

[PATCH 03/14]

page-replace-remove-sc-from-refill.patch

Source: <http://linux.derkeiler.com/Mailing-Lists/Kernel/2005-12/msg08491.html>

- *From:* Peter Zijlstra <a.p.zijlstra@xxxxxxxx>
 - *Date:* Fri, 30 Dec 2005 23:40:44 +0100
-

From: Peter Zijlstra <a.p.zijlstra@xxxxxxxx>

Remove the dependency on struct scan_control from refill_inactive_zone so we can move it into the page replace file which doesn't know anything about scan_control.

Signed-off-by: Peter Zijlstra <a.p.zijlstra@xxxxxxxx>

mm/vmscan.c | 12 +++-----
1 files changed, 3 insertions(+), 9 deletions(-)

Index: linux-2.6-git/mm/vmscan.c

```
=====
--- linux-2.6-git.orig/mm/vmscan.c 2005-12-10 17:13:56.000000000 +0100
+++ linux-2.6-git/mm/vmscan.c 2005-12-10 18:19:39.000000000 +0100
@@ -71,8 +71,6 @@ struct scan_control {
 /* Incremented by the number of pages reclaimed */
 unsigned long nr_reclaimed;

- unsigned long nr_mapped; /* From page_state */
-
 /* How many pages shrink_cache() should reclaim */
 int nr_to_reclaim;

@@ -730,12 +728,11 @@ done:
 * But we had to alter page->flags anyway.
 */
 static void
- refill_inactive_zone(struct zone *zone, struct scan_control *sc)
+ refill_inactive_zone(struct zone *zone, int nr_pages)
 {
 int pgmoved;
 int pgdeactivate = 0;
 int pgscanned;
- int nr_pages = sc->nr_to_scan;
 LIST_HEAD(l_hold); /* The pages which were snipped off */
 LIST_HEAD(l_inactive); /* Pages to go onto the inactive_list */
```

```
LIST_HEAD(l_active); /* Pages to go onto the active_list */
@@ -765,7 +762,7 @@ refill_inactive_zone(struct zone *zone,
 * mapped memory instead of just pagecache. Work out how much memory
 * is mapped.
 */
- mapped_ratio = (sc->nr_mapped * 100) / total_memory;
+ mapped_ratio = (read_page_state(nr_mapped) * 100) / total_memory;

/*
 * Now decide how much we really want to unmap some pages. The mapped
@@ -892,7 +889,7 @@ shrink_zone(struct zone *zone, struct sc
sc->nr_to_scan = min(nr_active,
(unsigned long)sc->swap_cluster_max);
nr_active -= sc->nr_to_scan;
- refill_inactive_zone(zone, sc);
+ refill_inactive_zone(zone, sc->nr_to_scan);
}

if (nr_inactive) {
@@ -991,7 +988,6 @@ int try_to_free_pages(struct zone **zone
}

for (priority = DEF_PRIORITY; priority >= 0; priority--) {
- sc.nr_mapped = read_page_state(nr_mapped);
sc.nr_scanned = 0;
sc.nr_reclaimed = 0;
sc.priority = priority;
@@ -1080,7 +1076,6 @@ loop_again:
sc.gfp_mask = GFP_KERNEL;
sc.may_writepage = 0;
sc.may_swap = 1;
- sc.nr_mapped = read_page_state(nr_mapped);

inc_page_state(pageoutun);

@@ -1397,7 +1392,6 @@ int zone_reclaim(struct zone *zone, gfp_
sc.gfp_mask = gfp_mask;
sc.may_writepage = 0;
sc.may_swap = 0;
- sc.nr_mapped = read_page_state(nr_mapped);
sc.nr_scanned = 0;
sc.nr_reclaimed = 0;
/* scan at the highest priority */
-
To unsubscribe from this list: send the line "unsubscribe linux-kernel" in
the body of a message to majordomo@xxxxxxxxxxxxxxxxxxx
More majordomo info at http://vger.kernel.org/majordomo-info.html
Please read the FAQ at http://www.tux.org/lkml/
```

- **References:**

- ◆ **[PATCH] vm: page-replace and clockpro**

- ◆ *From:* Peter Zijlstra

- Prev by Date: **[PATCH 5/9] clockpro-ignore token.patch**

- Next by Date: **[PATCH 8/9] clockpro-rename PG active.patch**

- Previous by thread: **[PATCH 5/9] clockpro-ignore token.patch**

- Next by thread: **[PATCH 8/9] clockpro-rename PG active.patch**

- Index(es):

- ◆ **Date**

- ◆ **Thread**