

如何为 VSCode 开发前端调试插件

-Edge Developer Tools for Visual Studio Code

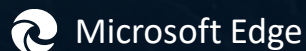
宋青见

EDGE · 开发者生态 · 首席产品经理

陈又箐

EDGE · 开发者工具 · 软件工程师

APRIL 2022



目录

- 撸代码之前的开场热身
 1. Edge 全面采用了 Chromium 内核 - CDP
 2. VSCode 内置了JS Debug, Edge Devtools增强了UI部分
 3. 一分钟全功能演示
- 架构分享、技术要点和代码示例
 1. Webviews和iFrame
 2. CDP 和 JS Debug的集成
 3. 利用VSCode 编辑器功能远程修改CSS
 4. Webhint 语法检查
 5. VSCode Commands 命令调用
- 联系我们

开场热身

Introducing the new Microsoft Edge

World-class performance with more privacy, more productivity
and more value while you browse

[Download now >](#)

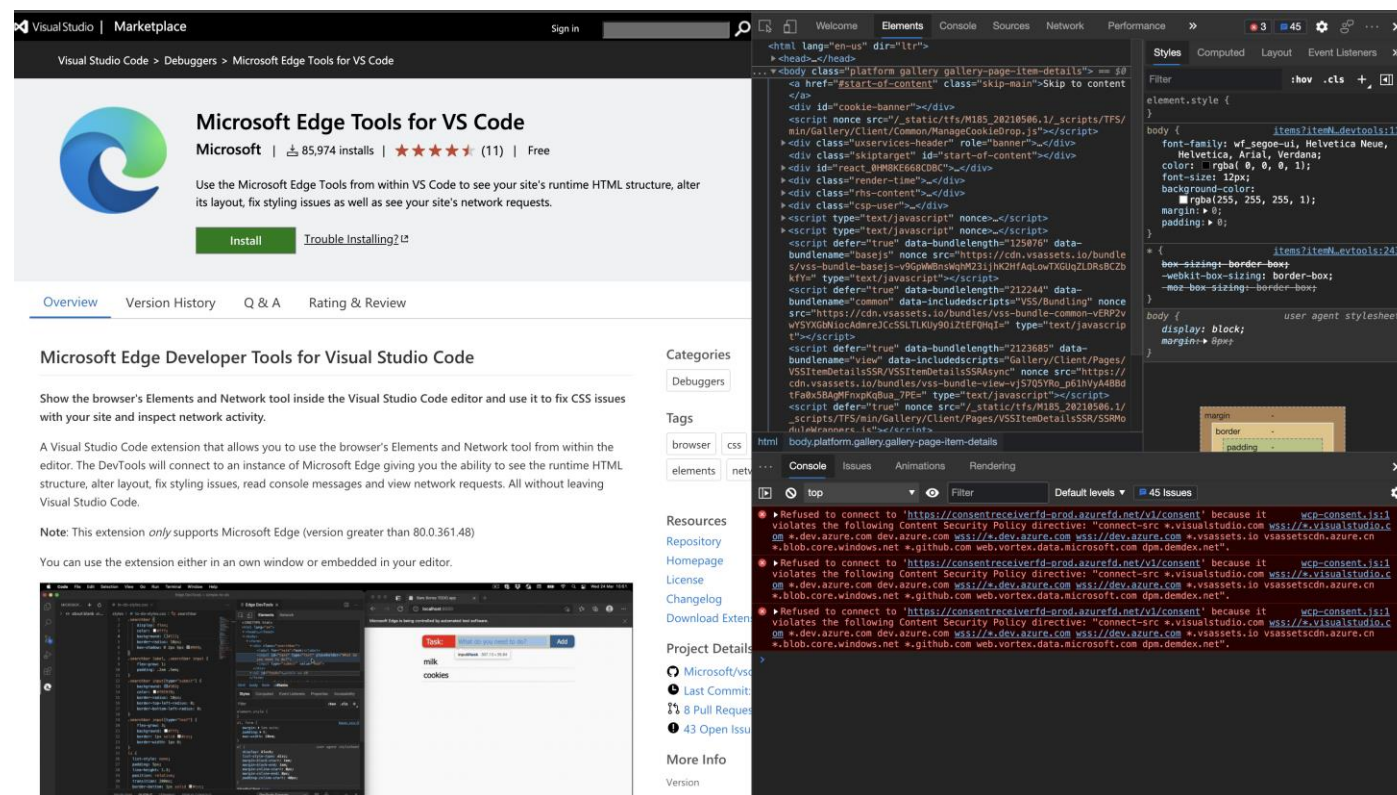


Edge 浏览器已经全面采用 Chromium 内核
Chrome DevTools Protocol - CDP



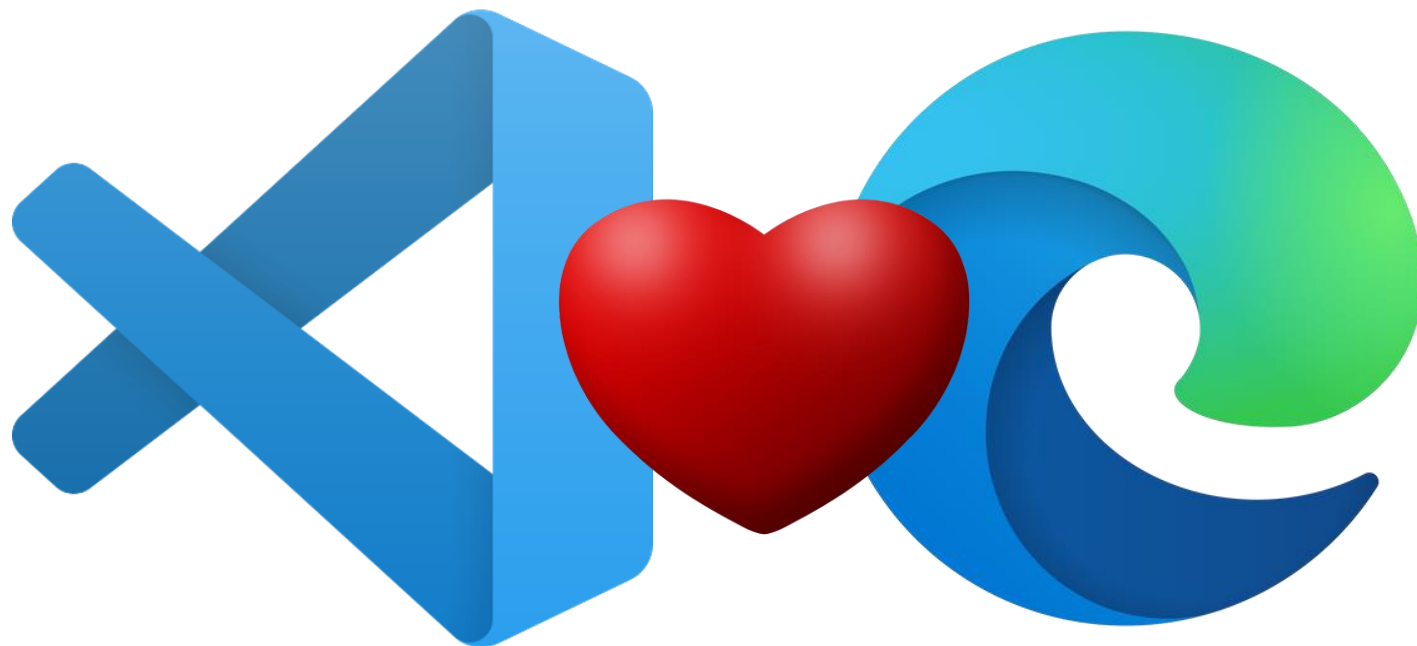
Microsoft Edge DevTools

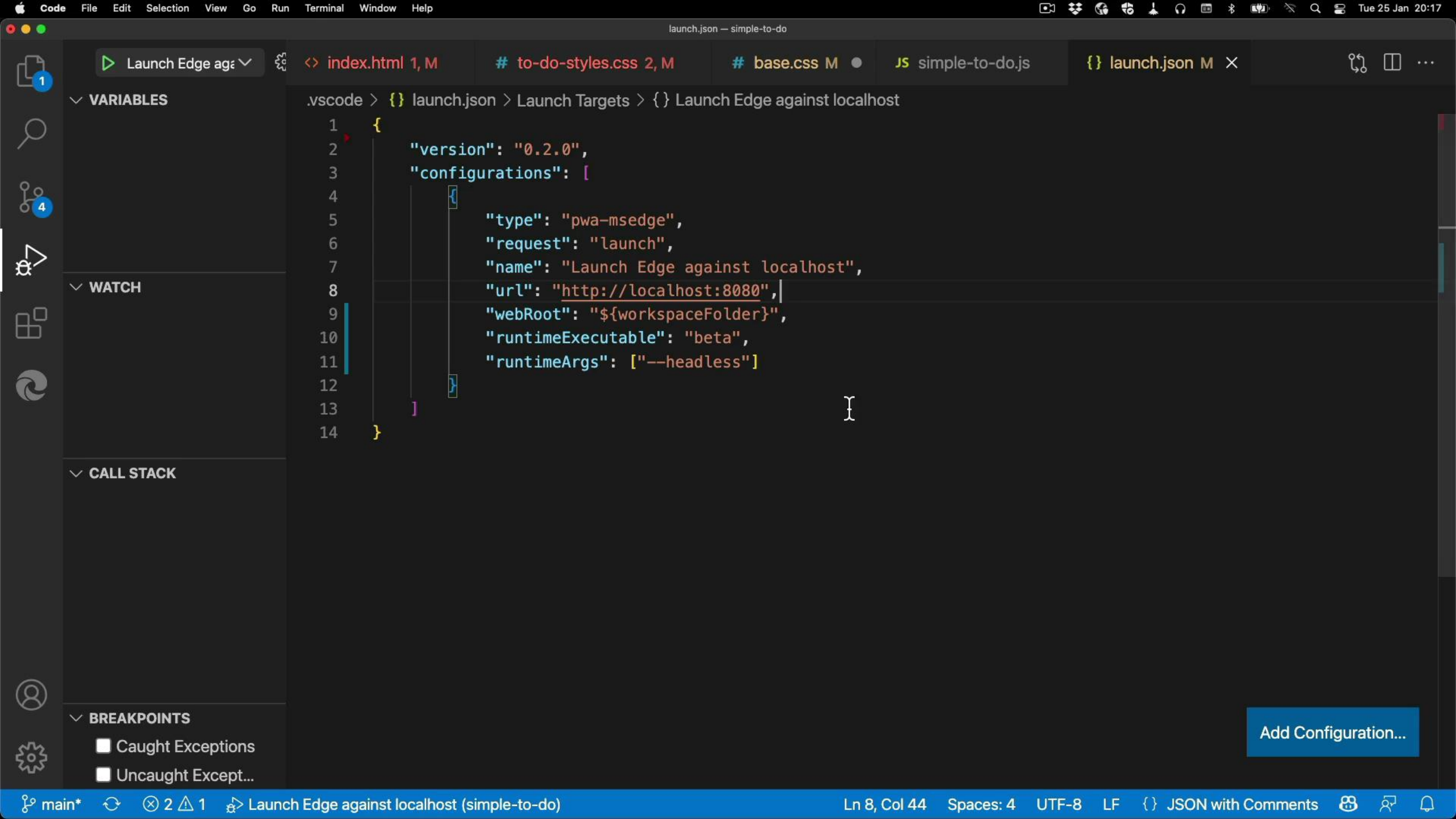
- 基于 Chromium
- 浏览器内 inspection of HTML, CSS and Rendering
- 图形化工具-比如 color pickers, animation sliders, grid/flexbox tools
- 网络监控-Network tool
- 代码编辑器，浏览器内编辑和调试！
- 更多有趣的功能.....



Edge DevTools 插件方式进入VSCode的好处

- 更符合程序员习惯的代码编辑和调试工具- 在VSCode
- 同时又兼具浏览器内丰富的可视化工具功能-用Edge DevTools 插件





Launch Edge ag

index.html 1, M

to-do-styles.css 2, M

base.css M

simple-to-do.js

launch.json M

VARIABLES

WATCH

CALL STACK

BREAKPOINTS

Caught Exceptions

Uncaught Except...

.vscode > {} launch.json > Launch Targets > {} Launch Edge against localhost

```
1  {
2    "version": "0.2.0",
3    "configurations": [
4      {
5        "type": "pwa-msedge",
6        "request": "launch",
7        "name": "Launch Edge against localhost",
8        "url": "http://localhost:8080",
9        "webRoot": "${workspaceFolder}",
10       "runtimeExecutable": "beta",
11       "runtimeArgs": ["--headless"]
12     }
13   ]
14 }
```

Add Configuration...


main* 2 1 Launch Edge against localhost (simple-to-do)

Ln 8, Col 44 Spaces: 4 UTF-8 LF {} JSON with Comments

Edge DevTools for Visual Studio Code



aka.ms/devtools-for-code



Microsoft Edge Tools for VS Code

Microsoft | 86,049 installs | ★★★★★ (11) | Free

Use the Microsoft Edge Tools from within VS Code to see your site's runtime HTML structure, alter its layout, fix styling issues as well as see your site's network requests.

[Install](#) [Trouble Installing?](#)

[Overview](#)

[Version History](#)

[Q & A](#)

[Rating & Review](#)

Microsoft Edge Developer Tools for Visual Studio Code

Show the browser's Elements and Network tool inside the Visual Studio Code editor and use it to fix CSS issues with your site and inspect network activity.

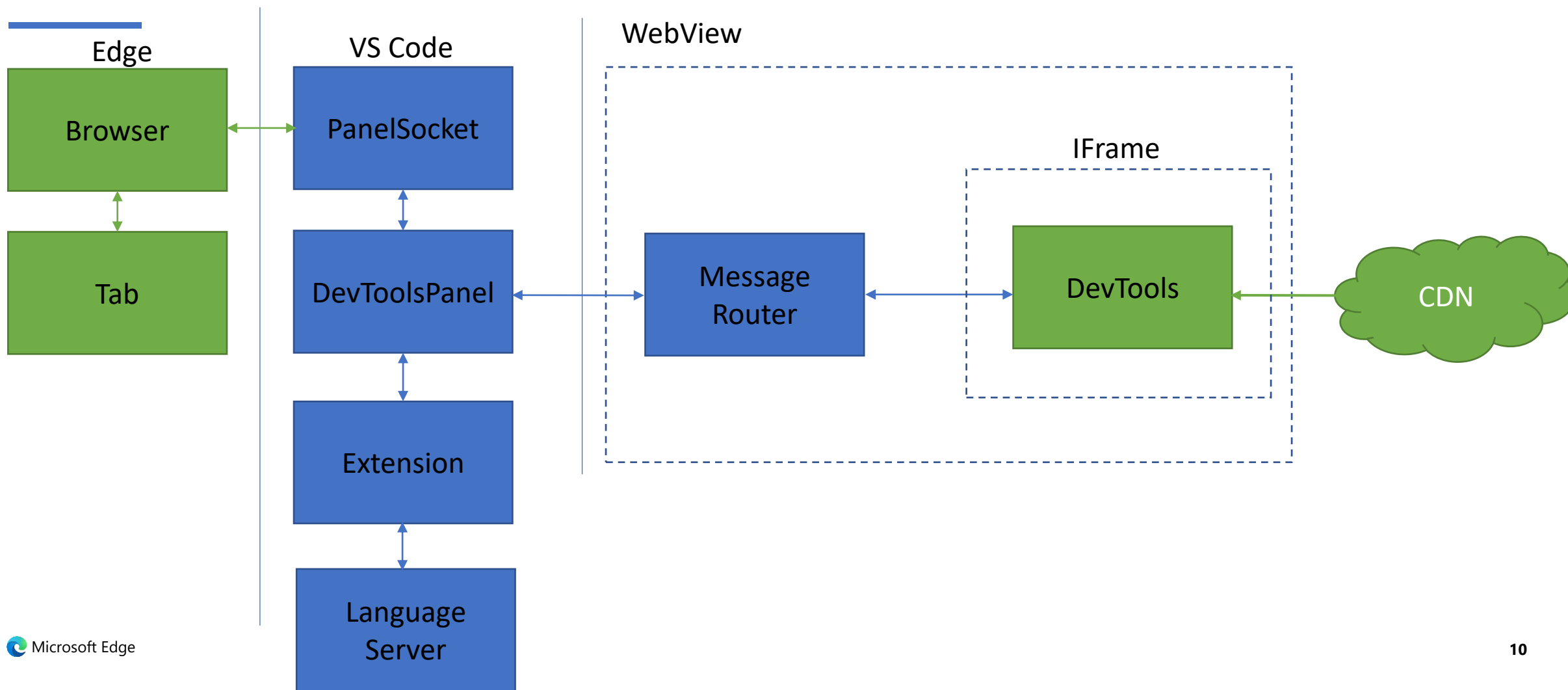
A Visual Studio Code extension that allows you to use the browser's Elements and Network tool from within the editor. The DevTools will connect to an instance of Microsoft Edge giving you the ability to see the runtime HTML structure, alter layout, fix styling issues, read console messages and view network requests. All without leaving Visual Studio Code.

Note: This extension *only* supports Microsoft Edge (version greater than 80.0.361.48)

You can use the extension either in an own window or embedded in your editor.

技术架构和代码要点

技术架构



Webviews 和 iFrame

WebView 代码示例

WebView Creation

```
const panel = vscode.window.createWebviewPanel(SETTINGS_STORE_NAME, SETTINGS_WEBVIEW_NAME, column, {  
    enableCommandUris: true,  
    enableScripts: true,  
    retainContextWhenHidden: true,  
});
```

Setting WebView Content

```
this.panel.webview.html = this.getCdnHtmlForWebview();
```

WebView 和 IFrame 架构



WebView 加入iFrame 代码示例

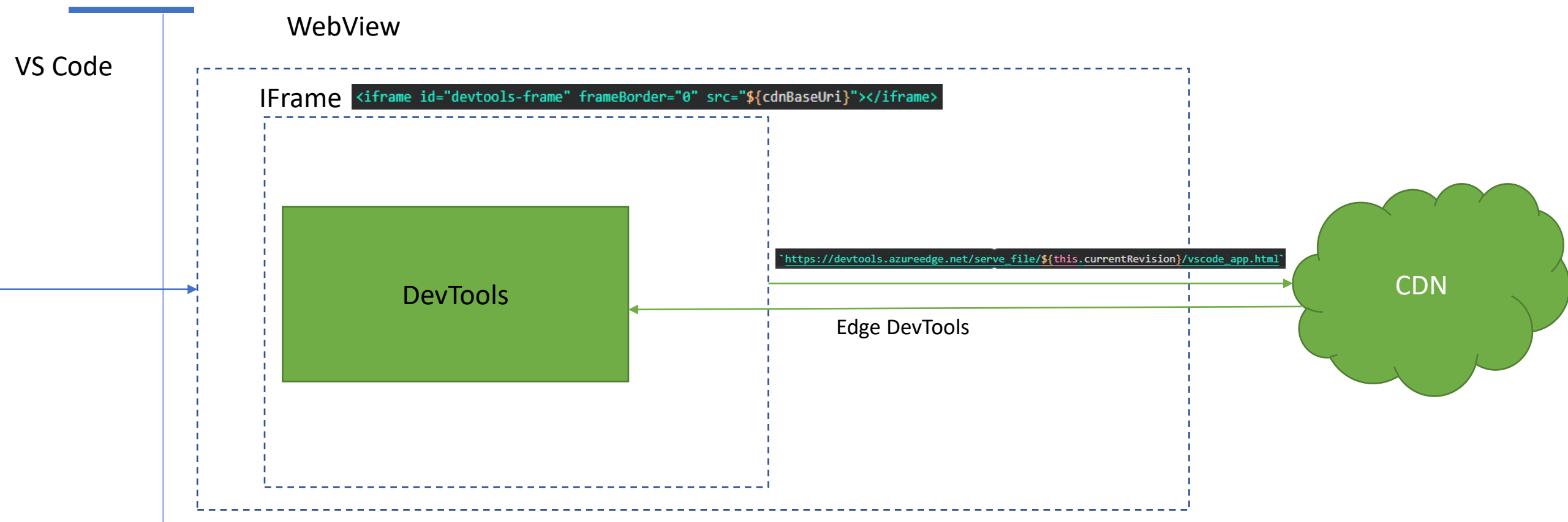
iFrame HTML Element

```
<iframe id="devtools-frame" frameborder="0" src="${cdnBaseUrl}"></iframe>
```

cdnBaseUrl Definition

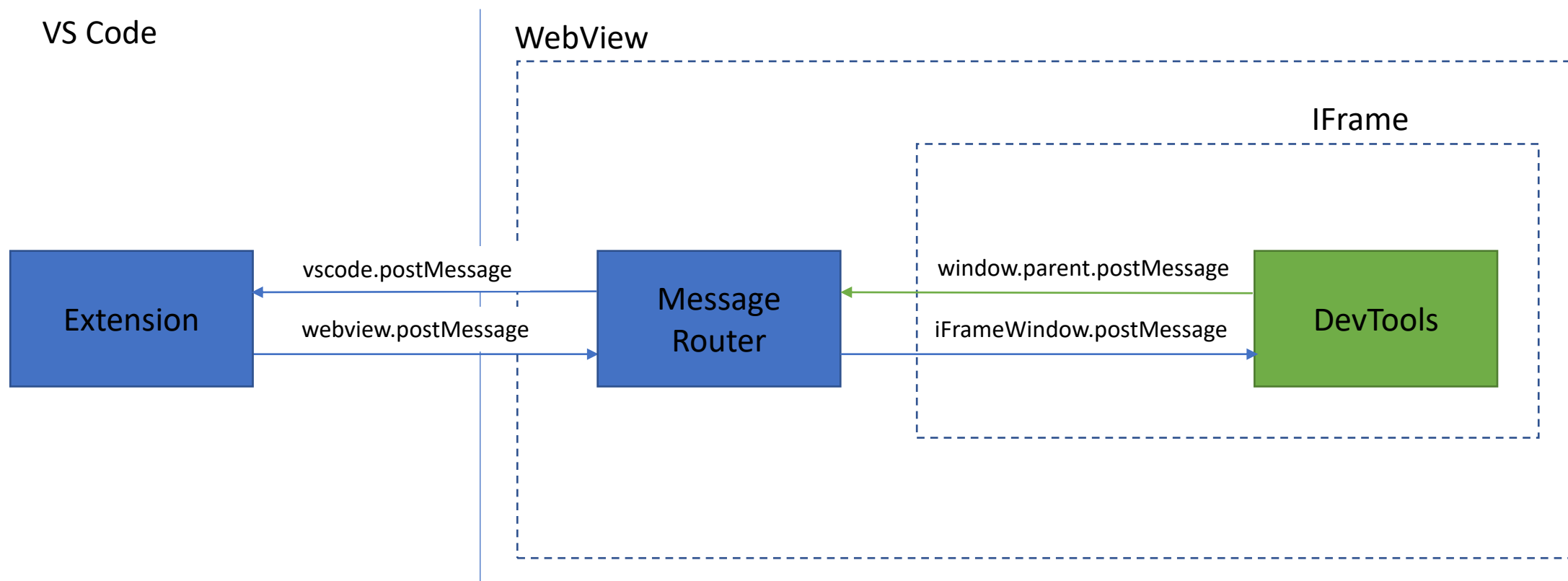
```
cdnBaseUrl = `https://devtools.azureedge.net/serve_file/${this.currentRevision}/vscode_app.html`;
```

WebView 和 IFrame 架构

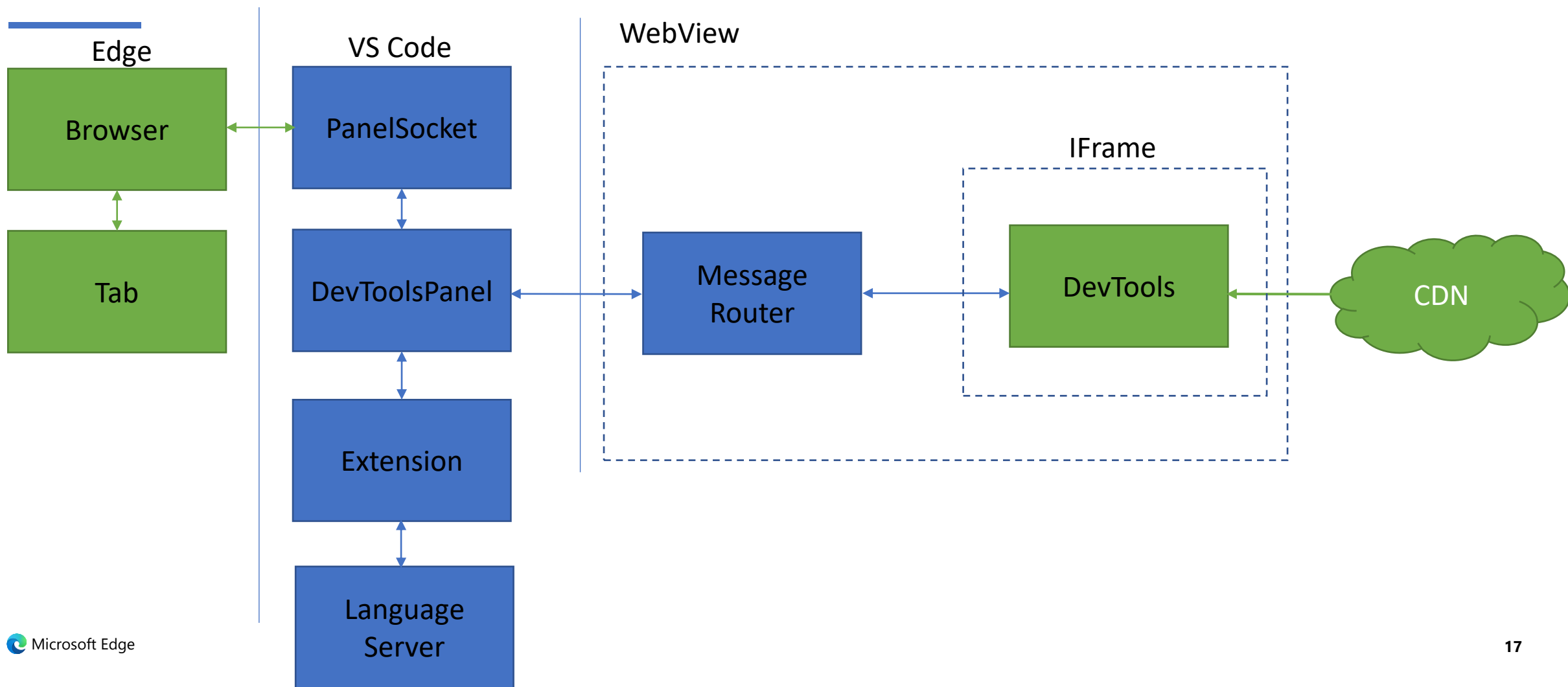


通信机制

VS Code



技术架构



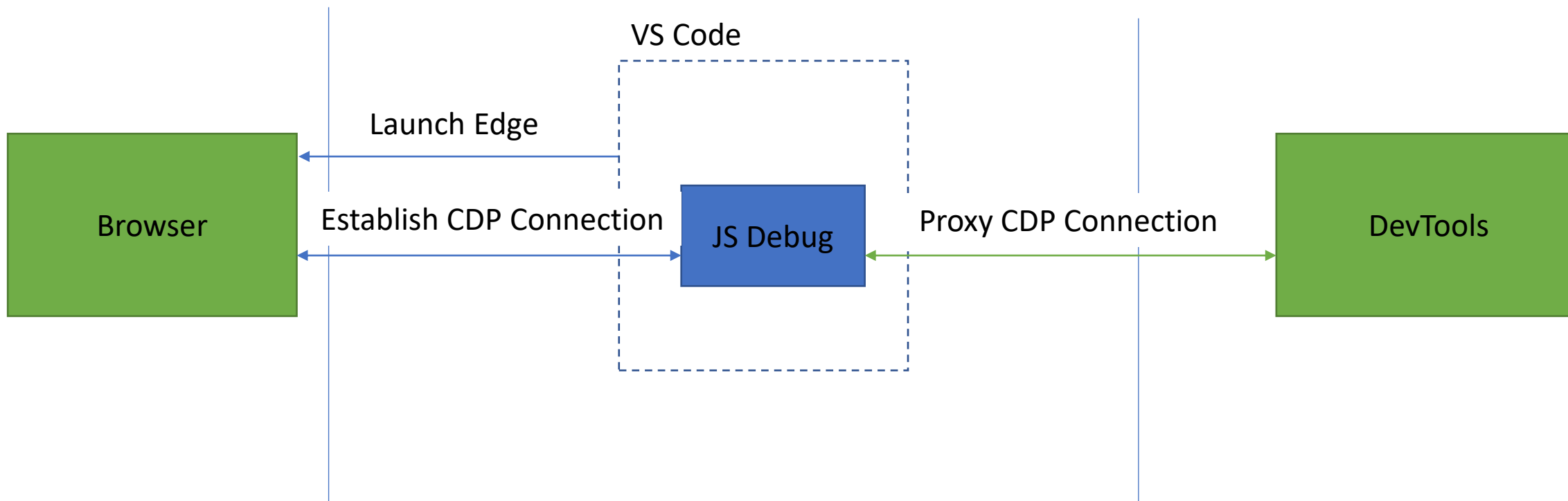
CDP 和 JS Debug的集成

注册 JS Debug CDP Proxy

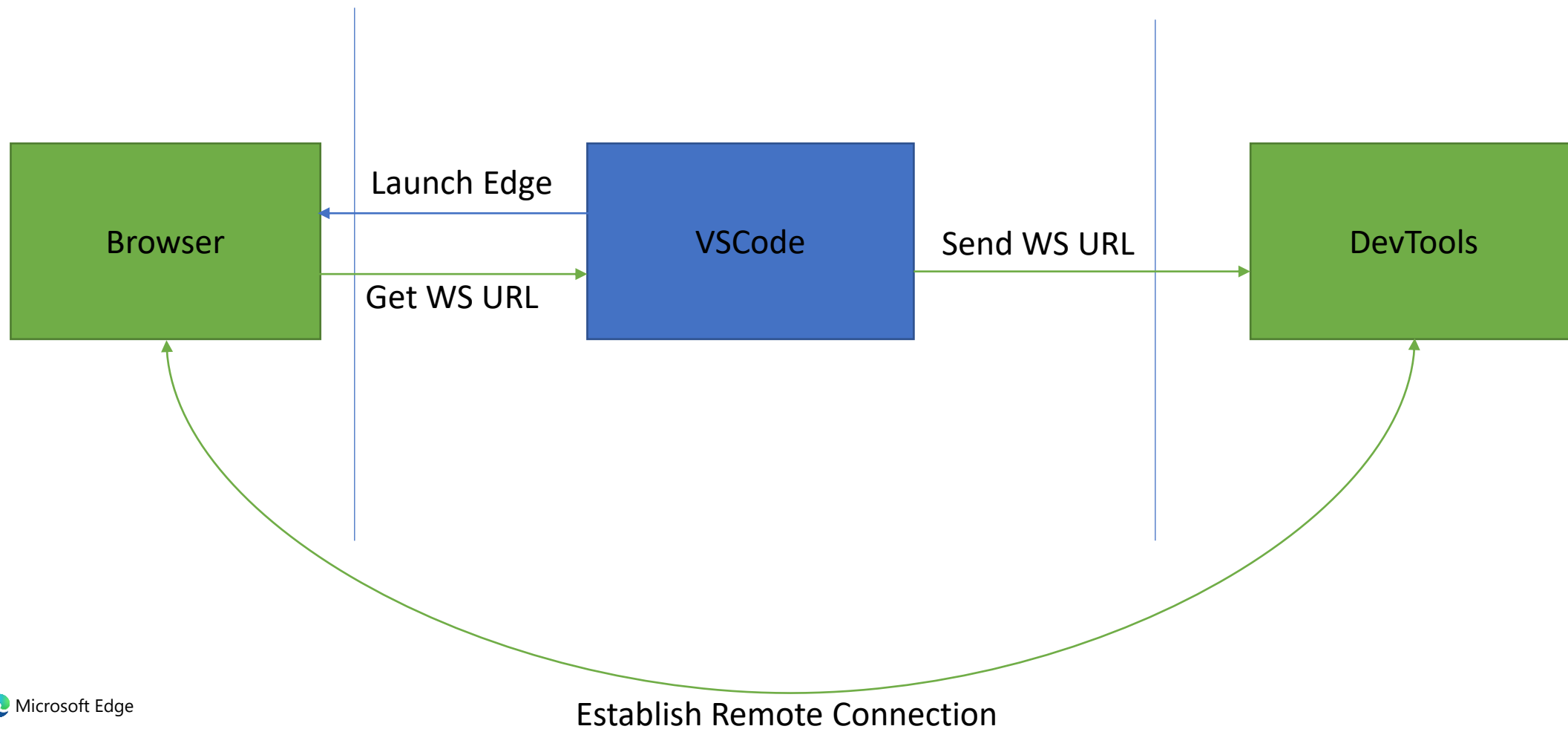
```
private registerForJsDebugSharedCDPEvents() {  
    // Register for CDP events from jsdebug:  
    // https://github.com/microsoft/vscode-js-debug/blob/main/CDP_SHARE.md  
    const registrationMessage = {  
        method: 'JsDebug.subscribe',  
        params: {  
            events: [  
                'Runtime.*',  
                'DOM.*',  
                'CSS.*',  
                'DOMDebugger.*',  
                'Network.*',  
                'Page.*',  
                'Target.*',  
                'Overlay.*',  
            ],  
        },  
    };  
    if (this.socket) {  
        this.socket.send(JSON.stringify(registrationMessage));  
    }  
}
```

[vscode-js-debug/CDP_SHARE.md at main · microsoft/vscode-js-debug \(github.com\)](https://github.com/microsoft/vscode-js-debug/blob/main/CDP_SHARE.md)

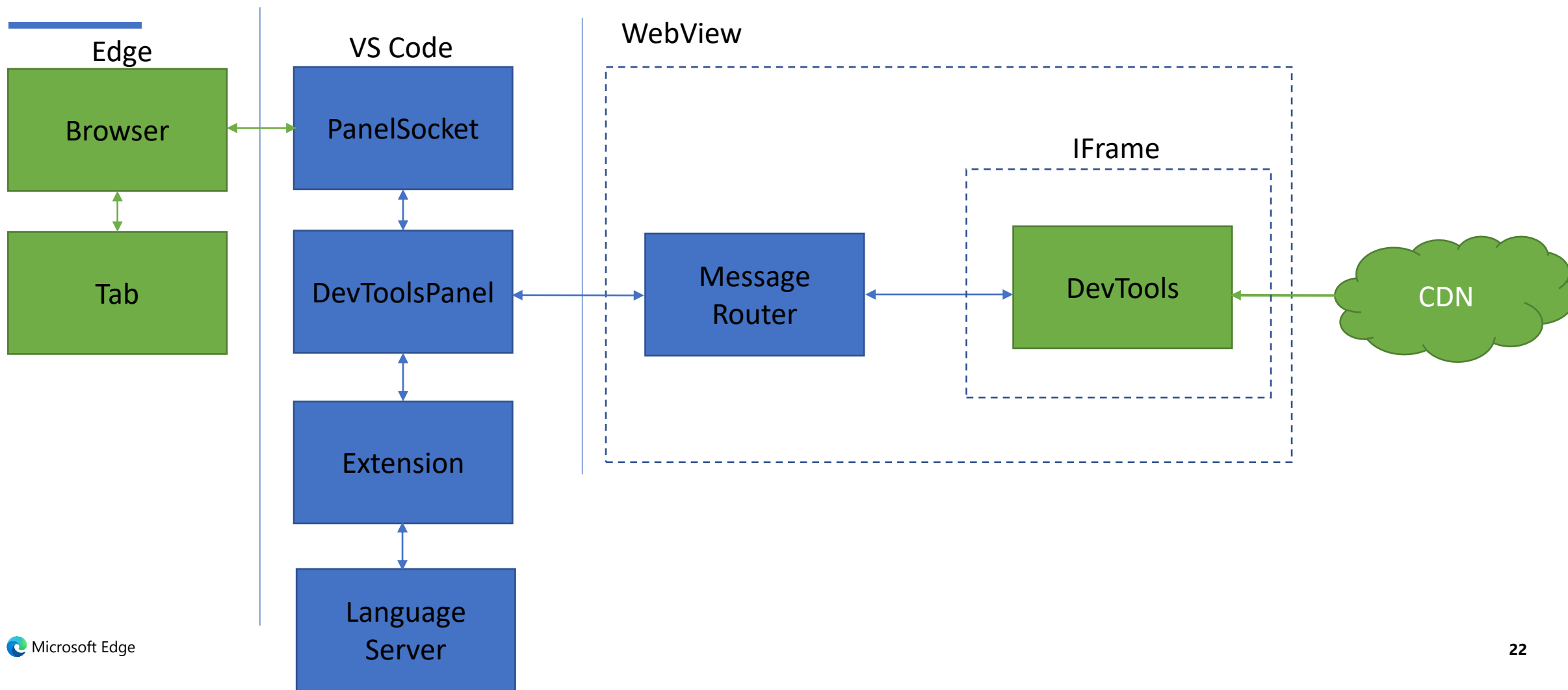
JS Debug 架构



非 JS Debug 架构



技术架构

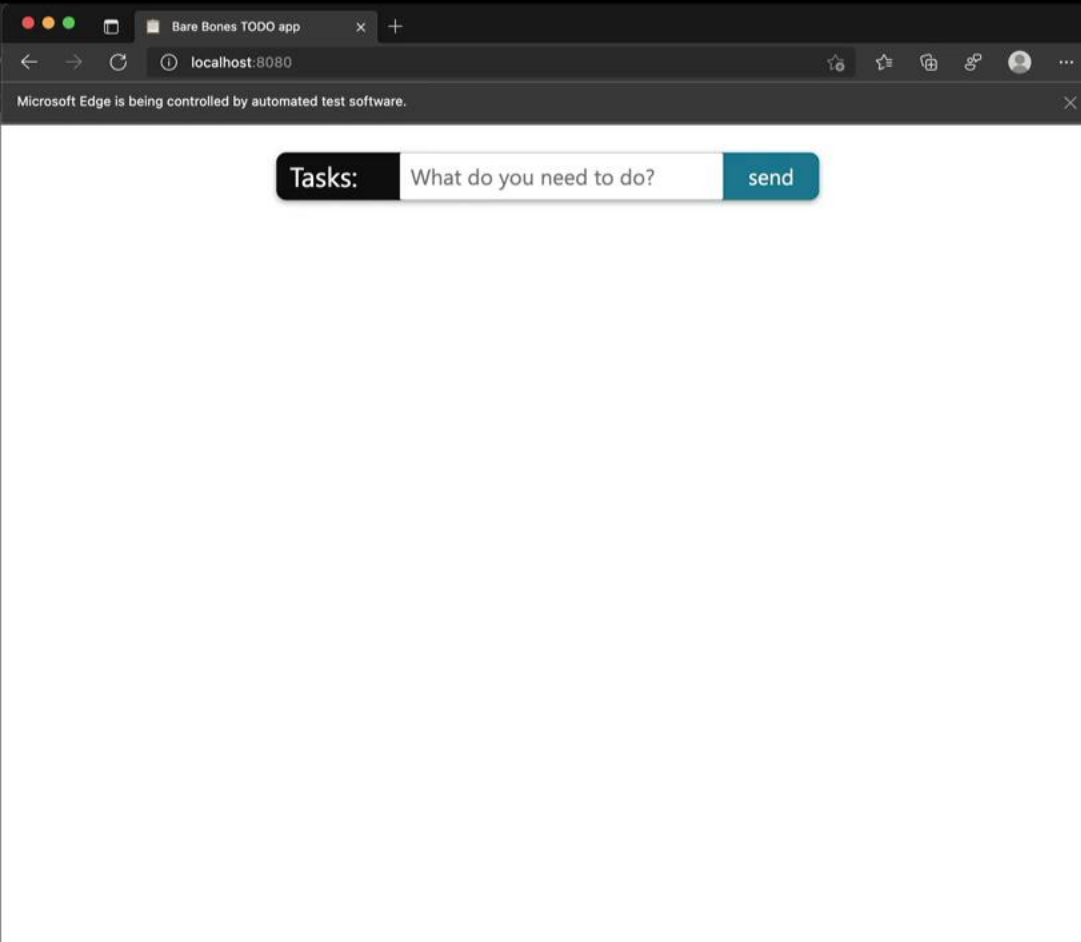


利用VSCode 编辑器功能 远程修改CSS

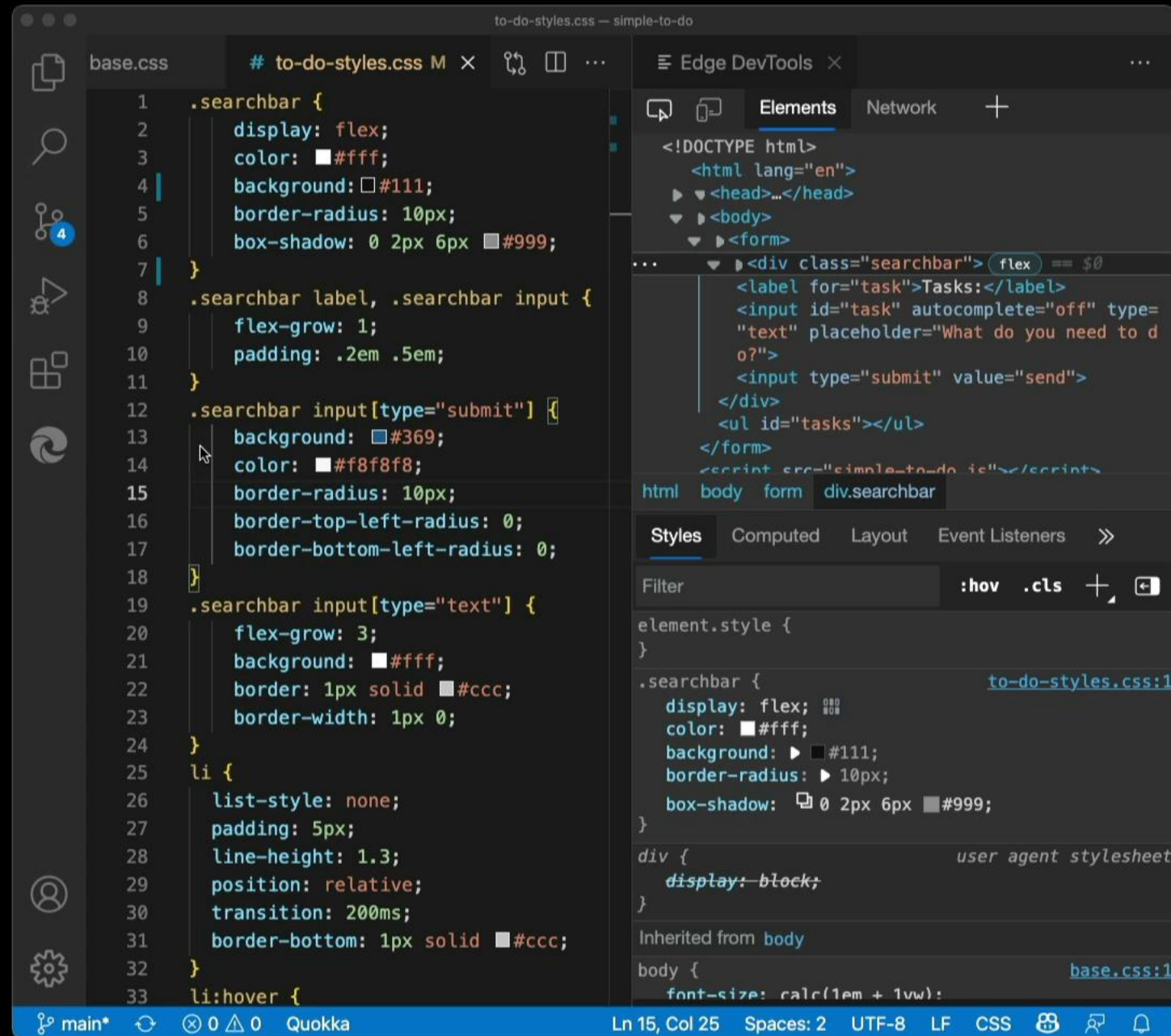


Microsoft Edge

CSS Mirror Editing aka.ms/devtools-for-code



@EdgeDevTools



同步DevTools 和 Workspace 文件修改

Opening a Workspace Files via URI

```
const doc = await vscode.workspace.openTextDocument(uri);
const viewColumn = this.panel.viewColumn === vscode.ViewColumn.One ? vscode.ViewColumn.Beside : vscode.ViewColumn.One;
const selection = line !== undefined && column !== undefined ? new vscode.Range(line, column, line, column) : undefined;
return await vscode.window.showTextDocument(doc, { preserveFocus: true, viewColumn, selection });
```

Replacing Text Editor File Contents

```
if (textEditor) {
    const fullRange = this.getDocumentFullRange(textEditor);
    const isSnapshotSameAsLastMirroredCSS = this.mirroredCSS.get(url) === textEditor.document.getText();
    if (!textEditor.document.isDirty || isSnapshotSameAsLastMirroredCSS)
    {
        this.mirroredCSS.set(url, newContent);
        void textEditor.edit(editBuilder => {
            editBuilder.replace(fullRange, newContent);
        });
    }
}
```

CSS 同步架构

VS Code

Workspaces

```
vscode.workspace.openTextDocument(uri);
```

Text Editor*

Extension

```
void textEditor.edit(editBuilder => {  
    editBuilder.replace(fullRange, newContent);  
});
```

cssMirrorContent

```
{  
    url: string,  
    newContent : string  
}
```

DevTools
+
Message Router

Language Server 和 Webhint 语法检查

2

webdebug

VARIABLES

WATCH

CALL STACK

webde... RUNNING

LOADED SCRIPTS

BREAKPOINTS

Caught Exceptions

Uncaught Except...

to-do-styles.css 2

base.css

JS simple-to-do.js

styles > # to-do-styles.css > .searchbar

1 .searchbar {

2 display: flex;

3 color: #fff;

4 background: black;

5 border-radius: 10px;

6 box-shadow: 0 5px 8px #333;

7 flex-direction: row;

8 }

9 .searchbar label, .searchbar input {

10 flex-grow: 1;

11 padding: .2em .5em;

12 -moz-animation: dummy 1s infinite;

13 }

Edge DevTools: Screencast

http://localhost:8080/ Desktop

Tasks: What do you need to do? send

test 2

Edge DevTools

Elements

Network

<!DOCTYPE html>

<html>

><head>...</head>

><body>

><form>

><div class="searchbar">flex == \$0

><label for="task">Tasks:</label>

><input id="task" autocomplete="off" type="text"

>placeholder="What do you need to do?">

><input type="submit" value="send">

></div>

>><ul id="tasks">...

></form>

><script src="simple-to-do.js"></script>

></body>

></html>

html

body

form

div.searchbar

Styles

Computed

Layout

Event Listeners

Properties

Filter

:hov .cls

element.style {

}

.searchbar {

display: flex;

color: #fff;

background: black;

border-radius: 10px;

box-shadow: 0 5px 8px #333;

flex-direction: row;

to-do-styles.css:1

main

2 1

webdebug (simple-to-do)

Webhint 语法检查 和 VSCode 诊断

Converting a Webhint Problem to a VS Code Diagnostic

```
// Translate a webhint problem into the VSCode diagnostic format.  
export const problemToDiagnostic = (problem: Problem, textDocument: TextDocument): Diagnostic => {  
  return {  
    code: problem.hintId,  
    codeDescription: { href: docHref },  
    message: `${problem.message}`,  
    range: {  
      end: { character: endColumn, line: endLine },  
      start: { character, line }  
    },  
    severity: webhintToDiagnosticSeverity(problem.severity)  
  };  
};
```

Sending diagnostics to VS Code

```
this.connection.sendDiagnostics(diagnostics);
```

从VS Code 启动 webhint Server

Defining Server and Client Options and Setting the Language Server

```
const serverOptions: ServerOptions = {
  debug: {
    args,
    module, // module = context.asAbsolutePath('node_modules/vscode-webhint/dist/src/server.js');
    options: { execArgv: ['--nolazy', '--inspect=6009'] },
    transport,
  },
  run: {
    args,
    module,
    transport,
  },
};

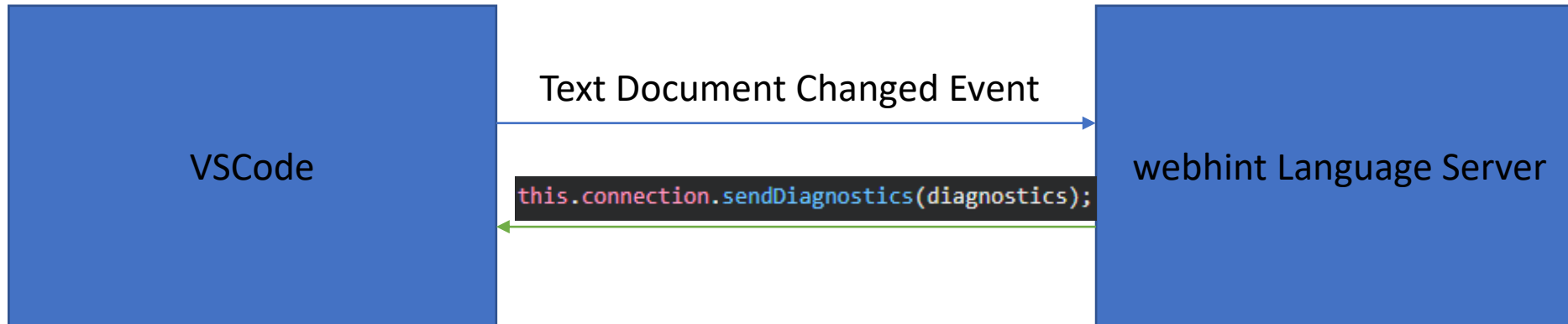
const clientOptions: LanguageClientOptions = {
  documentSelector: supportedDocuments,
  synchronize: {
    // Notify the server if a webhint-related configuration changes.
    fileEvents: vscode.workspace.createFileSystemWatcher('**/.hintrc'),
  },
};
```

Starting the Language Client

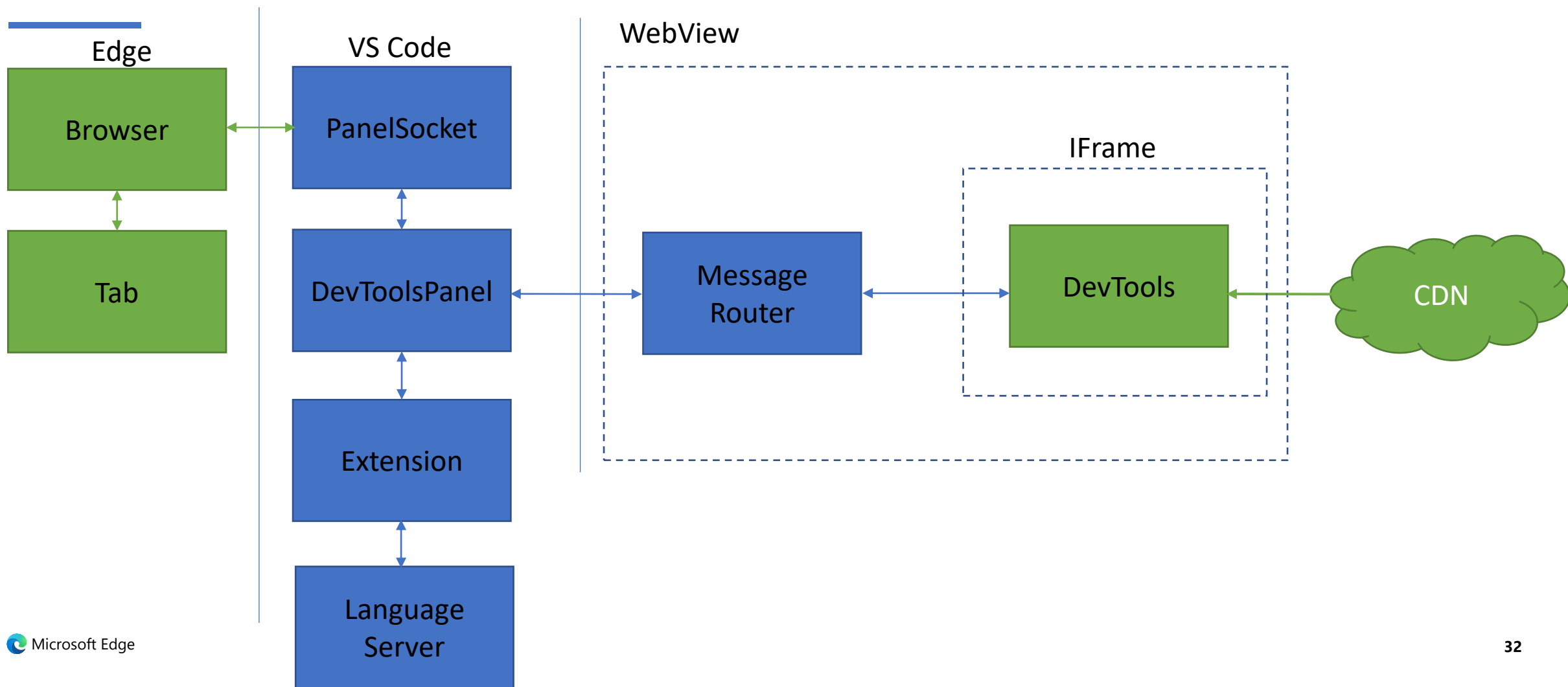
```
// Create and start the client (also starts the server).
client = new LanguageClient('Microsoft Edge Tools', serverOptions, clientOptions);

client.start();
```

Webhint Language Server 架构



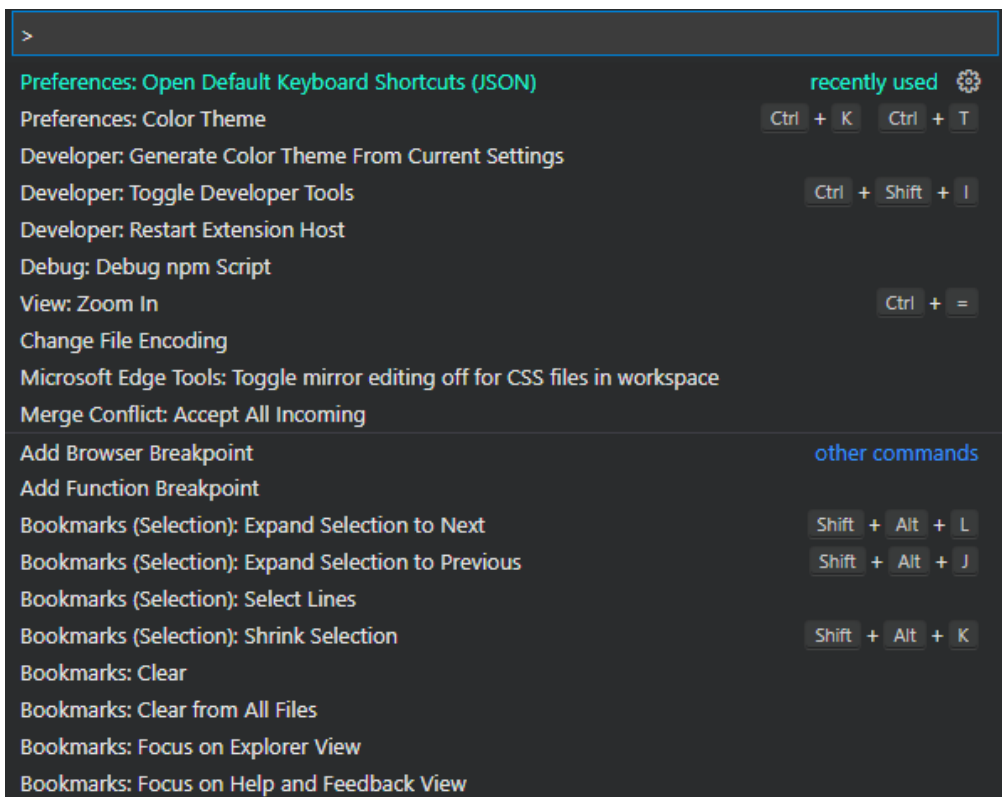
技术架构



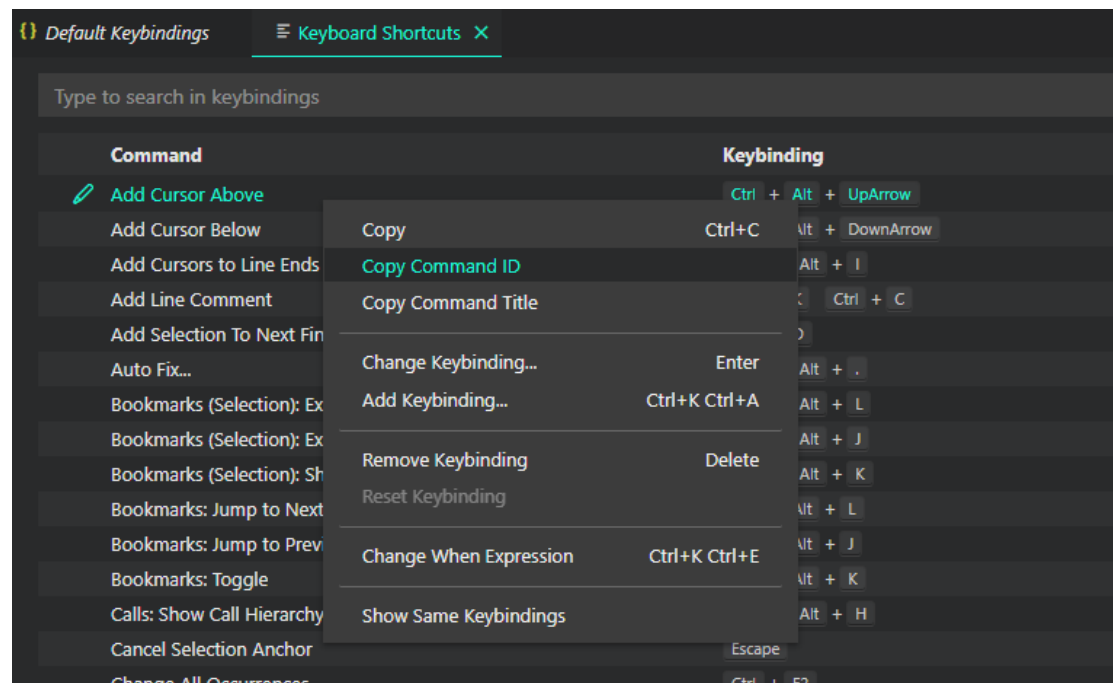
执行VS Code 指令

VScode指令

Command Menu (CTRL + SHIFT + P)



Getting Command IDs



VScode 指令代码示例

Opening launch.json using Built-in VS Code API

```
void vscode.commands.executeCommand('vscode.open', vscode.Uri.joinPath(workspaceUri, relativePath));
```

Defining Commands in our Extension

```
context.subscriptions.push(vscode.commands.registerCommand(
    `${SETTINGS_VIEW_NAME}.toggleInspect`,
    (enabled: boolean) => {
        if (ScreencastPanel.instance) {
            ScreencastPanel.instance.toggleInspect(enabled);
        }
    }
));
```

Calling Commands Defined by our Extension

```
if (cdpMsg.params.mode === 'none') {
    void vscode.commands.executeCommand(`${SETTINGS_VIEW_NAME}.toggleInspect`, { enabled: false });
} else if (cdpMsg.params.mode === 'searchForNode') {
    void vscode.commands.executeCommand(`${SETTINGS_VIEW_NAME}.toggleInspect`, { enabled: true });
}
```

请联系我们 帮助产品不断提高！

- aka.ms/devtools-for-code
- github.com/microsoft/vscode-edge-devtools

