Name

glCheckFramebufferStatus, glCheckNamedFramebufferStatus — check the completeness status of a framebuffer

C Specification

GLenum **glCheckFramebufferStatus**(GLenum *target*);

GLenum **glCheckNamedFramebufferStatus**(GLuint *framebuffer*,
GLenum *target*);

Parameters

`target`

Specify the target to which the framebuffer is bound for **glCheckFramebufferStatus**, and the target against which framebuffer completeness of `framebuffer` is checked for **glCheckNamedFramebufferStatus**.

`framebuffer`

Specifies the name of the framebuffer object for **glCheckNamedFramebufferStatus**

Description

glCheckFramebufferStatus and **glCheckNamedFramebufferStatus** return the completeness status of a framebuffer object when treated as a read or draw framebuffer, depending on the value of `target`.

For **glCheckFramebufferStatus**, the framebuffer checked is that bound to `target`, which must be `GL_DRAW_FRAMEBUFFER`, `GL_READ_FRAMEBUFFER` or `GL_FRAMEBUFFER`. `GL_FRAMEBUFFER` is equivalent to `GL_DRAW_FRAMEBUFFER`.

For **glCheckNamedFramebufferStatus**, `framebuffer` is zero or the name of the framebuffer object to check. If `framebuffer` is zero, then the status of the default read or draw framebuffer, as determined by `target`, is returned.

The return value is `GL_FRAMEBUFFER_COMPLETE` if the specified framebuffer is complete. Otherwise, the return value is determined as follows:

- `GL_FRAMEBUFFER_UNDEFINED` is returned if the specified framebuffer is the default read or draw framebuffer, but the default framebuffer does not exist.
- `GL_FRAMEBUFFER_INCOMPLETE_ATTACHMENT` is returned if any of the framebuffer attachment points are framebuffer incomplete.
- `GL_FRAMEBUFFER_INCOMPLETE_MISSING_ATTACHMENT` is returned if the framebuffer does not have at least one image attached to it.

- GL_FRAMEBUFFER_INCOMPLETE_DRAW_BUFFER is returned if the value of GL_FRAMEBUFFER_ATTACHMENT_OBJECT_TYPE is GL_NONE for any color attachment point(s) named by GL_DRAW_BUFFERi.

- GL_FRAMEBUFFER_INCOMPLETE_READ_BUFFER is returned if GL_READ_BUFFER is not GL_NONE and the value of GL_FRAMEBUFFER_ATTACHMENT_OBJECT_TYPE is GL_NONE for the color attachment point named by GL_READ_BUFFER.

- GL_FRAMEBUFFER_UNSUPPORTED is returned if the combination of internal formats of the attached images violates an implementation-dependent set of restrictions.

- GL_FRAMEBUFFER_INCOMPLETE_MULTISAMPLE is returned if the value of GL_RENDERBUFFER_SAMPLES is not the same for all attached renderbuffers; if the value of GL_TEXTURE_SAMPLES is the not same for all attached textures; or, if the attached images are a mix of renderbuffers and textures, the value of GL_RENDERBUFFER_SAMPLES does not match the value of GL_TEXTURE_SAMPLES.

- GL_FRAMEBUFFER_INCOMPLETE_MULTISAMPLE is also returned if the value of GL_TEXTURE_FIXED_SAMPLE_LOCATIONS is not the same for all attached textures; or, if the attached images are a mix of renderbuffers and textures, the value of GL_TEXTURE_FIXED_SAMPLE_LOCATIONS is not GL_TRUE for all attached textures.

- GL_FRAMEBUFFER_INCOMPLETE_LAYER_TARGETS is returned if any framebuffer attachment is layered, and any populated attachment is not layered, or if all populated color attachments are not from textures of the same target.

Additionally, if an error occurs, zero is returned.

Errors

GL_INVALID_ENUM is generated if target is not GL_DRAW_FRAMEBUFFER, GL_READ_FRAMEBUFFER or GL_FRAMEBUFFER.

GL_INVALID_OPERATION is generated by **glCheckNamedFramebufferStatus** if framebuffer is not zero or the name of an existing framebuffer object.

Version Support

Function / Feature Name	OpenGL Version											
	2.0	2.1	3.0	3.1	3.2	3.3	4.0	4.1	4.2	4.3	4.4	4.5
glCheckFramebufferStatus	-	-	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
glCheckNamedFramebufferStatus	-	-	-	-	-	-	-	-	-	-	-	✓

See Also

glGenFramebuffers, glDeleteFramebuffers glBindFramebuffer

Copyright