# DAOS Troubleshooting

## DAOS Errors

DAOS has its own error numbering that starts at 1000. The most common errors are documented in the table below.

|  |  |  |
| --- | --- | --- |
| DAOS Error | Value | Description |
| DER\_NO\_PERM | 1001 | No permission |
| DER\_NO\_HDL | 1002 | Invalid handle |
| DER\_INVAL | 1003 | Invalid parameters |
| DER\_NOSPACE | 1007 | No space left on storage target |
| DER\_NOSYS | 1010 | Function not implemented |
| DER\_IO | 2001 | Generic I/O error |
| DER\_ENOENT | 2003 | Entry not found |
| DER\_KEY2BIG | 2012 | Key is too large |
| DER\_IO\_INVAL | 2014 | IO buffers can't match object extents |

When an operation fails, DAOS returns a negative DER error. For a full list of errors, please check <https://github.com/daos-stack/cart/blob/master/src/include/gurt/errno.h> (DER\_ERR\_GURT\_BASE is equal to 1000 and DER\_ERR\_DAOS\_BASE is equal to 2000).

The function d\_errstr() is provided in the API to convert an error number to an error message.

## Debugging System

DAOS uses the debug system defined in [CaRT](https://github.com/daos-stack/cart) but more specifically the GURT library. Log files for both client and server are written to "/tmp/daos.log" unless otherwise set by D\_LOG\_FILE.

### Registered Subsystems/Facilities

The debug logging system includes a series of subsystems or facilities which define groups for related log messages (defined per source file). There are common facilities which are defined in GURT, as well as other facilities that can be defined on a per-project basis (such as those for CaRT and DAOS). DD\_SUBSYS can be used to set which subsystems to enable logging. By default all subsystems are enabled ("DD\_SUBSYS=all").

* DAOS Facilities:   
  common, tree, vos, client, server, rdb, pool, container, object, placement, rebuild, tier, mgmt, bio, tests
* Common Facilities (GURT):   
  MISC, MEM
* CaRT Facilities:   
  RPC, BULK, CORPC, GRP, LM, HG, PMIX, ST, IV

### Priority Logging

All macros that output logs have a priority level, shown in descending order below.

* D\_FATAL(fmt, ...) FATAL
* D\_CRIT(fmt, ...) CRIT
* D\_ERROR(fmt, ...) ERR
* D\_WARN(fmt, ...) WARN
* D\_NOTE(fmt, ...) NOTE
* D\_INFO(fmt, ...) INFO
* D\_DEBUG(mask, fmt, ...) DEBUG

The priority level that outputs to stderr is set with DD\_STDERR. By default in DAOS (specific to the project), this is set to CRIT ("DD\_STDERR=CRIT") meaning that all CRIT and more severe log messages will dump to stderr. This, however, is separate from the priority of logging to "/tmp/daos.log". The priority level of logging can be set with D\_LOG\_MASK, which by default is set to INFO ("D\_LOG\_MASK=INFO"), which will result in all messages excluding DEBUG messages being logged. D\_LOG\_MASK can also be used to specify the level of logging on a per-subsystem basis as well ("D\_LOG\_MASK=DEBUG,MEM=ERR").

### Debug Masks/Streams:

DEBUG messages account for a majority of the log messages, and finer-granularity might be desired. Mask bits are set as the first argument passed in D\_DEBUG(mask, ...). To accomplish this, DD\_MASK can be set to enable different debug streams. Similar to facilities, there are common debug streams defined in GURT, as well as other streams that can be defined on a per-project basis (CaRT and DAOS). All debug streams are enabled by default ("DD\_MASK=all").

* DAOS Debug Masks:
  + md = metadata operations
  + pl = placement operations
  + mgmt = pool management
  + epc = epoch system
  + df = durable format
  + rebuild = rebuild process
  + daos\_default = (group mask) io, md, pl, and rebuild operations
* Common Debug Masks (GURT):
  + any = generic messages, no classification
  + trace = function trace, tree/hash/lru operations
  + mem = memory operations
  + net = network operations
  + io = object I/Otest = test programs

### Common Use Cases

* Generic setup for all messages (default settings)

1. $ D\_LOG\_MASK=DEBUG
2. $ DD\_SUBSYS=all
3. $ DD\_MASK=all

* Disable all logs for performance tuning

1. $ D\_LOG\_MASK=ERR -> will only log error messages from all facilities
2. $ D\_LOG\_MASK=FATAL -> will only log system fatal messages

* Disable a noisy debug logging subsystem

1. $ D\_LOG\_MASK=DEBUG,MEM=ERR -> disables MEM facility by restricting all
2. logs from that facility to ERROR or higher priority only

* Enable a subset of facilities of interest

1. $ DD\_SUBSYS=rpc,tests
2. $ D\_LOG\_MASK=DEBUG -> required to see logs for RPC and TESTS less severe
3. than INFO (the majority of log messages)

* Fine-tune the debug messages by setting a debug mask

1. $ D\_LOG\_MASK=DEBUG
2. $ DD\_MASK=mgmt -> only logs DEBUG messages related to pool management

Refer to the DAOS Environment Variables documentation (Appendix B) for more information about the debug system environment.

## Common DAOS Problems

This section to be updated in a future revision.

## Bug Report

Bugs should be reported through our issue tracker[[1]](#footnote-2) with a test case to reproduce the issue (when applicable) and debug logs.

1. https://jira.hpdd.intel.com [↑](#footnote-ref-2)