Differences between Nvim and Vim8 full version

Shougo
Profile

- It is really needed?
- Shougo
- The dark powered Vim 暗黒美無王
- Uncock Vim Awe, The dark Vim maestro
- Vimrc 500 lines + toml configuration
- 90 plugins
Background

• Nvim(neovim) is used for 30%~40% of Vimmers in 2017 VimConf.
• But why are you use Vim8 or neovim?
• "Vim8 has almost same APIs. So, Nvim is deprecated."

Really???
Do you know Nvim advantages and disadvantages?

- I can explain them to you!
- Because I am the contributor of Vim and Nvim.
Note

- **O**: Nvim is better
- **X**: Vim is better
- **?**: It depends on your usage
?: Nvim is not BDFL project

- Nvim has many contributors
- Vim is commited by Bram
?: Nvim use funding for full time development

- https://salt.bountysource.com/teams/neovim
- Some people partly works for Nvim in weeks.
- Vim recommends to donate to Uganda. But it is not used for full time development
- :help donate
O: Nvim supports XDG defaults

- Nvim uses `$XDG_CONFIG_HOME/nvim/init.vim` instead of `~/.vim/vimrc`
- Nvim uses `$XDG_CONFIG_HOME/nvim` instead of `~/.vim`
O: Nvim defaults

- Nvim always `nocompatible` mode
- Nvim does not use `defaults.vim`
- Syntax, indent and filetype plugins are enabled in default
- Many options are enabled in default.
- `:help nvim-defaults` in Nvim
?: Nvim removes some features

- Crypto(\texttt{help encryption} in Vim8)
- \texttt{:smile} command
- \texttt{:help nvim-features-removed} in Nvim
O: Nvim uses shada instead of viminfo

- It is MessagePack binary format
- :help shada in Nvim
O: Nvim implements original APIs

• nvim prefixed **nvim_xxx** APIs
• **nvim_parse_expression()** - Parse a Vim Script expression
• **:help api** in Nvim
O: Nvim implements inccommand option

- It is live preview feature when substitute
- [Demo]
O: Nvim implements built-in LSP supports (WIP)

- You don't need to install external LSP plugins. https://github.com/neovim/neovim/pull/6856
- Deoplete-lsp uses it https://github.com/Shougo/deoplete-lsp
- [Demo]
?: Nvim re-implements GUI and Windows supports

- It is not stable like GVim, but many GUIs are already available
- Nvim-qt
  https://github.com/equalsraf/neovim-qt
- Gonvim
  https://github.com/akiyoshi/gonvim
  [Demo]?
?: Nvim re-implements TUI

• “term.c” is removed
• t_xxx variables are removed
• If Nvim works, but Vim8 may not work properly in terminal
• If Vim8 works, but Nvim may not work properly in terminal
X: Nvim re-implements external interfaces

- `if_python`, `if_ruby`
- And it is slow than Vim8...
O: Nvim remote plugins

- You can use many Languages than Vim8 to create Nvim plugins
  https://github.com/neovim/neovim/wiki/Related-projects#api-clients
- Many client are available!
O: Nvim includes Lua

• You can use Lua in Nvim!
• Note: The API is not compatible with if_lua yet.
Nvim's job and terminal feature

- Both are simple version and not compatible with Vim API.
- Because, Nvim implements the features before Vim8.
X: Nvim has not included complete Vim patches

- Nvim already includes many Vim patches.
- But it is not completed.
- You can contribute it!
Who should use Nvim?

- "I like Nvim development style."
- "I like Nvim original features."
- "I want to support Nvim development."
- "I want to create plugins without Vim script."
Who should use Vim8?

- "I like Vim8 development style."
- "I don't like Nvim incompatibility."
- "I want to use Vim HEAD features."
FAQ: Nvim is faster than Vim8?

• It depends on your environment and usage.
• Try it!
More?

• :help vim-differences in Nvim documentation
Extra

• In vimconf 2019, I will talk about the 4th generation plugins development.
• 1st generation plugins: neocomplcache.vim, vimshell.vim
• 2nd generation plugins: neocomplete.vim, unite.vim
• 3rd generation plugins: deoplete.nvim, denite.nvim
• 4th generation plugins: defx.nvim, deoppet.nvim
Thank you!