

[PATCH 09/14] page-replace-reinsert.patch

Source: <http://linux.derkeiler.com/Mailing-Lists/Kernel/2005-12/msg08493.html>

- *From:* Peter Zijlstra <a.p.zijlstra@xxxxxxxx>
 - *Date:* Fri, 30 Dec 2005 23:41:44 +0100
-

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

page-replace interface function:
page_replace_reinsert()

This function will reinsert those candidate pages that were not freed by try_pageout().

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

include/linux/mm_page_replace.h | 1 +
mm/page_replace.c | 29 +++++
mm/vmscan.c | 25 +-----
3 files changed, 31 insertions(+), 24 deletions(-)

Index: linux-2.6-git/include/linux/mm_page_replace.h

```
=====  
--- linux-2.6-git.orig/include/linux/mm_page_replace.h  
+++ linux-2.6-git/include/linux/mm_page_replace.h  
@@ -43,6 +43,7 @@ static inline void page_replace_activate  
{  
SetPageActive(page);  
}  
+void page_replace_reinsert(struct zone *, struct list_head *);
```

int isolate_lru_pages(int, struct list_head *, struct list_head *, int *);

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

```
=====  
--- linux-2.6-git.orig/mm/page_replace.c  
+++ linux-2.6-git/mm/page_replace.c  
@@ -1,6 +1,7 @@  
#include <linux/mm_page_replace.h>  
#include <linux/mm_inline.h>  
#include <linux/swap.h>  
+#include <linux/pagevec.h>
```

[PATCH 09/14] page-replace-reinsert.patch

```
void __page_replace_insert(struct zone *zone, struct page *page)
@@ -78,3 +79,31 @@ void page_replace_candidates(struct zone
mod_page_state_zone(zone, pgscan_direct, nr_scan);
}

+/*
+ * Put back any unfreeable pages.
+ */
+void page_replace_reinsert(struct zone *zone, struct list_head *page_list)
+{
+ struct pagevec pvec;
+
+ pagevec_init(&pvec, 1);
+ spin_lock_irq(&zone->lru_lock);
+ while (!list_empty(page_list)) {
+ struct page *page = lru_to_page(page_list);
+ BUG_ON(PageLRU(page));
+ SetPageLRU(page);
+ list_del(&page->lru);
+ if (PageActive(page))
+ add_page_to_active_list(zone, page);
+ else
+ add_page_to_inactive_list(zone, page);
+ if (!pagevec_add(&pvec, page)) {
+ spin_unlock_irq(&zone->lru_lock);
+ __pagevec_release(&pvec);
+ spin_lock_irq(&zone->lru_lock);
+ }
+ }
+ spin_unlock_irq(&zone->lru_lock);
+ pagevec_release(&pvec);
+}
+
```

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

```
=====
--- linux-2.6-git.orig/mm/vmscan.c
+++ linux-2.6-git/mm/vmscan.c
@@ -573,8 +573,6 @@ static int shrink_list(struct list_head
static void shrink_cache(struct zone *zone, struct scan_control *sc)
{
LIST_HEAD(page_list);
- struct pagevec pvec;
- struct page *page;
int nr_freed;

lru_add_drain();
@@ -589,28 +587,7 @@ static void shrink_cache(struct zone *zo
mod_page_state_zone(zone, pgsteal, nr_freed);
sc->nr_to_reclaim -= nr_freed;

- /*
```

[PATCH 09/14] page-replace-reinsert.patch

```
- * Put back any unfreeable pages.
- */
- pagevec_init(&pvec, 1);
- spin_lock_irq(&zone->lru_lock);
- while (!list_empty(&page_list)) {
- page = lru_to_page(&page_list);
- if (TestSetPageLRU(page))
- BUG();
- list_del(&page->lru);
- if (PageActive(page))
- add_page_to_active_list(zone, page);
- else
- add_page_to_inactive_list(zone, page);
- if (!pagevec_add(&pvec, page)) {
- spin_unlock_irq(&zone->lru_lock);
- __pagevec_release(&pvec);
- spin_lock_irq(&zone->lru_lock);
- }
- }
- spin_unlock_irq(&zone->lru_lock);
- pagevec_release(&pvec);
+ page_replace_reinsert(zone, &page_list);
}
```

/*

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: [\[PATCH 4/9\] clockpro-use-once.patch](#)
- Next by Date: [\[PATCH 2/9\] clockpro-nonresident-del.patch](#)
- Previous by thread: [\[PATCH 4/9\] clockpro-use-once.patch](#)
- Next by thread: [\[PATCH 2/9\] clockpro-nonresident-del.patch](#)
- Index(es):
 - ◆ [Date](#)
 - ◆ [Thread](#)