# DAOS Debugging

DAOS uses the debug system defined in [CaRT](https://github.com/daos-stack/daos/tree/master/src/cart), specifically the GURT library. Both server and client default log is stdout, unless otherwise set by D\_LOG\_FILE environment variable (client) or log\_file config parameter (server).

## 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 for. By default all subsystems are enabled (DD\_SUBSYS=all). - DAOS Facilities: [common, tree, vos, client, server, rdb, pool, container, object, placement, rebuild, tier, mgmt, eio, tests] - Common Facilities (GURT): [MISC, MEM] - CaRT Facilities: [RPC, BULK, CORPC, GRP, LM, HG, ST, IV]

## Priority Logging

All macros which 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 can be set with DD\_STDERR. By default in DAOS (specific to 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, …). In order 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 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/O - test = test programs

## Common Use Cases

* Generic setup for all messages (default settings)

$ export D\_LOG\_MASK=DEBUG  
 $ export DD\_SUBSYS=all  
 $ export DD\_MASK=all

* Disable all logs for performance tuning

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

* Disable a noisy debug logging subsystem

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

* Enable a subset of facilities of interest

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

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

$ export D\_LOG\_MASK=DEBUG  
 $ export DD\_MASK=mgmt # -> only logs DEBUG messages related to pool management

**See the** [**DAOS Environment Variables**](./environ.md) **documentation for more info about debug system environment.**