Android: memcg v1 -> v2

T.J. Mercier
(tjmercier@google.com)

What: Upgrade memcg from v1 to v2

Memcg accounts system memory use, and can limit memory on a per-cgroup basis

Why:

- 2 > 1
- dma-buf accounting (replace DMABUF_SYSFS_STATS)
- Proactive reclaim for background / cached apps (memory.reclaim)
- Per-application memory limits and reclaim protection (memory.min/low/high/max)
- Lower refaults: reclaim locality with MGLRU
    - When MGLRU and memcg are enabled, a LRU of memcgs is used during global reclaim instead of a scan across all memcgs.

Issues:

- Zombie cgroups (*mostly* solved)
  - https://lwn.net/Articles/932070/
  - https://lore.kernel.org/all/20230615234806.3390147-1-tjmercier@google.com/ (cgroup_css_set_fork)
  - Pixel driver bugs
  - https://lore.kernel.org/all/CABdmKX3SOXpcK85a7cx3iXrwUj=i1yXqEz9i9zNkx8mB=ZXQ8A@mail.gmail.com/ (cgroup.procs)
- Kswapd spinning without progress (solved)
  - https://lore.kernel.org/all/20230814151636.1639123-1-tjmercier@google.com/

Issues:

120 MB slab memory use increase vs no memcg (under investigation)

TODO:

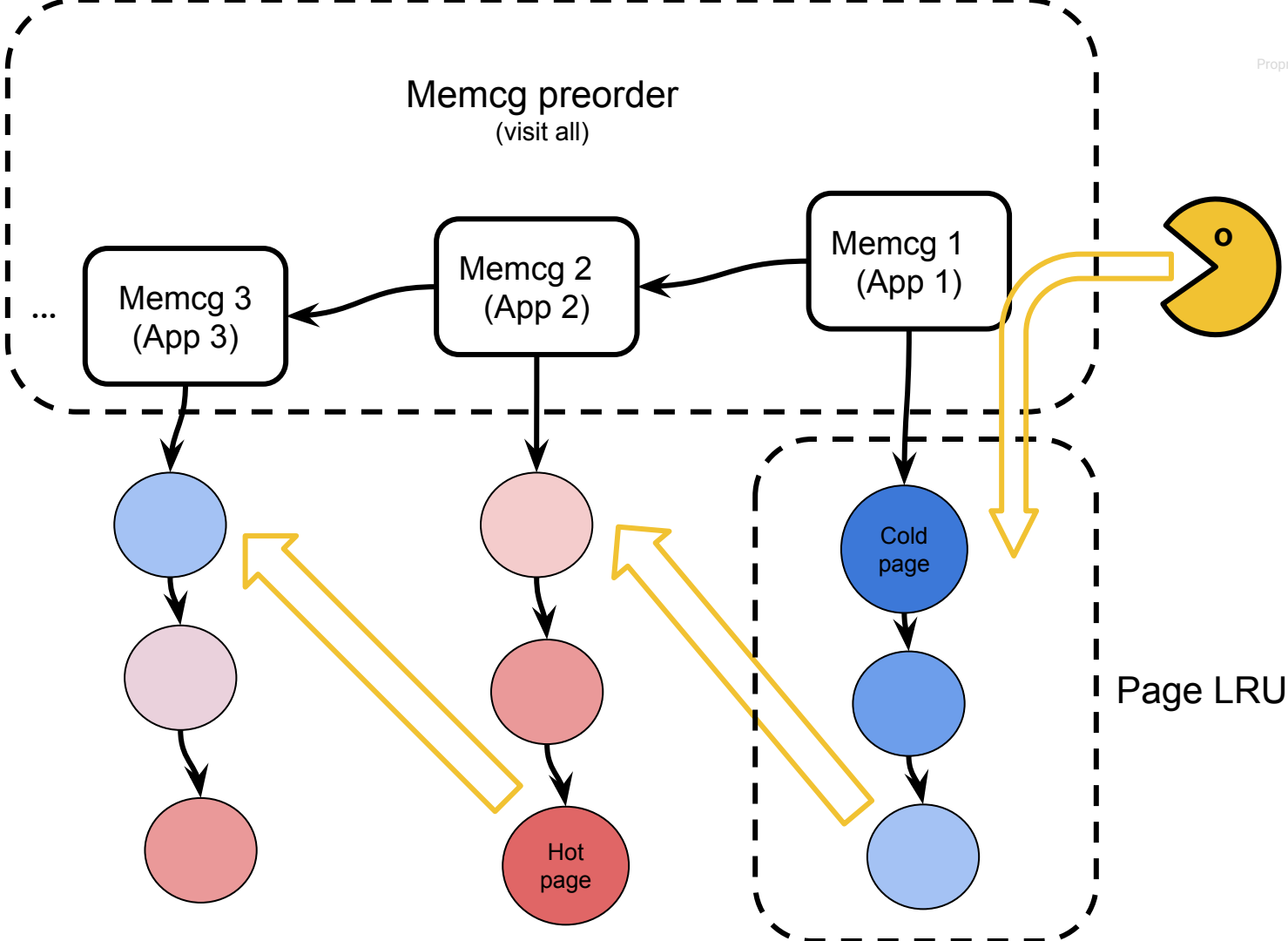
Perf testing on 6.1+

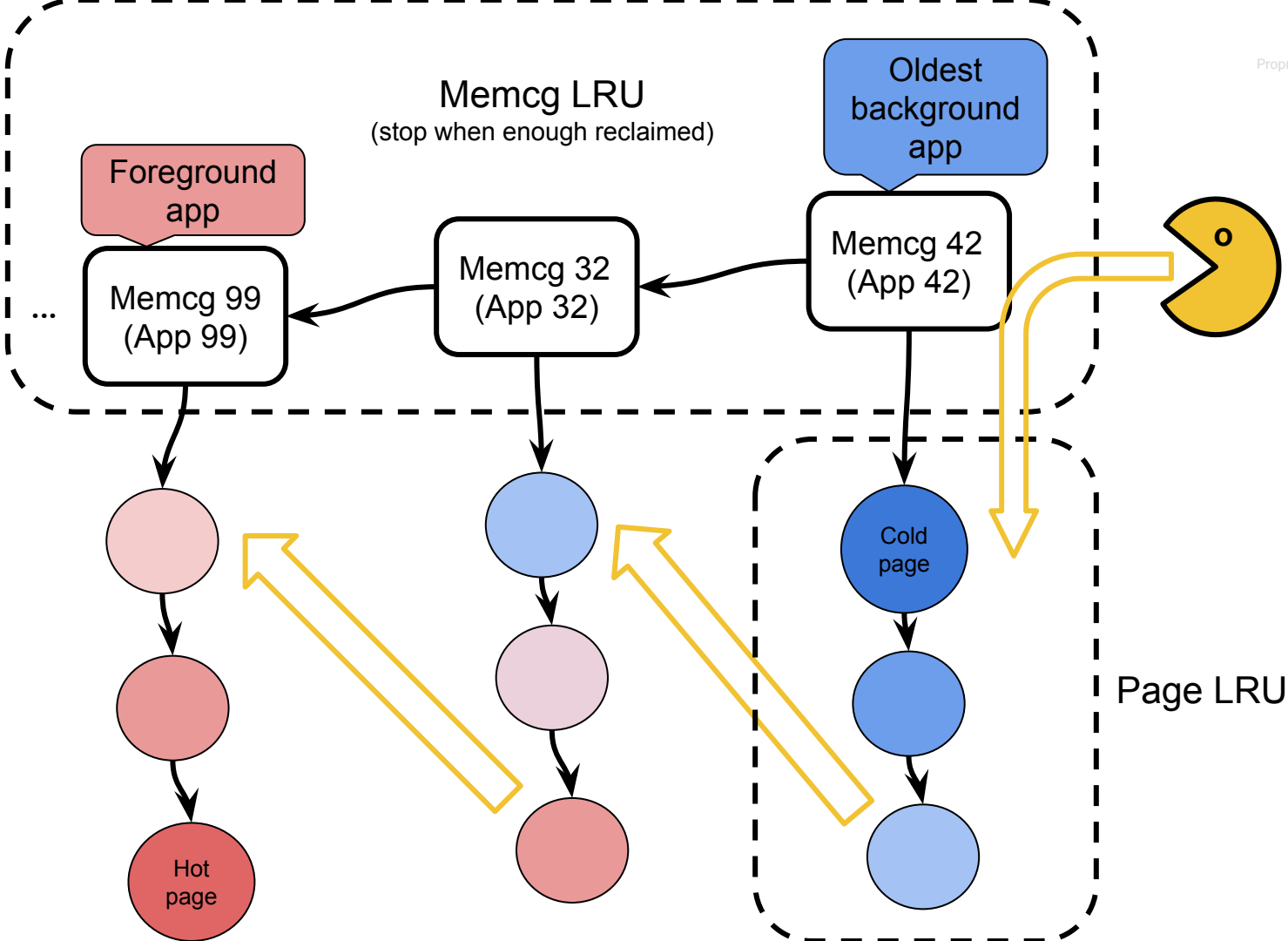
Improved perf for kmem accounting:

https://lore.kernel.org/all/20231019225346.1822282-1-roman.gushchin@linux.dev/

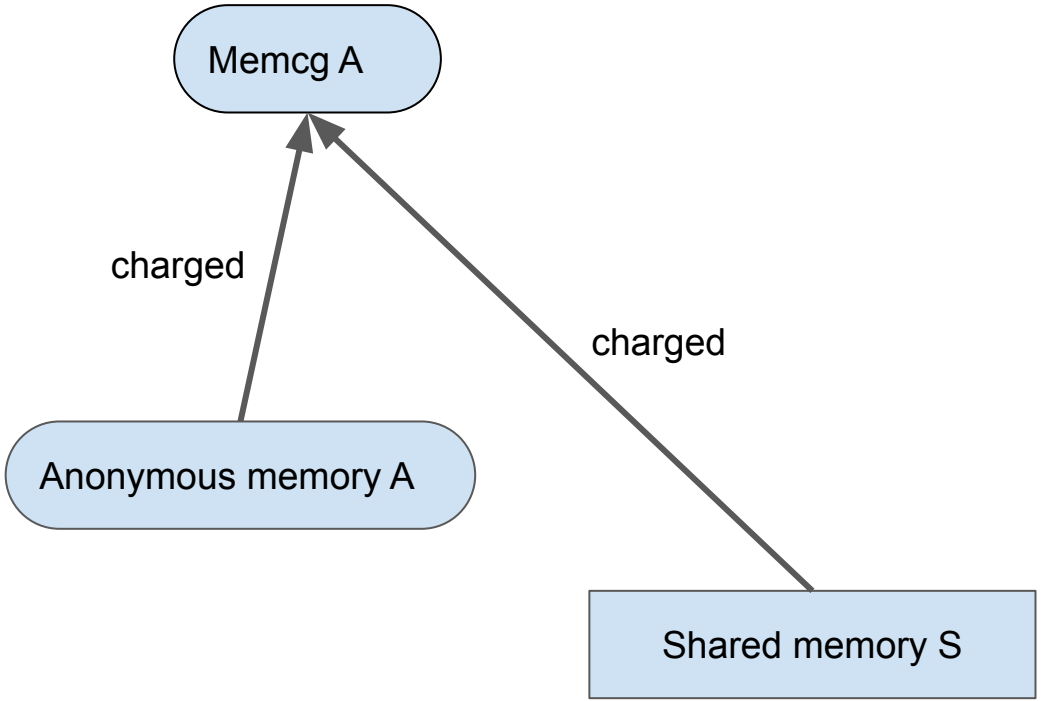
Discussion

What values to use for memcg upper limits?

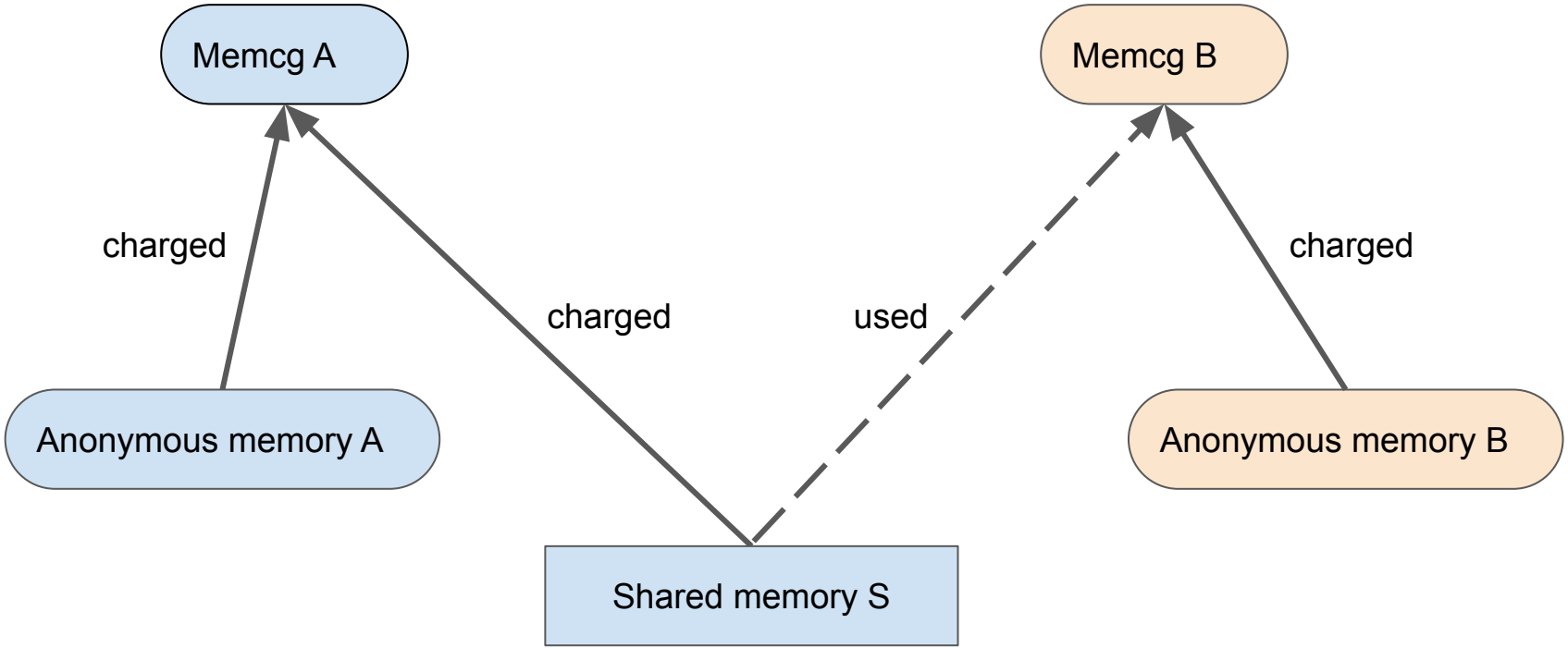




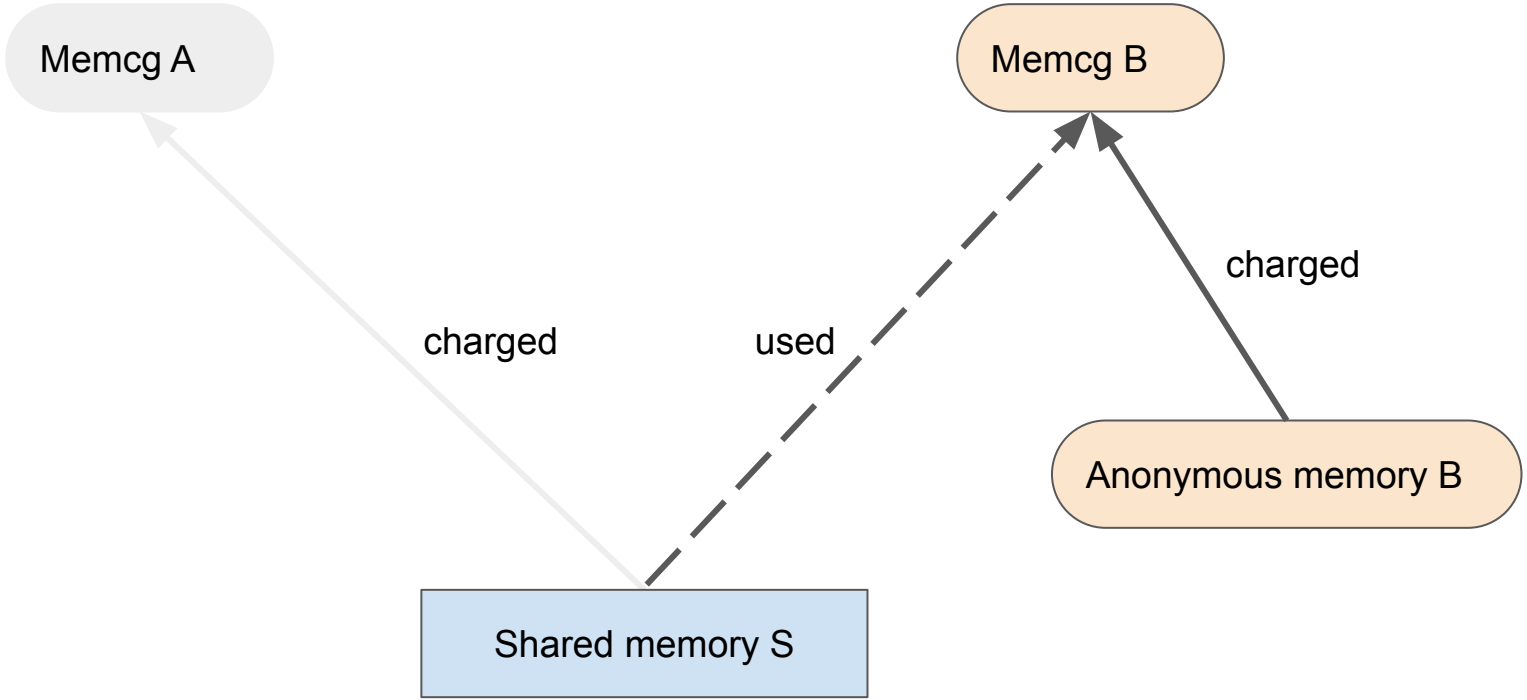
Memcg A Allocates Shared Memory S



Memcgs A and B Share Memory S



Memcg A Removed (Zombie)



Slab size increase in bytes (vs no memcg)

f2fs_inode_cache	34415935
kmalloc-64	32992223
kmalloc-8k	14463369
kmalloc-512	10991846
kmalloc-4k	4918051
dentry	4854776
fscrypt_info	2873291
radix_tree_node	2792981
f2fs_extent_tree	2233066
samsung-iommu-lv2table	2143715