

[ NCCL ]([index.html](#))

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

- \* [Overview of NCCL]([overview.html](#))

- \* [Setup]([setup.html](#))

- \* [Using NCCL]([usage.html](#))

- \* [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-nccluniqueids](#))

- \* [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 \(2.2 and later\)\]\(usage/groups.html#aggregated-operations-2-2-and-later\)](#)
- \* [\[Nonblocking Group Operation\]\(usage/groups.html#nonblocking-group-operation\)](#)
- \* [\[Point-to-point communication\]\(usage/p2p.html\)](#)
- \* [\[Sendrecv\]\(usage/p2p.html#sendrecv\)](#)
- \* [\[One-to-all \(scatter\)\]\(usage/p2p.html#one-to-all-scatter\)](#)
- \* [\[All-to-one \(gather\)\]\(usage/p2p.html#all-to-one-gather\)](#)
- \* [\[All-to-all\]\(usage/p2p.html#all-to-all\)](#)
- \* [\[Neighbor exchange\]\(usage/p2p.html#neighbor-exchange\)](#)
- \* [\[Thread Safety\]\(usage/threadsafety.html\)](#)
- \* [\[In-place Operations\]\(usage/inplace.html\)](#)
- \* [\[Using NCCL with CUDA Graphs\]\(usage/cudagraph.html\)](#)
- \* [\[User Buffer Registration\]\(usage/bufferreg.html\)](#)
  - \* [\[NVLink Sharp Buffer Registration\]\(usage/bufferreg.html#nvlink-sharp-buffer-registration\)](#)
  - \* [\[IB Sharp Buffer Registration\]\(usage/bufferreg.html#ib-sharp-buffer-registration\)](#)
  - \* [\[General Buffer Registration\]\(usage/bufferreg.html#general-buffer-registration\)](#)
  - \* [\[Memory Allocator\]\(usage/bufferreg.html#memory-allocator\)](#)
- \* [\[NCCL API\]\(api.html\)](#)
  - \* [\[Communicator Creation and Management Functions\]\(api/comms.html\)](#)

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

- \* [\[ncclGroupStart\]\(api/group.html#ncclgroupstart\)](#)
- \* [\[ncclGroupEnd\]\(api/group.html#ncclgroupend\)](#)
- \* [\[ncclGroupSimulateEnd\]\(api/group.html#ncclgroupsimulateend\)](#)
- \* [\[Point To Point Communication Functions\]\(api/p2p.html\)](#)
- \* [\[ncclSend\]\(api/p2p.html#ncclsend\)](#)
- \* [\[ncclRecv\]\(api/p2p.html#ncclrecv\)](#)
- \* [\[Types\]\(api/types.html\)](#)
- \* [\[ncclComm\\_t\]\(api/types.html#ncclcomm-t\)](#)
- \* [\[ncclResult\\_t\]\(api/types.html#ncclresult-t\)](#)
- \* [\[ncclDataType\\_t\]\(api/types.html#nccldatatype-t\)](#)
- \* [\[ncclRedOp\\_t\]\(api/types.html#ncclredop-t\)](#)
- \* [\[ncclScalarResidence\\_t\]\(api/types.html#ncclscalarresidence-t\)](#)
- \* [\[ncclConfig\\_t\]\(api/types.html#ncclconfig-t\)](#)
- \* [\[ncclSimInfo\\_t\]\(api/types.html#ncclsiminfo-t\)](#)
- \* [\[User Defined Reduction Operators\]\(api/ops.html\)](#)
- \* [\[ncclRedOpCreatePreMulSum\]\(api/ops.html#ncclredopcreatepremulsum\)](#)
- \* [\[ncclRedOpDestroy\]\(api/ops.html#ncclredopdestroy\)](#)
- \* [\[Migrating from NCCL 1 to NCCL 2\]\(nccl1.html\)](#)
- \* [\[Initialization\]\(nccl1.html#initialization\)](#)
- \* [\[Communication\]\(nccl1.html#communication\)](#)
- \* [\[Counts\]\(nccl1.html#counts\)](#)
- \* [\[In-place usage for AllGather and ReduceScatter\]\(nccl1.html#in-place-usage-for-allgather-and-reducescatter\)](#)
- \* [\[AllGather arguments order\]\(nccl1.html#allgather-arguments-order\)](#)
- \* [\[Datatypes\]\(nccl1.html#datatypes\)](#)
- \* [\[Error codes\]\(nccl1.html#error-codes\)](#)
- \* [\[Examples\]\(examples.html\)](#)

\* [Communicator Creation and Destruction

Examples](examples.html#communicator-creation-and-destruction-examples)

\* [Example 1: Single Process, Single Thread, Multiple

Devices](examples.html#example-1-single-process-single-thread-multiple-devices)

\* [Example 2: One Device per Process or

Thread](examples.html#example-2-one-device-per-process-or-thread)

\* [Example 3: Multiple Devices per

Thread](examples.html#example-3-multiple-devices-per-thread)

\* [Example 4: Multiple communicators per

device](examples.html#example-4-multiple-communicators-per-device)

\* [Communication Examples](examples.html#communication-examples)

\* [Example 1: One Device per Process or

Thread](examples.html#example-1-one-device-per-process-or-thread)

\* [Example 2: Multiple Devices per

Thread](examples.html#example-2-multiple-devices-per-thread)

\* [NCCL and MPI](mpi.html)

\* [API](mpi.html#api)

\* [Using multiple devices per process](mpi.html#using-multiple-devices-per-process)

\* [ReduceScatter operation](mpi.html#reducescatter-operation)

\* [Send and Receive counts](mpi.html#send-and-receive-counts)

\* [Other collectives and point-to-point

operations](mpi.html#other-collectives-and-point-to-point-operations)

\* [In-place operations](mpi.html#in-place-operations)

\* [Using NCCL within an MPI Program](mpi.html#using-nccl-within-an-mpi-program)

\* [MPI Progress](mpi.html#mpi-progress)

\* [Inter-GPU Communication with CUDA-aware

MPI](mpi.html#inter-gpu-communication-with-cuda-aware-mpi)

- \* [Environment Variables](env.html)
- \* [System configuration](env.html#system-configuration)
- \* [NCCL\_SOCKET\_IFNAME](env.html#nccl-socket-ifname)
  - \* [Values accepted](env.html#values-accepted)
- \* [NCCL\_SOCKET\_FAMILY](env.html#nccl-socket-family)
  - \* [Values accepted](env.html#id2)
- \* [NCCL\_SOCKET\_RETRY\_CNT](env.html#nccl-socket-retry-cnt)
  - \* [Values accepted](env.html#id3)
- \* [NCCL\_SOCKET\_RETRY\_SLEEP\_MSEC](env.html#nccl-socket-retry-sleep-msec)
  - \* [Values accepted](env.html#id4)
- \* [NCCL\_SOCKET\_NTHREADS](env.html#nccl-socket-nthreads)
  - \* [Values accepted](env.html#id5)
- \* [NCCL\_NSOCKS\_PERTHREAD](env.html#nccl-nsocks-perthread)
  - \* [Values accepted](env.html#id6)
- \* [NCCL\_CROSS\_NIC](env.html#nccl-cross-nic)
  - \* [Values accepted](env.html#id7)
- \* [NCCL\_IB\_HCA](env.html#nccl-ib-hca)
  - \* [Values accepted](env.html#id8)
- \* [NCCL\_IB\_TIMEOUT](env.html#nccl-ib-timeout)
  - \* [Values accepted](env.html#id9)
- \* [NCCL\_IB\_RETRY\_CNT](env.html#nccl-ib-retry-cnt)
  - \* [Values accepted](env.html#id10)
- \* [NCCL\_IB\_GID\_INDEX](env.html#nccl-ib-gid-index)
  - \* [Values accepted](env.html#id11)
- \* [NCCL\_IB\_ADDR\_FAMILY](env.html#nccl-ib-addr-family)
  - \* [Values accepted](env.html#id12)
- \* [NCCL\_IB\_ADDR\_RANGE](env.html#nccl-ib-addr-range)

\* [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)

- \* [NCCL\_IGNORE\_CPU\_AFFINITY](env.html#nccl-ignore-cpu-affinity)
  - \* [Values accepted](env.html#id27)
- \* [NCCL\_CONF\_FILE](env.html#nccl-conf-file)
  - \* [Values accepted](env.html#id28)
- \* [NCCL\_DEBUG](env.html#nccl-debug)
  - \* [Values accepted](env.html#id30)
- \* [NCCL\_DEBUG\_FILE](env.html#nccl-debug-file)
  - \* [Values accepted](env.html#id31)
- \* [NCCL\_DEBUG\_SUBSYS](env.html#nccl-debug-subsys)
  - \* [Values accepted](env.html#id32)
- \* [NCCL\_COLLNET\_ENABLE](env.html#nccl-collnet-enable)
  - \* [Value accepted](env.html#value-accepted)
- \* [NCCL\_COLLNET\_NODE\_THRESHOLD](env.html#nccl-collnet-node-threshold)
  - \* [Value accepted](env.html#id33)
- \* [NCCL\_TOPO\_FILE](env.html#nccl-topo-file)
  - \* [Value accepted](env.html#id34)
- \* [NCCL\_TOPO\_DUMP\_FILE](env.html#nccl-topo-dump-file)
  - \* [Value accepted](env.html#id35)
- \* [NCCL\_SET\_THREAD\_NAME](env.html#nccl-set-thread-name)
  - \* [Value accepted](env.html#id36)
- \* [Debugging](env.html#debugging)
- \* [NCCL\_P2P\_DISABLE](env.html#nccl-p2p-disable)
  - \* [Values accepted](env.html#id37)
- \* [NCCL\_P2P\_LEVEL](env.html#nccl-p2p-level)
  - \* [Values accepted](env.html#id38)
  - \* [Integer Values (Legacy)](env.html#integer-values-legacy)
- \* [NCCL\_P2P\_DIRECT\_DISABLE](env.html#nccl-p2p-direct-disable)



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

\* [NCCL\_SHM\_DISABLE](env.html#nccl-shm-disable)

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

\* [NCCL\_BUFFSIZE](env.html#nccl-buffersize)

\* [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)

- \* [NCCL\_P2P\_LL\_THRESHOLD](env.html#nccl-p2p-ll-threshold)
  - \* [Values accepted](env.html#id82)
- \* [NCCL\_ALLOC\_P2P\_NET\_LL\_BUFFERS](env.html#nccl-alloc-p2p-net-ll-buffers)
  - \* [Values accepted](env.html#id83)
- \* [NCCL\_COMM\_BLOCKING](env.html#nccl-comm-blocking)
  - \* [Values accepted](env.html#id84)
- \* [NCCL\_CGA\_CLUSTER\_SIZE](env.html#nccl-cga-cluster-size)
  - \* [Values accepted](env.html#id85)
- \* [NCCL\_MAX\_CTAS](env.html#nccl-max-ctas)
  - \* [Values accepted](env.html#id86)
- \* [NCCL\_MIN\_CTAS](env.html#nccl-min-ctas)
  - \* [Values accepted](env.html#id87)
- \* [NCCL\_NVLS\_ENABLE](env.html#nccl-nvls-enable)
  - \* [Values accepted](env.html#id88)
- \* [NCCL\_IB\_MERGE\_NICS](env.html#nccl-ib-merge-nics)
  - \* [Values accepted](env.html#id89)
- \* [NCCL\_MNNVL\_ENABLE](env.html#nccl-mnnvl-enable)
  - \* [Values accepted](env.html#id90)
- \* [NCCL\_RAS\_ENABLE](env.html#nccl-ras-enable)
  - \* [Values accepted](env.html#id91)
- \* [NCCL\_RAS\_ADDR](env.html#nccl-ras-addr)
  - \* [Values accepted](env.html#id92)
- \* [NCCL\_RAS\_TIMEOUT\_FACTOR](env.html#nccl-ras-timeout-factor)
  - \* [Values accepted](env.html#id93)

## \* Troubleshooting

### \* Errors

### \* RAS

- \* [\[RAS\]\(troubleshooting/ras.html\)](#)
- \* [\[Principle of Operation\]\(troubleshooting/ras.html#principle-of-operation\)](#)
- \* [\[RAS Queries\]\(troubleshooting/ras.html#ras-queries\)](#)
- \* [\[Sample Output\]\(troubleshooting/ras.html#sample-output\)](#)
- \* GPU Direct
- \* GPU-to-GPU communication
- \* GPU-to-NIC communication
- \* PCI Access Control Services (ACS)
- \* Topology detection
- \* Shared memory
- \* Docker
- \* Systemd
- \* Networking issues
- \* IP Network Interfaces
- \* IP Ports
- \* InfiniBand
- \* RDMA over Converged Ethernet (RoCE)

[\\_\\_\[NCCL\]\(index.html\)](#)

- \* [\[Docs\]\(index.html\)](#) »
- \* Troubleshooting
- \* [\[ View page source\]\(\\_sources/troubleshooting.rst.txt\)](#)

\* \* \*

# TroubleshootingÂ¶

Ensure you are familiar with the following known issues and useful debugging strategies.

## ## Errors¶

NCCL calls may return a variety of return codes. Ensure that the return codes are always equal to `ncclSuccess`. If any call fails and returns a value different from `ncclSuccess`, setting `NCCL_DEBUG` to `WARN` will make NCCL print an explicit warning message before returning the error.

Errors are grouped into different categories.

- \* `ncclUnhandledCudaError` and `ncclSystemError` indicate that a call to an external library failed.
- \* `ncclInvalidArgument` and `ncclInvalidUsage` indicates there was a programming error in the application using NCCL.

In either case, refer to the NCCL warning message to understand how to resolve the problem.

## ## RAS¶

Starting with version 2.24, NCCL includes a reliability, availability, and serviceability (RAS) subsystem to help with the diagnosis and debugging of crashes and hangs.

- \* [\[RAS\]\(troubleshooting/ras.html\)](#)

- \* [\[Principle of Operation\]\(troubleshooting/ras.html#principle-of-operation\)](#)
- \* [\[RAS Queries\]\(troubleshooting/ras.html#ras-queries\)](#)
- \* [\[Sample Output\]\(troubleshooting/ras.html#sample-output\)](#)

## ## GPU Direct¶

NCCL heavily relies on GPU Direct for inter-GPU communication. This refers to the ability for a GPU to directly communicate with another device, such as another GPU or a network card, using direct point-to-point PCI messages.

Direct point-to-point PCI messages can fail or perform poorly for a variety of reasons, like missing components, a bad configuration of a virtual machine or a container, or some BIOS settings.

## ### GPU-to-GPU communication¶

To make sure GPU-to-GPU communication is working correctly, look for the `p2pBandwidthLatencyTest` from the CUDA samples.

```
cd /usr/local/cuda/samples/1_Uutilities/p2pBandwidthLatencyTest  
sudo make  
./p2pBandwidthLatencyTest
```

The test should run to completion and report good performance between GPUs.

Another tool for checking GPU-to-GPU performance is called `nvbandwidth`. This can be downloaded and built from the code and instructions found here:

<<https://github.com/NVIDIA/nvbandwidth>>

### ### GPU-to-NIC communication¶

GPUs can also communicate directly with network cards using GPU Direct RDMA (GDRDMA). This requires having a compatible network cards and drivers, plus loading an extra kernel module called `nvidia-peermem`. The `nvidia-peermem` module is now supplied with the CUDA drivers, however it must be loaded on each node boot with:

```
sudo modprobe nvidia-peermem
```

GDRDMA can also be enabled by using the DMA-BUF feature of recent Linux kernels combined with the Open Source Nvidia GPU driver. In this case, NCCL will automatically detect and enable DMA-BUF so the nvidia-peermem module will not be necessary.

### ### PCI Access Control Services (ACS)¶

**\*\*Baremetal systems\*\***



IO virtualization (also known as VT-d or IOMMU) can interfere with GPU Direct by redirecting all PCI point-to-point traffic to the CPU root complex, causing a significant performance reduction or even a hang. You can check whether ACS is enabled on PCI bridges by running:

```
sudo lspci -vvv | grep ACSctl
```

If lines show `â€œSrcValid+â€•`, then ACS might be enabled. Looking at the full output of `lspci`, one can check if a PCI bridge has ACS enabled.

```
sudo lspci -vvv
```

If PCI switches have ACS enabled, it needs to be disabled. On some systems this can be done from the BIOS by disabling IO virtualization or VT-d. For Broadcom PLX devices, it can be done from the OS but needs to be done again after each reboot.

Use the command below to find the PCI bus IDs of PLX PCI bridges:

```
sudo lspci | grep PLX
```

Next, use `setpci` to disable ACS with the command below, replacing 03:00.0 by the PCI bus ID of each PCI bridge.

```
sudo setpci -s 03:00.0 ECAP_ACS+0x6.w=0000
```

Or you can use a script similar to this:

```
for BDF in `lspci -d "*:*:*" | awk '{print $1}'`; do
    # skip if it doesn't support ACS
    sudo setpci -v -s ${BDF} ECAP_ACS+0x6.w > /dev/null 2>&1
    if [ $? -ne 0 ]; then
        continue
    fi
    sudo setpci -v -s ${BDF} ECAP_ACS+0x6.w=0000
done
```

**\*\*Virtual machines\*\***

Virtual machines require ACS to function, hence disabling ACS is not an option. To run with maximum performance inside virtual machines, ATS needs to be enabled in network adapters.

## ## Topology detection¶

NCCL relies on /sys to discover the PCI topology of GPUs and network cards. When running inside a virtual machine or container, make sure /sys is properly mounted. Having /sys expose a virtual PCI topology can result in sub-optimal performance.

## ## Shared memory¶

To communicate between processes and even between threads of a process, NCCL creates shared memory segments in /dev/shm. The operating system's limits on these resources may need to be increased accordingly. Please see your system's documentation for details.

If insufficient shared memory is available, NCCL will fail to initialize.

Running with NCCL\_DEBUG=WARN will show a message similar to this:

```
NCCL WARN Error: failed to extend /dev/shm/nccl-03v824 to 4194660 bytes
```

## ### Docker¶

In particular, Docker containers default to limited shared and pinned memory resources. When using NCCL inside a container, please make sure to adjust the shared memory size inside the container, for example by adding the following arguments to the docker launch command line:

```
--shm-size=1g --ulimit memlock=-1
```

### Systemd

When running jobs using mpirun or SLURM, systemd may remove files in shared memory when it detects that the corresponding user is not logged in, in an attempt to clean up old temporary files. This can cause NCCL to crash during init with an error like:

```
NCCL WARN unlink shared memory /dev/shm/nccl-d5rTd0 failed, error: No such file or directory
```

Given mpirun and SLURM jobs can run on the node without the user being seen as logged in by systemd, system administrators need to disable that clean-up mechanism, which can be performed by SLURM epilogue scripts instead. To do this, the following line needs to be set in `/etc/systemd/logind.conf`:

```
RemoveIPC=no
```

Once updated, the daemons should be restarted with:

```
sudo systemctl restart systemd-logind
```

## ## Networking issues¶

### ### IP Network Interfaces¶

NCCL auto-detects which network interfaces to use for inter-node communication. If some interfaces are in the UP state but are not able to communicate between nodes, NCCL may try to use them anyway and therefore fail during the init functions or even hang.

For information about how to specify which interfaces to use, see the Environment Variables section, particularly the `NCCL_SOCKET_IFNAME` environment variable.

### ### IP Ports¶

NCCL opens TCP ports to connect processes together and exchange connection information. To restrict the range of ports used by NCCL, one can set the ``net.ipv4.ip_local_port_range`` property of the Linux kernel.

This example shows how to restrict NCCL ports to 50000-51000:

```
echo 50000 51000 > /proc/sys/net/ipv4/ip_local_port_range
```

Or to make this permanent, add a line to `/etc/sysctl.conf`:

```
echo "net.ipv4.ip_local_port_range = 50000 51000" >> /etc/sysctl.conf
```

Restricting the port range can be useful to open a corresponding range in the firewall, for example on Google Cloud:

```
gcloud compute --project=myproject firewall-rules create ncclnet0-ingress --direction=INGRESS  
--priority=1 --network=ncclnet --action=ALLOW --rules=tcp:50000-51000,22,1024-1039  
--destination-ranges=0.0.0.0/0 --target-tags=ncclnet
```

### InfiniBand¶

Before running NCCL on InfiniBand, running low-level InfiniBand tests (and in particular the `ib_write_bw` test) can help verify whether the nodes are able to communicate properly.

A common issue seen with InfiniBand is the library not being able to register sufficient pinned memory. In such cases you may see an error like:

```
NCCL WARN Call to ibv_create_qp failed
```

or

```
NCCL WARN Call to ibv_reg_mr failed
```

The solution is to remove the user limits on registering pinned memory. This can be done by adding these lines:

- \* soft memlock unlimited
- \* hard memlock unlimited

To the `/etc/security/limits.conf` configuration file or equivalent on your Linux distribution.

### ### RDMA over Converged Ethernet (RoCE)¶

Before running NCCL on RoCE, running low-level RDMA tests (and in particular the `ib_write_bw` test) can help verify whether the nodes are able to communicate properly.

A common issue seen with RoCE is the incorrect GID Index being selected for the RoCE v2 NICs. This can result in the following error:

```
NCCL WARN Call to ibv_modify_qp failed with error Invalid argument
```

With NCCL 2.21 and later the GID index is dynamically selected, but with prior versions the user would need to run:



show\_gids

And then set `NCCL\_IB\_GID\_INDEX` to the GID INDEX for the RoCE v2 VER GID.

With NCCL 2.21 and later releases, this environment variable should `_not_` be set.

Users may also need to set `NCCL\_IB\_TC` when using RoCE based networks. Refer to your vendor's documentation for the values this should be set to.

[Next ]([troubleshooting/ras.html](#) "RAS") [ Previous]([env.html](#) "Environment Variables")

\* \* \*

(C) Copyright 2020, NVIDIA Corporation

Built with [Sphinx](<http://sphinx-doc.org/>) using a [theme]([https://github.com/rtfd/sphinx\\_rtd\\_theme](https://github.com/rtfd/sphinx_rtd_theme)) provided by [Read the Docs](<https://readthedocs.org>).