Fixing the USB desaster

Gerd Hoffmann, Red Hat
KVM Forum 2011, Aug 16th
USB, one year ago

- Unloved and largely unmaintained.
- The few USB patches often got lost.
- USB 1.1 support only.
- No migration support.
- Known as “The thing which burns CPU when you enable the tablet” ;-)
Very brief USB intro

[root@fedora ~]# lsusb -vs4:5
Device Descriptor:
  iManufacturer           1 QEMU 0.15.50
  iProduct                3 QEMU USB Tablet
Configuration Descriptor:
  Interface Descriptor:
    bInterfaceClass         3 Human Interface Device
    bInterfaceSubClass      0 No Subclass
    bInterfaceProtocol      2 Mouse
Endpoint Descriptor:
  bLength                 7
  bDescriptorType         5
  bEndpointAddress     0x81  EP 1 IN
  bmAttributes            3
    Transfer Type            Interrupt
    Synch Type               None
    Usage Type               Data
  wMaxPacketSize     0x0008  1x 8 bytes
USB endpoints

- 16 endpoints per device.
- 4 endpoint types.
  - control (all devices, endpoint 0)
    - query descriptors, ...
  - bulk (usb-storage).
  - interrupt (mouse, kbd).
  - isochronous (audio, webcams).
- All endpoint transfers are started by the host.
USB hardware polling @ 1000 Hz

UHCI / OHCI host controller

IRQ

NAK

DATA

NAK

NAK

USB Tablet

click
Old and boring.
USB changes in 0.14
USB Descriptor overhaul

- Move from "uint8_t desc[] = { ... }" blobs to "struct USBDesc*".
- General cleanup.
- Needed for USB 2.0 “other speed” descriptors.
- Easy access to device properties, allows to move common emulation bits into generic code.
Physical ports addressing

- `device usb-hub, port=2`
- `device usb-storage, port=2.1, drive=...`
Migration support

• USB devices had no live migration support at all.
• They lost all state, including the device address, thus stopped responding.
• Surprising few problems because of that, guests just reset (HID) devices and go on.
• Fixed now for hub and HID devices.
• Others are still TODO.
  • starting with 0.15 they are at least tagged as being unmigratable.
Remote wakeup

- Devices can send wakeup requests (via hub) to the host controller.
- OS can suspend the USB bus when idle, wait for wakeups, then resume operation.
  - qemu stops burning CPU for usb device polling.
  - reduces power consumption on bare metal.
- QEMU USB hub and HID devices support it now.
- Guests don't use it by default due to broken hardware.
  - udev has rules to enable it for qemu HID devices.
Remote wakeup effect in powertop

- **Idle guest without remote wakeup:**
  Top causes for wakeups:
  
  - 89.2% (1977.3) qemu-kvm : hrtimer_start_range_ns (posix_timer_fn)
  - 4.5% (100.5) <kernel core> : hrtimer_start_range_ns (tick_sched_timer)
  - 3.8% (83.5) <interrupt> : eth0
  - 0.6% (14.0) qemu-kvm : hrtimer_start (kvm_timer_fn)
  - 0.3% (6.7) <kernel core> : hrtimer_start (tick_sched_timer)

- **Idle guest with remote wakeup:**
  Top causes for wakeups:
  
  - 29.2% (73.8) <interrupt> : eth0
  - 29.2% (73.8) qemu-kvm : hrtimer_start_range_ns (posix_timer_fn)
  - 20.9% (52.9) <kernel core> : hrtimer_start_range_ns (tick_sched_timer)
  - 4.5% (11.4) qemu-kvm : hrtimer_start (kvm_timer_fn)
  - 3.2% (8.2) <kernel core> : hrtimer_start (tick_sched_timer)
USB device emulation

- usb-storage
  - USB 2.0 support.
- usb-host
  - iso transfer buffers to keep the data stream going.
  - several webcams (1.1) are working now.
- As usual, bugfixes.
Just released.
USB changes in 0.15
USB 2.0 support

- A whole lot of USB subsystem fixes + cleanups.
  - Parts of them in 0.14 already
  - Device descriptors.
  - Device & Port speed.
- EHCI host adapter emulation.
- Bugfixes & adaptions for usb-host.
USB Companion controllers

```
[root@fedora ~]# lspci -s1d
00:1d.0 USB Controller: Intel Corporation 82801I (ICH9 Family) USB UHCI Controller #1 (rev 03)
00:1d.1 USB Controller: Intel Corporation 82801I (ICH9 Family) USB UHCI Controller #2 (rev 03)
00:1d.2 USB Controller: Intel Corporation 82801I (ICH9 Family) USB UHCI Controller #3 (rev 03)
00:1d.7 USB Controller: Intel Corporation 82801I (ICH9 Family) USB2 EHCI Controller #1 (rev 03)
```

qemu -readconfig docs/ich9-ehci-uhci.cfg
USB Redirection

- Hook up USB devices plugged into a remote machine, with the USB requests traveling over the network.
- TCP transport supported already.
- SPICE support coming soon.
Cutting edge.
USB changes in master
Support scatter lists

- USBPacket buffer is a iovec now.
- Allows to just map the guest buffers and pass them on.
  - avoids extra copying.
  - keeps more state in guest memory, simplifying migration support.
HID separation

- Separated out HID code which has use cases outside USB too.
- Remove funky (ab-)use of the usb devices in bluetooth and milkymist.
There is more.
Future plans for USB
Improve USB endpoints handling

- Maintain state and notify on changes.
- Stop calling into device emulation for each poll.
- Then try optimize host controller emulation to reduce CPU overhead.
  - Not sure it works out for UHCI.
  - xHCI should be alot easier.
UHCI data structures
xHCI data structures

- Device context array
  - Device context base
  - Device context base
  - Device context base
  - ...

- Device context
  - Slot context
  - EP context
  - EP context
  - ...

- TRB Ring
  - Transfer Request Block
  - Transfer Request Block
  - Transfer Request Block
  - ...

- Device context
  - Slot context
  - EP context
  - EP context
  - ...

- TRB Ring
  - Transfer Request Block
  - Transfer Request Block
  - Transfer Request Block
  - ...
Other TODO list items

- Migration support for more devices.
- xHCI emulation & USB 3.0 support.
  - There is some code from Hector Martin.
  - Does pass-through only, sidesteps all qemu USB subsystem issues by winding up libusb directly.
- Use descriptor structs more.
  - Only collected low-hanging fruit now.
- Switch usb-host to libusb?
That's it. Questions?