundContrast="0"
patternContrast="0" emphasis2Color="#000000" emphasis1Color="#000000"
patternColor="#000000" />
  <action emphasis2Contrast="0" emphasis1Contrast="0" backgroundContrast="0"
patternContrast="7" emphasis2Color="#000000" emphasis1Color="#000000"
backgroundColor="#000000" patternColor="#0000FF" />
  <selection emphasis2Contrast="0" emphasis1Contrast="0" backgroundContrast="0"
patternContrast="7" emphasis2Color="#000000" emphasis1Color="#000000"
backgroundColor="#000000" patternColor="#FF0000" />
  <background file="menu\root_menu.mpg" />
</menu>

<!-- Play All Dummy menu -->
<menu id="101">
  <preCommands>
    <command>
      <!-- GPRM11 = 1, 71 00 00 0b 00 01 00 00 -->
      <opcode>[0x71][0x00][0x00][0x0b][0x00][0x01][0x00][0x00]</opcode>
    </command>

    <command>
      <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 01 00 00 -->
      <opcode>[0x30][0x05][0x00][0x01][0x00][0x01][0x00][0x00]</opcode>
    </command>
  </preCommands>
</menu>
</menus>
</titleSet>
</dvd>

```

### 3.1.2.1.1.7.1 <command> Element

#### Description

Defines a Virtual Machine (VM) command.

```

<command>
<opcode></opcode>
</command>

```

#### Example

The following example shows a DVD that has title menu, root menu and three titles. The title menu is a dummy one. It simply redirects to the root menu. The root menu supports playing each clip individually and returns back or playing all clips in sequence.

```

<?xml version="1.0" encoding="utf-8"?>
<dvd xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" version="2.1.2.1"
xmlns="http://www.primosoftware.com/dvdbuilder">
  <videoManager>
    <firstPlayCommands>
      <command>
        <!-- JumpSS VTSM (vts 1, tt 1, menu ROOT), 30 06 00 01 01 83 00 00 -->
        <opcode>[0x30][0x06][0x0][0x01][0x01][0x83][0x0][0x0]</opcode>
      </command>
    </firstPlayCommands>

    <menus>
      <menu id="1" entry="title">
        <preCommands>

```

```

    <command>
      <!-- JumpSS VTSM (vts 1, tt 1, menu ROOT), 30 06 00 01 01 83 00 00 -->
      <opcode>[0x30][0x06][0x0][0x01][0x01][0x83][0x0][0x0]</opcode>
    </command>
  </preCommands>

  <background file="menu\title_menu.mpg" />
</menu>
</menus>

</videoManager>

<!-- GPRM11 -> Play All Mode (True=1, False=0) -->

<titleSet>
  <titles>
    <title id="100" chapters="00:00:00;">
      <videoObject file="../movies/3/columbiatristardvd_pal.mpg" />

      <postCommands>
        <command>
          <!-- if (GPRM11 == 1) Goto 3, 00 a1 00 0b 00 01 00 03 -->
          <opcode>[0x00][0xa1][0x00][0x0b][0x00][0x01][0x00][0x03]</opcode>
        </command>
        <command>
          <!-- CallSS VTSM (menu ROOT, rsm_cell 1), 30 08 00 00 01 83 00 00 -->
          <opcode>[0x30][0x08][0x00][0x00][0x01][0x83][0x00][0x00]</opcode>
        </command>

        <command>
          <!-- JumpVTS_PTT (tt 2, ptt 1), 30 05 00 01 00 02 00 00 -->
          <opcode>[0x30][0x05][0x00][0x01][0x00][0x02][0x00][0x00]</opcode>
        </command>
      </postCommands>

    </title>

    <title id="101" chapters="00:00:00;">
      <videoObject file="../movies/4/dreamworksanim_pal.mpg" />

      <postCommands>
        <command>
          <!-- if (GPRM11 == 1) Goto 3, 00 a1 00 0b 00 01 00 03 -->
          <opcode>[0x00][0xa1][0x00][0x0b][0x00][0x01][0x00][0x03]</opcode>
        </command>
        <command>
          <!-- CallSS VTSM (menu ROOT, rsm_cell 1), 30 08 00 00 01 83 00 00 -->
          <opcode>[0x30][0x08][0x00][0x00][0x01][0x83][0x00][0x00]</opcode>
        </command>

        <command>
          <!-- JumpVTS_PTT (tt 3, ptt 1), 30 05 00 01 00 03 00 00 -->
          <opcode>[0x30][0x05][0x00][0x01][0x00][0x03][0x00][0x00]</opcode>
        </command>
      </postCommands>
    </title>

    <title id="102" chapters="00:00:00;">
      <videoObject file="../movies/3/columbiatristardvd_pal.mpg" />

      <postCommands>
        <command>
          <!-- CallSS VTSM (menu ROOT, rsm_cell 1), 30 08 00 00 01 83 00 00 -->
          <opcode>[0x30][0x08][0x00][0x00][0x01][0x83][0x00][0x00]</opcode>
        </command>
      </postCommands>
    </title>
  </titles>
</menus>
  <menu id="2" entry="root">
    <preCommands>

```

```

    <command>
      <!-- GPRM11 = 0, 71 00 00 0b 00 00 00 00 -->
      <opcode>[0x71][0x00][0x00][0x0b][0x00][0x00][0x00][0x00]</opcode>
    </command>
  </preCommands>

  <postCommands>
    <command>
      <opcode>[0x20][0x04][0x0][0x0][0x0][0x0][0x0][0x01]</opcode>
      <!-- LinkPGCN 1, 20 04 00 00 00 00 01 -->
    </command>
  </postCommands>

  <!-- Play Title 1 button -->
  <button left="107" top="96" width="166" height="35">
    <command>
      <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 01 00 00 -->
      <opcode>[0x30][0x05][0x00][0x01][0x00][0x01][0x00][0x00]</opcode>
    </command>
  </button>

  <!-- Play Title 2 button -->
  <button left="107" top="155" width="170" height="37">
    <command>
      <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 02 00 00 -->
      <opcode>[0x30][0x05][0x00][0x01][0x00][0x02][0x00][0x00]</opcode>
    </command>
  </button>

  <!-- Play Title 3 button -->
  <button left="107" top="216" width="171" height="36">
    <command>
      <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 03 00 00 -->
      <opcode>[0x30][0x05][0x00][0x01][0x00][0x03][0x00][0x00]</opcode>
    </command>
  </button>

  <!-- Play All button -->
  <button left="107" top="401" width="112" height="36">
    <command>
      <!-- LinkPGCN 2, 20 04 00 00 00 00 02 -->
      <opcode>[0x20][0x04][0x0][0x0][0x0][0x0][0x0][0x02]</opcode>
    </command>
  </button>

  <mask file="menu\root_menu_mask.bmp" patternColor="#000000"
backgroundcolor="#FFFFFF" emphasisColor1="#000001" emphasisColor2="#000002" />
  <display emphasis2Contrast="0" emphasis1Contrast="0" backgroundContrast="0"
patternContrast="0" emphasis2Color="#000000" emphasis1Color="#000000"
patternColor="#000000" />
  <action emphasis2Contrast="0" emphasis1Contrast="0" backgroundContrast="0"
patternContrast="7" emphasis2Color="#000000" emphasis1Color="#000000"
backgroundcolor="#000000" patternColor="#0000FF" />
  <selection emphasis2Contrast="0" emphasis1Contrast="0" backgroundContrast="0"
patternContrast="7" emphasis2Color="#000000" emphasis1Color="#000000"
backgroundcolor="#000000" patternColor="#FF0000" />
  <background file="menu\root_menu.mpg" />
</menu>

  <!-- Play All Dummy menu -->
  <menu id="101">
    <preCommands>
      <command>
        <!-- GPRM11 = 1, 71 00 00 0b 00 01 00 00 -->
        <opcode>[0x71][0x00][0x00][0x0b][0x00][0x01][0x00][0x00]</opcode>
      </command>

      <command>
        <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 01 00 00 -->
        <opcode>[0x30][0x05][0x00][0x01][0x00][0x01][0x00][0x00]</opcode>
      </command>
    </preCommands>

```

```

    </menu>
  </menus>
</titleSet>
</dvd>

```

### 3.1.2.1.1.7.1.1 <opcode> Element

Specifies the Virtual Machine (VM) command opcode. The opcode is 8 bytes long. Each byte is specified in as hexadecimal value [0xXX].

```
<opcode> [0xXX] [0xXX] [0xXX] [0xXX] [0xXX] [0xXX] [0xXX] [0xXX] </opcode>
```

#### Remarks

#### Example

The following example shows a DVD that has title menu, root menu and three titles. The title menu is a dummy one. It simply redirects to the root menu. The root menu supports playing each clip individually and returns back or playing all clips in sequence.

```

<?xml version="1.0" encoding="utf-8"?>
<dvd xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" version="2.1.2.1"
xmlns="http://www.primosoftware.com/dvdbuilder">
  <videoManager>
    <firstPlayCommands>
      <command>
        <!-- JumpSS VTSM (vts 1, tt 1, menu ROOT), 30 06 00 01 01 83 00 00 -->
        <opcode>[0x30][0x06][0x00][0x01][0x01][0x83][0x00][0x00]</opcode>
      </command>
    </firstPlayCommands>

    <menus>
      <menu id="1" entry="title">
        <preCommands>
          <command>
            <!-- JumpSS VTSM (vts 1, tt 1, menu ROOT), 30 06 00 01 01 83 00 00 -->
            <opcode>[0x30][0x06][0x00][0x01][0x01][0x83][0x00][0x00]</opcode>
          </command>
        </preCommands>

        <background file="menu\title_menu.mpg" />
      </menu>
    </menus>

  </videoManager>

  <!-- GPRM11 -> Play All Mode (True=1, False=0) -->

  <titleSet>
    <titles>
      <title id="100" chapters="00:00:00;">
        <videoObject file="../movies/3/columbiatristardvd_pal.mpg" />
      </title>
    </titles>
    <postCommands>
      <command>
        <!-- if (GPRM11 == 1) Goto 3, 00 a1 00 0b 00 01 00 03 -->
        <opcode>[0x00][0xa1][0x00][0x0b][0x00][0x01][0x00][0x03]</opcode>
      </command>
      <command>
        <!-- CallSS VTSM (menu ROOT, rsm_cell 1), 30 08 00 00 01 83 00 00 -->
        <opcode>[0x30][0x08][0x00][0x00][0x01][0x83][0x00][0x00]</opcode>
      </command>
      <command>
        <!-- JumpVTS_PTT (tt 2, ptt 1), 30 05 00 01 00 02 00 00 -->
        <opcode>[0x30][0x05][0x00][0x01][0x00][0x02][0x00][0x00]</opcode>
      </command>
    </postCommands>
  </titleSet>

```

```

    </command>
  </postCommands>

</title>

<title id="101" chapters="00:00:00;">
  <videoObject file="../movies/4/dreamworksanim_pal.mpg" />

  <postCommands>
    <command>
      <!-- if (GPRM11 == 1) Goto 3, 00 a1 00 0b 00 01 00 03 -->
      <opcode>[0x00][0xa1][0x00][0x0b][0x00][0x01][0x00][0x03]</opcode>
    </command>
    <command>
      <!-- CallSS VTSM (menu ROOT, rsm_cell 1), 30 08 00 00 01 83 00 00 -->
      <opcode>[0x30][0x08][0x00][0x00][0x01][0x83][0x00][0x00]</opcode>
    </command>

    <command>
      <!-- JumpVTS_PTT (tt 3, ptt 1), 30 05 00 01 00 03 00 00 -->
      <opcode>[0x30][0x05][0x00][0x01][0x00][0x03][0x00][0x00]</opcode>
    </command>
  </postCommands>
</title>

<title id="102" chapters="00:00:00;">
  <videoObject file="../movies/3/columbiatristardvd_pal.mpg" />

  <postCommands>
    <command>
      <!-- CallSS VTSM (menu ROOT, rsm_cell 1), 30 08 00 00 01 83 00 00 -->
      <opcode>[0x30][0x08][0x00][0x00][0x01][0x83][0x00][0x00]</opcode>
    </command>
  </postCommands>
</title>

</titles>
<menus>
  <menu id="2" entry="root">
    <preCommands>
      <command>
        <!-- GPRM11 = 0, 71 00 00 0b 00 00 00 00 -->
        <opcode>[0x71][0x00][0x00][0x0b][0x00][0x00][0x00][0x00]</opcode>
      </command>
    </preCommands>

    <postCommands>
      <command>
        <opcode>[0x20][0x04][0x0][0x0][0x0][0x0][0x0][0x01]</opcode>
        <!-- LinkPGCN 1, 20 04 00 00 00 00 01 -->
      </command>
    </postCommands>

    <!-- Play Title 1 button -->
    <button left="107" top="96" width="166" height="35">
      <command>
        <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 01 00 00 -->
        <opcode>[0x30][0x05][0x00][0x01][0x00][0x01][0x00][0x00]</opcode>
      </command>
    </button>

    <!-- Play Title 2 button -->
    <button left="107" top="155" width="170" height="37">
      <command>
        <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 02 00 00 -->
        <opcode>[0x30][0x05][0x00][0x01][0x00][0x02][0x00][0x00]</opcode>
      </command>
    </button>

    <!-- Play Title 3 button -->
    <button left="107" top="216" width="171" height="36">
      <command>

```

```

        <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 03 00 00 -->
        <opcode>[0x30][0x05][0x00][0x01][0x00][0x03][0x00][0x00]</opcode>
    </command>
</button>

<!-- Play All button -->
<button left="107" top="401" width="112" height="36">
    <command>
        <!-- LinkPGCN 2, 20 04 00 00 00 00 02 -->
        <opcode>[0x20][0x04][0x00][0x00][0x00][0x00][0x00][0x02]</opcode>
    </command>
</button>

    <mask file="menu\root_menu_mask.bmp" patternColor="#000000"
backgroundcolor="#FFFFFF" emphasisColor1="#000001" emphasisColor2="#000002" />
    <display emphasis2Contrast="0" emphasis1Contrast="0" backgroundContrast="0"
patternContrast="0" emphasis2Color="#000000" emphasis1Color="#000000"
patternColor="#000000" />
    <action emphasis2Contrast="0" emphasis1Contrast="0" backgroundContrast="0"
patternContrast="7" emphasis2Color="#000000" emphasis1Color="#000000"
backgroundColor="#000000" patternColor="#0000FF" />
    <selection emphasis2Contrast="0" emphasis1Contrast="0" backgroundContrast="0"
patternContrast="7" emphasis2Color="#000000" emphasis1Color="#000000"
backgroundColor="#000000" patternColor="#FF0000" />
    <background file="menu\root_menu.mpg" />
</menu>

<!-- Play All Dummy menu -->
<menu id="101">
    <preCommands>
        <command>
            <!-- GPRM11 = 1, 71 00 00 0b 00 01 00 00 -->
            <opcode>[0x71][0x00][0x00][0x0b][0x00][0x01][0x00][0x00]</opcode>
        </command>

        <command>
            <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 01 00 00 -->
            <opcode>[0x30][0x05][0x00][0x01][0x00][0x01][0x00][0x00]</opcode>
        </command>
    </preCommands>
</menu>
</menus>
</titleSet>
</dvd>

```

### 3.1.2.1.1.8 <preCommands> Element

Specifies the VM commands to be executed before a menu or a title is presented.

```

<preCommands>
  <command/>
</preCommands>

```

#### Child Elements

Element	Description
<command> (Version 2.1.2)	Specifies a VM command.

#### Example

The following example shows a DVD that has title menu, root menu and three titles. The title menu is a dummy one. It simply redirects to the root menu. The root menu supports playing each clip individually and returns back or playing all clips in sequence.

```

<?xml version="1.0" encoding="utf-8"?>
<dvd xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" version="2.1.2.1"

```

```

xmlns="http://www.primosoftware.com/dvdbuilder">
  <videoManager>
    <firstPlayCommands>
      <command>
        <!-- JumpSS VTSM (vts 1, tt 1, menu ROOT), 30 06 00 01 01 83 00 00 -->
        <opcode>[0x30][0x06][0x0][0x01][0x01][0x83][0x0][0x0]</opcode>
      </command>
    </firstPlayCommands>

    <menus>
      <menu id="1" entry="title">
        <preCommands>
          <command>
            <!-- JumpSS VTSM (vts 1, tt 1, menu ROOT), 30 06 00 01 01 83 00 00 -->
            <opcode>[0x30][0x06][0x0][0x01][0x01][0x83][0x0][0x0]</opcode>
          </command>
        </preCommands>

        <background file="menu\title_menu.mpg" />
      </menu>
    </menus>

  </videoManager>

  <!-- GPRM11 -> Play All Mode (True=1, False=0) -->

  <titleSet>
    <titles>
      <title id="100" chapters="00:00:00;">
        <videoObject file="../movies/3/columbiatristardvd_pal.mpg" />

        <postCommands>
          <command>
            <!-- if (GPRM11 == 1) Goto 3, 00 a1 00 0b 00 01 00 03 -->
            <opcode>[0x00][0xa1][0x00][0x0b][0x00][0x01][0x00][0x03]</opcode>
          </command>
          <command>
            <!-- CallSS VTSM (menu ROOT, rsm_cell 1), 30 08 00 00 01 83 00 00 -->
            <opcode>[0x30][0x08][0x00][0x00][0x01][0x83][0x00][0x00]</opcode>
          </command>

          <command>
            <!-- JumpVTS_PTT (tt 2, ptt 1), 30 05 00 01 00 02 00 00 -->
            <opcode>[0x30][0x05][0x00][0x01][0x00][0x02][0x00][0x00]</opcode>
          </command>
        </postCommands>
      </title>

      <title id="101" chapters="00:00:00;">
        <videoObject file="../movies/4/dreamworksanim_pal.mpg" />

        <postCommands>
          <command>
            <!-- if (GPRM11 == 1) Goto 3, 00 a1 00 0b 00 01 00 03 -->
            <opcode>[0x00][0xa1][0x00][0x0b][0x00][0x01][0x00][0x03]</opcode>
          </command>
          <command>
            <!-- CallSS VTSM (menu ROOT, rsm_cell 1), 30 08 00 00 01 83 00 00 -->
            <opcode>[0x30][0x08][0x00][0x00][0x01][0x83][0x00][0x00]</opcode>
          </command>

          <command>
            <!-- JumpVTS_PTT (tt 3, ptt 1), 30 05 00 01 00 03 00 00 -->
            <opcode>[0x30][0x05][0x00][0x01][0x00][0x03][0x00][0x00]</opcode>
          </command>
        </postCommands>
      </title>

      <title id="102" chapters="00:00:00;">
        <videoObject file="../movies/3/columbiatristardvd_pal.mpg" />

```

```

    <postCommands>
      <command>
        <!-- CallSS VTSM (menu ROOT, rsm_cell 1), 30 08 00 00 01 83 00 00 -->
        <opcode>[0x30][0x08][0x00][0x00][0x01][0x83][0x00][0x00]</opcode>
      </command>
    </postCommands>
  </title>

</titles>
<menus>
  <menu id="2" entry="root">
    <preCommands>
      <command>
        <!-- GPRM11 = 0, 71 00 00 0b 00 00 00 00 -->
        <opcode>[0x71][0x00][0x00][0x0b][0x00][0x00][0x00][0x00]</opcode>
      </command>
    </preCommands>

    <postCommands>
      <command>
        <opcode>[0x20][0x04][0x0][0x0][0x0][0x0][0x0][0x01]</opcode>
        <!-- LinkPGCN 1, 20 04 00 00 00 00 01 -->
      </command>
    </postCommands>

    <!-- Play Title 1 button -->
    <button left="107" top="96" width="166" height="35">
      <command>
        <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 01 00 00 -->
        <opcode>[0x30][0x05][0x00][0x01][0x00][0x01][0x00][0x00]</opcode>
      </command>
    </button>

    <!-- Play Title 2 button -->
    <button left="107" top="155" width="170" height="37">
      <command>
        <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 02 00 00 -->
        <opcode>[0x30][0x05][0x00][0x01][0x00][0x02][0x00][0x00]</opcode>
      </command>
    </button>

    <!-- Play Title 3 button -->
    <button left="107" top="216" width="171" height="36">
      <command>
        <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 03 00 00 -->
        <opcode>[0x30][0x05][0x00][0x01][0x00][0x03][0x00][0x00]</opcode>
      </command>
    </button>

    <!-- Play All button -->
    <button left="107" top="401" width="112" height="36">
      <command>
        <!-- LinkPGCN 2, 20 04 00 00 00 00 02 -->
        <opcode>[0x20][0x04][0x0][0x0][0x0][0x0][0x0][0x02]</opcode>
      </command>
    </button>

    <mask file="menu\root_menu_mask.bmp" patternColor="#000000"
backgroundColor="#FFFFFF" emphasisColor1="#000001" emphasisColor2="#000002" />
    <display emphasis2Contrast="0" emphasis1Contrast="0" backgroundContrast="0"
patternContrast="0" emphasis2Color="#000000" emphasis1Color="#000000"
patternColor="#000000" />
    <action emphasis2Contrast="0" emphasis1Contrast="0" backgroundContrast="0"
patternContrast="7" emphasis2Color="#000000" emphasis1Color="#000000"
backgroundColor="#000000" patternColor="#0000FF" />
    <selection emphasis2Contrast="0" emphasis1Contrast="0" backgroundContrast="0"
patternContrast="7" emphasis2Color="#000000" emphasis1Color="#000000"
backgroundColor="#000000" patternColor="#FF0000" />
    <background file="menu\root_menu.mpg" />
  </menu>

  <!-- Play All Dummy menu -->

```

```

    <menu id="101">
      <preCommands>
        <command>
          <!-- GPRM11 = 1, 71 00 00 0b 00 01 00 00 -->
          <opcode>[0x71][0x00][0x00][0x0b][0x00][0x01][0x00][0x00]</opcode>
        </command>

        <command>
          <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 01 00 00 -->
          <opcode>[0x30][0x05][0x00][0x01][0x00][0x01][0x00][0x00]</opcode>
        </command>
      </preCommands>
    </menu>
  </menus>
</titleSet>
</dvd>

```

### 3.1.2.1.1.8.1 <command> Element

#### Description

Defines a Virtual Machine (VM) command.

```

<command>
<opcode></opcode>
</command>

```

#### Example

The following example shows a DVD that has title menu, root menu and three titles. The title menu is a dummy one. It simply redirects to the root menu. The root menu supports playing each clip individually and returns back or playing all clips in sequence.

```

<?xml version="1.0" encoding="utf-8"?>
<dvd xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" version="2.1.2.1"
xmlns="http://www.primosoftware.com/dvdbuilder">
  <videoManager>
    <firstPlayCommands>
      <command>
        <!-- JumpSS VTSM (vts 1, tt 1, menu ROOT), 30 06 00 01 01 83 00 00 -->
        <opcode>[0x30][0x06][0x00][0x01][0x01][0x83][0x00][0x00]</opcode>
      </command>
    </firstPlayCommands>

    <menus>
      <menu id="1" entry="title">
        <preCommands>
          <command>
            <!-- JumpSS VTSM (vts 1, tt 1, menu ROOT), 30 06 00 01 01 83 00 00 -->
            <opcode>[0x30][0x06][0x00][0x01][0x01][0x83][0x00][0x00]</opcode>
          </command>
        </preCommands>

        <background file="menu\title_menu.mpg" />
      </menu>
    </menus>

  </videoManager>

  <!-- GPRM11 -> Play All Mode (True=1, False=0) -->

  <titleSet>
    <titles>
      <title id="100" chapters="00:00:00;">
        <videoObject file="../movies/3/columbiatristardvd_pal.mpg" />

        <postCommands>
          <command>

```

```

        <!-- if (GPRM11 == 1) Goto 3, 00 a1 00 0b 00 01 00 03 -->
        <opcode>[0x00][0xa1][0x00][0x0b][0x00][0x01][0x00][0x03]</opcode>
    </command>
</command>
    <!-- CallSS VTSM (menu ROOT, rsm_cell 1), 30 08 00 00 01 83 00 00 -->
    <opcode>[0x30][0x08][0x00][0x00][0x01][0x83][0x00][0x00]</opcode>
</command>

<command>
    <!-- JumpVTS_PTT (tt 2, ptt 1), 30 05 00 01 00 02 00 00 -->
    <opcode>[0x30][0x05][0x00][0x01][0x00][0x02][0x00][0x00]</opcode>
</command>
</postCommands>
</title>

<title id="101" chapters="00:00:00;">
    <videoObject file="../movies/4/dreamworksanim_pal.mpg" />

    <postCommands>
        <command>
            <!-- if (GPRM11 == 1) Goto 3, 00 a1 00 0b 00 01 00 03 -->
            <opcode>[0x00][0xa1][0x00][0x0b][0x00][0x01][0x00][0x03]</opcode>
        </command>
        <command>
            <!-- CallSS VTSM (menu ROOT, rsm_cell 1), 30 08 00 00 01 83 00 00 -->
            <opcode>[0x30][0x08][0x00][0x00][0x01][0x83][0x00][0x00]</opcode>
        </command>

        <command>
            <!-- JumpVTS_PTT (tt 3, ptt 1), 30 05 00 01 00 03 00 00 -->
            <opcode>[0x30][0x05][0x00][0x01][0x00][0x03][0x00][0x00]</opcode>
        </command>
    </postCommands>
</title>

<title id="102" chapters="00:00:00;">
    <videoObject file="../movies/3/columbiatristardvd_pal.mpg" />

    <postCommands>
        <command>
            <!-- CallSS VTSM (menu ROOT, rsm_cell 1), 30 08 00 00 01 83 00 00 -->
            <opcode>[0x30][0x08][0x00][0x00][0x01][0x83][0x00][0x00]</opcode>
        </command>
    </postCommands>
</title>

</titles>
<menus>
    <menu id="2" entry="root">
        <preCommands>
            <command>
                <!-- GPRM11 = 0, 71 00 00 0b 00 00 00 00 -->
                <opcode>[0x71][0x00][0x00][0x0b][0x00][0x00][0x00][0x00]</opcode>
            </command>
        </preCommands>

        <postCommands>
            <command>
                <opcode>[0x20][0x04][0x0][0x0][0x0][0x0][0x0][0x01]</opcode>
                <!-- LinkPGCN 1, 20 04 00 00 00 00 01 -->
            </command>
        </postCommands>

        <!-- Play Title 1 button -->
        <button left="107" top="96" width="166" height="35">
            <command>
                <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 01 00 00 -->
                <opcode>[0x30][0x05][0x00][0x01][0x00][0x01][0x00][0x00]</opcode>
            </command>
        </button>

```

```

<!-- Play Title 2 button -->
<button left="107" top="155" width="170" height="37">
  <command>
    <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 02 00 00 -->
    <opcode>[0x30][0x05][0x00][0x01][0x00][0x02][0x00][0x00]</opcode>
  </command>
</button>

<!-- Play Title 3 button -->
<button left="107" top="216" width="171" height="36">
  <command>
    <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 03 00 00 -->
    <opcode>[0x30][0x05][0x00][0x01][0x00][0x03][0x00][0x00]</opcode>
  </command>
</button>

<!-- Play All button -->
<button left="107" top="401" width="112" height="36">
  <command>
    <!-- LinkPGCN 2, 20 04 00 00 00 00 02 -->
    <opcode>[0x20][0x04][0x00][0x00][0x00][0x00][0x00][0x02]</opcode>
  </command>
</button>

  <mask file="menu\root_menu_mask.bmp" patternColor="#000000"
backgroundColor="#FFFFFF" emphasisColor1="#000001" emphasisColor2="#000002" />
  <display emphasis2Contrast="0" emphasis1Contrast="0" backgroundContrast="0"
patternContrast="0" emphasis2Color="#000000" emphasis1Color="#000000"
patternColor="#000000" />
  <action emphasis2Contrast="0" emphasis1Contrast="0" backgroundContrast="0"
patternContrast="7" emphasis2Color="#000000" emphasis1Color="#000000"
backgroundColor="#000000" patternColor="#0000FF" />
  <selection emphasis2Contrast="0" emphasis1Contrast="0" backgroundContrast="0"
patternContrast="7" emphasis2Color="#000000" emphasis1Color="#000000"
backgroundColor="#000000" patternColor="#FF0000" />
  <background file="menu\root_menu.mpg" />
</menu>

<!-- Play All Dummy menu -->
<menu id="101">
  <preCommands>
    <command>
      <!-- GPRM11 = 1, 71 00 00 0b 00 01 00 00 -->
      <opcode>[0x71][0x00][0x00][0x0b][0x00][0x01][0x00][0x00]</opcode>
    </command>

    <command>
      <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 01 00 00 -->
      <opcode>[0x30][0x05][0x00][0x01][0x00][0x01][0x00][0x00]</opcode>
    </command>
  </preCommands>
</menu>
</menus>
</titleSet>
</dvd>

```

#### 3.1.2.1.1.8.1.1 <opcode> Element

Specifies the Virtual Machine (VM) command opcode. The opcode is 8 bytes long. Each byte is specified in as hexadecimal value [0xXX].

```
<opcode>[0xXX][0xXX][0xXX][0xXX][0xXX][0xXX][0xXX][0xXX]</opcode>
```

#### Remarks

#### Example

The following example shows a DVD that has title menu, root menu and three titles. The title menu is a dummy one. It simply redirects to the root menu. The root menu supports playing each clip individually and returns back or playing all clips in sequence.

```
<?xml version="1.0" encoding="utf-8"?>
<dvd xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" version="2.1.2.1"
xmlns="http://www.primosoftware.com/dvdbuilder">
  <videoManager>
    <firstPlayCommands>
      <command>
        <!-- JumpSS VTSM (vts 1, tt 1, menu ROOT), 30 06 00 01 01 83 00 00 -->
        <opcode>[0x30][0x06][0x0][0x01][0x01][0x83][0x0][0x0]</opcode>
      </command>
    </firstPlayCommands>

    <menu>
      <menu id="1" entry="title">
        <preCommands>
          <command>
            <!-- JumpSS VTSM (vts 1, tt 1, menu ROOT), 30 06 00 01 01 83 00 00 -->
            <opcode>[0x30][0x06][0x0][0x01][0x01][0x83][0x0][0x0]</opcode>
          </command>
        </preCommands>

        <background file="menu\title_menu.mpg" />
      </menu>
    </menu>

  </videoManager>

  <!-- GPRM11 -> Play All Mode (True=1, False=0) -->

  <titleSet>
    <titles>
      <title id="100" chapters="00:00:00;">
        <videoObject file="../movies/3/columbiatristardvd_pal.mpg" />

        <postCommands>
          <command>
            <!-- if (GPRM11 == 1) Goto 3, 00 a1 00 0b 00 01 00 03 -->
            <opcode>[0x00][0xa1][0x00][0x0b][0x00][0x01][0x00][0x03]</opcode>
          </command>
          <command>
            <!-- CallSS VTSM (menu ROOT, rsm_cell 1), 30 08 00 00 01 83 00 00 -->
            <opcode>[0x30][0x08][0x00][0x00][0x01][0x83][0x00][0x00]</opcode>
          </command>

          <command>
            <!-- JumpVTS_PTT (tt 2, ptt 1), 30 05 00 01 00 02 00 00 -->
            <opcode>[0x30][0x05][0x00][0x01][0x00][0x02][0x00][0x00]</opcode>
          </command>
        </postCommands>
      </title>

      <title id="101" chapters="00:00:00;">
        <videoObject file="../movies/4/dreamworksanim_pal.mpg" />

        <postCommands>
          <command>
            <!-- if (GPRM11 == 1) Goto 3, 00 a1 00 0b 00 01 00 03 -->
            <opcode>[0x00][0xa1][0x00][0x0b][0x00][0x01][0x00][0x03]</opcode>
          </command>
          <command>
            <!-- CallSS VTSM (menu ROOT, rsm_cell 1), 30 08 00 00 01 83 00 00 -->
            <opcode>[0x30][0x08][0x00][0x00][0x01][0x83][0x00][0x00]</opcode>
          </command>

          <command>
            <!-- JumpVTS_PTT (tt 3, ptt 1), 30 05 00 01 00 03 00 00 -->
```

```

        <opcode>[0x30][0x05][0x00][0x01][0x00][0x03][0x00][0x00]</opcode>
    </command>
</postCommands>
</title>

<title id="102" chapters="00:00:00;">
    <videoObject file="../movies/3/columbiatristardvd_pal.mpg" />

    <postCommands>
        <command>
            <!-- CallSS VTSM (menu ROOT, rsm_cell 1), 30 08 00 00 01 83 00 00 -->
            <opcode>[0x30][0x08][0x00][0x00][0x01][0x83][0x00][0x00]</opcode>
        </command>
    </postCommands>
</title>

</titles>
<menus>
    <menu id="2" entry="root">
        <preCommands>
            <command>
                <!-- GPRM11 = 0, 71 00 00 0b 00 00 00 00 -->
                <opcode>[0x71][0x00][0x00][0x0b][0x00][0x00][0x00][0x00]</opcode>
            </command>
        </preCommands>

        <postCommands>
            <command>
                <opcode>[0x20][0x04][0x0][0x0][0x0][0x0][0x0][0x01]</opcode>
                <!-- LinkPGCN 1, 20 04 00 00 00 00 00 01 -->
            </command>
        </postCommands>

        <!-- Play Title 1 button -->
        <button left="107" top="96" width="166" height="35">
            <command>
                <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 01 00 00 -->
                <opcode>[0x30][0x05][0x00][0x01][0x00][0x01][0x00][0x00]</opcode>
            </command>
        </button>

        <!-- Play Title 2 button -->
        <button left="107" top="155" width="170" height="37">
            <command>
                <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 02 00 00 -->
                <opcode>[0x30][0x05][0x00][0x01][0x00][0x02][0x00][0x00]</opcode>
            </command>
        </button>

        <!-- Play Title 3 button -->
        <button left="107" top="216" width="171" height="36">
            <command>
                <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 03 00 00 -->
                <opcode>[0x30][0x05][0x00][0x01][0x00][0x03][0x00][0x00]</opcode>
            </command>
        </button>

        <!-- Play All button -->
        <button left="107" top="401" width="112" height="36">
            <command>
                <!-- LinkPGCN 2, 20 04 00 00 00 00 02 -->
                <opcode>[0x20][0x04][0x0][0x0][0x0][0x0][0x0][0x02]</opcode>
            </command>
        </button>

        <mask file="menu\root_menu_mask.bmp" patternColor="#000000"
backgroundColor="#FFFFFF" emphasisColor1="#000001" emphasisColor2="#000002" />
        <display emphasis2Contrast="0" emphasis1Contrast="0" backgroundContrast="0"
patternContrast="0" emphasis2Color="#000000" emphasis1Color="#000000"
patternColor="#000000" />
        <action emphasis2Contrast="0" emphasis1Contrast="0" backgroundContrast="0"
patternContrast="7" emphasis2Color="#000000" emphasis1Color="#000000"

```

```

backgroundColor="#000000" patternColor="#0000FF" />
  <selection emphasis2Contrast="0" emphasis1Contrast="0" backgroundContrast="0"
patternContrast="7" emphasis2Color="#000000" emphasis1Color="#000000"
backgroundColor="#000000" patternColor="#FF0000" />
  <background file="menu\root_menu.mpg" />
</menu>

<!-- Play All Dummy menu -->
<menu id="101">
  <preCommands>
    <command>
      <!-- GPRM11 = 1, 71 00 00 0b 00 01 00 00 -->
      <opcode>[0x71][0x00][0x00][0x0b][0x00][0x01][0x00][0x00]</opcode>
    </command>

    <command>
      <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 01 00 00 -->
      <opcode>[0x30][0x05][0x00][0x01][0x00][0x01][0x00][0x00]</opcode>
    </command>
  </preCommands>
</menu>
</menus>
</titleSet>
</dvd>

```

### 3.1.2.2 <titles> Element

#### Description

Defines Video Title Set (VTS) titles.

```

<titles>
  <title>
</titles>

```

#### Child Elements

Element	Description
<title>	Adds a VTS title.

#### Remarks

A group of related video titles having similar display format (standard screen or wide screen) or aspect ratio comprises a Video Title Set (VTS). A single VTS can hold up to 99 titles. Each title can hold up to 10 Video Objects (VOB).

#### 3.1.2.2.1 <title> Element

#### Description

Defines a Video Title Set (VTS) title.

```

<title
  id="num"
  postNavigate="Menu=id;|Title=id;|Title=id; Chapter=num;"
  chapters="00:00:00;|00:00:00;hr:min:sec;">
  <videoObject>
  <subpicturePalette>
  <preCommands>
  <postCommands>
</title>

```

#### Required Attributes

Attribute	Option	Description
id	1 to 99	Title unique identifier.
chapters	00:00:00	Specifies the playback times at which title chapters begin. The time is expressed in hr:min:sec string format.  Example: The value "00:00:00; 00:01:00;" creates two chapters. The first chapter begins at 00:00:00 playback time and the second chapter begins at 00:01:00  Each title should have at least one chapter. If there is only one chapter the value of this attribute should be set to "00:00:00"

### Optional Attributes

Attribute	Option	Description
postNavigate		Specifies the command that will be executed after the title playback.
	Menu=id;	Jumps to a menu. The menu can be any menu in the Video Manager(VMG) or a root menu from any of the Video Title Sets (VTS).
	Title=id;	Jumps to a specific title.
	Title=id; Chapter=num;	Navigates to Chapter with number "num" from Title with ID "id". Example: Title=1;Chapter=1;

### Child Elements

Element	Description
<videoObject>	Specifies video content for the title.
<subpicturePalette> (Version 1.10)	Specifies the palette that will be used for displaying sub-picture/subtitle streams. If you do not define the sub-picture palette all subtitles will be shown in white color.
<preCommands> (Version 2.1.2)	Specifies the VM commands to be executed before a title is presented.
<postCommands> (Version 2.1.2)	Specifies the VM commands to be executed after a title is presented.

### Remarks

A group of related video titles having similar display format (standard screen or wide screen) or aspect ratio comprises a Video Title Set (VTS). A single VTS can hold up to 99 titles. Each title can hold up to 10 Video Objects (VOB).

If <postCommands>child element is specified then the attribute "postNavigate" is not required. If both <postCommands>child element and attribute "postNavigate" are specified then the program will use only the <postCommands>child element.

**Example**

The following example shows a DVD that has one title with two chapters. When the last chapter is played the DVD player will pause for 10 seconds and will navigate to the Video Manager menu.

```
<?xml version="1.0" encoding="utf-8"?>
<dvd version="1.1.0.1" xmlns="http://www.primosoftware.com/dvdbuilder">
  <videoManager firstPlayNavigate="Menu=50001;">
    <menus>
      <menu id="50001" entry="title">
        <background file="menu_50001.mpg" />
        <mask
          file="menu_50001_mask.bmp"
          patternColor="#000000"
          backgroundColor="#FFFFFF"
          emphasisColor1="#0000FF"
          emphasisColor2="#00FF00" />

        <display
          patternColor   ="#FF0000" patternContrast   ="15"
          backgroundColor="#000000" backgroundContrast="0"
          emphasis1Color ="#000000" emphasis1Contrast ="0"
          emphasis2Color ="#000000" emphasis2Contrast ="0"/>

        <selection
          patternColor   ="#0000FF" patternContrast   ="15"
          backgroundColor="#000000" backgroundContrast="0"
          emphasis1Color ="#000000" emphasis1Contrast ="0"
          emphasis2Color ="#000000" emphasis2Contrast ="0"/>

        <action
          patternColor   ="#00FF00" patternContrast   ="15"
          backgroundColor="#000000" backgroundContrast="0"
          emphasis2Color ="#000000" emphasis2Contrast ="0"
          emphasis1Color ="#000000" emphasis1Contrast ="0"/>

        <button left="10" top="10" width="74" height="51" navigate="Title=1;" />
      </menu>
    </menus>
  </videoManager>
  <titleSet>
    <titles>
      <title id="1" postNavigate="Menu=50001;" chapters="00:00:00;00:01:00">
        <videoObject file="1.mpg" pause="10"/>
      </title>
    </titles>
  </titleSet>
</dvd>
```

The following example shows a DVD that has two title sets each with two titles in it. After the playback of each title the DVD Player will navigate to the next title. When the last title is played the DVD player will navigate to the first title and start the sequence again.

```
<?xml version="1.0" encoding="utf-8"?>
<dvd version="1.1.0.1" xmlns="http://www.primosoftware.com/dvdbuilder">
  <videoManager firstPlayNavigate="Title=1;" />
  <titleSet>
    <titles>
      <title id="1" postNavigate="Title=2;" chapters="00:00:00;">
        <videoObject file="1.mpg" />
      </title>

      <title id="2" postNavigate="Title=3;" chapters="00:00:00;">
        <videoObject file="2.mpg" />
      </title>
    </titles>
  </titleSet>
  <titleSet>
    <titles>
      <title id="3" postNavigate="Title=4;" chapters="00:00:00;">
        <videoObject file="3.mpg" />
      </title>
    </titles>
  </titleSet>
</dvd>
```

```

</title>
<title id="4" postNavigate="Title=1;" chapters="00:00:00;">
  <videoObject file="4.mpg" />
</title>
</titles>
</titleSet>
</dvd>

```

### 3.1.2.2.1.1 <videoObject> Element

Specifies video content for the title.

```

<videoObject
  file="file"
  pause="num"
  callback="true|false"
  aspectRatio="4:3|16:9"
  resolution="720x480|704x480|352x480|352x240|720x576|704x576|352x576|352x288"
  frameRate="29.97|25"/>

```

#### Required Attributes

Attribute	Description
file	Specifies a file that contains an MPEG-2 movie.

#### Optional Attributes

Attribute	Option	Description
pause	0 to 254	Specifies the number of seconds to pause after the Video Object playback.
callback (Version 1.1.0.1)	true	Indicates that the video object data should be read from a stream object supplied by the user. The value of the 'file' attribute will be passed to the callback object for reference.
	false	Indicates that the video object data should be read from a standard file specified with the 'file' attribute.
aspectRatio (Version 1.1.0.1)	4:3	Indicates that the video stream of this video object use Normal (4:3) aspect ratio.
	16:9	Indicates that the video stream of this video object use Wide Screen (16:9) aspect ratio.
resolution (Version 1.1.0.1)	<b>NTSC:</b> 720x480   704x480   352x480   352x240 <b>PAL :</b> 720x576   704x576   352x576   352x288	<b>NTSC</b> 720 x 480 pixels MPEG-2 704 x 480 pixels MPEG-2 352 x 480 pixels MPEG-2 352 x 240 pixels MPEG-1 or MPEG-2  <b>PAL</b> 720 x 576 pixels MPEG-2 704 x 576 pixels MPEG-2 352 x 576 pixels MPEG-2 352 x 288 pixels MPEG-1 or MPEG-2

frameRate (Version 1.1.0.1)	<b>NTSC:</b> 29.97 <b>PAL :</b> 25	<b>NTSC</b> MPEG video must be encoded at 29.97 frames per second.  <b>PAL</b> MPEG video must be encoded at 25 frames per second.
--------------------------------	---------------------------------------	--

### Child Elements

Element	Description
<audioStream> (Version 2.1.1.1)	Specifies an audio elementary stream.
<videoAstream> (Version 2.1.1.1)	Specifies a video elementary stream.
<subpictureStream> (Version 2.1.1.1)	Specifies a subpicture elementary stream.

### Remarks

DVDBuilder automatically detects the aspect ratio, the resolution and the frame rate of each video stream. If the aspect ratio, resolution or frame rate are specified the values in the project XML the values detected from the video stream will be ignored. The reason for this behavior is to allow input from encoders/multiplexers which set wrong values for aspect ratio, resolution and frame rate in the MPEG-2 streams.

### Example

The following example shows a DVD that has one title made of two video objects. When the first video object has been played the DVD player will pause the playback for 10 seconds and then will continue with the second video object.

```
<?xml version="1.0" encoding="utf-8"?>
<dvd version="1.1.0.1" xmlns="http://www.primosoftware.com/dvdbuilder">
  <videoManager firstPlayNavigate="Title=1;" />
  <titleSet>
    <titles>
      <title id="1" chapters="00:00:00;">
        <videoObject file="1.mpg" pause="10"/>
        <videoObject file="2.mpg" pause="0"/>
      </title>
    </titles>
  </titleSet>
</dvd>
```

The following example shows a DVD that has one title made of two video objects. Aspect rate, resolution and frame rate are explicitly specified for each video object.

```
<?xml version="1.0" encoding="utf-8"?>
<dvd version="1.1.0.1" xmlns="http://www.primosoftware.com/dvdbuilder">
  <videoManager firstPlayNavigate="Title=1;" />
  <titleSet>
    <titles>
      <title id="1" chapters="00:00:00;">
        <videoObject file="1.mpg" aspectRatio="4:3" resolution="720x576" frameRate="25"/>
        <videoObject file="2.mpg" aspectRatio="4:3" resolution="720x576" frameRate="25"/>
      </title>
    </titles>
  </titleSet>
</dvd>
```

The following example shows a DVD that has one title made of a video object using audio/video elementary streams. Note that the version attribute in the <dvd> tag is set to 2.1.1.1, because the <audioStream> and <videoStream> tags require DVDBuilder 2.1.1.1 or later.

```
<?xml version="1.0" encoding="utf-8"?>
<dvd version="2.1.1.1" xmlns="http://www.primosoftware.com/dvdbuilder">
  <videoManager firstPlayNavigate="Title=1;" />
  <titleSet>

    <audioStreams>
      <stream languageCode="EN" mpegStreamID="0xC0" mpegSubstreamID="0x00" />
    </audioStreams>

    <titles>
      <title id="1" chapters="00:00:00">

        <videoObject>
          <audioStream file="moviel.mpa" format="MPA" />
          <videoStream file="moviel.mpv" />
        </videoObject>

      </title>
    </titles>
  </titleSet>
</dvd>
```

#### 3.1.2.2.1.1.1 <audioStream> Element

Specifies an audio stream.

```
<audioStream
  file="file" format="format" />
```

##### Required Attributes

Attribute	Description
file	Specifies an audio elementary stream file.
format	Specifies the audio elementary stream format. It must be AC3, MPA or DTS.

##### Example

The following example shows a DVD with one movie. Note that the version attribute in the <dvd> tag is set to 2.1.1.1, because the <audioStream> and <videoStream> tags require DVDBuilder 2.1.1.1 or later.

```
<?xml version="1.0" encoding="utf-8"?>
<dvd version="2.1.1.1" xmlns="http://www.primosoftware.com/dvdbuilder">
  <videoManager firstPlayNavigate="Title=1;" />
  <titleSet>

    <audioStreams>
      <stream languageCode="EN" mpegStreamID="0xC0" mpegSubstreamID="0x00" />
    </audioStreams>

    <titles>
      <title id="1" chapters="00:00:00">

        <videoObject>
          <audioStream file="moviel.mpa" format="MPA" />
          <videoStream file="moviel.mpv" />
        </videoObject>

      </title>
    </titles>
  </titleSet>
</dvd>
```

## Summary

**3.1.2.2.1.1.2 <videoStream> Element**

Specifies a video stream.

```
<videoStream
  file="file" />
```

**Required Attributes**

Attribute	Description
file	Specifies a MPEG-2 video elementary stream file.

**Example**

The following example shows a DVD with one movie. Note that the version attribute in the <dvd> tag is set to 2.1.1.1, because the <audioStream> and <videoStream> tags require DVDBuilder 2.1.1.1 or later.

```
<?xml version="1.0" encoding="utf-8"?>
<dvd version="2.1.1.1" xmlns="http://www.primosoftware.com/dvdbuilder">
  <videoManager firstPlayNavigate="Title=1;" />
  <titleSet>

    <audioStreams>
      <stream languageCode="EN" mpegStreamID="0xC0" mpegSubstreamID="0x00" />
    </audioStreams>

    <titles>
      <title id="1" chapters="00:00:00">

        <videoObject>
          <audioStream file="movie1.mpa" format="MPA" />
          <videoStream file="movie1.mpv" />
        </videoObject>

      </title>
    </titles>
  </titleSet>
</dvd>
```

## Summary

**3.1.2.2.1.1.3 <subpictureStream> Element**

Specifies a subpicture stream.

```
<subpictureStream
  file="file" />
```

**Required Attributes**

Attribute	Description
file	Specifies a MPEG-2 PS file that contains a DVD-Video subpictures.

**Example**

The following example shows a DVD with one movie with subpicture. Note that the version attribute in the <dvd> tag is set to 2.1.1.1, because the <audioStream>, <videoStream> and <subpictureStream> tags require DVDBuilder 2.1.1.1 or later.

```
<?xml version="1.0" encoding="utf-8"?>
<dvd version="2.1.1.1" xmlns="http://www.primosoftware.com/dvdbuilder">
  <videoManager firstPlayNavigate="Title=1;" />
```

```

<titleSet>
  <audioStreams>
    <stream languageCode="EN" mpegStreamID="0xC0" mpegSubstreamID="0x00" />
  </audioStreams>

  <subpictureStreams>
    <stream languageCode="EN" mpegStreamID="0xBD" mpegSubstreamID="0x20" />
  </subpictureStreams>

  <titles>
    <title id="1" chapters="00:00:00">

      <subpicturePalette>
        <color index="1" value="#FF00FF" />
        <color index="2" value="#00FF00" />
        <color index="3" value="#FF0000" />
        <color index="4" value="#FFFF00" />
      </subpicturePalette>

      <videoObject>
        <audioStream file="moviel.mpa" format="MPA" />
        <videoStream file="moviel.mpv" />
        <subpictureStream file="moviel.sp" />
      </videoObject>

    </title>
  </titles>
</titleSet>
</dvd>

```

#### Summary

### 3.1.2.2.1.2 <postCommands> Element

Specifies the VM commands to be executed after a menu or a title is presented.

```

<preCommands>
  <command />
</preCommands>

```

#### Child Elements

Element	Description
<command> (Version 2.1.2)	Specifies a VM command.

#### Example

The following example shows a DVD that has title menu, root menu and three titles. The title menu is a dummy one. It simply redirects to the root menu. The root menu supports playing each clip individually and returns back or playing all clips in sequence.

```

<?xml version="1.0" encoding="utf-8"?>
<dvd xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" version="2.1.2.1"
  xmlns="http://www.primosoftware.com/dvdbuilder">
  <videoManager>
    <firstPlayCommands>
      <command>
        <!-- JumpSS VTSM (vts 1, tt 1, menu ROOT), 30 06 00 01 01 83 00 00 -->
        <opcode>[0x30][0x06][0x01][0x01][0x83][0x00][0x00]</opcode>
      </command>
    </firstPlayCommands>

    <menus>
      <menu id="1" entry="title">

```

```

    <preCommands>
      <command>
        <!-- JumpSS VTSM (vts 1, tt 1, menu ROOT), 30 06 00 01 01 83 00 00 -->
        <opcode>[0x30][0x06][0x0][0x01][0x01][0x83][0x0][0x0]</opcode>
      </command>
    </preCommands>

    <background file="menu\title_menu.mpg" />
  </menu>
</menus>

</videoManager>

<!-- GPRM11 -> Play All Mode (True=1, False=0) -->

<titleSet>
  <titles>
    <title id="100" chapters="00:00:00;">
      <videoObject file="../movies/3/columbiatristardvd_pal.mpg" />

      <postCommands>
        <command>
          <!-- if (GPRM11 == 1) Goto 3, 00 a1 00 0b 00 01 00 03 -->
          <opcode>[0x00][0xa1][0x00][0x0b][0x00][0x01][0x00][0x03]</opcode>
        </command>
        <command>
          <!-- CallSS VTSM (menu ROOT, rsm_cell 1), 30 08 00 00 01 83 00 00 -->
          <opcode>[0x30][0x08][0x00][0x00][0x01][0x83][0x00][0x00]</opcode>
        </command>

        <command>
          <!-- JumpVTS_PTT (tt 2, ptt 1), 30 05 00 01 00 02 00 00 -->
          <opcode>[0x30][0x05][0x00][0x01][0x00][0x02][0x00][0x00]</opcode>
        </command>
      </postCommands>
    </title>

    <title id="101" chapters="00:00:00;">
      <videoObject file="../movies/4/dreamworksanim_pal.mpg" />

      <postCommands>
        <command>
          <!-- if (GPRM11 == 1) Goto 3, 00 a1 00 0b 00 01 00 03 -->
          <opcode>[0x00][0xa1][0x00][0x0b][0x00][0x01][0x00][0x03]</opcode>
        </command>
        <command>
          <!-- CallSS VTSM (menu ROOT, rsm_cell 1), 30 08 00 00 01 83 00 00 -->
          <opcode>[0x30][0x08][0x00][0x00][0x01][0x83][0x00][0x00]</opcode>
        </command>

        <command>
          <!-- JumpVTS_PTT (tt 3, ptt 1), 30 05 00 01 00 03 00 00 -->
          <opcode>[0x30][0x05][0x00][0x01][0x00][0x03][0x00][0x00]</opcode>
        </command>
      </postCommands>
    </title>

    <title id="102" chapters="00:00:00;">
      <videoObject file="../movies/3/columbiatristardvd_pal.mpg" />

      <postCommands>
        <command>
          <!-- CallSS VTSM (menu ROOT, rsm_cell 1), 30 08 00 00 01 83 00 00 -->
          <opcode>[0x30][0x08][0x00][0x00][0x01][0x83][0x00][0x00]</opcode>
        </command>
      </postCommands>
    </title>
  </titles>
</menuSet>
  <menu id="2" entry="root">

```

```

<preCommands>
  <command>
    <!-- GPRM11 = 0, 71 00 00 0b 00 00 00 00 -->
    <opcode>[0x71][0x00][0x00][0x0b][0x00][0x00][0x00][0x00]</opcode>
  </command>
</preCommands>

<postCommands>
  <command>
    <opcode>[0x20][0x04][0x0][0x0][0x0][0x0][0x0][0x01]</opcode>
    <!-- LinkPGCN 1, 20 04 00 00 00 00 01 -->
  </command>
</postCommands>

<!-- Play Title 1 button -->
<button left="107" top="96" width="166" height="35">
  <command>
    <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 01 00 00 -->
    <opcode>[0x30][0x05][0x00][0x01][0x00][0x01][0x00][0x00]</opcode>
  </command>
</button>

<!-- Play Title 2 button -->
<button left="107" top="155" width="170" height="37">
  <command>
    <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 02 00 00 -->
    <opcode>[0x30][0x05][0x00][0x01][0x00][0x02][0x00][0x00]</opcode>
  </command>
</button>

<!-- Play Title 3 button -->
<button left="107" top="216" width="171" height="36">
  <command>
    <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 03 00 00 -->
    <opcode>[0x30][0x05][0x00][0x01][0x00][0x03][0x00][0x00]</opcode>
  </command>
</button>

<!-- Play All button -->
<button left="107" top="401" width="112" height="36">
  <command>
    <!-- LinkPGCN 2, 20 04 00 00 00 00 02 -->
    <opcode>[0x20][0x04][0x0][0x0][0x0][0x0][0x0][0x02]</opcode>
  </command>
</button>

  <mask file="menu\root_menu_mask.bmp" patternColor="#000000"
backgroundcolor="#FFFFFF" emphasisColor1="#000001" emphasisColor2="#000002" />
  <display emphasis2Contrast="0" emphasis1Contrast="0" backgroundContrast="0"
patternContrast="0" emphasis2Color="#000000" emphasis1Color="#000000"
patternColor="#000000" />
  <action emphasis2Contrast="0" emphasis1Contrast="0" backgroundContrast="0"
patternContrast="7" emphasis2Color="#000000" emphasis1Color="#000000"
backgroundcolor="#000000" patternColor="#0000FF" />
  <selection emphasis2Contrast="0" emphasis1Contrast="0" backgroundContrast="0"
patternContrast="7" emphasis2Color="#000000" emphasis1Color="#000000"
backgroundcolor="#000000" patternColor="#FF0000" />
  <background file="menu\root_menu.mpg" />
</menu>

<!-- Play All Dummy menu -->
<menu id="101">
  <preCommands>
    <command>
      <!-- GPRM11 = 1, 71 00 00 0b 00 01 00 00 -->
      <opcode>[0x71][0x00][0x00][0x0b][0x00][0x01][0x00][0x00]</opcode>
    </command>

    <command>
      <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 01 00 00 -->
      <opcode>[0x30][0x05][0x00][0x01][0x00][0x01][0x00][0x00]</opcode>
    </command>

```

```

    </preCommands>
  </menu>
</menus>
</titleSet>
</dvd>

```

### 3.1.2.2.1.2.1 <command> Element

#### Description

Defines a Virtual Machine (VM) command.

```

<command>
<opcode></opcode>
</command>

```

#### Example

The following example shows a DVD that has title menu, root menu and three titles. The title menu is a dummy one. It simply redirects to the root menu. The root menu supports playing each clip individually and returns back or playing all clips in sequence.

```

<?xml version="1.0" encoding="utf-8"?>
<dvd xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" version="2.1.2.1"
xmlns="http://www.primosoftware.com/dvdbuilder">
  <videoManager>
    <firstPlayCommands>
      <command>
        <!-- JumpSS VTSM (vts 1, tt 1, menu ROOT), 30 06 00 01 01 83 00 00 -->
        <opcode>[0x30][0x06][0x00][0x01][0x01][0x83][0x00][0x00]</opcode>
      </command>
    </firstPlayCommands>

    <menu>
      <menu id="1" entry="title">
        <preCommands>
          <command>
            <!-- JumpSS VTSM (vts 1, tt 1, menu ROOT), 30 06 00 01 01 83 00 00 -->
            <opcode>[0x30][0x06][0x00][0x01][0x01][0x83][0x00][0x00]</opcode>
          </command>
        </preCommands>

        <background file="menu\title_menu.mpg" />
      </menu>
    </menu>

  </videoManager>

  <!-- GPRM11 -> Play All Mode (True=1, False=0) -->

  <titleSet>
    <titles>
      <title id="100" chapters="00:00:00;">
        <videoObject file="../movies/3/columbiatristardvd_pal.mpg" />

        <postCommands>
          <command>
            <!-- if (GPRM11 == 1) Goto 3, 00 a1 00 0b 00 01 00 03 -->
            <opcode>[0x00][0xa1][0x00][0x0b][0x00][0x01][0x00][0x03]</opcode>
          </command>
          <command>
            <!-- CallSS VTSM (menu ROOT, rsm_cell 1), 30 08 00 00 01 83 00 00 -->
            <opcode>[0x30][0x08][0x00][0x00][0x01][0x83][0x00][0x00]</opcode>
          </command>

          <command>
            <!-- JumpVTS_PTT (tt 2, ptt 1), 30 05 00 01 00 02 00 00 -->
            <opcode>[0x30][0x05][0x00][0x01][0x00][0x02][0x00][0x00]</opcode>

```

```

    </command>
  </postCommands>

</title>

<title id="101" chapters="00:00:00;">
  <videoObject file="../movies/4/dreamworksanim_pal.mpg" />

  <postCommands>
    <command>
      <!-- if (GPRM11 == 1) Goto 3, 00 a1 00 0b 00 01 00 03 -->
      <opcode>[0x00][0xa1][0x00][0x0b][0x00][0x01][0x00][0x03]</opcode>
    </command>
    <command>
      <!-- CallSS VTSM (menu ROOT, rsm_cell 1), 30 08 00 00 01 83 00 00 -->
      <opcode>[0x30][0x08][0x00][0x00][0x01][0x83][0x00][0x00]</opcode>
    </command>

    <command>
      <!-- JumpVTS_PTT (tt 3, ptt 1), 30 05 00 01 00 03 00 00 -->
      <opcode>[0x30][0x05][0x00][0x01][0x00][0x03][0x00][0x00]</opcode>
    </command>
  </postCommands>
</title>

<title id="102" chapters="00:00:00;">
  <videoObject file="../movies/3/columbiatristardvd_pal.mpg" />

  <postCommands>
    <command>
      <!-- CallSS VTSM (menu ROOT, rsm_cell 1), 30 08 00 00 01 83 00 00 -->
      <opcode>[0x30][0x08][0x00][0x00][0x01][0x83][0x00][0x00]</opcode>
    </command>
  </postCommands>
</title>

</titles>
<menus>
  <menu id="2" entry="root">
    <preCommands>
      <command>
        <!-- GPRM11 = 0, 71 00 00 0b 00 00 00 00 -->
        <opcode>[0x71][0x00][0x00][0x0b][0x00][0x00][0x00][0x00]</opcode>
      </command>
    </preCommands>

    <postCommands>
      <command>
        <opcode>[0x20][0x04][0x0][0x0][0x0][0x0][0x0][0x01]</opcode>
        <!-- LinkPGCN 1, 20 04 00 00 00 00 01 -->
      </command>
    </postCommands>

    <!-- Play Title 1 button -->
    <button left="107" top="96" width="166" height="35">
      <command>
        <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 01 00 00 -->
        <opcode>[0x30][0x05][0x00][0x01][0x00][0x01][0x00][0x00]</opcode>
      </command>
    </button>

    <!-- Play Title 2 button -->
    <button left="107" top="155" width="170" height="37">
      <command>
        <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 02 00 00 -->
        <opcode>[0x30][0x05][0x00][0x01][0x00][0x02][0x00][0x00]</opcode>
      </command>
    </button>

    <!-- Play Title 3 button -->
    <button left="107" top="216" width="171" height="36">
      <command>

```

```

        <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 03 00 00 -->
        <opcode>[0x30][0x05][0x00][0x01][0x00][0x03][0x00][0x00]</opcode>
    </command>
</button>

<!-- Play All button -->
<button left="107" top="401" width="112" height="36">
    <command>
        <!-- LinkPGCN 2, 20 04 00 00 00 00 02 -->
        <opcode>[0x20][0x04][0x00][0x00][0x00][0x00][0x02]</opcode>
    </command>
</button>

    <mask file="menu\root_menu_mask.bmp" patternColor="#000000"
backgroundcolor="#FFFFFF" emphasisColor1="#000001" emphasisColor2="#000002" />
    <display emphasis2Contrast="0" emphasis1Contrast="0" backgroundContrast="0"
patternContrast="0" emphasis2Color="#000000" emphasis1Color="#000000"
patternColor="#000000" />
    <action emphasis2Contrast="0" emphasis1Contrast="0" backgroundContrast="0"
patternContrast="7" emphasis2Color="#000000" emphasis1Color="#000000"
backgroundcolor="#000000" patternColor="#0000FF" />
    <selection emphasis2Contrast="0" emphasis1Contrast="0" backgroundContrast="0"
patternContrast="7" emphasis2Color="#000000" emphasis1Color="#000000"
backgroundcolor="#000000" patternColor="#FF0000" />
    <background file="menu\root_menu.mpg" />
</menu>

<!-- Play All Dummy menu -->
<menu id="101">
    <preCommands>
        <command>
            <!-- GPRM11 = 1, 71 00 00 0b 00 01 00 00 -->
            <opcode>[0x71][0x00][0x00][0x0b][0x00][0x01][0x00][0x00]</opcode>
        </command>

        <command>
            <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 01 00 00 -->
            <opcode>[0x30][0x05][0x00][0x01][0x00][0x01][0x00][0x00]</opcode>
        </command>
    </preCommands>
</menu>
</menus>
</titleSet>
</dvd>

```

### 3.1.2.2.1.2.1.1 <opcode> Element

Specifies the Virtual Machine (VM) command opcode. The opcode is 8 bytes long. Each byte is specified in as hexadecimal value [0xXX].

```
<opcode>[0xXX][0xXX][0xXX][0xXX][0xXX][0xXX][0xXX][0xXX]</opcode>
```

#### Remarks

#### Example

The following example shows a DVD that has title menu, root menu and three titles. The title menu is a dummy one. It simply redirects to the root menu. The root menu supports playing each clip individually and returns back or playing all clips in sequence.

```

<?xml version="1.0" encoding="utf-8"?>
<dvd xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" version="2.1.2.1"
xmlns="http://www.primosoftware.com/dvdbuilder">
    <videoManager>
        <firstPlayCommands>
            <command>

```

```

    <!-- JumpSS VTSM (vts 1, tt 1, menu ROOT), 30 06 00 01 01 83 00 00 -->
    <opcode>[0x30][0x06][0x00][0x01][0x01][0x83][0x00][0x00]</opcode>
  </command>
</firstPlayCommands>

<menus>
  <menu id="1" entry="title">
    <preCommands>
      <command>
        <!-- JumpSS VTSM (vts 1, tt 1, menu ROOT), 30 06 00 01 01 83 00 00 -->
        <opcode>[0x30][0x06][0x00][0x01][0x01][0x83][0x00][0x00]</opcode>
      </command>
    </preCommands>

    <background file="menu\title_menu.mpg" />
  </menu>
</menus>

</videoManager>

<!-- GPRM11 -> Play All Mode (True=1, False=0) -->

<titleSet>
  <titles>
    <title id="100" chapters="00:00:00;">
      <videoObject file="../movies/3/columbiatristardvd_pal.mpg" />

      <postCommands>
        <command>
          <!-- if (GPRM11 == 1) Goto 3, 00 a1 00 0b 00 01 00 03 -->
          <opcode>[0x00][0xa1][0x00][0x0b][0x00][0x01][0x00][0x03]</opcode>
        </command>
        <command>
          <!-- CallSS VTSM (menu ROOT, rsm_cell 1), 30 08 00 00 01 83 00 00 -->
          <opcode>[0x30][0x08][0x00][0x00][0x01][0x83][0x00][0x00]</opcode>
        </command>

        <command>
          <!-- JumpVTS_PTT (tt 2, ptt 1), 30 05 00 01 00 02 00 00 -->
          <opcode>[0x30][0x05][0x00][0x01][0x00][0x02][0x00][0x00]</opcode>
        </command>
      </postCommands>
    </title>

    <title id="101" chapters="00:00:00;">
      <videoObject file="../movies/4/dreamworksanim_pal.mpg" />

      <postCommands>
        <command>
          <!-- if (GPRM11 == 1) Goto 3, 00 a1 00 0b 00 01 00 03 -->
          <opcode>[0x00][0xa1][0x00][0x0b][0x00][0x01][0x00][0x03]</opcode>
        </command>
        <command>
          <!-- CallSS VTSM (menu ROOT, rsm_cell 1), 30 08 00 00 01 83 00 00 -->
          <opcode>[0x30][0x08][0x00][0x00][0x01][0x83][0x00][0x00]</opcode>
        </command>

        <command>
          <!-- JumpVTS_PTT (tt 3, ptt 1), 30 05 00 01 00 03 00 00 -->
          <opcode>[0x30][0x05][0x00][0x01][0x00][0x03][0x00][0x00]</opcode>
        </command>
      </postCommands>
    </title>

    <title id="102" chapters="00:00:00;">
      <videoObject file="../movies/3/columbiatristardvd_pal.mpg" />

      <postCommands>
        <command>
          <!-- CallSS VTSM (menu ROOT, rsm_cell 1), 30 08 00 00 01 83 00 00 -->
          <opcode>[0x30][0x08][0x00][0x00][0x01][0x83][0x00][0x00]</opcode>

```

```

    </command>
  </postCommands>
</title>

</titles>
<menu>
  <menu id="2" entry="root">
    <preCommands>
      <command>
        <!-- GPRM11 = 0, 71 00 00 0b 00 00 00 00 -->
        <opcode>[0x71][0x00][0x00][0x0b][0x00][0x00][0x00][0x00]</opcode>
      </command>
    </preCommands>

    <postCommands>
      <command>
        <opcode>[0x20][0x04][0x0][0x0][0x0][0x0][0x0][0x01]</opcode>
        <!-- LinkPGCN 1, 20 04 00 00 00 00 01 -->
      </command>
    </postCommands>

    <!-- Play Title 1 button -->
    <button left="107" top="96" width="166" height="35">
      <command>
        <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 01 00 00 -->
        <opcode>[0x30][0x05][0x00][0x01][0x00][0x01][0x00][0x00]</opcode>
      </command>
    </button>

    <!-- Play Title 2 button -->
    <button left="107" top="155" width="170" height="37">
      <command>
        <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 02 00 00 -->
        <opcode>[0x30][0x05][0x00][0x01][0x00][0x02][0x00][0x00]</opcode>
      </command>
    </button>

    <!-- Play Title 3 button -->
    <button left="107" top="216" width="171" height="36">
      <command>
        <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 03 00 00 -->
        <opcode>[0x30][0x05][0x00][0x01][0x00][0x03][0x00][0x00]</opcode>
      </command>
    </button>

    <!-- Play All button -->
    <button left="107" top="401" width="112" height="36">
      <command>
        <!-- LinkPGCN 2, 20 04 00 00 00 00 02 -->
        <opcode>[0x20][0x04][0x0][0x0][0x0][0x0][0x0][0x02]</opcode>
      </command>
    </button>

    <mask file="menu\root_menu_mask.bmp" patternColor="#000000"
backgroundColor="#FFFFFF" emphasisColor1="#000001" emphasisColor2="#000002" />
    <display emphasis2Contrast="0" emphasis1Contrast="0" backgroundContrast="0"
patternContrast="0" emphasis2Color="#000000" emphasis1Color="#000000"
patternColor="#000000" />
    <action emphasis2Contrast="0" emphasis1Contrast="0" backgroundContrast="0"
patternContrast="7" emphasis2Color="#000000" emphasis1Color="#000000"
backgroundColor="#000000" patternColor="#0000FF" />
    <selection emphasis2Contrast="0" emphasis1Contrast="0" backgroundContrast="0"
patternContrast="7" emphasis2Color="#000000" emphasis1Color="#000000"
backgroundColor="#000000" patternColor="#FF0000" />
    <background file="menu\root_menu.mpg" />
  </menu>

  <!-- Play All Dummy menu -->
  <menu id="101">
    <preCommands>
      <command>
        <!-- GPRM11 = 1, 71 00 00 0b 00 01 00 00 -->

```

```

    <opcode>[0x71][0x00][0x00][0x0b][0x00][0x01][0x00][0x00]</opcode>
  </command>

  <command>
    <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 01 00 00 -->
    <opcode>[0x30][0x05][0x00][0x01][0x00][0x01][0x00][0x00]</opcode>
  </command>
</preCommands>
</menu>
</menus>
</titleSet>
</dvd>

```

### 3.1.2.2.1.3 <preCommands> Element

Specifies the VM commands to be executed before a menu or a title is presented.

```

<preCommands>
  <command/>
</preCommands>

```

#### Child Elements

Element	Description
<command> (Version 2.1.2)	Specifies a VM command.

#### Example

The following example shows a DVD that has title menu, root menu and three titles. The title menu is a dummy one. It simply redirects to the root menu. The root menu supports playing each clip individually and returns back or playing all clips in sequence.

```

<?xml version="1.0" encoding="utf-8"?>
<dvd xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" version="2.1.2.1"
xmlns="http://www.primosoftware.com/dvdbuilder">
  <videoManager>
    <firstPlayCommands>
      <command>
        <!-- JumpSS VTSM (vts 1, tt 1, menu ROOT), 30 06 00 01 01 83 00 00 -->
        <opcode>[0x30][0x06][0x0][0x01][0x01][0x83][0x0][0x0]</opcode>
      </command>
    </firstPlayCommands>

    <menus>
      <menu id="1" entry="title">
        <preCommands>
          <command>
            <!-- JumpSS VTSM (vts 1, tt 1, menu ROOT), 30 06 00 01 01 83 00 00 -->
            <opcode>[0x30][0x06][0x0][0x01][0x01][0x83][0x0][0x0]</opcode>
          </command>
        </preCommands>

        <background file="menu\title_menu.mpg" />
      </menu>
    </menus>

  </videoManager>

  <!-- GPRM11 -> Play All Mode (True=1, False=0) -->

  <titleSet>
    <titles>
      <title id="100" chapters="00:00:00;">
        <videoObject file="../movies/3/columbiatristardvd_pal.mpg" />

      <postCommands>

```

```

    <command>
      <!-- if (GPRM11 == 1) Goto 3, 00 a1 00 0b 00 01 00 03 -->
      <opcode>[0x00][0xa1][0x00][0x0b][0x00][0x01][0x00][0x03]</opcode>
    </command>
    <command>
      <!-- CallSS VTSM (menu ROOT, rsm_cell 1), 30 08 00 00 01 83 00 00 -->
      <opcode>[0x30][0x08][0x00][0x00][0x01][0x83][0x00][0x00]</opcode>
    </command>

    <command>
      <!-- JumpVTS_PTT (tt 2, ptt 1), 30 05 00 01 00 02 00 00 -->
      <opcode>[0x30][0x05][0x00][0x01][0x00][0x02][0x00][0x00]</opcode>
    </command>
  </postCommands>

</title>

<title id="101" chapters="00:00:00;">
  <videoObject file="../movies/4/dreamworksanim_pal.mpg" />

  <postCommands>
    <command>
      <!-- if (GPRM11 == 1) Goto 3, 00 a1 00 0b 00 01 00 03 -->
      <opcode>[0x00][0xa1][0x00][0x0b][0x00][0x01][0x00][0x03]</opcode>
    </command>
    <command>
      <!-- CallSS VTSM (menu ROOT, rsm_cell 1), 30 08 00 00 01 83 00 00 -->
      <opcode>[0x30][0x08][0x00][0x00][0x01][0x83][0x00][0x00]</opcode>
    </command>

    <command>
      <!-- JumpVTS_PTT (tt 3, ptt 1), 30 05 00 01 00 03 00 00 -->
      <opcode>[0x30][0x05][0x00][0x01][0x00][0x03][0x00][0x00]</opcode>
    </command>
  </postCommands>
</title>

<title id="102" chapters="00:00:00;">
  <videoObject file="../movies/3/columbiatristardvd_pal.mpg" />

  <postCommands>
    <command>
      <!-- CallSS VTSM (menu ROOT, rsm_cell 1), 30 08 00 00 01 83 00 00 -->
      <opcode>[0x30][0x08][0x00][0x00][0x01][0x83][0x00][0x00]</opcode>
    </command>
  </postCommands>
</title>

</titles>
<menus>
  <menu id="2" entry="root">
    <preCommands>
      <command>
        <!-- GPRM11 = 0, 71 00 00 0b 00 00 00 00 -->
        <opcode>[0x71][0x00][0x00][0x0b][0x00][0x00][0x00][0x00]</opcode>
      </command>
    </preCommands>

    <postCommands>
      <command>
        <opcode>[0x20][0x04][0x0][0x0][0x0][0x0][0x0][0x01]</opcode>
        <!-- LinkPGCN 1, 20 04 00 00 00 00 01 -->
      </command>
    </postCommands>

    <!-- Play Title 1 button -->
    <button left="107" top="96" width="166" height="35">
      <command>
        <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 01 00 00 -->
        <opcode>[0x30][0x05][0x00][0x01][0x00][0x01][0x00][0x00]</opcode>
      </command>
    </button>

```

```

<!-- Play Title 2 button -->
<button left="107" top="155" width="170" height="37">
  <command>
    <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 02 00 00 -->
    <opcode>[0x30][0x05][0x00][0x01][0x00][0x02][0x00][0x00]</opcode>
  </command>
</button>

<!-- Play Title 3 button -->
<button left="107" top="216" width="171" height="36">
  <command>
    <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 03 00 00 -->
    <opcode>[0x30][0x05][0x00][0x01][0x00][0x03][0x00][0x00]</opcode>
  </command>
</button>

<!-- Play All button -->
<button left="107" top="401" width="112" height="36">
  <command>
    <!-- LinkPGCN 2, 20 04 00 00 00 00 02 -->
    <opcode>[0x20][0x04][0x0][0x0][0x0][0x0][0x0][0x02]</opcode>
  </command>
</button>

  <mask file="menu\root_menu_mask.bmp" patternColor="#000000"
backgroundColor="#FFFFFF" emphasisColor1="#000001" emphasisColor2="#000002" />
  <display emphasis2Contrast="0" emphasis1Contrast="0" backgroundContrast="0"
patternContrast="0" emphasis2Color="#000000" emphasis1Color="#000000"
patternColor="#000000" />
  <action emphasis2Contrast="0" emphasis1Contrast="0" backgroundContrast="0"
patternContrast="7" emphasis2Color="#000000" emphasis1Color="#000000"
backgroundColor="#000000" patternColor="#0000FF" />
  <selection emphasis2Contrast="0" emphasis1Contrast="0" backgroundContrast="0"
patternContrast="7" emphasis2Color="#000000" emphasis1Color="#000000"
backgroundColor="#000000" patternColor="#FF0000" />
  <background file="menu\root_menu.mpg" />
</menu>

<!-- Play All Dummy menu -->
<menu id="101">
  <preCommands>
    <command>
      <!-- GPRM11 = 1, 71 00 00 0b 00 01 00 00 -->
      <opcode>[0x71][0x00][0x00][0x0b][0x00][0x01][0x00][0x00]</opcode>
    </command>

    <command>
      <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 01 00 00 -->
      <opcode>[0x30][0x05][0x00][0x01][0x00][0x01][0x00][0x00]</opcode>
    </command>
  </preCommands>
</menu>
</menus>
</titleSet>
</dvd>

```

### 3.1.2.1.3.1 < command > Element

#### Description

Defines a Virtual Machine (VM) command.

```

<command>
<opcode></opcode>
</command>

```

#### Example

The following example shows a DVD that has title menu, root menu and three titles. The title menu is a dummy one. It

simply redirects to the root menu. The root menu supports playing each clip individually and returns back or playing all clips in sequence.

```
<?xml version="1.0" encoding="utf-8"?>
<dvd xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" version="2.1.2.1"
xmlns="http://www.primosoftware.com/dvdbuilder">
  <videoManager>
    <firstPlayCommands>
      <command>
        <!-- JumpSS VTSM (vts 1, tt 1, menu ROOT), 30 06 00 01 01 83 00 00 -->
        <opcode>[0x30][0x06][0x0][0x01][0x01][0x83][0x0][0x0]</opcode>
      </command>
    </firstPlayCommands>

    <menus>
      <menu id="1" entry="title">
        <preCommands>
          <command>
            <!-- JumpSS VTSM (vts 1, tt 1, menu ROOT), 30 06 00 01 01 83 00 00 -->
            <opcode>[0x30][0x06][0x0][0x01][0x01][0x83][0x0][0x0]</opcode>
          </command>
        </preCommands>

        <background file="menu\title_menu.mpg" />
      </menu>
    </menus>

  </videoManager>

  <!-- GPRM11 -> Play All Mode (True=1, False=0) -->

  <titleSet>
    <titles>
      <title id="100" chapters="00:00:00;">
        <videoObject file="../movies/3/columbiatristardvd_pal.mpg" />

        <postCommands>
          <command>
            <!-- if (GPRM11 == 1) Goto 3, 00 a1 00 0b 00 01 00 03 -->
            <opcode>[0x00][0xa1][0x00][0x0b][0x00][0x01][0x00][0x03]</opcode>
          </command>
          <command>
            <!-- CallSS VTSM (menu ROOT, rsm_cell 1), 30 08 00 00 01 83 00 00 -->
            <opcode>[0x30][0x08][0x00][0x00][0x01][0x83][0x00][0x00]</opcode>
          </command>

          <command>
            <!-- JumpVTS_PTT (tt 2, ptt 1), 30 05 00 01 00 02 00 00 -->
            <opcode>[0x30][0x05][0x00][0x01][0x00][0x02][0x00][0x00]</opcode>
          </command>
        </postCommands>

      </title>

      <title id="101" chapters="00:00:00;">
        <videoObject file="../movies/4/dreamworksanim_pal.mpg" />

        <postCommands>
          <command>
            <!-- if (GPRM11 == 1) Goto 3, 00 a1 00 0b 00 01 00 03 -->
            <opcode>[0x00][0xa1][0x00][0x0b][0x00][0x01][0x00][0x03]</opcode>
          </command>
          <command>
            <!-- CallSS VTSM (menu ROOT, rsm_cell 1), 30 08 00 00 01 83 00 00 -->
            <opcode>[0x30][0x08][0x00][0x00][0x01][0x83][0x00][0x00]</opcode>
          </command>

          <command>
            <!-- JumpVTS_PTT (tt 3, ptt 1), 30 05 00 01 00 03 00 00 -->
            <opcode>[0x30][0x05][0x00][0x01][0x00][0x03][0x00][0x00]</opcode>
          </command>
        </postCommands>
      </title>
    </titles>
  </titleSet>
</dvd>
```

```

        </command>
    </postCommands>
</title>

<title id="102" chapters="00:00:00;">
    <videoObject file="../movies/3/columbiatristardvd_pal.mpg" />

    <postCommands>
        <command>
            <!-- CallSS VTSM (menu ROOT, rsm_cell 1), 30 08 00 00 01 83 00 00 -->
            <opcode>[0x30][0x08][0x00][0x00][0x01][0x83][0x00][0x00]</opcode>
        </command>
    </postCommands>
</title>

</titles>
<menus>
    <menu id="2" entry="root">
        <preCommands>
            <command>
                <!-- GPRM11 = 0, 71 00 00 0b 00 00 00 00 -->
                <opcode>[0x71][0x00][0x00][0x0b][0x00][0x00][0x00][0x00]</opcode>
            </command>
        </preCommands>

        <postCommands>
            <command>
                <opcode>[0x20][0x04][0x0][0x0][0x0][0x0][0x0][0x01]</opcode>
                <!-- LinkPGCN 1, 20 04 00 00 00 00 01 -->
            </command>
        </postCommands>

        <!-- Play Title 1 button -->
        <button left="107" top="96" width="166" height="35">
            <command>
                <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 01 00 00 -->
                <opcode>[0x30][0x05][0x00][0x01][0x00][0x01][0x00][0x00]</opcode>
            </command>
        </button>

        <!-- Play Title 2 button -->
        <button left="107" top="155" width="170" height="37">
            <command>
                <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 02 00 00 -->
                <opcode>[0x30][0x05][0x00][0x01][0x00][0x02][0x00][0x00]</opcode>
            </command>
        </button>

        <!-- Play Title 3 button -->
        <button left="107" top="216" width="171" height="36">
            <command>
                <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 03 00 00 -->
                <opcode>[0x30][0x05][0x00][0x01][0x00][0x03][0x00][0x00]</opcode>
            </command>
        </button>

        <!-- Play All button -->
        <button left="107" top="401" width="112" height="36">
            <command>
                <!-- LinkPGCN 2, 20 04 00 00 00 00 02 -->
                <opcode>[0x20][0x04][0x0][0x0][0x0][0x0][0x0][0x02]</opcode>
            </command>
        </button>

        <mask file="menu\root_menu_mask.bmp" patternColor="#000000"
backgroundColor="#FFFFFF" emphasisColor1="#000001" emphasisColor2="#000002" />
        <display emphasis2Contrast="0" emphasis1Contrast="0" backgroundContrast="0"
patternContrast="0" emphasis2Color="#000000" emphasis1Color="#000000"
patternColor="#000000" />
        <action emphasis2Contrast="0" emphasis1Contrast="0" backgroundContrast="0"
patternContrast="7" emphasis2Color="#000000" emphasis1Color="#000000"
backgroundColor="#000000" patternColor="#0000FF" />

```

```

    <selection emphasis2Contrast="0" emphasis1Contrast="0" backgroundContrast="0"
patternContrast="7" emphasis2Color="#000000" emphasis1Color="#000000"
backgroundColor="#000000" patternColor="#FF0000" />
    <background file="menu\root_menu.mpg" />
  </menu>

  <!-- Play All Dummy menu -->
  <menu id="101">
    <preCommands>
      <command>
        <!-- GPRM11 = 1, 71 00 00 0b 00 01 00 00 -->
        <opcode>[0x71][0x00][0x00][0x0b][0x00][0x01][0x00][0x00]</opcode>
      </command>

      <command>
        <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 01 00 00 -->
        <opcode>[0x30][0x05][0x00][0x01][0x00][0x01][0x00][0x00]</opcode>
      </command>
    </preCommands>
  </menu>
</menus>
</titleSet>
</dvd>

```

### 3.1.2.2.1.3.1.1 <opcode> Element

Specifies the Virtual Machine (VM) command opcode. The opcode is 8 bytes long. Each byte is specified in as hexadecimal value [0xXX].

```
<opcode> [0xXX] [0xXX] [0xXX] [0xXX] [0xXX] [0xXX] [0xXX] [0xXX] </opcode>
```

#### Remarks

#### Example

The following example shows a DVD that has title menu, root menu and three titles. The title menu is a dummy one. It simply redirects to the root menu. The root menu supports playing each clip individually and returns back or playing all clips in sequence.

```

<?xml version="1.0" encoding="utf-8"?>
<dvd xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" version="2.1.2.1"
xmlns="http://www.primosoftware.com/dvdbuilder">
  <videoManager>
    <firstPlayCommands>
      <command>
        <!-- JumpSS VTSM (vts 1, tt 1, menu ROOT), 30 06 00 01 01 83 00 00 -->
        <opcode>[0x30][0x06][0x00][0x01][0x01][0x83][0x00][0x00]</opcode>
      </command>
    </firstPlayCommands>

    <menus>
      <menu id="1" entry="title">
        <preCommands>
          <command>
            <!-- JumpSS VTSM (vts 1, tt 1, menu ROOT), 30 06 00 01 01 83 00 00 -->
            <opcode>[0x30][0x06][0x00][0x01][0x01][0x83][0x00][0x00]</opcode>
          </command>
        </preCommands>

        <background file="menu\title_menu.mpg" />
      </menu>
    </menus>
  </videoManager>

  <!-- GPRM11 -> Play All Mode (True=1, False=0) -->

```

```

<titleSet>
  <titles>
    <title id="100" chapters="00:00:00;">
      <videoObject file="../movies/3/columbiatristardvd_pal.mpg" />

      <postCommands>
        <command>
          <!-- if (GPRM11 == 1) Goto 3, 00 a1 00 0b 00 01 00 03 -->
          <opcode>[0x00][0xa1][0x00][0x0b][0x00][0x01][0x00][0x03]</opcode>
        </command>
        <command>
          <!-- CallSS VTSM (menu ROOT, rsm_cell 1), 30 08 00 00 01 83 00 00 -->
          <opcode>[0x30][0x08][0x00][0x00][0x01][0x83][0x00][0x00]</opcode>
        </command>

        <command>
          <!-- JumpVTS_PTT (tt 2, ptt 1), 30 05 00 01 00 02 00 00 -->
          <opcode>[0x30][0x05][0x00][0x01][0x00][0x02][0x00][0x00]</opcode>
        </command>
      </postCommands>
    </title>

    <title id="101" chapters="00:00:00;">
      <videoObject file="../movies/4/dreamworksanim_pal.mpg" />

      <postCommands>
        <command>
          <!-- if (GPRM11 == 1) Goto 3, 00 a1 00 0b 00 01 00 03 -->
          <opcode>[0x00][0xa1][0x00][0x0b][0x00][0x01][0x00][0x03]</opcode>
        </command>
        <command>
          <!-- CallSS VTSM (menu ROOT, rsm_cell 1), 30 08 00 00 01 83 00 00 -->
          <opcode>[0x30][0x08][0x00][0x00][0x01][0x83][0x00][0x00]</opcode>
        </command>

        <command>
          <!-- JumpVTS_PTT (tt 3, ptt 1), 30 05 00 01 00 03 00 00 -->
          <opcode>[0x30][0x05][0x00][0x01][0x00][0x03][0x00][0x00]</opcode>
        </command>
      </postCommands>
    </title>

    <title id="102" chapters="00:00:00;">
      <videoObject file="../movies/3/columbiatristardvd_pal.mpg" />

      <postCommands>
        <command>
          <!-- CallSS VTSM (menu ROOT, rsm_cell 1), 30 08 00 00 01 83 00 00 -->
          <opcode>[0x30][0x08][0x00][0x00][0x01][0x83][0x00][0x00]</opcode>
        </command>
      </postCommands>
    </title>
  </titles>
  <menus>
    <menu id="2" entry="root">
      <preCommands>
        <command>
          <!-- GPRM11 = 0, 71 00 00 0b 00 00 00 00 -->
          <opcode>[0x71][0x00][0x00][0x0b][0x00][0x00][0x00][0x00]</opcode>
        </command>
      </preCommands>

      <postCommands>
        <command>
          <opcode>[0x20][0x04][0x0][0x0][0x0][0x0][0x0][0x01]</opcode>
          <!-- LinkPGCN 1, 20 04 00 00 00 00 01 -->
        </command>
      </postCommands>
    </menu>
  </menus>

```

```

<!-- Play Title 1 button -->
<button left="107" top="96" width="166" height="35">
  <command>
    <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 01 00 00 -->
    <opcode>[0x30][0x05][0x00][0x01][0x00][0x01][0x00][0x00]</opcode>
  </command>
</button>

<!-- Play Title 2 button -->
<button left="107" top="155" width="170" height="37">
  <command>
    <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 02 00 00 -->
    <opcode>[0x30][0x05][0x00][0x01][0x00][0x02][0x00][0x00]</opcode>
  </command>
</button>

<!-- Play Title 3 button -->
<button left="107" top="216" width="171" height="36">
  <command>
    <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 03 00 00 -->
    <opcode>[0x30][0x05][0x00][0x01][0x00][0x03][0x00][0x00]</opcode>
  </command>
</button>

<!-- Play All button -->
<button left="107" top="401" width="112" height="36">
  <command>
    <!-- LinkPGCN 2, 20 04 00 00 00 00 02 -->
    <opcode>[0x20][0x04][0x00][0x00][0x00][0x00][0x02]</opcode>
  </command>
</button>

  <mask file="menu\root_menu_mask.bmp" patternColor="#000000"
backgroundColor="#FFFFFF" emphasisColor1="#000001" emphasisColor2="#000002" />
  <display emphasis2Contrast="0" emphasis1Contrast="0" backgroundContrast="0"
patternContrast="0" emphasis2Color="#000000" emphasis1Color="#000000"
patternColor="#000000" />
  <action emphasis2Contrast="0" emphasis1Contrast="0" backgroundContrast="0"
patternContrast="7" emphasis2Color="#000000" emphasis1Color="#000000"
backgroundColor="#000000" patternColor="#0000FF" />
  <selection emphasis2Contrast="0" emphasis1Contrast="0" backgroundContrast="0"
patternContrast="7" emphasis2Color="#000000" emphasis1Color="#000000"
backgroundColor="#000000" patternColor="#FF0000" />
  <background file="menu\root_menu.mpg" />
</menu>

<!-- Play All Dummy menu -->
<menu id="101">
  <preCommands>
    <command>
      <!-- GPRM11 = 1, 71 00 00 0b 00 01 00 00 -->
      <opcode>[0x71][0x00][0x00][0x0b][0x00][0x01][0x00][0x00]</opcode>
    </command>

    <command>
      <!-- (Jump_VTS_PTT) Jump to Chapter, 30 05 00 01 00 01 00 00 -->
      <opcode>[0x30][0x05][0x00][0x01][0x00][0x01][0x00][0x00]</opcode>
    </command>
  </preCommands>
</menu>
</menus>
</titleSet>
</dvd>

```

### 3.1.2.2.2 <subpicturePalette> Element

Configures the sub-picture/subtitles color palette.

```

<subpicturePalette>
  <color>
</subpicturePalette>

```

**Child Elements**

Element	Description
<color>	Adds a sub-picture palette color.

**Remarks**

There can be up to 16 colors per sub-picture/subtitles palette.

**Version**

1.10

**Example**

The following example shows a DVD that has subtitles in three languages: Bulgarian, English and French. Viewers of this DVD will be able to select Bulgarian, English or French subtitles by pressing the Subtitles button on the remote control unit of their player. A sub-picture palette is defined with the <subpicturePalette> element, so the subtitles will be shown in yellow color.

```
<?xml version="1.0" encoding="utf-8"?>
<dvd version="1.1.0.1" xmlns="http://www.primosoftware.com/dvdbuilder">
  <videoManager firstPlayNavigate="Title=1;" />
  <titleSet>
    <subpictureStreams>
      <stream languageCode="BG" mpegStreamID="0xBD" mpegSubstreamID="0x20" />
      <stream languageCode="EN" mpegStreamID="0xBD" mpegSubstreamID="0x21" />
      <stream languageCode="FR" mpegStreamID="0xBD" mpegSubstreamID="0x22" />
    </subpictureStreams>

    <titles>
      <title id="1" chapters="00:00:00">
        <videoObject file="1.mpg" />
      </title>
      <subpicturePalette>
        <color index="1" value="#FFFF00" />
      </subpicturePalette>
    </titles>
  </titleSet>
</dvd>
```

**3.1.2.2.2.1 <color>****Description**

Specifies a color from the sub-picture color palette.

```
<color
  index="num"
  value="#ffffff" />
```

**Required Attributes**

Attribute	Option	Description
index	1 to 16	Index of the color in the sub-picture palette.
value	This is an RGB color value expressed in hexadecimal format ("#ffffff").	Color value.

### 3.1.2.3 <audioStreams> Element

Configures the audio streams of a title set.

```
<audioStreams>
  <stream>
</audioStreams>
```

#### Child Elements

Element	Description
<stream>	Adds an audio stream.

#### Remarks

There can be up to 8 audio streams in a video title set.

#### Version

1.10

#### Example

The following example shows a DVD that has one title and audio available in two languages: Bulgarian and English. Viewers of this DVD will be able to select Bulgarian or English audio from the language menu of the DVD player.

```
<?xml version="1.0" encoding="utf-8"?>
<dvd version="1.1.0.1" xmlns="http://www.primosoftware.com/dvdbuilder">
  <videoManager firstPlayNavigate="Title=1;" />
  <titleSet>
    <audioStreams>
      <stream languageCode="BG" mpegStreamID="0xBD" mpegSubstreamID="0x80" />
      <stream languageCode="EN" mpegStreamID="0xBD" mpegSubstreamID="0x81" />
    </audioStreams>

    <titles>
      <title id="1" chapters="00:00:00">
        <videoObject file="1.mpg" />
      </title>
    </titles>
  </titleSet>
</dvd>
```

#### 3.1.2.3.1 <stream> Element

Maps a MPEG-2 stream to a DVD language. The MPEG-2 stream defined by this element can be an audio stream or a subtitle/sub-picture stream.

```
<stream
  languageCode="string"
  mpegStreamID="num"
  mpegSubStreamID="num" />
```

#### Required Attributes

Attribute	Description
languageCode	A two letter language code (e.g. EN, FR). See DVD Language Codes (see page 109) for more information.
mpegStreamID	MPEG stream ID. A stream with this ID must exist in all video objects in the title set.

**Optional Attributes**

Attribute	Description
mpegSubstreamID	MPEG substream ID. Use this attribute when the value of the mpegStreamID attribute is set to 0xBD (private stream). A private stream with this substream ID must exist in all video objects in the title set.

**Audio Streams**

Stream	MPEG Stream ID	MPEG Substream ID
MP2 (MPEG-2 Audio)	From 0xC0 to 0xDF	N/A
AC-3 (Dolby Digital Audio)	0xBD	From 0x80 to 0x87
DTS (Digital Theater System Audio)	0xBD	From 0x88 to 0x8F
LPCM (Linear Pulse Code Modulation Audio)	0xBD	From 0xA0 to 0xA7

**Sub-picture Streams**

Stream	MPEG Stream ID	MPEG Substream ID
Subtitles	0xBD	From 0x20 to 0x3F

**Version**

1.10

**Example**

The following example shows a DVD that has one title with two audio streams (Bulgarian and English) and three subtitle streams (Bulgarian, English and French).

```
<?xml version="1.0" encoding="utf-8"?>
<dvd version="1.1.0.1" xmlns="http://www.primosoftware.com/dvdbuilder">
  <videoManager firstPlayNavigate="Title=1;" />
  <titleSet>
    <audioStreams>
      <stream languageCode="BG" mpegStreamID="0xBD" mpegSubstreamID="0x80" />
      <stream languageCode="EN" mpegStreamID="0xBD" mpegSubstreamID="0x81" />
    </audioStreams>

    <subPictureStreams>
      <stream languageCode="BG" mpegStreamID="0xBD" mpegSubstreamID="0x20" />
      <stream languageCode="EN" mpegStreamID="0xBD" mpegSubstreamID="0x21" />
      <stream languageCode="FR" mpegStreamID="0xBD" mpegSubstreamID="0x22" />
    </subPictureStreams>

    <titles>
      <title id="1" chapters="00:00:00">
        <videoObject file="1.mpg" />
      </title>
    </titles>
  </titleSet>
</dvd>
```

**3.1.2.4 <subpictureStreams> Element**

Configures the sub-picture/subtitle streams of a title set.

```
<subpictureStreams>
```

```
<stream>
</subpictureStreams>
```

### Child Elements

Element	Description
<stream>	Adds a sub-picture/subtitle stream.

### Remarks

There can be up to 32 sub-picture/subtitle streams in a video title set.

### Version

1.10

### Example

The following example shows a DVD that has subtitles in three languages: Bulgarian, English and French. Viewers of this DVD will be able to select Bulgarian, English or French subtitles by pressing the Subtitles button on the remote control unit of their player. Because there is no sub-picture palette defined (no <subpicturePalette> element) the subtitles will be shown in the default white color.

```
<?xml version="1.0" encoding="utf-8"?>
<dvd version="1.1.0.1" xmlns="http://www.primosoftware.com/dvdbuilder">
  <videoManager firstPlayNavigate="Title=1;" />
  <titleSet>
    <subpictureStreams>
      <stream languageCode="BG" mpegStreamID="0xBD" mpegSubstreamID="0x20" />
      <stream languageCode="EN" mpegStreamID="0xBD" mpegSubstreamID="0x21" />
      <stream languageCode="FR" mpegStreamID="0xBD" mpegSubstreamID="0x22" />
    </subpictureStreams>

    <titles>
      <title id="1" chapters="00:00:00">
        <videoObject file="1.mpg" />
      </title>
    </titles>
  </titleSet>
</dvd>
```

### 3.1.2.4.1 <stream> Element

Maps a MPEG-2 stream to a DVD language. The MPEG-2 stream defined by this element can be an audio stream or a subtitle/sub-picture stream.

```
<stream
  languageCode="string"
  mpegStreamID="num"
  mpegSubStreamID="num" />
```

### Required Attributes

Attribute	Description
languageCode	A two letter language code (e.g. EN, FR). See DVD Language Codes ( <a href="#">see page 109</a> ) for more information.
mpegStreamID	MPEG stream ID. A stream with this ID must exist in all video objects in the title set.

### Optional Attributes

Attribute	Description
mpegSubstreamID	MPEG substream ID. Use this attribute when the value of the mpegStreamID attribute is set to 0xBD (private stream). A private stream with this substream ID must exist in all video objects in the title set.

### Audio Streams

Stream	MPEG Stream ID	MPEG Substream ID
MP2 (MPEG-2 Audio)	From 0xC0 to 0xDF	N/A
AC-3 (Dolby Digital Audio)	0xBD	From 0x80 to 0x87
DTS (Digital Theater System Audio)	0xBD	From 0x88 to 0x8F
LPCM (Linear Pulse Code Modulation Audio)	0xBD	From 0xA0 to 0xA7

### Sub-picture Streams

Stream	MPEG Stream ID	MPEG Substream ID
Subtitles	0xBD	From 0x20 to 0x3F

### Version

1.10

### Example

The following example shows a DVD that has one title with two audio streams (Bulgarian and English) and three subtitle streams (Bulgarian, English and French).

```
<?xml version="1.0" encoding="utf-8"?>
<dvd version="1.1.0.1" xmlns="http://www.primosoftware.com/dvdbuilder">
  <videoManager firstPlayNavigate="Title=1;" />
  <titleSet>
    <audioStreams>
      <stream languageCode="BG" mpegStreamID="0xBD" mpegSubstreamID="0x80" />
      <stream languageCode="EN" mpegStreamID="0xBD" mpegSubstreamID="0x81" />
    </audioStreams>

    <subPictureStreams>
      <stream languageCode="BG" mpegStreamID="0xBD" mpegSubstreamID="0x20" />
      <stream languageCode="EN" mpegStreamID="0xBD" mpegSubstreamID="0x21" />
      <stream languageCode="FR" mpegStreamID="0xBD" mpegSubstreamID="0x22" />
    </subPictureStreams>

    <titles>
      <title id="1" chapters="00:00:00">
        <videoObject file="1.mpg" />
      </title>
    </titles>
  </titleSet>
</dvd>
```

# 4 DVD Language Codes

## DVD Language Codes

Language	Code	Language	Code
1. Abkhazian	AB	69. Lithuanian	LT
2. Afar	AA	70. Macedonian	MK
3. Afrikaans	AF	71. Malagasy	MG
4. Albanian	SQ	72. Malay	MS
5. Amharic, Ameharic	AM	73. Malayalam	ML
6. Arabic	AR	74. Maltese	MT
7. Armenian	HY	75. Maori	MI
8. Assamese	AS	76. Marathi	MR
9. Aymara	AY	77. Moldavian	MO
10. Azerbaijani	AZ	78. Mongolian	MN
11. Bashkir	BA	79. Nauru	NA
12. Basque	EU	80. Nepali	NE
13. Bengali, Bangla	BN	81. Norwegian (Norsk)	NO
14. Bhutani	DZ	82. Occitan	OC
15. Bihari	BH	83. Oriya	OR
16. Bislama	BI	84. Afan (Oromo)	OM
17. Breton	BR	85. Panjabi	PA
18. Bulgarian	BG	86. Pashto, Pushto	PS
19. Burmese	MY	87. Persian	FA
20. Byelorussian	BE	88. Polish	PL
21. Cambodian	KM	89. Portuguese	PT
22. Catalan	CA	90. Quechua	QU
23. Chinese	ZH	91. Rhaeto-Romance	RM
24. Corsican	CO	92. Romanian	RO
25. Hrvatski (Croatian)	HR	93. Russian	RU
26. Czech (Ceske)	CS	94. Samoan	SM

27. Dansk (Danish)	DA	95. Sangho	SG
28. Dutch (Nederlands)	NL	96. Sanskrit	SA
29. English	EN	97. Scots Gaelic	GD
30. Esperanto	EO	98. Serbo-Croatian	SH
31. Estonian	ET	99. Sesotho	ST
32. Faroese	FO	100. Serbian	SR
33. Fiji	FJ	101. Setswana	TN
34. Finnish	FI	102. Shona	SN
35. French	FR	103. Sindhi	SD
36. Frisian	FY	104. Singhalese	SI
37. Galician	GL	105. Siswati	SS
38. Georgian	KA	106. Slovak	SK
39. Deutsch (German)	DE	107. Slovenian	SL
40. Greek	EL	108. Somali	SO
41. Greenlandic	KL	109. Spanish (Espanol)	ES
42. Guarani	GN	110. Sundanese	SU
43. Gujarati	GU	111. Swahili	SW
44. Hausa	HA	112. Svenska (Swedish)	SV
45. Hebrew	IW	113. Tagalog	TL
46. Hindi	HI	114. Tajik	TG
47. Hungarian	HU	115. Tatar	TT
48. Islenka (Icelandic)	IS	116. Tamil	TA
49. Indonesian	IN	117. Telugu	TE
50. Interlingua	IA	118. Thai	TH
51. Interlingue	IE	119. Tibetan	BO
52. Inupiak	IK	120. Tigrinya	TI
53. Irish	GA	121. Tonga	TO
54. Italian	IT	122. Tsonga	TS
55. Japanese	JA	123. Turkish	TR
56. Javanese	JW	124. Turkmen	TK
57. Kannada	KN	125. Twi	TW

58. Kashmiri	KS	126. Ukranian	UK
59. Kazakh	KK	127. Urdu	UR
60. Kinyarwanda	RW	128. Uzbek	UZ
61. Kirghiz	KY	129. Vietnamese	VI
62. Kirundi	RN	130. Volapuk	VO
63. Korean	KO	131. Welsh	CY
64. Kurdish	KU	132. Wolof	WO
65. Laothian	LO	133. Yiddish	JI
66. Latin	LA	134. Yoruba	YO
67. Latvian, Lettish	LV	135. Xhosa	XH
68. Lingala	LN	136. Zulu	ZU
<b>Language</b>	<b>Code</b>	<b>Language</b>	<b>Code</b>

# 5 Symbol Reference

## 5.1 PrimoSoftware Namespace

### Namespaces

Namespace	Description
DVDBuilder ( <a href="#">see page 112</a> )	

### 5.1.1 PrimoSoftware::DVDBuilder Namespace

#### Namespaces


Namespace	Description
VR ( <a href="#">see page 132</a> )	Video Recording

#### Classes



Class	Description
DVDBuilder_ProgressEventArgs ( <a href="#">see page 128</a> )	Provides data for the DVDBuilder::OnProgress event. A DVDBuilder::OnProgress event is raised to report the progress of the DVD creation process.
DVDBuilder_StatusEventArgs ( <a href="#">see page 129</a> )	Provides data for the DVDBuilder::OnStatus event. A DVDBuilder::OnStatus event is raised to report the status of the dvd creation process.
Library ( <a href="#">see page 130</a> )	Class factory for DVDBuilder interfaces.

#### Interfaces






Interface	Description
↔ DataStream ( <a href="#">see page 114</a> )	The DataStream interface can be used to implement a data source for VOB(Video Object) data. To use this interface you must also provide implementation of the InputDataStreamFactory ( <a href="#">see page 123</a> ) interface.
↔ DVDBuilder ( <a href="#">see page 117</a> )	This interface provides methods that enable an application to build a DVD video file structure.
↔ InputDataStreamFactory ( <a href="#">see page 123</a> )	The InputDataStreamFactory interface is a class factory for creating and destroying DataStream ( <a href="#">see page 114</a> ) objects. This interface must be implemented by the application and advertised via the DVDBuilder::InputDataStreamFactory property.

 SubpictureEncoder ( <a href="#">see page 124</a> )	DVD subpicture stream encoder. Currently the only supported input is Spruce Technologies STL file with subtitles defined as images. Subtitle generation from text and fonts is currently not supported. The output is always MPEG-2 PS (Program Stream). For STL format description see: <a href="#">STL Subtitle File Format</a>
--	--

### Legend

	Interface
	Enumeration

### Structs, Records, Enums

Struct, Record, Enum	Description
 DVDBuilderError ( <a href="#">see page 167</a> )	The DVDBuilderError type defines a range of error codes which are returned by the DVDBuilder.Error ( <a href="#">see page 119</a> ) property. Each error code has an auxiliary message that helps to identify the error. It is called an <i>error hint</i> and is returned by the method DVDBuilder.ErrorHint ( <a href="#">see page 120</a> ). Some error codes do not have a hint since such is not needed to identify the error. Other error codes specify as hints invalid project element or the path to a problematic input file. The hint format is specific to the error code that it explains but there are some common conventions: <ul style="list-style-type: none"> <li>• A text in <i>italics</i> specifies... more (<a href="#">see page 167</a>)</li> </ul>
 DVDBuilderStatus ( <a href="#">see page 177</a> )	The DVDBuilderStatus type defines a range of status information. Members of the DVDBuilderStatus enumeration are passed to the DVDBuilder::OnStatus event, so that the application can respond to the changing status of the DVDBuilder ( <a href="#">see page 117</a> ) object being used.
 ErrorFacility ( <a href="#">see page 177</a> )	The ErrorFacility type defines the error source.
 SubpictureEncoderError ( <a href="#">see page 178</a> )	The SubpictureEncoderError type defines a range of error codes which are returned by the SubpictureEncoder.Error ( <a href="#">see page 125</a> ) property.
 SubpictureEncoderImageMaskColor ( <a href="#">see page 179</a> )	The SubpictureEncoderImageMaskColor type type defines the subpicture mask color indices.

### Types





Type	Description
DVDBuilder_ContinueEventHandler ( <a href="#">see page 180</a> )	Represents the method that handles the DVDBuilder::OnContinue event raised to check if dvd creation should continue. When you create a DVDBuilder_ContinueEventHandler delegate, you identify the method that will handle the event. To associate the event with your event handler, add an instance of the delegate to the event. The event handler is called whenever the event occurs, unless you remove the delegate. The declaration of your event handler must have the same parameters as the DVDBuilder_ContinueEventHandler delegate declaration.

DVDBuilder_ProgressEventHandler ( <a href="#">see page 181</a> )	<p>Represents the method that handles the DVDBuilder::OnProgress event raised to report the dvd creation process progress.</p> <p>When you create a DVDBuilder_ProgressEventHandler delegate, you identify the method that will handle the event. To associate the event with your event handler, add an instance of the delegate to the event. The event handler is called whenever the event occurs, unless you remove the delegate.</p> <p>The declaration of your event handler must have the same parameters as the DVDBuilder_ProgressEventHandler delegate declaration.</p>
DVDBuilder_StatusEventHandler ( <a href="#">see page 181</a> )	<p>Represents the method that handles the DVDBuilder::OnStatus event raised to report the status of the dvd creation process.</p> <p>When you create a DVDBuilder_StatusEventHandler delegate, you identify the method that will handle the event. To associate the event with your event handler, add an instance of the delegate to the event. The event handler is called whenever the event occurs, unless you remove the delegate.</p> <p>The declaration of your event handler must have the same parameters as the DVDBuilder_StatusEventHandler delegate declaration.</p>

## 5.1.1.1 Interfaces

The following table lists interfaces in this documentation.

### Interfaces

Interface	Description
 <a href="#">DataStream</a> ( <a href="#">see page 114</a> )	The DataStream interface can be used to implement a data source for VOB(Video Object) data. To use this interface you must also provide implementation of the <a href="#">InputDataStreamFactory</a> ( <a href="#">see page 123</a> ) interface.
 <a href="#">DVDBuilder</a> ( <a href="#">see page 117</a> )	This interface provides methods that enable an application to build a DVD video file structure.
 <a href="#">InputDataStreamFactory</a> ( <a href="#">see page 123</a> )	The InputDataStreamFactory interface is a class factory for creating and destroying <a href="#">DataStream</a> ( <a href="#">see page 114</a> ) objects. This interface must be implemented by the application and advertised via the DVDBuilder::InputDataStreamFactory property.
 <a href="#">SubpictureEncoder</a> ( <a href="#">see page 124</a> )	DVD subpicture stream encoder. Currently the only supported input is Spruce Technologies STL file with subtitles defined as images. Subtitle generation from text and fonts is currently not supported. The output is always MPEG-2 PS (Program Stream). For STL format description see: <a href="#">STL Subtitle File Format</a>

### Legend

	Interface
---	-----------

#### 5.1.1.1.1 DataStream

##### Class Hierarchy

PrimoSoftware::DVDBuilder::DataStream

**Visual Basic**

```
Public Interface DataStream
```

**C#**

```
public interface DataStream;
```

**Description**

The DataStream interface can be used to implement a data source for VOB(Video Object) data. To use this interface you must also provide implementation of the InputDataStreamFactory ([see page 123](#)) interface.







**Version**

1.10


**See Also**

InputDataStreamFactory ([see page 123](#))

**Members****Methods**

Method	Description
 Close ( <a href="#">see page 115</a> )	Closes the stream object.
 GetSize ( <a href="#">see page 115</a> )	Returns the size of the stream object. In some cases the GetSize method may be called without calling the Open ( <a href="#">see page 116</a> ) method first. This method should return a valid size even when the stream is closed.
 Open ( <a href="#">see page 116</a> )	Opens the stream object.
 Read ( <a href="#">see page 116</a> )	Reads a specified number of bytes from the stream object into memory starting at the current seek pointer.
 Seek ( <a href="#">see page 117</a> )	Changes the seek pointer to a new location relative to the beginning of the stream.
 Write ( <a href="#">see page 117</a> )	Writes a specified number of bytes into the stream object starting at the current seek pointer.

**Legend**

	Method
---	--------

**5.1.1.1.1.1 DataStream.Close****Visual Basic**

```
Public Sub Close()
```

**C#**

```
public void Close();
```

**Description**

Closes the stream object.

**5.1.1.1.1.2 DataStream.GetSize****Visual Basic**

```
Public Funtion GetSize() As Long
```

**C#**

```
public long GetSize();
```

**Returns**

The number of bytes in the stream.

When the stream size is unknown return 0. DVDBuilder (see page 117) will stop reading from the stream when the `DataStream.Read` (see page 116) method returns 0.

To indicate an error condition return -1.

**Description**

Returns the size of the stream object. In some cases the `GetSize` method may be called without calling the `DataStream.Open` (see page 116) method first. This method should return a valid size even when the stream is closed.

**5.1.1.1.1.3 DataStream.Open****Visual Basic**

```
Public Function Open() As Boolean
```

**C#**

```
public bool Open();
```

**Returns**

true Success.

false Failure.

**Description**

Opens the stream object.

**5.1.1.1.1.4 DataStream.Read****Visual Basic**

```
Public Function Read(ByVal readBuffer() As Byte, ByVal readBufferSize As Integer) As Integer
```

**C#**

```
public int Read(byte [] readBuffer, int readBufferSize);
```

**Parameters**

Parameters	Description
readBuffer	[out] Buffer into which the stream data is read.
readBufferSize	[in] Specifies the number of bytes to attempt to read from the stream object.

**Returns**

The number of bytes read from the stream object.

To indicate an error condition return -1.

**Description**

Reads a specified number of bytes from the stream object into memory starting at the current seek pointer.

### 5.1.1.1.1.5 DataStream.Seek

#### Visual Basic

```
Public Function Seek(ByVal offset As Long) As Boolean
```

#### C#

```
public bool Seek(long offset);
```

#### Parameters

Parameters	Description
offset	[in] Displacement to be added to the beginning of the stream.

#### Returns

true Success.

false Failure.

#### Description

Changes the seek pointer to a new location relative to the beginning of the stream.

### 5.1.1.1.1.6 DataStream.Write

#### Visual Basic

```
Public Function Write(ByVal data() As Byte, ByVal dataLength As Integer, ByRef bytesWritten As Integer) As Boolean
```

#### C#

```
public bool Write(byte[] data, int dataLength, out int bytesWritten);
```

#### Parameters

Parameters	Description
data	[in] A buffer with the data that should be written to the stream.
dataLength	[in] Number of bytes to write to the stream object.
bytesWritten	[out] a variable that receives the actual number of bytes written to the stream object.

#### Returns

true Success.

false Failure.

#### Description

Writes a specified number of bytes into the stream object starting at the current seek pointer.

### 5.1.1.1.2 DVDBuilder

#### Class Hierarchy

```
PrimoSoftware::DVDBuilder::DVDBuilder
```

#### Visual Basic

```
Public Interface DVDBuilder
    Inherits IDisposable
```

**C#**

```
public interface DVDBuilder : IDisposable;
```















**Description**

This interface provides methods that enable an application to build a DVD video file structure.




**Version**

1.00


**Members****Properties**

Property	Description
  Error ( <a href="#">see page 119</a> )	Gets the error code for the last operation.  See the Error Handling ( <a href="#">see page 4</a> ) topic for a detailed description.
  ErrorFacility ( <a href="#">see page 120</a> )	Gets the error source of the last operation. If the last operation is successful the value of this property is ErrorFacility::Success.  See the Error Handling ( <a href="#">see page 4</a> ) topic for a detailed description.
  ErrorHint ( <a href="#">see page 120</a> )	Gets a string that helps to pinpoint the reason for a DVDBuilder error.  The hint is specific for each DVDBuilderError ( <a href="#">see page 167</a> ). See DVDBuilderError ( <a href="#">see page 167</a> ) for details. The hint is also available for system errors, i.e. when ErrorFacility ( <a href="#">see page 120</a> ) is ErrorFacility.System. When there's no hint an empty string is returned. This method never returns null (Nothing).
  InputDataStreamFactory ( <a href="#">see page 120</a> )	Sets a factory object to be used for creating and destroying DataStream ( <a href="#">see page 114</a> ) objects. The DVDBuilder engine calls the provided factory class for each <videoObject> XML element which have its "callback" attribute set to "true". The data stream objects which are returned by the factory are then used for reading the actual video object data.
 OutputFolder ( <a href="#">see page 122</a> )	Sets the output directory for the DVD files.
  Project ( <a href="#">see page 122</a> )	Sets the DVD project XML from a memory buffer. The data in the buffer must be encoded as UTF-8 or UTF-16.
 ProjectFile ( <a href="#">see page 122</a> )	Sets a DVD project file. The DVD project file contains XML that configures the DVD titles, video objects and menus.
  SystemErrorDomain ( <a href="#">see page 122</a> )	Gets the error source of the last operation when there's a system error. See the Error Handling ( <a href="#">see page 4</a> ) topic for a detailed description.






**Events**

Event	Description
 OnContinue ( <a href="#">see page 121</a> )	Raised to check if DVD creation process should continue. The event handler receives an argument of type EventArgs containing data related to this event.
 OnProgress ( <a href="#">see page 121</a> )	Raised to report the progress of the DVD creation process. The event handler receives an argument of type DVDBuilder_ProgressEventArgs ( <a href="#">see page 128</a> ) containing data related to this event.
 OnStatus ( <a href="#">see page 121</a> )	Raised to report the DVDBuilder object status. The event handler receives an argument of type DVDBuilder_StatusEventArgs ( <a href="#">see page 129</a> ) containing data related to this event.

**Methods**

Method	Description
 Build ( <a href="#">see page 123</a> )	Builds a DVD file structure.

**Legend**

	Property
	read only
	write only
	Event
	Method

**5.1.1.1.2.1 DVDBuilder.Error****Visual Basic**

```
Public ReadOnly Property Error As Integer
```

**C#**

```
public int Error {get;}
```

**Description**

Gets the error code for the last operation.

See the Error Handling ([see page 4](#)) topic for a detailed description.

**Version**

1.10

**See Also**

DVDBuilderError ([see page 167](#))

### 5.1.1.1.2.2 DVDBuilder.ErrorFacility

#### Visual Basic

```
Public ReadOnly Property ErrorFacility As Integer
```

#### C#

```
public int ErrorFacility {get;}
```

#### Description

Gets the error source of the last operation. If the last operation is successful the value of this property is ErrorFacility::Success.

See the Error Handling ([see page 4](#)) topic for a detailed description.

#### Version

2.0

#### See Also

DVDBuilderError ([see page 167](#))

### 5.1.1.1.2.3 DVDBuilder.ErrorHint

#### Visual Basic

```
Public ReadOnly Property ErrorHint As String
```

#### C#

```
public string ErrorHint {get;}
```

#### Description

Gets a string that helps to pinpoint the reason for a DVDBuilder ([see page 117](#)) error.

The hint is specific for each DVDBuilderError ([see page 167](#)). See DVDBuilderError ([see page 167](#)) for details.

The hint is also available for system errors, i.e. when DVDBuilder.ErrorFacility ([see page 120](#)) is ErrorFacility.System.

When there's no hint an empty string is returned. This method never returns null (Nothing).

#### Version

2.0

### 5.1.1.1.2.4 DVDBuilder.InputDataStreamFactory

#### Visual Basic

```
Public WriteOnly Property InputDataStreamFactory As InputDataStreamFactory
```

#### C#

```
public InputDataStreamFactory InputDataStreamFactory {set;}
```

#### Description

Sets a factory object to be used for creating and destroying DataStream ([see page 114](#)) objects.

The DVDBuilder ([see page 117](#)) engine calls the provided factory class for each <videoObject> XML element which have

its "callback" attribute set to "true". The data stream objects which are returned by the factory are then used for reading the actual video object data.

**Version**

1.10

### 5.1.1.1.2.5 DVDBuilder.OnContinue

**Visual Basic**

```
Public Event OnContinue As DVDBuilder_ContinueEventHandler
```

**C#**

```
public event DVDBuilder_ContinueEventHandler OnContinue;
```

**Description**

Raised to check if DVD creation process should continue.

The event handler receives an argument of type EventArgs containing data related to this event.

**See Also**

DVDBuilder\_ContinueEventHandler ([↗](#) see page 180)

### 5.1.1.1.2.6 DVDBuilder.OnProgress

**Visual Basic**

```
Public Event OnProgress As DVDBuilder_ProgressEventHandler
```

**C#**

```
public event DVDBuilder_ProgressEventHandler OnProgress;
```

**Description**

Raised to report the progress of the DVD creation process.

The event handler receives an argument of type DVDBuilder\_ProgressEventArgs ([↗](#) see page 128) containing data related to this event.

**See Also**

DVDBuilder\_ProgressEventHandler ([↗](#) see page 181), DVDBuilder\_ProgressEventArgs ([↗](#) see page 128)

### 5.1.1.1.2.7 DVDBuilder.OnStatus

**Visual Basic**

```
Public Event OnStatus As DVDBuilder_StatusEventHandler
```

**C#**

```
public event DVDBuilder_StatusEventHandler OnStatus;
```

**Description**

Raised to report the DVDBuilder ([↗](#) see page 117) object status.

The event handler receives an argument of type DVDBuilder\_StatusEventArgs ([↗](#) see page 129) containing data related to this event.

**See Also**

DVDBuilder\_StatusEventHandler ([↗](#) see page 181), DVDBuilder\_StatusEventArgs ([↗](#) see page 129)

**5.1.1.1.2.8 DVDBuilder.OutputFolder****Visual Basic**

```
Public Property OutputFolder() As String
```

**C#**

```
public string OutputFolder {get; set;}
```

**Description**

Sets the output directory for the DVD files.

**5.1.1.1.2.9 DVDBuilder.Project****Visual Basic**

```
Public WriteOnly Property Project As Byte()
```

**C#**

```
public byte [] Project {set;}
```

**Description**

Sets the DVD project XML from a memory buffer. The data in the buffer must be encoded as UTF-8 or UTF-16.

**Version**

1.10

**5.1.1.1.2.10 DVDBuilder.ProjectFile****Visual Basic**

```
Public Property ProjectFile() As String
```

**C#**

```
public string ProjectFile {get; set;}
```

**Description**

Sets a DVD project file. The DVD project file contains XML that configures the DVD titles, video objects and menus.

**See Also**

DVD Project File ([↗](#) see page 7)

**5.1.1.1.2.11 DVDBuilder.SystemErrorDomain****Visual Basic**

```
Public ReadOnly Property SystemErrorDomain As Integer
```

**C#**

```
public int SystemErrorDomain {get;}
```

**Returns**

The source of the last system error.

**Description**

Gets the error source of the last operation when there's a system error.

See the Error Handling (see page 4) topic for a detailed description.

**Version**

2.0

**5.1.1.1.2.12 DVDBuilder.Build****Visual Basic**

```
Public Function Build() As Boolean
```

**C#**

```
public bool Build();
```

**Description**

Builds a DVD file structure.

**5.1.1.1.3 InputDataStreamFactory****Class Hierarchy**

```
PrimoSoftware::DVDBuilder::InputDataStreamFactory
```

**Visual Basic**

```
Public Interface InputDataStreamFactory
```

**C#**

```
public interface InputDataStreamFactory;
```

**Description**

The InputDataStreamFactory interface is a class factory for creating and destroying [DataStream](#) (see page 114) objects. This interface must be implemented by the application and advertised via the `DVDBuilder::InputDataStreamFactory` property.



**Version**

1.10

**See Also**

[DataStream](#) (see page 114)

**Members****Methods**

Method	Description
 <a href="#">Create</a> (see page 124)	Creates a <a href="#">DataStream</a> (see page 114) object.
 <a href="#">Destroy</a> (see page 124)	Destroys a <a href="#">DataStream</a> (see page 114) object.

**Legend**

	Method
---	--------

**5.1.1.1.3.1 InputDataStreamFactory.Create****Visual Basic**

```
Public Function Create(ByVal file As String) As DataStream
```

**C#**

```
public DataStream Create(string file);
```

**Parameters**

Parameters	Description
file	The value of the "file" attribute of a <videoObject> XML element.

**Description**

Creates a DataStream (see page 114) object.

**5.1.1.1.3.2 InputDataStreamFactory.Destroy****Visual Basic**

```
Public Sub Destroy(ByVal stream As DataStream)
```

**C#**

```
public void Destroy(DataStream stream);
```

**Parameters**

Parameters	Description
stream	The stream object to be destroyed.

**Description**

Destroys a DataStream (see page 114) object.

**5.1.1.1.4 SubpictureEncoder****Class Hierarchy**

```
PrimoSoftware::DVDBuilder::SubpictureEncoder
```

**Visual Basic**

```
Public Interface SubpictureEncoder
    Inherits IDisposable
```

**C#**

```
public interface SubpictureEncoder : IDisposable;
```

**Description**









DVD subpicture stream encoder. Currently the only supported input is Spruce Technologies STL file with subtitles defined as images. Subtitle generation from text and fonts is currently not supported. The output is always MPEG-2 PS (Program Stream).

For STL format description see: [STL Subtitle File Format](#)




**Version**

2.1.1




**Members****Properties**

Property	Description
  Error ( <a href="#">see page 125</a> )	Gets the error code for the last operation.  See the Error Handling ( <a href="#">see page 4</a> ) topic for a detailed description.
  ErrorFacility ( <a href="#">see page 126</a> )	Gets the error source of the last operation. If the last operation is successful the value of this property is ErrorFacility::Success.  See the Error Handling ( <a href="#">see page 4</a> ) topic for a detailed description.
  ErrorHint ( <a href="#">see page 126</a> )	Gets a string that helps to pinpoint the reason for a SubpictureEncoder error.
  SystemErrorDomain ( <a href="#">see page 126</a> )	Gets the error source of the last operation when there's a system error. See the Error Handling ( <a href="#">see page 4</a> ) topic for a detailed description.

**Methods**

Method	Description
 Encode ( <a href="#">see page 127</a> )	Encodes the input file into a DVD-Video subpicture stream.
 GetImageMaskColor ( <a href="#">see page 127</a> )	Gets the image mask color with the specified index.
 SetImageMaskColor ( <a href="#">see page 127</a> )	Sets the image mask color with the specified index.

**Legend**

	Property
	read only
	Method

**5.1.1.1.4.1 SubpictureEncoder.Error****Visual Basic**

```
Public ReadOnly Property Error As Integer
```

**C#**

```
public int Error {get;}
```

**Description**

Gets the error code for the last operation.

See the Error Handling ([see page 4](#)) topic for a detailed description.

**See Also**

[SubpictureEncoderError](#) (see page 178)

**5.1.1.1.4.2 SubpictureEncoder.ErrorFacility****Visual Basic**

```
Public ReadOnly Property ErrorFacility As Integer
```

**C#**

```
public int ErrorFacility {get;}
```

**Description**

Gets the error source of the last operation. If the last operation is successful the value of this property is `ErrorFacility::Success`.

See the [Error Handling](#) (see page 4) topic for a detailed description.

**See Also**

[SubpictureEncoderError](#) (see page 178)

**5.1.1.1.4.3 SubpictureEncoder.ErrorHint****Visual Basic**

```
Public ReadOnly Property ErrorHint As String
```

**C#**

```
public string ErrorHint {get;}
```

**Description**

Gets a string that helps to pinpoint the reason for a [SubpictureEncoder](#) (see page 124) error.

**5.1.1.1.4.4 SubpictureEncoder.SystemErrorDomain****Visual Basic**

```
Public ReadOnly Property SystemErrorDomain As Integer
```

**C#**

```
public int SystemErrorDomain {get;}
```

**Returns**

The source of the last system error.

**Description**

Gets the error source of the last operation when there's a system error.

See the [Error Handling](#) (see page 4) topic for a detailed description.

### 5.1.1.1.4.5 SubpictureEncoder.Encode

#### Visual Basic

```
Public Function Encode(ByVal inputFile As String, ByVal outputFile As String ) As Boolean
```

#### C#

```
public bool Encode(string inputFile, string outputFile);
```

#### Parameters

Parameters	Description
inputFile	Path to the input file. The input file should be in the STL format developed by Spruce Technologies. For STL format description see: <a href="#">STL Subtitle File Format</a> The subtitles must be defined as BMP or TIFF images. Subtitle generation from text and fonts is currently not supported.
outputFile	Path to the output file. The format of the output file is always MPEG-2 PS (Program Stream).

#### Description

Encodes the input file into a DVD-Video subpicture stream.

### 5.1.1.1.4.6 SubpictureEncoder.GetImageMaskColor

#### Visual Basic

```
Public Function GetImageMaskColor(ByVal index As SubpictureEncoderImageMaskColor) As Color
```

#### C#

```
public Color GetImageMaskColor(SubpictureEncoderImageMaskColor index);
```

#### Description

Gets the image mask color with the specified index.

### 5.1.1.1.4.7 SubpictureEncoder.SetImageMaskColor

#### Visual Basic

```
Public Sub SetImageMaskColor(ByVal index As SubpictureEncoderImageMaskColor, ByVal color As Color)
```

#### C#

```
public void SetImageMaskColor(SubpictureEncoderImageMaskColor index, Color color);
```

#### Description

Sets the image mask color with the specified index.

## 5.1.1.2 Classes

The following table lists classes in this documentation.

**Classes**

Class	Description
DVDBuilder_ProgressEventArgs (see page 128)	Provides data for the DVDBuilder::OnProgress event. A DVDBuilder::OnProgress event is raised to report the progress of the DVD creation process.
DVDBuilder_StatusEventArgs (see page 129)	Provides data for the DVDBuilder::OnStatus event. A DVDBuilder::OnStatus event is raised to report the status of the dvd creation process.
Library (see page 130)	Class factory for DVDBuilder (see page 112) interfaces.

**5.1.1.2.1 DVDBuilder\_ProgressEventArgs****Class Hierarchy**

PrimoSoftware::DVDBuilder::DVDBuilder\_ProgressEventArgs

**Visual Basic**

```
Public Class DVDBuilder_ProgressEventArgs
    Inherits EventArgs
```

**C#**

```
public class DVDBuilder_ProgressEventArgs : EventArgs;
```

**Description**


Provides data for the DVDBuilder::OnProgress event.

A DVDBuilder::OnProgress event is raised to report the progress of the DVD creation process.

**See Also**

DVDBuilder.OnProgress (see page 121)

**Members****Data Members**

Data Member	Description
 Percent (see page 128)	The percent for the progress of the DVD creation process.

**Legend**

	Data Member
---	-------------

**5.1.1.2.1.1 DVDBuilder\_ProgressEventArgs.Percent****Visual Basic**

```
Public Percent As System::Int32
```

**C#**

```
public System::Int32 Percent;
```

**Description**

The percent for the progress of the DVD creation process.

**5.1.1.2.2 DVDBuilder\_StatusEventArgs****Class Hierarchy**

PrimoSoftware::DVDBuilder::DVDBuilder\_StatusEventArgs

**Visual Basic**

```
Public Class DVDBuilder_StatusEventArgs
    Inherits EventArgs
```

**C#**

```
public class DVDBuilder_StatusEventArgs : EventArgs;
```

**Description**


Provides data for the DVDBuilder::OnStatus event.

A DVDBuilder::OnStatus event is raised to report the status of the dvd creation process.

**See Also**

DVDBuilder.OnStatus ([see page 121](#))

**Members****Data Members**

Data Member	Description
 Status ( <a href="#">see page 129</a> )	Status code defined by a member of the DVDBuilderStatus ( <a href="#">see page 177</a> ) enumeration.

**Legend**

	Data Member
---	-------------

**5.1.1.2.2.1 DVDBuilder\_StatusEventArgs.Status****Visual Basic**

```
Public Status As DVDBuilderStatus
```

**C#**

```
public DVDBuilderStatus Status;
```

**Description**

Status code defined by a member of the DVDBuilderStatus ([see page 177](#)) enumeration.

**See Also**

DVDBuilderStatus ([see page 177](#))

### 5.1.1.2.3 Library

#### Class Hierarchy

PrimoSoftware::DVDBuilder::Library

#### Visual Basic

```
Public Class Library
```

#### C#

```
public class Library;
```

#### Description

Class factory for PrimoSoftware::DVDBuilder Namespace (see page 112) interfaces.

#### Members

##### Methods

Method	Description
◆ CreateDVDBuilder (see page 130)	The CreateDVDBuilder method creates a new DVDBuilder (see page 117) object.
◆ CreateSubpictureEncoder (see page 130)	The CreateVideoRecorder (see page 131) method creates a new object for recording DVD video.
◆ CreateVideoRecorder (see page 131)	The CreateVideoRecorder method creates a new object for recording DVD video.
◆ Initialize (see page 131)	Initializes the managed code library. This method must be called before using any other method or object from this library.
◆ SetLicenseInfo (see page 131)	Sets customer license information. To obtain a license key please contact DVDBuilder (see page 117) SDK Sales at sales@dvdbuilder.com
◆ Terminate (see page 132)	This method must be called at the end of the application to cleanup the managed resources.

#### Legend

◆	Method
---	--------

#### 5.1.1.2.3.1 Library.CreateDVDBuilder

##### Visual Basic

```
Public Shared Function CreateDVDBuilder() As DVDBuilder
```

##### C#

```
public static DVDBuilder CreateDVDBuilder();
```

##### Description

The CreateDVDBuilder method creates a new DVDBuilder (see page 117) object.

#### 5.1.1.2.3.2 Library.CreateVideoRecorder

##### Visual Basic

```
Public Shared Function Library.CreateVideoRecorder() As VideoRecorder
```

**C#**

```
public static VideoRecorder Library.CreateVideoRecorder();
```

**Description**

The Library.CreateVideoRecorder (see page 131) method creates a new object for recording DVD video.

**Version**

2.00

### 5.1.1.2.3.3 Library.CreateVideoRecorder

**Visual Basic**

```
Public Shared Function CreateVideoRecorder() As VideoRecorder
```

**C#**

```
public static VideoRecorder CreateVideoRecorder();
```

**Description**

The CreateVideoRecorder method creates a new object for recording DVD video.

**Version**

2.00

### 5.1.1.2.3.4 Library.Initialize

**Visual Basic**

```
Public Shared Sub Initialize()
```

**C#**

```
public static void Initialize();
```

**Description**

Initializes the managed code library. This method must be called before using any other method or object from this library.

### 5.1.1.2.3.5 Library.SetLicenseInfo

**Visual Basic**

```
Public Sub SetLicenseInfo(ByVal licenseName As String, ByVal licenseCompany As String,  
ByVal licenseKey As String)
```

**C#**

```
public void SetLicenseInfo(string licenseName, string licenseCompany, string licenseKey);
```

**Description**

Sets customer license information. To obtain a license key please contact DVDBuilder (see page 117) SDK Sales at sales@dvdbuilder.com

**Version**

2.00

### 5.1.1.2.3.6 Library.Terminate

#### Visual Basic

```
Public Shared Sub Terminate()
```

#### C#

```
public static void Terminate();
```

#### Description

This method must be called at the end of the application to cleanup the managed resources.

## 5.1.1.3 PrimoSoftware::DVDBuilder::VR Namespace

Video Recording

#### Version

2.0

#### Interfaces

Interface	Description
↔ DeviceError ( <a href="#">see page 134</a> )	This interface represents the error state of a single device. An instance of this interface can be obtained via VideoRecorder::GetDeviceError.
↔ FileSystemDeviceConfig ( <a href="#">see page 136</a> )	This interface configures an Hdd device used for DVD video recording. The object obtained via VRDevice.Config ( <a href="#">see page 154</a> ) should be cast to this interface if the VRDevice.Type ( <a href="#">see page 156</a> ) is VRDeviceType.FileSystem.
↔ OpticalDiscDeviceConfig ( <a href="#">see page 137</a> )	This interface configures a burner device used for DVD video recording. The object obtained via VRDevice.Config ( <a href="#">see page 154</a> ) should be cast to this interface if the device type is VRDeviceType.OpticalDisc ( <a href="#">see page 166</a> ).
↔ Title ( <a href="#">see page 138</a> )	This interface is used to get information about a particular DVD-Video title. An instance of this interface can be obtained via TitleEnumerator ( <a href="#">see page 139</a> ).
↔ TitleEnumerator ( <a href="#">see page 139</a> )	This interface is used to enumerate all DVD-Video titles in a recording. An instance of this interface can be obtained via VideoRecorder.GetTitles ( <a href="#">see page 149</a> ). When it's not needed anymore the instance must be disposed.
↔ VideoRecorder ( <a href="#">see page 140</a> )	This interface represents the DVD video recorder and is used to carry out the DVD recording. A VideoRecorder object can be created through Library.CreateVideoRecorder ( <a href="#">see page 131</a> ).

↔ VRDevice (see page 152)	<p>This interface represents a virtual device used a storage of a DVD video recording. The underlying physical device is either a file system or an optical disc burner.</p> <p>A VRDevice object can be created with VRDevicePlugin.CreateOpticalDiscDevice (see page 162) or VRDevicePlugin.CreateFileSystemDevice (see page 162).</p> <p>A VideoRecorder (see page 140) object needs at least one VRDevice. A VRDevice object should be added to the device list of the video recorder: VideoRecorder.Devices (see page 144) The device object must be initialized successfully before it can be used for recording or reading.</p>
↔ VRDeviceList (see page 157)	<p>This interface is used to add/remove devices to the VideoRecorder (see page 140). The recorder writes the DVD content to all devices in its device list.</p> <p>The device list can be obtained via VideoRecorder.Devices (see page 144).</p>
↔ VRDevicePlugin (see page 160)	<p>This interface creates the video recording devices used by VideoRecorder (see page 140). The recorder writes the DVD video to all devices in its device list.</p> <p>A VRDevicePlugin can be created through VideoRecorder.LoadDevicePlugin (see page 149).</p>

### Legend

↔	Interface
📄	Enumeration

### Structs, Records, Enums

Struct, Record, Enum	Description
📄 VideoRecorderError (see page 163)	The VideoRecorderError type defines the possible errors returned by VideoRecorder.Error (see page 144) when the error facility is VideoRecorder (see page 140).
📄 VRDeviceError (see page 165)	The VRDeviceError type defines the possible errors returned by VRDevice.Error (see page 154) and by DeviceError.Error (see page 135) when the error facility is VRDevice (see page 152).
📄 VRDevicePluginError (see page 166)	The VRDevicePluginError type defines the possible errors returned by VRDevicePlugin.Error (see page 161).
📄 VRDeviceType (see page 166)	The VRDeviceType defines video recording storage devices.

### 5.1.1.3.1 Interfaces

The following table lists interfaces in this documentation.

#### Interfaces

Interface	Description
↔ DeviceError (see page 134)	This interface represents the error state of a single device. An instance of this interface can be obtained via VideoRecorder::GetDeviceError.
↔ FileSystemDeviceConfig (see page 136)	This interface configures an Hdd device used for DVD video recording. The object obtained via VRDevice.Config (see page 154) should be cast to this interface if the VRDevice.Type (see page 156) is VRDeviceType.FileSystem.

↔ OpticalDiscDeviceConfig (see page 137)	This interface configures a burner device used for DVD video recording. The object obtained via VRDevice.Config (see page 154) should be cast to this interface if the device type is VRDeviceType.OpticalDisc (see page 166).
↔ Title (see page 138)	This interface is used to get information about a particular DVD-Video title. An instance of this interface can be obtained via TitleEnumerator (see page 139).
↔ TitleEnumerator (see page 139)	This interface is used to enumerate all DVD-Video titles in a recording. An instance of this interface can be obtained via VideoRecorder.GetTitles (see page 149). When it's not needed anymore the instance must be disposed.
↔ VideoRecorder (see page 140)	This interface represents the DVD video recorder and is used to carry out the DVD recording. A VideoRecorder object can be created through Library.CreateVideoRecorder (see page 131).
↔ VRDevice (see page 152)	This interface represents a virtual device used a storage of a DVD video recording. The underlying physical device is either a file system or an optical disc burner. A VRDevice object can be created with VRDevicePlugin.CreateOpticalDiscDevice (see page 162) or VRDevicePlugin.CreateFileSystemDevice (see page 162). A VideoRecorder (see page 140) object needs at least one VRDevice. A VRDevice object should be added to the device list of the video recorder: VideoRecorder.Devices (see page 144) The device object must be initialized successfully before it can be used for recording or reading.
↔ VRDeviceList (see page 157)	This interface is used to add/remove devices to the VideoRecorder (see page 140). The recorder writes the DVD content to all devices in its device list. The device list can be obtained via VideoRecorder.Devices (see page 144).
↔ VRDevicePlugin (see page 160)	This interface creates the video recording devices used by VideoRecorder (see page 140). The recorder writes the DVD video to all devices in its device list. A VRDevicePlugin can be created through VideoRecorder.LoadDevicePlugin (see page 149).

**Legend**

↔	Interface
---	-----------

**5.1.1.3.1.1 DeviceError****Class Hierarchy**

PrimoSoftware::DVDBuilder::VR::DeviceError







**Visual Basic****Public Interface DeviceError****C#****public interface DeviceError;****Description**

This interface represents the error state of a single device. An instance of this interface can be obtained via VideoRecorder::GetDeviceError.



**Version**

2.0

**See Also**VideoRecorder.GetDeviceError ([see page 147](#))**Members****Properties**

Property	Description
  Error ( <a href="#">see page 135</a> )	Gets the error code of the last operation. See the Error Handling ( <a href="#">see page 4</a> ) topic for a detailed description.
  ErrorFacility ( <a href="#">see page 135</a> )	Gets the error source of the last operation. See the Error Handling ( <a href="#">see page 4</a> ) topic for a detailed description.
  SystemErrorDomain ( <a href="#">see page 136</a> )	Gets the error source of the last operation when there's a system error. See the Error Handling ( <a href="#">see page 4</a> ) topic for a detailed description.

**Legend**

	Property
	read only

**5.1.1.3.1.1.1 DeviceError.Error****Visual Basic**

```
Public ReadOnly Property Error As Integer
```

**C#**

```
public int Error {get;}
```

**Description**

Gets the error code of the last operation.

See the Error Handling ([see page 4](#)) topic for a detailed description.

**Version**

2.0

**5.1.1.3.1.1.2 DeviceError.ErrorFacility****Visual Basic**

```
Public ReadOnly Property ErrorFacility As Integer
```

**C#**

```
public int ErrorFacility {get;}
```

**Description**

Gets the error source of the last operation.

See the Error Handling ([see page 4](#)) topic for a detailed description.

**Version**

2.0

**5.1.1.3.1.1.3 DeviceError.SystemErrorDomain****Visual Basic**

```
Public ReadOnly Property SystemErrorDomain As Integer
```

**C#**

```
public int SystemErrorDomain {get;}
```

**Description**

Gets the error source of the last operation when there's a system error.

See the Error Handling (see page 4) topic for a detailed description.

**Version**

2.0

**5.1.1.3.1.2 FileSystemDeviceConfig****Class Hierarchy**

```
PrimoSoftware::DVDBuilder::VR::FileSystemDeviceConfig
```

**Visual Basic**

```
Public Interface FileSystemDeviceConfig
```

**C#**

```
public interface FileSystemDeviceConfig;
```

**Description**

This interface configures an Hdd device used for DVD video recording.

The object obtained via VRDevice.Config (see page 154) should be cast to this interface if the VRDevice.Type (see page 156) is VRDeviceType.FileSystem.



**Version**

2.0

**Members****Properties**

Property	Description
  Folder (see page 136)	Gets the output folder on the hard drive.

**Legend**

	Property
	read only

**5.1.1.3.1.2.1 FileSystemDeviceConfig.Folder****Visual Basic**

```
Public Property Folder() As String
```

**C#**

```
public string Folder {get;}
```

**Description**

Gets the output folder on the hard drive.

**Version**

2.0

**5.1.1.3.1.3 OpticalDiscDeviceConfig****Class Hierarchy**

PrimoSoftware::DVDBuilder::VR::OpticalDiscDeviceConfig

**Visual Basic**

```
Public Interface OpticalDiscDeviceConfig
```

**C#**

```
public interface OpticalDiscDeviceConfig;
```

**Description**






This interface configures a burner device used for DVD video recording.

The object obtained via VRDevice.Config (see page 154) should be cast to this interface if the device type is VRDeviceType (see page 166).



**Version**

2.0

**Members****Properties**

Property	Description
  DeviceHandle (see page 137)	Gets the OS handle of the device.
  DriveLetter (see page 138)	Gets the drive letter of a DVD burner device
 VolumeLabel (see page 138)	Sets / Gets the volume label of the DVD disc. If required the volume label should be set before calling FinalizeMedia but after the device is successfully initialized. In case the volume label is not set before finalizing the disc the label is blank. The volume label can be read after the device is initialized.

**Legend**

	Property
	read only

**5.1.1.3.1.3.1 OpticalDiscDeviceConfig.DeviceHandle****Visual Basic**

```
Public Property DeviceHandle() As IntPtr
```

**C#**

```
public IntPtr DeviceHandle {get;}
```

**Returns**

OS device handle

**Description**

Gets the OS handle of the device.

**Version**

2.0

**5.1.1.3.1.3.2 OpticalDiscDeviceConfig.DriveLetter****Visual Basic**

```
Public Property DriveLetter() As Char
```

**C#**

```
public char DriveLetter {get;}
```

**Description**

Gets the drive letter of a DVD burner device

**5.1.1.3.1.3.3 OpticalDiscDeviceConfig.VolumeLabel****Visual Basic**

```
Public Property VolumeLabel() As String
```

**C#**

```
public string VolumeLabel {get; set;}
```

**Parameters**

Parameters	Description
volumeLabel	[in] disc volume label. The maximum length allowed for the volume label is 32 characters. Only capital letters from A to Z, digits from 0 to 9, and the special characters _ (underscore) are allowed.

**Description**

Sets / Gets the volume label of the DVD disc. If required the volume label should be set before calling FinalizeMedia but after the device is successfully initialized. In case the volume label is not set before finalizing the disc the label is blank. The volume label can be read after the device is initialized.

**Version**

2.0

**5.1.1.3.1.4 Title****Class Hierarchy**

```
PrimoSoftware::DVDBuilder::VR::Title
```

**Visual Basic**

```
Public Interface Title
```

**C#**

```
public interface Title;
```

**Description**

This interface is used to get information about a particular DVD-Video title. An instance of this interface can be obtained via [TitleEnumerator](#) (see page 139).

**Version**

2.0



**See Also**

[TitleEnumerator.Item](#) (see page 140)

**Members****Properties**

Property	Description
  <a href="#">Duration</a> (see page 139)	Gets the title duration in seconds as a floating point value.

**Legend**

	Property
	read only

**5.1.1.3.1.4.1 Title.Duration****Visual Basic**

```
Public ReadOnly Property Duration As Double
```

**C#**

```
public double Duration {get;}
```

**Description**

Gets the title duration in seconds as a floating point value.

**Version**

2.0

**5.1.1.3.1.5 TitleEnumerator****Class Hierarchy**

```
PrimoSoftware::DVDBuilder::VR::TitleEnumerator
```

**Visual Basic**

```
Public Interface TitleEnumerator
Inherits IDisposable
```

**C#**

```
public interface TitleEnumerator : IDisposable;
```

**Description**

This interface is used to enumerate all DVD-Video titles in a recording. An instance of this interface can be obtained via [VideoRecorder.GetTitles](#) (see page 149). When it's not needed anymore the instance must be disposed.



**Version**

2.0


**See Also**

VideoRecorder.GetTitles (see page 149), Title (see page 138)

**Members****Properties**

Property	Description
 <b>R</b> Count (see page 140)	Gets the number of titles.
 <b>R</b> Item (see page 140)	Gets an object that describes a DVD-Video title. The lifetime of the title object is controlled by the title enumerator.

**Legend**

	Property
<b>R</b>	read only

**5.1.1.3.1.5.1 TitleEnumerator.Count****Visual Basic**

```
Public ReadOnly Property Count As Integer
```

**C#**

```
public int Count {get;}
```

**Description**

Gets the number of titles.

**Version**

2.0

**5.1.1.3.1.5.2 TitleEnumerator.Item****Visual Basic**

```
Public Default ReadOnly Property Item(ByVal index As Integer) As Title
```

**C#**

```
public Title this[int index] {get;}
```

**Description**

Gets an object that describes a DVD-Video title. The lifetime of the title object is controlled by the title enumerator.

**Version**

2.0

**5.1.1.3.1.6 VideoRecorder****Class Hierarchy**

```
PrimoSoftware::DVDBuilder::VR::VideoRecorder
```

**Visual Basic**

```
Public Interface VideoRecorder
    Inherits IDisposable
```

**C#**

```
public interface VideoRecorder : IDisposable;
```

**Description**

This interface represents the DVD video recorder and is used to carry out the DVD recording.

A VideoRecorder object can be created through Library.CreateVideoRecorder ([see page 131](#)).














**Version**



2.0

**See Also**







Library.CreateVideoRecorder ([see page 131](#))























**Members****Properties**

Property	Description
 AllowMuxBufferUnderflow ( <a href="#">see page 143</a> )	Indicates whether or not a multiplexer buffer underflow is allowed.
  AverageBitrate ( <a href="#">see page 144</a> )	Gets the average bitrate (bits per second) of the video since the start of the recording. The returned bitrate may be 0 when the first several seconds of the input video has not been processed yet by the video recorder. This should be checked explicitly when calculating the remaining recording time.
  Devices ( <a href="#">see page 144</a> )	Gets the device list of the VideoRecorder. The device list can be used to add or remove devices. The VideoRecorder needs at least one device in order to start the recording process.
  Error ( <a href="#">see page 144</a> )	Gets the error code of the last operation. See the Error Handling ( <a href="#">see page 4</a> ) topic for a detailed description.
  ErrorFacility ( <a href="#">see page 145</a> )	Gets the error source of the last operation. See the Error Handling ( <a href="#">see page 4</a> ) topic for a detailed description.
  IsFinalized ( <a href="#">see page 145</a> )	Gets information whether the video is finalized. Returns true when the video is finalized, false otherwise. If the finalization state cannot be obtained VideoRecorder.ErrorFacility ( <a href="#">see page 145</a> ) is non-zero.
  IsFinalizeSupported ( <a href="#">see page 145</a> )	Gets information whether the video supports finalization. If it is supported then the FinalizeMedia ( <a href="#">see page 146</a> ) method can be used to make discs more compatible for playing/reading. Returns true when the video supports finalization, false otherwise. If the finalization support cannot be obtained VideoRecorder.ErrorFacility ( <a href="#">see page 145</a> ) is non-zero.




 <b>MediaFreeSpace</b> ( <a href="#">see page 146</a> )	<p>Gets the usable free space (in bytes) for video recording. The returned value is the minimum free space of all attached devices. The video recorder can successfully stop the recording and finalize the video content even when the free space is zero.</p> <p>This value can be used to estimate the remaining recording time as well (in combination with <a href="#">GetAverageBitrate</a>). A good estimate of the available recording time in seconds is: <math>\text{MediaFreeSpace} * 8 / \text{AverageBitrate}</math> (<a href="#">see page 144</a>).</p> <p>During recording this method should be used to estimate the remaining recording space instead of <a href="#">VRDevice.MediaFreeSpace</a> (<a href="#">see page 155</a>).</p>
 <b>SystemErrorDomain</b> ( <a href="#">see page 146</a> )	<p>Gets the error source of the last operation when there's a system error.</p> <p>See the <a href="#">Error Handling</a> (<a href="#">see page 4</a>) topic for a detailed description.</p>

## Methods

Method	Description
 <b>FinalizeMedia</b> ( <a href="#">see page 146</a> )	<p>Finalizes VideoRecorder instance. This is required for DVD-Video so that the content can be played by a conventional DVD player or even read by the operating system. The media used by the VideoRecorder cannot be ejected manually while the finalization is in progress.</p>
 <b>GetDeviceError</b> ( <a href="#">see page 147</a> )	<p>Gets an object that can be used to query the error state of a specified device after a parallel operation of the video recorder.</p> <p>This method should be used immediately after one of the following parallel operations: <a href="#">PrepareMedia</a> (<a href="#">see page 149</a>), <a href="#">Start</a> (<a href="#">see page 150</a>), <a href="#">Write</a> (<a href="#">see page 151</a>), <a href="#">Stop</a> (<a href="#">see page 151</a>), and <a href="#">FinalizeMedia</a> (<a href="#">see page 146</a>). Only in this case the <a href="#">GetDeviceError</a> method is guaranteed to return a consistent error state for the specified device.</p>
 <b>GetIsFinalized</b> ( <a href="#">see page 147</a> )	<p>Gets information whether the video in all attached devices is finalized. If the finalized state cannot be obtained <a href="#">VideoRecorder.ErrorFacility</a> (<a href="#">see page 145</a>) is non-zero.</p>
 <b>GetMediaFreeSpace</b> ( <a href="#">see page 148</a> )	<p>Gets the usable free space (in bytes) for video recording. The returned value is the free space for the specified device. The video recorder can successfully stop the recording and finalize the video content even when the free space is zero.</p> <p>This value can be used to estimate the remaining recording time as well (in combination with <a href="#">GetAverageBitrate</a>). A good estimate of the available recording time in seconds is: <math>\text{MediaFreeSpace} * 8 / \text{AverageBitrate}</math> (<a href="#">see page 144</a>).</p> <p>During recording this method should be used to estimate the remaining recording space instead of <a href="#">VRDevice.MediaFreeSpace</a> (<a href="#">see page 155</a>).</p>
 <b>GetTitles</b> ( <a href="#">see page 149</a> )	<p>Gets an object that can be used to enumerate all DVD-Video titles in the specified device. The object returned by this method must be disposed when it's not needed anymore in order to free the occupied memory.</p>
 <b>LoadDevicePlugin</b> ( <a href="#">see page 149</a> )	<p>The <a href="#">LoadDevicePlugin</a> method creates a device plugin object.</p>

 PrepareMedia (  see page 149)	Prepares the media in the attached devices for writing. It is an integral part of the Start (  see page 150) and StartAsync (  see page 150) methods. It is not necessary to call PrepareMedia in order to start the recording. However the Start (  see page 150) method may take much less time if PrepareMedia has been called beforehand.
 Start (  see page 150)	Starts the recording process. All VideoRecorder.Devices (  see page 144) must be initialized before this method is called. After the recording has been started data must be passed to the recorder using the Write (  see page 151) method. When the Start method returns successfully the media in VideoRecorder.Devices (  see page 144) are locked and cannot be ejected manually. The media are unlocked when the recording is stopped.
 StartAsync (  see page 150)	Starts the recording process asynchronously. It is identical to VideoRecorder.Start (  see page 150) except that StartAsync has a non-blocking behaviour. The video recorder starts a background task to prepare the media in all attached devices for writing. This is a potentially lengthy operation (especially for DVD+R discs) and StartAsync does not wait for it to finish. A VideoRecorder object is ready to accept data through the Write (  see page 151) method as soon as StartSync returns even though the actual recording process is postponed after all media is prepared. If the video recorder cannot accept more data it will set the VideoRecorderError::StartAsyncBufferFull error when data is... more (  see page 150)
 Stop (  see page 151)	Stops the recording process. All data buffered in the VideoRecorder is flushed to VideoRecorder.Devices (  see page 144). This method is synchronous and it may take a while (approx. 20 sec.) before the method returns. All media used in the recording are unlocked after the method returns successfully.
 Write (  see page 151)	Passes data to the VideoRecorder to be written to all devices. This method must be called continuously after Start (  see page 150) has been called successfully and as long as there is available data to be written. If there is no more data Stop (  see page 151) must be called.

**Legend**

	Property
	read only
	Method

**5.1.1.3.1.6.1 VideoRecorder.AllowMuxBufferUnderflow****Visual Basic**

```
Public Property AllowMuxBufferUnderflow As Boolean
```

**C#**

```
public bool AllowMuxBufferUnderflow {get; set;}
```

**Description**

Indicates whether or not a multiplexer buffer underflow is allowed.

**Version**

2.0.5

#### 5.1.1.3.1.6.2 VideoRecorder.AverageBitrate

##### Visual Basic

```
Public ReadOnly Property AverageBitrate As Integer
```

##### C#

```
public int AverageBitrate {get;}
```

##### Description

Gets the average bitrate (bits per second) of the video since the start of the recording. The returned bitrate may be 0 when the first several seconds of the input video has not been processed yet by the video recorder. This should be checked explicitly when calculating the remaining recording time.

##### Version

2.0

##### See Also

[VideoRecorder.GetMediaFreeSpace](#) (see page 148), [VideoRecorder.MediaFreeSpace](#) (see page 146), [VideoRecorder.Start](#) (see page 150), [VideoRecorder.Stop](#) (see page 151)

#### 5.1.1.3.1.6.3 VideoRecorder.Devices

##### Visual Basic

```
Public ReadOnly Property Devices As VRDeviceList
```

##### C#

```
public VRDeviceList Devices {get;}
```

##### Description

Gets the device list of the [VideoRecorder](#) (see page 140). The device list can be used to add or remove devices. The [VideoRecorder](#) (see page 140) needs at least one device in order to start the recording process.

##### Version

2.0

#### 5.1.1.3.1.6.4 VideoRecorder.Error

##### Visual Basic

```
Public ReadOnly Property Error As Integer
```

##### C#

```
public int Error {get;}
```

##### Description

Gets the error code of the last operation.

See the [Error Handling](#) (see page 4) topic for a detailed description.

##### Version

2.0

#### 5.1.1.3.1.6.5 VideoRecorder.ErrorFacility

##### Visual Basic

```
Public ReadOnly Property ErrorFacility As Integer
```

##### C#

```
public int ErrorFacility {get;}
```

##### Description

Gets the error source of the last operation.

See the Error Handling (see page 4) topic for a detailed description.

##### Version

2.0

#### 5.1.1.3.1.6.6 VideoRecorder.IsFinalized

##### Visual Basic

```
Public ReadOnly Property IsFinalized As Boolean
```

##### C#

```
public bool IsFinalized {get;}
```

##### Description

Gets information whether the video is finalized.

Returns true when the video is finalized, false otherwise. If the finalization state cannot be obtained VideoRecorder.ErrorFacility (see page 145) is non-zero.

##### Version

2.0

##### See Also

VideoRecorder.FinalizeMedia (see page 146), VideoRecorder.IsFinalizeSupported (see page 145)

#### 5.1.1.3.1.6.7 VideoRecorder.IsFinalizeSupported

##### Visual Basic

```
Public ReadOnly Property IsFinalizeSupported As Boolean
```

##### C#

```
public bool IsFinalizeSupported {get;}
```

##### Description

Gets information whether the video supports finalization. If it is supported then the VideoRecorder.FinalizeMedia (see page 146) method can be used to make discs more compatible for playing/reading.

Returns true when the video supports finalization, false otherwise. If the finalization support cannot be obtained VideoRecorder.ErrorFacility (see page 145) is non-zero.

##### Version

2.0

**See Also**

[VideoRecorder.FinalizeMedia](#) (see page 146), [VideoRecorder.IsFinalized](#) (see page 145)

**5.1.1.3.1.6.8 VideoRecorder.MediaFreeSpace****Visual Basic**

```
Public ReadOnly Property MediaFreeSpace As Int64
```

**C#**

```
public long MediaFreeSpace {get;}
```

**Description**

Gets the usable free space (in bytes) for video recording. The returned value is the minimum free space of all attached devices. The video recorder can successfully stop the recording and finalize the video content even when the free space is zero.

This value can be used to estimate the remaining recording time as well (in combination with [GetAverageBitrate](#)). A good estimate of the available recording time in seconds is:  $\text{MediaFreeSpace} * 8 / \text{VideoRecorder.AverageBitrate}$  (see page 144).

During recording this method should be used to estimate the remaining recording space instead of [VRDevice.MediaFreeSpace](#) (see page 155).

**Version**

2.0

**See Also**

[VideoRecorder.AverageBitrate](#) (see page 144), [VideoRecorder.Stop](#) (see page 151), [VideoRecorder.FinalizeMedia](#) (see page 146), [VideoRecorder.GetMediaFreeSpace](#) (see page 148), [VRDevice.MediaFreeSpace](#) (see page 155)

**5.1.1.3.1.6.9 VideoRecorder.SystemErrorDomain****Visual Basic**

```
Public ReadOnly Property SystemErrorDomain As Integer
```

**C#**

```
public int SystemErrorDomain {get;}
```

**Description**

Gets the error source of the last operation when there's a system error.

See the [Error Handling](#) (see page 4) topic for a detailed description.

**Version**

2.0

**5.1.1.3.1.6.10 VideoRecorder.FinalizeMedia****Visual Basic**

```
Public Function FinalizeMedia() As Boolean
```

**C#**

```
public bool FinalizeMedia();
```

**Returns**

true. The video is finalized successfully.  
false. The video could not be finalized.

**Description**

Finalizes VideoRecorder (see page 140) instance. This is required for DVD-Video so that the content can be played by a conventional DVD player or even read by the operating system. The media used by the VideoRecorder (see page 140) cannot be ejected manually while the finalization is in progress.

**Version**

2.0

**See Also**

VideoRecorder.IsFinalizeSupported (see page 145), VideoRecorder.IsFinalized (see page 145)

**5.1.1.3.1.6.11 VideoRecorder.GetDeviceError****Visual Basic**

```
Public Function GetDeviceError(deviceIndex as Integer) As DeviceError
```

**C#**

```
public DeviceError GetDeviceError(int deviceIndex);
```

**Parameters**

Parameters	Description
deviceIndex	[in] A zero-based index of the device for which the error state is requested.

**Returns**

An object that contains information about the error state of a video recording device or null if the error state is not available.

**Description**

Gets an object that can be used to query the error state of a specified device after a parallel operation of the video recorder.

This method should be used immediately after one of the following parallel operations: VideoRecorder.PrepareMedia (see page 149), VideoRecorder.Start (see page 150), VideoRecorder.Write (see page 151), VideoRecorder.Stop (see page 151), and VideoRecorder.FinalizeMedia (see page 146). Only in this case the GetDeviceError method is guaranteed to return a consistent error state for the specified device.

**Version**

2.0

**5.1.1.3.1.6.12 VideoRecorder.GetIsFinalized****Visual Basic**

```
Public Function GetIsFinalized(deviceIndex As Integer) As Boolean
```

**C#**

```
public bool GetIsFinalized(int deviceIndex);
```

**Parameters**

Parameters	Description
deviceIndex	[in] The index of the device that will be checked for a finalized video content.

**Returns**

The return value is true when the video is finalized and false when it's not finalized.

**Description**

Gets information whether the video in all attached devices is finalized. If the finalized state cannot be obtained VideoRecorder.ErrorFacility (see page 145) is non-zero.

**Version**

2.0

**See Also**

VideoRecorder.FinalizeMedia (see page 146), VideoRecorder.IsFinalizeSupported (see page 145), VideoRecorder.ErrorFacility (see page 145)

**5.1.1.3.1.6.13 VideoRecorder.GetMediaFreeSpace****Visual Basic**

```
Public Function GetMediaFreeSpace(deviceIndex as Integer) As Int64
```

**C#**

```
public long GetMediaFreeSpace(int deviceIndex);
```

**Parameters**

Parameters	Description
deviceIndex	[in] A zero-based index in the video recorder device list. Specifies the device for which the remaining free space is requested.

**Returns**

The remaining free space (in bytes) that can be used for video recording.

**Description**

Gets the usable free space (in bytes) for video recording. The returned value is the free space for the specified device. The video recorder can successfully stop the recording and finalize the video content even when the free space is zero.

This value can be used to estimate the remaining recording time as well (in combination with GetAverageBitrate). A good estimate of the available recording time in seconds is:  $\text{VideoRecorder.MediaFreeSpace} \times 8 / \text{VideoRecorder.AverageBitrate}$  (see page 146).

During recording this method should be used to estimate the remaining recording space instead of VRDevice.MediaFreeSpace (see page 155).

**Version**

2.0

**See Also**

VideoRecorder.AverageBitrate (see page 144), VideoRecorder.Stop (see page 151), VideoRecorder.FinalizeMedia (see page 146), VideoRecorder.MediaFreeSpace (see page 146), VRDevice.MediaFreeSpace (see page 155)

**5.1.1.3.1.6.14 VideoRecorder.GetTitles****Visual Basic**

```
Public Function GetTitles(deviceIndex as Integer) As TitleEnumerator
```

**C#**

```
public TitleEnumerator GetTitles(int deviceIndex);
```

**Parameters**

Parameters	Description
deviceIndex	[in] A zero-based index in the video recorder device list. Specifies the device for which the DVD-Video titles will be retrieved. The device must be initialized before calling this method.

**Returns**

An object that contains information about all titles on the specified disc.

**Description**

Gets an object that can be used to enumerate all DVD-Video titles in the specified device. The object returned by this method must be disposed when it's not needed anymore in order to free the occupied memory.

**Version**

2.0

**5.1.1.3.1.6.15 VideoRecorder.LoadDevicePlugin****Visual Basic**

```
Public Shared Function LoadDevicePlugin() As VRDevicePlugin
```

**C#**

```
public static VRDevicePlugin LoadDevicePlugin();
```

**Parameters**

Parameters	Description
path	[in] A file system path to the plugin library.

**Description**

The LoadDevicePlugin method creates a device plugin object.

**Version**

2.0

**5.1.1.3.1.6.16 VideoRecorder.PrepareMedia****Visual Basic**

```
Function PrepareMedia() As bool
```

**C#**

```
bool PrepareMedia();
```

**Returns**

The return value is 1 when the media in all attached devices have been prepared successfully.

The return value is 0 when the media preparation has failed.

**Description**

Prepares the media in the attached devices for writing. It is an integral part of the `VideoRecorder.Start` (see page 150) and `VideoRecorder.StartAsync` (see page 150) methods. It is not necessary to call `PrepareMedia` in order to start the recording. However the `VideoRecorder.Start` (see page 150) method may take much less time if `PrepareMedia` has been called beforehand.

**Version**

2.0

**See Also**

`VideoRecorder.Start` (see page 150)

**5.1.1.3.1.6.17 VideoRecorder.Start****Visual Basic**

```
Public Function Start() As Boolean
```

**C#**

```
public bool Start();
```

**Returns**

true. The recording process started successfully. The media used in the recording are locked and cannot be ejected manually.

false. The recording process failed to start.

**Description**

Starts the recording process. All `VideoRecorder.Devices` (see page 144) must be initialized before this method is called. After the recording has been started data must be passed to the recorder using the `VideoRecorder.Write` (see page 151) method. When the `Start` method returns successfully the media in `VideoRecorder.Devices` (see page 144) are locked and cannot be ejected manually. The media are unlocked when the recording is stopped.

**Version**

2.0

**See Also**

`VideoRecorder.Write` (see page 151), `VideoRecorder.Stop` (see page 151)

**5.1.1.3.1.6.18 VideoRecorder.StartAsync****Visual Basic**

```
Function StartAsync() As bool
```

**C#**

```
bool StartAsync();
```

**Returns**

The return value is 1 when the recording process has started successfully. The media used in the recording are locked and cannot be ejected manually.

The return value is 0 when the the recording process has failed to start.

**Description**

Starts the recording process asynchronously. It is identical to `VideoRecorder.Start` (see page 150) except that `StartAsync` has a non-blocking behaviour. The video recorder starts a background task to prepare the media in all attached devices for writing. This is a potentially lengthy operation (especially for DVD+R discs) and `StartAsync` does not wait for it to finish. A `VideoRecorder` (see page 140) object is ready to accept data through the `VideoRecorder.Write` (see page 151) method as soon as `StartSync` returns even though the actual recording process is postponed after all media is prepared. If the video recorder cannot accept more data it will set the `VideoRecorderError::StartAsyncBufferFull` error when data is passed through the `VideoRecorder.Write` (see page 151) method. Then the `VideoRecorder.Stop` (see page 151) method should be called to stop the recording. The video recorder is designed to accept at least 2 minutes of MPEG-2 video at max birate.

**Version**

2.0

**See Also**

`VideoRecorder.Write` (see page 151), `VideoRecorder.Stop` (see page 151), `VideoRecorder.Start` (see page 150)

**5.1.1.3.1.6.19 VideoRecorder.Stop****Visual Basic**

```
Public Function Stop() As Boolean
```

**C#**

```
public bool Stop();
```

**Returns**

true. The recording has been stopped successfully. The media used in the recording are unlocked and can be ejected manually.

false. An error occurred while trying to stop the recording.

**Description**

Stops the recording process. All data buffered in the `VideoRecorder` (see page 140) is flushed to `VideoRecorder.Devices` (see page 144). This method is synchronous and it may take a while (approx. 20 sec.) before the method returns. All media used in the recording are unlocked after the method returns successfully.

**Version**

2.0

**See Also**

`VideoRecorder.Start` (see page 150)

**5.1.1.3.1.6.20 VideoRecorder.Write****Visual Basic**

```
Public Function Write(dataPtr As IntPtr, dataSize As Integer) As Boolean  
Public Function Write(data As Byte()) As Boolean
```

**C#**

```
public bool Write(IntPtr dataPtr, int dataSize);
public bool Write(byte[] data);
```

**Parameters**

Parameters	Description
dataPtr	[in] A pointer to data buffer that has to be written. The data is in the unmanaged memory.
dataSize	[in] The size (in bytes) of the data buffer specified by the dataPtr parameter
data	[in] The data buffer that has to be written. The data is in the managed memory.

**Returns**

true. The data has been accepted by the recorder.

false. The recorder has failed to accept the data or one or more of the attached devices have failed. Use the `VideoRecorder.GetDeviceError` (see page 147) method to inspect the error state of the attached device when the `Write` method returns false. If all devices have failed the recording process has to be stopped using the `VideoRecorder.Stop` (see page 151) method and no more data must be passed to the recorder by calling the `Write` method. The recording may continue as long as there's at least one active (non-failed) device.

**Description**

Passes data to the `VideoRecorder` (see page 140) to be written to all devices. This method must be called continuously after `VideoRecorder.Start` (see page 150) has been called successfully and as long as there is available data to be written. If there is no more data `VideoRecorder.Stop` (see page 151) must be called.

**Version**

2.0

**See Also**

`VideoRecorder.Start` (see page 150), `VideoRecorder.Stop` (see page 151)

**5.1.1.3.1.7 VRDevice****Class Hierarchy**

PrimoSoftware::DVDBuilder::VR::VRDevice

**Visual Basic**

```
Public Interface VRDevice
Inherits IDisposable
```

**C#**

```
public interface VRDevice : IDisposable;
```

**Description**

This interface represents a virtual device used a storage of a DVD video recording. The underlying physical device is either a file system or an optical disc burner.

















A `VRDevice` object can be created with `VRDevicePlugin.CreateOpticalDiscDevice` (see page 162) or `VRDevicePlugin.CreateFileSystemDevice` (see page 162).

A `VideoRecorder` (see page 140) object needs at least one `VRDevice`. A `VRDevice` object should be added to the device list of the video recorder: `VideoRecorder.Devices` (see page 144) The device object must be initialized successfully before it can be used for recording or reading.




**Version**

2.0



**Members****Properties**

Property	Description
  Config ( <a href="#">see page 154</a> )	Returns a device configuration object. It should be cast to the appropriate device configuration interface depending on the device type.
  Error ( <a href="#">see page 154</a> )	Gets the error code of the last operation. See the Error Handling ( <a href="#">see page 4</a> ) topic for a detailed description.
  ErrorFacility ( <a href="#">see page 154</a> )	Gets the error source of the last operation. See the Error Handling ( <a href="#">see page 4</a> ) topic for a detailed description.
  MediaFreeSpace ( <a href="#">see page 155</a> )	Gets the remaining free space in bytes. If the free space cannot be obtained VRDevice.ErrorFacility ( <a href="#">see page 154</a> ) is non-zero.
  MedialsBlank ( <a href="#">see page 155</a> )	Gets whether the disc is blank. Returns true when the disc is blank, false otherwise. If the blank state cannot be obtained VRDevice.ErrorFacility ( <a href="#">see page 154</a> ) is non-zero. In this case the return value should be ignored.
  MedialsReWritable ( <a href="#">see page 155</a> )	Gets whether the disc is rewritable. Returns true when the disc is rewritable (DVD-RW, DVD+RW, DVD-RAM), false otherwise (DVD-R, DVD+R). If the rewritable property cannot be obtained VRDevice.ErrorFacility ( <a href="#">see page 154</a> ) is non-zero. In this case the return value should be ignored.
  SystemErrorDomain ( <a href="#">see page 156</a> )	Gets the error source of the last operation when there's a system error. See the Error Handling ( <a href="#">see page 4</a> ) topic for a detailed description.
  Type ( <a href="#">see page 156</a> )	Gets the device type of a video recording device.

**Methods**

Method	Description
 EraseMedia ( <a href="#">see page 156</a> )	Erases a rewritable disc. When a disc is erased it becomes blank. The disc cannot be ejected manually while it is being erased.
 Initialize ( <a href="#">see page 157</a> )	Initializes VRDevice. This must be done before the device object can be used for recording or reading existing data. Video Recorder should start the recording process (VideoRecorder.Start ( <a href="#">see page 150</a> )) only when all device objects in its device list are initialized successfully. The device object may be initialized before or after it is added to the device list.
 NotifyOSFileSystemChanged ( <a href="#">see page 157</a> )	Notifies the operating system that the file contents have changed. This is necessary to see the updated files in the file browser.

**Legend**

	Property
	read only



#### 5.1.1.3.1.7.1 VRDevice.Config

##### Visual Basic

```
Public ReadOnly Property Config As Object
```

##### C#

```
public object Config {get;}
```

##### Returns

Storage configuration object

##### Description

Returns a device configuration object. It should be cast to the appropriate device configuration interface depending on the device type.

##### Version

2.0

#### 5.1.1.3.1.7.2 VRDevice.Error

##### Visual Basic

```
Public ReadOnly Property Error As Integer
```

##### C#

```
public int Error {get;}
```

##### Description

Gets the error code of the last operation.

See the Error Handling (see page 4) topic for a detailed description.

##### Version

2.0

#### 5.1.1.3.1.7.3 VRDevice.ErrorFacility

##### Visual Basic

```
Public ReadOnly Property ErrorFacility As Integer
```

##### C#

```
public int ErrorFacility {get;}
```

##### Description

Gets the error source of the last operation.

See the Error Handling (see page 4) topic for a detailed description.

##### Version

2.0

#### 5.1.1.3.1.7.4 VRDevice.MediaFreeSpace

##### Visual Basic

```
Public ReadOnly Property MediaFreeSpace As Long
```

##### C#

```
public long MediaFreeSpace {get;}
```

##### Description

Gets the remaining free space in bytes. If the free space cannot be obtained VRDevice.ErrorFacility (see page 154) is non-zero.

##### Version

2.0

#### 5.1.1.3.1.7.5 VRDevice.MediaIsBlank

##### Visual Basic

```
Public ReadOnly Property MediaIsBlank As Boolean
```

##### C#

```
public bool MediaIsBlank {get;}
```

##### Description

Gets whether the disc is blank.

Returns true when the disc is blank, false otherwise. If the blank state cannot be obtained VRDevice.ErrorFacility (see page 154) is non-zero. In this case the return value should be ignored.

##### Version

2.0

##### See Also

VRDevice.EraseMedia (see page 156)

#### 5.1.1.3.1.7.6 VRDevice.MediaIsRewritable

##### Visual Basic

```
Public ReadOnly Property MediaIsRewritable As Boolean
```

##### C#

```
public bool MediaIsRewritable {get;}
```

##### Description

Gets whether the disc is rewritable.

Returns true when the disc is rewritable (DVD-RW, DVD+RW, DVD-RAM), false otherwise (DVD-R, DVD+R). If the rewritable property cannot be obtained VRDevice.ErrorFacility (see page 154) is non-zero. In this case the return value should be ignored.

##### Version

2.0

#### 5.1.1.3.1.7.7 VRDevice.SystemErrorDomain

##### Visual Basic

```
Public ReadOnly Property SystemErrorDomain As Integer
```

##### C#

```
public int SystemErrorDomain {get;}
```

##### Description

Gets the error source of the last operation when there's a system error.

See the Error Handling (see page 4) topic for a detailed description.

##### Version

2.0

#### 5.1.1.3.1.7.8 VRDevice.Type

##### Visual Basic

```
Public ReadOnly Property Type As VRDeviceType
```

##### C#

```
public VRDeviceType Type {get;}
```

##### Description

Gets the device type of a video recording device.

##### Version

2.0

#### 5.1.1.3.1.7.9 VRDevice.EraseMedia

##### Visual Basic

```
Public Function EraseMedia() As Boolean
```

##### C#

```
public bool EraseMedia();
```

##### Returns

true. The disc is erased successfully.

false. The disc cannot be erased. Non rewritable disc (DVD+R, DVD-R) cannot be erased.

##### Description

Erases a rewritable disc. When a disc is erased it becomes blank. The disc cannot be ejected manually while it is being erased.

##### Version

2.0

##### See Also

VRDevice.MedialsBlank (see page 155), VRDevice.MedialsReWritable (see page 155)

#### 5.1.1.3.1.7.10 VRDevice.Initialize

##### Visual Basic

```
Public Function Initialize() As Boolean
```

##### C#

```
public bool Initialize();
```

##### Returns

true. The VRDevice (see page 152) is successfully initialized and can be used.

false. The VRDevice (see page 152) cannot be used.

##### Description

Initializes VRDevice (see page 152). This must be done before the device object can be used for recording or reading existing data.

Video Recorder should start the recording process (VideoRecorder.Start (see page 150)) only when all device objects in its device list are initialized successfully. The device object may be initialized before or after it is added to the device list.

##### Version

2.0

#### 5.1.1.3.1.7.11 VRDevice.NotifyOSFileSystemChanged

##### Visual Basic

```
Public Function NotifyOSFileSystemChanged() As Boolean
```

##### C#

```
public bool NotifyOSFileSystemChanged();
```

##### Returns

true. The operation has succeeded.

false. The operation has failed.

##### Description

Notifies the operating system that the file contents have changed. This is necessary to see the updated files in the file browser.

##### Version

2.0

#### 5.1.1.3.1.8 VRDeviceList

##### Class Hierarchy

```
PrimoSoftware::DVDBuilder::VR::VRDeviceList
```

##### Visual Basic

```
Public Interface VRDeviceList
```

##### C#

```
public interface VRDeviceList;
```

##### Description

This interface is used to add/remove devices to the VideoRecorder (see page 140). The recorder writes the DVD content

to all devices in its device list.

The device list can be obtained via `VideoRecorder.Devices` (see page 144).

### Version



2.0

### See Also




`VideoRecorder.Devices` (see page 144), `VRDevicePlugin.CreateOpticalDiscDevice` (see page 162), `VRDevicePlugin.CreateFileSystemDevice` (see page 162)

### Members




#### Properties

Property	Description
 <b>R</b> <code>Count</code> (see page 158)	Gets the number of items in the list.
 <code>Item</code> (see page 158)	Gets/sets an item with a given index.

#### Methods

Method	Description
 <code>Add</code> (see page 159)	Adds a <code>VRDevice</code> (see page 152) object to the list.
 <code>Clear</code> (see page 159)	Removes all items from the list. The device objects are not automatically disposed.
 <code>RemoveAt</code> (see page 159)	Removes an item with the specified index from the list. The removed device is not automatically disposed.

### Legend

	Property
	read only
	Method

#### 5.1.1.3.1.8.1 VRDeviceList.Count

##### Visual Basic

```
Public ReadOnly Property Count As Integer
```

##### C#

```
public int Count {get;}
```

##### Description

Gets the number of items in the list.

##### Version

2.0

#### 5.1.1.3.1.8.2 VRDeviceList.Item

##### Visual Basic

```
Public Default Property Item(index As Integer) As VRDevice
```

##### C#

```
public VRDevice this[int index] {get; set;}
```

**Parameters**

Parameters	Description
index	[in] The zero based index of an item.
item	[in] The item that should be set at the specified position in the device list.

**Description**

Gets/sets an item with a given index.

**Version**

2.0

**5.1.1.3.1.8.3 VRDeviceList.Add****Visual Basic**

```
Public Sub Add(item As VRDevice)
```

**C#**

```
public void Add(VRDevice item);
```

**Parameters**

Parameters	Description
item	[in] A VRDevice (see page 152) object

**Description**

Adds a VRDevice (see page 152) object to the list.

**Version**

2.0

**5.1.1.3.1.8.4 VRDeviceList.Clear****Visual Basic**

```
Public Sub Clear()
```

**C#**

```
public void Clear();
```

**Description**

Removes all items from the list.

The device objects are not automatically disposed.

**Version**

2.0

**5.1.1.3.1.8.5 VRDeviceList.RemoveAt****Visual Basic**

```
Public Sub RemoveAt(item As Integer)
```

**C#**

```
public void RemoveAt(int item);
```

**Parameters**

Parameters	Description
index	[in] The index of the item to be removed from the list.

**Description**

Removes an item with the specified index from the list. The removed device is not automatically disposed.

**Version**

2.0

**5.1.1.3.1.9 VRDevicePlugin****Class Hierarchy**

PrimoSoftware::DVDBuilder::VR::VRDevicePlugin

**Visual Basic**

```
Public Interface VRDevicePlugin
Inherits IDisposable
```

**C#**

```
public interface VRDevicePlugin : IDisposable;
```

**Description**

This interface creates the video recording devices used by VideoRecorder ([see page 140](#)). The recorder writes the DVD video to all devices in its device list.

A VRDevicePlugin can be created through VideoRecorder.LoadDevicePlugin ([see page 149](#)).




**Version**

2.0




**See Also**

VideoRecorder.Devices ([see page 144](#)), VideoRecorder.LoadDevicePlugin ([see page 149](#))




**Members****Properties**

Property	Description
 <b>Error</b> ( <a href="#">see page 161</a> )	Gets the error code of the last operation. See the Error Handling ( <a href="#">see page 4</a> ) topic for a detailed description.
 <b>ErrorFacility</b> ( <a href="#">see page 161</a> )	Gets the error source of the last operation. See the Error Handling ( <a href="#">see page 4</a> ) topic for a detailed description.
 <b>SystemErrorDomain</b> ( <a href="#">see page 161</a> )	Gets the error source of the last operation when there's a system error. See the Error Handling ( <a href="#">see page 4</a> ) topic for a detailed description.

**Methods**

Method	Description
 CreateFileSystemDevice (see page 162)	Creates a File System device that can be added to the video recorder device list. The device is created from a file system path.
 CreateOpticalDiscDevice (see page 162)	Creates an Optical Disc device that can be added to the video recorder device list. The device is created either from a drive letter or from an OS handle.
 GetAPIRevision (see page 163)	Returns the API revision (version) of the DVD VR (see page 132) device plugin The high order byte is the major version and the low order byte is the minor version of the API.

**Legend**

	Property
	read only
	Method

**5.1.1.3.1.9.1 VRDevicePlugin.Error****Visual Basic**

```
Public ReadOnly Property Error As Integer
```

**C#**

```
public int Error {get;}
```

**Description**

Gets the error code of the last operation.

See the Error Handling (see page 4) topic for a detailed description.

**Version**

2.0

**5.1.1.3.1.9.2 VRDevicePlugin.ErrorFacility****Visual Basic**

```
Public ReadOnly Property ErrorFacility As Integer
```

**C#**

```
public int ErrorFacility {get;}
```

**Description**

Gets the error source of the last operation.

See the Error Handling (see page 4) topic for a detailed description.

**Version**

2.0

**5.1.1.3.1.9.3 VRDevicePlugin.SystemErrorDomain****Visual Basic**

```
Public ReadOnly Property SystemErrorDomain As Integer
```

**C#**

```
public int SystemErrorDomain {get;}
```

**Description**

Gets the error source of the last operation when there's a system error.

See the Error Handling ([see page 4](#)) topic for a detailed description.

**Version**

2.0

**5.1.1.3.1.9.4 VRDevicePlugin.CreateFileSystemDevice****Visual Basic**

```
Public Function CreateFileSystemDevice(path As String) As VRDevice
```

**C#**

```
public VRDevice CreateFileSystemDevice(string path);
```

**Parameters**

Parameters	Description
path	[in] File system path.

**Returns**

An instance of a video recording device.

**Description**

Creates a File System device that can be added to the video recorder device list. The device is created from a file system path.

**Version**

2.0

**See Also**

VRDeviceType ([see page 166](#)), VRDevice ([see page 152](#)), VideoRecorder.Devices ([see page 144](#))

**5.1.1.3.1.9.5 VRDevicePlugin.CreateOpticalDiscDevice****Visual Basic**

```
Public Function CreateOpticalDiscDevice(letter As Char) As VRDevice
Public Function CreateOpticalDiscDevice(handle As IntPtr) As VRDevice
```

**C#**

```
public VRDevice CreateOpticalDiscDevice(char letter);
public VRDevice CreateOpticalDiscDevice(IntPtr handle);
```

**Parameters**

Parameters	Description
letter	[in] Drive letter.
handle	[in] OS device handle.

**Returns**

An instance of video recording device.

**Description**

Creates an Optical Disc device that can be added to the video recorder device list. The device is created either from a drive letter or from an OS handle.

**Version**

2.0

**See Also**

VRDeviceType ([see page 166](#)), VRDevice ([see page 152](#)), VideoRecorder.Devices ([see page 144](#))

**5.1.1.3.1.9.6 VRDevicePlugin.GetAPIRevision****Visual Basic**

```
Public Function GetAPIRevision() As Short
```

**C#**

```
public short GetAPIRevision();
```

**Description**

Returns the API revision (version) of the DVD PrimoSoftware::DVDBuilder::VR Namespace ([see page 132](#)) device plugin. The high order byte is the major version and the low order byte is the minor version of the API.





**Version**

2.0


**5.1.1.3.2 Structs, Records, Enums**

The following table lists structs, records, enums in this documentation.

**Enumerations**

Enumeration	Description
 VideoRecorderError ( <a href="#">see page 163</a> )	The VideoRecorderError type defines the possible errors returned by VideoRecorder.Error ( <a href="#">see page 144</a> ) when the error facility is VideoRecorder ( <a href="#">see page 140</a> ).
 VRDeviceError ( <a href="#">see page 165</a> )	The VRDeviceError type defines the possible errors returned by VRDevice.Error ( <a href="#">see page 154</a> ) and by DeviceError.Error ( <a href="#">see page 135</a> ) when the error facility is VRDevice ( <a href="#">see page 152</a> ).
 VRDevicePluginError ( <a href="#">see page 166</a> )	The VRDevicePluginError type defines the possible errors returned by VRDevicePlugin.Error ( <a href="#">see page 161</a> ).
 VRDeviceType ( <a href="#">see page 166</a> )	The VRDeviceType defines video recording storage devices.

**Legend**

	Enumeration
---	-------------

**5.1.1.3.2.1 VideoRecorderError****Visual Basic**

```
Public Enum VideoRecorderError
```

**C#**

```
public enum VideoRecorderError
```

**Members**

Members	Description
Success	Success
InvalidStructure	Invalid DVD structure found in the IFO file on the disc.
RecordingNotStarted	The recorder must be started in order to complete the required operation.
RecordingAlreadyStarted	The recorder must be stopped in order to complete the required operation.
MultiplexerError	The input MPEG-2 stream cannot be re-multiplexed.
InvalidVideoStreamCount	The input stream must contain one video stream.
InvalidVideoCodec	The input stream must be encoded in MPEG-2 format.
InvalidVideoAspectRatio	The input video stream has invalid aspect ratio. The valid aspect ratios are 4:3   16:9
InvalidVideoResolution	The input video stream has invalid resolution. The valid video resolutions are:  NTSC: 720x480   704x480   352x480   352x240  PAL: 720x576   704x576   352x576   352x288
InvalidVideoFrameRate	The input video stream has invalid frame rate. The valid frame rates are 29.97 fsp (NTSC) and 25 fps (PAL).
InvalidAudioStreamCount	The input MPEG-2 stream contains more than 8 audio streams.
InvalidAudioSamplingFrequency	One or more input audio streams have invalid sampling frequency. It must be either 48KHz or 96KHz.
InvalidAudioCodec	One or more input audio streams are not in the expected format. The valid audio codec are: LPCM (16, 20, 24 bit), MPEG-1 Audio and AC-3 (Dolby Digital), DTS (Digital Theater System).
InvalidAudioBitrate	One or more input audio streams have invalid bitrate. The allowed max bitrates are: LPCM: 6144 kbit/s MPEG-1 Audio: 384 kbit/s AC-3 (Dolby Digital): 448 kbit/s DTS (Digital Theater System): 1536 kbit/s

DeviceError	<p>One or more of the devices used by the video recorder have failed after calling one of the following methods: VideoRecorder.PrepareMedia (see page 149), VideoRecorder.Start (see page 150), VideoRecorder.Write (see page 150), VideoRecorder.Stop (see page 150), and VideoRecorder.FinalizeMedia (see page 150).</p> <p>In order to get a more detailed information enumerate the device error state using VideoRecorder.GetDeviceError (see page 147).</p> <p>The device error state is reset upon calling one of the following methods: VideoRecorder.PrepareMedia (see page 149), VideoRecorder.Start (see page 150) and VideoRecorder.FinalizeMedia (see page 146).</p> <p>When VideoRecorder.Write (see page 151) is called any of the attached devices may fail and the error state is preserved until the Stop method is called. The failed devices are not used till the end of the recording. When the application gets a device error after calling VideoRecorder.Write (see page 151) it may either stop the recording process or it may continue to write to the video recorder if at least one of the devices has a clean error state (DeviceError.ErrorFacility (see page 135) is ErrorFacility.Success (see page 177)).</p> <p>When VideoRecorder.Stop (see page 151) is called the video recorder tries to stop the devices that have a clean error state and those that have run out of space. Their error state is updated. The other devices are keep their error state.</p> <p>See Error Handling (see page 4)</p>
DeviceNotSet	There's no device attached to the video recorder.
StartAsyncBufferFull	The recording process has been started asynchronously using the StartAsync method and the video recorder cannot accept more data through the Write method. Usually this is a sign that the media cannot be prepared for writing within certain time and the Stop method must be called to stop the recording. All data buffered in the video recorder is lost.
InternalError	Unexpected error in Video Recorder.
InvalidDeviceIndex	The device index is not in the allowed range. The specified index is not a valid index either in the device list or in the device error state list.

**Description**

The VideoRecorderError type defines the possible errors returned by VideoRecorder.Error (see page 144) when the error facility is VideoRecorder (see page 140).

**Version**

2.0

**5.1.1.3.2.2 VRDeviceError****Visual Basic**

```
Public Enum VRDeviceError
```

**C#**

```
public enum VRDeviceError
```

**Members**

Members	Description
Success	No error. The operation is successful.
IncompatibleMedia	The device can neither read from nor write to the disc. A DVD disc is required.
MediaNotPresent	There is no disc in the device.
FileSystemError	Cannot recognize the existing file system on the disc. The disc can be used after being erased.
NotInitialized	In order to complete the required operation the device must be initialized first.
OutOfFreeSpace	There's no free space on the disc.

**Description**

The VRDeviceError type defines the possible errors returned by VRDevice.Error (see page 154) and by DeviceError.Error (see page 135) when the error facility is VRDevice (see page 152).

**Version**

2.0

**5.1.1.3.2.3 VRDevicePluginError****Visual Basic**

```
Public Enum VRDevicePluginError
```

**C#**

```
public enum VRDevicePluginError
```

**Members**

Members	Description
Success	No error. The operation is successful.
NotSupported	The requested operation is not supported by the device plugin.

**Description**

The VRDevicePluginError type defines the possible errors returned by VRDevicePlugin.Error (see page 161).

**Version**

2.0

**5.1.1.3.2.4 VRDeviceType****Visual Basic**

```
Public Enum VRDeviceType
```

**C#**

```
public enum VRDeviceType
```

**Members**

Members	Description
FileSystem	The video will be recorded to a file system device like hard disk
OpticalDisc	The video will be recorded to an optical disc device

**Description**

The VRDeviceType defines video recording storage devices.






**Version**

2.0


## 5.1.1.4 Structs, Records, Enums

The following table lists structs, records, enums in this documentation.

**Enumerations**

Enumeration	Description
 DVDBuilderError ( <a href="#">see page 167</a> )	<p>The DVDBuilderError type defines a range of error codes which are returned by the DVDBuilder.Error (<a href="#">see page 119</a>) property.</p> <p>Each error code has an auxiliary message that helps to identify the error. It is called an <i>error hint</i> and is returned by the method DVDBuilder.ErrorHint (<a href="#">see page 120</a>). Some error codes do not have a hint since such is not needed to identify the error. Other error codes specify as hints invalid project element or the path to a problematic input file. The hint format is specific to the error code that it explains but there are some common conventions:</p> <ul style="list-style-type: none"> <li>• A text in <i>italics</i> specifies... more (<a href="#">see page 167</a>)</li> </ul>
 DVDBuilderStatus ( <a href="#">see page 177</a> )	The DVDBuilderStatus type defines a range of status information. Members of the DVDBuilderStatus enumeration are passed to the DVDBuilder::OnStatus event, so that the application can respond to the changing status of the DVDBuilder ( <a href="#">see page 117</a> ) object being used.
 ErrorFacility ( <a href="#">see page 177</a> )	The ErrorFacility type defines the error source.
 SubpictureEncoderError ( <a href="#">see page 178</a> )	The SubpictureEncoderError type defines a range of error codes which are returned by the SubpictureEncoder.Error ( <a href="#">see page 125</a> ) property.
 SubpictureEncoderImageMaskColor ( <a href="#">see page 179</a> )	The SubpictureEncoderImageMaskColor type defines the subpicture mask color indices.

**Legend**

	Enumeration
---	-------------

### 5.1.1.4.1 DVDBuilderError

**Visual Basic**

```
Public Enum DVDBuilderError
```

**C#**

```
public enum DVDBuilderError
```

## Members

Members	Description
Success	= 0  The operation is successful.  Hint: <i>none</i>
SystemError	= 0x1  <b>This error constant is deprecated.</b> Use the DVDBuilder.ErrorFacility (see page 120) property to detect a system error and the DVDBuilder.Error (see page 119) property to get the actual error code. See the Error Handling (see page 4) topic for a detailed description. See DVDBuilder.ErrorFacility (see page 120)
Interrupted	= 0x2  Operation was interrupted by user.  Hint: <i>none</i>
InternalError	= 0x3  Unexpected error in DVDBuilder (see page 117)  Hint: <i>msg   none</i>
NoProject	= 0x100  No project is specified (neither buffer, not filename)  Hint: <i>none</i>
InvalidProjectXml	= 0x101  Malformed project XML  Hint: <i>line:num column:num error-message</i>
ProjectVersionMissing	= 0x102  Project version is not specified in the project  Hint: <i>none</i>
UnsupportedProjectVersion	= 0x103  The project version is higher then maximum recognized by DVDBuilder (see page 117)  Hint (maximum supported project version by DVDBuilder (see page 117)): <i>a.b.c.d</i>
VideoManagerMissing	= 0x104  There's no video manager in the project file. A <videoManager> element is required.  Hint: <i>none</i>

VideoManagerFirstPlayMissing	<p>= 0x105</p> <p>The video manager needs a first play command. It is executed when the DVD disc is loaded in the player. The command is set by the attribute 'firstPlayNavigate' of the &lt;videoManager&gt; element.</p> <p>Hint: <i>none</i></p>
InvalidButton	<p>= 0x106</p> <p>A &lt;button&gt; element is not defined correctly.</p> <p>Hint (one of the following):  &lt;videoManager&gt; &lt;menu id="num"&gt; &lt;button attributes&gt;  &lt;titleSet nn&gt; &lt;menu id="num"&gt; &lt;button attributes&gt;</p>
InvalidMenu	<p>= 0x107</p> <p>A &lt;menu&gt; element is not defined correctly.</p> <p>Hint (The invalid menu is either in the video manager or in a title set):  &lt;videoManager&gt; &lt;menu id="num"&gt;  &lt;titleSet nn&gt; &lt;menu id="num"&gt;</p>
MenuIdMissing	<p>= 0x108</p> <p>A &lt;menu&gt; element has no 'id' attribute.</p> <p>Hint (The invalid menu is either in the video manager or in a title set):  &lt;videoManager&gt;  &lt;titleSet nn&gt;</p>
DuplicateMenuId	<p>= 0x109</p> <p>A menu has the same id as another menu and the conflict cannot be resolved. Menu id's must be unique within a titleset and within the video manager. Additionally the root menu of a titleset must not conflict with the root menu of another titleset.</p> <p>Hint (The invalid menu is either in the video manager or in a title set):  &lt;videoManager&gt; &lt;menu id="duplicate-menu-id"&gt;  &lt;titleSet nn&gt; &lt;menu id="duplicate-menu-id"&gt;</p>
NoTitlesets	<p>= 0x10A</p> <p>A DVD must have at least one video titleset.</p> <p>Hint: <i>none</i></p>
NoTitles	<p>= 0x10B</p> <p>There are no titles in the titleset.</p> <p>Hint: <i>none</i></p>

NoMenus	= 0x10C  The <menus> element is empty. When used it must contain at least one <menu> element.  Hint: <i>none</i>
TitleIdMissing	= 0x10D  A <title> element has no 'id' attribute.  Hint (The titleset where the title is defined): <titleSet <i>nn</i> >
DuplicateTitleId	= 0x10E  The attribute 'id' with the same value is used in more than one <title>.  Hint (The titleset where the duplicate title is defined and the title itself): <titleSet <i>nn</i> > <title id="duplicate-title-id">
NoChapters	= 0x10F  A title must have at least one chapter.  Hint: <i>title-id</i>
NoVob	= 0x110  A title must have at least one video object.  Hint: <i>title-id</i>
VobFileMissing	= 0x111  A <videoObject> element has no 'file' attribute or it's empty.  Hint: <i>parent-title-id</i>
TooManyTitles	= 0x112  A DVD disc can contain up to 99 titles.  Hint: <i>none</i>
FirstChapterNotZeroTime	= 0x113  The first chapter in a title must start from 00:00:00  Hint: <i>title-id</i>
ChaptersOutOfOrder	= 0x114  The defined chapters must be successive in time.  Hint: <i>title-id</i>

InvalidChapterTime	= 0x115  The chapter time should be before the end of the video object; it should be in the form hh:mm:ss.  Hint: <i>title-id</i>
InvalidProjectVobAspectRatio	= 0x116  The aspect ratio specified in the project file is invalid. It must be either 4:3 or 16:9.  Hint: <title id="num"> file="file-path"
InvalidProjectVobResolution	= 0x117  The resolution is not specified correctly in the project file. Allowed resolutions: NTSC: 720x480   704x480   352x480   352x240 PAL: 720x576   704x576   352x576   352x288  Hint: <title id="num"> file="file-path"
InvalidProjectVobFrameRate	= 0x118  The framerate is not specified correctly in the project file. It must be either 29.97 fps (NTSC) or 25 fps (PAL).  Hint: <title id="num"> file="file-path"
InvalidProject	<b>This error constant is deprecated.</b> Use the specific project errors defined in the DVDBuilderError enum.
MenuVobTooBig	= 0x119 The menu VOB cannot be bigger than 1GB.  Hint: <i>none</i>
ElementaryStreamFileMissing	= 0x11A  An elementary stream (<videoStream>, <audioStream> or <subpictureStream>) element has no 'file' attribute or it's empty.  Hint: <i>parent-title-id</i>
InvalidAudioEsFormat	= 0x11B  The audio format is invalid. The audio format must be MPA, AC3 or DTS.  Hint: <audioStream file="file-path" format="audio-format"
InvalidElementaryStream	= 0x11C  The specified elementary stream file is invalid.  Hint (one of the following): <videoStream file="file-path" <audioStream file="file-path" <subpictureStream file="file-path"

InvalidVideoFrameRate	<p>= 0x200</p> <p>The frame rate of the input video stream must be either 29.97 fps for NTSC or 25 fps for PAL.</p> <p>Hint (one of the following):  <code>&lt;videManager&gt; &lt;menu id="num"&gt; file="file-path"</code>  <code>&lt;titleSet nn&gt; &lt;menu id="num"&gt; file="file-path"</code>  <code>&lt;titleSet nn&gt; &lt;title id="num"&gt; file="file-path" [(callback)]</code></p>
InvalidVideoFormat	<p>= 0x201</p> <p>The format of the input video must be either MPEG-1 or MPEG-2 program stream.</p>
InvalidVideoAspectRatio	<p>= 0x202</p> <p>The aspect ratio of the input video must be either 4:3 or 16:9.</p> <p>Hint (one of the following):  <code>&lt;videManager&gt; &lt;menu id="num"&gt; file="file-path"</code>  <code>&lt;titleSet nn&gt; &lt;menu id="num"&gt; file="file-path"</code>  <code>&lt;titleSet nn&gt; &lt;title id="num"&gt; file="file-path" [(callback)]</code></p>
InvalidVideoResolution	<p>= 0x203</p> <p>The resolution of the input video must be one of the following:  NTSC: 720x480   704x480   352x480   352x240  PAL: 720x576   704x576   352x576   352x288</p> <p>Hint (one of the following):  <code>&lt;videManager&gt; &lt;menu id="num"&gt; file="file-path"</code>  <code>&lt;titleSet nn&gt; &lt;menu id="num"&gt; file="file-path"</code>  <code>&lt;titleSet nn&gt; &lt;title id="num"&gt; file="file-path" [(callback)]</code></p>
InconsistentVideoStreams	<p>= 0x204</p> <p>A specific group of videos is required to have the same video parameters. The requirement applies to:</p> <ol style="list-style-type: none"> <li>1. all video objects in a titleset;</li> <li>2. all menu backgrounds in a titleset or in the video manager</li> </ol> <p>Hint (one of the following):  <code>&lt;videoManager&gt; &lt;menus&gt;</code>  <code>&lt;titleSet nn&gt; &lt;menus&gt;</code>  <code>&lt;titleSet nn&gt; &lt;titles&gt;</code></p>

InvalidVideoStream	<p><b>This error constant is deprecated.</b></p> <p>Use the specific video and multiplexer errors defined in the DVDBuilder (see page 117) enum.</p> <p>The input video stream is not in the expected format. The valid video streams are:</p> <p><b>NTSC</b></p> <p>Codec: MPEG-2; Maximum GOP Size: 18 frames; Maximum Bitrate: 9.8 Mbit/s  Aspect Ratio: 4:3   16:9  Resolution: 720x480   704x480   352x480   352x240  Frame Rate: 29.97 fps</p> <p><b>PAL</b></p> <p>Codec: MPEG-2; Maximum GOP Size: 15 frames; Maximum Bitrate: 9.8 Mbit/s  Aspect Ratio: 4:3   16:9  Resolution: 720x576   704x576   352x576   352x288  Frame Rate: 25 fps</p>
InvalidAudioFormat	<p>= 0x300</p> <p>The format of the input audio streams must be one of the following: LCPM, AC3, MPEG-1 Layer II or DTS.</p> <p>Hint (one of the following):</p> <pre>&lt;videManager&gt; &lt;menu id="num"&gt; file="file-path" &lt;titleSet nn&gt; &lt;menu id="num"&gt; file="file-path" &lt;titleSet nn&gt; &lt;title id="num"&gt; file="file-path" [(callback)]</pre>
InvalidAudioFrequency	<p>= 0x301</p> <p>The frequency (sampling rate) of the input audio must be either 48000 Hz or 96000 Hz.</p> <p>Hint (one of the following):</p> <pre>&lt;videManager&gt; &lt;menu id="num"&gt; file="file-path" &lt;titleSet nn&gt; &lt;menu id="num"&gt; file="file-path" &lt;titleSet nn&gt; &lt;title id="num"&gt; file="file-path" [(callback)]</pre>

InvalidAudioStream	<p><b>This error constant is deprecated.</b></p> <p>Use the specific video and multiplexer errors defined in the DVDBuilder (see page 117) enum.</p> <p>One or more input audio streams are not in the expected format. The valid audio streams are:</p> <p><b>LPCM (Linear Pulse Code Modulation)</b></p> <p>Sampling Frequency: 48KHz   96KHz  Bits Per Sample: 16   20   24  Maximum Bitrate: 6144 Kbit/s  Maximum Channels: 8</p> <p><b>MPEG-1 Audio (MPEG-1 Layer II)</b></p> <p>Sampling Frequency: 48KHz  Maximum Bitrate: 384 Kbit/s  Maximum Channels: 2</p> <p><b>MPEG-2 Audio (MPEG-1 Layer II + Extension Stream)</b></p> <p>Sampling Frequency: 48KHz  Maximum Bitrate: 912 Kbit/s  Maximum Channels: 8. The 7th and 8th channels are called Left-Center and Right-Center. This is equivalent to a 7.1 configuration.</p> <p><b>AC-3 (Dolby Digital)</b></p> <p>Sampling Frequency: 48KHz  Maximum Bitrate: 448 Kbit/s  Maximum Channels: 5.1</p> <p><b>DTS (Digital Theater System)</b></p> <p>Sampling Frequency: 48KHz  Maximum Bitrate: 1536 Kbit/s  Maximum Channels: 5.1</p>
SubpictureEncodingError	<p>= 0x400</p> <p>Subpicture encoding error. The encoded subpicture size exceeds 52KB.</p> <p>Hint: <i>bitmap-file</i></p>

InvalidBitmapDimensions	<p>= 0x401</p> <p>The bitmap width and height must be positive and divisible by 2 (even numbers).</p> <p>A positive height indicates that the row order is the usual upside-down.</p> <p><a href="http://en.wikipedia.org/wiki/BMP_file_format">http://en.wikipedia.org/wiki/BMP_file_format</a>: Pixels are stored "upside-down" with respect to normal image raster scan order, starting in the lower left corner, going from left to right, and then row by row from the bottom to the top of the image. Uncompressed Windows bitmaps can also be stored from the top row to the bottom, if the image height value is negative.</p> <p>Hint: <i>bitmap-file</i></p>
UnexpectedBitmapColor	<p>= 0x402</p> <p>The bitmap contains a color that is different from the 4 colors described in the project: pattern, background, emphasis1 and emphasis2.</p> <p>Hint: <i>bitmap-file</i></p>
InvalidBitmap	<p>= 0x403</p> <p>The input file is not a valid bitmap.</p> <p>Hint: <i>bitmap-file</i></p>
UnsupportedBitmapCompression	<p>= 0x404</p> <p>The input file is a compressed bitmap. An uncompressed bitmap is required.</p> <p>Hint: <i>bitmap-file</i></p>
UnsupportedBitmapColorDepth	<p>= 0x405</p> <p>The input bitmap does not have the supported color depth. The supported color depth is: 8-bit, 24-bit or 32-bit.</p> <p>Hint: <i>bitmap-file</i></p>
UnsupportedBitmapFormat	<p><b>This error constant is deprecated.</b></p> <p>Use the specific bitmap errors in the DVDBuilderError enum.</p>
MultiplexerError	<p>= 0x500</p> <p>A general MPEG multiplexer error.</p> <p>Hint: file="video-file" [(callback)]</p>
MultiplexerParams	<p>= 0x501</p> <p>Invalid parameters (e.g. no audio/video streams, or streams with invalid format) are passed to the multiplexer.</p> <p>Hint: file="video-file" [(callback)]</p>

MultiplexerUnderrun	= 0x502  The multiplexer has run out of input data.  Hint: file="video-file" [(callback)]
DataStreamError	= 0x1F01  A call to an DataStream (see page 114) method failed.  Hint: file="video-file" [(callback)]
InvalidNavigationCommand	= 0x1F02  Invalid navigation command. Wrong title ID, menu ID or chapter number found in a navigational attribute.  Hint (one of the following):  <videoManager firstPlayNavigate="command">  <titleSet nn> <title id="id" postNavigate="command">  <titleSet nn> <menu id="id"> <button navigate="command">  <videoManager> <menu id="id"> <button navigate="command">

### Description

The DVDBuilderError type defines a range of error codes which are returned by the DVDBuilder.Error (see page 119) property.

Each error code has an auxiliary message that helps to identify the error. It is called an *error hint* and is returned by the method DVDBuilder.ErrorHint (see page 120). Some error codes do not have a hint since such is not needed to identify the error. Other error codes specify as hints invalid project element or the path to a problematic input file. The hint format is specific to the error code that it explains but there are some common conventions:

- A text in *italics* specifies portions of the hint that are replaced by a concrete value.
- When several project elements are placed one after another, separated by a space, this represents a (partial) hierarchy of project elements leading to the element where the error is located. The first element in a series is located near the root of the xml structure. Each successive element is located deeper.

#### Hint samples (definition and actual string):

hint: *none*

actual string: empty

hint: line:*num* column:*num* *error-message*

actual string: line:1 column:5 Unknown error.

hint: x.x.x.x

actual string: 1.1.0.1

hint: <videoManager> <menu id="*num*"> <button *attributes*>

actual string: <videoManager> <menu id="100"> <button left="10" top=(missing) height="30" width="100" navigate=(missing)>

This hint specifies a button with missing 'top' and 'navigate' attributes. The button is located in a menu with an id '10'. The menu itself is in the video manager.

hint: <titleSet *nn*> <menu id="duplicate-menu-id">

actual string: <titleSet 02> <menu id="5">

This hint species a menu with id '5'. The menu is located in the second titleset in the dvd structure. NOTE: Titlesets are numbered from 01 to 99, but the numbering is implicitly based on their order in the project file.

hint: <titleSet *nn*> <title id="num"> file="file-path" [(callback)]

actual string: <titleSet 01> <title id="1"> file="C:\tmp\movies\test.mpg"

This hint specifies a video file "C:\tmp\movies\test.mpg" which is defined in a title with id '1'. The title is located in the first titleset defined in the project.

### See Also

DVDBuilder.ErrorHint (see page 120)

## 5.1.1.4.2 DVDBuilderStatus

### Visual Basic

```
Public Enum DVDBuilderStatus
```

### C#

```
public enum DVDBuilderStatus
```

### Members

Members	Description
WritingVOB	Writing Video Manager and Video Title Set video objects, VOB files.
WritingIFO	Writing Video Manager and Video Title Set information files like: VIDEO_TS.IFO, VIDEO_TS.BUP, VTS_XX_0.IFO and VTS_XX_0.BUP.

### Description

The DVDBuilderStatus type defines a range of status information. Members of the DVDBuilderStatus enumeration are passed to the DVDBuilder::OnStatus event, so that the application can respond to the changing status of the DVDBuilder (see page 117) object being used.

### Version

1.00

## 5.1.1.4.3 ErrorFacility

### Visual Basic

```
Public Enum ErrorFacility
```

### C#

```
public enum ErrorFacility
```

### Members

Members	Description
Success	No error. The operation is successful.
System	Operating System Error.

VideoRecorder	Video Recorder Error. See VideoRecorderError (see page 163).
VRDevicePlugin	Device Plugin Error. See VRDevicePluginError (see page 166).
VRDevice	Device Error. See VRDeviceError (see page 165).
PBEngine	Primo Burner Error
DVDBuilder	DVDBuilder Error. See DVDBuilderError (see page 167)
SubpictureEncoder	Subpicture Encoder Error. See SubpictureEncoderError (see page 178)

**Description**

The ErrorFacility type defines the error source.

**Version**

2.0

**5.1.1.4.4 SubpictureEncoderError****Visual Basic**

```
Public Enum SubpictureEncoderError
```

**C#**

```
public enum SubpictureEncoderError
```

**Members**

Members	Description
Success	= 0 The operation is successful.
InternalError	= 0x601 Internal error. An unexpected error in SubpictureEncoder (see page 124).
SyntaxError	= 0x602 Syntax error in the input file.
UnsupportedFeature	= 0x603 An unsupported feature is specified in the input file.
ReadImageError	= 0x604 The input image can not be loaded.
InvalidBitmapDimensions	= 0x605 The bitmap width and height must be positive and divisible by 2 (even numbers).
UnexpectedBitmapColor	= 0x606 The bitmap contains a color that is different from the 4 colors described in the project.

SubpictureEncodingError	= 0x607  Subpicture encoding error. The encoded subpicture size exceeds 52KB.
-------------------------	---

**Description**

The SubpictureEncoderError type defines a range of error codes which are returned by the SubpictureEncoder.Error (see page 125) property.

**See Also**

SubpictureEncoder.ErrorHint (see page 126)

### 5.1.1.4.5 SubpictureEncoderImageMaskColor

**Visual Basic**

```
Public Enum SubpictureEncoderImageMaskColor
```

**C#**

```
public enum SubpictureEncoderImageMaskColor
```

**Members**

Members	Description
Background	Background color index.
Pattern	Pattern color index.
Emphasis1	Emphasis1 color index.
Emphasis2	Emphasis2 color index.

**Description**

The SubpictureEncoderImageMaskColor type type defines the subpicture mask color indices.

**See Also**

SubpictureEncoder.GetImageMaskColor (see page 127), SubpictureEncoder.SetImageMaskColor (see page 127)

### 5.1.1.5 Types

The following table lists types in this documentation.

**Types**

Type	Description
DVDBuilder_ContinueEventHandler (see page 180)	Represents the method that handles the DVDBuilder::OnContinue event raised to check if dvd creation should continue. When you create a DVDBuilder_ContinueEventHandler delegate, you identify the method that will handle the event. To associate the event with your event handler, add an instance of the delegate to the event. The event handler is called whenever the event occurs, unless you remove the delegate. The declaration of your event handler must have the same parameters as the DVDBuilder_ContinueEventHandler delegate declaration.

DVDBuilder_ProgressEventHandler (🔗 see page 181)	<p>Represents the method that handles the DVDBuilder::OnProgress event raised to report the dvd creation process progress.</p> <p>When you create a DVDBuilder_ProgressEventHandler delegate, you identify the method that will handle the event. To associate the event with your event handler, add an instance of the delegate to the event. The event handler is called whenever the event occurs, unless you remove the delegate.</p> <p>The declaration of your event handler must have the same parameters as the DVDBuilder_ProgressEventHandler delegate declaration.</p>
DVDBuilder_StatusEventHandler (🔗 see page 181)	<p>Represents the method that handles the DVDBuilder::OnStatus event raised to report the status of the dvd creation process.</p> <p>When you create a DVDBuilder_StatusEventHandler delegate, you identify the method that will handle the event. To associate the event with your event handler, add an instance of the delegate to the event. The event handler is called whenever the event occurs, unless you remove the delegate.</p> <p>The declaration of your event handler must have the same parameters as the DVDBuilder_StatusEventHandler delegate declaration.</p>

### 5.1.1.5.1 DVDBuilder\_ContinueEventHandler

#### Visual Basic

```
Public Delegate Function DVDBuilder_ContinueEventHandler(ByVal sender As Object, ByVal e As EventArgs) As Boolean
```

#### C#

```
public delegate bool DVDBuilder_ContinueEventHandler(object sender, EventArgs e);
```

#### Parameters

Parameters	Description
sender	[in] The source of the event.
e	[in] An EventArgs object that contains the event data.

#### Returns

true The dvd creation should continue.

false The dvd creation should be aborted.

#### Description

Represents the method that handles the DVDBuilder::OnContinue event raised to check if dvd creation should continue.

When you create a DVDBuilder\_ContinueEventHandler delegate, you identify the method that will handle the event. To associate the event with your event handler, add an instance of the delegate to the event. The event handler is called whenever the event occurs, unless you remove the delegate.

The declaration of your event handler must have the same parameters as the DVDBuilder\_ContinueEventHandler delegate declaration.

#### See Also

DVDBuilder.OnContinue (🔗 see page 121)

### 5.1.1.5.2 DVDBuilder\_ProgressEventHandler

#### Visual Basic

```
Public Delegate Sub DVDBuilder_ProgressEventHandler(ByVal sender As Object, ByVal e As DVDBuilder_ProgressEventArgs)
```

#### C#

```
public delegate void DVDBuilder_ProgressEventHandler(object sender, DVDBuilder_ProgressEventArgs e);
```

#### Parameters

Parameters	Description
sender	[in] The source of the event.
e	[in] A DVDBuilder_ProgressEventArgs (see page 128) object that contains the event data.

#### Description

Represents the method that handles the DVDBuilder::OnProgress event raised to report the dvd creation process progress.

When you create a DVDBuilder\_ProgressEventHandler delegate, you identify the method that will handle the event. To associate the event with your event handler, add an instance of the delegate to the event. The event handler is called whenever the event occurs, unless you remove the delegate.

The declaration of your event handler must have the same parameters as the DVDBuilder\_ProgressEventHandler delegate declaration.

#### See Also

DVDBuilder.OnProgress (see page 121), DVDBuilder\_ProgressEventArgs (see page 128)

### 5.1.1.5.3 DVDBuilder\_StatusEventHandler

#### Visual Basic

```
Public Delegate Sub DVDBuilder_StatusEventHandler(ByVal sender As Object, ByVal e As DVDBuilder_StatusEventArgs)
```

#### C#

```
public delegate void DVDBuilder_StatusEventHandler(object sender, DVDBuilder_StatusEventArgs e);
```

#### Parameters

Parameters	Description
sender	[in] The source of the event.
e	[in] A DVDBuilder_StatusEventArgs (see page 129) object that contains the event data.

#### Description

Represents the method that handles the DVDBuilder::OnStatus event raised to report the status of the dvd creation process.

When you create a DVDBuilder\_StatusEventHandler delegate, you identify the method that will handle the event. To associate the event with your event handler, add an instance of the delegate to the event. The event handler is called whenever the event occurs, unless you remove the delegate.

The declaration of your event handler must have the same parameters as the DVDBuilder\_StatusEventHandler delegate declaration.

**See Also**

DVDBuilder.OnStatus ([see page 121](#)), DVDBuilder\_StatusEventArgs ([see page 129](#))

## Index

### <

<action> Element 17, 58  
 <audioStream> Element 14, 55, 86  
 <audioStreams> Element 105  
 <background> Element 13, 53  
 <button> Element 18, 59  
 <color> 104  
 <command> Element 20, 28, 35, 43, 60, 68, 76, 91, 98  
 <display> Element 16, 56  
 <dvd> Element 7  
 <firstPlayCommands> Element 40  
 <mask> Element 15, 55  
 <menu> Element 9, 49  
 <menus> Element 9, 49  
 <opcode> Element 22, 30, 38, 46, 63, 71, 78, 93, 101  
 <postCommands> Element 33, 65, 88  
 <preCommands> Element 25, 73, 96  
 <selection> Element 17, 57  
 <stream> Element 105, 107  
 <subpicturePalette> Element 103  
 <subpictureStream> Element 87  
 <subpictureStreams> Element 106  
 <title> Element 81  
 <titles> Element 81  
 <titleSet> Element 48  
 <videoManager> Element 8  
 <videoObject> Element 84  
 <videoStream> Element 14, 54, 87

### D

DataStream 114
 

- Close 115
- GetSize 115
- Open 116
- Read 116
- Seek 117
- Write 117

 DeviceError 134
 

- Error 135

ErrorFacility 135
 

- SystemErrorDomain 136

 DVD Language Codes 109  
 DVD Project File 7  
 DVDBuilder 117
 

- Build 123
- Error 119
- ErrorFacility 120
- ErrorHint 120
- InputDataStreamFactory 120
- OnContinue 121
- OnProgress 121
- OnStatus 121
- OutputFolder 122
- Project 122
- ProjectFile 122
- SystemErrorDomain 122

 DVDBuilder\_ContinueEventHandler 180  
 DVDBuilder\_ProgressEventArgs 128
 

- Percent 128

 DVDBuilder\_ProgressEventHandler 181  
 DVDBuilder\_StatusEventArgs 129
 

- Status 129

 DVDBuilder\_StatusEventHandler 181  
 DVDBuilderError 167  
 DVDBuilderStatus 177

### E

Error Handling 4  
 ErrorFacility 177

### F

FileSystemDeviceConfig 136
 

- Folder 136

### I

InputDataStreamFactory 123
 

- Create 124
- Destroy 124

**L**

## Library 130

- CreateDVDBuilder 130
- CreateSubpictureEncoder 130
- CreateVideoRecorder 131
- Initialize 131
- SetLicenseInfo 131
- Terminate 132

**O**

## OpticalDiscDeviceConfig 137

- DeviceHandle 137
- DriveLetter 138
- VolumeLabel 138

**P**

## PrimoSoftware namespace 112

## PrimoSoftware::DVDBuilder namespace 112

## PrimoSoftware::DVDBuilder::VR namespace 132

**R**

## Release Notes 1

**S**

## SubpictureEncoder 124

- Encode 127
- Error 125
- ErrorFacility 126
- ErrorHint 126
- GetImageMaskColor 127
- SetImageMaskColor 127
- SystemErrorDomain 126

## SubpictureEncoderError 178

## SubpictureEncoderImageMaskColor 179

**T**

## Title 138

- Duration 139

## TitleEnumerator 139

- Count 140

Item 140

**V**

## VideoRecorder 140

- AllowMuxBufferUnderflow 143
- AverageBitrate 144
- Devices 144
- Error 144
- ErrorFacility 145
- FinalizeMedia 146
- GetDeviceError 147
- GetIsFinalized 147
- GetMediaFreeSpace 148
- GetTitles 149
- IsFinalized 145
- IsFinalizeSupported 145
- LoadDevicePlugin 149
- MediaFreeSpace 146
- PrepareMedia 149
- Start 150
- StartAsync 150
- Stop 151
- SystemErrorDomain 146
- Write 151

## VideoRecorderError 163

## VRDevice 152

- Config 154
- EraseMedia 156
- Error 154
- ErrorFacility 154
- Initialize 157
- MediaFreeSpace 155
- MedialsBlank 155
- MedialsReWritable 155
- NotifyOSFileSystemChanged 157
- SystemErrorDomain 156
- Type 156

## VRDeviceError 165

## VRDeviceList 157

- Add 159
- Clear 159
- Count 158

Item 158

RemoveAt 159

VRDevicePlugin 160

CreateFileSystemDevice 162

CreateOpticalDiscDevice 162

Error 161

ErrorFacility 161

GetAPIRevision 163

SystemErrorDomain 161

VRDevicePluginError 166

VRDeviceType 166