
Subject: [RFC][PATCH] memory cgroup enhancements updated [1/10]
try_to_free_mem_cgroup_pages bugfix
Posted by [KAMEZAWA Hiroyuki](#) on Fri, 19 Oct 2007 09:29:52 GMT
[View Forum Message](#) <> [Reply to Message](#)

Because NODE_DATA(node)->node_zonelist[] is guaranteed to contain all necessary zones, it is not necessary to use for_each_online_node.

And this for_each_online_node() makes reclaim routine start always from node 0. This is bad.

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

mm/vmscan.c | 8 +++-----
1 file changed, 3 insertions(+), 5 deletions(-)

Index: devel-2.6.23-mm1/mm/vmscan.c

```
=====
--- devel-2.6.23-mm1.orig/mm/vmscan.c
+++ devel-2.6.23-mm1/mm/vmscan.c
@@ -1375,15 +1375,13 @@ unsigned long try_to_free_mem_cgroup_pag
     .mem_cgroup = mem_cont,
     .isolate_pages = mem_cgroup_isolate_pages,
 };
- int node;
+ int node = numa_node_id();
     struct zone **zones;
     int target_zone = gfp_zone(GFP_HIGHUSER_MOVABLE);

- for_each_online_node(node) {
-     zones = NODE_DATA(node)->node_zonelist[target_zone].zones;
-     if (do_try_to_free_pages(zones, sc.gfp_mask, &sc))
+ zones = NODE_DATA(node)->node_zonelist[target_zone].zones;
+ if (do_try_to_free_pages(zones, sc.gfp_mask, &sc))
     return 1;
- }
     return 0;
 }
#endif
```

Containers mailing list
Containers@lists.linux-foundation.org
<https://lists.linux-foundation.org/mailman/listinfo/containers>
