



# The GUI-fication of Neovim

Akin Sowemimo

Hi 🖐️

Software Developer



Monzo

Based in the UK

Github ⇒ [Akin909](#)

Twitter ⇒ [@Akin\\_So](#)

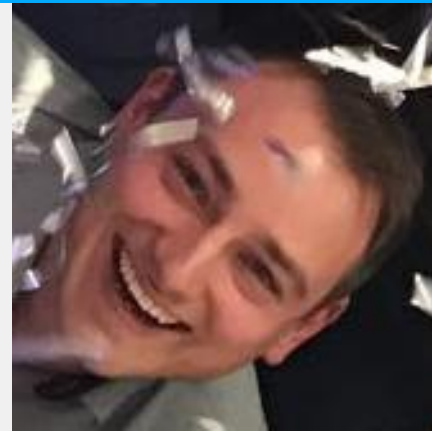
# The Team



Bryan



Ryan



Manuel



Tal



Akin

## Talking Points

- What is Oni?
- Oni's Architecture
- Oni's Extensibility
- Where are we now?
- The Future of Oni

## **What is Oni?**

Oni is a Graphical User Interface(GUI) for Neovim.

It allows users to have all the power of modal editing without the limitations of a terminal UI

But Oni does more than that.

Oni allows Neovim to have features of an IDE such as integration with *language servers* & more *advanced syntax highlighting*

# Syntax Highlighting 🥰

```

26 import * as React from 'react';
25 import { RouteComponentProps } from 'react-router-dom';
24
23 import {
22   GET_LINE_STATUSES,
21   GetLineStatusesQuery,
20 } from '../..../graphql/queries/tfl';
19
18 import TubeLines from '../..../components/organisms/tubeLines';
17 import ErrorHandler from '../..../components/organisms/errorHandler';
16
15 type Props = RouteComponentProps<{}>;
14
13 export default class TubeLinesContainer extends React.PureComponent<Props> {
12   handleClick = (line: string) => () => {
11     this.props.history.push(`/lines/${line}`);
10   };
9
8   render() {
7     return (
6       <GetLineStatusesQuery
5         query={GET_LINE_STATUSES}
4         pollInterval={10_000}
3         displayName="GetLineStatuses"
2       >
1       <ErrorHandler Akin909, 3 months ago, Add main layout to routes as well as a bac
27   <ErrorHandler
1     loaded={Boolean(data && data.lines)}
2     loading={loading}
3     error={error}
4     data={data}
5     render={({ lines }) => (
6       <TubeLines
7         lines={lines}
8         onClick={this.handleClick}
9       />
10    )}
11  />
12  )}
13 </GetLineStatusesQuery>
14 );
15 }

```

Theme ⇒ Night Owl

# Language Server Features



```
initVimNotification.setContent(\n  "init.vim found",\n  `We found an init.vim file would you like Oni to use it?\n  This will result in Oni being reloaded`,\n)\n\ninitVimNotification.SetButtons([\n  {\n    title: "Yes",\n    callback: () => {\n      this._configuration.setValues(\n        { "_internal.hasCheckedInitVim": true, "oni.loadInitVim": true },\n        true,\n      )\n      commandManager.executeCommand("oni.debug.reload")\n    }\n  }\n])
```

*Akin, 7 months ago, Feature: add prompt to use init vim if one is found*



## Oni's Architecture

Oni is an electron app, written with React & Typescript which speaks with Neovim via its *msgpack API*. The editor text is rendered via `webgl` or `canvas`

## Where are we now? 🤔

Some of Oni's most recently implemented features include -

- **Syntax highlighting** based on Textmate
- **File explorer** - which prioritises navigation via the keyboard using vim-like mappings
- **Git Integration** - Only git is supported right now but the framework exists to add more version control providers
- **Achievements** - This functionality is aimed at getting new users interested in learning to use vim
- **Browser Extension** - This allows users to open a browser as an oni split as an oni split.

**5min Demo** 🙏 😞

# Extensibility

Oni can be extended via plugins.

```
const activate = Oni => {
  // Create a new status bar item using oni's api
  const gitHubRepoItem = Oni.statusBar.createItem(1, gitHubRepo)

  // We use react and export it as a dependency for plugins to use
  const React = Oni.dependencies.React

  // Create a command a user can call from the command palette
  const openGitHubRepoCommand = () => {
    Oni.commands.executeCommand("browser.openUrl", "https://github.com/onivim/oni")
  }

  const setGitHubRepo = () => {
    const gitHubIcon = Oni.ui.createIcon({
      name: "github",
      size: Oni.ui.iconSize.Default,
    })

    const element = React.createElement("div", { onClick: openGitHubRepoCommand }, gitHubIcon)
    gitHubRepoItem.setContents(element)
  }

  setGitHubRepo()
  gitHubRepoItem.show()
};

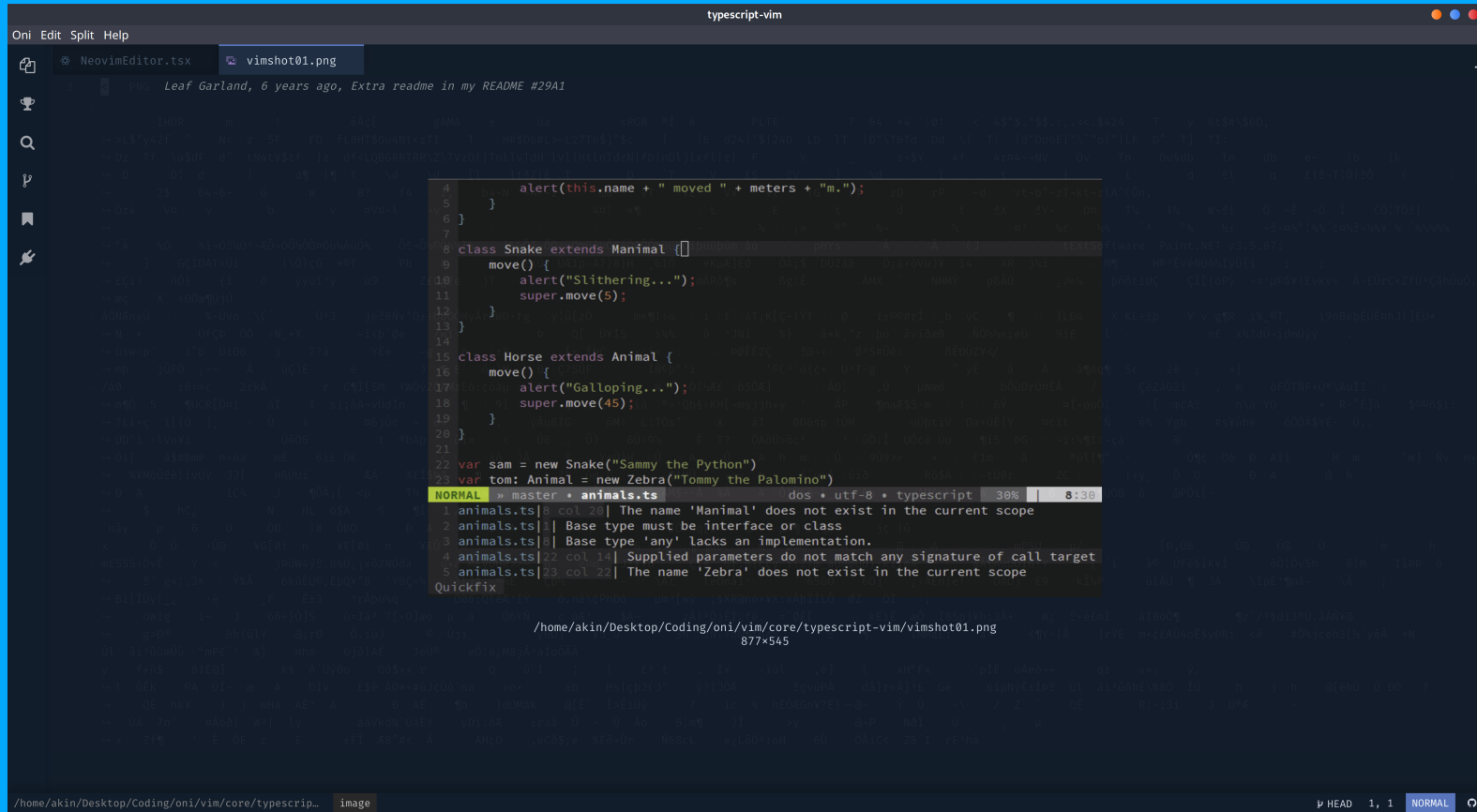
module.exports = {
  activate
};
```

# Some Oni Plugins

# Git Blame Plugin

```
const ligaturesEnabled = this._configuration.getValue("editor.fontLigatures")
this._renderer =
  this._configuration.getValue("editor.renderer") === "webgl" ? new WebGLRenderer(ligaturesEnabled)
  : new CanvasRenderer() Manuel Hornung, 6 months ago, Add experimental WebGL Renderer (#2120) #479F
```

# Image Layer Plugin



```
typescirt-vim
Oni Edit Split Help
NeovimEditor.tsx vimshot01.png
Leaf Garland, 6 years ago, Extra readme in my README #29A1
4     alert(this.name + " moved " + meters + "m.");
5   }
6 }
7
8 class Snake extends Manimal {
9   move() {
10    alert("Slithering...");
11    super.move(5);
12  }
13 }
14
15 class Horse extends Animal {
16   move() {
17    alert("Gallopig...");
18    super.move(45);
19  }
20 }
21
22 var sam = new Snake("Sammy the Python")
23 var tom: Animal = new Zebra("Tommy the Palomino")
NORMAL | master | animals.ts | dos | utf-8 | typescript | 30% | 8:30
1 animals.ts|8 col 20| The name 'Manimal' does not exist in the current scope
2 animals.ts|1| Base type must be interface or class
3 animals.ts|8| Base type 'any' lacks an implementation.
4 animals.ts|22 col 14| Supplied parameters do not match any signature of call target
5 animals.ts|23 col 22| The name 'Zebra' does not exist in the current scope
Quickfix
/home/akin/Desktop/Coding/oni/vim/core/typescirt-vim/vimshot01.png
877x545
/home/akin/Desktop/Coding/oni/vim/core/typescirt-vim/ image | HEAD 1, 1 NORMAL
```

# CSV Layer Plugin

1799\_test.csv

1 1,3.56,7.6, Bryan Phelps, 6 months ago, Fix #1799 - Large macro application (#1801) #0C2E

1 1,0.437479,1,

2 2,2.01333,4.6,

3 2,0.556062,1,

4 3,2.88,8.4,

5 3,0.461792,0.95,

CSV PREVIEW - "/Users/akinyulife/Desktop/Coding/oni/test/collateral/1799\_test.csv"

	3.56	7.6
1	0.437479	1
2	2.01333	4.6
2	0.556062	1
3	2.88	8.4
3	0.461792	0.95
4	2.76	8
4	0.446372	1
5	5.27333	13.6
5	0.280798	0.645
6	4.71333	9.8
6	0.296257	0.725
7	5.52667	14.2
7	0.274231	0.545
8	5.36667	10.4
8	0.288089	0.566667
9	6.68	12.8
9	0.235846	0.62
10	5.14667	11.6
10	0.287271	0.631667
11	5.67333	12.6
11	0.281639	0.649167
12	6.44667	13.8
12	0.235897	0.517619
13	8.34667	15.6
13	0.184609	0.485278
14	7.28	14.2
14	0.23081	0.419444
15	7.41333	13.6
15	0.218091	0.369683
16	6.7	18

34 18,8.29333,16.2,

35 18,0.193528,0.426667,

36 19,8.02,15.8,

1



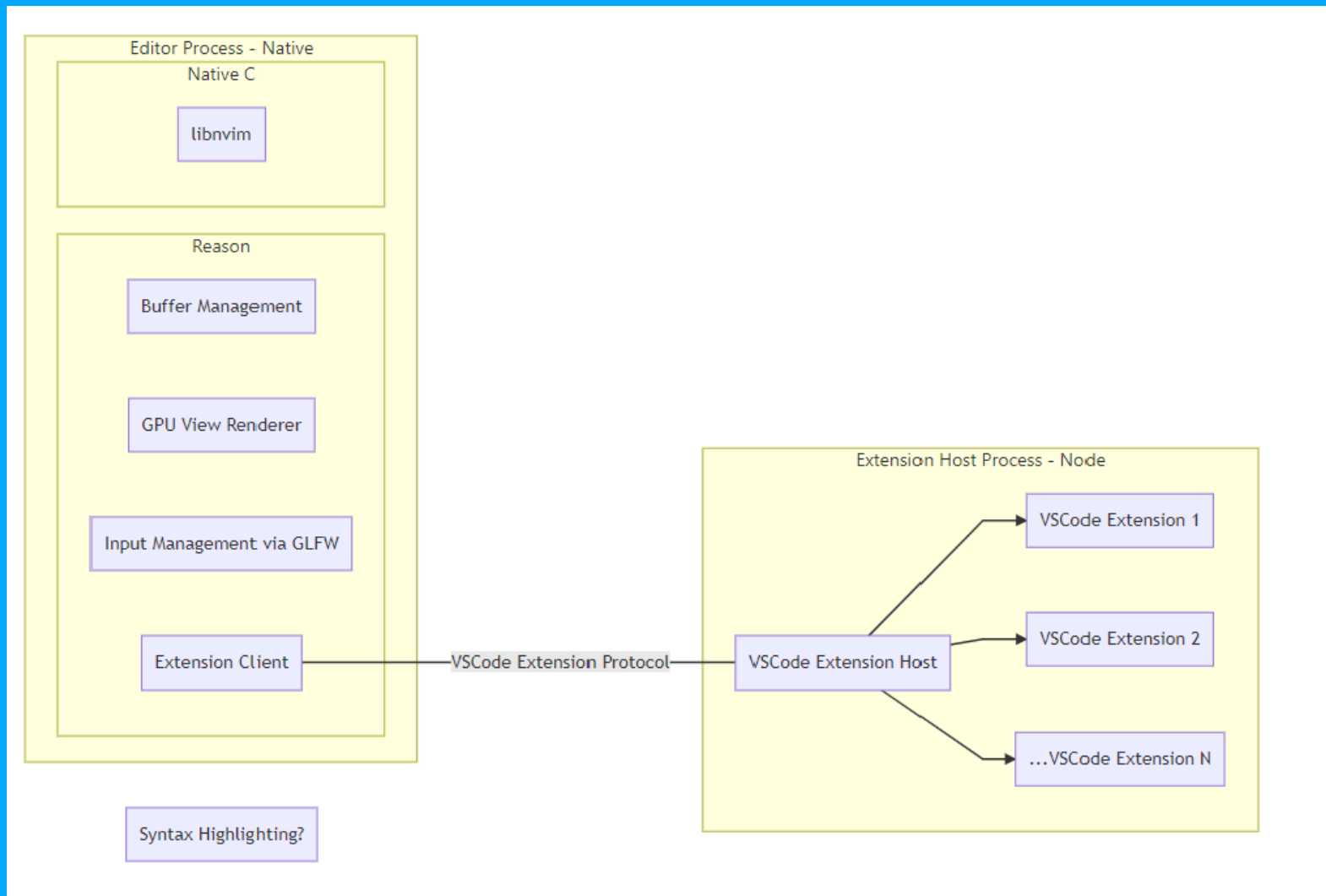
## **Where are we going?**

Performance has always been one of the major advantages of using vim, it has always had a very quick startup time and not been too memory intensive.

We feel that whilst electron and typescript have provided us the advantages of the ecosystem the performance tradeoff might be too great. So ...

We are prototyping Oni V2 (reOni?)  
which will use a highly performant  
**ReasonML** based desktop application  
framework we are developing called  
**Revery.**

# reOni (Oni V2) Proposed architecture



## Useful Links

- Oni website - <https://www.onivim.io/>
- Oni Repository - <https://github.com/onivim/oni>
- Oni Api documentation - <https://onivim.github.io/oni-api/globals.html>
- Discord Link - <https://discord.gg/7maEAxV>
- Revery - <https://github.com/bryphe/revery>
- Revery Quick start - <https://github.com/bryphe/revery-quick-start>

Questions?