

---

Subject: [PATCH 1/8] CGroup Files: Clean up whitespace in struct cftype  
Posted by [Paul Menage](#) on Fri, 20 Jun 2008 23:43:59 GMT

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

This patch removes some extraneous spaces from method declarations in struct cftype, to fit in with conventional kernel style.

Signed-off-by: Paul Menage <[menage@google.com](mailto:menage@google.com)>

---

```
include/linux/cgroup.h | 32 ++++++-----  
1 file changed, 16 insertions(+), 16 deletions(-)
```

Index: cws-2.6.26-rc5-mm3/include/linux/cgroup.h

```
=====  
--- cws-2.6.26-rc5-mm3.orig/include/linux/cgroup.h  
+++ cws-2.6.26-rc5-mm3/include/linux/cgroup.h  
@@ -205,48 +205,48 @@ struct cftype {  
 * subsystem, followed by a period */  
 char name[MAX_CFTYPE_NAME];  
 int private;  
- int (*open) (struct inode *inode, struct file *file);  
- ssize_t (*read) (struct cgroup *cgrp, struct cftype *cft,  
- struct file *file,  
- char __user *buf, size_t nbytes, loff_t *ppos);  
+ int (*open)(struct inode *inode, struct file *file);  
+ ssize_t (*read)(struct cgroup *cgrp, struct cftype *cft,  
+ struct file *file,  
+ char __user *buf, size_t nbytes, loff_t *ppos);  
/*  
 * read_u64() is a shortcut for the common case of returning a  
 * single integer. Use it in place of read()  
 */  
- u64 (*read_u64) (struct cgroup *cgrp, struct cftype *cft);  
+ u64 (*read_u64)(struct cgroup *cgrp, struct cftype *cft);  
/*  
 * read_s64() is a signed version of read_u64()  
 */  
- s64 (*read_s64) (struct cgroup *cgrp, struct cftype *cft);  
+ s64 (*read_s64)(struct cgroup *cgrp, struct cftype *cft);  
/*  
 * read_map() is used for defining a map of key/value  
 * pairs. It should call cb->fill(cb, key, value) for each  
 * entry. The key/value pairs (and their ordering) should not  
 * change between reboots.  
 */  
- int (*read_map) (struct cgroup *cont, struct cftype *cft,  
- struct cgroup_map_cb *cb);
```

```

+ int (*read_map)(struct cgroup *cont, struct cftype *cft,
+   struct cgroup_map_cb *cb);
/*
 * read_seq_string() is used for outputting a simple sequence
 * using seqfile.
 */
- int (*read_seq_string) (struct cgroup *cont, struct cftype *cft,
-   struct seq_file *m);
+ int (*read_seq_string)(struct cgroup *cont, struct cftype *cft,
+   struct seq_file *m);

- ssize_t (*write) (struct cgroup *cgrp, struct cftype *cft,
-   struct file *file,
-   const char __user *buf, size_t nbytes, loff_t *ppos);
+ ssize_t (*write)(struct cgroup *cgrp, struct cftype *cft,
+   struct file *file,
+   const char __user *buf, size_t nbytes, loff_t *ppos);

/*
 * write_u64() is a shortcut for the common case of accepting
 * a single integer (as parsed by simple strtoull) from
 * userspace. Use in place of write(); return 0 or error.
 */
- int (*write_u64) (struct cgroup *cgrp, struct cftype *cft, u64 val);
+ int (*write_u64)(struct cgroup *cgrp, struct cftype *cft, u64 val);
/*
 * write_s64() is a signed version of write_u64()
 */
- int (*write_s64) (struct cgroup *cgrp, struct cftype *cft, s64 val);
+ int (*write_s64)(struct cgroup *cgrp, struct cftype *cft, s64 val);

/*
 * trigger() callback can be used to get some kick from the
@@ -256,7 +256,7 @@ struct cftype {
*/
int (*trigger)(struct cgroup *cgrp, unsigned int event);

- int (*release) (struct inode *inode, struct file *file);
+ int (*release)(struct inode *inode, struct file *file);
};

struct cgroup_scanner {

--
```

---

Containers mailing list  
[Containers@lists.linux-foundation.org](mailto:Containers@lists.linux-foundation.org)  
<https://lists.linux-foundation.org/mailman/listinfo/containers>

---