## Appendix D: Replacement of origin driver with the jUSB driver

Once we want to develop a device with the Java USB API it comes to the point where we need to replace the origin driver from the device and force the device to use the JUSB driver. This section explains how we can configure any USB device to the JUSB driver. There are two situations we come across when we attach a device to the USB bus:

- The hardware assistant starts and looks for an appropriate INF file that contains further information about the driver to be installed.
- The operating system automatically loads the appropriate driver for the USB device. This is usually done for devices associated with a class driver. Windows supports all known class drivers from the standard USB specification.

Both situations demand for a different procedure to install the JUSB driver. There are some aspects to concern about, which will be explained in the next two sections.

### I: Install the jUSB driver

Register the JUSB Driver

Before we can configure a USB device to use the JUSB driver, we need to install the driver. If the USB device is configured with an INF file, we can pass to 0. In all other cases, we need to register the JUSB driver to the Windows registry and put the driver file JUSB.sys into the \WINDOWS\SYSTEM32\DRIVERS folder. To register this driver double click on the jusb.reg file in the “\JUSBDriver\Treiber Installations Dateien”. This will add the JUSB driver to the registry. For additional devices we do not have to repeat this process, because the JUSB driver remains in the registry as long as no changes to the registry are made.
The entry can be found in HKEY_LOCAL_MACHINE\SYSTEM\ControlSet001 \Services\JUSB

### II: USB Device with an INF file

If an USB device comes with its own driver attached on an external disk, then in most cases an INF-file is supplied for that driver. The first thing we have to do if we do not know the vendor and product id of the device is reading the first paragraph of 0 to get the id’s. In a second step we do prepare the INF file for the JUSB driver for our own needs. Open the jusb.INF file (should be concerned as a default file) from the *\InstallationFiles\JusbDriver* folder (Table 53 shows the content of jusb.inf file). The INF file contains several sections which are denoted within brackets ( ‘[‘, ‘]’ ). To distinguish this INF file from other INF files using the JUSB driver we save the default jusb.INF file under an appropriate name which belongs to the USB device we want to add, e.g. the device we want to work with is called MyPen [20].

1. Save jusb.inf as e.g. jusb-mypen.inf
2. section **[SourceDiskFiles]**:
   change JUSB.inf to jusb-mypen.inf
3. section **[JavaUSBDevices]**
   change two times the VID and PID according to the VID and PID the USB device is identified.
4. section **[JUSB.Files.Inf]**
   change JUSB.inf to jusb-mypen.inf
5. section **[String]**
   change VID and PID. The DeviceDesc string has to be set, so that it starts with the “**JUSB Driver --:**”.
6. Save jusb-mypen.inf in the *\InstallationFiles\JusbDriver* folder.

```
; Installation inf for the JUSB (Java USB) driver
```

```
;
; (c) Copyright 2003 ETH Zürich Institute for Information Systems
;

[Version]
Signature="$CHICAGO$"
Class=USB
ClassGUID={36FC9E60-C465-11CF-8056-444553540000}
provider=%ETHGLOBIS%
DriverVer=07/28/2003

[SourceDisksNames]
1="JUSB Installation Disk",,,

[SourceDisksFiles]
JUSB.sys = 1
JUSB.inf = 1

[Manufacturer]
%MfgName%=JavaUSBDevices

[JavaUSBDevices]
%USB\VID_0A93&PID_0002.DeviceDesc%=JUSB.Dev, USB\VID_0A93&PID_0002
…

[JUSB.Files.Ext]
JUSB.sys

[JUSB.Files.Inf]
JUSB.Inf

 [Strings]
ETHGLOBIS="ETH Zürich Institute for Information Systems"
MfgName="Stahl"
;FriendlyDeviceName has always to start with "JUSB Driver --:"
;
;In case of a device having a friendly device name starts with "JUSB Driver --:"
;the Java USB API will put that device in the JUSB class, In the other case
;the device will be put to the NonJUSB class.
;
USB\VID_0A93&PID_0002.DeviceDesc="JUSB Driver --:  JUSB Device"
JUSB.SvcDesc="JUSB.Sys Java USB Driver"
```

**Table 53: Fragment of the jusb.inf file. Highlighted are the sections that have to be modified to fit for other USB devices**

Replace the driver using the Device Manager in Windows. According to the example we are looking for a device called MyPen. At this point we can update the driver. The sources for the jUSB driver are found in the *InstallationFiles\JusbDriver*.

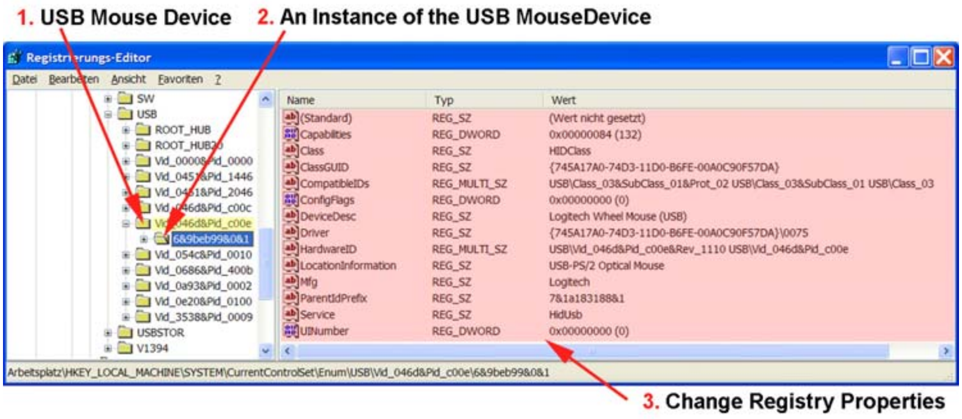
### III: Class USB Devices

A class USB device is usually automatically recognised by the Windows operating system and the corresponding driver is loaded for that device. The hardware assistant may inform that it found a new USB device and its driver has been successfully loaded. If we have such a device, then we need to edit the registry to make that device available to the JUSB driver. The steps are related to a mouse which is the class of a Human Interface Device (HID). A mouse can be attached to the computer and without any settings the mouse can be used. This is exactly what we do first of all, we attach the USB device we want to configure for the JUSB driver to the USB bus. In a second step we need to know the vendor (VID) and product id (PID) of the device. This can be done in the following ways:

- Start RunUSBControllerTest.java. This will do a scan of the USB bus and display its attached devices. With the friendly device name we are able to identify the device we just attached. If we have two identical devices, detach one of them, so we are sure to see only the connected device we are looking for. Look at the uniqueID string to get the vendor id ( Vid_xxxx ) and the product id ( Pid_xxxx ).

- Use the usbview executable from the DDK to get those informations.
- Look in the registry under HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Enum\USB. This approach is like look and guess. The first two approaches are recommended to retrieve the VID and PID.

As soon we know the PID and the VID we start the registry editor (use Start->Run and type “regedit”). Go to the following folder:
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Enum\USB.
In that folder are all USB devices that once have been attached to the USB bus. It contains a ROOT_HUB and a ROOT_HUB20 folder in which the settings and information about the root hubs on the system can be found. Plenty of folders named like Vid_xxxx&Pid_xxxx represent devices belonging to those VID and PID. The mouse we are going to configure for the JUSB driver has VID 046d and PID c00e ( in Figure 20 (1.) a folder name containing the VID and PID).



| Name | Typ | Wert |
|---|---|---|
| (Standard) | REG_SZ | (Wert nicht gesetzt) |
| Capabilities | REG_DWORD | 0x00000084 (132) |
| Class | REG_SZ | HIDClass |
| ClassGUID | REG_SZ | {745A17A0-74D3-11D0-B6FE-00A0C90F57DA} |
| CompatibleIDs | REG_MULTI_SZ | USB\Class_03&SubClass_01&Prot_02 USB\Class_03&SubClass_01 USB\Class_03 |
| ConfigFlags | REG_DWORD | 0x00000000 (0) |
| DeviceDesc | REG_SZ | Logitech Wheel Mouse (USB) |
| Driver | REG_SZ | {745A17A0-74D3-11D0-B6FE-00A0C90F57DA}\0075 |
| HardwareID | REG_MULTI_SZ | USB\Vid_046d&Pid_c00e&Rev_1110 USB\Vid_046d&Pid_c00e |
| LocationInformation | REG_SZ | USB-PS/2 Optical Mouse |
| Mfg | REG_SZ | Logitech |
| ParentIdPrefix | REG_SZ | 7&1a183188&1 |
| Service | REG_SZ | HidUsb |
| UINumber | REG_DWORD | 0x00000000 (0) |

Arbeitsplatz\HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Enum\USB\Vid_046d&Pid_c00e\6&9beb99&0&1

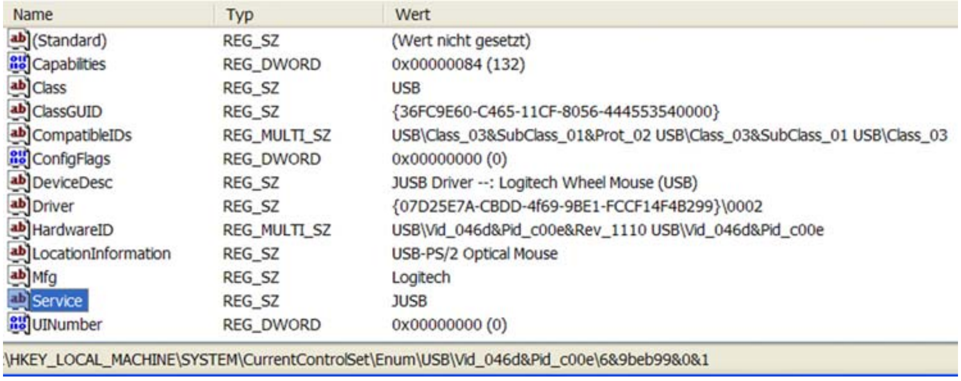
**Figure 20: Registry entries in HKLM\SYSTEM\CurrentControlSet\Enum-\USB**

This means we have found the mouse device in the registry. The subfolders ( Figure 20 (2.)) contain instances of a mouse device. We may have more USB devices with the same VID and PID and for each of those an instance will be created. If we select this instance (in our example there is only one) we get some registry properties on the right hand side To change the properties of that device instance, the security attributes of the instance folder may have to be changed to write access otherwise the registry entry cannot be changed (for more information see 0). Table 54 lists the entries that have to be changed to configure the device for the JUSB driver.

| | |
|---|---|
| Class | **USB** |
| ClassGUID | **{36FC9E60-C465-11CF-8056-444553540000}**<br>Is the system supplied setup class for USB [26]. |
| DeviceDesc | **JUSB Driver --: <the recent DeviceDesc name>** |
| Driver | **{07D25E7A-CBDD-4f69-9BE1-FCCF14F4B299}\nnnn**<br>{07D25E7A-CBDD-4f69-9BE1-FCCF14F4B299}: is the GUID for the JUSB Driver Interface Class. This GUID is defined in JUSBDriver\sys\guids.h.<br>nnnn: a number like 0001. For every JUSB Driver increment this number. There should not exist another JUSB driver with the same number. |
| ParentIdPrefix | **delete** the complete entry |
| Service | **JUSB** |

**Table 54:Change of Regitry Entries for a JUSB Device**

The complete list of settings for a JUSB device is presented in Figure 21



| Name | Typ | Wert |
|---|---|---|
| (Standard) | REG_SZ | (Wert nicht gesetzt) |
| Capabilities | REG_DWORD | 0x00000084 (132) |
| Class | REG_SZ | USB |
| ClassGUID | REG_SZ | {36FC9E60-C465-11CF-8056-444553540000} |
| CompatibleIDs | REG_MULTI_SZ | USB\Class_03&SubClass_01&Prot_02 USB\Class_03&SubClass_01 USB\Class_03 |
| ConfigFlags | REG_DWORD | 0x00000000 (0) |
| DeviceDesc | REG_SZ | JUSB Driver --: Logitech Wheel Mouse (USB) |
| Driver | REG_SZ | {07D25E7A-CBDD-4f69-9BE1-FCCF14F4B299}\0002 |
| HardwareID | REG_MULTI_SZ | USB\Vid_046d&Pid_c00e&Rev_1110 USB\Vid_046d&Pid_c00e |
| LocationInformation | REG_SZ | USB-PS/2 Optical Mouse |
| Mfg | REG_SZ | Logitech |
| Service | REG_SZ | JUSB |
| UINumber | REG_DWORD | 0x00000000 (0) |

\HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Enum\USB\Vid_046d&Pid_c00e\6&9beb99&0&1

**Figure 21: Registry entries for a jUSB device**

In Windows XP a reboot of the system is not required, but at least the device has to be unplugged and plugged again to the bus so the change in the registry takes effect. As soon these properties are set, the device is now being recognised as a JUSB device and will process all request to the JUSB driver. In fact the assumption is made that the JUSB driver is already in the C:\WINDOWS\SYSTEM32\DRIVERS folder. Otherwise it needs to be copied to that location.
The example with the mouse shows the effect of not working correct anymore. The mouse can be moved but no cursor can be seen. In other words, the mouse device is in the control of the JUSB driver, hopefully not out of control!

### IV: How to change Registry Security Attributes

We may encounter the following error messages as displayed in Table 55 when trying to modify the registry values.

| | |
|---|---|
| **Windows 2000/XP:** | Error Editing Value<br>Cannot edit <value>: Error writing the value’s new content |

**Table 55: Error while trying to modify registry entries**

The reason for that failure is because the Windows operating system in the basic setting only grants read access to registry entries. To change the security permission we must log in as system administrator and perform the following steps depending on the Windows version:

**Windows 2000:**

1. Security permissions have to be set with RegEdt32 (Start→Run, enter RegEdt32).
   Select the HKEY_LOCAL_MACHINE window. Open the following folders: *SYSTEM*, *ControlSet001*, *Enum*. Select the *USB* folder and choose Security→Permissions… in the registry editor menu. Give “Everyone” full control access and apply the new changes. Close RegEdt32.
2. Start regedit (Start→Run, enter regedit) and do the necessary changes as described in 0.

**Windows XP:**

1. Start regedit (Start→Run, enter regedit) and choose the folder where new settings have to be applied. Right click and go to Permissions… and set “Everyone” to full control access.
2. In case no Permissions… field is available in the context menu, other settings in the folder options have to be done (Explorer→Tools→Folder Options→View) under Advanced settings, clear “*Use simple file sharing [Recommended]*”. After this setting the security tag should be visible.