NSOpenGLPixelBuffer Class Reference

Cocoa > Graphics & Imaging

2007-01-31

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

Simultaneously published in the United States and Canada.

Contents

NSOpenGLPixelBuffer Class Reference

Inherits from	NSObject
Conforms to	NSObject (NSObject)
Framework	/System/Library/Frameworks/AppKit.framework
Availability	Available in Mac OS X v10.3 and later.
Companion guide	Cocoa Drawing Guide
Declared in	NSOpenGL.h
Related sample code	Quartz Composer Offline Rendering Quartz Composer Texture

Overview

The `NSOpenGLPixelBuffer` class gives Cocoa OpenGL implementations access to accelerated offscreen rendering. With this offscreen rendering you could, for instance, draw into the pixel buffer, then use the contents as a texture map elsewhere. Typically you initialize an `NSOpenGLPixelBuffer` object using the `initWithTextureTarget:textureInternalFormat:textureMaxMipMapLevel:pixelsWide:pixelsHigh:` (page 6) method and attach the resulting object to an OpenGL context with the `setPixelBuffer:cubeMapFace:mipMapLevel:currentVirtualScreen:` method of `NSOpenGLContext`.

Tasks

Initializing an OpenGL Pixel Buffer

`- initWithTextureTarget:textureInternalFormat:textureMaxMipMapLevel:pixelsWide:pixelsHigh:` (page 6)

Returns an `NSOpenGLPixelBuffer` object initialized with the specified parameters.

Obtaining Information About an OpenGL Pixel Buffer

`- pixelsHigh` (page 7)

Returns the height of the receiver's texture (in pixels).

- `pixelsWide` (page 7)
  Returns the width of the receiver's texture (in pixels).
- `textureInternalFormat` (page 8)
  Returns the internal format of the receiver's texture.
- `textureMaxMipMapLevel` (page 8)
  Returns the maximum mipmap level of the receiver's texture.
- `textureTarget` (page 8)
  Returns the texture target of the receiver.

Instance Methods

initWithTextureTarget:textureInternalFormat:textureMaxMipMapLevel:pixelsWide:pixelsHigh:

Returns an `NSOpenGLPixelBuffer` object initialized with the specified parameters.

```
- (id)initWithTextureTarget:(GLenum)target textureInternalFormat:(GLenum)format
    textureMaxMipMapLevel:(GLint)maxLevel pixelsWide:(GLsizei)pixelsWide
    pixelsHigh:(GLsizei)pixelsHigh
```

Parameters

target
  The texture object. This value should be one of the following: `GL_TEXTURE_2D`,`GL_TEXTURE_CUBE_MAP`, or `GL_TEXTURE_RECTANGLE_EXT`.

format
  The base internal format of the texture. This value should be `GL_RGB`, `GL_RGBA`, or `GL_DEPTH_COMPONENT`.

maxLevel
  The desired maximum mipmap level of the structure, starting with zero.

pixelsWide
  The width of the texture (in pixels) in the pixel buffer.

pixelsHigh
  The height of the texture (in pixels) in the pixel buffer.

Return Value
An initialized `NSOpenGLPixelBuffer` object or `nil` if the initialization failed. Initialization can fail if there is inconsistency among the parameter values. See the OpenGL documentation for `glTexImage2D` for more information.

Discussion
The value you pass to the *target* parameter defines several other constraints that are then applied to the remaining parameters. The list below gives the values you can pass to *target* and the additional constraints.

- `GL_TEXTURE_2D`
- `GL_TEXTURE_CUBE_MAP` - the values in *pixelsWide* and *pixelsHigh* must be equal.
- `GL_TEXTURE_RECTANGLE_EXT` - *maxLevel* must be zero.

Normally, when using the `GL_TEXTURE_2D` and `GL_TEXTURE_CUBE_MAP` targets, you must specify width and height values that are powers of two. When the `ARB_texture_non_power_of_two` extension is present, however, some types of hardware can support values that are not powers of two. You should check for the presence of this extension before specifying non power-of-two values.

If the texture map cannot be created, you can use the `glGetError` function to get the error code.

Availability
Available in Mac OS X v10.3 and later.

Related Sample Code
Quartz Composer Offline Rendering
Quartz Composer Texture

Declared In
`NSOpenGL.h`

pixelsHigh

Returns the height of the receiver's texture (in pixels).

```
- (GLsizei)pixelsHigh
```

Return Value
The height of the texture (in pixels).

Availability
Available in Mac OS X v10.3 and later.

See Also
`- pixelsWide` (page 7)

Declared In
`NSOpenGL.h`

pixelsWide

Returns the width of the receiver's texture (in pixels).

```
- (GLsizei)pixelsWide
```

Return Value
The width of the texture (in pixels).

Availability
Available in Mac OS X v10.3 and later.

See Also
`- pixelsHigh` (page 7)

Declared In
`NSOpenGL.h`

textureInternalFormat

Returns the internal format of the receiver's texture.

```
- (GLenum)textureInternalFormat
```

Return Value
The texture format, which can be one of the following values: `GL_RGB`, `GL_RGBA`, or `GL_DEPTH_COMPONENT`.

Availability
Available in Mac OS X v10.3 and later.

Declared In
`NSOpenGL.h`

textureMaxMipMapLevel

Returns the maximum mipmap level of the receiver's texture.

```
- (GLint)textureMaxMipMapLevel
```

Return Value
The maximum mipmap level.

Availability
Available in Mac OS X v10.3 and later.

Declared In
`NSOpenGL.h`

textureTarget

Returns the texture target of the receiver.

```
- (GLenum)textureTarget
```

Return Value
The texture target, which can be one of the following values: `GL_TEXTURE_2D`, `GL_TEXTURE_CUBE_MAP`, or `GL_TEXTURE_RECTANGLE_EXT`.

Availability
Available in Mac OS X v10.3 and later.

Declared In
`NSOpenGL.h`

Document Revision History

This table describes the changes to *NSOpenGLPixelBuffer Class Reference*.

Date	Notes
2007-01-31	Removed references to 1D pbuffer targets, which are not supported. Corrected information about power-of-two width and height values. Updated for Mac OS X v10.5.
2006-05-23	First publication of this content as a separate document.

Document Revision History

Index