Subject: Re: [PATCH v3 15/28] slub: always get the cache from its page in kfree
Posted by Christoph Lameter on Tue, 29 May 2012 14:42:03 GMT
View Forum Message <> Reply to Message

On Fri, 25 May 2012, Glauber Costa wrote:

> struct page already have this information. If we start chaining
> caches, this information will always be more trustworthy than
> whatever is passed into the function

Yes but the lookup of the page struct also costs some cycles. SLAB in
!NUMA mode and SLOB avoid these lookups and can improve their freeing
speed because of that.

> diff --git a/mm/slub.c b/mm/slub.c
> index 0eb9e72..640872f 100644
> --- a/mm/slub.c
> +++ b/mm/slub.c
> @@ -2598,10 +2598,14 @@ redo:
> void kmem_cache_free(struct kmem_cache *s, void *x)
> {
>   struct page *page;
> + bool slab_match;
>
>   page = virt_to_head_page(x);
>
> - slab_free(s, page, x, _RET_IP_);
> + slab_match = (page->slab == s) | slab_is_parent(page->slab, s);
> + VM_BUG_ON(!slab_match);

Why add a slab_match bool if you do not really need it?