

---

Subject: [PATCH] memory cgroup enhancements take 4 [4/8] remember "a page is on active list of cgroup or

Posted by [KAMEZAWA Hiroyuki](#) on Wed, 31 Oct 2007 10:29:26 GMT

[View Forum Message](#) <> [Reply to Message](#)

---

Remember page\_cgroup is on active\_list or not in page\_cgroup->flags.

Against 2.6.23-mm1.

Changelog v2->v3

- renamed #define PCGF\_ACTIVE to PAGE\_CGROUP\_FLAG\_ACTIVE.

Changelog v1->v2

- moved #define to out-side of struct definition

Signed-off-by: KAMEZAWA Hiroyuki <kamezawa.hiroyu@jp.fujitsu.com>

Signed-off-by: YAMAMOTO Takashi <yamamoto@valinux.co.jp>

mm/memcontrol.c | 10 ++++++---  
1 file changed, 7 insertions(+), 3 deletions(-)

Index: devel-2.6.23-mm1/mm/memcontrol.c

-----  
--- devel-2.6.23-mm1.orig/mm/memcontrol.c

+++ devel-2.6.23-mm1/mm/memcontrol.c

@@ -86,6 +86,7 @@ struct page\_cgroup {  
 int flags;

};

#define PAGE\_CGROUP\_FLAG\_CACHE (0x1) /\* charged as cache \*/

+#define PAGE\_CGROUP\_FLAG\_ACTIVE (0x2) /\* page is active in this cgroup \*/

enum {

MEM\_CGROUP\_TYPE\_UNSPEC = 0,

@@ -213,10 +214,13 @@ clear\_page\_cgroup(struct page \*page, str

static void \_\_mem\_cgroup\_move\_lists(struct page\_cgroup \*pc, bool active)

{

- if (active)

+ if (active) {

+ pc->flags |= PAGE\_CGROUP\_FLAG\_ACTIVE;

list\_move(&pc->lru, &pc->mem\_cgroup->active\_list);

- else

+ } else {

+ pc->flags &= ~PAGE\_CGROUP\_FLAG\_ACTIVE;

list\_move(&pc->lru, &pc->mem\_cgroup->inactive\_list);

+ }

}

int task\_in\_mem\_cgroup(struct task\_struct \*task, const struct mem\_cgroup \*mem)

```
@@ -425,7 +429,7 @@ noreclaim:
    atomic_set(&pc->ref_cnt, 1);
    pc->mem_cgroup = mem;
    pc->page = page;
- pc->flags = 0;
+ pc->flags = PAGE_CGROUP_FLAG_ACTIVE;
  if (ctype == MEM_CGROUP_CHARGE_TYPE_CACHE)
    pc->flags |= PAGE_CGROUP_FLAG_CACHE;
  if (page_cgroup_assign_new_page_cgroup(page, pc)) {
```

---

Containers mailing list  
Containers@lists.linux-foundation.org  
<https://lists.linux-foundation.org/mailman/listinfo/containers>

---