

CityJS™  
ATHENS  
2024



**Anthony Fu**  
NuxtLabs

**ESLint One for All Made Easy**

al

# Anthony Fu

Core team member of  Vite  Vue and  Nuxt

Creator of  Vitest  Slidev  UnoCSS  Type Challenges  Elk

Maintainer of  ESLint Stylistic  Shiki  Twoslash

Working at  NuxtLabs



 [antfu.me](https://antfu.me)  [antfu](#)  [antfu@webtool.ls](mailto:antfu@webtool.ls)  [antfu7](#)



# ESLint

## One for All

*Made Easy*

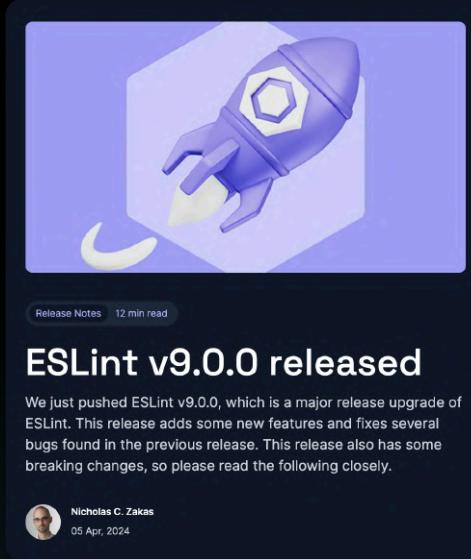


CityJS Athens

June 8th 2024

# ESLint v9.0.0

released 64 days ago



The image shows a screenshot of a dark-themed digital card for the ESLint v9.0.0 release. At the top is a purple hexagonal icon with a white outline. Below it, the text "ESLint v9.0.0" is displayed in a large, bold, light blue font. Underneath that, the text "released 64 days ago" is shown in a smaller, gray font. The main visual is a stylized illustration of a rocket ship launching, with a yellow flame at its base and a white curved line indicating motion against a light blue background. Below the illustration, there's a small button labeled "Release Notes" and "12 min read". The central title "ESLint v9.0.0 released" is in a large, bold, white font. A detailed description follows: "We just pushed ESLint v9.0.0, which is a major release upgrade of ESLint. This release adds some new features and fixes several bugs found in the previous release. This release also has some breaking changes, so please read the following closely." At the bottom left is a small circular profile picture of a man, identified as "Nicholas C. Zakas". To the right of the profile picture, the name "Nicholas C. Zakas" is written in white, and below it, the date "05 Apr, 2024".



Heard about the new ESLint [Flat Config](#)?



Using **Flat Config** already?

# Legacy Config

eslintrc config

- Multiple sources `.eslintrc`, `.eslintrc.js`, `.eslintrc.json`, `package.json`, etc.
- Convention based `extends`
- Package name based `plugins`
- Inheritance tree can be complex

```
// .eslintrc.json
{
  "extends": [
    "eslint:recommended",
    "plugin:@typescript-eslint/recommended"
  ],
  "plugins": [
    "vue",
    "n"
  ],
  "rules": {
    "vue/html-indent": ["error", 2]
  },
  "overrides": [
    // ...
  ]
}
```

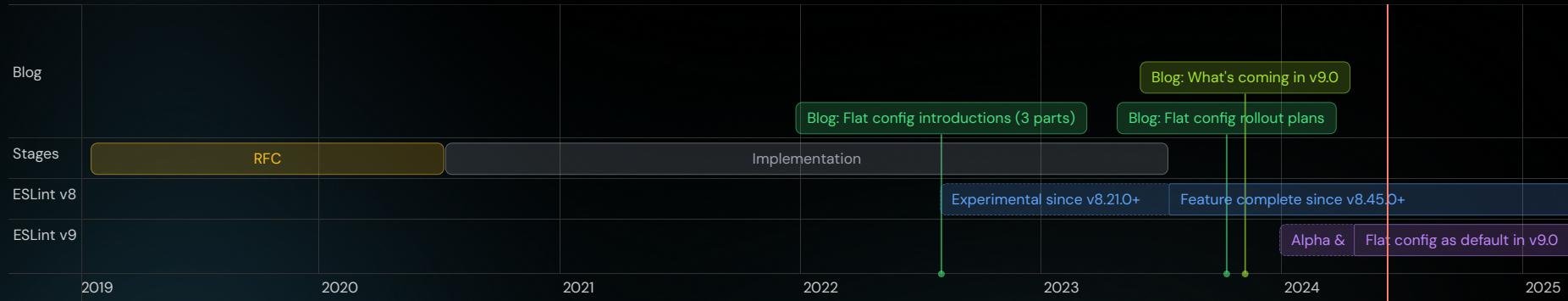
# Flat Config

- Single source `eslint.config.js` & `.cjs` `.mjs`
- Explicit native imports
- Plugins are objects capability to rename plugins
- Composable, easier to trace back

```
// eslint.config.js
import eslint from '@eslint/js'
import typescript from '@eslint-typescript/eslint-plugin'
import vue from 'eslint-plugin-vue'
import n from 'eslint-plugin-n'

export default [ // export an array of configs
  eslint.configs.recommended,
  ...typescript.configs.recommended,
  {
    plugins: {
      vue,
      node: n, // do a rename here
    },
    rules: {
      'vue/html-indent': ['error', 2]
    }
  }
]
```

# Flat Config



Data Source: Flat config rollout timeline by Nicholas C. Zakas

- RFC was created at January 2019
- Experimental in `v8.21.0`
- Stable in `v8.45.0`
- Default in `v9.0.0`
- JavaScript config with full control
- Simplified inheritance and overriding
- Flexible, Dynamic, Customizable

# Compatibility

Migrate from legacy format

## @eslint/eslintrc

Runtime compatibility layer for `extends` converting legacy config to flat config

```
import { FlatCompat } from '@eslint/eslintrc'
import js from '@eslint/js'

const compat = new FlatCompat()

export default [
  // mimic ESLintRC-style extends
  ...compat.extends(
    'standard',
    'example'
  ),
]
```

```
@eslint/migrate-config
```

CLI tool to convert legacy config to flat config

```
npx @eslint/migrate-config .eslintrc.json
```

```
// .eslintrc.json
{
  "env": {
    "node": true,
    "es6": true
  },
  "parserOptions": {
    "ecmaVersion": 2020,
    "sourceType": "module"
  },
  "extends": [
    "eslint:recommended",
    "plugin:ava/recommended",
    "prettier"
  ],
  "plugins": ["prettier", "import"],
  "rules": {
    "prettier/prettier": 2,
    "ava/no-ignored-test-files": 0,
    "ava/only-one-test-file": 0
  }
}
```



```
// eslint.config.mjs
import prettier from 'eslint-plugin-prettier'
import _import from 'eslint-plugin-import'
import { FlatCompat } from '@eslint/eslintrc'

const compat = new FlatCompat()
export default [
  ...compat.extends(
    'eslint:recommended',
    'plugin:ava/recommended',
    'prettier'
  ),
  {
    plugins: {
      prettier,
      import: _import,
    },
    languageOptions: {
      ecmaVersion: 2020,
      sourceType: 'module'
    }
  }
]
```

# Toolings

New tools and possibilities with Flat Config

# Config Inspector

```
eslint --inspect-config
```

- Visualize your config
  - Understand the composition
  - In-place documentations
  - File path tester

```
// eslint.config.js
import antfu from '@antfu/eslint-config'

export default antfu({
  vue: true,
  typescript: false,
  stylistic: {
    indent: 2,
    semi: true,
  },
})
```

 ESLint Config Inspector v0.4.10

/Users/antfu/i/talks/2024-06-08/src/eslint.demo.config.ts  
Composed with 29 config items, updated 6 minutes ago

Configs Rules Files ⚡

Q Test matching with filepath...

Expand All | Collapse All

# Flat Config Utils

Community

antfu/eslint-flat-config-utils

```
import eslint from '@eslint/js'
import typescript from 'typescript-eslint'
import unocss from '@unocss/eslint-plugin'
import vue from 'eslint-plugin-vue'

export default [
  eslint.configs.recommended,
  ...typescript.configs.recommend,
  ...await unocss(),
  {
    files: ['*.vue'],
    ...vue.configs['vue3-recommand'],
    rules: {
      ...vue.configs['vue3-recommand'].rules,
      'vue/html-indent': ['error', 2]
    }
  }
]

// (pesudo code for demo)
```



```
import { compose } from 'eslint-flat-config-utils'
import eslint from '@eslint/js'
import typescript from 'typescript-eslint'
import unocss from '@unocss/eslint-plugin'
import vue from 'eslint-plugin-vue'

export default compose(
  eslint.configs.recommended,
  typescript.configs.recommend, // auto spread
  unocss() // auto await in parallel
)
  .append( // chainable extensions
    vue.configs['vue3-recommand']
  )
  // override any configs with their name or index
  .override('vue', {
    files: ['*.vue'],
    rules: {
      'vue/html-indent': ['error', 2]
    }
  })
)
```

# ESLint Typegen

Community

antfu/eslint-typegen

```
/**  
 * @type {import('eslint').Linter.FlatConfig[]}  
 */  
const configs = [  
  {  
    rules: {}  
    'eslint-comments/disable-enable-pair': 'off',  
    'import'      You, 12 seconds ago • Uncommitted changes  
  },  
  {  
    ':':  
      'import/consistent-type-specifier-style?  
      'import/default?  
      'import/dynamic-import-chunkname?  
      'import/export?  
      'import(exports-last?  
      'import/extensions?  
      'import/first?  
      'import/group-exports?  
      'import/max-dependencies?  
      'import/named?  
      'import/namespace?  
      'import/newline-after-import?
```

(property) RuleOptions['import/consistent-type-specifier-style']?: Linter.RuleEntry<ImportConsistentTypeSpecifierStyle> | undefined

Enforce or ban the use of inline type-only markers for named imports.

@see — <https://github.com/un-ts/eslint-plugin-import-x/blob/v0.4.4/docs/rules/consistent-type-specifier-style.md>

# One for All

One config for all projects

## Legacy Config

```
{  
  "extends": [  
    "@antfu/eslint-config",  
    "@antfu/eslint-config-ts",  
    "@antfu/eslint-config-vue",  
    "@antfu/eslint-config-vue-ts"  
    // ...provide every combination?  
  ],  
  "rules": {  
    // ...a lot overrides  
    "indent": ["error", 4],  
    "@typescript-eslint/indent": ["error", 4],  
    "jsx-indent": ["error", 4],  
    "vue/indent": ["error", 4]  
  }  
}
```

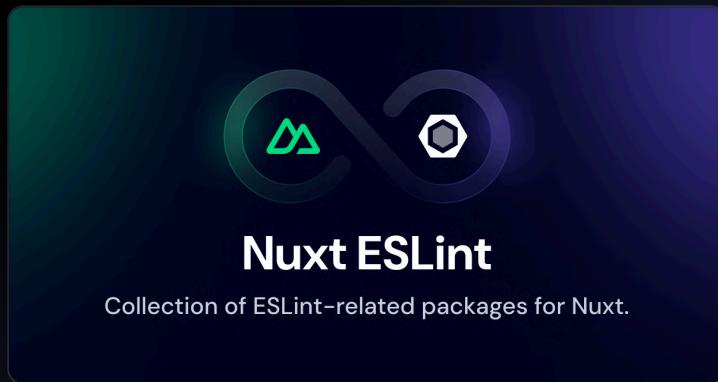
## Flat Config

```
import antfu from '@antfu/eslint-config'  
  
export default antfu({  
  vue: true,  
  typescript: true,  
  stylistic: {  
    indent: 4  
  }  
  // ...  
})
```

- ✓ Shared configs can take user options.
- ✓ One single config, adapts to all projects.
- ✓ Minimal configures as Prettier.
- ✓ Powerful and customizable as ESLint.

# Project-aware Configs

Example:  Nuxt ESLint



[eslint.nuxt.com](https://eslint.nuxt.com)

```
// Generated by Nuxt based on your project
import withNuxt from './.nuxt/eslint.config.mjs'

export default withNuxt(
  // your custom config goes here
)
```

# ESLint is More than a Linter

Mature and powerful AST Toolkit

# ESLint can be a... Formatter



# ESLint Stylistic

Collection of stylistic ESLint rules.  
Formatting and linting in one go.

`eslint.style`

Configs in VS Code

```
[  
  // Auto fix on save  
  "editor.codeActionsOnSave": {  
    "source.fixAll.eslint": "explicit"  
  },  
  
  // Silent the stylistic rules in your IDE,  
  // but still auto fix them  
  "eslint.rules.customizations": [  
    { "rule": "@stylistic/*", "severity": "off" }  
  ]  
}
```

# ESLint can be a... Codemod

antfu/eslint-plugin-command



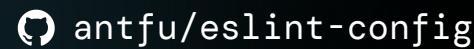
# ESLint can be a... Linter for other Languages

Language	Plugin	Authors
typescript	@typescript-eslint	 TYPESCRIPT-ESLINT  BRADZACHER  JOSHUAKGOLDBERG
vue	eslint-plugin-vue	  OTA-MESHI  VUEJS
svelte	eslint-plugin-svelte	  OTA-MESHI  SVELTEJS
astro	eslint-plugin-astro	  OTA-MESHI
json	eslint-plugin-jsonc	  OTA-MESHI
yml	eslint-plugin-yml	  OTA-MESHI
toml	eslint-plugin-toml	  OTA-MESHI
graphql	graphql-eslint	  DIMAMACHINA
html	html-eslint	  YEONJUAN
mdx	eslint-mdx	  JOUNQIN
Other formats*	eslint-plugin-format	  ANTFU

ESLint RFC #99 – ESLint Language Plugins

# One for All

One config for all projects  
One tool for *everything*\*



★ 3.2k

# @antfu/eslint-config

Code Style ⚙️ Anthony Fu

## Quick Start

```
npx @antfu/eslint-config@latest
```

## Features

- Auto fix for formatting aimed to be used standalone **without** Prettier
- Reasonable defaults, best practices, only one line of config
- Work with TypeScript, JSX, Vue, JSON, YAML, Toml, Markdown, Out-of-box.
- Opinionated, but very customizable
- ESLint Flat config, compose easily!
- Optional React, Svelte, UnoCSS, Astro, Solid support
- Optional formatters support for formatting CSS, HTML, XML, etc.
- **Style principle:** Minimal for reading, stable for diff, consistent
  - Sorted imports, dangling commas
  - Single quotes, no semi
  - Using ESLint Stylistic
- Respects `.gitignore` by default
- Supports ESLint v9.0+ or v8.50.0+

# Thank you!

Slides on [antfu.me](http://antfu.me)