af

Anthony Fu・アンソニ

Core team member of Vite Vue and Muxt

Creator of 🛷 Vitest 🌘 Slidev 🦠 Type Challenges 🚅 UnoCSS U VueUse

Maintainer of

ESLint Stylistic

Shiki

Twoslash

Working at A NuxtLabs

♀ antfu.me ♀ antfu ❤ antfu.me ※ antfujp (日本語)



2ヶ月前にパリから 東京に引っ越した!

よろしくお願いします!



llog Projects Talks Sponsors 후 🗖 😠 🛱 🔊 🕒

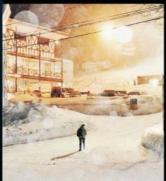
おはよう、東京!

Apr 8 · 7min

→ English

7年前、初めて日本を訪れたときのこと。大阪、京都、北海道を旅して、そのときからずっと「いつか日本に住みたいな」って思ってたんだ。この旅は本当に特別で、世界の見方が変わるような体験だったよ。日本の細部へのこだわりや職人技への情熱、それに伝統と現代の融合の仕方とか、今でも僕の心に強く残ってるんだ。

2018年の旅行で撮った写真を紹介する:







美瑛 Biei, Hokkaido



京都 Kyoto

言語としての日本語にも惹かれたんだ。日本語には独特なリズムや音楽性があって、すごく 魅力的に感じるよ。アニメ好きとして、**たくさんの作品**を見ながら自然と日本語に触れてき たけど、本気で勉強したことはないんだ(日本語がパラパラで通じないかも)。 The New

ESLint Config

with Type Safety

We Heard about the new ESLint Flat Config? フラットコンフィグ 知っていますか?

Using Flat Config already?

フラットコンフィグ を使っていますか?

© ESLint v9.0.0

Released 412 days ago



2024-04-05



2025-05-22

Legacy Config eslintre config

- Multiple formats .eslintrc, .eslintrc.js,
 .eslintrc.json, package.json, etc.
- Convention based extends
- Package name based plugins
- Inheritance tree can be complex
- No types

```
// .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 & .mjs .cjs also supports .ts file!
- Explicit native imports
- Plugins are objects capability to rename plugins
- Composable, easier to trace back
- TypeScript friendly

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

export default [ // export an array of configs
    eslint.configs.recommended,
    ...typescript.configs.recommended,
    {
        plugins: {
            vue,
            node: n, // do a rename here
        }.
```

Flat Config



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

- RFC was created at January 2019
- Experimental in v8.21.0
- Stable since v8.45.0
- Default since v9.0.0
- Ships type declarations since v9.10.0 (@types/eslint is no longer needed)
- Supports TypeScript config since v9.18.0 (eslint.config.ts)

Migration @eslint/migrate-config

CLI tool to convert legacy config to flat config

```
npx @eslint/migrate-config .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/no-import-test-files": 0
```

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

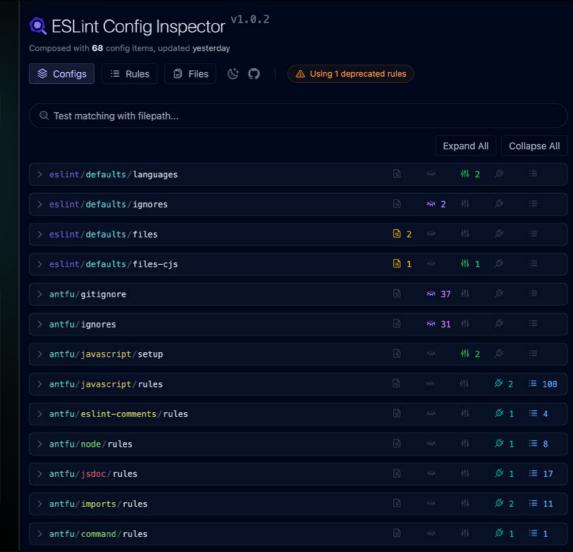
Config Inspector Official

```
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,
  },
})
```



Legacy Config

```
"extends": [
 "@antfu/eslint-config",
 "@antfu/eslint-config-ts",
 "@antfu/eslint-config-vue"
 "@antfu/eslint-config-vue-ts
"rules": {
 "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 accept user options.
- One config adapts to all projects.
- ▼ Type-safe, with inline isdocs documentation.
- Simple configuration like Prettier.
- Powerful and flexible like ESLint.

Flat Config Utils Community

ntfu/eslint-flat-config-utils

```
files: ['*.vue'],
vue configs['vue3-recommand'],
rules: {
 ...vue.configs['vue3-recommand'].rules,
  'vue/html-indent': ['error', 2]
```

 \rightarrow

```
.append( // chainable extensions
 vue.configs['vue3-recommand']
.override('vue', {
 files: ['*.vue'],
 rules: {
   'vue/html-indent': ['error', 2]
```

ESLint Typegen Community

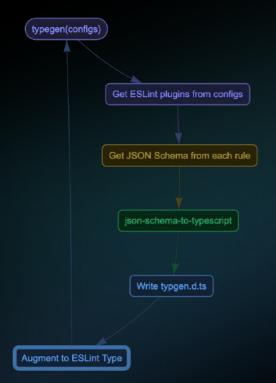
ntfu/eslint-typegen

```
/// <reference path="./eslint-typegen.d.ts" />
import typegen from 'eslint-typegen
export default typegen([
   rules: [
      style/indent': ['error', '2'], Type '"2"' is not assignable to type 'number | "tab" | undefined'.
     'style/quotes': ['error', 'single'],
     'node/no-unsupported-features/node-builtins': ['error', {
        ':' exp⊗ node/no-unpublished-bin?
                                                            (property) 'node/no-unsupported-feat ×

    node/no-unpublished-import?

                                                            ures/node-builtins'?: Linter.RuleEnt
              ry<NodeNoUnsupportedFeaturesNodeBuil
              tins> | undefined
              disallow unsupported Node.js built-in APIs on the
                                                            specified version
                                                            @see - https://github.com/eslint-community/eslint-
                                                            plugin-n/blob/HEAD/docs/rules/no-unsupported-
                                                            features/node-builtins.md
```

How it works



```
declare module 'eslint' {
 namespace Linter {
   interface RulesRecord extends RuleOptions {}
```

ESLint is More than a Linter

Mature and powerful AST Toolkit

ESLint can be a... Formatter



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

eslint.style

Configs in X VS Code

```
{
  // Auto fix on save
  "editor.codeActionsOnSave": {
    "source.fixAll.eslint": "explicit"
  },

  // Silent the stylistic rules in you IDE,
  // but still auto fix them
  "eslint.rules.customizations": [
        { "rule": "@stylistic/*", "severity": "off" }
  ]
}
```

ESLint can be a... Codemod

antfu/eslint-plugin-command

```
▶ 0:00
```

ESLint can be a... Linter for other Languages

Language	Plugin	Maintainers
TS TypeScript	@typescript-eslint	
V Vue	eslint-plugin-vue	OTA-MESHI ▼ VUEJS
Svelte	eslint-plugin-svelte	OTA-MESHI SVELTEJS
🐧 Astro	eslint-plugin-astro	OTA-MESHI
O JSON	eslint-plugin-jsonc	OTA-MESHI
× YAML	eslint-plugin-yml	OTA-MESHI
[T] TOML	eslint-plugin-toml	OTA-MESHI
🗘 GraphQL	graphql-eslint	S DIMAMACHINA
5 HTML	html-eslint	2 YEONJUAN
MUX MDX	eslint-mdx	JOUNQIN
Other formats*	eslint-plugin-format	ANTFU ANTFU

ESLint RFC #99 - ESLint Language Plugins

One for All

One config for all projects
One tool for everything*

antfu/eslint-config

@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+

The Future of Linting

What about native toolings like biome, oxlint, deno lint?



Quick Take: Already usable in some scenarios, but not yet ready for majority.

- Extremely fast performance
- Unified toolchain, reusing AST
- Multi-threaded, parallel file linting

Cons

- ⚠ Limited plugin ecosystem
- ⚠ Limited language support
- ▲ Static configuration only

Anthony's Takes

- Native linters are valuable additions to the ecosystem
- ESLint and native linters will coexist until feature parity is reached
- Integration paths: ESLint in native linters or vice versa
- Type-aware linting via tsgo for native linters
- Working with the oxlint team to explore better integration

ありがとうございます!

Slides available at antfu.me