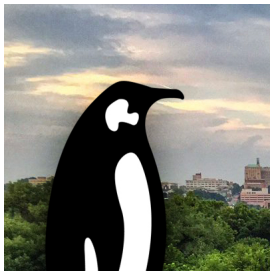

Contribution ID: **232** Type: **not specified**

Shared CXL 3 memory: what will be required?

Monday, 13 November 2023 16:45 (25 minutes)

CXL 3 introduces sharable fabric-attached memory (FAM). I would like to lay out some use cases and lead a discussion as to what functionality will be needed in the cxl and dax stack to make such use cases possible.

This would start with a brief overview of DCD and tagged capacity. Tagged capacity creates a namespace of memory allocations or regions (by tag) that apps can use to find the memory of interest. Sharable tagged capacity is file-like, in that it is a named object that can be memory mapped. It is also pmem-like, in that the contents may already be initialized when an app maps it - and the contents may survive after an app unmaps it. (there are additional cxl 3.1 details that don't belong in this abstract due to confidentiality, but should be un-embargoed by the time of the conference.)

I'll make the case that sharable tagged capacity should never be onlined as system-ram by default, just as pmem should not be configured as system-ram by default. Sharing system-ram memory across hosts is problematic, although it might make sense through a "force" option.

Even though it is possible for apps to mmap data sets in tagged capacity (e.g. /sys/devices/dax/<tag>), I'll argue that dax is not file-like enough to be the complete solution - we'd like to support all apps that can mmap files without forcing them to understand devdax.

Given the need for something "more file-like" than dax, I'll suggest ways that devdax needs to evolve to support a sharable file system sitting on sharable tagged capacity. In particular, devdax will need to inherit the iomap* functionality from the fsdax/pmem support.

Since the vfs layer already supports DAX files via the S_DAX flag, I'll argue that the MVP "famfs" is not all that heavy a lift - with appropriate limitations. I'll present a first cut at an acceptable set of limitations to make famfs practically possible.

I'll also point out some app classes that could adapt readily to shared data sets in shared FAM. In particular, the data science tool chain has many apps and tools that already know how to format data sets to store in files for efficient ability to mmap and use vector instructions without reorganizing data in memory (the "zero copy" formats such as Apache Arrow).

Finally, time permitting, I'll present a brief overview of a famfs prototype that we have developed - which draws heavily on ramfs and hugetlbfs, plus xfs for dax file support.

Primary author: GROVES, John (Micron)

Presenter: GROVES, John (Micron)

Session Classification: Compute Express Link MC

Track Classification: LPC Microconference: Compute Express Link MC