

Re: [PATCH] sched: smpnice work around for active\_load\_balance()

## Re: [PATCH] sched: smpnice work around for active\_load\_balance()

---

*Source:* <http://linux.derkeiler.com/Mailing-Lists/Kernel/2006-03/msg09683.html>

---

- *From:* "Siddha, Suresh B" <[suresh.b.siddha@xxxxxxxxx](mailto:suresh.b.siddha@xxxxxxxxx)>
  - *Date:* Tue, 28 Mar 2006 11:25:21 -0800
- 

On Tue, Mar 28, 2006 at 05:00:50PM +1100, Peter Williams wrote:

Problem:

It is undesirable for HT/MC packages to have more than one of their CPUs busy if there are other packages that have all of their CPUs idle. This

We need to balance even if the other packages are not idle.. For example, consider a 4-core DP system, if we have 6 runnable (assume same priority) processes, we want to schedule 3 of them in each package..

Today's active load balance implementation is very simple and generic. And hence it works smoothly with dual and multi-core.. Please read my OLS 2005 paper which talks about different scheduling scenarios and also how we were planning to implement Power savings policy in case of multi-core.. I had a prototype patch for doing this, which I held it up before going on vacation, as it needed some rework with your smpnice patch in place.. I will post a patch on top of current mainline for your reference.

```
+ } else if (!busiest_has_loaded_cpus && avg_load < max_load) {
```

I haven't fully digested the result of this patch but should this be `avg_load < max_load` or `avg_load > max_load` ?

Either way, I can show scheduling scenarios which will fail..

```
- if (rq->raw_weighted_load > max_load && rq->nr_running > 1) {  
+ if (rq->nr_running > 1) {  
+ if (rq->raw_weighted_load > max_load || !busiest_is_loaded) {  
+ max_load = rq->raw_weighted_load;  
+ busiest = rq;  
+ busiest_is_loaded = 1;  
+ }
```

Re: [PATCH] sched: smpnice work around for active\_load\_balance()

Re: [PATCH] sched: smpnice work around for active\_load\_balance()

```
+ } else if (!busiest_is_loaded && rqi->raw_weighted_load > max_load) {
```

Please note the point that same scheduling logic has to work for all the different levels of scheduler domains... I think these checks complicates the decisions as we go up in the scheduling hirerachy.. Please go through the HT/MC/MP/Numa combinations and with same/different priority processes for different scenarios..

Even with no HT and MC, this patch has still has issues in the presence of different priority tasks... consider a simple DP system and run two instances of high priority tasks(simple infinite loop) and two normal priority tasks. With "top" I observed that these normal priority tasks keep on jumping from one processor to another... Ideally with smpnice, we would assume that each processor should have two tasks (one high priority and another one with normal priority) ..

thanks,

suresh

-

To unsubscribe from this list: send the line "unsubscribe linux-kernel" in the body of a message to majordomo@xxxxxxxxxxxxxxxxx

More majordomo info at <http://vger.kernel.org/majordomo-info.html>

Please read the FAQ at <http://www.tux.org/lkml/>