[NCCL](index.html)

[2.25](https://docs.nvidia.com/deeplearning/sdk/nccl-archived/index.html)

    * [ncclGetLastError](api/comms.html#ncclgetlasterror)
    * [ncclGetErrorString](api/comms.html#ncclgeterrorstring)
    * [ncclGetVersion](api/comms.html#ncclgetversion)
    * [ncclGetUniqueId](api/comms.html#ncclgetuniqueid)
    * [ncclCommInitRank](api/comms.html#ncclcomminitrank)
    * [ncclCommInitAll](api/comms.html#ncclcomminitall)
    * [ncclCommInitRankConfig](api/comms.html#ncclcomminitrankconfig)
    * [ncclCommInitRankScalable](api/comms.html#ncclcomminitrankscalable)
    * [ncclCommSplit](api/comms.html#ncclcommsplit)
    * [ncclCommFinalize](api/comms.html#ncclcommfinalize)
    * [ncclCommDestroy](api/comms.html#ncclcommdestroy)
    * [ncclCommAbort](api/comms.html#ncclcommabort)
    * [ncclCommGetAsyncError](api/comms.html#ncclcommgetasyncerror)
    * [ncclCommCount](api/comms.html#ncclcommcount)
    * [ncclCommCuDevice](api/comms.html#ncclcommcudevice)
    * [ncclCommUserRank](api/comms.html#ncclcommuserrank)
    * [ncclCommRegister](api/comms.html#ncclcommregister)
    * [ncclCommDeregister](api/comms.html#ncclcommderegister)
    * [ncclMemAlloc](api/comms.html#ncclmemalloc)
    * [ncclMemFree](api/comms.html#ncclmemfree)
  * [Collective Communication Functions](api/colls.html)
    * [ncclAllReduce](api/colls.html#ncclallreduce)
    * [ncclBroadcast](api/colls.html#ncclbroadcast)
    * [ncclReduce](api/colls.html#ncclreduce)
    * [ncclAllGather](api/colls.html#ncclallgather)
    * [ncclReduceScatter](api/colls.html#ncclreducescatter)
  * [Group Calls](api/group.html)

  * [Values accepted](env.html#id13)

* [NCCL_IB_ROCE_VERSION_NUM](env.html#nccl-ib-roce-version-num)

  * [Values accepted](env.html#id14)

* [NCCL_IB_SL](env.html#nccl-ib-sl)

  * [Values accepted](env.html#id15)

* [NCCL_IB_TC](env.html#nccl-ib-tc)

  * [Values accepted](env.html#id16)

* [NCCL_IB_FIFO_TC](env.html#nccl-ib-fifo-tc)

  * [Values accepted](env.html#id17)

* [NCCL_IB_RETURN_ASYNC_EVENTS](env.html#nccl-ib-return-async-events)

  * [Values accepted](env.html#id18)

* [NCCL_OOB_NET_ENABLE](env.html#nccl-oob-net-enable)

  * [Values accepted](env.html#id19)

* [NCCL_OOB_NET_IFNAME](env.html#nccl-oob-net-ifname)

  * [Values accepted](env.html#id20)

* [NCCL_UID_STAGGER_THRESHOLD](env.html#nccl-uid-stagger-threshold)

  * [Values accepted](env.html#id21)

* [NCCL_UID_STAGGER_RATE](env.html#nccl-uid-stagger-rate)

  * [Values accepted](env.html#id22)

* [NCCL_NET](env.html#nccl-net)

  * [Values accepted](env.html#id23)

* [NCCL_NET_PLUGIN](env.html#nccl-net-plugin)

  * [Values accepted](env.html#id24)

* [NCCL_TUNER_PLUGIN](env.html#nccl-tuner-plugin)

  * [Values accepted](env.html#id25)

* [NCCL_PROFILER_PLUGIN](env.html#nccl-profiler-plugin)

  * [Values accepted](env.html#id26)

  * [Values accepted](env.html#id39)

* [NCCL_SHM_DISABLE](env.html#nccl-shm-disable)

  * [Values accepted](env.html#id40)

* [NCCL_BUFFSIZE](env.html#nccl-buffsize)

  * [Values accepted](env.html#id41)

* [NCCL_NTHREADS](env.html#nccl-nthreads)

  * [Values accepted](env.html#id42)

* [NCCL_MAX_NCHANNELS](env.html#nccl-max-nchannels)

  * [Values accepted](env.html#id43)

* [NCCL_MIN_NCHANNELS](env.html#nccl-min-nchannels)

  * [Values accepted](env.html#id44)

* [NCCL_CHECKS_DISABLE](env.html#nccl-checks-disable)

  * [Values accepted](env.html#id45)

* [NCCL_CHECK_POINTERS](env.html#nccl-check-pointers)

  * [Values accepted](env.html#id46)

* [NCCL_LAUNCH_MODE](env.html#nccl-launch-mode)

  * [Values accepted](env.html#id47)

* [NCCL_IB_DISABLE](env.html#nccl-ib-disable)

  * [Values accepted](env.html#id48)

* [NCCL_IB_AR_THRESHOLD](env.html#nccl-ib-ar-threshold)

  * [Values accepted](env.html#id49)

* [NCCL_IB_QPS_PER_CONNECTION](env.html#nccl-ib-qps-per-connection)

  * [Values accepted](env.html#id50)

* [NCCL_IB_SPLIT_DATA_ON_QPS](env.html#nccl-ib-split-data-on-qps)

  * [Values accepted](env.html#id51)

* [NCCL_IB_CUDA_SUPPORT](env.html#nccl-ib-cuda-support)

  * [Values accepted](env.html#id52)

* [NCCL_IB_PCI_RELAXED_ORDERING](env.html#nccl-ib-pci-relaxed-ordering)
  * [Values accepted](env.html#id53)
* [NCCL_IB_ADAPTIVE_ROUTING](env.html#nccl-ib-adaptive-routing)
  * [Values accepted](env.html#id54)
* [NCCL_IB_ECE_ENABLE](env.html#nccl-ib-ece-enable)
  * [Values accepted](env.html#id55)
* [NCCL_MEM_SYNC_DOMAIN](env.html#nccl-mem-sync-domain)
  * [Values accepted](env.html#id56)
* [NCCL_CUMEM_ENABLE](env.html#nccl-cumem-enable)
  * [Values accepted](env.html#id57)
* [NCCL_CUMEM_HOST_ENABLE](env.html#nccl-cumem-host-enable)
  * [Values accepted](env.html#id58)
* [NCCL_NET_GDR_LEVEL (formerly NCCL_IB_GDR_LEVEL)](env.html#nccl-net-gdr-level-formerly-nccl-ib-gdr-level)
  * [Values accepted](env.html#id59)
  * [Integer Values (Legacy)](env.html#id60)
* [NCCL_NET_GDR_READ](env.html#nccl-net-gdr-read)
  * [Values accepted](env.html#id61)
* [NCCL_NET_SHARED_BUFFERS](env.html#nccl-net-shared-buffers)
  * [Value accepted](env.html#id62)
* [NCCL_NET_SHARED_COMMS](env.html#nccl-net-shared-comms)
  * [Value accepted](env.html#id63)
* [NCCL_SINGLE_RING_THRESHOLD](env.html#nccl-single-ring-threshold)
  * [Values accepted](env.html#id64)
* [NCCL_LL_THRESHOLD](env.html#nccl-ll-threshold)
  * [Values accepted](env.html#id65)
* [NCCL_TREE_THRESHOLD](env.html#nccl-tree-threshold)

  * [Values accepted](env.html#id66)

* [NCCL_ALGO](env.html#nccl-algo)

  * [Values accepted](env.html#id67)

* [NCCL_PROTO](env.html#nccl-proto)

  * [Values accepted](env.html#id68)

* [NCCL_NVB_DISABLE](env.html#nccl-nvb-disable)

  * [Value accepted](env.html#id69)

* [NCCL_PXN_DISABLE](env.html#nccl-pxn-disable)

  * [Value accepted](env.html#id70)

* [NCCL_P2P_PXN_LEVEL](env.html#nccl-p2p-pxn-level)

  * [Value accepted](env.html#id71)

* [NCCL_RUNTIME_CONNECT](env.html#nccl-runtime-connect)

  * [Value accepted](env.html#id72)

* [NCCL_GRAPH_REGISTER](env.html#nccl-graph-register)

  * [Value accepted](env.html#id74)

* [NCCL_LOCAL_REGISTER](env.html#nccl-local-register)

  * [Value accepted](env.html#id75)

* [NCCL_LEGACY_CUDA_REGISTER](env.html#nccl-legacy-cuda-register)

  * [Value accepted](env.html#id76)

* [NCCL_SET_STACK_SIZE](env.html#nccl-set-stack-size)

  * [Value accepted](env.html#id77)

* [NCCL_GRAPH_MIXING_SUPPORT](env.html#nccl-graph-mixing-support)

  * [Value accepted](env.html#id79)

* [NCCL_DMABUF_ENABLE](env.html#nccl-dmabuf-enable)

  * [Value accepted](env.html#id80)

* [NCCL_P2P_NET_CHUNKSIZE](env.html#nccl-p2p-net-chunksize)

  * [Values accepted](env.html#id81)

* * *

Using NCCLÂ¶

Using NCCL is similar to using any other library in your code:

1. Install the NCCL library on your system
2. Modify your application to link to that library
3. Include the header file nccl.h in your application
4. Create a communicator (see [Creating a Communicator](usage/communicators.html#communicator-label))
5. Use NCCL collective communication primitives to perform data communication. You can familiarize yourself with the [NCCL API](api.html#api-label) documentation to maximize your usage performance.

Collective communication primitives are common patterns of data transfer among a group of CUDA devices. A communication algorithm involves many processors that are communicating together. Each CUDA device is identified within the communication group by a zero-based index or rank. Each rank uses a communicator object to refer to the collection of GPUs that are intended to work together. The creation of a communicator is the first step needed before launching any communication operation.

* [Creating a Communicator](usage/communicators.html)
    * [Creating a communicator with options](usage/communicators.html#creating-a-communicator-with-options)
    * [Creating a communicator using multiple ncclUniqueIds](usage/communicators.html#creating-a-communicator-using-multiple-ncccluniqueids)
    * [Creating more communicators](usage/communicators.html#creating-more-communicators)

* [Using multiple NCCL communicators concurrently](usage/communicators.html#using-multiple-nccl-communicators-concurrently)
    * [Finalizing a communicator](usage/communicators.html#finalizing-a-communicator)
    * [Destroying a communicator](usage/communicators.html#destroying-a-communicator)
* [Error handling and communicator abort](usage/communicators.html#error-handling-and-communicator-abort)
* [Asynchronous errors and error handling](usage/communicators.html#asynchronous-errors-and-error-handling)
  * [Fault Tolerance](usage/communicators.html#fault-tolerance)
  * [Collective Operations](usage/collectives.html)
    * [AllReduce](usage/collectives.html#allreduce)
    * [Broadcast](usage/collectives.html#broadcast)
    * [Reduce](usage/collectives.html#reduce)
    * [AllGather](usage/collectives.html#allgather)
    * [ReduceScatter](usage/collectives.html#reducescatter)
  * [Data Pointers](usage/data.html)
  * [CUDA Stream Semantics](usage/streams.html)
* [Mixing Multiple Streams within the same ncclGroupStart/End() group](usage/streams.html#mixing-multiple-streams-within-the-same-ncclgroupstart-end-group)
  * [Group Calls](usage/groups.html)
* [Management Of Multiple GPUs From One Thread](usage/groups.html#management-of-multiple-gpus-from-one-thread)
* [Aggregated Operations (2.2 and later)](usage/groups.html#aggregated-operations-2-2-and-later)
    * [Nonblocking Group Operation](usage/groups.html#nonblocking-group-operation)
  * [Point-to-point communication](usage/p2p.html)
    * [Sendrecv](usage/p2p.html#sendrecv)

[Next](usage/communicators.html "Creating a Communicator") [
Previous](setup.html "Setup")

* * *