To Online or Not Online CXL memory

東京 2025
LINUX PLUMBERS CONFERENCE
TOKYO, JAPAN / DEC. 11-13, 2025

The problem:

```
# daxctl list
[
 {
   "chardev":"dax1.0",
   "size":412316860416,
   "target_node":3,
   "align":2097152,
   "mode":"system-ram",
   "online_memblocks":192,
   "total_memblocks":192,
   "movable":true
 },
]
# daxctl -reconfigure-device -mode=devdax dax1.0
error reconfiguring devices: Device or resource busy
reconfigured 0 devices
#
```

So what happens here?

Daxctl is writing to /sys/devices/system/memory/memXX, waiting for the memory to become off-line

Triggering an uevent

Runs uevent rules

One of the rule is auto-onlining memory

Daxctl is aborting with ‘EBUSY’

Actual problem, first iteration:

Memory is configured via sysfs entries in /sys/devices/system/memory/memXXX

Organized in 'memblocks'

Each 'write' triggers an uevent, which causes udev rules to run.

Each 'open' without 'O_EXCL' is triggering udev to sytheticise a 'change' event.

-> Potential chain of uevents

-> Outcome unpredictable

Actual problem, second iteration:

Sysfs attributes are organized in memblocks

(The example system has 640 memblocks)

Every single sysfs directory for the CXL device needs to be modified

(That is every single of the 192 memblocks)

Every single 'write' might fail, causing 'daxctl' to revert the operation by again 'write' to the sysfs attribute, which in turn again might fail.

Not very robust.

Actual problem, third iteration:

Command-line policy 'auto-movable'

Requires a 'ratio' between Kernel and movable memory.

In the case of CXL: all memory should be 'movable'

At the same time: CXL memory might (and in some cases: will be) hot-pluggable

Causing the admin to 'guess' how much memory will be added over the life-time of the system.

Not a good abstraction. Can we move to specify memory regions here?

東京 2025

LINUX PLUMBERS CONFERENCE

TOKYO, JAPAN / DECEMBER 11-13, 2025