

support for unicode dead keys

Source: <http://linux.derkeiler.com/Mailing-Lists/Kernel/2005-03/7058.html>

From: Jirka Bohac (jbohac_at_suse.cz)

Date: 03/22/05

Date: Tue, 22 Mar 2005 16:00:35 +0100
To: lkml <linux-kernel@vger.kernel.org>

Hello,

the current keyboard mapping code allows for unicode characters in the keyboard maps. There is also a "compose" table which allows characters to be combined from two (=dead key) or three (=compose key) different keystrokes. This table, however, does not support unicode. This patch adds unicode support for composed chars:

```
diff -Naur 2.6.11-bk-orig/drivers/char/defkeymap.c_shipped
2.6.11-bk-d2/drivers/char/defkeymap.c_shipped
--- 2.6.11-bk-orig/drivers/char/defkeymap.c_shipped 2005-01-20 15:28:26.000000000 +0100
+++ 2.6.11-bk-d2/drivers/char/defkeymap.c_shipped 2005-02-15 16:09:26.000000000 +0100
@@ -222,7 +222,7 @@
     NULL,
 };

-struct kbdiacr accent_table[MAX_DIACR] = {
+struct kbdiacruc accent_table[MAX_DIACR] = {
     {'\'', 'A', '\300'}, {'\'', 'a', '\340'},
     {'\'', 'A', '\301'}, {'\'', 'a', '\341'},
     {'\^', 'A', '\302'}, {'\^', 'a', '\342'},
diff -Naur 2.6.11-bk-orig/drivers/char/keyboard.c 2.6.11-bk-d2/drivers/char/keyboard.c
--- 2.6.11-bk-orig/drivers/char/keyboard.c 2005-01-20 15:28:57.000000000 +0100
+++ 2.6.11-bk-d2/drivers/char/keyboard.c 2005-02-15 16:09:26.000000000 +0100
@@ -329,7 +329,7 @@
 * in utf-8 already. UTF-8 is defined for words of up to 31 bits,
 * but we need only 16 bits here
 */
-static void to_utf8(struct vc_data *vc, ushort c)
+static void to_utf8(struct vc_data *vc, unsigned int c)
 {
     if (c < 0x80)
         /* 0***** */
@@ -391,7 +391,7 @@
 * Otherwise, conclude that DIACR was not combining after all,
 * queue it and return CH.
```

Linux–Kernel: support for unicode dead keys

```
*/
–static unsigned char handle_diacr(struct vc_data *vc, unsigned char ch)
+static unsigned int handle_diacr(struct vc_data *vc, unsigned char ch)
{
    int d = diacr;
    int i;
@@ –614,11 +614,26 @@

static void k_self(struct vc_data *vc, unsigned char value, char up_flag, struct pt_regs *regs)
{
+ unsigned int v;
+
+
    if (up_flag)
        return; /* no action, if this is a key release */

– if (diacr)
– value = handle_diacr(vc, value);
+ if (diacr) {
+ v = handle_diacr(vc, value);
+
+ if (kbd->kbdmode == VC_UNICODE) {
+ to_utf8(vc, v & 0xFFFF);
+ return;
+ }
+
+ /*
+ * this makes at least latin–1 compose chars work
+ * even when using unicode keymap in non–unicode mode
+ */
+ value = v & 0xFF;
+ }

    if (dead_key_next) {
        dead_key_next = 0;
@@ –637,7 +652,7 @@
    {
        if (up_flag)
            return;
– diacr = (diacr ? handle_diacr(vc, value) : value);
+ diacr = (diacr ? handle_diacr(vc, value) & 0xff : value);
    }

/*
diff –Naur 2.6.11–bk–orig/drivers/char/vt_ioctl.c 2.6.11–bk–d2/drivers/char/vt_ioctl.c
— 2.6.11–bk–orig/drivers/char/vt_ioctl.c 2005–01–20 15:28:36.000000000 +0100
+++ 2.6.11–bk–d2/drivers/char/vt_ioctl.c 2005–02–15 20:06:24.000000000 +0100
@@ –583,19 +583,46 @@
    case KDGKBDIACR:
    {
        struct kbdiacrs __user *a = up;
```

Linux–Kernel: support for unicode dead keys

```
+ struct kbdiacr diacr;
+ int i;

        if (put_user(accent_table_size, &a->kb_cnt))
            return -EFAULT;
- if (copy_to_user(a->kbdiacr, accent_table, accent_table_size*sizeof(struct kbdiacr)))
+ for (i = 0; i < accent_table_size; i++) {
+ diacr.diacr = accent_table[i].diacr;
+ diacr.base = accent_table[i].base;
+ diacr.result = accent_table[i].result > 0xff ? 0 : accent_table[i].result;
+ if (copy_to_user(a->kbdiacr + i, &diacr, sizeof(struct kbdiacr)))
+ return -EFAULT;
+ }
+ return 0;
+ }
+
+ case KDGKBDIACRUC:
+ {
+ struct kbdiacrsuc32 __user *a = up;
+ struct kbdiacruc32 diacr;
+ int i;
+
+ if (put_user(accent_table_size, &a->kb_cnt))
            return -EFAULT;
+ for (i = 0; i < accent_table_size; i++) {
+ diacr.diacr = accent_table[i].diacr;
+ diacr.base = accent_table[i].base;
+ diacr.result = accent_table[i].result;
+ if (copy_to_user(a->kbdiacruc + i, &diacr, sizeof(struct kbdiacruc32)))
+ return -EFAULT;
+ }
        return 0;
    }

    case KDSKBDIACR:
    {
        struct kbdiacrs __user *a = up;
+ struct kbdiacr diacr;
        unsigned int ct;
-
+ int i;
+
        if (!perm)
            return -EPERM;
        if (get_user(ct, &a->kb_cnt))
@@ -603,8 +630,37 @@
        if (ct >= MAX_DIACR)
            return -EINVAL;
        accent_table_size = ct;
- if (copy_from_user(accent_table, a->kbdiacr, ct*sizeof(struct kbdiacr)))
+ for (i = 0; i < ct; i++) {
```

Linux-Kernel: support for unicode dead keys

```
+ if (copy_from_user(&diacr, a->kbdiacr + i, sizeof(struct kbdiacr)))
+ return -EFAULT;
+ accent_table[i].diacr = diacr.diacr;
+ accent_table[i].base = diacr.base;
+ accent_table[i].result = diacr.result;
+ }
+ return 0;
+ }
+
+ case KDSKBDIACRUC:
+ {
+ struct kbdiacrsuc32 __user *a = up;
+ struct kbdiacruc32 diacr;
+ unsigned int ct;
+ int i;
+
+ if (!perm)
+ return -EPERM;
+ if (get_user(ct, &a->kb_cnt))
+ return -EFAULT;
+ if (ct >= MAX_DIACR)
+ return -EINVAL;
+ accent_table_size = ct;
+ for (i = 0; i < ct; i++) {
+ if (copy_from_user(&diacr, a->kbdiacruc + i, sizeof(struct kbdiacruc32)))
+ return -EFAULT;
+ accent_table[i].diacr = diacr.diacr;
+ accent_table[i].base = diacr.base;
+ accent_table[i].result = diacr.result;
+ }
+ return 0;
+ }
```

```
diff -Naur 2.6.11-bk-orig/drivers/s390/char/defkeymap.c 2.6.11-bk-d2/drivers/s390/char/defkeymap.c
--- 2.6.11-bk-orig/drivers/s390/char/defkeymap.c 2005-01-20 15:28:52.000000000 +0100
+++ 2.6.11-bk-d2/drivers/s390/char/defkeymap.c 2005-02-15 15:32:01.000000000 +0100
@@ -148,7 +148,7 @@
     0,
 };
```

```
-struct kbdiacr accent_table[MAX_DIACR] = {
+struct kbdiacruc accent_table[MAX_DIACR] = {
    {'^', 'c', '\003'}, {'^', 'd', '\004'},
    {'^', 'z', '\032'}, {'^', '\012', '\000'},
};
```

```
diff -Naur 2.6.11-bk-orig/drivers/s390/char/keyboard.c 2.6.11-bk-d2/drivers/s390/char/keyboard.c
--- 2.6.11-bk-orig/drivers/s390/char/keyboard.c 2005-01-20 15:28:25.000000000 +0100
+++ 2.6.11-bk-d2/drivers/s390/char/keyboard.c 2005-02-15 18:30:57.000000000 +0100
@@ -87,11 +87,11 @@
     goto out_func;
     memset(kbd->fn_handler, 0, sizeof(fn_handler_fn *) * NR_FN_HANDLER);
```

Linux-Kernel: support for unicode dead keys

```
    kbd->accent_table =
- kcalloc(sizeof(struct kbdiacr)*MAX_DIACR, GFP_KERNEL);
+ kcalloc(sizeof(struct kbdiacruc)*MAX_DIACR, GFP_KERNEL);
    if (!kbd->accent_table)
        goto out_fn_handler;
    memcpy(kbd->accent_table, accent_table,
- sizeof(struct kbdiacr)*MAX_DIACR);
+ sizeof(struct kbdiacruc)*MAX_DIACR);
    kbd->accent_table_size = accent_table_size;
    return kbd;

@@ -464,7 +464,6 @@
kbd_ioctl(struct kbd_data *kbd, struct file *file,
          unsigned int cmd, unsigned long arg)
{
- struct kbdiacrs __user *a;
  void __user *argp;
  int ct, perm;

@@ -485,17 +484,45 @@
  case KDSKBSSENT:
      return do_kdgkb_ioctl(kbd, argp, cmd, perm);
  case KDGKBDIACR:
- a = argp;
-
+ {
+ struct kbdiacrs __user *a = argp;
+ struct kbdiacr diacr;
+ int i;
+
      if (put_user(kbd->accent_table_size, &a->kb_cnt))
          return -EFAULT;
- ct = kbd->accent_table_size;
- if (copy_to_user(a->kbdiacr, kbd->accent_table,
- ct * sizeof(struct kbdiacr)))
+ for (i = 0; i < kbd->accent_table_size; i++) {
+ diacr.diacr = kbd->accent_table[i].diacr;
+ diacr.base = kbd->accent_table[i].base;
+ diacr.result = kbd->accent_table[i].result > 0xff ? 0 : accent_table[i].result;
+ if (copy_to_user(a->kbdiacr + i, &diacr, sizeof(struct kbdiacr)))
+ return -EFAULT;
+ }
+ return 0;
+ }
+ case KDGKBDIACRUC:
+ {
+ struct kbdiacrsuc32 __user *a = argp;
+ struct kbdiacruc32 diacr;
+ int i;
+
+ if (put_user(kbd->accent_table_size, &a->kb_cnt))
```

Linux-Kernel: support for unicode dead keys

```
        return -EFAULT;
+ for (i = 0; i < kbd->accent_table_size; i++) {
+ diacr.diacr = kbd->accent_table[i].diacr;
+ diacr.base = kbd->accent_table[i].base;
+ diacr.result = accent_table[i].result;
+ if (copy_to_user(a->kbdiacruc + i, &diacr, sizeof(struct kbdiacruc32)))
+ return -EFAULT;
+ }
        return 0;
+ }
    case KDSKBDIACR:
- a = argp;
+ {
+ struct kbdiacrs __user *a = argp;
+ struct kbdiacr diacr;
+ int i;
+
        if (!perm)
            return -EPERM;
        if (get_user(ct, &a->kb_cnt))
@@ -503,10 +530,37 @@
        if (ct >= MAX_DIACR)
            return -EINVAL;
        kbd->accent_table_size = ct;
- if (copy_from_user(kbd->accent_table, a->kbdiacr,
- ct * sizeof(struct kbdiacr)))
+ for (i = 0; i < ct; i++) {
+ if (copy_from_user(&diacr, a->kbdiacr + i, sizeof(struct kbdiacr)))
+ return -EFAULT;
+ accent_table[i].diacr = diacr.diacr;
+ accent_table[i].base = diacr.base;
+ accent_table[i].result = diacr.result;
+ }
+ return 0;
+ }
+ case KDSKBDIACRUC:
+ {
+ struct kbdiacrsuc32 __user *a = argp;
+ struct kbdiacruc32 diacr;
+ int i;
+
+ if (!perm)
+ return -EPERM;
+ if (get_user(ct, &a->kb_cnt))
            return -EFAULT;
+ if (ct >= MAX_DIACR)
+ return -EINVAL;
+ kbd->accent_table_size = ct;
+ for (i = 0; i < ct; i++) {
+ if (copy_from_user(&diacr, a->kbdiacruc + i, sizeof(struct kbdiacruc32)))
+ return -EFAULT;
```

Linux-Kernel: support for unicode dead keys

```
+ accent_table[i].diacr = diacr.diacr;
+ accent_table[i].base = diacr.base;
+ accent_table[i].result = diacr.result;
+ }
    return 0;
+ }
    default:
        return -ENOIOCTLCMD;
    }
diff -Naur 2.6.11-bk-orig/drivers/s390/char/keyboard.h 2.6.11-bk-d2/drivers/s390/char/keyboard.h
--- 2.6.11-bk-orig/drivers/s390/char/keyboard.h 2005-01-20 15:28:56.000000000 +0100
+++ 2.6.11-bk-d2/drivers/s390/char/keyboard.h 2005-02-15 16:09:26.000000000 +0100
@@ -25,7 +25,7 @@
    unsigned short **key_maps;
    char **func_table;
    fn_handler_fn **fn_handler;
- struct kbdiacr *accent_table;
+ struct kbdiacruc *accent_table;
    unsigned int accent_table_size;
    unsigned char diacr;
    unsigned short sysrq;
diff -Naur 2.6.11-bk-orig/include/linux/kbd_diacr.h 2.6.11-bk-d2/include/linux/kbd_diacr.h
--- 2.6.11-bk-orig/include/linux/kbd_diacr.h 2005-01-20 15:28:19.000000000 +0100
+++ 2.6.11-bk-d2/include/linux/kbd_diacr.h 2005-02-15 16:32:48.000000000 +0100
@@ -2,7 +2,7 @@
#define _DIACR_H
#include <linux/kd.h>

-extern struct kbdiacr accent_table[];
+extern struct kbdiacruc accent_table[];
extern unsigned int accent_table_size;

#endif /* _DIACR_H */
diff -Naur 2.6.11-bk-orig/include/linux/kd.h 2.6.11-bk-d2/include/linux/kd.h
--- 2.6.11-bk-orig/include/linux/kd.h 2005-01-20 15:28:40.000000000 +0100
+++ 2.6.11-bk-d2/include/linux/kd.h 2005-02-15 17:57:08.000000000 +0100
@@ -122,12 +122,25 @@
    unsigned int kb_cnt; /* number of entries in following array */
    struct kbdiacr kbdiacr[256]; /* MAX_DIACR from keyboard.h */
};
+
#define KDGKBDIACR 0x4B4A /* read kernel accent table */
#define KDSKBDIACR 0x4B4B /* write kernel accent table */

+struct kbdiacruc32 {
+ __u32 diacr, base, result;
+};
+struct kbdiacrusc32 {
+ unsigned int kb_cnt; /* number of entries in following array */
+ struct kbdiacruc32 kbdiacruc[256]; /* MAX_DIACR from keyboard.h */
+};
```

support for unicode dead keys

Linux-Kernel: support for unicode dead keys

```
+
+define KDGBKBDIACRUC 0x4BFA /* read kernel accent table - UCS */
+define KDSKBDIACRUC 0x4BFB /* write kernel accent table - UCS */
+
struct kbkeycode {
    unsigned int scancode, keycode;
};
+
#define KDGETKEYCODE 0x4B4C /* read kernel keycode table entry */
#define KDSETKEYCODE 0x4B4D /* write kernel keycode table entry */

diff -Naur 2.6.11-bk-orig/include/linux/keyboard.h 2.6.11-bk-d2/include/linux/keyboard.h
--- 2.6.11-bk-orig/include/linux/keyboard.h 2005-01-20 15:28:20.000000000 +0100
+++ 2.6.11-bk-d2/include/linux/keyboard.h 2005-02-15 16:33:33.000000000 +0100
@@ -28,6+28,12 @@
extern unsigned short *key_maps[MAX_NR_KEYMAPS];
extern unsigned short plain_map[NR_KEYS];
extern unsigned char keyboard_type;
+
+struct kbdiacruc {
+ unsigned char diacr, base;
+ unsigned short result;
+};
+
#endif

#define MAX_NR_FUNC 256 /* max nr of strings assigned to keys */
```

(I did not manage to identify the author of the original patch that this one is based on...)

To take advantage of this kernel patch, a patched version of the kbd package can be used. You can get my kbd patch at <http://jikos.cz/~jbohac/tmp/kbd-1.12-to-2.00.diff>
Please note, that this patch does much more changes to kbd than necessary for unicode compose chars to be used. I will split it into smaller patches eventually...

Regards,

Jirka

```
--
Jirka Bohac <jbohac@suse.cz>
SUSE Labs, SUSE CR
-
```

```
To unsubscribe from this list: send the line "unsubscribe linux-kernel" in
the body of a message to majordomo@vger.kernel.org
More majordomo info at http://vger.kernel.org/majordomo-info.html
Please read the FAQ at http://www.tux.org/lkml/
```