Subject: [PATCH][for -mm] per-zone and reclaim enhancements for memory controller take 3 [5/10] calculate act
Posted by KAMEZAWA Hiroyuki on Tue, 27 Nov 2007 03:02:55 GMT
View Forum Message <> Reply to Message

calculate active/inactive imbalance per memory cgroup.

Changelog V1 -> V2:
 - removed "total" (just count inactive and active)
 - fixed comment
 - fixed return type to be "long".

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

 include/linux/memcontrol.h | 8 ++++++++
 mm/memcontrol.c | 14 ++++++++++++++
 2 files changed, 22 insertions(+)

Index: linux-2.6.24-rc3-mm1/mm/memcontrol.c
===
--- linux-2.6.24-rc3-mm1.orig/mm/memcontrol.c 2007-11-27 10:44:19.000000000 +0900
+++ linux-2.6.24-rc3-mm1/mm/memcontrol.c 2007-11-27 11:19:51.000000000 +0900
@@ -437,6 +437,20 @@
  rss = (long)mem_cgroup_read_stat(&mem->stat, MEM_CGROUP_STAT_RSS);
  return (int)((rss * 100L) / total);
 }
+/*
+ * This function is called from vmscan.c. In page reclaiming loop. balance
+ * between active and inactive list is calculated. For memory controller
+ * page reclaiming, we should use using mem_cgroup's imbalance rather than
+ * zone's global lru imbalance.
+ */
+long mem_cgroup_reclaim_imbalance(struct mem_cgroup *mem)
+{
+ unsigned long active, inactive;
+ /* active and inactive are the number of pages. 'long' is ok.*/
+ active = mem_cgroup_get_all_zonestat(mem, MEM_CGROUP_ZSTAT_ACTIVE);
+ inactive = mem_cgroup_get_all_zonestat(mem, MEM_CGROUP_ZSTAT_INACTIVE);
+ return (long) (active / (inactive + 1));
+}

 unsigned long mem_cgroup_isolate_pages(unsigned long nr_to_scan,
      struct list_head *dst,
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 10:44:19.000000000 +0900
+++ linux-2.6.24-rc3-mm1/include/linux/memcontrol.h 2007-11-27 11:19:00.000000000 +0900
@@ -65,6 +65,8 @@

```
  * For memory reclaim.
  */
 extern int mem_cgroup_calc_mapped_ratio(struct mem_cgroup *mem);
+extern long mem_cgroup_reclaim_imbalance(struct mem_cgroup *mem);
+


 #else /* CONFIG_CGROUP_MEM_CONT */
@@ -142,6 +144,12 @@
 {
  return 0;
 }
+
+static inline int mem_cgroup_reclaim_imbalance(struct mem_cgroup *mem)
+{
+ return 0;
+}
+
 #endif /* CONFIG_CGROUP_MEM_CONT */

 #endif /* _LINUX_MEMCONTROL_H */
```