

Triky ve Vimu (i bez pluginů)

■ Autor Pavel Tišnovský, Red Hat
■ Email <ptisnovs 0x40 redhat 0x2e com>
■ Datum 2019-10-06

Obsah přednášky (1)

- ▶ Začínáme...
- ▶ Základní konfigurační volby
- ▶ Vyhledávání v textu
- ▶ Nahrazení
- ▶ Volání externích nástrojů, změna aktuálního dokumentu
- ▶ Volby způsobu zobrazení textů
- ▶ Tabby nebo mezery?
- ▶ Nastavení wrap/showbreak/linebreak/breakindent
- ▶ Chování textového kurzoru při pohybu v textu
- ▶ Stroj času zabudovaný do Vimů

Obsah přednášky (2)

- ▶ Trocha matematiky
- ▶ Zneužití speciálního registru = (vložení číselné sekvence)
- ▶ Vizualní režim a režim výběru
- ▶ Výběr vertikálních bloků, práce s bloky
- ▶ Vertikální bloky a funkce typu „najdi a nahrad“
- ▶ Režim virtuální editace
- ▶ Chování myši, výběr bloků přes Shift+..., celkové chování Vimů
- ▶ Chování editoru při chybě
- ▶ Zvýraznění aktuálního řádku/sloupce
- ▶ Digraphs (dvojznaky)

Obsah přednášky (3)

- ▶ Režim „ex“
- ▶ Režim „ed“
- ▶ Zobrazení informací o pozici v textu a o vybraném znaku
- ▶ Další klávesové zkratky
- ▶ Zvýraznění syntaxe
- ▶ Pravítko a stavový řádek
- ▶ Užitečné externí nástroje
- ▶ Lepší využití klávesy Caps Lock
- ▶ Vtípky
- ▶ Odkazy

Začínáme...

- ▶ Otevření souboru editovaného minule
\$ vim
 - ◆ otevře se prázdný nepojmenovaný dokument
- ▶ Poté stisknout Ctrl+o Ctrl+o
...voilà

Základní konfigurační volby

-
- ▶ Vypnutí kompatibility s editorem Vi
 - :set nocompatible
 - ▶ Zobrazení pravítka
 - :set ruler
 - ▶ Zobrazení a nastavení obsahu pravítka
 - :set laststatus=...
 - :set statusline=...
 - ▶ Expanze či zákaz expanze tabů
 - :set (no)expandtab
 - :retab
 - ▶ Chování klávesy backspace
 - :set backspace=[indent,eol,start]

Základní konfigurační volby (pokračování)

- ▶ Nastavení typu souboru (konce řádků)
 - :set filetype=[dos,unix]
- ▶ Chování bloků
 - :set selection=exclusive
 - ♦ bloky se chovají podobně jako v ostatních editorech
 - ♦ (bude ukázáno dále)
- ▶ Seznam souborů atd. po TAB
 - :set wildmenu
- ▶ Některé soubory je vhodné ignorovat
 - :set wildignore=*~,*.bak,*.log,*.aut,*.dvi,*.o

Kde se změnilo nastavení konkrétní volby???

- ▶ Může se jednat o více míst
 - ♦ systémový .vimrc
 - ♦ uživatelský .vimrc
 - ♦ .vimrc v aktuálním adresáři
 - ♦ atc.
- ▶ Zjištění zdroje poslední změny
 - :verbose set nocompatible?
 - :verbose map <Space>

Vyhledávání v textu

- ▶ Nalezení slova pod kurzorem
 - ♦ směrem ke konci textu
 - * celá slova
 - g* i části slov
 - ♦ směrem k začátku textu
 - # celá slova
 - g# i části slov
- ▶ Vyhledání (a doskok) na párovou závorku
 - %

Vyhledávání v textu

- ▶ Vyhledání se zobrazením výskytů A čísel řádků
 - :g/tab/#

- `:set ignorecase`
- `:set smartcase`
- V příkazovém režimu (command mode)
 - ◆ `Ctrl+R Ctrl+W`
vložení slova (jako *) na příkazovou řádku
ideální pro příkaz /
 - ◆ `Ctrl+R Ctrl+A`
vložení slova (jako #) na příkazovou řádku
ideální pro příkaz /

Vyhledávání v textu

- Zobrazení historie hledání
`:his s`
- Zobrazení okna s historií hledání
`q/`
 - ◆ (uzavře se pomocí `:q`)

Nahrazení

- Jediný výskyt na řádku
`:s/foo/bar/foo`
- Jediný výskyt na každém řádku v celém souboru
`:%s/foo/bar/`
- Všechny výskyty na řádku
`:s/foo/bar/g`
- Výskyty schválené uživatelem
`:s/foo/bar/gc`
- Všechny výskyty `public => public int`, ovšem jen pokud již není nahrazeno
`:s/public\(int\)\@!/public int/g`
- Zopakování posledního příkazu pro náhradu
`:&`

Vložení výsledku příkazu do souboru na aktuální řádek

```
:.!date  
:.!whoami
```

Hexa editor (aplikace současný soubor)

- Nástroj `xxd` je součástí instalace Vimů (`vim-enhanced`)
 - `:%!xxd`
 - `:%!xxd -r`

Volby způsobu zobrazení textů

- Zobrazení čísel řádků
`:set number`
- Zobrazení relativních čísel (aktuální řádek=0)
`:set relativenumber`
- Zkombinovat lze i obě předchozí volby
 - ◆ relativní řádky + aktuální řádek má absolutní index
- Zobrazení řídicích znaků

```
:set list
```

Volby způsobu zobrazení textů (pokračování)

- ▶ Způsob „zobrazení“ tabulačních zarážek

```
:set tabstop=xxx
```
- ▶ Rychlý doskok na párovou závorku

```
:set showmatch
```
- ▶ Definice párových znaků (typicky závorek)

```
:set matchpairs=(:),[:],{:},<:>
```
- ▶ Způsob zobrazení řídicích znaků

```
:set display
```

 - ◆ (viz další slajdy)

Relativní čísla řádků

- ▶ Pomáhá při mnoha operacích typu 5j 10dd atd.

```
:set relativenumber
```

Zobrazení řídicích znaků

- ▶ Ve formátu ^znak
 - ◆ například ^K
- ▶ Hexa hodnota znaku
 - ◆ například <0b>
- ▶ Změna nastavení

```
:set display=
```

```
:set display=uhex
```

Taby nebo mezery?

- ▶ Viditelné přes `:set list`
- ▶ Lze vytvořit zvláštní syntaktické kategorie

```
:syn match Tab "\t"
:syn match Tab2 "\t\t"
:hi def Tab  ctermbg=lightgreen guibg=#e0ffe0
:hi def Tab2 ctermbg=lightred  guibg=#ffe0e0
```

Zalamování textu

- ▶ Pouze vizuální zalamování (do souborů se nevkládá CR)

```
:set wrap
```
- ▶ Vypnutí vizuálního zalamování

```
:set nowrap
```
- ▶ Vizuální zalamování na hranicích slov (vhodné pro texty)

```
:set linebreak
```
- ▶ Vypnutí vizuálního zalamování na hranicích slov

```
:set nolinebreak
```
- ▶ Označení zalomených (obrazových) řádků znakem/znaky

```
:set showbreak=.
```

```
:set showbreak=\\
```
- ▶ Automatické zalamování na xxx sloupci (vkládá se CR)

```
:set textwidth=xxx
```

```
:set tw=xxx
```

Volba wrap a řádky na konci obrazovky

- ▶ Buď @ v případě dlouhého řádku
 - ◆ dlouhodobá tradice
- ▶ Popř. „zobraz co nejvíce skutečného textu“
 - :set display+=lastline

Nastavení breakindent

- ▶ Bere do úvahy odsazení řádku
 - ◆ obrazovkové řádky jsou odsazené též
- ▶ Kombinace s linebreak a wrap
- ▶ Samozřejmě též se showbreak
- ▶ Použití
 - ◆ (La)TeX
 - ◆ AsciiDoc
 - ◆ Markdown
 - ◆ plain old ASCII text
 - ◆ ...

Nastavení breakindent (ukázky)

1. nowrap
2. wrap
3. wrap + showbreak=...
4. wrap + linebreak
5. wrap + breakindent
6. wrap + linebreak + breakindent

Chování textového kurzoru při pohybu v textu

- ▶ Dostatek zobrazeného textu před/za aktuálním řádkem
 - :set scrolloff=...
 - ◆ scroll offset (ne scroll off)
- ▶ Naváže dvě (či více oken) pro synchronizaci posunu
 - :set cursorbind
 - ◆ význam při použití ve více oknech současně

Stroj času zabudovaný do Vimů

- ▶ Takzvané „undo branches“
- ▶ Posun mezi jednotlivými verzemi dokumentu
 - g-
 - g+
 - ◆ pochopitelně lze zkombinovat s počtem opakování

Stroj času zabudovaný do Vimů

- ▶ Posun na předchozí verze dokumentu
 - :earlier {count} přechod na předchozí stav souboru
 - :earlier {N}s lze zadat časový interval v sekundách
 - :earlier {N}m nebo minutách
 - :earlier {N}h dokonce i hodinách

- ▶ :earlier {N}d pokud pracujete nepřetržitě několik dnů... :-)
- ▶ Opak operace :earlier
 :later
- ▶ Postupné odvolání změn
 Ctrl+R

Trocha matematiky

- ▶ Zvýšení a snížení hodnoty, na které je kurzor nebo první hodnoty ZA kurzorem
 - ◆ CTRL+A
 - ◆ CTRL+X
- ▶ (Trik) doskok na první číslo (jedno jaké, s několika výjimkami)
 - ◆ CTRL+A CTRL+X
- ▶ Výpočty ve vkládacím režimu
 - ◆ CTRL+R=6*7
 - ◆ CTRL+R=0xffffffff
 - ◆ CTRL+R=pow(2,10)

Zneužití speciálního registru =

- ▶ Vložení sekvence čísel do souboru v insert režimu
 - ◆ CTRL+R=range(1,10)
 - ◆ CTRL+R=range(1,20,3)
 - ◆ CTRL+R=range(10,0,-2)

Vizuální režim versus režim výběru

- ▶ Selection mode
 - ◆ částečně podobné chování editorů v MS Windows
 - ◆ lze vybírat pomocí Shift+šipky atd.
 - (viz dále popsaná volba keymodel)
 - ◆ pohyb bez Shift režim výběru ukončí
 - Esc taktéž režim výběru ukončí
 - ◆ po výběru a stisku viditelného znaku se:
 - výběr (vybraný text) vymaže
 - editor se přepne do insert režimu
 - ◆ lze nastavit i další možnosti pomocí
 - :set selectmode
 - :set selection
 - :set keymodel

Vizuální režim versus režim výběru

- ▶ Visual mode
 - ◆ výběr začíná v, V, Ctrl+V
 - ovšem tato volba může chování změnit
 - :set selectmode
 - ◆ fungují zde běžné operátory c, y, d
 - ◆ obnovení původního výběru: gv
 - ◆ lze nastavit i další možnosti pomocí
 - :set selection

Vizuální režim versus režim výběru

:set selectmode=seznam_hodnot

Hodnota	Význam
mouse	select mode pomocí myši
key	select mode pomocí shift+šipky
cmd	select mode začíná v, V, Ctrl+V

Výběr textu

:set selection=

- ♦ vybírá se znak pod kurzorem?
- ♦ lze vybrat znak za koncem řádku?

Hodnota	Za koncem řádku	Znak pod kurzorem
old	ne	ano
inclusive	ano	ano
exclusive	ano	ne

Výběr vertikálních bloků, práce s bloky

- ▶ Standardně klávesová zkratka Ctrl+V
- ▶ IMHO lepší nastavení
 - :noremap ,v <C-V>
 - ♦ nekoliduje s případným mapováním Ctrl+V na „paste“
- ▶ Běžné operátory c, y, d
- ▶ Při „yank“ a „delete“ se zapamatuje, že se jedná o blok
 - ♦ jak?
 - ♦ trik s Ctrl+J
- ▶ Možnost kombinace s „virtualedit“ při nestejně dlouhých řádcích
 - ♦ viz další slajd

Vertikální bloky a funkce typu „najdi a nahraď“

- ▶ Po výběru bloku stisk :
- ▶ Lze zapsat například
 - : '<,'>s-foo-bar-g
- ▶ Ovšem náhrada se provede v celém rozsahu textu od prvního vybraného řádku do řádku posledního
- ▶ Lze náhradu provést skutečně jen ve vybraném vertikálním bloku?
- ▶ Řešení
 - ♦ použití \%V
 - : '<,'>s-\%Vfoo-bar-g

Režim virtuální editace

:set virtualedit=

Hodnota	Význam
---------	--------

block	povolení virtuální editace při práci s vizuálními bloky
insert	povolení virtuální editace ve vkládacím režimu
all	povolení virtuální editace ve všech režimech
onemore	umožňuje přesun kurzoru pouze ZA poslední znak

Režim virtuální editace

(ukázka s vertikálními bloky)

Volba mousemodel

- Určuje chování myši
 - ◆ rozdíl X/MS Windows
 - ◆ většinou se týká pravého popř. prostředního tlačítka myši

Volba keymodel

- Týká se výběru bloků pomocí Shift+posun
- ```
:set keymodel
```

| Volba                   | Význam                                                             |
|-------------------------|--------------------------------------------------------------------|
| startsel, stopsel<br>"" | výběr pomocí Shift+šipky<br>skok na další slovo pomocí Shift+šipky |

## Celkové chování editoru

- Nastavení několika voleb jediným příkazem :behave
 

```
:behave mswin
```

```
:behave xterm
```

| Volba      | mswin              | xterm     |
|------------|--------------------|-----------|
| selectmode | mouse, key         | ""        |
| mousemodel | popup              | extend    |
| keymodel   | startstel, stopsel | ""        |
| selection  | exclusive          | inclusive |

## Chování editoru při chybě

- Pípnutí
  - ◆ kolegové vás budou mít moc rádi
- Krátký „záblesk“ celé obrazovky
  - ◆ dosti nepříjemné
  - ◆ pomalé (terminálové sekvence atd.)
- Řešením je vypnutí obou způsobů
 

```
:set novisualbell
```

```
:set vb t_vb=""
```

## Zvýraznění aktuálního řádku/sloupce



- ▶ Může se hodit při editacích YAML souborů apod.
  - :set cursorcolumn
  - :set cursorline
- ▶ Změna barvy aktuálního řádku a/nebo sloupce
  - :hi CursorLine guibg=#2d2d2d
  - :hi CursorColumn guibg=#2d2d2d
- ▶ Poměrně „podivné“ chování při nastavení :wrap

## Digraphs (dvojznaky)

- ▶ Umožňuje vložení vybraného (Unicode) znaku do textu
- ▶ Povolení tohoto režimu
  - :set digraph
- ▶ Použití dvojznaků
  - ◆ Ctrl+K znak1 znak2
    - dostupné vždy
  - ◆ znak1 BACKSPACE znak2
    - jen při povolené volbě :digraphs
    - emuluje a vylepšuje chování psacího stroje
    - klasické BACKSPACE po delším časovém intervalu

## Digraphs (dvojznaky)

- ▶ Zobrazení všech dvojznaků
  - :digraphs
- ▶ Registrace nového dvojznaku
  - :digraphs znak1 znak2 kód\_unicode\_znaku
- ▶ Například
  - :digraphs o` 243
  - :digraphs o/ 248
  - :digraphs O/ 216
  - :digraphs 12 189
- ▶ Výsledky
  - ò
  - ø
  - Ø
  - ½

## Režim „ex“

- ▶ Klávesa Q
  - ◆ někdy stisknuta omylem
  - ◆ vi - návrat zpět

## Režim „ed“

- „ed“
- ▶ Ideový předchůdce editoru Vi (který je předchůdcem Vimu)
- ▶ Příkazy editoru „ed“ se zadávají v command režimu
- ▶ Obecný tvar příkazů
  - :[address [,address]]command[parameters]
- ▶ Adresy
  - . aktuální řádek
  - \$ poslední řádek

2 druhý řádek  
- předchozí řádek  
+ následující řádek  
+2 relativní řádek +2  
-2 relativní řádek -2  
0 nad prvním řádkem (například pro move atd.)  
/foobar/ řádek s textem odpovídajícím regexpu  
?foobar? řádek s textem odpovídajícím regexpu



## Režim „ed“ (pokračování)

- Příkazy „ed“u
  - a append
  - c change
  - d delete
  - m move (musí se zadat další adresa cíle)
    - :.m\$
    - :.m-2
    - :1m\$

## Režim „ed“ (složitější příklady)

- Přesun řádku obsahujícího text „LinuxDays“ na začátek souboru  
:/LinuxDays/m0
- Přesun řádku obsahujícího text „LinuxDays“ na konec souboru  
:/LinuxDays/m\$
- Změna všech znaků „-“ na předchozím řádku za „&mdash“;  
:-1s/-/\&mdash;/g
- Obalení aktuální řádky (může vizuálně přesahovat jeden řádek) do značky pro odstavec  
:s/./<p>\r&\r<\p>/

## Zobrazení informací o pozici v textu a o vybraném znaku

- Zobrazení informace o pozici kurzoru v dokumentu  
gCtrl+G
- Zobrazení kódu znaku pod kurzorem  
ga
- Zobrazení UTF-8 sekvence pro znak pod kurzorem  
g8
- Znaky pro otestování  
 

## Další klávesové zkratky

- Pohyb v historii pozic kurzoru
  - g;
  - g,
- Otevření soubor, na jehož názvu se nachází kurzor  
gf

## Zvýraznění syntaxe

- Dvojice příkazů

- :syntax
- :highlight
- První příkaz
  - :syntax
    - ◆ bez parametrů vypíše syntaktické skupiny
    - ◆ parametry: typ určení textu + jméno + samotné určení
- Druhý příkaz
  - :highlight
    - ◆ bez parametrů vypíše vazbu syntaktické skupiny na „barvičky“
    - ◆ parametry: jméno term= cterm= ctermfg= ctermbg= guifg= guibg=

### Zvýraznění syntaxe

---

```
:syn match Tab "\t"
:syn match Tab2 "\t\t"
:syn match Tab3 "\t\t\t"
:syn match Tab4 "\t\t\t\t"
:hi Tab ctermbg=lightgreen guibg=#e0ffe0
:hi Tab2 ctermbg=lightred guibg=#ffe0e0
:hi Tab3 ctermbg=lightblue guibg=#e0e0ff
:hi Tab4 ctermbg=lightcyan guibg=#ffffe0
```

### Zvýraznění syntaxe

---

```
:syn match vimcommand "\:.*"
:syn match odrazka1 "^\\%d9654.*"
:syn match odrazka2 "^ [^:]*"
:syn match odrazka3 "^ ."
:syn match nadpis "^ [A-Z]*"
:hi odrazka1 ctermfg=lightgreen guifg=darkgreen
:hi odrazka2 ctermfg=yellow guifg=brown
:hi odrazka3 ctermfg=lightcyan guifg=red
:hi nadpis ctermfg=white guifg=blue
:hi vimcommand ctermbg=yellow guibg=#c0ffc0
```

### Pravítko a stavový řádek

---

| Název parametru | Typ parametru | Význam                                                                 |
|-----------------|---------------|------------------------------------------------------------------------|
| ruler           | pravdivostní  | povolení zobrazení pravítka                                            |
| showcmd         | pravdivostní  | povolení zobrazení příkazu během jeho zápisu na pravítku               |
| showmode        | pravdivostní  | povolení zobrazení aktuálně nastaveného režimu editoru na pravítku     |
| rulerformat     | řetězec       | definice formátu zobrazení informací na pravítku                       |
| statusline      | řetězec       | definice formátu zobrazení informací na stavovém řádku                 |
| laststatus      | číslo 0, 1, 2 | řízení způsobu zobrazení stavového řádku pro nejspodnější okno editoru |

Volba :rulerformat

---

```
:set rulerformat=%t%y
:set rulerformat=%t\ %y
:set rulerformat=%c\ %l/%L
:set rulerformat=%o\ [%b=0x%B]
```

Volba :rulerformat

---

| Kód  | Význam                                                      |
|------|-------------------------------------------------------------|
| %f   | jméno souboru s relativní cestou                            |
| %F   | jméno souboru s plnou cestou                                |
| %t   | pouze jméno souboru                                         |
| %y   | typ souboru                                                 |
| %m   | příznak, že soubor byl změněn (modified)                    |
| %r   | příznak, že soubor je určen pro čtení (read-only)           |
| %l   | číslo řádku, na němž se nachází kurzor                      |
| %L   | celkový počet řádků v souboru                               |
| %c   | číslo sloupce, na němž se nachází kurzor                    |
| %v   | číslo virtuálního sloupce                                   |
| %o   | offset od začátku souboru (začíná se od jedničky)           |
| %b   | hodnota znaku pod textovým kurzorem v deka                  |
| %B   | hodnota znaku pod textovým kurzorem v hexa                  |
| %##  | mezi znaky ## je vloženo jméno skupiny definované highlight |
| %{ } | mezi znaky {} je vloženo volání uživatelské funkce          |
| %(   | začátek skupiny značek/textu, kterým lze přiřadit šířku     |
| %)   | konec skupiny značek/textu, kterým lze přiřadit šířku       |

Barvy na pravítku

---

```
:highlight RulerFileName guifg=#cc8080 ctermfg=red
:highlight RulerColumn guifg=#cccc80 ctermfg=yellow
:highlight RulerLine guifg=#80cc80 ctermfg=green
:highlight RulerOffset guifg=#8080cc ctermfg=blue
:set rulerformat=%40(##RulerFileName#%t\ ##RulerColumn#%c\
##RulerLine#%l/%L\ \ ##RulerOffset#%o\ ##Normal#[%b=0x%B]%)
```

Stavový řádek

---

```
:setlocal statusline=%55(%{strftime('%a\ %b\ %e\ %I:%M\ %p')})\ %5l,
%-6(%c%V%)\ %P%)
```

Užitečné externí nástroje

---

- ▶ xxd
- ▶ fmt
- ▶ tablify
- ▶ sort
- ▶ grep
- ▶ cut
- ▶ m4

