

[RFC: 2.6 patch] the overdue removal of drivers/pcmcia/pcmcia_ioctl.c

Source: <http://linux.derkeiler.com/Mailing-Lists/Kernel/2006-03/msg08312.html>

- *From:* Adrian Bunk <bunk@xxxxxxxx>
 - *Date:* Fri, 24 Mar 2006 02:46:34 +0100
-

This patch contains the overdue removal of drivers/pcmcia/pcmcia_ioctl.c plus the fallout of additional cleanups after this removal.

This patch contains the scheduled removal of drivers/pcmcia/pcmcia_ioctl.c plus the fallout of additional cleanups after this removal.

Signed-off-by: Adrian Bunk <bunk@xxxxxxxx>

Documentation/Changes | 19
Documentation/feature-removal-schedule.txt | 17
Documentation/ioctl-mess.txt | 26
arch/m68k/Kconfig | 4
arch/sparc/Kconfig | 4
arch/sparc64/Kconfig | 4
drivers/net/Kconfig | 3
drivers/pcmcia/Kconfig | 16
drivers/pcmcia/Makefile | 1
drivers/pcmcia/cs.c | 17
drivers/pcmcia/cs_internal.h | 3
drivers/pcmcia/ds.c | 23
drivers/pcmcia/ds_internal.h | 19
drivers/pcmcia/pcmcia_ioctl.c | 796 -----
drivers/pcmcia/pcmcia_resource.c | 5
drivers/pcmcia/rsrc_mgr.c | 80 ---
drivers/pnp/pnpbios/Kconfig | 3
include/pcmcia/cs.h | 1
include/pcmcia/ds.h | 43 -
include/pcmcia/ss.h | 6
scripts/ver_linux | 3
21 files changed, 19 insertions(+), 1124 deletions(-)

--- linux-2.6.14-mm1-full/Documentation/feature-removal-schedule.txt.old 2005-11-07 19:31:51.000000000 +0100

+++ linux-2.6.14-mm1-full/Documentation/feature-removal-schedule.txt 2005-11-07

[RFC: 2.6 patch] the overdue removal of drivers/pcmcia/pcmcia_ioctl.c

19:32:09.000000000 +0100

@ @ -92,23 +92,6 @ @

-What: PCMCIA control ioctl (needed for pcmcia-cs [cardmgr, cardctl])
-When: November 2005
-Files: drivers/pcmcia/: pcmcia_ioctl.c
-Why: With the 16-bit PCMCIA subsystem now behaving (almost) like a
- normal hotpluggable bus, and with it using the default kernel
- infrastructure (hotplug, driver core, sysfs) keeping the PCMCIA
- control ioctl needed by cardmgr and cardctl from pcmcia-cs is
- unnecessary, and makes further cleanups and integration of the
- PCMCIA subsystem into the Linux kernel device driver model more
- difficult. The features provided by cardmgr and cardctl are either
- handled by the kernel itself now or are available in the new
- pcmciautils package available at
- <http://kernel.org/pub/linux/utils/kernel/pcmcia/>
-Who: Dominik Brodowski <linux@xxxxxxxx>
-

-
What: ip_queue and ip6_queue (old ipv4-only and ipv6-only netfilter queue)

When: December 2005

Why: This interface has been obsoleted by the new layer3-independent

--- linux-2.6.14-mm1-full/Documentation/ioctl-mess.txt.old 2005-11-07 19:49:59.000000000 +0100

+++ linux-2.6.14-mm1-full/Documentation/ioctl-mess.txt 2005-11-07 19:50:09.000000000 +0100

@ @ -1283,32 +1283,6 @ @

I: struct drm_lock

O: -

-DS_ACCESS_CONFIGURATION_REGISTER
-DS_ADJUST_RESOURCE_INFO
-DS_BIND_MTD
-DS_BIND_REQUEST
-DS_EJECT_CARD
-DS_GET_CONFIGURATION_INFO
-DS_GET_DEVICE_INFO
-DS_GET_FIRST_REGION
-DS_GET_FIRST_TUPLE
-DS_GET_FIRST_WINDOW
-DS_GET_MEM_PAGE
-DS_GET_NEXT_DEVICE
-DS_GET_NEXT_REGION
-DS_GET_NEXT_TUPLE
-DS_GET_NEXT_WINDOW
-DS_GET_STATUS
-DS_GET_TUPLE_DATA
-DS_INSERT_CARD
-DS_PARSE_TUPLE
-DS_REPLACE_CIS

[RFC: 2.6 patch] the overdue removal of drivers/pcmcia/pcmcia_ioctl.c

[RFC: 2.6 patch] the overdue removal of drivers/pcmcia/pcmcia_ioctl.c

- DS_RESET_CARD
- DS_RESUME_CARD
- DS_SUSPEND_CARD
- DS_UNBIND_REQUEST
- DS_VALIDATE_CIS

-

N: DV1394_IOC_RECEIVE_FRAMES

I: (unsigned int) arg

O: -

--- linux-2.6.14-mm1-full/drivers/pcmcia/Kconfig.old 2005-11-07 19:33:17.000000000 +0100

+++ linux-2.6.14-mm1-full/drivers/pcmcia/Kconfig 2005-11-07 19:43:18.000000000 +0100

@@ -65,21 +65,7 @@

to be loaded from userspace to work correctly. If you say Y here, and your userspace is arranged correctly, this will be loaded automatically using the in-kernel firmware loader and the hotplug subsystem, instead of relying on cardmgr from pcmcia-cs to do so.

-

- If unsure, say Y.

-

-config PCMCIA_IOCTL

- bool "PCMCIA control ioctl (obsolete)"

- depends on PCMCIA

- default y

- help

- If you say Y here, the deprecated ioctl interface to the PCMCIA

- subsystem will be built. It is needed by cardmgr and cardctl

- (pcmcia-cs) to function properly.

-

- You should use the new pcmciautils package instead (see

- <file:Documentation/Changes> for location and details).

+ subsystem.

If unsure, say Y.

--- linux-2.6.14-mm1-full/include/pcmcia/ds.h.old 2005-11-07 19:49:21.000000000 +0100

+++ linux-2.6.14-mm1-full/include/pcmcia/ds.h 2005-11-07 19:49:44.000000000 +0100

@@ -51,49 +51,6 @@

u_int CardOffset;

} mtd_info_t;

-typedef union ds_ioctl_arg_t {

- adjust_t adjust;

- config_info_t config;

- tuple_t tuple;

- tuple_parse_t tuple_parse;

- client_req_t client_req;

- cs_status_t status;

- conf_reg_t conf_reg;

- cisinfo_t cisinfo;

- region_info_t region;

- bind_info_t bind_info;

[RFC: 2.6 patch] the overdue removal of drivers/pcmcia/pcmcia_ioctl.c

```
– mtd_info_t mtd_info;
– win_info_t win_info;
– cisdump_t cisdump;
–} ds_ioctl_arg_t;
–
–#define DS_ADJUST_RESOURCE_INFO_IOWR('d', 2, adjust_t)
–#define DS_GET_CONFIGURATION_INFO_IOWR('d', 3, config_info_t)
–#define DS_GET_FIRST_TUPLE_IOWR('d', 4, tuple_t)
–#define DS_GET_NEXT_TUPLE_IOWR('d', 5, tuple_t)
–#define DS_GET_TUPLE_DATA_IOWR('d', 6, tuple_parse_t)
–#define DS_PARSE_TUPLE_IOWR('d', 7, tuple_parse_t)
–#define DS_RESET_CARD_IO ('d', 8)
–#define DS_GET_STATUS_IOWR('d', 9, cs_status_t)
–#define DS_ACCESS_CONFIGURATION_REGISTER_IOWR('d', 10, conf_reg_t)
–#define DS_VALIDATE_CIS_IOR ('d', 11, cisinfo_t)
–#define DS_SUSPEND_CARD_IO ('d', 12)
–#define DS_RESUME_CARD_IO ('d', 13)
–#define DS_EJECT_CARD_IO ('d', 14)
–#define DS_INSERT_CARD_IO ('d', 15)
–#define DS_GET_FIRST_REGION_IOWR('d', 16, region_info_t)
–#define DS_GET_NEXT_REGION_IOWR('d', 17, region_info_t)
–#define DS_REPLACE_CIS_IOWR('d', 18, cisdump_t)
–#define DS_GET_FIRST_WINDOW_IOR ('d', 19, win_info_t)
–#define DS_GET_NEXT_WINDOW_IOWR('d', 20, win_info_t)
–#define DS_GET_MEM_PAGE_IOWR('d', 21, win_info_t)
–
–#define DS_BIND_REQUEST_IOWR('d', 60, bind_info_t)
–#define DS_GET_DEVICE_INFO_IOWR('d', 61, bind_info_t)
–#define DS_GET_NEXT_DEVICE_IOWR('d', 62, bind_info_t)
–#define DS_UNBIND_REQUEST_IOW ('d', 63, bind_info_t)
–#define DS_BIND_MTD_IOWR('d', 64, mtd_info_t)
–
#ifdef __KERNEL__
#include <linux/device.h>

--- linux-2.6.14-mm1-full/include/pcmcia/ss.h.old 2005-11-07 19:36:04.000000000 +0100
+++ linux-2.6.14-mm1-full/include/pcmcia/ss.h 2005-11-07 19:54:29.000000000 +0100
@@ -272,10 +272,6 @@
struct work_struct device_add; /* for adding further pseudo-multifunction
* devices */

–#ifdef CONFIG_PCMCIA_IOCTL
– struct user_info_t *user;
– wait_queue_head_t queue;
–#endif
#endif

/* cardbus (32-bit) */
@@ -290,8 +286,6 @@

};
```

```
-struct pcmcia_socket * pcmcia_get_socket_by_nr(unsigned int nr);
```

```
-
```

```
extern void pcmcia_parse_events(struct pcmcia_socket *socket, unsigned int events);  
--- linux-2.6.14-mm1-full/include/pcmcia/cs.h.old 2005-11-07 19:55:12.000000000 +0100  
+++ linux-2.6.14-mm1-full/include/pcmcia/cs.h 2005-11-07 20:02:24.000000000 +0100  
@@ -387,7 +387,6 @@  
int pcmcia_get_first_window(window_handle_t *win, win_req_t *req);  
int pcmcia_get_next_window(window_handle_t *win, win_req_t *req);  
int pcmcia_get_status(struct pcmcia_device *p_dev, cs_status_t *status);  
-int pcmcia_get_mem_page(window_handle_t win, memreq_t *req);  
int pcmcia_map_mem_page(window_handle_t win, memreq_t *req);  
int pcmcia_modify_configuration(struct pcmcia_device *p_dev, modconf_t *mod);  
int pcmcia_register_client(client_handle_t *handle, client_req_t *req);  
--- linux-2.6.14-mm1-full/arch/m68k/Kconfig.old 2005-11-07 19:37:26.000000000 +0100  
+++ linux-2.6.14-mm1-full/arch/m68k/Kconfig 2005-11-07 19:37:32.000000000 +0100  
@@ -69,9 +69,7 @@
```

and the newer 32 bit CardBus cards. If you want to use CardBus cards, you need to say Y here and also to "CardBus support" below.

- To use your PC-cards, you will need supporting software from David
- Hinds' pcmcia-cs package (see the file <file:Documentation/Changes>
- for location). Please also read the PCMCIA-HOWTO, available from
- + Please also read the PCMCIA-HOWTO, available from
- <<http://www.tldp.org/docs.html#howto>>.

To compile this driver as modules, choose M here: the

```
--- linux-2.6.14-mm1-full/arch/sparc/Kconfig.old 2005-11-07 19:37:45.000000000 +0100  
+++ linux-2.6.14-mm1-full/arch/sparc/Kconfig 2005-11-07 19:37:53.000000000 +0100  
@@ -97,9 +97,7 @@
```

and the newer 32 bit CardBus cards. If you want to use CardBus cards, you need to say Y here and also to "CardBus support" below.

- To use your PC-cards, you will need supporting software from David
- Hinds' pcmcia-cs package (see the file <file:Documentation/Changes>
- for location). Please also read the PCMCIA-HOWTO, available from
- + Please also read the PCMCIA-HOWTO, available from
- <<http://www.tldp.org/docs.html#howto>>.

To compile this driver as modules, choose M here: the

```
--- linux-2.6.14-mm1-full/arch/sparc64/Kconfig.old 2005-11-07 19:38:01.000000000 +0100  
+++ linux-2.6.14-mm1-full/arch/sparc64/Kconfig 2005-11-07 19:38:07.000000000 +0100  
@@ -241,9 +241,7 @@
```

and the newer 32 bit CardBus cards. If you want to use CardBus cards, you need to say Y here and also to "CardBus support" below.

- To use your PC-cards, you will need supporting software from David
- Hinds' pcmcia-cs package (see the file <file:Documentation/Changes>
- for location). Please also read the PCMCIA-HOWTO, available from

[RFC: 2.6 patch] the overdue removal of drivers/pcmcia/pcmcia_ioctl.c

+ Please also read the PCMCIA-HOWTO, available from
<<http://www.tldp.org/docs.html#howto>>.

To compile this driver as modules, choose M here: the

--- linux-2.6.14-mm1-full/Documentation/Changes.old 2005-11-07 19:38:28.000000000 +0100

+++ linux-2.6.14-mm1-full/Documentation/Changes 2005-11-07 19:39:30.000000000 +0100

@@ -59,7 +59,6 @@

o reiser4progs 1.0.0 # fsck.reiser4 -V

o xfsprogs 2.6.0 # xfs_db -V

o pcmciautils 004

-o pcmcia-cs 3.1.21 # cardmgr -V

o quota-tools 3.09 # quota -V

o PPP 2.4.0 # pppd --version

o isdn4k-utils 3.1pre1 # isdnctrl 2>&1|grep version

@@ -204,17 +203,9 @@

PCMCIAutils

-PCMCIAutils replaces pcmcia-cs (see below). It properly sets up
-PCMCIA sockets at system startup and loads the appropriate modules
-for 16-bit PCMCIA devices if the kernel is modularized and the hotplug
-subsystem is used.

-

-Pcmcia-cs

-

-PCMCIA (PC Card) support is now partially implemented in the main
-kernel source. The "pcmciautils" package (see above) replaces pcmcia-cs
-for newest kernels.

+PCMCIAutils replaces pcmcia-cs. It properly sets up PCMCIA sockets at
+system startup and loads the appropriate modules for 16-bit PCMCIA
+devices if the kernel is modularized and the hotplug subsystem is used.

Quota-tools

@@ -385,10 +376,6 @@

o <<ftp://ftp.kernel.org/pub/linux/utils/kernel/pcmcia/>>

-Pcmcia-cs

-o <<http://pcmcia-cs.sourceforge.net/>>

-

Quota-tools

o <<http://sourceforge.net/projects/linuxquota/>>

--- linux-2.6.14-mm1-full/drivers/net/Kconfig.old 2005-11-07 19:41:56.000000000 +0100

+++ linux-2.6.14-mm1-full/drivers/net/Kconfig 2005-11-07 19:42:21.000000000 +0100

@@ -1709,8 +1709,7 @@

If you want to plug a network (or some other) card into the PCMCIA
(or PC-card) slot of your laptop instead (PCMCIA is the standard for

[RFC: 2.6 patch] the overdue removal of drivers/pcmcia/pcmcia_ioctl.c

[RFC: 2.6 patch] the overdue removal of drivers/pcmcia/pcmcia_ioctl.c

credit card size extension cards used by all modern laptops), you

- need the pcmcia-cs package (location contained in the file
- <file:Documentation/Changes>) and you can say N here.
- + can say N here.

Laptop users should read the Linux Laptop home page at

<<http://www.linux-on-laptops.com/>> or

```
--- linux-2.6.14-mm1-full/drivers/pnp/pnpbios/Kconfig.old 2005-11-07 19:43:33.000000000 +0100
```

```
+++ linux-2.6.14-mm1-full/drivers/pnp/pnpbios/Kconfig 2005-11-07 19:43:40.000000000 +0100
```

```
@@ -34,9 +34,6 @@
```

Also some buggy systems will fault when accessing certain features in the PNPBIOS /proc interface (e.g. "boot" configs).

- See the latest pcmcia-cs (stand-alone package) for a nice set of
- PNPBIOS /proc interface tools (lspnp and setpnp).

-

Unless you are debugging or have other specific reasons, it is recommended that you say N here.

```
--- linux-2.6.14-mm1-full/scripts/ver_linux.old 2005-11-07 19:44:04.000000000 +0100
```

```
+++ linux-2.6.14-mm1-full/scripts/ver_linux 2005-11-07 19:44:09.000000000 +0100
```

```
@@ -48,9 +48,6 @@
```

```
xfs_db -V 2>&1 | grep version | awk \
```

```
'NR==1{print "xfsprogs ", $3}'
```

```
-cardmgr -V 2>&1 | grep version | awk \
```

```
-'NR==1{print "pcmcia-cs ", $3}'
```

-

```
quota -V 2>&1 | grep version | awk \
```

```
'NR==1{print "quota-tools ", $NF}'
```

```
--- linux-2.6.16-mm1-full/drivers/pcmcia/Makefile.old 2006-03-24 01:35:01.000000000 +0100
```

```
+++ linux-2.6.16-mm1-full/drivers/pcmcia/Makefile 2006-03-24 01:35:35.000000000 +0100
```

```
@@ -11,7 +11,6 @@
```

```
obj-$(CONFIG_PCCARD) += pcmcia_core.o
```

```
pcmcia-y += ds.o pcmcia_resource.o
```

```
-pcmcia-$(CONFIG_PCMCIA_IOCTL) += pcmcia_ioctl.o
```

```
obj-$(CONFIG_PCMCIA) += pcmcia.o
```

```
obj-$(CONFIG_PCCARD_NONSTATIC) += rsrc_nonstatic.o
```

```
--- linux-2.6.16-mm1-full/drivers/pcmcia/ds.c.old 2006-03-24 01:36:24.000000000 +0100
```

```
+++ linux-2.6.16-mm1-full/drivers/pcmcia/ds.c 2006-03-24 01:37:01.000000000 +0100
```

```
@@ -55,7 +54,9 @@
```

```
#define ds_dbg(lvl, fmt, arg...) do { } while (0)
```

```
#endif
```

```
-spinlock_t pcmcia_dev_list_lock;
```

```
+static spinlock_t pcmcia_dev_list_lock;
```

+

```
+static struct bus_type pcmcia_bus_type;
```

```
/*=====*/  
  
@@ -328,7 +329,7 @@  
  
/* pcmcia_device handling */  
  
-struct pcmcia_device * pcmcia_get_dev(struct pcmcia_device *p_dev)  
+static struct pcmcia_device * pcmcia_get_dev(struct pcmcia_device *p_dev)  
{  
struct device *tmp_dev;  
tmp_dev = get_device(&p_dev->dev);  
@@ -337,7 +338,7 @@  
return to_pcmcia_dev(tmp_dev);  
}  
  
-void pcmcia_put_dev(struct pcmcia_device *p_dev)  
+static void pcmcia_put_dev(struct pcmcia_device *p_dev)  
{  
if (p_dev)  
put_device(&p_dev->dev);  
@@ -577,7 +578,8 @@  
*/  
static DEFINE_MUTEX(device_add_lock);  
  
-struct pcmcia_device * pcmcia_device_add(struct pcmcia_socket *s, unsigned int function)  
+static struct pcmcia_device * pcmcia_device_add(struct pcmcia_socket *s,  
+ unsigned int function)  
{  
struct pcmcia_device *p_dev, *tmp_dev;  
unsigned long flags;  
@@ -1152,13 +1154,11 @@  
case CS_EVENT_CARD_REMOVAL:  
s->pcmcia_state.present = 0;  
pcmcia_card_remove(skt, NULL);  
- handle_event(skt, event);  
break;  
  
case CS_EVENT_CARD_INSERTION:  
s->pcmcia_state.present = 1;  
pcmcia_card_add(skt);  
- handle_event(skt, event);  
break;  
  
case CS_EVENT_EJECTION_REQUEST:  
@@ -1169,7 +1169,6 @@  
case CS_EVENT_RESET_PHYSICAL:  
case CS_EVENT_CARD_RESET:  
default:  
- handle_event(skt, event);  
break;
```

```
}

@@ -1231,9 +1230,6 @@
*/
msleep(250);

-#ifdef CONFIG_PCMCIA_IOCTL
- init_waitqueue_head(&socket->queue);
-#endif
INIT_LIST_HEAD(&socket->devices_list);
INIT_WORK(&socket->device_add, pcmcia_delayed_add_pseudo_device, socket);
memset(&socket->pcmcia_state, 0, sizeof(u8));
@@ -1274,7 +1270,7 @@
};

-struct bus_type pcmcia_bus_type = {
+static struct bus_type pcmcia_bus_type = {
.name = "pcmcia",
.uevent = pcmcia_bus_uevent,
.match = pcmcia_bus_match,
@@ -1293,8 +1289,6 @@
bus_register(&pcmcia_bus_type);
class_interface_register(&pcmcia_bus_interface);

- pcmcia_setup_ioctl();
-
return 0;
}
fs_initcall(init_pcmcia_bus); /* one level after subsys_initcall so that
@@ -1303,8 +1297,6 @@

static void __exit exit_pcmcia_bus(void)
{
- pcmcia_cleanup_ioctl();
-
class_interface_unregister(&pcmcia_bus_interface);

bus_unregister(&pcmcia_bus_type);
--- linux-2.6.16-mm1-full/drivers/pcmcia/rsrc_mgr.c.old 2006-03-24 01:37:57.000000000 +0100
+++ linux-2.6.16-mm1-full/drivers/pcmcia/rsrc_mgr.c 2006-03-24 01:38:13.000000000 +0100
@@ -21,86 +21,6 @@
#include "cs_internal.h"

-#ifdef CONFIG_PCMCIA_IOCTL
-
-#ifdef CONFIG_PCMCIA_PROBE
-
- static int adjust_irq(struct pcmcia_socket *s, adjust_t *adj)
- {
```

```
- int irq;
- u32 mask;
-
- irq = adj->resource.irq.IRQ;
- if ((irq < 0) || (irq > 15))
- return CS_BAD_IRQ;
-
- if (adj->Action != REMOVE_MANAGED_RESOURCE)
- return 0;
-
- mask = 1 << irq;
-
- if (!(s->irq_mask & mask))
- return 0;
-
- s->irq_mask &= ~mask;
-
- return 0;
-}
-
-#else
-
-#static inline int adjust_irq(struct pcmcia_socket *s, adjust_t *adj) {
- return CS_SUCCESS;
-}
-
-#endif
-
-int pcmcia_adjust_resource_info(adjust_t *adj)
- {
- struct pcmcia_socket *s;
- int ret = CS_UNSUPPORTED_FUNCTION;
- unsigned long flags;
-
- down_read(&pcmcia_socket_list_rwsem);
- list_for_each_entry(s, &pcmcia_socket_list, socket_list) {
-
- if (adj->Resource == RES_IRQ)
- ret = adjust_irq(s, adj);
-
- else if (s->resource_ops->adjust_resource) {
-
- /* you can't use the old interface if the new
- * one was used before */
- spin_lock_irqsave(&s->lock, flags);
- if ((s->resource_setup_new) &&
- !(s->resource_setup_old)) {
- spin_unlock_irqrestore(&s->lock, flags);
- continue;
- } else if (!(s->resource_setup_old))
```

[RFC: 2.6 patch] the overdue removal of drivers/pcmcia/pcmcia_ioctl.c

```
- s->resource_setup_old = 1;
- spin_unlock_irqrestore(&s->lock, flags);
-
- ret = s->resource_ops->adjust_resource(s, adj);
- if (!ret) {
- /* as there's no way we know this is the
- * last call to adjust_resource_info, we
- * always need to assume this is the latest
- * one... */
- spin_lock_irqsave(&s->lock, flags);
- s->resource_setup_done = 1;
- spin_unlock_irqrestore(&s->lock, flags);
- }
- }
- }
- up_read(&pcmcia_socket_list_rwsem);
-
- return (ret);
-}
-EXPORT_SYMBOL(pcmcia_adjust_resource_info);
-
-#endif
-
int pcmcia_validate_mem(struct pcmcia_socket *s)
{
if (s->resource_ops->validate_mem)
--- linux-2.6.16-mm1-full/drivers/pcmcia/cs_internal.h.old 2006-03-24 01:38:44.000000000 +0100
+++ linux-2.6.16-mm1-full/drivers/pcmcia/cs_internal.h 2006-03-24 01:39:19.000000000 +0100
@@ -126,10 +126,7 @@
/* In cs.c */
extern struct rw_semaphore pcmcia_socket_list_rwsem;
extern struct list_head pcmcia_socket_list;
-int pcmcia_get_window(struct pcmcia_socket *s, window_handle_t *handle, int idx, win_req_t *req);
-int pccard_get_configuration_info(struct pcmcia_socket *s, struct pcmcia_device *p_dev, config_info_t
*config);
int pccard_reset_card(struct pcmcia_socket *skt);
-int pccard_get_status(struct pcmcia_socket *s, struct pcmcia_device *p_dev, cs_status_t *status);

struct pcmcia_callback{
--- linux-2.6.16-mm1-full/drivers/pcmcia/cs.c.old 2006-03-24 01:41:04.000000000 +0100
+++ linux-2.6.16-mm1-full/drivers/pcmcia/cs.c 2006-03-24 01:43:39.000000000 +0100
@@ -291,23 +291,6 @@
EXPORT_SYMBOL(pcmcia_unregister_socket);

-struct pcmcia_socket * pcmcia_get_socket_by_nr(unsigned int nr)
- {
- struct pcmcia_socket *s;
-
- down_read(&pcmcia_socket_list_rwsem);
```

[RFC: 2.6 patch] the overdue removal of drivers/pcmcia/pcmcia_ioctl.c

```
- list_for_each_entry(s, &pcmcia_socket_list, socket_list)
- if (s->sock == nr) {
- up_read(&pcmcia_socket_list_rwsem);
- return s;
- }
- up_read(&pcmcia_socket_list_rwsem);
-
- return NULL;
-
-}
-EXPORT_SYMBOL(pcmcia_get_socket_by_nr);
-
/**
 * The central event handler. Send_event() sends an event to the
 * 16-bit subsystem, which then calls the relevant device drivers.
--- linux-2.6.16-mm1-full/drivers/pcmcia/pcmcia_resource.c.old 2006-03-24 01:45:21.000000000 +0100
+++ linux-2.6.16-mm1-full/drivers/pcmcia/pcmcia_resource.c 2006-03-24 01:45:59.000000000 +0100
@@ -200,9 +199,9 @@
EXPORT_SYMBOL(pcmcia_access_configuration_register);

-int pcmcia_get_configuration_info(struct pcmcia_socket *s,
- struct pcmcia_device *p_dev,
- config_info_t *config)
+static int pcmcia_get_configuration_info(struct pcmcia_socket *s,
+ struct pcmcia_device *p_dev,
+ config_info_t *config)
{
config_t *c;

@@ -265,40 +264,6 @@
EXPORT_SYMBOL(pcmcia_get_configuration_info);

-/** pcmcia_get_window
- */
-int pcmcia_get_window(struct pcmcia_socket *s, window_handle_t *handle,
- int idx, win_req_t *req)
-{
- window_t *win;
- int w;
-
- if (!s || !(s->state & SOCKET_PRESENT))
- return CS_NO_CARD;
- for (w = idx; w < MAX_WIN; w++)
- if (s->state & SOCKET_WIN_REQ(w))
- break;
- if (w == MAX_WIN)
- return CS_NO_MORE_ITEMS;
- win = &s->win[w];
- req->Base = win->ctl.res->start;
```

[RFC: 2.6 patch] the overdue removal of drivers/pcmcia/pcmcia_ioctl.c

```
- req->Size = win->ctl.res->end - win->ctl.res->start + 1;
- req->AccessSpeed = win->ctl.speed;
- req->Attributes = 0;
- if (win->ctl.flags & MAP_ATTRIB)
- req->Attributes |= WIN_MEMORY_TYPE_AM;
- if (win->ctl.flags & MAP_ACTIVE)
- req->Attributes |= WIN_ENABLE;
- if (win->ctl.flags & MAP_16BIT)
- req->Attributes |= WIN_DATA_WIDTH_16;
- if (win->ctl.flags & MAP_USE_WAIT)
- req->Attributes |= WIN_USE_WAIT;
- *handle = win;
- return CS_SUCCESS;
-} /* pcmcia_get_window */
-EXPORT_SYMBOL(pcmcia_get_window);
-
-
/** pccard_get_status
 *
 * Get the current socket state bits. We don't support the latched
 @@ -367,21 +332,6 @@

-/** pcmcia_get_mem_page
 *
 * Change the card address of an already open memory window.
 */
-int pcmcia_get_mem_page(window_handle_t win, memreq_t *req)
-{
- if ((win == NULL) || (win->magic != WINDOW_MAGIC))
- return CS_BAD_HANDLE;
- req->Page = 0;
- req->CardOffset = win->ctl.card_start;
- return CS_SUCCESS;
-} /* pcmcia_get_mem_page */
-EXPORT_SYMBOL(pcmcia_get_mem_page);
-
-
int pcmcia_map_mem_page(window_handle_t win, memreq_t *req)
{
struct pcmcia_socket *s;
--- linux-2.6.16-mm1-full/drivers/pcmcia/pcmcia_ioctl.c 2006-03-23 12:01:09.000000000 +0100
+++ /dev/null 2006-02-12 01:05:26.000000000 +0100
@@ -1,796 +0,0 @@
-/**
- * pcmcia_ioctl.c --- ioctl interface for cardmgr and cardctl
- *
- * This program is free software; you can redistribute it and/or modify
- * it under the terms of the GNU General Public License version 2 as
- * published by the Free Software Foundation.
```

```

- *
- * The initial developer of the original code is David A. Hinds
- * <dahinds@xxxxxxxxxxxxxxxxxxxxxxxx>. Portions created by David A. Hinds
- * are Copyright (C) 1999 David A. Hinds. All Rights Reserved.
- *
- * (C) 1999 David A. Hinds
- * (C) 2003 - 2004 Dominik Brodowski
- */
-
-/*
- * This file will go away soon.
- */
-
-
-#include <linux/kernel.h>
-#include <linux/module.h>
-#include <linux/init.h>
-#include <linux/major.h>
-#include <linux/errno.h>
-#include <linux/ioctl.h>
-#include <linux/proc_fs.h>
-#include <linux/poll.h>
-#include <linux/pci.h>
-#include <linux/workqueue.h>
-
-#define IN_CARD_SERVICES
-#include <pcmcia/cs_types.h>
-#include <pcmcia/cs.h>
-#include <pcmcia/cistpl.h>
-#include <pcmcia/ds.h>
-#include <pcmcia/ss.h>
-
-#include "cs_internal.h"
-#include "ds_internal.h"
-
-static int major_dev = -1;
-
-/* Device user information */
-#define MAX_EVENTS 32
-#define USER_MAGIC 0x7ea4
-#define CHECK_USER(u) \
-(((u) == NULL) || ((u)->user_magic != USER_MAGIC))
-
-typedef struct user_info_t {
- u_int user_magic;
- int event_head, event_tail;
- event_t event[MAX_EVENTS];
- struct user_info_t *next;
- struct pcmcia_socket *socket;
-} user_info_t;
```

```
-
-
-#ifdef DEBUG
-extern int ds_pc_debug;
-#define cs_socket_name(skt) ((skt)->dev.class_id)
-
-#define ds_dbg(lvl, fmt, arg...) do { \
- if (ds_pc_debug >= lvl) \
- printk(KERN_DEBUG "ds: " fmt , ## arg); \
-} while (0)
-#else
-#define ds_dbg(lvl, fmt, arg...) do { } while (0)
-#endif
-
-static struct pcmcia_device *get_pcmcia_device(struct pcmcia_socket *s,
- unsigned int function)
- {
- struct pcmcia_device *p_dev = NULL;
- unsigned long flags;
-
- spin_lock_irqsave(&pcmcia_dev_list_lock, flags);
- list_for_each_entry(p_dev, &s->devices_list, socket_device_list) {
- if (p_dev->func == function) {
- spin_unlock_irqrestore(&pcmcia_dev_list_lock, flags);
- return pcmcia_get_dev(p_dev);
- }
- }
- spin_unlock_irqrestore(&pcmcia_dev_list_lock, flags);
- return NULL;
- }
-
-/* backwards-compatible accessing of driver ---- by name! */
-
-static struct pcmcia_driver *get_pcmcia_driver(dev_info_t *dev_info)
- {
- struct device_driver *drv;
- struct pcmcia_driver *p_drv;
-
- drv = driver_find((char *) dev_info, &pcmcia_bus_type);
- if (!drv)
- return NULL;
-
- p_drv = container_of(drv, struct pcmcia_driver, drv);
-
- return (p_drv);
- }
-
-#ifdef CONFIG_PROC_FS
-static struct proc_dir_entry *proc_pccard = NULL;
-
```

```

-static int proc_read_drivers_callback(struct device_driver *driver, void *d)
-{
- char **p = d;
- struct pcmcia_driver *p_drv = container_of(driver,
- struct pcmcia_driver, drv);
-
- *p += sprintf(*p, "%-24.24s 1 %d\n", p_drv->drv.name,
-#ifdef CONFIG_MODULE_UNLOAD
- (p_drv->owner) ? module_refcount(p_drv->owner) : 1
-#else
- 1
-#endif
- );
- d = (void *) p;
-
- return 0;
-}
-
-static int proc_read_drivers(char *buf, char **start, off_t pos,
- int count, int *eof, void *data)
-{
- char *p = buf;
-
- bus_for_each_drv(&pcmcia_bus_type, NULL,
- (void *) &p, proc_read_drivers_callback);
-
- return (p - buf);
-}
-#endif
-
-/*=====
-
- These manage a ring buffer of events pending for one user process
-
-=====*/
-
-static int queue_empty(user_info_t *user)
-{
- return (user->event_head == user->event_tail);
-}
-
-static event_t get_queued_event(user_info_t *user)
-{
- user->event_tail = (user->event_tail+1) % MAX_EVENTS;
- return user->event[user->event_tail];
-}
-
-static void queue_event(user_info_t *user, event_t event)
-{
- user->event_head = (user->event_head+1) % MAX_EVENTS;

```

[RFC: 2.6 patch] the overdue removal of drivers/pcmcia/pcmcia_ioctl.c

```
- if (user->event_head == user->event_tail)
- user->event_tail = (user->event_tail+1) % MAX_EVENTS;
- user->event[user->event_head] = event;
-}
-
-void handle_event(struct pcmcia_socket *s, event_t event)
-{
- user_info_t *user;
- for (user = s->user; user; user = user->next)
- queue_event(user, event);
- wake_up_interruptible(&s->queue);
-}
-
-
-/*=====
-
- bind_request() and bind_device() are merged by now. Register_client()
- is called right at the end of bind_request(), during the driver's
- ->attach() call. Individual descriptions:
-
- bind_request() connects a socket to a particular client driver.
- It looks up the specified device ID in the list of registered
- drivers, binds it to the socket, and tries to create an instance
- of the device. unbind_request() deletes a driver instance.
-
- Bind_device() associates a device driver with a particular socket.
- It is normally called by Driver Services after it has identified
- a newly inserted card. An instance of that driver will then be
- eligible to register as a client of this socket.
-
- Register_client() uses the dev_info_t handle to match the
- caller with a socket. The driver must have already been bound
- to a socket with bind_device() -- in fact, bind_device()
- allocates the client structure that will be used.
-
-=====*/
-
-static int bind_request(struct pcmcia_socket *s, bind_info_t *bind_info)
-{
- struct pcmcia_driver *p_drv;
- struct pcmcia_device *p_dev;
- int ret = 0;
- unsigned long flags;
-
- s = pcmcia_get_socket(s);
- if (!s)
- return -EINVAL;
-
- ds_dbg(2, "bind_request(%d, '%s')\n", s->sock,
- (char *)bind_info->dev_info);
-}
```

[RFC: 2.6 patch] the overdue removal of drivers/pcmcia/pcmcia_ioctl.c

```
- p_drv = get_pcmcia_driver(&bind_info->dev_info);
- if (!p_drv) {
- ret = -EINVAL;
- goto err_put;
- }
-
- if (!try_module_get(p_drv->owner)) {
- ret = -EINVAL;
- goto err_put_driver;
- }
-
- spin_lock_irqsave(&pcmcia_dev_list_lock, flags);
- list_for_each_entry(p_dev, &s->devices_list, socket_device_list) {
- if (p_dev->func == bind_info->function) {
- if ((p_dev->dev.driver == &p_drv->drv)) {
- if (p_dev->cardmgr) {
- /* if there's already a device
- * registered, and it was registered
- * by userspace before, we need to
- * return the "instance". */
- spin_unlock_irqrestore(&pcmcia_dev_list_lock, flags);
- bind_info->instance = p_dev;
- ret = -EBUSY;
- goto err_put_module;
- } else {
- /* the correct driver managed to bind
- * itself magically to the correct
- * device. */
- spin_unlock_irqrestore(&pcmcia_dev_list_lock, flags);
- p_dev->cardmgr = p_drv;
- ret = 0;
- goto err_put_module;
- }
- } else if (!p_dev->dev.driver) {
- /* there's already a device available where
- * no device has been bound to yet. So we don't
- * need to register a device! */
- spin_unlock_irqrestore(&pcmcia_dev_list_lock, flags);
- goto rescan;
- }
- }
- spin_unlock_irqrestore(&pcmcia_dev_list_lock, flags);
-
- p_dev = pcmcia_device_add(s, bind_info->function);
- if (!p_dev) {
- ret = -EIO;
- goto err_put_module;
- }
-
- rescan:
```

[RFC: 2.6 patch] the overdue removal of drivers/pcmcia/pcmcia_ioctl.c

```
- p_dev->cardmgr = p_drv;
-
- /* if a driver is already running, we can abort */
- if (p_dev->dev.driver)
- goto err_put_module;
-
- /*
- * Prevent this racing with a card insertion.
- */
- mutex_lock(&s->skt_mutex);
- bus_rescan_devices(&pcmcia_bus_type);
- mutex_unlock(&s->skt_mutex);
-
- /* check whether the driver indeed matched. I don't care if this
- * is racy or not, because it can only happen on cardmgr access
- * paths...
- */
- if (!(p_dev->dev.driver == &p_drv->drv))
- p_dev->cardmgr = NULL;
-
- err_put_module:
- module_put(p_drv->owner);
- err_put_driver:
- put_driver(&p_drv->drv);
- err_put:
- pcmcia_put_socket(s);
-
- return (ret);
-} /* bind_request */
-
-#ifdef CONFIG_CARDBUS
-
- static struct pci_bus *pcmcia_lookup_bus(struct pcmcia_socket *s)
- {
- if (!s || !(s->state & SOCKET_CARDBUS))
- return NULL;
-
- return s->cb_dev->subordinate;
- }
-#endif
-
- static int get_device_info(struct pcmcia_socket *s, bind_info_t *bind_info, int first)
- {
- dev_node_t *node;
- struct pcmcia_device *p_dev;
- struct pcmcia_driver *p_drv;
- unsigned long flags;
- int ret = 0;
-
-#ifdef CONFIG_CARDBUS
- /*
```

```
- * Some unbelievably ugly code to associate the PCI cardbus
- * device and its driver with the PCMCIA "bind" information.
- */
- {
- struct pci_bus *bus;
-
- bus = pcmcia_lookup_bus(s);
- if (bus) {
- struct list_head *list;
- struct pci_dev *dev = NULL;
-
- list = bus->devices.next;
- while (list != &bus->devices) {
- struct pci_dev *pdev = pci_dev_b(list);
- list = list->next;
-
- if (first) {
- dev = pdev;
- break;
- }
-
- /* Try to handle "next" here some way? */
- }
- if (dev && dev->driver) {
- strncpy(bind_info->name, dev->driver->name, DEV_NAME_LEN);
- bind_info->major = 0;
- bind_info->minor = 0;
- bind_info->next = NULL;
- return 0;
- }
- }
- }
-#endif
-
- spin_lock_irqsave(&pcmcia_dev_list_lock, flags);
- list_for_each_entry(p_dev, &s->devices_list, socket_device_list) {
- if (p_dev->func == bind_info->function) {
- p_dev = pcmcia_get_dev(p_dev);
- if (!p_dev)
- continue;
- goto found;
- }
- }
- spin_unlock_irqrestore(&pcmcia_dev_list_lock, flags);
- return -ENODEV;
-
- found:
- spin_unlock_irqrestore(&pcmcia_dev_list_lock, flags);
-
- p_drv = to_pcmcia_drv(p_dev->dev.driver);
- if (p_drv && !p_dev->_locked) {
```

[RFC: 2.6 patch] the overdue removal of drivers/pcmcia/pcmcia_ioctl.c

```
- ret = -EAGAIN;
- goto err_put;
- }
-
- if (first)
- node = p_dev->dev_node;
- else
- for (node = p_dev->dev_node; node; node = node->next)
- if (node == bind_info->next)
- break;
- if (!node) {
- ret = -ENODEV;
- goto err_put;
- }
-
- strncpy(bind_info->name, node->dev_name, DEV_NAME_LEN);
- bind_info->major = node->major;
- bind_info->minor = node->minor;
- bind_info->next = node->next;
-
- err_put:
- pcmcia_put_dev(p_dev);
- return (ret);
-} /* get_device_info */
-
-
- static int ds_open(struct inode *inode, struct file *file)
- {
- socket_t i = iminor(inode);
- struct pcmcia_socket *s;
- user_info_t *user;
- static int warning_printed = 0;
-
- ds_dbg(0, "ds_open(socket %d)\n", i);
-
- s = pcmcia_get_socket_by_nr(i);
- if (!s)
- return -ENODEV;
- s = pcmcia_get_socket(s);
- if (!s)
- return -ENODEV;
-
- if ((file->f_flags & O_ACCMODE) != O_RDONLY) {
- if (s->pcmcia_state.busy) {
- pcmcia_put_socket(s);
- return -EBUSY;
- }
- else
- s->pcmcia_state.busy = 1;
- }
-
- }
```

[RFC: 2.6 patch] the overdue removal of drivers/pcmcia/pcmcia_ioctl.c

[RFC: 2.6 patch] the overdue removal of drivers/pcmcia/pcmcia_ioctl.c

```
- user = kmalloc(sizeof(user_info_t), GFP_KERNEL);
- if (!user) {
- pcmcia_put_socket(s);
- return -ENOMEM;
- }
- user->event_tail = user->event_head = 0;
- user->next = s->user;
- user->user_magic = USER_MAGIC;
- user->socket = s;
- s->user = user;
- file->private_data = user;
-
- if (!warning_printed) {
- printk(KERN_INFO "pcmcia: Detected deprecated PCMCIA ioctl "
- "usage.\n");
- printk(KERN_INFO "pcmcia: This interface will soon be removed from "
- "the kernel; please expect breakage unless you upgrade "
- "to new tools.\n");
- printk(KERN_INFO "pcmcia: see http://www.kernel.org/pub/linux/
- utils/kernel/pcmcia/pcmcia.html for details.\n");
- warning_printed = 1;
- }
-
-
- if (s->pcmcia_state.present)
- queue_event(user, CS_EVENT_CARD_INSERTION);
- return 0;
- } /* ds_open */
-
-
- static int ds_release(struct inode *inode, struct file *file)
- {
- struct pcmcia_socket *s;
- user_info_t *user, **link;
-
- ds_dbg(0, "ds_release(socket %d)\n", iminor(inode));
-
- user = file->private_data;
- if (CHECK_USER(user))
- goto out;
-
- s = user->socket;
-
- /* Unlink user data structure */
- if ((file->f_flags & O_ACCMODE) != O_RDONLY) {
- s->pcmcia_state.busy = 0;
- }
- file->private_data = NULL;
- for (link = &s->user; *link; link = &(*link)->next)
- if (*link == user) break;
- if (link == NULL)
```

```
- goto out;
- *link = user->next;
- user->user_magic = 0;
- kfree(user);
- pcmcia_put_socket(s);
-out:
- return 0;
-} /* ds_release */
=
-/*=====*/
=
-static ssize_t ds_read(struct file *file, char user *buf,
- size_t count, loff_t *ppos)
-{
- struct pcmcia_socket *s;
- user_info_t *user;
- int ret;
=
- ds_dbg(2, "ds_read(socket %d)\n", iminor(file->f_dentry->d_inode));
=
- if (count < 4)
- return -EINVAL;
=
- user = file->private_data;
- if (CHECK_USER(user))
- return -EIO;
=
- s = user->socket;
- if (s->pcmcia_state.dead)
- return -EIO;
=
- ret = wait_event_interruptible(s->queue, !queue_empty(user));
- if (ret == 0)
- ret = put_user(get_queued_event(user), (int user *)buf) ? -EFAULT : 4;
=
- return ret;
-} /* ds_read */
=
-/*=====*/
=
-static ssize_t ds_write(struct file *file, const char user *buf,
- size_t count, loff_t *ppos)
-{
- ds_dbg(2, "ds_write(socket %d)\n", iminor(file->f_dentry->d_inode));
=
- if (count != 4)
- return -EINVAL;
- if ((file->f_flags & O_ACCMODE) == O_RDONLY)
- return -EBADF;
=
- return -EIO;
=
```

[RFC: 2.6 patch] the overdue removal of drivers/pcmcia/pcmcia_ioctl.c

```
-} /* ds write */
=  
-/* ===== */
=  
-/* No kernel lock - fine */
-static u_int ds_poll(struct file *file, poll_table *wait)
{
  struct pcmcia_socket *s;
  user_info_t *user;
=  
  ds_dbg(2, "ds_poll(socket %d)\n", iminor(file->f_dentry->d_inode));
=  
  user = file->private_data;
  if (CHECK_USER(user))
  return POLLERR;
  s = user->socket;
  /*
  * We don't check for a dead socket here since that
  * will send cardmgr into an endless spin.
  */
  poll_wait(file, &s->queue, wait);
  if (!queue_empty(user))
  return POLLIN | POLLRDNORM;
  return 0;
  } /* ds poll */
=  
-/* ===== */
=  
-extern int pcmcia_adjust_resource_info(adjust_t *adj);
=  
-static int ds_ioctl(struct inode *inode, struct file *file,
  u_int cmd, u_long arg)
{
  struct pcmcia_socket *s;
  void user *uarg = (char user *)arg;
  u_int size;
  int ret, err;
  ds_ioctl_arg_t *buf;
  user_info_t *user;
=  
  ds_dbg(2, "ds_ioctl(socket %d, %#x, %#lx)\n", iminor(inode), cmd, arg);
=  
  user = file->private_data;
  if (CHECK_USER(user))
  return -EIO;
=  
  s = user->socket;
  if (s->pcmcia_state.dead)
  return -EIO;
=  
  size = (cmd & IOCSIZE_MASK) >> IOCSIZE_SHIFT;
```

[RFC: 2.6 patch] the overdue removal of drivers/pcmcia/pcmcia_ioctl.c

[RFC: 2.6 patch] the overdue removal of drivers/pcmcia/pcmcia_ioctl.c

```
- if (size > sizeof(ds_ioctl_arg_t)) return -EINVAL;
=  
- /* Permission check */  
- if (!(cmd & IOC_OUT) && !capable(CAP_SYS_ADMIN))  
- return -EPERM;  
=  
- if (cmd & IOC_IN) {  
- if (!access_ok(VERIFY_READ, uarg, size)) {  
- ds_dbg(3, "ds_ioctl(): verify read = %d\n", -EFAULT);  
- return -EFAULT;  
- }  
- }  
- if (cmd & IOC_OUT) {  
- if (!access_ok(VERIFY_WRITE, uarg, size)) {  
- ds_dbg(3, "ds_ioctl(): verify write = %d\n", -EFAULT);  
- return -EFAULT;  
- }  
- }  
- buf = kmalloc(sizeof(ds_ioctl_arg_t), GFP_KERNEL);  
- if (!buf)  
- return -ENOMEM;  
=  
- err = ret = 0;  
=  
- if (cmd & IOC_IN) copy_from_user((char *)buf, uarg, size);  
=  
- switch (cmd) {  
- case DS_ADJUST_RESOURCE_INFO:  
- ret = pcmcia_adjust_resource_info(&buf->adjust);  
- break;  
- case DS_GET_CONFIGURATION_INFO:  
- if (buf->config.Function &&  
- (buf->config.Function >= s->functions))  
- ret = CS_BAD_ARGS;  
- else {  
- struct pcmcia_device *p_dev = get_pcmcia_device(s, buf->config.Function);  
- ret = pccard_get_configuration_info(s, p_dev, &buf->config);  
- pcmcia_put_dev(p_dev);  
- }  
- break;  
- case DS_GET_FIRST_TUPLE:  
- mutex_lock(&s->skt_mutex);  
- pcmcia_validate_mem(s);  
- mutex_unlock(&s->skt_mutex);  
- ret = pccard_get_first_tuple(s, BIND_FN_ALL, &buf->tuple);  
- break;  
- case DS_GET_NEXT_TUPLE:  
- ret = pccard_get_next_tuple(s, BIND_FN_ALL, &buf->tuple);  
- break;  
- case DS_GET_TUPLE_DATA:  
- buf->tuple.TupleData = buf->tuple.parse.data;
```

[RFC: 2.6 patch] the overdue removal of drivers/pcmcia/pcmcia_ioctl.c

[RFC: 2.6 patch] the overdue removal of drivers/pcmcia/pcmcia_ioctl.c

```
- buf->tuple.TupleDataMax = sizeof(buf->tuple_parse.data);  
- ret = pccard_get_tuple_data(s, &buf->tuple);  
- break;  
- case DS_PARSE_TUPLE:  
- buf->tuple.TupleData = buf->tuple_parse.data;  
- ret = pccard_parse_tuple(&buf->tuple, &buf->tuple_parse.parse);  
- break;  
- case DS_RESET_CARD:  
- ret = pccard_reset_card(s);  
- break;  
- case DS_GET_STATUS:  
- if (buf->status.Function &&  
- (buf->status.Function >= s->functions))  
- ret = CS_BAD_ARGS;  
- else {  
- struct pcmcia_device *p_dev = get_pcmcia_device(s, buf->status.Function);  
- ret = pccard_get_status(s, p_dev, &buf->status);  
- pcmcia_put_dev(p_dev);  
- }  
- break;  
- case DS_VALIDATE_CIS:  
- mutex_lock(&s->skt_mutex);  
- pcmcia_validate_mem(s);  
- mutex_unlock(&s->skt_mutex);  
- ret = pccard_validate_cis(s, BIND_FN_ALL, &buf->cisinfo);  
- break;  
- case DS_SUSPEND_CARD:  
- ret = pcmcia_suspend_card(s);  
- break;  
- case DS_RESUME_CARD:  
- ret = pcmcia_resume_card(s);  
- break;  
- case DS_EJECT_CARD:  
- err = pcmcia_eject_card(s);  
- break;  
- case DS_INSERT_CARD:  
- err = pcmcia_insert_card(s);  
- break;  
- case DS_ACCESS_CONFIGURATION_REGISTER:  
- if ((buf->conf_reg.Action == CS_WRITE) && !capable(CAP_SYS_ADMIN)) {  
- err = -EPERM;  
- goto free_out;  
- }  
=  
- ret = CS_BAD_ARGS;  
=  
- if (!(buf->conf_reg.Function &&  
- (buf->conf_reg.Function >= s->functions))) {  
- struct pcmcia_device *p_dev = get_pcmcia_device(s, buf->conf_reg.Function);  
- if (p_dev)  
- ret = pcmcia_access_configuration_register(p_dev, &buf->conf_reg);
```

[RFC: 2.6 patch] the overdue removal of drivers/pcmcia/pcmcia_ioctl.c

[RFC: 2.6 patch] the overdue removal of drivers/pcmcia/pcmcia_ioctl.c

```
- pcmcia_put_dev(p_dev);  
- }  
- break;  
- case DS_GET_FIRST_REGION:  
- case DS_GET_NEXT_REGION:  
- case DS_BIND_MTD:  
- if (!capable(CAP_SYS_ADMIN)) {  
- err = -EPERM;  
- goto free_out;  
- } else {  
- static int printed = 0;  
- if (!printed) {  
- printk(KERN_WARNING "2.6 kernels use pcmciamtd instead of memory_cs.c and do not require  
special\n");  
- printk(KERN_WARNING "MTD handling any more.\n");  
- printed++;  
- }  
- }  
- err = -EINVAL;  
- goto free_out;  
- break;  
- case DS_GET_FIRST_WINDOW:  
- ret = pcmcia_get_window(s, &buf->win_info.handle, 0,  
&buf->win_info.window);  
- break;  
- case DS_GET_NEXT_WINDOW:  
- ret = pcmcia_get_window(s, &buf->win_info.handle,  
buf->win_info.handle->index + 1, &buf->win_info.window);  
- break;  
- case DS_GET_MEM_PAGE:  
- ret = pcmcia_get_mem_page(buf->win_info.handle,  
&buf->win_info.map);  
- break;  
- case DS_REPLACE_CIS:  
- ret = pcmcia_replace_cis(s, &buf->cisdump);  
- break;  
- case DS_BIND_REQUEST:  
- if (!capable(CAP_SYS_ADMIN)) {  
- err = -EPERM;  
- goto free_out;  
- }  
- err = bind_request(s, &buf->bind_info);  
- break;  
- case DS_GET_DEVICE_INFO:  
- err = get_device_info(s, &buf->bind_info, 1);  
- break;  
- case DS_GET_NEXT_DEVICE:  
- err = get_device_info(s, &buf->bind_info, 0);  
- break;  
- case DS_UNBIND_REQUEST:  
- err = 0;
```

[RFC: 2.6 patch] the overdue removal of drivers/pcmcia/pcmcia_ioctl.c

```
- break;
- default:
- err = -EINVAL;
- }
=
- if ((err == 0) && (ret != CS_SUCCESS)) {
- ds_dbg(2, "ds_ioctl: ret = %d\n", ret);
- switch (ret) {
- case CS_BAD_SOCKET: case CS_NO_CARD:
- err = -ENODEV; break;
- case CS_BAD_ARGS: case CS_BAD_ATTRIBUTE: case CS_BAD_IRQ:
- case CS_BAD_TUPLE:
- err = -EINVAL; break;
- case CS_IN_USE:
- err = -EBUSY; break;
- case CS_OUT_OF_RESOURCE:
- err = -ENOSPC; break;
- case CS_NO_MORE_ITEMS:
- err = -ENODATA; break;
- case CS_UNSUPPORTED_FUNCTION:
- err = -ENOSYS; break;
- default:
- err = -EIO; break;
- }
- }
=
- if (cmd & IOC_OUT) {
- if ( copy to user(uarg, (char *)buf, size))
- err = -EFAULT;
- }
=
-free out:
-kfree(buf);
-return err;
-} /* ds_ioctl */
=
-/*=====*/
=
-static struct file_operations ds_fops = {
- .owner = THIS_MODULE,
- .open = ds_open,
- .release = ds_release,
- .ioctl = ds_ioctl,
- .read = ds_read,
- .write = ds_write,
- .poll = ds_poll,
-};
=
-void init_pcmcia_setup_ioctl(void) {
- int i;
=
```

[RFC: 2.6 patch] the overdue removal of drivers/pcmcia/pcmcia_ioctl.c

```
- /* Set up character device for user mode clients */  
- i = register_chrdev(0, "pcmcia", &ds_fops);  
- if (i < 0)  
- printk(KERN_NOTICE "unable to find a free device # for "  
- "Driver Services (error=%d)\n", i);  
- else  
- major_dev = i;  
=  
+#ifdef CONFIG_PROC_FS  
- proc_pccard = proc_mkdir("pccard", proc_bus);  
- if (proc_pccard)  
- create_proc_read_entry("drivers", 0, proc_pccard, proc_read_drivers, NULL);  
+#endif  
-}  
=  
=  
-void exit_pcmcia_cleanup_ioctl(void) {  
+#ifdef CONFIG_PROC_FS  
- if (proc_pccard) {  
- remove_proc_entry("drivers", proc_pccard);  
- remove_proc_entry("pccard", proc_bus);  
-}  
+#endif  
- if (major_dev != -1)  
- unregister_chrdev(major_dev, "pcmcia");  
-}  
--- linux-2.6.16-mm1-full/drivers/pcmcia/ds_internal.h.old 2006-03-24 02:06:28.000000000 +0100  
+++ linux-2.6.16-mm1-full/drivers/pcmcia/ds_internal.h 2006-03-24 02:06:41.000000000 +0100  
@@ -1,23 +1,4 @@  
/* ds_internal.h - internal header for 16-bit PCMCIA devices management */  
  
-extern spinlock_t pcmcia_dev_list_lock;  
-extern struct bus_type pcmcia_bus_type;  
=  
-extern struct pcmcia_device * pcmcia_get_dev(struct pcmcia_device *p_dev);  
-extern void pcmcia_put_dev(struct pcmcia_device *p_dev);  
=  
-struct pcmcia_device * pcmcia_device_add(struct pcmcia_socket *s, unsigned int function);  
=  
extern int pcmcia_release_configuration(struct pcmcia_device *p_dev);  
  
+#ifdef CONFIG_PCMCIA_IOCTL  
-extern void init_pcmcia_setup_ioctl(void);  
-extern void exit_pcmcia_cleanup_ioctl(void);  
-extern void handle_event(struct pcmcia_socket *s, event_t event);  
-extern int handle_request(struct pcmcia_socket *s, event_t event);  
+#else  
-static inline void init_pcmcia_setup_ioctl(void) { return; }  
-static inline void exit_pcmcia_cleanup_ioctl(void) { return; }  
-static inline void handle_event(struct pcmcia_socket *s, event_t event) { return; }  
-static inline int handle_request(struct pcmcia_socket *s, event_t event) { return CS_SUCCESS; }
```

[RFC: 2.6 patch] the overdue removal of drivers/pcmcia/pcmcia_ioctl.c

[RFC: 2.6 patch] the overdue removal of drivers/pcmcia/pcmcia_ioctl.c

-#endif

--- linux-2.6.16-mm1-full/drivers/pcmcia/pcmcia_resource.c.old 2006-03-24 02:14:23.000000000 +0100

+++ linux-2.6.16-mm1-full/drivers/pcmcia/pcmcia_resource.c 2006-03-24 02:14:51.000000000 +0100

@@ -271,8 +271,9 @@

* SocketState yet: I haven't seen any point for it.

*/

-int pccard_get_status(struct pcmcia_socket *s, struct pcmcia_device *p_dev,

- cs_status t *status)

+static int pccard_get_status(struct pcmcia_socket *s,

+ struct pcmcia_device *p_dev,

+ cs_status t *status)

{

config_t *c;

int val;

=

To unsubscribe from this list: send the line "unsubscribe linux-kernel" in
the body of a message to majordomo@xxxxxxxxxxxxxxxxx

More majordomo info at <http://vger.kernel.org/majordomo-info.html>

Please read the FAQ at <http://www.tux.org/lkml/>