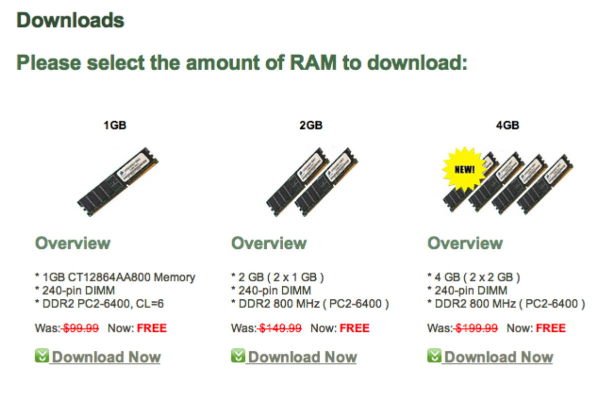
facebook

Linux memory management at scale

Chris Down (github: cdown)
Kernel Engineering, Facebook

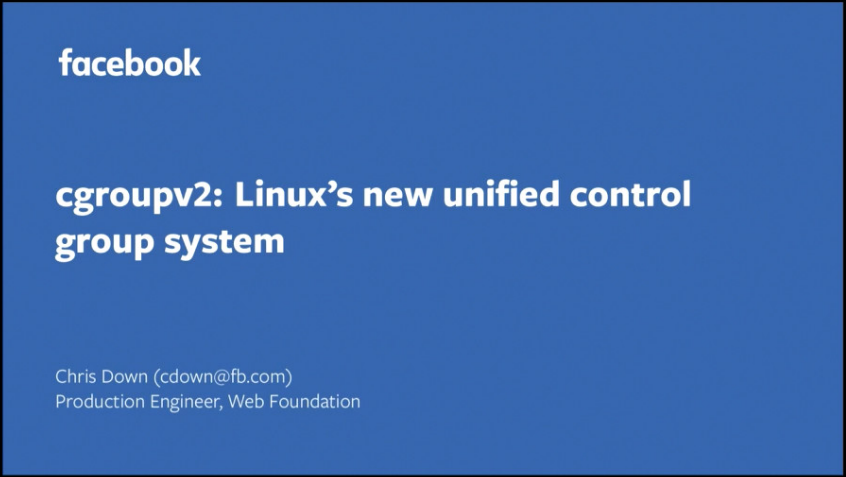


Downloads

Please select the amount of RAM to download:

1GB	2GB	4GB
Overview	**Overview**	**Overview**
* 1GB CT12864AA800 Memory * 240-pin DIMM * DDR2 PC2-6400, CL=6	* 2 GB (2 x 1 GB) * 240-pin DIMM * DDR2 800 MHz (PC2-6400)	* 4 GB (2 x 2 GB) * 240-pin DIMM * DDR2 800 MHz (PC2-6400)
Was: ~~$99.99~~ Now: **FREE**	Was: ~~$149.99~~ Now: **FREE**	Was: ~~$199.99~~ Now: **FREE**
Download Now	Download Now	Download Now

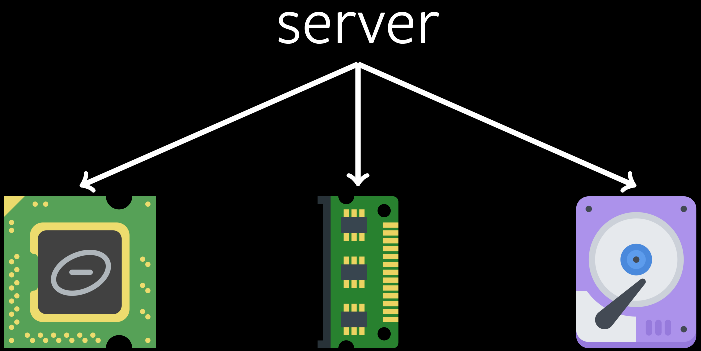
- Give you the knowledge to make better use of memory
- Be able to build more resilient systems through resource control
- Bust some common misconceptions about memory management



facebook

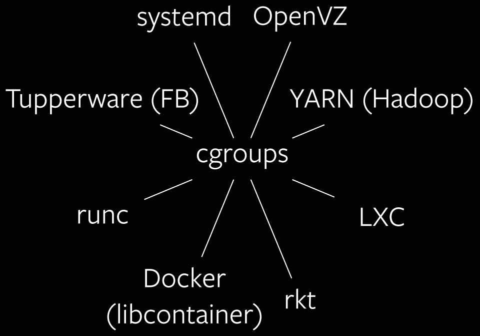
cgroupv2: Linux's new unified control group system

Chris Down (cdown@fb.com)
Production Engineer, Web Foundation



server

Who uses cgroups?



systemd

OpenVZ

Tupperware (FB)

YARN (Hadoop)

cgroups

runc

LXC

Docker (libcontainer)

rkt

Image: Spc. Christopher Hernandez, US Military Public Domain

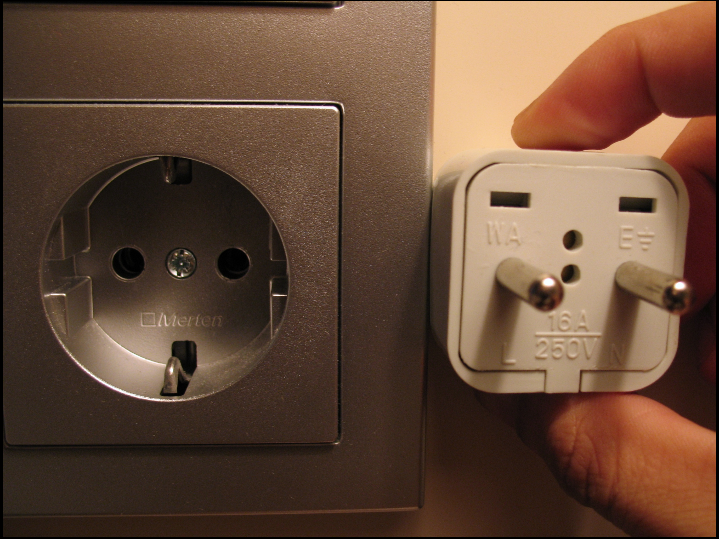
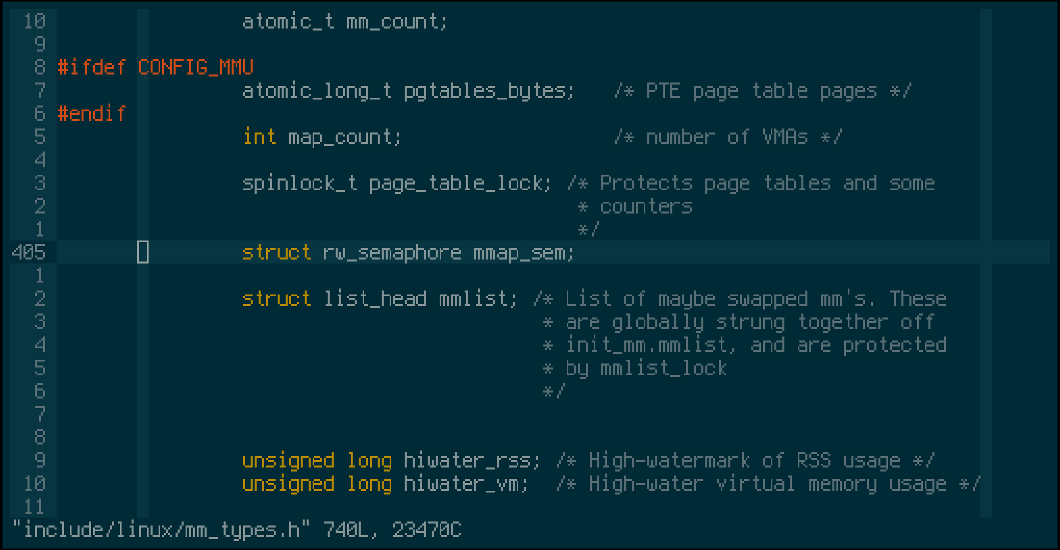
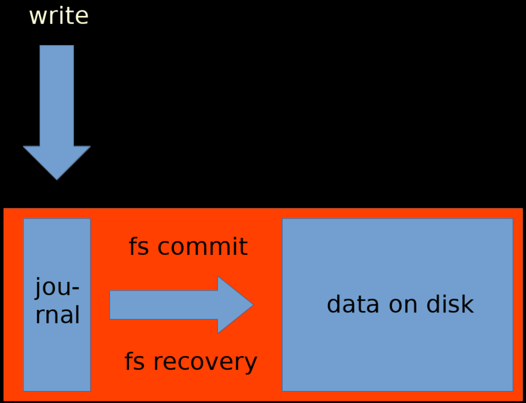


Image: Simon Law on Flickr, CC-BY-SA



```
                atomic_t mm_count;

#ifdef CONFIG_MMU
                atomic_long_t pgtables_bytes;   /* PTE page table pages */
#endif
                int map_count;                  /* number of VMAs */

                spinlock_t page_table_lock; /* Protects page tables and some
                                             * counters
                                             */
                struct rw_semaphore mmap_sem;

                struct list_head mmlist; /* List of maybe swapped mm's. These
                                          * are globally strung together off
                                          * init_mm.mmlist, and are protected
                                          * by mmlist_lock
                                          */


                unsigned long hiwater_rss; /* High-watermark of RSS usage */
                unsigned long hiwater_vm;  /* High-water virtual memory usage */

"include/linux/mm_types.h" 740L, 23470C
```


write
jou-
rnal
fs commit
fs recovery
data on disk

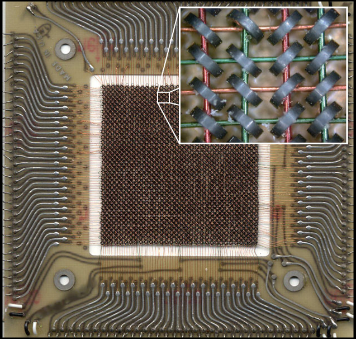
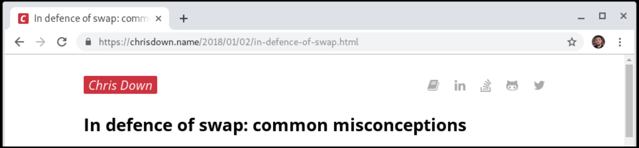


Image: Orion J on Wikimedia Commons, CC-BY

- Memory is divided in to multiple “types”: anon, cache, buffers, sockets, etc
- “Reclaimable” or “unreclaimable” is important, but not guaranteed
- RSS is kinda bullshit, sorry



Chris Down

In defence of swap: common misconceptions

`bit.ly/whyswap`

- Swap isn’t about emergency memory, in fact that’s probably harmful
- Instead, it increases reclaim equality and reliability of forward progress of the system
- Also promotes maintaining a small positive pressure (similar to `make -j cores+1`)

In defence of swap: comm

https://chrisdown.name/2018/01/02/in-defence-of-swap.html

Chris Down

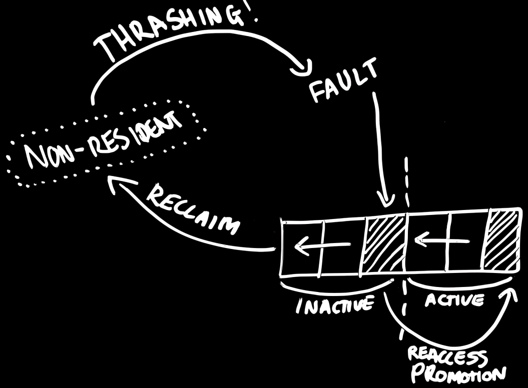
In defence of swap: common misconceptions

`bit.ly/whyswap`

- Swap isn’t about emergency memory, in fact that’s probably harmful
- Instead, it increases reclaim equality and reliability of forward progress of the system
- Also promotes maintaining a small positive pressure (similar to `make -j cores+1`)

Out of memory at line: 1

- OOM killer is reactive, not proactive, based on reclaim failure
- Hotness obscured by MMU (`pte_young`), we don't know we're OOMing ahead of time
- Can be very, very late to the party, and sometimes go to the wrong party entirely



- kswapd reclaim: background, started when resident pages goes above a threshold
- Direct reclaim: blocks application when have no memory available to allocate frames
- Tries to reclaim the coldest pages first
- Some things might not be reclaimable. Swap can help here (`bit.ly/whyswap`)

psi

“If I had more of this resource, I could probably run *N*% faster”

- Find bottlenecks
- Detect workload health issues before they become severe
- Used for resource allocation, load shedding, pre-OOM detection

```
root@web # cat /sys/fs/cgroup/system.slice/memory.pressure
some avg10=0.21 avg60=0.22 avg300=0.19 total=4760988587
full avg10=0.21 avg60=0.22 avg300=0.19 total=4681731696
```

oomd

`bit.ly/fboomd`

- Early-warning OOM detection and handling using new memory pressure metrics
- Highly configurable policy/rule engine
- Workload QoS and context-aware decisions

WARRANTY VOID IF REMOVED

io.latency

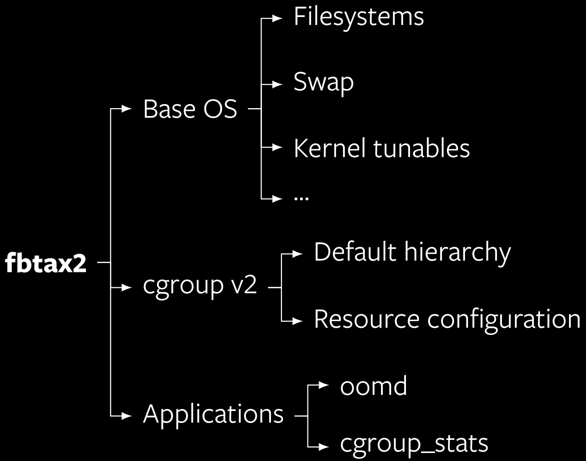
- Best-effort avg (or p90) completion latency guarantee
- More work-conserving — can do as much IO as you like, if you don’t affect others
- Supports do-first-pay-later “credit card” approach

Shift to “protection” mentality

- Limits (eg. memory.{high,max}) really don’t compose well
- Prefer protection (memory.{low,min}) if possible
- Protections affect memory reclaim behaviour

fbtax2

- **Workload protection**: Prevent non-critical services degrading main workload
- **Host protection**: Degrade gracefully if machine cannot sustain workload
- **Usability**: Avoid introducing performance or operational costs



fbtax2

Base OS

Filesystems

Swap

Kernel tunables

...

cgroup v2

Default hierarchy

Resource configuration

Applications

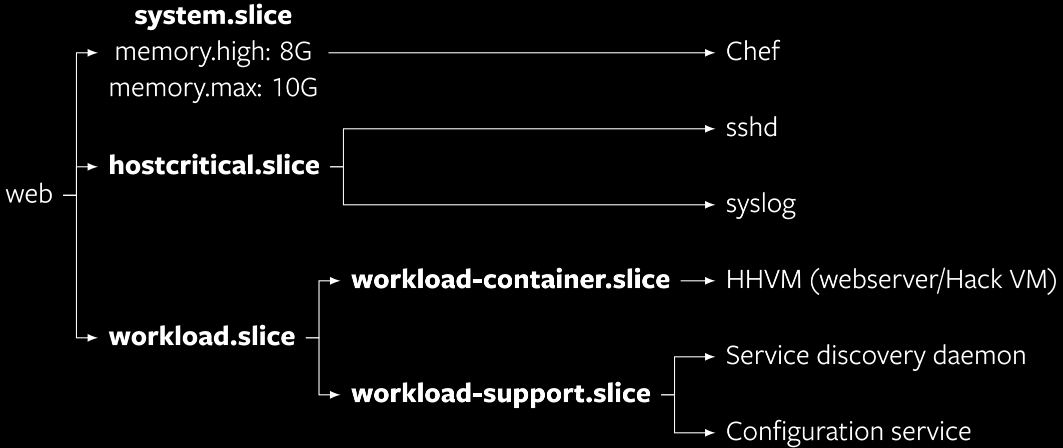
oomd

cgroup_stats

Base OS

- btrfs as /
  - ext4 has priority inversions
  - All metadata is annotated
- Swap
  - Yes, you really still want it (`bit.ly/whyswap`)
  - Allows memory pressure to build up gracefully
  - Usually disabled on main workload
  - btrfs swap file support to avoid tying to provisioning
- Kernel tunables
  - `vm.swappiness`
  - Writeback throttling (wbt)

fbtax2 cgroup hierarchy: old



web

system.slice
memory.high: 8G
memory.max: 10G

Chef

hostcritical.slice

sshd

syslog

workload.slice

workload-container.slice

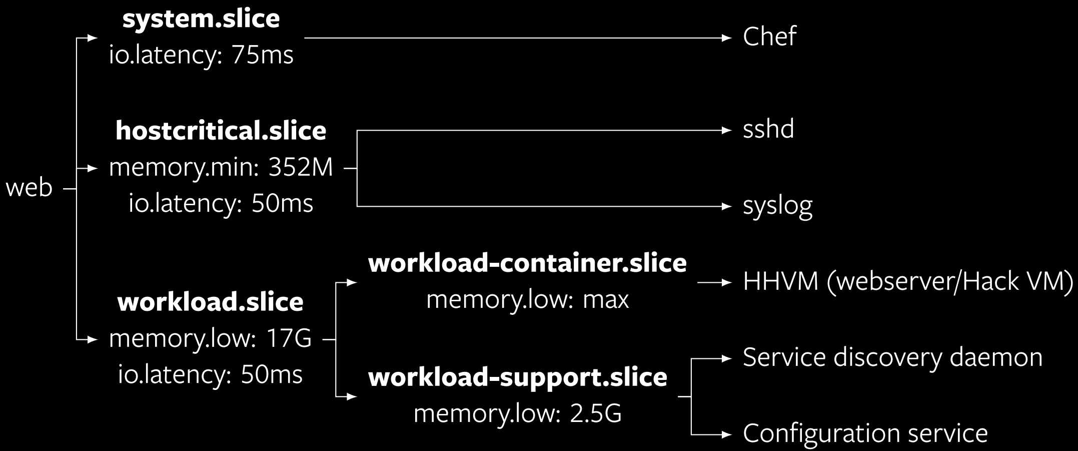
HHVM (webserver/Hack VM)

workload-support.slice

Service discovery daemon

Configuration service

fbtax2 cgroup hierarchy



web

system.slice
io.latency: 75ms

Chef

hostcritical.slice
memory.min: 352M
io.latency: 50ms

sshd

syslog

workload.slice
memory.low: 17G
io.latency: 50ms

workload-container.slice
memory.low: max

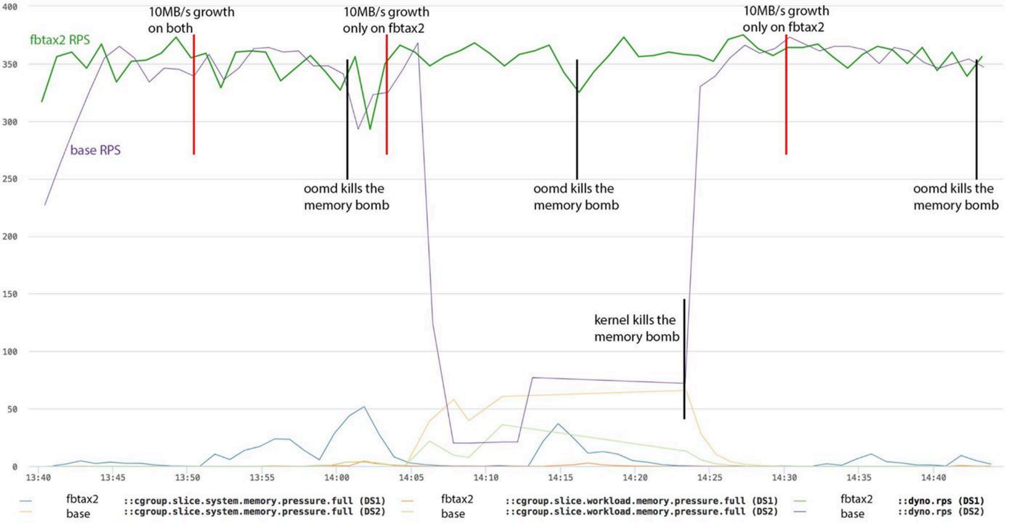
HHVM (webserver/Hack VM)

workload-support.slice
memory.low: 2.5G

Service discovery daemon

Configuration service

webservers: protection against memory starvation



Try it out: `bit.ly/fbtax2`

facebook