Very Large Contiguous Regions in Userspace

Christoph Lameter <cl@linux.com> @qant
Mike Kravetz

Linux Plumbers Conference 2018
Vancouver

Overview

Continuation from 2017 LPC

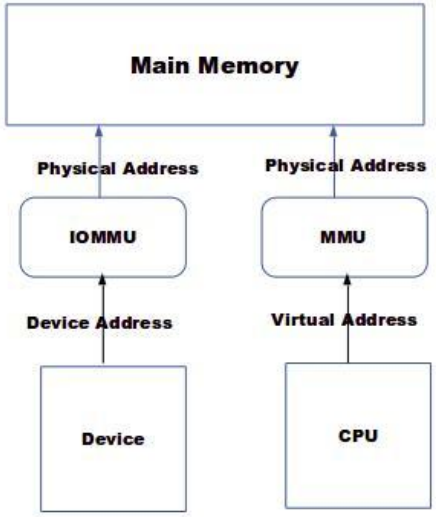
Restatement of issue/requirements

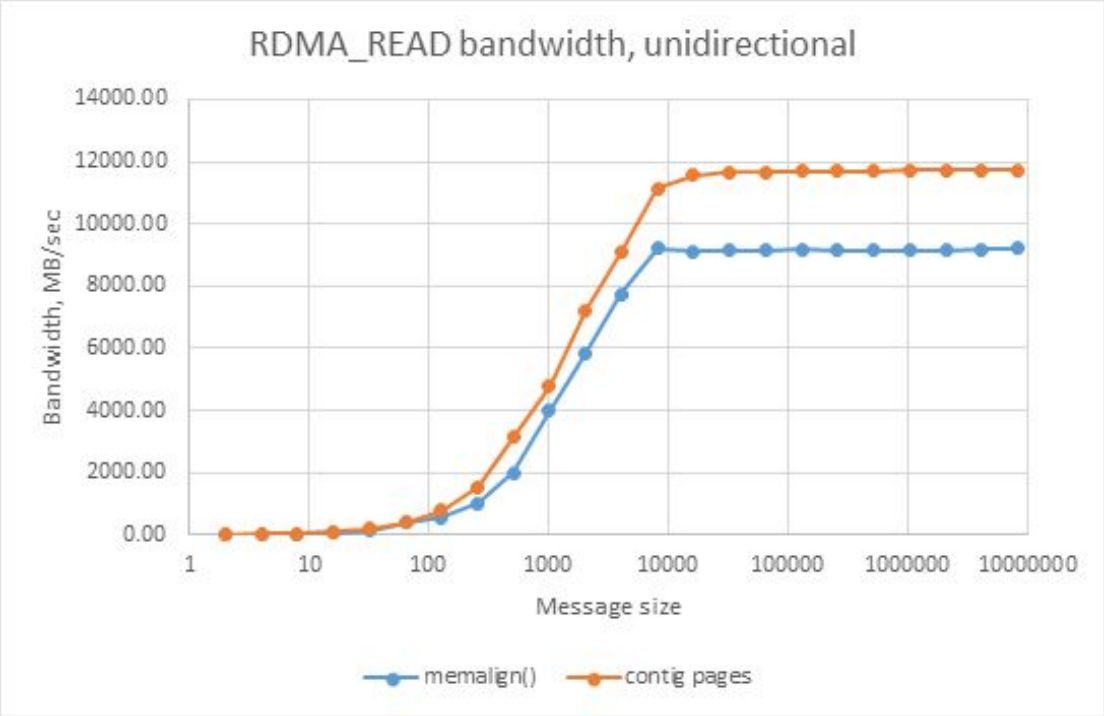
Efforts in area since 2017

Is Fragmented Memory Bad for Us?

- Software Solution:
  - Virtually mapped contiguous areas.
    MMU Maps: Virtual Address → physical address
  - In Linux: Demand paging and reclaim.
- Hardware Solution:
  - IOMMU serves as MMU for devices
  - DMA can do vector I/O
    - Gather data from fragmented memory blocks
    - Scatter data to fragmented memory blocks
    - Hence DMA scatter/gather

So why bother?





ib_read_bw			
size	memalign()	contiguous pages	improvement
2	8.74	10.65	22%
4	14.83	23.47	58%
8	35.64	48.52	36%
16	76.10	96.02	26%
32	141.45	195.92	39%
64	406.52	383.80	-6%
128	543.58	780.68	44%
256	1,018.24	1,545.88	52%
512	2,003.87	3,133.44	56%
1,024	4,000.76	4,761.60	19%
2,048	5,837.46	7,216.94	24%
4,096	7,747.90	9,077.66	17%
8,192	9,224.03	11,140.40	21%
16,384	9,109.37	11,561.70	27%
32,768	9,133.99	11,647.37	28%
65,536	9,133.65	11,662.10	28%
131,072	9,179.69	11,694.01	27%
262,144	9,150.13	11,691.50	28%
524,288	9,149.54	11,706.13	28%
1,048,576	9,149.36	11,714.85	28%
2,097,152	9,161.54	11,715.61	28%
4,194,304	9,176.27	11,716.65	28%
8,388,608	9,199.97	11,716.78	27%

Mellanox ConnectX-5 Ex, EDR, back-to-back
Intel(R) Xeon(R) Gold 6148 CPU @ 2.40GHz
MLNX_OFED_LINUX-4.1-4.0.8.0

Efforts Since Last Year

RFC "Protect larger order pages from breaking up"?

RFC “In Kernel Contig Alloc Interface”

HACK “hugetlbfs extension”

RFC "Protect larger order pages from breaking up"?

In lieu of addressing the defragmentation problem.

Like Huge page reservations certain orders of pages can be protected from being broken up

Controversial hack although in use at a large isp in Europe.

RFC “In Kernel Contig Alloc Interface”

Not CMA

Not directly available for userspace

Came out of last year’s LPC discussions

mmap(MAP_CONTIG) RFC

alloc_contig_pages()

Replicates much of hugetlbfs gigantic page allocator

Not limited to pageblock size as buddy allocator

Uses alloc_contig_range() at low level

Depends on migration of pages/pageblocks

Prefers MIGRATE_MOVEABLE

What about use for DMA?

alloc_contig_pages()

This is hard. See Vlastmil Babka presentation on “The hard work behind large physical allocations in the kernel"

Drivers must use this interface to provide their own interfaces (mmap) to user space

HACK “hugetlbfs extension”

Allocate/reserve large contig areas at boot time

Requires preallocation/reserved memory

Came out of Mellanox RDMA use case

- IB device MMU 2GB pages
- Host CPU 2MB pages