

clCreateContext

Creates an OpenCL context.

```
cl_context clCreateContext(const cl_context_properties *properties,
                          cl_uint num_devices,
                          const cl_device_id *devices,
                          (void CL_CALLBACK *pfn_notify) (
                          const char *errinfo,
                              const void *private_info, size_t cb,
                              void *user_data
                          ),
                          void *user_data,
                          cl_int *errcode_ret)
```

Description

An OpenCL context is created with one or more devices. Contexts are used by the OpenCL runtime for managing objects such as command-queues, memory, program and kernel objects and for executing kernels on one or more devices specified in the context.

Parameters

properties

Specifies a list of context property names and their corresponding values. Each property name is immediately followed by the corresponding desired value. The list is terminated with 0. *properties* can be NULL in which case the platform that is selected is implementation-defined. The list of supported *properties* is described in the table below.

If the extension [cl_khr_dx9_media_sharing](#) is enabled, then *properties* specifies a list of context property names and their corresponding values. Each property is followed immediately by the corresponding desired value. The list is terminated with zero. If a property is not specified in *properties*, then its default value (listed in the table below) is used (it is said to be specified implicitly). If *properties* is NULL or empty (points to a list whose first value is zero), all attributes take on their default values.

If the extension [cl_khr_d3d10_sharing](#) is enabled, then *properties* specifies a list of context property names and their corresponding values. Each property is followed immediately by the corresponding desired value. The list is terminated with zero. If a property is not specified in *properties*, then its default value is used (it is said to be specified implicitly). If *properties* is NULL or empty (points to a list whose first value is zero), all attributes take on their default value.

If the extension [cl_khr_gl_sharing](#) is enabled, then *properties* points to an attribute list, which is a array of ordered <attribute name, value> pairs terminated with zero. If an attribute is not specified in *properties*, then its default value is used (it is said to be specified implicitly). If *properties* is NULL or empty (points to a list whose first value is zero), all attributes take on their default values..

cl_context_properties enum	Property value	Description
----------------------------	----------------	-------------

cl_context_properties enum	Property value	Description
CL_CONTEXT_PLATFORM	cl_platform_id	Specifies the platform to use.
CL_CONTEXT_INTEROP_USER_SYNC	cl_bool	Specifies whether the user is responsible for synchronization between OpenCL and other APIs. Please refer to the specific sections in the OpenCL 1.2 extension specification that describe sharing with other APIs for restrictions on using this flag. If CL_CONTEXT_INTEROP_USER_SYNC is not specified, a default of CL_FALSE is assumed.
CL_CONTEXT_D3D10_DEVICE_KHR	ID3D10Device*	If the cl_khr_d3d10_sharing extension is enabled, specifies the ID3D10Device* to use for Direct3D 10 interoperability. The default value is NULL.
CL_GL_CONTEXT_KHR	0, OpenGL context handle	OpenGL context to associated the OpenCL context with (available if the cl_khr_gl_sharing extension is enabled)
CL_EGL_DISPLAY_KHR	EGL_NO_DISPLAY, EGLDisplay handle	EGLDisplay an OpenGL context was created with respect to (available if the cl_khr_gl_sharing extension is enabled)
CL_GLX_DISPLAY_KHR	None, X handle	X Display an OpenGL context was created with respect to (available if the cl_khr_gl_sharing extension is enabled)
CL_CGL_SHAREGROUP_KHR	0, CGL share group handle	CGL share group to associate the OpenCL context with (available if the cl_khr_gl_sharing extension is enabled)
CL_WGL_HDC_KHR	0, HDC handle	HDC an OpenGL context was created with respect to (available if the cl_khr_gl_sharing extension is enabled)

cl_context_properties enum	Property value	Description
CL_CONTEXT_ADAPTER_D3D9_KHR	IDirect3DDevice9 *	Specifies an IDirect3DDevice9 to use for D3D9 interop (if the cl_khr_dx9_media_sharing extension is supported).
CL_CONTEXT_ADAPTER_D3D9EX_KHR	IDirect3DDeviceEx*	Specifies an IDirect3DDevice9Ex to use for D3D9 interop (if the cl_khr_dx9_media_sharing extension is supported).
CL_CONTEXT_ADAPTER_DXVA_KHR	IDXVAHD_Device *	Specifies an IDXVAHD_Device to use for DXVA interop (if the cl_khr_dx9_media_sharing extension is supported).
CL_CONTEXT_D3D11_DEVICE_KHR	ID3D11Device *	Specifies the ID3D11Device * to use for Direct3D 11 interoperability. The default value is NULL.

num_devices

The number of devices specified in the *devices* argument.

devices

A pointer to a list of unique devices returned by [clGetDeviceIDs](#) or sub-devices created by [clCreateSubDevices](#) for a platform.

pfn_notify

A callback function that can be registered by the application. This callback function will be used by the OpenCL implementation to report information on errors during context creation as well as errors that occur at runtime in this context. This callback function may be called asynchronously by the OpenCL implementation. It is the application's responsibility to ensure that the callback function is thread-safe. If *pfn_notify* is NULL, no callback function is registered. The parameters to this callback function are:

errinfo is a pointer to an error string.

private_info and *cb* represent a pointer to binary data that is returned by the OpenCL implementation that can be used to log additional information helpful in debugging the error.

user_data is a pointer to user supplied data.

NOTE: There are a number of cases where error notifications need to be delivered due to an error that occurs outside a context. Such notifications may not be delivered through the *pfn_notify* callback. Where these notifications go is implementation-defined.

user_data

Passed as the *user_data* argument when *pfn_notify* is called. *user_data* can be NULL.

errcode_ret

Returns an appropriate error code. If *errcode_ret* is NULL, no error code is returned.

Notes

clCreateContext and **clCreateContextFromType** perform an implicit retain. This is very helpful for 3rd party libraries, which typically get a context passed to them by the application. However, it is possible that the application may delete the context without informing the library. Allowing functions to attach to (i.e. retain) and release a context solves the problem of a context being used by a library no longer being valid.

Errors

clCreateContext returns a valid non-zero context and *errcode_ret* is set to CL_SUCCESS if the context is created successfully. Otherwise, it returns NULL value with the following error values returned in *errcode_ret*:

- CL_INVALID_PLATFORM if *properties* is NULL and no platform could be selected or if platform value specified in *properties* is not a valid platform. (If the extension [cl_khr_gl_sharing](#) is enabled, then this error is replaced with CL_INVALID_GL_SHAREGROUP_REFERENCE_KHR; see below.)
- CL_INVALID_PROPERTY if context property name in *properties* is not a supported property name, if the value specified for a supported property name is not valid, or if the same property name is specified more than once. However if the extension [cl_khr_gl_sharing](#) is enabled, then CL_INVALID_PROPERTY is returned if an attribute name other than those listed in the table for *properties* above or if CL_CONTEXT_INTEROP_USER_SYNC is specified in *properties*.
- CL_INVALID_PROPERTY if an attribute name other than those specified in table 4.5 or if CL_CONTEXT_INTEROP_USER_SYNC is specified in *properties*.
- CL_INVALID_VALUE if *devices* is NULL; if *num_devices* is equal to zero; or if *pfn_notify* is NULL but *user_data* is not NULL.
- CL_INVALID_DEVICE if *devices* contains an invalid device.
- CL_INVALID_OPERATION if Direct3D 10 interoperability is specified by setting CL_INVALID_D3D10_DEVICE_KHR to a non-NULL value, and interoperability with another graphics API is also specified (if the [cl_khr_d3d10_sharing](#) extension is enabled).
- CL_DEVICE_NOT_AVAILABLE if a device in *devices* is currently not available even though the device was returned by [clGetDeviceIDs](#).
- CL_OUT_OF_RESOURCES if there is a failure to allocate resources required by the OpenCL implementation on the device.
- CL_OUT_OF_HOST_MEMORY if there is a failure to allocate resources required by the OpenCL implementation on the host.
- CL_INVALID_D3D10_DEVICE_KHR if the Direct3D 10 device specified for interoperability is not compatible with the devices against which the context is to be created (if the [cl_khr_d3d10_sharing](#) extension is enabled).

- `CL_INVALID_D3D10_DEVICE_KHR` if the value of the property `CL_CONTEXT_D3D10_DEVICE_KHR` is non-NULL and does not specify a valid Direct3D 10 device with which the *cl_device_ids* against which this context is to be created may interoperate (if the [cl_khr_d3d10_sharing](#) extension is enabled).
- `CL_INVALID_GL_SHAREGROUP_REFERENCE_KHR` when an invalid OpenGL context or share group object handle is specified in *properties* (only if the [cl_khr_gl_sharing](#) extension is enabled).
- `CL_INVALID_GL_SHAREGROUP_REFERENCE_KHR` if no OpenGL or OpenGL ES context or share group is specified in the attribute list given to [clCreateContext](#) and any of the commands in section 9.7 are called. (if the [cl_khr_gl_sharing](#) extension is enabled)
- `CL_INVALID_GL_SHAREGROUP_REFERENCE_KHR` if the [cl_khr_gl_sharing](#) extension is enabled and if a context was specified by any of the following means:
 - A context specified for an EGL-based OpenGL ES or OpenGL implementation by setting the attributes `CL_GL_CONTEXT_KHR` and `CL_EGL_DISPLAY_KHR`.
 - A context was specified for a GLX-based OpenGL implementation by setting the attributes `CL_GL_CONTEXT_KHR` and `CL_GLX_DISPLAY_KHR`.
 - A context was specified for a WGL-based OpenGL implementation by setting the attributes `CL_GL_CONTEXT_KHR` and `CL_WGL_HDC_KHR`.

and any of the following conditions hold:

- The specified display and context attributes do not identify a valid OpenGL or OpenGL ES context.
- The specified context does not support buffer and renderbuffer objects.
- The specified context is not compatible with the OpenCL context being created (for example, it exists in a physically distinct address space, such as another hardware device, or does not support sharing data with OpenCL due to implementation restrictions).
- `CL_INVALID_GL_SHAREGROUP_REFERENCE_KHR` if a share group was specified for a CGL-based OpenGL implementation by setting the attribute `CL_CGL_SHAREGROUP_KHR`, and the specified share group does not identify a valid CGL share group object (if the [cl_khr_gl_sharing](#) extension is enabled).
- `CL_INVALID_OPERATION` if a context was specified as described above and any of the following conditions hold:
 - A context or share group object was specified for one of CGL, EGL, GLX, or WGL and the OpenGL implementation does not support that window-system binding API.
 - More than one of the attributes `CL_CGL_SHAREGROUP_KHR`, `CL_EGL_DISPLAY_KHR`, `CL_GLX_DISPLAY_KHR`, and `CL_WGL_HDC_KHR` is set to a non-default value.
 - Both of the attributes `CL_CGL_SHAREGROUP_KHR` and `CL_GL_CONTEXT_KHR` are set to non-default values.
 - Any of the devices specified in the *devices* argument cannot support OpenCL

objects which share the data store of an OpenGL object, as described in section 9.7.

- `CL_INVALID_DX9_MEDIA_ADAPTER_KHR` if the media adapter specified for interoperability is not compatible with the devices against which the context is to be created (only if the [cl_khr_dx9_media_sharing](#) extension is supported).
- `CL_INVALID_ADAPTER_KHR` if any of the values of the properties `CL_CONTEXT_ADAPTER_D3D9_KHR`, `CL_CONTEXT_ADAPTER_D3D9EX_KHR` or `CL_CONTEXT_ADAPTER_DXVA_KHR` is non-NULL and does not specify a valid media adapter with which the `cl_device_ids` against which this context is to be created may interoperate (only if the [cl_khr_dx9_media_sharing](#) extension is supported).
- `CL_INVALID_OPERATION` if interoperability is specified by setting `CL_CONTEXT_ADAPTER_D3D9_KHR`, `CL_CONTEXT_ADAPTER_D3D9EX_KHR` or `CL_CONTEXT_ADAPTER_DXVA_KHR` to a non-NULL value, and interoperability with another graphics API is also specified (only if the [cl_khr_dx9_media_sharing](#) extension is supported).
- `CL_INVALID_OPERATION` if Direct3D 11 interoperability is specified by setting `CL_INVALID_D3D11_DEVICE_KHR` to a non-NULL value, and interoperability with another graphics API is also specified (only if the [cl_khr_d3d11_sharing](#) extension is supported).
- `CL_INVALID_D3D11_DEVICE_KHR` if the value of the property `CL_CONTEXT_D3D11_DEVICE_KHR` is non-NULL and does not specify a valid Direct3D 11 device with which the `cl_device_ids` against which this context is to be created may interoperate (only if the [cl_khr_d3d11_sharing](#) extension is supported).
- `CL_INVALID_D3D11_DEVICE_KHR` if the Direct3D 11 device specified for interoperability is not compatible with the devices against which the context is to be created (only if the [cl_khr_d3d11_sharing](#) extension is supported).

Specification

 [OpenCL Specification](#)

Also see

[clGetDeviceIDs](#), [clCreateContextFromType](#), [clRetainContext](#), [clReleaseContext](#), [clGetContextInfo](#), [Cardinality Diagram](#)



Copyright © 2007-2011 The Khronos Group Inc. Permission is hereby granted, free of charge, to any person obtaining a copy of this software and/or associated documentation files (the "Materials"), to deal in the Materials without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Materials, and to permit persons to whom the Materials are furnished to do so, subject to the condition that this copyright notice and permission notice shall be included in all copies or substantial portions of the Materials.