The pNFS RDMA layout

Christoph Hellwig

High-level idea

- If we have byte addressable storage we can turn the usual RDMA model upside down:
  - The server registers regions of memory and hands out handles to it
  - The client can do raw RDMA READ / WRITE operations for the data transfer

device_addr4

- The pNFS device needs to identify an entity to which the memory registrations are bound:
  - At least a “protection domain” in RDMA terms
  - Possibly a specific QP
- Connection management is an open issue:
  - Reuse NFS/RDMA connections?
  - New connection management protocol using RDMA/CM

layouts

- Layouts need to give the client the following information:
  - Device and QP(s) to operate on
  - Handle (R_key / stag) to read / write from
  - Fixed offset into the MR handle
  - File-relative offset
  - Length

layout details

- Client side copy on write processing: For a given layout the READ and WRITE targets might be different.
  - The current draft has multiple “extents” inside a layout, which may overlap for reads vs writes
  - This allows easy code reuse from those layouts, but seems over-complicated
  - Maybe allow separate handles for reads vs writes in the layout itself?

Cache flushing / posted writes

- RDMA Write operations to PCIe MMIO regions require explicit flushing using a read from the device
- Cached mapping on the memory bus might require explicit cache flushes
- Non-cached mappings on the memory bus or upcoming cache coherent interconnects might have different requirements, including no explicit flushing after writes at all

flushing and LAYOUTCOMMIT

- For now all flushing is handled by LAYOUTCOMMIT
  - Works reliably, but involves the MDS for every synchronous write operations
  - LAYOUTCOMMIT should become optional ala flexfiles for cases where no flushing is required
  - We need to support RDMA-native flush operations when the become available

Open issues:

- Connection model and connection management
- Do we need extents below the layout?
- How do we support RDMA-level flush operations without knowing how they are going to look like in detail?
- Do we need hints for the layout management on the client?

Questions?