
Subject: Re: [PATCH 02/11] memcg: Reclaim when more than one page needed.
Posted by [Glauber Costa](#) on Wed, 27 Jun 2012 10:03:01 GMT

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

On 06/26/2012 08:09 AM, David Rientjes wrote:

```
> @@ -2206,7 +2214,7 @@ static int mem_cgroup_do_charge(struct mem_cgroup *memcg,  
gfp_t gfp_mask,  
>> * unlikely to succeed so close to the limit, and we fall back  
>> * to regular pages anyway in case of failure.  
>> */  
>>- if (nr_pages == 1 && ret)  
>>+ if (nr_pages <= NR_PAGES_TO_RETRY && ret)  
>> return CHARGE_RETRY;
```

Changed to costly order.

One more thing. The original version of this patch included a `cond_resched()` here, that was also removed. From my re-reading of the code in `page_alloc.c` and `vmscan.c` now, I tend to think this is indeed not needed, since any `cond_resched()`s that might be needed to ensure the safety of the code will be properly inserted by the reclaim code itself, so there is no need for us to include any when we signal that a retry is needed.

Do you/others agree?
