A whirlwind tour through the input development efforts

Peter Hutterer

XDC 2019, Montreal

Outline

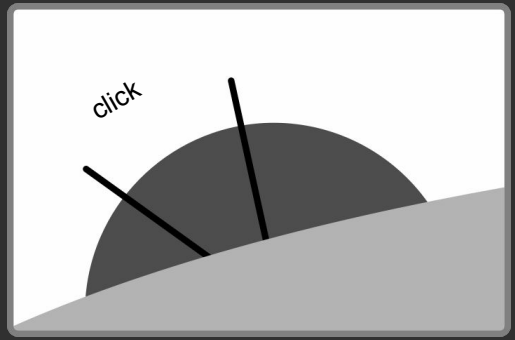
Outline

high-resolution wheel scrolling

hardware

1 detent per 15 or 20°

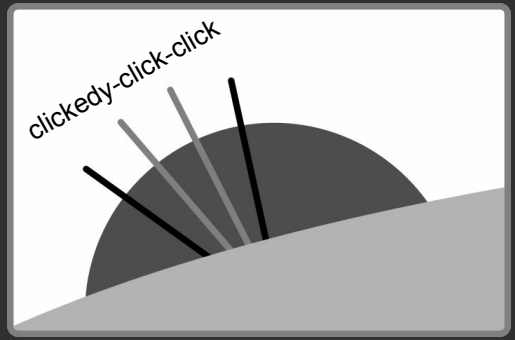
- **value 1 per detent**



hardware

hi-res mice have a "multiplier"

- **multiplier per detent**
  - **4 events with value 1**
  - **1 event with value 4**
  - **2 events with value 2**



firmware

- **HID Resolution Multiplier**
- **Logitech HID++**

kernel 5.0

New axes using a 120 base:

`REL_WHEEL_HI_RES`
`REL_HWHEEL_HI_RES`

This is a separate event stream

```
REL_WHEEL_HI_RES 30
------- SYN_REPORT -------
REL_WHEEL_HI_RES 30
------- SYN_REPORT -------
REL_WHEEL_HI_RES 30
------- SYN_REPORT -------
REL_WHEEL_HI_RES 30
REL_WHEEL        1
------- SYN_REPORT -------
```

libinput

LIBINPUT_EVENT_POINTER_AXIS

- **axis source (wheel, finger, …)**
- **delta in degrees or pixels**
- **discrete steps**

libinput

LIBINPUT_EVENT_POINTER_AXIS_WHEEL

- axis source == wheel
- delta in degrees
- discrete steps == 0
- v120 value

This is a separate event stream

wayland

`wl_pointer.axis_source`

`wl_pointer.axis` (pixels)

`wl_pointer.axis_discrete`

`wl_pointer.frame`

We only have one `wl_pointer`

wayland

Issues:

- no-one likes discrete 0
- merged `wl_pointer`
- magic distance 10
- `wl_pointer.axis_source` needs `wl_pointer.axis`

wayland

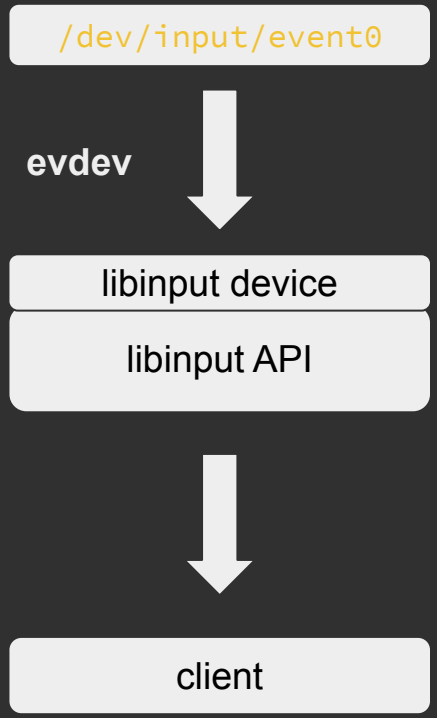
wl_pointer.pointer_axis_v120

Needs to be a separate event stream

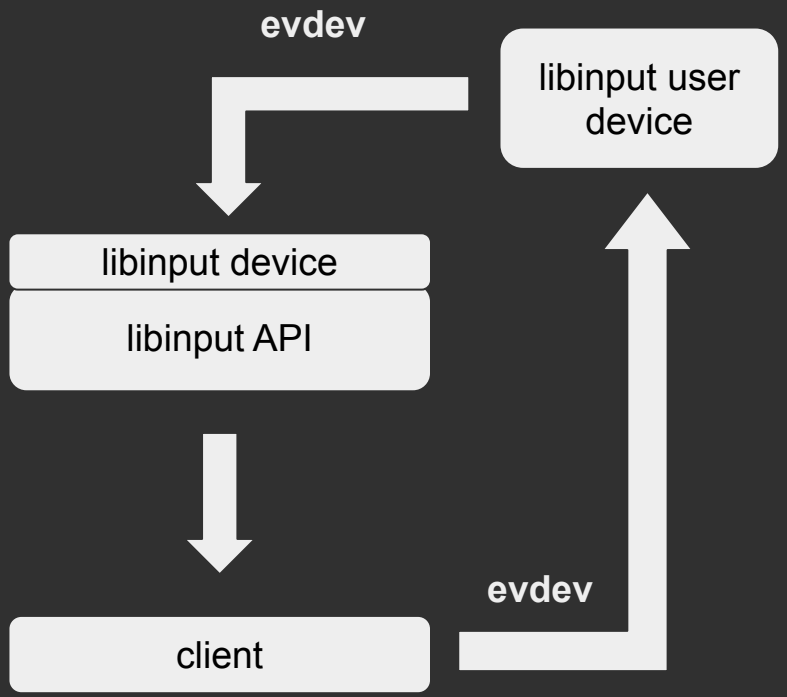
```
wl_pointer.axis_source   wheel
wl_pointer.axis_v120     40
wl_pointer.frame
wl_pointer.axis_source   wheel
wl_pointer.axis_v120     40
wl_pointer.frame
wl_pointer.axis_source   wheel
wl_pointer.axis_v120     40
wl_pointer.axis          10
wl_pointer.axis_discrete 1
wl_pointer.frame
```

libinput user devices

kernel devices



user devices



but why?

libinput test suite

- takes ~60min
- timeout-based failures
- needs root
- needs uinput
- needs udev
- messes with the session

user devices

- any fd will do
- no udev dependency
- potential gettime-fd

not in my lib!

Issues:

- udev leaks into the API
- quirks … ???

`libinput-testing.so`

XKB user configuration

changing a single key is hard!

XKB

Required:

- a custom layout
- include directory shuffling

```
partial alphanumeric_keys
xkb_symbols "custom" {
      include "us(basic)"
      name[Group1]= "English (US)";
      key.type[group1]="ONE_LEVEL";

      key <TLDE> { [ VoidSymbol ] };
};
```

xmodmap

```
keycode 49 = NoSymbol
```

XKB pieces

RMLVO → KCGST

- rules: everyone uses "evdev"
- models: everyone uses "evdev"

`evdev.xml` for GUIs

rules files

```
! option     =    symbols
  grp:shift_toggle = +group(shifts_toggle)
  altwin:menu      = +altwin(menu)
  altwin:meta_alt  = +altwin(meta_alt)
  altwin:alt_win   = +altwin(alt_win)
  altwin:ctrl_win  = +altwin(ctrl_win)

! model     =     keycodes
  pc98      =     evdev(pc98)
  applealu_jis  =       evdev+macintosh(jisevdev)
 $jollamodels   =       evdev+jolla(jolla)
  olpc      =     evdev+olpc(olpc)
  olpcm     =     evdev+olpc(olpcm)
  *         =     evdev
```

Not enough useful wildcards

user rules

- Default to `$HOME/.xkb/rules`
- Add `include` statement

```
! option     = symbols
  totally:awesome   = +some(option)

! include %S/evdev
```

user rules

`evdev.xml` parsers

- libgnome-desktop
- Qt
- libxklavier

```
<xi:include href="...">
```

`evdev.lst` parsers:

- loginctl

ratbag-emu

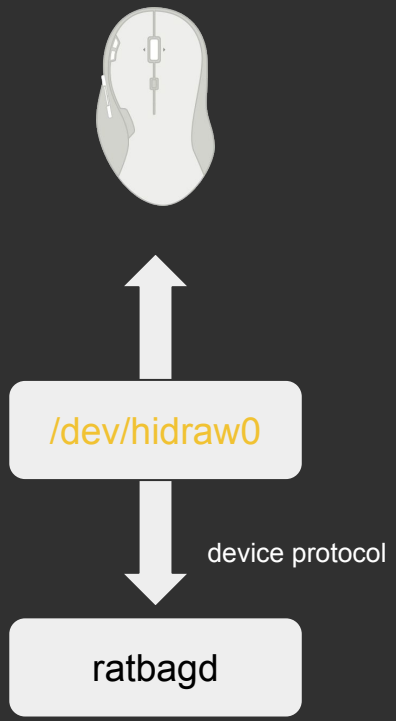
Filipe Laíns, Logitech

mouse firmware emulator

https://github.com/libratbag/ratbag-emu

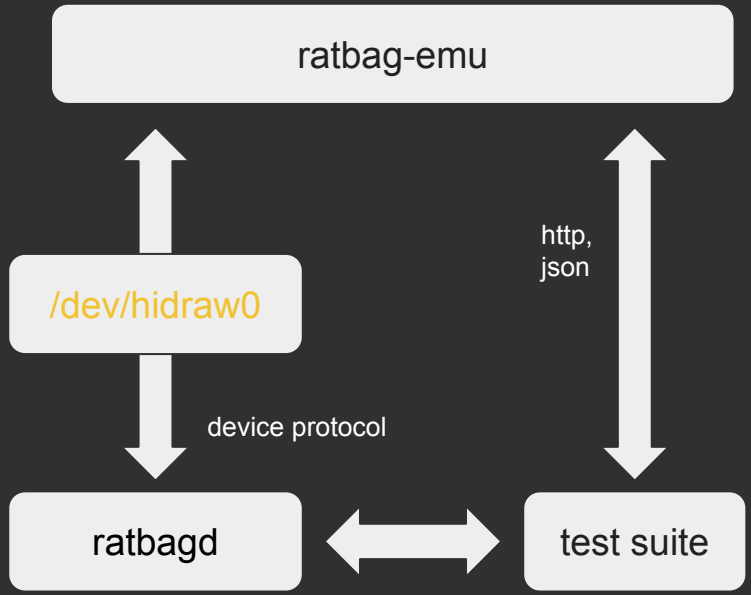
ratbag-emu

Filipe Laíns, Logitech



ratbag-emu

Filipe Laíns, Logitech



Outline

Totem

dell.com

Totem

kernel 4.19

- `ABS_MT_TOOL MT_TOOL_DIAL`

libinput

- `LIBINPUT_TABLET_TOOL_TYPE_TOTEM`
- `libinput_event_tablet_tool_get_size_major`
- `libinput_event_tablet_tool_get_size_minor`

Wayland (proposed)

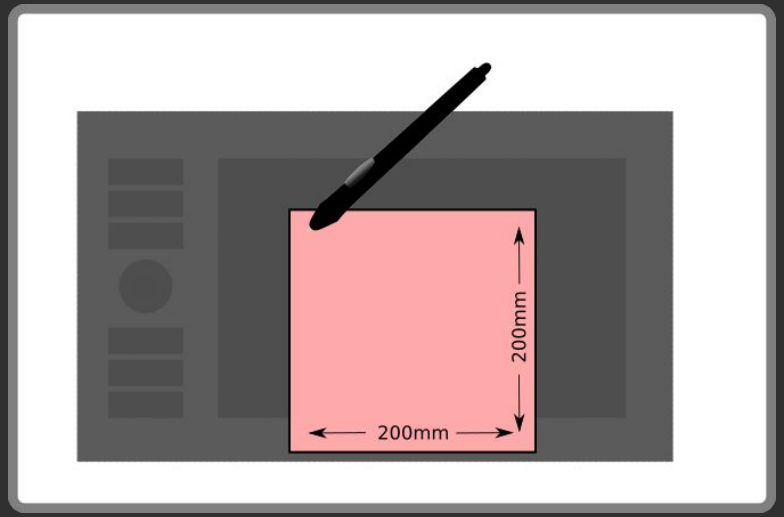
- `wl_tablet_tool.type`
- `wl_tablet_tool.size`

Totem

Applications:

- none
- ???
- yeah, right

Touch arbitration



Only on screen tablets atm

Tuhi

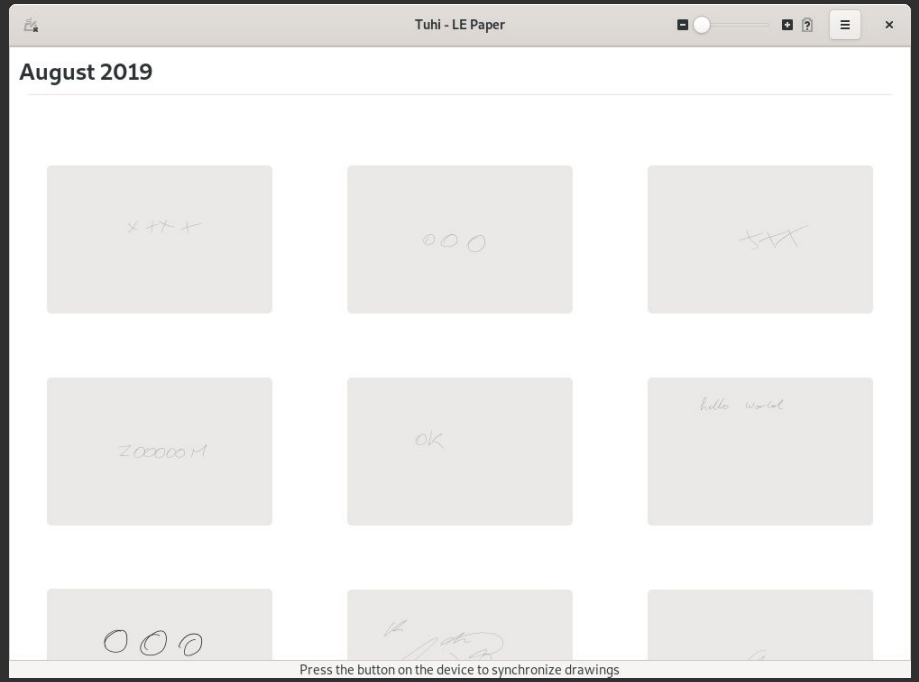
by Jakub Steiner

GTK application for Wacom Smartpads

- Bamboo Spark
- Bamboo Folio
- Bamboo Slate
- Intuos Pro

http://github.com/tuhiproject/tuhi

Tuhi


Tuhi - LE Paper
August 2019
Press the button on the device to synchronize drawings

xf86-input-evdev

- last commit May 2018
- 2.10.0 release 4 years ago
  - 19 commits since

xf86-input-synaptics

- 1.9.0 release Nov 2016
  - 9 commits since

The libinput party bus

- libinput 1.9.0..
  - ~1100 commits
  - ~980 by me
  - crickets tag: help needed
- gitlab a mixed blessing
  - much more efficient
  - CI is a blessing (-ish)
  - but zero reviews

libratbag

by Andy Fitzsimon

- ETIME/EINTR
- Still no stable DBus API

libevdev

nothing to report

libevdev

python-libevdev

https://readthedocs.org/projects/python-libevdev/

libinput quirks

devices need "this is broken" tags

udev hwdb

- free key/value storage
- arbitrary lookup rules

udev hwdb

hwdb become unmaintainable

- too hard to debug
- `udevadm hwdb --update` vs `systemd-hwdb update`
- disconnected "configuration"

libinput quirks

```
[Logitech M570]
MatchName=*Logitech M570*
ModelTrackball=1

# Marble Mouse claims to have a middle button
[Logitech Marble Mouse Trackball]
MatchUdevType=mouse
MatchBus=usb
MatchVendor=0x46D
MatchProduct=0xC408
AttrEventCodeDisable=BTN_MIDDLE

[Logitech K400]
MatchUdevType=mouse
MatchBus=usb
MatchVendor=0x046D
MatchProduct=0x4024
ModelBouncingKeys=1
```

hid-generic rebinding

hid-generic unbinds when there's a better driver available

libinput record

Replacement for **`evemu`**

- **YAML instead of custom format**

Debugging tool in the libinput git repo

Outline

hold gestures

```
HOLD BEGIN (1fg)

HOLD CANCEL

POINTER MOTION
```

```
HOLD BEGIN (2fg)

HOLD CANCEL

POINTER AXIS

HOLD BEGIN

HOLD END
```

Summary

- high resolution wheel scrolling
- libinput user devices
- XKB user configuration
- ratbag-emu firmware emulator
- Dell Canvas Totem
- Touch arbitration
- Tuhi for Wacom Smartpads
- xf86-input-evdev/synaptics in maintenance mode
- libinput's bus factor is bad
- libinput quirks/record/replay
- hold gestures