

[PATCH 02/14] page-replace-try_pageout.patch

Source: <http://linux.derkeiler.com/Mailing-Lists/Kernel/2005-12/msg08475.html>

- *From:* Peter Zijlstra <a.p.zijlstra@xxxxxxxxxx>
 - *Date:* Fri, 30 Dec 2005 23:40:34 +0100
-

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

Move the functionality of the shrink_list() loop into its own function.

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

mm/vmscan.c | 324 ++++++-----
1 file changed, 174 insertions(+), 150 deletions(-)

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

```
-----  
--- linux-2.6-git.orig/mm/vmscan.c  
+++ linux-2.6-git/mm/vmscan.c  
@@ -51,6 +51,16 @@ typedef enum {  
PAGE_CLEAN,  
} pageout_t;  
  
+/* possible outcome of try_pageout() */  
+typedef enum {  
+ /* unable to perform pageout */  
+ PAGEOUT_KEEP,  
+ /* unable to perform pageout, page is in active use */  
+ PAGEOUT_ACTIVATE,  
+ /* pageout succeeded, page is gone */  
+ PAGEOUT_SUCCESS,  
+} try_pageout_t;  
+  
struct scan_control {  
/* Ask refill_inactive_zone, or shrink_cache to scan this many pages */  
unsigned long nr_to_scan;  
@@ -382,189 +392,203 @@ static pageout_t pageout(struct page *pa  
return PAGE_CLEAN;  
}  
  
-/*  
- * shrink_list adds the number of reclaimed pages to sc->nr_reclaimed  
- */  
-static int shrink_list(struct list_head *page_list, struct scan_control *sc)
```

[PATCH 02/14] page-replace-try_pageout.patch

```
+static try_pageout_t try_pageout(struct page *page, struct scan_control *sc)
{
- LIST_HEAD(ret_pages);
- struct pagevec freed_pvec;
- int pgactivate = 0;
- int reclaimed = 0;
-
- cond_resched();
-
- pagevec_init(&freed_pvec, 1);
- while (!list_empty(page_list)) {
- struct address_space *mapping;
- struct page *page;
- int may_enter_fs;
- int referenced;
-
- cond_resched();
-
- page = lru_to_page(page_list);
- list_del(&page->lru);
+ struct address_space *mapping;
+ int may_enter_fs;
+ int referenced;

- if (TestSetPageLocked(page))
- goto keep;
+ if (TestSetPageLocked(page))
+ goto keep;

- BUG_ON(PageActive(page));
+ BUG_ON(PageActive(page));

+ sc->nr_scanned++;
+ /* Double the slab pressure for mapped and swapcache pages */
+ if (page_mapped(page) || PageSwapCache(page))
sc->nr_scanned++;
- /* Double the slab pressure for mapped and swapcache pages */
- if (page_mapped(page) || PageSwapCache(page))
- sc->nr_scanned++;

- if (PageWriteback(page))
- goto keep_locked;
+ if (PageWriteback(page))
+ goto keep_locked;

- referenced = page_referenced(page, 1);
- /* In active u
```