
Subject: [PATCH 08/11] driver core: Implement tagged directory support for device classes.

Posted by [Benjamin Thery](#) on Fri, 06 Jun 2008 15:48:14 GMT

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

driver core: Implement tagged directory support for device classes.

This patch enables tagging on every class directory if struct class has tag_ops.

In addition device_del and device_rename were modified to use sysfs_delete_link and sysfs_rename_link respectively to ensure when these operations happen on devices whose classes have tag_ops that they work properly.

Signed-off-by: Eric W. Biederman <ebiederm@xmission.com>

Signed-off-by: Benjamin Thery <benjamin.thery@bull.net>

```
drivers/base/class.c | 30 ++++++-----
drivers/base/core.c  | 41 ++++++-----
include/linux/device.h | 2 ++
3 files changed, 50 insertions(+), 23 deletions(-)
```

Index: linux-mm/drivers/base/class.c

=====

```
--- linux-mm.orig/drivers/base/class.c
+++ linux-mm/drivers/base/class.c
@@ -134,6 +134,17 @@ static void remove_class_attrs(struct cl
 }
 }

+static int class_setup_tagging(struct class *cls)
+{
+ const struct sysfs_tagged_dir_operations *tag_ops;
+
+ tag_ops = cls->tag_ops;
+ if (!tag_ops)
+ return 0;
+
+ return sysfs_enable_tagging(&cls->subsys.kobj, tag_ops);
+}
+
int class_register(struct class *cls)
{
int error;
@@ -162,11 +173,22 @@ int class_register(struct class *cls)
cls->subsys.kobj.ktype = &class_ktype;
```

```

    error = kset_register(&cls->subsys);
- if (!error) {
- error = add_class_attrs(class_get(cls));
- class_put(cls);
- }
+ if (error)
+ goto out;
+
+ error = class_setup_tagging(cls);
+ if (error)
+ goto out_unregister;
+
+ error = add_class_attrs(cls);
+ if (error)
+ goto out_unregister;
+
+out:
    return error;
+out_unregister:
+ kset_unregister(&cls->subsys);
+ goto out;
}

```

void class_unregister(struct class *cls)

Index: linux-mm/drivers/base/core.c

```

=====
--- linux-mm.orig/drivers/base/core.c
+++ linux-mm/drivers/base/core.c
@@ -617,6 +617,10 @@ static struct kobject *get_device_parent
    kobject_put(k);
    return NULL;
}
+ /* If we created a new class-directory setup tagging */
+ if (dev->class->tag_ops)
+ sysfs_enable_tagging(k, dev->class->tag_ops);
+
+ /* do not emit an uevent for this simple "glue" directory */
    return k;
}
@@ -751,12 +755,13 @@ static void device_remove_class_symlinks

    if (dev->kobj.parent != &dev->class->subsys.kobj &&
        device_is_not_partition(dev))
- sysfs_remove_link(&dev->class->subsys.kobj, dev->bus_id);
+ sysfs_delete_link(&dev->class->subsys.kobj, &dev->kobj,
+ dev->bus_id);
    #else
    if (dev->parent && device_is_not_partition(dev))

```

```

sysfs_remove_link(&dev->kobj, "device");

- sysfs_remove_link(&dev->class->subsys.kobj, dev->bus_id);
+ sysfs_delete_link(&dev->class->subsys.kobj, &dev->kobj, dev->bus_id);
#endif

sysfs_remove_link(&dev->kobj, "subsystem");
@@ -1282,6 +1287,15 @@ int device_rename(struct device *dev, ch
strncpy(old_device_name, dev->bus_id, BUS_ID_SIZE);
strncpy(dev->bus_id, new_name, BUS_ID_SIZE);

+#ifndef CONFIG_SYSFS_DEPRECATED
+ if (dev->class && (dev->kobj.parent != &dev->class->subsys.kobj)) {
+ error = sysfs_rename_link(&dev->class->subsys.kobj,
+ &dev->kobj, old_device_name, new_name);
+ if (error)
+ goto out;
+ }
+#endif
+
error = kobject_rename(&dev->kobj, new_name);
if (error) {
strncpy(dev->bus_id, old_device_name, BUS_ID_SIZE);
@@ -1290,24 +1304,13 @@ int device_rename(struct device *dev, ch

#ifdef CONFIG_SYSFS_DEPRECATED
if (old_class_name) {
+ error = -ENOMEM;
new_class_name = make_class_name(dev->class->name, &dev->kobj);
- if (new_class_name) {
- error = sysfs_create_link(&dev->parent->kobj,
- &dev->kobj, new_class_name);
- if (error)
- goto out;
- sysfs_remove_link(&dev->parent->kobj, old_class_name);
- }
- }
-#else
- if (dev->class) {
- sysfs_remove_link(&dev->class->subsys.kobj, old_device_name);
- error = sysfs_create_link(&dev->class->subsys.kobj, &dev->kobj,
- dev->bus_id);
- if (error) {
- dev_err(dev, "%s: sysfs_create_symlink failed (%d)\n",
- __func__, error);
- }
+ if (new_class_name)
+ error = sysfs_rename_link(&dev->parent->kobj,

```

```
+    &dev->kobj,  
+    old_class_name,  
+    new_class_name);  
}  
#endif
```

Index: linux-mm/include/linux/device.h

```
=====  
--- linux-mm.orig/include/linux/device.h  
+++ linux-mm/include/linux/device.h  
@@ -199,6 +199,8 @@ struct class {  
    int (*suspend)(struct device *dev, pm_message_t state);  
    int (*resume)(struct device *dev);  
  
+ const struct sysfs_tagged_dir_operations *tag_ops;  
+  
    struct pm_ops *pm;  
};
```

--

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