

Posted by [Cedric Le Goater](#) on Tue, 02 Oct 2007 08:46:11 GMT

From: Cedric Le Goater <clg@fr.ibm.com>

if 'data' is a pointer to a namespace, it makes it easy to create a single super_block per namespace. This is something we need for the mqueue file system, and other single super_block internal file systems.

Dave, can you review this one to make sure the helper routine belongs to fs/super.c ?

Thanks

Signed-off-by: Cedric Le Goater <clg@fr.ibm.com>

— — —

```
fs/super.c      | 32 ++++++
include/linux/fs.h | 4 ++++
2 files changed, 36 insertions(+)
```

Index: 2.6.23-rc8-mm2/fs/super.c

```
--- 2.6.23-rc8-mm2.orig/fs/super.c
```

+++ 2.6.23-rc8-mm2/fs/super.c

```
@@ -853,6 +853,38 @@ int get_sb_single(struct file_system_type
```

```
EXPORT_SYMBOL(get_sb_single);
```

```
+static int compare_data(struct super_block *sb, void *data)
+{
+ return (void *)sb->s_fs_info == data;
+}
+
+int get_sb_single_per_data(struct file_system_type *fs_type,
+ int flags, void *data,
+ int (*fill_super)(struct super_block *, void *, int),
+ struct vfsmount *mnt)
+{
+ struct super_block *s;
+ int error;
+
+
```

```

+ s = sget(fs_type, compare_data, set_anon_super, data);
+ if (IS_ERR(s))
+ return PTR_ERR(s);
+ if (!s->s_root) {
+ s->s_flags = flags;
+ s->s_fs_info = data;
+ error = fill_super(s, data, flags & MS_SILENT ? 1 : 0);
+ if (error) {
+ up_write(&s->s_umount);
+ deactivate_super(s);
+ return error;
+ }
+ s->s_flags |= MS_ACTIVE;
+ }
+ do_remount_sb(s, flags, data, 0);
+ return simple_set_mnt(mnt, s);
+}
+EXPORT_SYMBOL(get_sb_single_per_data);
+
+ struct vfsmount *
+ vfs_kern_mount(struct file_system_type *type, int flags, const char *name, void *data)
+ {
Index: 2.6.23-rc8-mm2/include/linux/fs.h
=====
--- 2.6.23-rc8-mm2.orig/include/linux/fs.h
+++ 2.6.23-rc8-mm2/include/linux/fs.h
@@ -1437,6 +1437,10 @@ extern int get_sb_single(struct file_sys
    int flags, void *data,
    int (*fill_super)(struct super_block *, void *, int),
    struct vfsmount *mnt);
+extern int get_sb_single_per_data(struct file_system_type *fs_type,
+ int flags, void *data,
+ int (*fill_super)(struct super_block *, void *, int),
+ struct vfsmount *mnt);
extern int get_sb_nodev(struct file_system_type *fs_type,
    int flags, void *data,
    int (*fill_super)(struct super_block *, void *, int),
--

```

Containers mailing list
Containers@lists.linux-foundation.org
<https://lists.linux-foundation.org/mailman/listinfo/containers>
