
Subject: [PATCH][for -mm] per-zone and reclaim enhancements for memory controller take 3 [6/10] remember recl

Posted by [KAMEZAWA Hiroyuki](#) on Tue, 27 Nov 2007 03:03:52 GMT

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

Functions to remember reclaim priority per cgroup (as zone->prev_priority)

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

```
include/linux/memcontrol.h | 23 ++++++
mm/memcontrol.c            | 20 ++++++
2 files changed, 43 insertions(+)
```

Index: linux-2.6.24-rc3-mm1/mm/memcontrol.c

```
=====
--- linux-2.6.24-rc3-mm1.orig/mm/memcontrol.c 2007-11-27 11:19:51.000000000 +0900
+++ linux-2.6.24-rc3-mm1/mm/memcontrol.c 2007-11-27 11:22:14.000000000 +0900
@@ -132,6 +132,7 @@
 */
 spinlock_t lru_lock;
 unsigned long control_type; /* control RSS or RSS+Pagecache */
+ int prev_priority; /* for recording reclaim priority */
/*
 * statistics.
 */
@@ -452,6 +453,25 @@
 return (long) (active / (inactive + 1));
 }

+/*
+ * prev_priority control...this will be used in memory reclaim path.
+ */
+int mem_cgroup_get_reclaim_priority(struct mem_cgroup *mem)
+{
+ return mem->prev_priority;
+}
+
+void mem_cgroup_note_reclaim_priority(struct mem_cgroup *mem, int priority)
+{
+ if (priority < mem->prev_priority)
+ mem->prev_priority = priority;
+}
+
+void mem_cgroup_record_reclaim_priority(struct mem_cgroup *mem, int priority)
+{
+ mem->prev_priority = priority;
+}
+
```

```
unsigned long mem_cgroup_isolate_pages(unsigned long nr_to_scan,
    struct list_head *dst,
    unsigned long *scanned, int order,
```

Index: linux-2.6.24-rc3-mm1/include/linux/memcontrol.h

```
=====
--- linux-2.6.24-rc3-mm1.orig/include/linux/memcontrol.h 2007-11-27 11:19:00.000000000 +0900
+++ linux-2.6.24-rc3-mm1/include/linux/memcontrol.h 2007-11-27 11:22:14.000000000 +0900
@@ -67,6 +67,11 @@
extern int mem_cgroup_calc_mapped_ratio(struct mem_cgroup *mem);
extern long mem_cgroup_reclaim_imbalance(struct mem_cgroup *mem);

+extern int mem_cgroup_get_reclaim_priority(struct mem_cgroup *mem);
+extern void mem_cgroup_note_reclaim_priority(struct mem_cgroup *mem,
+    int priority);
+extern void mem_cgroup_record_reclaim_priority(struct mem_cgroup *mem,
+    int priority);

#else /* CONFIG_CGROUP_MEM_CONT */
@@ -150,6 +155,24 @@
    return 0;
}

+static inline int mem_cgroup_get_reclaim_priority(struct mem_cgroup *mem,
+    int priority)
+{
+    return 0;
+}
+
+static inline void mem_cgroup_note_reclaim_priority(struct mem_cgroup *mem,
+    int priority)
+{
+    return 0;
+}
+
+static inline void mem_cgroup_record_reclaim_priority(struct mem_cgroup *mem,
+    int priority)
+{
+    return 0;
+}
+
#endif /* CONFIG_CGROUP_MEM_CONT */

#endif /* _LINUX_MEMCONTROL_H */
```

Containers mailing list

Containers@lists.linux-foundation.org

