
Plug-in Mechanism for Flow Linter

28 April 2020

Kunihiko Toumura

Research and Development Group
Hitachi, Ltd.

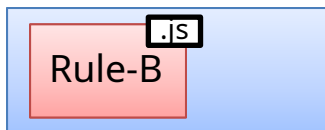
1. Overall architecture

Update since last discussion:

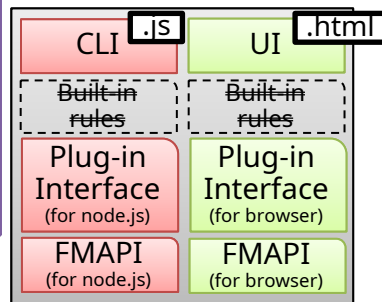
- Eliminate "Built-in Rules" in Main module to simplify main module structure.
- Introduce API calls between editor and runtime.

Note: Plug-in developer could generate a browser-side code automatically (e.g. using WebPack, etc.).

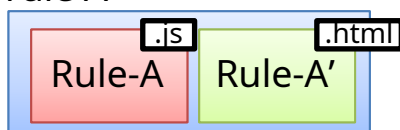
Plug-in for validation rule A



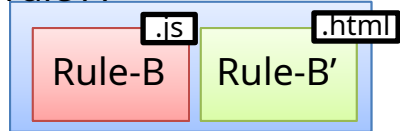
Main module of Flow Linter



Plug-in for validation rule A



Plug-in for validation rule A

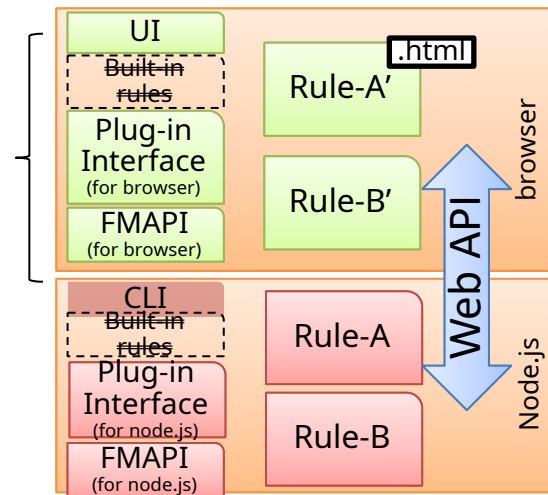


Runtime/
CLI-side

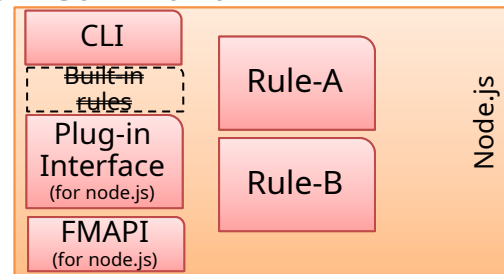
Browser-
side

Linter as a Node-RED editor function

Load to Browser using same mechanism as nodes.



Linter as a CLI Command



2. Browser-side code auto-generation

- Using 'webpack'
 - optionally, use 'Babel' to convert from modern code to browser-compatible code

Excerpt from webpack.config.js:

```
...
entry: './src/rule.js',
output: {
  filename: 'bundle.js',
  path: path.resolve(__dirname, 'dist'),
  library: normalisePluginName('no-func-name'),
  libraryTarget: 'var'
},
plugins: [
  new CleanWebpackPlugin(),
  new HtmlWebpackPlugin({
    filename: 'plugin.html',
    templateContent: ''
  }),
  new ScriptExtHtmlWebpackPlugin({
    inline: 'bundle.js'
  }),
  new CopyWebpackPlugin([
    { from: 'src/plugin.js' }
  ]),
],
...
```

← source file of a rule module

← convert module to function ('nrlint-plugin-no-func-name' module to 'nrlintNoFuncName' object)

← embed javascript code to <script> in HTML file

← just copy a (empty) server-side code

3. Issue in Browser-side code auto-generation

- Almost all rules are simple, so this auto-generation will work.
- Some package uses dynamic module loading or uses filesystems
 - e.g. ESLint
 - 'eslint4b' (ESLint for browser; I use it in the rule) eliminate them by hand. But we cannot assume that all package have such treatment.
- For the rule which uses such code:
 - the rule module provide Web API to call server-side function.
 - not a public API. Same as '/debug/...' in the Debug node.
 - the rule should minimize API calls to the server-side.
 - for example, to check a code in function node on server-side, the rule calls Web API only when the function nodes are modified, rather than when the any nodes in the flow are modified.