

# Profile Manager

## Uživatelská dokumentace

Jan Škoda

2. dubna 2014

Profile manager je nástroj pro detekci přítomnosti notebooku v několika předdefinovaných lokacích a přizpůsobení pracovního prostředí pro tyto lokace. Je určen pro OS Linux a pomocí vkompilovaných i externích modulů vnímá *smysly* stav periferií notebooku, určuje *lokaci* a podle ní *akcemi* ovlivňuje chování systému. To vše je konfigurovatelné v textovém souboru a rozšiřitelné externími moduly implementovanými v libovolném programovacím jazyce.

## Spuštění a závislosti

Aplikace je implementovaná v jazyce C++ a využívá knihovny Boost, Google logging library, xrandr, x11, gtk3, inotifytools a usb-1.0. Pro použití modulu *wifi\_scan* je navíc potřeba balík *sudo*. Vše je dostupné ve všech obvyklých balíčkovacích systémech. Rozhraním je téměř výlučně konfigurační soubor, dále je program spouštěn z příkazové řádky a může být použit i jako systémová služba. Příklad spuštění programu a zároveň ukázka použití jediného argumentu:

```
./profile_manager -C /cesta/k/konfiguracnimu_souboru.cfg
```

Pokud není cesta k konfiguraci uvedena argumentem, hledá program soubor `profile_manager.conf` v aktuální složce. Program je možné bezpečně ukončit stiskem *Ctrl-C* nebo signálem *SIGTERM*.

## Příklad konfigurace

Uvažme notebook, který se běžně přesouvá mezi lokacemi **doma** a **kolej**. Kromě toho je k němu občas ve škole připojen **projektor**. Doma bude k notebooku připojena USB myš, na koleji zase USB myš, klávesnice a externí monitor, který rozšiřuje pracovní plochu doprava nahoru. Naopak při připojení projektoru chceme nastavit klonování obrazu. Doma i na koleji je navíc k dispozici bezdrátová síť.

Tuto situaci můžeme obsluhovat následující konfigurací:

```
[senses]
monitor_kolej = monitor.id=Monitor-XYZ
monitor_nejaky = monitor.id=Any
kolej_wifi = wifi.id=Kolej_wifi
doma_wifi = wifi.id=Domaci_wifi

doma_mys = usb.id=046d:c03f
kolej_kbd = usb.id=04fc:05d8
kolej_mys = usb.id=04da:046f

[actions]
monitor_vpravo = xrandr.vga=right-of
monitory_zarovnane_dole = xrandr.vga-pos=1280x-224
monitor_rozliseni = xrandr.vga-resolution=1280x1024

monitor_klonovani = xrandr.vga=clone
monitor_pomer = xrandr.vga-resolution=1024x768
```

```
[locations]
kolej = kolej_wifi & (kolej_kbd | kolej_mys)
doma = doma_wifi & doma_mys
projektor = monitor_nejaky & !monitor_kolej

[rules]
kolej = monitor_rozliseni,monitory_zarovnane_dole,monitor_vpravo
projektor = monitor_klonovani,monitor_pomer
```

## Popis konfigurace

Celá konfigurace je rozdělena do sekcí *senses*, *actions*, *locations* a *rules*. Každá konfigurační položka (řádka) má formát **identifikátor = definice**. Definice je buďto infixový výraz (v sekci *locations*), seznam předtím definovaných identifikátorů (sekce *rules*) nebo rovnost (v sekcích *senses* a *actions*).

V sekci **locations** definujeme lokace, ve kterých se může notebook nacházet. To je pro maximální flexibilitu definováno infixovým výrazem složených z identifikátorů smyslů. Takže například v lokaci `loc = a & b` jsme tehdy, pokud platí smysly *a* a *b*. Podporovány jsou operátory `&`, `|`, `^`, `!` a samozřejmě závorky.

V sekci **rules** definujeme seznam akcí, které se mají provést v dané lokaci. Jednotlivé akce jsou zadány identifikátorem akce ze sekce *actions* a odděleny čárkami.

V sekci **senses** definujeme tzv. smysly. Jsou to identifikátory, které buďto platí anebo ne a tím vypovídají o stavu periférií notebooku. Každý smysl testuje jednu proměnnou nějakého smyslového modulu na rovnost oproti nějaké hodnotě. Například `mysl = monitor.id=Acer XYZ` testuje zda je proměnná `id` modulu `monitor` rovna "Acer XYZ". Jinými slovy jestli je připojen tento monitor.

V sekci **actions** definujeme akce, kterými můžeme ovlivňovat konfiguraci počítače podle detekované lokace. Konfigurační položky (řádky) mají stejnou syntaxi jako v sekci *senses*, jen je sématika operátoru `=` přiřazení namísto porovnání.

## Moduly a jejich vlastnosti

### Smyslové moduly

**monitor** je smyslový modul s jedinou proměnnou *id*. Ta je rovna identifikátoru zařízení, nebo *Any*, pokud je připojeno VGA zařízení. V opačném případě je *id* rovno *None*.

**wifi** je smyslový modul s jedinou proměnnou *id*. Ta je rovna ESSID připojené WiFi sítě.

**wifi\_scan** je smyslový modul s jedinou proměnnou *id*. Ta je rovna ESSID všech WiFi sítí v okolí. Aby mohl být modul použit, musíte dovolit vašemu uživateli přístup ke scannování WiFi sítí. To realizujte tak, že přidělíte svému uživateli *sudo* práva na spuštění příkazu `iw dev wlan0 scan` (vizte `man xrandr`).

**usb** je smyslový modul s jedinou proměnnou *id*. Ta je rovna identifikátorům všech připojených USB zařízení. Identifikátor je ve tvaru `vendorId:productId`, kde obě *id* jsou malými písmeny zapsané 4znakové hexa stringy vč. počátečních nul. Tento identifikátor se dá získat například příkazem `lsusb`.

### Akční moduly

**xrandr** má 3 proměnné:

- *vga* – Pozice externí obrazovky. Možné hodnoty: *right-of*, *left-of*, *below*, *above*.

- *vga-pos* – Posun externí obrazovky v pixelech. Např. *1280x-224*. Vizte `man xrandr`, .
- *vga-resolution* – Rozlišení externí obrazovky. Např. *1024x768*. Seznam rozlišení podporovaných vaším výstupem zjistíte spuštěním příkazu `xrandr`.

**execute** má jedinou proměnnou *command*. Její nastavení spustí zadaný příkaz, kterým je obvykle nějaký uživatelský bash skript. Modul execute nepodporuje argumenty!