Improving DMA

2011-10-06 DevSummit WG

Improvements

- The API itself
    - Inconsistent across architectures
    - Fuzzy in behavior
    - Lacking features to guarantee coherency or maximize performance
- Duplication of code

Improvements (cont'd)

- Better memory allocation
- Better alignment handling
- Cached vs. uncached memory
- Generic remapping support
- Locality awareness (NUMA, QPI, DCA)
- Optimize high bandwidth devices

Consequence

- Device tree awareness for tags
- DMA memory allocator:
  - physical memory allocation
  - cache alignment
  - handling of cached/uncached pages
- Add bandwidth hints to tags

Remapping KOBJ

- Used for bounce buffering
- I/O MMUs
- Aperture-based remapping (NUMA)

DMA from user space

- Need mmap(2) support
- VM to interact with DMA allocator?
- POSIX typed memory objects?
- Device pager approach?

Action Items

- Create wiki to document the work
- Collect use cases and requirements
  - Reach out to driver writers
  - Keep virtualization in mind
  - Determine impact of hot plug
- Define KOBJ interface(s) and API

Action Items (cont'd)

- Prototype and test
- Expose on /head (3rd party testing)
- Tie up loose ends and fix bugs
- For 10.0: switch over