
Subject: [PATCH 6/15] Make input layer use seq_list_xxx helpers

Posted by [xemul](#) on Fri, 18 May 2007 09:33:28 GMT

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

This is essentially just a renaming of the existing functions as the seq_list_start() and seq_list_next() copies already exist in the input.c file. Now we have them in the generic place.

Signed-off-by: Pavel Emelianov <xemul@openvz.org>

```
diff --git a/drivers/input/input.c b/drivers/input/input.c
index ccd8aba..da83c17 100644
--- a/drivers/input/input.c
+++ b/drivers/input/input.c
@@ -471,37 +471,16 @@ static unsigned int input_proc_devices_p
     return 0;
 }

-static struct list_head *list_get_nth_element(struct list_head *list, loff_t *pos)
-{
-    struct list_head *node;
-    loff_t i = 0;
-
-    list_for_each(node, list)
-        if (i++ == *pos)
-            return node;
-
-    return NULL;
-}
-
-static struct list_head *list_get_next_element(struct list_head *list, struct list_head *element, loff_t
*pos)
-{
-    if (element->next == list)
-        return NULL;
-
-    ++(*pos);
-    return element->next;
-}
-
static void *input_devices_seq_start(struct seq_file *seq, loff_t *pos)
{
    /* acquire lock here ... Yes, we do need locking, I knowi, I know... */

-    return list_get_nth_element(&input_dev_list, pos);
```

```

+ return seq_list_start(&input_dev_list, *pos);
}

static void *input_devices_seq_next(struct seq_file *seq, void *v, loff_t *pos)
{
- return list_get_next_element(&input_dev_list, v, pos);
+ return seq_list_next(v, &input_dev_list, pos);
}

static void input_devices_seq_stop(struct seq_file *seq, void *v)
@@ -592,13 +571,13 @@ static void *input_handlers_seq_start(st
{
/* acquire lock here ... Yes, we do need locking, I knowi, I know... */
seq->private = (void *) (unsigned long) *pos;
- return list_get_nth_element(&input_handler_list, pos);
+ return seq_list_start(&input_handler_list, *pos);
}

static void *input_handlers_seq_next(struct seq_file *seq, void *v, loff_t *pos)
{
seq->private = (void *) (unsigned long) (*pos + 1);
- return list_get_next_element(&input_handler_list, v, pos);
+ return seq_list_next(v, &input_handler_list, pos);
}

static void input_handlers_seq_stop(struct seq_file *seq, void *v)

```
