

---

Subject: Re: [PATCH v5] slab: Ignore internal flags in cache creation  
Posted by [akpm](#) on Thu, 18 Oct 2012 22:42:03 GMT

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

---

On Wed, 17 Oct 2012 15:36:51 +0400

Glauber Costa <[glommer@parallels.com](mailto:glommer@parallels.com)> wrote:

> Some flags are used internally by the allocators for management  
> purposes. One example of that is the CFLGS\_OFF\_SLAB flag that slab uses  
> to mark that the metadata for that cache is stored outside of the slab.  
>  
> No cache should ever pass those as a creation flags. We can just ignore  
> this bit if it happens to be passed (such as when duplicating a cache in  
> the kmem memcg patches).

I may be misunderstanding this, but...

If some caller to `kmem_cache_create()` is passing in bogus flags then  
that's a bug, and it is undesirable to hide such a bug in this fashion?

> Because such flags can vary from allocator to allocator, we allow them  
> to make their own decisions on that, defining SLAB\_AVAILABLE\_FLAGS with  
> all flags that are valid at creation time. Allocators that doesn't have  
> any specific flag requirement should define that to mean all flags.  
>  
> Common code will mask out all flags not belonging to that set.

---