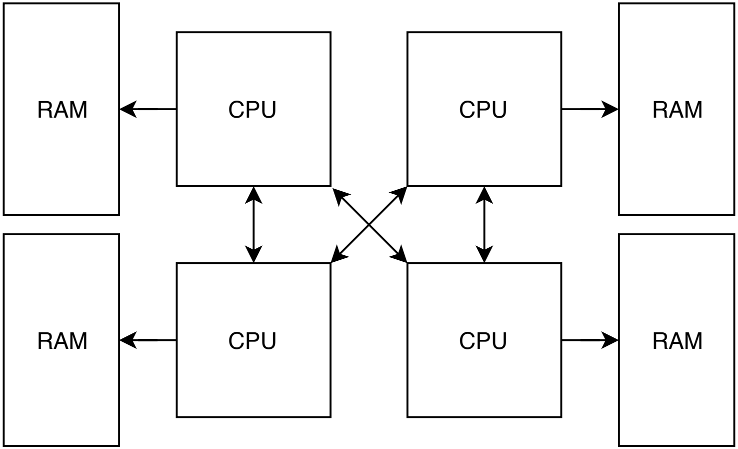
NUMA

Mark Johnston
markj@FreeBSD.org

FreeBSD Bay Area Vendor Summit
October 11, 2019

Non-Uniform Memory Access



OS Responsibilities

Minimize remote memory accesses

- Avoid remote access latency penalty
- Avoid bottlenecking on cross-domain interconnect

Requirements:

- Balance resource utilization
- Allow applications to provide hints (scheduling, memory allocation)
- Handle local memory shortages gracefully
- Affinitize static data structures

APIs

Kernel:

- `bus_get_domain(9)`, `bus_dma_tag_set_domain(9)`
- `malloc_domainset(9)`, `kmem_malloc_domainset(9)`
- `uma_zalloc_domain(9)` (slow!)

Userspace:

- `cpuset(1)`
- `cpuset_getdomain(2)` `cpuset_setdomain(2)`

Review: Domain Selection Policies, `domainset(9)`

`DOMAINSET_POLICY_ROUNDROBIN`

- Cycle through domains: `d = iter++ % ds->ds_cnt`
- `0, 1, 2, 3, 0, 1, 2, 3, 0, ...`

`DOMAINSET_POLICY_FIRSTTOUCH`

- Pick the domain of the current CPU: `d = PCPU_GET(domain)`
- Userland default, good for short-lived processes

`DOMAINSET_POLICY_PREFER`

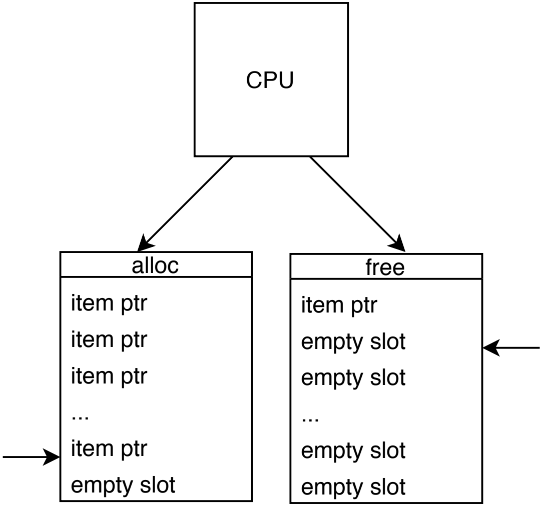
- Pick the domain specified in the policy: `d = ds->ds_prefer`
- Fall back to round-robin when free pages are scarce

`DOMAINSET_POLICY_INTERLEAVE`

- Round-robin with a stride
- `0, 0, ..., 0, 1, 1, ..., 1, 0, 0, ...`
- Superpage-friendly: use a stride of 512
- Kernel default

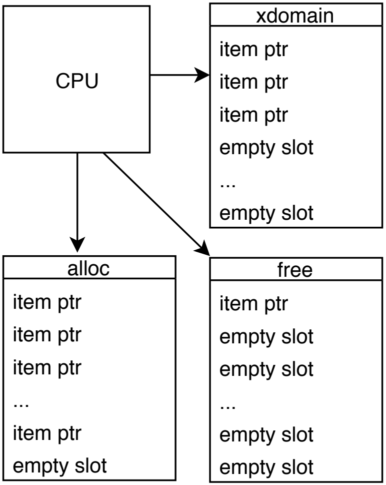
Review: UMA per-CPU caches

- Bucket: dynamically allocated array
- Items allocated from `alloc` bucket
- Items freed to `free` bucket
- Buckets are swapped if empty (alloc) or full (free)
- Per-domain cache of full buckets
- Slow path: lock the zone, check bucket cache

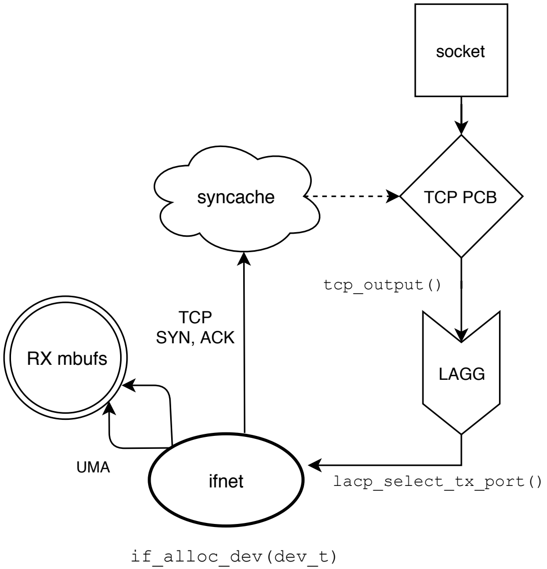


`options UMA_XDOMAIN`

- On free, find item domain
- Cache in `free` if `domain == PCPU_GET(domain)`, else `xdomain`
- Slow path: lock the zone, drain `xdomain`
- Special optimization for 2 domains



Network affinity



vm_page_array (amd64 only)

- One vm_page structure per 4KB page
- vm_page_array allocated early during boot
- Physically contiguous $\rightarrow$ allocated from single domain
- Unfriendly to first-touch allocation policy
- Now backed by “correct” memory, up to 2MB boundaries

FreeBSD

Other Data Structures

- PCPU area (amd64)
- ULE per-CPU thread queues
- callout wheel
- vm_page locks (by removing their usage)
- Kernel thread stacks

Memory-bound pgbench on a 2-socket system, r353116

```
Core | IPC  | Instructions | Cycles   |  Local DRAM accesses | Remote DRAM Accesses
   0   0.46        1097 M    2402 M      1467 K                759 K
   1   0.45        1090 M    2402 M      1464 K                766 K
   2   0.46        1095 M    2402 M      1556 K                801 K
   3   0.46        1096 M    2402 M      1445 K                755 K
   4   0.46        1099 M    2402 M      1507 K                787 K
   5   0.45        1091 M    2402 M      1550 K                813 K
   6   0.46        1099 M    2402 M      1482 K                785 K
   7   0.45        1092 M    2402 M      1509 K                790 K
   8   0.46        1100 M    2402 M      1469 K                771 K
   9   0.45        1090 M    2402 M      1535 K                800 K
  10   0.46        1094 M    2402 M      1585 K                830 K
  11   0.45        1092 M    2402 M      1507 K                777 K
  12   0.46        1099 M    2402 M      1481 K                776 K
  13   0.46        1095 M    2402 M      1482 K                780 K
  14   0.46        1094 M    2402 M      1535 K                793 K
  15   0.45        1092 M    2402 M      1516 K                776 K
  16   0.41         992 M    2402 M       796 K               1256 K
  17   0.41         991 M    2402 M       763 K               1208 K
  18   0.43        1040 M    2402 M       851 K               1365 K
  19   0.43        1034 M    2402 M       860 K               1390 K
  20   0.43        1042 M    2402 M       840 K               1332 K
  21   0.43        1030 M    2402 M       852 K               1404 K
  22   0.43        1035 M    2402 M       857 K               1392 K
  23   0.43        1035 M    2402 M       836 K               1335 K
  24   0.43        1039 M    2402 M       834 K               1341 K
  25   0.43        1034 M    2402 M       830 K               1335 K
  26   0.43        1040 M    2402 M       838 K               1339 K
  27   0.43        1035 M    2402 M       841 K               1335 K
  28   0.43        1040 M    2402 M       835 K               1321 K
  29   0.43        1038 M    2402 M       818 K               1319 K
  30   0.43        1041 M    2402 M       806 K               1269 K
  31   0.43        1031 M    2402 M       831 K               1327 K
```

Future Direction

- Continue affinitizing static kernel data structures
  - e.g., `vm_reserv_array`, `vm_dom[]`
- Taskqueue affinity
- NUMA awareness in UMA by default
- Improve NUMA support on !amd64
- ...?

FreeBSD