

DBX Cheat Sheet

[] indicates optional values

Loading a program

dbx *program_name*

dbx - *process_id*

Running a program

(dbx) run [*args*]

(dbx) run [*args*] [<*input*]
[>*output*]

Examining core files

dbx *program_name core*

dbx - *core*

Breakpoints

(dbx) stop in *function_name*

(dbx) stop at *file:line*

(dbx) status

(dbx) delete *all/#breakpoint*

Stepping

next [*n*]

step [*n*]

cont

Examining the call stack

where

Examining variables

print *expression*

display *expression*

undisplay *expression*

undisplay 0

Examining memory

examine *address* [/ [*count*]
[*format*]

Tracing Execution

Display lines as executed
(dbx) trace event_spec
[*modifier*]

e.g.,

trace step

trace next-in *function*

trace change *variable*

trace in *func* -thread *th_id*

(dbx) dbxenv trace_speed
number

(dbx) trace -file *output_file*

Debugging multithreaded

threads

thread

thread *thread_id*

lwps

Miscellaneous

list [*function_name*]

dump (show local variables)

regs

detach

Walking the call stack

up [-h] [*number*]

down [-h] [*number*]

frame [-h] [*number*]

Runtime checking

(dbx) check -memuse

(dbx) check -access

Generates reports about
memory access/leaks/use

Mismatched cores

e.g., when debugging a core
generated on machine X
(customer) using machine Y
(developer)

(dbx) dbxenv

core_lo_pathmap on

(dbx) pathmap

paths_to_correct_libs

(dbx) pathmap /install/

/local/dbg/install/

(dbx) debug program core

Fix and continue

Dbx allows you to fix code
and continue debugging
without rebuilding/restarting
the debug session. To fix
your code:

1. Change source

2. (dbx) fix

This rebuilds the .o, loads it
in memory and runs till the
fix. This does not change
the exe, so the exe has to be
rebuilt post debugging.

Exit

(dbx) quit

Debugging a Program With dbx

<http://docs.sun.com/source/819-3683/>