
Subject: [PATCH 5/9] deal with movement of tasks across groups
Posted by [Srivatsa Vaddagiri](#) on Thu, 12 Apr 2007 17:57:11 GMT
[View Forum Message](#) <> [Reply to Message](#)

When a task moves between groups (as initiated by an administrator), it has to be removed from the runqueue of its old group and added to the runqueue of its new group. This patch defines this move operation.

Signed-off-by : Srivatsa Vaddagiri <vatsa@in.ibm.com>

1 file changed, 28 insertions(+)

```
diff -puN kernel/sched.c~task_move kernel/sched.c
--- linux-2.6.20/kernel/sched.c~task_move 2007-04-11 12:00:51.000000000 +0530
+++ linux-2.6.20-vatsa/kernel/sched.c 2007-04-12 11:07:16.000000000 +0530
@@ -7449,4 +7449,32 @@ static int sched_get_quota(struct task_g
    return cpu_quota(tg);
}

+/*
+ * Move a task from one group to another. If the task is already on a
+ * runqueue, this involves removing the task from its old group's runqueue
+ * and adding to its new group's runqueue.
+ */
+static void sched_move_task(struct task_grp *tg_new, struct task_grp *tg_old,
+ struct task_struct *tsk)
+{
+ struct rq *rq;
+ unsigned long flags;
+
+ if (tg_new == tg_old)
+ return;
+
+ rq = task_rq_lock(tsk, &flags);
+
+ if (tsk->array) {
+ /* Set tsk->group to tg_old here */
+ deactivate_task(tsk, rq);
+ /* Set tsk->group to tg_new here */
+ __activate_task(tsk, rq);
+ }
+
+ task_rq_unlock(rq, &flags);
```

```
+  
+ return;  
+}  
+  
#endif /* CONFIG_CPUMETER */
```

```
--  
--  
Regards,  
vatsa
```

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