

Zellij – Come Develop Rust Plugins with us!



<https://zellij.dev>

by Aram Drevekenin (@imsnif)



What is a terminal emulator?

Terminal Emulator

- Graphical Application
- Intercepts ANSI/VT
- Displays text on screen
- eg. Alacritty, gnome-terminal, iTerm2

Shell

- Interface to the OS
- Filesystem Access
- Process creation and management
- Scripting
- eg. bash, fish, zsh

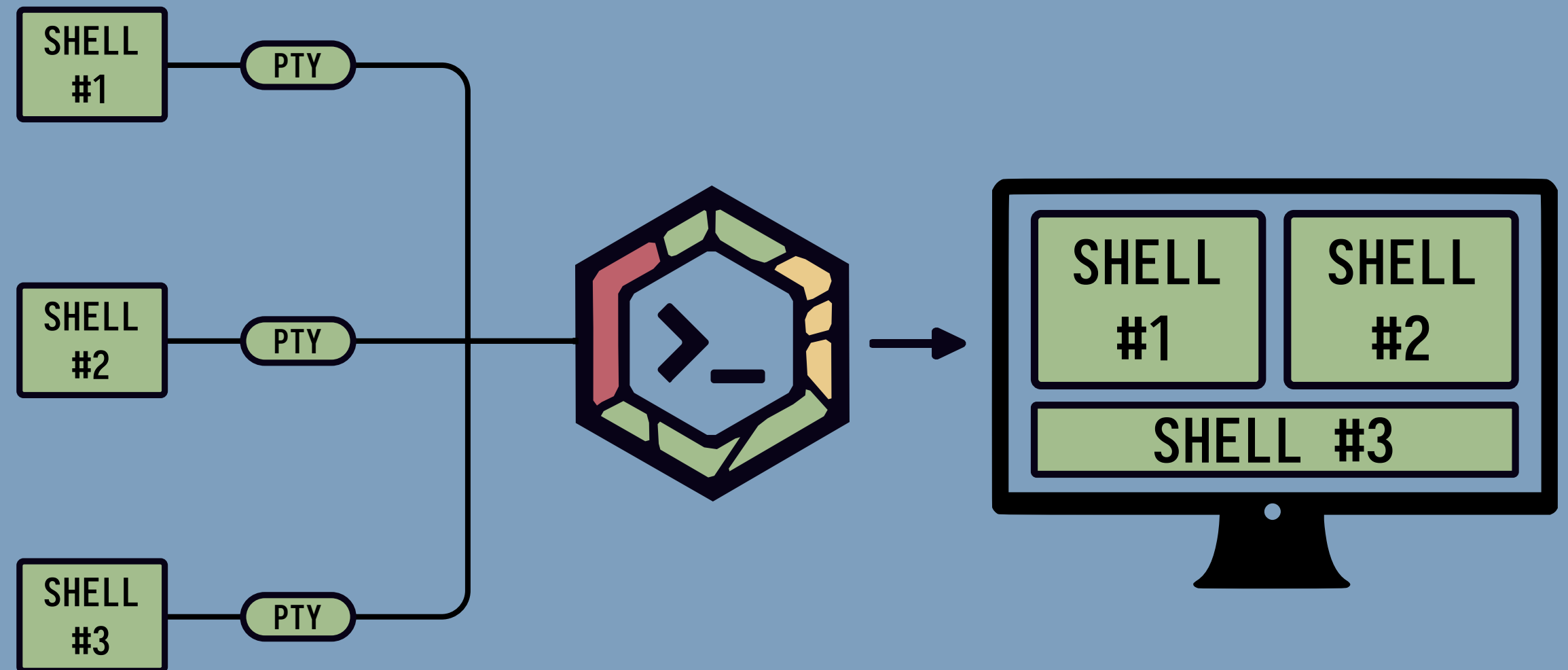
PTY

- Abstracts STDIN/STDOUT
- Tracks primary/secondary pids
- Tracks state such as columns / rows



What's a terminal multiplexer?

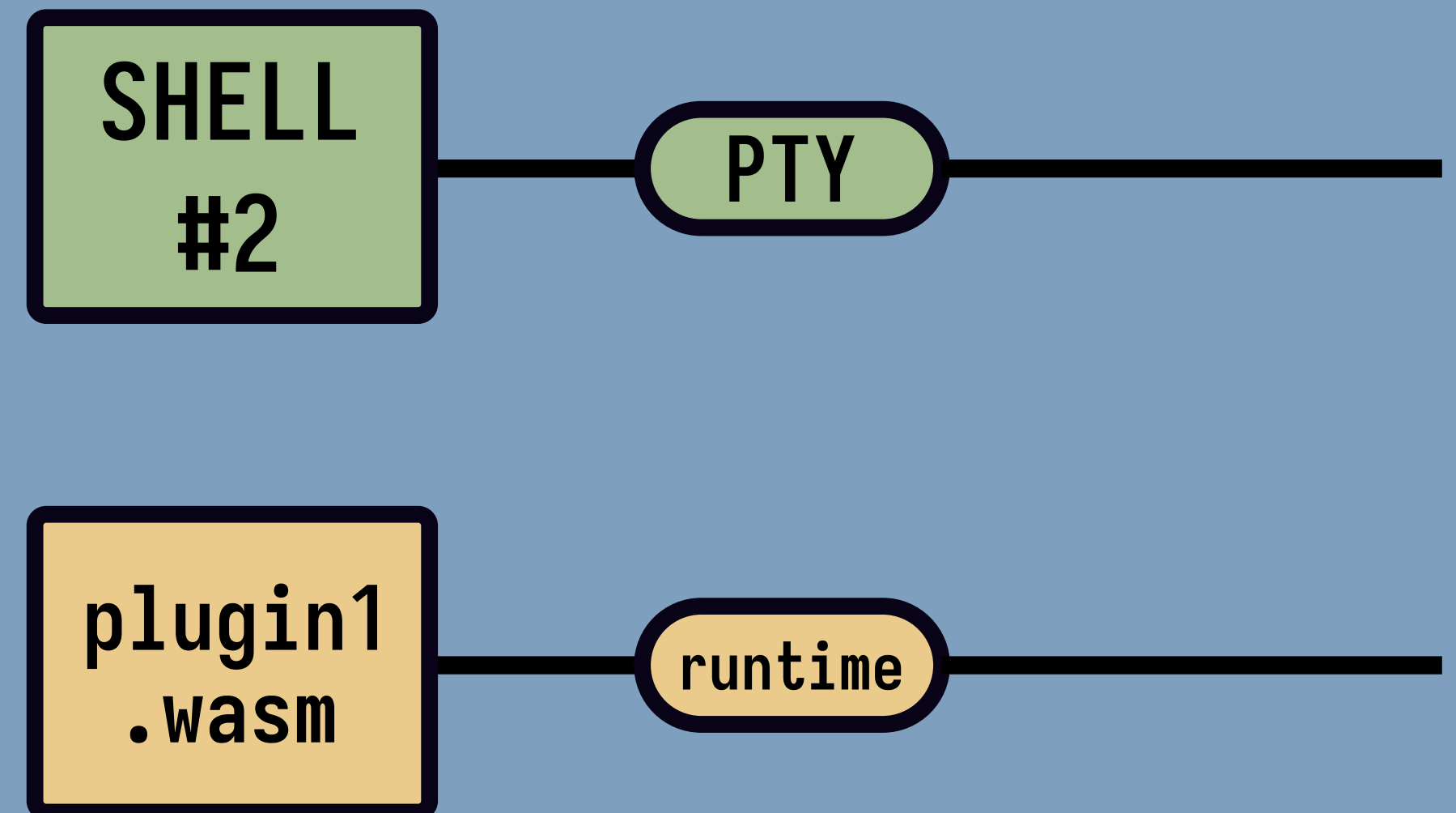
- Manages pty connections
- Keeps session alive
- Arranges panes on screen
- Tracks their viewport



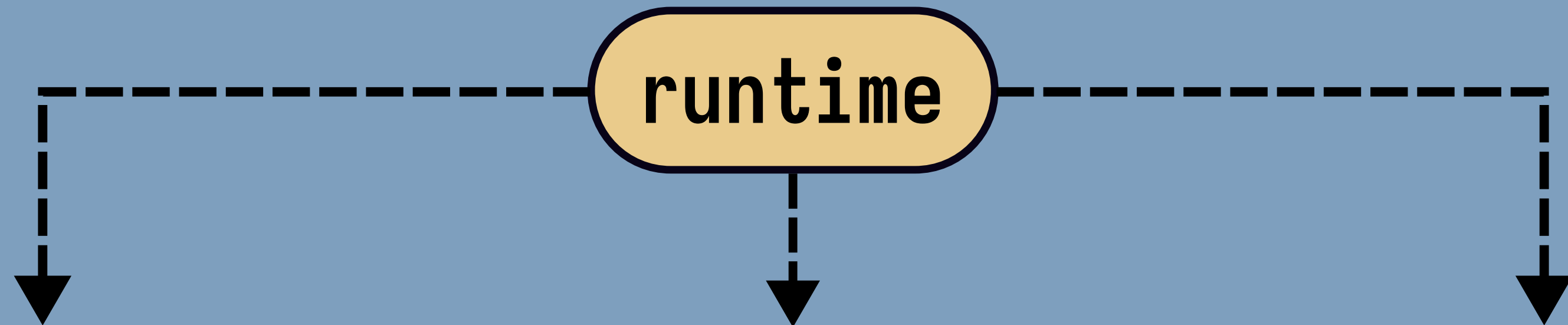


But what if...

- Easier distribution
- Sandboxing
- Easier development
 - * *Composability*
 - * *UI components*
 - * *Output buffer*
- Stronger capabilities
 - * *Affect Zellij*
 - * *Know whole application state*
 - * *Multplayer support out of the box*



How does it work?



Updates and Rendering

Wakes plugin up on subscribed events.

Uses same ANSI/VT interpreter as terminals, but also provides an output buffer.

Exported host functions

```
open_terminal_pane
rename_pane
new_tab_with_layout
...
```

Communication with host

Serializes /
Deserializes complex
data structures over
STDIN/STDOUT

Sandboxes filesystem
access

zellij-tile (Rust SDK)



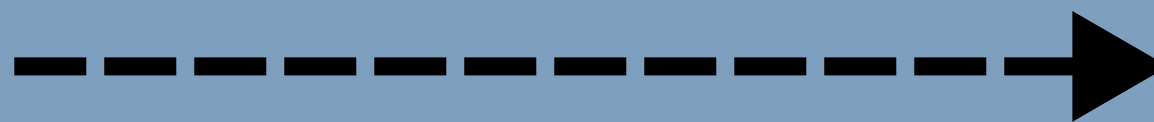
Main Plugin

```
#[derive(Default)]
struct State {}

register_plugin!(State)

impl ZellijPlugin for State {
    fn load()
    fn update(Event)
    fn render(rows, cols)
}
```

post_message_to(search, ..)



post_message_to_plugin(..)



Zero or more Workers

```
#[derive(Default)]
struct SearchWorker {}

impl ZellijWorker ...
    fn on_message(..)

register_worker!(
    SearchWorker,
    search_worker,
    SEARCH_WORKER
)
```

Development Environment

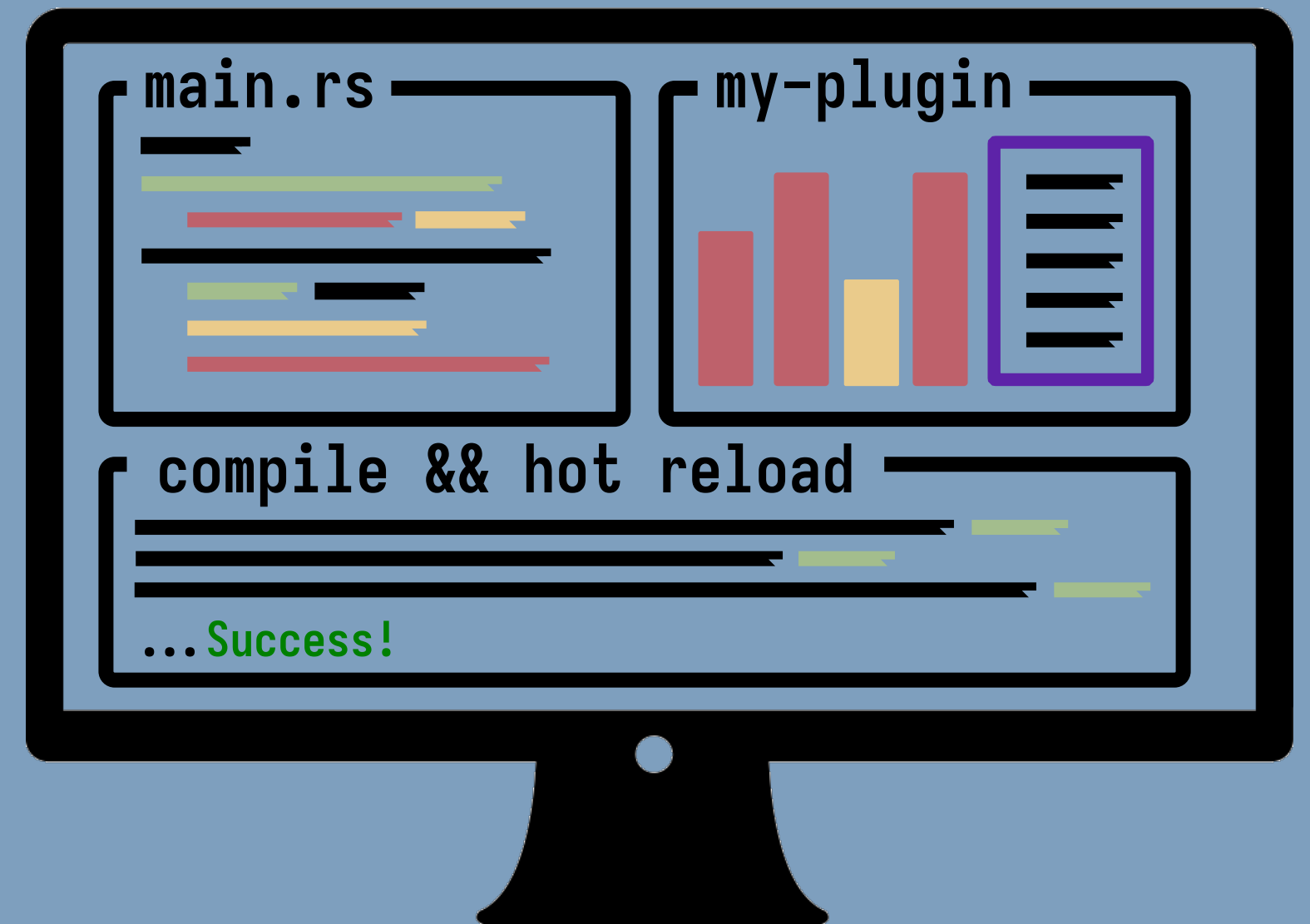


- rust-plugin-example

<https://github.com/zellij-org/rust-plugin-example>

- dev.kdl layout for building and hot reloading

- Plugins can also be developed outside of the terminal!



Plugin Example – Monocle



Monocle is a fuzzy finder for file names and contents.

- Plugin with UI
- Uses **skim** for fuzzy finding
- Uses background workers for file-search and file-contents-search
- Uses the **ignore** crate to traverse fs, skipping hidden files, .gitignore, etc.

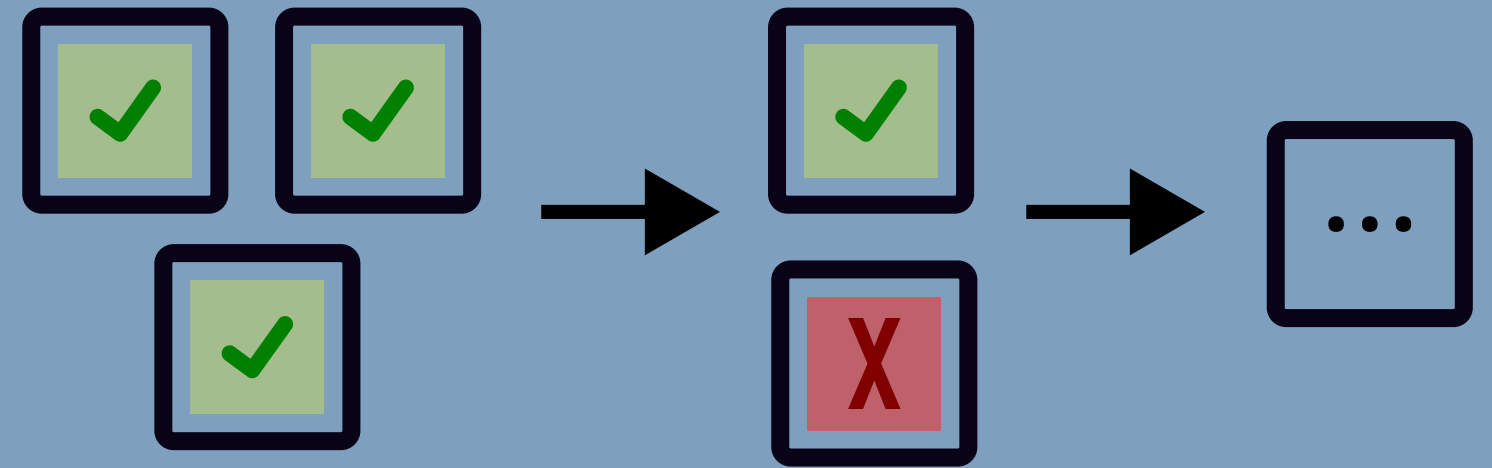


Plugin Example – Multitask



Multitask is a mini-CI for your local environments

- Plugin without UI
- Uses your \$EDITOR as an interface
- Starts a new tab with a custom layout
- Finds out the status of its invoked commands from PaneInfo events





Questions?



Thank you!

Check out Zellij:

<https://github.com/zellij-org/zellij>

<https://zellij.dev>

Learn more about Zellij plugins:

<https://zellij.dev/documentation/plugins>

Zellij screencasts and tutorials:

<https://zellij.dev/screencasts>

Join us on:

 Discord

 Matrix

Follow us on:

 Mastodon

 Twitter